Evaluating and Improving the Accuracy of Computational Gene-Finding on Mammalian DNA Sequences

by

Sanja Rogic

Abstract

This thesis presents work in one of the main research areas in Computational Biology: computational gene-finding in higher eukaryotic genomic DNA. Programs for identification of gene structures have been in existence for more than a decade, but today they are used more extensively than ever to analyze the enormous amount of sequence data coming from various genome sequencing projects. Consequently, their impact on research in the area of genomics and beyond is substantial.

The thesis has two distinguishable parts: the first presents an evaluation and comprehensive analysis of the current generation of gene-finding programs. For this purpose a new, thoroughly filtered and biologically validated test dataset of genomic sequences was assembled. The basic prediction accuracy of the programs tested was calculated and the relationships between various sequence and prediction features and programs' accuracy were analyzed. The second part of the thesis presents the development and results of methods for combination of the predictions from two gene-finding programs. Three methods were developed, each having some advantages over the other two, and each of them offering higher prediction accuracy on the test dataset than any gene-finding program currently available.

Contents

List of tables

List of figures

Chapter 1

Introduction

This thesis presents work in one of the main research areas in Computational Biology: the identification of exon/intron structures of genes in higher eukaryotic genomic DNA sequences. Computational gene-finding has been an active area of research for the last 15 years and has became increasingly attractive in the recent years with the development of techniques for automated DNA sequencing that allowed for large scale genomic sequencing. This resulted in a steady influx of raw genomic data that can only be efficiently analyzed by computational approaches. A large body of literature on the subject of gene prediction as well as dozens of developed gene-finding algorithms further illustrate the importance of this area of research, but despite the considerable effort gene-finding still remains an open problem.

This thesis has two goals:

1) to offer an independent comparative evaluation and analysis of the current generation of gene-finding programs;
2) based on this analysis, to explore ways to integrate predictions from two gene-finding programs in order to obtain better prediction accuracy.

The work presented here is interdisciplinary in nature, since the results of primarily biological interest are obtained by computational approaches.

1.1 Motivation

In this era of intensive genomic sequencing, when millions of bases of genomic DNA are sequenced daily in genome centres worldwide and the list of completely sequenced genomes from different organisms is growing rapidly, tools for interpreting the content of these genomes are more important than ever. The first step in deciphering the DNA sequence information is finding all the genes contained in a sequence and elucidating their structure. Although many gene-finding programs have been developed in the past 10 years and their prediction accuracy is constantly improving, we are still far away from completely automatic gene discovery with 100% accuracy. Current programs, although very good in discovering the majority of coding nucleotides (more than 90% predicted correctly) and moderately good in discovering exact exon boundaries (70-75% of exons predicted correctly) are still weak when it comes to predicting complete gene structures: less than 50% of predicted genes correspond exactly to the actual genes. Consequently, predictions given by these programs need to be verified by other evidence such as similarity to a cDNA sequence, or similarity to a known protein or expressed sequence tag (EST) sequence. However, in many cases this additional evidence is not available: it has been shown that only a fraction of newly discovered genes have identifiable homologs in the current databases (Oliver *et al.*, 1992; Wilson *et al.*, 1994; Durham *et al.*, 1999). *Ab initio* gene-finding remains the only available computational approach for identifying novel genes that do not have detectable similarities to known proteins and hence the predictions thus obtained have significant effect on our understanding of the genomes and on future experimental directions.

Recognizing the strengths and weaknesses of gene-finding programs and knowing their prediction accuracy levels are therefore essential for drawing realistic and truthful conclusions from the obtained results. This indicates a need for a comprehensive evaluation of existing gene-finding programs, which would be a valuable resource not only for users but also for developers of the programs. Two analysis of this kind have been done in the past, the last one in 1995/96 by Burset and Guigo (1996). Since then many programs have been developed or upgraded and although some of them are extensively used to identify genes in newly sequenced clones or genomes, independently computed accuracy results are not available. This motivated us to conduct an evaluation of the programs for gene prediction made available after Burset and Guigo published the results of their analysis.

Seven gene-finding programs: FGENES, GeneMark.hmm, Genie, Genscan, HMMgene, Morgan and MZEF were tested. For evaluation purposes a new, thoroughly filtered and biologically validated dataset of genomic sequences was developed that does not overlap with the training sets of the programs analyzed. For all the programs tested the basic accuracy measures introduced by Burset and Guigo (1996) were calculated. The accuracy of the programs was also examined as a function of various sequence and prediction features, such as: G+C content of the sequence, length and type of exons, signal type and score of the exon prediction. The results obtained offer an insight into the strengths and weaknesses of each individual program as well as of computational gene-finding in general.

This evaluation of programs has laid a foundation for our further research in combining the results of gene-finding programs in order to obtain higher prediction accuracy. Improving the accuracy would lead to faster, less expensive and above all more accurate interpretation of sequenced genomes, and thus any advance in this direction would be beneficial in many ways.

Three novel methods were developed for combining the predictions from Genscan and HMMgene. The methods primarily attempt to improve exon level prediction accuracy by identifying more probable exon boundaries and by eliminating false positive exon predictions. Each of the methods improved prediction accuracy on all three datasets it was tested on. The improvements were obtained at both nucleotide and exon levels, but were more substantial on the latter. Of particular practical interest are the improvements obtained on a long genomic sequence: the substantially decreased number of false positive exons resulted in the significantly increased specificity of the prediction while the sensitivity was still comparable to the sensitivity of the individual programs.

1.2 Outline of the thesis

Chapter 2 provides a brief description of biological background on the structure of genes and contents of the genomes; it also gives an overview of methods for computational gene identification. After descriptions of the novel test dataset HMR195, the programs tested and the accuracy measures applied, the chapter presents the results and discussion of the gene-finding evaluation performed. Chapter 3 gives the motivation and background for combining the predictions from gene-finding programs. It further introduces the newly developed combination methods and discusses the results of their testing on three different datasets. Chapter 4 gives an overall conclusion of the thesis and offers directions for further work. Appendix A discusses the implementation of the methods introduced in Chapter 3. Appendix B gives a list of the accession numbers and definition lines from GenBank entries for the sequences in HMR195. Two sample output files from Genscan and HMMgene are shown in Appendix C. For the reader's convenience, a glossary of biological terms used in this thesis is given in Appendix D.

Chapter 2

Evaluation of gene-finding programs

2.1 Background

2.1.1 Gene structure

The genes of most eukaryotic organisms are neither continuous nor contiguous: they are separated by long stretches of intergenic DNA and their coding sequences are interrupted by non-coding introns. Coding sequences occupy just a small fraction of a typical higher eukaryotic genome; the extreme example is the human genome where an estimate of that fraction at 3% (Duret *et al.*, 1995) was recently confirmed for chromosome 22 (Dunham *et al.*, 1999). To obtain a continuous coding sequence, which will be translated into a protein sequence, genes are transcribed into mRNA molecules that subsequently undergo complex processing to remove intronic sequences and assemble gene exons. However, assembly of the gene exons in the mature mRNA is not always the same; Mironov *et al.* (1999) have found that at least 35% of human genes are alternatively spliced - having more then one possible exon assembly. The arrangement of genes in genomes is also prone to exceptions. Although usually separated with an intergenic region, there are examples of genes nested within each other (Dunham *et al.*, 1999) - one gene located in an intron of another gene or examples of overlapping genes on the same (Schulz and Butler, 1989; Ashburner *et al.*, 1999) or opposite (Cooper *et al.*, 1998) DNA strands. The presence of pseudogenes, nonfunctional sequences resembling real genes, which are distributed in numerous copies throughout the genome, further complicates identification of true protein coding genes.

Regulatory regions play a crucial role in gene expression and their identification is needed to fully comprehend a gene's function, activity and role in cellular processes. The location of regulatory regions relative to their target gene is not uniquely determined: the basic regulatory elements, such as the TATA and CAT boxes, are usually found in the upstream proximity of the transcription start site (TSS), while the other elements, such as enhancers and silencers, can be located in distant upstream and downstream regions of a gene and sometimes even within the introns of the gene.

This brief overview of genome organization and gene architecture highlights the complexity of gene identification in the sequences of uncharacterized DNA. For further reading see Griffiths *et al.* (1996).

2.1.2 Computational methods for identification of genes

There are several methods for experimental discovery of genes, but they are time-consuming and costly. Accordingly, for the last 15 years researchers have been developing computational methods for gene-finding that could automate, or facilitate, identification of genes. Two basic approaches have been established for computational gene-finding: the sequence similarity search or *lookup* method (Fickett, 1996) and the integrated compositional and signal search or *template* method (Fickett, 1996). The latter one is also commonly referred to as *ab initio* gene finding.

Sequence similarity search is a well established computational method for gene discovery, which has been used extensively with considerable success. It is based on sequence conservation due to the functional constraints and is used to search for regions of similarity between uncharacterized sequence of interest and already characterized sequences in a sequence database. A query sequence can be compared with DNA, protein or EST sequences or it can be searched for known sequence motifs. If a query sequence is found to be significantly similar to an already annotated sequence (DNA or protein), we can - assuming that these two sequences are homologous, i.e., have common evolutionary origin -

use the information from the annotated sequence to maybe infer gene structure or function of the query sequence. Comparison with an EST database can provide information if the sequence of interest is transcribed, i.e., contains a gene coding for a protein, but will only give incomplete clues about the structure of the whole gene or its function.

Although sequence similarity search has been proven useful in many cases, it has been shown that only a fraction of newly discovered sequences have identifiable homologs in the current databases (Oliver *et al.*, 1992; Wilson *et al.*, 1994; Dunham *et al.*, 1999). Furthermore, Green *et al.* (1993) suggested that currently known proteins may already include representatives of most ancient conserved regions (or ACRs, regions of protein sequences showing highly significant similarity across phyla) and that new sequences not similar to any database sequence are unlikely to contain ACRs. The proportion of vertebrate genes with no detectable similarity in other phyla is estimated to be around 50% (Claverie, 1997). This is supported by recent analysis of human chromosome 22 (Dunham *et al.*, 1999) where only 50% of the proteins are found to be similar to previously known proteins.

These results suggest that, even today, only one half of all new vertebrate genes may be discovered by sequence similarity search across phyla. Considering that a complete vertebrate genome is still not available and that the most prominent vertebrate organisms in GenBank (Benson *et al.*, 2000), *Homo sapiens* and *Mus musculus*, have only ~18% and ~0.5% of their genomes present in finished sequences, respectively (data from April, 2000) it is obvious that sequence similarity search within vertebrates is currently limited. When more vertebrate sequences become available in GenBank (such as mouse, zebrafish, or pufferfish), matches within phyla will be more likely and this will facilitate detection of genes coding for non-ACR-containing proteins.

The second computational approach for prediction of genes structures in the genomic DNA sequences, termed the *template* approach, integrates coding statistics with signal detection into one framework. Coding statistics are measures indicative of protein coding function since they behave differently on coding and non-coding regions. A number of these measures have been evaluated in Fickett and Tung (1992) and it has been concluded that the in-phase hexamer measure, which measures the frequency of occurrence of oligonucleotides of length six in a specific reading frame, is the most effective. Indeed, this measure was used successfully in many recently developed programs such as GeneMark.hmm (Lukashin and Borodovski, 1998), Genscan (Burge, 1997) and HMMgene (Krogh, 1997). This coding statistic is usually implemented as a 5th order Hidden Markov Model (HMM) (the theory of HMMs is reviewed in Rabiner, 1989).

Signal sensors attempt to mimic closely processes occurring within the cell. They are intended to identify sequence signals, usually just several-nucleotides-long subsequences, which are recognized by cell machinery and are initiators of certain processes. The signals that are usually modeled by gene-finding programs are: promoter elements, start and stop codons, splice sites, and polyA sites. Many different pattern recognition methods have been used as signal detectors, including simple consensus sequences, weight matrices, weight arrays, neural network, and decision trees .

DNA sequence signals have low information content; they are usually degenerate and highly unspecific since it is almost impossible to distinguish the signals truly processed by the cell from those that are apparently non-functional. Therefore, signal sensors are not sufficient to elucidate gene structure and it is necessary to combine them with coding statistics methods in order to obtain satisfactory predictive power.

Both codon statistics and signal models are 'learned' from a training set: frequencies of oligonucleotide occurrence in different regions of the genes are calculated from sequences in the training set and the signal models are constructed using the multiple alignment of the signal sequences from the training set.

There is also a group of programs that integrate a third component in their systems: similarity with an annotated sequence. Examples of such programs are GeneID+ (Guigo *et al.*, 1992), GeneParser3 (Snyder and Stormo, 1995), Procrustes (Gelfand, 1996) and AAT (Huang *et al.*, 1997).

Existing gene-finding programs have been designed to identify gene structure simpler than the intricate structure described above; most of the programs, especially the older ones, are trained to identify just one gene in a sequence, rarely predicting any promoter elements. Some progress has been made with recently developed programs, which are capable of identifying more complex genomic structure: any number of genes with either complete or partial structure. This is the case with Genie (Kulp *et al.*, 1996), GeneMark.hmm, Genscan, and HMMgene. Still, regulatory regions and polyA sites usually remain unidentified, 5' and 3' untranslated regions are not specified, alternative splice variants are not considered, and overlapping or nested genes are not detected.

Nevertheless, the prediction of the coding sequence of typical genes is an important first step in deciphering the content of any genome and gene-finding programs are used extensively for this task with considerable success.

2.1.3 Evaluation of gene-finders

Since in many cases there is no additional evidence to support the gene predictions provided by *ab initio* gene-finding programs, it is very important to know the accuracy of these programs. The reliability of the programs concerns both users and developers. Lab bench experiments are often based on the gene/exon predictions and they usually require a substantial investment in time and resources. This is why it is important for a user to know how well a certain algorithm performs, what its strengths and weaknesses are, and how to interpret a particular score given by the program. For developers it is valuable to know the current state of the art, to relate the programs' efficiency and reliability to the methods used and to recognize the weaknesses that need to be addressed.

Previous comparative analyses of gene-finding programs have been performed by Snyder and Stormo in 1995 and Burset and Guigo in 1996. Snyder and Stormo (1995) analyzed three gene-finding programs GeneID, GRAIL (Xu *et al.*, 1994) (two versions), and GeneParser (three versions of the program) on rather limited test sets containing 28 and 34 sequences. More comprehensive evaluation of gene structure prediction programs was done by Burset and Guigo (1996). These authors tested 9 programs on a test set of 570 sequences and introduced a number of performance metrics to measure accuracy of prediction on three levels: nucleotide, exon, and gene level. Some of these measures were known and used before (sensitivity, specificity, and correlation coefficient at the nucleotide level) and some were newly introduced (approximate correlation, sensitivity, and specificity at the exon level). The authors also investigated the behaviour of the programs on sequences with errors (frameshift mutations), sequences with differing G+C content and sequences from different phylogenic groups within the vertebrates. This comprehensive analysis has been a valuable resource for both users and developers of gene prediction programs, and the Burset/Guigo dataset has been extensively used as a benchmark dataset for testing new generations of programs.

In the last four years, since the Burset/Guigo analysis was published, many new programs have been developed. For most of them the accuracy measures have been reported for the Burset/Guigo dataset. The reason for concern is not just that authors tested their programs themselves, but also in many cases it is not clear how the sequences from a program's training set overlap with the Burset/Guigo test set. It is realistic to assume that, in many cases, the training sets of these programs do overlap with the Burset/Guigo dataset since it used to contain the vast majority of available vertebrate genomic sequences.

This lack of independent performance results of gene-finding programs motivated the development of an evaluation similar to one done by Burset and Guigo. To evaluate gene-finding programs meaningfully it is necessary to do it uniformly on one test set of sequences. It is also important to avoid using sequences used for the training of programs analyzed or otherwise the accuracy of the programs may be overestimated. The next section describes the assembly and the characteristics of the new dataset of genomic sequences that was used as a test set in the analysis.

2.2 The novel test dataset HMR195

The primary requirement for the construction of the dataset to be used for the evaluation of gene structure prediction programs was to exclude sequences already used for training those programs. Since for some of the programs the training datasets are not specified, the best solution was to choose only sequences entered into GenBank after the programs were developed and trained. For that reason only sequences submitted to GenBank after August, 1997 were considered.

Although we first considered including only human sequences in the dataset, after a few filtering steps it was obvious that the size of the dataset would be relatively small, so we decided to expand the list of organisms. Sequences from *Mus musculus* and *Rattus norvegicus* were included because the mouse and rat genomes are relatively well studied and a number of murine sequences are present in GenBank. Also human, mouse and rat genomes are phylogenetically close enough to be analyzed with the same parameter files used in the gene-finding programs (parameter values specific to mouse or rat sequences are not available for any program). 2.6.5 on phylogenetic specificity offers further discussion and justification of this hypothesis.

With these considerations the dataset was constructed as follows:

DNA sequences were extracted from GenBank release 111.0 (April 1999). The basic requirements in sequence selection were:

- the sequence was entered in GenBank after August, 1997;
- the source organism is *Homo sa*piens, *Mus musculus* or *Rattus norvegicus* ;
- only genomic sequences that contain exactly one gene were considered;
- mRNA sequences and sequences containing pseudo genes or alternatively spliced genes were excluded.

Sequences collected according to those principles were further filtered to meet the following requirements:

- all annotated coding sequences started with the ATG initiation codon and ended with one of the stop codons: TAA, TAG, TGA;
- all exons had dinucleotide AG at their acceptor site and dinucleotide GT at their donor site;
- sequences that did not contain any nucleotides in their 5' or 3' UTR were discarded;
- sequences longer than 200,000 bp were discarded because some of the programs analyzed can only accept sequences up to that length;
- sequences whose coding region contains in-frame stop codons were discarded.

Sequences that passed these filtering steps were further subjected to non-redundancy testing. All-against-all neighbour search with the Entrez Browser (command line Entrez - Nentrcmd from the NCBI tool kit) (Schuler *et al.*, 1996) was performed and if two sequences were linked as neighbors only one of them was selected to enter the final dataset. Neighbour linkage in Entrez represents high similarity between two sequences.

The final restriction of the dataset was done to confirm exon locations annotated in the GenBank records. For each sequence in the dataset we used the BLAST algorithm (Altschul *et al.*, 1990; Altschul *et al.*, 1997) to find a corresponding mRNA sequence that had been independently sequenced and not derived from the genomic sequence. If such an mRNA sequence existed, the sim4 program (Florea *et al.*, 1998) was used to align the genomic sequence and the mRNA sequence. The result of the sim4 alignment is the list of exon locations, which were then compared with the annotation in the GenBank record. Only those sequences whose exon annotation perfectly matched the sim4 results were selected for the final version of the dataset. Unfortunately, this analysis could not confirm the start location of the initial exon and the end location of the terminal exon, since mRNAs also contain 5' and 3' untranslated regions (UTR) that also align to the genomic sequence, so these annotations remained unconfirmed.

