# APPENDIX

# Computational Methods for Numerical Taxonomy

For the benefit of the reader interested in familiarizing himself with the computational details of the various techniques discussed in Chapters 6 and 7, the Appendix shows some of the most common computations in numerical taxonomy. While all computational steps will be illustrated, we cannot, of course, explain the fundamentals of statistical procedure in this section. Readers lacking any statistical background should refer to one of the numerous books on the subject (for example, Simpson, Roe, and Lewontin, 1960, or Steel and Torrie, 1960, for biological statistics). All the necessary formulas have already been given in the previous sections; presented below are small illustrative examples which bring out some of the problems involved in doing the actual computations and which may help those readers who are not too facile with mathematical computations to visualize what is happening. Indeed, since much of the work will in the future be done on computers, the present examples will serve to show what is going on "behind the scenes." In Section A.1 we shall briefly review how characters may be coded and scaled in order to present them in a data matrix. We shall then discuss the computation of various estimates of affinity or similarity in Section A.2. This is followed by a discussion of methods of clustering in Section A.3. In Section A.4 some of the ancillary methods, such as studies of frequency distributions of similarity coefficients and computation of cophenetic correlations, are illustrated, and the final section makes brief mention of the problems of computer programming for numerical taxonomy.

## A.1. Choice and coding of characters

Detailed techniques for coding characters have already been discussed in Sections 5.2 and 5.3 and need not be repeated here. Table A-1 shows an $n \times t$

**Table A-1.** *Original Data Matrix*

| Characters | OTU's | | | | | | Mean | S.D. |
|---|---|---|---|---|---|---|---|---|
| | **A** | **B** | **C** | **D** | **E** | **F** | | |
| 1 | 1 | 8 | 1 | 7 | 2 | 5 | 4.000 | 3.098 |
| 2 | 1 | 6 | 1 | 6 | 1 | 3 | 3.000 | 2.449 |
| 3 | 6 | 1 | 5 | 1 | 4 | 2 | 3.167 | 2.137 |
| 4 | 1 | 0 | 1 | 0 | 1 | 4 | 1.167 | 1.472 |
| 5 | NC | 6 | 3 | 6 | NC | 1 | 4.000 | 2.449 |
| 6 | NC | 2 | NC | 3 | 1 | 1 | 1.750 | 0.957 |
| 7 | 8 | 2 | 7 | 2 | 5 | 5 | 4.833 | 2.483 |
| 8 | 1 | 6 | 1 | 6 | 3 | 4 | 3.500 | 2.258 |
| 9 | 1 | 8 | 1 | 8 | 2 | 4 | 4.000 | 3.286 |
| 10 | 6 | 1 | 6 | 1 | 5 | 2 | 3.500 | 2.429 |
| 11 | 3 | 3 | 3 | 3 | 3 | 3 | 3.000 | 0.000 |

data matrix for six OTU's labeled **A** through **F** and eleven characters numbered one through eleven.

It should be pointed out immediately that the example here is entirely for illustrative purposes. A numerical taxonomic study should not be based on eleven characters alone. However, so far as computational details are concerned, the smaller the example the easier it will be to present. Also, it is not customary to label OTU's by letter in computational work. In larger studies, and especially when digital computers are employed, referring to the OTU's by number is essential. The characters considered in this entirely hypothetical example are multistate characters varying in their range from three states for character 6 to eight states for characters 1 and 9. Most of the characters presented here have been coded with "1" as the lowest class. However, since the characters will eventually be standardized, the numerical value of the lowest character state is immaterial. All sets of characters can have a constant subtracted from them without changing their standard deviation or their standardized character state codes. Thus, for example, character 7 has the character state codes 8, 2, 7, 2, 5, 5 which could be recoded to read 7, 1, 6, 1, 4, 4 by subtraction of 1. The standard deviation of these characters would be the same regardless of whether the 1 had been subtracted or not; hence the standardization would give identical results. All possible character states may be shown in a data matrix, as for example the three-state character 6, where states 1, 2, and 3 all occur. On the other hand, some of the intermediate states may not exist in a given data matrix, as is the case with most of the other characters in the matrix of Table A-1. This could be so for three reasons. (1) The character state codes may actually refer to counts of certain structures, such as number of

bristles or segments in an insect, number of leaflets in a plant, or number of tentacles in an animal possessing these, or they may refer to the concentration of a certain substance in an arithmetic scale or in a logarithmic scale when such seems appropriate. Thus, for example, character 4 may represent the characteristic of the logarithm of the concentration of a given substance $X$ in the blood of the organisms with which we are working. In such a scheme, character state code 0 would represent that the substance is undetectable—for example, less than 10 micrograms per unit volume; character state code 1 would then represent up to 100 micrograms per unit volume, and character state code 4 would represent from 10,000 to 100,000 micrograms per unit volume. (2) A second reason why all character states are not present may be that we are working only with a section of a larger study. However, in such a case it may be argued that the standardization employed should be based on all the characters rather than only on the sample used here.

(3) A third reason could be that the taxonomist working on the group may know from his experience and comparative knowledge of other groups that there are in fact intermediate states between the extremes that he is examining at the moment. Thus, for example, in character 1 the states are coded from 1 to 8, yet character state codes 3, 4, and 6 are missing. It may well be that the taxonomist knows that according to his scheme of coding such character state codes occur in other OTU's but not in the ones in the present study. Therefore he may feel that to code as shown here is more appropriate than to use a coding scheme limited entirely to the character states in the OTU's he is at present studying.

Two further points of interest in the original data matrix should be noted. Character 11 has the identical character state code for all the OTU's in the example. This is therefore a character that is not admissible in a numerical taxonomic study (see Section 5.3.3.4). It should not have been included at all. Its presence is immediately obvious in this small study; however, in a large study invariant characters may not be noticed during the preparation of the data matrix (this may quite easily happen if the data are processed by automatic data-processing equipment). In such a case the next step, standardization of characters, would automatically remove character 11 from consideration. This is so because its standard deviation is zero, and it will be impossible to compute standardized character state codes for this character since such a procedure would necessitate division by zero.

Second, characters 5 and 6 show the letters NC in place of character state codes for certain OTU's. NC stands for "no comparison." Such a score is indicated when data are missing because of damaged specimens or when characters or organs are missing and some property of these can therefore not be measured or evaluated. The wing veins in wingless insects or some chemical attribute of chlorophyll in plants that do not possess chlorophyll would be cases in point. This whole subject has been discussed in detail in Section 6.5, "Un-

warranted Comparisons." No comparison or calculation is made regarding the resemblance between two OTU's for any particular character when one or both of the OTU's involved are scored NC. Thus in comparing **A** and **B**, two NC's are involved—one for character 5, the other for character 6—and the total number of valid comparisons is only 8 (remember that character 11 is omitted from the study altogether). The OTU's **C** and **E** also have eight valid comparisons. OTU's **B** and **D** on the other hand, can be compared in all ten characters, while **E** and **F** can be compared in 9 characters. It should be noted that the number of NC's should not be excessive, to ensure the relevance of a comparison (the percentage of valid comparisons over the total number of characters involved in a study) remaining fairly high. In the present study the lowest relevance is between **A** and the other OTU's and between **C** and **E**, where eight out of ten possible comparisons can be made, or a relevance of 0.8. The other comparisons in the present example have relevances of 0.9 or 1.0.

Some hesitation may be felt on how to score OTU's which seem to be intermediate between two arbitrarily assigned character state codes. Thus, having erected a scale, say, from 1 to 6, one may find a specimen which seems to be more or less intermediate between 4 and 5. How should it be scored? Twice in published criticisms of numerical taxonomy this issue has been raised as an apparent example of a lack of objectivity in numerical taxonomy equal to that of classical systematics. This is not so! When a specimen is truly intermediate, so that it should fall on the borderline between 4 and 5, arbitrary assignment to one or the other of these groups does not result in a large degree of distortion of the similarity coefficients computed from such data. This is a well-accepted and proven principle of statistics, and only through ignorance or misunderstanding of statistical procedures would this issue have been raised in the first place. Furthermore, if a digital computer is employed, where character state codes do not have to be a fixed number of digits, it is perfectly acceptable to record the character as 4.5 in the best judgment of the taxonomist and have the machine compute a more exact value of the similarity coefficient. However, a little experimenting with numbers along this line will convince anyone that the differences are very minor indeed between the similarity coefficients obtained by such a procedure and those based on characters arbitrarily grouped into one of the two neighboring states.

The next step in processing the data for further computation of correlation or distance coefficients is the standardization of the characters, discussed in Section 6.2.2. To do this we calculate the mean and standard deviation of each character—that is, of each row of the data matrix. Before we proceed to do this we should review the symbolism which we have adopted for a data matrix. This is shown in Table A-2, where the entries are of the form $X_{ij}$, which stands for the character state value of character $i$ in the OTU $j$. There are $t$ OTU's and $n$ characters in a study. In the example of Table A-1, $t = 6$ and $n$ (after the elimination of the invariant character) $= 10$.

