Natural Vector Method of Characterizing, Clustering and Phylogeny of DNA, Genome and Proteins

BY

Mo Deng
B.A. (Northwest Normal University) 2003
M.S. (East China Normal University) 2006

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Mathematics
in the Graduate College of the
University of Illinois at Chicago, 2011

Chicago, Illinois

Defense Committee:
Stephen Yau, Chair and Advisor
David Nicholls
Jie Yang
Jan Verschelde
Clement Yu, Department of Computer Science

To my families:

my parents Jishan Deng and Gaihua Jia,

and my wife Yidan Chen

ACKNOWLEDGMENTS

I am heartily thankful to my advisor, Stephen Yau, whose encouragement, guidance and support from the initial to the final level enabled me to develop an understanding of the subject. I am grateful to Professor Jie Yang who helps me a lot improve my projects. I appreciate Professor Jing Wang, Professor Gyorgy Turan, and Professor Jiasong Wang from Nanjing University for their great assistance. I am indebted to Chenglong Yu, Postdoc at Institute of Mathematical Sciences, The Chinese University of Hong Kong, Dr. Lu Zheng at Carnegie Mellon University. Part of this thesis grew out of some joint work with them. I appreciate Professor David Nicholls, Professor Jan Verschelde and Professor Clement Yu for being my committee members. I also want to give appreciation to my friends and colleagues Dr. Changchuan Yin, Dr. Eb Armah, Dr. Bo Zhao, Mr. Yang Jiao, and so on.

Moreover, I would like to show my gratitude to my parents, Jishan Deng and Gaihua Jia, and my parents-in-law, Tiangui Chen and Lihua Wang, for their tremendous supports, both emotionally and financially.

My greatest thanks goes to my wife, Yidan Chen who achieved her master degree from UIC. During the past four years, she contributed a lot to me and the family. Her love and support have made my life exceptional. I am really lucky to have such a beautiful partner.

Lastly, I offer my regards and blessings to all of those who supported me in any respect during the completion of this thesis.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (Continued)

FIGURE **PAGE**

LIST OF ABBREVIATIONS

DNA	Deoxyribonucleic acid
RNA	Ribonucleic acid
ZC	Z-curve and Cumulative Matrix
HRV	Human Rhinovirus
PKC	Protein Kinase C
MSA	Multiple Sequence Alignment
NV	Natural Vector

SUMMARY

With the development of biotechnology, more and more biological sequence information has been acquired. The number of sequences in GenBank has been growing exponentially in the past 20 years (http://www.ncbi.nlm.nih.gov). There are almost 8 million sequences in non-redundant (NR) database of protein sequences, including the complete genomes of 1800 different species. This large body of data is doubling in size every 28 months. Many computational and statistical methods for the comparison of biological sequences (DNA, genome or protein sequences) have been proposed. It still remains one of the most active and important research areas in bioinformatics and computational biology.

Two different methodologies for studying the sequence comparison (i.e., the similarity of sequences) are known as alignment-based and alignment-free methods. Alignment-based method is widely used by scientists. However, the search for optimal solutions using sequence alignment-based methods is encountered with great difficulty in computational aspect with regard to large biological databases, especially when comparing three or more biological sequences at a time, i.e., multiple sequence alignment. In fact, multiple sequence alignment is an NP-hard problem. Therefore, it is very necessary to develop the alignment-free approaches to overcome the critical limitations of alignment-based methods.

In chapter 2, we introduce an alignment-free approach, natural vector method, to characterize a biological sequence as a natural vector. We mathematically prove that the sequence and its natural vector are in one-to-one correspondence. This contribution allows us to embed the biological sequence space as a subspace of Euclidean space. More importantly, natural vector method is much faster and more accurate than the-state-of-the-art methods. Therefore, we can globally compare all the existing DNA, genome sequences within the space in a very short time whereas the conventional multiple alignment methods can never achieve it. In addition, the evolutionary properties of an unknown DNA, genome sequence can be predicted in the existing space by simply computing its associated natural vector. We conduct our method on new outbreak of A (H1N1) genes and genomes, human rhinovirus gnomes (HRV), mammalian mitochondrial genomes. The results indicate that natural vector method is much faster and more accurate than the-state-of-the-art methods in specifying the homology of DNA or genome sequences.

Secondly, we use the natural vector method to construct the protein space, a subspace of Euclidean space in chapter 3. Similarly, we can prove the one-to-one correspondence between a protein sequence and its natural vector.

In chapter 3, we use the natural vector method to reconstruct the phylogenetic tree for protein sequences. As illustration, protein kinase C (PKC) family and beta globin family sequences are tested based on their 60-dimensional natural vectors.

In chapter 4, we introduce a novel method, ZC-method, to classify intron-less sequences from intron-containing sequences. This proposed method extends Z-curve method and improve the accuracy significantly. In applications, we test the original data of Z-curve method and another large dataset by using ZC-method and other state-of-the-art methods including Genscan, N-scan and Z-curve. The result shows the proposed ZC-method is more accurate than others.

CHAPTER 1

INTRODUCTION

1.1 Conventional Methods in Reconstructing The Phylogenetic Trees for Nucleotide Level Sequences

About one hundred and fifty years ago, Charles Robert Darwin(1) published the famous theory called "natural selection" in his book "On the Origin of Species". He claimed that all species of life on earth are descended from common ancestors based on the geographical distribution of wildlife and fossils. The idea and a simple draft of the phylogenetic tree were also provided in his book. Later, with the development of evolutionary biology, it is necessary to find a new practicable method to construct the phylogenetic tree automatically instead of using fossils manually. A simple phylogenetic tree of human races is depicted in Figure 1.

It is well known that DNA is transferred from organisms to their offspring. During the transmission, A few changes always take place in DNA sequences. Hence, the organisms who share a lineage and are descended from a common ancestor must have more similar DNA, RNA and protein sequences. In 1977, Carl Woese(2) firstly analyzed the phylogenetic relationship based on 16S ribosomal RNA, which became a standard for the research of evolutionary biology based on genetic sequences. Later, two methodologies for studying the sequence comparison (i.e., the similarity of sequences) were created, alignment-based and alignment-free methods.

For the alignment-based method, there are two major approaches to infer the phylogenetic

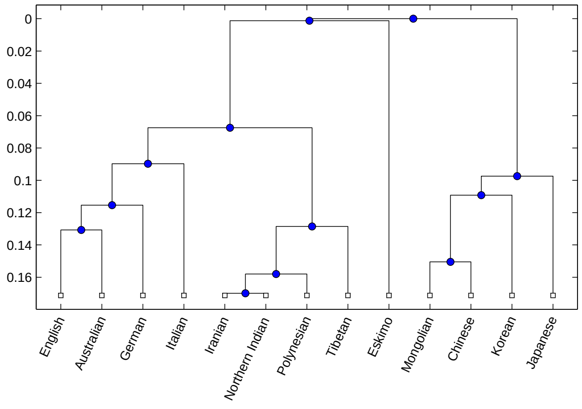

Figure 1. A simple phylogenetic tree of some human race

tree: distance matrix method and maximum likelihood method. Currently, two methods (after the alignment), neighbor-joining (distance matrix method) and maximum likelihood are always used to implement the evolutionary relations among species. Neighbor-joining calculates genetic distance from multiple sequence alignment (MSA), thus it assigns a model to calculate the distance matrix. Maximum likelihood method applies an explicit evolutionary model to infer the phylogenetic tree estimation. In fact all these existing methods for phylogenetic inference require the multiple alignment of the sequences and assume some sort of models. Most common models of DNA evolution are JC69 model (3), K80 model (4), F81 model (5), HKY85 model (6), GTR model (7), T92 model (8), TN93 model (9), etc. There are many multiple sequence alignment computer programs available, such as Clustal(23), Muscle(24) and MAFFT(25). Those programs can yield good alignment results and create an accurate phylogenetic tree. However, the search for optimal solutions using sequence alignment-based methods is encountered with great difficulty in computational aspect with regard to large biological databases, especially when comparing three or more biological sequences at a time, i.e., multiple sequence alignment. In fact, multiple sequence alignment is an NP-hard problem. Moreover, human intervention is required beforehand to choose a model to reconstruct the phylogenetic tree. The different choices of evolutionary models totally yield different results. In other words, these results are sometimes inconsistent. In addition, the accuracy of current methods for reconstructing phylogenetic tree is positively correlated with alignment accuracy (measured using SP-scores). The fact is that all current methods for multiple alignments have high error rates when sequences evolve with many indels (insertions and deletions) and substitutions. Therefore, it is very necessary to develop the alignment-free approaches to overcome the critical limitations of alignment-based methods.

Many alignment-free methods have been reported to analyze the large volume of DNA sequences. One of them is the graphical representation of sequences, which provides a simple way of viewing, sorting, and comparing various gene structures. One of the first attempts towards developing a graphical technique for representing DNA sequences was from Hamori (52), who used a three-dimensional H curve to represent a DNA sequence. However, sophisticated computer graphic tools are needed to generate this H curve, which limits its widespread use. Later, Gates (53) proposed a two-dimensional graphical representation that is simpler than H curve. However, Gates' graphical representation has high degeneracy caused by path overlapping and crossing itself. For example, the sequences ACTG, ACTGA, ACTGAC, etc have the same graphical representation. In mathematical terms, the sequence degeneracy forms repetitive closed loops or circuits in the DNA graph. Many efforts have been made to avoid it (54; 19). Yau et al (18) in 2003 reported a new two-dimensional graphical representation of gene sequences. This method is not natural because artificial values were assigned to each nucleotide base. For example, $(\frac{1}{2}, -\frac{\sqrt{3}}{2}) \rightarrow A$.

These motivate us to create a novel method without these drawbacks. We introduce a new method–natural vector method in chapter 2 and 3, which can reconstruct the accurate phylogenetic tree in linear time and much faster than the multiple sequence alignment programs and much accurate than alignment-free methods. In our natural vector method, we associate each DNA sequence a natural vector, a specific mathematical description of distributions of nucleotides in DNA governing all biological information encoded by DNA sequence. The construction of natural vectors will be introduced in next chapter 3.

1.2 Protein Universe

The protein universe, a concept first mentioned in 1992 (47), is the collection of all proteins of every biological species that lives or has lived on earth. According to recent PNAS paper by M. Levitt (48), there are almost 8 million sequences in non-redundant (NR) database of protein sequences, including the complete genomes of 1800 different species. This large body of data is doubling in size every 28 months. Coming to understand with the protein universe is unarguably central (48; 49). Thus, researchers are using the evolutionary relatedness of all life on earth to understand the protein universe (48; 50). Protein universe is a poorly defined and mysterious entity to scientists. An obvious way to reveal the nature of the protein universe is to cluster sequences into families by similarities (measured, say, by the percentage of identical amino acids when suitably aligned). Appreciable levels of similarity generally imply homology or descent from a common ancestor, which allows related sequences to be grouped into families (51). Here we approach the problem differently. We use the natural vector method to realize the protein universe as a subspace of Euclidean space, which we call protein space. Then we can globally compare all the existing proteins within this protein space in a short time with high accuracy. As applications, we perform our method on protein kinase C (PKC) family dataset and beta globin dataset. The results show that natural vector method on protein sequences is more efficient and accurate than state-of-the-art methods.

1.3 Intron-containing and Intron-less Sequences

Deoxyribonucleic acid (DNA) is a nucleic acid which carries genetic information for the biological development of all cellular forms of life and many viruses, which consists of two long strands with the double helix structure. Each strand has the direction from 5' end to 3' end and consists of four type nucleotides including Adenine (A), Guanine (G), Cytosine (C), and Thymine (T). Adenine pairs with Thymine and Cytosine pairs with Guanine to form the double helix structure.

The main biological function of DNA sequences is that it can be encoded into protein sequences via RNA transcription. However, not the whole DNA sequences are encoded to protein sequences. Some regions are removed when the DNA sequences are transcribed into RNA, which we call Introns. The regions transcribed into RNA are called Exons or Intron-less sequences. Exons are encoded into the protein sequences except the untranslated regions, which are important for efficient translation of the transcript and for controlling the rate of transla-

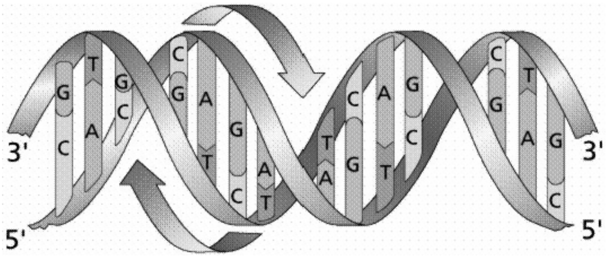

Figure 2. The double helix structure of DNA

tion. An intron is a DNA region within a gene that is not translated into protein. Introns are always call as "Junk" DNA which will be removed from DNA when transcribing into RNA. Therefore, it is a very important to create methods to distinguish an intron-less (exon) sequence from an intron-containing sequence. Figure 3 provides an example of an intron-containing and intron-less sequences.

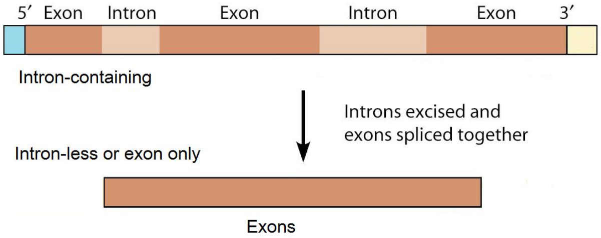

Figure 3. Intron-containing and Intron-less sequences

Distinguishing intron-containing and intron-less (exon) DNA sequences is an important topic in Bioinformatics. Intron-less and intron-containing genes have different biological properties and statistical characteristics. Congruent with the Spearman's rank correlation, the comparison of intron-less and intron-containing genes showed significantly reduced expression for intron-less genes when compared to intron-containing genes (57). These observations raised interesting questions about the role of intron-less (exon) and intron-containing genes. On the other hand, Peng et al (58) have discovered that long-range correlation existed in the intron-containing genes, but did not exist in the intron-less genes. This work was based on simple random-walk model of DNA sequences, in which a pyrimidine led to a step up and a purine a step down. Consequently, the walk resulted in a definite landscape for a given sequence and only one parameter was calculated based on the landscape. This parameter was proposed to distinguish between the intron-containing and intron-less genes. However, further study showed that this finding cannot be used as a general method to identify intron-less genes (59; 60). They pointed out that there were some basic drawbacks in the work of Peng et al. First, the DNA sequence cannot uniquely be described by the pyrimidine & purine walk. Secondly, although Buldyrev et al (62) considered six other possible walks besides the pyrimidine & purine walk, the cross-correlations between any two walks were totally ignored. Zhang et al (60) introduced a Z-curve consisting of three parameters, in which the cross-correlations between any two parameters of the Z-curve were considered. As an application, they used the Z-curve method to classify a dataset consisting of 100 intron-containing and 100 intron-less genes. The discriminant accuracy as high as 89.0% can be obtained by using Fisher's linear discriminant algorithm. Although the distributions of three different biological types were displayed in Z-curve, it did not reveal the cross-correlations of distances between the nucleic bases, which are also important parameters to classify genes into intron-containing and intron-less. In recent years, a number of methods have been developed for DNA/protein clustering or classification, gene prediction and exon/intron parsing. However, all these methods can provide predictions for splice sites of exons. Thus, these gene-finding models or gene-parsing systems provided a prediction of precise (predicted) splice sites of the exons/introns in the gene, while also producing the intron-bearing status of a gene. Our problem is for classifying genes as to their intron-bearing status only in order to improve the prediction accuracy. In Chapter 4, we will introduce our new ZC-method, which can classify intron-containing sequences from intron-less sequences with higher accuracy than GENSCAN, N-SCAN and Z-curve methods.

CHAPTER 2

NATURAL VECTOR METHOD TO CLUSTER AND RECONSTRUCT THE PHYLOGENETIC TREES FOR HOMOLOGOUS GENOMES

2.1 Introduction

Computational and statistical methods to cluster the DNA or protein sequences have been successfully applied in clustering DNA, protein sequences and microarray data (10; 11; 12; 13; 14; 15; 16; 17). Yau and his group showed that the method of genomic space was an efficient way to cluster the DNA or protein sequences (18; 19; 20; 21; 22). In the papers (18; 20) each nucleic base or amino acid was assigned a specific value. For example, nucleic base adenine A was assigned to a pair $(\frac{1}{2}, -\frac{\sqrt{3}}{2})$ (18). However, that method is not natural although it can be used successfully to represent a DNA sequence in the Cartesian coordinate plane. Here ”natural” means that the parameters used are inherently attached to the DNA or protein sequences. The parameters used in this work are based on the numbers and distributions of nucleotides in the sequence, which is a natural way to describe these sequences.

We propose a method of characterizing DNA sequences, which indicates that a specific mathematical description of distributions of nucleotides in a DNA sequence can govern all biological information. To each DNA sequence we associate a natural sequence of parameters, called a natural vector, describing the numbers and distributions of nucleotides in the sequence. We show that the correspondence between a natural vector and a DNA sequence is one-to-one. A natural distance between two genes is the distance between their corresponding natural vectors. This creates a genome space with biological distance, which allows us to do phylogenetic analysis in the most natural and easy manner. This alignment-free method is much faster than conventional multiple sequence alignment methods. Multiple sequence alignment (MSA) can be seen as a generalization of pairwise sequence alignment instead of aligning two sequences, k sequences are aligned simultaneously. MSA is the most powerful method to analyze the genetic sequences and most of state-of-art algorithms are constructed based on it (23; 24; 25), it is however an NP-hard computational optimization problem which is implausible for a huge amount of sequences (26). The complexity is $O(n^k)$ given k sequences with length n.

2.2 Construction of Natural Vector for DNA Sequence

A natural vector of a DNA sequence is composed of a series of numerical numbers which describe the statistics of the nucleotides in the sequence.

Let us first introduce the definition of normalized central moments which is the most important part of natural vector method. Normalized central moments are defined as follows:

$$D_j^k = \sum_{i=1}^{n_k} \frac{(s[k][i] - \mu_k)^j}{n_k^{j-1} n^{j-1}}, j = 1, 2, \ldots, n_k. \tag{2.1}$$

where $k = A, C, G, T$. Here, n_k denotes the number of each nucleic base k of the DNA sequence and n is the total length of DNA sequence. $s[k][i]$ is the distance from the first nucleotide (regarded as origin) to the ith nucleotide k in the DNA sequence. $T_k = \sum_{i=1}^{n_k} s[k][i]$ denotes the total distance of each set of A, C, G, T to the origin, $k = A, C, G, T$. $\mu_k = \frac{T_k}{n_k}$ is the arithmetic mean of the distance of each nucleic base. Therefore, we have the sequences of central moments: $< D_1^A, D_1^C, D_1^G, D_1^T, \ldots, D_{n_A}^A, D_{n_C}^C, D_{n_G}^G, D_{n_T}^T >$. We note that for each $k = A, C, G, T$,

$$D_1^k = \sum_{i=1}^{n_k}(s[k][i] - \mu_k) = T_k - n_k(T_k/n_k) = 0. \tag{2.2}$$

Thus, The normalized central moments start from the second moment D_2^k. That is, $< D_2^A, D_2^C, D_2^G, D_2^T, \ldots, D_{n_A}^A, D_{n_C}^C, D_{n_G}^G, D_{n_T}^T >$. The second normalized central moment describes the variance of the distance distribution for each base:

$$D_2^k = \sum_{i=1}^{n_k} \frac{(s[k][i] - \mu_k)^2}{n_k n}, \text{ where } k = A, C, G, T. \tag{2.3}$$

Our method described below is to give a complete understanding of the distribution of four nucleotides A, C, G and T.

(1) The quantities of four nucleotides: A, C, G and T of a DNA sequence are chosen as the first four parameters of the natural vector. Four integers n_A, n_C, n_G and n_T denote the numbers of nucleic bases A, C, G and T in the DNA sequence.

(2) The second group of numerical parameters are the mean value of total distance for each of the four nucleotide bases: $\mu_k = \frac{T_k}{n_k}$, $k = A, C, G, T$. As a simple illustration of a DNA sequence GTTCAATACT. Total distance of adenine A is $T_A = 4 + 5 + 7 = 16$, since the distance of origin to the three nucleotide As is 4,5 and 7, repectively. Then $\mu_A = \frac{T_A}{n_A} = \frac{16}{3}$. The arithmetic mean value of total distance for other nucleotide base G, C and T can be obtained in the same way.

(3) The final group of parameters that we include in the natural vector are composed of normalized central moments. The first normalized central moment is 0 (equation 2.2), thus we start with the second normalized central moment. The second normalized central moment is the variance of the distance distribution for each base (equation 2.3). If the distribution of each nucleotide base is different, DNA sequences cannot be the same even though they may have the same nucleotide contents and the same total distance measurement. Therefore, the information about the distribution has also been included in the natural vector. As described above, each subset of numerical parameters is not sufficient to annotate DNA sequences. However, the combined numerical parameters are sufficient to characterize each DNA sequence. So the natural vector is given as follows:

$$< n_A, \mu_A, D_1^A, \ldots, D_{n_A}^A, n_C, \mu_C, D_1^C, \ldots, D_{n_C}^C, n_G, \mu_G, D_1^G, \ldots, D_{n_G}^G, n_T, \mu_T, D_1^T, \ldots, D_{n_T}^T > . \tag{2.4}$$

In order to express the vector elegantly, we rewrite it as follows:

$$< n_A, n_C, n_G, n_T, \mu_A, \mu_C, \mu_G, \mu_T, D_1^A, D_1^C, D_1^G, D_1^T, \ldots, D_{n_A}^A, D_{n_C}^C, D_{n_G}^G, D_{n_T}^T > . \tag{2.5}$$

Alternatively, the natural vector can be written as

$$< n_A, n_C, n_G, n_T, \mu_A, \mu_C, \mu_G, \mu_T, D_1^A, D_1^C, D_1^G, D_1^T, \ldots, D_{n_\pi}^A, D_{n_\pi}^C, D_{n_\pi}^G, D_{n_\pi}^T > . \tag{2.6}$$

where $n_\pi = max\{n_A, n_C, n_G, n_T\}$. By the definition, $D_j^k = 0$, if $j > n_k$. For instance, this case happens when we compute $< D_{20}^A, D_{20}^C, D_{20}^G, D_{20}^T >$ if there are 19 A, 15 C, 21 G and 22 T in the sequence. Therefore, the 20th moment will be $< 0, 0, D_{20}^G, D_{20}^T >$. The natural vector is obtained by concatenating the first group of parameters (the number of each base) and the second group of parameters (the mean value of total distance of each base) to the normalized central moments.

In the next section, we will prove that the higher moments converge to 0 for a random generated sequence or a biological sequence.

2.3 Construction of Natural Vector for Genomes

We have already obtained a good numerical characterization (natural vector) to represent a DNA sequence. Now we will use this tool to construct a natural vector for genomes. It is known that the structure of genomes is very complicated. It may be single-stranded or double-stranded, and in a linear, circular or segmented structure. Thus, we should consider the different structures when constructing the natural vector for genomes. For the simplest genome structures, linear single-strand forms, we can treat them as linear DNA sequences. That is, every genome corresponds to a general DNA sequence. Thus, we can utilize our method to construct the natural vector for genomes. In order to use whole genome information to make comparative analysis among genomes, we can use the first N dimensional natural vector

$$< n_A, n_C, n_G, n_T, \mu_A, \mu_C, \mu_G, \mu_T, D_1^A, D_1^C, D_1^G, D_1^T, \dots, D_S^A, D_S^C, D_S^G, D_S^T > \tag{2.7}$$

of the whole natural vector of a genome sequence to represent a genome where $S = \frac{N}{4} - 1$. Thus, using the Euclidean distance between each pair of vectors for comparison, we can perform phylogenetic and clustering analysis for genome sequences.

For the circular single-strand genomes, the construction of natural vector of a genome is more complicated because we do not know which point is the starting point in this circular DNA sequence. In this case, we treat every point as the starting point in this circular sequence of length n, and then we get n linear single-strand genomes. For every linear single-strand genome sequence, we can compute its $(n+4)-$dimensional natural vector. Then we take average to get a normalized vector. For circular single-strand genomes, we use the first N dimensional natural vector (as shown above) of this normalized vector of each genome to do clustering and phylogenetic analysis. For the double-stranded genomes, we need to state that the natural vector of reverse complementary sequence is not the same as the original sequence. Generally, for the double-stranded genomes, we treat them as two single-stranded genomes. We use the above method (linear or circular) to get two $(n+4)-$dimensional natural vectors for these two single-stranded sequences, and then take average for them to get a general natural vector

$$< n_A, n_C, n_G, n_T, \mu_A, \mu_C, \mu_G, \mu_T, D_1^A, D_1^C, D_1^G, D_1^T, \ldots, D_{n_A}^A, D_{n_C}^C, D_{n_G}^G, D_{n_T}^T > . \tag{2.8}$$

By using the first N dimensional natural vector of this general natural vector for a genome, we can do clustering analysis for those genomes. Here we need to point out that the two strands of some genomes (e.g. some bacterial genomes) are differentiated by their nucleotide content, which are called the heavy strand and the light strand respectively. The two strands have different masses because one has a higher proportion of heavier nucleic acids and its complement has a lower proportion. In this case, we just treat them as the single-stranded (by using the heavy strand) genomes to construct the natural vectors. For a genome consisting of k segments, we compute the natural vector for each DNA and then concatenate these k natural vectors to represent the natural vector for a segmented genome. For example, each segmented influenza A (H1N1) genome consists of 8 segments. So we can compute the natural vector for each segment and then concatenate these 8 natural vectors together. In our experiment, a 12-dimensional natural vector can characterize each A (H1N1) segment, then 96-dimensional natural vector can be used to characterize an influenza A (H1N1) genome.

2.4 The Proof of Corresponding Theorem

Theorem 1: Higher moments in the natural vector of a sequence converge to 0.

Proof. By the equation (2.1),

$$D_j^k = \sum_{i=1}^{n_k} \frac{(s[k][i]-\mu_k)^j}{n_k^{j-1} n^{j-1}} \leq \sum_{i=1}^{n_k} \frac{max_i |s[k][i]-\mu_k|^j}{n_k^{j-1} n^{j-1}}$$

$$= \frac{max_i |s[k][i]-\mu_k|^j}{n_k^{j-2} n^{j-1}} \leq \frac{n^j}{n_k^{j-2} n^{j-1}} = \frac{n}{n_k^{j-2}}$$

It is clear that $n_k \geq 2$, otherwise, $s[k][i] - \mu_k = 0$. On one hand, from the viewpoint of probability, we suppose that the expectation value of any nucleic base $n_k = n/4$ (uniform distribution) for a sequence with given length n, therefore

$$\lim_j \frac{n}{n_k^{j-2}} = \lim_j \frac{n}{(n/4)^{j-2}}$$

$$= \lim_j \frac{n \cdot 4^{j-2}}{n^{j-2}} = \lim_j \frac{4^{j-2}}{n^{j-3}}$$

Obviously, this limit goes to 0 as j approaches n_k. □

On the other hand, we can simply discuss the number n_k from GC-content. GC-content (or guanine-cytosine content) in molecular biology, is the percentage of nitrogenous bases on a DNA molecule which are either guanine or cytosine. GC-content is found to be variable with different organisms. Because of the nature of the genetic code, it is however virtually impossible for an organism to have a genome with a GC-content approaching either 0% or 100%. A species with an extremely low GC-content is Plasmodium falciparum (GC%=∼ 20%)(27). Therefore, for any simulated dataset or biological dataset, D_j^k always converges to 0 when j approaches n_k. For a simulation, we generate a random sequence with length of 10000 nucleotides by using Hidden Markov model (Matlab, bioinformatic toolbox). The simulated sequence contains 2345 A, 2761 C, 2544 G and 2350 T. For adenines $D_4^A = 0.00182$, i.e., higher normalized central moments starting from 4th moment will converge to 0.

One of the most important things in this thesis is that we can prove that the correspondence between a DNA sequence and its natural vector is one-to-one.

Theorem 2: Suppose a DNA sequence has the number n of nucleotides. Then the correspondence between a DNA sequence and its natural vector

$$< n_A, n_C, n_G, n_T, \mu_A, \mu_C, \mu_G, \mu_T, D_1^A, D_1^C, D_1^G, D_1^T, \ldots, D_{n_A}^A, D_{n_C}^C, D_{n_G}^G, D_{n_T}^T > . \tag{2.9}$$

is one-to-one.

