DNA AND CELL BIOLOGY
Volume 26, Number 7, 2007

Pp. 477–483
DOI: 10.1089/dna.2007.0583

Prediction of Primate Splice Site Using Inhomogeneous Markov Chain and Neural Network

LIBIN LIU, YEE-KIN HO, and STEPHEN YAU

ABSTRACT

The inhomogeneous Markov chain model is used to discriminate acceptor and donor sites in genomic DNA sequences. It outperforms statistical methods such as homogeneous Markov chain model, higher order Markov chain and interpolated Markov chain models, and machine-learning methods such as *k*-nearest neighbor and support vector machine as well. Besides its high accuracy, another advantage of inhomogeneous Markov chain model is its simplicity in computation. In the three states system (acceptor, donor, and neither), the inhomogeneous Markov chain model is combined with a three-layer feed forward neural network. Using this combined system 3175 primate splice-junction gene sequences have been tested, with a prediction accuracy of greater than 98%.

INTRODUCTION

With the development of technology, DNA sequencing has become easier now than ever before. The number of DNA sequences deposited in GenBank increased exponentially in recent years. The number of newly sequenced genomes has also dramatically increased. One of the most significant events is the completion of the human genome project in 2003. The abundance of these data demands highly accurate computational tools to extract useful information. Identification of protein-coding genes in genomic DNA sequences has been one of the most challenging topics in the last few decades. In prokaryotes, the coding region is the single open reading frame (ORF), while in eukaryotes genes are usually organized as exons and introns. Genomes of eukaryotes, especially higher eukaryotes, may have less than 10% coding sequence. Therefore, to find out the genes from eukaryotes, it is important to find splice sites. Several computational methods have been developed to predict splice sites, using either stand-alone splice site finders, or gene finders, which use splice finder as subroutine (Stormo, 2000; Pertea *et al.*, 2001). The performance of splice finder directly affects the performance of gene finder. If a splice finder method identifies all the splice sites correctly, it would identify almost all the protein-coding regions correctly.

Artificial neural network was originally developed to model the information process in the brain (Wu and McLarty, 2000). However, its development led to a highly efficient machine-learning algorithm that is independent of biological processes. One of the most important properties of the neural network is its capability to learn from examples. Due to this property, it has been successfully implemented in many fields with immature theory but abundant data. Applications of neural network in bioinformatics include coding region recognition, transcriptional and translational signal prediction, protein secondary and tertiary structure prediction, protein folding class prediction, protein family classification, and so on. These successful applications stimulate more research to implement neural network in other areas of bioinformatics.

Markov chain model is a well-established statistical model used in the prediction of a variety of signal sites (Durbin *et al.*, 1998; Roelin *et al.*, 2003). As it reflects the correlation between neighboring nucleotide bases, this model and its variations are widely used. Higher order Markov chain model usually has more accuracy than lower order Markov chain model. However, in the higher order model, estimating all transition probabilities may be difficult. Interpolated Markov chain model overcomes this problem by combining probabilities from different order models (Avery, 2002; Deshpande and Karypis, 2002). Hidden Markov models also have been used to detect homogeneous DNA segments (Boys *et al.*, 2000).

In this paper, we propose the use of the inhomogeneous Markov chain model in splice site prediction. Compared to other Markov chain models (Blaker and Merz, 1998), this one offers higher accuracy with less computation. It has been

Department of Mathematics, Statistics and Computer Science, University of Illinois at Chicago, Chicago, Illinois.

successfully used to discriminate donor and acceptor sites. For the three states system (donor, acceptor, and others), neural network is combined with the inhomogeneous Markov chain model to make the accuracy of prediction greater than 98%.