**Table A-2.** *Symbolism of a Data Matrix*

| | OTU's | | | | |
|---|---|---|---|---|---|
| Characters | 1 | 2 | 3 | ··· | $t$ |
| 1 | $X_{1,1}$ | $X_{1,2}$ | $X_{1,3}$ | ··· | $X_{1,t}$ |
| 2 | $X_{2,1}$ | $X_{2,2}$ | $X_{2,3}$ | ··· | $X_{2,t}$ |
| 3 | $X_{3,1}$ | $X_{3,2}$ | $X_{3,3}$ | ··· | $X_{3,t}$ |
| · | · | · | · | | · |
| · | · | · | · | | · |
| · | · | · | · | | · |
| $n$ | $X_{n,1}$ | $X_{n,2}$ | $X_{n,3}$ | ··· | $X_{n,t}$ |

The first step in standardizing the characters is the computation of the mean and standard deviation of each character of the data matrix as shown in Table A-1. The mean of character 1, given in the next to last column of Table A-1, is calculated as

$$\frac{1}{t}\sum_{j=1}^{t} X_{1j} = \bar{X}_1.$$

This formula signifies the addition of all $t$ scores for the first row, the sum being divided by $t$. Numerically, it is

$$\frac{1+8+1+7+2+5}{6} = \frac{24}{6} = 4.00.$$

The standard deviation is calculated as follows:

$$s_1 = \left\{\frac{1}{t-1}\left[\sum_{j=1}^{t} X_{1j}^2 - \frac{\left(\sum_{j=1}^{t} X_{1j}\right)^2}{t}\right]\right\}^{1/2},$$

where $\sum_{j=1}^{t} X_{1j}^2$ represents every score squared and summed; for example, in the first row we would have $1^2 + 8^2 + 1^2 + 7^2 + 2^2 + 5^2 = 144$. From this we subtract the so-called correction term, which is the square of the sum of the scores (found above for the computation of the mean) divided by $t$. This results in the "sum of squares," which is then divided by $t - 1$ to yield the variance, the square root of which is the standard deviation. Hence the standard deviation for character 1 equals

$$\left\{\frac{1}{5}\left(144 - \frac{(24)^2}{6}\right)\right\}^{1/2} = 3.098.$$

In cases where NC's occur in a character, the summation is, of course, not over $t$ scores, but over $t$ scores minus the number of NC's. This applies both to the mean and to the standard deviation. Similarly, division in such a case is by $t$

minus the number of NC's for the mean and by $t$ − (number of NC's + 1) for the variance.

Standardized character states are computed as

$$X'_{ij} = \frac{X_{ij} - \overline{X}_{i\cdot}}{s_i},$$

where $X'_{ij}$ is the standardized character state code for character $i$ and OTU $j$, while $X_{ij}$ is the raw score for this character state, and $\overline{X}_{i\cdot}$ and $s_i$ are the mean and standard deviation of character $i$, respectively. Thus, for instance, the first character state for OTU 1 is a "1". Since the mean and standard deviation for character 1 are 4.000 and 3.098, respectively, the first standardized character state code is

$$\frac{1.000 - 4.000}{3.098} = -0.97.$$

The mean and standard deviation of standardized character state codes are 0 and 1, respectively. Table A-3 lists the data matrix with all characters standardized. Since the mean is 0, it is obvious that some standardized character state codes will be negative. When calculations are carried out on a computer, negative values and positive values are handled with equal facility. However, for desk calculator operations negative values are to be avoided as much as possible. Therefore, since the present example was to be processed on a desk calculator, the character state codes were transformed into positive values by addition of a constant, 5.00. These transformed character state codes are shown in Table A-4. For the purposes of computing correlation coefficients or distance coefficients the addition of this constant is of no consequence and will automatically cancel itself.

**Table A-3.** *Data Matrix with Characters Standardized*

| | OTU's | | | | | |
|---|---|---|---|---|---|---|
| Characters | **A** | **B** | **C** | **D** | **E** | **F** |
| 1 | −.97 | 1.29 | −.97 | .97 | −.65 | .32 |
| 2 | −.82 | 1.22 | −.82 | 1.22 | −.82 | .00 |
| 3 | 1.33 | −1.01 | .86 | −1.01 | .39 | −.55 |
| 4 | −.11 | −.79 | −.11 | −.79 | −.11 | 1.92 |
| 5 | NC | .82 | −.41 | .82 | NC | −1.22 |
| 6 | NC | .26 | NC | 1.31 | −.78 | −.78 |
| 7 | 1.28 | −1.14 | .87 | −1.14 | .07 | .07 |
| 8 | −1.11 | 1.11 | −1.11 | 1.11 | −.22 | .22 |
| 9 | −.91 | 1.22 | −.91 | 1.22 | −.61 | .00 |
| 10 | 1.03 | −1.03 | 1.03 | −1.03 | .62 | −.62 |

**Table A-4.** *Data Matrix with Characters Standardized (5.00 added to remove negative numbers)*

| Characters | OTU's | | | | | |
|---|---|---|---|---|---|---|
| | **A** | **B** | **C** | **D** | **E** | **F** |
| 1 | 4.03 | 6.29 | 4.03 | 5.97 | 4.35 | 5.32 |
| 2 | 4.18 | 6.22 | 4.18 | 6.22 | 4.18 | 5.00 |
| 3 | 6.33 | 3.99 | 5.86 | 3.99 | 5.39 | 4.45 |
| 4 | 4.89 | 4.21 | 4.89 | 4.21 | 4.89 | 6.92 |
| 5 | NC | 5.82 | 4.59 | 5.82 | NC | 3.78 |
| 6 | NC | 5.26 | NC | 6.31 | 4.22 | 4.22 |
| 7 | 6.28 | 3.86 | 5.87 | 3.86 | 5.07 | 5.07 |
| 8 | 3.89 | 6.11 | 3.89 | 6.11 | 4.78 | 5.22 |
| 9 | 4.09 | 6.22 | 4.09 | 6.22 | 4.39 | 5.00 |
| 10 | 6.03 | 3.97 | 6.03 | 3.97 | 5.62 | 4.38 |

## A.2. The computation of coefficients of resemblance

We shall first discuss the computation of correlation and distance coefficients and subsequently proceed to a discussion of the calculation of association coefficients. We do this because the first two coefficients are usually based on standardized multistate characters, as discussed in the previous section, while association coefficients are somewhat different in nature.

*Correlation coefficients* are conveniently computed by the computational formula

$$r_{jk} = \frac{\sum_{i=1}^{n} X_{ij}X_{ik} - \frac{1}{n}\left(\sum_{i=1}^{n} X_{ij}\right)\left(\sum_{i=1}^{n} X_{ik}\right)}{\left\{\left[\sum_{i=1}^{n} X_{ij}^2 - \frac{1}{n}\left(\sum_{i=1}^{n} X_{ij}\right)^2\right]\left[\sum_{i=1}^{n} X_{ik}^2 - \frac{1}{n}\left(\sum_{i=1}^{n} X_{ik}\right)^2\right]\right\}^{1/2}},$$

which is exactly equal to the formula

$$r_{jk} = \frac{\sum_{i=1}^{n} (X_{ij} - \bar{X}_j)(X_{ik} - \bar{X}_k)}{\left\{\sum_{i=1}^{n} (X_{ij} - \bar{X}_j)^2 \sum_{i=1}^{n} (X_{ik} - \bar{X}_k)^2\right\}^{1/2}}$$

given in Section 6.2.2 for the Pearson product-moment correlation coefficient. By way of illustration we shall calculate below the correlation between OTU's **B** and **C** by the computational formula. The formula may be broken down into the following steps. We need to find the sum of the character state codes for OTU $j$ and similarly for OTU $k$, or quantities $\sum_{i=1}^{n} X_{ij}$ and $\sum_{i=1}^{n} X_{ik}$. We need then to find the sum of the squares of these character state codes ($\sum_{i=1}^{n} X_{ij}^2$ and $\sum_{i=1}^{n} X_{ik}^2$) and finally the sum of the products of the character state codes of the

two OTU'S ($\sum_{i=1}^{n} X_{ij}X_{ik}$). These terms and $n$ are all that we need to evaluate the correlation coefficient. Table A-5 shows the computational steps for finding the correlation between OTU's **B** and **C**. The standardized character state codes for **B** are shown followed by their squares. In the next column are the standardized character state codes for **C** followed by their squares, and in the final column we find the products between **B** and **C**. In an actual computation on a desk calculator, individual items and their squares would not be shown; they would be accumulated as running totals. Most modern desk calculators provide facilities for computing the five quantities simultaneously; however, when these are computed simultaneously it may not be possible to carry as many decimal places as shown in this example. It should be clear that when one (or both) of the OTU's has an NC code for one of the characters, as in the present case, the summation is carried out only for those characters recorded in both OTU's. Thus $n$ for this example necessarily reduces to 9. The remaining computational