Proof. To prove the theorem, we first need prove that for any given proper DNA natural vector, we can recover the corresponding DNA sequence. That is, for a natural vector, n_k, μ_k are known, $k = A, C, G, T$. Then we need to locate the position of each nucleotide A, C, G and T in the sequence. We next concentrate on normalized central moments to derive the position of each nucleotide in the sequence. Let us first denote $z_{[k]i} = s[k][i] - \mu_k$, then the normalized central moments can be simplified as:

$$D_j^k = \sum_{i=1}^{n_k} \frac{z_{[k]i}^j}{n_k^{j-1} n^{j-1}}, j = 1, 2, \ldots, n_k, \tag{2.10}$$

where n is the total length of the DNA sequence. If we can find the value of each $z_{[k]i}$, then the position of each nucleotide A, C, G and T can be located and consequently the DNA sequence can be recovered. To solver for $z_{[k]i}$, let $\delta_{[k]j} = D_j^k n_k^{j-1} n^{j-1}$, then the $\delta_{[k]j}$ can be obtained by D_j^k and n_k (generally each $n_k \geq 2$). For a given natural vector, n_k is known for each nucleic base A, C, G and T. So we need to solve for $z_{[k]i}$ corresponding to one of the n_k. Clearly $\delta_{[k]j}$ and $z_{[k]i}$ have the relation as below:

$$\begin{cases} \delta_{[k]1} & = & z_{[k]1} & + & z_{[k]2} & + & \dots & + & z_{[k]n_k} \\ \delta_{[k]2} & = & z_{[k]1}^2 & + & z_{[k]2}^2 & + & \dots & + & z_{[k]n_k}^2 \\ \multicolumn{9}{l}{\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots} \\ \delta_{[k]n_k} & = & z_{[k]1}^{n_k} & + & z_{[k]2}^{n_k} & + & \dots & + & z_{[k]n_k}^{n_k} \end{cases}$$

$z_{[k]1}, z_{[k]2}, \dots, z_{[k]n_k}$ are roots of a polynomial. We choose nucleotide A as an illustration, thus

$$z^{n_A} + a_{n_A-1}z^{n_A-1} + \dots + a_1 z + a_0 = (z - z_1)(z - z_2)\dots(z - z_{n_A}). \tag{2.11}$$

Therefore, we can recover $z_1, z_2, \dots, z_{n_A}$ if we can solve each coefficient $a_0, a_1, \dots, a_{n_A-1}$ ($a_{n_A} = 1$). Let p_d ($d = 1, 2, \dots, n_A$) be the elementary symmetric polynomials in $z_1, z_2, \dots, z_{n_A}$,

$$p_1 = \sum_{1}^{n_A} z_i, p_2 = \sum_{i<j}^{n_A} z_i z_j, p_3 = \sum_{i<j<l}^{n_A} z_i z_j z_l, \dots, p_{n_A} = z_1 z_2 \dots z_{n_A} \tag{2.12}$$

Then by comparison of the coefficients in series

$$a_{n_A-1} = -z_1 - z_2 - \ldots - z_{n_A} = -p_1 \tag{2.13}$$

$$a_{n_A-2} = z_1 z_2 + z_1 z_3 + \ldots + z_{n_A-1} z_{n_A} = \sum_{i<j} z_i z_j = p_2 \tag{2.14}$$

$$\cdots \tag{2.15}$$

$$a_0 = (-1)^{n_A} z_1 z_2 \ldots z_{n_A} = (-1)^{n_A} p_{n_A} \tag{2.16}$$

I.e.,

$$p_1 = -a_{n_A-1}, p_2 = a_{n_A-2}, \ldots, p_{n_A} = (-1)^{n_A} a_0. \tag{2.17}$$

By using Newton's famous identities (28):

$$\delta_{n_A} - p_1 \delta_{n_A-1} + \ldots + (-1)^{n_A-1} p_{n_A-1} \delta_1 + n_A (-1)^{n_A} p_{n_A} = 0, \tag{2.18}$$

where $\delta_1 = \sum_{i=1}^{n_A} z_i, \delta_2 = \sum_{i=1}^{n_A} z_i^2, \ldots, \delta_{n_A} = \sum_{i=1}^{n_A} z_i^{n_A}$. Then each coefficient a_i of the polynomial can be obtained by δ_j as shown below:

$$\begin{cases} a_{n_A} & = & 1 \\ a_{n_A-1} & = & (-1)\delta_1 \\ a_{n_A-2} & = & \frac{1}{2}(\delta_1^2 - \delta_2) \\ a_{n_A-3} & = & (-1)^3 \frac{1}{6}(\delta_1^3 - 3\delta_1\delta_2 + 2\delta_3) \\ a_{n_A-4} & = & \frac{1}{24}(\delta_1^4 - 6\delta_1^2\delta_2 + 3\delta_2^2 + 8\delta_1\delta_3 - 6\delta_4) \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \end{cases}$$

I.e., $a_{n_A} = 1, a_{n_A-1} = -p_1, a_{n_A-2} = p_2, \ldots$. As a result, the coefficients of the polynomial

$$a_0 + a_1 z + a_2 z^2 + \ldots + z^{n_A} = (z - z_1)(z - z_2)\ldots(z - z_{n_A})$$

can be obtained from the natural vector and thus the set of all roots can be obtained.

Next we need identify each root $z_1, z_2, \ldots, z_{n_A}$. Clearly, $z_i - z_{i+l} = s[A][i] - s[A][i+l]$. For any $l > 0$, $s[A][i] < s[A][i+l]$ by the definition of $s[k][i]$. Then $z_i < z_{i+l}$. As a consequence, z_i is strictly increasing and each root can be identified by this property. That is, each value of $s[A][i]$ can be obtained. Therefore, the position of each nucleotide A can be obtained. We can locate all positions of all nucleotides C, G and T in the same way.

The unique corresponding sequence can be recovered from a given natural vector.

On the other hand, given a DNA sequence, the number of each nucleic base and the distance of each nucleic base to the origin are determined. Based on this information, we can calculate T_k, μ_k. So it is easy to construct a natural vector. Obviously, any two different sequences are distinct in length, nucleotide content or nucleotide arrangement. Therefore, these natural vectors are completely different. $\square$

So far we have proved that the correspondence between a DNA sequence and its natural vector is in one-to-one. Now let us define the biological distance between two sequences as the Euclidean distance between two corresponding natural vectors.

Given two natural vectors k, k':

$$k =< n_A, n_C, n_G, n_T, \mu_A, \mu_C, \mu_G, \mu_T, D_2^A, D_2^C, D_2^G, D_2^T, \ldots, D_{n_A}^A, D_{n_C}^C, D_{n_G}^G, D_{n_T}^T >, \tag{2.19}$$

$$k' =< n'_A, n'_C, n'_G, n'_T, \mu'_A, \mu'_C, \mu'_G, \mu'_T, D_2'^A, D_2'^C, D_2'^G, D_2'^T, \ldots, D_{n_A}'^A, D_{n_C}'^C, D_{n_G}'^G, D_{n_T}'^T > . \tag{2.20}$$

The Euclidean distance between them is

$$L(k, k') = \sqrt{\sum_j \sum_i (j_i - j'_i)^2}, \tag{2.21}$$

where $j = n, \mu, D_2^i, \ldots, D_{n_i}^i$, $i = A, C, G, T$.

Note that the first normalized central moment is

$$D_1^k = \sum_{i=1}^{n_k} (s[k][i] - \mu_k) = T_k - n_k(T_k/n_k) = 0.$$

Therefore, we always start from the second normalized central moment.

2.5 Distribution of Pair-wise Distance L of Natural Vectors Under Simulation

It is an interesting problem to know the distribution of distance L of natural vectors for random sequences. This is performed by taking any sequence and generating a large number of shuffles of the sequence so that the total numbers of each nucleotide base are preserved. For example, we select a human mitochondrial genome(GenBank ID: V00662), and then shuffle it 1000 times so that the total numbers of A, C, G, T are preserved. We first use 12 dimensional natural vectors. Therefore, 500500 different Ls will be generated, and then we plot the histogram of those Ls and estimate this probability distribution directly from the Ls by plotting a density curve. The histogram shows the distribution of frequency of those Ls. When we take 12 or 16 dimensional natural vectors to draw the histogram, there is no obvious differences between those two histograms, which means 12 dimensional natural vectors are stable. The histogram is shown here(Figure 4).

2.6 Simulation of Gene Rearrangements

Gene rearrangements play important roles in evolution. The order of genes and transcription regions are changed during evolution by gene rearrangements such as DNA inversions and transpositions, which do not affect the gene content of the chromosomes. For example, inversions of large genomic fragments are often observed even between closely related species.

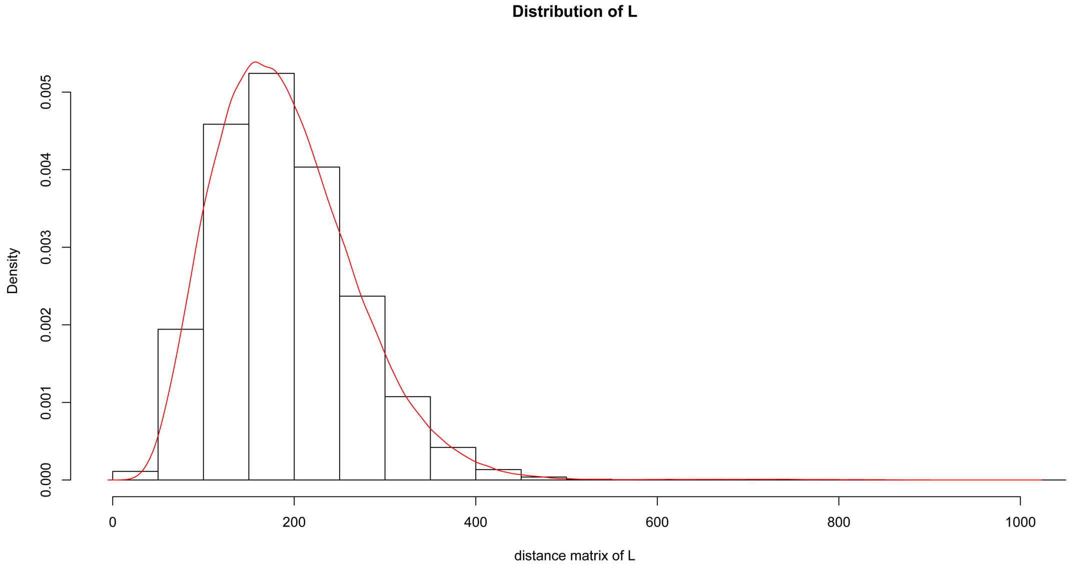

Figure 4. The distribution of distance L for random sequences.

The phylogenetic analysis based on gene order is challenging as it requires detailed complex gene order data in genomes and intensive computation. In order to test that our method is stable for genomic rearrangement and biologically meaningful, we consider the following simulated experiment. We choose a human mitochondrial genome denoted by human, and then inverse its two genes ATPase 6 and Cytochrome oxidase to get a simulated genome, which is denoted by human-inv. We can treat this new simulated genome as the result of the inversion of genes from the original human mitochondrial genome. Next we randomly generate a genome sequence which has the same length and nucleotide content as the original human genome, which we denote by human-ran. Thus, we have 3 genomes of the same length and nucleotide content: human, human-inv and human-ran. In addition, we also choose another mitochondrial genome, chimpanzee as comparison since chimpanzee and human are evolutionarily close. By using the 16-dimensional natural vector, we calculate the distance among these 4 genomes and get a distance matrix in Table I. we found that the distance between human and human-inv is as small as 1.7 units. This means that the gene-inverted genome still has a very short distance to the original genome even if some gene rearrangement happens in this genome. As a result, the original genome and its gene rearrangement genome cannot be treated separately by using our method since the evolutionarily close genome of human is chimpanzee which has the distance of 16.88 units to human. More importantly, this simulation demonstrates that our method can be applied to do clustering or phylogenetic analysis. If two genetic sequences are close in the distance, they should be close in the evolutionary tree. For example, the distance between real human genomes and the genome of chimpanzee is 16.88 units. Since the distance between the original human genome and all shuffled genomes ranges from 114.49 to 1009.00 with the mean value of 190.60 units (Figure 4), all those randomly shuffled sequences cannot be clustered together with the human.

Therefore, natural vector method can be used to cluster the biological species even if they are only different in gene arrangement.

TABLE I: DISTANCE MATRIX OF SIMULATION OF GENE REARRANGEMENT

	human	human-inv	human-ran
human-inv	1.717323		
human-ran	127.27331	128.14110	
chimpanzee	16.87587	17.047305	104.02812

2.7 Gene Deletion and Translocation

In genetics, a deletion (also called gene deletion, deficiency, or deletion mutation) is a mutation (a genetic aberration) in which a part of a chromosome or a part of a DNA sequence is missing. Deletion is the loss of genetic material. Any number of nucleotides can be deleted, from a single base to an entire piece of chromosome. Small deletions are less likely to be fatal; large deletions are usually fatal - there are always variations based on which genes are lost. Most medium-sized deletions lead to recognizable human disorders.

Gene translocation (also called chromosomal translocation) means the movement of a gene fragment from one location to another, which often alters or abolishes expression. Like gene deletion, translocation affects genetic changes which lead to the disease.

Since the number of nucleotides of a gene to be deleted is random, we can design a simulation to test our method on a genome and its variation with some deletion. We choose norway rat complete mitochondrial genome (Rat) as an example. This mitochondrial genome has the length of 16300 bp. Now the whole piece of tRNA-encoding gene "tRNA-Glu" is deleted, then the resulting genome is named as "Rat-del". So we relocation this gene from the original position to the position between coding gene "NADH subunit 1" and gene "tRNA-Ile". The resulting genome is called as "Rat-trs". The last genome, "Rat-ran" is randomly generated from original norway rat.

By using the 16-dimensional natural vector, we calculate the distance among these 4 genomes and get a distance matrix in Table II.

TABLE II: DISTANCE MATRIX OF GENE DELETION AND TRANSLOCATION

	Rat	Rat-del	Rat-trs
Rat-del	2.91056		
Rat-trs	1.87739	2.33671	
Rat-ran	339.05662	321.45721	330.25188

From the Table II, a single gene deletion or translocation does not affect a genome too much. However, the natural vector of the genome may be changed a lot if many genes are deleted or translocated since the variation of this species will have a very different structure. In this case, our method can also be used to detect which individual genome has been changed by deleting or translocating many genes.

2.8 Results

2.8.1 Clustering Analysis of Influenza A H1N1 Genomes

As applications for our method, we first use our method to analyze the new influenza A (H1N1) virus. Recent reports of widespread transmission of swine-origin influenza A (H1N1) viruses in humans in Mexico, the United States, and elsewhere, highlighted this ever-present threat to global public health (29). Much effort has been made by using the experimental method and many important results have been obtained in the past (30; 31). Pigs have been hypothesized to act as a mixing vessel for the reassortment of avian, swine, and human influenza viruses and might play an important role in the emergence of novel influenza viruses capable of causing a human pandemic (32; 33; 34). There were many reports of recent transmissions of swine influenza viruses in humans (35). The new strain was initially described as triple reassortants of viruses from pigs, humans, and birds, called triple-reassortant swine influenza A (H1) viruses, which have circulated in pigs for more than a decade (29). Subsequent analysis suggested it was a reassortment of just two strains, both found in swine (31). Although initial reports identified the new strain as swine influenza (i.e., a zoonosis originating in swine), its origin is unknown from the point of view of whole genomes. Here we used our proposed method to verify the origin of A (H1N1) genomes. To demonstrate that our natural vector can be truly useful for answering biological questions, we performed hierarchical clustering analysis on the natural vectors of the genes of the swine influenza A (H1N1) virus. The Euclidean distance was used to measure the distance between natural vectors. Genomes of the outbreak of swine influenza A (H1N1), North American and Eurasian swine influenza virus genomes, avian and human seasonal influenza virus genomes were analyzed here. Each complete genome contains 8 complete gene-coding segments. So we use 96-dimensional natural vector to represent a whole genome since each segment can be characterized very well by using a 12-dimensional natural vector. Based on our novel mathematical method and the result, we can predict that the genome of new swine influenza A (H1N1) is similar to swine viruses rather than human seasonal influenza and avian viruses. Using the natural vector and hierarchial clustering method, we have reconstructed the complex reassortment history of the outbreak of swine influenza A (H1N1), summarized in Figure 5.

Our analysis shows that the swine influenza A (H1N1) genome is nested within a well-established triple-reassortant swine influenza A and Eurasian swine influenza A lineage (that is, a lineage circulating primarily in swine before the current outbreak). Natural vector method and hierarchical clustering method are used to reconstruct the phylogenetic tree for nucleotide sequences of the whole genome sequences of selected influenza viruses. The selected viruses are chosen to be representative from among all available relevant sequences in GenBank. The data can be found in Table III. Sequences have both high and low divergence to avoid biasing the distribution of branch lengths. Strains are representative of the major gene lineages from different hosts. The robustness of individual nodes of the tree is assessed using a bootstrap resampling analysis with 1000 replicates shown in Figure 6. From this figure, we can clearly see that new influenza A (H1N1) viruses originate from North American triple-reassortant swine virus and Eurasian classical swine virus lineage. In addition, we also analyzed 8 segments polymerase PB2, PB1, PA, hemagglutinin HA, neuraminidase NA, nucleocapsid NP, matrix protein MP

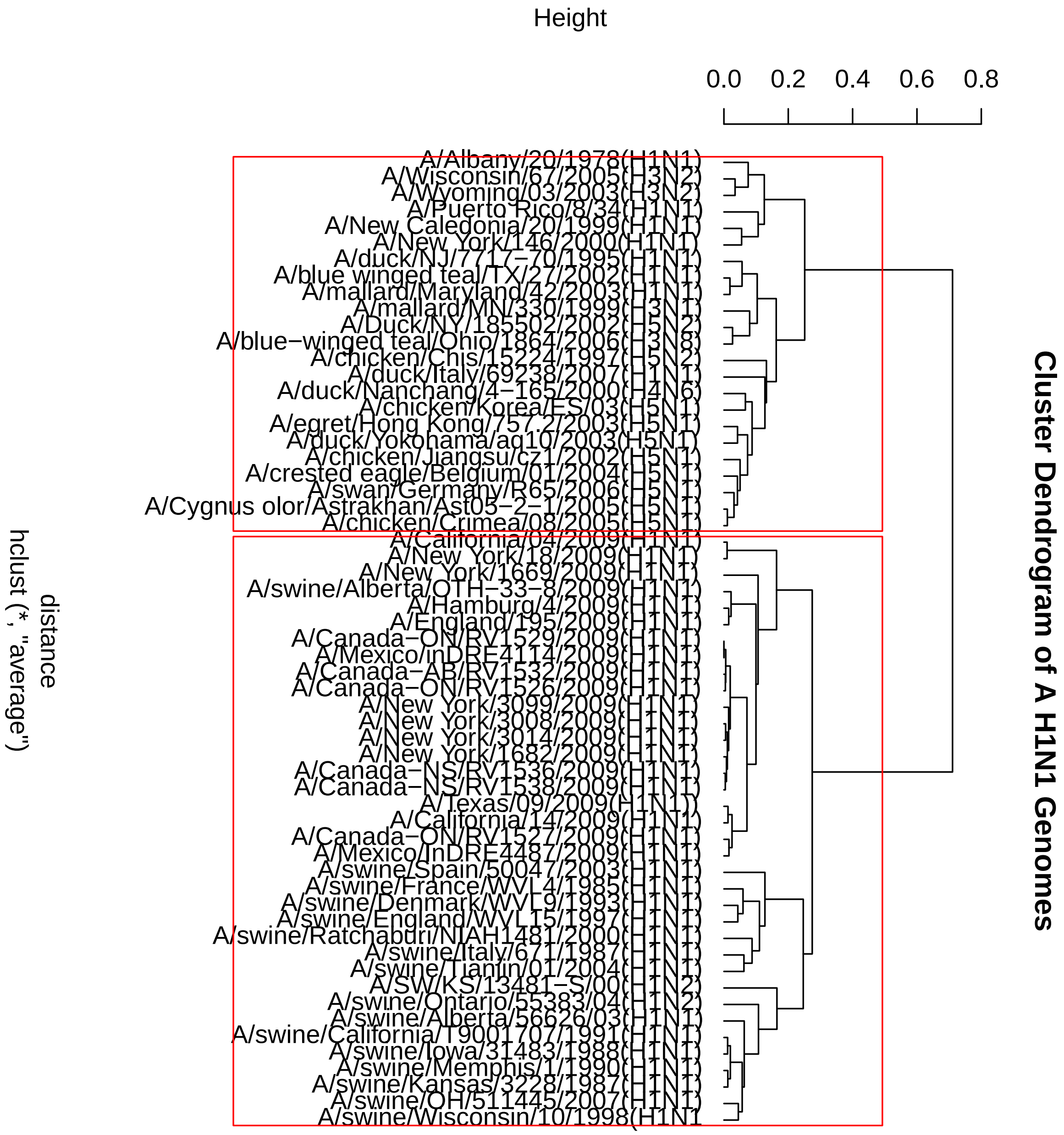
Cluster Dendrogram of A H1N1 Genomes
Height
0.0
0.2
0.4
0.6
0.8
distance
hclust (*, "average")
A/Albany/20/1978(H1N1)
A/Wisconsin/67/2005(H3N2)
A/Wyoming/03/2003(H3N2)
A/Puerto Rico/8/34(H1N1)
A/New Caledonia/20/1999(H1N1)
A/New York/146/2000(H1N1)
A/duck/NJ/7717−70/1995(H1N1)
A/blue winged teal/TX/27/2002(H1N1)
A/mallard/Maryland/42/2003(H1N1)
A/mallard/MN/330/1999(H3N1)
A/Duck/NY/185502/2002(H5N2)
A/blue−winged teal/Ohio/1864/2006(H3N8)
A/chicken/Chis/15224/1997(H5N2)
A/duck/Italy/69238/2007(H1N1)
A/duck/Nanchang/4−165/2000(H4N6)
A/chicken/Korea/ES/03(H5N1)
A/egret/Hong Kong/757.2/2003(H5N1)
A/duck/Yokohama/aq10/2003(H5N1)
A/chicken/Jiangsu/cz1/2002(H5N1)
A/crested eagle/Belgium/01/2004(H5N1)
A/swan/Germany/R65/2006(H5N1)
A/Cygnus olor/Astrakhan/Ast05−2−1/2005(H5N1)
A/chicken/Crimea/08/2005(H5N1)
A/California/04/2009(H1N1)
A/New York/18/2009(H1N1)
A/New York/1669/2009(H1N1)
A/swine/Alberta/OTH−33−8/2009(H1N1)
A/Hamburg/4/2009(H1N1)
A/England/195/2009(H1N1)
A/Canada−ON/RV1529/2009(H1N1)
A/Mexico/InDRE4114/2009(H1N1)
A/Canada−AB/RV1532/2009(H1N1)
A/Canada−ON/RV1526/2009(H1N1)
A/New York/3099/2009(H1N1)
A/New York/3008/2009(H1N1)
A/New York/3014/2009(H1N1)
A/New York/1682/2009(H1N1)
A/Canada−NS/RV1536/2009(H1N1)
A/Canada−NS/RV1538/2009(H1N1)
A/Texas/09/2009(H1N1)
A/California/14/2009(H1N1)
A/Canada−ON/RV1527/2009(H1N1)
A/Mexico/InDRE4487/2009(H1N1)
A/swine/Spain/50047/2003(H1N1)
A/swine/France/WVL4/1985(H1N1)
A/swine/Denmark/WVL9/1993(H1N1)
A/swine/England/WVL15/1997(H1N1)
A/swine/Ratchaburi/NIAH1481/2000(H1N1)
A/swine/Italy/671/1987(H1N1)
A/swine/Tianjin/01/2004(H1N1)
A/SW/KS/13481−S/00(H1N2)
A/swine/Ontario/55383/04(H1N2)
A/swine/Alberta/56626/03(H1N1)
A/swine/California/T9001707/1991(H1N1)
A/swine/Iowa/31483/1988(H1N1)
A/swine/Memphis/1/1990(H1N1)
A/swine/Kansas/3228/1987(H1N1)
A/swine/OH/511445/2007(H1N1)
A/swine/Wisconsin/10/1998(H1N1

Figure 5. Phylogenetic tree of influenza A H1N1 genomes.

and nonstructural gene NS respectively in A H1N1 genome. Our result showed that HA, NP, NS genes resemble those of classical swine influenza A viruses and PB2, PB1, PA genes resemble those of triple-reassortant swine influenza A viruses circulating in pigs in North America while the genes NA and MP are most closely related to those in influenza A viruses circulating in swine populations in Eurasia. The clustering result of these 8 gene segments by our method coincides with the phylogenetic analysis results from Garten et al (30) and Novel Swine-Origin Influenza A (H1N1) Virus Investigation Team (31). These conclusions have been widely accepted by other scientists (36) in scientific community. Therefore, this result shows that Kou et al.'s conclusion (37) was not fully convincing since they showed that PB2 and PA genes came from avian influenza virus and PB1 from human seasonal influenza virus. The gene and genome data are provided in the section of appendix. They can also be downloaded from Flu Database of GenBank (http://www.ncbi.nlm.nih.gov/genomes/FLU/FLU.html). The phylogenetic trees of those 8 segmented genes of A H1N1 genomes are shown in Appendices.

2.8.2 Statistical Analysis of Phylogenetic Tree of Influenza A H1N1 Genomes

To demonstrate the robustness and accuracy of the result above, we use statistical bootstrapping method to test. In statistics, bootstrapping is a computer-based method for assigning measures of accuracy to sample estimates (38). For assessing the uncertainty in hierarchical cluster analysis on influenza A (H1N1) genomes, p-values are calculated via multiscale bootstrap resampling for each cluster in hierarchical clustering. In Figure 6, we apply our method to analyze 59 influenza viruses based on their whole genomes. From the clustering result, we can see that the new swine influenza A (H1N1) viruses resemble the triple-reassortant swine influenza

A virus and Eurasian classical swine lineages. For each cluster in the hierarchical clustering, p-values are calculated via multiscale bootstrap resampling. Red values are AU (approximately unbiased) p-values, and green values are BP (bootstrap probability) values. Clusters with AU larger than 95% are strongly supported by data. The hypothesis that ”the cluster does not exist” is rejected with significance level 0.05; roughly speaking, these highlighted clusters exist and can be stably observed if we increase the number of observation. We conduct hierarchical cluster analysis with multiscale bootstrap with number of bootstrapping 1000 using average method and Euclidean distance. The new influenza A (H1N1) viruses resemble the swine virus (in the same cluster).

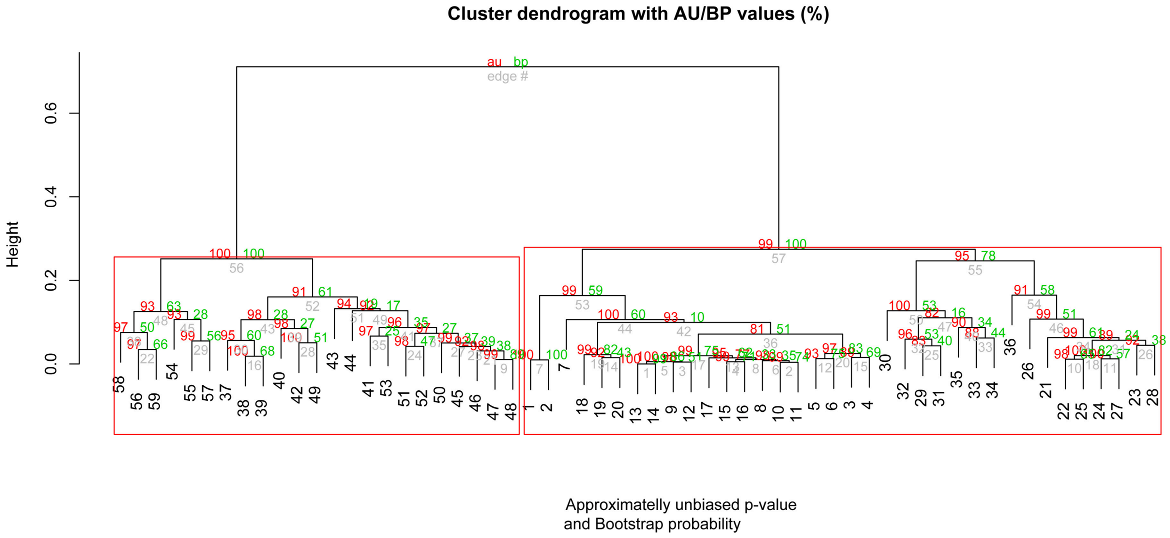

Figure 6. Multiscale bootstrap resampling on A H1N1 genomes.

For obtained approximately unbiased p-values, the AU p-values themselves include sampling error, since they are also computed by a limited number of bootstrap samples. The standard errors of AU p-values is shown in Figure 7. All clusters with AU p-values$\geq$ 95%, we can say that these highlighted clusters may stably be observed if we increase the number of observations (genomes). As the plot shows, all clusters have standard errors smaller than 0.05.

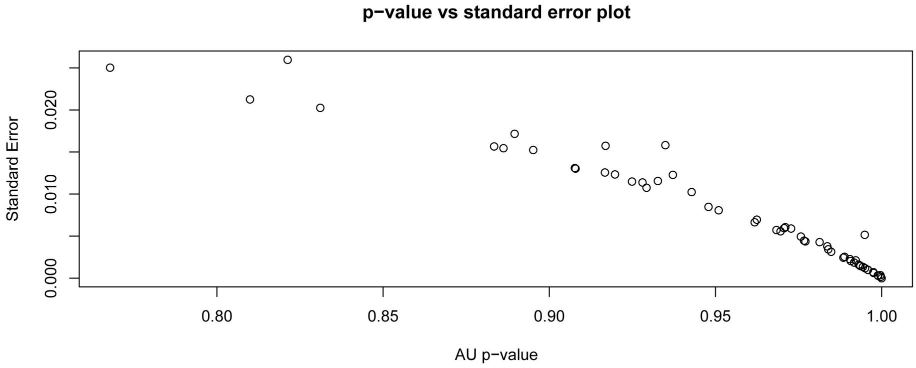

Figure 7. Standard error of approximately unbiased p-values

2.8.3 Phylogenetic Tree of HRV Genomes

In the second application, we apply our approach to study another group of viruses, human rhinovirus (HRV). Infection by HRV is a major cause of upper and lower respiratory disease worldwide and displays considerable phenotypic variation. Recently, Palmenberg et al (39) reported a comprehensive sequencing and analyzed result for all known HRV genomes based on the whole genome. In their article, the authors used the multiple alignment method to reconstruct the evolutionary tree. In that tree, five groups HRV-A, HRV-B, HRV-C, HEV-B and HEV-C were clearly identified. Now we use our natural vector method to do clustering for the same dataset (Table IV). We associate each whole genome sequence with a natural vector, and then by computing the Euclidean distances among these natural vectors we obtain the evolutionary tree (Figure 8) for all HRVs. MEGA software is used to draw the tree (40). According to our result, the five clusters HRV-A, HRV-B, HRV-C, HEV-B, and HEV-C are clearly separated from each other. But it takes 18 seconds for our novel method to get the clustering result while it takes more than 19 hours for multiple alignment method. Both methods yield the same clustering result.