The resulting dataset contains 195 sequences with exactly one complete, either single-exon or multi-exon, gene and it is named HMR195.

HMR195 has the following characteristics:

- the ratio of human:mouse:rat sequences is 103:82:10 ;
- the mean length of the sequences in the set is 7,096 bp;
- the number of single-exon genes is 43 and the number of multi-exon genes is 152;
- the average number of exons per gene is 4.86;
- the mean exon length is 208 bp, the mean intron length is 678 bp, and the mean coding length of a gene is 1,015 bp (~340 amino acids);
- the proportion of coding sequence in this dataset is 14%, of the intronic sequence 46%, and of the intergenic DNA 40%.

The HMR195 dataset is available at *http://www.cs.ubc.ca/labs/beta/genefinding/*.

This dataset is not a typical subset of sequences from human and murine genomes: the fraction of coding sequence in the dataset (14%) is much higher than the estimated 3% for these genomes. The mean coding length of ~340 amino acids is shorter than the calculated mean of ~450 aa for *C. elegans* and *S. cerevisiae* (Zhang, 2000). It is realistic to expect that the average protein in human, mouse and rat will be at least this long, since the analysis in Zhang (2000) shows that protein length seems to increase with the complexity of organism. Also, the average number of exons in a gene in HMR195, 4.86, is lower than the calculated 5.4 for human chromosome 22 (Dunham *et al.*, 1999). These discrepancies are a direct product of biases in GenBank and other public sequence databases, which are discussed below in Section 2.7. With the current limited amount of data it is not yet feasible to generate a dataset which would perfectly model human and murine genomes. Also, the proportion of single-exon genes in HMR195 substantially exceeds this proportion in real genomes. Again, overrepresentation of these sequences in GenBank is a source of this discrepancy. Since there were no other biological reasons to eliminate single-exon genes we chose to keep them within the dataset.

2.3 Programs tested

All gene-finding programs made available after the evaluation by Burset and Guigo in 1996 were considered for this analysis. Since the goal of this experiment was to asses the programs that solely use statistics and pattern recognition methods for gene-finding, programs that use other resources, such as database similarity search, were not included in the testing. Also, only programs trained on vertebrate sequences were considered.

The seven programs tested were (in alphabetical order) FGENES, GeneMark.hmm, Genie, Genscan, HMMgene, Morgan, and MZEF.

Some of the programs analyzed allow the user to change some of the parameters of the program (e.g. prior probability for MZEF and exon size in Morgan), depending on the properties of the input sequences. Although this might be beneficial for expert users working on specific sequences it is not suitable for automatic testing of large sequence dataset. Therefore, all the programs analyzed here were run with the suggested default parameters.

All programs were installed and run locally except for Genie, which was accessed through the Genie web server *http://www.fruitfly.org/seq_tools/genie.html*. The programs were run on a SUN Ultra 60 computer, under the Solaris 5.6 operating system.

The programs analyzed in this survey are enumerated below. For each program a short description of methods used by the program is given, information about its training set, the parameter files used when running it, the subset of the HMR195 dataset it was tested on, and some characteristics of its output format.

1. FGENES (Solovyev and Salamov, 1997) [version 1.6]. Information about this program can be found on the Sanger Center Computational Genomic Group web site *http://genomic.sanger.ac.uk/gf/gf.html* and details about an earlier version of the program FGENEH can be found in Solovyev *et al.* (1995). FGENES uses dynamic programming to find the optimal combination of exons, promoters and polyA sites detected by a pattern recognition algorithm, constructing a set of gene models along a given sequence. The model is very flexible and allows prediction of single- and multi-genes in a sequence, that are either complete or partial. The program has been trained on a non-redundant dataset of 660 human sequences extracted from GenBank release 100. Details about the dataset can be found in (Salamov and Solovyev, 1997). The type (first, internal, last, single) and location of each exon is specified in the output of the program, and for each exon there is an associated score for the prediction.

All the sequences from HMR195 were submitted to FGENES, which predicted genes in 190 out of 195 sequences.

2. GeneMark.hmm (Borodovsky *et al.*, 1998) [version 2.2]. Initially, this program was developed for bacterial gene-finding (Lukashin and Borodovsky, 1998) and it has been only recently modified to predict gene structure in eukaryotic organisms. A paper about the eukaryotic version of the program has not been published, but from the program's web site *at http://genemark.biology.gatech.edu/GeneMark/* it can be concluded that it uses an explicit state duration HMM, which is often used in gene-finding programs (Genie, Genscan). The optimal gene candidates selected by the HMM and dynamic programming are further processed by a ribosomal binding site recognition algorithm. The dataset used for training is not described. The output is similar to that of FGENES, but no scores are given.

In this analysis GeneMark.hmm was run with the `human.mtx` matrix for every sequence in HMR195, and it predicted genes in every sequence.

3. Genie (Kulp *et al.*, 1996) [version 1.x and version 2.1, from October 1999]. Similarly to GeneMark.hmm, Genie uses a generalized HMM with arbitrary length distributions associated with some states of the model. The system is described as modular, since each state is trained separately and new states can be easily added. The mechanisms underlying some states are neural networks for splicing sites, with Markov chains for coding regions. The training set is assembled from the human sequences extracted from GenBank release 89.0 (1995) and details describing sequences and filtering processes can be found at *http://www.fruitfly.org/sequence/human-datasets.html*. This dataset has also been used for training other gene-finding systems (HMMgene, Genscan). Genie can predict single- or multiple-exon genes and any number of them in the sequence. The Genie web site is at *http://www.fruitfly.org/seq_tools/genie.html*.

During the testing period a new version of Genie, 2.1, became available, so we used the opportunity to test both versions. In this thesis only the results of the new Genie's prediction will be considered and the name Genie will refer to version 2.1 of this program. The results for the 1.x version of Genie exist but are not presented here.

In order to test the new, upgraded version of Genie we sent all our sequences to Martin Reese at Lawrence Berkeley National Laboratory who ran them through the program. Genie's output is in GFF (General Feature Format) format with the location and score for each feature in the sequence. Genie predicted genes in 180 out of 195 sequences.

4. Genscan (Burge, 1997; Burge and Karlin, 1997) [version 1.0]. In this program the structure of the genomic sequence is modeled by an explicit state duration HMM. The states of this HMM are probabilistic models themselves. Signals are modeled by weight matrices, weight arrays and *maximal dependence decomposition* (Burge, 1997), a new technique used for recognition of donor sites. Genscan's model can predict the absence of genes or the presence of a single gene or multiple genes, which can be either complete or partial. It also has the option to predict suboptimal exons, which are defined as potential exons with a probability higher than a certain threshold but which are not contained in the optimal parse of the sequence. This type of exon can potentially represent alternatively spliced exons. Genscan was trained on Kulp and Reese's dataset of human genomic sequences and an additional set of 1999 human cDNA sequences was used for training the coding region HMM. The maximal length of the input sequence for this version of Genscan is 200 kb. The output of Genscan is similar to the output of the other programs, giving information about exon locations and their probabilistic scores, but scores for other sequence features such as splicing sites are also given. The web version of Genscan is *at http://CCR-081.mit.edu/GENSCAN.html*.

Genscan was run with parameter file `HumanIso.smat` for all the sequences in HMR195. It predicted genes in 192 out of 195 sequences.

5. HMMgene (Krogh, 1997) [version 1.1d]. The program is based on HMMs and it is trained using a criterion called *conditional maximum likelihood*, which maximizes the probability of correct prediction. If the sequence analyzed already has some subregions identified (hits to EST or protein database, repeated elements), those regions can be locked as coding or non-coding and then submitted to HMMgene. The underlying gene structure model can predict both partial and complete genes in sequence and any number of them. The program has the option to give more than one prediction, which could indicate alternative splicing of the gene in the sequence. The dataset of human single- and multi-exon genes collected by Kulp and Reese was used for the training of this program. The output is given in GFF format, slightly different from that used by Genie: it does not give the location of the splicing sites, but only of the exons, whose type is also specified. HMMgene's web site is at *http://www.cbs.dtu.dk/services/HMMgene/*.

Every sequence from the testing dataset HMR195 was submitted to the program, which predicted genes in 190 out of 195 sequences.

6. Morgan (Salzberg *et al.*, 1998) [version from June 1997]. The underlying method behind Morgan is a combination of decision trees, dynamic programming and Markov chains. The most distinctive technique used is a decision tree classifier that classifies subsequences into different classes: initial, internal, and final exons. Morgan has been trained on the Burset and Guigo dataset of 570 sequences containing only multi-exon genes and for that reason its prediction is limited to only this class of genes. Also it is not capable of analyzing sequences that contain symbols other then A, C, G, T (e.g., N, M, R, Y) which further reduces the number of sequences from HMR195 that can be used for the analysis. Morgan has the standard output with exon locations and probability scores. The recommended length of DNA sequence is up to 200 kb.

Morgan was tested on 127 acceptable sequences from HMR195 and it predicted a gene in every sequence analyzed.

7. MZEF (Zhang, 1997) [version from April 1998]. It uses a quadratic discriminant function to distinguish between two classes: coding and non-coding. Its training set consists of 3440 human exons extracted from GenBank release 87.0 and it's trained to predict only internal coding exons. The output of the program gives the location of every internal exon predicted, along with a probability score for it and some other measures for different reading frames. MZEF can only analyze sequences shorter than 200 kb. The program has an option to set the prior probability for the sequence analyzed which depends on gene density and G+C content of the sequence. The default value of 0.4 was used for the prior when submitting the sequences from the HMR195 dataset to MZEF. The program's web site is at *http://sciclio.cshl.org/genefinder/*.

Since MZEF can predict only internal exons, only sequences that contain more than two exons from the dataset HMR195 were considered. The accuracy measures were calculated considering only annotated internal exons. There were 119 of the HMR195 sequences and it predicted exons in 111 of them.

2.4 Measuring predictive accuracy

In order to evaluate the predictive accuracy of a gene-finding program we need to compare the exons predicted by the program with the actual coding exons, as annotated in the GenBank record under the "CDS" feature (annotated non-coding exons are not considered, since the programs analyzed do not predict them). From this comparison, nucleotide and exon level accuracy measures were calculated.

2.4.1 Nucleotide level accuracy

If we define the values *TP* (true positives), *TN* (true negatives), *FP* (false positives) and *FN* (false negatives) as follows:

TP - the number of coding nucleotides predicted correctly as coding
TN - the number of non-coding nucleotides predicted correctly as non-coding
FP - the number of non-coding nucleotides predicted incorrectly as coding
FN - the number of coding nucleotides predicted incorrectly as non-coding

then we define sensitivity as the proportion of coding nucleotides that are correctly predicted as coding:

$$Sn = \frac{TP}{TP + FN}$$

and specificity as the proportion of nucleotides predicted as coding that are actually coding:

$$Sp = \frac{TP}{TP + FP}$$

These are widely used measurements of accuracy for gene prediction programs.

Both *Sn* and *Sp* range independently over [0,1], with perfect prediction occurring only when both measures are equal to 1. Each of these measures is not sufficient by itself because perfect sensitivity can be obtained if all the nucleotides were predicted as coding and perfect specificity if all nucleotides were predicted as non-coding.

A single measure that captures both specificity and sensitivity called the correlation coefficient (*CC*) is defined as:

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

This measure has been extensively used for evaluating gene structure prediction programs, but it has the undesirable property that it is not defined for some sequences (e.g. if there is not any coding region in an input sequence or an input sequence has been predicted to be entirely non-coding). A measure with similar characteristics, but defined under any circumstance, is the *approximate correlation* (*AC*), introduced in Burset and Guigo (1996), defined as:

$$AC = (ACP - 0.5) * 2$$

where *ACP* is the average conditional probability defined as:

$$ACP = \frac{1}{4}\left(\frac{TP}{TP + FN} + \frac{TP}{TP + FN} + \frac{TN}{TN + FN} + \frac{TN}{TN + FP}\right)$$

Since at least two of the conditional probabilities in this formula are always defined, *ACP* can always be calculated as the average of the ones defined. *CC* and *AC* range over [-1,1] and usually are close to each other whenever *CC* is defined.

Nucleotide level accuracy measures indicate how good the 'search by content' element of the program is, but they don't tell us much about the 'search by signal' component. For measuring those prediction characteristics we use exon level prediction accuracy.

2.4.2 Exon level accuracy

Exon level prediction is also estimated by sensitivity and specificity, but in this case true positives are exactly predicted exons (identical to an annotated exon). The formulas for exon level sensitivity (*ESn*) and specificity (*ESp*) are:

$$ESn = \frac{TE}{AE} \qquad ESp = \frac{TE}{PE}$$

where *TE* (true exons) is the number of exactly predicted exons and *AE* and *PE* are the numbers of annotated and predicted exons, respectively.

Similarly to nucleotide level accuracy these measures cannot be used alone and usually their average is used as a reliable measure of program's exon level accuracy.

Sometimes, knowing just the proportion of the exactly predicted exons may underestimate the performance of the program, especially if its 'search by signal' component is weaker. To get a better estimate of the prediction accuracy of the analyzed programs we can also considered other categories of predicted and annotated exons. Predicted exons can be divided into four categories: exactly predicted, partially predicted (only one exon boundary is correctly predicted), overlapped (neither exon boundary is correct, but it overlaps an actual exon) and wrong (does not overlap any actual exon). Analogously, annotated exons can be divided in those that are exactly predicted, partially predicted, overlapped and missed (not overlapped with any predicted exon).

The tables show values for:

CRa - proportion of annotated exons that are correctly predicted

CRp - proportion of predicted exons that are exactly correct

PCa - proportion of partially predicted annotated exons

PCp - proportion of predicted exons that are partially correct

OL - proportion of predicted exons that overlap actual exons

ME - proportion of missed exons

WE - proportion of wrong exons.

From the definition of the exon level sensitivity we can see that this measure is not defined when a program does not predict any exons in a sequence. In this case, 0 is assigned to both *ESn* and *ESp*, and this sequence will not be considered when calculating the average values for the whole dataset. Even though *ESn* is defined for every sequence containing actual exons we do not average it over the sequences for which *ESp* is not defined in order to obtain a more realistic relationship between two measures (in real genomic sequences the sequences without exon prediction would be less common than the sequences without actual exons). On the sequence level *CRa* and *CRp* are identical to *ESn* and *ESp*, except for *PE*=0 when *CRp*=0, but they are averaged over all sequences in the dataset. For this reasons we used *CRa* and *CRp* as more credible measures when programs were run on the subsets of the dataset and when sequences without predictions could strongly influence the results for *ESn* and *ESp*.

2.5 Implementation and results of the evaluation

Sequences from the HMR195 dataset were run through the seven gene prediction programs. For each sequence, the exons predicted on the forward strand (predictions for the reverse strand were ignored) were compared to the actual coding exons, as annotated in the GenBank 'CDS' feature. Although all of the programs tested, except Morgan, can predict genes and exons on both DNA strands simultaneously, the GenBank records for most of the sequences in HMR195 contain only annotation for the Watson/plus strand and consequently only prediction for that strand could be confirmed. From this comparison accuracy measures at the nucleotide and the exon level were computed and then averaged.

For each gene-finding program tested a Perl script was written that parses the prediction from the program and compares predicted exons to the annotated exons read from the annotation file for HMR195. The separate scripts were necessary since the outputs of the programs tested are different for each program. For each sequence in HMR195 all the accuracy measures are calculated and written to the output file. The final output from each

Perl script are the averaged nucleotide and exon level prediction accuracy measures on the HMR195 dataset. The scripts for the computation of the accuracy measures for Genscan and HMMgene are posted on the web site at *http://www.cs.ubc.ca/labs/beta/genefinding/*.

The gene level accuracy measures were not computed since the prediction of the entire gene structure is still unreliable and seldom used. As an illustration, Dunham *et al.* (1999) identified 94% at least partially predicted exons on human chromosome 22 using Genscan, but only 20% of genes had all exons predicted exactly.

We chose to use averaging by sequence, where measures are first calculated for each gene then averaged over all genes, as opposed to averaging by base, where measures are summed for all sequences and then averaged by nucleotide or by exon, depending on the measure type. The former is thought to give better indication of the success rate for the individual sequence entry. For discussion see Dong and Searls (1995) and Burset and Guigo (1996).

The measures are averaged only over sequences for which they are defined. This might overestimate the values for *Sn*, *Sp*, *AC*, *CC*, *ESn*, *ESp* and the average of the latter two (*CR*, *PC*, *OL*, *WE* and *ME* are always defined) on the sets that have many sequences without prediction. So, in order to have a realistic estimate of the gene-finders' performance one must also look at the number of sequences where no genes were predicted. The accuracy measures for all of the programs analyzed, averaged for the entire HMR195 dataset are presented in Table 1.

The next step was to examine accuracy as a function of various sequence or prediction features, such as: G+C content of the sequence, length and type of the annotated and the predicted exons, signal type for both annotated and predicted genes, and the score/probability of the exon prediction. For each of these characteristics the dataset was divided into the subsets exhibiting different value ranges or types of the characteristic examined and the accuracy measures were calculated and averaged over all the sequences belonging to a particular subset. For each of the programs and each of the sequence or prediction features a separate Perl script was written. The results obtained are presented in Tables 2-6.

2.6 Discussion

Comparing the results presented in Burset and Guigo (1996) with the results obtained in this study (Table 1) it is apparent that the new generation of programs has, overall, substantially higher prediction accuracy then the programs analyzed by Burset and Guigo in 1996. At that time the program with the best approximate correlation (among the programs not using any database similarity search) was FGENEH, with $AC = 0.78$, while the highest AC in 1999 is 0.91, exhibited by both Genscan and HMMgene. On the exon level, $(ESn+ESp)/2$ has increased from 0.64 for FGENEH to 0.76 for HMMgene. Also, earlier gene-finders were programmed to have low false positive rates at the expense of losing valid predictions, which resulted in, on average, 20% higher specificity than sensitivity. Programs of the new generation are tuned to have equally high sensitivity and specificity, which is more desirable.

These improvements have come about as a result of developing more accurate models for gene structure that are capable of recognizing many different gene features in the sequence. Most of the gene-finders analyzed use explicit duration HMMs with associated length distributions for each state. These models of genomic structure are hierarchical, with generalized HMMs modeling the overall gene structure, where states of the model are independent probabilistic models themselves, such as HMMs and neural networks. Also, new methods have been developed for signal recognition, such as maximal dependence decomposition used for donor site recognition in Genscan and neural networks in Genie. The training sets are carefully selected and the average number of training sequences in the dataset has increased, allowing for more diversity in genomic content of the training sequences.