**Table A-5.** *Computational Steps for Obtaining Correlation Coefficients*

| OTU's **B** and **C** | | | | |
|---|---|---|---|---|
| **B** | $\mathbf{B}^2$ | **C** | $\mathbf{C}^2$ | **BC** |
| 6.29 | 39.5641 | 4.03 | 16.2409 | 25.3487 |
| 6.22 | 38.6884 | 4.18 | 17.4724 | 25.9996 |
| 3.99 | 15.9201 | 5.86 | 34.3396 | 23.3814 |
| 4.21 | 17.7241 | 4.89 | 23.9121 | 20.5869 |
| 5.82 | 33.8724 | 4.59 | 21.0681 | 26.7138 |
| (5.26) | — | NC | — | — |
| 3.86 | 14.8996 | 5.87 | 34.4569 | 22.6582 |
| 6.11 | 37.3321 | 3.89 | 15.1321 | 23.7679 |
| 6.22 | 38.6884 | 4.09 | 16.7281 | 25.4398 |
| 3.97 | 15.7609 | 6.03 | 36.3609 | 23.9391 |
| 46.69 | 252.4501 | 43.43 | 215.7111 | 217.8354 |

$$r_{BC} = \frac{217.8354 - \dfrac{(46.69)(43.43)}{9}}{\left\{\left[252.4501 - \dfrac{(46.69)^2}{9}\right]\left[215.7111 - \dfrac{(43.43)^2}{9}\right]\right\}^{1/2}}$$

$$= \frac{217.8354 - 225.3052}{[(252.4501 - 242.2173)(215.7111 - 209.5739)]^{1/2}}$$

$$= \frac{-7.4698}{[(10.2328)(6.1372)]^{1/2}}$$

$$= \frac{-7.4698}{7.9247} = -0.943$$

steps are quite simple, according to the formula, and are shown in the lower half of Table A-5.

In a similar manner the other correlations between all pairs of OTU's are computed. The coefficients are then conventionally placed into a matrix, usually a half matrix with the correlation coefficients placed below the principal diagonal of the matrix; Table A-6 shows the arrangement for all pairs of OTU's in our illustrative example.

**Table A-6.** *Product-moment Correlation Coefficients Between All Pairs of OTU's Computed by the Method of Table A-5*

| | | OTU's | | | | | |
|---|---|---|---|---|---|---|---|
| | | **A** | **B** | **C** | **D** | **E** | **F** |
| | **A** | *X* | | | | | |
| | **B** | −.934 | *X* | | | | |
| OTU's | **C** | .990 | −.943 | *X* | | | |
| | **D** | −.933 | .946 | −.939 | *X* | | |
| | **E** | .854 | −.819 | .882 | −.884 | *X* | |
| | **F** | −.338 | −.100 | −.213 | −.201 | −.095 | *X* |

When the OTU's are based on the same number of characters (where there are no NC codes in a data matrix), the sums of squares (the expressions under the square root in the denominator of the correlation coefficient) remain constant for each column of the correlation matrix. Similarly, the sums of $X$'s employed in the numerator of the correlation coefficient are constant for each column of the data. Desk calculator operators should then set up the flow of the computation in such a way as to take advantage of this fact. Thus on one sheet is assembled a matrix of the products of two variables (we shall call it matrix 1). In another matrix (2) on a second sheet, the sums of $X$'s divided by $\sqrt{n}$ are placed around the margin (both at the left of the rows and at the heads of the columns), and the appropriate product of the marginal terms is subtracted from the product of the two variables in matrix 1, the difference being entered in matrix 2. Thus matrix 2 represents the numerator of the correlation term. The denominator is computed as the product of the square roots of the sums of squares (again conveniently entered at the margins of a third sheet) and entered into matrix 3. Finally, in a fourth matrix are entered the quotients of the corresponding terms of matrices 2 and 3; that is, the numerator of the correlation coefficient expression divided by its denominator. Thus matrix 4 is the correlation coefficient matrix.

Readers wishing to practice the computations in this example are cautioned against expecting their results to agree to the last decimal place with those presented here. These data have been computed carrying four decimal places

**Table A-7.** *Product-moment Correlation Coefficients Between All Pairs of OTU's Computed with the Sums of the Character State Codes and Their Squares and Products Rounded to One Decimal Place Only*

| | | OTU's | | | | | |
|---|---|---|---|---|---|---|---|
| | | **A** | **B** | **C** | **D** | **E** | **F** |
| | **A** | *X* | | | | | |
| | **B** | −.933 | *X* | | | | |
| OTU's | **C** | .990 | −.941 | *X* | | | |
| | **D** | −.933 | .946 | −.938 | *X* | | |
| | **E** | .855 | −.818 | .882 | −.884 | *X* | |
| | **F** | −.338 | −.101 | −.213 | −.202 | −.094 | *X* |

throughout, and if the data are treated in this manner the results should be very close to ours. However, the number of decimal places carried and rounding off at various stages during the computational cycle will affect the final values of the correlation coefficients. Thus, rounding off to one decimal place during the calculations produced the matrix shown in Table A-7. As can be seen by comparing this table with Table A-6, the differences are not appreciable. All other considerations being equal, it is, of course, preferable to keep as many places during the computation as possible and to round off figures after the correlation coefficients have been computed. Those persons using computers should also be aware that the output of a machine will vary depending on the precision with which the computations are carried on inside the machine. As an example we show Table A-8, which reproduces the output of a computation carried out on the IBM 650 digital computer, using floating decimal points and eight significant figures for all computations, including the initial computation of the standardized character state codes. Assuming that the data in Table A-8

**Table A-8.** *Copy of Output of Computer Program (COR-DIST on IBM 650 Computer) Showing the Correlation Coefficients Between All Pairs of OTU's Computed on the Basis of Eight Significant Figures*

| | | OTU's | | | | | |
|---|---|---|---|---|---|---|---|
| | | **A** | **B** | **C** | **D** | **E** | **F** |
| | **A** | *X* | | | | | |
| | **B** | −.93391829 | *X* | | | | |
| OTU's | **C** | .99014559 | −.94198138 | *X* | | | |
| | **D** | −.93274176 | .94611327 | −.93848123 | *X* | | |
| | **E** | .85499380 | −.81824288 | .88234334 | −.88400459 | *X* | |
| | **F** | −.33844704 | −.10110804 | −.21292392 | −.20196453 | −.093913755 | *X* |

are closest to the true but unobtainable values based on an infinitely precise arithmetic, we note that the figures in Table A-6 based on four decimal places are not always closer to the figures of Table A-8 than are those of Table A-7, based on one decimal place. However, none of the matrices is appreciably different from the others.

We take up next the computation of *taxonomic distance*, computed as $d_{jk}$, which is the average distance in its square root form, as described in Section 6.2.3.2. The formula used for this computation is

$$d_{jk} = \left( \frac{\sum_{i=1}^{n} X_{ij}^2 + \sum_{i=1}^{n} X_{ik}^2 - 2 \sum_{i=1}^{n} X_{ij} X_{ik}}{n} \right)^{1/2},$$

which is the convenient computational formula for the mathematically identical expression

$$d_{jk} = \left[ \frac{\sum_{i=1}^{n} (X_{ij} - X_{ik})^2}{n} \right]^{1/2} = \left( \frac{\Delta_{jk}^2}{n} \right)^{1/2},$$

which was shown in Section 6.2.3.2. It will be noted that the expressions in the computational formula are the very same ones employed in the computation of the correlation coefficient. It is therefore quite convenient to compute correlations and distances simultaneously. Once the computationally tedious portions of the operations have been completed, the final computation of both correlations and distances is simple and probably worthwhile. The computer program employed in the laboratory at the University of Kansas carries out the computation of correlations and distances simultaneously. Using the intermediate results obtained in Table A-5, we can compute the average distance between OTU's **B** and **C** as follows:

$$d_{BC} = \left( \frac{252.4501 + 215.7111 - 2(217.8354)}{9} \right)^{1/2} = (3.6100)^{1/2} = 1.900.$$

The average distances and their squares computed in this manner (again carrying four decimal places during our computations) are shown in Table A-9. Readers will remember that distances indicate the opposite of correlation coefficients; thus the greater the distance between any two OTU's, the less their phenetic relationship.

As an example of the computation of an *association coefficient* we shall present the simplest of these, the simple matching coefficient of Sokal and Michener. It will be remembered that association coefficients are based on character states coded "zero" and "one" only. These could be attributes, "zero" standing for absence and "one" for presence, or these two states may represent two attributes which cannot be linearly ordered, such as black and red. However, two-state coding can also be applied to linearly ordered measurements which are arbitrarily divided into two classes. This is what we have done here (Table

**Table A-9.** *Average Distance Coefficients Between All Pairs of OTU's**

| | | OTU's | | | | | |
|---|---|---|---|---|---|---|---|
| | | **A** | **B** | **C** | **D** | **E** | **F** |
| OTU's | **A** | *X* | 4.347 | 0.049 | 4.178 | 0.437 | 2.097 |
| | **B** | 2.085 | *X* | 3.610 | 0.120 | 2.304 | 1.915 |
| | **C** | 0.221 | 1.900 | *X* | 3.460 | 0.252 | 1.674 |
| | **D** | 2.044 | 0.347 | 1.860 | *X* | 2.541 | 2.190 |
| | **E** | 0.661 | 1.518 | 0.502 | 1.594 | *X* | 0.968 |
| | **F** | 1.448 | 1.384 | 1.294 | 1.480 | 0.984 | *X* |

* $d_{ik}^2$ above principal diagonal, $d_{jk}$ below.

