john.edwards@acm.org

BIOCHEM218, Autumn 2001, Project

Organism specific Amino Acid Substitution Matrices

Abstract

Amino acid substitution matrices are widely used in the field of computational molecular biology as a means of optimising protein sequence alignments. Examples of substitution matrices include the PAM matrix [DAYHOFF79] and the BLOSUM matrix [HENIKOFF92].

Certain substitution matrices are better suited to certain alignments and one of the most common optimisations is to use a substitution matrix that is tailored to the presumed evolutionary distance between the sequences. The PAM and BLOSUM matrix families both provide different matrices suited to aligning sequences at different evolutionary distances, however it is difficult to determine the optimal evolutionary distance without first making an alignment, which itself requires the use of a substitution matrix. A common solution is to use a substitution matrix that performs reasonably well at most evolutionary distances e.g. BLOSUM62. Certain other substitution matrices are optimised to structure, as opposed to identity, for example [KOSHI95]

The aim of this work is to create organism-specific substitution matrices, the advantage being that an optimal matrix for source and target sequences - or potentially an asymmetric matrix tailored to both - could be selected using information available prior to the alignment being made.

Several organism-level statistics are evaluated as part of the work, including the effect of positional base composition [KNIGHT01] and codon usage frequencies [NAKAMURA00]. Dinucleotide genome signatures such as those described by [GENTLES01] are also briefly considered.

In the first phase of the project, substitution matrices are produced using a neural network approach based on work carried out by [LIN01], however these matrices fail to deliver any useful performance improvements. Possible causes and solutions are discussed in the report.

The second phase of the project considers organism specific biases within the context of substitution groups [WU96], in order to confirm that some organism level optimisation might be possible. This work demonstrates that within certain conserved substitution groups, different organisms favour different amino acids.

Finally, in order to examine potential underlying mechanisms for this behaviour, low order codon substitution groups are produced and evaluated.

Neural network approach

Lin et al [LIN01] report an elegant approach to the production of substitution matrices using an artificial neural network. The network was trained using a large set of aligned residue pairs produced from structural alignments with varying sequence identities. In addition to the aligned pairs, the neural network was also presented with the sequence identity of the alignment. Substitution matrices for a range of sequence identities could be read line by line from the trained neural network by presenting the each of the 20 amino acids in turn to the input layer, along with the desired sequence identity.

This work uses a similar approach, but instead of conditioning each alignment with the sequence identity, organism-level sequence statistics are used e.g. base composition.

Data

In common with Lin's work, alignments were produced using the CATH protein structure classification database [ORENGO97] as a source index. Different CATH classification levels were used to provide a set of structural alignments between homologues across the full range of sequence identity. Sequence identity was used to produce different sets of test alignments, so that the performance of the resultant neural networks could be evaluated at different levels of sequence identity.

The CATH-derived index was then used to generate pair-wise structural alignments using both SSAP and LOCK, although only SSAP derived data are presented here.

Codon Usage data was obtained from Nakamura's Codon Usage Database [NAKAMURA00]. The Codon Usage data was normalised (in a relational sense) and loaded into a relational database. Several lookup tables were created to allow per-organism codon and base composition statistics to be extracted from the dataset quickly and easily using standard SQL:

- Codon usage frequency per gene
- Codon usage frequency per organism
- Overall GC content per gene/organism
- Positional GC content per gene/organism
- N_C. Effective number of codons used in a gene

The pdb_source index was obtained from the Protein Data Bank [BERMAN00] and loaded into the database to allow organism-level codon usage frequency data to be joined to the relevant PDB structures and hence to the alignments. The join was not 100% efficient (i.e. certain PDB organisms were not matched in the Codon Usage database and vice versa) which reduced the effective number of protein alignments that could be used in the study.

Codon Usage totals with less than 20 coding sequences per organism were excluded on the basis that this minimum sample size would be required to produce realistic genomic averages. Knight [KNIGHT01] used the same Codon Usage database and reported that increasing the cut off to 50 or 100 coding sequences had no effect on the averages, save reducing the available dataset.

Network topology

The experiment used a modified version of the feed forward neural net employed by [LIN01]. The software was ported to Java (from C) to allow the tests to be run across various OS platforms. The new version of the software was modified to allow network topology and learning rates to be varied using a text configuration file, to allow rapid prototyping.

