Rationale for substitution matrices

- Substitution matrices are a way of keeping track of the structural, physical and chemical properties of the amino acids in proteins, in such a fashion that less detrimental mutations are penalized less than those mutations which completely destroy the function of the protein.
- Instead of the simple scoring scheme of +1 for a match, -1 for a mismatch, and -2 for a gap, a variety of scores are assigned, depending on which pair of amino acids is being switched.

Creation of a PAM matrix

1. Align sequences that are highly homologous (I.e., 85% identical).
    1. Minimize ambiguity in alignments.
    2. Minimize the number of coincident mutations.
2. Reconstruct phylogenetic trees and infer ancestral sequences.
3. Tally replacements "accepted" by natural selection, in all pair-wise comparisons (each is the number of times amino acid *j* was replaced by amino acid *i* in all comparisons).
4. Compute amino acid mutability (i.e., the propensity of a given amino acid, *j*, to be replaced).
5. Combine data from 3 & 4 to produce a *Mutation Probability Matrix* for one PAM of evolutionary distance.
6. Calculate *Log Odds Matrix* for similarity scoring
    1. Divide each element of the Mutation Data Matrix, M, by the frequency of occurrence of each residue.
7. The Log Odds Matrix, is calculated from the relatedness odds matrix by taking the log of each R_{ij}.
8. Different protein families manifest different PAM rates.

Properties of PAM matrices

1. The probability that an amino acid will change is on the order of 1% for each amino acid. The probability that it will stay the same is on the order of 99% for each amino acid.
2. The Mutation Probability Matrix, M1, *defines* a unit of evolutionary change: specifically, 1 PAM (Accepted Point Mutation per 100 residues).
    1. The matrix can be used to simulate evolution by using a random number generator to select fate of each residue in the sequence according to the probability given in the table.
    2. Exposing a 100 residue protein sequence of *average* composition to the evolutionary change represented by M1 results in one amino acid change, *on average*.
3. Successive application of M1 on a sequence yields 2, 3, 4... PAMs of evolutionary change.
4. The following operations are equivalent:
    1. Successive application of M1 on a sequence.
    2. Matrix multiplication of M1 by itself, M1*M1, followed by operation on a sequence.
    3. Scaling the elements of M1 by a constant of proportionality, = 1,2,3... (this enables the direct calculation of a matrix for any desired PAM distance).

Assumptions in the PAM model

1. Replacement at any site depends only on the amino acid at that site and the probability given by the table (Markov model).
2. Sequences that are being compared have average amino acid composition.

Selecting an optimal PAM matrix

1. As the evolutionary distance increases, the information content of the corresponding PAM matrix decreases.
2. As the information content of the PAM matrix decreases a longer region of similarity is required to generate a sufficiently high score to be distinguishable from chance.
3. Regions of similarity in real proteins are found in narrower "blocks" as the evolutionary distance increases.

Sources of error in the PAM model

- Many sequences depart from average composition.
- Rare replacements were observed too infrequently to resolve relative probabilities accurately (for 36 pairs no replacements were observed).
- Errors in 1PAM are magnified in the extrapolation to 250PAM.
- The Markov process is an imperfect representation of evolution: Distantly related sequences usually have islands (blocks) of conserved residues. This implies that replacement is not equally probable over entire sequence.