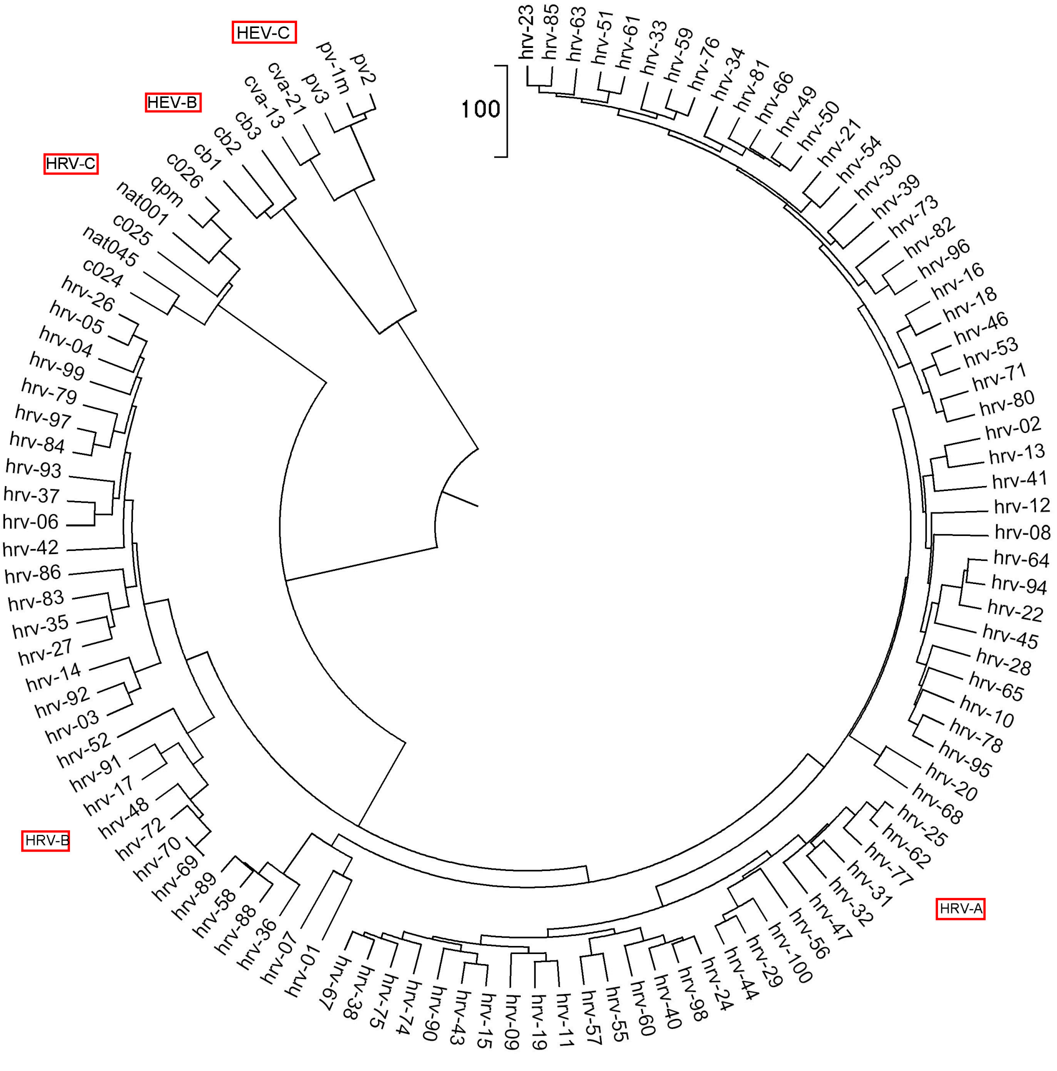

Figure 8. Phylogenetic tree of HRV genomes

2.8.4 Phylogenetic Tree of 36 Mammalian Mitochondrial Genomes

The third application is about mammalian mitochondrial genome. We consider the phylogeny of mitochondrial genomes. Mitochondrial DNA is not highly conserved and has a rapid mutation rate, thus it is very useful for studying the evolutionary relationships of organisms (41). We extracted 36 representative cases of complete mammalian mitochondrial genome sequences from the GenBank, each of which has length of more than 16000 nucleotides. Data is described in Table V. Here we use the first 16 moments of the natural vector, to characterize these 36 genomes. By computing the Euclidean distances between these points, we obtain the distance matrix for these 36 organisms. The phylogenetic tree is shown in Figure 9. The result shows these 36 genomes are well clustered into 8 clusters: Erinaceomorpha, Primates, Carnivore, Perissodactyla, Cetacea, Artiodactyla, Lagomorpha and Rodentia. This result coincides with the conclusion found by Liu et al (42), Raina et al (43), and Kullberg et al (44).

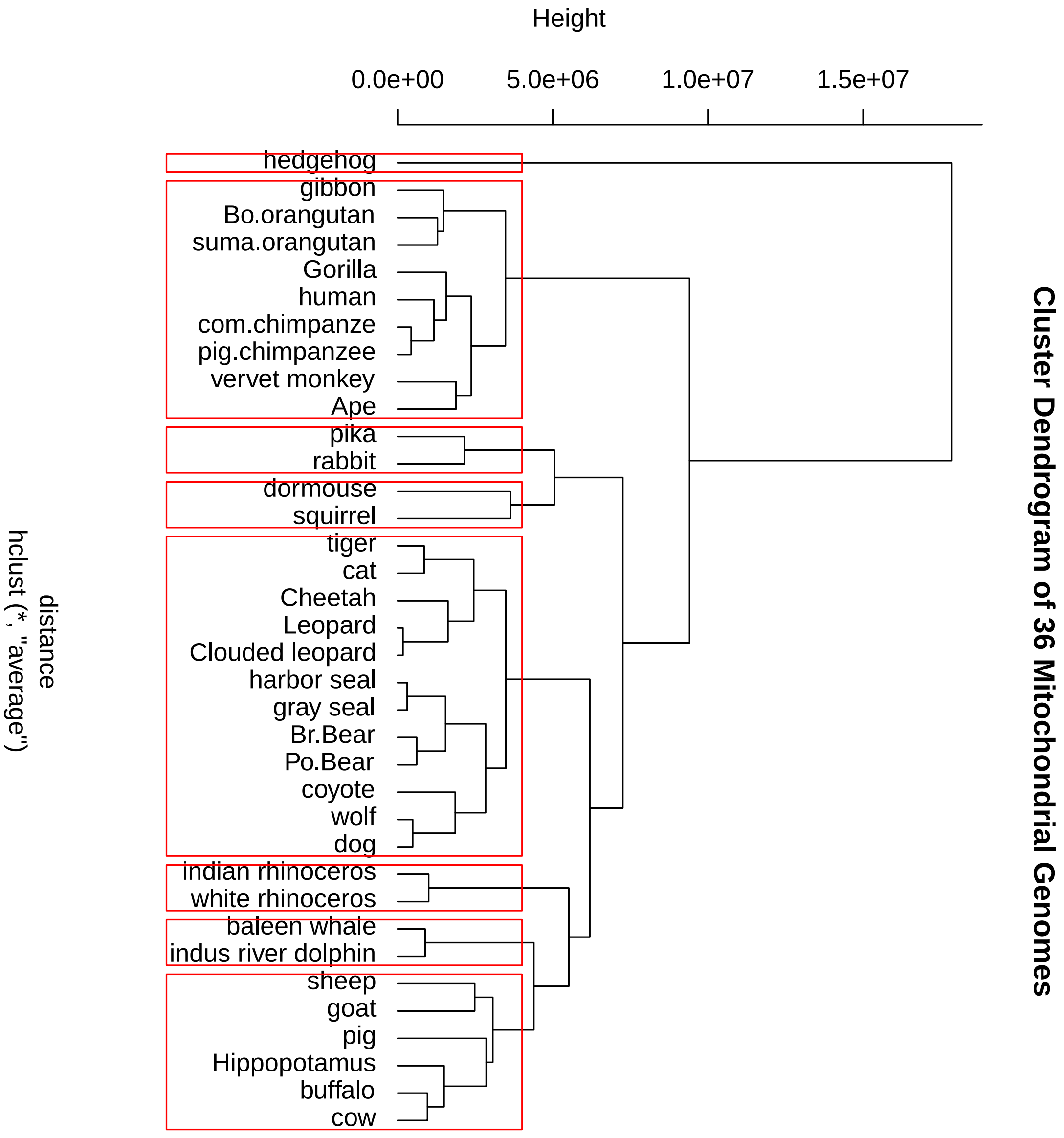

Figure 9. Phylogenetic tree of 36 mitochondrial genomes

2.9 Comparison of Natural Vector Method and State-of-the-Art Methods

We randomly pick up 21 genomes of influenza A H1N1 on different hosts to do the comparison of current standard methods involving evolutionary models (please see Table VI for data description). The different choices of these evolutionary models lead to different results which are sometimes not consistent (please refer Figure 10). Phylogenetic trees are reconstructed by using maximum likelihood (ML) alignment method with Jukes-Cantor model (Figure 10 a), neighbor-joining (NJ) method with Kimura 2 parameter model (Figure 10 b) and with Jukes-Cantor model (Figure 10 c). It is clear that swine flu viruses are not clustered correctly using ML method (Figure 10 a). NJ method with Kimura and Jukes-Cantor models yields totally different phylogenetic trees. Kimura model fails to distinguish the origin of A H1N1 virus since A H1N1 genomes are all very far away from other genomes (Figure 10 b), while Jukes-Cantor model fails to cluster swine flu viruses correctly(Figure 10 c). Moreover, there is no evidence to show which model can best fit all biological datasets without human intervention. Our method does not involve these models and it totally depends on the natural vectors constructed from the whole sequences. Therefore, this method is stable, natural and produces a unique clustering or phylogenetic result.

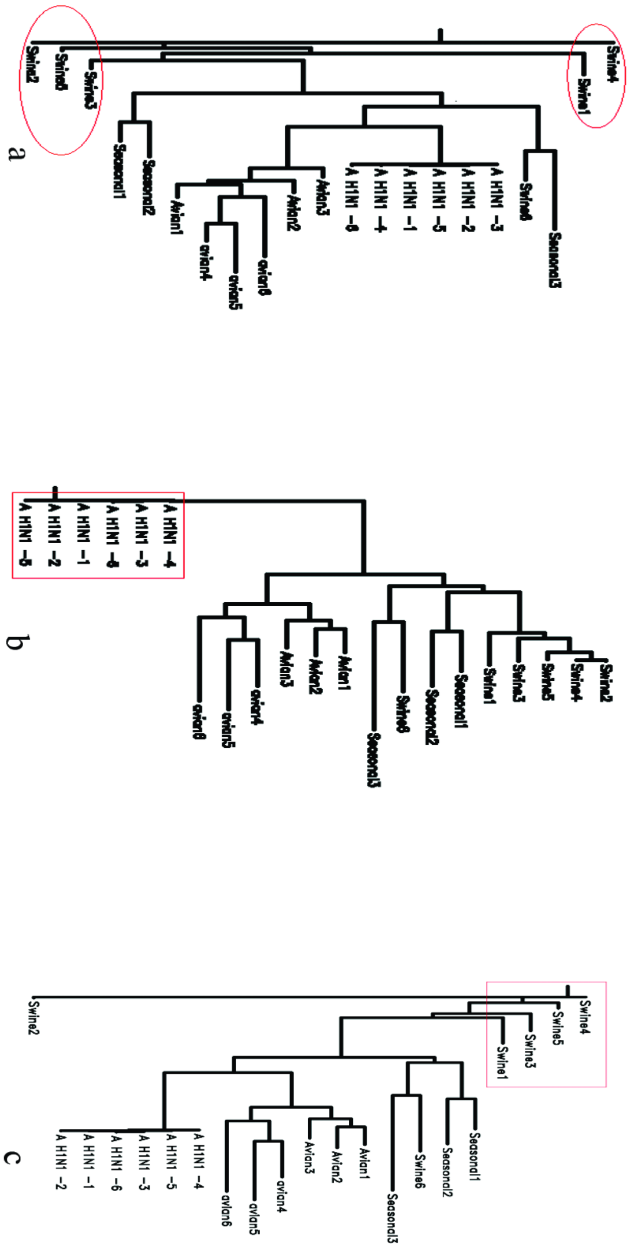

Figure 10. a: Maximum likelihood method with J-C model on a subset of A H1N1 genome dataset. b: Neighbor-joining method with kimura 2 distance. c: Neighbor-joining method with J-C distance.

In order to convincingly demonstrate the speed of our method, we compare the computation time of our method as well as other standard and state-of-the-art methods including ClustalW2, Muscle and MAFFT (23; 24; 25). Clustal is the most widely used general purpose multiple sequence alignment program for DNA or proteins. It produces biologically meaningful multiple sequence alignments of divergent sequences. Muscle and MAFFT are recognized as the most accurate and most speedy algorithms for phylogenies. The comparison result is shown in Figure 11.

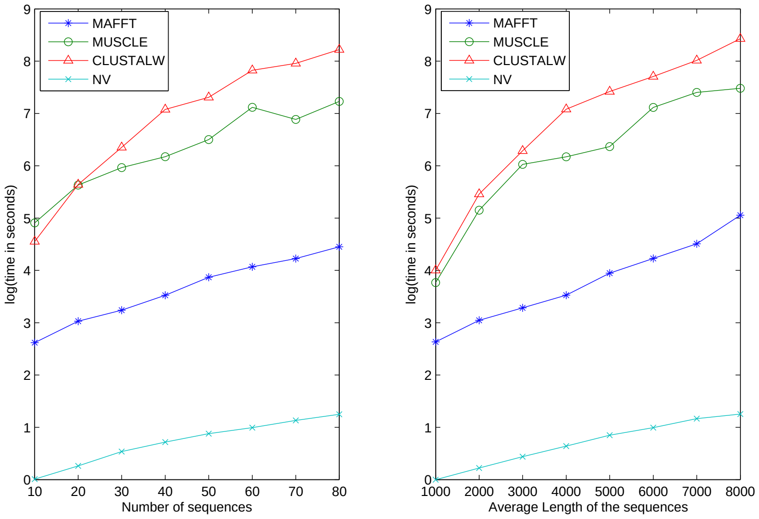

Figure 11. Computation time of natural vector method and state-of-the-art methods

we do the test on two sets of sequences (these datasets were provided by Bo Zhao). The first set includes 8 sub datasets. Each dataset contains 10, 20, 30, 40, 50, 60, 70 and 80 sequences respectively, where the lengths of all the sequences are around 4000 bp. Another set is composed of 8 datasets. Each single sub dataset has 40 sequences. The lengths of all sequences in these 8 sub datasets are 1000, 2000, 3000, 4000, 5000, 6000, 7000 and 8000 bp respectively. We build the trees on each dataset of the two sets by using the four methods and record the time that each method takes. The results in the Figure 11 shows that our method is much faster than the other three methods. The time of our method increases linearly as the number of sequences or the length of sequences increase, whereas the acceleration of the time for other three methods is much bigger. The actual time differences are much larger than the visual differences in the figure since we are using the log(time) as the label of y-axis.

2.10 Conclusion

We create a new mathematical method to characterize a genetic sequence as a natural vector and then do clustering or phylogenetic tree based on it. A natural vector system to represent a DNA sequence is introduced, and the correspondence between a DNA sequence and its natural vector is mathematically proved to be one-to-one. With this natural vector system, each genome sequence can be represented as a multidimensional vector. Genomes with close evolutionary relationship and similar properties are plotted close to each other when we construct the phylogenetic tree. Thus, it will provide a new powerful tool for analyzing and annotating genomes and their phylogenetic relationships. Our method is easier and quicker in handling whole or partial genomes than multiple alignment methods. There are four major advantages to our method: (1) once a genome space has been constructed, it can be stored in a database. There is no need to reconstruct the genome space for any subsequent application, whereas in multiple alignment methods, realignment is needed for add-on new sequences. (2) One can have global comparison of all genomes simultaneously, which no other existing method can achieve. Recently, we have finished the computation of a dataset with 27643 sequences. It takes us one hour to compute the distance matrix while it needs 4 years to finish it using multiple alignment method. (3) Our method is quicker than alignment methods and easier to manipulate, because not all dimensions of natural vectors are needed for computing. Instead, only the first several dimensions of natural vectors are good enough to cluster DNA sequences or genomes. Generally, we select the first N dimensions such that the clustering result remains stable even if we choose higher moments. N=12 in our experiments is good enough to characterize all sequences. We can compare all genes, DNA and genome sequences with different lengths by truncating all different (n+4) natural vectors into the same number of dimensions. The one-to-one correspondence between the truncated natural vectors (with 12 or more dimensions) and sequences is still valid. (4) The current standard methods involve the evolutionary models. The different choices of these evolutionary models lead to different results which are sometimes not consistent. This motivates us to create a new mathematical method without any model. Our method does not involve these models and it totally depends on the natural vectors constructed from the whole sequences. Therefore, this method is stable, natural and produces a unique clustering or phylogenetic result.

Although natural vector method can be used to reconstruct the phylogenetic trees of DNA sequences, genes and whole genomes, however this method may not be a suitable substitute for local multiple sequence alignment when one wants to identify the similarity of genomic subsequences and does not know a priori which subsequences to identify.

CHAPTER 3

NATURAL VECTOR METHOD ON PROTEIN SEQUENCES

3.1 Introduction

The protein universe, a concept first mentioned in 1992 (47), is the collection of all proteins of every biological species that lives or has lived on earth. According to recent PNAS paper by M. Levitt (48), there are almost 8 million sequences in non-redundant (NR) database of protein sequences, including the complete genomes of 1800 different species. This large body of data is doubling in size every 28 months. Coming to understand with the protein universe is unarguably central (48; 49). Thus, researchers are using the evolutionary relatedness of all life on earth to understand the protein universe (48; 50). Protein universe is a poorly defined and mysterious entity to scientists. An obvious way to reveal the nature of the protein universe is to cluster sequences into families by similarities (measured, say, by the percentage of identical amino acids when suitably aligned). Appreciable levels of similarity generally imply homology or descent from a common ancestor, which allows related sequences to be grouped into families (51). Here we approach the problem differently. We use the natural vector method to realize the protein universe as a subspace of Euclidean space, which we call protein space. We can globally compare all the existing proteins within this protein space. Many methods have been reported to analyze the huge amounts of genes. One of them is the graphical representation of gene sequences, which is a very powerful tool for visual comparison of gene sequences.

Hamori first used a three-dimensional H curve to represent a gene sequence (52). Gates later published a two-dimensional graphical representation that is simpler than the H curve (53). However, Gates' graphical representation has high degeneracy. Yau et al reported previously a new two-dimensional graphical representation of gene sequences (18), which has no circuit or degeneracy, so the correspondence between gene sequences and gene graphs is one-to-one. Lately, many graphical representation methods for gene and genome sequences have been proposed (54; 22; 55), however, the method to make a protein sequence graph has never been shown although statistical method has been used to analyze protein structures and evolution (56; 19). Unlike dealing with a gene or DNA sequence from only four nucleotides, dealing with a protein sequence, from 20 amino acids, is more complicated. Yau et al designed another graphical method to successfully represent proteins in a protein map via moment vectors (20). Although this method can be successfully used to cluster the proteins, the choice of parameters is artificial. Recently, Carr et al have created a rapid method for characterization of proteins using feature vectors (21). As illustration, protein kinase C (PKC) family are used to test that method. This method is quick in handling the clustering of proteins, however, there is no proof of one-to-one correspondence between a protein sequence and a feature vector. Besides, it requires scanning the whole database to compute the theoretical mean of the feature vector. We associate to each protein a natural vector, which allows us to put protein universe as a subspace in Euclidean space. Then clustering or comparison of proteins can be conducted based on the distance calculated from the natural vectors. We are able to realize protein universe as a subspace of Euclidean space because the protein sequences and natural vectors are in one-to-one correspondence. Our natural vector can also be used to predict the properties of proteins whose functions are not yet determined.

3.2 Method

Since the natural vector method for DNA or genome sequence has been stated in chapter 2, the natural vector method for protein sequence can be described similarly by replacing 4 nucleotide bases with 20 amino acids.

Thus the next description below is to give a complete understanding of the distribution of 20 amino acids.

(1) The quantities of the 20 amino acids of a protein sequence are chosen as the first 20 parameters of the natural vector. 20 integers $n_A, n_R, n_N, \ldots, n_V$ denote the numbers of 20 amino acids in a protein sequence.

(2) The second group of 20 numerical parameters are the arithmetic mean value of total distance for each of the 20 amino acid bases:$\mu_k = \frac{T_k}{n_k}$, $k = A, R, N, \ldots, V.$

(3) The final group of parameters that we include in the natural vector are composed of normalized central moments. The first normalized central moment is:

$$D_1^k = \sum_{i=1}^{n_k}(s[k][i] - \mu_k) = T_k - n_k(\frac{T_k}{n_k}) = 0.$$

Because the first central moment is zero, we start with the second normalized central moment. The second normalized central moment is the variance of the distance distribution for each amino acid base:

$$D_2^k = \sum_{i=1}^{n_k} \frac{(s[k][i] - \mu_k)^2}{n_k n}$$

where k is each of 20 amino acid. If the distribution of each amino acid base is different, protein sequences cannot be similar even though they may have the same amino acid contents and the same total distance measurement. Therefore, the information about distribution has also been included in the natural vector. As described above, each subset of numerical parameters is not sufficient to annotate protein sequences. However, the combined numerical parameters are sufficient to characterize each protein sequence. So the natural vector is given as follows:

$$< n_A, \mu_A, D_2^A, \ldots, D_{n_A}^A, n_R, \mu_R, D_2^R, \ldots, D_{n_R}^R, \ldots, n_V, \mu_V, D_2^V, \ldots, D_{n_V}^V > \tag{3.1}$$

In order to express the vector elegantly, we rewrite it as follows:

$$< n_A, n_R, \ldots, n_V, \mu_A, \mu_R, \ldots, \mu_V, D_2^A, D_2^R, \ldots, D_2^V, \ldots, D_{n_A}^A, D_{n_R}^R, \ldots, D_{n_V}^V >, \tag{3.2}$$

Alternatively, the natural vector can be written as

$$< n_A, n_R, \ldots, n_V, \mu_A, \mu_R, \ldots, \mu_V, D_2^A, D_2^R, \ldots, D_2^V, \ldots, D_{n_\pi}^A, D_{n_\pi}^R, \ldots, D_{n_\pi}^V >, \tag{3.3}$$

where $n_\pi = \max\{n_A, n_R, \ldots, n_V\}$. By this definition, $D_j^k = 0$, if $j > n_k$. The natural vector is obtained by concatenating the first group of parameters (the number of each base) and the second group of parameters (the mean value of total distance of each base) to the normalized central moments. Obviously, higher moments converge to 0 for a random generated sequence since for any given k,

$$D_j^k = \sum_{i=1}^{n_k} \frac{(s[k][i] - \mu_k)^j}{n_k^{j-1} n^{j-1}} \leq n_k \frac{(n/2)^j}{n_k^{j-1} n^{j-1}}$$

$$= \frac{(n/2)^j}{n_k^{j-2} n^{j-1}} \leq \frac{n^j}{2^j \cdot n_k^{j-2} n^{j-1}} = \frac{n}{2^j \cdot n_k^{j-2}}$$

If one specific amino acid does not exist, i.e., $n_k = 0$, its μ_k, D_j^k are zeros. On the other hand, it is clear that $n_k \geq 2$, otherwise, $s[k][i] - \mu_k = 0$, which yield $D_j^k = 0$. From the viewpoint of probability, we can assume that the expectation value of any amino acid base is $n_k = n/20$ (uniform distribution) for a protein sequence with given length n. Clearly, this limit goes to 0 as j approaches a large value of n_k. For example, given a protein with length $n = 120$, we theoretically assume that there are 6 alanine A following discrete uniform distribution, then the 4th central moment $D_5^k \leq \frac{120}{2^4 6^2} = 0.208$, which is very close to 0. That is, the higher central moments converge to 0. In this example, with the increasing of the higher moments, D_j^k is approaching 0.

We have used natural vector to obtain a good numerical characterization of a protein sequence. In the biological applications, we use the first several truncated moments from the representation (we use 60 dimensional moments in the biological applications). As we showed above, higher moments converge to 0 which do not contribute to the computation of the distance.

3.3 The Corresponding Theorem

One of the most important contributions is that we can prove that the correspondence between a protein sequence and its natural vector is one-to-one. Actually the proof of protein sequence is very similar to that of DNA sequence.

Theorem: Suppose a protein sequence has the number n of amino acids, then the correspondence between the sequence and its natural vector

$$< n_A, n_R, \ldots, n_V, \mu_A, \mu_R, \ldots, \mu_V, D_2^A, D_2^R, \ldots, D_2^V, \ldots,$$

$$D_{n_A}^A, D_{n_R}^R, \ldots, D_{n_V}^V >, \qquad (1)$$

is one-to-one.

Proof. To prove the theorem, first we need to demonstrate that for any given proper protein natural vector, we can recover the corresponding protein sequence. Let us first denote $z_i = s[k][i] - \mu_k$, then the normalized central moments can be simplified as:

$$D_j^k = \sum_{i=1}^{n_k} \frac{z_i^j}{n_k^{j-1} n^{j-1}}, j = 1, 2, \ldots, n_k,$$

where n is the total length of the sequence. If we can find the value of each z_i, the protein sequence can be recovered. To solve for z_i, let $\delta_j = D_j^k n_k^{j-1} n^{j-1}$, then the δ_j can be obtained by D_j^k and n_k. For a given natural vector, n_k is known for each amino acid base A, R,..., or V. So we need to solve for each z_i corresponding to each one of the n_k, say n_A. The value of z_i corresponding to other amino acid bases could be obtained in the same way.

Clearly δ_j and z_i have the relation as below:

$$\begin{cases} \delta_1 = z_1 + z_2 + \ldots + z_{n_A} \\ \delta_2 = z_1^2 + z_2^2 + \ldots + z_{n_A}^2 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ \delta_{n_A} = z_1^n + z_2^n + \ldots + z_{n_A}^n \end{cases}$$

$z_1, z_2, \ldots, z_{n_A}$ are roots of a symmetric polynomial $a_0 + a_1 z + a_2 z^2 + \ldots + a_{n_A} z^{n_A} = (z - z_1)(z - z_2)\ldots(z - z_{n_A})$. Let p_d $(d = 1, 2, \ldots, n)$ be the elementary symmetric polynomials in $z_1, z_2, \ldots, z_n$, i.e.,

$$p_1 = \sum_1^{n_A} z_i, p_2 = \sum_{i<j} z_i z_j, p_3 = \sum_{i<j<l} z_i z_j z_l,$$

$$\ldots, p_{n_A} = z_1 z_2 \ldots z_n$$

Then

$$p_1 = -a_{n_A - 1}, p_2 = a_{n_A - 2}, \ldots, p_{n_A} = (-1)^{n_A} a_0.$$

By using Newton's famous identities (28):

$$\delta_d - p_1 \delta_{d-1} + \ldots + (-1)^{d-1} p_{d-1} \delta_1 + (-1)^d p_d = 0,$$

where $d = 1, 2, \ldots, n$, p_d is the elementary symmetric polynomials in $z_1, z_2, \ldots, z_{n_A}$. a_i can be obtained by δ_j as shown below:

$$\begin{cases} a_n & = & 1 \\ a_{n-1} & = & (-1)\delta_1 \\ a_{n-2} & = & \frac{1}{2}(\delta_1^2 - \delta_2) \\ a_{n-3} & = & (-1)^3\frac{1}{6}(\delta_1^3 - 3\delta_1\delta_2 + 2\delta_3) \\ a_{n-4} & = & \frac{1}{24}(\delta_1^4 - 6\delta_1^2\delta_2 + 3\delta_2^2 + 8\delta_1\delta_3 - 6\delta_4) \\ \multicolumn{3}{l}{\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots} \end{cases}$$

As a result, the coefficients of the symmetric polynomial $a_0 + a_1 z + a_2 z^2 + \ldots + z^n = (z - z_1)(z - z_2)\ldots(z - z_n)$ can be confirmed, and the set of all roots can be obtained. Next we need to identify each root $z_1, z_2, \ldots, z_n$.

$z_i - z_{i+l} = s[A][i] - s[A][i + l]$. For any $l > 0$, $s[A][i] < s[A][i + l]$ by the definition of $s[A][i]$. It is obviously that $z_i < z_{i+l}$. As a consequence z_i is strictly increasing and each root can be identified by this property, which means each value of $s[A][i]$ can be obtained. Since μ_A is known for a given natural vector, $s[A][i]$ can also be obtained.

Similarly, we can find all $s[k][i]$ for $k = R, N, \ldots, V$ respectively. Therefore, the unique corresponding a protein sequence can be recovered based on all $s[k][i]$, k is an amino acid base.

On the other hand, given a protein sequence, the number of each amino acid and the distance of each amino acid to the origin are determined. Based on this information, we can compute T_k and μ_k. So it is easy to construct a natural vector. Clearly, any two different protein sequences are distinct in length, number of each amino acid base or amino acid arrangement. Thus, the two corresponding natural vectors are completely different.

Therefore, we have successfully proved that the correspondence between a protein sequence and its natural vector is one-to-one. □

Remark We can see that $D_1^k = \sum_{i=1}^{n_k} z_i$, thus

$$D_1 = \sum_{i=1}^{n_k} (s[k][i] - \mu_k)$$

$$= T_k - n_k(\frac{T_k}{n_k}) = 0$$

So the natural vector sequence does not need D_1^k.

So far we have characterized a protein sequence by a natural vector with series of moments, next we can define the natural distance between two proteins. a natural distance between two proteins is the distance between their corresponding natural vectors, which is the classic Euclidean distance between two natural vectors:

$$L(j, j') = \sqrt{\sum_j \sum_i (j_i - j_i')^2}$$

where $i = A, R, \ldots, V$. $j = n, \mu, D_2^i, D_3^i, \ldots, D_{n_i}^i$,

$n_i$is the number of each A, R, …, V.