A-10) by way of an example. We have taken the standardized data matrix from Table A-3 and have coded them in such a way that all scores equal to or less than zero have been given a "zero" score. All scores larger than zero have been given a "one" score. Such a division does not always have to be made at a given fixed point, however, but can lie at any place between the extremes of the range of the character states. To illustrate this, character 4 has been coded "zero" only for those standardized character states less than $-0.11$ and has been coded "one" for $-0.11$ and all higher states. The main reason for converting multistate characters into two-state characters in the present example is an illustrative one, so that data already obtained could be proc-

**Table A-10.** *Standardized Character State Codes of Table A-3 Converted into Two-state Characters by Calling All Scores $\leqq 0$ Equal to "Zero" and All Scores $> 0$ Equal to "1"* *

| | OTU's | | | | | |
|---|---|---|---|---|---|---|
| Characters | **A** | **B** | **C** | **D** | **E** | **F** |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 2 | 0 | 1 | 0 | 1 | 0 | 0 |
| 3 | 1 | 0 | 1 | 0 | 1 | 0 |
| 4 | 1 | 0 | 1 | 0 | 1 | 1 |
| 5 | NC | 1 | 0 | 1 | NC | 0 |
| 6 | NC | 1 | NC | 1 | 0 | 0 |
| 7 | 1 | 0 | 1 | 0 | 1 | 1 |
| 8 | 0 | 1 | 0 | 1 | 0 | 1 |
| 9 | 0 | 1 | 0 | 1 | 0 | 0 |
| 10 | 1 | 0 | 1 | 0 | 1 | 0 |

* Character 4 was coded "zero for scores $< -.11$ and "1" for scores $\geqq -.11$

essed. Ordinarily one would not wish to carry out such coding, since it would lose information. If association coefficients are computed by preference, a method of converting multistate characters into two-state characters should be used, so as not to lose information, as suggested in Section 5.3.6.

The more common reason for employing two-state characters is that only two states are available for study, often without any logical linear order. Such characters are frequently employed in microbiology, and the hypothetical data treated below may be thought of in this framework.

The data of Table A-3, recoded into two-state characters as indicated above, are presented in Table A-10. Character state codes for which no comparison is possible are again labeled NC. Reference to Section 6.2.1 will show the arrangement of data for computation of a coefficient of association.

| | | OTU $j$ + | OTU $j$ − | |
|---|---|---|---|---|
| OTU $k$ | + | $n_{JK}$ | $n_{jK}$ | $n_K$ |
| | − | $n_{Jk}$ | $n_{jk}$ | $n_k$ |
| | | $n_J$ | $n_j$ | $n$ |

The data consist of $n$ scores for two OTU's labeled $j$ and $k$. They are subdivided

**Table A-11.** *The Data of Table A-10 Arranged in 2 × 2 Tables for Computation of Coefficients of Association Between All Pairs of OTU's*

| | | OTU's | | | | | |
|---|---|---|---|---|---|---|---|
| | | **A** | **B** | **C** | **D** | **E** | **F** |
| | **A** | *X* | | | | | |
| | **B** | 0 4<br>4 0 | *X* | | | | |
| | **C** | 4 0<br>0 4 | 0 5<br>4 0 | *X* | | | |
| OTU's | **D** | 0 4<br>4 0 | 6 0<br>0 4 | 0 4<br>5 0 | *X* | | |
| | **E** | 4 0<br>0 4 | 0 5<br>4 0 | 4 0<br>0 4 | 0 5<br>4 0 | *X* | |
| | **F** | 2 2<br>2 2 | 2 4<br>2 2 | 2 2<br>2 3 | 2 4<br>2 2 | 2 2<br>2 3 | *X* |

**Table A-12.** *Necessary Quantities for Computation of Coefficients of Association Between All Pairs of OTU's**

| | OTU Pairs | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | AB | AC | AD | AE | AF | BC | BD | BE | BF | CD | CE | CF | DE | DF | EF |
| $n_{JK}$ | 0 | 4 | 0 | 4 | 2 | 0 | 6 | 0 | 2 | 0 | 4 | 2 | 0 | 2 | 2 |
| $n_{Jk}$ | 4 | 0 | 4 | 0 | 2 | 4 | 0 | 4 | 2 | 5 | 0 | 2 | 4 | 2 | 2 |
| $n_{jK}$ | 4 | 0 | 4 | 0 | 2 | 5 | 0 | 5 | 4 | 4 | 0 | 2 | 5 | 4 | 2 |
| $n_{jk}$ | 0 | 4 | 0 | 4 | 2 | 0 | 4 | 0 | 2 | 0 | 4 | 3 | 0 | 2 | 3 |
| $n_J$ | 4 | 4 | 4 | 4 | 4 | 4 | 6 | 4 | 4 | 5 | 4 | 4 | 4 | 4 | 4 |
| $n_j$ | 4 | 4 | 4 | 4 | 4 | 5 | 4 | 5 | 6 | 4 | 4 | 5 | 5 | 6 | 5 |
| $n_K$ | 4 | 4 | 4 | 4 | 4 | 5 | 6 | 5 | 6 | 4 | 4 | 4 | 5 | 6 | 4 |
| $n_k$ | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 5 | 4 | 5 | 4 | 4 | 5 |
| $m$ | 0 | 8 | 0 | 8 | 4 | 0 | 10 | 0 | 4 | 0 | 8 | 5 | 0 | 4 | 5 |
| $u$ | 8 | 0 | 8 | 0 | 4 | 9 | 0 | 9 | 6 | 9 | 0 | 4 | 9 | 6 | 4 |
| $n$ | 8 | 8 | 8 | 8 | 8 | 9 | 10 | 9 | 10 | 9 | 8 | 9 | 9 | 10 | 9 |

* Data taken from Table A-11.

into positive and negative classes for each of the two operational taxonomic units. Capital subscripts indicate positive or "1" states and lower-case subscripts show negative or "0" states. We designate as $n_{JK}$ the number of characters in which both OTU's are positive and $n_{jk}$ as the number in which both are negative. The number of characters positive for one OTU and negative for the other are $n_{Jk}$ and $n_{jK}$, respectively. Marginal totals are $n_J$ and $n_K$ for positive characters of OTU $j$ and $k$, respectively; similarly, $n_j$ and $n_k$ are the marginal totals for the negative characters. The following symbolism was established for convenience in writing formulas:

$m = n_{JK} + n_{jk}$, the number of characters in "matched" cells;
$u = n_{Jk} + n_{jK}$, the number of characters in "unmatched" cells;
$n = m + u$.

Table A-11 shows the data of the two-state characters in Table A-10 arranged as $2 \times 2$ tables for the computation of coefficients of association of pairs of OTU's. The arrangement of the frequencies in these tables is as in the schematic example shown above. Marginal totals are not shown because of the simplicity of the tables, but in any real analysis these would probably be necessary in order to complete the computations. From Table A-11 it is obvious which pairs of OTU's show special association and which do not. Since the formula for the simple matching coefficient is $S_{SM} = m/n$, we can see from Table A-12 that, for example, the simple matching coefficient between OTU's **A** and **B** is zero.

**Table A-13.** *Coefficients of Association Listed in Tables 6-1 and 6-2 Computed from the Quantities in Table A-12*

| Coefficient | AB | AC | AD | AE | AF | BC | BD | BE | BF | CD | CE | CF | DE | DF | EF |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *SM* | 0 | 1.00 | 0 | 1.00 | .50 | 0 | 1.00 | 0 | .40 | 0 | 1.00 | .56 | 0 | .40 | .56 |
| *J* | 0 | 1.00 | 0 | 1.00 | .33 | 0 | 1.00 | 0 | .25 | 0 | 1.00 | .33 | 0 | .25 | .33 |
| *RR* | 0 | .50 | 0 | .50 | .25 | 0 | .60 | 0 | .20 | 0 | .50 | .22 | 0 | .20 | .22 |
| *D* | 0 | 1.00 | 0 | 1.00 | .50 | 0 | 1.00 | 0 | .40 | 0 | 1.00 | .50 | 0 | .40 | .50 |
| $un_1$ | 0 | 1.00 | 0 | 1.00 | .67 | 0 | 1.00 | 0 | .57 | 0 | 1.00 | .71 | 0 | .57 | .71 |
| $un_2$ | 0 | 1.00 | 0 | 1.00 | .20 | 0 | 1.00 | 0 | .14 | 0 | 1.00 | .20 | 0 | .14 | .20 |
| *RT* | 0 | 1.00 | 0 | 1.00 | .33 | 0 | 1.00 | 0 | .25 | 0 | 1.00 | .38 | 0 | .25 | .38 |
| *K*1 | 0 | ∞ | 0 | ∞ | .50 | 0 | ∞ | 0 | .33 | 0 | ∞ | .50 | 0 | .33 | .50 |
| $un_3$ | 0 | ∞ | 0 | ∞ | 1.00 | 0 | ∞ | 0 | .67 | 0 | ∞ | 1.25 | 0 | .67 | 1.25 |
| *K*2 | 0 | 1.00 | 0 | 1.00 | .50 | 0 | 1.00 | 0 | .42 | 0 | 1.00 | .50 | 0 | .42 | .50 |
| $un_4$ | 0 | 1.00 | 0 | 1.00 | .50 | 0 | 1.00 | 0 | .42 | 0 | 1.00 | .55 | 0 | .42 | .55 |
| O | 0 | 1.00 | 0 | 1.00 | .50 | 0 | 1.00 | 0 | .41 | 0 | 1.00 | .50 | 0 | .41 | .50 |
| $un_5$ | 0 | 1.00 | 0 | 1.00 | .25 | 0 | 1.00 | 0 | .17 | 0 | 1.00 | .30 | 0 | .17 | .30 |
| *H* | −1.00 | 1.00 | −1.00 | 1.00 | 0 | −1.00 | 1.00 | −1.00 | −.20 | −1.00 | 1.00 | .11 | −1.00 | −.20 | .11 |
| *Y* | −1.00 | 1.00 | −1.00 | 1.00 | 0 | −1.00 | 1.00 | −1.00 | −.33 | −1.00 | 1.00 | .20 | −1.00 | −.33 | .20 |
| $\phi$ | −1.00 | 1.00 | −1.00 | 1.00 | 0 | −1.00 | 1.00 | −1.00 | −.17 | −1.00 | 1.00 | .10 | −1.00 | −.17 | .10 |