ALGORITHM DESCRIPTION

Inhomogeneous Markov model

Statistical models have been used for a long time in DNA sequence analysis. The simplest statistical model is the "nucleotide dice." The DNA sequence could be regarded as an outcome of continuous tossing of a nucleotide dice with four sides. Each side of the dice represents a nucleotide, A, T, G, or C. The parameters of this model are the probabilities of each nucleotide: P_A, P_T, P_G, P_C. The sum of these four probabilities equals one. We could estimate these parameters by scanning the DNA sequence database and counting the number of each nucleotide. The ratio of the number of each nucleotide to the total number of nucleotides in the sequence represents its probability. Thus, the probability of a new DNA sequence can be computed using the following formula:

$$P = \prod_{i=1}^{n} P_i = P_A^{n_A} P_T^{n_T} P_G^{n_G} P_C^{n_C},$$

where n is the total length of the DNA sequence; n_S, $S \in$ (A, T, G, C) is the number of each nucleotide.

In this model, parameters are the same for all the positions. So it only reflects the number of each nucleotide without considering the order of nucleotides. In multidice model, we use different dices for different positions. Parameters could be estimated by scanning the aligned DNA sequence database. So the probability of a given DNA sequence could be calculated as

$$P = \prod_{i=1}^{n} P_{s_i}$$

P_{s_i} is the probability of nucleotide $S \in$ (A, T, G, C) in position i.

In the above models each nucleotide is assumed to be independent of others. Obviously it does not reflect the reality. It is easy to notice that some nucleotides correlate with others in DNA sequences. A first order Markov chain is a sequence of random variables where the probability of S_i depends only on the preceding nucleotide, S_{i-1}. $P(S_i|S_{i-1}) = P(S_i|S_{i-1}, S_{i-2}, \ldots, S_1)$

There are a total of 16 parameters in the first order Markov chain: P(A|A), P(A|T),…, P(G|G), of which 12 are independent parameters. These could be easily estimated by counting the number of di-nucleotides in the DNA sequence database. Transition probabilities are computed as follows:

$$P(S_i|S_{i-1}) = \frac{f(S_{i-1}S_i)}{\sum_{x\in\{A,T,C,G\}} f(S_{i-1}x)}$$

f(XY) denotes the number of occurrences of string *XY* in the database. Given a new sequence, its probability could be computed as

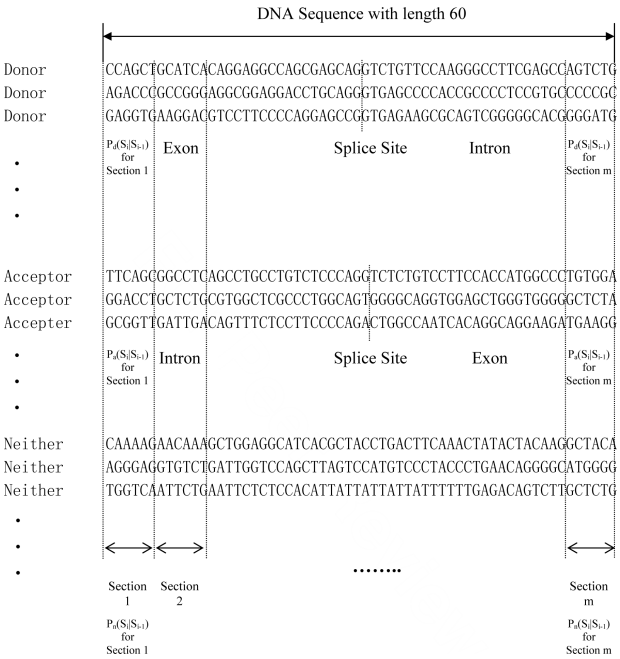

FIG. 1. Splice junction site dataset and the inhomogeneous Markov chain model.

$$P = \prod_{i=1}^{n} P(S_i|S_{i-1})$$

For the kth order Markov chain model, the current nucleotide S_i depends on the preceding k nucleotides. The total number of independent parameters of this model is $4^{k+1} - 4$. Nucleotide-dice model could be regarded as 0th order Markov chain model. Higher order Markov chain usually does better job in modeling the correlation between nucleotides. As the number of parameters increases exponentially with order, absence of some kth combination of nucleotides in the training data may cause problems in the training process.