3.4 Applications

3.4.1 Phylogenetic Tree of PKC Family

We use the natural vector method to do clustering and reconstruct phylogenetic trees. The first example is protein kinase C (PKC) family. Kinases are proteins which modify other proteins by phosphorylation, the covalent addition of phosphate groups. The PKC family is a large multigene family of serine/threonine kinases. Six main groups of PKCs can be identified by domain architecture: conventional, novel, atypical, PKCμ-like, fungal PKC1, and PKC-related kinases. The first three of these groups can be further categorized into subtypes. In general, the PKC domain architecture consists of a regulatory region and a catalytic domain. The regulatory region contains several functional domains of varying types. True PKCs are classified as conventional, novel or atypical based on the functional domains present in the regulatory regions of the PKCs. Briefly, conventional PKCs contain subtypes $\alpha, \beta I, \beta II$ and γ; novel PKCs contain subtypes θ, ϵ, δ and η; and atypical PKCs contain λ or ι, and ζ. The catalytic domain is more conserved and more commonly used for differentiating between families of protein kinases. However, it is also useful in characterizing the PKCμ-like kinases, which contain markedly different catalytic regions from the rest of the protein kinase C members.
60 moments in the natural vector are used to reconstruct the phylogenetic tree.

The dendrogram (Figure 12) created by neighbor-joining method of the PKC natural vectors successfully recreates the phylogenetic relationships between PKC architectural types and highlighted the degree of difference between PKC-related kinases, PKC1s and other PKCs. PKC data can be available in Table VII.

On the comparison of our result, we perform the most state-of-the-art method, MAFFT on the same dataset. The result shows that some nPKC ϵ proteins and a cPKCγ proteins (cPKC 058) are not correctly clustered(Figure 13).

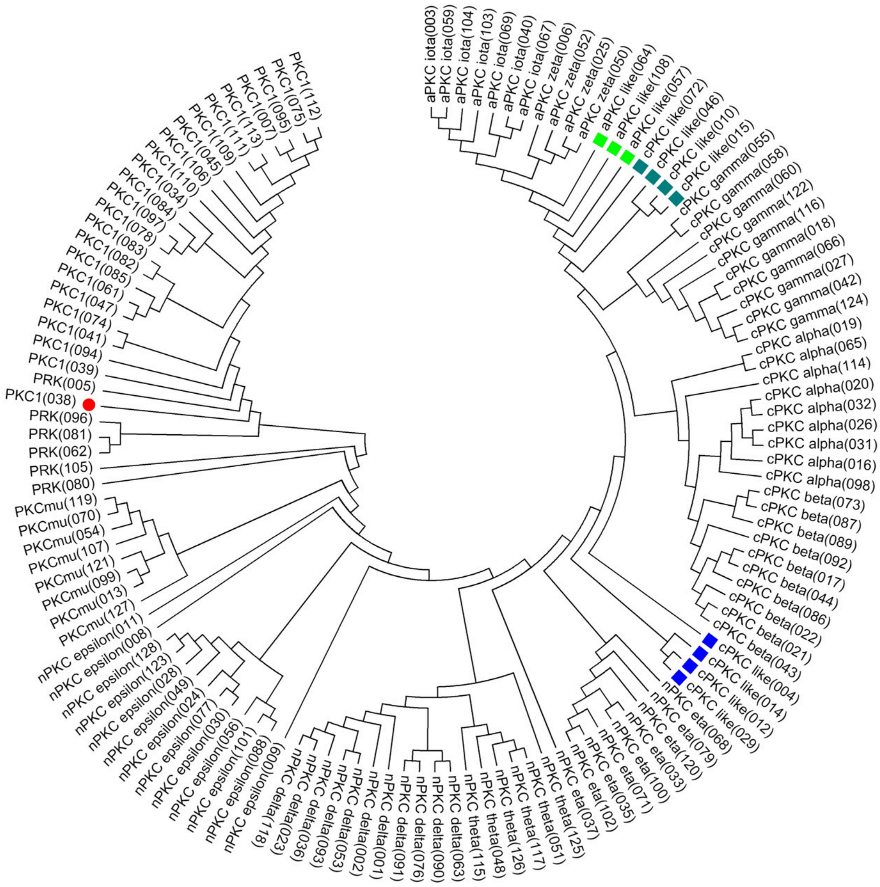

Figure 12. Neighbor-joining method with distance of natural vectors on PKC family

Not only does the natural vector method provide an accurate result, but it is much more efficient

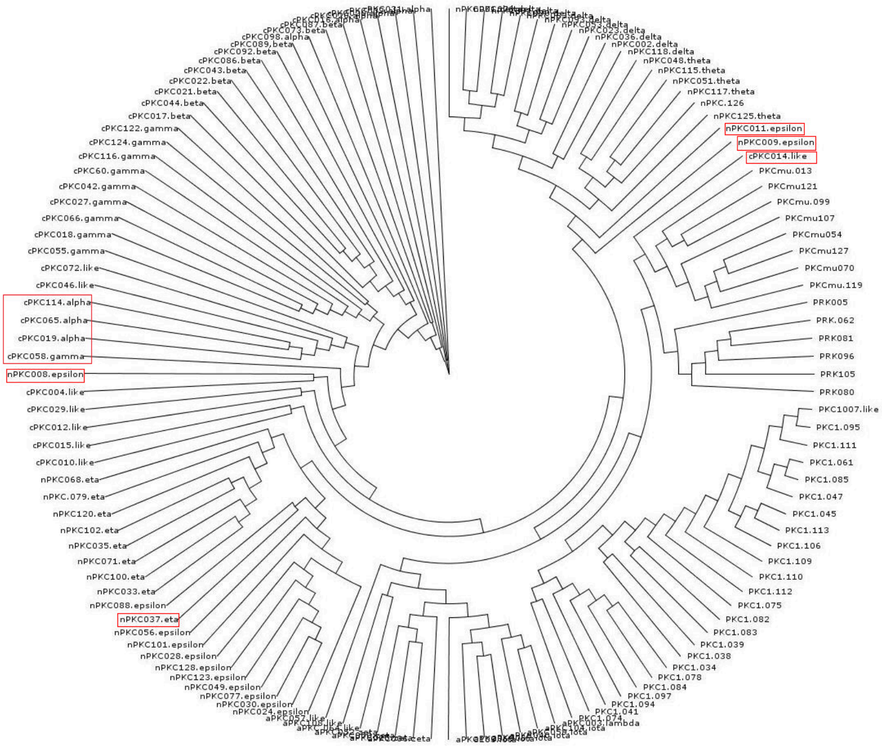

Figure 13. MAFFT on PKC family

in computation time than other phylogenetic methods. It takes only 2 seconds to complete the distance matrix of natural vectors while MAFFT needs 55.4 seconds to get the distance.

3.4.2 Phylogenetic Tree of Beta-globin Family

Another biological application is about beta-globin sequences. Fifty beta-globin protein sequences of different species were extracted from Swiss-Prot (http://au.expasy.org/). Using 60 dimensional natural vector method on this dataset, we can cluster the 50 beta-globin sequences using hierarchical clustering method with average distance on the distance matrix provided by natural vectors (Figure 14). Data is described in Table VIII.

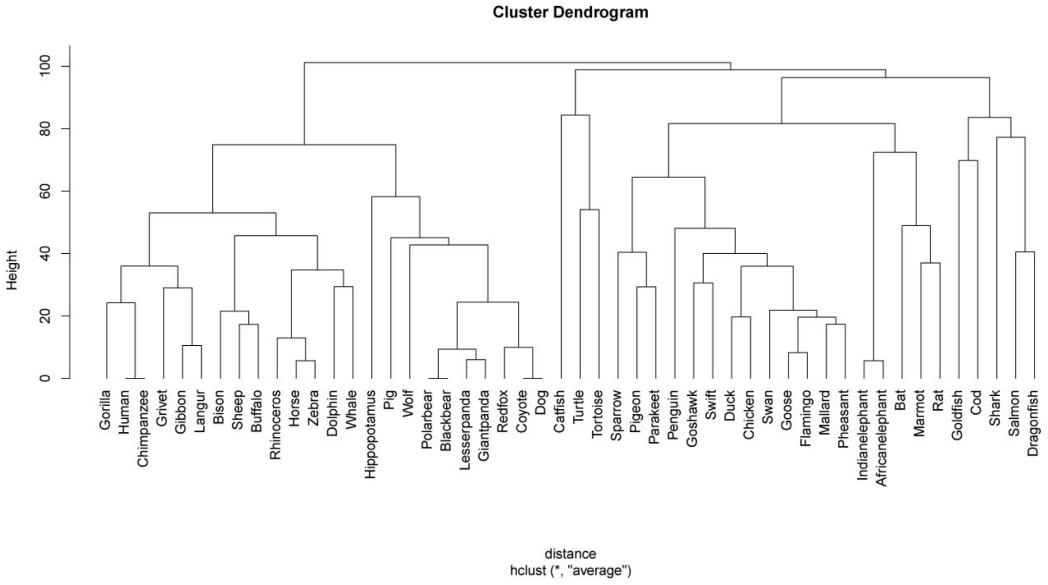

Figure 14. Hierarchical clustering method with the distance of natural vectors on beta-globin proteins

From Figure 14, we note that these 50 beta-globins are correctly separated into several sub clusters: primates, perissodactyla, artiodactyla, cetacea, carnivora, avian, proboscidea, reptilian,chiroptera, rodentia and fish species. Because the chimpanzee beta-globin sequence is the same as the human beta-globin sequence, these two proteins have the same natural vectors. For the same reason, dog and coyote beta-globin sequences have the same natural vectors, as do black bear and polar bear beta-globin sequences. Compared with the result obtained from MAFFT, not only does our method provide a high accurate result, but it just takes 1 second to complete the distance matrix while MAFFT has to take 31.7 seconds.

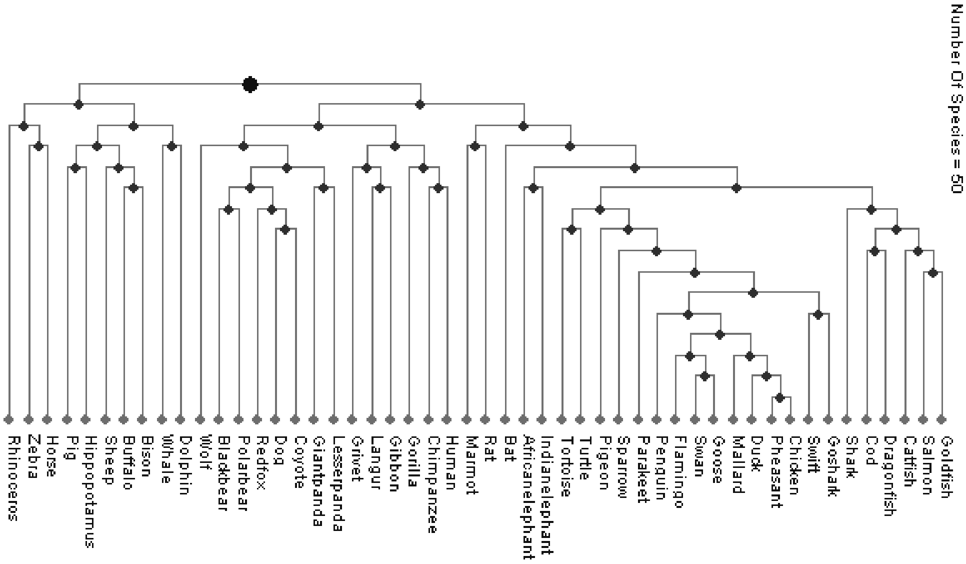

Figure 15. MAFFT on beta-globin proteins

3.5 Conclusion

Natural vector method is a very useful tool for clustering or phylogenetic analysis on protein sequences. Each protein can be represented as a natural vector, which has been mathematically proved that it is one-to-one corresponding to a protein. Moreover, in biological applications only a truncated natural vector can be used to successfully do clustering or phylogenetic trees on proteins rather than the whole natural vector. Therefore, it saves much time in computation. Compared with current phylogenetic methods, not only does our proposed natural vector method provide a high accurate result, but more importantly, it is much quicker in computation time, which guarantees us to do clustering and phylogenetic analysis on a huge number of sequences.

CHAPTER 4

A NEW APPROACH TO CLASSIFY DNA SEQUENCES INTO INTRON-LESS AND INTRON-CONTAINING SEQUENCES

4.1 Introduction

An important problem for geneticists as well as computer scientists involve classifying particular items into common groups. Here we focus on classifying sequences of DNA into either an intron-containing or an intron-less (exon only). Intron-less and intron-containing genes have different biological properties and statistical characteristics. Congruent with the Spearman's rank correlation, the comparison of intron-less and intron-containing genes showed significantly reduced expression for intron-less genes when compared to intron-containing genes (57). These observations raised interesting questions about the role of intron-less and intron-containing genes. On the other hand, Peng et al (58) have discovered that long-range correlation existed in the intron-containing genes, but did not exist in the intron-less genes. This work was based on simple random-walk model of DNA sequences, in which a pyrimidine led to a step up and a purine a step down. Consequently, the walk resulted in a definite landscape for a given sequence and only one parameter was calculated based on the landscape. This parameter was proposed to distinguish between the intron-containing and intron-less genes. However, further study showed that this finding cannot be used as a general method to identify intron-less genes (59; 60). They pointed out that there were some basic drawbacks in the work of Peng et al (58).

Firstly, the DNA sequence cannot uniquely be described by the pyrimidine & purine walk. Secondly, although Buldyrev et al (62) considered six other possible walks besides the pyrimidine & purine walk, the cross-correlations between any two walks were totally ignored. Zhang et al (60; 63) introduced a Z-curve consisting of three parameters, in which the cross-correlations between any two parameters of the Z-curve were considered. As an application, they used the Z-curve method to classify a dataset consisting of 100 intron-containing and 100 intron-less genes. The discriminant accuracy as high as 89.0% can be obtained by using Fisher's linear discriminant algorithm. Although the distributions of three different biological types were displayed in Z-curve, it did not reveal the cross-correlations of distances between the nucleic bases, which are also important parameters to classify genes into intron-containing and intron-less.

In recent twenty years, a number of methods have been developed for DNA/protein clustering or classification, gene prediction and exon/intron parsing. Similar as Zhang's Z-curve method, Ma (64) created a model based on position weight function to describe genes by transforming them into quaternary numbers. Especially, this method indicates that E.coli K12's genome and the eukaryote yeast's genome have different strengths of single nucleotide periodicities. Yau et al (18) first developed DNA representation method in 2003, Yau and his colleagues have been studying the efficient methods to cluster and classify DNA and proteins (65; 19; 67; 66; 20; 21; 22). The first successful methods for gene prediction and exon/intron parsing are based on a generalized hidden Markov model (GHMM) framework. The best-known example of this method is the program GENSCAN (68), which in 1997 was shown to be dramatically more accurate than the previous state-of-the-art prediction programs. GENSCAN is easy to use and remains a popular bioinformatics tool. More recent *de novo* gene predictors have been created, including ROSETTA (69), CEM (70), TWINSCAN (71), N-SCAN (72), SLAM (73), EvoGene (74), ExoniPhy (75), DOGFISH (76), EXONSCAN (77). *De novo* gene predictors additionally made use of aligned DNA sequence from other genomes (78). Alignments can increase predictive accuracy since protein-coding genes exhibit distinctive patterns of conservation. Another type of predictors, including Pairagon (79), GenomeWise (80) and EXOGEAN (81), made use of expression data, usually EST or cDNA alignments. These methods can provide predictions for splice sites of exons. Thus, these modern gene-finding or gene-parsing systems provided a prediction of precise (predicted) splice sites of the exons/introns in the gene, while also producing the intron-bearing status of a gene.

Our problem is for classifying genes as to their intron-bearing status only. Implementing some classical methods such as GENSCAN, EXONSCAN and N-SCAN on Zhang's dataset, the result shows that these methods are not very effective for our problem, comparing with Z-curve and our ZC method. These classical methods adopted the alignment technique which is primarily useful for locating the coordinates of splice sites of each genes. In addition, they are performing very well on protein-coding portion of genes but not on the whole gene. We perform our ZC method on the same dataset provided in Zhang' paper (60). The average accuracy of 96.5% can be obtained by our method (Table XVII). In order to avoid the statistical bias due to the small dataset, we introduced another dataset consisting of 1000 intron-containing and 1000 intron-less genes. As a result, the accuracy of 93.05% can be achieved (Table XVII and Figure 20), which is significantly better than Z-curve method and other state-of-the-art algorithms. In this chapter, we investigate three new parameters which are based on the cross-correlations between the distribution of distances of nucleic bases in any two sequences. Those new parameters together with Zhang's original parameters and the value of their total standard deviation can be used to significantly improve the accuracy of classification on intron-bearing status of genes.

In Section 4.2, we discuss the proposed method by introducing Z-curve and cumulative distance, calculating the feature parameters based on detrended fluctuation analysis and selecting support vector machine classifier. We demonstrate the performance of the proposed method using real biological datasets in Section 4.3 and draw a conclusion of our study in Section 4.4.

4.2 Method

4.2.1 Z-curve and Background

First let us recall the Z-curve theory of a DNA sequence. It was first defined and developed by Zhang and his colleagues (83; 82; 60). Consider a DNA sequence with N bases. Let the number of steps be denoted by n, i.e., $n = 1, 2, \ldots, N$. We count the cumulative numbers of base A, C, G, T which occur in the subsequence from the first to the nth base in the DNA sequence. Those cumulative numbers are denoted by A_n, C_n, G_n and T_n respectively. The

Z-curve is a three-dimensional curve which consists of a series of nodes P_n $(n = 1, 2, \ldots, N)$, whose coordinates are denoted by x_n, y_n and z_n. It is shown that

$$\begin{cases} x_n &= 2(A_n + G_n) - n \\ y_n &= 2(A_n + C_n) - n \\ z_n &= 2(A_n + T_n) - n \end{cases}$$

where $n = 1, 2, \ldots, N$ and $A_0 = C_0 = G_0 = T_0 = 0$. The connection of the nodes $P_0 = 0$, $P_1, \ldots, P_N$ one by one by straight lines is defined as the Z curve of the DNA sequence.

Zhang et al (83) proved that there existed a one-to-one correspondence between the DNA sequence and the Z curve. Therefore, the Z-curve could be used to calculate all the information of DNA sequence. Besides, x_n, y_n and z_n have biological implications. x_n displays the distribution of bases of purine-pyrimidine types along the DNA sequence. When the number of purine bases in the subsequence from the first to the nth base is greater than that of pyrimidine bases, $x_n > 0$, otherwise, $x_n < 0$. y_n displays the distribution of bases of amino-ketone types along the sequence. When the number of amino bases in the subsequence from first to the nth base is greater than that of ketone bases, $y_n > 0$, otherwise, $y_n < 0$. z_n displays the distribution of bases of weak-strong hydrogen (H) bond types along the sequence. When the number of weak H bond bases is greater than that of strong H bond bases, $z_n > 0$, otherwise, $z_n < 0$.

Z-curve method displays three biological types of a sequence and this method shows that it is useful to classify genes into intron-containing and intron-less. However more intrinsic parameters should be discovered for the higher accurate classification. Here we introduce cumulative distance, which is important in the proposed ZC method.

D^m is the cumulative distance of each nucleotide base to the first nucleotide in m steps. So $D^m = D^m_A + D^m_C + D^m_G + D^m_T$, where each D^m_j represents the cumulative distance of all nucleotides of nucleic base j to the first one in m steps, $j = A, C, G, T$.

$D^m_j = \sum_{i=1}^m t^j_i$, $j = A, C, G, T$, t^j_i is the distance from the first nucleotide to the last nucleotide of nucleic base j in i steps. For example, $< ACCTCGC >$ is a gene sequence. For nucleic base C, $t^c_1 = 0$, $t^c_2 = 1$, $t^c_3 = 2$, $t^c_4 = 0$, $t^c_5 = 4$, $t^c_6 = 0$, $t^c_7 = 6$, so $D^7_C = 13$. So we can compute $D^7_A = 0, D^7_G = 5$ and $D^7_T = 3$ in the same way. Three types of cumulative distances can be defined similarly as follows:

$$\begin{cases} D_n &= (D^n_A + D^n_G) - (D^n_C + D^n_T) \\ E_n &= (D^n_A + D^n_C) - (D^n_G + D^n_T) \\ H_n &= (D^n_A + D^n_T) - (D^n_C + D^n_G) \end{cases}$$

x_n, y_n, z_n display the distributions of three different biological types. But more importantly, D_n, E_n, H_n reveal the cross-correlation of the "position" of each nucleic base. These cumulative distances are natural objects and they provide more biological information based on DNA sequences.

4.2.2 Construction of Cumulative Distance Matrix

Detrended fluctuation analysis was first introduced by Peng et al (58). It is a scaling analysis method used to estimate long-range power-law correlation parameters in noisy signals. It was developed by Peng et al for the analysis of DNA sequences.

Zhang et al (60) extended Peng et al's result by constructing a matrix based on x_n, y_n and z_n. Motivated by their methods, we construct a 3×3 matrix based on D_n, E_n and H_n to compute three new feature parameters λ, θ, ϕ.

The algorithmic steps are provided as follows:

(1) Set a window with width l, $l = 2^n, n = 1, 2, 3, 4, 5$ and move the window from the site l_0.

(2) Calculate the variation of each distribution at the two ends of the window,

$$\begin{aligned} \Delta D_l &= D_{l+l_0} - D_{l_0}, \\ \Delta E_l &= E_{l+l_0} - E_{l_0}, \\ \Delta H_l &= H_{l+l_0} - H_{l_0}. \end{aligned}$$

(3) Shift the window sequentially from the beginning site $l_0 = 1$ to $l_0 = 2$ and so on, up to $l_0 = N - l + 1$, where N is the length of the sequence. For each value of l_0 starting from 1 to $N - l + 1$, calculate each corresponding $\Delta D_l, \Delta E_l, \Delta H_l$.

(4) Define the fluctuation functions

$$\begin{aligned}
\rho_{DD}(l) &= \sqrt{((\overline{\Delta D_l \Delta D_l}) - (\overline{\Delta D_l})(\overline{\Delta D_l}))^2}, \\
\rho_{EE}(l) &= \sqrt{((\overline{\Delta E_l \Delta E_l}) - (\overline{\Delta E_l})(\overline{\Delta E_l}))^2}, \\
\rho_{HH}(l) &= \sqrt{((\overline{\Delta H_l \Delta H_l}) - (\overline{\Delta H_l})(\overline{\Delta H_l}))^2},
\end{aligned}$$

$$\begin{aligned}
\rho_{DE}(l) = \rho_{ED}(l) &= \sqrt{((\overline{\Delta D_l \Delta E_l}) - (\overline{\Delta D_l})(\overline{\Delta E_l}))^2}, \\
\rho_{DH}(l) = \rho_{HD}(l) &= \sqrt{((\overline{\Delta D_l \Delta H_l}) - (\overline{\Delta D_l})(\overline{\Delta H_l}))^2}, \\
\rho_{EH}(l) = \rho_{HE}(l) &= \sqrt{((\overline{\Delta E_l \Delta H_l}) - (\overline{\Delta H_l})(\overline{\Delta E_l}))^2},
\end{aligned}$$

where the bars indicate an average over all site l_0 in the sequence. The matrix of fluctuation functions is as follows:

$$F = \begin{pmatrix} \rho_{DD}(l) & \rho_{DE}(l) & \rho_{DH}(l) \\ \rho_{ED}(l) & \rho_{EE}(l) & \rho_{EH}(l) \\ \rho_{HD}(l) & \rho_{HE}(l) & \rho_{HH}(l) \end{pmatrix}$$

Obviously F is a real and symmetric matrix. Denote the three eigenvalues of F by ϵ_1, ϵ_2 and ϵ_3, and let $\epsilon_1 > \epsilon_2 > \epsilon_3$.

Based on fluctuation analysis, we find that

$$\epsilon_1 \propto l^{\lambda}, \epsilon_2 \propto l^{\theta}, \epsilon_3 \propto l^{\phi},$$

where λ, θ and ϕ are three parameters, determined by the slopes in the log-log plots. In other words, ϵ_i is a proportional function of l^j, $i = 1, 2, 3$ and $j = \lambda, \theta, \phi$, i.e., $\epsilon_i = a \times l^j$, $a \neq 0$. Because of the nonlinear scaling of the axes, a function of the form $y = a \times l^b$ will appear as a straight line on a log-log graph, in which b is the slope of the line. Therefore, the parameters λ, θ, ϕ can be computed by estimating each slope of log-log graph corresponding to ϵ_1, ϵ_2 and ϵ_3 from the numerical data.

(5) Estimate the slope λ, θ, ϕ of each log-log graph corresponding to $\epsilon_1, \epsilon_2, \epsilon_3$ computed in step (4).

For any given gene sequence, we can calculate three parameters λ, θ, ϕ by using these 5 algorithmic steps.

Using this same procedure, if we construct the matrix based on x_n, y_n and z_n, the coordinates of Z-curve, then D_n, E_n, H_n will be replaced by x_n, y_n, z_n respectively and α, β, γ can be computed eventually in the same algorithm by setting the window size $l = 2^n$, $n = 1, 2, 3, 4, 5,\ 6, 7$. The detail can also be available in (60).

Zhang et al found that when $l = 2^n$, $n = 1, 2, 3, 4, 5, 6, 7$, these seven points can be fitted by a line very well for calculating α, β and γ. We adopt these seven points and found that the line is not fitted very well for calculating λ, θ, ϕ in some cases when $l = 2^6$ and $l = 2^7$, especially in the case of calculating the value of slope ϕ. So the values of $l = 2^n$, $n = 1, 2, 3, 4, 5$ are adopted in order to reduce the error for determining the slope ϕ and improve the computational efficiency. Besides, the line fitted by those l's is perfect. Even if the linearity is not so perfect in several cases, the squared error E^2 with respect to the slope and intercept parameters is minimized and the unique straight line can also be obtained by performing a least-squares fit of the data.

After determining λ, θ, ϕ, we have a feature vector consisting of parameters $\alpha, \beta, \gamma, \lambda, \theta$, ϕ and σ, where σ is the standard deviation of $(\alpha, \beta, \gamma, \lambda, \theta, \phi)$. Then a machine learning method based on a support vector machine (SVM) combined with a Gaussian radial basis kernel function (RBF) and a 7-feature vector was developed for prediction of intron-less and intron-containing genes based only on the primary sequences.

4.2.3 Support Vector Machine (SVM)

In this section, we will sketch the idea of support vector machine (SVM). The SVM was first introduced by Vapnik (61), which is a supervised learning method that analyzes data used for classification and regression analysis. The standard SVM takes a set of input data, and predicts, for each given input, which of two possible classes the input is a member of, which makes the SVM a non-probabilistic binary linear classifier. Since an SVM is a classifier, then given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that predicts whether a new example falls into one category or the other. Intuitively, an SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on. More formally, a support vector machine constructs a hyperplane or set of hyperplanes in a high or infinite dimensional space, which can be used for classification, regression or other tasks. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training data points of any class (so-called functional margin), since in general the larger the margin the lower the generalization error of the classifier.

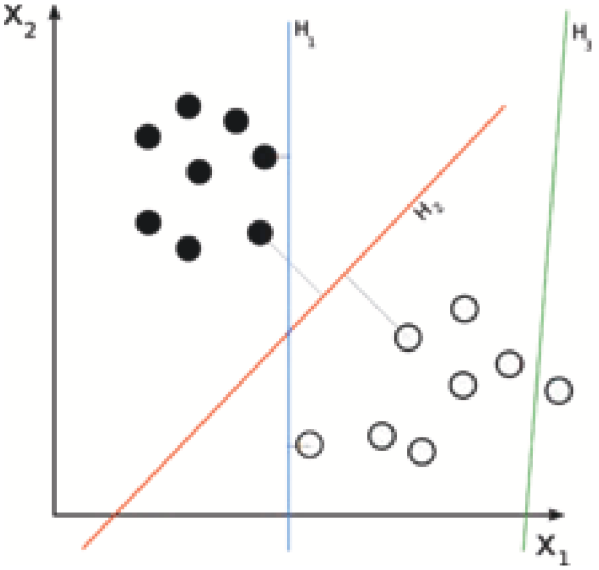

Figure 16. A simple SVM line (red) separates two classes of points with maximum margin

H3 (green) does not separate the two classes. H1 (blue) does, with a small margin and H2 (red) with the maximum margin.

Formalization.

We are given some training data $\mathcal{D}$, a set of n points of the form

$$\mathcal{D} = \{(x_i, y_i) | x_i \in \mathbb{R}^p, y_i \in \{-1, 1\}\}_{i=1}^{n} \tag{4.1}$$

where the y_i is either 1 or -1, indicating the class to which the point x_i belongs. Each x_i is a p-dimensional real vector. We want to find the maximum-margin hyperplane that divides the points having $y_i = 1$ from those having $y_i = -1$. Any hyperplane can be written as the set of points $\mathbf{x}$ satisfying

$$\mathbf{w} \cdot \mathbf{x} - b = 0, \quad \text{where} \cdot \text{denotes the dot product} \tag{4.2}$$

The vector $\mathbf{w}$ is a normal vector: it is perpendicular to the hyperplane. The parameter $\frac{b}{||\mathbf{w}||}$ determines the offset of the hyperplane from the origin along the normal vector $\mathbf{w}$.

We want to choose the $\mathbf{w}$ and b to maximize the margin, or distance between the parallel hyperplanes that are as far apart as possible while still separating the data. These hyperplanes can be described by the equations $\mathbf{w} \cdot \mathbf{x} - b = 1$ and $\mathbf{w} \cdot \mathbf{x} - b = -1$.