$un_1$, etc. stands for unnamed coefficients number 1, etc., numbered in the order of their presentation in Table 6-1.

For OTU's **A** and **C** it is 1, and so forth. In Table A-13 we have shown all the coefficients listed in Tables 6-1 and 6-2, using the quantities in Table A-12.

## A.3. Clustering methods

Numerous clustering methods have been applied in numerical taxonomy, but we do not propose to present all of them in detail here. A review of the methods has already been given in Section 7.3.2. We shall discuss only the most commonly used types; the others are only minor modifications thereof and are easily applied.

We first demonstrate the *weighted variable-group method*, using Spearman's sums of variables method for recomputing the correlation coefficients. The possibility of reversals in the coefficients on recalculating the correlation matrices after the initial clustering step has already been described in Section 7.3.2.4. When some of the coefficients are appreciably negative, as they generally are when characters have been standardized, the reversals become quite large. In the present example we have rather high negative correlations, brought about by the artificial nature of the matrix with which we are dealing. In real cases negative correlations of magnitude $-0.9$ would not be likely.

The first step in clustering—by any of the group methods—is to find the mutually highest correlations as central points of the clusters to be formed. By a mutually highest correlation we mean a correlation between any two OTU's which is higher than the correlation of these OTU's with any other OTU. When working with a desk calculator, it is convenient to represent the matrix of correlation coefficients in symmetrical form. This is shown in Table A-14, where the correlations of Table A-6 have been copied as a full (symmetric)

**Table A-14.** *The Correlation Coefficients of Table A-6 Copied as a Full (Symmetric) Matrix, with Decimal Points Omitted. The Highest Coefficient for Each Column Has Been Underlined*

| | **A** | **B** | **C** | **D** | **E** | **F** |
|---|---|---|---|---|---|---|
| **A** | *X* | −934 | 990 | −933 | 854 | −338 |
| **B** | −934 | *X* | −943 | 946 | −819 | −100 |
| **C** | 990 | −943 | *X* | −939 | 882 | −213 |
| **D** | −933 | 946 | −939 | $X_1$ | −884 | −201 |
| **E** | 854 | −819 | 882 | −884 | *X* | −095 |
| **F** | −338 | −100 | −213 | −201 | −095 | *X* |

matrix, omitting decimal points, which are understood to be in front of the leading digit. Next we underline the highest correlation in the column of each OTU (see Table A-14). We find that OTU **A** is correlated at level 990 with

OTU **C** and that the highest correlation of **C** is also with **A**. Thus the correlation between **A** and **C** is a mutually highest correlation and OTU's **A** and **C** will therefore form a cluster. We note, however, that **E** is also most highly correlated with **C** (882). However, **C** has a higher correlation with **A** than it has with **E**. Therefore the correlation between **C** and **E** is not a mutually highest correlation and **E** does not initiate a cluster. The highest correlation of OTU **B** is with **D** (946), and conversely **D**'s highest correlation is with **B**. Therefore **B** and **D** form a cluster. **F**'s highest correlation is with **E**, but **E**'s highest correlation is with **C**. Therefore neither **E** nor **F** are included in the initial clustering process. Thus at the conclusion of the first clustering cycle we find the following clusters:

$$\mathbf{A} + \mathbf{C}, \qquad \mathbf{B} + \mathbf{D}, \qquad \mathbf{E}, \qquad \mathbf{F}$$

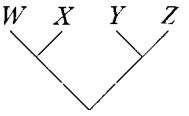
In any variable group method, more than two members are permitted to join a cluster during one clustering cycle. We therefore determine whether **E** or **F** should be permitted to join any of the present clusters. In a variable group method a criterion for cluster formation has to be furnished. If adding a new member to a cluster would produce an average correlation between the newcomer and the established cluster *lower than the previous level of junction by more than the criterion*, the prospective member is not admitted. In the initial study of Sokal and Michener (1958), the criterion value was set, somewhat arbitrarily, at 0.03; as it turned out, this was a satisfactory value for their study. If we accept this criterion in the present example, just for purposes of illustration, we first have to calculate the average correlation of prospective members with the clusters they are likely to join. The OTU **E** is highly correlated both with **A** and with **C** and it therefore appears to be a likely candidate for the already established cluster **A** + **C**. The average correlation of **E** with **A** + **C** is 0.868:

$$r_{AE} = 0.854, \qquad r_{CE} = 0.882,$$

$$\bar{S}_n = \frac{1}{2}(0.854 + 0.882) = 0.868.$$

Sokal and Michener (1958) used $\bar{L}_n$ as the symbol for the average correlation; we now prefer to call it $\bar{S}_n$. The difference from the correlation $r_{AC} = 0.990$ to $\bar{S}_n = 0.868$ is 0.122, considerably in excess of our criterion of 0.03. Thus during the first clustering cycle **E** does not join cluster **A** + **C**. It is obvious that **F** with its relatively low correlation with any other OTU will not join any of the established clusters during the present clustering cycle and thus the cycle can be concluded. In a more realistic example, involving more OTU's than the present one, a number of possible members may have to be examined and some of them may in fact join the cluster. In such a case, after three members have formed a cluster, one will have to calculate the average correlation of the three cluster members with a fourth possible member in order to decide whether the clustering should cease or whether the fourth member should be admitted into the cluster. It may even be that, in a group of four mutually highly cor-

related individuals, two initial clusters of two members would form, and the two clusters of two each would come together in a single cluster without lowering $\overline{S}_n$ appreciably:




At this stage we have to recalculate the correlation of all clusters and unclustered OTU's among themselves in preparation for the next clustering cycle. For this we use Spearman's sums of variables formula. This formula is

$$r_{qQ} = \frac{\square qQ}{\sqrt{q + 2\Delta q}\sqrt{Q + 2\Delta Q}},$$

where $\square qQ$ is the sum of all correlations between members of one group with the other group, $\Delta q$ is the sum of all correlations between members of the first group, $\Delta Q$ is a similar sum between members of the second group, $q$ is the number of OTU's in group 1, and $Q$ the number of OTU's in group 2. Whenever we have to calculate a correlation between a cluster and a single OTU, Spearman's formula reduces to

$$r_{xq} = \frac{\sum r_{xq}}{\sqrt{q + 2\Delta q}},$$

where the numerator refers to the sum of all the correlations of the single OTU with members of the cluster.

In recalculating the correlation matrix both these formulas will be employed. The computational steps are as follows:

$$\begin{aligned}\square(A + C)(B + D) &= r_{AB} + r_{AD} + r_{CB} + r_{CD} \\ &= -0.934 + (-0.933) + (-0.943) + (-0.939) \\ &= -3.749.\end{aligned}$$

$$\sqrt{2 + 2(r_{AC})} = 1.9950,$$

$$\sqrt{2 + 2(r_{BD})} = 1.9728,$$

$$r_{(A+C)(B+D)} = \frac{-3.749}{1.9950 \times 1.9728} = -0.953,$$

$$r_{(A+C)E} = \frac{r_{AE} + r_{CE}}{\sqrt{2 + 2(r_{AC})}} = \frac{0.854 + 0.882}{1.9950} = 0.870.$$

Similarly,

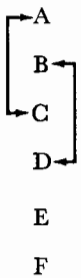
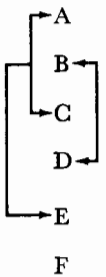
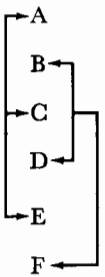
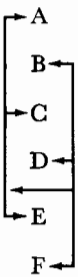
$$r_{(A+C)F} = -0.276, \qquad r_{(B+D)E} = -0.863, \qquad r_{(B+D)F} = -0.153.$$

The correlation between **E** and **F** is, of course, unchanged.

