CONTENTS

CHAPTER 28

MAXIMUM ENTROPY METHOD FOR COMPOSITION VECTOR METHOD (RAYMOND H.-F. CHAN, ROGER W. WANG, JEFF C.-F. WONG)

28.1 INTRODUCTION

In the past few decades, a large volume of molecular sequences has been collected, from which the evolution and traits of the related living organisms are investigated. These sequences all look simple; for instance, the DNA sequence, no matter how long it is, contains only four different nucleotides `A`, `C`, `G` and `T`; so it is not surprising that on the surface these sequences themselves cannot tell us much. In order to reveal the hidden information, the use of the so-called sequence comparison is an essential tool. Sequence comparison methods can be divided into two main categories: alignment-based [16, 18, 38, 39, 44, 52] and alignment-free [27, 30, 42, 45, 54].

The alignment-based methods use the dynamic programming (DP) method to "align" the sequences and then find the similarity and dissimilarity after the alignment. To compare two sequences of length n by any alignment-based method, both the computational cost and the memory requirement are $\mathcal{O}(n^2)$ [56]. Because of the accuracy of

(String Processing and Application to Biological Sequences, draft). By Mourad Elloumi and Albert Y. Zomaya (eds.)

the DP method, the alignment-based methods are widely used for analyzing gene sequences. However, different gene sequences may give different evolutionary results. For instance, based on the 16rRNA sequences, birds, which are more closely related to crocodilians, were grouped with mammals [58]. In addition, based on the gene sequences for MHG-CoA reductase, Archaeoglobus fulgidus, a definite archaean, was assigned into the bacteria group [15]. Nowadays, with the advent of sequence techniques, whole genome sequences have been generally accepted as excellent tools for the study of species differences and evolution [17]. However, aligning the whole genomes is a very challenging problem, since every species has its own gene content and gene order, and we do not know which two genes can be truly aligned. Furthermore, as the length of the genome sequences are usually very long, it is impossible to align the genome sequences due to the cost of the computational time and the memory requirement.

The alignment-free methods, in turn, are developed for overcoming the difficulty of the analysis of the whole genome phylogeny. They can be divided into three classes:

- the gene content method [45],
- the data compression method [26], and
- the composition vector (CV) method [27, 42].

For the gene content method, the distance between two species is defined by their number of common genes divided by the total number of genes in the genome sequences. The data compression method uses the distance between the compressed information from the genome sequences as the distance between the species. For the CV method, the composition vector is first constructed for each species based on its whole genome sequence, and the distance between the composition vectors is used as the distance between species. In this chapter, we shall only shed some light on the CV method.

The CV method was proposed by Hao *et al.* [27] for the phylogeny of bacteria, and it was very successful. The CV method generally consists of four steps as follows:

1. Construct the frequency vectors — different methods for constructing the frequency vectors are discussed based on the different types of biological sequences that are input.
2. Construct the composition vectors — the composition vectors are constructed with each entry being the signal-to-noise ratio. Several kinds of models are introduced for estimating the noise.

3. Compute the distance between composition vectors — several distance measures are introduced and analyzed.

4. Build the phylogenetic trees — we use the neighbor-joining method to draw the phylogenetic trees.

As we shall see below, there is a link between the maximum entropy optimization and some existing denoising formulas. Maximum entropy is being increasingly used as a general and powerful technique for making the classification of species through the biological sequences from noise itself when the data in the signal is obscured by noise and bias (e.g., [28]). Entropy can be justified in information-theoretic terms. Not only will we present some denoising formulas and suggest which one is the optimal one, but we will also introduce several models for the CV method and show that the CV method can also be applied successfully for phylogenetic analysis of tetrapod, hepatitis B virus, mammal and chorophlast. We even show that the CV method can provide some reasonable results where the alignment methods failed (see Example 1).

This chapter is divided into four sections. Section 1 gives the introduction. Section 2 includes the general formulation of the CV method. Section 3 presents the results using the CV method with different denoising formulas and compares them with other existing results. Section 4 gives the concluding remarks.

28.2 MODELS AND ENTROPY OPTIMIZATION

In Section 28.2.1, a list of formal definitions for the biological terms is introduced. In Section 28.2.2, two of the most common denoising formulas in literature are revisited: that advocated by Hao *et al.*'s formula [27, 42] and Yu *et al.*'s formula [60]. In particular, under the framework of the constrained optimization problem with the maximum entropy approach, we provide three new denoising formulas by means of the CV method, cf., (28.14), (28.16) and (28.17). Based on the angle-based distance approach, various types of distance formulas are also introduced in Section 28.2.3. Phylogenetic tree construction is described in Section 28.2.4.

28.2.1 Definitions

Definition 1 Consider a molecular sequence (DNA*/RNA† sequence or peptide/amino acid sequence) of length N. Any consecutive k molecules within the sequence are called a *k-string*, where $1 \leq k \leq N$.

Definition 2 The *observed frequency* $f(\alpha_1\alpha_2\cdots\alpha_k)$ of a k-string $\alpha_1\alpha_2\cdots\alpha_k$ is defined as

$$f(\alpha_1\alpha_2\cdots\alpha_k) = \frac{g(\alpha_1\alpha_2\cdots\alpha_k)}{N-k+1}, \tag{28.1}$$

where $g(\alpha_1\alpha_2\cdots\alpha_k)$ is the number of times that $\alpha_1\alpha_2\cdots\alpha_k$ appears in the sequence.

Let us define the frequency vector for the gene sequence and genome sequence, respectively.

Definition 3 For a gene sequence, whether a DNA sequence or a RNA sequence, there are 4^k possible k-strings. A vector is constructed with the frequency defined in (28.1) for each entry, and is called the *frequency vector*.

Consider the following nucleotide sequence that consists of `A, C, G, T` such that

`GACTACTACT.`

*Deoxyribonucleic acid
†Ribonucleic acid

Set $k = 3$ and $N - k + 1 = 8$. The total number of possible different 3-string sequences is then 4^3 and the frequency vector is given as follows:

$$\begin{bmatrix} f(\mathtt{AAA}) \\ f(\mathtt{AAC}) \\ \vdots \\ f(\mathtt{ACT}) \\ \vdots \\ f(\mathtt{CTA}) \\ \vdots \\ f(\mathtt{GAC}) \\ \vdots \\ f(\mathtt{TAC}) \\ \vdots \\ f(\mathtt{TTG}) \\ f(\mathtt{TTT}) \end{bmatrix}_{4^3} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 3/8 \\ \vdots \\ 2/8 \\ \vdots \\ 1/8 \\ \vdots \\ 2/8 \\ \vdots \\ 0 \\ 0 \end{bmatrix}.$$

It is worth mentioning that a window of the length of k-string is used that slides it through the sequences by shifting one window at a time to look for the frequencies of each k-string.

There are three kinds of sequences available from the whole genome sequence:

1. The whole DNA sequence

For the whole DNA sequence, the frequency of appearance of a k-string is also defined in (28.1).

2. The protein-coding DNA sequences

Definition 4 For the protein-coding DNA sequences, the observed frequency of a k-string $\alpha_1\alpha_2\cdots\alpha_k$ in the whole sequence is defined as [42]

$$f(\alpha_1\alpha_2\cdots\alpha_k) = \frac{\sum_{j=1}^{m} g_j(\alpha_1\alpha_2\cdots\alpha_k)}{\sum_{j=1}^{m}(N_j - k + 1)}, \tag{28.2}$$

where m is the number of protein-coding gene sequences from the whole genome, $g_j(\alpha_1\alpha_2\cdots\alpha_k)$ is the number of times that $\alpha_1\alpha_2\cdots\alpha_k$ appears in the jth DNA sequence, and N_j is the length of the jth DNA sequence. A *frequency vector* is then constructed with each entry containing all the frequencies defined in (28.2).

3. The amino acid sequences of all protein-coding sequences

For the amino acid sequences of all protein-coding sequences, the frequency vector can be constructed similarly, with each entry defined in (28.2). A vector of length 20^k is then constructed.

28.2.1.1 Signal-to-Noise Ratio It is generally accepted that the phylogenetic signals in the biological sequence data are often obscured by noise and bias [10]. The relation between the signal and the noise can be formulated as a single mathematical formula, referred to as the composition vector. Given M molecular sequences, M frequency vectors of the same length $|\Omega|^k$ were defined earlier, where

$$|\Omega| = \begin{cases} 4, & \text{if the sequence is the DNA/RNA type,} \\ 20, & \text{if the sequence is the peptide/amino acid type.} \end{cases} \tag{28.3}$$

Definition 5 For each $f(\alpha_1\alpha_2\cdots\alpha_k)$, the frequency of appearance of the k-string $\alpha_1\alpha_2\cdots\alpha_k$ defined in (28.1), we will estimate its noise and denote it by $q(\alpha_1\alpha_2\cdots\alpha_k)$. Then the *composition vector* of one species is the $|\Omega|^k$-vector, where each nonzero entry equals

$$\frac{f(\alpha_1\alpha_2\cdots\alpha_k) - q(\alpha_1\alpha_2\cdots\alpha_k)}{q(\alpha_1\alpha_2\cdots\alpha_k)},$$

the *signal-to-noise ratio* of the k-string $\alpha_1\alpha_2\cdots\alpha_k$.

