

Discriminant analysis and its application in DNA sequence motif recognition

Michael Q. Zhang

Date received (in revised form): 21st August 2000

Abstract

Identification of functional motifs in a DNA sequence is fundamentally a statistical pattern recognition problem. Discriminant analysis is widely used for solving such problems. This paper will review two basic parametric methods: LDA (linear discriminant analysis) and QDA (quadratic discriminant analysis). We will demonstrate their usage in recognition of splice sites and exons in the human genome.

Keywords: *discriminant analysis, functional motif, LDA, QDA, exon, intron*

INTRODUCTION

Almost all of the hereditary information of a living cell is encoded in its genomic DNA sequence. Most important of all are the genes and their regulatory elements. According to the central dogma, genetic information flows as DNA → RNA → protein. Namely, a gene is first transcribed into a pre-mRNA, this transcript is subsequently processed (ie capped, spliced and polyadenylated) and the mature mRNA transcript is then transported from the nucleus into the cytoplasm for translation into the gene product – a functional protein. Therefore, identification of the coding regions is often the first task that is undertaken after a genomic DNA is sequenced. A typical eukaryotic gene structure and its corresponding mature mRNA transcript are depicted in Figure 1. Among 16 possible types of exons,¹ *itexons* (internal

translated or internal coding exons, presented by a black box in the figure) are the most studied. There are many ways to identify itexons in a genomic DNA sequence. Experimentally, the two most popular methods are cDNA sequencing and exon trapping. Computationally, if the cDNA (or a closely related cDNA) sequence is known, a simple alignment between the genomic sequence and the cDNA sequence is sufficient. Otherwise, an exon prediction method must be used. Without knowing in precise detail the mechanism of pre-mRNA splicing, all current exon prediction algorithms would have to rely on some statistical models. The basic assumption is that every functional motif (such as an exon) in a genome should have some (unknown) distinct sequence features that can distinguish it from the surrounding regions (such as the flanking introns).

Dr Micheal Zhang
Cold Spring Harbor Laboratory,
Watson School of Biological
Sciences,
Post Office Box 100,
1 Bungtown Road,
Cold Spring Harbor,
NY 11724

Tel: +1 (516) 367 8393
Fax: +1 (516) 367 8461
E-mail: Mzhang@cshl.org

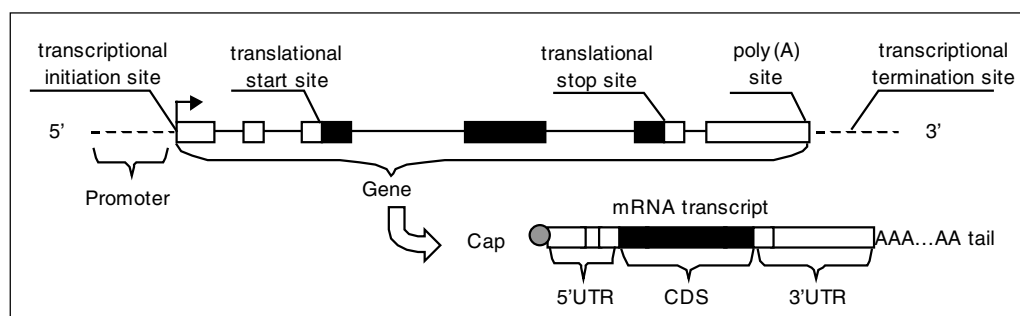


Figure 1: A gene has six exons

Identification of a sequence motif thus falls into the category of statistical pattern recognition. There are numerous books on theory and methodology of statistical pattern recognition, for instance: McLachlan,² Fukunaga,³ Bishop⁴ and Duda and Hart.⁵ Discriminant analysis is a powerful statistical pattern recognition method which has been applied to many DNA sequence motif finding problems, such as splice site prediction,⁶ exon/introns prediction,⁷⁻⁹ translational start site prediction,⁹ polyadenylation site prediction^{10,11} and promoter prediction.^{12,13}

WHAT IS DISCRIMINANT ANALYSIS?

Discriminant analysis is a general statistical method for classification. Given N objects, how can one assign each object into K known classes with minimum errors? For simplicity, we only consider the case of $K=2$, although the theory can be easily generalised to $K > 2$. In order to distinguish one class object from another, two things are needed. Firstly, a set of feature variables $x = \{x_\alpha : \alpha=1, \dots, p\}$ and secondly a decision rule (ie classifier) C such that given the measured values x^i for the i th object, C would be able to map it into either class I (denoted by $+$) or class II (denoted by $-$, see Figure 2). In practice, choosing the set of feature variables that is most discriminative with respect to the two classes is the key. For example, the sex hormone level is a much better discriminative feature variable than the

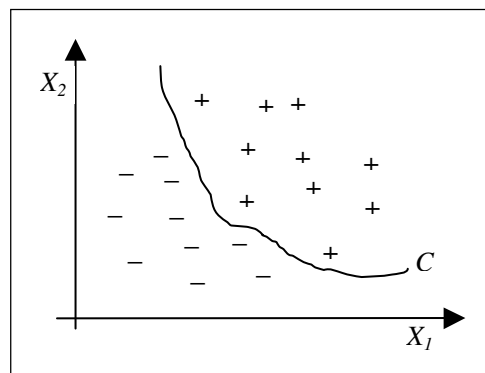


Figure 2: A classifier C separates $N=20$ sample points in $K = 2$ feature space

colour of skins when classifying people into males and females. Although there are many systematic methods for selecting better feature variables, it is still like a black art, which depends heavily on the master's insight to the nature of the subject. In this paper, we shall assume the set of feature variables is decided (or given) and hence we can represent the N objects to be classified as N sample points x^i in the p -dimensional feature space. Discriminant theory is to provide us with the mathematical tools for finding the optimal classifier in the sense of minimising the classification errors.