With the accuracy measures at the nucleotide level as high as 0.91 for Genscan's and HMMgene's AC we might conclude that the problem of computational gene-finding is almost solved. But looking at the results for exon sensitivity and specificity and their average, we see that the goal is still far away. Why is there such a gap between AC and $(ESn+ESp)/2$? Since ESn and ESp are defined as TE divided by AE and PE, respectively, this means that in calculating these two measures only exons with both boundaries predicted

Programs	**# of sequences**	***Nucleotide accuracy***				***Exon accuracy***							
		Sn	***Sp***	***AC***	***CC***	***ESn***	***ESp***	***(ESn+Esp)/2***	***ME***	***WE***	***PCa***	***PCp***	***OL***
FGENES	195 (5)	0.86	0.88	0.84 ± 0.19	0.83	0.67	0.67	0.67 ± 0.32	0.12	0.09	0.20	0.17	0.02
GeneMark.hmm	195 (0)	0.87	0.89	0.84 ± 0.18	0.83	0.53	0.54	0.54 ± 0.36	0.13	0.11	0.29	0.27	0.09
Genie	195 (15)	0.91	0.90	0.89 ± 0.16	0.88	0.71	0.70	0.71 ± 0.30	0.19	0.11	0.15	0.15	0.02
Genscan	195 (3)	0.95	0.90	0.91 ± 0.12	0.91	0.70	0.70	0.70 ± 0.32	0.08	0.09	0.21	0.19	0.02
HMMgene	195 (5)	0.93	0.93	0.91 ± 0.13	0.91	0.76	0.77	0.76 ± 0.30	0.12	0.07	0.14	0.14	0.02
Morgan	127 (0)	0.75	0.74	0.70 ± 0.21	0.69	0.46	0.41	0.43 ± 0.26	0.20	0.28	0.28	0.25	0.07
MZEF	119 (8)	0.70	0.73	0.68 ± 0.21	0.66	0.58	0.59	0.59 ± 0.28	0.32	0.23	0.08	0.16	0.01

Table 1: Nucleotide and exon level accuracy – For each sequence in the HMR195 dataset, the exons predicted on the forward (+) strand were compared to the annotated exons. The standard measures of predictive accuracy on nucleotide and exon level were calculated for each sequence and averaged over all sequences for which they were defined. This was done separately for each of the programs tested. # of sequences – number of sequences effectively analyzed by each program; in parentheses is the number of sequences where the absence of gene was predicted; *Sn* –nucleotide level sensitivity; *Sp* – nucleotide level specificity, *AC* – approximate correlation; *CC* – correlation coefficient; *ESn* – exon level sensitivity; *ESp* - exon level specificity; *ME* – missed exons; *WE* – wrong exons; *PCa* – proportion of real exons that were partially predicted (only one exon boundary correct); *PCp* – proportion of predicted exons that were only partially correct; *OL* – proportion of predicted exons that overlap an actual exon. *AC* and *(ESn+ESp)/2* are given with standard deviation.

correctly will be considered. An almost perfectly predicted exon, covering the whole sequence of an actual exon, but exceeding the splicing site by just few nucleotides will not be counted in TE. In order to predict the exact boundaries of an exon a program has to have a strong "search by signal" component - signal sensors for identifying start and stop codons and splicing sites. However, signal detection, especially of start and stop codon, is probably the weakest component of current gene-finding programs, as can be observed in Table 5. Although discrimination among coding and non-coding regions, most often done by measuring the hexamer frequencies in these regions, has shown to be quite successful, signal recognition could still be improved. There has been significant effort to improve prediction of acceptor and donor splice sites and many different methods have been used for this task, such as neural networks and maximal dependence decomposition (the methods for splice site detection are not known to us for all the programs analyzed). The success of these methods is apparent in Table 5. On the other hand we are not aware of any systematic effort to tackle the problem of start and stop codon detection. These signals are considered to have low information content and they are usually detected by using weight (positional) matrices, weight arrays that capture dependencies between adjacent nucleotides or, in the case of Genie, neural networks for translation initiation site.

The tendency to miss actual signals can also be observed from the proportion of partially predicted exons (*PCa*) that ranges from 0.08 for Genie to 0.29 for GeneMark.hmm (Table 1). GeneMark.hmm and Morgan besides having high proportion of *PCa* also have a relatively high proportion of *OL* (0.09 and 0.07, respectively, in Table 1) and according to these results they are the programs with the poorest signal detection. If we add the number of the partially predicted exons to the number of correctly predicted exons and use this number for calculating *ESn* and *ESp* then *AC* and *(ESn+ESp)/2* would have similar values.

2.6.1 G+C content

The human genome (and the genomes of other warm-blooded vertebrates) is not a structurally homogenous sequence of nucleotides. Instead, it's a mosaic of isochores, long (>300 kb, on average) DNA regions whose base composition is locally homogenous, but varies significantly between disjoint regions. The genome is usually divided into 5 different compositional categories: L1, L2 (A+T rich regions), H1, H2 and H3 (G+C rich regions) in increasing order of G+C%. It has been observed (Bernardi, 1993) that L1+L2 constitute approximately 60% of human genome, H1+H2 30% and H3 only 5%. These compositional regions widely vary in gene density: Zoubak *et al.* (1996) calculated that L1+L2 regions have a relative gene concentration of 4%, H1+H2 20% and H3 76% respectively. This means that the gene density in very G+C rich DNA segments is almost 20 times higher than in A+T rich regions.

Important structural properties of genes are found to be strongly correlated with G+C content (Duret *et al.*, 1995): genes from G+C poor isochores code for proteins that are on average longer then those from G+C rich isochores, intronic DNA is on average three times longer in L1+L2 than in H3 and the number of introns per gene is higher in L1+L2 than in H3.

How does compositional variability in genomic sequences affect performance of gene-finding programs? Burset and Guigo (1996), Snyder and Stormo (1995) and Lopez *et al.* (1994) have shown in their analyses that gene-finding programs usually perform worse when the G+C content is low. The proposed reasons for this anomaly are that G+C rich genes have stronger codon bias that makes them easier to identify and that they are more frequent than the genes in A+T rich isochores. In another study Guigo and Fickett (1995) have shown that coding statistics used by gene-finding programs (codon, dicodon and hexamer frequency) are strongly dependent on G+C content.

It is obvious that if a program has only one set of parameters intended to model gene structure (oligonucleotide frequency, length of coding and intergenic region, exon and intron length and number) it will not be able to perform equally well in both A+T and G+C rich sequences due to the significant structural differences between genes in these sequences. The reason why programs perform better for G+C rich sequences could also be because they are trained on the sequence subset of GenBank, which is biased towards G+C rich sequences. According to Duret *et al.* (1995) genes from G+C rich isochores are much more frequently sequenced than those from G+C poor isochores.

C+G content	***< 40 %***		***40 –50 %***		***50 –60 %***		***> 60%***	
	AC	***(Esn+Esp) /2***	***AC***	***(Esn+Esp) /2***	***AC***	***(Esn+Esp) /2***	***AC***	***(Esn+Esp) /2***
FGENES	0.84	0.70	0.81	0.64	0.85	0.71	0.87	0.66
GeneMark.hmm	0.79	0.48	0.80	0.46	0.87	0.62	0.85	0.48
Genie	0.85	0.69	0.85	0.60	**0.92**	0.75	0.87	**0.79**
Genscan	**0.94**	**0.80**	**0.91**	0.66	0.91	0.74	0.88	0.70
HMMgene	0.91	0.76	0.90	**0.73**	**0.92**	**0.79**	**0.91**	0.77
Morgan	0.65	0.29	0.72	0.49	0.69	0.43	0.69	0.37
MZEF	0.66	0.71	0.65	0.50	0.70	0.62	0.58	0.53

Table 2: Accuracy versus G+C content - The HMR195 dataset was partitioned according to the G+C% content of the sequences. For each program, *AC* and *(Esn+Esp)/2* are averaged over all sequences belonging to particular partition for which they are defined. The best result ina column is given in bold face.

Recently some programs, such as Genscan, HMMgene and MZEF tested in this survey, have adopted the approach of using distinct, empirically derived model parameters for distinct G+C compositional regions.

Table 2 presents the programs' accuracy measures on the sequences with different G+C content. The HMR195 dataset was partitioned into four groups according to G+C content of the sequences. These groups are closely related to previously defined isochores except that the very G+C rich isochore was split into two groups because it was heavily populated. 7% (14/195) of the sequences came from L1+L2 isochores (more precisely with G+C% ≤ 40%), 35% (69/195) of sequences from H1+H2 (40% < G+C% ≤ 50%) and 58% (112/195) of sequences from H3 (G+C% > 50%), which were subsequently split into two groups (50% < G+C% ≤60% and G+C% > 60%), containing 93 and 19 sequences, respectively. These percentages are significantly inconsistent with the results from Bernardi (1993), which points out the huge bias for the G+C rich sequences in GenBank.

Consistent with the observations made in Burset and Guigo (1996), it seems that some programs are sensitive to G+C content of a sequence, performing better when the sequence is G+C rich. Programs exhibiting this behaviour in our analysis are FGENES on the nucleotide level, GeneMark.hmm and Genie on both levels and HMMgene marginally on the exon level. Among programs that are known to use different parameter sets for different G+C content, the Genscan and HMMgene prediction accuracy is relatively independent of the base composition, but MZEF still has very variable results, especially on the exon level, that are not proportional to the G+C content of a sequence. The situation is similar for Morgan.

There is one peculiarity in Table 2: all the programs, except Morgan, have the lowest accuracy measures averaged on the sequences with G+C content between 40 and 50 %. Since this is not the region with the lowest G+C composition, it is not clear if the program really do perform most poorly for this type of sequences or there is some characteristic of the test set that causes this slight drop in prediction accuracy.

2.6.2 Exon length

The length distributions of different gene elements differ considerably between each other. Introns seem to have an approximate geometric length distribution (Burge, 1997; Hawkins, 1988), which is a characteristic of a discrete stochastic process with the ‘memoryless’ property (Karlin and Taylor, 1975). This supports the idea that introns do not have any significant constraints on their length, except that the minimal number of nucleotides (70-80) is required (Wieringa *et al.*, 1984).

On the other hand, exons have significant functional constraints. The exon length plays an important role in proper splicing and inclusion in the mature mRNA (Dominski and Kole, 1991). These constraints have shaped the exon length distribution quite differently from a geometric distribution. The length distribution depends on the exon type. Internal exons have length distribution close to a Gaussian distribution with a broad peak between

100 and 170 bp (Hawkins, 1988). Hawkins calculated the mean internal exon length to be 137 bp (in the HMR195 dataset 136 bp) and he observed very few exons shorter than 50 and longer than 300 nucleotides. Length distributions for initial and terminal coding exons are not recognizable statistical distributions. They are still substantially peaked around 60 and 160 bp respectively (Hawkins, 1988; Burge, 1997), but do not have a steep drop-off in density after 300 bp. Both types of exons are more variable in length than internal exons and their calculated means are 134 bp for initial exons and 198 bp for terminal exons (Hawkins, 1988) (in the HMR195 dataset 207 bp for initial and 265 bp for terminal). For the fourth class of exons, single-exons, the length distribution is not known, but in general they are much longer than any other type of exons and their mean length is calculated to be 1300 bp (Hawkins, 1988) (in the HMR195 dataset 1010 bp).

In our analysis we have grouped exons by both their annotated length and their predicted length and averaged the accuracy measures in each group. Since many programs tested in this analysis (Genie, GeneMark.hmm, Genscan, HMMgene) use explicit duration HMMs, which have length distribution associated with each state of the model, it is interesting to see how these distributions influence the accuracy of their exon prediction.

From Table 3 it can be observed that the general trend of all the programs is to have a very low proportion of correctly predicted short exons, which then slowly but monotonically rises with the length of annotated exons. For almost all of the programs exons are most accurately predicted if their length ranges between 75 and 200 nucleotides (these exons were the most common: 560 out of 839). The exons longer than 200 nucleotides (the HMR195 dataset contained 131 of these exons) seem more difficult to predict correctly and the accuracy measures drop further as the length increases. The exception is HMMgene that predicts longer exons with the same accuracy as the more common medium length exons.

The exons shorter than 25 bases (there were only 17) are missed in 41% of cases for FGENES up to 88% for MZEF. The most plausible explanation for this phenomenon is that the length of the coding region is too short to be clearly distinguished from surrounding non-coding regions. Also, there is biochemical evidence that this type of exon is inefficiently spliced in vivo without the presence of special splicing activating sequences (Dominski and Kole, 1991). And finally, the associated length distributions used by some programs do not

	Length range of exons in bp						
Programs	***0 – 24***	***25 – 49***	***50 – 74***	***75 – 99***	***100 – 199***	***200 – 299***	***300 +***
FGENES	**0.45** (0.33)	0.55 (0.42)	**0.71** (0.64)	0.80 (0.75)	0.80 (0.81)	0.71 (0.61)	0.59 (0.66)
GeneMark.hmm	0.05 (0.12)	0.39 (0.51)	0.60 (0.58)	0.77 (0.72)	0.75 (0.73)	0.67 (0.62)	0.46 (0.45)
Genie	0.27 (0.18)	0.53 (0.47)	0.60 (0.66)	0.80 (0.81)	0.70 (0.83)	0.71 (0.68)	0.69 (0.69)
Genscan	0.18 (0.29)	0.45 **(0.81)**	0.68 **(0.79)**	**0.89** **(0.85)**	**0.84** (0.76)	**0.87** (0.71)	0.66 (0.65)
HMMgene	0.23 **(0.42)**	**0.59** (0.76)	0.64 (0.75)	0.79 (0.77)	0.80 **(0.85)**	0.78 **(0.72)**	**0.77** **(0.74)**
Morgan	0.30 (0.14)	0.37 (0.14)	0.38 (0.31)	0.61 (0.57)	0.51 (0.57)	0.51 (0.41)	0.42 (0.35)
MZEF	0.00 (0.00)	0.16 (0.44)	0.32 (0.45)	0.40 (0.58)	0.49 (0.73)	0.45 (0.53)	0.12 (0.26)

Table 3: Accuracy versus exon length - The HMR195 dataset was partitioned according to the length of the annotated exons. For each program, *CRa* - the proportion of real exons that are correctly predicted (the upper number) and *CRp* – the proportion of predicted exons that are correct (the number in parentheses), are averaged over all sequences belonging to particular partition.

favour very short exons, and depending how these distributions are used by the systems this may cause poor prediction for this type of exons.

Although very long exons are less likely to be predicted correctly than medium length exons, they are most unlikely to be completely missed. The number of partially predicted exons longer than 300 nucleotides is relatively large (data not shown) and only less then 7% of them are completely missed (the exception is MZEF with 33% of exons missed).

Finally, it can be noted from Table 3 that there is usually a significant difference between *CRa* and *CRp* for very short exons. The reason for this is that while programs FGENES, Genie and Morgan overpredict short exons the rest of the programs underpredict them - the total number of short exons predicted by each of these programs is much lower than the actual number of exons of the same size. Again, this may be the consequence of exon length distribution built into the gene-finding programs. This discrepancy in number of real and predicted exons is much smaller for the longer exons.

2.6.3 Exon type and signal prediction

Table 4 summarizes accuracy measures for different exon types. What can be observed is a striking difference between the proportion of correctly predicted internal exons on one side and the proportion of correctly predicted initial and terminal exons on the other side. This difference is partially eroded with a high number of partially predicted initial and terminal exons (data not shown), especially if we allow the predicted exon to be of any type, but still initial and terminal exons are more likely to be completely missed than internal exons. For single-exon genes the situation is similar in the sense that the *CR* is usually significantly lower than for internal exons (the exceptions are HMMgene and Genie), but they have very high values for *PC* (the extreme case is GeneMark.hmm with *CRa*=0.30 and *PCa*=0.56 for single exons). The difference is that single exon genes are very rarely missed and the proportion of missed exons of this type is the lowest among all exon types and all programs. The only program that is almost equally successful in predicting exons of any type is HMMgene, which also has the highest proportion of correctly predicted exons (*CRp*) for initial, terminal and single exon among all other programs. This HMMgene characteristic surely contributes to its excellent results on the HMR195 dataset.

Why are initial, terminal and single exons more difficult to identify? The only obvious structural differences between different types of exons are the signals bordering them: there are no studies showing that codon usage (hexamer frequency) fluctuates between different exon types. The difference in exon length could be a possible reason, since internal exons (136 bp in the HMR195 dataset) belong to a group of exons more likely to be identified correctly than exons longer than 200 bp which is the case with initial and terminal exons (207 bp and 265 bp, respectively) (see Table 3). However, the difference in accuracy

	Exon type			
Programs	***Initial***	***Internal***	***Terminal***	***Single***
FGENES	0.64 (0.55)	0.79 (0.78)	0.66 (0.58)	0.58 (0.83)
GeneMark.hmm	0.40 (0.48)	0.78 (0.72)	0.52 (0.51)	0.30 (0.65)
Genie	0.49 (0.45)	0.76 (0.82)	0.61 (0.57)	0.70 (0.68)
Genscan	0.57 (0.71)	**0.87** (0.76)	0.67 **(0.73)**	0.63 **(0.83)**
HMMgene	**0.68** **(0.72)**	0.78 **(0.83)**	**0.70** **(0.73)**	**0.77** (0.79)
Morgan	0.35 (0.35)	0.55 (0.46)	0.36 (0.36)	-
MZEF	-	-	-	-

Table 4: Accuracy versus exon type - The HMR195 dataset was partitioned according to the type of the annotated exons. For each program, *CRa* (the upper number) and *CRp* (the number in parentheses), are averaged over all sequences belonging to that particular partition.

level observed in Table 3 do not compensate for the high differences observed in Table 4. The hypothesis that signal prediction is mainly responsible for the difference we see in accuracy levels is supported by the results in Table 5: detection of start and stop codons is much less accurate than of acceptor and donor sites (again, the exception is HMMgene) and the difference in accuracy level is proportional to the accuracy level difference for initial and terminal exons versus internal exons in Table 4. As noted above, during the assembly of HMR195 we were not able to validate the locations of annotated start and stop codons. Consequently, prediction accuracy measures calculated for these signals, as well as subsequent analysis and discussion strongly rely on the correctness of their annotation in GenBank.

	Signal type			
Programs	***Start codon***	***Acceptor site***	***Donor site***	***Stop codon***
FGENES	0.67 (0.63)	0.80 (0.77)	0.85 (0.82)	0.75 (0.72)
GeneMark.hmm	0.46 (0.60)	0.81 (0.75)	0.82 (0.78)	0.57 (0.64)
Genie	0.56 (0.57)	0.77 (0.82)	0.78 (0.83)	0.72 (0.73)
Genscan	0.61 **(0.78)**	**0.87** (0.80)	**0.90** (0.84)	0.76 **(0.86)**
HMMgene	**0.75** **(0.78)**	0.81 **(0.85)**	0.83 **(0.87)**	**0.78** (0.81)
Morgan	0.43 (0.43)	0.66 (0.57)	0.65 (0.56)	0.39 (0.39)
MZEF	-	0.59 (0.65)	0.66 (0.73)	-

Table 5: Accuracy versus signal type - The HMR195 dataset was partitioned according to the signal type in the annotated genes. For each program, *CRa* (the upper number) and *CRp* (the number in parentheses), are averaged over all sequences belonging to that particular partition.