A range of different input data and network topologies were tested. In each case the first 20 inputs of the input layer represents a binary 1-of-c input for each of the 20 amino acids. For example, Alanine (Ala, A) would set the value of the first input unit to 1 and the remaining nineteen to 0. The output layer also features 20 units and follows an identical naming convention. The output layer uses a normalised exponential, or *softmax* activation function, which ensures that all output values lie in the range 0 to 1 and sum to unity. This allows the outputs to be treated as probabilities of class membership. Following propagation of the net, the value presented at the first output node represents the posterior probability of the input residue aligning with the target residue.

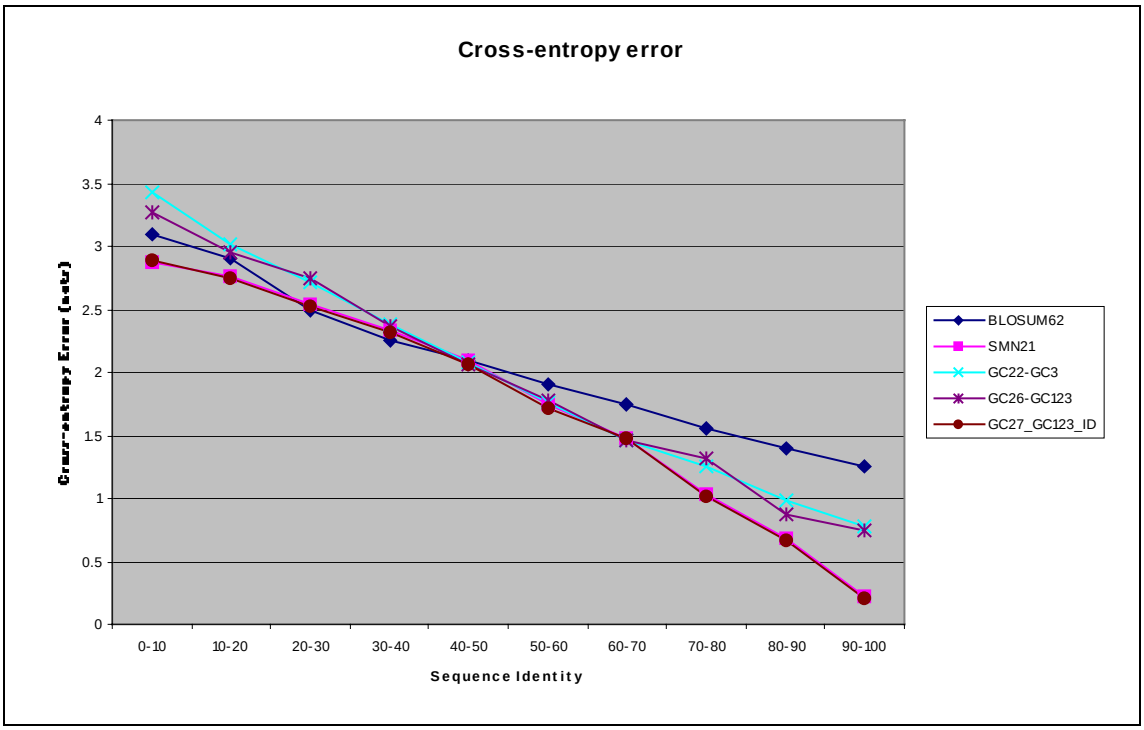
For each training iteration, one residue of an aligned pair was presented to the input layer, the other as the target for the output layer. Additional conditioning information (e.g. GC content for source and target organism) is also presented to the input layer.

The following table shows how the inputs and outputs might be configured for one training cycle with an Alanine/Proline alignment at an identity of 89%:

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	id
s	**1**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**0.89**
t	0	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	0	0	0	0	0	
	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	

The following section describes several variations of this standard 20*20 setup. In each case, different numbers of hidden units were tested empirically before an optimal rate of convergence was found. All inputs were normalised against the relevant maximum / minimum values, so as to improve the performance of the neural network.

Results



The graph shows the cross entropy errors for a range of neural network topologies and training sets. The motivation for each topology and the results obtained are discussed below.

SMN21

This is the original 20 + identity topology as reported by [LIN01], however it was repeated to validate the neural network set-up and the training/testing data set. The results obtained were comparable to those originally reported with the resulting probability matrix outperforming conventional matrices (BLOSUM62 shown in graph) across all identities.

GC composition

[KNIGHT01] presents a case for the causality of the correlation between genome GC content and frequencies of certain codons and amino acids, suggesting that nucleotide composition drives codon usage and not vice versa.