To integrate position information in the Markov chain, we use the inhomogeneous Markov chain model. DNA sequences are divided into several sections. Parameters of Markov chain model are estimated for each section. The probability of a new DNA sequence is computed as

$$P = \prod_{i=1}^{m} \prod_{j=1}^{n_i} P_i(S_j|S_{j-1}), \qquad (*)$$

where m is the number of sections; n_i is the number of nucleotides in ith section.

To predict splice junction using inhomogeneous first order Markov chain model, we need to estimate the transition probabilities for the donor and acceptor sites. As shown in Figure 1, each training DNA sequence has a total length of 60 and is divided into m sections. Splice junction is in the middle of the sequence, either donor or acceptor. Parameters of donor training data are estimated for each section and the same is done for the acceptor training data. So we have $32m$ parameters totally, of which $16m$ parameters are for the donor splice junction and the

rest are for the acceptor splice junction. When an unknown DNA sequence of length 60 is given, we can compute the following discriminator to see whether the middle of the sequence is a donor or an acceptor site.

$$d = \log\left(\frac{\prod_{i=1}^{m}\prod_{j=1}^{n_i} P_{d,i}(S_j|S_{j-1})}{\prod_{i=1}^{m}\prod_{j=1}^{n_i} P_{a,i}(S_j|S_{j-1})}\right),$$

where m is the number of sections; n_i is the number of nucleotides in section i; $P_{d,i}(S_j|S_{j-1})$ is the transition probability from S_{j-1} to S_j in the donor splice junction of section i; $P_{a,i}(S_j|S_{j-1})$ is the transition probability in the acceptor splice junction of section i. If $d > 0$, the new DNA sequence could be the donor splice junction; otherwise, it could be the acceptor splice junction.

It is easy to extend the inhomogeneous Markov chain from the first to the kth order. To achieve this, we only need to find out the kth transition probabilities for each section.

Neural network

First order inhomogeneous Markov chain model can achieve above 95% accuracy in discriminating donor and acceptor splice sites. For the system to discriminate three states, donor, acceptor, and neither, we could compute the following three probabilities and choose the biggest one as output:

$$P_d = \prod_{i=1}^{m}\prod_{j=1}^{n_i} P_{d,i}(S_j|S_{j-1})$$

$$P_a = \prod_{i=1}^{m}\prod_{j=1}^{n_i} P_{a,i}(S_j|S_{j-1})$$

$$P_n = \prod_{i=1}^{m}\prod_{j=1}^{n_i} P_{n,i}(S_j|S_{j-1})$$

P_a: probability of the middle of the given sequence to be an acceptor splice junction; P_d: probability of the middle of the given sequence to be a donor splice junction; P_n: probability of the middle of the given sequence to be neither a donor nor an acceptor splice junction.

The accuracy of the three states system is lower than that of the two states system. Let us look for the reason in the model itself. The three states system model integrates position information and correlation between neighboring nucleotides. If we take logarithm on both sides of equation (∗), we get

$$\log P = \sum_{i=1}^{m}\sum_{j=1}^{n_i} \log P_i(S_j|S_{j-1})$$

This equation sums up the probability contribution from each section. To know if each section has the same weight of contribution or if some sections have larger weight of contribution than others, we use neural network to assign weight to each section.