The situation is a bit more complex for single-exon genes: on the one hand they contain both start and terminal codons which should complicate their identification even further, but on the other their average length in the dataset is 1010 bp, which according to the analysis in Section 2.6.2 makes them hard to predict exactly, but difficult to miss. This directly corresponds to the results in Table 4.

What can also be observed in Table 4 is that for programs FGENES, GeneMark.hmm and Genscan there is a significant difference between *CRa* and *CRp* for the single-exons. Analogously to the case of very short exons the cause of this phenomenon is that these programs are conservative in predicting single-exon genes: the number of single-exons predicted by any of these programs is much lower then the number of real ones in the dataset.

The consequence of this is that single-exons predicted by these programs have a very good chance of being correct, while many real ones remain unidentified.

2.6.4 Exon probabilities and scores

Each of the programs evaluated in this study, except GeneMark.hmm, has a scoring scheme for its exon prediction. Genscan, HMMgene and MZEF have a probability score for each exon predicted that is supposed to be a quantitative measure of the likelihood that the given exon is correct. Morgan's scores were originally intended to be probabilities but that intention was not followed through subsequent upgrades and what is left is a scale with no formal meaning except that very high scores result from motifs that Morgan has seen before. Genie uses the bit score against a background distribution that is dependent on the length of the predicted exon and thus can not be meaningfully compared to other exon scores. The way FGENES calculates exon scores is not known to us.

Since the nature of Morgan's and Genie's scores makes them uninformative for a user, we tested the reliability of the exon scores for the other four programs: FGENES, Genscan, HMMgene and MZEF.

The results for FGENES (not presented) appear to show that its scores are not directly useful. Most of the exons predicted have a score less than 10 and *CR* values average to the similar levels for any subregion on the scale from 0 to10. The only informative exon score is above 10, since, at least in the HMR195 dataset, these exons are correctly predicted in 90% of cases, which is significantly higher than *CR* for exons with lower scores. However, scores this high are rarely assigned to an exon prediction.

The accuracy measures for different regions of probability scores for Genscan, HMMgene and MZEF are displayed in Table 6. What can be observed is that *CR* values are monotonically rising with the increase in exon probability. For Genscan and HMMgene these values are usually close to the lower boundary of a probability range (the exception is the probability region 0.90 - 0.95, where *CR* values are lower than probabilities). MZEF, on the other hand, significantly overestimates probabilities for its exon predictions. If the partially predicted exons are included (the results for *PC* are not shown) *CR* values will reach and

	Probability range of predicted exons				
Programs	***0.00 – 0.50***	***0.50 – 0.75***	***0.75 – 0.90***	***0.90 – 0.95***	***0.95 +***
Genscan	0.32	0.45	0.75	**0.84**	0.94
HMMgene	0.32	**0.65**	**0.79**	0.83	**0.95**
MZEF	-	0.43	0.54	0.64	0.74

Table 6: Accuracy versus probability - The HMR195 dataset was partitioned according to probability of the predicted exons. For each program, *CRp* - proportion of predicted exons that are correct, is averaged over all sequences belonging to that particular partition.

sometimes overreach the upper boundary of a probability region (*CR* for MZEF will correspond to probability region average).

This analysis shows that in the case of Genscan and HMMgene the exon probability score can be a very useful guide to the reliability of the exon prediction.

2.6.5 Phylogenetic specificity

All of the programs analyzed in this survey were trained on human sequences, except Morgan, which was trained on the dataset of vertebrate sequences collected by Burset and Guigo (1996). Since the dataset used to test the programs was composed of 103 human and 92 murine (82 *Mus musculus* and 10 *Rattus norvegicus*) sequences we wanted to investigate if such a phylogenetic mix can corrupt the performance of the gene-finding programs, especially those calibrated for human sequences.

Results for *AC* on nucleotide level and for *(ESn+ESp)/2* on exon level for each of the programs, but separately for human and murine sequences, are given in Table 7. It can be observed that the difference in accuracy measures between human and mouse/rat are

Programs	***Trained on***	***Nucleotide accuracy - AC***		***Exon accuracy – (ESn+ESp)/2***	
		Human	***Murine***	***Human***	***Murine***
FGENES	Human	0.85	0.82	0.67	0.68
GeneMark.hmm	human	0.83	0.84	0.56	0.51
Genie	human	0.88	0.89	0.73	0.67
Genscan	human	0.89	0.92	0.72	0.70
HMMgene	human	0.90	0.92	0.74	0.79
Morgan	vertebrate	0.64	0.75	0.43	0.44
MZEF	human	0.65	0.66	0.59	0.58

Table 7: Phylogenetic specificity – The HMR195 dataset was split into two species subsets containing 103 human and 92 murine sequences. For each subset and each program *AC* and *(ESn+ESp)/2* were averaged over all sequences belonging to the particular subset.

marginal. Even more interesting is that in most cases the values for murine sequences are higher then the values for human sequence, even though the model parameters of the programs were learned from the set of human sequences.

It is likely that such differences are not statistically significant and that they would also be observed if the results on two different human sequence sets were compared. This hypothesis is also supported by comparison of the human and mouse grammars constructed by Dong and Searls (1994) where no statistically significant differences were found.

2.7 Conclusions from the evaluation of gene-finding programs

The results obtained in this analysis indicate that the new generation of programs has significantly higher accuracy than the programs analyzed in Burset and Guigo (1996). Comparing the programs with the highest approximation correlation in their study and our study, we find that it has improved from 0.78 (FGENEH) to 0.91 (Genscan and HMMgene), a 17% increase, while the highest averaged exon sensitivity and specificity has improved from 0.64 (FGENEH) to 0.76 (HMMgene), a 19% increase.

The behaviour of the programs on the sequences with different G+C content is not systematic: some programs' accuracy appears to be slightly dependent on the G+C content, while programs such as Genscan and HMMgene, which use different parameter sets for different G+C content, perform steadily for any G+C content.

The accuracy of exon prediction is dependent on the length of the exon. The general trend of the programs is to have a very low proportion of correctly predicted short exons, which then rises with the length of annotated exons. For almost all of the programs 'medium' exons, whose length ranges between 70 and 200 nucleotides, are most accurately predicted. The accuracy decreases again for exons longer then 200 bp (the exception is HMMgene), but very few of them are missed completely.

The analysis of accuracy prediction as a function of the exon type reveals that internal exons are much more likely to be predicted correctly than other types of exons. The cause of this phenomenon is a weakness in the detection of start and stop codons, which border other types of exons. Initial and terminal exons are most likely to be missed completely, while single exons although difficult to predict exactly (they contain both start and stop signals) are rarely missed due to their substantial length.

Among all the programs analyzed only Genscan and HMMgene have reliable scores for exon prediction.

Our goal was not to obtain the ultimate accuracy results for the programs tested, but rather to conduct the first independent, comparative evaluation of the recently developed gene-finding algorithms. Obtaining definitive accuracy results is an impossible task since the performance of the programs is very sensitive to the dataset they are tested on, as observed by many authors.

Our evaluation was based on a dataset that was carefully prepared, containing only ‘text book’ genes. Even if the sequences had been selected in more flexible manner, they would still be biased because the present public sequence databases are biased: genes that are more difficult to isolate or to sequence (e.g., very long genes found in A+T rich regions) are underrepresented, while there is a great deal of redundancy with overrepresentation of some gene families. Also genes currently present in databases reflect the interests of the scientific community (e.g., disease genes) and are not a random sample of the genome. More details about biases in the sequence databases can be found in (Duret *et al.*, 1995). The evaluation of gene-finding programs on more realistic sequence datasets (longer genomic sequences with more complex gene structure and less coding density) would almost certainly result in considerably lower accuracy measures than those obtained in this study. The results presented here should be considered as upper bound estimates of the programs' accuracy when they are used on typical genomic sequences. This situation may improve when the programs get retrained on new, more diverse genomic sequences.

There are certain assumptions that had to be made in order to obtain accuracy measures for the programs tested. Although we were able to validate exon/intron boundaries, we did not have a methodology to confirm start and stop codon positions and therefore had to assume that they were correctly annotated in GenBank. Also, 5' and 3' flanking sequences were assumed to be exonless and every prediction made in those regions was considered incorrect, which might eliminate some perfectly valid predictions. On the other hand, it is very unlikely that some predicted internal exons were in fact real, but not previously detected, because in that case we would have observed an unaligned mRNA piece when using the sim4 algorithm. The possibility exists that some of the genes in the HMR195 testset have other, still unknown, splice variants, and that some of the exons predicted actually belong to some of them.

Although, the programs for gene structure predictions have greatly improved in the last decade, from the simple ORF finders to sophisticated heterogeneous systems incorporating various evidence for gene structure, even the best of them cannot be used autonomously for the detection of genes and other genomic elements. The programs still have a considerable proportion of incorrect and missed exons and additional evidence is usually needed to confirm their predictions. Also, they concentrate only on detection of coding exons, while 5' and 3' UTRs, promoter elements and polyA sites often remain undetected. Elucidation of complex genome organization, such as nested and overlapping genes or alternative splicing, has not yet been considered by any program. Even the signal sensors, especially for start and stop codon, which have been in use for a long time, seem to be rather weak and should allow significant room for improvement.

To achieve the ultimate goal of automatic annotation of genomes, better understanding of the biological processes involved in transcription, mRNA processing and translation is required. However, improvements can also be made by further development of existing methods, especially signal sensors and regulatory regions models, and calibration of programs' parameters on more diverse genomic sequences.

Chapter 3

Combining predictions from gene-finding programs

In the second part of the thesis a different approach to improving gene prediction accuracy is explored; instead of attempting to improve some components of gene prediction algorithms or designing a new one from a scratch we developed methods for combining the evidence (i.e., predictions) from two gene-finding experts (i.e., programs). This approach was motivated by previous attempts to use gene-finding programs in this manner as well as by the results of the evaluation presented in previous chapter. The evaluation allowed us to identify the most accurate programs, which were used as gene-finding experts, and also to recognize the weaknesses of gene prediction that could be remedied by this approach and the strengths of the individual programs that could be used to achieve the highest possible accuracy. This chapter describes in details the motivation for and implementation and results of methods for combining the predictions from two gene-finding programs.

3.1 Background and related work

Current gene prediction programs are sophisticated systems that integrate many different methods for identifying elements of the genes, as discussed in the previous chapter. The set of methods used and the way they are integrated differs among individual programs, as well as the sequence training sets used to build signal models and tune programs' parameters. Being distinct in their architecture and training, programs often give different gene structure predictions for the same DNA sequence. This characteristic of programs' predictions has motivated several authors to investigate the benefits of combining several gene-finding programs.

Burset and Guigo (1996) investigated the correlation between six *ab initio* gene-finding programs that they evaluated. The approximate correlation of the predictions at the nucleotide level varied from 49% to 68% and the average exon accuracy varied from 24% to 47%, when predictions from two programs were compared. The exons predicted by all of the programs tested were correct in 99% of cases and the proportion of exons completely missed by any of the programs was 1%. This analysis shows that the programs' predictions are considerably different and that each program can contribute to finding all annotated exons (only 1% missed), maximally increasing the sensitivity of the prediction, but decreasing specificity. On the other hand, if an exon is predicted by all programs that almost certainly guarantees the correctness of the prediction (only 1% wrong), maximally increasing the specificity of the prediction, but decreasing sensitivity. This indicates that by combining several gene-finding programs it is possible to improve prediction accuracy, but in order to improve overall accuracy sensitivity and specificity have to be increased simultaneously.

A more comprehensive study of methods for combining gene-finding programs was done by Murakami and Takagi (1998). They used five different methods to combine four gene-finding programs: FEXN (Solovyev *et al.*, 1994), GeneParser3 (Snyder and Stormo, 1995), Genscan (Burge and Karlin, 1997) and Grail2 (Uberbacher and Mural, 1991). The methods they tested were: the AND-based method, the OR-based method, the HIGHEST method, the RULE method and the BOUNDARY method. The first two methods coincide with Burset and Guigo's approach of accepting only exons predicted by all the programs or accepting exons predicted by any of the programs. The other three methods use normalized exon or splice site scores to decide on the exon candidates. While approximate correlation was significantly improved when FEXN, GeneParser3 and Grail2 were combined by some of the methods (up to 10%), improvements were more marginal when Genscan was used because it was more difficult to outperform Genscan's high prediction accuracy. The best result of this analysis was a 4.7% increase of *AC* and a 2.5% increase of average exon accuracy comparing to the best individual program (Genscan).

The approach that we am proposing in this thesis is different from the ones previously described. Rather then combining several gene-finding programs including those with generally low prediction accuracy we decided to combine only the two best-performing ones. Relying on the results of the evaluation of recently developed programs presented in the previous chapter as well as on some analysis described in the next couple of sections the two best candidates for this study were chosen.

3.2 Selection of the programs and datasets

3.2.1 Correlation between the programs

Five of the seven tested programs were selected to investigate correlation between pairs of programs. Each of these five programs can predict one or more single- or multi-exon genes in a sequence. The remaining two programs were not considered because of their limitations: Morgan can predict only a single multi-exon gene in a sequence and MZEF can predict only internal exons.

The results of the correlation analysis are shown in Table 8. For each pair of programs we calculated the number of exons predicted exactly (both exon boundaries predicted correctly) by at least one of the two programs. The numbers of exactly predicted

	FGENES	GeneMark.hmm	Genie	Genscan	HMMgene
FGENES	697 (0.74)				
GeneMark.hmm	799 (0.84)	625 (0.66)			
Genie	824 (0.87)	773 (0.82)	651 (0.69)		
Genscan	840 (0.89)	796 (0.84)	807 (0.85)	735 (0.78)	
HMMgene	825 (0.87)	811 (0.86)	793 (0.84)	826 (0.87)	715 (0.75)

Table 8: Correlation between the programs – For each pair of the programs we calculated the number of exons predicted correctly for the whole HMR195 dataset by at least one of the two programs. The number in parenthesis is the proportion of real exons that were predicted correctly. The numbers on the diagonal are the results for individual programs. The total number of annotated (real) exons for the HMR195 is 948.

exons by a single program are given on the diagonal of the Table 8. The total number of annotated exons for the HMR195 dataset is 948. Comparing the numbers on the diagonal of Table 8 with the numbers below it we can see that any pair of programs can predict more exons correctly than any single program. The reason for this is that each program when compared to any other program has a set of exactly predicted exons that were not predicted by the other program. Consequently, any of the gene-finding programs investigated could contribute to the sensitivity of the prediction of any other program. But, of course, this simplistic approach would simultaneously decrease the specificity of the prediction and our goal is to combine the predictions in such a way that the correctly predicted exons are preserved while wrong exons are discarded.

3.2.2 Exon probability scores

As discussed in the previous chapter, each of the programs evaluated in this study, except GeneMark.hmm, has a scoring scheme for its exon prediction. However, our analysis has shown that only Genscan and HMMgene exon probability scores, which give the quantitative measure of the likelihood that the given exon is correct, are reliable. Figure 1 shows therelationship between Genscan and HMMgene exon probability scores and the proportion of exactly predicted exons. We can see that there is an approximate linear dependence between these two variables for both programs and that the proportion of exactly predicted exons monotonically increases with the increase of exon probability score (disregarding a small anomaly for Genscan). This means that the exons with the higher scores are usually more accurate then the exons with lower scores and in the case of HMMgene the likelihood

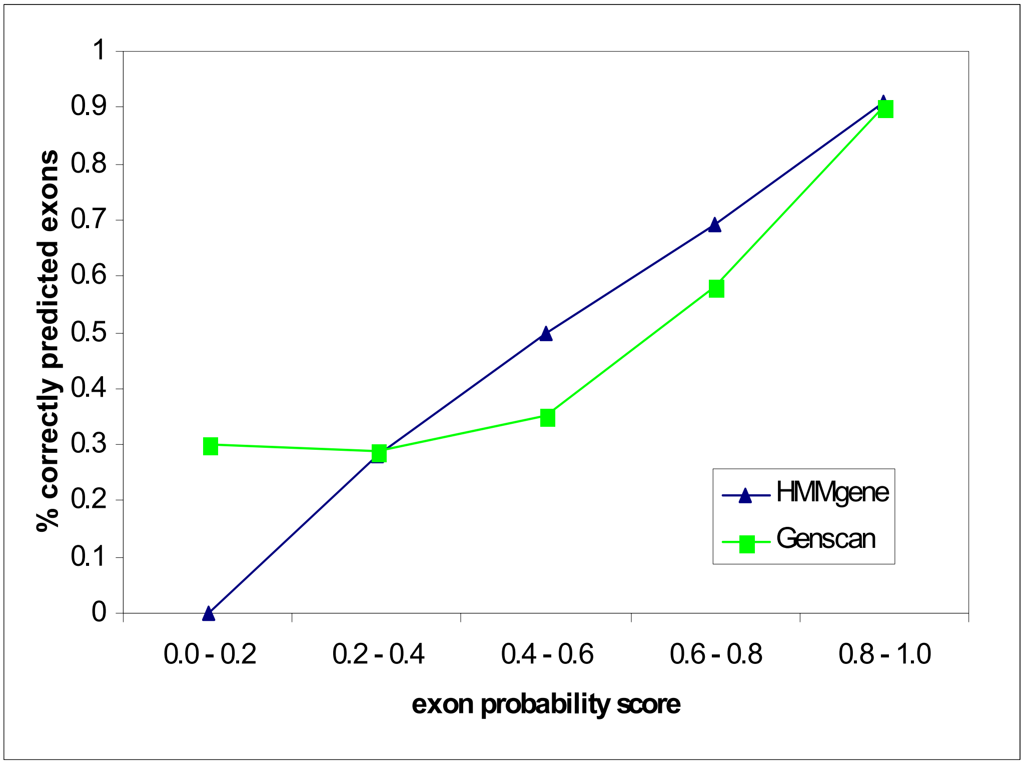

Figure 1: Reliability of exon probability scores – Genscan's and HMMgene's proportion of correctly predicted exons vs. exon probability scores.

of correct prediction is almost perfectly estimated with the exon score. This characteristic of Genscan and HMMgene exon probability scores makes them very useful guides in deciding on the correctness of a predicted exon.

3.2.3 Selection of the programs

Considering the results of the gene prediction program evaluation and Table 1 we concluded that Genscan and HMMgene were the best candidates for our study since they have the best overall prediction accuracy and almost the highest number of correctly predicted exons when combined (only the FGENES - Genscan combination has a better score) (Table 8). However, the most important reason for selecting these two programs is the reliability of their exon probability scores as shown in Figure 1, since the methods developed rely on these scores when combining the predictions from the programs.