In general, the (Bayesian) theory assumes the sample points were drawn from two distinct distributions $p(x|+) = f_+(x)$ and $p(x|-) = f_-(x)$. If these conditional distributions and the *a priori* probabilities π_+ and π_- (for a randomly chosen sample being in class $+$ or $-$, respectively) are known, then the *a posteriori* probability $q_+(x)$ of seeing the data x and it belonging to class $+$ is given by the Bayes formula

$$q_+(x) = \frac{\pi_+ f_+(x)}{\pi_+ f_+(x) + \pi_- f_-(x)}$$

this is because

$$\begin{aligned} q_+(x) &= p(+|x) = \frac{p(+,x)}{p(x)} \\ p(x) &= p(x|+)\pi_+ + p(x|-)\pi_- \\ p(x) &= p(x|+)\pi_+ + p(x|-)\pi_- \end{aligned}$$

A discriminant function $h(x)$ is defined as the log-likelihood ratio

$$h(x) = \ln [q_+(x) / q_-(x)]$$

One can choose the decision boundary C_B (the *Bayes decision rule*) as the hyper-surface $h(x) = 0$, because for any given sample point x^i , it would be more likely to belong to class $+$ if $h(x^i) > 0$. By assigning x^i to class $+$, one would make an error with probability $q_-(x^i) < q_+(x^i)$. Similarly, by assigning x^i to class $-$ when $h(x^i) < 0$, one would make an error with probability $q_+(x^i) < q_-(x^i)$. In general for

any decision rule C , the total error (the *Bayes error*)

$$\begin{aligned} \varepsilon &= \text{probability of misclassification} \\ &= \int_{R_+} q_-(x) dx + \int_{R_-} q_+(x) dx \end{aligned}$$

where the regions R_+ and R_- are classified to $+$ and $-$ by C , respectively.

LDA AND QDA

When samples are drawn from two different normal distributions

$$\begin{aligned} f_k(x) &= \frac{1}{(2\pi)^{p/2} |\Sigma_k|^{1/2}} \exp\left\{-\frac{1}{2}(x - \mu_k)^T \Sigma_k^{-1} (x - \mu_k)\right\} \\ &= \frac{1}{(2\pi)^{p/2} |\Sigma_k|^{1/2}} \exp\left\{-\frac{1}{2} \Delta^2(x, \mu_k)\right\} \end{aligned}$$

where μ_k and Σ_k are the mean and the covariance matrix for the class k ($k = +$ or $-$, $|\Sigma_k|$ is the determinant of the $p \times p$ matrix and $\Delta^2(x, \mu_k)$ is called Mahalanobis distance between two vectors x, μ_k), the discriminant function will be a quadratic function of x (through Δ^2):

$$h(x) = -\frac{1}{2} \left[\Delta^2(x, \mu_+) - \Delta^2(x, \mu_-) + \ln \frac{|\Sigma_+|}{|\Sigma_-|} \right] + \gamma_{\pm} \quad (1)$$

where $\gamma_{\pm} = \ln(\pi_+ / \pi_-)$. Geometrically, the decision boundary is a quadratic hyper-surface in p -dimensions (Figure 3) when $\Sigma_+ \neq \Sigma_-$. Using such a quadratic discriminant function for classification is called QDA (quadratic discriminant analysis).

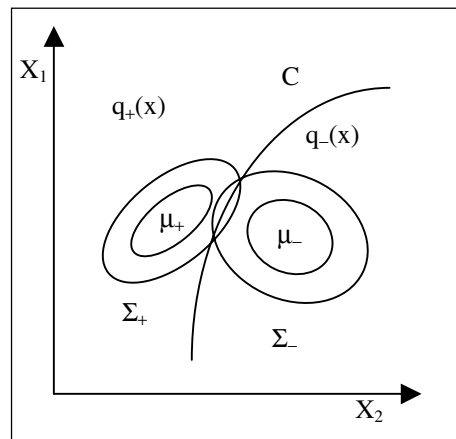


Figure 3: Quadratic decision boundary for normal distribution

When $\Sigma_+ = \Sigma_- = \Sigma$, the quadratic terms in $h(x)$ will be cancelled out:

$$\begin{aligned} h(x) &= (\mu_+ - \mu_-)^T \Sigma^{-1} x - \frac{1}{2} \\ & \quad (\mu_+^T \Sigma^{-1} \mu_+ - \mu_-^T \Sigma^{-1} \mu_-) + \gamma_{\pm} \end{aligned} \quad (2)$$