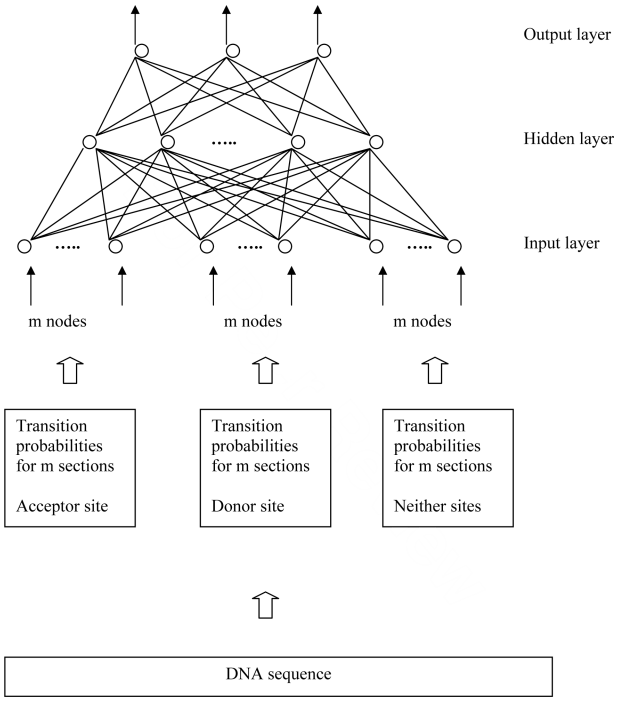

FIG. 2. Structure of the neural network.

As shown in Figure 2, the system is designed as a three-layer feed forward fully connected neural network. The first layer is the input layer. The number of neurons in this layer is $3m$, where m is the number of sections. Probability contribution of each section for each state corresponds to each neuron in the input layer. The input data is normalized with zero means and unity standard deviations. The second layer is the hidden layer. We try different number of neurons and determine the optimal number of neurons in the hidden layer. The last layer is the output layer, which contains three neurons. The outcomes of neural network are converted to indication vectors $\langle 1\ 0\ 0\rangle$, $\langle 0\ 1\ 0\rangle$, and $\langle 0\ 0\ 1\rangle$, which represent the donor, acceptor, and neither states, respectively. The conversion is realized by using compete function, which converts the highest element in the vector to one and the other two elements to zero.

The initial weights are assigned randomly between 0 and 1. Back propagation method is used to minimize the mean square error of all neurons. The number of iterations is set as 2000, but this number could be increased if the error drops dramatically after 2000 iterations.

RESULTS AND DISCUSSION

To evaluate the performance of the combined system described above, we need a DNA database with donor and acceptor splice junctions accurately annotated. This database should also contain DNA sequences that are neither donor nor acceptor. To compare the performance of this system with other algorithms, we choose the primate splice junction gene sequences dataset, which is distributed as part of the UCI KDD

TABLE 1. THE ACCURACIES OF TWO STATES SYSTEM PREDICTION ON DONOR AND ACCEPTOR SITES WITH THE FIRST ORDER INHOMOGENEOUS MARKOV CHAIN MODEL

Number of sections	*Part 1*	*Part 2*	*Part 3*	*Part 4*	*Part 5*	*Part 6*	*Part 7*	*Part 8*	*Part 9*	*Part 10*	*Average*
0	0.7961	0.7778	0.8105	0.8105	0.7647	0.6863	0.7582	0.8039	0.7368	0.7500	**0.7695**
2	0.9085	0.9281	0.9085	0.9150	0.8497	0.9150	0.9216	0.8618	0.8882	0.9211	**0.9018**
4	0.9281	0.9346	0.9346	0.9216	0.8627	0.9085	0.9542	0.9013	0.8882	0.9276	**0.9161**
6	0.9477	0.9542	0.9150	0.9346	0.9346	0.9281	0.9346	0.9145	0.9013	0.9408	**0.9305**
10	0.9608	0.9869	0.9542	0.9542	0.9673	0.9542	0.9673	0.9605	0.9342	0.9605	**0.9600**

Archive (Blaker and Merz, 1998). Many other algorithms have been evaluated using this dataset.

Dataset description

All examples in the dataset are selected from GenBank. The donor and acceptor categories contain all gene splice junctions for primates in GenBank 64.1. The nonsplice category contains known DNA sequences that do not include splice sites. Each entry includes a category (donor, acceptor, neither), a name, and 60 nucleotides. The possible junction sites are in the middle of the 60 nucleotides. There are a total of 3190 entries, of which 767 are donor sites, 768 are acceptor sites, and 1655 are neither. Some DNA sequences contain special characters besides A, T, G, and C to indicate the ambiguity of nucleotide. After removing these entries, there are 762 donor sites, 765 acceptor sites, and 1648 neither sites. Thus the total number of entries used in the experiments is 3175.