The BLOSUM family of matrices

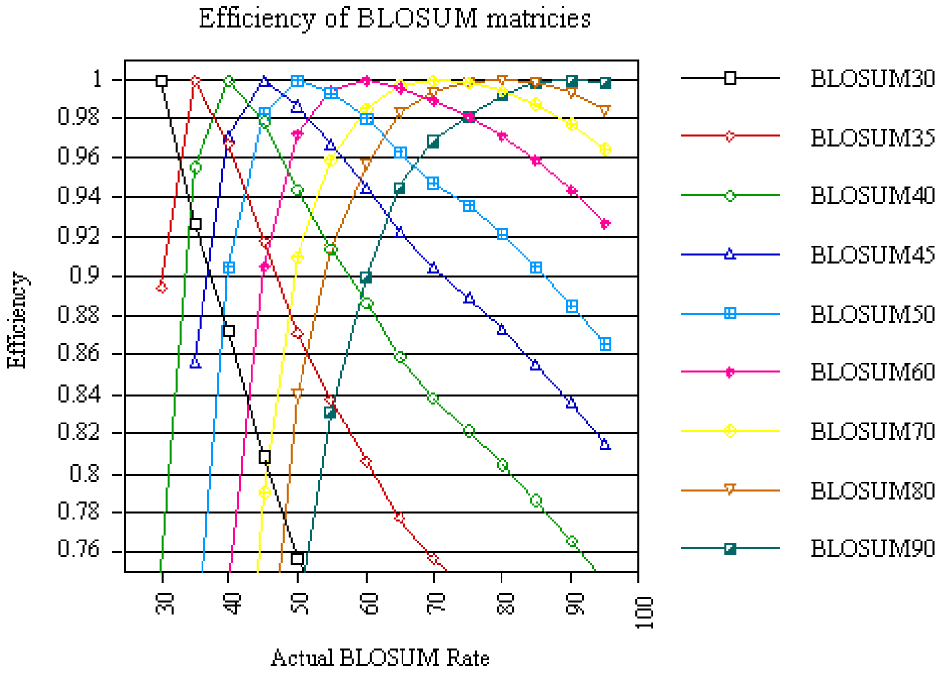
There are three principal differences between the Blosum and PAM matrices.

1. The PAM matrices are based on an explicit evolutionary model (that is, replacements are counted on the branches of a phylogenetic tree), whereas the Blosum matrices are based on an implicit rather than explicit model of evolution.
2. The sequence variability in the alignments used to count replacements. The PAM matrices are based on mutations observed throughout a global alignment, this includes both highly conserved and highly mutable regions. The Blosum matrices are based only on highly conserved regions in series of alignments forbidden to contain gaps.
3. The Blosum procedure uses groups of sequences within which not all mutations are counted the same. A small block of lipid binding proteins, taken from the Blocks database, is shown below and used to illustrate the process of counting replacements that underlies the Blosum similarity matrices.

BLOSUM: Blocks Substitution Matrix

1. Starting data is conserved blocks from the *Blocks* database
   1. Aligned, ungapped sequences
   2. Widely varying similarity, but measures are taken to avoid biasing the sample with frequently occurring highly related sequences.
2. Tallies of replacements are made by straight forward tallying of all pairs of aligned residues.
   1. The observed frequency of each pair is:
      $q_{ij} = f_{ij}$ /(total number of residue pairs)
   2. This includes cases of i=j (i.e. no replacement observed).
   3. The expected frequency of each pair is essentially the product of the frequencies of each residue in the data set.
3. Similar sequences in a block, above a threshold percent similarity are clustered and members of the cluster count fractionally toward the final tally.
   1. Reduces the number of identical pairs (AA, SS, TT, etc., matches) in the final tallies.
   2. Somewhat analogous to increasing the PAM distance.
   3. If clustering threshold is 80%, final matrix is BLOSUM 80.
   4. Clustering at 62% reduces the number of blocks contributing to the table by 25% (still 1.25e6 pairs contributed)
   5. Least frequent amino acid pair replacement was observed 2369 times.