the Bayes decision boundary will become linear (hyper-plane as seen in Figure 4). Although linear decision boundaries are optimal (in the Bayes sense) only for normal distributions with equal covariance matrices, because of its simplicity, one may always want to know how well one can do with just a linear discriminant function for arbitrary class of distributions. A general linear discriminant function can be written as $h(x) = V^T x + v$. This means that x is projected onto a vector V and the variable $y = V^T x$ in the projected linear space is classified according whether $y > v$ or $y < v$. Suppose the means and variances in the projected subspace are

$$\begin{aligned} \eta_{\pm} &= E\{h(x) | \pm\} = V^T \mu_{\pm} + v \\ \text{and } \sigma_{\pm}^2 &= \text{Var}[h(x) | \pm] = V^T \Sigma_{\pm} V, \end{aligned}$$

the most popular choice for the optimal V is

$$V = \left(\frac{1}{2} \Sigma_+ + \frac{1}{2} \Sigma_- \right)^{-1} (\mu_+ - \mu_-) \quad (3)$$

which maximises the *Fisher criterion*¹⁴ $(\eta_+ - \eta_-)^2 / (\sigma_+^2 + \sigma_-^2)$. One notices that the Fisher coefficient (3) will reduce to that of (2) when $\Sigma_+ = \Sigma_-$, although

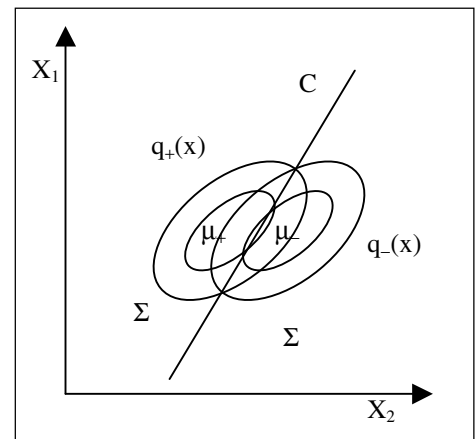


Figure 4: Linear decision boundary for normal distribution when $\Sigma_+ = \Sigma_-$

minimisation of the Fisher criterion cannot provide an optimal value for the constant threshold n which may be chosen by minimising the classification errors in the linear subspace. Using a linear discriminant function (often the Fisher discriminant function) for classification is called LDA (linear discriminant analysis).

In real applications, one normally does not know the distributions. One should always try to transform variables so that they are approximately normal (there are many techniques for doing this, for instance, the Box–Cox transformation¹⁵). Even if one assumes some parametric distributions, one still has to estimate the parameters using the training data. LDA is more robust, because it does not require normality of the distributions and it has fewer parameters to be estimated. But if one has sufficient data and the decision boundary is intrinsically non-linear (two class distributions have very different shapes as indicated by $\Sigma^+ \neq \Sigma^-$), QDA can be superior. Of course, there are also other non-parametric methods that are beyond the scope of this paper. Discriminant analysis can be done equally well by neural networks or machine learning approaches. Iteration algorithms

are used to estimate the decision boundary or the distribution parameters. Here we only focus on multivariate statistical approaches for the analytical clarity.

A PEDAGOGICAL EXAMPLE: SPLICE SITES RECOGNITION BY LDA

Two splice sites define every internal exon. There is an acceptor site (3' splice site or 3'ss) with a conserved dinucleotide AG at the upstream intron boundary and a donor site (5' splice site or 5'ss) with a conserved dinucleotide GT at the downstream intron boundary. (We ignore the rare class introns that do not follow the GT..AG rule.) Using the dataset⁸ itexon.gb¹⁶ ftp://cshl.org/pub/science/mzhanglab/human_exons/ and S-PLUS® (trade mark of Mathsoft Inc.) with the MASS library,¹⁷ we now demonstrate how to use LDA to identify functional splice sites.

Itexon.gb contains 3,440 human internal coding exons in a multi-FASTA format. Each exon also has a 54 nt flanking intron sequence on each side. The following S-PLUS codes (each command line starts with a '>') demonstrate a typical LDA exercise.

Given a donor scoring matrix and an acceptor scoring matrix (log-odd ratios¹)

```
> hdonor.sco
      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]      [,8]      [,9]
[1,] 4.927625 5.455039 3.657552 -6.708694 -6.708694 5.337298 5.651656 3.407280 4.182753
[2,] 4.940989 3.940082 2.771444 -6.708694 -6.708694 2.532078 3.429272 3.076291 4.177265
[3,] 4.306864 4.004144 5.751008 5.990247 -6.708694 5.131641 3.822682 5.783425 4.396518
[4,] 3.857558 4.083199 3.409864 -6.708694 5.990247 2.271352 3.642136 3.126140 5.234395
> hacceptor.sco
      [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]      [,8]      [,9]
[1,] 3.880829 3.741861 3.642345 3.612792 3.426912 3.426912 3.328560 3.657692 3.700662
[2,] 4.904257 4.883169 4.852447 4.861092 4.811186 4.912976 4.933451 5.031327 5.075265
[3,] 3.939965 3.961359 3.894347 3.809521 3.846549 3.757379 3.846549 3.894347 3.598628
[4,] 5.104587 5.151940 5.215812 5.237274 5.293456 5.246280 5.225385 5.065679 5.090574
      [,10]     [,11]     [,12]     [,13]     [,14]     [,15]     [,16]     [,17]
[1,] 3.548945 3.426912 4.475351 2.803553 5.990254 -6.715080 4.526246 4.435813
[2,] 5.114224 4.946127 4.865121 5.640631 -6.715080 -6.715080 3.950054 4.429618
[3,] 3.290467 3.290467 4.444408 0.532712 -6.715080 5.990254 5.445529 4.752387
[4,] 5.144678 5.304566 4.555519 4.601891 -6.715080 -6.715080 3.700662 4.947867
```