The correlations between the taxa at this clustering stage and for successive stages are shown in the lower part of Table A-15. One problem during this

**Table A-15.** *Successive Clustering of Correlation Matrix by* WVGM, *Using Spearman's Sums of Variables Method for Recalculating Correlations*

Matrix

| 1* | 2 | 3 | 4 |
|---|---|---|---|
|  |  |  |  |



Matrix 2

| | A′ | B′ | E | F | |
|---|---|---|---|---|---|
| A′ | *X* | −953 | 870 | −276 | |
| B′ | −953 | *X* | −863 | −153 | |
| E | 870 | −863 | *X* | −095 | A′ = A + C |
| F | −276 | −153 | −095 | *X* | B′ = B + D |

Matrix 3

| | A″ | B′ | F | |
|---|---|---|---|---|
| A″ | *X* | −939 | −192 | |
| B′ | −939 | *X* | −153 | |
| F | −192 | −153 | *X* | A″ = A′ + E |

Matrix 4

| | A″ | B″ | |
|---|---|---|---|
| A″ | *X* | −869 | |
| B″ | −869 | *X* | B″ = B′ + F |

*Matrix 1 is Table A-14

procedure is the labeling of the new clusters. The system adopted here is quite simple and works well for small matrices. Cluster **A** + **C** is now called **A′** and is defined at the right margin of Table A-15. Similarly, **B** + **D** has become **B′**, and successive clusters including **A′** and **B′** will be called **A″** and **B″** as their composition changes. When a large number of variables is employed, prime notation becomes too unwieldy; a convenient notation for desk calculator operations has been to use the letter or number of the first (leftmost) OTU in the group being clustered, together with a superscript or subscript identifying

the clustering cycle during which this group has been formed. Different workers have different schemes for keeping track of their clusters, and no uniformly accepted system has been developed. When clustering is carried out on a computer, a different method has to be adopted. In some existing computer programs groups are renumbered at each clustering cycle, and a record is kept in the machine identifying the code numbers of the taxa at each clustering cycle in terms of the previous cycle. This record is then punched out with the output and permits the investigator to identify the stems at each clustering cycle. We have found it very helpful to draw a schema as shown at the top of Table A-15. The criteria for successive clustering are the same as in the first clustering cycle, and the procedure can be easily followed by working through the example shown in Table A-15. The results of this clustering process can be represented in the dendrogram shown in Figure A-1.

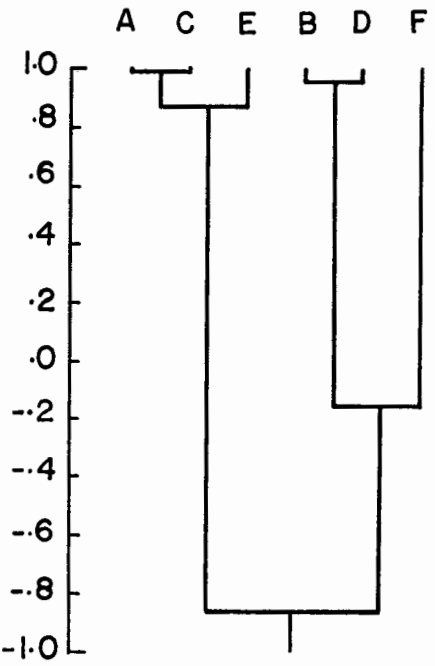



FIGURE A-1

*Dendrogram of the relations among six OTU's based on the WVGM clustering procedure of correlation coefficients shown in Table A-15. The ordinate is shown in correlation coefficient scale,* r.

We might now take time out to discuss the various modifications of this clustering procedure which have been suggested. The most common variant is the *weighted pair-group method*, which permits only the two most highly correlated stems to join at each clustering cycle. Thus this method needs no criterion for admitting further joiners during any one cycle. In the present instance, the pair-group method gives identical results with the weighted variable group methods because no more than two stems came together during one clustering cycle anyway. Another variation is an *unweighted group method*, which during each clustering cycle recomputes correlations based not on the previous matrix but on the initial matrix. Thus, in Table A-15, if we were to calculate one of the correlations shown in matrix 3 by an unweighted method, these would not be based on the correlation coefficients found in matrix 2, but, after considering which OTU's go to make up the two clusters which are joining in matrix 3, would be based directly on the original correlation coefficients of matrix 1. We shall evaluate the correla-

tion between **A''** and **B'**. **A''** consists of **A** + **C** + **E**, and **B'** consists of **B** + **D**. By the unweighted method each OTU has equal weight within each group, while in the weighted method **E** has a weight equal to the stem **A** + **C**. The computations for the unweighted correlation between **A''** and **B'** are as follows (using Spearman's sums of variables method):

$$\square(A + C + E)(B + D)$$
$$= r_{AB} + r_{AD} + r_{BC} + r_{CD} + r_{BE} + r_{DE}$$
$$= (-0.934) + (-0.933) + (-0.943) + (-0.939) + (-0.819) + (-0.884)$$
$$= -5.452,$$

$$\sqrt{3 + 2(r_{AC} + r_{AE} + r_{CE})} = \sqrt{8.452} = 2.9072,$$

$$\sqrt{2 + 2(r_{BD})} = 1.9728,$$

$$r_{(A+C+E)(B+D)} = \frac{-5.452}{2.9072 \times 1.9728} = -0.951.$$

This correlation of $-0.951$ is appreciably different from the $-0.939$ obtained by the weighted variable group method. This example also illustrates the "reversals" induced by Spearman's method. The average correlation is lower than any of its constituent correlations. Near the lower limit of $r$ this tendency can result in values of $r$ slightly less than $-1.0$. Thus unweighted $r_{(A+C+E)(B+D+F)} = -1.013$.

Finally, instead of Spearman's sums of variables method, *simple averages* could be used. This should not be done with correlation coefficients, since their variance depends on their magnitude, but can be done with transformations of correlation coefficients either to Fisher's $z$ or to angles which represent the arc cosines of the correlation coefficients. Such transformations can be looked up in conventional statistical or mathematical tables, or they are automatically generated on a computer. Thus the correlation 0.700 would be 0.867 when transformed to $z$ and 45.63° when transformed to an angle (arc cos $r$). A correlation coefficient of 0.0 corresponds to a $z$ of 0.0 and arc cos of 90°; $r = 1$ corresponds to $z = \infty$ and arc cos $r = 0°$. The transformed coefficients then can be averaged and retransformed to correlations if desired. The most common use for averages rather than Spearman's correlations would be in distance coefficients. Table A-16 shows a weighted pair-group method for clustering the distance coefficients. Since most clustering methods work from high coefficients to low ones and since the closest taxonomic units have smaller distances than less related taxonomic units, the complements of the distances are computed, generally the ten-complements (= 10 − taxonomic distance). The successive matrices using this method are also shown in Table A-16. They were computed by the weighted pair-group method, so that no more than two stems joined at any one cluster. The new distances between clusters are always calculated as simple averages and thus correspond to the $\overline{S}_n$ values as described before. The interested reader should be able to reproduce easily for himself the figures

**Table A-16.** *Ten-complements of Distance Coefficients Clustered Successively by* WPGM, *Using Averages of Distance Coefficients to Form Successive Matrices**

| | | OTU's | | | | | |
|---|---|---|---|---|---|---|---|
| | OTU's | A | B | C | D | E | F |
| Matrix 1 | A | *X* | | | | | |
| | B | 7.915 | *X* | | | | |
| | C | 9.779 | 8.100 | *X* | | | |
| | D | 7.956 | 9.653 | 8.140 | *X* | | |
| | E | 9.339 | 8.482 | 9.498 | 8.406 | *X* | |
| | F | 8.552 | 8.616 | 8.706 | 8.520 | 9.016 | *X* |
| | | A′ | B′ | E | F | | |
| Matrix 2 | A′ | *X* | | | | | |
| | B′ | 8.028 | *X* | | | | |
| | E | 9.419 | 8.444 | *X* | | | |
| | F | 8.629 | 8.568 | 9.016 | *X* | | |
| | | A″ | B′ | F | | A′ = A + C | |
| Matrix 3 | A″ | *X* | | | | B′ = B + D | |
| | B′ | 8.236 | *X* | | | A″ = A′ + E | |
| | F | 8.823 | 8.568 | *X* | | A‴ = A″ + F | |
| | | A‴ | B′ | | | | |
| Matrix 4 | A‴ | *X* | | | | | |
| | B′ | 8.402 | *X* | | | | |

* Data taken from Table A-9.