28.2.2 Denoising formulas

Let us review some existing denoising formulas for removing noises in the phylogenetic signals.

28.2.2.1 Hao's Formula Given any molecular sequence, Hao *et al.* [27, 42] employed the following formula

$$q^{\text{Hao}}(\alpha_1\alpha_2\cdots\alpha_k) = \begin{cases} \dfrac{f(\alpha_1\cdots\alpha_{k-1})f(\alpha_2\cdots\alpha_k)}{f(\alpha_2\cdots\alpha_{k-1})}, & \text{if } f(\alpha_2\cdots\alpha_{k-1}) \neq 0, \\ 0, & \text{otherwise.} \end{cases} \tag{28.4}$$

to estimate the noise of the k-string $\alpha_1\cdots\alpha_k$, where $f(\mathtt{u})$ is the frequency of appearance of any string $\mathtt{u}$ in the sequence. To find the noise of the k-string by (28.4), using the $(k-2)$th order Markov assumption (together with the joint and conditional probability) [36], the appearance frequencies of the $(k-1)$-string and the $(k-2)$-string are established. If the denominator in (28.4) $f(\alpha_2\cdots\alpha_{k-1})$ is found to be zero, then it means that the $(k-2)$-string does not appear in the sequence. Obviously the $(k-1)$-strings $\alpha_1\cdots\alpha_{k-1}$ and $\alpha_2\cdots\alpha_k$ will also not appear in the sequence, and then

$$f(\alpha_1\cdots\alpha_{k-1}) = f(\alpha_2\cdots\alpha_k) = 0.$$

When this degeneracy case happens, one can simply let

$$q^{\text{Hao}}(\alpha_1\cdots\alpha_k) = 0.$$

Formula (28.4) is derived from the observed frequency $f(\cdot)$ that represents the probability. It was Brendel *et al.* [4] in 1986 who originally introduced (28.4) for revealing the functional and evolutionary relatedness of word sequence. Hao *et al.* [27, 42] used formula (28.4) for the phylogenetic analysis of prokaryotes, based on whole genome sequences.

28.2.2.2 Yu's Formula Given any molecular sequence, Yu *et al.* [60] proposed the following formula

$$q^{\text{Yu}}(\alpha_1\alpha_2\cdots\alpha_k) = \frac{f(\alpha_1)f(\alpha_2\cdots\alpha_k) + f(\alpha_1\cdots\alpha_{k-1})f(\alpha_k)}{2} \tag{28.5}$$

to find the noise of the k-string $\alpha_1\alpha_2\cdots\alpha_k$, where $f(\mathtt{u})$ is the appearance frequency of any string $\mathtt{u}$ in the sequence. A salient feature of (28.5) is that it takes the average of the sum of two independent events with respect to $f(\alpha_1)f(\alpha_2\cdots\alpha_k)$ and $f(\alpha_1\cdots\alpha_{k-1})f(\alpha_k)$.

Formula (28.5) is taken from the observed frequency $f(\cdot)$ that represents the probability. Application of (28.5) was common in the area of complex and dynamic systems, e.g., [59]. Yu *et al.* [60] used formula (28.5) for the phylogenetic analysis of prokaryotes, chloroplasts and other phylogenetic problems, based on whole genome sequences.

28.2.2.3 Establishing denoising formulas using the maximum entropy principle For the sake of simplicity, we only consider DNA/RNA sequences in our formulation, but the amino acid sequences can be used in a similar fashion.

Let us consider the following constraints [28]:

$$\begin{cases} q(\mathtt{vA}) + q(\mathtt{vC}) + q(\mathtt{vG}) + q(\mathtt{vT}) = f(\mathtt{v}), \\ q(\mathtt{Av}) + q(\mathtt{Cv}) + q(\mathtt{Gv}) + q(\mathtt{Tv}) = f(\mathtt{v}), \end{cases} \tag{28.6}$$

where $q(\cdot)$ is the frequency to be maximized from the entropy when the observed frequency $f(\mathtt{v})$ for all $(k-1)$-strings $\mathtt{v}$ are given. The solution of the optimization problem is (28.4). We assume that the noises of the k-strings are related to (28.6), i.e., $q(\mathtt{vA}) + q(\mathtt{vC}) + q(\mathtt{vG}) + q(\mathtt{vT})$ and $q(\mathtt{Av}) + q(\mathtt{Cv}) + q(\mathtt{Gv}) + q(\mathtt{Tv})$ are known functions of $\mathtt{v}$, and we assume that the two sums are not identical to each other since their values can be changed and will lead to different denoising formulas as we shall see below.

28.2.2.4 Formulation of the optimization problem Let us propose our noise model as follows: The noise $q(\cdot)$ of the 4^k's k-strings satisfies

$$\begin{cases} q(\mathtt{vA}) + q(\mathtt{vC}) + q(\mathtt{vG}) + q(\mathtt{vT}) = l(\mathtt{v}), \\ q(\mathtt{Av}) + q(\mathtt{Cv}) + q(\mathtt{Gv}) + q(\mathtt{Tv}) = r(\mathtt{v}), \end{cases} \tag{28.7}$$

where $l(\mathtt{v})$ and $r(\mathtt{v})$ are given non-negative numbers for each $(k-1)$-string $\mathtt{v}$, and the right hand sides of (28.7) are obtained from the observed frequencies of any given sequence. Note that in (28.7), depending on the choice of $(k-1)$-strings, there are $(2 \cdot 4^{k-1})$ constraints and 4^k unknowns. Thus, when the number of constraints is fewer than the number of unknowns, the system is under-determined and the solution is not unique.

To obtain the unique $q(\mathtt{u})$, we maximize their entropy. More precisely, let $q_i \equiv q(\mathtt{u}_i)$ be the noise of the k-string $\mathtt{u}_i$, then we obtain q_i by solving the constrained maximisation problem:

$$\begin{aligned} &\text{maximize} \quad -\sum_{i=1}^{4^k} q_i \log q_i \\ &\text{subject to} \quad \begin{cases} q_i \text{ satisfies (28.7)}, \\ q_i \geq 0 \text{ for all } i. \end{cases} \end{aligned} \tag{28.8}$$

We note that $-q_i \log q_i$ is the entropy of q_i.

28.2.2.5 Solution of the optimization problem According to Pevzner [41], the best k-string for a sequence of length N is $\log_4 \left[\frac{N(N-1)}{2}\right]$. Thus if

$N = 1000$, then the best k-string is about 10. Hence the optimization problem (28.8) will have about one million unknowns, and it is seemingly difficult to solve such a constraint problem. However, we have the following useful result.

Lemma 1 For $k \geq 2$, the problem (28.8) is decoupled into 4^{k-2} subproblems of size 8-by-16 each.

Proof

- Let us first see the structure/pattern of the coefficient matrix in (28.8) when $k = 3$. The other choice of k can be used similarly. As

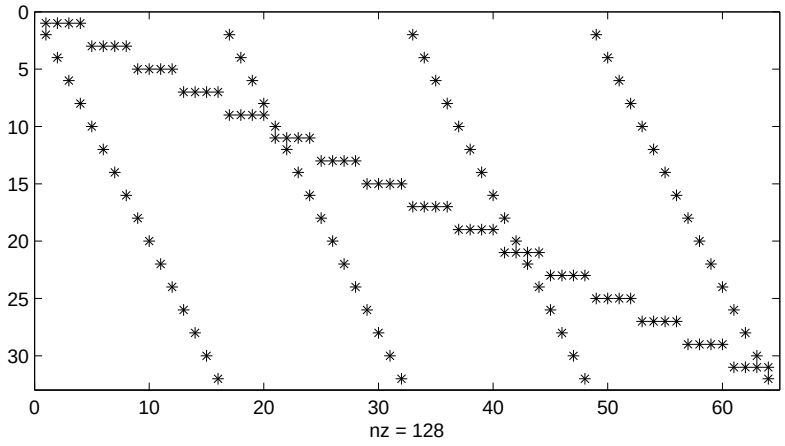

Figure 28.1 The pattern of the coefficient matrix in (28.8).

shown in Figure 28.1, the matrix is sparse and binary (containing 0 and 1 only), and the nonzero entries can be divided into two categories: the nonzero entries located on the "diagonal part" of the matrix form one category while the rest of the nonzero entries form the other category. To decouple these two categories, one simply rearranges the order of the equations:

– put the original odd-order equations first and
– locate the even-order equations later.