3.2.4 Sequence datasets used

The previously described HMR195 sequence dataset, containing 195 human, mouse and rat sequences, was used to develop and test methods for combining the Genscan and HMMgene predictions. We do not consider it a typical training dataset because it was mostly used to study the strengths and weaknesses of the programs and only one method parameter was derived from it. However, to ensure independent testing of the methods' two additional control datasets were used: the Burset/Guigo dataset and a *Drosophila melanogaster Adh* region used in the Genome Annotation Assessment Project (GASP) (Reese *et al.*, 2000).

The dataset assembled by Burset and Guigo (1996) consists of 570 vertebrate genomic sequences containing exactly one multi-exon gene. Similarly to the HMR195 dataset it has been filtered to exclude anomalous sequences.

The *Drosophila melanogaster Adh* region is nearly 3 Mb long and has been extensively studied for the last 20 years (Ashburner, 2000). For the GASP experiment two different annotation sets were used to evaluate the gene-finding programs' predictions: st1 and st3. The first set, called standard set 1, contained only highly accurate annotations, confirmed by aligning full-length cDNA sequences from this region with the high-quality genomic sequences. This approach left out many potential genes that did not have a matching cDNA. St1 originally contained annotation for 43 transcripts, but after some incorrect sequences were removed, the number of genes is 38. The second and more complete annotation set, st3, containing 222 gene structures, was compiled by biology experts using information from various sources: BLAST results, PFAM alignments, high scoring Genscan and Genefinder predictions, ORFFinder results, full-length cDNA alignments and alignments with genes from GenBank. Out of 222 annotated genes only 40 were based solely on strong Genscan and Genefinder predictions.

3.3 Combining the programs' predictions

This section describes methods for integrating the Genscan and HMMgene predictions. Our goal was to provide computationally straightforward techniques for combining the output from these two programs. On the assumption that they can be considered partially independent sources of evidence for gene structure, it should be possible to use output from the programs as follows: when either program is quite confident of its exon prediction use it regardless; in the cases where both programs are less certain of their exon prediction use it if they both agree.

Both Genscan and HMMgene produce output files for each DNA sequence submitted. The output files give the details of the gene structure predictions made by the programs. Each file contains enumerated exons with their location, type and probability score. Exons are labeled according to the gene they belong to. The sample Genscan and HMMgene output files are given in Appendix C. The methods that are described below use the information from the output files to decide on the candidate exons. Three different algorithms EUI (Exon Union-Intersection), GI (Gene Intersection) and EUI_frame (Exon Union-Intersection with Reading Frame Consistency) are described:

Algorithm EUI (Exon Union-Intersection)

1. Consider all the Genscan and HMMgene exons that have exon probability score greater or equal to a threshold p_{th}. The regions predicted by at least one of the programs are labeled as EUI exons (exon union - see Figure 2).

2. Consider all the Genscan and HMMgene exons that have exon probability score less than p_{th}. The regions predicted by both programs are labeled as EUI exons (exon intersection - see Figure 2).

Consequently, a Genscan or HMMgene exon that does not overlap any exon predicted by the other program will be accepted if its exon probability is greater or equal to p_{th} and refused otherwise.

There is one exception for step 1: if Genscan's internal exon has the same right boundary (donor site) as HMMgene's initial exon (both exons have the score greater or equal to p_{th}) choose HMMgene's exon prediction as an EUI exon. This ‘initial exon rule’ was incorporated into the EUI method after the analysis showed that Genscan often predicts initial exons as internal, which have the correct donor site but false acceptor site preceding the true ATG codon. HMMgene's predictions of the initial exons are more accurate (Table 5).

Algorithm GI (Gene Intersection)

1. For each program's prediction select regions predicted as genes (genes are treated as continuous sequence from the beginning of the first predicted exon in the gene to the end of the last predicted exon). Regions predicted by both programs are labeled as GI genes (gene intersection - see Figure 2).

2. Apply the EUI method to those exons that completely belong to GI genes (where both exon boundaries are within a GI gene).

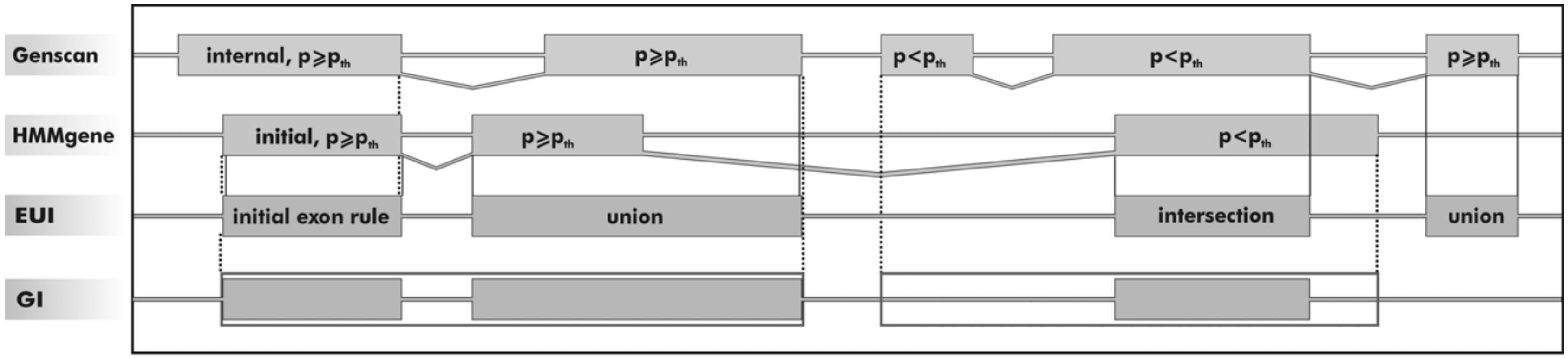

Figure 2: Graphical representations of EUI and GI methods - The dotted lines mark the boundaries of the GI genes and the solid lines mark the boundaries of EUI exons. The labels on the EUI exons indicate which part of EUI algorithm was used to determine the exon.

This approach is primarily designed for identification of genes in long genomic regions where another level of constraints, namely considering only exons that belong to regions predicted as genes by both programs, helps further eliminate numerous wrong exons typical for *ab initio* predictions in the long sequences.

Algorithm EUI _frame (Exon Union-Intersection with Reading Frame Consistency)

This method applies the EUI method to the Genscan and HMMgene predictions while maintaining reading frame consistency:

1. For each program's prediction determine the gene boundaries and to each gene assign a gene probability calculated as the average of exon probability scores for all the exons contained in that gene. For each predicted exon determine the positions of acceptor and donor site in a reading frame of a gene it belongs to.
2. If the gene predicted by Genscan overlaps the gene predicted by HMMgene, choose the one with the higher gene probability to impose the reading frame. Apply the EUI method to the exons belonging to the selected genes accepting EUI exons only if they are in the chosen reading frame.

The threshold value p_{th} that is used in all three methods has been empirically derived using the HMR195 dataset. The optimal value is p_{th}=0.775. However, the methods' accuracy results show very low sensitivity to the threshold variation, as can be observed in Figure 3. The average exon accuracy varies from 0.78 to 0.81 for EUI method and 0.79 to 0.82 for the GI method when the threshold value changes from 0.45 to 0.95. For both methods *(ESn+ESp)/2* peaks when p_{th} is between 0.75 and 0.80, and accordingly the average of these two values is chosen to be the threshold value. The details about the algorithms previously described can be found in Appendix A.

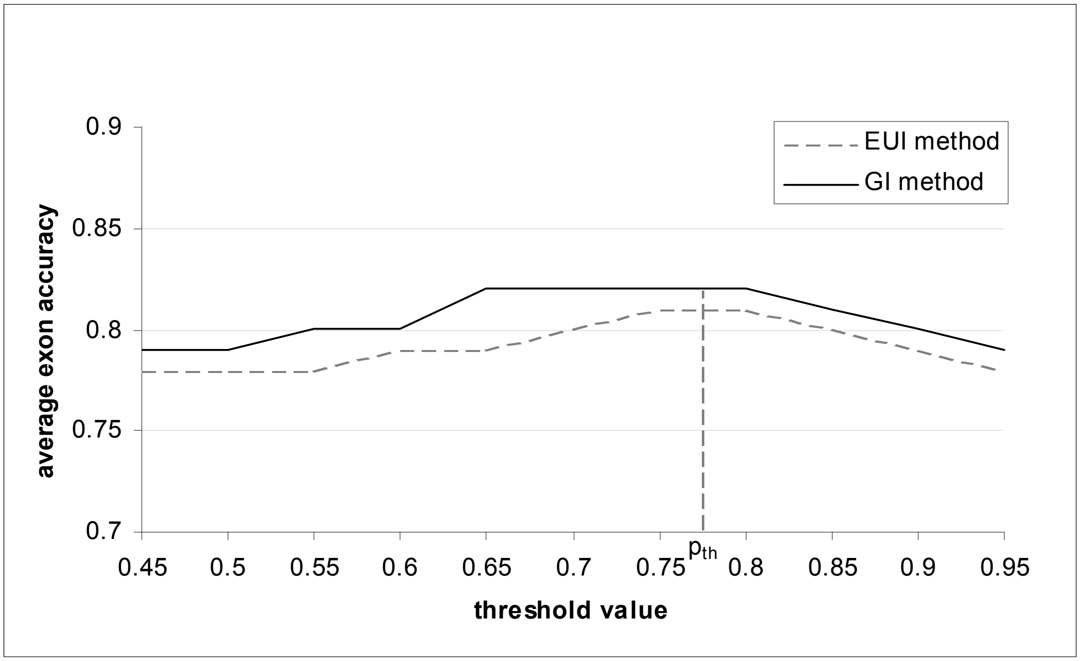

Figure 3: Threshold sensitivity – Average exon accuracy vs. threshold value. The optimal threshold value is $p_{th} = 0.775$.

3.4 Results

Accuracy measures for the three methods as well as for Genscan and HMMgene on the HMR195 dataset are given in Table 9. The numbers in bold indicate an improvement when compared to either of the two programs. It can be observed that each of the methods outperforms Genscan and HMMgene in all categories except for nucleotide level sensitivity (*Sn*) and proportion of missed exons (*ME*). The results in Table 9 suggest that each of three methods improve specificity more than sensitivity at both the nucleotide and exon levels. While sensitivity is decreased at the nucleotide level from 0.95 for Genscan to 0.91 - 0.94 for the methods, specificity is increased from 0.93 for HMMgene to 0.96 for GI method (3.2% increase) and 0.95 for EUI and EUI_frame methods (2.2% increase). At the exon level, sensitivity increased from 0.76 for HMMgene to 0.79 for EUI (3.9% increase) and 0.78 for

GI and EUI_frame methods (2.6% increase), while specificity increased from 0.77 for HMMgene to 0.86 for GI (11.7% increase) and 0.83 for EUI and EUI_frame methods (7.8% increase). These numbers also imply that improvements are substantially better at the exon level than at the nucleotide level, which is also supported by an increase of 2.2% in *AC* and an increase of 7.9% in *(ESn+ESp)/2*, when comparing only the highest accuracy values for the programs and the methods. While the number of missed exons was not improved by either of the methods, the number of wrong exons was substantially decreased: Genscan predicted 104 wrong exons, HMMgene 81 and the GI method only 44.

Results for the Burset/Guigo control set are summarized in Table 10. Bold numbers in Table 10 have the same pattern of appearance as in Table 9, which indicates that improvements are accomplished in the same categories. Similarly to the results in Table 9 improvements are better for specificity at both levels and generally better for exon level measures than for nucleotide level measures. The increases in accuracy values for this dataset were somewhat lower than for the HMR195 dataset.

METHODS	***# no prediction***	***Nucleotide accuracy***			***Exon accuracy***				
		Sn	***Sp***	***AC***	***ESn***	***ESp***	***(ESn+Esp)/2***	***ME***	***WE***
Genscan	3	0.95	0.90	0.91	0.70	0.70	0.70	0.08 (76)	0.09 (104)
HMMgene	5	0.93	0.93	0.91	0.76	0.77	0.76	0.12 (128)	0.07 (81)
EUI	3	0.94	**0.95**	**0.93**	**0.79**	**0.83**	**0.81**	0.10 (104)	**0.04** (55)
GI	15	0.91	**0.96**	**0.92**	**0.78**	**0.86**	**0.82**	0.19 (149)	**0.03** (43)
EUI_frame	3	0.93	**0.95**	**0.93**	**0.78**	**0.83**	**0.80**	0.11 (115)	**0.03** (46)

Table 9: Results for HMR195 – For each sequence in the HMR195 test set, the forward (+) strand exons in the default outputs of the programs tested were compared to the annotated exons. The standard measures of predictive accuracy on nucleotide and exon level were calculated for each sequence and averaged over all sequences for which they were defined. The second column gives the number of sequences where no prediction was made. The numbers in parenthesis in the last two columns are the actual numbers of missed and wrong exons, respectively.

METHODS	*# no prediction*	*Nucleotide accuracy*			*Exon accuracy*				
		Sn	*Sp*	*AC*	*ESn*	*ESp*	*(ESn+Esp)/ 2*	*ME*	*WE*
Genscan	8	0.94	0.93	0.92	0.78	0.81	0.80	0.09 (203)	0.05 (188)
HMMgene	38	0.93	0.94	0.92	0.81	0.83	0.82	0.14 (308)	0.04 (139)
EUI	20	0.94	**0.96**	**0.93**	**0.83**	**0.88**	**0.85**	0.12 (250)	**0.03** (98)
GI	43	0.91	**0.97**	**0.93**	**0.82**	**0.90**	**0.86**	0.18 (386)	**0.02** (67)
EUI_frame	27	0.93	**0.96**	**0.93**	**0.83**	**0.88**	**0.85**	0.13 (286)	**0.03** (87)

Table 10: Results for Burset/Guigo dataset – For each sequence in the Burset/Guigo test set, the forward (+) strand exons in the default outputs of the programs tested were compared to the annotated exons and the standard measures of accuracy calculated.

The results on the 3Mb *Adh* D*rosophila* region are shown in Table 11. The values for *Sn*, *ESn* and *ME* are calculated using annotation set st1 and the values for *Sp*, *ESp* and *WE* are calculated using st3. The rationale for this lies in the way these sets are built: st1 contains a subset of all genes in the *Adh* region that are correct in the details, while the st3 dataset is believed to be complete but the confidence in its correctness is not as high as for the st1 dataset. Thus, sensitivity, which is the measure of how well a program can predict the real coding features in a sequence, is more accurately estimated from st1 because we are sure that these annotations are correct. On the other hand, specificity, which is the measure of how well a program avoids false positive predictions, is better estimated from st3, which is thought to be complete. Similarly to the results for the previous two datasets, the three introduced methods have improved specificity more than sensitivity. At the nucleotide level specificity increased from 0.62 for Genscan to 0.75 for GI and EUI_frame methods (21.0% increase) and 0.69 for EUI method (11.3% increase), while the sensitivity values for the methods were less than or equal to the ones for the programs. At the exon level specificity

METHODS	***# of predicted exons***	***Nucleotide accuracy***		***Exon accuracy***			
		Sn	***Sp***	***ESn***	***ESp***	***ME***	***WE***
Genscan	1696	0.96	0.62	0.59	0.40	0.14 (15)	0.51 (873)
HMMgene	2101	0.95	0.61	0.49	0.19	0.14 (16)	0.66 (1379)
EUI	1376	0.96	**0.69**	**0.62**	0.40	**0.13** (14)	**0.46** (632)
GI	1043	0.92	**0.75**	0.56	**0.49**	0.19 (21)	**0.35** (366)
EUI_frame	912	0.83	**0.75**	0.55	**0.53**	0.23 (25)	**0.35** (318)

Table 11: Results for *Drosophila Adh* region – *Sn*, *ESn* and *ME* are reported for st1 annotation set and *Sp*, *ESp* and *WE* are reported for st3 annotation set. All the methods are tested on the both strands of *Adh* region.

increased from 0.40 for Genscan to 0.49 for GI (22.5% increase) and 0.53 for EUI_frame (32.5% increase), while sensitivity increased by 5.1%, (from 0.59 to 0.62) for EUI method and slightly decreased for the rest two methods when compared to the programs' best sensitivity result *ESn*=0.59. The EUI method has the lowest *ME* among the programs and the methods, while GI and EUI-frame have missed 6 and 5 more exons than Genscan, respectively. The last column in Table 11, showing the proportion of the wrong exons, illustrates the most important advantage of the methods over Genscan and HMMgene when used on a long genomic region: the number of false positive exons decreased from 873 for Genscan and 1379 for HMMgene to 632 for EUI, 366 for the GI and 318 for EUI-frame methods. The overall high numbers for *WE* are the result of a known shortcoming of gene-finding programs: overpredicting exons and genes in long stretches of genomic sequences (Dunham, 1999).

The results for HMMgene shown in Table 11 differ from those shown in Reese (2000) and Krogh (2000) for two reasons: first, the results that we report are only for *ab initio* gene-finding without using any of the additional sources of evidence, which have been incorporated in HMMgene for GASP purposes (Krogh, 2000) and second, the st1 standard set that we used is a refined version of the set used for the original GASP evaluation.

3.5 Discussion

The analysis of gene-finding programs presented in the previous chapter shows that the weakest component of the current programs is signal detection, especially the detection of initiation and termination codons, which lowers the exon level prediction accuracy. From the definitions of the exon level sensitivity and specificity in Section 2.4.2 it is obvious that *TE* value is directly proportional to *ESn* and *ESp*. Therefore, if the correct splice site is missed, even by just a couple of nucleotides, the predicted exon will not be counted as a 'true' exon, which simultaneously decreases *ESn* and *ESp*. Thus, the exon prediction accuracy could be improved in two ways: identifying the correct exon boundaries would increase the number of 'true' exons, at the same time increasing both the exon sensitivity and specificity, and reducing the number of predicted exons (*PE*) would increase exon specificity. Of course, only the dismissal of the falsely predicted exons would be beneficial for the overall increase in *ESp*.

The EUI method, which is also incorporated in the other two methods introduced above, attempts to simultaneously find more probable exon boundaries and to discard the low confidence exons. As shown in Burset and Guigo (1996) and Murakami and Tagaki (1998), selecting the union of the exons predicted by two programs (OR-method) would result in increased sensitivity but decreased specificity and analogously, the intersection of the exons (AND-method) would increase specificity but decrease sensitivity. The EUI method integrates these two approaches by using them selectively depending on the confidence in exon correctness. When the probability scores for the two overlapping predicted exons are high (greater than or equal to p_{th}) the coding region predicted by either of the programs is chosen to be a resulting EUI exon. This potentially increases the sensitivity of the prediction, which is already supposed to be specific according to Figure 1 (proportion of the correctly predicted exons is almost equivalent to the specificity). When the exon scores for the two overlapping exons are low (less than p_{th}), the region predicted to be coding by both of the programs is selected to be the resulting exon, which potentially improves the specificity of the prediction. A 'stand-alone' exon that does not overlap with any exon predicted by the other program will be accepted only if it has an exon score greater or equal to p_{th}. This further improves exon specificity by eliminating low probability exons that have a high chance of being wrong.