BLOSUM matrix efficiency



The *Struct* matrix

- 32 three-dimensional structures from 11 families
- Only substitutions whose alpha carbon atoms are no more than 1.2 Å apart after superposition were considered

Scoring insertions and deletions

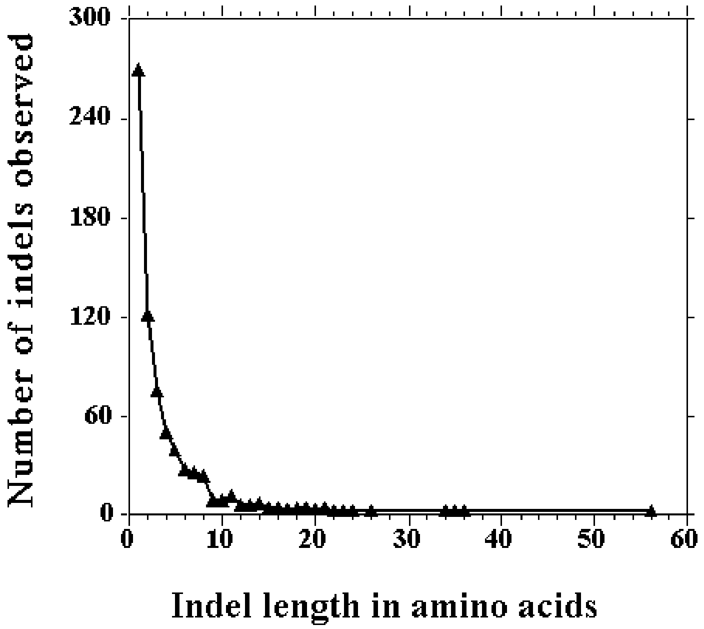
- The selection of appropriate scores for insertions and deletions, the gap penalties, is as important as selecting the similarity scores for the success of a database search.
- Gap penalties are critical in determining the expected maximum score for random sequences being compared with your query sequence.
- While we want to force alignments to have relatively few gaps we need to allow them to be fairly long.

Measuring statistical significance

- Using the Database Search to Create a Reference Distribution
  - How often will an event at least as extreme as the one just observed happen if these events are the result of a well defined, specific, random process?
- Using Information Theory and the Length of the Alignment

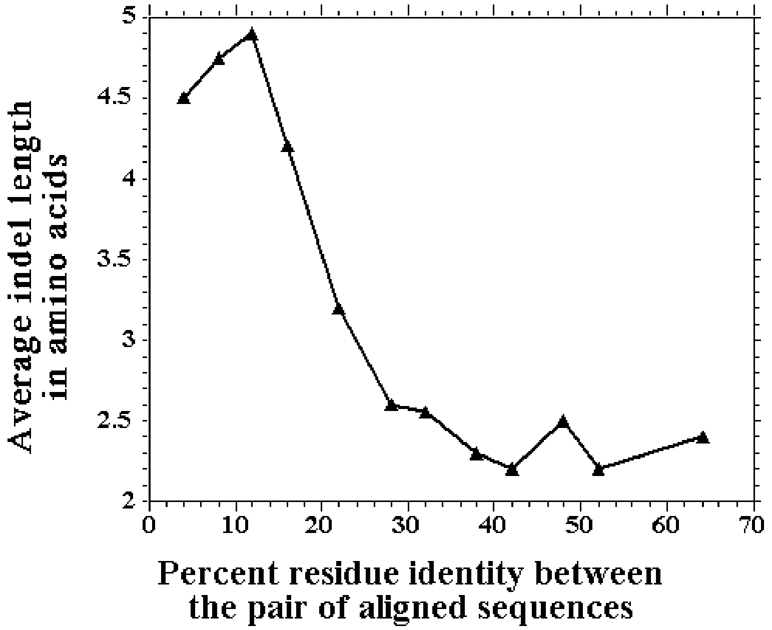
Characteristic of lengths of observed insertions and deletions.

