The Comparative Method
by
H.M. Hubey

Abstract
A thorough overview of the problems, assumptions and methods of comparative linguistics is given. It is shown that the concept of distance (or its inverse similarity) is of paramount concern both for phonological and semantic aspects of the comparative method. Different distance metrics, their results and their relationships to the heuristic and intuitive comparative method is shown. The importance of basic vocabulary, and the formalization of such concepts in the form of Swadesh-like lists is demonstrated and explained. Relationships of distinctive features, shared innovations and shared retentions in family trees is discussed in terms of distance and similarity. The role of regular sound change, its meaning, its importance, and its heuristic value as manifested in various distance metrics is demonstrated.

1. The Comparative Method

It is obvious that the method compares sets of morphemes (bound morphemes or free morphemes) from two languages to each other. The question of which morphemes are compared, and how the results are evaluated are often left foggy. It is taught via example which leaves students wondering exactly which parts of the method are salient. What is done is that the degree of similarity (both semantically and phonologically) between a specific set of morphemes chosen according to some assumptions is then judged (again fuzzily) and the final result is again evaluated. Among the set of procedures are phrases such as "regular sound correspondence", "cognate", "nursery words", and "basic vocabulary". These will be discussed as they are relevant to the comparative method or the historical method. The similarity (often "phonetic similarity") is the inverse of "distance", and mathematical spaces that have a distance defined on them are called metric spaces. Therefore here we will use only the concept of distance since the concept of similarity is derivable from it rather easily. For example, if d(x,y) is the distance between x and y, then we can easily relate similarity s(x,y) to distance as s(x,y)=1-d(x,y). It should be noted here that the similarity and distance are"normalized" in that maximum allowable value is 1, and the minium is 0. Therefore if d(x,y)=0, it means that x and y are identical, hence for any variable z, d(z,z)=0. It is easily seen that s(z,z)=1 or z is maximally similar to itself (i.e. identical). An alleged metric, or *measurement scale* (please see the appendix) which cannot discriminate an identity is surely very much suspect. What good is a thermometer that cannot measure a given temperature correctly twice i.e. is not reliable (see the appendices on measurement theory).

Thus we can work with the concept of distance itself like the rest of the mathematical sciences. Semantic distance will be extremely difficult to define except using some fuzzy measures; Zadeh intended fuzzy logic especially for the social sciences, especially to be used in senses in which we can make distinctions such as hot, very hot, somewhat hot, not hot, not too hot, warm, very warm, cold, not very cold, etc. As for phonetic. phonological, acoustic, perceptual, sound distances, there are many ways in which can be accomplished. Here the word "signal" will be used since it is a representation of the acoustic manifestation of what we say are "speech sounds"; phonetic and phonological are essentially high and low resolution depictions of the same, and perceptual dis-

tance is already implicitly used in making these low/hi resolution representations of speech sounds. Furthermore, there is nothing to stop anyone from using any of the distance measures used on the signal in many speech recognition programs many of which are already commercially available at commodity prices.

Define M_j^{α} to be the jth morpheme of language alpha (α). Then we define $d_p(M_j^{\alpha}, M_k^{\beta})$ to be the signal distance between the jth morpheme of the alpha language and the kth morpheme of the beta language. This distance can be obtained from the distances between the phonemes as a low resolution distance in many ways, including the simplest such distance, Hamming distance as used in Hubey [1998]. Other more sophisticated and more realistic distances can also be seen in Hubey[1994].

Similarly define $d_s(M_j^{\alpha}, M_k^{\beta})$ to be the semantic distance between these morphemes. The comparative method works by comparing the (bound and/or free) morphemes of two languages and is semantically driven. To see what these mean, let us examine special cases. As an example, let us use English as a metalanguage and try "dog" in some virtual language family X. Suppose we have several words with this meaning in two languages; the set $\{M_1^a, M_2^a, M_3^a\}$ in language A and the set $\{M_1^b, M_2^b, M_3^b, M_4^b\}$ in language B. Which ones do we choose to be in the list i.e. the set of comparanda? We can represent this situation as a bipartite graph as shown in Figure I.

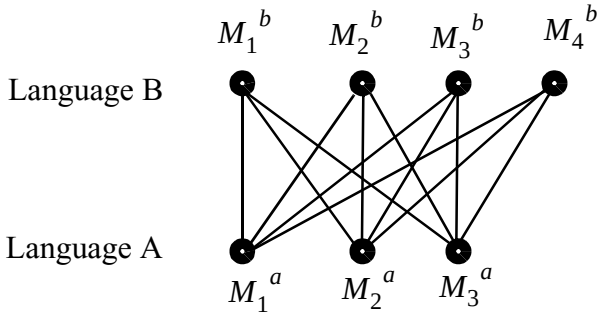


Figure I: The Complete Bipartite Graph of Correspondence. The graph is $K_{3,4}$.

A complete bipartite graph $K_{n,m}$ is a graph whose vertex/node set is divided into two mutually exclusive sets such that there is an edge between every pair of vertices between these sets. Obviously $K_{n,m}$ has mn edges. We must choose one of these edges which represent the relation "potential cognate". On what basis do we make this decision? The fuzzy answer is that we try somehow to minimize the semantic distance but also the signal distance. For example, if we are given the choices {aka, aba, utu, opo} in one language and {akka, kabar, zultar} in another language, even if *aka* meant 'dog' and *akka* meant 'wolf' and even if *zultar* meant *dog* we might be tempted to select {aka,akka} as the potential cognate pair. In this case we are making an attempt to minimize some function of *both the signal and the semantic distance.* Therefore our algorithm minimizes some function $\Phi(d_p(M_\mu^a, M_\upsilon^b), d_s(M_\mu^a, M_\upsilon^b))$ of the semantic and signal distances

so that when we have actually chosen our potential cognate pair this function is minimized for that specific pair. This part of the algorithm is purposely left foggy because it is exactly that way in real life as practiced by real linguists. Despite denials that they indulge in no such thing, it is quite easily discernable from the examples in books that something along these lines is attempted, and indeed, there is no other way.
Since we have now made the procedure explicit, at least to a degree, it will now be easier for those using the comparative method to either clarify what algorithm to use or for empiricists to study what has been done and attempt to obtain what algorithm is actually (and implicitly) in use.

Suppose we compare some language gamma (γ) to language to itself, i.e a dialect of itself. Then we would be comparing the "corresponding" morphemes $d_p(M_j^{\gamma}, M_j^{\gamma})$ where "correspondence" here is driven by semantics meaning that we index the words using the semantics. In other words we might say that we select morphemes such that $d_p(M_j^{\gamma}, M_j^{\gamma}) \approx d_p(M_j^{\gamma}, M_j^{\gamma})$ and $d_s(M_j^{\gamma}, M_j^{\gamma}) \approx d_s(M_j^{\gamma}, M_j^{\gamma})$ .Obviously if we compared the same dialect to itself the distances would be identical and hence we would have exact equivalence, instead of approximate equivalnence thus we would have $d_p(M_j^{\gamma}, M_j^{\gamma}) - d_p(M_j^{\gamma}, M_j^{\gamma}) = 0$ and $d_s(M_j^{\gamma}, M_j^{\gamma}) - d_s(M_j^{\gamma}, M_j^{\gamma}) = 0$. There are, of course, apparent problems with this scenario. One of them is that the term 'cognate' is apparently not defined in terms of distance (or its inverse similarity) but in terms of something else, corresponding phonemes. This is a problem of 'multiple scales' and of measurement theory... There are similar problems in other sciences. For example, in problems of intelligence, the brain to body ratio (b/B) correlates very well with intelligence when it is applied over a large scale, from the lower order animals up to humans, but fails to be significant when only humans are being considered. We can see here that signal and semantic distance is highly correlated with geneticity when the same language or dialects of the same language or very closely related languages are being compared but apparently fails to be germane when distantly related languages are being compared. This will be treated at the end after other problems of the comparative method are outlined and the algorithm given, and more and related complexity is discussed. A related problem is that of shared features, retention of features and parallel development. Obviously these ideas are related to those of cladistics in biology. Just as obviously, it will add even more complexity to the comparative method and cannot be discussed until at least the complete (even if simple) algorithm is given and discussed.

2. The List

We now apparently know how to choose potential cognate pairs. There are still more complications which will be evident when we try to create a whole list of such pairs. Which and what kinds of words (morphemes) do we use for our test and how many of these shall we have? Obviously, there are some assumptions that go into this method. We might think that more the merrier and choose the whole language. In the case of comparing dialects or the language to itself, there is no difference for all practical purposes. However the phenomenon of copying (also known as borrowing) complicates the situation since some of the words (free morphemes usually) might have come from another language. Regular sound change will not resolve this problem since even copied/borrowed words show regular sound patterns. The assumption is that some words "basic

vocabulary" (BV) are resistant to copying and thus they should be used. It is here that we are faced with the problem of BV. How do we choose these words? Here we see more complexity in choosing potential cognate pairs, and also the BV. Suppose both the word for 'wolf' and 'dog' were to be included in the BV. Then it is obvious that we cannot initially separate the potential words into two sets and then try the algorithm above, since we might decide to match up 'wolf' in one language with 'dog' in another language. We would have to consider both sets together and maximize or minimize some function of both the signal and the semantic distance. Therefore although the general idea of computing distances and comparing morphemes to each other holds we have to consider the whole BV set together, however it could be done in several stages. In the first stage we can treat the semantically-driven matchings independently of each other, and to do that we have to make sure that we select a set of words which are maximally semantically distinct from each other. The Swadesh list is a formalization of this concept. It is supposed to be a set of words that are (i) semantically distinct (i.e. large semantic distances amongst the words) and (ii) at the same time words that are resistant to copying by other languages.

(i) The first part is relatively easy to state assuming that we have the required semantic distances. Compute the semantic distance between every pair of words in the set, and then maximize some function of these distances. Therefore a simple example of such an operation is to *Maximize* $\sum_{i<j} d_s(M_i^m, M_j^m)$. The superscript m merely denotes that this is being done in a metalanguage and $i<j$ denotes the fact that we do not consider the case i=j and that the distance measure is symmetric so we need only compute the distance for only a single occurrence of given i and j. For N words we will have to compute N(N+1)/2 such distances. We could decide that we want to *Maximize* $\prod_{i<j} d_s(M_i^m, M_j^m)$, or we might try exponential or logarithmic forms. A simple heuristic algorithm for selecting m words(the list) from n words(the complete language) would be to start with the complete graph of the n words in which each edge weight is the inverse semantic distance(i.e. similarity) s_{ij} between the words. We would then sum up the edge weights for each node (word) $S_j = \sum s_{ij}$. Then we would successively delete the node with the largest S_j until we are left with only m words. The algorithm is given in Appendix A.

(ii) How do we justify that assumption that some words are resistant to copying? The general belief is that some words, such as hi-tech words, are easily borrowed, and that they should not be used in comparison tests. However, there were probably technology words during many periods of human history. Even the pronouns could have been copied at some distant time in the past. Innovation is an extremely difficult thing in every aspect of life, and it is much easier to recognize a good solution to a problem and to copy than it is to innovate; this must never be forgotten. Even morphological and syntactic forms can be copied from other languages [Thomasen & Kaufman,1988]. As unbelievable as it might sound, the alleged resistance of some words to copying is essentially circular. It came out of the study of the IE language family. In order to claim that these languages constitute a family, we have to see that some of these words from the set of basic vocabulary resemble each other across a variety of languages both in semantics and in the signal shape/form. Therefore it was the acceptance of the IE family based on these resemblances (i.e. small distances) that gave rise to the acceptance of the belief that some words are resistant to

copying/borrowing. Lest this sound uncharitable, we should note that although it adds more complication, the truth is closer to the explanation that both of them occurred together and positively influenced each other. In that sense they are inseparable. Therefore we have two assumptions;

I.) The BV comparison does not display great changes and is resistant to copying. (How do we know this? Why,of course, in the so-called IE languages which we know constitute a family by (II), the BV does not show any great changes, actually less changes than some of the other words).

II). The so-called IE languages constitute a family. (How do we know this? Why, of course, because the BV of the IE language family does not show great changes and this implies by (I) that these languages constitute a family.)

It is certainly internally consistent. But so are these two statements:

i) I am Napoleon and this person here is General Marat.
ii) Yes, I am General Marat and this person here, I testify, is Napoleon.

Now, consistency is something demanded of axiomatic, or formal systems of mathematics, but more is demanded of other sciences; they must also be in agreement with reality. In the physical sciences we can test [consistent] mathematical models to see which ones are in concordance with the facts. In historical linguistics we can only know about a few thousand years of recorded history much of which is still shrouded in mystery. The only thing we can know for sure, and which we can use is that certain events are too improbable to have occurred due to chance and thus are likely due to some other process. Large number of concordances of words (morphemes) between sets of languages cannot be due to chance and hence are due to either copying from each other or descent from a common ancestral language (knowing full well that even borrowed words are descended from a common ancestral language). First we have to compute what “large number of concordances” are. In other words we need to have numbers (that can be seen in many places, some simple, some sophisticated and some incorrect, for example, Bender [1962], Cowan[1962], Ringe[1992,1995], Hubey[1994,1999a]). Secondly, we have to clarify what it means to be descended from a common ancestral language. After all, language is something like an infection or genes that is carried by humans. It is not easy to think only about an abstract concept called language without making use of abstract tools of mathematics. Therefore since there is so much writing already available in IE and AA languages, they will have to be used as test beds for mathematical models of historical linguistics.