The pattern of the new coefficient matrix is shown in Figure 28.2.

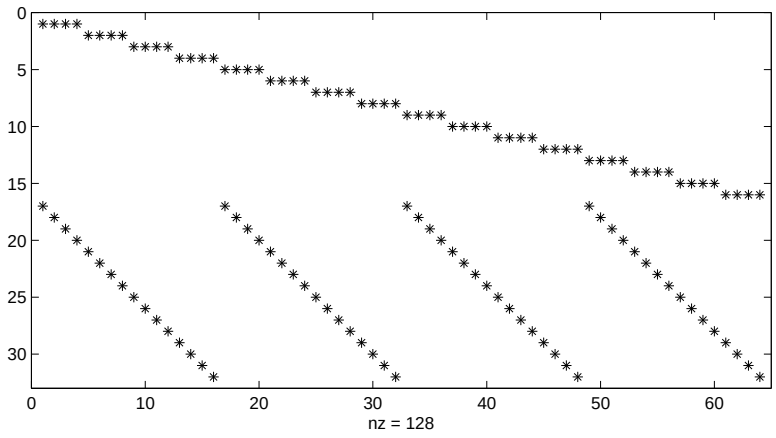

Figure 28.2 The pattern of the coefficient matrix after the permutation.

- In Figure 28.2, there are only four unknown variables q_1, q_2, q_3 and q_4 in the first row, and these four variables are also contained, respectively in four other rows of the matrix: the 17th, 18th, 19th and 20th. After carefully examining these four rows, we find that the following 16 variables

$$q_{16*(i-1)+j}, \qquad \forall\ i, j = 1, 2, 3, 4,$$

will not be found anywhere else but are totally contained in the following eight constraints, given in ascending order:

$$\text{Constraint}: \quad 1, \quad 5, \quad 9, \quad 13, \quad 17, \quad 18, \quad 19, \quad 20.$$

These eight constraints are clearly divorced from other constraints. Moreover, the 3-strings are contained in the above eight constraints if and only if they can be written in the following form

$$\mathtt{LAR}, \quad \forall\ \mathtt{L}, \mathtt{R} \in \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}.$$

Similarly, three groups of eight constraints are formed if and only if we also have the following cases:

$$\begin{aligned} &\texttt{LCR}, \quad \forall\ \texttt{L},\texttt{R} \in \{\texttt{A},\texttt{C},\texttt{G},\texttt{T}\},\\ &\texttt{LGR}, \quad \forall\ \texttt{L},\texttt{R} \in \{\texttt{A},\texttt{C},\texttt{G},\texttt{T}\},\\ &\texttt{LTR}, \quad \forall\ \texttt{L},\texttt{R} \in \{\texttt{A},\texttt{C},\texttt{G},\texttt{T}\}. \end{aligned}$$

Now we conclude that the original system can be decomposed into four sub-systems (see Figure 28.3) and the 3-strings are contained in each sub-system if and only if the 3-strings can be written in any of the four forms.

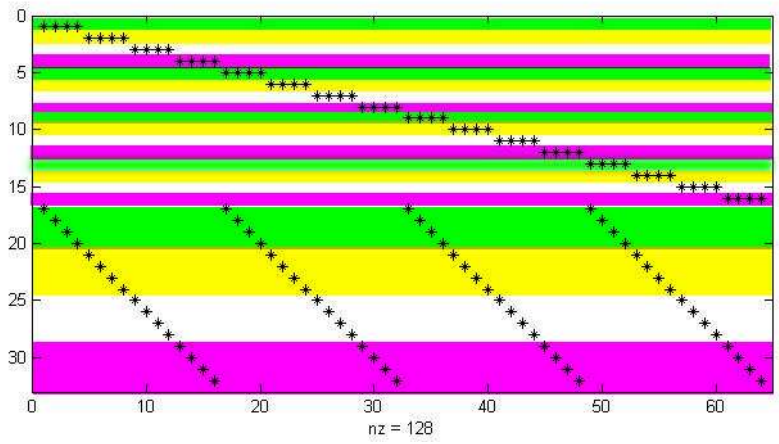

Figure 28.3 The figure for the decoupling of the permuted coefficient matrix.

With this idea, we now consider the general formulation of the equality constraints when $k \geq 3$. Let us rewrite the k-strings in the left hand sides of (28.7) as $\texttt{LwR}$, where $\texttt{w}$ is a $(k-2)$-string. By exhausting different combinations of $\texttt{L}$ and $\texttt{R}$, we obtain a system

of the following constraints for each $\mathtt{w}$:

$$\begin{cases} q(\mathtt{AwA}) + q(\mathtt{AwC}) + q(\mathtt{AwG}) + q(\mathtt{AwT}) = l(\mathtt{Aw}), \\ q(\mathtt{CwA}) + q(\mathtt{CwC}) + q(\mathtt{CwG}) + q(\mathtt{CwT}) = l(\mathtt{Cw}), \\ q(\mathtt{GwA}) + q(\mathtt{GwC}) + q(\mathtt{GwG}) + q(\mathtt{GwT}) = l(\mathtt{Gw}), \\ q(\mathtt{TwA}) + q(\mathtt{TwC}) + q(\mathtt{TwG}) + q(\mathtt{TwT}) = l(\mathtt{Tw}), \\ q(\mathtt{AwA}) + q(\mathtt{CwA}) + q(\mathtt{GwA}) + q(\mathtt{TwA}) = r(\mathtt{wA}), \\ q(\mathtt{AwC}) + q(\mathtt{CwC}) + q(\mathtt{GwC}) + q(\mathtt{TwC}) = r(\mathtt{wC}), \\ q(\mathtt{AwG}) + q(\mathtt{CwG}) + q(\mathtt{GwG}) + q(\mathtt{TwG}) = r(\mathtt{wG}), \\ q(\mathtt{AwT}) + q(\mathtt{CwT}) + q(\mathtt{GwT}) + q(\mathtt{TwT}) = r(\mathtt{wT}). \end{cases} \tag{28.9}$$

From (28.9), one notices that the right hand side cannot be set arbitrarily but must satisfy

$$\begin{aligned} l(\mathtt{Aw}) + l(\mathtt{Cw}) + l(\mathtt{Gw}) + l(\mathtt{Tw}) &= \sum_{\mathtt{L},\mathtt{R}\in\{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}} q(\mathtt{LwR}) \\ &= r(\mathtt{wA}) + r(\mathtt{wC}) + r(\mathtt{wG}) + r(\mathtt{wT}). \end{aligned} \tag{28.10}$$

Inspection of (28.9) also indicated that for each $\mathtt{w}$ a decoupled system is formed. In fact, for each $\mathtt{w}_i$, the unknowns $q(\mathtt{Lw}_i\mathtt{R})$ for different $\mathtt{L}$ and $\mathtt{R}$ can only occur in the constraints (28.9) for that particular $\mathtt{w}_i$, and will never occur in the constraints for $\mathtt{w}_j$, $j \neq i$. This is because $\mathtt{L}_i\mathtt{w}_i\mathtt{R}_i$ can never be equal to $\mathtt{L}_j\mathtt{w}_j\mathtt{R}_j$ for any possible $\mathtt{L}_i$, $\mathtt{L}_j$, $\mathtt{R}_i$, and $\mathtt{R}_j$. Obviously the objective function in (28.8) is already decoupled for each $\mathtt{w}_i$ as each term in the objective function involves only one $q(\mathtt{Lw}_i\mathtt{R})$. Hence we see that the optimization problem (28.8) can be decoupled into 4^{k-2} subproblems of the form (28.9). ■

Problem (28.8) can now be solved readily. To solve the subproblems, let us rewrite them as:

$$\begin{array}{ll} \text{maximize} & -\sum_{i,j=1}^{4} p_{ij} \log p_{ij} \\ \text{subject to} & \begin{cases} p_{i1} + p_{i2} + p_{i3} + p_{i4} = l_i, & i = 1,2,3,4 \\ p_{1j} + p_{2j} + p_{3j} + p_{4j} = r_j, & j = 1,2,3,4 \end{cases} \\ \text{and} & p_{ij} \geq 0, \quad i,j = 1,2,3,4, \end{array} \tag{28.11}$$

where p_{ij} are the unknowns $q(\mathtt{LwR})$ to be sought in (28.9).

Theorem 2 [8] The solution to (28.11) is

$$p_{ij} = \begin{cases} \dfrac{l_i r_j}{\sigma}, & \text{if } \sigma \neq 0, \\ 0, & \text{if } \sigma = 0, \end{cases} \tag{28.12}$$

where $\sigma \equiv l_1 + l_2 + l_3 + l_4 = r_1 + r_2 + r_3 + r_4$ (c. f. (28.10)).