Pascarella and Argos (1992)



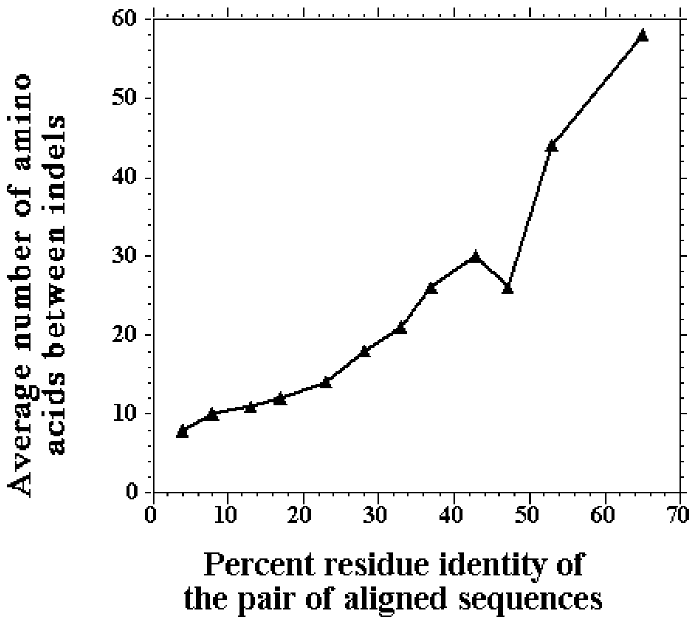
Characteristic of lengths of observed insertions and deletions vs. percent identity

Pascarella and Argos (1992)



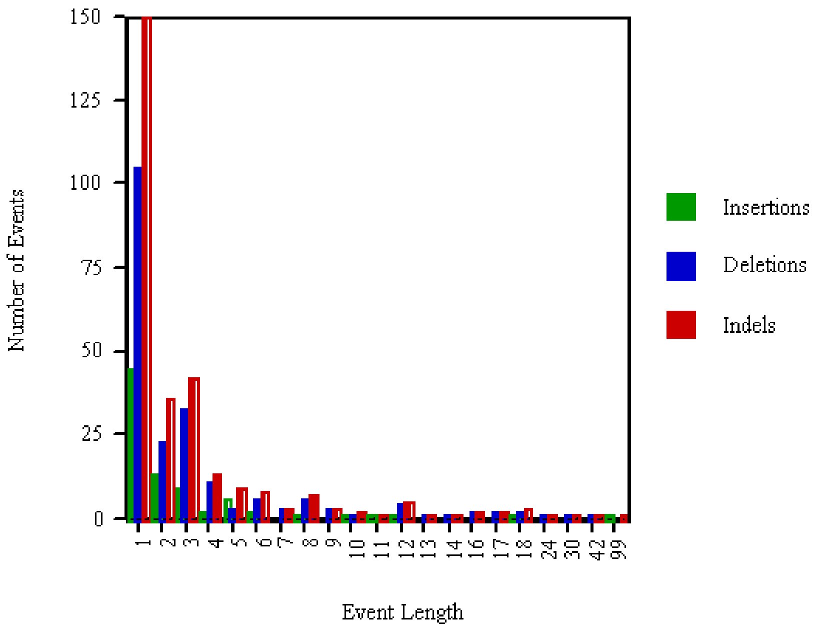
Number of residues observed between insertions and deletions vs. percent identity

Pascarella and Argos (1992)



Characteristic of lengths of observed insertions and deletions in gene-pseudogene comparisons

Gu and Li (1995)



Substitution matrix conclusions

- Expect to find many small gaps in your alignment
  - Small open gap penalty (on the same scale as the worst residue mismatch score)
  - Gap extend penalty of approximately half this size for the majority of the cases
- The sequences that you are aligning will dictate the proper penalties to use

Similarity searching and large databases

- As we have seen, the time for searching large databases is an important consideration, and it becomes necessary to balance speed and the sensitivity of the search.
- Exhaustive methods, such as Needleman-Wunsch and Smith-Waterman, generally guarantee an optimal answer but take far too long to compare a sequence with a database.
- In an attempt to gain speed with an acceptable loss of sensitivity, approximate methods have been derived from these algorithms.