Furthermore, it is not completely true that the statements above are circular, or they can be made so that they are not circular. To conduct a proper experiment we’d have to trace at least a single family which has enough writing (a dictionary would be best) over many centuries so we can verify that at least in the case of this family some words do not get copied. In order to have sufficient belief in the assumption of resistance, we would have to verify this for many languages. Suppose we traced say, Latin and its derivatives (descendants and others that copied words from it, and also languages which eventually came to be replaced by Latin because the speakers were incorporated into the Roman Empire) and quantify how much change occurred in both the meanings and the signals of the morphemes. Where does copying end, and where does the copying language change sufficiently to be in the Latin (Italic, Romance, etc) family? We would need to use many

other examples, as well, in order to have reasonable confidence, at least a quantifiable confidence, in what words are best suited to be included in the BV, and also what kinds of mathematical models best describe the historical processes of languages. Much of what has been said already can be formalized into such models, and has been done already [Hubey,1999a]. Also related to this problem is that of 'nursery talk' which alleges that some signal forms exist accross many languages because they are invented by children (like onomotopeic words) and that they should not be used. Psychological experiments show that the opposite is true since infants are exposed to their mother's voice even in the womb, and are capable of discriminating the sounds and words of their native languages within weeks or months after birth, hence the infants are imitating the words which they have heard from their parents[Hauser,1997, USNWR,1998]. Even more material on this topic can be found in Jablonski and Aiello[1998]. Furthermore it is still a topic of heated discussion in the psychological sciences as can be seen in [Marcus, et al,1999]. These are probably the strongest evidence for protoworld of some sorts since words like {ata,ana,ati,...} now cannot be summarily dismissed and curious minds will want to know why Hittite has 'atta' and 'anna' for father and mother, and why the Karachay-Balkar [Kipchak] language of the North Caucasus has the 'parental words' {ata,ana, atta,anna,appa,akka, amma}.

3. Shared Innovations and Shared Retentions: possibly more red herrings

Even after the resolution of the problems of how to select the words for comparison, and deciding what kind of a distance metric to use, and what words to avoid and what not to avoid, we are still faced with problems on a larger scale. Eventually, after the analysis we will have accumulated some processed data which we can use. Let us call these 'features' or 'distinctive features' of the languages in question. We still have to somehow make use of these features to create a family-descent tree. To know what to do with real data, we should first decide what we would notice in cases in which we already knew the answer. Suppose we have the distinctive features of the protolanguage and its two daughter languages A, and B. We can write these as vectors;

3.1)
$$P=\begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \end{bmatrix},\quad A=\begin{bmatrix} a_1 \\ b_1 \\ c \\ d_1 \\ e_1 \\ f_1 \\ g \end{bmatrix},\quad B=\begin{bmatrix} a_2 \\ b_1 \\ c_2 \\ d \\ e_1 \\ f_2 \\ g \end{bmatrix},\quad C=\begin{bmatrix} a_2 \\ b_1 \\ c_3 \\ d \\ e_1 \\ f_3 \\ g \end{bmatrix}$$

The protolanguage has six features represented by the letters as shown. Wherever we see the same symbol in A or B it means that these are retentions. The symbols with subscripts are innovations, thus b1 is a shared innovation since both A and B possess it. Similarly g is a shared retention since it is retained exactly as it is in the protolanguage. Now we can compute distances based on the distinctive features quite easily using the metric discussed in Hubey[1998] and Hubey[1994]; d(P,A)=5, d(P,B)=5, d(P,C)=5, d(A,B)=4, d(A,C)=4, d(B,C)=2. We should note that this is not normalized distance as in Hubey[1998] or as in the appendix. So far it looks like B and C should be in the same family since the distance between them is the minimum of the distances amongst the daughter languages. For this case we can also compute the shared retentions between lan-

guages also R(A,B)=1, R(A,C)=1, and R(B,C)=2. However we should note that if we had only the daughter languages and were faced with this problem creating a family tree or reconstructing the protolanguage we would not be able to compute the shared retentions at all. In this case, we can also compute the shared innovations, which is also a kind of distance, I(A,B)=2, I(A,C)=2, and I(B,C)=3. When we create a family, we are making a statement that these languages in the same family are closer to each other than to those outside the family. According to the distances, B and C are closer and hence should be in one [sub]family. Since the number of shared innovations is also considered to be what determines how family trees are constructed, seeing that the number of shared innovations is highest between B and C, we are again prodded to put these two languages into the [sub]same family. Now as the shared innovations distance increases the distance between the languages becomes smaller and thus makes the languages 'closer'. Let us look at another example, where the languages share no features at all as shown below

3.2) $$P = \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \end{bmatrix}, \quad A = \begin{bmatrix} a_1 \\ b_1 \\ c_1 \\ d_1 \\ e_1 \\ f_1 \\ g_1 \end{bmatrix}, \quad B = \begin{bmatrix} a_2 \\ b_2 \\ c_2 \\ d_2 \\ e_2 \\ f_2 \\ g_2 \end{bmatrix}, \quad C = \begin{bmatrix} a_2 \\ b_3 \\ c_3 \\ d_3 \\ e_3 \\ f_3 \\ g_3 \end{bmatrix}$$

Now the daughter languages share nothing, i.e I(A,B)=I(A,C)=I(B,C)=0. The distances are given by d(A,B)==d(A,C)=d(B,C)=7. We note that we should use normalized distances so that the numbers are in the interval [0,1] so that language families can be compared to each other. Suppose now we change one of the features in one of the languages to match another language's feature. For example, change a_2 in B to a_1. Immediately, the distance is decremented by one and the shared innovations is incremented by 1, thus d(A,B)=6, and I(A,B)=1. The same holds for any feature between any pair of languages. Now suppose we have some retentions as below

3.3) $$P = \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \end{bmatrix}, \quad A = \begin{bmatrix} a \\ b \\ c_1 \\ d_1 \\ e_1 \\ f_1 \\ g_1 \end{bmatrix}, \quad B = \begin{bmatrix} a \\ b \\ c_2 \\ d_2 \\ e_2 \\ f_2 \\ g_2 \end{bmatrix}, \quad C = \begin{bmatrix} a \\ b \\ c_3 \\ d_3 \\ e_3 \\ f_3 \\ g_3 \end{bmatrix}$$

Then the distances are d(A,B)=d(A,C)=d(B,C)=5, and the shared innovations still remain as I(A,B)=I(A,C)=I(B,C)=0 but the shared retentions have increased; R(A,B)=R(A,C)=R(B,C)=2. Both the shared retentions and the shared innovations make languages similar to each other. Hence with an increase in SI and SR, similarity increases and distance decreases. Therefore instead of the relationship d(x,y) + s(x,y)=1, in this case (since we are not using normalized distances yet) we have the identity

3.4) $$D(X,Y) + I(X,Y) + R(X,Y) = N$$

where N is the number of distinctive features and S(X,Y)=I(X,Y)+R(X,Y). To make this equation normalized we only need to divide everything by N. If we now change say c_2 in B to c_1, then D(A,B) decreases by one, and I(A,B) increases by one, thus the accounting equation remains valid. We can write this equation as

3.5) $$D(X,Y) + I(X,Y) = N - R(X,Y)$$

The increase/decrease in D(X,Y) results in decrease/increase in I(X,Y). The total number of distinctive features of the protolanguage is constant, therefore if the number of shared retentions changes on the right hand side, there must be a corresponding change on the left hand side meaning that if a shared retention is no longer shared, then it is either a shared innovation so that I(X,Y) increases or it is not in which case D(X,Y) increases. Therefore, we can use either I(X,Y) or D(X,Y) in our determination of family relationships. Of course, if we are attempting to reconstruct a language or create a family tree, we first determine N, i.e. in how many features do the languages in question vary. This is, in effect, the distinctive features space of the language family.Indeed we do not really know how many features the language should have we only notice the number of changes amongst the languages. The only way we can tell what N was would be by comparing this language family to another language family. Let us now consider the last case except without the protolanguage since this is the typical situation.

3.6) $$A = \begin{bmatrix} a \\ b \\ c_1 \\ d_1 \\ e_1 \\ f_1 \\ g_1 \end{bmatrix}, \quad B = \begin{bmatrix} a \\ b \\ c_2 \\ d_2 \\ e_2 \\ f_2 \\ g_2 \end{bmatrix}, \quad C = \begin{bmatrix} a \\ b \\ c_3 \\ d_3 \\ e_3 \\ f_3 \\ g_3 \end{bmatrix}$$

Now it is obvious that the features shared by all the languages would never have been noticed except when comparing this to some other language family. Therefore we would have really only been able to notice

3.7) $$A = \begin{bmatrix} c_1 \\ d_1 \\ e_1 \\ f_1 \\ g_1 \end{bmatrix}, \quad B = \begin{bmatrix} c_2 \\ d_2 \\ e_2 \\ f_2 \\ g_2 \end{bmatrix}, \quad C = \begin{bmatrix} c_3 \\ d_3 \\ e_3 \\ f_3 \\ g_3 \end{bmatrix}$$

This means that the only really relevant part of the equation should be

3.8) $$I(X,Y) + D(X,Y) = M = \text{constant} \quad \text{or} \quad I(X,Y) = M - D(X,Y)$$

if we did know (somehow) which features are innovations and which are retentions. But it seems that this stage has to follow earlier stages in which we first attempt to create families or subfamilies. In that case, just as we can use shared innovations to create a subfamily, we are also forced to use shared retentions to create subfamilies exactly because those languages which have still retained features are still close to each other, after all before the innovations they were all one language. Pronouncements to the effect that shared retentions are not important cannot be correct. All we can see so far is that it is distance that matters, since the shared innovations can be obtained from it. But since distance also reflects shared retentions and shared innovations it also reflects genetic relationships. The only time shared retentions can be important is to determine the direction of the genetic links in time, that is basically all. But all of this is only for pairs of languages and not for a set of languages and so far we have not yet finished the comparative method algorithm. Suppose now, instead of the case in eq. (3.6) in which every daughter shared the retentions a and b, we have the situation as below in which B and C share the retention a and A and C share the retention b but that we do not know which are retentions and which are innovations.

3.9)
$$A = \begin{bmatrix} a_1 \\ b \\ c_1 \\ d_1 \\ e_1 \\ f_1 \\ g_1 \end{bmatrix}, \quad B = \begin{bmatrix} a \\ b_2 \\ c_2 \\ d_2 \\ e_2 \\ f_2 \\ g_2 \end{bmatrix}, \quad C = \begin{bmatrix} a \\ b \\ c_3 \\ d_3 \\ e_3 \\ f_3 \\ g_3 \end{bmatrix}$$

Since we don't know the form of the protolanguage we cannot tell which of these features are retentions and which are innovations. The computations are now D(A,B)=6, D(A,C)=6, and D(B,C)=6. They are all equidistant from each other. The shared innovations have not changed from the previous case so the shared innovations (if we knew them) I(A,B)=I(A,C)=I(B,C)=0 also would tell us that they are equidistant. As can be seen if the shared innovations between any two of the descendants increases it would do so at the expense of the shared retentions thus an increase in one (shared innovations) would imply a decrease in the other (distance). Suppose we have the situation as below

$$A = \begin{bmatrix} a_1 \\ b \\ c_1 \\ d_1 \\ e_1 \\ f_1 \\ g_1 \end{bmatrix}, \quad B = \begin{bmatrix} a_1 \\ b_2 \\ c_2 \\ d_2 \\ e_2 \\ f_2 \\ g_2 \end{bmatrix}, \quad C = \begin{bmatrix} a \\ b \\ c_3 \\ d_3 \\ e_3 \\ f_3 \\ g_3 \end{bmatrix}, \quad D = \begin{bmatrix} a \\ b_2 \\ c_3 \\ d_3 \\ e_3 \\ f_3 \\ g_3 \end{bmatrix}$$

If we did not know anything about the protolanguage, we could not know if the feature a (in C and D) or feature b (in B and D) is an innovation or retention (or indeed parallel development having nothing to do with descent). However, in both cases we can still use the concept of distance to compute distances. Now, even as we see that C and D share feature a as a retention, it is still a fact

that they are closer to each other because of it and hence this closeness still has to be accounted for in the generation of family trees. All we can see is that unless there is parallel convergence, A and B seem to have innovated a_1, but then C, and D exactly because they have not innovated feature a must still be considered to be in the same subfamily. After all, if the protolanguage did not itself constitute a family what is the point of historical linguistics? To see why this must be so, consider some language (which is a potential protolanguage since every language is). Let the number of features of some protolanguage be N. Split the speakers of this language into M groups physically so that any innovations will not spread across the protolanguage but will only remain withinthe isolated group. Now at this point in time, these languages(!) still constitute a single language because they still retain (!) all the features of this language. Of course, unless the speakers of this language fell from the sky, it split off from another language so that these shared retentions are then shared innovations from the perspective of the sisters of the protolanguage. At this point how can we tell that these languages constitute a language family (and they do) if we look only for shared innovations in the restricted (and common sense) in linguistics? Obviously there are no shared innovations and thus we'd be forced into the ridiculous conclusion that they are not related. This is essentially the reason why Ringe's method [1992,1995] would produce a ridiculous result that English is not genetically related to English [see Hubey[1999a]]. But if we stick to the general (and correct) reasoning that distance (or its inverse similarity) determines genetic relationships, then we will have no problem at all in determining that these languages are indeed a family (i.e. the same language).