The relationship between Genscan and HMMgene prediction scores is shown in Figure 4. Each exon predicted by either of the two programs is represented by a data point in the graph. If two exons overlap they are represented by one dot whose coordinates correspond to the Genscan and HMMgene exon scores. The dots on the x- and y-axes represent exons predicted only by one program. We can distinguish three classes of exons from this scatter plot: the exons on the axes of the plot, which are 'stand-alone' exons, the exons predicted by both programs (they do not have to be exactly identical) with very high score, and the exons predicted by both programs whose scores from the two programs are not tightly correlated. This graph further emphasizes the non-correlation hypothesis for the two programs: first, there are many exons predicted by only one program, as shown in Table 8, and also even if the two predictions overlap, very often their scores do not agree closely.

Figure 5 presents all the false positive exon predictions made by either program. The exons are represented in the same way as in Figure 4. Figure 5 clearly shows that most of the wrong exons predicted by one program were not predicted by the other – only 55 of 447 dots in the graph are not found on the axes. Comparing Figure 4 to Figure 5, we can see that the false exons predicted by both programs are buried among numerous true predictions and it appears to be impossible to distinguish them using solely the exon scores. However, the exons plotted on the axes of the graph in Figure 5 can be easily excluded if we choose to keep only the exon predicted by both programs. This is exactly what the EUI algorithm is doing, except that it also retains all the 'stand-alone' exons with the probability greater than the threshold p_{th}. Figure 5 shows that dense clusters of dots on the axes of the plot are terminated around p_{th} and there are fewer false positives with a score higher than p_{th}. The value for p_{th} determines the trade-off between sensitivity and specificity and by choosing p_{th}=0.775 we are making them as balanced as possible.

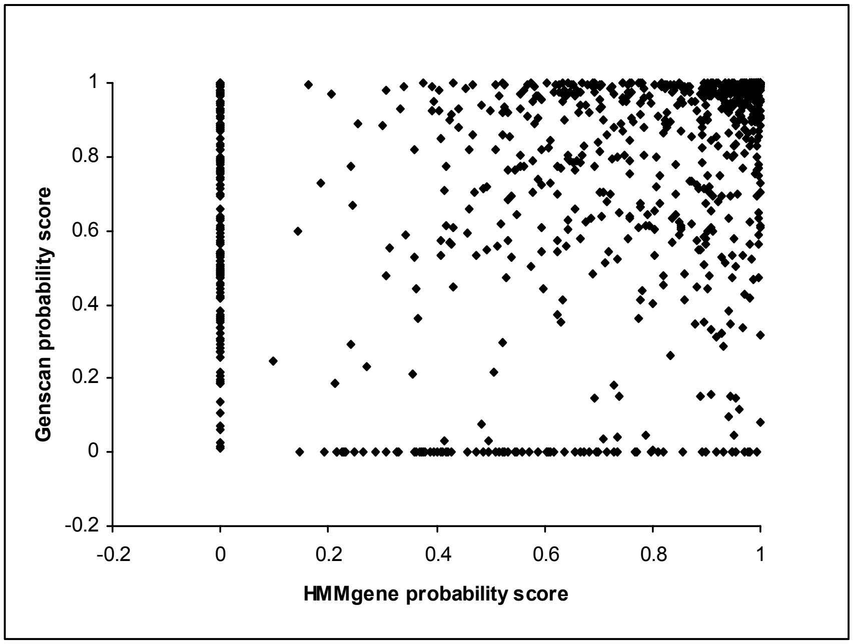

Figure 4: Probability scores of all the exons predicted by Genscan and HMMgene.

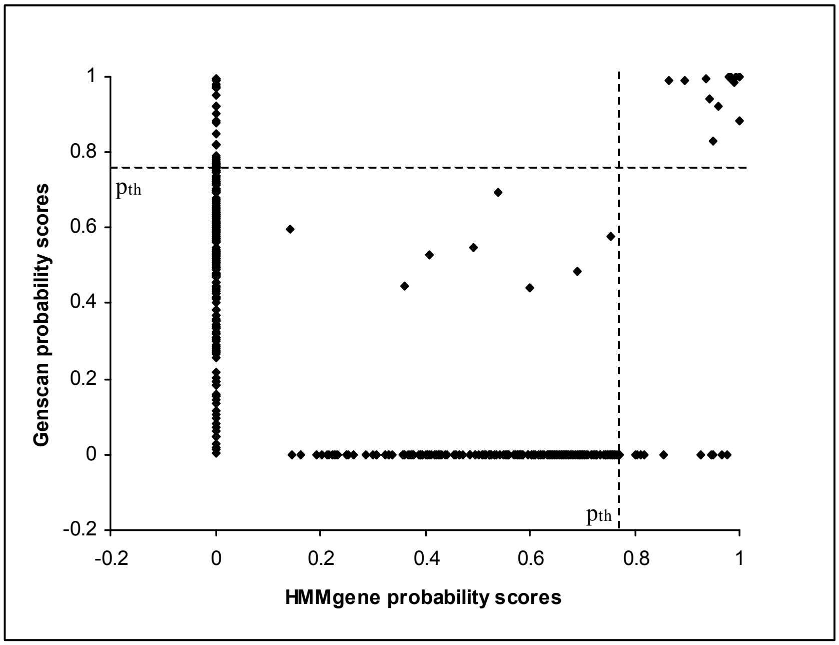

Figure 5: Probability scores of all the false positive exons predicted by Genscan and HMMgene.

The EUI method was primarily designed to improve prediction accuracy on the relatively short sequences containing only one gene, which resemble the sequences used for training of gene prediction programs. Genscan and HMMgene do rather well when predicting genes in these sequences: the majority of the actual exons is identified, at least partially, and the fraction of false positive exons is only around 5%. Although this results in fairly high accuracy measures at the nucleotide level, the exon level accuracy is affected by weakness of the signal detection, which often misses exact exon boundaries. In order to improve the prediction accuracy EUI attempts to correct exon boundaries using the union and intersection of exons; only a very small number of exons get discarded due to low exon scores. This approach gives more correctly predicted exons than any other method resulting in the highest exon sensitivity for each of the test datasets.

On the other hand, GI was designed for longer genomic sequences containing more than one gene, for which gene-finding programs generally make more false positive predictions. To reduce the high rate of wrong exons GI first chooses gene candidates to be those regions predicted as genes by both programs. In this algorithmic step many genes that are predicted by just one program and many exons that do not belong completely to the newly selected GI gene get eliminated. In the next step the EUI method is applied to the resulting GI genes. These two rounds of exon elimination get rid of many falsely predicted exons resulting in considerably higher specificity than both programs and the EUI method.

As can be inferred from the definition of the methods, EUI or GI exons that belong to the same gene are not guaranteed to be in the same reading frame. Frame consistency is lost when exon boundaries are changed by applying the EUI algorithm. In order to investigate the effect of frame consistency on EUI method we designed the EUI_frame method that uses the EUI algorithm to combine the predictions from Genscan and HMMgene, while maintaining a single reading frame. The program with the highest average exon score dictates the reading frame of the final prediction: exons whose boundaries are modified by the EUI method or high scoring ‘stand-alone’ exons (score $\geq p_{th}$) will be accepted in the final prediction only if they do not disrupt the chosen reading frame. Surprisingly, this method gave almost identical results to those of EUI on HMR195 and Burset/Guigo datasets. After the analysis of the results it was found that EUI_frame missed some of the exons that were correctly predicted by EUI and at the same time eliminated some of the wrong exons predicted by EUI. These differences were proportionally too small to change the overall prediction accuracy except for the slight decrease in the sensitivity. Being trained on sequences similar to those from HMR195 and Burset/Guigo datasets, the Genscan and HMMgene predictions on these two datasets are fairly accurate and similar: overlapping exons are usually in the same reading frame and there are not many false positive predictions that could disrupt the reading frame. This is why EUI and EUI_frame have almost identical results on the first two datasets. However, the *Adh* region sequence is much longer than any of the training sequences and contains a couple of hundred of genes, which presents a serious challenge for any gene-finding program. The Genscan and HMMgene prediction accuracy for this region is substantially weaker than for the other two datasets: while most of the coding nucleotides have been identified correctly in many cases exact exon boundaries are missed resulting in much lower exon sensitivity and specificity. The major problem is the huge number of the wrong exons, which outcomes in the drastic decrease in the specificity at both levels. These characteristics of the Genscan and HMMgene predictions resulted in many reading frame disruptions in the EUI genes and thus caused elimination of more than 400 of exons when EUI_frame was applied. Most of the dismissed exons were false positives, but a few of the 'true' exons were also sacrificed. The discrepancy between the EUI and EUI_frame results is notable: due to the twofold decrease in the number of wrong exons EUI_frame has substantially higher specificity at both levels than EUI, but at the same time sensitivity was decreased, especially at the nucleotide level owing to the exceptionally large size of exons missed by EUI_frame method.

By selecting more probable exon boundaries exon level accuracy is directly improved. This does not have to affect the nucleotide level accuracy significantly since the correct splice site could have been missed by just a couple of nucleotides and the correction will just slightly change *Sn*, *Sp* and *AC*. This explains why exon level accuracy is more improved than nucleotide level accuracy, as observed in Tables 9-11. Another phenomenon, observable for all three datasets, is that specificity is improved more than sensitivity at both levels. Since it is impossible for the EUI and GI methods to predict an exon that was initially missed by both programs, which would directly improve sensitivity of the prediction, the methods attempt to improve the accuracy of the predictions by correcting the exon boundaries and eliminating potentially wrong exons. The effect of this is that EUI and GI have approximately one half as many wrong exons as the individual programs, which primarily improves *Sp* and *ESp*.

Although Tables 9-11 show that the methods introduced have improved accuracy measures for all three datasets they were tested on, the level of improvement varies among them. The results on the Burset/Guigo dataset show the lowest increase in the accuracy measures. This dataset has been available since 1996 and it contained the vast majority of available vertebrate genomic sequences at the time it was assembled. It is realistic to assume that, in many cases, the training sets of gene-finding programs developed afterwards overlap with the Burset/Guigo dataset and this is probably the case with the training datasets of Genscan and HMMgene. This assumption is supported by the programs' high accuracy results on this dataset, shown in Table 10. Since the programs have been trained on at least a subset of Burset/Guigo dataset, their predictions are often correct and identical. Consequently, the combination of their predictions does not improve prediction accuracy as much as for the new HMR195 dataset.

The highest increase in prediction specificity is achieved on the *Adh* region. In this region the GI and EUI_frame methods have 21% higher specificity at nucleotide level, while at the exon level GI has 22.5% and EUI_frame 32.5% higher specificity when compared to the Genscan's accuracy results. This unusually high increase in specificity is a direct result of decreased number of false positive predictions. In long genomic sequences, such as the sequence of the *Adh* region, gene-finding programs make many false exon predictions, which lowers specificity at both levels. The effect of this shortcoming is also observable in our tables: the specificity values for Genscan and HMMgene at both levels are substantially lower for the *Adh* region than for the other two datasets. Each of the methods succeeded in eliminating many of the wrong exons predicted by Genscan and HMMgene, EUI_frame being the most successful by having approximately one quarter of the false positive exon predictions of HMMgene. However, this substantially increased specificity was also coupled with decreased sensitivity for the GI and EUI_frame methods. The decrease was marginal at the exon level since GI and EUI_frame had just a few correctly predicted exons less than

Genscan, but more substantial at the nucleotide level due to the unusually large size of the exons completely missed by the methods.

Since Genscan was used to build the st3 annotation set, it is obvious that the values in *Sp*, *ESp* and *WE* columns are not truly independent results of Genscan's and the methods' performance in the long genomic regions. Although only 40 of 222 annotated genes in st3 did not have any additional evidence except for strong Genscan and Genefinder prediction, it is very likely that the authors of st3 were also relying on Genscan's exon boundaries when other evidence were available. This can be inferred from the significantly higher *ESp* (and lower *WE*) for Genscan than for HMMgene, which cannot be observed for other datasets. However, our goal is to show the performance of the methods, rather than to give an independent evaluation of the programs on the *Adh* region and for that purpose the results in Table 11 are useful, showing that even though st3 was tailored using Genscan's predictions our methods have higher accuracy than Genscan.

3.6 Conclusion from the combination of predictions from gene-finding programs

We have developed three methods, EUI, GI and EUI_frame, for combining exon predictions from two gene-finding programs, Genscan and HMMgene, which successfully improve prediction accuracy, especially on long genomic sequences. The improvements have been obtained at both the nucleotide and exon levels and for all three datasets used for testing. The major advantage of the methods is the elimination of many false positive exon predictions, which directly improves the specificity at both levels.

While other sources of evidence, such as database or EST matches, are indispensable in the search for genes it is definitely worthwhile improving accuracy of *ab initio* gene prediction, which is essential when other evidence is not available. Our study demonstrates that the accuracy of gene-finders can be improved exploiting only currently available methods. Using the Genscan and HMMgene predictions as two partially independent sources of evidence and integrating them using variations of one basic approach the methods succeeded in correcting the exon boundaries, getting more exactly predicted exons, and in eliminating many false positive exons. The three methods presented have different strengths and are suitable for different purposes, depending whether sensitivity, specificity or reading frame consistency is the more valued characteristic of the predictions.

Chapter 4

Conclusion and future work

Separate conclusions for the evaluation of gene-finding programs and the combination of programs' predictions are given at the end of Chapter 2 and Chapter 3, respectively. Here, we will discuss the overall contributions of the thesis and give some directions for future work.

The evaluation of gene-finding programs given in this thesis is the first independent analysis of the new generation of programs. Since these programs are extensively used to analyze newly sequenced genomes and their predictions have a significant effect on interpretation of the genomic data, it is essential to have a realistic conception about their performance. The analysis of various aspect of the programs' prediction accuracy presented in Chapter 2 is comprehensive, unbiased and carefully carried out on a new independent dataset and as such could serve as a valuable reference to the gene-finding community.

The analysis of the programs' performance in relationship with some features of the input sequence as well as some features of the predicted genes pinpoints the strength and weaknesses of the individual programs analyzed and the gene-finding programs in general. Knowing the particular prediction features of the available programs might help users select the best program for a specific task. Also, identifying the drawbacks of the programs will make the users more aware of the potential mistakes that programs make that could lead to inaccurate interpretations and conclusions.

One of the important contributions of our research is the construction of the HMR195 dataset. There is a high demand for the datasets of this profile, not only for training and testing of the gene-finding programs, but also for other similar purposes (e.g., TSS and promoter recognition, splice site recognition) and very few datasets similar to HMR195 are available. The high quality of the dataset lies in its thoroughly filtered sequences, its non-redundancy and especially in its biologically validated annotation, which is a rare feature among other DNA datasets.

One of the conclusions drown from the analysis of gene-finding programs is that current programs, while significantly improved when compared to the programs of the older generation, are still not capable of autonomous gene discovery. Considering the huge amount of sequence data needed to be analyzed daily and that results of these gene identification analysis are basis for further lengthy and costly experiments it is clear that any improvements in gene prediction accuracy would have a significant practical importance. Although the methods presented in Chapter 3 are computationally straightforward and mostly intuitively and empirically derived their prediction accuracy is higher than for any single gene-finding program. Besides the practical value of the accuracy results, this approach could also serve as a basis or inspiration for development of a future generation of programs.

Overall, the contributions of this thesis are primarily of interest to biologist and are more practical than theoretical, though not without theoretical impact.

Future work

The methods developed in this thesis work well in practice; however, we did not offer much in terms of theoretical foundation. Therefore, one immediate extension of the work presented here would be to set up a theoretical framework that would explain the success of the methods in a more formal way. There is a substantial body of research on the combination of evidence from two or more experts (Spiegelhalter *et al*., 1990); this is one of the directions one could explore in search of a theoretical framework that would also allow the computation of reliable exon scores and might lead to further improvements in prediction accuracy.

Another direction one could pursue, based on the results of this thesis, is further development and improvement of the methods for combining the predictions from gene-finding programs. There are two possible approaches: improving the combination of *ab initio* gene-finders, either by modifying the methods presented in this thesis to include more than two programs, or elaborating on the existing algorithms by taking into account some additional characteristics of the exon predictions given by Genscan and HMMgene. The other obvious approach is to include other sources of evidence, such as similarity to a protein or EST sequence, or the presence of repeat elements in a query sequence. Recently, some gene-finding programs have been developed or upgraded to use prior knowledge about the sequence when searching for genes (HMMgene (Krogh, 1997), GeneScope (Burge, in preparation)), and it has been shown that this approach results in higher prediction accuracy. However, these integrated systems incorporate only one gene-finding program and our results presented in this work suggest that using additional programs would lead to further improvements.

It is our opinion that the newly assembled HMR195 dataset was the most suitable set of genomic sequences for the purposes of the gene-finding evaluation and analysis presented in Chapter 2. Nevertheless, the sequences in this dataset are not typical of sequences submitted to gene prediction algorithms for analysis: they are relatively short sequences containing only one complete gene, which is usually less complex than a typical mammalian gene (shorter, less exons), and flanked by unrealistically short stretches of intergenic regions. With the human genome sequencing project well underway, more realistic sequences that are well annotated (which is currently not the case) will become available and these should be used to reevaluate gene-finding programs and provide more credible results. Currently, a more realistic dataset could be obtained by also including sequences containing partial and multiple genes. The problems associated with the selection of these sequences are doubtful annotation (many partial genes are not in the final finished form) and increased difficulty of verification of the annotation.

Finally, as our analysis in Chapter 2 pointed out, the weakest component of current gene-finding programs is signal detection and attempting to improve signal sensors appears to be a fruitful research direction.

Appendix A

Implementation of the algorithms for combining Genscan and HMMgene predictions

All three algorithms presented in Section 3.3 are implemented in the Perl programming language. Each program first parses two input files, the Genscan and HMMgene output files for the same DNA sequence, obtaining the information about the exons predicted by the programs. For each exon, acceptor and donor site locations and exon type and score are stored in separate arrays. From this point the algorithms differ and they are described separately:

The EUI algorithm

Once the predicted exons are read from the input files they are divided into two groups: in the first group are the exons that have the probability score ≥ p_{th} and in the second group are the exons with the lower score. Here is the brief outline of the remainder of the algorithm:

For each Genscan and HMMgene exon from the first group:

- If Genscan's exon is the type "Initial" and has the same donor site as the HMMgene's exon, HMMgene's exon will be chosen for EUI exon

- If two exons overlap:
    - For the acceptor site of the EUI exon select the left boundary of the exon that is closer to the beginning of the input sequence
    - For the donor site of the EUI exon select the right boundary of the exon that is closer to the end of the input sequence
- All the exons from this group that do not overlap any other exon are accepted as EUI exons

For all the EUI exons found:

- If the two adjacent exons overlap, merge them (this can occur if, for example, Genscan's exon overlaps two HMMgene's exons)

For each Genscan and HMMgene exon from the second group:

- If two exons overlap:
    - For the acceptor site of the EUI exon select the left boundary of the exon that is closer to the end of the input sequence
    - For the donor site of the EUI exon select the right boundary of the exon that is closer to the beginning of the input sequence
- 'Stand-alone' exons from this group are not considered as EUI exons

Sort the EUI exons by their location in the sequence and print them out to an output file.