We used a 10-fold validation in the experiments to estimate the performance of the systems. The whole dataset is divided into 10 equal-sized disjoint partitions. Each partition contains approximately 25% donor sites, 25% acceptor sites, and 50% neither sites. For each partition, we use all data outside the partition to train the system and then test the system in the partition. The reported accuracy represents the average of the accuracies computed for all 10 partitions.

The two states system

The two states system discriminates only the donor and acceptor sites. The performance of this system is shown in Table 1. The DNA sequences are divided into 0, 2, 4, 6, and 10 sections. Section 0 is the homogeneous Markov chain model. From the table we can see that inhomogeneous Markov chain model greatly improves the performance compared to the homogeneous model. The prediction accuracy of the homogeneous model is only around 77%. With the inhomogeneous Markov chain model, the accuracy of prediction jumps to 90% even with only two sections and continuously increases with the number of sections. It reaches above 96% when the number of sections is 10.

Table 2 gives a comparison of the performance of inhomogeneous Markov chain model and of other algorithms (Deshpande and Karypis, 2002). The highest accuracy of prediction of other algorithms is 93.3%, for the algorithm *k*-nearest neighbor when $k = 5$. The predictions of most homogeneous

TABLE 2. COMPARISON OF THE PERFORMANCE OF INHOMOGENEOUS MARKOV CHAIN MODEL WITH OTHER ALGORITHMS

Algorithm			*Accuracy*
Inhomogeneous Markov chain	$n = 0$		0.7695
	$n = 2$		0.9018
	$n = 4$		0.9161
	$n = 6$		0.9305
	$n = 10$		0.9600
K-nearest neighbor		Cosine	0.7294
	$k = 1$	Global	0.9220
		Local	0.9115
		Cosine	0.7360
	$k = 5$	Global	0.9390
		Local	0.9200
		Cosine	0.7494
	$k = 20$	Global	0.9155
		Local	0.8866
Simple Markov chain and SVM	Order = 0	SVM	0.7439
		Markov	0.7399
	Order = 1	SVM	0.7812
		Markov	0.7700
	Order = 2	SVM	0.8342
		Markov	0.8041
	Order = 3	SVM	0.8768
		Markov	0.8454
Interpolated Markov chain and SVM	Order = 1	SVM	0.7727
		Markov	0.7530
	Order = 2	SVM	0.8022
		Markov	0.7864
	Order = 3	SVM	0.8277
		Markov	0.8153
Selective Markov chain and SVM	Order = 1	SVM	0.7865
		Markov	0.7721
	Order = 2	SVM	0.8343
		Markov	0.8146
	Order = 3	SVM	0.8769
		Markov	0.8500

TABLE 3. THE ACCURACIES OF TWO STATES SYSTEM PREDICTION ON DONOR AND ACCEPTOR SITES WITH THE SECOND ORDER INHOMOGENEOUS MARKOV CHAIN MODEL

Number of sections	*Part 1*	*Part 2*	*Part 3*	*Part 4*	*Part 5*	*Part 6*	*Part 7*	*Part 8*	*Part 9*	*Part 10*	*Average*
0	0.8170	0.8366	0.8301	0.7908	0.7778	0.8301	0.8039	0.7632	0.8289	0.8553	**0.8134**
2	0.9346	0.9542	0.9020	0.8954	0.8824	0.9346	0.9085	0.9342	0.9013	0.9079	**0.9155**
4	0.9539	0.9216	0.9346	0.9281	0.9216	0.8954	0.9608	0.9150	0.9605	0.9079	**0.9299**
6	0.9539	0.9281	0.9281	0.9085	0.9020	0.9150	0.9281	0.9216	0.9539	0.9013	**0.9241**
10	0.9276	0.8758	0.9346	0.9281	0.9216	0.9020	0.9477	0.9216	0.9605	0.9065	**0.9280**

Markov chain models and their variations are 70–80% accurate. None of them is above 90%, even with the help of support vector machine (SVM). On the contrary, prediction accuracies of inhomogeneous Markov chain models are above 90% and the highest is 96%. So we can say that inhomogeneous Markov chain model outperforms homogeneous Markov chain model.