There is probably some resistance or incredulousness among those who are not trained in any of the mathematical sciences to say that a language is really a language family. However, problems of this type occurred many centuries ago, and such obstinacies were all given up eventually. For centuries zero was not considered to be a number. How is it possible for something that does not exist to be anything at all? But zero is a number, and so are negative numbers and imaginary numbers. If not, then students get confused, for example, with the fact that "velocity is constant" also applies to bodies at rest (i.e. velocity zero). It does not mean "velocity does not exist" which some students attempt and get confused as a result. It means velocity is zero. Exactly similar reasoning applies in logic and with which students (and not only students) also have many problems. For example, a statement such as "all pink elephants taller than 100 feet are sterile" is usually thought to be a false statement. However in logic, this is a true statement. It is so, because no such elephant can be found, and if there is no counterexample to an assertion, the statement cannot be shown to be false. In bivalent logic, whatever is not false is true, therefore something that cannot possibly be false, then must be true [see for example, Hubey[1999]]. This is also the reason why 'falsificationism' (vigorously defended by the philosopher Popper) is now at the root of philosophy of science. That is also why a method that claims to determine geneticity of two languages can and must be submitted to a test in which we can see if it can determine whether the two languages are really the same language. In the same way, a language is itself a language family since if n languages can constitute a language (sub)family, so can 1 language. All the features of this single-language family are retentions, and distances are zero. At the same time all of these shared retentions are shared innovations which differentiate it from its sister languages. If all we have are some languages which we think are genetically related, then all we have is distance, and it was distance that was used in Hubey[1998]. Suppose now that the situation after some time, where N=7 and M=7 is as below

$$A = \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f_1 \\ g_1 \end{bmatrix}, \quad B = \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f_2 \\ g_2 \end{bmatrix}, \quad C = \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f_3 \\ g_3 \end{bmatrix}, \quad D = \begin{bmatrix} a_4 \\ b_4 \\ c_4 \\ d_4 \\ e_4 \\ f_4 \\ g_4 \end{bmatrix}, \quad E = \begin{bmatrix} a_5 \\ b_4 \\ c_5 \\ d_4 \\ e_5 \\ f_5 \\ g_5 \end{bmatrix}, \quad F = \begin{bmatrix} a_5 \\ b \\ c_5 \\ d_6 \\ e_6 \\ f_6 \\ g_6 \end{bmatrix}, \quad G = \begin{bmatrix} a_5 \\ b_4 \\ c_7 \\ d_6 \\ e_6 \\ f_6 \\ g_7 \end{bmatrix}$$

Would anyone seriously argue that the set of languages {A,B,C} should not be put into a subfamily because they share no innovations? What they have are shared retentions, but according to some linguists (who have tried to import the cladistics idea --itself a fuzzy concept-- from biology), these features should not be used to create (sub)families. They certainly do constitute a (sub)family, because they are still a lot like the protolanguage because they still retain 5 out of the 7 features of the protolanguage. Is it possible to argue that a language (e.g. the protolanguage) is not genetically related to itself? This is the same problem as in Ringe method; can a language not be genetically related to itself? Now, the rest of the languages do have shared innovations so those can be used to determine their relationship to each other, but in both cases we see that it is really the degree of similarity (or distance) that we use for genetic determination. The innovations and retentions would, of course, be used in the determination of what the protolanguage looked like, so that if the family tree is a digraph (directed graph) then the retentions and innovations will be used to determine the direction of arrows (and thus time). It is easy to produce a tree from distances between languages. The example from IE from Hubey[1998] is repeated below in Figure 2. It was left this way purposefully because the knowledge that is necessary to produce a family tree where we note ancestry is not available in the features, but additional information and assumptions are necessary.

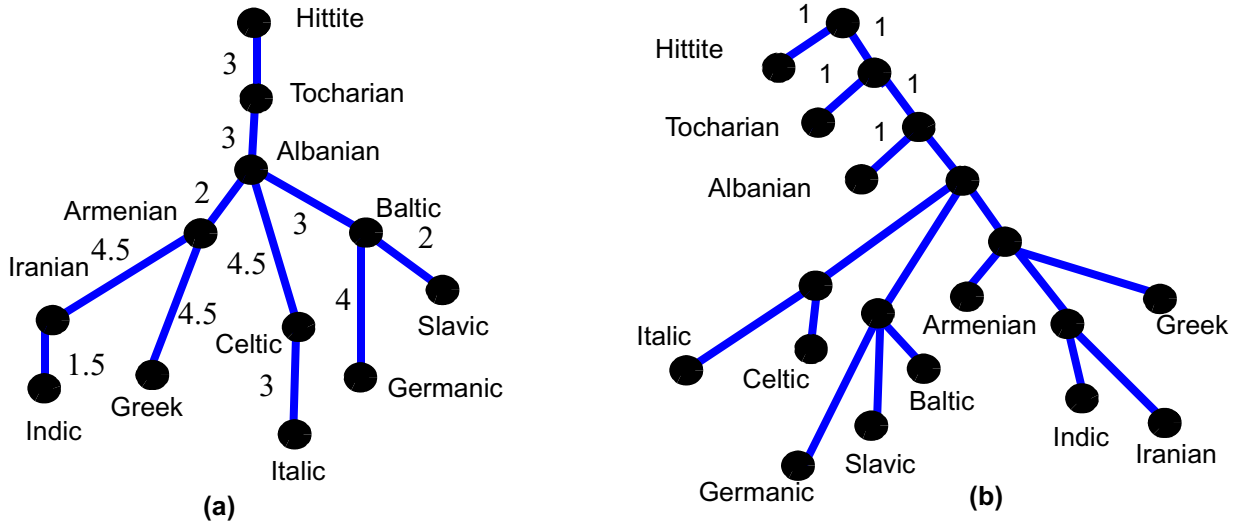


Figure 2: Trees of the Indo-European Family: (a) The Minimal Spanning Tree, (b) a Branching Tree. Note that no node has been identified as a "root" in the spanning tree. Since in graphs only connectivity matters, the graph (a) is equivalent to the one in Hubey[1998]. To create a branching family-descent tree more assumptions or rules in the heuristics are needed. Such a branching tree should resemble something like that in (b). Knowledge outside of linguistics was used to select Hittite near the root of the tree.

4. Regular Sound Change

The only heuristic which historical linguists seem to be able to agree upon is that of "regular sound change". How valuable is it really, and how significant is it? The detailed and rigorous answers and a research programme can be found in Hubey[1999a]. Here simple analogical arguments will be used. First everyone is familiar with the Birthday Problem, that of determining the probability that at least two persons will have the same birthday among a group of persons of size n. The answer is that the odds are approximately 50-50 that at least two persons have the same birthday when n=23. Similarly if we find 23 putative cognate sets, and use only about 19 consonants (so that there are about 365 possible sound changes) then the odds are again approximately 50-50 that at least one of those sound changes will be repeated and hence will constitute a regular sound change. With say 200 putative cognates many of the sound changes will be repeated, therefore the regularity of sound change is really a side effect of quantity. The case in which the heuristic is very significant is when the number of comparanda is small. The use of rigorous mathematical methods (e.g probability theory) will prevent the use of needless argument over heuristics. However it is possible to use the concept "regular sound change" without the explicit use of probability theory and as a part of the determination of "distance". Intuitively, we can see that a high degree of regular sound change is something whose probability of occurrence is lower than that of many different kinds of sound change, thus it is a heuristic that is in some ways a substitute for probability theory. It has been used as a type of distance in the works of Raman[....] in which the number of phonological changes required to change a word into another word is used as a distance metric. This is simply another way of determining the Hamming distance since it would take that many changes to make two words identical. It can be used in conjunction with the concept of distance in two ways.

(I) First, the number of regular sound changes can be used on one of the sets of the comparanda before the distances are computed, and used to modify the distance computation. For example, suppose that we are comparing three languages X,Y,Z. Suppose further that the distances amongst the languages according to some metric are equal. If, however, after making some sound changes in one set, say according to N sound change rules, d(X,Y) is much less than d(X,Z) we can use this to infer that X and Y should probably be placed in a subfamily.

(II) Second way in which regularity of sound change would be to use some phonology space (see for example, Hubey[1999]) to create distances amongst sounds so that the metric is better suited than the simplest metric in which each sound change in a word counts equal whereas we might want to assign higher value for the change p>h than p>b using a heuristic that p→h was probably realized something like p→k→x→h. The latter would take longer to accomplish and thus should automatically imply a longer time horizon.

(III) Third way we can use regularity of sound change would be to further refine the first in using the number of regular sound changes to modify the distances computed. In this we would use (a) the principle that regular sound changes imply greater affinity and therefore those pairs of languages that can be brought closer together after regular sound changes should be considered to be more closely related than those without, and (b) those pairs of languages that can be brought closer together with a smaller number of regular sound changes are displaying a closer affinity

than those that require a larger number of sound changes. All of these are based on a simple assumption that longer time periods create greater distances amongst languages, especially if they are geographically separated. An example of some distance metrics is given below in Table I.

Table 1:

	Distance or weighting description	Case 1	Case 2	Case 3
	Suppose we find these changes in a small sample of text. How shall we use a signal (phonetic/phonological) distance that is harmonious with linguistics sense.	pan→man pas→mas pat→mat par→mar	pan→man pas→bas pat→wat par→far	pan→gan pas→ras pat→kat par→dar
H_0	Every sound change is equal i.e. each sound change gets 1 pt.	4	4	4
H_1	Number of distinct sound changes	1	4	4
H_2	Additive Combination; H_0+H_1	5	8	8
H_3	Multiplicative Combination; H_0*H_1	4	16	16
H_4	Each different sound change gets a weight *	1	4	>4
H_5	Additive Combination; H_0+H_4	5	8	>8
H_6	Multiplicative Combination; H_0*H_4	4	16	>16
	** The changes {b →m, b→p, b→w, b →f} in Case 2 all involve bilabials and probably are smaller scale changes than the set of changes {b→g, b→r,b →k,b→d}. For example some of these changes, if they really were endogeneous changes, might have gone thru stages such as p→p^h→f→h→0, or p→t→k→g and might be considered to be indicative of temporal distance.* These distances are only suggestive. They should also be normalized.			

There are other considerations as well in Table I; in order to be able to use a weighting scheme as in H_4 we need to know what phonemes were the earliest developed and these can go back as far as hundreds of thousands of years. Since we do not have access to such knowledge, we might make yet another assumption. We might borrow another concept from biology; the earliest sounds in use among humans might have traveled the same route as that children follow when learning their language's phonemes. Furthermore phonemes that are observed in almost all of the world's languages probably have been in existence the longest. These are /ptksn/, and the vowels /iua/ exist in almost all of the world's languages with the possible exception of Kabardian. The reasons for the development of these particular sounds is easily explicable (see Hubey[1994] or Hubey[1999b]). They are merely ways of dividing up the available phonological space so that they are maximally distinct. For example, the earliest sounds infants develop is a centralized lax vowel like an [a] and a bilabial like [p],[b] or [m]. The vowel can be produced when the articulators and particularly the tongue is in a relaxed position. The bilabial is visible, and is probably something the infant notices when people are talking.

5. Reconstruction and Geneticity

There is an often repeated refrain that geneticity cannot be determined without having reconstructed the protolanguage. Let's indicate ith sound change from the set of basic words of a daughter language, D_1 to its protolanguage L_P, by P_i^d i=1,2,3,..n. Therefore if we apply all of these relations one after another, we can then transform the words of D_1 in the list to the words of the other (presumably related) language. Since relations are associative (see appendix), we can denote the composition of all of these changes (relations) as P. Suppose similarly that relation Q relates a second daughter language D_2 to L_P. This situation can be seen in the diagram on Figure 3. Under the neogrammarian assumption, we should be able to obtain the original words of the protolanguage if the relation is invertible. Therefore, we can then attempt to reconstruct the protolanguage from a set of such relations. It should not be a surprise, therefore, that we should be able to find that the changes from language D_1 to D_2 are also regular. In other words $L_P=P(D_1)$ and $L_P=Q(D_2)$. If both relations are invertible then $D_1=P^{-1}(Q(D_2))$ and $D_2=Q^{-1}(P(D_1))$ where Q^{-1} and P^{-1} are the inverse relations. Intuitively it is clear why this must be so (within some practical limits). After all, the protolanguage itself is reconstructed from the daughter languages, therefore the sound changes, and the putative sounds of the original protolanguage are also derived from the raw data in the form of the daughter languages. Furthermore, there must at least be a tacit acceptance of the existence of the family before reconstruction is attempted since everything about the protolanguage depends on the languages which are accepted to be daughter languages. We should note that the relations P and Q are the actual sound changes from the daughter languages to the protolanguage. However, the general and abstract relation "regular sound change" (i.e. a rule based set of changes) is a symmetric relation. If there are a set of rules that that we can use to change one set of words to another, then there are rules that will allow us to change in the other direction. It should be noted that this concept is not about the real and practical aspects of reconstructing a language. The description is about a situation in which we were in possession of such

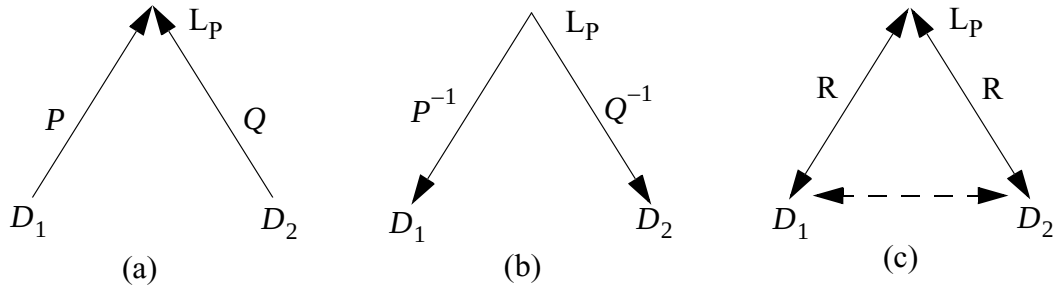


Figure 3: Regular Sound Change Relations: The 'regular sound change' relation is symmetric and transitive as in (c).

rules. Obviously, reconstruction is a manifestation of a particular set of rules that has been deduced from the evidence of daughter languages. Thus it is merely another aspect of the relationship of the daughter languages to each other. Therefore in considering only the relation of "regular sound change", it is clear that it is a symmetric relation as in Figure 3.c. Therefore the relation, R, (regular sound change) is also transitive (dotted lines in Figure 3.c). As is obvious, there is no reason in principle why geneticity calculations and algorithms cannot be based on the putative daughter languages, as indeed is done in real life applications of the comparative method.

6. The Comparison Algorithm

Nobody really knows how the human brain manages to spot resemblances or how it stores them although there are theories. The closest we have come to modeling the functioning of the brain seems to be in the workings of artificial neural networks. But it is not necessary for computer programs to duplicate the functioning of the human brain. Typically, if one is reading about a list of words of closely related languages or dialects, the semantic distances are already zero or close to zero, and the signal distances are also small. Then the algorithm to be followed by the reader is that which will confirm or verify that the phonological distances are small.