Extract 20 itexons with the flanking intron sequences without any 'N's

```

> s <- scan("c:itexon.gb", "")
> s1<- character(20); s0 <- ""; j <- 0
> for(i in 1:length(s)){
+   if(s[i] == ">"){
+     k <- 0
+     if(j != 0){ s1[j] <- s0; s0 <- ""}
+     j <- j+1
+   }
+   k <- k+1
+   if(k > 4){ s0 <- paste(s0,s[i],sep="")}
+ }
> s1[20] <- s0

```

Extract 20 pairs of true donor and acceptor sites and 20 pairs of pseudo ones

```

> a.true <- character(20)
> d.true <- character(20)
> for(i in 1:20){
+   a.true[i] <- substring(s1[i],40,56)
+   s.len <- nchar(s1[i])
+   d.true[i] <- substring(s1[i],s.len-56,s.len-56+8)
+ }
> s2 <- character(20)
> for(i in 1:20){
+   s.len <- nchar(s1[i])
+   s2[i] <- substring(s1[i],51,s.len)
+ }
> w <- regexpr(".....ag..",s2)+50
> a.false <- character(20)
> for(i in 1:20){
+   a.false[i] <- substring(s1[i],w[i],w[i]+17-1)
+ }
> w <- regexpr("...gt...",s1)
> d.false <- character(20)
> for(i in 1:20){
+   d.false[i] <- substring(s1[i],w[i],w[i]+9-1)
+ }

```

Calculate the scores (using 'mat.search' function) and split 10 pairs of scores as the training set and the other 10 as the test set

```

> donor <- sapply(c(d.true,d.false),mat.search,hdonor.sco)
> acceptor <- sapply(c(a.true,a.false),mat.search,hacceptor.sco)
> donor.train <- donor[c(1:10,21:30)]
> donor.test <- donor[c(11:20,31:40)]
> acceptor.train <- acceptor[c(1:10,21:30)]
> acceptor.test <- acceptor[c(11:20,31:40)]
> group <- c(rep(1,10),rep(0,10))

```

Calculate the LDA classifier using the training set

```

> s.lda <- lda(cbind(donor.train,acceptor.train),group)
> s.lda
Call:
lda.matrix(cbind(donor.train, acceptor.train), grouping = group)
Prior probabilities of groups:
0      1
0.5    0.5
Group means:
donor.train acceptor.train
0    42.76343    76.78946
1    46.25065    84.95855
Coefficients of linear discriminants:
LD1
donor.train 0.1372213
acceptor.train 0.3278489

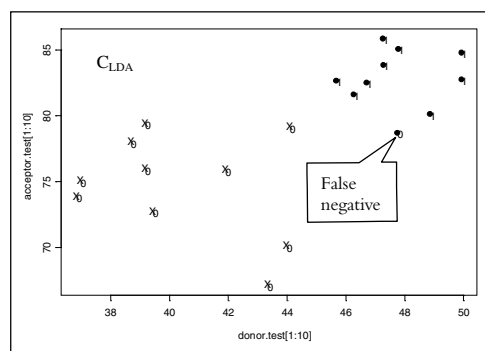
```

Make prediction on the test set

```
> ss.test <- cbind(donor.test,acceptor.test)
> s.pred <- predict(s.lda,ss.test)
> table(group,s.pred$class)
      0      1
0     10     0
1      1     9
```

Hence there is no false positives, but one false negative: make a plot

```
> win.graph()
>
plot(donor.test[1:10],acceptor.test[1:10],xlim=range(donor.test),ylim=range(acceptor.test))
> points(donor.test[11:20],acceptor.test[11:20],pch="X")
> text(donor.test+0.1,acceptor.test,as.character(s.pred$class))
```

**Figure 5:**

XXXXXXXXXXXXXXXX XXX

One can see from Figure 5, even for such a small sample size, LDA was able to classify 20 test acceptor–donor pairs correctly except one false negative (one can replace ‘lda’ everywhere by ‘qda’ and get a QDA result). Finding true splice sites in a large genome is far more difficult because many putative sites can occur simply by chance.¹ One needs to use more discriminant features (especially the content bias between exons and introns in addition to the boundary signals) and more sophisticated statistical algorithms, such as neural nets, hidden Markov models, support vector machines, decision trees or non-linear discriminant analysis.

MZEF:⁸ AN ITEXON RECOGNITION ALGORITHM BY QDA

Statistical studies^{1,18} of human exons and their flanking regions have revealed several characteristic features that may be used as discriminant feature variables.