28.2.2.6 Denoising formulas In this section, we derive some new denoising formulas that maximize the entropy. Two approaches are introduced:

1. The first approach is to apply existing formulas such as (28.4) and (28.5) in the left hand side of (28.9) to derive the right hand side functions $l(\cdot)$ and $r(\cdot)$, respectively.

2. The second approach is to apply existing formulas directly to the right hand side of (28.9).

For the first approach, two formulas are obtained.

Corollary 3 [8] For any 1-strings $\mathtt{Y}$ and $\mathtt{Z}$ and any $(k-2)$-string $\mathtt{w}$,

$$q(\mathtt{YwZ}) = \frac{f(\mathtt{Yw})f(\mathtt{wZ})}{f(\mathtt{w})}. \tag{28.13}$$

Formula (28.13) is identical to (28.4). Thus we have formally proved the claim in [28] that formula (28.4) satisfies the maximum entropy principle.

Let us examine Yu's formula (28.5) to see whether a new denoising formula can be derived.

Corollary 4 [8] For any 1-strings $\mathtt{Y}$ and $\mathtt{Z}$ and any $(k-2)$-string $\mathtt{w}$,

$$q(\mathtt{YwZ}) = \frac{1}{4\sigma}\left[f(\mathtt{Yw}) + f(\mathtt{Y})\sum_{\mathtt{R}} f(\mathtt{wR})\right]\left[f(\mathtt{wZ}) + f(\mathtt{Z})\sum_{\mathtt{L}} f(\mathtt{Lw})\right], \tag{28.14}$$

where

$$\sigma = \frac{1}{2}\left[\sum_{\mathtt{L}} f(\mathtt{Lw}) + \sum_{\mathtt{R}} f(\mathtt{wR})\right].$$

This formula, which satisfies the maximum entropy principle, is different from (28.5).

Our second approach to create new formulas stem from the following observation.

Lemma 5 [8] For all $(k-1)$-strings $\mathtt{w}$ and 1-strings $\mathtt{L}$ and $\mathtt{R}$, let $l(\mathtt{Lw}) = \alpha q(\mathtt{Lw})$ and $r(\mathtt{wR}) = \beta q(\mathtt{wR})$, where α and β are treated as normalization constants to fulfill the equality condition of (28.10). Then by (28.12),

$$q(\mathtt{LwR}) = \frac{q(\mathtt{Lw})q(\mathtt{wR})}{q(\mathtt{w})}. \tag{28.15}$$

To obtain two other new formulas, all we have to do is to substitute formulas (28.4) and (28.5) into the right hand side of (28.15). One can easily check that (28.15) satisfies (28.10).

Corollary 6 [8] Let $\mathtt{w}$ be $\mathtt{YxZ}$, where $\mathtt{x}$ is a $(k-4)$-string, and $\mathtt{Y}$ and $\mathtt{Z}$ are 1-strings. Then by using Hao's formula (28.4), (28.15) becomes

$$q(\mathtt{LYxZR}) = \begin{cases} \dfrac{f(\mathtt{LYx})f(\mathtt{YxZ})f(\mathtt{x})f(\mathtt{YxZ})f(\mathtt{xZR})}{\big[f(\mathtt{Yx})\big]^2\big[f(\mathtt{xZ})\big]^2}, & \text{if } f(\mathtt{Yx})f(\mathtt{xZ}) \neq 0, \\ 0, & \text{otherwise.} \end{cases} \tag{28.16}$$

Corollary 7 [8] Let $\mathtt{w}$ be $\mathtt{YxZ}$, where $\mathtt{x}$ is a $(k-4)$-string, and $\mathtt{Y}$ and $\mathtt{Z}$ are 1-strings. Then by using Yu's formula (28.5), (28.15) becomes

$$q(\mathtt{LYxZR}) = \begin{cases} \dfrac{[f(\mathtt{L})f(\mathtt{YxZ}) + f(\mathtt{LYx})f(\mathtt{Z})][f(\mathtt{Y})f(\mathtt{xZR}) + f(\mathtt{YxZ})f(\mathtt{R})]}{2[f(\mathtt{Y})f(\mathtt{xZ}) + f(\mathtt{Yx})f(\mathtt{Z})]}, \\ \qquad\qquad \text{if } f(\mathtt{Y})f(\mathtt{xZ}) + f(\mathtt{Yx})f(\mathtt{Z}) \neq 0, \\ 0, \qquad \text{otherwise.} \end{cases} \tag{28.17}$$

We remark that only nucleotide sequences were considered here. The decoupled constraint matrices of the optimization problem are thereby of size $(2 \cdot 4^{k-1})$-by-4^k. If amino acid sequences are considered, the decoupled systems will be of size $(2 \cdot 20^{k-1})$-by-20^k. However, denoising formulas can still be derived similarly and their forms will be the same.

As observed, different right hand side functions $l(\mathtt{v})$ and $r(\mathtt{v})$ in (28.9) provide different denoising formulas. In this work, we provide two approaches for defining them. For the four data sets tested in this work, formula (28.16) and formula (28.17) each have their own merits. We note that only Hao's formula (28.4) and Yu's formula (28.5) were used in constructing the right hand sides. One may use other existing formulas to construct new denoising formulas via our two approaches.

28.2.3 Distance measure

Let $n = 4^k$ be the length of the composition vector, and $\mathbb{S}$ be the set of composition vectors. To find the evolutionary distance between

two species $\mathcal{A}$ and $\mathcal{B}$, we will compute the distance $d(\mathbf{a}, \mathbf{b})$ between their composition vectors $\mathbf{a} = (a_i)_{i=1}^n$ and $\mathbf{b} = (b_i)_{i=1}^n \in \mathbb{S}$, respectively. This distance is then used to represent the distance between their corresponding species. Assume the reciprocal of the length of the composition vector is computable, and assume there are no composition vectors $\mathbf{c}$ and $\mathbf{d}$ in $\mathbb{S}$ such that

$$\mathbf{c} = -\mathbf{d}. \tag{28.18}$$

This assumption is reasonable as (28.18) will be rarely occur in real applications (e.g., see (28.21)).

Once the conversion of sequences into frequency vectors was established, a variety of distances $d(\mathbf{a}, \mathbf{b})$ were immediately calculated. In the following, we will introduce some of angle-based distance measures which are widely utilized in practice [3, 27, 42, 49, 50, 54, 55]. We remark that those distances must satisfy the following conditions:

(1) (Non-negativity) $0 \leq d(\mathbf{a}, \mathbf{b}) < +\infty$ for all $\mathbf{a}$ and $\mathbf{b} \in \mathbb{S}$.

(2) (Identity of indiscernibles) $d(\mathbf{a}, \mathbf{b}) = 0$ if and only if $\mathbf{a} = \mathbf{b}$.

(3) (Symmetry) $d(\mathbf{a}, \mathbf{b}) = d(\mathbf{b}, \mathbf{a})$ for all $\mathbf{a}$ and $\mathbf{b} \in \mathbb{S}$.

But the "triangle inequality" of the "metric distance"

$$d(\mathbf{a}, \mathbf{b}) \leq d(\mathbf{a}, \mathbf{c}) + d(\mathbf{c}, \mathbf{b}), \qquad \forall \mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{S}, \tag{28.19}$$

is not required for those distances.

28.2.3.1 Angle-based distance To measure the distance between the composition vectors $\mathbf{a}$ and $\mathbf{b} \in \mathbb{S}$, it is common to employ the cosine of their angle as defined below [2],

$$\cos\theta = \frac{\mathbf{a}^T\mathbf{b}}{\|\mathbf{a}\| \cdot \|\mathbf{b}\|}, \tag{28.20}$$

where $\|\cdot\|$ is the Euclidean vector norm (i.e., $\|\mathbf{a}\| = \sqrt{\sum_{i=1}^n (a_i)^2}$).

It was Stuart *et al.* [49, 50] who were the first to introduce the angle distance for the phylogenetic analysis. A formula is given by

$$d^{\text{Stuart}}(\mathbf{a}, \mathbf{b}) = -\log\left(\frac{1+\cos\theta}{2}\right) = -\log\left[\frac{1}{2}\left(1 + \frac{\mathbf{a}^T\mathbf{b}}{\|\mathbf{a}\| \cdot \|\mathbf{b}\|}\right)\right]. \tag{28.21}$$

Formula (28.21) is a distance on the set $\mathbb{S}$. Using the Cauchy-Schwarz inequality, one can show that d^{Stuart} satisfies the first and the second

conditions of a distance. Moreover, d^{Stuart} satisfies the third condition. Because of characteristics of the logarithm function, $d^{\text{Stuart}}(\mathbf{a}, \mathbf{b})$ can be sufficiently large if the angle between $\mathbf{a}$ and $\mathbf{b}$ is large enough. For this reason, this distance can be utilized for the phylogenetic analysis of the data set where the species are far away from each other. In fact, Formula (28.21) has been applied successfully for the phylogenetic analysis of whole genomes of bacteria and vertebrates [47, 48, 49, 50, 57].