On this basis, the following GC-based input layer configurations were produced and evaluated. If successful, the use of GC data would have the advantage of being relatively compact as opposed to individual codon statistics, which would require a large number of input nodes.

GC22-GC3

This configuration featured two additional inputs; the total GC content of the 3rd codon position for both the source and target organism. [KNIGHT01] notes that the GC content at the third codon position correlates strongly with the overall GC content and points out that, for example, lysine (K) and arginine (R) are highly correlated with GC content and that that they can easily substitute for one another in proteins. We

would therefore expect that two organisms with the same GC bias would have a lower probability for K-R substitution than two organisms with opposing GC biases.

The results for GC22-GC3 show performance better than BLOSUM62 at identities of around 50% and above, but worse performance below this similarity level. GC22-GC3 performs worse than SMN21 at all levels of identity.

GC26-GC123

A variation on GC22-GC3, this configuration split the input data so that the GC total at each codon position was presented for both source and target residue. This was done to determine whether the additional positional information would improve the performance of the neural network.

The configuration performed marginally better than GC22-G3 at 0-20% identity and 80-100% identity, but slightly worse at 70-80% identity. At all other identities performance was identical with that observed with GC22-GC3; better than BLOSUM above 50%, worse that SMN21 at all identities.

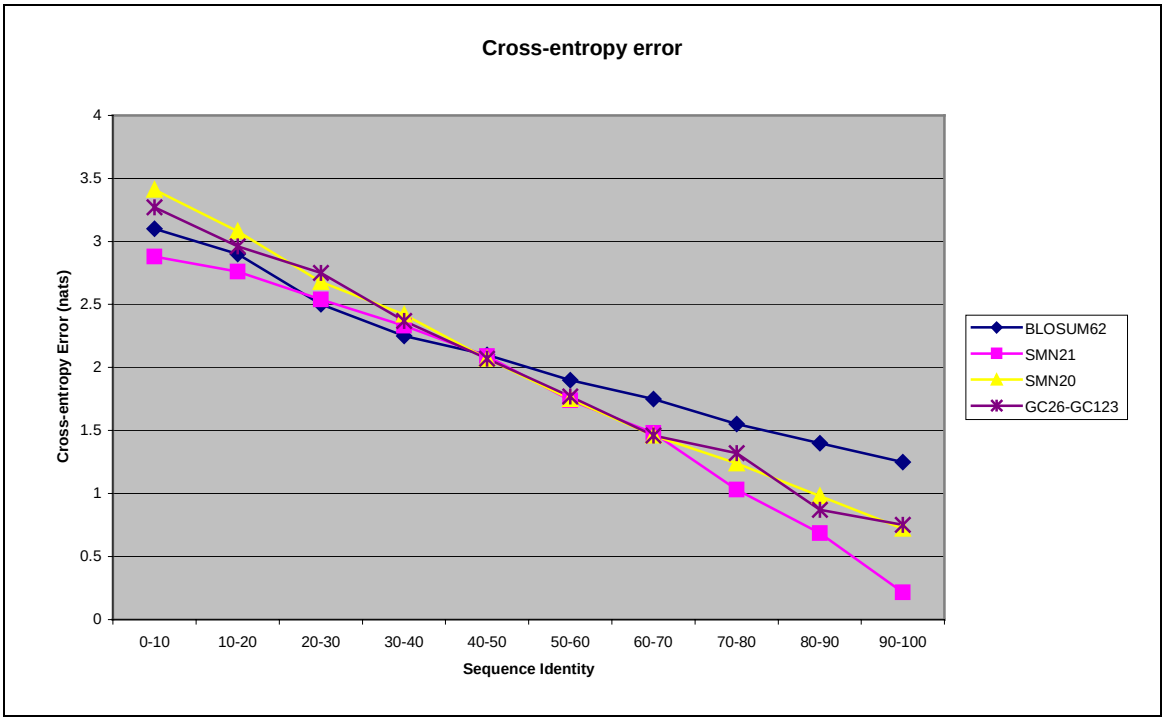
BLOSUM62

This result is included on the graph as reported in [LIN01] and serves as a useful benchmark for the other configurations.

SMN20

In order to isolate the effect of the conditioning inputs for the most effective non-identity network topology (GC26-GC123), a version of SMN21 was created, but without the identity input. The resulting SMN20 performed on a par with GC26-GC123 at all identities, which indicates that the bulk of any improvement over BLOSUM62 is largely a result of the neural network's ability to generalise alignments rather than the specific effect of the GC conditioning inputs.