We also extended the two states system from first order to second order inhomogeneous Markov chain model. The results are shown in Table 3. Comparison between the performances of these two models is shown in Figure 3. We can see that the second order model performs slightly better than the first order model when the number of sections is less than 6. When the number of sections is greater than or equal to 6, performance of the second order model is poor compared to that of the first order model. Higher order Markov chain models usually perform better than the lower order models, but easily encounter problems in the training process. The training data might not contain enough tri-nucleotides to estimate transition parameters. With increase in the number of sections, each section contains lesser and lesser training data. If some sections do not contain some specific tri-nucleotide, pseudo counts are used. This might lead to errors in future prediction. This is why the second order model outperforms the first order model when the number of sections is small, but shows poorer performance when the number of sections is large. So the first order inhomogeneous Markov chain model can be used if we only need to discriminate between donor and acceptor sites.

The three states system

To find splice junction sites in unknown genomic sequences, distinguishing only the donor and acceptor sites is not enough. The system should be able to tell whether the middle of any window of length 60 is donor, acceptor, or neither. We still use the first order inhomogeneous Markov chain model and the results are shown in Table 4. As seen, the accuracy of prediction

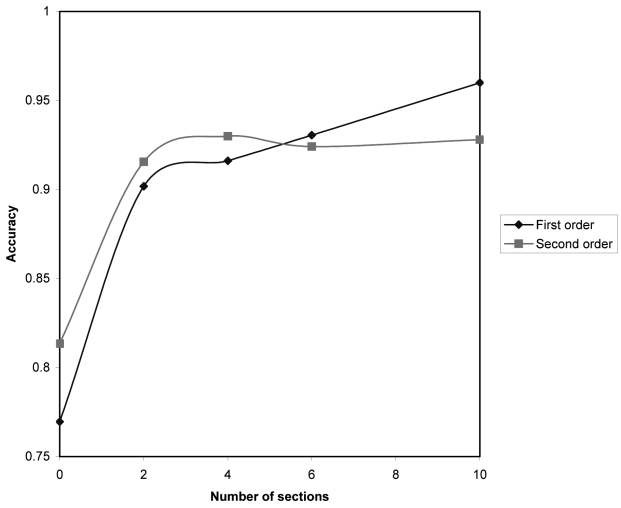

FIG. 3. Comparison of the first order and second order inhomogeneous Markov chain models.

increases with the number of sections as for the two states system. However, the accuracy of prediction is much lower than that of the two states system. Even when the number of sections is 10, the accuracy is less than 80%.

We combine the three states system with the artificial neural network to improve its performance. Different sections play

TABLE 4. THE ACCURACIES OF THREE STATES SYSTEM PREDICTION ON DONOR, ACCEPTOR, AND NEITHER SITE WITH THE FIRST ORDER INHOMOGENEOUS MARKOV CHAIN MODEL

Number of sections	*Part 1*	*Part 2*	*Part 3*	*Part 4*	*Part 5*	*Part 6*	*Part 7*	*Part 8*	*Part 9*	*Part 10*	*Average*
2	0.6456	0.6006	0.6101	0.6226	0.6478	0.6321	0.6415	0.6447	0.6562	0.6171	**0.6318**
4	0.68084	0.6855	0.6761	0.6918	0.7013	0.6635	0.6918	0.7138	0.6972	0.6677	**0.6869**
6	0.7215	0.7310	0.7296	0.7390	0.7233	0.7453	0.7358	0.7201	0.7547	0.7445	**0.7345**
10	0.8145	0.8270	0.7893	0.7987	0.7987	0.8050	0.7830	0.7855	0.7722	0.8133	**0.7987**