MinDistance =0
TotalDistance =0
FOR μ*=1 To LastWord DO*
    Distance= $d_p(M_\mu{}^a, M_\mu{}^b)$
    TotalDistance= TotalDistance + Distance
NEXT μ

First we see that we might have done the calculation as 'TotalDistance= TotalDistance*Distance' instead of additively. Furthermore, both the additive and the multiplicative versions as specific examples of a distance metric between the two lists; they are merely the simplest versions. In this case, we do not really have to do any comparisons since this this list presumably has already been prepared for the reader and already passed the test. There could be a confirmatory test by using an idea due to Oswalt [see further below]. One can do a modulo shift of one of the lists and then recompute the distances and verify that for each shift the TotalDistance is greater than for the correct matching. If we wanted to test two languages, we would first have to create a list of comparanda, and in the creation of this list we would be guided by the ideas discussed, mainly that of the 'basic vocabulary' list. If we are given many potential words for every word in the BV, then we would have to have the computation include the selection of the words for the list while we are also testing the words. One way to proceed would be to first go through the list and compute some distance metric for the list from the first element in each set of putative cognates. For example, if for the first semantic concept we have n words from the first language and m words from the second language, we would compute a distance metric from the first word in each list. We would do the same thing for each semantic concept on the list. This is easily accomplished by the algorithm

GetFirstWordInEachList;
FOR SemanticConcept=FirstConcept to LastConcept DO
    ComputeDistanceBetweenTheFirstWords;
    D[SemanticConcept]= StoreDistance(SemanticConcept);
    TotalDistance=TotalDistance + Distance
    GetNextWordInNextSemanticConceptInEachList;
NEXT SemanticConcept

After this TotalDistance is computed, we assign this to a MinimumTotalDistance and then permute the words in the list and compute new distance for each semantic concept. If this new distance for this semantic concept is less than the previous distance for this semantic concept, we

substitute this distance for the older one, subtract the old distance from the MinimumTotalDistance, and add the new distance to the MinimumTotalDistance.

```
MinimumTotalDistance=TotalDistance
FOR SemanticConcept=1 to LastConcept DO
    REPEAT
        GetNewWordInSemanticConceptInList1;
        REPEAT
            GetNewWordInSemanticConceptInList2;
            X=ComputeNewDistance(WordInList1,WordInList2);
            IF (X<D[SemanticConcept]) THEN
            {MinimumTotalDistance=MinimumTotalDistance+X-D[SemanticConcept];
            D[SemanticConcept] = X;}
        UNTIL (NoMoreWordsInSemanticConceptInList2)
    UNTIL (NoMoreWordsInSemanticConceptInList1)
NEXT SemanticConcept;
```

It should be noted that this works after some preprocessing. In other words, the basic semantic concepts must already be in place, and the possible cognates for each semantic concept have already been listed, and most importantly each such word is assigned to only one semantic concept slot. If that is not the case, then we need to check the solution to make sure that the same word has not been used twice in the minimization of total distance. In the case that words have been assigned to multiple semantic concept slots, we need another algorithm, one that checks every possibility. In this case we would still start off with some TotalDistance which would be quite arbitrary, and then we would go through every permutation of the words to minimize a total distance. A brute-force approach to this problem would be;

```
MinimumTotalDistance=TotalDistance
FOR SemanticConcept=FirstConcept to LastConcept DO
    REPEAT
        GetNextWordInSemanticConceptInList1;
        REPEAT
            GetNextWordInList2;
            X=ComputeNewDistance(WordInList1,WordInList2);
            IF (X<D[SemanticConcept]) THEN
              {MinimumTotalDistance=MinimumTotalDistance+X-D[SemanticConcept];
              D[SemanticConcept] = X;}
        UNTIL (NoMoreWordsInList2)
    UNTIL (NoMoreWordsInSemanticConceptInList1)
NEXT SemanticConcept;
```

We should note that for each semantic concept, all the words in that semantic concept in List1 is compared to all the words in List2, so that eventually, everything in List1 will be compared to everything in List2 and the minimum distance between the lists will be computed. The heuristic of regular sound change can obviously be exploited here in the computation of distance as shown in the section regular sound change. Nothing specific about the distance computation is given with

the exception that the total distance is additive which can obviously be changed to multiplicative or any other suitable form. The intuitive application of the regular sound change heuristic accomplishes what the last algorithm describes. What is left to do now is to decide if the result of the algorithm can be due to chance. There are many ways of doing this, none of them perfect. The problems attending such a task have been explained in other places, for example Hubey[1999a], Embleton[1991], Oswalt[1991], Ringe[1992,1995]. Ringe's attempt is discussed elsewhere[Hubey,1999a]. Here a short discussion of Oswalt's attempt follows. The problem Oswalt attempts is a heuristic that compares the result of the best matching of the list (the result of formal or informal application of the above algorithm) to what he calls the 'background score', what would have been obtained when random matchings are made, a problem which Ringe [1992,1995,1998] attempted. Oswalt essentially computes the crosscorrelation of the the two lists. To see how it would work, suppose we have two sequences of numbers of the type [-1,1,1,1,1-1,-1,-1,1,-1,-1,....] denoted by x_i and y_i. We compute the crosscorrelation of these sequences as

$$R_{xy}(k) = \sum_{j}^{\infty} x_j y_{j+k} \tag{6.1}$$

The upper limit has been purposefully set to infinity. For applications in real life where the data is always finite the results will only approximate the conclusions derivable from the infinite case above. Let us consider a simpler case in which x=y so that we are computing the autocorrelation. Then we can see that if the numbers are random, the products will always be +1 or -1 and the total will probably be zero for each k in R(k), with the exception of R(0) which can never be zero since the products are always (+1)(+1) or (-1)(-1) so that for an infinite sequence R(0) is infinite. For a finite case R(0) will probably be greater than every k. This is essentially Oswalt's test. Since the data is finite the upper limit for k is N-1 where N is the number of pairs being compared. The comparison is done modulo N thus we should write

$$R_{xy}(k \mod N) = \sum_{j}^{N-1} x_j y_{(j+k) \mod N} \tag{6.2}$$

Therefore if there is any determinism in the putative cognate list, we should obtain a better score (Oswalt uses similarity, not distance) for R(0) than for any other case. In the shift test, one can easily compute the maximum possible value, and thus compare other results to this maximum. The autocorrelation can be used as a test of randomness[Hubey,1997].

7. Summary & Conclusions

A clear and reasonably thorough although mostly intuitive explication of the comparative method has been given. A more rigorous discussion would have to make use of probability theory, statistical testing and stochastic processes. Statistical methods including generalizations of lexicostatistical and glottochronological ones can be found in Embleton[1991]. It is clear that Oswalt's [1991] idea of testing the result of the putative cognates against 'background' matching is much better suited for the task than Ringe's[1992,1995] attempts whose faults are shown in Hubey[1999a]. Baxter's criticism[1998] of Oswalt's shift test [1991] can be overcome quite easily by incorporating regular sound change into the distance metric as shown in this paper.

Appendix A: Algorithms

An algorithm is a set of step by step instructions for accomplishing a specific task. In the age of digital computers, the algorithmic way of thinking has become even more important than in the past. However, it has not become any easier, for like other branches of mathematics, algorithmic thinking is no more natural than, say, probability theory or even logic. Most people are so accustomed to simply carrying on with their lives without paying much attention to how they do what they do, that they find it difficult to produce algorithms for things they do every day. One of the problems is that it does not even occur to them that some things really need an algorithm. Algorithms are generally written using special kinds of languages (pseudo-code) or special types of symbols. Among the various symbolisms are flow charts, Nassi-Schneiderman charts, syntax charts, structure charts. In this paper pseudo code will be used because it sits partway between natural language and procedural computer languages. It is well-known that programs can be written using only three kinds of structures; (i) sequence/statement, (ii) IF...THEN and (ii) LOOP. The IF-THEN construction is generall of the form

```
        IF (condition)  THEN
A.1)            Perform_X
        ELSE
                Perform_Y.
```

The condition is some kind of an expression that computes to true (T) or false (F). If during runtime, the condition is true, then Perform_X is executed, and if the condition is not true then Perform_Y is executed. Perform_X (and Y) is a set/block of statements which might contain IF...THEN statements. Probably the three most comon loops are (WHILE...WEND), (REPEAT...UNTIL), and the (FOR...NEXT). The While loop is a top-test loop therefore the loop may never get executed. The form of it (in Basic-like syntax) is

```
        WHILE (condition)
                Statement
A.2)            .....
                Statement.
        WEND
...
```

When the While statement is executed, the condition is tested, and if it is true the loop is entered. The statements in the loop are executed. When the end of the loop (WEND) is encountered the execution goes back to the top (While statement) and the condition is tested again. As long as the condition is true, the body of the loop is executed repeatedly. The Repeat statement gets executed at least once because this particular loop is tested at the bottom. The syntax is of the form;

```
        REPEAT
                Statement
A.3)            ...
                Statement
        UNTIL (condition).
```

In this loop, the condition at the bottom is tested after the loop has been executed at least once. Thus the condition in this loop is really a terminating condition and not a continuation condition as in the While loop. The For loop is an indexed loop. It is very useful if we want to repeat a set of operation a known number of times. Its syntax is of form,

```
        FOR index=Begin To End DO
                Statement
A.4)            ...
                Statement
        NEXT index
```

The variable index is initialized to the value Begin, then the loop is executed. When the bottom of the loop is reached (NEXT statement) then the execution goes back to the top, the index is incremented, and the loop repeated. The loop is repeated until the value of the index reaches the End value.

Using these constructs we can write a short algorithm, for example, to change every initial letter of a set of strings (a string of letters, i.e. a word) to the letter ‘A’. We can write this

```
        Get/Read Word;
        WHILE (there are more words)
A.5)            Change first letter to A;
                Get/Read Word;
        WEND
```

Appendix B: The Measurement Problem

Dimensions/units: Aside from the basic ideas that we have all seen in high school math there are other important ideas. For example, we often need to make things comparable to each other. That is most easily done if we use numbers. Before we can even do that we have to make sure that the objects that we deal with are *quantifiable* in some way and that we can measure them (with number naturally).

It turns out that this problem often-stated as "you cannot add apples and oranges" has many facets. One is that the things we measure in physics (and hence engineering) come in fundamental dimensions. For example, dimensions of that particular branch of physics called mechanics consists of M {mass}, L {length}, and T {time}. These dimensions are measured in units (which are almost completely arbitrary). For example length can be measured in inches, feet, centimeters, kilometers, nanometers etc. Time is measured in seconds, hours, days, years etc. The MLT system can also be changed to FLT (i.e. force instead of mass because of Newton's famous law of physics F=ma). In general systems of units are chosen to make calculations of physical equations simple. That is the case with the SI (System Internationale). For electrical phenomena we need one more dimension, Q (charge), and for thermal phenomena we need θ (temperature). Unfortunately, such a system of dimensions do not seem to exist for any other science except those that can be derived from physics, such as chemistry, biochemistry, biology etc. However, strangely enough, although we can multiply quantities of different units (and dimensions) to get other quantities of different dimensions, we cannot add quantities of different dimensions. We can see the same problem in economics in which we cannot add the dollar cost (nominal prices) of objects across time, even if they are all nominally dollars. Instead we add their real values (which we compute by taking into account other factors such as inflation rate).In physics these ideas are bound up within the concept of dimensional analysis. Dimensional analysis is a method of reducing the number and complexity of experimental variables which affect a given physical phenomenon, using a sort of compacting technique. If a phenomenon depends upon n dimensional variables, dimensional analysis will reduce the problem to only k dimensionless variables, where the reduction n-k=1,2,3 or 4 depending on the problem (phonological space)

Dimensional analysis has other side benefits. One is the savings in money spent on experiments. The other savings is that it provides scaling laws which tell us how things should change in shape as their size changes. The method is based on the *Principle of Dimensional Homogeneity* which says that any equation that expresses a proper relationship between the variables of a physical process or system will have each of its additive terms possess the same dimensions. It is based on Rayleigh's "method of dimensions" in (Theory of Sound,1887).

Mass and Surface Area

The mass of an animal grows proportional to L^3 but its surface area is only proportional to L^2. This has great effect on several things. First, as animals get larger they have to have thicker cross sections of bones to support all that weight. Second, for falling objects, the drag force is proportional to surface area but gravitational attraction is proportional to mass so that larger objects have higher terminal velocities. For an average man, the terminal velocity is about 120 mph which pretty much guarantees that he will sustain a great deal of damage from falls whereas for an insect the terminal velocity is so small that nothing will happen if thrown out of an airplane high up in

the sky. Friction is also proportional to velocity so that although we can move through water with great ease at small velocities, the drag is too large for very small increases in velocity so that a fast walk through water chest deep is virtually impossible. However, because of the same reasons, a large animal can shake off the water after a dip in the ocean but a small insect will be weighted down by a very large weight of water sticking to it in the form of a water droplet. Many other things having to do with scaling of living things such as metabolism, oxygen consumption, heat exhaustion, cooling etc. can be found in Schmidt-Nielsen [1984].

Normalization

In addition to problems of different dimensions, we try to change numbers into those in some standard range so that we can compare them more easily. It would be even better for practical purposes to put numbers into a standard interval [0,1]. This is essentially what is done in probability theory, fuzzy logic and other fields.

Example *Normalizing Grades on Tests*

One way to normalize test grades is simply to divide every grade by the highest grade in class. This guarantees that the highest grade in class is 1.0. If any student received a zero for a test, it will still remain zero. The easiest way to grade tests is actually in binary; we just note if they passed or failed.

Example *Boxing normalization*

In order to be able to compare one boxing match to another a standard scoring system is used in which the same number of referees are used to score the bout, and for each round at least one boxer must be given 10 points.

Extensive-Intensive Variables

It is often remarked in narratives that a fundamental difference exists which can be characterized by the words **quantitative** vs. **qualitative**. Often what is meant by the word qualitative is "intensive" instead of “extensive” since concepts often characterized as a quality can also be quantified. A **state** of a system is characterized by a set of parameters. If we split this system in half some of the parameters will obey $X_1+X_2=X_s$ and others will obey $x_1=x_2=x_s$. The former (upper case) are extensive parameters, and the former intensive parameters. If a system consisting of a lot of 10,000 TVs is split into two sets at random, the quality of the two subsystems will equal each other and also the quality of the TVs of the whole original system. In a socio-economic system, inflation rate is an intensive variable, where as total consumption is an extensive variable. In a language, the lexicon is an extensive variable, whereas the typology, morphology, and syntax are intensive properties or quantities.

Example *Gymnastics & Diving*

In gymnastics and diving the scoring system is based on 10, but the total score is obtained by multiplying the *raw score* by the degree-of-difficulty (an intensive parameter) so we have something like

B.1) $$S_{total} = S_{raw} \cdot D_{difficulty} .$$

Example *Combined Boyle's Law and Charles' Law*

These laws are $p_1V_1/T_1=p_2V_2/T_2$=constant which can also be written as pV=nRT. These equations display another characteristic of systems in that the product pV is comprised of an *intensive* and an *extensive* parameter which happens often in nature.