The following graph shows how SMN20 performs in relation to BLOSUM62, SMN21 and GC26-GC123.



Others

The following configurations were also tested, however the results were worse than those described earlier and are therefore not reported in detail.

Base Composition

As GC alone may not always the best way to compare base composition between genomes and genes, an alternative configuration was devised which showed each base separately. This configuration presented 8 additional inputs, once for each base content for both source and target organisms i.e. A_{SOURCE}, C_{SOURCE}, G_{SOURCE}, T_{SOURCE}, A_{TARGET}, C_{TARGET}, G_{TARGET}, T_{TARGET}. Again, the values were normalised against the overall values from the CU database.

CUF128

This included the complete codon usage frequencies for both organisms as the conditioning inputs, however the network was incapable of converging with this many input nodes.

Dinucleotide composition

The relative dinucleotide abundances for each organism were derived from [GENTLES01] and used to condition the training inputs, however the results obtained were worse than without the conditioning data, suggesting that the additional data may have only served to add noise to the training inputs

Discussion

None of the matrices produced using organism-level statistics perform as well across sequence identities as those that included the sequence identity as a conditioning input, however they do perform better than traditional substitution matrices at certain identities.

This result was initially taken as a positive indication that organism-level statistics could help to produce more effective substitution matrices, however the SMN20 (with no conditioning input) was found to perform equally well.

The conclusion was that the neural network approach to producing substitution matrices is extremely effective even without the sequence identity and the incorporation of organism-specific base composition and codon statistics did not measurably improve the performance of the neural network. In fact, the data suggests that the additional data may have only served to add noise to the training inputs and, as such, reduced rather than improved the effectiveness of the net.

There are several possible explanations for the failure of this organism level data to have a clear positive effect on the performance of the neural network.

Firstly, the addition of the organism specific training data significantly increases the number of training alignments required in order for the neural network to properly generalise the relationships between amino acid substitution probabilities and the organism specific statistic. The method used to produce the alignment index from CATH used the hierarchical nature of the CATH families to ensure that each range of sequence identity was adequately represented, however there was insufficient data to ensure that each range of organism specific statistic (e.g. GC bias at codon position 3) was also adequately represented within these groups.

Secondly, while bacterial genomes differ in mean G+C and have small variation about that mean, Mammalian genomes all have approx the same mean G+C but with large variation within the genome [SUEOKA62]. It is therefore possible that systematic effects would be visible if only bacterial alignments are used. An attempt was made to generate a bacteria only dataset from the structural alignments, however the resultant dataset was deemed of insufficient size to proceed with, so it was not possible to test this hypothesis.

A final possibility is that there is in fact no organism-level statistic that can be used to predict inter-organism variations in substitution probabilities for amino acids.

Organism specific substitution groups

The neural network analyses based on organism-level base composition statistics did not produce a significant result, so attention was then given to organism specific substitutions, to see if any biases could be observed at the amino acid level. If some specific organism level biases could be identified, then it might be possible to relate these back to a more fundamental organism level statistic (e.g. GC bias).

Initial work used the BLOSUM matrix generation software [HENIKOFF92] to produce substitution matrices from BLOCKS data filtered for each organism, however no significant differences in the resultant matrices were evident.

Substitution groups [WU96] were then considered, as these allow for a more sensitive approach to detecting amino acid bias than simply filtering the alignment data at the organism level. Substitution groups could be generated as normal, but each organism's contribution to each group member would also be recorded. This

approach allows for the detection of organism specific amino acid biases *within* the constraints of each substitution group.

For example, if a given alignment conforms with the substitution group FWY, then one might assume that mutations between the three residues F,W and Y *at that position* are effectively synonymous. Accordingly, any organism-specific mutational pressure (e.g. compositional bias, dinucleotide abundance) is free to assume organism-specific equilibrium within these boundaries. If, in this example, we were to examine an organism with a high GC bias, then we might expect that organism to favour W at a conserved FWY position, as it has the highest GC content of the three amino acids. Such tendency might not be exhibited so clearly elsewhere in the alignment, or on average throughout the genome, as more vigorous functional and structural constraints would probably outweigh any weak underlying bias.

An alternative substitution group analysis program was derived from [WU96], which recorded the relative contribution of different organisms to each amino acid in each substitution group. Only organisms contributing to greater than 20 different alignments for each conserved group are included.