Note that if the training data are linearly separable, we can select the two hyperplanes of the margin in a way that there are no points between them and then try to maximize their distance. By using geometry, we find the distance between these two hyperplanes is $\frac{2}{||\mathbf{w}||}$, so we want to minimize $||\mathbf{w}||$. As we also have to prevent data points falling into the margin, we add the following constraint: for each i either

$$\mathbf{w} \cdot \mathbf{x_i} - b \geq 1 \qquad \text{for } x_i \text{ of the first class}$$

or

$$\mathbf{w} \cdot \mathbf{x_i} - b \leq -1 \qquad \text{for } x_i \text{ of the second class}$$

This can be written as:

$$y_i(\mathbf{w} \cdot \mathbf{x_i} - b) \geq 1, \qquad \text{for all } 1 \leq i \leq n \tag{4.3}$$

We can formalize this to get the optimization problem:

Minimize (in $\mathbf{w}, b$)

$$||\mathbf{w}||$$

subject to (for any $i = 1, 2, \ldots, n$)

$$y_i(\mathbf{w} \cdot \mathbf{x_i} - b) \geq 1.$$

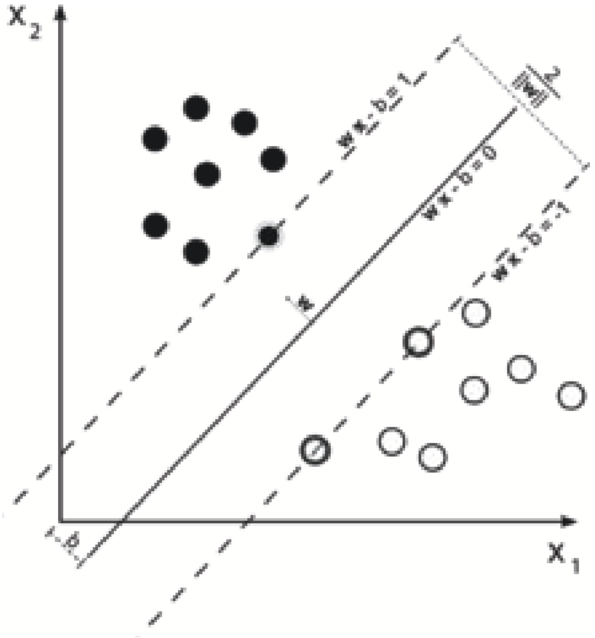

Figure 17. maximum-margin hyperplane for an SVM trained with samples from two classes. Samples on the margin are called the support vectors

Primal form

The optimization problem presented above is difficult to solve since it depends on $||\mathbf{w}||$, the norm of $\mathbf{w}$, which involves a square root. Thus we alter the equation by substituting $||\mathbf{w}||$ with

$$\frac{1}{2}||\mathbf{w}||^2$$

(the factor 1/2 being used for mathematical convenience) without changing the solution. This is a quadratic programming (QP) optimization problem. More clearly:

Minimize (in $\mathbf{w}$,b)

$$\frac{1}{2}||w||^2$$

subject to (for any $i = 1, 2, \ldots, n$)

$y_i(\mathbf{w} \cdot \mathbf{x_i} - b) \geq 1.$

This problem can be expressed by means of non-negative Lagrange multiplier α_i as

$$\min_{w,b} \max_{\alpha} \{J(\mathbf{w}, b, \alpha) : J(\mathbf{w}, b, \alpha) = \{\frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{n} \alpha_i [y_i(\mathbf{w} \cdot \mathbf{x_i} - b) - 1]\}\} \tag{4.4}$$

The solution to the constrained optimization problem is determined by the saddle point of the Lagrange function $J(\mathbf{w}, b, \alpha)$, which has to be minimized with respect to $\mathbf{w}$ and b; it also has to be maximized with respect to α. Thus differentiating $J(\mathbf{w}, b, \alpha)$ with respect to $\mathbf{w}$ and b and setting the results equal to zero, we get the following two conditions of optimality:

$$\text{Condition 1:} \frac{\partial J(\mathbf{w}, b, \alpha)}{\partial \mathbf{w}} = 0, \qquad \text{Condition 2:} \frac{\partial J(\mathbf{w}, b, \alpha)}{\partial b} = 0 \tag{4.5}$$

After rearrangement of terms, the Condition 1 yields

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x_i} \tag{4.6}$$

and the Condition 2 yields

$$\sum_{i=1}^{n} \alpha_i y_i = 0 \tag{4.7}$$

Given such a constrained optimization problem, scientists prefer to construct another problem called *dual problem.* The dual problem has the same optimal value as the primal problem, but with the Lagrange multipliers providing the optimal solution.

To postulate the dual problem for our primal problem, we first expand Eq.(4.4), term by term as follows:

$$J(\mathbf{w}, b, \alpha) = \frac{1}{2}\mathbf{w} \cdot \mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i \mathbf{w} \cdot \mathbf{x_i} - b\sum_{i=1}^{n} \alpha_i y_i + \sum_{i=1}^{n} \alpha_i \tag{4.8}$$

By Eq. (4.6,4.7), setting $J(\mathbf{w}, b, \alpha) = Q(\alpha)$, we can reformulate Eq.(4.8) as

$$Q(\alpha) = \sum_{i-1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x_i^T} \mathbf{x_j} = \sum_{i-1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j k(x_i, x_j) \tag{4.9}$$

Now the dual problem is stated as follows:

$$\text{maximize: } Q(\alpha) = \sum i = 1^n \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x_i^T} \cdot \mathbf{x_j} \tag{4.10}$$

$$\text{subject to: } \sum_{i=1}^{n} \alpha_i y_i = 0 \tag{4.11}$$

$$\alpha \geq 0 \tag{4.12}$$

Have determined the Lagrange multipliers α, we may compute the optimum weight $\mathbf{w}$ using Eq.(4.6).

4.2.4 Kernel Function

In Eq. (4.9), the kernel is defined by $k(x_i, x_j) = x_i \cdot x_j$. Some common kernels include Polynomial, Radial Basis Function, Hyperbolic tangent, etc. The effectiveness of SVM depends on the selection of kernel, the kernel's parameters, and soft margin parameter C. In fact The kernel function $K(\cdot, \cdot)$ dominates the learning capability of the SVM. In our dataset, we choose Radial Basis Kernel Function $K(x_i, x_j) = exp(-\gamma||x_i - x_j||^2)$, which has a single parameter γ, to predict the intronless from intron-containing genes.

More details about SVM are available in (61).

4.2.5 SVM Parameter Optimization

As in many multivariate statistical models, the performances of the SVM for classification depend on the combination of several parameters. In general, the SVM involves two classes of parameters: the penalty parameter C and kernel type K. C is a regularization parameter that controls the tradeoff between maximizing the margin and minimizing the training error. The kernel type K is another important parameter. In the radial basis function used in this study, γ is an important parameter to dominate the generalization ability of SVM by regulating the amplitude of the kernel function. Accordingly, two parameters, C and γ, should be optimized. The parameter optimization is performed by using a grid search approach within a limited range. Predict accuracy associated with mean-square-error is used to select the parameters:

$$\text{Predict accuracy} = 1 - \text{MSE}/(1-(-1))^2.$$

During SVM classification, each data point represents a pair (geneID, y); if the gene is experimentally intronless, y is assigned 1, otherwise y is -1.

4.2.6 K-fold Cross Validation

After all the seven parameters are determined, we can perform the K-fold cross validation to estimate the accuracy of our predictive model. In K-fold cross validation, the original sample is randomly partitioned into K subsamples. Of the K subsamples, a single subsample is retained as the validation data for testing the model, and the remaining K-1 subsamples are used as training data. The cross-validation process is then repeated K times (the folds), with each of the K subsamples used exactly once as the validation data. The K results from the folds then can be averaged (or otherwise combined) to produce a single estimation.

4.3 Applications and Results

The same dataset (60), one hundred intron-less genes and one hundred intron-containing genes selected randomly from Genbank and EMBL database, were used here. Parameters $\alpha, \beta, \gamma, \lambda, \theta$, ϕ and σ for both sets are calculated by using the algorithm in section 2. We randomly pick up 12 gene sequences as an illustration. The corresponding parameters are computed and listed in Table XVII. The first six DNA sequences are intron-less and the last six are intron-containing. The 7 feature parameters of each sequence can be computed by using the method described in section 4.2. Additional output suppressed to save space. Sevendimension numerical parameters for each sequence can be calculated. An optimal hyperplane for separating intron-less and intron-containing sequences can be obtained by implementing support vector machine classifier based on this seven-dimensional feature set.

Figure 18 shows an example: the linearity of log-log plots of genes on the value λ, θ and ϕ. We can see that eigenvalues ϵ_1, ϵ_2 and ϵ_3 are perfectly fitted by the lines with slope λ, θ and ϕ when $l = 2^n$, $n = 1, 2, 3, 4, 5$. This example shows the linearity of log-log plots based on our three feature parameters. The slope of each line, indicated on the graph, is λ, θ and ϕ respectively.

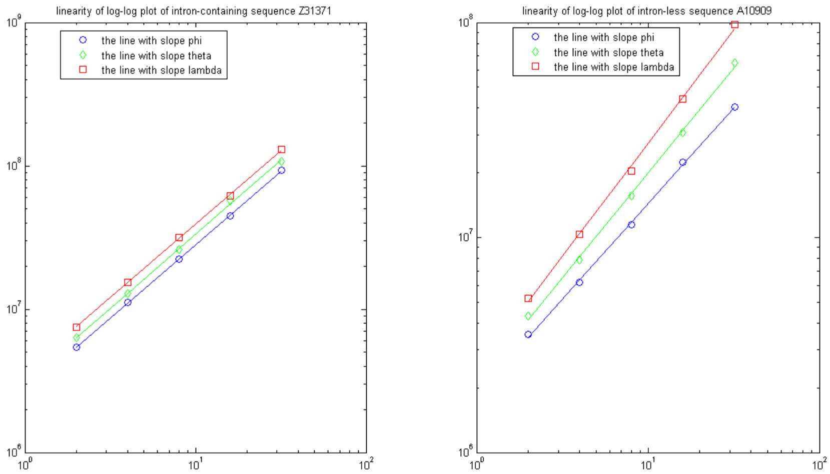

Figure 18. Linearity of log-log plots of gene Z31371 and A10909 randomly selected from the dataset

GENESCAN, EXONSCAN, N-SCAN, Zhang's Z-curve and our ZC method are implemented on this dataset in order to compare the results. Since the output of gene parsing and finding system provides us with the (predicted) beginning and ending coordinates of exons/introns in these sequences, it is easy for us to determine whether the gene is intron-bearing or not based on the prediction.

For Zhang's original dataset, 5-fold cross validation was performed. the original sample was partitioned into 5 (the folds) subsamples, each of which consists of 20 intron-less and 20 intron-containing genes. Of the 5 subsamples, a single subsample was retained as the validation data for testing the model, and the remaining 4 (i.e., $K - 1$ folds) subsamples were used as training dataset. By grid search method, the best model is obtained using $(C, \gamma) = (1, 0.125)$, yielding a misclassification error of 3.5%. The discriminant accuracy is defined as follows:

$$\mathrm{p} = \frac{\text{The number of all correct discriminations}}{\text{The number of sequences in the testing dataset}}.$$

For the intron-less genes, the prediction is regarded as "false" if it predicts the splice sites between exons and introns. This confirms the existence of introns. For the intron-containing genes, the prediction is regarded as "false" if prediction shows "single exon". For the intron-containing or intron-less genes which do contain exons, the prediction is regarded as "false" if the predicted answer is "no exon". The programs of GENESCAN, EXONSCAN, N-SCAN are performed directly on the testing set. In each round we calculate the value of p, then the mean value $\overline{p}$ can be obtained consequently (See Table XVIII). Table XVIII is the prediction results of the test datasets of Zhang's original dataset. Mean and variance are averaged by the results of five test sets. ZC method can accurately classify the gene sequences into intron-containing and intron-less with accuracy as high as 96.50%, which is better than Z-Curve (90.00%) and other state-of-the-art algorithms.

The Z-curve method and our new method are able to predict the intron-bearing status of genes with higher accuracy than the-state-of-the-art gene parsing algorithms.

To establish that the new four feature parameters account for the improved classification accuracy, we can run SVM with RBF on our new parameter set and Zhang's original feature set. 5-fold cross validation is performed on the same dataset (Table XIX).

In Table XIX, we perform support vector machine (SVM) with radial basis kernel function (RBF) on original parameters α, β, γ, new parameters$\lambda, \theta, \phi, \sigma$ and their combination, respectively. In the column of Methods, SVM+RBF 1 means that SVM with RBF is performed on Zhang's original feature parameters α, β, γ; The average accuracy is 91.00%. SVM+RBF 2: SVM with RBF is performed on the new four parameter sets λ, θ, ϕ and their standard deviation σ; The average accuracy is 94.00%. SVM+RBF 3: SVM with RBF is performed on the all 7 feature parameters (zhang's original and four new parameters); The average accuracy is 96.50%. The result shows that the accuracy of ZC method to classify intron-containing and intron-less sequences is not simply due to the superiority of SVM with RBF. The improved classification accuracy is mainly due to the combination of new parameters and Zhang's original parameters.

From the result, we can see that the high accuracy of ZC method is not simply due to the superiority of support vector machine. The result shows that the improved classification accuracy is mainly due to the combination of new feature set and Zhang's original feature set.

Another example of real biological data analysis. To avoid the statistical bias of small dataset, we use another dataset and test the result. This dataset contains 1000 intron-less genes (prokaryotic genome completely) which are selected randomly from UniProtKB/ Swiss-Prot (release 15.1) and 1000 intron-containing genes selected randomly from Genbank database (release 170). The classical gene parsing system new GENSCAN released on the website (http://genes.mit.edu/ GENSCAN.html), N-SCAN (http://mblab.wustl.edu/), Zhang's Z-curve and ZC method were implemented. To avoid the biasedness of the discriminant accuracy defined in (60), 5-fold cross-validation was used. During SVM classification, the parameter ranges were given as follows: $C \in (2^{-1}, \ldots, 2^{8})$, $\gamma \in (2^{-8}, \ldots, 2^{8})$. The best model in the parameter range is obtained using $C = 16$ and $\gamma = 0.015625$. The prediction error profile has a minimum value at $(C, \gamma) = (16, 2^{-6})$, indicating that the optimal values of C and γ for constructing SVM models are 16 and 2^{-6}, respectively. Please refer the Figure 19 for parameter selection.

Using the optimal values of C and γ, the prediction model was constructed based on the training set by using the SVM learning algorithm with the Gaussian radial kernel function. To minimize data dependence on the prediction model, fivefold cross validation sampling method described in section 4.2 was prepared. Each training set consisted of 1600 sequences; half of them were randomly selected from data of intron-less sequences, and the other half were randomly selected from data of intron-containing sequences. Each test set was constructed

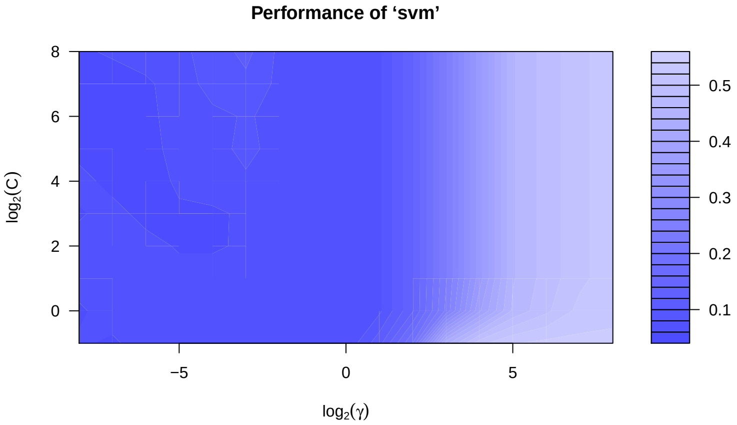

Figure 19. The best parameters for SVM model

with another 400 sequences. The prediction results are listed in Table XX and Figure 20. In Table XX and Figure 20, a larger dataset composed of 1000 intron-containing and 1000 intronless genes is used for training and testing. The accuracy of our method ZC is 93.05 ± 1.14% which is better than that of Z-curve (86.15%), N-SCAN (81.85%) and GENSCAN (76.60%).

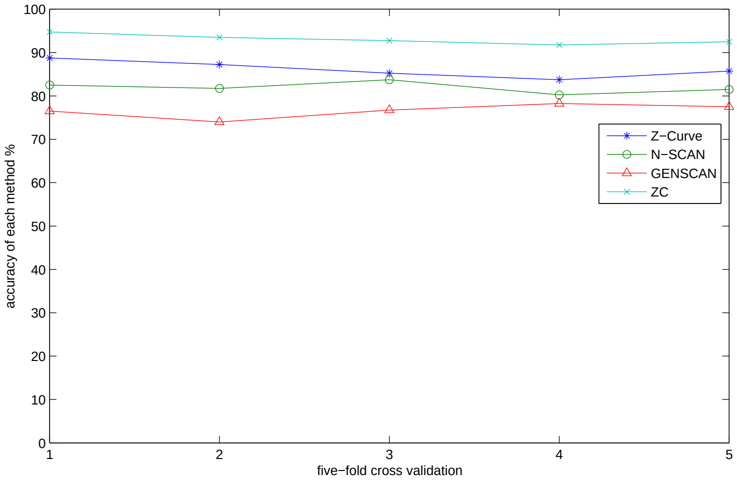

Figure 20. Comparison of our ZC method and Z-curve, GENSCAN and N-SCAN

4.4 Conclusion

In this work, a high-accurate predictive method has been proposed for the prediction of a given gene belonging to intron-containing or intron-less by coupling SVM with RBF. In comparison with previous literatures, the predictive performance has been significantly enhanced. It is anticipated that the current method can be a complementary tool for distinguishing intron-less genes from intron-containing genes. First, seven feature parameters α, β, γ, λ, θ, ϕ and σ can be computed using the algorithmic steps in section 4.2. Secondly, support vector machine classifier with radial basis kernel function can be performed on those seven parameters to classify the genes.

The parameter α was the only one proposed by Peng et al (58) to distinguish between the intron-less and intron-containing sequences; β and γ were proposed by Zhang et al (60); other four feature parameters λ, θ, ϕ and σ are introduced in ZC method. These feature parameters can be used to discover more information hidden within the genes. Support vector machine classifier with Gaussian radial basis kernel function(RBF) is implemented in ZC method.

Computation Environment The programs for calculating 7 feature vectors were written in Matlab and C++. The programs implementing SVM were written by using R package ”e1071” which was based on the core of the libsvm 2.8 package (http://www.csie.ntu.edu. tw/ cjlin/libsvm). All programs were run on a 2.66GHz CPU with 6 GB DDR3 SDRAM.

APPENDICES

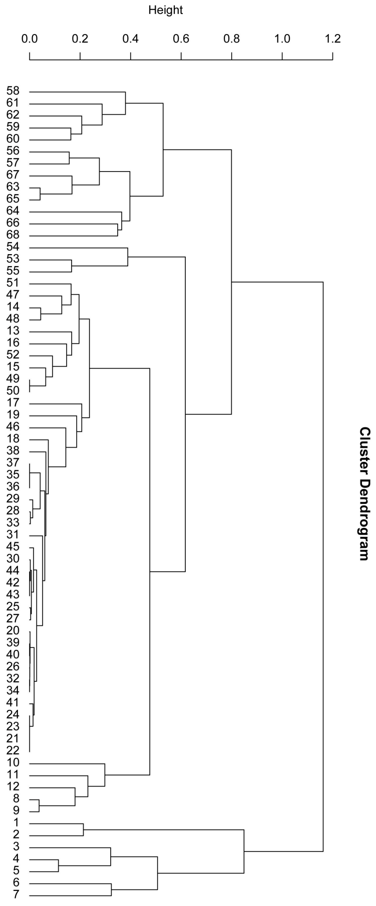

Figure 21. Phylogenetic tree of PB2 segmented gene of A H1N1 genomes

Classical swine: 1-5; human seasonal (H1N1): 6-7; American avian: 8-12; triple reassortant swine: 13-19, 46-52; new swine influenza A (H1N1): 20-45; Eurasian avian: 53-55; Eurasian swine: 56-68. These gene information is provided in Table VIII.

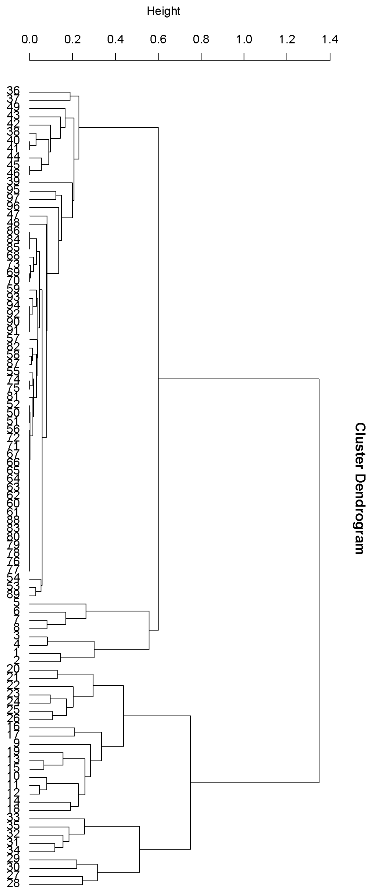

Figure 22. Phylogenetic tree of PB1 segmented gene of A H1N1 genomes

classical swine: 1-4; human seasonal (H1N1): 5-8; Eurasian swine: 9-19; Eurasian avian: 20-26; American avian: 27-30; human seasonal (H3N2): 31-35; triple reassortant swine: 36-49, 95- 97; new swine influenza A (H1N1): 50-94. These gene information is provided in Table IX.

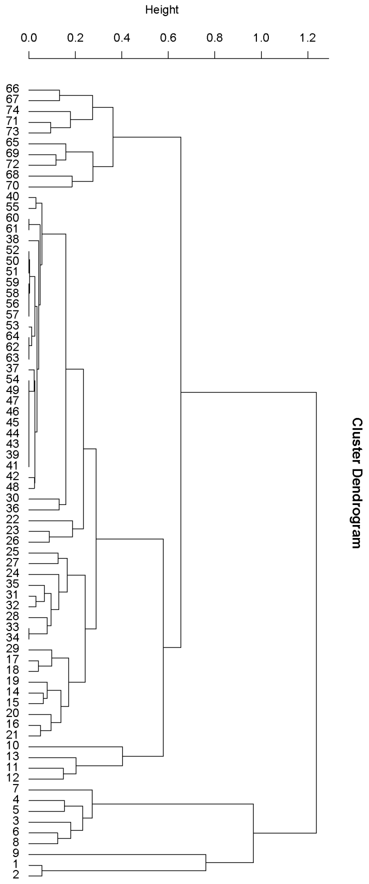

Figure 23. Phylogenetic tree of PA segmented gene of A H1N1 genomes

classical swine: 1-8; human seasonal (H1N1): 9; Eurasian avian: 10-13; American avian: 14-21; triple reassortant swine: 22-36; new swine influenza A (H1N1): 37-64; Eurasian swine: 65-74. These gene information is provided in Table X.

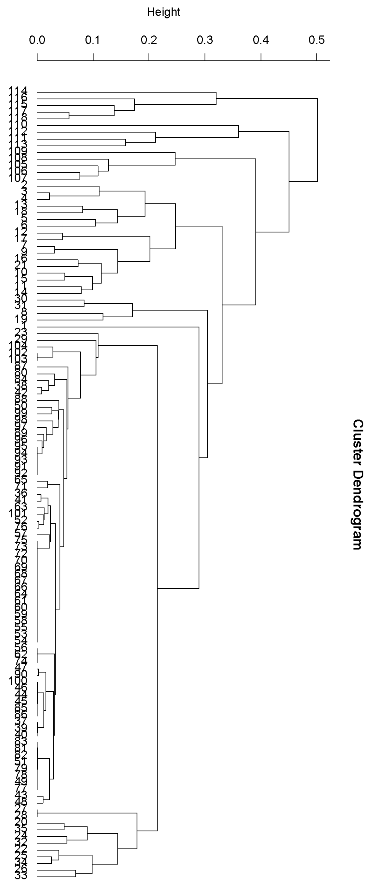

Figure 24. Phylogenetic tree of HA segmented gene of A H1N1 genomes

classical swine: 1-35; new swine influenza A (H1N1): 36-104; human seasonal: 105-109; American avian: 110- 113; Eurasian swine: 114-118. These gene information is provided in Table XI.

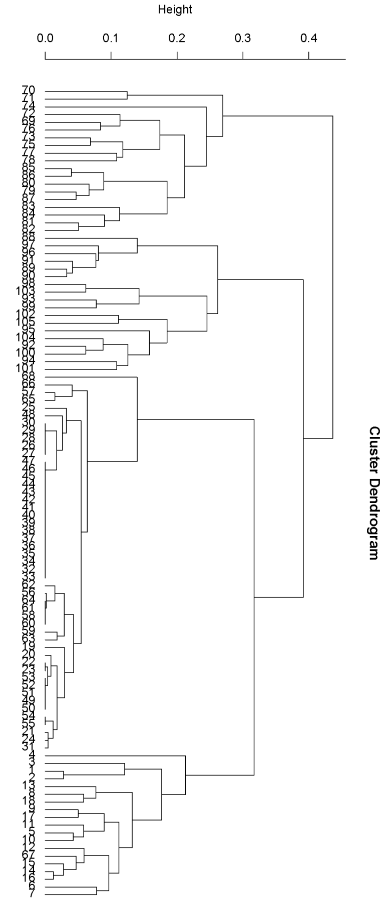

Figure 25. Phylogenetic tree of NP segmented gene of A H1N1 genomes

classical swine: 1-18, 67-68; new swine influenza A (H1N1): 19-66; human seasonal: 69; American avian: 70-78; Eurasian avian: 79-87; Eurasian swine: 88-105. These gene information is provided in Table XII.

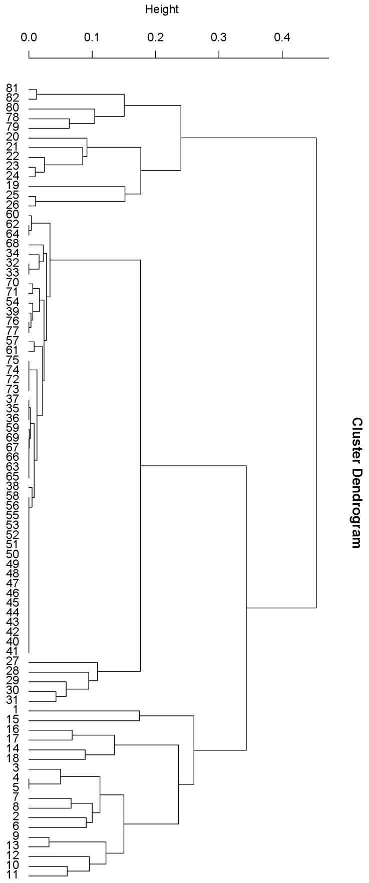

Figure 26. Phylogenetic tree of NA segmented gene of A H1N1 genomes

classical swine: 1-13; human seasonal: 14-18; American avian: 19-26; Eurasian swine: 27-31; new swine influenza A (H1N1): 32-77; Eurasian avian: 78-82. These gene information is provided in Table XIII.

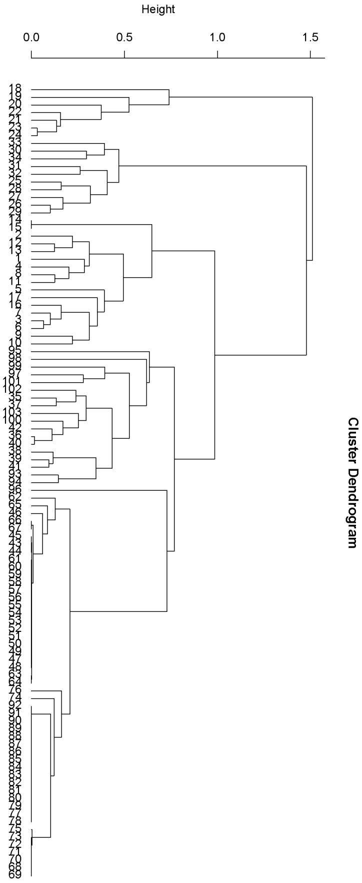

Figure 27. Phylogenetic tree of MP segmented gene of A H1N1 genomes

classical swine: 1-17; human seasonal: 18-24; American avian: 25-29; Eurasian avian: 30-34; Eurasian swine: 35-42, 93-103; new swine influenza A (H1N1): 43-92. These gene information is provided in Table XIV.

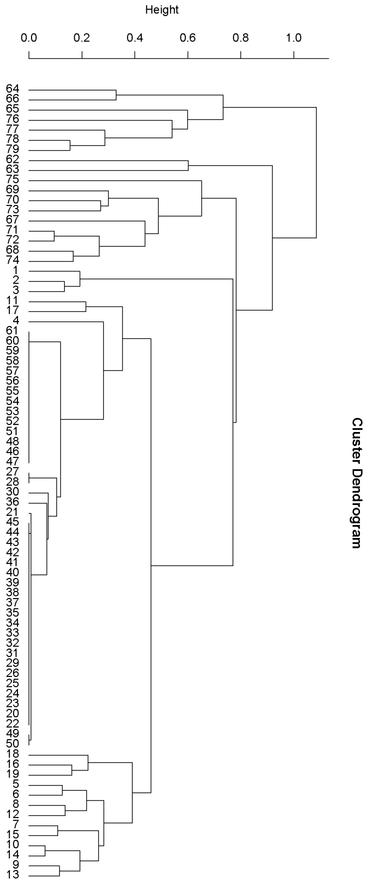

Figure 28. Phylogenetic tree of NS segmented gene of A H1N1 genomes

classical swine: 1-19; new swine influenza A (H1N1): 20-61; human seasonal: 62-63; Eurasian avian: 64-66; Eurasian swine: 67-75; American avian: 76-79. These gene information is provided in Table XV.