Example *Brain and Body Mass*

One way to make different animals comparable is to compare not their brain capacities but the ratio of their brain mass, b, to their body mass B. When this is done there is an almost straight line (log-log plot). An equation of type $\ln(b)=\alpha\ln(B) + \ln(\gamma)$ is really one of form $b=\gamma B^{\alpha}$ and many things in nature seem to obey the *power laws*.

Different kind of normalization

Suppose we want to represent physical agility or physical capability of athletes from various different tests. Suppose we only use three tests; (i)endurance/stamina; (ii) reflex, reaction-time, and (iii)strength. How should we represent these three qualities (as quantities)? As the simplest such measure we can simply make three separate bits (i.e. zero or one) which will represent the possession or lack of the relevant property (such as a pass/fail grade) which we can write as 000,001,010,011,100,101,110, and 111. Or we can decide to give them grades in the normalized interval [0,1] for each of the three separate tests. Of course, we can easily increase the number of such tests to five or ten, and we can also increase the dimensionality of the problem but plotting more than 3 dimensions is very difficult. Hence, it is easy to deal with such high dimensional problems using only symbols and logic. To continue the example of 3 dimensions, we can make bar charts, pie charts or we can plot them on a 3-dimensional graph. Then we can represent each person as a point in three dimensions {x,y,z}. We call such ordered **n-tuples** or **vectors**. A vector is obviously a simpler case of a matrix. It is a 1 by n matrix.

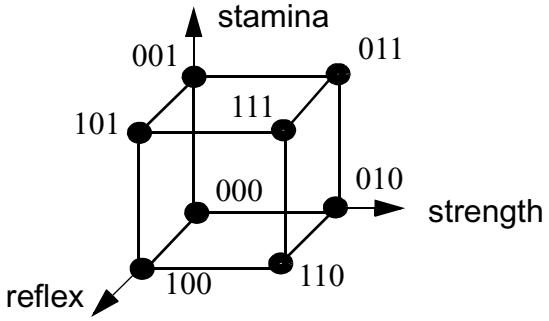


Figure B:1: Pass/Fail Physical Agility Space: We can define physical ability as some kind of a linear combination of the three factors. This is another representation of parallelism or simultaneity in (phase) space.

Example *Color Space*

A perfect example of a three dimensional vector of cognitive science is color. As we know all the colors (for all practical purposes) can be obtained (additively) from the three basic primaries, Red, Green and Blue, RGB. The gray scale runs from black to white along the diagonal. The great advantage of using multiple dimensional space is the accuracy of such representations of much phenomena. We all know what colors are but they would be virtually impossible to

explain to someone who was congenitally blind. If we did attempt to "explain" colors by explaining that "black is the absence of color and white is a mixture of all the colors" it is likely that the blind person would think of colors as what we call "gray scale".

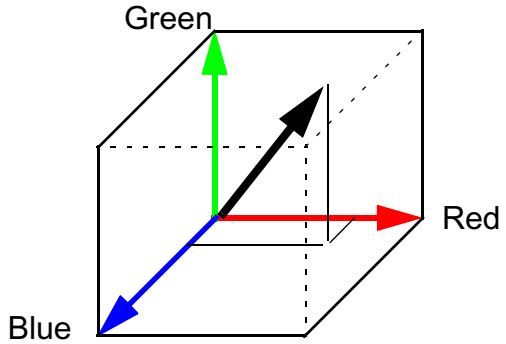


Figure B:2: Color (Vector) Space: All the colors can be created from the so-called primary colors by additive mixing. We can think of colors, therefore, as vectors (n-tuples, or arrays) in color space.

The primary colors are vectors:

B.2) $$\mathbf{r}=\begin{bmatrix}1\\0\\0\end{bmatrix};\ \mathbf{g}=\begin{bmatrix}0\\1\\0\end{bmatrix};\ \text{and}\ \mathbf{b}=\begin{bmatrix}0\\0\\1\end{bmatrix}.$$

Since a vector consists of ordered elements, the first entry refers to redness, second to greenness and the third to blueness. Thus the red vector **r** has only a 1 in the redness-place and zeroes elsewhere. Similarly for the other primary colors, **g**, and **b**. We suspect, then, that the other colors will be some combination of these primary colors. What this boils down to is that we want to add different proportions of the primaries to create other colors so that we will multiply the primary colors by some number less than one (so that it is a small proportion) and then add them all to get some other color $\mathbf{c}_{any}$, so that

B.3) $$\mathbf{c}_{any}=p_r\begin{bmatrix}1\\0\\0\end{bmatrix}+p_g\begin{bmatrix}0\\1\\0\end{bmatrix}+p_b\begin{bmatrix}0\\0\\1\end{bmatrix}$$

where p_r=proportion of red, p_g=proportion of green and p_b=proportion of blue. If we had $p_r=p_g=p_b=0.5$ we will obtain a gray since the diagonal of the color space that runs from black to white is called gray-scale. We can represent this particular gray as

B.4) $$\mathbf{c}_{gray}=0.5\begin{bmatrix}1\\0\\0\end{bmatrix}+0.5\begin{bmatrix}0\\1\\0\end{bmatrix}+0.5\begin{bmatrix}0\\0\\1\end{bmatrix}=\begin{bmatrix}0.5\\0\\0\end{bmatrix}+\begin{bmatrix}0\\0.5\\0\end{bmatrix}+\begin{bmatrix}0\\0\\0.5\end{bmatrix}=\begin{bmatrix}0.5\\0.5\\0.5\end{bmatrix}$$

Matrices are also called tensors of rank 2, and vectors are tensors of rank 1. Therefore the ordinary single numbers are called tensors of rank 0 or simply scalars. In the example above we saw the rules for scalar-vector multiplication and vector addition, but not vector multiplication. The final result for this particular gray is that it has 0.5 proportion of red, green and blue since those are the vector components.However, if we do make an analogy to the 3D space in which we live with the exception that the dimensions of color are not homogenous like our space dimensions, it is more likely to be understood better. For a more detailed look at color, see Hubey [1997].

Accuracy and Precision

There is usually no thought given to the possibility of measuring something accurately but not precisely, or precisely but not accurately.

Example *Significant Digits*

Suppose we want to compute the area of a rectangle with width w and length L, but the measurements are not and can never be to infinite precision but have errors in them as shown. Therefore our calculation is really $A = (L \pm \Delta L)(W \pm \Delta w)$ instead of A=Lw where ΔL and Δw are the errors with which we measure L and W.

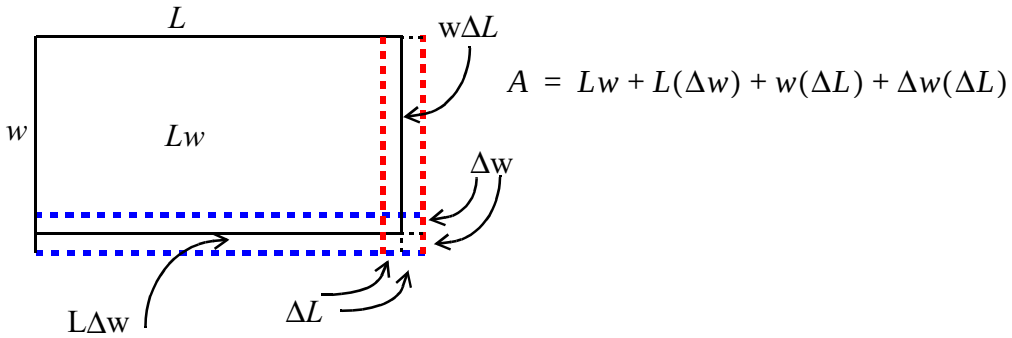


Figure B:3: Errors in Measurement: By choosing A=Lw we mean that we drop the other small terms which are small error terms that we cannot get rid of.

Therefore $A = Lw + L(\Delta w) + w(\Delta L) + \Delta w(\Delta L)$. If the error is about one tenth of the actual size that is measured, then the last product is about one-hundredth and can be dropped introducing no more than an error of one-hundredth in the final answer for the area. Dropping the other terms will introduce an error of about ten percent. As an example, suppose we measured L=5.2 and w=3.1232123432. The product is 16.2407041846 but all of these digits are not significant in the sense that they are a part of the error since L has not been measured to any more accuracy than 2 digits. For example the error in L can be on the order of 0.1, therefore L could in reality be between 5.0 and 5.2. If we use these then we see that 15.616061716 < A <16.553025419. Therefore since the lowest precision number is L, and since it only has 2 significant digits (i.e. the 5.2 since the last digit could be error) then the answer A is only significant to two digits; therefore A=16.0 (correct basically to two significant digits) and which in a sense is an average since it is something like the median value in the interval [15.6,16.5]. This number is approximately equal to the average of the upper and lower limits of A whose first few digits are is 16.08.

In simple terms, when we make computations such as finding the area of a piece of land we should make sure that our calculated answers do not give the appearance of being better than they are. The resolving power of our tools is and must always be greater than the reach of our concepts. It is easy to know the resolution of our physical instruments and thus perform error analysis. However, in the non-physical sciences we are forced to create different kinds of instruments with which to measure things. There are different types of problems associated with social sciences.

Example *Paleontology*

In paleontology one is often required and forced to make deep claims on the basis of partial bones of long-dead ex-living things. In order to find some regularity one is forced to take into account basically shapes and size of bones from which conclusions are obtained about the species. Obviously, precision and accuracy are highly correlated in these eye-balling measurements. Judging from the tremendous variation in size and shape of a single living species such as dogs, one should be very careful announcing that a given set of bones belongs to a different species of hominids. There are two species of monkeys for which the differences in skeletons cannot be ascertained. There are other species (for example, dogs) whose sizes vary greatly. Therefore much of the *lumper vs. splitter* arguments are probably not decidable from the fossils since the precision of the instruments and theory is insufficient to make such determinations.

Reliability and Validity

So habituated are we to measuring things in this modern age that we scarcely give thought to the possibility that what is being represented as a number may be meaningless. That was the case in almost all the examples given above which were mostly from the physical sciences with analogical extensions to other fields. In measurement theory, we use the terms *validity* and *reliability.* Validity of the measurement is that the metric actually measures what we intend to measure. In physical measurements there is usually no such problem. Validity also comes in different flavors such as construct-validity, criterion-related validity, and content-validity. Reliability refers to the consistency of measurements taken using the same method on the same subject.

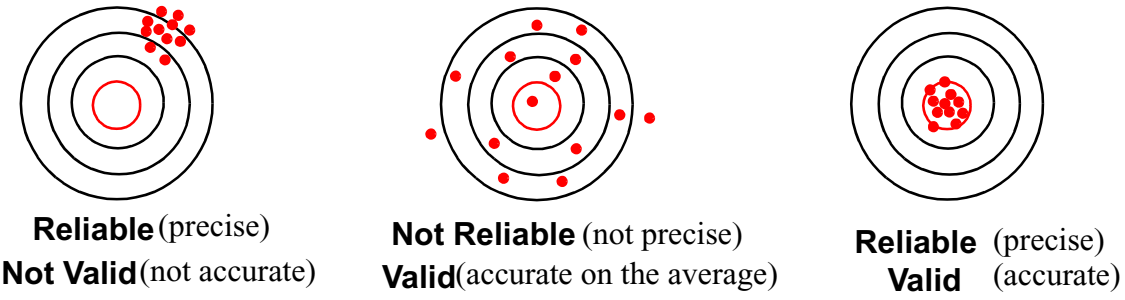


Figure B:5: Reliability and Validity Analogy: One normally expects accuracy to increase with precision. However in the social sciences they are independent.

In the case of concocting an instrument which can measure the genetic affiliation of languages, the bare minimum we expect should be that it is able at least to recognize that two languages are the same language or that the two languages are so close that they are dialects. What kind of a temperature scale would not be able to discriminate that the temperature T in one day was the same as the temperature T on another day? What kind of a ruler could not discriminate that the standard length of a piece of paper is exactly the same as another sheet? If there are attempts to create metrics (algorithms) using probability theory, these algorithms should at least be tested for validity and reliability for known cases. The clearest case of a known answer is when a language is tested against itself. Any alleged method, algorithm, instrument, or metric that cannot clearly and unequivocally show that the two languages are identical (when they are identical) cannot be said to have passed the validity test.

Scales or Levels of Measurement

Before we try to measure or normalize quantities we should know what kinds of measurements we have. They determine if we can multiply those numbers, add them, rank them etc. Accordingly measurements are classified as: (i) Ratio scale, (ii) Interval scale, (iii) Ordinal scale, or (iv) Nominal scale.

***I. Ratio Scale*:** The highest level of measurement scale is that of ratio/absolute scale. A ratio scale requires an absolute or nonarbitrary zero, and on such a scale we can multiply (and divide) numbers knowing that the result is meaningful. The standard length measurement using a ruler is an absolute or ratio scale.

Example *Distance*

Probably the most common measurement that people are familiar with is that of distance. It is such a general and common-sensical idea that mathematicians have abstracted from it whatever properties it has that makes it so useful and have extended it to mathematical spaces so that this idea, is in fact, used and useful in the previous ideas of measurements. The requirement that the concept of distance satisfies is:

B.5) $$d(x,z) \le d(x,y) + d(y,z)$$

The concept of "distance" or "distance metric" or "metric spaces" is motivated by the simple concept illustrated below.

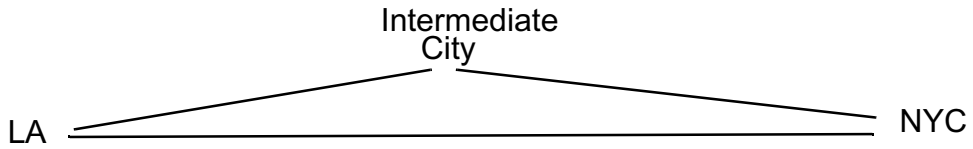


Figure B:6: Concept of Distance Metric: Any detour from NYC to LA cannot be a shorter than the direct distance between the two since distance is measured as the shortest distance between two points.

If we substitute from the figure above we can see that the distance from LA to NYC can never be greater than the distance from LA to some intermediate city plus the distance from that intermediate city to NYC. Any space in which distance is defined is a metric space.

Example *Hamming Distance*

Hamming distance is the number of bits by which two bitstrings differ. For example the distance between the bitstring 1111 and 0000 is 4 since the corresponding bits of the two bitstrings differ in 4 places. The distance between 1010 and 1111 is two, and the distance between 1010 and 0000 is also two.