One of the most powerful features is the in-frame hexamer frequency bias which measures the codon bias as well as codon–codon correlation. Since many of the feature variables are not independent, their intrinsic correlation (as reflected in the covariance matrix among these variables) are characteristic of the functional constraints. For example:

- The donor site score is correlated to the pairing acceptor score, which is consistent to the ‘exon definition model’¹⁹ because of the interaction among different splicing factors across an exon.
- The splice site scores are correlated with the exon length as the exon becomes short (< 50 base pairs, bp), the quality of the splice sites must be high in order to avoid exon skipping.
- Different regions of a vertebrate chromosome are segregated into different isochores²⁰ (large regions of approximately constant GC-content), all compositional measures of genes (hence exons) residing in different isochores will be biased by the genomic GC-content.
- Based on these studies, a QDA algorithm (MZEF) was developed. At that time, GRAIL2²¹ (a neural network exon finder using 22 features) and HEXON/Fgenex⁷ (a LDA exon finder using 17 features) were the best. The

possibility of using fewer feature variables (as many are highly correlated) and still achieving comparable or better performance was tested by using non-linear discriminant analysis that can exploit specific correlation structures through covariance matrices. The results came back positively.

Forty-three completely sequenced genes were randomly selected as the test set and the other sequences in itexon.gb were used to generate the training set of 1879 true exons and 184,217 pseudo-exons (defined as an open reading frame flanked by AG..GT). The training set was first divided into a high-GC set and a low-GC set (cutoff $G+C=0.48$). Nine feature variables were then computed for each set and QDA parameters (sample means and covariant matrices) were estimated. The nine feature variables are (1) intron-exon boundary transition; (2) branch site; (3) acceptor site; (4) exon size; (5) 6-mer exon preference; (6) strand preference; (7) frame preference; (8) donor site; and (9) exon-intron boundary transition. Readers are referred to the original paper⁸ for details. The statistics on the test set are quoted in Table 1. Other statistics are quoted in Table 2 for the ALLSEQ dataset²² that did not overlap with our training set. It should be noted that these datasets were biased because they contained little or no intergenic regions and they were also biased towards exon-rich regions. Later statistics seemed to indicate the overall *ab initio* exon finder accuracy on the whole genome scale is only about 50 per cent. Although many investigators tried to integrate EST/cDNA database matches into gene-finders, the author believes it should be done separately because it would be difficult to attribute a bad result to an erroneous match or to an inferior classifier.

As larger genomic contigs become available, assembly of exons into full gene models by dynamic programming has become more popular. Gene assembly

(such as Genscan²³) can eliminate false positive exons and hence increase the accuracy by imposing frame compatibility and distance constraints; it has been used in many genome centres for annotating large-scale genomic sequences. Since our understanding of promoter architecture, polyadenylation and alternative splicing mechanisms is still very poor, terminal exons and alternative exons cannot be predicted reliably. As a consequence, gene assembly can result in fusing neighbouring genes or missing alternative exons. Improving techniques for recognising individual types of exons will have complementary utility especially when analysing fragments that may not contain the full gene or studying local competing exons for functional characterisation of fine structure of a gene. MZEF is reportedly the best for finding short internal exons (Manpreet Katari, personal communication). In order to assemble exons one lets MZEF output a few overlapping exons in order to choose those that are phase compatible. One should also enter a lower prior probability (such as 0.001) for gene poor (usually high A+T) regions or for good-quality exons in the initial run.

Table 1: Comparison of performance on the test set (upper-case: exon level; lower-case: nucleotide level)

Program	SN	SP	sn	sp	cc
GRAIL2	0.51	0.57	0.79	0.85	0.80
HEXON	0.71	0.65	0.88	0.80	0.83
MZEF	0.84	0.92	0.88	0.95	0.90

(sn = sensitivity, sp = specificity and cc = correlation coefficient)

Table 2: Comparison of performance on the ALLSEQ set

Program	SN	SP	sn	sp	cc
GRAIL2	0.53	0.60	0.79	0.92	0.83
Fgeneh	0.73	0.78	0.83	0.93	0.85
MZEF	0.78	0.86	0.87	0.95	0.89

More recently, we have developed programs: CorePromoter¹³ (for mapping core-promoters and the transcriptional start sites with ~100 bp resolution), CpG_Promoter¹⁴ (for promoter associated CpG island mapping within ~2 kb of the transcription start sites) and JTEF²⁴ (for identifying the last exons using Polyadq¹² as a subsystem). These tools will help to find the ends of genes for function/regulation studies or for better genome annotations by reducing artificial gene fusion. The example below will demonstrate how one may be able to use these QDA-based discrimination tools to annotate a human gene in practice.

ITGB2 AND APP GENE ANALYSIS: A CASE STUDY WITH MZEF AND RELATED TOOLS

Human chromosome 21 is one of the first two human chromosomes fully sequenced (up to a number of known gaps)²⁵ by an international team of researchers. MZEF⁸, Genscan²⁴ and Grail²² were used as the *ab initio* gene

prediction programs. The paper reported that 'MZEF tends to over-predict exons compared with Grail and Genscan. In particular for the large APP gene'. Here, MZEF results for the analysis of two genes in the chromosome 21 are shown, to find out whether this is true.