TABLE III: THE LIST OF 59 A H1N1 GENOMES

No. on tree	Genome Strain Designation
1	Influenza A virus(A/California/04/2009(H1N1))
2	Influenza A virus(A/New York/18/2009(H1N1))
3	Influenza A virus(A/Canada-ON/RV1527/2009(H1N1))
4	Influenza A virus(A/Mexico/InDRE4487/2009(H1N1))
5	Influenza A virus(A/Texas/09/2009(H1N1))
6	Influenza A virus(A/California/14/2009(H1N1))
7	Influenza A Virus(A/New York/1669/2009(H1N1))
8	Influenza A Virus(A/New York/1682/2009(H1N1))
9	Influenza A virus(A/Canada-AB/RV1532/2009(H1N1))
10	Influenza A virus(A/Canada-NS/RV1536/2009(H1N1))
11	Influenza A virus(A/Canada-NS/RV1538/2009(H1N1))
12	Influenza A virus(A/Canada-ON/RV1526/2009(H1N1))
13	Influenza A virus(A/Canada-ON/RV1529/2009(H1N1))
14	Influenza A virus(A/Mexico/inDRE4114/2009(H1N1))
15	Influenza A virus(A/New York/3008/2009(H1N1))
16	Influenza A virus(A/New York/3014/2009(H1N1))
17	Influenza A virus(A/New York/3099/2009(H1N1))
18	Influenza A virus(A/swine/Alberta/OTH-33-8/2009(H1N1))
19	Influenza A virus(A/Hamburg/4/2009(H1N1))
20	Influenza A virus(A/England/195/2009(H1N1))

21	Influenza A virus (A/swine/Alberta/56626/03(H1N1))
22	Influenza A virus (A/swine/California/T9001707/1991(H1N1))
23	Influenza A virus (A/swine/OH/511445/2007(H1N1))
24	Influenza A virus (A/swine/Memphis/1/1990(H1N1))
25	Influenza A virus (A/swine/Iowa/31483/1988(H1N1))
26	Influenza A virus (A/swine/Ontario/55383/04(H1N2))
27	Influenza A virus (A/swine/Kansas/3228/1987(H1N1))
28	Influenza A virus (A/swine/Wisconsin/10/1998(H1N1)
29	Influenza A virus (A/swine/Denmark/WVL9/1993(H1N1))
30	Influenza A virus (A/swine/Spain/50047/2003(H1N1))
31	Influenza A virus (A/swine/England/WVL15/1997(H1N1))
32	Influenza A virus (A/swine/France/WVL4/1985(H1N1))
33	Influenza A virus (A/swine/Italy/671/1987(H1N1))
34	Influenza A virus (A/swine/Tianjin/01/2004(H1N1))
35	Influenza A virus (A/swine/Ratchaburi/NIAH1481/2000(H1N1))
36	Influenza A virus (A/SW/KS/13481-S/00(H1N2))
37	Influenza A virus (A/duck/NJ/7717-70/1995(H1N1))
38	Influenza A virus (A/blue winged teal/TX/27/2002(H1N1))
39	Influenza A virus (A/mallard/Maryland/42/2003(H1N1))
40	Influenza A virus (A/mallard/MN/330/1999(H3N1))
41	Influenza A virus (A/duck/Nanchang/4-165/2000(H4N6))
42	Influenza A virus (A/Duck/NY/185502/2002(H5N2))

43	Influenza A virus (A/chicken/Chis/15224/1997(H5N2))
44	Influenza A virus (A/duck/Italy/69238/2007(H1N1))
45	Influenza A virus (A/crested eagle/Belgium/01/2004(H5N1))
46	Influenza A virus (A/swan/Germany/R65/2006(H5N1))
47	Influenza A virus (A/Cygnus olor/Astrakhan/Ast05-2-1/2005(H5N1))
48	Influenza A virus (A/chicken/Crimea/08/2005(H5N1))
49	Influenza A virus (A/blue-winged teal/Ohio/1864/2006(H3N8))
50	Influenza A virus (A/chicken/Jiangsu/cz1/2002(H5N1))
51	Influenza A virus (A/egret/Hong Kong/757.2/2003(H5N1))
52	Influenza A virus (A/duck/Yokohama/aq10/2003(H5N1))
53	Influenza A virus (A/chicken/Korea/ES/03(H5N1))
54	Influenza A virus (A/Puerto Rico/8/34(H1N1))
55	Influenza A virus (A/New Caledonia/20/1999(H1N1))
56	Influenza A virus (A/Wisconsin/67/2005(H3N2))
57	Influenza A virus (A/New York/146/2000(H1N1))
58	Influenza A virus (A/Albany/20/1978(H1N1))
59	Influenza A virus (A/Wyoming/03/2003(H3N2))

TABLE IV: THE LIST OF HRV GENOMES

Number	Genome names on tree	Accession Number	species	seq length
1	cva-13	AF499637	HEV-C	7458
2	cva-21	AF546702	HEV-C	7406
3	pv-1m	V01149	HEV-C	7440
4	pv-2	M12197	HEV-C	7440
5	pv-3	K01392	HEV-C	7431
6	cb1	M16560	HEV-B	7389
7	cb2	AF081485	HEV-B	7403
8	cb3	M33854	HEV-B	7399
9	hrv-03	DQ473485	HRV-B	7208
10	hrv-04	DQ473490	HRV-B	7212
11	hrv-05	FJ445112	HRV-B	7212
12	hrv-06	DQ473486	HRV-B	7216
13	hrv-14	L05355	HRV-B	7212
14	hrv-17	EF173420	HRV-B	7219
15	hrv-26	FJ445124	HRV-B	7211
16	hrv-27	FJ445186	HRV-B	7217
17	hrv-35	FJ445187	HRV-B	7224
18	hrv-37	EF173423	HRV-B	7216
19	hrv-42	FJ445130	HRV-B	7223
20	hrv-48	DQ473488	HRV-B	7214

21	hrv-52	FJ445188	HRV-B	7216
22	hrv-69	FJ445151	HRV-B	7211
23	hrv-70	DQ473489	HRV-B	7223
24	hrv-72	FJ445153	HRV-B	7216
25	hrv-79	FJ445155	HRV-B	7224
26	hrv-83	FJ445161	HRV-B	7230
27	hrv-84	FJ445162	HRV-B	7201
28	hrv-86	FJ445164	HRV-B	7213
29	hrv-91	FJ445168	HRV-B	7221
30	hrv-92	FJ445169	HRV-B	7233
31	hrv-93	EF173425	HRV-B	7215
32	hrv-97	FJ445172	HRV-B	7207
33	hrv-99	FJ445174	HRV-B	7208
34	hrv-01	FJ445111	HRV-A	7137
35	hrv-02	X02316	HRV-A	7102
36	hrv-07	FJ445176	HRV-A	7146
37	hrv-08	FJ445113	HRV-A	7108
38	hrv-09	FJ445177	HRV-A	7132
39	hrv-10	FJ445178	HRV-A	7137
40	hrv-11	EF173414	HRV-A	7125
41	hrv-12	EF173415	HRV-A	7124
42	hrv-13	FJ445116	HRV-A	7140

43	hrv-15	DQ473493	HRV-A	7134
44	hrv-16	L24917	HRV-A	7124
45	hrv-18	FJ445118	HRV-A	7119
46	hrv-19	FJ445119	HRV-A	7135
47	hrv-20	FJ445120	HRV-A	7163
48	hrv-21	FJ445121	HRV-A	7134
49	hrv-22	FJ445122	HRV-A	7129
50	hrv-23	DQ473497	HRV-A	7025
51	hrv-24	FJ445190	HRV-A	7132
52	hrv-25	FJ445123	HRV-A	7126
53	hrv-28	DQ473508	HRV-A	7148
54	hrv-29	FJ445125	HRV-A	7123
55	hrv-30	FJ445179	HRV-A	7099
56	hrv-31	FJ445126	HRV-A	7131
57	hrv-32	FJ445127	HRV-A	7133
58	hrv-33	FJ445128	HRV-A	7133
59	hrv-34	FJ445189	HRV-A	7119
60	hrv-36	DQ473505	HRV-A	7141
61	hrv-38	FJ445180	HRV-A	7136
62	hrv-39	AY751783	HRV-A	7136
63	hrv-40	FJ445129	HRV-A	7138
64	hrv-41	DQ473491	HRV-A	7145

65	hrv-43	FJ445131	HRV-A	7129
66	hrv-44	DQ473499	HRV-A	7123
67	hrv-45	FJ445132	HRV-A	7114
68	hrv-46	DQ473506	HRV-A	7149
69	hrv-47	FJ445133	HRV-A	7132
70	hrv-49	DQ473496	HRV-A	7109
71	hrv-50	FJ445135	HRV-A	7118
72	hrv-51	FJ445136	HRV-A	7152
73	hrv-53	DQ473507	HRV-A	7143
74	hrv-54	FJ445138	HRV-A	7134
75	hrv-55	DQ473511	HRV-A	7036
76	hrv-56	FJ445140	HRV-A	7136
77	hrv-57	FJ445141	HRV-A	7134
78	hrv-58	FJ445142	HRV-A	7140
79	hrv-59	DQ473500	HRV-A	7135
80	hrv-60	FJ445143	HRV-A	7139
81	hrv-61	FJ445144	HRV-A	7139
82	hrv-62	FJ445145	HRV-A	7131
83	hrv-63	FJ445146	HRV-A	7141
84	hrv-64	FJ445181	HRV-A	7129
85	hrv-65	FJ445147	HRV-A	7162
86	hrv-66	FJ445148	HRV-A	7139

87	hrv-67	FJ445149	HRV-A	7135
88	hrv-68	FJ445150	HRV-A	7164
89	hrv-71	FJ445152	HRV-A	7161
90	hrv-73	DQ473492	HRV-A	7140
91	hrv-74	DQ473494	HRV-A	7120
92	hrv-75	DQ473510	HRV-A	7137
93	hrv-76	FJ445182	HRV-A	7128
94	hrv-77	FJ445154	HRV-A	7136
95	hrv-78	FJ445183	HRV-A	7145
96	hrv-80	FJ445156	HRV-A	7138
97	hrv-81	FJ445157	HRV-A	7116
98	hrv-82	FJ445160	HRV-A	7123
99	hrv-85	FJ445163	HRV-A	7140
100	hrv-88	DQ473504	HRV-A	7143
101	hrv-89	FJ445184	HRV-A	7152
102	hrv-90	FJ445167	HRV-A	7124
103	hrv-94	FJ445185	HRV-A	7132
104	hrv-95	FJ445170	HRV-A	7110
105	hrv-96	FJ445171	HRV-A	7134
106	hrv-98	FJ445173	HRV-A	7133
107	hrv-100	FJ445175	HRV-A	7140
108	qpm	EF186077	HRV-C	6917

109	nat001	EF077279	HRV-C	7079
110	c024	EF582385	HRV-C	7099
111	nat045	EF077280	HRV-C	7015
112	c026	EF582387	HRV-C	7086
113	c025	EF582386	HRV-C	7114

TABLE V: THE LIST OF 36 MAMMALIAN MITOCHONDRIAL GENOMES

Number	Genome names on tree	Accession Number
1	human	V00662
2	pigmy chimpanzee	D38116
3	common chimpanzee	D38113
4	gibbon	X99256
5	cheetah	NC_005212
6	vervet monkey	AY863426
7	ape	NC_002764
8	bornean orangutan	D38115
9	sumatran orangutan	NC_002083
10	gorilla	D38114
11	cat	U20753
12	dog	U96639
13	pig	AJ002189
14	sheep	AF010406
15	goat	AF533441
16	cow	V00654
17	buffalo	AY488491
18	wolf	EU442884
19	tiger	EF551003

20	leopard	EF551002
21	indian rhinoceros	X97336
22	white rhinoceros	Y07726
23	Indus river dolphin	NC_005275
24	brown bear	AF303110
25	polar bear	AF303111
26	Clouded leopard	NC_008450
27	rabbit	AJ001588
28	hedgehog	X88898
29	dormouse	AJ001562
30	squirrel	AJ238588
31	baleen whale	X72204
32	gray seal	NC_001602
33	harbor seal	NC_001325
34	coyote	DQ480511
35	pika	NC_011029
36	Hippopotamus	AP003425

TABLE VI: THE LIST OF 21 GENOMES USED FOR COMPARISONS TO SUBSTITUTIONS AND INDELS

name on tree	Genome Strain Description
A(H1N1)-1	Influenza A virus(A/New York/18/2009(H1N1))
A(H1N1)-2	Influenza A virus(A/Canada-ON/RV1527/2009(H1N1))
A(H1N1)-3	Influenza A virus(A/Mexico/InDRE4487/2009(H1N1))
A(H1N1)-4	Influenza A virus(A/Texas/09/2009(H1N1))
A(H1N1)-5	Influenza A virus(A/California/14/2009(H1N1))
A(H1N1)-6	Influenza A Virus(A/New York/1669/2009(H1N1))
swine1	Influenza A virus (A/swine/Alberta/56626/03(H1N1))
swine2	Influenza A virus (A/swine/California/T9001707/1991(H1N1))
swine3	Influenza A virus (A/swine/Nebraska/123/1977(H1N1))
swine4	Influenza A virus (A/swine/Memphis/1/1990(H1N1))
swine5	Influenza A virus (A/swine/Iowa/31483/1988(H1N1))
swine6	Influenza A virus (A/swine/Ontario/55383/04(H1N2))
seasonal1	Influenza A virus (A/Puerto Rico/8/34(H1N1))
seasonal2	Influenza A virus (A/New Caledonia/20/1999(H1N1))
seasonal3	Influenza A virus (A/Wisconsin/67/2005(H3N2))
avian1	Influenza A virus (A/duck/NJ/7717-70/1995(H1N1))
avian2	Influenza A virus (A/blue winged teal/TX/27/2002(H1N1))
avian3	Influenza A virus (A/mallard/Maryland/42/2003(H1N1))
avian4	Influenza A virus (A/mallard/MN/330/1999(H3N1))

avian5	Influenza A virus (A/blue-winged teal/Ohio/1864/2006(H3N8))
avian6	Influenza A virus (A/Duck/NY/185502/2002(H5N2))

TABLE VII: THE LIST OF 128 PKC FAMILY

Number	GenBank ID	PKC Cluster	Number	GenBank ID	PKC Cluster
1	NP_001006133	nPKC	65	Q4AED6	cPKC
2	NP_001008716	nPKC	66	Q4R4U2	cPKC
3	NP_001012707	aPKC	67	Q5R4K9	aPKC
4	O01715	cPKC	68	Q5TZD4	nPKC
5	O17874	PRK	69	Q62074	aPKC
6	O19111	aPKC	70	Q62101	PKCmu
7	O42632	PKC1	71	Q64617	nPKC
8	O61224	nPKC	72	Q69G16	cPKC
9	O61225	nPKC	73	Q6AZF7	cPKC
10	O62567	cPKC	74	Q6BI27	PKC1
11	O62569	nPKC	75	Q6C292	PKC1
12	O76850	cPKC	76	Q6DCJ8	nPKC
13	O94806	PKCmu	77	Q6DUV1	nPKC
14	O96942	cPKC	78	Q6FJ43	PKC1
15	O96997	cPKC	79	Q6GNZ7	nPKC
16	P04409	cPKC	80	Q6P5Z2	PRK
17	P05126	cPKC	81	Q6P748	PRK
18	P05129	cPKC	82	Q6UB96	PKC1
19	P05130	cPKC	83	Q6UB97	PKC1
20	P05696	cPKC	84	Q75BT0	PKC1

21	P05771	cPKC	85	Q76G54	PKC1
22	P05772	cPKC	86	Q7LZQ8	cPKC
23	P09215	nPKC	87	Q7LZQ9	cPKC
24	P09216	nPKC	88	Q7QCP8	nPKC
25	P09217	aPKC	89	Q7SY24	cPKC
26	P10102	cPKC	90	Q7SZH7	nPKC
27	P10829	cPKC	91	Q7SZH8	nPKC
28	P10830	nPKC	92	Q7T2C5	cPKC
29	P13677	cPKC	93	Q86XJ6	nPKC
30	P16054	nPKC	94	Q86ZV2	PKC1
31	P17252	cPKC	95	Q873Y9	PKC1
32	P20444	cPKC	96	Q8IUV5	PRK
33	P23298	nPKC	97	Q8J213	PKC1
34	P24583	PKC1	98	Q8JFZ9	cPKC
35	P24723	nPKC	99	Q8K1Y2	PKCmu
36	P28867	nPKC	100	Q8K2K8	nPKC
37	P34885	nPKC	101	Q8MXB6	nPKC
38	P36582	PKC1	102	Q8NE03	nPKC
39	P36583	PKC1	103	Q90XF2	aPKC
40	P41743	aPKC	104	Q91569	aPKC
41	P43057	PKC1	105	Q91948	PRK
42	P63318	cPKC	106	Q99014	PKC1

43	P68403	cPKC	107	Q9BZL6	PKCmu
44	P68404	cPKC	108	Q9GSZ3	aPKC
45	P87253	PKC1	109	Q9HF10	PKC1
46	P90980	cPKC	110	Q9HGK8	PKC1
47	Q00078	PKC1	111	Q9UVJ5	PKC1
48	Q02111	nPKC	112	Q9Y792	PKC1
49	Q02156	nPKC	113	Q9Y7C1	PKC1
50	Q02956	aPKC	114	XM_391874	cPKC
51	Q04759	nPKC	115	XP_001066028	nPKC
52	Q05513	aPKC	116	XP_001116804	cPKC
53	Q05655	nPKC	117	XP_001147999	nPKC
54	Q15139	PKCmu	118	XP_001250401	nPKC
55	Q16974	cPKC	119	XP_234108	PKCmu
56	Q16975	nPKC	120	XP_421417	nPKC
57	Q19266	aPKC	121	XP_540151	PKCmu
58	Q25378	cPKC	122	XP_541432	cPKC
59	Q28EN9	aPKC	123	XP_583587	nPKC
60	Q2NKI4	cPKC	124	XP_602125	cPKC
61	Q2U6A7	PKC1	125	XP_683138	nPKC
62	Q3UEA6	PRK	126	XP_849292	nPKC
63	Q498G7	nPKC	127	XP_851386	PKCmu
64	Q4AED5	aPKC	128	XP_851861	nPKC

TABLE VIII: THE LIST OF 50 BETA-GLOBIN PROTEINS

Beta globin	Accession No	Beta globin	Accession No	Beta globin	Accession No
Human	AAA16334	Goshawk	P08851	Lesser panda	P18982
Giant panda	P18983	Sheep	P02075	Duck	P02114
Mallard	P02115	Goose	P02118	Rat	P02090
Penguin	P80216	Swift	P15165	Coyote	P60526
Catfish	O13163	Bison	P09422	Swan	P68944
Buffalo	P67819	Dog	P60524	Chimpanzee	P68873
Dolphin	P18990	Goldfish	P02140	Polar bear	P68011
Rhinoceros	P02066	Chicken	CAA23700	Wolf	P23020
Turtle	P13274	Pigeon	P11342	Black bear	P68012
Indian elephant	P02084	African elephant	P02085	Tortoise	P83123
Grivet	P08028	Gorilla	P02024	Shark	P02143
Hippopotamus	P19016	Horse	P02062	Gibbon	P02025
Whale	P18984	Bat	P24660	Redfox	P21201
Marmot	P08853	Salmon	CAA49580	Sparrow	P07406
Pheasant	P02113	Flamingo	P02121	Pig	CAA60490
Dragonfish	P29625	Parakeet	P21668	Zebra	P67824
Cod	O13077	Langur	P02032		

TABLE IX: THE LIST OF PB2 GENES

No. on tree	Access ID.	Strain Description
1	M73515	A/swine/Iowa/15/1930(H1N1)
2	M55469	A/swine/1976/1931(H1N1)
3	CY026146	A/Wisconsin/301/1976(H1N1)
4	DQ280205	A/swine/Ontario/55383/04(H1N2)
5	DQ280189	A/swine/Ontario/57561/03(H1N1)
6	CY033629	A/New Caledonia/20/1999(H1N1)
7	CY034123	A/Wisconsin/67/2005(H3N2)
8	EU026109	A/duck/NY/13152-13/1994(H1N1)
9	AY619970	A/swine/Ontario/42729A/01(H3N3)
10	EU026021	A/mallard/MD/161/2002(H1N1)
11	AY619954	A/swine/Saskatchewan/18789/02(H1N1)
12	EU880827	A/turkey/CA/358533/2005(H4N8)
13	AF285892	A/Swine/Ontario/01911-1/99 (H4N6)
14	AF455736	A/Swine/Indiana/P12439/00 (H1N2)
15	AF250131	A/Swine/Indiana/9K035/99 (H1N2)
16	EU301177	A/swine/Korea/JNS06/2004(H3N2)
17	EU015993	A/swine/Guangxi/13/2006(H1N2)
18	AF455733	A/Swine/North Carolina/93523/01 (H1N2)
19	AY233387	A/duck/NC/91347/01(H1N2)
20	GQ160597	A/Nebraska/03/2009(H1N1)

21	GQ168878	A/New York/15/2009(H1N1)
22	GQ168876	A/New York/10/2009(H1N1)
23	FJ984351	A/New York/18/2009(H1N1)
24	GQ117021	A/New York/22/2009(H1N1)
25	GQ117035	A/California/14/2009(H1N1)
26	GQ149632	A/Mexico/4604/2009(H1N1)
27	GQ168870	A/Indiana/09/2009(H1N1)
28	GQ117076	A/Arizona/02/2009(H1N1)
29	FJ998206	A/Mexico/InDRE4487/2009(H1N1)
30	CY039908	A/New York/1682/2009(H1N1)
31	GQ132138	A/Mexico/InDRE4114/2009(H1N1)
32	GQ162180	A/Mexico/4108/2009(H1N1)
33	GQ149617	A/Mexico/4486/2009(H1N1)
34	GQ162168	A/Mexico/4603/2009(H1N1)
35	FJ966079	A/California/04/2009(H1N1)
36	FJ966976	A/California/07/2009(H1N1)
37	FJ984365	A/California/08/2009(H1N1)
38	GQ117089	A/Texas/07/2009(H1N1)
39	GQ117070	A/Minnesota/02/2009(H1N1)
40	FJ966955	A/California/05/2009(H1N1)
41	FJ966963	A/California/06/2009(H1N1)
42	GQ117047	A/Texas/08/2009(H1N1)

43	GQ168885	A/Texas/04/2009(H1N1)
44	GQ168879	A/Texas/06/2009(H1N1)
45	GQ117027	A/Texas/09/2009(H1N1)
46	AF342824	A/Wisconsin/10/98 (H1N1)
47	EU798929	A/swine/Korea/CAS05/2004(H3N2)
48	AF455737	A/Swine/Illinois/100085A/01 (H1N2)
49	DQ469987	A/swine/Ontario/33853/2005(H3N2)
50	DQ469971	A/swine/British Columbia/28103/2005(H3N2)
51	EU604691	A/swine/OH/511445/2007(H1N1)
52	DQ280213	A/swine/Ontario/53518/03(H1N1)
53	AB473548	A/duck/Mongolia/47/2001(H7N1)
54	AY676023	A/chicken/Korea/ES/03(H5N1)
55	DQ464357	A/swan/Germany/R65/2006(H5N1)
56	CY037965	A/swine/Belgium/WVL2/1983(H1N1)
57	CY038004	A/swine/England/WVL7/1992(H1N1)
58	EF101747	A/Philippines/344/2004(H1N2)
59	FJ415615	A/swine/Zhejiang/1/2007(H1N1)
60	CY009379	A/swine/Spain/33601/2001(H3N2)
61	CY010587	A/swine/Spain/53207/2004(H1N1)
62	CY009899	A/Swine/Spain/50047/2003(H1N1)
63	CY038020	A/swine/Denmark/WVL9/1993(H1N1)
64	AJ293920	A/Hong Kong/1774/99(H3N2)

65	AB434293	A/swine/Ratchaburi/NIAH550/2003(H1N1)
66	AB434285	A/swine/Ratchaburi/NIAH1481/2000(H1N1)
67	EF101754	A/Thailand/271/2005(H1N1)
68	AB434325	A/swine/Chachoengsao/NIAH587/2005(H1N1)

TABLE X: THE LIST OF PB1 GENES

No. on tree	Access ID.	Strain Description
1	M55472	A/swine/1976/1931(H1N1)
2	CY026145	A/Wisconsin/301/1976(H1N1)
3	CY024931	A/Ohio/3559/1988(H1N1)
4	CY027161	A/swine/Iowa/24297/1991(H1N1)
5	CY021803	A/Albany/20/1978(H1N1)
6	CY000456	A/New York/146/2000(H1N1)
7	CY033628	A/New Caledonia/20/1999(H1N1)
8	CY020163	A/Western Australia/77/2005(H1N1)
9	EF101748	A/Philippines/344/2004(H1N2)
10	CY038005	A/swine/England/WVL7/1992(H1N1)
11	CY038021	A/swine/Denmark/WVL9/1993(H1N1)
12	AJ293921	A/Hong Kong/1774/99(H3N2)
13	AB434326	A/swine/Chachoengsao/NIAH587/2005(H1N1)
14	AB434286	A/swine/Ratchaburi/NIAH1481/2000(H1N1)
15	AB434294	A/swine/Ratchaburi/NIAH550/2003(H1N1)
16	EF101753	A/Thailand/271/2005(H1N1)
17	CY010586	A/swine/Spain/53207/2004(H1N1)
18	CY009898	A/Swine/Spain/50047/2003(H1N1)
19	FJ415614	A/swine/Zhejiang/1/2007(H1N1)
20	EU026108	A/duck/NY/13152-13/1994(H1N1)

21	FJ432760	A/duck/Italy/69238/2007(H1N1)
22	CY025147	A/turkey/Italy/4617/1999(H7N1)
23	AB268552	A/duck/Mongolia/47/2001(H7N1)
24	DQ464361	A/swan/Germany/R65/2006(H5N1)
25	AY676027	A/chicken/Korea/ES/03(H5N1)
26	CY034780	A/duck/Vietnam/NCVD06/2005(H5N1)
27	AY619955	A/swine/Saskatchewan/18789/02(H1N1)
28	EU026020	A/mallard/MD/161/2002(H1N1)
29	CY032662	A/gadwall/California/HKWF100/2007(H6N1)
30	CY033418	A/American wigeon/California/HKWF42/2007(H6N1)
31	CY008162	A/Beijing/1/68(H3N2)
32	CY009362	A/England/72(H3N2)
33	CY003070	A/Memphis/1/90(H3N2)
34	CY008266	A/Canterbury/06/2002(H3N2)
35	CY034122	A/Wisconsin/67/2005(H3N2)
36	DQ280214	A/swine/Ontario/53518/03(H1N1)
37	DQ280206	A/swine/Ontario/55383/04(H1N2)
38	AY233388	A/duck/NC/91347/01(H1N2)
39	AF455725	A/Swine/North Carolina/93523/01 (H1N2)
40	AF251413	A/Swine/Iowa/533/99 (H3N2)
41	AF251429	A/Swine/Minnesota/593/99 (H3N2)
42	AF455729	A/Swine/Illinois/100085A/01 (H1N2)

43	AF455728	A/Swine/Indiana/P12439/00 (H1N2)
44	DQ469988	A/swine/Ontario/33853/2005(H3N2)
45	EU409945	A/swine/Ohio/24366/07(H1N1)
46	EU604692	A/swine/OH/511445/2007(H1N1)
47	DQ889683	A/Iowa/CEID23/2005(H1N1)
48	AF342823	A/Wisconsin/10/98 (H1N1)
49	EU015992	A/swine/Guangxi/13/2006(H1N2)
50	GQ117034	A/California/14/2009(H1N1)
51	CY039907	A/New York/1682/2009(H1N1)
52	FJ998226	A/Mexico/InDRE4487/2009(H1N1)
53	GQ132179	A/Mexico/InDRE4114/2009(H1N1)
54	FJ966965	A/California/06/2009(H1N1)
55	GQ117075	A/Arizona/02/2009(H1N1)
56	GQ168881	A/New York/13/2009(H1N1)
57	GQ160546	A/Texas/23/2009(H1N1)
58	FJ966080	A/California/04/2009(H1N1)
59	GQ168852	A/New York/11/2009(H1N1)
60	FJ984367	A/California/08/2009(H1N1)
61	GQ149683	A/Mexico/4603/2009(H1N1)
62	GQ168850	A/New York/06/2009(H1N1)
63	GQ168844	A/New York/12/2009(H1N1)
64	GQ117120	A/New York/09/2009(H1N1)

65	GQ160596	A/Nebraska/03/2009(H1N1)
66	GQ168877	A/New York/15/2009(H1N1)
67	GQ149675	A/Mexico/4482/2009(H1N1)
68	GQ117088	A/Texas/07/2009(H1N1)
69	GQ168880	A/Michigan/02/2009(H1N1)
70	GQ117069	A/Minnesota/02/2009(H1N1)
71	GQ117093	A/Indiana/09/2009(H1N1)
72	GQ117020	A/New York/22/2009(H1N1)
73	GQ149633	A/Mexico/4604/2009(H1N1)
74	GQ162176	A/Mexico/4486/2009(H1N1)
75	GQ168847	A/Massachusetts/06/2009(H1N1)
76	GQ168873	A/Massachusetts/07/2009(H1N1)
77	GQ168864	A/New York/23/2009(H1N1)
78	FJ984392	A/New York/19/2009(H1N1)
79	FJ966978	A/California/07/2009(H1N1)
80	GQ168867	A/New York/20/2009(H1N1)
81	FJ984373	A/New York/10/2009(H1N1)
82	FJ984353	A/New York/18/2009(H1N1)
83	GQ122094	A/Texas/15/2009(H1N1)
84	FJ966958	A/California/05/2009(H1N1)
85	GQ168875	A/Nebraska/02/2009(H1N1)
96	GQ168884	A/Colorado/03/2009(H1N1)