Example *Phonological Distance: Distinctive Features*

In phonology, the basic primitive objects are phonemes. They are descriptions of the basic building blocks of speech and are usually described in binary as the presence or absences of

specific characteristics such as voicing, rounding, frication, plosivity etc. Since we can represent these as bitstrings the Hamming distance can be used to measure the distance between phonemes [Hubey, 1994]. This metric is sometimes called 'city block' metric because of the way distances are measured. For binary-valued variables this can be shown clearly on a Karnaugh map (see Hubey[1994]) as shown below.

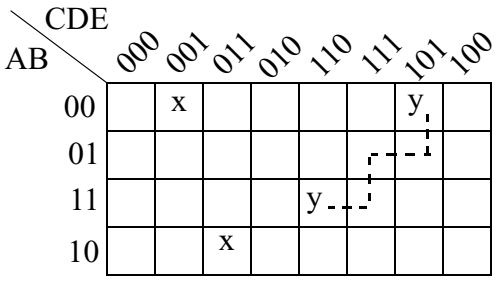


Figure B:7: Karnaugh Map for Phonological Distance [Hubey[1994]].

Five binary variables are plotted on the Karnaugh map. The values of the A and B variables are writen vertically and in what is called a "reflected code". The bitstrings 00,01,11, and 10 are placed so that the adjacent bitstrings differ only by one bit. The three variables C,D, and E are represented similarly so that again the adjacent bitstrings differ by a single bit. The two y's represent the bitstrings 00101 and 11110, thus the distance is 4. A typical (city block) distance between the y's is shown in dotted lines. However the distance between the two x's is 2 which can be verified from the bitstrings 00001 and 10011. To see this pictorially, the map must be wrapped on a torus (see Hubey[1994]).

Example *What's a Bird?*

The concept of distinctive features can also be used in conjunction with fuzzy logic in artificial intelligence to describe (or define) objects, such as a bird, fruit or a chair. For example, a set of simple properties such as "has feathers", "is bipedal" and "flies" is generally sufficient to fuzzily define 'bird' for intelligent entities (such as humans).

II. Interval Scale: However, not everything that can be measured or represented with integers (or real numbers) constitutes a ratio/absolute scale.

Example For example, the Fahrenheit temperature scale is only an interval scale. The differences on an interval scale (such as the Fahrenheit scale) are valid and meaningful and correct, but multiplication/division is not. For example, 100F is not twice as hot as 50F.

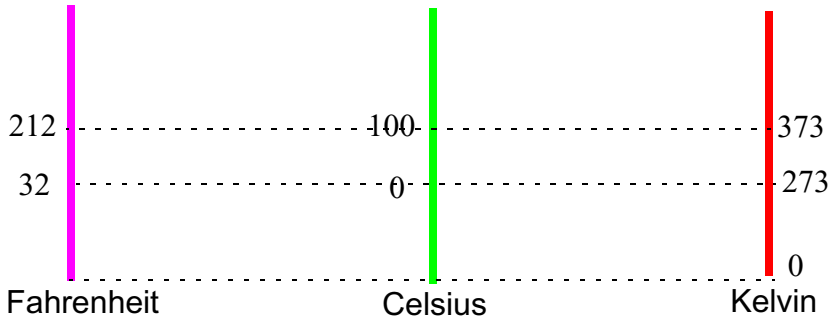


Figure B:8: Various Temperature Scales: For calculating things like engine efficiency only the absolute (ratio) scale i.e. Kelvin temperature scale, can be used.

It took Lord Kelvin and greater development of thermodynamics theory to show that the Fahrenheit scale gave wrong answers when used in some problems in thermodynamics Kelvin was able to show that the lowest temperature achievable was about -460F (-273C) and that this temperature should have constituted absolute zero on the temperature scale. Since then despite all attempts no temperature lower than absolute zero has been achieved in laboratories.

III. Ordinal Scale: The next level on the measurement scale is the ordinal scale, a scale in which things can simply be ranked according to some number but the differences are not valid. In the ordinal scale we can make judgements such as A>B. Therefore if A>B and B>C, then we can conclude that A>C. In the ordinal scale there is no information about the magnitude of the differences between elements. We cannot use operations such as +, -, * or / on the ordinal scale.

Example *Likert Scale*:
It is possible to obtain an ordinal scale from questionnaires. One of the most common, if not the most common is the multiple-choice test which has the choices: extremely likely/agreeable, likely/agreeable, neutral, unlikely/disagreeable, and extremely/very unlikely/disagreeable.

IV. Nominal Scale: The lowest level of measurement and the simplest in science is that of classification. In classifying we attempt to sort elements into categories with respect to a particular attribute. This is the nominal scale. On this scale we can only say if some element possesses a particular attribute but cannot even rank them according to some scale on a hierarchy based on the intensity of possession of that attribute. We can only think of creating sets based on the possession of some property and apply the operations for sets. In this sense the set operations are the most primitive of operations of mathematics. It ranks so low on the scale or hierarchy that we all instinctively do it. Whatever kind of logic that flows from this must obviously be related to set theory in some way. Obviously classification is about nothing more than set theory, and although may be useful in at least providing some logical consistency, and a boon to common sense, it is far from creating a science like any of the quantitative sciences. The road ahead for linguistics is clear.

Appendix C: Relations

Definition **Binary relation** α from a set X to a set Y (or between two sets X and Y) is a subset R_α of the cartesian product $X \times Y$ ($R_\alpha \subseteq X \times Y$). The set $D(R_\alpha)$ of all objects x such that for some y, $\langle x, y\rangle \in R_\alpha$ is called the **domain** of R_α. Similarly, the set $R(R_\alpha)$ of all objects y such that for some x, $\langle x, y\rangle \in R_\alpha$ is called the **range** of R_α.

Many kinds of relations exist but there are some that are quite general and interesting in their own right, for example, reflexive (and irreflexive), symmetric (and anti-symmetric) and transitive.

Definition R is *reflexive* $\equiv \forall x(xRx)$.

Definition R is *irreflexive* $\forall x((x, x) \notin R)$. A relation R on X is irreflexive if for every $x \in X$, $(x, x) \notin R$. In other words, there is no $x \in X$ such that xRx.

Definition R is *symmetric* $\equiv \forall x \forall y(xRy \Rightarrow yRx)$. A relation R is called asymmetric if $(a, b) \in R$ implies $(b, a) \notin R$.

Definition R is *anti-symmetric* $\equiv \forall x \forall y(xRy \wedge yRx \Rightarrow (x = y))$ The contrapositive of this which is equivalent (obviously) is $a \neq b \Rightarrow \overline{aRb} \vee \overline{bRa}$ or using different notation $a \neq b \Rightarrow (a, b) \notin R \vee (b, a) \notin R$

Definition Asymmetry: if xRy then *not*(yRx). Therefore asymmetric means not symmetric.

Definition R is *transitive* $\equiv \forall x \forall y \forall z(xRy \wedge yRz \Rightarrow xRz)$

Reflexivity and irreflexivity depend only on the diagonal. If some elements of the diagonal are 1 while others are 0 (in the relation in matrix form) then the relation is neither reflexive nor irreflexive. In the digraph (directed graph) of a symmetric relation all arcs/edges are bidirectional. In an anti-symmetric relation, no arc/edge has a mate that goes in the opposite direction. Reflexivity, and symmetry can be spotted quite easily either on the graph of the relation or the zero-one matrix of the relation.

The regularity of sound change, the "working horse" assumption of comparative linguistics, is a transitive relation. For example, if the sound changes from the protolanguage P to one of its daughter languages $D^k{}_1$ (the immediate daughter i.e. level 1 descendant, in the kth branch) is regular, and if the sound changes from $D^k{}_1$ to its daughter $D^k{}_2$ is regular, then the sound change from P to $D^k{}_2$ is also regular.The same holds for every descendant for every branch. Therefore the relation from one daughter language to another daughter language is also regular.

Representation of Relations

1. Set-theoretic Representation of Relations

Relations can be written/expressed simply as a set of ordered pairs:

C.1a) $R_1=\{(1,1),(1,2),(2,1),(2,2),(3,4),(4,1),(4,4)\}$
C.1b) $R_2=\{(1,1),(1,2),(1,4),(2,1),(2,2),(3,3),(4,1),(4,4)\}$
C.1c) $R_3=\{(2,1),(3,1),(3,2),(4,1),(4,2),(4,3)\}$
C.1d) $R_4=\{(1,1),(1,2),(1,3),(1,4),(2,2),(2,3),(2,4),(3,3),(3,4),(4,4)\}$

2. Matrix Representation of Relations

We can represent relations via adjacency matrices. In this representation a 1 entry in the a_{ij} position indicates that there is an edge from ith vertex to the jth vertex.

C.2)

$$\begin{bmatrix}1&1&0&0\\1&1&0&0\\0&0&0&1\\1&0&0&1\end{bmatrix} \quad \begin{bmatrix}1&1&0&1\\1&1&0&0\\0&0&1&0\\1&0&0&1\end{bmatrix} \quad \begin{bmatrix}0&0&0&0\\1&0&0&0\\1&1&0&0\\1&1&1&0\end{bmatrix} \quad \begin{bmatrix}1&1&1&1\\0&1&1&1\\0&0&1&1\\0&0&0&1\end{bmatrix}$$

not reflexive	reflexive	not reflexive	reflexive
not irreflexive	not irreflexive	irreflexive	not irreflexive
not symmetric	symmetric	not symmetric	not symmetric
not antisymmetric	anti-symmetric	anti-symmetric	anti-symmetric

Figure C:1: Relations in #1 and whether they are reflexive, irreflexive, symmetric or anti-symmetric. Reflexiivity and symmetry have to do with the main diagonal of the relation matrix. Transitivity is harder to determine.

3. Graph-theoretic Representation of Relations

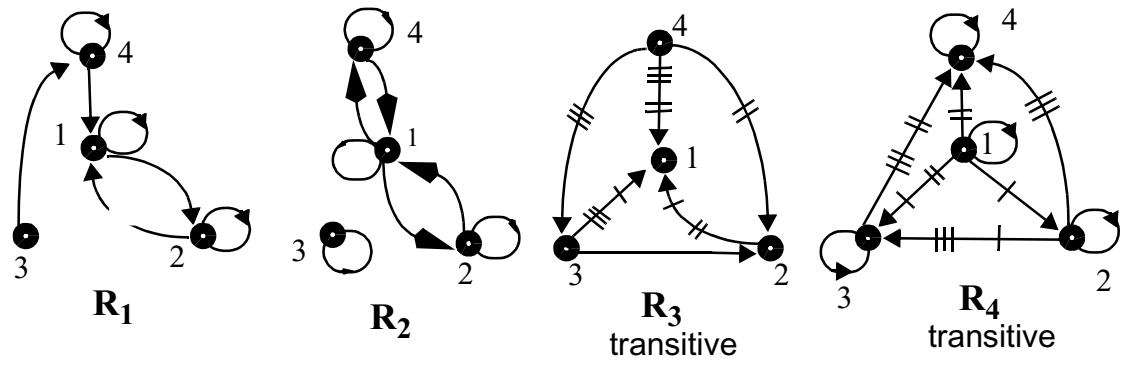
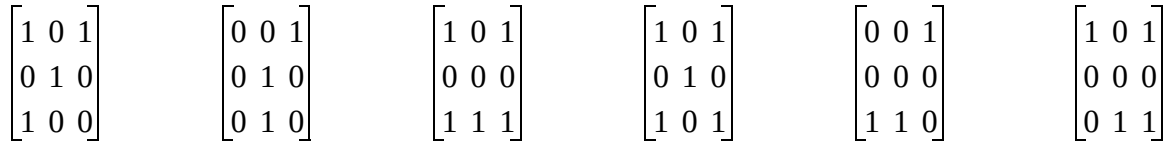


Figure C:2: Graphs of Figure C:1 and whether they are transitive: The edge triplets labeled by tick marks dellineate relations involved in transitivity.



symmetric	anti-symm	neither	symmetric	neither	anti-symm
neither	neither	neither	reflexive	irreflexive	neither

Figure C:3: Are the relations symmetric or reflexive?

Boolean matrices, and Relations

As can be seen above, these matrices are of the special type called zero-one matrices. These matrices are used to represent discrete structures as shown above. In the matrix multiplication of the earlier section, the laws of arithmetic were used. It is possible to use Boolean operations instead of arithmetic operations on these zero-one matrices. In this case the addition operation gets replaced by $\vee$ and multiplication by $\wedge$. The **join** of two zero-one matrices A, and B is the matrix G whose elements are $g_{ij} = a_{ij} \vee b_{ij}$. The meet of two zero-one matrices A, and B is a matrix E whose elements are $e_{ij} = a_{ij} \wedge b_{ij}$. Note that this is not normal (arithmetic)matrix multiplication. The matrices A and B must be the same size. The analogue of matrix multiplication for zero-one matrices is the Boolean product of two zero-one matrices. It is simply the product of two zero-one matrices using $\vee$ and $\wedge$ instead of addition and multiplication, respectively. The Boolean product of the two matrices $A \odot B$ above is:

C.3)

$$\underset{2\times 3}{\begin{bmatrix}1&1&1\\1&1&1\end{bmatrix}} = \underset{2\times 4}{\begin{bmatrix}0&1&1&0\\1&0&1&0\end{bmatrix}} \underset{4\times 3}{\begin{bmatrix}0&1&1\\1&1&0\\1&0&1\\1&0&0\end{bmatrix}}$$

Figure C:4: Matrix Multiplication for Relations using Boolean Operations: Addition and multiplication are Boolean operations; the algorithm to calculate every element is the same as matrix multiplication.

If R and S are relations so then are $R \cap S$, $R \cup S$, $R \oplus S$, $R - S$, $S - R$. For example if W is the set of roots/stems of a given language, and P is the set of phonemes of the language, then we can define relations as follows. Let the relation F be the set of ordered pairs (p,w) where p occurs in word w in the word-final position. Let I be the set of ordered pairs (p,w) where the phoneme p occurs in the word-initial position. Then $F \cap I$ is the set of all ordered pairs (p,w) in which the phoneme p occurs both in the initial and final position in the word w. The $F \cup I$ is the set of all ordered pairs (p,w) in which p occurs either in the initial or final position in word w. $F \oplus I$ is the relation in which the phoneme occurs either in the initial or final position but not both; $F - I$ is

the relation in which p occurs in the final position but not in the initial position; $I - F$ is the relation in which the word appears in the initial position but not the final position.