Results

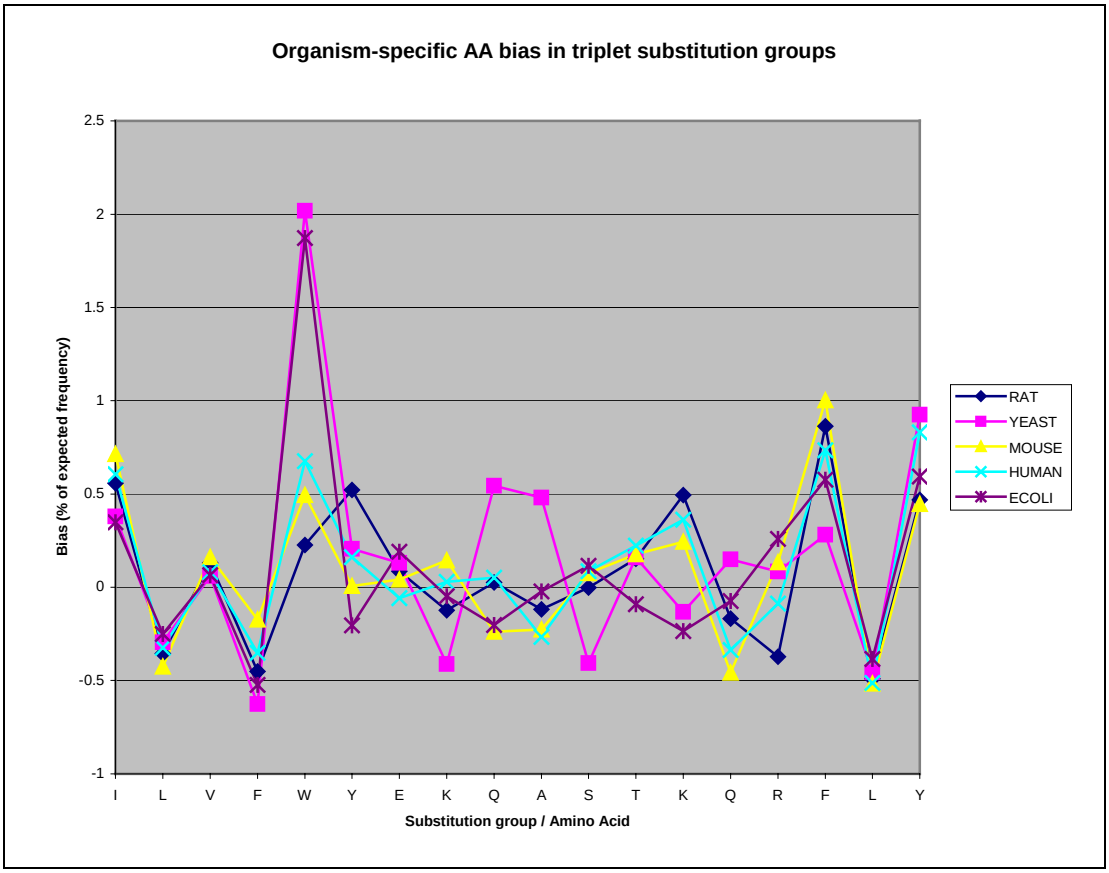
Result were obtained for substitution groups of size two and three. The results below examine the 6 conserved triplet substitution groups as described in [WU96], namely ILV, FWY, EKQ, AST, KQR and FLY.

The first graph shows for each organism how the frequency of each amino acid within each substitution group differs in relation to the expected frequency, calculated according to the relative abundance each member amino acid in the organism as a whole.

e.g. ILV in RAT

RAT	I	L	V
observed	0.353805	0.301736	0.344459
overall	0.048010	0.099018	0.064413
expected	0.227064	0.468298	0.304638

The bias is shown as a percentage of the expected frequency of that amino acid within the group, for example Tryptophan (W) occurs 200% more than would be expected in FWY for YEAST/ECOLI, but is relatively unbiased for other organisms.



In ILV, Leucine(L) is favoured in all organisms, particularly Mouse, Rat and Human. Isoleuceine(I) is under-represented to the same degree. Very little bias is shown for Valine by any of the organisms. A possible explanation at the codon level is that Leucine is coded for by 6 codons, whereas Isoleucine has only three.