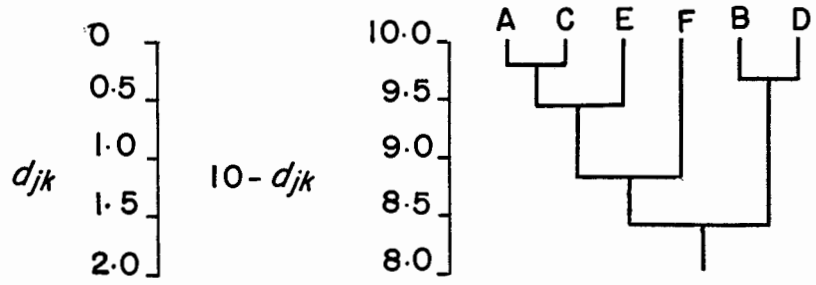



FIGURE A-2

*Dendrogram of the relations among six OTU's based on the WPGM clustering procedure using averages of distance coefficients shown in Table A-16. The ordinate is shown in distance scale,* d, *as well as in the ten-complement scale* (*10.0* − d) *in which the computations had been carried out.*

shown in Table A-16. The results of this clustering method are shown in Figure A-2.

## A.4. Miscellaneous techniques

The representation of taxonomic relationships in the form of dendrograms gives rise to problems. First, there is inevitable distortion of the relationships by trying to represent an essentially multidimensional relationship in two dimensions. Second, different relations can be obtained by using different similarity coefficients and methods of clustering. In order to measure the amount of distortion introduced by such techniques, the method of cophenetic relationships was developed (Section 7.4). This consists of drawing horizontal lines across a dendrogram, as shown in Figure A.3, which is the dendrogram based on Figure A.1 with the horizontal, so-called phenon lines drawn at regular intervals across the figure. The cophenetic classes delimited by these phenon lines are coded "1" through "8," and the relationship between any pair of OTU's is assigned the cophenetic value of the cophenetic class in which the two OTU's connect. For instance, **A** and **C** have a cophenetic value of 8, since **A** and **C** connect in the cophenetic class 8. On the other hand, **B** and **F**

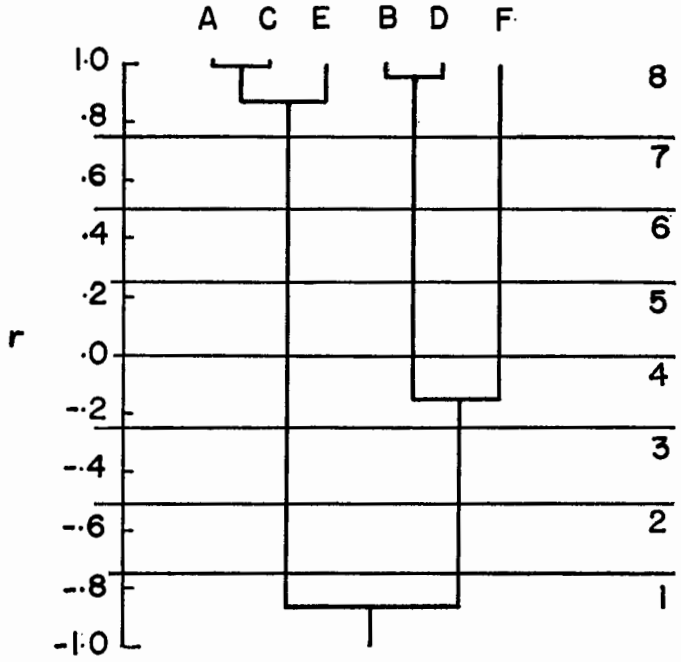



FIGURE A-3

*Dendrogram of Figure A-1 with phenon lines drawn in to illustrate the computation of cophenetic values.*

have a cophenetic value of 4 because they connect in the phenon class 4. The matrix of cophenetic relationships of the OTU's is shown in Table A-17.

**Table A-17.** *Cophenetic Values Between All Pairs of OTU's, Based on the Dendrogram of Figure A.3*

| | | OTU's | | | | | |
|---|---|---|---|---|---|---|---|
| | | **A** | **B** | **C** | **D** | **E** | **F** |
| | **A** | *X* | | | | | |
| | **B** | 1 | *X* | | | | |
| OTU's | **C** | 8 | 1 | *X* | | | |
| | **D** | 1 | 8 | 1 | *X* | | |
| | **E** | 8 | 1 | 8 | 1 | *X* | |
| | **F** | 1 | 4 | 1 | 4 | 1 | *X* |

Such a matrix can now be strung out in single file, first column followed by second column and so forth, and compared against either the original correlation coefficients or against a similarly strung out matrix of cophenetic values for a dendrogram obtained by another method. Such computations have not been carried out here because with so few OTU's the resulting correlation coefficient would be quite unreliable. The computation of correlations between such linearly arranged matrices follows exactly the same formula and procedures as have already been described.

If ranks are to be assigned to various taxa, phenon lines may also be used for this purpose. For instance, if the hypothetical OTU's in our present study are species and it is felt that they should all belong to one genus, then a phenon line might be drawn across Figure A.1 at $r = -0.500$ to represent the subgeneric level; then two subgenera [(−.5)–phenons] could be recognized: **A** + **C** + **E** and **B** + **D** + **F**. On the other hand, if one wished to raise this line to $r = -0.100$, three subgenera [(−.1)–phenons] could be delimited: **A** + **C** + **E**, **B** + **D**, and **F**.

Some remarks on the significance of coefficients of similarity are pertinent here. Section 6.2.4 discussed why the significance of such coefficients in a similarity matrix is not an important issue in numerical taxonomic work. The heterogeneity of the column vectors makes ordinary tests of significance inappropriate. However, lacking better ones we might employ the conventional tests as rough guide lines. Thus for a simple association coefficient, such as $S_{SM}$ or $S_J$, we can use the standard error of the binomial as an approximation. The standard error would then be

$$\text{S.E.}_S = \sqrt{\frac{S(1 - S)}{n}},$$

where $S$ is the association coefficient and $n$ is the number of characters. For

example, an association coefficient of 0.30 based on 60 characters would have a standard error of $\sqrt{\frac{(0.30)(0.70)}{60}} = 0.0592$. This standard error can be used to set approximate confidence limits to the estimate of the association coefficients, using the normal distribution, in view of the large number of characters on which estimates are generally based in numerical taxonomy. Ninety-five percent confidence limits are calculated as

$$0.30 \pm (1.96)(0.0592) = 0.184 \longrightarrow 0.416.$$

For 99% confidence limits we would replace 1.96 by 2.58. More exact con-

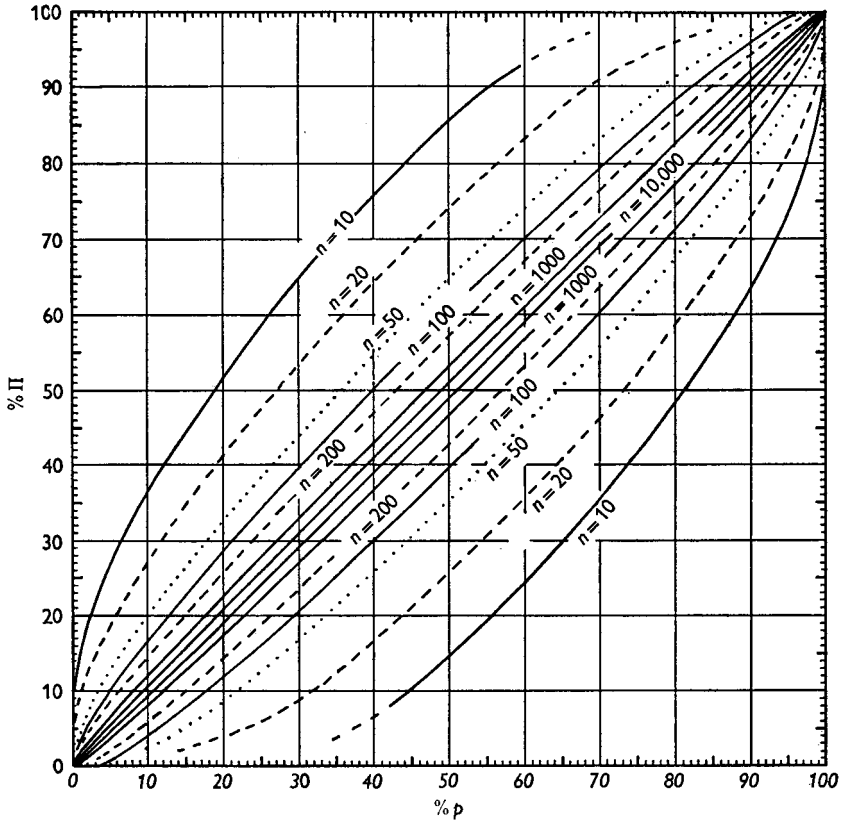



FIGURE A-4

*The 95% confidence limits of the binomial distribution for various values of* n. *For example, if the proportion,* Π, *of positive values in a large population is 70% and one takes repeated samples of size* n = 20, *then in 95% of the cases the observed proportion of positives,* p, *will lie between 45.7% and 88.1%.* [*Reproduced with permission from the 12th Symposium of the Society for General Microbiology.*]

fidence limits can be obtained from Figure A-4, in which confidence bands based on various sample sizes, $n$, are shown, including values of $S$ close to 0 or 1.

To test whether the significance of the association coefficient is significantly different from zero, we test whether $S/\text{S.E.}_S$ is greater than 1.64 or 2.33 at the 5% or 1% level, respectively (this is a one-tail test, since $S$ cannot be negative). Thus for an association coefficient of 0.30 based on 60 characters, $0.30/0.0592 = 5.07$, which indicates significance at $P < 0.01$.