Composition of Relations

Let R: $X \rightarrow Y$ and S: $Y \rightarrow X$ be two relations. The composition of R and S, denoted by $R \bullet S$ contains the pairs (x,z) if and only if there is an intermediate object y such that (x,y) is in R and (y,z) is in S. Therefore $x(R \bullet S)z = \exists y(xR(y \wedge yRz))$. Note that xRy can also be written as Rxy or R(x,y) or $(x, y) \in R$. The composition of relations is given by the Boolean product of the matrices of the relations. An example is shown below.

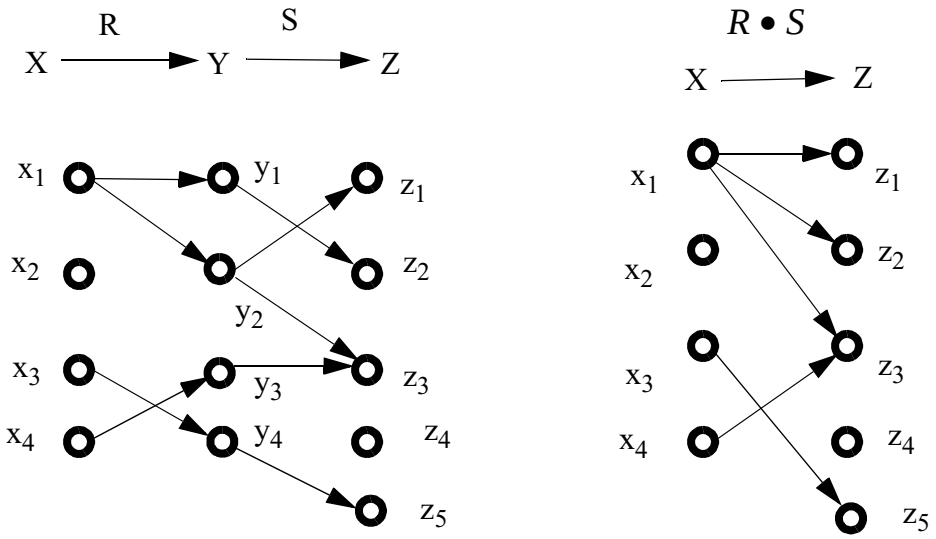


Figure C:5: The Graph of Composition of Relations. One can check the Boolean product of the X and Y matrices to see that the composition of relations is calculated by that product. Suppose set X is the set of basic words in the protolanguage, and let set Y be its daughter languages. According o the Neogrammarian Assumption, Y cannot be a single daughter language since some words of the protolanguage change to more than one 'offspring' words. In other words some of the members of set Y are cognates, and are reflexes of the corresponding protoword. For example, y_1 and y_2 are cognates and are reflexes of x_1. Then let set Z is the set of daughter languages of the daughter languages, for which the set Y is a protolanguage since every language is a potential protolanguage for its descendants. In the above example, we see that z_3 is a word that may be in two languages and shows a convergence of two separate words. In reall languages such things are possible. Now the composition of these two relations is also a relation and maps the original protolanguage to its second level daughter languages. It can be seen in this for example that word z_3 is a convergence of two words x_1 and x_4 in the original protolanguage. We can see from this simple example that the methods of historical/comparative linguistics can easily be discussed in terms of sets and relations.

From the graph above we can see that the matrices for the relations are

C.4) $$R=\begin{bmatrix}1&1&0&0\\0&0&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix} \text{ and } S=\begin{bmatrix}0&1&0&0&0\\1&0&1&0&0\\0&0&1&0&0\\0&0&0&0&1\end{bmatrix}.$$

The Boolean product is given by

C.5) $$R \bullet S=\begin{bmatrix}1&1&1&0&0\\0&0&0&0&0\\0&0&0&0&1\\0&0&1&0&0\end{bmatrix} = \begin{bmatrix}1&1&0&0\\0&0&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}\begin{bmatrix}0&1&0&0&0\\1&0&1&0&0\\0&0&1&0&0\\0&0&0&0&1\end{bmatrix}$$

Theorem Composition is an associative operation: $(R \bullet S) \bullet T = R \bullet (S \bullet T)$

Proof:
Using *suppressed summation* tensor notation (first used by A. Einstein)

C.6) $$(R \bullet S) \bullet T = (r_{ij}s_{jk})t_{km} = q_{ik}t_{km} = v_{im}$$
$$R \bullet (S \bullet T) = r_{ij}(s_{jk}t_{km}) = r_{ij}u_{jm} = v_{im}$$

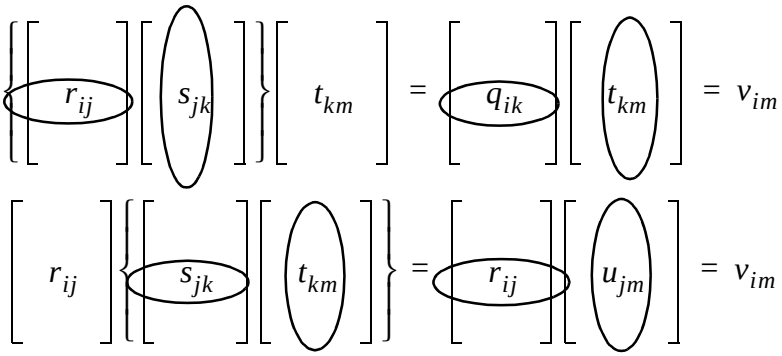
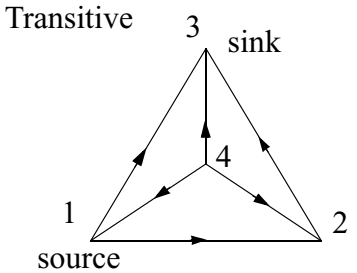


Figure C:6: Composition Multiplication

Fact If R is a relation on the set A, then the powers R^n, n=1,2,3.. are defined inductively by $R^1 = R$ and $R^{n+1} = R^n \bullet R$. An example is shown below.



$$\begin{bmatrix}0&1&1&0\\0&0&1&0\\0&0&0&0\\1&1&1&0\end{bmatrix} \quad \begin{bmatrix}0&0&1&0\\0&0&0&0\\0&0&0&0\\0&1&1&0\end{bmatrix} \quad \begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&1&0\\0&0&0&0\end{bmatrix}$$
R $\quad$ R^2 $\quad$ R^3

Figure C:7: Powers of the Matrix of Relation R

Fact The relation R on a set X is transitive *if and only if* $R^n \subseteq R$ for n=1,2,3...

Definition A transitive closure for relation R is a relation R^* that contains R, is transitive and is as small as possible. In other words R^* contains R, is transitive and is contained in every transitive relation that contains R.

Definition The connectivity relation R^c of a relation R is the pairs (x,y) such that there is a path between x and y in R. Since R^n consists of pairs (x,y) such that there is a path of length n from x to y, then it follows that R^c is the union of all the sets R^n or $R^c = R \cup R^2 \cup R^3 \cup \ldots = \bigcup R^n$.

Fact The transitive closure R^* of a relation R is the connectivity relation R^c.

Equivalence Relations

Definition A relation on a set is called an equivalence relation if it is reflexive, symmetric and transitive.

Example Congruence Modulo m: Show that the relation

$$R = \{(a,b) | a \equiv b (mod\ \ m)\}$$

is an equivalence relation on the set of integers.

We know that $a \equiv b(mod\ \ m)$ if and only if m divides a-b. Note that a–a=0 is divisible by m since $0 = 0 \cdot m$. Hence, $a \equiv a(mod\ \ m)$, therefore it is reflexive. Then if a-b is divisible by m then so is b–a since b–a=–(a-b) therefore it is symmetric. Suppose $a \equiv b(mod\ \ m)$ and $b \equiv c(mod\ \ m)$. Therefore, from the definition of the modulo operation, a–b=km and b–c=nm. Adding these together we obtain a–c=(a–c)+(b–c)=km+nm=(k+n)m. Therefore $a \equiv c(mod\ \ m)$.

Definition If R is an equivalence relation on a set X, then the set of all elements that are related to an element x of X is called an equivalence class of x. The equivalence class of x with respect to (wrt) R is denoted by $[x]_R$. In other words $[x]_R = \{y | (x,y) \in R\}$. If $b \in [x]_R$ then b is called a representative of this equivalence class.

Fact Let R be an equivalence relation on a set S. Then the equivalence classes of R form a partition of S. Conversely, given a partition $\{A_i | i \in I\}$ of the set S, there is an equivalence relation R that has the sets A_i, $i \in I$ as its equivalence class. If the reflexes of any protolanguage could be guaranteed to exist in one and only one present-day daughter language then the relation of regular sound change would create equivalence classes so that every reflex of every word of the protolanguage would exist in only one class (i.e. one daughter language).

Inverses

Definition If R is a relation from a set A to a set B, then the inverse relation from B to A, denoted by R^{-1} is the set of ordered pairs $\{(b,a)|(a,b)\in R\}$. The complementary relation R is the set of ordered pairs $\{(a,b)|(a,b)\notin R\}$. The graph of the inverse relation has the same edges except that the arrows are in the opposite directions. If the relation is symmetric, the relation is self inverse. Think about flipping the matrix along the diagonal.The inverse is, in effect, *undo*. One can see this quite plainly in various functions for example, $\ln(e^x)=e^{\ln x}=x$, and $\sin(\arcsin(x))=\arcsin(\sin(x))=x$. Similarly if we have $f(x) = \alpha x + \beta$, then the inverse function is

C.7) $$f^{-1}(x) = \frac{x-\beta}{\alpha}.$$

Therefore

C.8) $$f(f^{-1}(x)) = f\left(\frac{x-\beta}{\alpha}\right) = \alpha\left(\frac{x-\beta}{\alpha}\right)+\beta = x$$

and

C.9) $$f^{-1}(f(x)) = f^{-1}(\alpha x+\beta) = \frac{(\alpha x+\beta)-\beta}{\alpha} = x\ .$$

Appendix D: Functions

Definition A binary relation f from set X to set Y is called a **function** if for every $x \in X$ there is a unique $y \in Y$ such that $\langle x, y\rangle \in f$.In other words, the function is a rule which assigns to each element of X some element of Y. It is customary to write y=f(x) instead of writing $\langle x, y\rangle \in f$.In this case, x is called the **argument** of the function f and the corresponding y is called the **image** of x under f. Sometimes this notation is extended to the whole set so that we denote the range of as R_f=f(X).

Note that this definition requires that a function must satisfy two conditions to qualify. The first condition is that every $x \in X$ (the domain) must be related to some element $y \in Y$ (the codomain); that is D_f=X which means that the domain of f is not a subset of X but the whole set. Similarly, the range of f is denoted by R_f and $R_f \subseteq Y$. In graphical notation every vertex $x \in X$ must have an arc emanating from it. The second condition is the uniqueness condition; that is one and only one element of Y is mapped to each x. In this definition, one cannot have two arcs emanating from any vertex since it would imply that the element mapped to x is not unique. Using $\wedge$ for conjunction (logical AND) we write this as

D.1) $$(\langle x, y_m\rangle \in f \wedge \langle x, y_n\rangle \in f) \Rightarrow (y_m = y_n)$$

or as

D.2) $$(y_m = f(x)) \wedge (y_n = f(x)) \Rightarrow (y_m = y_n)\ .$$

This means that one element of X cannot be mapped to many elements of Y (*one-to-many* is not a function!). But, the condition does not say that many elements in X cannot be mapped to one element of Y (*many-to-one* mapping is a function!). However, the first condition is relaxed by some authors [Tremblay & Manohar, 1975:193]. When this requirement is relaxed i.e. $D_f \subseteq X$, then it is a partial function [Preparata & Yeh, 1973:29].

Example In linguistics a natural language's speech sounds are categorized so that a minimal number of sound shapes are used to write the words or speech sounds of that language. Let this minimal set of sounds be the set P. One way to describe these speech sounds is by a simplified description of the positions of the speech articulators which are called distinctive features of the sound. For example a simple division of simple vowels (not compounds such as diphthongs or tripthongs) is by the two way position of the tongue high-low and front-back along with the motion of the lips round-unrounded. Thus u is {round, back, low} and the sound i is {unround,front,high}. If we now depict these phonemes as 100 and 001 respectively then the Hamming distance between these two i.e. H(u,i)=2. Similarly H(a,u)=H(010,100)=2.

Example Define *f(word,place,transition)* as a function from $A \times N \times P$ to A where A is the set of phonological words in a language, N is the set of natural integers and P is the set of phonemes. Therefore the function f(*pit*,2,*e*)=pet since it says to change the 2nd phoneme of the word *pit* into e by which the word *pit* changes to *pet*.

Definition A function f:$X \to Y$ is called **onto** (surjective) if R_f=Y; otherwise it is called *into*. Another way of saying this is that the function is onto if for any $y \in Y$ there is at least one $x \in X$ such that y=f(x). Graphically, this means that each vertex of Y is reached by at least one arc from X, which means again that *many-to-one* is a function.

Definition A function is **one-to-one** (injective) if distinct elements of X are mapped to distinct elements of Y. In other words $(x_m \neq x_n) \Rightarrow (f(x_m) \neq f(x_n))$ or the contrapositive $(f(x_m) = f(x_n)) \Rightarrow (x_m = x_n)$. Note that we had already allowed *many-to-one* so this restriction means that it is not allowed. Therefore two elements of X (i.e such as x_m and x_n) cannot map to the same element y (i.e. f(x) unless x_m=x_n. So if x_m does not equal x_n, then they can't map to the same y in Y.

Definition A one-to-one correspondence (or bijection) is a function that is both one-to-one and onto.