Hao *et al.* [27, 42] proposed the following formula

$$d^{\text{Hao}}(\mathbf{a}, \mathbf{b}) = \frac{1 - \cos\theta}{2} = \frac{1}{2}\left(1 - \frac{\mathbf{a}^T\mathbf{b}}{\|\mathbf{a}\| \cdot \|\mathbf{b}\|}\right). \tag{28.22}$$

One can verify that this measure is a distance satisfying the three conditions. Since the cosine value computed by (28.20) varies between -1 and 1, the function value $d^{\text{Hao}}(\mathbf{a}, \mathbf{b})$ is normalized to the interval (0, 1). Formula (28.22) is widely utilized and achieved great success in the phylogenetic analysis of whole genomes of bacteria, viruses, and vertebrates [11, 12, 21, 22, 31, 42, 60].

Although Hao's distance is defined based on the cosine of the angle, it has a close relationship with the Euclidean distance. To see how it works, let us take two vectors $\mathbf{c}$ and $\mathbf{d} \in \mathbb{R}^2$. The angle θ is a one-to-one mapping of the following vector

$$\frac{\mathbf{c}}{\|\mathbf{c}\|} - \frac{\mathbf{d}}{\|\mathbf{d}\|}.$$

Moreover, for the length of this vector, we have by the law of cosines [6] that

$$\left\| \frac{\mathbf{c}}{\|\mathbf{c}\|} - \frac{\mathbf{d}}{\|\mathbf{d}\|} \right\|^2 = 2 - 2\cos\theta.$$

Generally, we have the following property for Hao's distance (28.22). Given any vectors $\mathbf{a}$ and $\mathbf{b} \in \mathbb{S}$, their Hao's distance relates to the Euclidean distance between their normalized vectors $\frac{\mathbf{a}}{\|\mathbf{a}\|}$ and $\frac{\mathbf{b}}{\|\mathbf{b}\|}$ as follows:

$$\mathrm{d}^{\text{Hao}}(\mathbf{a}, \mathbf{b}) = \frac{1}{4}\left\| \frac{\mathbf{a}}{\|\mathbf{a}\|} - \frac{\mathbf{b}}{\|\mathbf{b}\|} \right\|^2. \tag{28.23}$$

It can be observed from (28.23) that Hao's distance is the square of a Euclidean distance and thereby does not satisfy the triangle inequality. If the triangle inequality is further required for the distance, we can define

$$d^{\text{NUD}}(\mathbf{a}, \mathbf{b}) = \frac{1}{2}\left\| \frac{\mathbf{a}}{\|\mathbf{a}\|} - \frac{\mathbf{b}}{\|\mathbf{b}\|} \right\|, \tag{28.24}$$

i.e., the Euclidean distance between their normalized vectors. This distance satisfies all the conditions of a "metric distance", and is the square root of Hao's distance. In addition, we can directly use the angle to measure the distance. Define

$$d^{\text{angle}}(\mathbf{a},\mathbf{b}) = \frac{1}{\pi}\arccos\Big(\frac{\mathbf{a}^T\mathbf{b}}{\|\mathbf{a}\|\cdot\|\mathbf{b}\|}\Big). \tag{28.25}$$

We see

$$d^{\text{angle}}(\mathbf{a},\mathbf{b}) \in (0,\ \ 1),$$

for all vectors $\mathbf{a}$, $\mathbf{b} \in \mathbb{S}$, and $d^{\text{angle}}(\mathbf{a},\mathbf{b})$ is a "metric distance". Newly defined distances (28.24) and (28.25) will be tested on more realistic data sets (e.g., see Example 4).

28.2.4 Phylogenetic tree construction

Given the molecular sequences for any n species $\mathcal{C}_1, \mathcal{C}_2, \cdots, \mathcal{C}_n$, $n \geq 4$, we construct their frequency vectors, and then the composition vectors $\mathbf{c}_1,\ \mathbf{c}_2,\ \cdots,\ \mathbf{c}_n$. The distance d_{ij}, $i,j = 1,2,\cdots,n$, between any two composition vectors $\mathbf{c}_i$ and $\mathbf{c}_j$ can be obtained by the angle-based distance measure described in Section 28.2.3. A distance matrix consists of the collection of the pairwise distances for all n species and is given by

$$\begin{array}{c} \\ \mathcal{C}_1 \\ \mathcal{C}_2 \\ \mathcal{C}_3 \\ \vdots \\ \mathcal{C}_n \end{array}
\begin{array}{c}
\begin{array}{ccccc} \mathcal{C}_1 & \mathcal{C}_2 & \mathcal{C}_3 & \cdots & \mathcal{C}_n \end{array} \\
\left[\begin{array}{ccccc}
0 & d_{12} & d_{13} & \cdots & d_{1n} \\
d_{21} & 0 & d_{23} & \cdots & d_{2n} \\
d_{31} & d_{32} & 0 & \cdots & d_{3n} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
d_{n1} & d_{n2} & d_{n3} & \cdots & 0
\end{array}\right]
\end{array}.$$

Any distance-based phylogenetic tree construction method may be employed to build the tree, for instance, the Fitch-Margoliash (FM) method [20], the Unweighted Pair Group Method with Arithmetic Mean method (UPGMA) [53], the neighbor-joining (NJ) method [43], etc. It is worth mentioning that the FM method is not feasible when the number of species is larger than 100, and the information about the branch length of the tree is not available if the UPGMA method is used. In this work, all the trees will be drawn by the NJ method. This algorithm is available in several software packages, for instance, PHYLIP [19], SPLITSTREE [29], MEGA [51], etc.

28.3 APPLICATION AND DICUSSION

Four worked examples are given. Until otherwise stated, for the purpose of distance measurement, Equation (28.23) is used throughout all the computational experiments. All the figures are produced by MEGA and SPLITSTREE.

28.3.1 Example 1

The phylogenetic relationship among tetrapods has been widely discussed in the area of phylogeny and evolution. One early topic was whether birds are more closely related to crocodilians or to mammals. Based on the traditional classification and the results that stemmed from a large amount of molecular, morphological and paleontological data, birds are thought to be grouped with crocodilians. However, many studies based on 18S rRNA sequences supported the grouping of birds and mammals [58]. Using the CV method without denoising, and with each of the five denoising formulas, every taxa were grouped to their corresponding amphibian, reptile, bird or mammal clade. However, inspection of Figure 28.4 indicated that with the denoising formula (28.16), the bird and reptile clades were grouped together, and the two oryctolaguses of the mammal clade are well-grouped. When the CV method was used without denoising, or with the denoising formulas (28.4), (28.5), (28.14), or (28.17), birds were grouped with mammals, and the two oryctolaguses were not grouped together. For further discussion, see [9].

28.3.2 Example 2

The characteristics of hepatitis B virus (HBV) genotype C subgroups in Hong Kong and their relationship with HBV genotype C in other parts of Asia were investigated by Chan *et al.* [7]. The full genome nucleotide sequences of 49 HBV genotype C isolated from Hong Kong local patients, together with 69 published HBV genotype C and 12 well-known HBV non-genotype C were first collected.

The multiple sequence alignment method [52] was used to align those sequences and the distance matrix was then obtained. One phylogenetic tree, called the "NJ tree", was thereby constructed by the NJ method [43, 51]. In their NJ tree, the HBV genotype C were divided into 2 subgroups:

- the genotype Ce,

- the genotype Cs.

Using the CV method without denoising, or with any denoising formula (28.4), (28.5), (28.14) or (28.17) respectively, every HBV of different genotype was correctly grouped to its corresponding genotype subgroup. In particular, the 49 genotype C isolated from Hong Kong were identically separated into 2 subgroups, genotype Ce and genotype Cs, where 39 isolates ($\sim 80\%$) belonged to the genotype Cs subgroup, and 10 isolates ($\sim 20\%$) belonged to the genotype Ce subgroup. Using (28.13), (28.14) and (28.17), the phylogenetic tree is the same. The CV tree with denoising formulas (28.14) was shown in Figure 28.5.

28.3.3 Example 3

In our third set of computational experiments, we have applied two denoising formulas (28.14) and (28.17) to the set of 20 complete mtDNA sequences, including 6 Primates, 8 Ferungulates (artiodactyls + cetaceans + perissodactyls + carnivores), 2 Rodents, and 3 outgroups (marsupials and monotremes), which is the same set of species as (e.g., [5]).

The phylogenetic relationship among mammals has been a long standing problem in the area of phylogeny and evolution. Using (28.14) and (28.17), the phylogenetic tree is the same. Figure 28.6 shows the phylogenetic trees calculated by (28.17) with $k = 14$. The result of (28.17) is identical to the one done in [5]. In Figure 28.6, four trees among primates, ferungulates, rodents and the outgroup are well-grouped using the CV method.