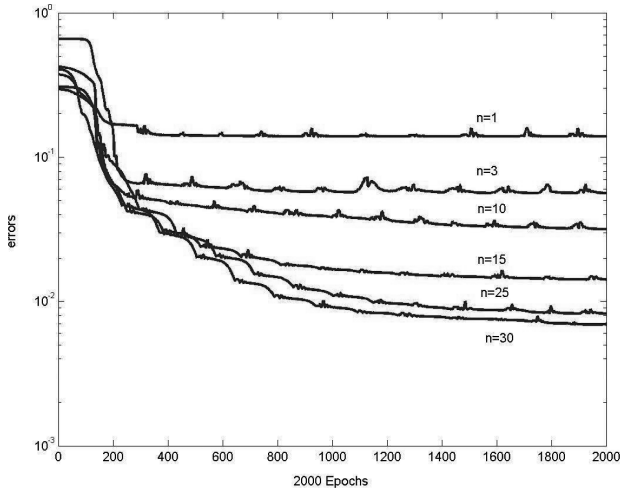

FIG. 4. Relationship between mean square error and the number of epochs in networks with different number of neurons in the hidden layer.

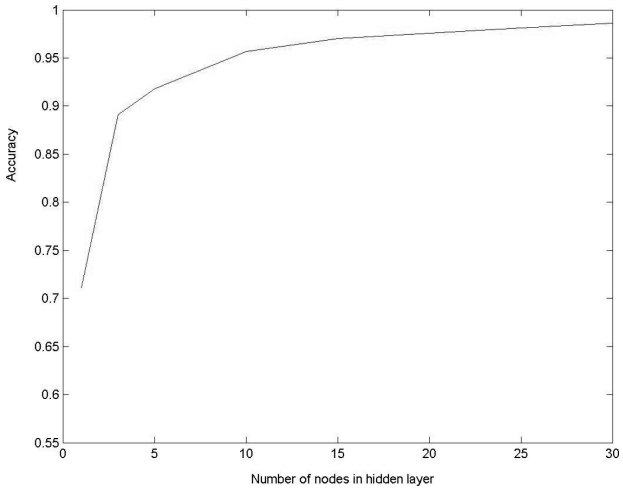

FIG. 5. Relationship between accuracy rate and the number of neurons in the hidden layer.

different roles in the prediction of splice junction sites. The main function of neural network is to add the weight on every section and map the relationship between DNA sequences and splice junction sites. We use a three-layer feed forward neural network. The number of neurons in the first layer is $3m$, where m is the number of sections. Each neuron corresponds to every state and every section. The output layer contains three neurons. The outcomes of neural network are converted to the indicator vectors $\langle 1, 0, 0\rangle$, $\langle 0, 1, 0\rangle$, and $\langle 0, 0, 1\rangle$, which represent the acceptor, donor, and neither states, respectively. The transfer function is a logistic sigmoidal function. This function is differentiable and could be used in the back propagation training.

There is no theoretical guide regarding how to choose the number of neurons in the hidden layers. So the optimization of network is focused on the number of sections and the number of neurons in the hidden layer. We begin with the network having 10 sections. For this neural network, there are 30 input neurons. We try different number of neurons in the hidden layer. From Figure 4, we can see that with increase in the number of neurons in the hidden layer, the mean square error decreases. We also test the performance of the network. Because the initial weights of the network are randomly selected, we test every setting of the network five times and get the average as the accuracy of performance. The results are summarized in Table 5. The accuracy increases from 71.1% to 98.5% with increase in number of neurons in the hidden layer (Fig. 5).

We repeat the same procedures for networks with 2, 4, and 6 sections and choose the configuration with the highest accuracy. From Figure 6, we see that when the number of sections increases, the performance improves. When the number of sections is 10 and the number of neurons in the hidden layer is 30, the accuracy reaches its maximum, 98.5%.