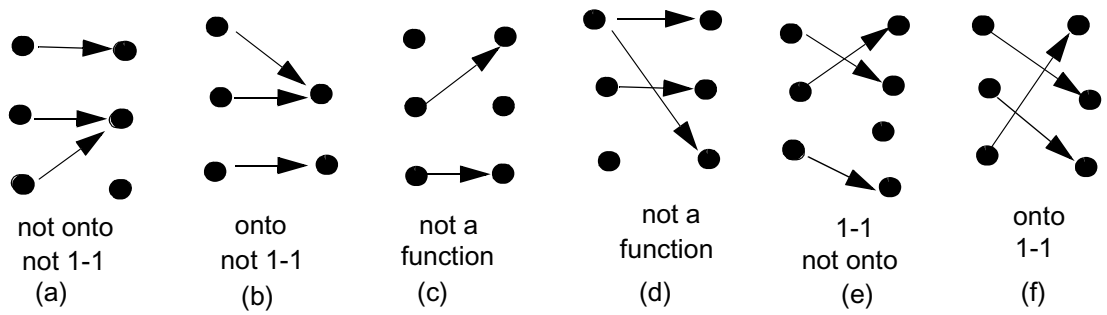


Figure D:1: Functions and nonfunctions
In real life the relation from the set of words of a protolanguage and the set of words of any of its descendants is not a function at all but is more like that of (c). Some words of the protolanguage do not get transmitted forward in time, and some words in the daughter language do not descend from any of the words of the protolanguage. The implicit algorithms used in historical linguistics (the comparative method) assumes that we can select the subset of the words of any language so that we can separate them into three classes: (i) those descended directly from the ancestors of the daughter language all the way from the protolanguage, (ii) those that are copied from ancestors of other daughter languages and (iii) those that are copied from or left over from other families. The relation between the protolanguage and those of set (i) is definitely not onto and is almost 1-1. It is not onto because every word in the daugher language does not come from the protolanguage. It is almost 1-1 because every word of the protolanguage does not show up in every daughter language. However the relation from a subset of the protolanguage to its daughters is 1-1. The trick is to reconstruct this subset. At the same time, a function from a daughter language's set of words descended from its protolanguage to its protolanguage is 1-1 (but not onto) since every word in this set must descend from the protolanguage but since some of the words of the protolanguage have been lost we cannot recover them from a single language. That is also a problem of reconstruction or geneticity or family determination.

Definition A function f: $X \to Y$ is said to be invertible if its inverse f^{-1} is a function from Y to X. A function is invertible if and only if it is both one-to-one and onto. If a function is not invertible, then it is not a one-to-one correspondence since the inverse of such a function does not exist.

Example *Cladistics*

Cladograms are essentially hypotheses about the pattern of nested evolutionary novelties postulated to occur among a group of organisms, and are therefore branching diagrams of organisms. As a general rule, organisms that share many similarities are likely to be related. The inference of phylogenetic relationships requires basic distinction between three types of similarity; *convergent similarity* (developed independently), and *homologous similarity* (based on inheritance from a common ancestor) which gets further divided into two classes; *primitive features* (those that were present in the initial common ancestor of the group) and *derived features* (those that were developed in a later ancestor within the genetic tree). Only shared derived similarities indicate the pattern of relationships in a tree. If we consider n homologous distinctive features, and out of this m of them are derived features between two species, then (n−m)/n is a normalized measure of similarity between the two species. In normalized distance measure, distance and similarity are related by d(x,y)=1−s(x,y) or s(x,y)=1−d(x,y). In this sense distance is a measure of dissimilarity so that if s(x,y)=(n−m)/n then d(x,y)=1−(n−m)/n=m/n. Therefore we can find the distance between every member of the group being considered, and then find the minimal spanning tree so that each organism is adjacent to the closest (most similar) organism.

Example *Phonemes as Vectors*

It is standard in phonology to define (or describe) the basic building blocks of speech, phonemes, in terms of articulatory, acoustic and perceptual terms, using the concept of "distinctive features". These features normally revolve around the descriptions of the primary articulators; the tongue position, the lips, the velum and the glottis. The lip positions are usually described as round or unround for vowels and bilabial for consonants such as p, b, and m, and labio-dental for the constants such as f and v. The tongue position normally low resolution descriptions has two degrees of freedom; height and placement from the front of the articulatory channel to the back. Many consonants are described simply by the name of the part of the mouth where the the tongue makes a constriction such as velar, laryngeal, dental. In addition the velum is used for nasalization and the glottis for voicing. In addition, speech sounds are divided up into sets depending on some real or alleged characteristics of these sounds which naturally makes them belong together.

In very broad ways we can divide up these speech sounds into 4 classes; Vowels (V), Consonants (C), Semivowels (S), and Quasiconsonants (Q). The first three are commonly used; the fourth comes from Hubey [1994]. The usefulness of a classification in powers of two is obvious. In addition there are fundamental reasons why speech sounds should be divided into these four groups. In order to make a four way division we need two binary features. In this case one of them is the fact that vowels and the quasiconsonants (liquids and fricatives) are steady-state sounds. They can be produced while the articulators are held steady. The consonants and semivowels cannot be produced except via a motion of the articulators. The other binary distinctive feature is what might be called resonance or frication-to-resonance ratio which is analogous to the signal-to-noise ratio. Vowels and semivowels are naturally identified as speech sounds with highly peaked (resonant)

spectra, while plosives such as p,t,k have almost constant power spectra (in which it is almost impossible to see any resonance) and voiced plosives in which one sees a spectra resembling a signal with a very high level of noise. Just as the semivowels are those sounds which have some characteristics of consonants (because of the motion of the articulators) the quasiconsonants have some characteristics of vowels in that they have some resonance (and can be produced while the articulators are in steady-state) so that their power spectra lies somewhere between pure frication (such as for sibilants s, sh or f) and a vowel with a lot of noise. The quasiconsonants have peaked power spectra with choppiness that is due to the noise.We can denote a phoneme as a vector in many ways. The simplest (and not necessarily the most accurate or the best for any given purpose) is to denote it by using a set of binary distinctive features. Any of the set of binary features (which are called oppositions in common linguistics terminology) such as from Ladefoged may be used. In this case, the phoneme is simply an n-tuple or a vector

Example *Phonemes as Fuzzy Vectors*

We can have low resolution phoneme description using only five dimensional fuzzy vectors using the dimensions front-back, round-unround, nasal-nonnasal, voiced-unvoiced and motion-steady_state. The distance for this should be like this: if the first digit is the same then the distance at the first level is zero, but if the first digit is not the same then the distance is 1. So we need a hierarchical distance. If the classification is of the form $D_1D_2...D_{n-1}D_n$ then we compare the two strings and weight each difference at each level differently. The only thing we look for at the same level is if they are the same or not. If they are the same the distance at that level is zero. If they are different then the distance at that level is 1. If the tree could be produced so that a hierarchy is exhibited also from the left to right at each level then we could use the absolute value of the differences (or square of the difference) as a measure of distance at that level. Of course, these partial distances will also have to be weighted appropriately. What this means is that we should create a distance scheme of the sort:

D.3) $$d(x,y) = \sum_{i=1}^{n} w_i \cdot (x_i \oplus y_i)$$

This scheme would work for color distance when R, G, and B are represented as say, 3 bits each for then since these three bits are intensities, the most significant bits (MSB) should be counted more heavily in the distance so we can just divide by the bit position counting from the MSB so that we have $w_j=1/j$. In this case, the maximum score would be

D.4) $$\frac{1}{1}+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+\ldots+\frac{1}{n}.$$

This is the Harmonic series. The sum can be found from [Graham et al, 1989: 41]

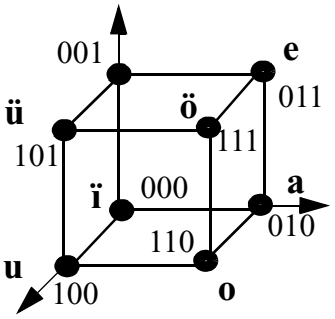
D.5) $$\sum_{0 \le k < n} H_k = nH_n - n$$

It would be much easier to use the weighting $w_i=(1/k)^i$ where k=1 yields the unweighted case and for k >1, we have the geometric series so that we can easily compute the normalizing factor, therefore the distance function is;

D.6) $$d(x,y) = \sum_{i=1}^{n} w_i(x_i \oplus y_i) = \frac{1-k}{1-k^{n+1}} \sum_{i=1}^{n} w_i(x_i \oplus y_i)$$

Example *Simple Phoneme Distances*

We can compute distances between phonemes using different ways.



Hamming Distance example

$$d(i,o) = i_1 \oplus o_1 + i_2 \oplus o_2 + i_3 \oplus o_3 = 3$$

Euclidean Distance example

$$d(i,o) = \sqrt{(|0-1|)^2 + (|0-1|)^2 + (|1-0|)^2} = \sqrt{3}$$

Generalized Distance:

$$d(i,o) = \sqrt[2n]{(|0-1|)^{2n} + (|0-1|)^{2n} + (|1-0|)^{2n}}$$

Figure D:2: Distance Examples on the Ordinal Cube of Wowels [Hubey,1994]
Clearly, if we substitute 1 for n in the general distance formula, we obtain the standard Euclidean distance however for phonology, perception or other purposes other values may be used. It might be useful to use other (combination) distance metrics. Bold letters are used for vectors (phonemes) Hubey[1994].

The 3D vowel cube can be seen to be many things in one. We can think of it as a vector space for vowels except that it has to be twisted and sheared into shape to fit the actual human speech sounds based on formants [Hubey,1994]. One needs to use the 3D versions of the scale, shear, translate and rotate matrices to fit the cube into data obtained from actual speech of humans.

Appendix E: Partially Ordered Sets: Posets

Definition A relation R on a set S is called a **partial ordering**, or **partial order** if it is reflexive, antisymmetric, and transitive. A set S together with a partial ordering R is called a partially ordered set, or poset, and is denoted by (S,R).

Example The "greater than or equal to" relation ($\geq$) is a partial ordering of the integers. Since $x \geq x$ for every integer the relation is reflexive. Whenever $x \geq y$ and $y \geq x$ then x=y, therefore it is antisymmetric. Whenever $x \geq y$ and $y \geq z$ then $x \geq z$ and hence it is transitive.

Example The "prerequisite tree" is not a tree but a partial order.In this definition the word really means prerequisite or corequisite. Since every course is required for itself the relation is reflexive. Whenever a course x is required for course y and course y is also required for course x, then it is the same course, therefore it is antisymmetric. Whenever course x is required for course y and course y is required for course z then course x is required for course z and hence it is transitive.

Example Descent is a partial order with the proviso that every language is its own descendant. This would mean that descent is analogous to $\geq$ instead of merely >. We could have instead invented another word but it is not necessary. The relation is anti-symmetric because if x is a descendant of y, y is not a descendant of x. And it is transitive. Similarly, we can use the sound correspondance or sound change in ways in which it is symmetric or antisymmetric.

Example The implication in logic $\Rightarrow$ is a partial ordering. It is reflexive because every proposition implies itself as the Law of Identity. It is antisymmetric because if $p \Rightarrow q$ and $q \Rightarrow p$ then $p \Leftrightarrow q$. It is transitive because if $p \Rightarrow q$ and $q \Rightarrow r$, then $p \Rightarrow r$.

References

Baxter, W. (1998) Response to Oswalt and Ringe, in Sprung from Some Common Source edited by S. Lamb and E. Mitchell, Stanford University Press, Stanford, California.

Bender, M. (1969) Chance CVC Correspondances in Unrelated Languages, Language, Np.45, 519-531.

Cowan, H. (1962) Statistical Determination of Linguistic Relationship, Studia Linguistica, 16, pp.57-96.

Durie, M., and M. Ross (eds) (1996) The Comparative Method Reviewed, Oxford University Press, New York.

Embleton, S. (1991) Mathematical Methods of Genetic Classification, in Sprung from Some Common Source edited by S. Lamb and E. Mitchell, Stanford University Press, Stanford, California.

Graham, R., D. Knuth, and O. Patashnik (1989) Concrete Mathematics: A foundation for Computer Science, Addison-Wesley.

Greenberg, J. (1960) A Quantitative Approach to the Morphological Typology of Language, International Journal of American Linguistics, No. 3, July 1960, pp.178-194.

Hubey, H.M. (1998) Quantitative Methods in Historical Linguistics and Applications to *PIE, History of Language, September..

Hubey, H.M. (1994) Mathematical and Computational Linguistics, Mir Domu Tvoemu, Moscow, Russia.

Hubey, H.M. (1999) The Diagonal Infinity, World Scientific, Singapore.

Hubey, H.M. (1999a) Mathematical Methods in Historical Linguistics: their use, misuse, and abuse, submitted to Diachronica.

Hubey, H.M. (1999b) Vector Phonological Spaces for Speech, accepted for publication in the Journal of the International Quantitative Linguistics Association.

Hubey, H.M. (1997) Logic, physics, physiology, and topology of color, Brain and Behavioral Sciences, vol 22, pp.191-194.

Hubey, H.M. and A. Gutierrez (1997) Testing Random Numbers via Cumulants, Proceedings of the Twenty-third Annual Pittsburgh Conference on Modeling and Simulation , May 1997, pp. 147-152.

Marcus, G., S. Vijayan, S. Rao, and P. Vishton (1999) Rule Learning by Seven-Month-Old Infants, Science, vol. 283, Jan 1, pp.77-80. Follow ups can be seen in the letters to the editor in Science, vol 284, 16 April 1999, pp. 434-437.

Oswalt, R. (1991) A Method for Assessing Distant Linguistic Relationships, in Sprung from Some Common Source edited by S. Lamb and E. Mitchell, Stanford University Press, Stanford, California.

Preparata & Yeh (1973) Introduction to Discrete Structures for Computer Science and Engineering, Addison-Wesley Publishing Company, Reading, MA.

Ringe, D. (1992) On Calculating the Factor of Chance in Language Comparison, The American Philosophical Society. Transactions of the American Philosophical Society, vol. 82, Philadelphia.

Ringe, D. (1995) 'Nostratic' and the Factor of Chance, Diachronica XII:1.55-74.

Salmons, J. and B. Joseph (1998) Nostratic: Sifting the Evidence, John Benjamins, Amsterdam.

Thomason, S.G. and T. Kaufman (1988) Language Contact, Creolization, and Genetic Linguistics, University of California Press, Berkeley, CA.

Tremblay & Manohar (1975) Discrete Mathematical Structures with Applications to Computer Science, McGraw-Hill Computer Science Series, New York
USN&WR (1998) Baby Talk, June 15, 1998.