ITGB2, integrin β -2, gene encodes the integrin β chain β -2. Integrins are integral cell-surface glycoproteins composed of a α chain and a β chain. A given chain may combine with multiple partners resulting in different receptors. They are known to participate in cell adhesion as well as cell-surface mediated signaling. This gene resides in a high G+C region and has a length about 35 kb containing 16 exons (14 itexons). The Genbank annotation (confirmed by cDNAs) and the predictions by both MZEF and Genscan are shown in Table 3. MZEF command line is shown with the default prior probability = 0.02 (assuming the chromosome 21 annotation team was using the default parameters for all the gene-finding programs), reverse strand and no overlaps. Genscan exons predicted in the same

Table 3 Annotated and predicted exons at the ITGB2 locus on chromosome 21

GenBank AL163300 34920 bp	MZEF ch21.seq 2 0.02 0	Genscan
1 297561..297493		1
2 287457 287454..287397		2 287457..287397
3 287044..286956	3 287033 ..286956	3 287044..285728 285844..285728
4 283767..283587	4 283767..283587	4 283767..283587
5 280207..280037	5 280207..280037	5 280207..283587
6 278405..278164	6 278405..278164	6 278405..278164
7 277147..276992	7 277147..276992	7 277147..276992
8 275834..275739	8 275834..275739	8 275945 ..275739 273639..273525
9 271732..271643	9 271732..271643	9 271770 ..271112
10 270216..270076	10 270202 ..270076	10 270202 ..270076
11 268668..268481	11 268668..268481	11 268668..268481
12 266894..266650	12 266894..266650	12 266894..266650
13 266167..265948	13	13 266167..265948
14 265567..265365	14 265567..265365	14 265567..265365
15 263574..263408	15 263574..263408	15 263574..263408
16 263102..263030 262642		262300..261026 258077 16

Errors are in bold. | indicates a CDS boundary.

Table 4 Annotated and predicted exons at the APP locus on chromosome 21.

GenBank:D87675 301692 bp	MZEF 2 0.02 0	Genscan
1 9001 9148..9204	1 98188.98253	1 9003..9204 11471..11603 34160..34244 34916..35021
2 64121..64288	2 64121..64288 81125..81222 (Alu)	2 64121..64288
3 86196..86325	3 86196..86325 98188..98253 119034..119099	3 86196..86325
4 122920..123032	4	4 122920..123032
5 125075..125268	5 125075..125268 129637..129674	5 125075..125268 139630..139711 144384..144569 147465..147574
6 154226..154428	6 154226..154428	6 154226..154428
7* 176087..176254	7*	7* 176087..176254
8* 178853..178909	8* 178853..178909 181429..181521? (Strong EST)	8* 181429..181521? 184376..184392 187130..187292
9 193794..193927	9 193794..192927	9 193794..193927
10 200243..200317	10 200243..200317	10 200243..200317
11 201043..201201	11 201043..201201 205399..205457 205766..205877 209616..209666	11 202043.. 201249
12 220515..220643	12 220515..220643	12 220515..220643
13 221581..221680	13 221611..221680 247010..247084	13 221581..221680 227698..227759 229502
14 264310..264531	14 264310..264531 265844..265951 269119..269173	14 262783 264291 ..264531
15* 271195..271248	15*	15* 271195..271248 272310..272449 274321
16 278599..278699	16 278599..278699	16 276196 278646..278699
17 284404..284550	17 284404..284550	17 284404..284550
18 294502..294603 295722	18	18 294502..294603 295449

Errors are in bold. | indicates a CDS boundary.

locus and in the same strand are also shown. The accuracy of the exon prediction methods are fairly comparable: $SN = SP = 10/13 = 0.77$ for MZEF and $SN = 10/11 = 0.91$, $SP = 10/16 = 0.63$ for Genscan. In this example, one would notice that the predicted exon 270202..270076 by MZEF is not frame-compatible. If one had tried to

output two more overlapping exons, one would have had found the true exon with a slightly lower posterior probability score. The lesson is to run with 0 overlap first and then try to find an overlapping substitute for a frame-incompatible exon.

APP gene encodes an amyloid precursor protein that is cleaved to form amyloid, a major component in amyloid

plaques. This gene became very famous after its mutations were linked to Alzheimer's disease and it has been subjected intensive studies worldwide. It was analysed around the end of 1996 as a test for MZEF before it was deposited into GenBank and the result was published in the original MZEF paper.⁸ Now there are more cDNAs and alternative spliced transcripts being discovered, it is interesting to revisit the gene prediction again after four years. APP resides in a very low C+G region and contains 18 exons among which exons 7, 8 and 15 are known to be alternatively spliced (exon skipping). Table 4 lists the GenBank annotation based on the cDNAs and predictions of MZEF and Genscan with the default parameters. Genscan predicted three separate genes in this locus on the APP strand, to compare with MZEF, this study only counted exons between the two true terminal ones. Again no worse false-positive rate was found for MZEF, as $SN = 12/15 = 0.80$, $SP = 12/25 = 0.48$ for MZEF and $SN = 12/16 = 0.75$, $SP = 12/26 = 0.46$ for GENSCAN. It is interesting that two (exons 7 and 15) of the three alternative exons were missed by MZEF and the other (exon 8) by GENSCAN. One false positive of MZEF is an Alu repeat which could have been masked out if repeats had been screened. (In practice, one always want to mask out the known repeats during the preprocessing.) Another 'false positive' exon (181429..181521) predicted by both programs had a very strong EST hit and receives high scores; the author believes it is a real exon. Another lesson to be learnt is that one should not discard 'false positives' too easily if they have been predicted by several *ab initio* gene-finding programs with high scores, even if there is not yet a cDNA/EST hit. Experimental methods, database matches and *ab initio* predictions are always complementary to each other. It is wrong to assume that *ab initio* methods are no longer useful because more and more cDNAs are becoming available. For one

thing, you can never be sure that you have a complete set of transcript variants for each gene. A theoretical understanding of exon definition and gene regulation is just too important to be ignored even if one could have got them all from the database matches.