28.3.4 Example 4

In our last computational experiments, we have applied a denoising formula (28.17) to the set of 34 choroplast genomes (or complete protein genome sequences), including 2 Archaea, 7 Chlorophyte *s.l.*, 8 Eubacteria, 3 Eukaryote, 1 Glaucophyte, 4 Rhodophyte *s.l.*, and 9 Seed plants, which is the same set of species as [12, 60].

Figure 28.7 shows the phylogenetic trees calculated by (28.17) with $k = 6$. The result of (28.17) mostly agrees with the one done in [60]. Some salient features of Figure 28.7 can be summarized as follows:

- Based on the widely accepted endosymbiotic theory that chloroplasts sprang from a cyanobacteria-like ancestor [24, 25, 35], all the chloroplast genomes yield a clade branched in the domain of Eubacteria and are diverged from a most recent common ancestor at cyanobacteria. Our denoising formula is able to identify

cyanobacteria as the most closely related prokaryotes of chloroplasts, even though massive gene transfering from the endosymbiont to the nucleus of the host cell was found [32, 33, 34].

- The chloroplasts are divided into two major clades:

  1. The green plants sensu lato, or chlorophytes *s.l.* [40] include all taxa with a chlorophyte chloroplast, both primary and secondary endosymbioses in origin.
  2. The glaucophyte Cyanophora and members of rhodophytes *s.l.* refer to rhodophytes (or red algae, Cyanidium and Porphyra in the tree) and their secondary symbiotic derivatives (the heterokont Odontella and the cryotphyte Guillarida).

- Inspection of Figure 28.7 shows that cyanophora is mixed into rhodophytes *s.l.*. These findings have been reported in [14, 46], despite the fact that the glaucophyte (cyanophora is grouped into glaucophyte) represents the earliest branch in chloroplast evolution with the green plants *s.l.* and rhodophytes *s.l.* as sister taxa [1, 33, 34, 37]

- In chlorophyte *s.l.*, the green algae (i.e., Chlorella, Mesostigma, and Nephroselmis) and Euglena are basal to that lineage and the seed plants cluster together as a derived group. But, the relationships among the other taxa (i.e., Marchantia, Psilotum, and Chaetosphaeridium) deviate slightly from our traditional understanding, probably because of limited taxon sampling in these primitive green plants [60].

- Similar to the result of [12], Chlorella is connected between Euglena and a clade of Mesostigma and Nephroselmis.

As a check, different distance formulas (28.21), (28.23), (28.24) and (28.25), were used and the results of each formula shown no sizable differences. In addition, they yielded the same phylogenetic tree.

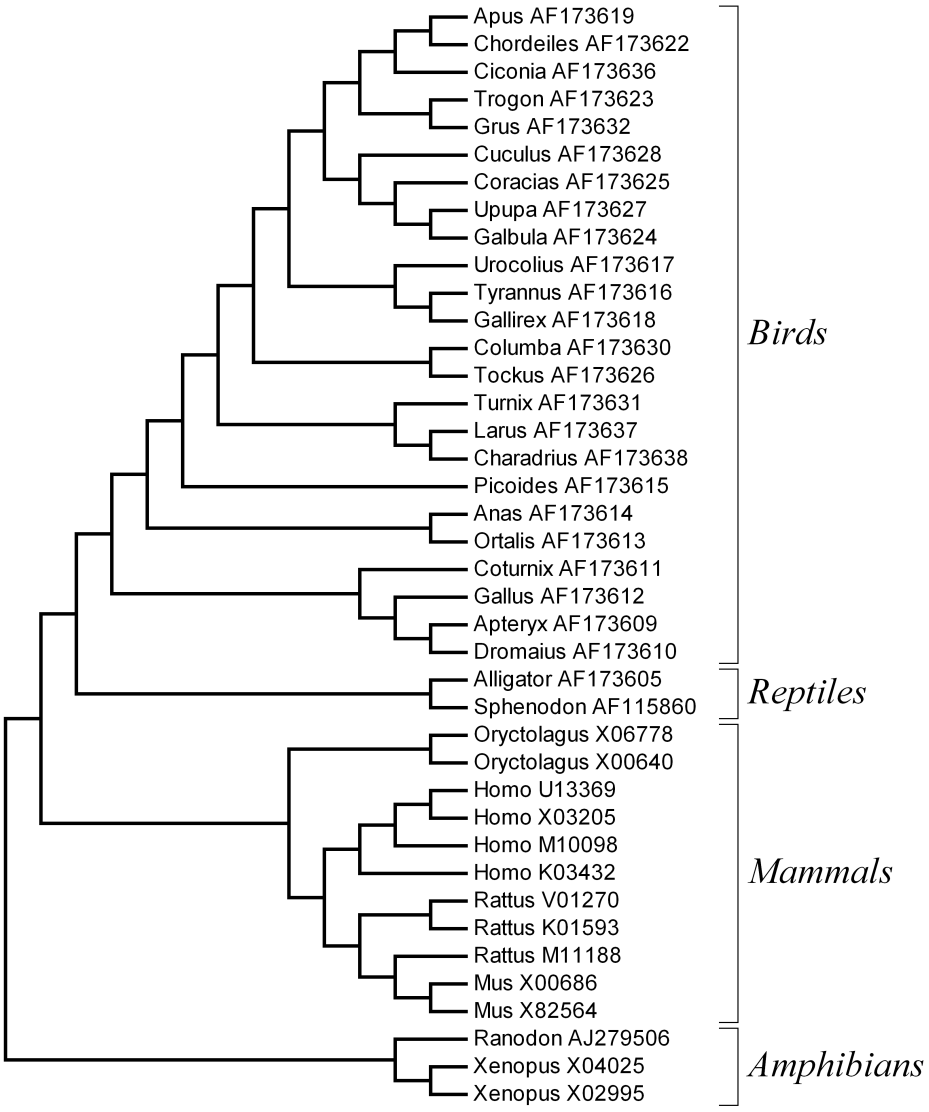

Figure 28.4 The CV tree ($k = 10$) based on the 18S rRNA dataset of tetrapods analyzed by Xia *et al.* [58].

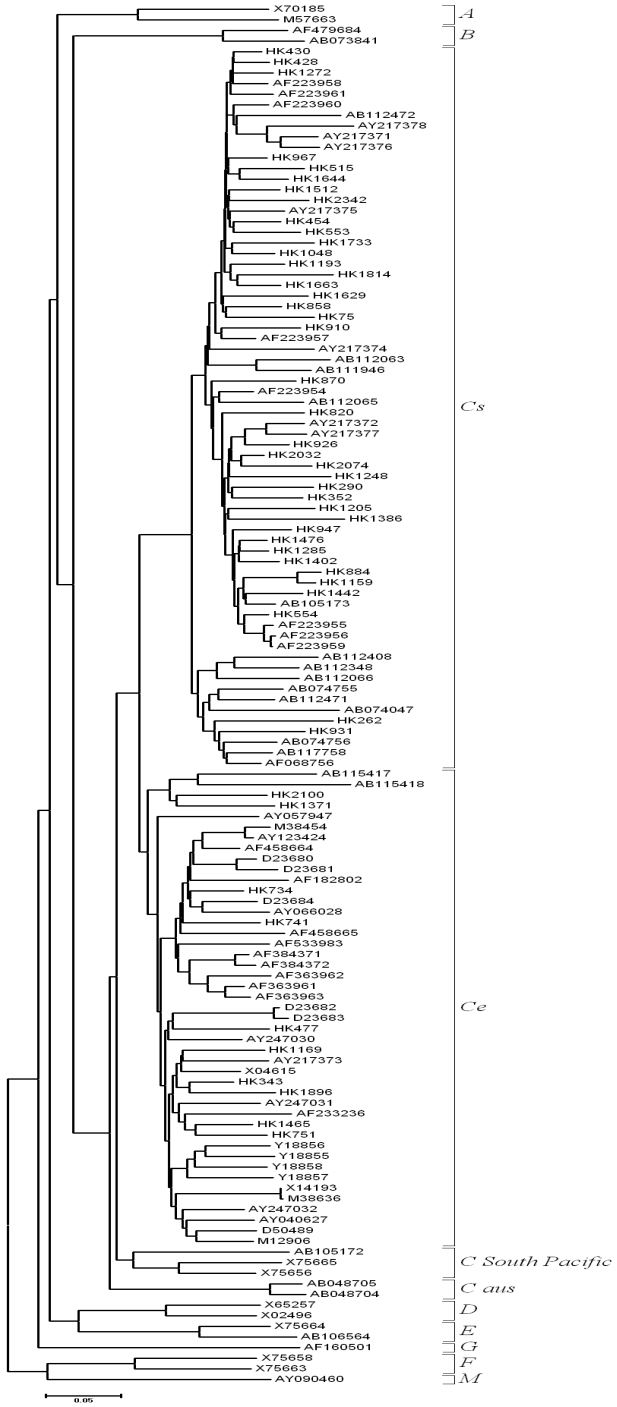

Figure 28.5 The CV tree ($k = 11$) based on the dataset of complete nucleotide sequences of HBV analyzed by Chan *et al.* [7].