For the correlation coefficient, standard errors are best calculated after a transformation of the correlation coefficient to Fisher's $z$. This transformation is given in most sets of statistical tables and has already been discussed above. The standard error for $z$ is

$$\text{S.E.}_z = \sqrt{\frac{1}{N-3}}.$$

Thus for a correlation coefficient of 0.70 based on 60 characters we would compute 95% confidence limits as follows:

$$\text{S.E.}_z = \sqrt{\frac{1}{57}} = 0.132,$$

$$r = 0.70 \text{ corresponds to } z = 0.8673.$$

$$\text{Confidence limits: } 0.8673 \pm (1.96)(0.132) = 0.6086 \longrightarrow 1.1260.$$

These limits correspond to $0.54 \longrightarrow 0.81$ on the $r$ scale. Confidence limits of correlation coefficients can also be looked up from a graph shown in Table 15 of the *Biometrika Tables for Statisticians* (Pearson and Hartley, 1958).

The significance of correlation coefficients can be tested as $z/\text{S.E.}_z$ after they have been transformed to $z$ values. Thus the coefficient discussed above, 0.8673, is tested as follows:

$$\frac{0.8673}{0.132} = 6.57, \text{ which is greater than } 2.58;$$

therefore the coefficient is significantly different from zero at $P < 0.01$.

The computation of confidence limits for the distance $d$ is rather involved. Readers interested in the details are asked to consult Rohlf (1962) and Rohlf and Sokal (1963). Since the computations are involved, we have thought it better to tabulate expected values and upper and lower 95% bounds for the distance for four values of $n$, with $n = 20$, 50, 100, and 200. The values of $n$ of interest to the reader are likely to be less than 200 and the bounds for a given value of $n$ can be found by interpolation. These values are presented in Table A-18. In the first and last columns of this table are given the expected value

**Table A-18.** *Expected Values, Limits Enclosing 95% of the Distances, and Approximate Standard Errors for Average Distance Coefficients at Four Sample Sizes ($n$ = number of characters upon which coefficients are based)*

| | | 95% Limits | | |
|---|---|---|---|---|
| $n$ | $\mathcal{E}(d)$ | Lower | Upper | S.E.$_d$ |
| 20 | 1.397 | 0.979 | 1.849 | 0.222 |
| 50 | 1.407 | 1.138 | 1.690 | 0.141 |
| 100 | 1.411 | 1.218 | 1.610 | 0.100 |
| 200 | 1.412 | 1.274 | 1.551 | 0.071 |

of $d$ and the standard error for $d$, $\sigma_d$, respectively. These are based on the formulas of Section 6.2.3.2:

$$\mathcal{E}(d) = \frac{(n-1)!}{\left[\left(\frac{n}{2}-1\right)!\right]^2 2^{n-2}}\sqrt{\frac{\pi}{n}},$$

$$\sigma_d^2 = 2 - [\mathcal{E}(d)]^2.$$

The limits shown in Table A-18 are not based on the standard error but on the more exact distribution of distances as derived by Rohlf (1962). However, the approximate standard errors shown here should be adequate for most users. These are computed as follows: for $n = 100$ characters, for example,

$$\log \mathcal{E}(d) = \log (n-1)! + \log \sqrt{\pi} - 2\log\left[\left(\frac{n}{2}-1\right)!\right] - (n-2)\log 2 - \frac{1}{2}\log n$$

$$= \log 99! + \log \sqrt{\pi} - 2\log 49! - 98\log 2 - \frac{1}{2}\log 100$$

$$= 155.970{,}0037 + 0.24858 - (2 \times 62.784{,}1049) - (98 \times 0.30103) - \frac{1}{2}(2.00000)$$

$$= 0.14943,$$

$$\mathcal{E}(d) = 1.4107,$$

$$\sigma_d^2 = 2 - [1.4107]^2 = 0.00993,$$

$$\sigma_d = 0.100.$$

Thus, to test the significance of a distance value of 0.8 based on 100 characters, we could proceed as follows:

$$\frac{d_{jk} - \mathcal{E}(d)}{\text{S.E.}_d} = \frac{0.800 - 1.411}{0.100} = -6.11,$$

which is a highly significant difference (greater than 2.58, hence $P < 0.01$).

## A.5. Computer programming for numerical taxonomy

We shall not give detailed flow charts and programs for computer processing in numerical taxonomy, although these exist, because they change very rapidly as computer facilities improve and new methodologies are developed. Programs can be obtained from various sources, and the interested reader is referred to a newsletter concerned with numerical taxonomy, "Taxometrics" (write: Mr. L. R. Hill, Progetto Sistematica Actinomiceti, Istituto "P. Stazzi," Via Celoria 10, Milano, Italy), in which descriptions of programs and their availability will be published. In general it has been our experience that programs are not too easily transferred from one computer installation to another; even if two installations have the same computer the machines usually have slightly different specifications. Therefore programs often need to be reworked to a slighter or greater extent to make them compatible. As computers get faster and the so-called automatic coding systems become more flexible and powerful, it is likely that programs for numerical taxonomy will no longer be written in machine language but in one of the automatic coding systems. In the United States FORTRAN, developed by IBM, and in Europe ALGOL seem to be the most widely used languages for addressing a great variety of computers. FORTRAN programs are currently being written for all numerical taxonomy procedures, but these will have to be recompiled at different computation centers in order to adapt them to the specific installations. The average taxonomist has probably never seen a computer and is quite likely to shy away from getting involved in the computing business. We would like to emphasize that learning a language such as FORTRAN is extremely simple and that a little acquaintance with the computational aspects of the work would pay copious dividends. On the other hand, as we have repeatedly emphasized throughout the book, the data can in fact be sent out to a number of installations for processing if the taxonomist is not interested in doing any of the computing himself.

Programs that are to be written for numerical taxonomy should have detailed write-ups to enable machine operators unfamiliar with the computations and taxonomists unfamiliar with computers to do the work with maximum facility. Each program write-up should

1. describe the general idea of what the program will do;
2. describe the maximum capacity of the program—how many OTU's can be processed and how many characters;
3. describe in detail the actual algebra of the program, unless it is a very standard procedure in which reference to a publication would be adequate;

4. give detailed operating instructions of what to do on a given computer to execute this program;

5. specify the exact format in which the data must be presented to the computer;

6. specify in detail the format in which the output of the program will be presented;

7. add time estimates for executing the program on a given computer;

8. provide some examples of prepared data input in order to amplify point 5;

9. provide some examples of the output in order to amplify point 6;

10. provide a small example completely worked out in order to give the worker a test case on which to check the program on his own machine.

Eight different kinds of computer programs for numerical taxonomy are envisaged at present:

Group 1 would be *control programs* for large, powerful machines. These would be master programs which would control the succeeding programs (labeled 2 through 8 here). By this is meant that the master program would call up different subprograms which could coordinate an entire numerical taxonomic study from the initial presentation of the data to the final establishment of a classification, calling on the subsequent subprograms as they are needed and directing the flow of the operations. Such a master program is feasible only on large computers, on which various sections of the program would be stored on magnetic tape units and would be called up as they are needed. On the smaller computers, such as the IBM 650, such a master program is not feasible, and the flow of operations is directed by the operator, who runs the data through repeatedly, using different programs.

Group 2 are *language translation programs*. These programs are still in their infancy. They would take descriptions of characters in words and be programmed in such a way as to convert these into numerical codes in some logical and consistent fashion. Such programs eventually may be able to remove much of the tedium of coding characters from the shoulders of the taxonomist.

Group 3 *processes and converts characters* to prepare them for the calculation of similarity coefficients. These procedures would convert the data to the form necessary for whatever program is to compute the similarity coefficients. Standardization of characters would fall into this category, as would transposition of matrices to change input by rows to input by columns.

Group 4 programs *calculate affinity or similarity coefficients* between pairs of OTU's according to the formulas shown earlier in the book.

Group 5 are *cluster analysis programs*. These take the output of Group 4 programs and make clusters from these data. They often also convert the clusters into dendrograms, printing these in some suitable form.

Group 6 are the *data extraction programs*. These too are only in their infancy. They extract data, answering specific questions which are addressed to the

study. So far, work of this sort has usually been done with the printed output, the investigator himself checking into certain situations which seem of interest to him. However, it is quite feasible on larger computers to include such procedures as part of the computer program. Thus any questions such as "What does character 33 in OTU 45 look like?" or "How do OTU's 37 and 89 compare for characters 13 through 35?" can easily be looked up by the machine and the results printed in suitable form for the benefit of the investigator.

Group 7 are *interstudy coordination programs*. Such programs would store and sort previous studies, establish reference taxa and their characters, and correlate different studies. Again such procedures would require large computers and are at present only in their earliest developmental stages.

Group 8 are *publication programs*. These would convert the output into legible and publishable form, such as diagnostic keys with descriptions of characters and organisms.

The above outline is based in part on a memorandum by Sneath and Rohlf in *Taxometrics* **2**, December 1962.