If prior probability $p = 0.002$ had been used (instead of the default $p = 0.02$), a more specific prediction (Table 5) would have resulted, at the expense of sacrificing some sensitivity: $SN = 10/16 = 0.63$, $SP = 10/12 = 0.83$ for $p = 0.002$. High specificity (low false positive) is desirable when one wants higher-quality exons (for designing polymerase chain reaction (PCR) primers or for the initial runs).

As MZEF is designed only for itexon prediction, the Cold Spring Harbor Laboratory has been working on special discrimination algorithms in order to detect other types of exons. The cutting edge technology for gene finding is to detect terminal exons. At the Cold Spring Harbor Computational Biology Workshop²⁶ a novel last exon prediction

Table 5: MZEF prediction for $p = 0.002$.

MZEF 2 0.002 0	
1	
43050..43155	
2 64121..64288	
3	
4	
5 125075..125268	
6 154226..154428	
7	
8 178853..178909	
9 193794..193927	
10 200243..200317	
11	
12 220515..220643	
13	
14 264310..264531	
269119..269173	
15	
16 278599..278699	
17 284404..284550	
18	

Table 6: CpG_Promoter prediction:

CpG islands	Promoter-associated
8813..9319	+
9328..9547	+
9761..10203	+
117256..117511	-
176132..176342	-
257735..257942	-
261475..261750	-

Table 7: Core_Promoter prediction:

TSS	Score
8921	0.100
8923	0.094
8920	0.089
8919	0.084
8922	0.078
8918	0.058
8783	0.056

algorithm that is called JTEF²⁵ was reported. The first exon prediction problem is more difficult, mainly because there have not been enough accurate and annotated 5'-sequences (promoters and 5'UTRs). The key is to predict promoter and the transcriptional start site (TSS) reliably. It is proposed to subdivide this problem into three related promoter (large-scale, a proximal and a core promoter) prediction problems.²⁷ CpG_Promoter¹⁴ has been developed for large-scale mapping and Core-Promoter^{28,13} for fine-scale TSS mapping. These gene-finding tools are now available at the web site.²⁹ Using APP gene prediction as an example, it is demonstrated how one could use these tools to identify the ends of APP gene. From the output of CpG_Promoter (Table 6), the first three CpG-islands were linked to the promoter region (with resolution of ~2 kb). Based on this information, a 2 kb sequence (8001..10000) was extracted, Core_Promoter predicted 8921 (Table 7) to be the transcriptional start site (with a

Table 8: Last exon prediction (JTEF):

* 89301 89332 89723 POS TERM [0.691238]
* 98517 98672 100365 POS TERM [0.589478]
* 98517 98672 100372 POS TERM [0.596260]
* 98517 98672 100376 POS TERM [0.599228]
* 98517 98672 100380 POS TERM [0.601334]
* 98517 98672 100476 POS TERM [0.603293]
* 154226 154481 155453 POS TERM [0.645983]
* 154226 154481 156173 POS TERM [0.645983]
* 154226 154481 157124 POS TERM [0.645983]
* 173950 174244 174307 POS TERM [0.999951]
* 173950 174244 175247 POS TERM [0.705701]
* 173950 174244 175330 POS TERM [0.705701]
* 173950 174244 175877 POS TERM [0.705701]
* 173950 174244 175972 POS TERM [0.705701]
* 187185 187333 187353 POS TERM [0.565787]
* 201012 201143 202839 POS TERM [0.789249]
* 201012 201143 204279 POS TERM [0.799420]
* 208585 208608 209209 POS TERM [0.881959]
* 208585 208608 209281 POS TERM [0.881347]
* 236985 237166 239274 POS TERM [0.633897]
* 236985 237166 240299 POS TERM [0.633897]
* 272255 272449 273414 POS TERM [0.965201]
* 272255 272449 274321 POS TERM [0.965198]
* 272255 272449 274412 POS TERM [0.965198]
* 294502 294603 295449 POS TERM [0.984603]
* 294502 294603 295454 POS TERM [0.983722]
* 294502 294603 295458 POS TERM [0.983622]
* 294502 294603 295506 POS TERM [0.983080]
* 294502 294603 295700 POS TERM [0.983052]
* 294502 294603 295997 POS TERM [0.983052]

resolution of ~100 bp), which is indeed pretty close to 9001, the annotated TSS in GenBank. To detect the last exon, JTEF was used and the output is shown in Table 8. The exon 294502..294603|295449 has the highest score (0.984603), where 295449 was predicted to be the putative polyA-signal. This agrees very well with the GenBank annotation as well as the Genscan prediction of the last exon in its third predicted gene.