87	GQ168887	A/Ohio/07/2009(H1N1)
88	GQ162199	A/Mexico/4115/2009(H1N1)
89	GQ168854	A/South Carolina/09/2009(H1N1)
90	GQ117026	A/Texas/09/2009(H1N1)
91	GQ160570	A/Texas/22/2009(H1N1)
92	FJ969526	A/Texas/04/2009(H1N1)
93	GQ168860	A/Texas/05/2009(H1N1)
94	GQ117046	A/Texas/08/2009(H1N1)
95	AF250130	A/Swine/Indiana/9K035/99 (H1N2)
96	EU301400	A/swine/Korea/JNS06/2004(H3N2)
97	EU798909	A/swine/Korea/CAS05/2004(H3N2)

TABLE XI: THE LIST OF PA GENES

No. on tree	Access ID.	Strain Description
1	M26076	A/swine/Iowa/15/1930(H1N1)
2	CY009633	A/swine/1931(H1N1)
3	CY026144	A/Wisconsin/301/1976(H1N1)
4	CY024930	A/Ohio/3559/1988(H1N1)
5	CY027160	A/swine/Iowa/24297/1991(H1N1)
6	DQ280215	A/swine/Ontario/53518/03(H1N1)
7	DQ280207	A/swine/Ontario/55383/04(H1N2)
8	DQ280191	A/swine/Ontario/57561/03(H1N1)
9	CY034121	A/Wisconsin/67/2005(H3N2)
10	AB268553	A/duck/Mongolia/47/2001(H7N1)
11	CY025146	A/turkey/Italy/4617/1999(H7N1)
12	FJ432759	A/duck/Italy/69238/2007(H1N1)
13	AY676031	A/chicken/Korea/ES/03(H5N1)
14	EU026107	A/duck/NY/13152-13/1994(H1N1)
15	EU880825	A/turkey/CA/358533/2005(H4N8)
16	AF285890	A/Swine/Ontario/01911-1/99 (H4N6)
17	CY032663	A/gadwall/California/HKWF100/2007(H6N1)
18	CY033419	A/American wigeon/California/HKWF42/2007(H6N1)
19	AY619956	A/swine/Saskatchewan/18789/02(H1N1)
20	AY619972	A/swine/Ontario/42729A/01(H3N3)

21	EU026019	A/mallard/MD/161/2002(H1N1)
22	EU798897	A/swine/Korea/CY10/2007(H3N2)
23	EU798889	A/swine/Korea/CAS05/2004(H3N2)
24	AY233389	A/duck/NC/91347/01(H1N2)
25	AF455717	A/Swine/North Carolina/93523/01 (H1N2)
26	EU301368	A/swine/Korea/JNS06/2004(H3N2)
27	AF455720	A/Swine/Indiana/P12439/00 (H1N2)
28	AD250129	A/Swine/Indiana/9K035/99 (H1N2)
29	DQ889684	A/Iowa/CEID23/2005(H1N1)
30	DQ469989	A/swine/Ontario/33853/2005(H3N2)
31	EU409947	A/swine/Ohio/24366/07(H1N1)
32	EU604693	A/swine/OH/511445/2007(H1N1)
33	AF251417	A/Swine/Iowa/533/99 (H3N2)
34	AF251433	A/Swine/Minnesota/593/99 (H3N2)
35	AF342822	A/Wisconsin/10/98 (H1N1)
36	EU015991	A/swine/Guangxi/13/2006(H1N2)
37	GQ149627	A/Mexico/4603/2009(H1N1)
38	GQ149636	A/Mexico/4604/2009(H1N1)
39	FJ984393	A/New York/19/2009(H1N1)
40	FJ984346	A/New York/11/2009(H1N1)
41	GQ168865	A/New York/23/2009(H1N1)
42	GQ149676	A/Mexico/4482/2009(H1N1)

43	FJ984380	A/New York/15/2009(H1N1)
44	FJ984354	A/New York/18/2009(H1N1)
45	GQ117037	A/California/14/2009(H1N1)
46	FJ984374	A/New York/10/2009(H1N1)
47	FJ984359	A/New York/31/2009(H1N1)
48	GQ117109	A/Michigan/02/2009(H1N1)
49	GQ117115	A/New York/13/2009(H1N1)
50	FJ981619	A/Texas/04/2009(H1N1)
51	FJ966970	A/Texas/05/2009(H1N1)
52	GQ117049	A/Texas/08/2009(H1N1)
53	GQ117029	A/Texas/09/2009(H1N1)
54	GQ132176	A/Mexico/InDRE4114/2009(H1N1)
55	FJ998223	A/Mexico/InDRE4487/2009(H1N1)
56	GQ149653	A/Mexico/4108/2009(H1N1)
57	GQ117095	A/Indiana/09/2009(H1N1)
58	GQ149620	A/Mexico/4486/2009(H1N1)
59	CY039906	A/New York/1682/2009(H1N1)
60	FJ966957	A/California/05/2009(H1N1)
61	FJ966964	A/California/06/2009(H1N1)
62	FJ966081	A/California/04/2009(H1N1)
63	FJ966977	A/California/07/2009(H1N1)
64	FJ984368	A/California/08/2009(H1N1)

65	EF101755	A/Thailand/271/2005(H1N1)
66	AB434327	A/swine/Chachoengsao/NIAH587/2005(H1N1)
67	CY038006	A/swine/England/WVL7/1992(H1N1)
68	EF101746	A/Philippines/344/2004(H1N2)
69	AB434295	A/swine/Ratchaburi/NIAH550/2003(H1N1)
70	CY009897	A/Swine/Spain/50047/2003(H1N1)
71	CY009377	A/swine/Spain/33601/2001(H3N2)
72	AJ293922	A/Hong Kong/1774/99(H3N2)
73	CY010585	A/swine/Spain/53207/2004(H1N1)
74	FJ415616	A/swine/Zhejiang/1/2007(H1N1)

TABLE XII: THE LIST OF HA GENES

No. on tree	Access ID.	Strain Description
1	CY026139	A/Wisconsin/301/1976(H1N1)
2	AB434328	A/swine/Chachoengsao/NIAH587/2005(H1N1)
3	EU296603	A/swine/Chonburi/05CB1/2005(H1N1)
4	EF101749	A/Thailand/271/2005(H1N1)
5	AB434288	A/swine/Ratchaburi/NIAH1481/2000(H1N1)
6	AB434296	A/swine/Ratchaburi/NIAH550/2003(H1N1)
7	CY024925	A/Ohio/3559/1988(H1N1)
8	EF101741	A/Philippines/344/2004(H1N2)
9	CY027155	A/swine/Iowa/24297/1991(H1N1)
10	AF222026	A/Swine/Wisconsin/125/97(H1N1)
11	AY060046	A/SW/CO/17871/01(H1N2)
12	AY129156	A/Swine/Korea/CY02/02(H1N2)
13	AY060049	A/SW/MO/1877/01(H1N2)
14	EU798779	A/swine/Korea/CAS08/2005(H1N1)
15	DQ889689	A/Iowa/CEID23/2005(H1N1)
16	EU139832	A/swine/Iowa/00239/2004(H1N1)
17	FJ986618	A/Iowa/01/2006(H1N1)
18	AY233393	A/duck/NC/91347/01(H1N2)
19	AF455676	A/Swine/North Carolina/98225/01(H1N2)
20	AF455677	A/Swine/North Carolina/93523/01 (H1N2)

21	AF342821	A/Wisconsin/10/98 (H1N1)
22	AF250124	A/Swine/Indiana/9K035/99 (H1N2)
23	AF455682	A/Swine/Illinois/100084/01 (H1N2)
24	AF455675	A/Swine/Ohio/891/01(H1N2)
25	EU139830	A/swine/Minnesota/00194/2003(H1N2)
26	FJ986619	A/Wisconsin/87/2005(H1N1)
27	FJ986620	A/Ohio/01/2007(H1N1)
28	EU604689	A/swine/OH/511445/2007(H1N1)
29	EU139831	A/swine/Kansas/00246/2004(H1N2)
30	EU798787	A/swine/Korea/PZ14/2006(H1N2)
31	EU798784	A/swine/Korea/Asan04/2006(H1N2)
32	AF455680	A/Swine/Indiana/P12439/00 (H1N2)
33	AY038014	A/Turkey/MO/24093/99(H1N2)
34	EF556201	A/swine/Guangxi/17/2005(H1N2)
35	EF556199	A/swine/Guangxi/13/2006(H1N2)
36	GQ117119	A/Colorado/03/2009(H1N1)
37	GQ160607	A/California/13/2009(H1N1)
38	GQ221818	A/California/11/2009(H1N1)
39	FJ969511	A/California/10/2009(H1N1)
40	FJ966952	A/California/05/2009(H1N1)
41	FJ966960	A/California/06/2009(H1N1)
42	GQ160606	A/California/12/2009(H1N1)

43	GQ162172	A/Mexico/4269/2009(H1N1)
44	GQ149692	A/Mexico/4115/2009(H1N1)
45	GQ162174	A/Mexico/4593/2009(H1N1)
46	GQ162182	A/Mexico/4176/2009(H1N1)
47	GQ221788	A/Arizona/02/2009(H1N1)
48	GQ160550	A/Texas/23/2009(H1N1)
49	GQ160605	A/Wisconsin/08/2009(H1N1)
50	GQ162200	A/Mexico/4635/2009(H1N1)
51	GQ149630	A/Mexico/4603/2009(H1N1)
52	FJ984397	A/Ohio/07/2009(H1N1)
53	GQ117043	A/Massachusetts/06/2009(H1N1)
54	GQ160599	A/Nebraska/03/2009(H1N1)
55	GQ168851	A/New York/06/2009(H1N1)
56	GQ160542	A/New York/05/2009(H1N1)
57	GQ160527	A/New York/37/2009(H1N1)
58	GQ221786	A/New York/19/2009(H1N1)
59	GQ117024	A/New York/22/2009(H1N1)
60	GQ117040	A/California/14/2009(H1N1)
61	GQ117103	A/Massachusetts/07/2009(H1N1)
62	GQ168661	A/New York/31/2009(H1N1)
63	FJ984364	A/New York/23/2009(H1N1)
64	GQ117082	A/New York/20/2009(H1N1)

65	GQ168655	A/New York/09/2009(H1N1)
66	FJ984347	A/New York/11/2009(H1N1)
67	GQ117116	A/New York/13/2009(H1N1)
68	GQ160601	A/New York/24/2009(H1N1)
69	FJ984375	A/New York/10/2009(H1N1)
70	GQ160566	A/New York/30/2009(H1N1)
71	FJ984337	A/New York/12/2009(H1N1)
72	FJ984355	A/New York/18/2009(H1N1)
73	GQ160602	A/New York/39/2009(H1N1)
74	GQ160568	A/New York/04/2009(H1N1)
75	GQ160543	A/New York/35/2009(H1N1)
76	GQ160567	A/New York/34/2009(H1N1)
77	GQ117112	A/Michigan/02/2009(H1N1)
78	GQ168620	A/Texas/19/2009(H1N1)
79	GQ149671	A/Mexico/4482/2009(H1N1)
80	GQ162195	A/Mexico/4502/2009(H1N1)
81	GQ162191	A/Mexico/4595/2009(H1N1)
82	GQ162197	A/Mexico/4575/2009(H1N1)
83	GQ149634	A/Mexico/4604/2009(H1N1)
84	GQ149647	A/Mexico/4486/2009(H1N1)
85	GQ162204	A/Mexico/4646/2009(H1N1)
86	FJ998208	A/Mexico/InDRE4487/2009(H1N1)

87	GQ132145	A/Mexico/InDRE4114/2009(H1N1)
88	CY039901	A/New York/1682/2009(H1N1)
89	GQ162183	A/Mexico/4627/2009(H1N1)
90	GQ221798	A/Texas/04/2009(H1N1)
91	FJ984385	A/Texas/06/2009(H1N1)
92	GQ117051	A/Texas/08/2009(H1N1)
93	GQ168671	A/Texas/09/2009(H1N1)
94	GQ168861	A/Texas/05/2009(H1N1)
95	GQ160574	A/Texas/22/2009(H1N1)
96	GQ162170	A/Mexico/4108/2009(H1N1)
97	GQ162194	A/Mexico/3955/2009(H1N1)
98	GQ117097	A/Indiana/09/2009(H1N1)
99	GQ117091	A/Texas/07/2009(H1N1)
100	GQ117067	A/Arizona/01/2009(H1N1)
101	GQ122097	A/Texas/15/2009(H1N1)
102	FJ966974	A/California/07/2009(H1N1)
103	FJ971076	A/California/08/2009(H1N1)
104	GQ117044	A/California/04/2009(H1N1)
105	AF389118	A/Puerto Rico/8/34/Mount Sinai(H1N1)
106	AJ344014	A/New Caledonia/20/99/(H1N1)
107	EU516297	A/Florida/3/2006(H1N1)
108	CY030230	A/Brisbane/59/2007(H1N1)

109	FJ686964	A/Washington/10/2008(H1N1)
110	FJ432754	A/duck/Italy/69238/2007(H1N1)
111	EU026102	A/duck/NY/13152-13/1994(H1N1)
112	AY619961	A/swine/Saskatchewan/18789/02(H1N1)
113	EU260014	A/mallard/MD/161/2002(H1N1)
114	CY038007	A/swine/England/WVL7/1992(H1N1)
115	CY037936	A/swine/England/WVL14/1996(H1N1)
116	CY038023	A/swine/Denmark/WVL9/1993(H1N1)
117	CY010580	A/swine/Spain/53207/2004(H1N1)
118	CY009892	A/Swine/Spain/50047/2003(H1N1)

TABLE XIII: THE LIST OF NP GENES

No. on tree	Access ID.	Strain Description
1	CY024928	A/Ohio/3559/1988(H1N1)
2	CY027158	A/swine/Iowa/24297/1991(H1N1)
3	AF251415	A/Swine/Iowa/533/99 (H3N2)
4	AF251431	A/Swine/Minnesota/593/99 (H3N2)
5	AF153254	A/Swine/Minnesota/9088-2/98 (H3N2)
6	AY233394	A/duck/NC/91347/01(H1N2)
7	EU301304	A/swine/Korea/JNS06/2004(H3N2)
8	EU798857	A/swine/Korea/CY10/2007(H3N2)
9	AF455704	A/Swine/Indiana/P12439/00 (H1N2)
10	AF455701	A/Swine/North Carolina/93523/01 (H1N2)
11	FJ374515	A/swine/Shanghai/1/2007(H1N2)
12	EU850621	A/swine/Guangxi/17/2005(H1N2)
13	EU015990	A/swine/Guangxi/13/2006(H1N2)
14	AF342819	A/Wisconsin/10/98 (H1N1)
15	DQ469991	A/swine/Ontario/33853/2005(H3N2)
16	DQ889686	A/Iowa/CEID23/2005(H1N1)
17	EU258936	A/swine/Missouri/2124514/2006(H2N3)
18	EU604694	A/swine/OH/511445/2007(H1N1)
19	FJ984384	A/Texas/06/2009(H1N1)
20	GQ122092	A/Texas/15/2009(H1N1)

21	GQ117063	A/Arizona/01/2009(H1N1)
22	GQ149693	A/Mexico/4115/2009(H1N1)
23	GQ160569	A/Texas/22/2009(H1N1)
24	FJ984366	A/California/08/2009(H1N1)
25	GQ149637	A/Mexico/4604/2009(H1N1)
26	GQ117107	A/Michigan/02/2009(H1N1)
27	GQ168618	A/Texas/19/2009(H1N1)
28	GQ149678	A/Mexico/4482/2009(H1N1)
29	GQ160603	A/Wisconsin/08/2009(H1N1)
30	GQ149682	A/Mexico/4603/2009(H1N1)
31	GQ149648	A/Mexico/4516/2009(H1N1)
32	FJ984363	A/New York/23/2009(H1N1)
33	FJ984345	A/New York/11/2009(H1N1)
34	GQ117101	A/Massachusetts/07/2009(H1N1)
35	FJ984391	A/New York/19/2009(H1N1)
36	GQ117113	A/New York/13/2009(H1N1)
37	GQ117041	A/Massachusetts/06/2009(H1N1)
38	GQ117083	A/New York/20/2009(H1N1)
39	FJ984352	A/New York/18/2009(H1N1)
40	FJ984358	A/New York/31/2009(H1N1)
41	FJ984341	A/New York/06/2009(H1N1)
42	FJ984336	A/New York/12/2009(H1N1)

43	GQ117033	A/California/14/2009(H1N1)
44	FJ984372	A/New York/10/2009(H1N1)
45	GQ117019	A/New York/22/2009(H1N1)
46	GQ160595	A/Nebraska/03/2009(H1N1)
47	FJ984379	A/New York/15/2009(H1N1)
48	GQ117098	A/Ohio/07/2009(H1N1)
49	FJ969512	A/California/04/2009(H1N1)
50	GQ117045	A/Texas/08/2009(H1N1)
51	FJ981618	A/Texas/04/2009(H1N1)
52	FJ981609	A/Texas/05/2009(H1N1)
53	GQ117025	A/Texas/09/2009(H1N1)
54	GQ132161	A/Mexico/InDRE4114/2009(H1N1)
55	FJ998217	A/Mexico/InDRE4487/2009(H1N1)
56	CY039904	A/New York/1682/2009(H1N1)
57	GQ117092	A/Indiana/09/2009(H1N1)
58	GQ149685	A/Mexico/4108/2009(H1N1)
59	FJ966961	A/California/06/2009(H1N1)
60	GQ117074	A/Arizona/02/2009(H1N1)
61	GQ117087	A/Texas/07/2009(H1N1)
62	GQ117068	A/Minnesota/02/2009(H1N1)
63	GQ149645	A/Mexico/4486/2009(H1N1)
64	FJ966953	A/California/05/2009(H1N1)

65	GQ117104	A/Nebraska/02/2009(H1N1)
66	GQ117117	A/Colorado/03/2009(H1N1)
67	DQ280193	A/swine/Ontario/57561/03(H1N1)
68	DQ280209	A/swine/Ontario/55383/04(H1N2)
69	CY034119	A/Wisconsin/67/2005(H3N2)
70	AF156415	A/Turkey/California/189/66(H9N2)
71	EU743125	A/turkey/CA/6878/1979(H5N3)
72	AY619958	A/swine/Saskatchewan/18789/02(H1N1)
73	AY619974	A/swine/Ontario/42729A/01(H3N3)
74	EU880823	A/turkey/CA/358533/2005(H4N8)
75	CY033421	A/American wigeon/California/HKWF42/2007(H6N1)
76	EU742873	A/turkey/CA/8651-C/2002(H5N2)
77	EU026105	A/duck/NY/13152-13/1994(H1N1)
78	EU026017	A/mallard/MD/161/2002(H1N1)
79	AB268554	A/duck/Mongolia/47/2001(H7N1)
80	FJ432757	A/duck/Italy/69238/2007(H1N1)
81	DQ321111	A/duck/Shantou/4610/2003(H5N1)
82	AY676039	A/chicken/Korea/ES/03(H5N1)
83	DQ464359	A/swan/Germany/R65/2006(H5N1)
84	CY034777	A/duck/Vietnam/NCVD06/2005(H5N1)
85	CY024893	A/turkey/Italy/3488/1999(H7N1)
86	CY021544	A/turkey/Italy/4426/2000(H7N1)

87	AJ627488	A/turkey/Italy/4603/1999(H7N1)
88	CY037968	A/swine/Belgium/WVL2/1983(H1N1)
89	CY038008	A/swine/England/WVL7/1992(H1N1)
90	CY037937	A/swine/England/WVL14/1996(H1N1)
91	CY037945	A/swine/England/WVL15/1997(H1N1)
92	AB434297	A/swine/Ratchaburi/NIAH550/2003(H1N1)
93	AB434289	A/swine/Ratchaburi/NIAH1481/2000(H1N1)
94	EF101752	A/Thailand/271/2005(H1N1)
95	AB434329	A/swine/Chachoengsao/NIAH587/2005(H1N1)
96	CY038024	A/swine/Denmark/WVL9/1993(H1N1)
97	AJ293924	A/Hong Kong/1774/99(H3N2)
98	EU924272	A/swine/Nordkirchen/IDT1993/2003(H3N2)
99	FJ805965	A/swine/Belgium/1/1998(H1N1)
100	CY010567	A/swine/Spain/54008/2004(H3N2)
101	FJ415617	A/swine/Zhejiang/1/2007(H1N1)
102	CY009375	A/Swine/Spain/50047/2003(H1N1)
103	CY009895	A/Swine/Spain/50047/2003(H1N1)
104	CY010583	A/swine/Spain/53207/2004(H1N1)
105	FJ798781	A/swine/Hungary/19774/2006(H1N1)

TABLE XIV: THE LIST OF NA GENES

No. on tree	Access ID.	Strain Description
1	EU139833	A/swine/Iowa/15/1930(H1N1)
2	CY026141	A/Wisconsin/301/1976(H1N1)
3	CY024927	A/Ohio/3559/1988(H1N1)
4	CY027157	A/swine/Iowa/24297/1991(H1N1)
5	EU851998	A/Ohio/01/2007(H1N1)
6	EU604690	A/swine/OH/511445/2007(H1N1)
7	AF342820	A/Wisconsin/10/98 (H1N1)
8	EU139839	A/swine/North Carolina/36883/2002(H1N1)
9	EU798819	A/swine/Korea/CAS08/2005(H1N1)
10	DQ889687	A/Iowa/CEID23/2005(H1N1)
11	DQ145538	A/swine/Minnesota/00395/2004(H3N1)
12	EU139842	A/swine/Iowa/00239/2004(H1N1)
13	EU798818	A/swine/Korea/CAN01/2004(H1N1)
14	CY033624	A/New Caledonia/20/1999(H1N1)
15	EU100633	A/Florida/3/2006(H1N1)
16	EU124136	A/Solomon Islands/3/2006(H1N1)
17	CY030233	A/Brisbane/59/2007(H1N1)
18	FJ686963	A/Washington/10/2008(H1N1)
19	EU026104	A/duck/NY/13152-13/1994(H1N1)
20	EU026016	A/mallard/MD/161/2002(H1N1)

21	AY619960	A/swine/Saskatchewan/18789/02(H1N1)
22	CY032714	A/northern shoveler/California/HKWF383/2007(H6N1)
23	CY032666	A/gadwall/California/HKWF100/2007(H6N1)
24	CY033422	A/American wigeon/California/HKWF42/2007(H6N1)
25	CY035851	A/northern pintail/California/HKWF151/2007(H6N1)
26	CY032926	A/ring-necked duck/California/HKWF662/2007(H6N1)
27	EU296604	A/swine/Chonburi/05CB1/2005(H1N1)
28	CY038025	A/swine/Denmark/WVL9/1993(H1N1)
29	EU045389	A/swine/Italy/65296/2004(H1N1)
30	EU045388	A/swine/Italy/53949/2004(H1N1)
31	CY010582	A/swine/Spain/53207/2004(H1N1)
32	FJ969517	A/California/04/2009(H1N1)
33	FJ966973	A/California/09/2009(H1N1)
34	GQ122096	A/Texas/15/2009(H1N1)
35	GQ149691	A/Mexico/4115/2009(H1N1)
36	GQ162193	A/Mexico/3955/2009(H1N1)
37	FJ998214	A/Mexico/InDRE4487/2009(H1N1)
38	GQ132155	A/Mexico/InDRE4114/2009(H1N1)
39	GQ149631	A/Mexico/4604/2009(H1N1)
40	GQ117022	A/New York/22/2009(H1N1)
41	GQ168666	A/New York/13/2009(H1N1)
42	FJ984350	A/New York/18/2009(H1N1)

43	GQ117102	A/Massachusetts/07/2009(H1N1)
44	GQ117036	A/California/14/2009(H1N1)
45	FJ984340	A/New York/06/2009(H1N1)
46	GQ117102	A/Massachusetts/07/2009(H1N1)
47	FJ984390	A/New York/19/2009(H1N1)
48	FJ984378	A/New York/15/2009(H1N1)
49	GQ117084	A/New York/20/2009(H1N1)
50	GQ162196	A/Mexico/4575/2009(H1N1)
51	GQ160547	A/Texas/23/2009(H1N1)
52	FJ984371	A/New York/10/2009(H1N1)
53	GQ168660	A/New York/31/2009(H1N1)
54	GQ117108	A/Michigan/02/2009(H1N1)
55	GQ168651	A/New York/11/2009(H1N1)
56	GQ168627	A/New York/23/2009(H1N1)
57	GQ162173	A/Mexico/4593/2009(H1N1)
58	GQ168672	A/New York/12/2009(H1N1)
59	FJ981614	A/Texas/04/2009(H1N1)
60	CY039903	A/New York/1682/2009(H1N1)
61	GQ117077	A/Arizona/02/2009(H1N1)
62	GQ162201	A/Mexico/4486/2009(H1N1)
63	GQ168632	A/Texas/08/2009(H1N1)
64	GQ117064	A/Arizona/01/2009(H1N1)

65	GQ162169	A/Mexico/4108/2009(H1N1)
66	FJ966969	A/Texas/05/2009(H1N1)
67	FJ984383	A/Texas/06/2009(H1N1)
68	GQ117071	A/Minnesota/02/2009(H1N1)
69	GQ162171	A/Mexico/4269/2009(H1N1)
70	GQ162181	A/Mexico/4176/2009(H1N1)
71	GQ117094	A/Indiana/09/2009(H1N1)
72	FJ971075	A/California/06/2009(H1N1)
73	GQ117118	A/Colorado/03/2009(H1N1)
74	FJ966956	A/California/05/2009(H1N1)
75	GQ117105	A/Nebraska/02/2009(H1N1)
76	GQ160571	A/Texas/22/2009(H1N1)
77	GQ168670	A/Texas/09/2009(H1N1)
78	AB302788	A/duck/Mongolia/47/2001(H7N1)
79	AB470667	A/duck/Mongolia/116/2002(H1N1)
80	FJ432756	A/duck/Italy/69238/2007(H1N1)
81	DQ464355	A/swan/Germany/R65/2006(H5N1)
82	EU233677	A/chicken/Korea/IS/2006(H5N1)

TABLE XV: THE LIST OF MP GENES

No. on tree	Access ID.	Strain Description
1	CY026140	A/Wisconsin/301/1976(H1N1)
2	DQ280196	A/swine/Ontario/57561/03(H1N1)
3	CY027156	A/swine/Iowa/24297/1991(H1N1)
4	CY024926	A/Ohio/3559/1988(H1N1)
5	AF455688	A/Swine/Indiana/P12439/00 (H1N2)
6	AF251414	A/Swine/Iowa/533/99 (H3N2)
7	AF153258	A/Swine/Minnesota/9088-2/98 (H3N2)
8	AF342818	A/Wisconsin/10/98 (H1N1)
9	EU015989	A/swine/Guangxi/13/2006(H1N2)
10	EU850622	A/swine/Guangxi/17/2005(H1N2)
11	AY233392	A/duck/NC/91347/01(H1N2)
12	AF455685	A/Swine/North Carolina/93523/01 (H1N2)
13	EU301241	A/swine/Korea/JNS06/2004(H3N2)
14	DQ469993	A/swine/Ontario/33853/2005(H3N2)
15	EU604695	A/swine/OH/511445/2007(H1N1)
16	DQ889688	A/Iowa/CEID23/2005(H1N1)
17	DQ280210	A/swine/Ontario/55383/04(H1N2)
18	EF190977	A/hvPR8/34(H1N1)
19	DQ849011	A/Wyoming/3/2003(H3N2)
20	CY034117	A/Wisconsin/67/2005(H3N2)

21	CY039088	A/Brisbane/10/2007(H3N2)
22	CY033623	A/New Caledonia/20/1999(H1N1)
23	CY031391	A/Brisbane/59/2007(H1N1)
24	EU516163	A/California/27/2007(H1N1)
25	EU026015	A/mallard/MD/161/2002(H1N1)
26	CY033423	A/American wigeon/California/HKWF42/2007(H6N1)
27	EU880821	A/turkey/CA/358533/2005(H4N8)
28	CY032723	A/northern pintail/California/HKWF792/2007(H3N8)
29	AY619976	A/swine/Ontario/42729A/01(H3N3)
30	AB268555	A/duck/Mongolia/47/2001(H7N1)
31	CY024891	A/turkey/Italy/3488/1999(H7N1)
32	CY025142	A/turkey/Italy/4617/1999(H7N1)
33	FJ432755	A/duck/Italy/69238/2007(H1N1)
34	CY034775	A/duck/Vietnam/NCVD06/2005(H5N1)
35	EF101742	A/Philippines/344/2004(H1N2)
36	AB434299	A/swine/Ratchaburi/NIAH550/2003(H1N1)
37	AB434291	A/swine/Ratchaburi/NIAH1481/2000(H1N1)
38	CY038026	A/swine/Denmark/WVL9/1993(H1N1)
39	EU478802	A/swine/England/17394/96(H1N2)
40	CY037939	A/swine/England/WVL14/1996(H1N1)
41	CY038010	A/swine/England/WVL7/1992(H1N1)
42	AJ293925	A/Hong Kong/1774/99(H3N2)

43	GQ117090	A/Texas/07/2009(H1N1)
44	GQ122095	A/Texas/15/2009(H1N1)
45	GQ117073	A/Minnesota/02/2009(H1N1)
46	FJ969510	A/California/10/2009(H1N1)
47	FJ966954	A/California/05/2009(H1N1)
48	FJ969537	A/California/07/2009(H1N1)
49	FJ969532	A/California/08/2009(H1N1)
50	GQ162188	A/Mexico/4486/2009(H1N1)
51	FJ966962	A/California/06/2009(H1N1)
52	GQ117066	A/Arizona/01/2009(H1N1)
53	GQ117096	A/Indiana/09/2009(H1N1)
54	FJ998211	A/Mexico/InDRE4487/2009(H1N1)
55	CY039902	A/New York/1682/2009(H1N1)
56	GQ168619	A/Texas/19/2009(H1N1)
57	GQ162175	A/Mexico/4603/2009(H1N1)
58	GQ117078	A/Arizona/02/2009(H1N1)
59	FJ969513	A/California/04/2009(H1N1)
60	GQ149625	A/Mexico/4115/2009(H1N1)
61	GQ132150	A/Mexico/InDRE4114/2009(H1N1)
62	GQ162203	A/Mexico/4646/2009(H1N1)
63	GQ160549	A/Texas/23/2009(H1N1)
64	FJ966972	A/California/09/2009(H1N1)