Compared to previous applications of neural network in predicting splice junction sites, our method has much higher accuracy rate. In the previous method, DNA sequences or their numeric representations were input into the network directly. It contained information only of this single sequence. In

TABLE 5. THE ACCURACY OF THREE STATES SYSTEM PERFORMANCE WITH DIFFERENT NUMBER OF NEURONS IN THE HIDDEN LAYER

Number of hidden nodes						*Average*
1	0.7114	0.7100	0.7121	0.7125	0.7104	**0.7113**
3	0.8828	0.8968	0.8898	0.9014	0.8846	**0.8910**
5	0.9142	0.9185	0.9217	0.9136	0.9213	**0.9179**
10	0.9591	0.9563	0.9584	0.9577	0.9521	**0.9567**
15	0.9675	0.9703	0.9706	0.9738	0.9689	**0.9702**
25	0.9811	0.9846	0.9818	0.9822	0.9790	**0.9817**
30	0.9842	0.9885	0.9853	0.9839	0.9867	**0.9857**

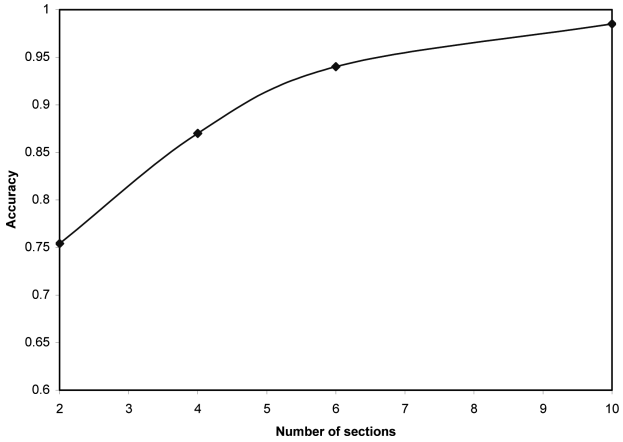

FIG. 6. Relationship between accuracy rate and the number of sections.

comparison, in our improved method DNA sequences are preprocessed using inhomogeneous Markov chain before being fed into the network. Due to this preprocess, the input of network contains information not only about this sequence but also about the whole database. Consequently, our new method helps to reduce the errors in prediction.

CONCLUSION

Inhomogeneous Markov chain model contains information on position and correlations between nucleotides. It outperforms the higher order and interpolated Markov chain models in discriminating donor sites and acceptor sites. In the three states system (donor, acceptor, or neither), inhomogeneous Markov chain model is combined with neural network. The prediction accuracy is 98.5%.

ACKNOWLEDGMENT

We thank one of the reviewers for the suggestion to revise this paper.

REFERENCES

AVERY, P.J. (2002). Fitting interconnected Markov chain models—DNA sequences and test cricket matches. Statistician **51,** 267–278.

BLAKER, C.L., and MERZ, C.J. (1998). UCI repository of machine learning dataset. http://kdd.ics.uci.edu (retrieved Oct. 2004).

BOYS, R.J., HENDSERSON, D.A., and WILKINSON, D.J. (2000). Detecting homogeneous segments in DNA sequences by using hidden Markov models. Appl. Stat. **49,** 269–285.

DESHPANDE, M., and KARYPIS, G. (2002). Evaluation of techniques for classifying biological sequences. In *Proceedings of the 6th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, pp. 417–431.

DURBIN, R., EDDY, S., KROGH, A., and MITCHINSON, G. (1998). *Biological Sequence Analysis* (Cambridge University Press, Cambridge).

PERTEA, M., LIN, X., and SALZBERG, S. (2001). GeneSplicer: a new computational method for splice site prediction. Nucleic Acids Res. **29,** 1185–1190.

ROELIN, D., RICHARD, H., and PRUM, B. (2003). SIC: a tool to detect short inverted segments in biological sequence. Nucleic Acids Res. **31,** 3669–3671.

STORMO, G.A. (2000). Gene-finding approaches for eukaryotes. Genome Res. **10,** 394–397.

WU, C., and McLARTY, J. (2000). *Neural Networks and Genome Informatics. Methods in Computational Biology and Biochemistry*, Vol. 1. A.K. Konopka, Ser. Ed. (Elsevier Science, New York, NY).

Address reprint requests to:
Stephen Yau
Department of Mathematics, Statistics and Computer Science
University of Illinois at Chicago
Chicago, IL 60607-7045

E mail: yau@uic.edu

Received for publication February 5, 2007; received in revised form February 16, 2007; accepted March 5, 2007.