The GI algorithm

While reading the exons from the input files this algorithm also finds the locations of the genes predicted in the sequence.

For each Genscan and HMMgene gene:

- If two genes overlap:
  - For the beginning of the GI gene select the left boundary of the gene that is closer to the end of the input sequence
  - For the end of the GI gene select the right boundary of the gene that is closer to the beginning of the input sequence
- 'Stand-alone' genes that do not overlap any other gene predicted by the other program are not considered as GI genes

The exons that completely belong to one of the GI genes are divided in two groups according to their scores as described in EUI algorithm. From this point on the algorithm continues the same as the EUI algorithm.

The EUI_frame algorithm

Similarly to the first part of the GI algorithm, the EUI_frame algorithm also computes the locations of the genes predicted in the input sequence and to each gene assigns a gene probability as the average of the scores of the exons belonging to that gene.

The next step in the algorithm is to determine the acceptor and donor site positions in a reading frame for each exon predicted. Position in the reading frame can be 0, 1 or 2 depending if a splice site is right between the codons, after the first nucleotide in the codon or after the second nucleotide in the codon. The reading frame position of the acceptor site of an exon (excluding the initial one) is equal to the reading frame position of the donor site of the previous exon (see Figure 6).

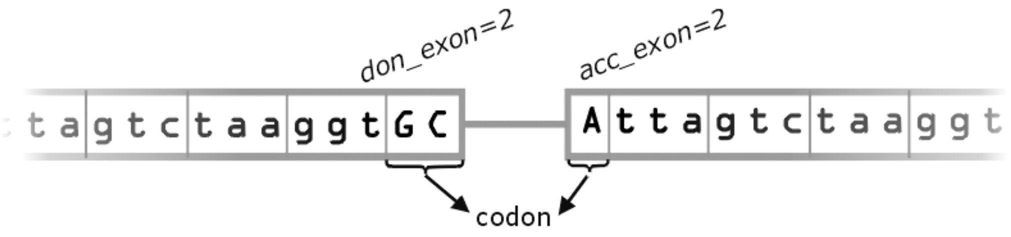

Figure 6: Determining values for *acc_exon* and *don_exon*.

In the following pseudo-algorithm for determining reading frame position variables *acc_frame* and *don_frame* denote the position of the acceptor and donor site in the reading frame, respectively, and *beg_exon* and *end_exon* for the location of the exon acceptor and donor site, respectively.

```
For each Genscan and HMMgene gene:
    • If the first exon belonging to the gene is the type “Initial”
        o For the first exon
            ▪ $acc\_frame = 0$
            ▪ $don\_frame = (end\_exon - beg\_exon + 1) \% 3$
        o for each following exon until the end of the gene
            ▪ $acc\_frame = don\_frame$(of previous exon)
            ▪ $don\_frame = (end\_exon - beg\_exon + 1 - (3 - acc\_frame)) \% 3$
    • If the first exon belonging to the gene is the type “Single”
        o $acc\_frame = 0$
        o $don\_frame = 0$
    • If gene’s first exon is not the type “Initial”, but its last exon is the type
      “Terminal”
        o for the last exon
            ▪ $don\_frame = 0$
            ▪ $acc\_frame = (end\_exon - beg\_exon + 1) \% 3$
        o for each preceding exon until the beginning of the gene
```

        - $don\_frame = acc\_frame$(of previous exon)
        - $acc\_frame = (end\_exon - beg\_exon + 1 - don\_frame) \% 3$
- If all the exons in the gene are the type "Internal"
    - Find the exon predicted by the other program that has the same exon boundaries as any of the "Internal" exons
    - Copy *acc_frame* and *don_frame* from that exon and then proceed 'left' and 'right' to the first and the last exon, computing the values for *acc_frame* and *don_frame* as previously described

Divide the predicted exons into two groups according to their scores.

For each Genscan and HMMgene gene:

- If two genes overlap:
    - Select the one with the higher gene probability to dictate the reading frame. All the exons belonging to that gene with the score ≥ p_{th} are initially accepted as EUI exons (but some of their boundaries might change if they overlap with the exons predicted by the other program)
    - If two exons overlap:
        - change EUI exon's left boundary only if the left boundary of the exon belonging to the lower probability gene is closer to the beginning of the input sequence and

          $$(beg\_exon\_EUI - beg\_exon) \% 3 = 0$$
        - change EUI exon's right boundary only if the right boundary of the exon belonging to the lower probability gene is closer to the end of the input sequence and

          $$(end\_exon - end\_exon\_EUI) \% 3 = 0$$

Sort the EUI exons by their location in the input sequence.

All the exons belonging to the lower probability gene that have score ≥ p_{th} and do not overlap any of the EUI exons are stored as EUI_cand exons. Also, all the intersections of the exons from the second group are stored as EUI_cand exons.

Sort the EUI_cand exons by their location.

- Select all the EUI_cand exons that are located before the first EUI exon:
    - Going from the last (closes to the first EUI exon) to the first exon
        - The first exon that has $\mathit{don\_exon} = \mathit{acc\_exon\_EUI}$ is accepted
        - The exon preceding the accepted exon is accepted if

          $\mathit{don\_exon} = \mathit{acc\_exon}$(accepted exon)

- Select all the EUI_cand exons that are located after the last EUI exon:
    - Going from the first (closest to the last EUI exon) to the last exon
        - The first exon that has $\mathit{acc\_exon} = \mathit{don\_exon\_EUI}$ is accepted
        - The exon following the accepted exon is accepted if

          $\mathit{acc\_exon} = \mathit{don\_exon}$(accepted exon)
- For each pair of the adjacent EUI exons select all the EUI_cand exons that are located between the two EUI exons:
    - Going from the first (closest to the 'left' EUI exon) to the last exon
        - The first exon that has $\mathit{acc\_exon} = \mathit{don\_exon\_EUI}$ is stored
        - The exon following the stored exon is also stored if

          $\mathit{don\_exon} = \mathit{acc\_exon}$(accepted exon)
    - If the last stored exon has $\mathit{don\_exon} = \mathit{acc\_exon\_EUI}$ then all the stored exons are accepted as EUI exons, else go 'backwards' to the first stored exon until $\mathit{don\_exon} = \mathit{acc\_exon\_EUI}$ and then accept all the stored exon up to that one

Sort the EUI exons by their location in the sequence and print them out to an output file.

Appendix B

List of sequences in HMR195

Accession	**Definition line**
AB016625	Homo sapiens OCTN2 gene
AF008216	Homo sapiens candidate tumor suppressor pp32r1 (PP32R1) gene
AF092047	Homo sapiens homeobox protein Six3 (SIX3) gene
AF096303	Homo sapiens putative sterol reductase SR-1 (TM7SF2) gene
AF019563	Homo sapiens DAP12 gene
AB012922	Homo sapiens MTA1-L1 gene
U25134	Human carbonic anhydrase V (CA-V) gene
U17081	Human fatty acid binding protein (FABP3) gene
AB021866	Homo sapiens KIP gene
AF039704	Homo sapiens lysosomal pepstatin insensitive protease (CLN2) gene
AF082802	Homo sapiens telencephalin (ICAM5) gene
AF039954	Homo sapiens CC chemokine LCC-1 precursor gene
AB018249	Homo sapiens gene for CC chemokine LEC
AF099731	Homo sapiens connexin 31.1 (GJB5) gene
AF099730	Homo sapiens connexin 31 (GJB3) gene
AF039401	Homo sapiens calcium-dependent chloride channel-1 (hCLCA1) gene
AF084941	Homo sapiens pre-T cell receptor alpha chain 1 precursor gene
AF059675	Homo sapiens putative RNA helicase Ski2w (SKI2W) gene
AF007189	Homo sapiens claudin 3 (CLDN3) gene

AF016898	Homo sapiens B-ATF gene
AF076214	Homo sapiens prophet of Pit1 (PROP1) gene
AB012113	Homo sapiens gene for CC chemokine PARC precursor
AB019534	Homo sapiens gene for cathepsin L2
AF080237	Homo sapiens Rho GDP-dissociation inhibitor gamma (ARHGDIG) gene
AF071596	Homo sapiens apoptosis inhibitor (IEX-1L) gene
U43842	Homo sapiens bone morphogenetic protein-4 (hBMP-4) gene
AF053455	Homo sapiens tetraspan TM4SF (TSPAN-5) gene
AF071216	Homo sapiens beta defensin 2 (HBD2) gene
AF058761	Homo sapiens ribosomal protein S12 gene
Y16791	Homo sapiens hHa5 gene
AB016492	Homo sapiens hJTB gene
AF001689	Homo sapiens ribosomal protein L23A (RPL23A) gene
AF029081	Homo sapiens 14-3-3 sigma protein promoter and gene
U96846	Human natural killer protein group 2-F (NKG2-F) gene
AF027152	Homo sapiens P450 25-hydroxyvitamin D-1 alpha hydroxylase
U55058	Human uroguanylin gene
AF068624	Homo sapiens 5-aminolevulinate synthase 2 (ALAS2) gene
AF053069	Homo sapiens NADH:ubiquinone dehydrogenase 51 kDa subunit (NDUFV1)
AF032437	Homo sapiens mitogen activated protein kinase activated protein
AF007876	Homo sapiens Na,K-ATPase beta 2 subunit gene
AF051160	Homo sapiens tyrosine phosphatase (PRL-1) gene
AF022382	Homo sapiens UDP-galactose 4' epimerase (GALE) gene
AF045999	Homo sapiens rod cGMP phosphodiesterase delta subunit (PDEd) gene
U53447	Homo sapiens PAPS synthase gene
AF009356	Homo sapiens regulator of G-protein signaling-16 (RGS16) gene
AF019409	Homo sapiens uncoupling protein 2 (UCP2) gene
AF015224	Homo sapiens mammaglobin gene
AF042782	Homo sapiens galanin receptor type 2 (GALR2) gene
AF037207	Homo sapiens persyn gene

AB016243	Homo sapiens gene for regulatory factor 2 of sodium/hydrogen
AF049259	Homo sapiens keratin 13 gene
AB012668	Homo sapiens hFuc-T VII gene for selectin-ligand synthase
AF052572	Homo sapiens chemokine receptor CXCR4 gene
AF042001	Homo sapiens zinc finger protein slug (SLUG) gene
AF055475	Homo sapiens GAGE-8 gene
AF055903	Homo sapiens cathepsin W gene
AF053630	Homo sapiens monocyte/neutrophil elastase inhibitor gene
AF027148	Homo sapiens myogenic determining factor 3 (MYOD1) gene
AB007828	Homo sapiens gene for necdin
AF044311	Homo sapiens gamma-synuclein gene
AF061327	Homo sapiens cyclin-dependent kinase 4 inhibitor D p19 gene
AF059650	Homo sapiens histone deacetylase 3 (HDAC3) gene
AB010874	Homo sapiens gene for ribosomal protein L41
AF071552	Homo sapiens N-acetyltransferase-1 (NAT1) gene, NAT1*26A allele
AF059734	Homo sapiens homeodomain transcription factor (HESX1) gene
AF009962	Homo sapiens CC-chemokine receptor (CCR-5) gene, delta-32 allele
AB013139	Homo sapiens gene for NBS1
AF013711	Homo sapiens 22 kDa actin-binding protein (SM22) gene
AF065396	Homo sapiens retinoic X receptor B gene
AF058762	Homo sapiens galanin receptor subtype 2 (GALNR2) gene
AF043105	Homo sapiens glutathione S-transferase mu 3 (GSTM3) gene
AF065988	Homo sapiens keratocan gene
AF026564	Homo sapiens RNA binding protein II (RBMII) gene
AF037438	Homo sapiens short chain L-3-hydroxyacyl-CoA dehydrogenase (SCHAD)
AF028233	Homo sapiens distal-less homeobox protein (DLX3) gene
AB007546	Homo sapiens gene for LECT2
AF058293	Homo sapiens D-dopachrome tautomerase gene
AF055080	Homo sapiens winged-helix transcription factor forkhead 5 gene
AF037062	Homo sapiens retinol dehydrogenase gene

AB009589	Homo sapiens gene for Osteomodulin
AF047383	Homo sapiens uroporphyrinogen decarboxylase (UROD) gene
U31468	Homo sapiens homeobox protein (GBX2) gene
AF036329	Homo sapiens gonadotropin-releasing hormone precursor
AF042084	Homo sapiens heparan glucosaminyl
AF040714	Homo sapiens homeobox protein A10 (HOXA10) gene
AF039307	Homo sapiens homeobox A11 (HOXA11) gene
AF031237	Homo sapiens CC chemokine receptor 5 (CCR5) gene
AF005058	Homo sapiens chemokine receptor (CXCR-4) gene
AF037372	Homo sapiens cytochrome oxidase subunit VIIa-H precursor (COX7AH)
D89060	Homo sapiens DNA for oligosaccharyltransferase
AF032455	Homo sapiens aldose reductase gene
AB003730	Homo sapiens gene for polyubiquitin
AB006987	Homo sapiens gene for 25-hydroxyvitamin D3 1-alpha-hydroxylase
AF015812	Homo sapiens RNA helicase p68 (HUMP68) gene
AF015954	Homo sapiens lymphopain gene
D67013	Homo sapiens DNA for alpha2-HS glycoprotein (AHSG)
D38752	Homo sapiens gene for fibroblast growth factor-8
D83956	Homo sapiens HLA-B gene (HLA-B*0801 allele)
AF016052	Homo sapiens zinc finger protein ZNF191 (ZNF191) gene
AB002059	Homo sapiens DNA for Human P2XM
AJ223321	Homo sapiens RP58 gene
U76254	Human neuropeptide Y receptor type 2 gene
AF017115	Homo sapiens cytochrome c oxidase subunit IV precursor (COX4) gene
AF016190	Mus musculus connexin-36 (Cx36) gene
U40494	Mus musculus augmenter of liver regeneration (Alr) gene
AF057156	Mus musculus small proline-rich protein 1A (Sprr1a) gene
AF006668	Mus musculus uroguanylin gene
AF002719	Mus musculus secretory leukoprotease inhibitor gene
AF019045	Mus musculus vesicular acetylcholine transporter gene

AF074912	Mus musculus chemokine receptor CX3CR1 gene
U83462	Mus musculus serotonin N-acetyltransferase (AANAT) gene
AF077860	Mus musculus helix-loop-helix protein Id2 gene
AF043289	Mus musculus muscle-specific serine kinase 1 gene
AF037313	Mus musculus potassium inward rectifier 6.2 (Kir6.2) gene
AF035672	Mus musculus MHC class I related protein 1 (MR1) gene
AF001797	Mus musculus glucosidase I gene
U44436	Mus musculus bradykinin B1 receptor gene
U50355	Rattus norvegicus neutrophil defensin 4 (RatNP-4) gene
U50353	Rattus norvegicus defensin 3a (RatNP-3a) gene
AB017361	Mus musculus Kip/Cip gene
AF059211	Mus musculus cholesterol 25-hydroxylase gene
AF069778	Mus musculus A3 adenosine receptor gene
AF035684	Mus musculus beta chemokine TCA4 gene
AF024513	Mus musculus menin (Men1) gene
AB015637	Rattus norvegicus gene for alpha(1,2) fucosyltransferase
AF045662	Mus musculus cell cycle checkpoint control protein Mrad9 gene
U96809	Mus musculus chromatin structural protein homolog (Supt4h) gene
AF093853	Mus musculus 1-Cys peroxiredoxin protein 2 gene
AF092536	Mus musculus heat shock protein hsp40-3 gene
AF074856	Mus musculus C1q/MBL/SPA receptor C1qRp (C1qrp) gene
AF039602	Mus musculus extracellular superoxide dismutase (SOD3) gene
AF057301	Mus musculus keratocan (Ktcn) gene
AF029791	Mus musculus UDP-Gal:betaGlcNAc beta 1,3-galactosyltranferase-II
AF026469	Mus musculus ubiquitin-specific protease (Unp) gene
AF031426	Mus musculus small unique nuclear receptor co-repressor (SUN-CoR)
U84903	Mus musculus L23 mitochondrial-related protein (L23mrp) gene
AF016697	Mus musculus chemokine receptor gene
AF022651	Mus musculus macrosialin gene
AF031826	Mus musculus leukocystatin gene

AF053757	Mus musculus complement C3a anaphylatoxin receptor (C3ar) gene
AF015881	Mus musculus nuclear factor erythroid-related factor 1 (Nrf1) gene
AF029875	Rattus norvegicus muscle carnitine palmitoyltransferase I (CPTI)
AF046000	Mus musculus rod cGMP phosphodiesterase delta subunit (Pde6d) gene
AF024524	Mus musculus LIM domain binding protein 1 (Ldb1) gene
U96626	Mus musculus chondroadherin gene
AF042784	Mus musculus galanin receptor type 2 (GalR2) gene
AF042783	Mus musculus galanin receptor type 3 (GalR3) gene
AF078705	Mus musculus vascular adhesion protein-1 gene
AF034569	Mus musculus anticoagulant protein C gene
AF064081	Mus musculus alpha-sarcoglycan gene
AB012159	Mus musculus gene for uncoupling protein-2
AF068199	Mus musculus D-dopachrome tautomerase gene
AF079877	Mus musculus cyclin G2 (Ccng2) gene
AF030522	Mus musculus stannin gene
AF079528	Mus musculus IER5 (Ier5) gene
AF003255	Mus musculus alpha-N-acetylglucosaminidase gene
AF029240	Rattus norvegicus MHC class Ib RT1.S3 (RT1.S3) gene
AF033620	Mus musculus platelet endothelial tetraspan antigen-3 (Peta3) gene
AB015652	Mus musculus gene for DJ-1
U82792	Mus musculus allograft inflammatory factor-1 gene
AF009614	Mus musculus homeobox containing nuclear transcriptional factor
AB007139	Mus musculus Psme3 gene for PA28 gamma subunit
AB010149	Mus musculus gene for PACAP ligand precursor
AB009967	Mus musculus gene for HES2
AF060229	Mus musculus homeobox protein MSX3 (Msx3) gene
AF006203	Rattus norvegicus insulin-like growth factor binding protein
AF052695	Rattus norvegicus cell cycle protein p55CDC gene
AF010405	Mus musculus fork head transcription factor (Hfh-1L) gene
AB011595	Mus musculus gene for eIF4A