References

1. Zhang, M. Q. (1998), 'Statistical features of human exons and their flanking regions', *Hum. Mol. Genet.*, Vol. 7, pp. 919–932.
2. McLachlan, G. J. (1992), 'Discriminant Analysis and Atatistical Pattern Recognition', John Wiley, New York.
3. Fukunaga, K. (1990), 'Introduction to Statistical Pattern Recognition', 2nd Edn., Academic Press, New York.
4. Bishop, C. M. (1995), 'Neural Networks for Pattern Recognition', Clarendon Press, Oxford.
5. Duda, R. O. and Hart, P. E. (1973), 'Pattern Classification and Scene Analysis', John Wiley, New York.
6. Solovyev, V. V. and Lawrence, C. (1993), 'Identification of human gene functional regions based on oligonucleotide composition', 'Proc. 1st International Conference on Intelligent Systems for Molecular Biology', AAAI from Menlo Park, CA, pp. 371–379
7. Solovyev, V. V., Salamov, A. A. and Lawrence, C. B. (1994), 'Predicting internal exons by oligonucleotide composition and discriminant analysis of spliceable open reading frames', *Nucl. Acid. Res.*, Vol. 22, pp. 5156–5163.
8. Zhang, M. Q. (1997), 'Identification of protein coding region in the human genome by quadratic discriminant analysis', *Proc. Natl Acad. Sci. USA*, Vol. 94, pp. 565–568.
9. Salamov, A. A., Nishikawa, T. and Swindells, M. B. (1998), 'Assessing protein coding integrity in cDNA sequencing projects', *Bioinformatics*, Vol. 14, pp. 383–390.
10. Salamov, A. A. and Solovyev, V. V. (1997), 'Recognition of 3'-processing sites of human mRNA precursors', *CABIOS*, Vol. 13, pp. 23–28.
11. Tabaska, J. and Zhang, M. Q. (1999), 'Detection of polyadenylation signals in human DNA sequences', *Gene*, Vol. 231, pp. 77–86.
12. Zhang, M. Q. (1998), 'Identification of human gene core promoters in silico', *Genome Res.*, Vol. 8, pp. 319–326.
13. Ioshikhes, I. P. and Zhang, M. Q. (2000), 'Large-scale human promoter mapping using CpG islands', *Nature Genet.*, in press.
14. Fisher, R. A. (1936), 'The use of multiple measurements in taxonomic problems', *Ann. Eugen.*, Vol. 7, pp. 179–188.
15. Box, G. E. P. and Cox, D. R. (1964), 'An analysis of transformations', *J. R. Statist. Soc. B*, Vol. 26, pp. 211–252.
16. ftp://cshl.org/pub/science/mzhanglab/human_exons/
17. Venables, W. N. and Ripley, B. D. (1999), 'Modern Applied Statistics with S-PLUS', 3rd Edn., Springer.
18. Fickett, J. W. and Tung, C. S. (1992), 'Assesment of protein coding measures', *Nucl. Acid. Res.*, Vol. 20, pp. 6441–6450.
19. Robberson, B. L., Cote G. J. and Berget, S. M. (1990), 'Exon definition may facilitate splice site selection in RNAs with multiple exons', *Mol. Cell. Biol.*, Vol. 10, pp. 84–94.
20. Bernardi, G., Mouchiroud, D. and Gautir, C. (1990), 'Compositional patterns invertebrate genomes: conservation and change in evolution', *J. Mol. Evol.*, Vol. 28, pp. 7–18.
21. Uberbacher, E. C. and Mural, R. J. (1991), 'Locating protein coding region in human DNA sequences using a multiple sensor – neural net approach', *Proc. Natl Acad. Sci. USA*, Vol. 88, pp. 11261–11265.
22. Buset, M. and Guigo, R. (1996), 'Evaluation of gene structure prediction programs', *Genomics*, Vol. 34, pp. 353–367.
23. Burge, C. and Karlin, S. (1997), 'Prediction of complete gene structures in human genomic DNA', *J. Mol. Biol.*, Vol. 268, pp. 78–94.
24. Tabaska, J. E., Davuluri, R. and Zhang, M. Q. (2000), 'A novel 3'-terminal exon recognition algorithm', presented at CSHL Computational Biology Workshop 9/99 and EBI Gene-finding Workshop, June 2000. Manuscript in preparation.
25. Hattori *et al.* (2000), 'The DNA sequence of human chromosome 21', *Nature*, Vol. 405, pp. 311–319.
26. Cold Spring Harbor Computational Biology Workshop: Bridging the gap between sequence and function, organised by Zhang, M. Q., Koonin, E. and Uberbacher, E., September 1999, Cold Spring Harbor, NY.
27. Zhang, M. Q. (1997), 'On a new strategy of promoter recognition', Georgia Tech International Conference on Bioinformatics: Gene discovery *in silico*, 6th–9th November, Georgia Tech. University, Atlanta, GA.
28. Zhang, M. Q. (1998), 'A discrimination study of human core-promoters', in proceedings of PSB'98. 4th–9th January, Maui, Hawaii. Altman, R., Donker, A. K., Hunter, L. and Klein, T. E., Eds, World Scientific, Singapore, pp. 240–251.
29. <http://www/cshl.org/mzhanglab/>