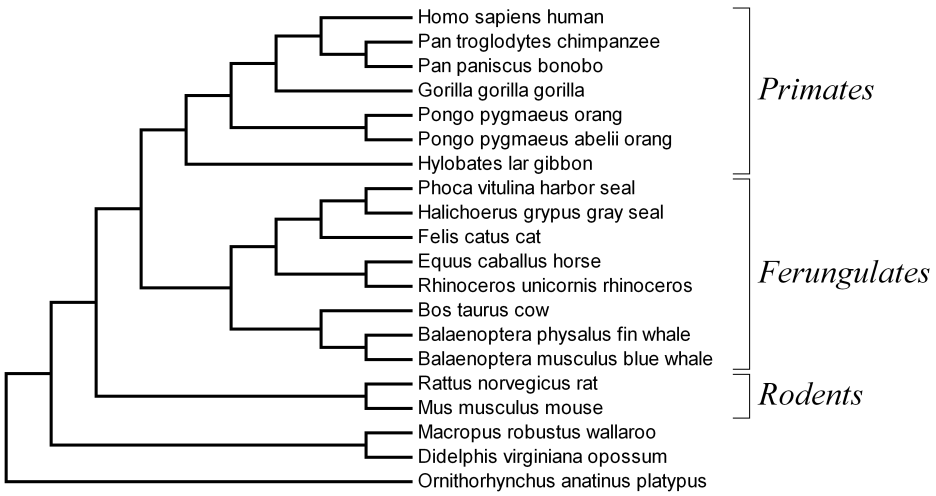

Figure 28.6 The CV tree ($k = 14$) based on the dataset of complete mtDNA sequences of mammal analyzed by Cao *et al.* [5].

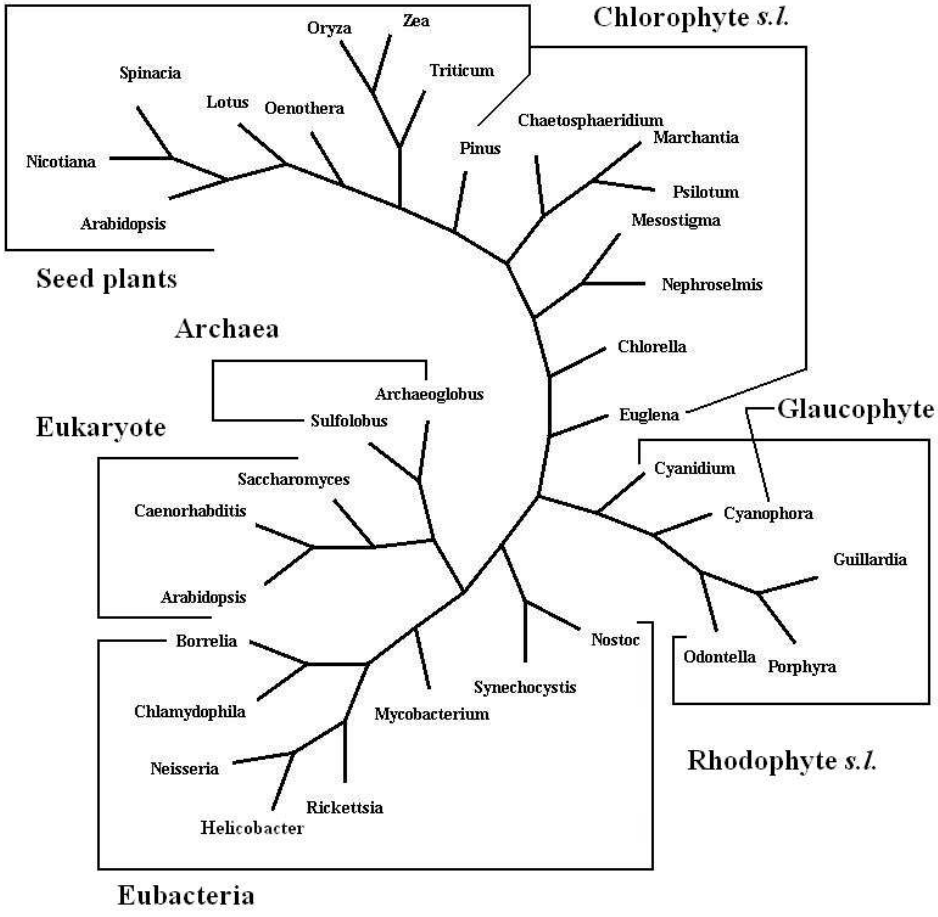

Figure 28.7 The CV tree ($k = 6$) based on the dataset of complete protein genome sequences of choroplast analyzed by Yu *et al.* [60].

28.4 CONCLUDING REMARKS

In this chapter, one kind of alignment-free method, namely, the composition vector (CV) method, was introduced. Compared with the multiple sequence alignment methods which are widely employed, the CV method has several advantages. For instance, it can be used for phylogenetic analysis of whole genome sequences of bacteria, viruses, etc. [12, 42, 60], where the sequence alignment methods all failed. Our denoising formulas worked well in the classification of HBV, mammal and chloroplast. As a systematic method for studying the classification of species, no scoring matrix or gap penalty [56] was required by the CV method. For computing the distance between two species, its operation cost is $\mathcal{O}(N \log N)$ and the memory requirement is $\mathcal{O}(N)$, where N is the length of the longer sequence. With the development of sequence techniques, more and more complete genome sequences are available, and these advantages are becoming more important or even necessary for sequence comparison methods.

The method described in this work has been written in MATLAB for the preparation of input data and in FORTRAN 90 for the rest of the numerical computations. The program is distributed by R. W. WANG and J. C.-F. WONG and can be obtained by anonymous ftp from our ftp size at the following internet website:
http://www.math.cuhk.edu/∼jwong. For example, the current version of the program allows a maximum of 34 library species to be analyzed on 64-byte machines (compilers).

References

1. J. Adachi, P. J. Waddell, W. Martin, and M. Hasegawa, *Plastid genome phylogeny and a model of amino acid substitution for proteins encoded by chloroplast DNA*, Journal of Molecular Evolution, **50** (2000), 348–358.

2. M. W. Berry, Z. Drmac, and E. R. Jessup, *Matrices, vector spaces, and information retrieval*, SIAM Review, **41** (1999), 335–362.

3. B. E. Blaisdell, *A measure of the similarity of sets of sequences not requiring sequence alignment*, Proc. Natl Acad. Sci. **83** (1986), 5155–5159.

4. V. Brendel, J. S. Beckmann, and E. N. Trifonov, *Linguistics of nucleotide sequences: morphology and comparison of vocabularies*, Journal of Biomolecular Structure and Dynamics, **4** (1986), 11–20.

5. Y. Cao, A. Janke, P. J. Waddell, M. Westerman, O. Takenaka, S. Murata, N. Okada, S. Paabo, M. Hasegawa, Conflict among individual mitochondrial proteins in resolving the phylogeny of eutherian orders, , Journal of Molecular Evolution, **47** (1998), 307–322.

6. J. Casey, *A Treatise on Spherical Trigonometry: and Its Application to Geodesy and Astronomy with Numerous Examples*, London: Longmans, Green, and Company (1889).

7. H. L. Y. Chan *et al.*, *Epidemiological and virological characteristics of 2 subgroups of Hepatitis B virus genotype C*, The Journal of Infectious Diseases, **191** (2005), 2022–2032.

8. R. H. Chan, T. H. Chan, and R. W. Wang, *Composition vector method based on maximum entropy principle for sequence comparison*, Research ReportMATH-09-05 (367), Department of Mathematics, The Chinese University of Hong Kong.

9. R. H. Chan, T. H. Chan, and R. W. Wang, *Composition vector method based on maximum entropy principle for sequence comparison*, sumbitted for publication.

10. R. L. Charlebois, R. G. Beiko, and M. A. Ragan, *Microbial phylogenomics: Branching out*, Nature, **421** (2003), 217–217.

11. K. H. Chu, C. P. Li, and J. Qi, *Ribosomal RNA as molecular barcodes: a simple correlation analysis without sequence alignment*, Bioinformatics, **22** (2006), 1690–1710.

12. K. H. Chu, J. Qi, Z. G. Yu, and V. Anh, *Origin and phylogeny of chloroplasts: A simple correlation analysis of complete genomes*, Molecular Biology and Evolution, **21** (2004), 200–206.

13. T. M. Cover, and J. A. Thomas, *Elements of Information Theory.* A Wiley-Interscience Publication (2006).

14. J. De Las Rivas, J. J. Lozano, and A. R. Ortiz, *Comparative analysis of chloroplast genomes: Functional annotation, genome-based phylogeny, and deduced evolutionary patterns*, Genome Research , **12** (2002), 567–583.