65	GQ149638	A/Mexico/4604/2009(H1N1)
66	GQ162179	A/Mexico/4108/2009(H1N1)
67	GQ162178	A/Mexico/4516/2009(H1N1)
68	GQ117050	A/Texas/08/2009(H1N1)
69	GQ117031	A/Texas/09/2009(H1N1)
70	GQ160573	A/Texas/22/2009(H1N1)
71	GQ162192	A/Mexico/4482/2009(H1N1)
72	FJ981617	A/Texas/04/2009(H1N1)
73	FJ981608	A/Texas/05/2009(H1N1)
74	FJ984381	A/Texas/06/2009(H1N1)
75	GQ117111	A/Michigan/02/2009(H1N1)
76	GQ168849	A/Massachusetts/06/2009(H1N1)
77	FJ984369	A/New York/10/2009(H1N1)
78	GQ160600	A/New York/24/2009(H1N1)
79	FJ984376	A/New York/15/2009(H1N1)
80	GQ168863	A/New York/20/2009(H1N1)
81	FJ984398	A/Ohio/07/2009(H1N1)
82	FJ984348	A/New York/18/2009(H1N1)
83	FJ984388	A/New York/19/2009(H1N1)
84	GQ117023	A/New York/22/2009(H1N1)
85	GQ117039	A/California/14/2009(H1N1)
86	FJ984342	A/New York/11/2009(H1N1)

87	GQ168845	A/New York/12/2009(H1N1)
88	GQ168859	A/New York/31/2009(H1N1)
89	FJ984338	A/New York/06/2009(H1N1)
90	GQ168883	A/New York/13/2009(H1N1)
91	GQ168866	A/New York/23/2009(H1N1)
92	GQ168863	A/New York/20/2009(H1N1)
93	EU478840	A/swine/Muesleringen-S./IDT4263/05(H3N2)
94	EU478835	A/swine/Hertzen/IDT4317/05(H3N2)
95	AB434331	A/swine/Chachoengsao/NIAH587/2005(H1N1)
96	EF101750	A/Thailand/271/2005(H1N1)
97	FJ805964	A/swine/Belgium/1/1998(H1N1)
98	FJ415612	A/swine/Zhejiang/1/2007(H1N1)
99	CY009373	A/swine/Spain/33601/2001(H3N2)
100	CY009893	A/swine/Spain/50047/2003(H1N1)
101	EU478816	A/swine/Norden/IDT2308/03(H1N2)
102	CY010581	A/swine/Spain/53207/2004(H1N1)
103	FJ798779	A/swine/Hungary/19774/2006(H1N1)

TABLE XVI: THE LIST OF NS GENES

No. on tree	Access ID.	Strain Description
1	EF101751	A/Thailand/271/2005(H1N1)
2	CY026143	A/Wisconsin/301/1976(H1N1)
3	CY024929	A/Ohio/3559/1988(H1N1)
4	DQ280192	A/swine/Ontario/57561/03(H1N1)
5	AF251416	A/Swine/Iowa/533/99 (H3N2)
6	EU301336	A/swine/Korea/JNS06/2004(H3N2)
7	EU735822	A/turkey/OH/313053/2004(H3N2)
8	DQ469990	A/swine/Ontario/33853/2005(H3N2)
9	DQ469974	A/swine/British Columbia/28103/2005(H3N2)
10	DQ889685	A/Iowa/CEID23/2005(H1N1)
11	EU798877	A/swine/Korea/CY10/2007(H3N2)
12	AF342817	A/Wisconsin/10/98 (H1N1)
13	AF153262	A/Swine/Minnesota/9088-2/98 (H3N2)
14	AF455712	A/Swine/Indiana/P12439/00 (H1N2)
15	FJ374516	A/swine/Shanghai/1/2007(H1N2)
16	EU850623	A/swine/Guangxi/17/2005(H1N2)
17	EU015988	A/swine/Guangxi/13/2006(H1N2)
18	AF455709	A/Swine/North Carolina/93523/01 (H1N2)
19	AY233390	A/duck/NC/91347/01(H1N2)
20	FJ969514	A/California/04/2009(H1N1)

21	GQ117072	A/Minnesota/02/2009(H1N1)
22	GQ149687	A/Mexico/4108/2009(H1N1)
23	GQ117065	A/Arizona/01/2009(H1N1)
24	GQ168853	A/Texas/08/2009(H1N1)
25	FJ966966	A/Texas/05/2009(H1N1)
26	FJ984382	A/Texas/06/2009(H1N1)
27	GQ117030	A/Texas/09/2009(H1N1)
28	GQ160572	A/Texas/22/2009(H1N1)
29	GQ149694	A/Mexico/4115/2009(H1N1)
30	GQ122093	A/Texas/15/2009(H1N1)
31	GQ149680	A/Mexico/4603/2009(H1N1)
32	FJ969533	A/California/08/2009(H1N1)
33	GQ160604	A/Wisconsin/08/2009(H1N1)
34	GQ168871	A/Indiana/09/2009(H1N1)
35	FJ971074	A/California/06/2009(H1N1)
36	GQ117106	A/Nebraska/02/2009(H1N1)
37	GQ160548	A/Texas/23/2009(H1N1)
38	GQ149635	A/Mexico/4604/2009(H1N1)
39	GQ168869	A/Texas/07/2009(H1N1)
40	GQ132167	A/Mexico/InDRE4114/2009(H1N1)
41	FJ998220	A/Mexico/InDRE4487/2009(H1N1)
42	FJ969538	A/California/07/2009(H1N1)

43	GQ149643	A/Mexico/4486/2009(H1N1)
44	GQ117110	A/Michigan/02/2009(H1N1)
45	CY039905	A/New York/1682/2009(H1N1)
46	GQ168882	A/New York/13/2009(H1N1)
47	GQ160598	A/Nebraska/03/2009(H1N1)
48	GQ168846	A/New York/22/2009(H1N1)
49	GQ168848	A/Massachusetts/06/2009(H1N1)
50	GQ168874	A/Massachusetts/07/2009(H1N1)
51	FJ984343	A/New York/11/2009(H1N1)
52	FJ984334	A/New York/12/2009(H1N1)
53	GQ117038	A/California/14/2009(H1N1)
54	FJ984356	A/New York/31/2009(H1N1)
55	FJ984370	A/New York/10/2009(H1N1)
56	FJ984339	A/New York/06/2009(H1N1)
57	FJ984377	A/New York/15/2009(H1N1)
58	FJ984389	A/New York/19/2009(H1N1)
59	FJ984361	A/New York/23/2009(H1N1)
60	FJ984349	A/New York/18/2009(H1N1)
61	GQ168868	A/New York/20/2009(H1N1)
62	EU097929	A/New Caledonia/20/1999(H1N1)
63	CY034120	A/Wisconsin/67/2005(H3N2)
64	AY676051	A/chicken/Korea/ES/03(H5N1)

65	DQ464358	A/swan/Germany/R65/2006(H5N1)
66	CY030283	A/duck/Vietnam/NCVD-6/2007(H5N1)
67	AJ293941	A/Hong Kong/1774/99(H3N2)
68	FJ415613	A/swine/Zhejiang/1/2007(H1N1)
69	AB434300	A/swine/Ratchaburi/NIAH550/2003(H1N1)
70	EF101744	A/Philippines/344/2004(H1N2)
71	EU924274	A/swine/Nordkirchen/IDT1993/2003(H3N2)
72	FJ798782	A/swine/Hungary/19774/2006(H1N1)
73	CY009896	A/Swine/Spain/50047/2003(H1N1)
74	AB434292	A/swine/Ratchaburi/NIAH1481/2000(H1N1)
75	AB434332	A/swine/Chachoengsao/NIAH587/2005(H1N1)
76	CY033424	A/American wigeon/California/HKWF42/2007(H6N1)
77	AY619957	A/swine/Saskatchewan/18789/02(H1N1)
78	EU880824	A/turkey/CA/358533/2005(H4N8)
79	AY300993	A/duck/NY/186875/02(H5N2)

TABLE XVII: THE FEATURE PARAMETERS OF 12 SAMPLES

Genbank Acce. No.	α	β	γ	λ	θ	ϕ	σ
A00033	0.386	0.500	0.491	0.9933	0.9702	0.9415	0.2839
A17677	0.508	0.544	0.565	1.0757	1.0307	0.8789	0.2588
A11542	0.422	0.470	0.497	1.0431	1.0104	0.8742	0.2876
A22239	0.465	0.495	0.480	1.0673	1.0353	0.9351	0.2951
A24782	0.513	0.572	0.634	1.1773	1.1755	1.1214	0.3233
Z31371	0.500	0.526	0.558	1.0233	1.0336	1.0200	0.2732
M28289	0.576	0.635	0.632	1.3088	1.2991	1.0684	0.3462
V01510	0.575	0.617	0.716	1.3347	1.2171	1.0992	0.3429
U25810	0.560	0.657	0.623	1.2284	1.2740	1.1454	0.3340
M13580	0.480	0.622	0.624	1.2339	1.2114	1.0537	0.3337
U06674	0.507	0.498	0.555	1.1821	1.1609	0.9518	0.3274
J02989	0.518	0.577	0.537	1.2192	1.2463	1.0363	0.3495

TABLE XVIII: 5-FOLD CROSS VALIDATION ON ZHANG'S DATASET(%)

Methods	1	2	3	4	5	average
ExonScan	70.00	67.50	77.50	72.50	70.00	71.50 ± 3.79
Genescan	78.50	80.50	83.50	82.50	76.50	80.30 ± 2.86
N-SCAN	82.50	85.00	90.00	82.50	87.50	85.50 ± 3.26
Z-Curve	87.50	92.50	92.50	87.50	90.00	90.00 ± 2.50
ZC method	95.00	97.50	97.50	97.50	95.00	96.50 ± 1.37

TABLE XIX: 5-FOLD CROSS VALIDATION ON FEATURE PARAMETERS(%)

Methods	1	2	3	4	5	average
SVM+RBF 1	87.50	90.00	95.00	90.00	92.50	91.00 ± 2.85
SVM+RBF 2	92.50	95.00	95.00	97.50	90.00	94.00 ± 2.85
SVM+RBF 3	95.00	97.50	97.50	97.50	95.00	96.50 ± 1.37

TABLE XX: PREDICTION RESULTS FROM DIFFERENT METHODS(%)

Methods	1	2	3	4	5	average
GENSCAN	76.50	74.00	76.75	78.25	77.50	76.60 ± 1.61
N-SCAN	82.50	81.75	83.75	80.25	81.50	81.95 ± 1.29
Z-Curve	88.75	87.25	85.25	83.75	85.75	86.15 ± 1.90
ZC method	94.75	93.50	92.75	91.75	92.50	93.05 ± 1.14

CITED LITERATURE

1. Darwin C.: On the Origin of Species. John Murray, 1859.

2. Woese, C. and Fox, G.: Phylogenetic structure of the prokaryotic domain: the primary kingdoms. Proceedings of the National Academy of Sciences, 74(11):5088–5090, 1977.

3. Jukes, T.H. and Cantor, C.R.: Evolution of protein molecules. In Munro, H.N.. Mammalian protein metabolism. New York: Academic Press., pp.21–123, 1969.

4. Kimura, M.: A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. Journal of Molecular Evolution, 16(2): 111–120, 1980.

5. Felsenstein, J.: Evolutionary trees from DNA sequences: a maximum likelihood approach. Journal of Molecular Evolution, 17(6): 368–376, 1981.

6. Hasegawa, M., Kishino, H. and Yano, T.: Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. Journal of Molecular Evolution, 22(2): 160–174, 1985.

7. Tavaré, M.: Some probabilistic and statistical problems in the analysis of DNA sequences. Lectures on Mathematics in the Life Sciences (American Mathematical Society), 17: 57–86, 1986.

8. Tamura, K.: Estimation of the number of nucleotide substitutions when there are strong transition-transversion and G+C-content biases. Molecular Biology and Evolution, 9(4): 678–687, 1992.

9. Tamura, K. and Nei, M.: Estimation of the number of nucleotide substitutions in the control region of mitochondrial DNA in humans and chimpanzees. Molecular Biology and Evolution, 10(3): 512–526, 1993.

10. Amano, K. and Nakamura, H.: Self-organizing clustering: a novel non-hierarchical method for clustering large amount of DNA sequences.

    Genome Informatics, 14:575–576, 2003.

11. Emrich, S.J., Kalyanaraman, A. and Aluru, A.: Algorithms for large-scale clustering and assembly of biological sequence data.

    Handbook of Computational Molecular Biology, ChapmanHall / CRC Press, Boca Raton, FL.

12. FitzGerald, P.C., Shlyakhtenko, A., Mir, A. and Vinson, C.: Clustering of DNA sequences in human promoters.

    Genome Research, 14:1562–1574, 2004.

13. Waterman, S.M.: Introduction to Computational Biology: Maps, Sequences and Genomes.

    ChapmanHall / CRC Press, Boca Raton, FL, 431pp.

14. Abe, T., Kanaya, S., Kinouchi, M., Ichiba, Y., Kozuki, T. and Ikemura, T.: Informatics for unveiling hidden genome signatures.

    Genome Research, 13:693–702, 2003.

15. Chuzhanova, N.A., Jones, A.J. and Margetts, S.: Feature selection for genetic sequence classification.

    Bioinformatics, 14:139–143, 1998.

16. Karlin, S. and Ladunga, I.: Comparisons of eukaryotic genomic sequences. Proceedings of National Academic Science, USA, 91: 12832–12836, 1994.

17. Nakashima, H., Ota, M., Nishikawa, K. and Ooi, T.: Genes from nine genomes are seperated into their organisms in the dinucleotide composition space.

    DNA Research, 5:251–259, 1998.

18. Yau, S., Wang, J. and Niknejad, A.: Dna sequence representation without degeneracy. Nucleic Acids Research, 31(12):3078–3080, 2003.

19. Liu, L., Ho, Y.-K. and Yau, S.S.-T.: Clustering DNA sequences by feature vectors. Molecular Phylogenetic Evolution, 41:64–69, 2006.

20. Yau, S.S.-T., Yu, C. and He, R.: A protein map and its application. DNA and Cell Biology, 27(5):241–250, 2008.

21. Carr, K., Murray, E., Armah, E., He, R.L. and Yau, S.S.-T.: A rapid method for characterization of protein relatedness using feature vectors. PloS ONE, 5(3):e9550, 2010.

22. Yu, C., Liang, Q., Yin, C., He, R.L. and Yau, S.S.-T.: A novel construction of genome space with biological geometry. DNA Research, 17(3):155–168, 2010.

23. Larkin M., Blackshields, G. and Brown, .N.: Clustal w and clustal x version 2.0. Bioinformatics, 23(21):2947–8, Nov 2007.

24. Edgar, R.C.: MUSCLE: a multiple sequence alignment method with reduced time and space complexity. BMC Bioinformatics, 5:113, 2004.

25. Katoh, K., Misawa, K., Kuma, K. and Miyata, T.: MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. Nucleic Acids Research, 30(14): 3059–3066, 2002.

26. Wang, L. and Jiang, T.: On the complexity of multiple sequence alignment. Journal of Computational Biology, 13(7): 1323–1339, 1994.

27. Musto, H., Cacciò, S., Bernardi, G.: Compositional constraints in the extremely GC-poor genome of Plasmodium falciparum. Memorias do Instituto Oswaldo Cruz, 92(6): 835–841, 1997.

28. Jacobson, N.: Basic Algebra, Vol. 1. Hindustan Publishing Corporation (India), pp.135, 1974.

29. Shinde, V., Bridges, C.B., Uyeki, T.M., Shu, B. and Balish, A.: Triple-reassortant swine influenza A (H1) in humans in the United States, 2005-2009. New England Journal of Medicine, 360: 2616–2625, 2009.

30. Garten, R.J., Davis, C.T., Russell, C.A., Shu, B. and Lindstrom, S.: Antigenic and Genetic Characteristics of Swine-Origin 2009 A (H1N1) Influenza Viruses Circulating in Humans. Science, 325(5937): 197–201, 2009.

31. Novel Swine-Origin Influenza A (H1N1) Virus Investigation Team : Emergence of a novel swine-origin influenza A (H1N1) virus in humans. New England Journal of Medicine, 360: 2605–2615, 2009.

32. Scholtissek, C.: Pigs as 'mixing vessels' for the creation of new pandemic influenza A viruses. Medical Principles and Practice, 2: 65–71, 1990.

33. Ito, T., Couceiro, J.N., Kelm, S. Baum, L.G. and Krauss, S.: Molecular basis for the generation in pigs of influenza A viruses with pandemic potential. Journal of Virology, 72: 7367–7373, 1998.

34. Ma, W., Kahn, R.E., Richt, J.A.: The pig as a mixing vessel for influenza viruses: human and veterinary implications. Journal of Molecular Genetic Medicine, 3: 158–166, 2009.

35. Belshe, R.B.: Implications of the emergence of a novel H1 Influenza virus. New England Journal of Medicine, 360(25): 2667–2668, 2009.

36. Kingsford, C., Nagarajan, Salzberg, S.: 2009 swine-origin influenza A (H1N1) resembles previous influenza isolates. PloS ONE, 4(7): e6402.doi:10.1371/journal.pone.0006402, 2009.

37. Kou, Z., Hu, S. and Li, T.: Genome evolution of novel influenza A (H1N1) viruses in humans. China Science Bulletin, 54: 2159–2163, 2009.

38. Efron, B. and Tibshirani, J.R.: An introduction to bootstrap. Chapman&Hall, New York, pp456, 1993.

39. Palmenberg, A., Spiro, D., Kuzmickas, R., Wang, S. and Djikeng, A.: Sequencing and analyses of all known human rhinovirus genomes reveal structure and evolution. Science, 324: 55–59, 2009.

40. Kumar, S., Dudley, J., Nei, M. and Tamura, K.: MEGA: A biologist-centric software for evolutionary analysis of DNA and protein sequences. Brief Bioinformatics, 9: 299–306, 2008.

41. Brown, W.M., Prager, E.M., Wang, A. and Wilson, A.C.: Mitochondrial DNA sequences of primates: Tempo and mode of evolution. Journal of Molecular Evolution, 18: 225–239, 1982.

42. Liu, F., Miyamoto, M., Freire, N., Ong, P. and Tennant, M.: Molecular and morphological supertrees for eutherian (placental) mammals. Science, 291: 1786–1789, 2001.

43. Raina, S., Faith, J., Disotell, T., Seligmann, H., Stewart, C. and Pollock, D.: Evolution of base-substitution gradients in primate mitochondrial genomes. Genome Research, 15(5):665–673, 2005.

44. Kullberg, M., Nilsson, M., Arnason, U., Harley, E. and Janke, A.: Housekeeping genes for phylogenetic analysis of Eutherian relationships. Molecular Biology Evolution, 23(8): 1493–1503, 2006.

45. Murtagh, F.: Multidimensional clustering algorithms. In: Chambers JM, Gordesch J, Klas A, Lebart L, Sint PP, editors. COMPSTAT Lectures. Physica-Verlag/Springer Press, Vienna, 4: 131pp, 1985.

46. Kamimura, T., Shimodaira, H., Imoto, S., Kim, S., Tashiro, K.: Multiscale bootstrap analysis of gene networks based on Bayesian networks and nonparametric regression. Genome Informatics, 14: 350–351, 2003.

47. Ladunga, I.: Phylogenetic continuum indicates galaxies in the protein universe: preliminary results on the natural group structures of proteins.
Journal of Molecular Evolution, 4:358–375, 1992.

48. Levitt, M.: Nature of the protein universe.

Proceedings of National Academy of Sciences USA, 106: 11079–11084, 2009.

49. Levitt, M.: Growth of novel protein structural data. Proceedings of National Academy of Sciences USA, 104: 3183–3188, 2007.

50. Li, W., Jaroszewski, L. and Godzik, A.: Clustering of highly homologous sequences to reduce the size of large protein databases. Bioinformatics, 17: 282–283, 2001.

51. Fitch, W.M.: Distinguishing homologous from analogous proteins. Systematic Zoology, 19: 99–113, 1970.

52. Hamori, E.: Novel DNA sequence representation. Nature, 314: 585–586, 1985.

53. Gates, M.A.: Simpler DNA sequence representations. Nature, 316: 219, 1985.

54. Randic, M., Vracko, M., Lers, N. and Plavsic, D.: Novel 2-D graphical representation of DNA sequences and their numerical characterization. Chemical Physics Letters, 368: 1–6, 2003.

55. Grantham, R.: Amino acid difference formula to help explain protein evolution. Science, 185: 862–864, 1974.

56. Jones, D.D.: Amino acid properties and side-chain orientation in proteins: a cross correlation approach. Journal of Theoretical Biology, 50: 167–184, 1975.

57. Lanier, W., Moustafa, D., Bhattacharya, D. and Comeron, J.: EST analysis of Osterococcus lucimarinus, the most compact Eukaryotic genome, shows an excess of introns in highly expressed genes. PloS ONE, 3(5):1–7, 2008.

58. Buldyrev, S.V., Peng, C.-K. and Stanley, H.E.: Long-range correlation properties of coding and noncoding DNA sequences: Genbank analysis. Physics Review E, 51: 5084–5091, 1995.

59. Prabhu, V.V. and Claverie, J.-M.: Correlations in intronless DNA. Nature, 359: 782, 1992.

60. Zhang, C.-T., Lin, Z.-S., Yan, M. and Zhang, R.: A novel approach to distinguish between intron-containing and intronless genes based on the format of Z curves. Journal of Theoretical Biology, 192:467–473, 1998.

61. Vapnik, V.N.: The Nature of Statistical Learning Theory. Springer, 1995.

62. Buldyrev, S.V., Peng, C.-K. and Stanley, H.E.: Long-range correlation properties of coding and noncoding DNA sequences: Genbank analysis. Physics Review E, 51: 5084–5091, 1995.

63. Zhang, C.-T., Zhang, R. and Ou, H.-Y.: The Z curve database: a graphic representation of genome sequences. Bioinformatics, 19:593–599, 2003.

64. Ma, BG.: How to describe genes: Enlightenment from the quaternary number system. BioSystems, 90:20–27, 2007.

65. Yin, C. and Yau, S.S.-T.: A Fourier characteristic of coding sequences: origins and a non-Fourier approximation. Journal of Computational Biology, 12(9):1153–1165, 2005.

66. Yin, C. and Yau, S.S.-T.: Prediction of protein coding regions by the 3-base periodicity analysis of a DNA sequence. Journal of Theoretical Biology, 247:687–694, 2007.

67. Liu, L., Ho, Y.-K. and Yau, S.S.-T.: Prediction of primate splice site using inhomogeneous markov chain and neural network. DNA and Cell Biology, 26:477–483, 2007.

68. Burge, C. and Karlin, S.: Prediction of complete gene structures in human genomic DNA. Journal of Molecular Biology, 268: 78–94, 1997.

69. Batzoglou, S. and Pachter, L.: Human and mouse gene structure: Comparative analysis and application to exon prediction. Genome Research, 10(7):950–958, July 2000.

70. Bafna, V. and Huson, D.H.: The conserved exon method for gene finding. Proceedings of International Conference on Intelligent Systems for Molecular Biology, 8:3–12, 2000.

71. Korf, I., Flicek, P., Duanm, D. and Brent, M.R.: Integrating genomic homology into gene structure prediction. Bioinformatics, 17(supp 1):140–148, 2001.

72. Gross, S.S. and Brent, M.R.: Using multiple alignments to improve gene prediction. Journal of Computational Biology, 13(2):379–393, 2006.

73. Alexandersson, M., Cawley, S. and Pachter L.: SLAM: Cross-species gene finding and alignment with a generalized pair hidden Markov model. Genome Research, 13:496–502, 2003.

74. Pedersen, J.S. and Hein, J.: Gene finding with a hidden Markov model of genome structure and evolution. Bioinformatics, 19(2):219–227, 2003.

75. Siepel, A. and Haussler, D.: Combining phylogenetic and hidden Markov models in biosequence analysis. Journal of Computational Biology, 11(2):413–428, 2004.

76. Carter, D. and Durbin, R.: Vertebrate gene finding from multiple-species alignments using a two-level strategy. Genome Biology, 7(supp 1):S6, 2006.

77. Shu, J.H., Yun, S.C., Lin, C.Y. and Tang, C.Y.: MUSCLE: EXONSCAN: EXON prediction with Signal detection and Coding region AligNment in homologous sequences. Proceedings of the 2005 ACM symposium on Applied computing., 202–203, 2005.

78. Brent, M.R.: Genome annotation past, present, and future: How to define an ORF at each locus. Genome Research, 15: 1777–1786, 2005.

79. Arumugam, M., Wei, C., Brown, R. and Brent, M.R.: Pairagon+N-SCAN_EST: a model-based gene annotation pipeline. Genome Biology, 7(supp 1): S5, 2006.

80. Birney, E., Clamp, M. and Durbin, R.: GeneWise and Genomewise. Genome Research, 14: 988–995, 2004.

81. Djebali, S., Delaplace, F. and Crollius, H.R.: Exogean: a framework for annotating protein-coding genes in eukaryotic genomic DNA. Genome Biology, 7(supp 1):S7, 2006.

82. Zhang, C.-T.: A symmetrical theory of DNA sequences and its applications. Journal of Theoretical Biology, 187: 297–306, 1997.

83. Zhang, R. and Zhang, C.-T.: Z curves, an intuitive tool for visualizing and analyzing the DNA sequences. Journal of Biomolecular Structure Dynamics, 11:767–782, 1994.

84. Aquino-Pinero, E. and Valle, N.R.-D.: Characterization of a protein kinase C gene in Sporothrix schenckii and its expression during the yeast-to-mycelium transition. Medical Mycology, 40: 185–199, 2003.

85. Hurley, J.H., Newton, A.C., Parker, P.J., Blumberg, P.M. and Nishizuka, Y.: Taxonomy and function of C1 protein kinase C homology domains. Protein Science, 6: 477–480, 1997.

86. Webb, B.L.J., Hirst, S.J. and Giembycz, M.A.: Protein kinase C isoenzymes: a review of their structure, regulation and role in regulating airways smooth muscle tone and mitogenesis. British Journal of Pharmacology, 130: 1433–1452, 2000.

87. Gross, S.S., Do, C.B., Sirota, M. and Batzoglou, S.: Using multiple alignments to improve gene prediction. Journal of Computational Biology, 13(2):379–393, 2006.

88. Nekrutenko, A. and Li, M.: Assessment of compositional heterogeneity within and between eukaryotic genomes. Genome Research, 10(12):1986–1995, 2000.

89. Gate, M.: Simpler DNA sequence representations. Nature, 316(6025):219, Jul 18-24 1985.

90. R Development Core Team: R: A Language and Environment for Statistical Computing. Vienna Austria R Foundation for Statistical Computing, 2009. ISBN 3900051070.

91. Smith, G., Vijaykrishna, D., Bahl, J., Lycett, S., Worobey, M., Pybus,O., Ma, S., Cheung, C., Raghwani, J., Bhatt, S., Peiris, J., Guan, Y. and Rambaut, A.: Origins and evolutionary genomics of the 2009 swine-origin h1n1 influenza a epidemic. Nature, 459(7250):1122–1125, June 2009.

92. Solovyov, A., Palacios, G., Briese, T., Lipkin, W. and Rabadan, R.: Cluster analysis of the origins of the new influenza a(h1n1) virus. Eurosurveillance, 14, 2009.

93. Fitch, W. and Margoliash, E.: Construction of phylogenetic trees. Science, 155(760):279–84, Jan 1967.

94. Boore, J. and Brown, W.: Big trees from little genomes: mitochondrial gene order as a phylogenetic tool. Current Opinion in Genetics and Development, 8(6):668–674, 1998.

95. Wolstenholme, D.: Animal mitochondrial DNA: Structure and evolution. International Review of Cytology, 141:173 – 216, 1992.

Web site references

http://www.ncbi.nlm.nih.gov/; Genbank official home page.

ftp://ftp.ebi.ac.uk/pub/databases/swissprot/special_selections/; Web site for downloading prokaryotic genes.

http://genes.mit.edu/GENSCAN.html; GENSCAN paper and software from Burge Lab page.

http://genes.mit.edu/exonscan/; EXONSCAN paper and software from Burge Lab page.

http://mblab.wustl.edu/; N-SCAN paper and software from Brent Lab page.

VITA

Contact

Mo Deng

322 Science and Engineering Offices (M/C 249)

851 S. Morgan Street, Chicago, IL 60607

`deng_mo@hotmail.com`

Education

- PH.D in Mathematical Computer Science, University of Illinois at Chicago. 2011
- M.S. in Operations Research, East China Normal University, China. 2006
- B.S. in Mathematics, Northwest Normal University, China. 2003

Work Experience

- Teaching Assistant at University of Illinois at Chicago

  Chicago,IL,USA, 2006.8-2011.5

- R&D Division, Financial Engineer, Huashang Fund, Ltd

  Beijing, China, 2010.12-2011.1

Publications

1. A novel method of characterizing genetic sequences: genome space with biological distance and applications (with C. Yu, Q. Liang, R. He and Stephen Yau), PloS ONE, 6(3), 2011, e17293.
2. A novel method of realizing the nature of protein universe by means of natural protein space (with R. He and S. Yau), 10pp. in ms., submitted for publication.

3. A new approach to classify DNA sequences into intron-containing and intron-less sequences (with C. Yu, L. Zheng and S. Yau), 15pp. in ms., submitted for publication.
4. DNA Sequence Comparison by a Novel Probabilistic Method (with C. Yu and S. Yau), Information Sciences, 181(8): 1484–1492, 2011.
5. Minimum cycle bases of graphs on surfaces (with R. Han), Discrete Mathematics, 307: 2654–2660, 2007.