AF027181	Rattus norvegicus neurabin II gene
AF007558	Mus musculus hemochromatosis (HFE) gene
AB003306	Mus musculus DNA for PSMB5
U77350	Rattus norvegicus chemokine receptor CCR5 gene
U77349	Rattus norvegicus chemokine receptor CCR2 gene
U29186	Mus musculus short incubation prion protein Prnpa gene
U93050	Mus musculus poly(A) binding protein II (mPABII) gene
U47815	Mus musculus stanniocalcin gene
AF024570	Mus musculus DNA polymerase delta catalytic subunit (pold1) gene
AF030199	Mus musculus type 1 sigma receptor gene
AB001735	Mus musculus DNA for ADAMTS-1
AB010281	Mus musculus gene for neuromedin B receptor
U70368	Mus musculus hematopoietic-specific IL-2 deubiquitinating enzyme
AB009694	Mus musculus gene for mafF
AB009693	Mus musculus gene for mafG
AF025818	Mus musculus A/J fibrinogen-like protein (fgl2) gene
AF035680	Mus musculus cathelicidin (Cramp) gene
AF019074	Mus musculus erythroid Kruppel-like factor (EKLF) gene
AF013262	Mus musculus lumican (Ldc) gene
AF022948	Mus musculus neuropeptide Y Y5 receptor gene
AF007900	Mus musculus fetuin (Ahsg) gene
D85562	Mus musculus DNA for proteasome subunit MECL1
AF012244	Mus musculus cerberus-like (Cer-l) gene
AF017128	Mus musculus fos-related antigen 1 (fra-1) gene
D89572	Mus musculus gene for ryudocan core protein
U89486	Mus musculus agouti-related protein (Agrp) gene

Appendix C

Sample Genscan and HMMgene output files

Genscan output file

```
GENSCAN 1.0 Date run: 13-Jan-100    Time: 11:48:03

Sequence AF096303 : 5744 bp : 60.03% C+G : Isochore 4 (57 - 100 C+G%)

Parameter matrix: H

Predicted genes/exons:

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

 1.01 Term +    534    794  261  2  0  113   48   560 0.992  51.03
 1.02 PlyA +    894    899    6                                1.05

 2.00 Prom +   1043   1082   40                               -5.95
 2.01 Init +   1703   2069  367  1  1   86    3   446 0.147  31.33
 2.02 Intr +   2423   2617  195  0  0   39   94   137 0.146  10.00
 2.03 Intr +   2694   2797  104  1  2   77   71   168 0.999  15.06
 2.04 Intr +   3938   4057  120  1  0  100   78   160 0.623  18.06
 2.05 Intr +   4137   4305  169  2  1  112   61   232 0.999  23.62
 2.06 Intr +   4538   4618   81  0  0   75   93    85 0.997   8.46
 2.07 Intr +   4700   4822  123  0  0  113  101     6 0.974   6.13
 2.08 Term +   5118   5278  161  1  2   92   55   366 0.999  32.76
 2.09 PlyA +   5563   5568    6                                1.05

Suboptimal exons with probability > 0.100

Exnum Type S .Begin ...End .Len Fr Ph B/Ac Do/T CodRg P.... Tscr..
----- ---- - ------ ------ ---- -- -- ---- ---- ----- ----- ------

S.001 Sngl +   1703   2077  375  1  0   86   50   452 0.819  35.26
S.002 Init +   2467   2617  151  0  1   35   94   227 0.813  16.34
S.003 Intr +   3938   4078  141  1  0  100   59   173 0.376  17.33
```

HMMgene output file

```
## gff-version 1
## date: Thu Jan 13 12:33:27 2000
## HMMgene1.1d (human model sim10gc.D.bsmod)

# SEQ: AF096303 5744 (+) A:1072 C:1656 G:1792 T:1224
AF096303  HMMgene1.1d   exon_1   534   762   0.33   +   1   bestparse:cds_1
AF096303  HMMgene1.1d   exon_2   1719  2069  0.694  +   1   bestparse:cds_1
AF096303  HMMgene1.1d   exon_3   2423  2617  0.955  +   1   bestparse:cds_1
AF096303  HMMgene1.1d   exon_4   2694  2797  0.996  +   0   bestparse:cds_1
AF096303  HMMgene1.1d   exon_5   3938  4057  0.849  +   0   bestparse:cds_1
AF096303  HMMgene1.1d   exon_6   4137  4305  0.933  +   1   bestparse:cds_1
AF096303  HMMgene1.1d   exon_7   4700  4822  0.977  +   1   bestparse:cds_1
AF096303  HMMgene1.1d   lastex   5118  5278  1.001  +   0   bestparse:cds_1
AF096303  HMMgene1.1d   CDS      1     5278  0.264  +   .   bestparse:cds_1
# SEQ: AF096303 5744 (-) A:1224 C:1792 G:1656 T:1072
AF096303  HMMgene1.1d   firstex  3996  4204  0.359  -   2   bestparse:cds_1
AF096303  HMMgene1.1d   lastex   2470  2602  0.490  -   0   bestparse:cds_1
AF096303  HMMgene1.1d   CDS      2470  4204  0.221  -   .   bestparse:cds_1
```

The columns in the Genscan's output, from left to right, are: the gene and exon number of each predicted exon, the type of exon or signal, the DNA strand of the predicted feature, the beginning and ending position of the predicted feature, the length of the predicted feature, the "absolute reading frame" of the predicted exon" (a codon ending at position *x* in the reading sequence has reading frame *x* mod 3), the "net phase" of the predicted exon (exon length modulo three), the acceptor and donor site scores, the coding score, the exon probability and the exon score.

The columns in the HMMgene's output, from left to right, are: sequence identifier, the version of the program used for prediction, the type of exon or signal, the beginning and ending position of the predicted feature, probabilistic exon score, the DNA strand of the predicted feature, the reading frame (for exons it is the position of the donor in the frame), group (gene) to which prediction belong.

Appendix D

Glossary

aa (amino acid) - organic compounds that generally contain an amino and a carboxyl group. Twenty amino acids are the subunits, which are polymerised to form proteins.

acceptor site - a site in pre-mRNA that corresponds to the 3'-end of the intron and the 5'-end of the next exon

ACR (ancient conserved regions) - regions of protein sequences showing highly significant similarity across phyla

alternative splicing - the process by which different mRNAs are produced from the same primary transcript, through variations in the splicing pattern of the transcript

bp (base pair) - in a nucleic acid double helix, a purine and a pyrimidine on different strands that interact by hydrogen bonding, most commonly a GC or AT pair.

cDNA - synthetic DNA transcribed from a specific RNA trough the reaction of a specific enzyme

donor site - the site on pre-mRNA that corresponds to the 3'-end of an exon and the 5'-end of an intron

enhancer - a regulatory sequence that can elevate levels of transcription from adjacent promoter

EST (expression sequence tag) - a partial coding sequence isolated at random from a cDNA library; used for identification and mapping of coding sequences, for discovery of new genes and (by reference to sequence data banks) for discovery of identities with other genes

eukaryotes - one of the three domains of the living organisms that are characterized with cells containing nucleus and cellular membranes. The other two domains are eubacteria and archaea.

exon - from *expressed region*, i.e. a region of a eukaryotic gene that encodes a sequence of amino acids

frame-shift mutation - the insertion or deletion of nucleotide pair or pairs, causing a disruption of the translational reading frame

genome - the entire complement of genetic material in a chromosome set

homology - the similarity in base sequences of genes or amino acid sequences of proteins that denotes a common evolutionary origin; also the similarity of structure or function of proteins that is due to a common evolutionary origin

isochores - genomic regions whose base composition is locally homogenuous, but varies significantly between disjoint regions

intron - from *intervening sequence region*; defined as a non-coding polynucleotide sequence that interrupts the coding sequences, the exons, of a gene. This segment is initially transcribed, but the transcript is not found in the functional mRNA.

kb (kilo bases) - 1,000 base pairs

mRNA (messenger RNA) - an RNA molecule transcribed from DNA of a gene, and from which a protein is translated by the action of ribosomes

ORF (open reading frame) - a section of a sequenced piece of DNA that begins with a start codon and ends with a stop codon

phylum **[**plural = **phyla] -** a subdivision of a kingdom encompassing all forms of life with the same distinctive body plan

polyA site - a site where polyadenine sequence is attached during mRNA processing

promoter - a regulatory region a short distance from the 5' end of a gene that acts as a binding site for transcription protein

pseudogene - a DNA sequence that is homologous to a structural gene, but cannot be expressed because it has no continuous open reading frame

ribosomal binding site - a nucleotide sequence proximal to the translation initiation codon (ATG). It is thought to be involved in initiation of translation by helping the mRNA bind to the ribosome.

transcription - a synthesis of RNA using DNA template

transcription start site (TSS) - the position in a gene where the mRNA synthesis start. The first nucleotide transcribed is denoted +1.

translation - the synthesis of a protein directed by mRNA

silencer - a DNA sequence which acts in the opposite direction of an enhancer to inhibit the transcription of a gene

splice site - acceptor or donor splice site

untranslated region (UTR) - a genomic DNA sequence that is not translated into an RNA sequence

vertebrates - animals with a backbone

Bibliography

Altschul, S.F., Gish, W., Miller, W., Myers, E.W., and Lipman, D.J. 1990. Basic local alignment search tool. *J. Mol. Biol.* **215:** 403-410.

Altschul, S.F., Madden, T.L., Schaffer, A.A., Zhang, J, Zhang, Z., Miller, W., and Lipman, D.J. 1997. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res*. **25(17):** 3389-3402.

Ashburner, M. 2000. A biologist's view of the *Drosophila* genome annotation assessment. *Genome Res.* **10:** 391-393.

Ashburner, M., Mishra, S., Roote, J., Lewis, S.E., Blazej, R., Davis, T., Doyle, C., Galle, R., George, R., Harris, N. et al. 1999. An exploration of the sequence of a 2.9 Mb region of the genome of drosophila melanogaster. The adh region. *Genetics* **153:**179-219.

Benson, D.A., Karsch-Mizrachi, I., Lipman, D.J., Ostell, J., Rapp, B.A., and Wheeler, D.L. 2000. GenBank. *Nucleic Acids Res.* **28(1):** 15-18.

Bernardi, G. 1993. The isochore organization of the human genome and its evolutionary history - a review. *Gene* **135:** 57-66.

Burge, C. and Karlin, S. 1997. Prediction of complete gene structure in human genomic DNA. *J. Mol. Biol.* **268:** 78-94.

Burge, C. 1997. Identification of complete gene structure in human genomic DNA. PhD thesis. Stanford University, Stanford, CA.

Burset, M. and Guigo, R. 1996. Evaluation of gene structure prediction programs. *Genomics* **34:** 353-367.

Claverie, J.-M. 1997.Computational methods for the identification of genes in vertebrate genomic sequences. *Hum*. *Mol*. *Genet*. **6(10):** 1735-1744.

Cooper, P.R., Smilinich, N.J., Day, C.D., Nowak, N.J., Reid, L.H., Pearsall, R.S., Reece, M., Prawitt, D., Landers, J., Housman, D.E., Winterpacht, A., Zabel, B.U., Pelletier, J., Weissman, B.E., Shows, T.B., and Higgins, M.J. 1998. Divergently transcribed overlapping genes expressed in liver and kidney and located in the 11p15.5 imprinted domain. *Genomics* **49:**38-51.

Dominski, Z. and Kole, R. 1991. Selection of splice sites in pre-mRNA with short internal exons. *Mol. Cell Biol.* **11(12):** 6075-6083.

Dong, S. and Searls, D.B. 1994. Gene structure prediction by linguistics methods. *Genomics* **23:** 540-551.

Duret, L., Mouchiroud, D., and Gautier, C. 1995. Statistical analysis of vertebrate sequences reveals that long genes are scarce in GC-rich isochores. *J. Mol. Evol.* **40:** 308-317.

Dunham, I., Shimizu, N., Roe, B.A., Chissoe, S., *et al*. 1999. The DNA sequence of human chromosome 22. *Nature* **402:** 489-495.

Ficket, W.F. and Tung, C.-S. 1992. Assesment of protein coding measures. *Nucleic Acids Res*. **20(24):** 6441-6450.

Fickett, J.W. 1996. The gene identification problem: an overview for developers. *Computer Chem.* **20(1):** 103-118.

Florea, L., Hartzell, G., Zhang, Z., Rubin, G.M., and Miller, W. 1998. A computer program for aligning a cDNA sequence with a genomic DNA sequence. *Genome Res.* **8(9):** 967-974.

Gelfand, M.S., Mironov, A.A., and Pevzner, P.A. 1996. Gene recognition via spliced sequence alignment. *Proc. Natl. Acad. Sci.* **93(17):** 9061-9066.

Green, P., Lipman, D., Hillier, L., Waterston, R., States, D., and Claverie, J.M. 1993. Ancient conserved regions in new gene sequences and the protein databases. *Science* **259:** 1711-1716.

Griffiths, A.J.F., Miller, J.H., Suzuki, D.T., Lewontin, R.C., and Gelbart, W.M. 1996*. An introduction to genetic analysis*, 6th edition. W. H. Freeman and Company, New York.

Guigo, R., Knudsen, S., Drake, N., and Smith, T.F. 1992. Prediction of gene structure. *J. Mol. Biol.* **226:** 141-157.

Guigo, R. and Fickett, J.W. 1995. Distinctive sequence features in protein coding, genic non-coding, and intergenic human DNA. *J. Mol. Biol.* **253:** 51-60.

Haussler, D. 1998. Computational Genefinding. *Trends Biochem Sci*, Supplementary Guide to Bioinformatics pp. 12-15.

Hawkins, J.D. 1988. A survey on intron and exon lengths. *Nucleic Acids Res.* **16**: pp. 9893-9908.

Huang, X., Adams, M.D., Zhou, H., and Kerlavage, A.R. 1997. A tool for analyzing and annotating genomic sequences. *Genomics* **46:** 37-45.

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

Krogh, A. 1997. Two methods for improving performance of an HMM and their application for gene-finding. In *Proceedings of the Fifth International Conference on Intelligent Systems for Molecular Biology* (ed. Gaasterland, T., Karp, P., Karplus, K., Ouzounis, C., Sander, C., and Valencia, A.), pp. 179-186. AAAI Press, Menlo Park, CA.

Krogh, A. 2000.Using database matches with HMMgene for automated gene detection in *Drosophila*. *Genome Res.* **10:** 523-528.

Kulp, D., Haussler, D., Reese, M.G., and 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* (ed. States, D., Agarwal, P., Gaasterland, T., Hunter, L, and Smith, R.), pp. 134-142. AAAI Press, Menlo Park, CA.

Lopez, R., Larsen, F. and Prydz, H. 1994. Evaluation of the exon predictions of the GRAIL software. *Genomics* **24:** 133-136.

Lukashin, A.V. and Borodovsky, M. 1998. GeneMark.hmm: new solutions for gene-finding. *Nucleic Acids Res.* **26:** 1107-1115.

Mironov,A.A., Ficket, J.W., and Gelfand, M.S. 1999. Frequent Alternative splicing of human genome. *Genome Res.* **9:** 1288-1293.

Murakami, K. and Tagaki, T. 1998. Gene recognition by combination of several gene-finding programs. *Bioinformatics* **14(8):** 665-675.

Oliver, S.G., van der Aart, Q. J., Agostoni-Carbone, M.L., Aigle, M., Alberghina, L., Alexandraki D., Antoine, G., Anwar R., Ballesta, J.P., Benit, P., *et al.* 1992. The complete DNA sequence of yeast chromosome III. *Nature* **357:** 38-46.

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

Reese, M.G., Hartzell, G., Harris, N.L., Ohler, U., Abril, J.F., and Lewis, S.E. 2000. Genome annotation assessment in drosophila melanogaster. *Genome Res*. **10:** 483-501.

Salamov, A.A. and Solovyev, V.V. 1997. Recognition of 3'-processing sites of human mRNA precursor. *CABIOS* **13(1):** 23-28.

Salzberg, S., Delcher, A., Fasman, K., and Henderson, J. 1998. A decision tree system for finding genes in DNA. *J. Comp. Biol.* **5(4):** 667-680.

Schuler, G.D., Epstein, J.A., Ohkawa, H., and Kans, J.A. 1996. Entrez: molecular biology database and retrieval system. *Methods Enzymol.* **266**:141-162.

Schulz, R.A. and Butler, B.A. 1989. Overlapping genes of Drosophila melanogaster: organization of the z600-gonadal-Eip28/29 gene cluster. *Genes Dev.* **3(2):** 232 –242.

Snyder, E.E. and 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. 1995. Identification of human gene structure using linear discriminant functions and dynamic programming. *In Proceedings of the Third International Conference on Intelligent Systems for Molecular Biology* (ed.Rawling, C., Clark, D., Altman, R., Hunter, L., Lengauer, T., and Wodak, S.), pp. 367-375. AAAI Press, Menlo Park, CA.

Solovyev, V.V., Salamov, A.A., Lawrence, C.B. 1995. Identification of human gene structure using linear discriminant functions and dynamic programming. *In Proceedings of the Third International Conference on Intelligent Systems for Molecular Biology* (ed. Rawling, C., Clark, D., Altman, R., Hunter, L., Lengauer, T., and Wodak, S.), pp. 367-375. AAAI Press, Menlo Park, CA.

Spiegelhalter, D.J., Franklin, R.C.G., and Bull, K. 1990. Assessment, criticism and improvement of imprecise subjective probabilities for a medical expert system. *Uncertainty in Artificial Intelligence 5* (ed. Henrion, M., Shachter, R.D., Kanal, L.N., Lemmer, J.F.), pp. 285-294. Elsevier Science Publishers B.V. (North-Holland).

Uberbacher, E.C. and Mural, R.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.

Wieringa, B., Hofer, E., and 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.

Wilson, R., Ainscough, R., Anderson, K., Baynes, C., Berks, M., Bonfield, J., Burton, J., Connell, M., Copsey, T., Cooper, J., *et al*. 1994. 2.2 Mb of contiguous nucleotide sequence from chromosome III of C. elegans. *Nature* **368:** 32-38.

Xu, Y., Einstein, J.R., Mural, R.J., Shah, M., and Uberbacher, E.C. 1994. 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* (ed. Altman, R., Brutlag, D., Karp, P., Lathrop, R., and Searls, D.), pp. 376-384. AAAI Press, Menlo Park, CA.

Zhang, I. 2000. Protein-length distributions for the three domains of life. *Trends Genet.* **16(3):**107-109.

Zhang, M.Q. 1997. Identification of protein coding regions in the human genome by quadratic discriminant analysis. *Proc. Natl. Acad. Sci.* **94:** 565-568.

Zoubak, S., Clay, O., and Bernardi, G. 1996. The gene distribution of the human genome. *Gene* **174:** 95-102.