15. W. F. Doolittle, *Phylogenetic classification and the universal tree*, Science, **284** (1999), 2124–2128.

16. R. C. Edgar, *MUSCLE: multiple sequence alignment with high accuracy and high throughput*, Nucleic Acids Research, **32(5)** (2004), 1792–1797.

17. J. A. Eisen, and C. M. Fraser, *Phylogenomics: intersection of evolution and genomics*, Science, **300** (2003), 1706–1707.

18. D. F. Feng, and R. F. Doolittle, *Progressive sequence alignment as a prerequisite to correct phylogenetic trees*, Journal of Molecular Evolution, **25** (1987), 351–360.

19. J. Felsenstein, *PHYLIP (phylogeny inference package) version 3.5c.*, distributed by the author at
http://evolution.genetics.washington.edu/phylip.html.

20. W. M. Fitch, and E. Margoliash, *Construction of phylogenetic trees*, Science, **155** (1967), 279 – 284.

21. L. Gao, and J. Qi, *Whole genome molecular phylogeny of large dsDNA viruses using composition vector method*, BMC Evolutionary Biology, **7** (2007), 1–7.

22. L. Gao, J. Qi, H. Wei, Y. Sun, and B. L. Hao, *Molecular phylogeny of coronaviruses including human SARS-CoV*, Chinese Science Bulletin, **48** (2003), 1170–1174.

23. J. F. Gentleman, and R. C. Mullin, *The distribution of the frequency of occurrence of nucleotide subsequences, based on their overlap capability*, Biometrics, **45** (1989), 35–52.

24. M. W. Gray, *The endosymbiont hypothesis revisited*, International Review Of Cytology , **141** (1992), 233–357.

25. M. W. Gray, *Evolution of organellar genomes*, Current Opinion in Genetics & Development, **9**, (1999) 678 – 687.

26. S. Grumbach, and F. Tahi, *Compression of DNA sequences*, Data Compression Conference, IEEE Computer Society Press, Sbowbird, Utah, USA.

27. B. L. Hao, J. Qi, and B. Wang, *Prokaryotic phylogeny based on complete genomes without sequence alignment*, Modern Physics Letters B, **2** (2003), 1–4.

28. R. Hu, and B. Wang, *Statistically significant strings are related to regulatory elements in the promoter regions of Saccharomyces cerevisiae*, Physica A, **290** (2001), 464–474.

29. D. H. Huson, and D. Bryant, *Application of phylogenetic networks in evolutionary studies*, Molecular Biology and Evolution, **23(2)** (2005), 254–267.

30. M. Li, J. H. Badger, X. Chen, S. Kwong, P. Kearney, and H. Zhang, *An information-based sequence distance and its application to hole mitochondrial genome phylogeny*, Bioinformatics, **17** (2001), 149–154.

31. G. Lu, S. Zhang, and X. Fang, *An improved string composition method for sequence comparison*, BMC Bioinformatics, **9 (Suppl 6)** (2008), S15.

32. W. Martin, and R. G. Herrmann, *Gene transfer from organelles to the nucleus: How much, what happens, and why?* Plant Physiology, **118** (1998), 9–17.

33. W. Martin, B. Stoebe, V. Goremykin, S. Hansmann, M. Hasegawa, and K. V. Kowallik, *Gene transfer to the nucleus and the evolution of chloroplasts*, Nature, vol 393,1998, pp. 162-165.

34. W. Martin, , T. Rujan, E. Richly, A. Hansen, S. Cornelsen, T. Lins, D. Leister, B. Stoebe, M. Hasegawa, and D. Penny, *Evolutionary analysis of Arabidopsis, cyanobacterial, and chloroplast genomes reveals plastid phylogeny and thousands of cyanobacterial genes in the nucleus* Proc. Natl. Acad. Sci. U.S.A., **99** (2002), 12246–12251.

35. G. I. McFadden, *Chloroplast origin and integration*, Plant Physiology, **125** (2001b), 50–53.

36. S. P. Meyn, and R. L. Tweedie, *Markov Chains and Stochastic Stability*, London: Springer-Verlag, (1993).

37. D. Moreira, H. LE Guyader, and H. Ppilippe, *The origin of red algae and the evolution of chloroplasts*, Nature, **405** (2000), 69–72.

38. S. B. Needleman, and C. D. Wunsch, *A general method applicable to the search for similarities in the amino acid sequence of two proteins*, Journal of Molecular Biology, **48** (1970), 443–453.

39. C. Notredame, D. G. Higgins, and J. Heringa, *T-Coffee: a novel method for fast and accurate multiple sequence alignment*, Journal of Molecular Biology, **302(1)** (2000), 205–217.

40. J. D. Palmer and C. F. Delwiche, *The origin and evolution of plastids and their genomes*. In Molecular Systematics of Plants II DNA Sequencing (eds. Soltis, D.E., Soltis, P.S. and Doyle, J.J.), (1998), 345–409. Kluwer, London.

41. P. A. Pevzner, *Computational Molecular Biology: an algorithmic approach*. MIT Press, Cambridge, MA (2000), pp. 75.

42. J. Qi, B. Wang, and B. L. Hao, *Whole proteome prokaryote phylogeny without sequence alignment: A k-string composition approach*, Journal of Molecular Evolution, **58(1)** (2004), 1–11.

43. N. Saitou, and M. Nei, *The neighbor-joining method: a new method for reconstructing phylogenetic trees*, Molecular Biology and Evolution, **4(4)** (1987), 406–425.

44. T. F. Smith, and M. S. Waterman, *Identification of common molecular sequences*, Journal of Molecular Biology, **147** (1981), 195–197.

45. B. Snel, P. Bork, and M. A. Huynen, *Genome phylogeny based on gene content*, Nature Genetics, **21** (1999), 108–110.

46. V. L. Stirewalt, C. B. Michalowski, W. Loffelhardt, H. J. Bohnert, and D. A. Bryant, *Nucleotide sequence of the cyanelle genome from Cyanophora paradoxa*, Plant Molecular Biology Reporter, **13** (1995), 327–332.

47. G. W. Stuart, and M. W. Berry, *A comprehensive whole genome bacterial phylogeny using correlated peptide motifs defined in a igh dimensional vector space*, Journal of Bioinformatics and Computational Biology **1(3)** (2003), 475–493.

48. G. W. Stuart, and M. W. Berry, *An SVD-based comparison of nine whole eukaryotic genomes supports a coelomate rather than ecdysozoan lineage*, BMC Bioinformatics **5:204** (2004).

49. G. W. Stuart, K. Moffett, and S. Baker, *Integrated gene and species phylogenies from unaligned whole genome protein sequences*, Bioinformatics **62** (2002), 100–108.

50. G. W. Stuart, K. Moffett, and J. J. Leader, *A comprehensive vertebrate phylogeny using vector representations of protein sequences from whole genomes*, Molecular Biology and Evolution **19** (2002), 554–562.

51. K. Tamura, J. Dudley, M. Nei, and S. Kumar, *MEGA4: molecular evolutionary genetics analysis (MEGA) software version 4.0*, Molecular Biology and Evolution **24** (2007), 1596–1599.

52. J. D. Thompson, D. G. Higgins, and T. J. Gibson, *Clustal W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice*, Nucleic Acids Research, **22(22)** (1994), 4673–4680.

53. UPGMA Software.
http://pubmlst.org/software/analysis/start/manual/upgma.shtml

54. S. Vinga, and J. Almeida, *Alignment free sequence comparison-a review*, Bioinformatics, **19** (2003), 513–523.

55. J. Wang, and X. Zheng, *WSE, a new sequence distance measure based on word frequencies*, Mathematical Biosciences, **215** (2008), 78 –83.

56. M. S. Waterman, *Introduction to Computational Biology: Maps, sequences and genomes*, Chapman and Hall–CRC Press, (1995).

57. X. Wu, X. F. Wan, G. Wu, D. Xu, and G. Lin, *Phylogenetic analysis using complete signature information of whole genomes and clustered Neighbor-Joining method*, International Journal of Bioinformatics Research and Applications, **2** (2006), 219–248.

58. X. Xia , Z. Xie, and K.M. Kjer, *18S ribosomal RNA and tetrapod phylogeny*, Systematic Biology, **52** (2003), 283–295.

59. H. M. Xie, *Grammatical Complexity and One-dimensional Dynamical Systems*, World Scientific. Singapore, (1996).

60. Z. G. Yu, L. Q. Zhou, V. Anh, K. H. Chu, S. C. Long, and J. Q. Deng, *Phylogeny of prokaryotes and chloroplasts revealed by a simple composition approach on all protein sequences from whole genome without sequence alignment*, Journal of Molecular Evolution, **60** (2005), 538–545.