The FWY triplet shows some extreme biases with Tryptophan (W) over represented in all organisms, particularly YEAST and ECOLI. Phenylalinine (F) is correspondingly under-represented in all organisms except MOUSE. The general bias towards Trptophan is probably explained by the fact that it is coded for by just a single codon, however it is not clear why W should be so favoured in YEAST and ECOLI

EKQ and AST are relatively unbiased for all organisms except YEAST, which is strongly positively biased for Glutamine (Q) and Alanine (A) and negatively biased for Lysine (K) and Serine (S).

KQR shows MOUSE, RAT and HUMAN with a shared positive bias to Lysine (K) and a negative bias to Glutamine(Q), while YEAST and ECOLI are negatively correlated. RAT and HUMAN are both negatively biased towards Arginine(R), whereas MOUSE, ECOLI and YEAST demonstrate a positive bias.

Finally, FLY shows some extreme biases which are followed by all organisms; Phenylalinine (F) and Tyrosine(Y) are over represented by between 50 and 100%, while Lysine(L) is underrepresented to a corresponding degree. Again the codon rule appears to apply, as F and Y are both coded for by just 2 codons each, whereas L has 6.

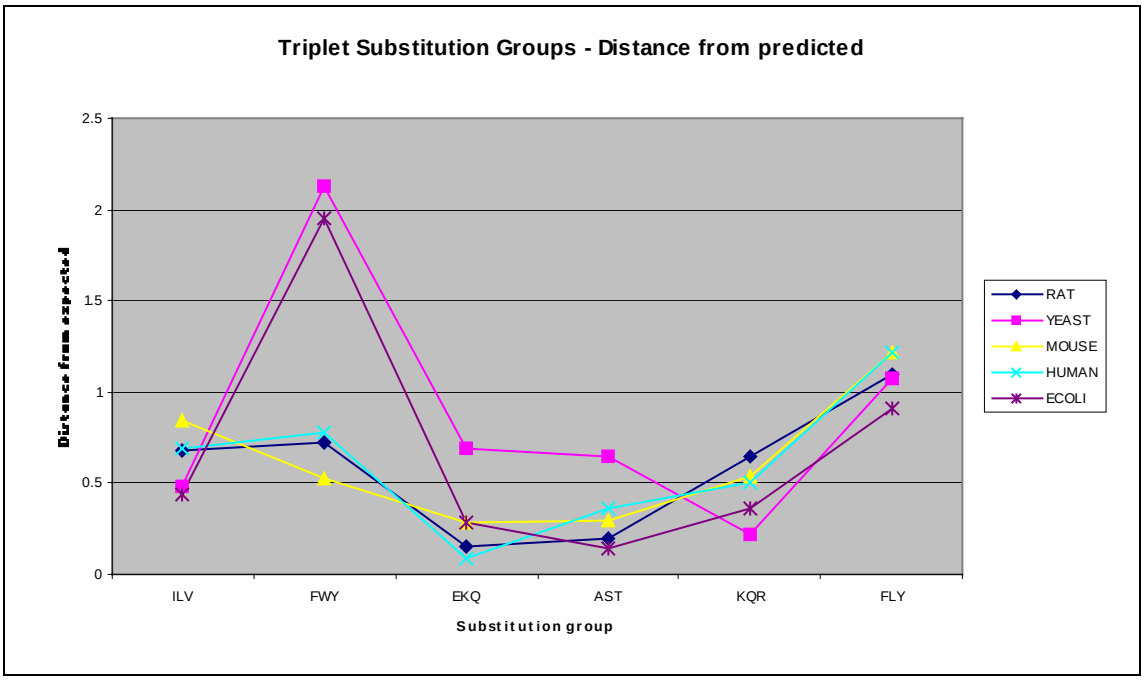
Of the 14 distinct amino acids featured in these triplets, four are present in more than one group (QKLF), which allows us to compare biases between groups. Q follows approximately the same bias in both groups, being under represented by MOUSE and over-represented by YEAST in both groups (EKQ,KQR). K is less consistent, even at the ordinal level (positive bias first):

- K:EKQ (MOUSE, HUMAN, ECOLI, RAT, YEAST)
- K:KQR (RAT,HUMAN,MOUSE,YEAST,ECOLI)

L is strongly under represented by all organisms in both groups (ILV, FLY)

F is under represented for all organisms in FWY and over represented in FLY, however the bias order is broadly preserved (ECOLI and YEAST have least F in both groups.)

The second graph shows the root squared distance from the predicted frequency for each substitution group.



From the group perspective, EKQ and AST are the close to the predicted distance for all organisms, except YEAST. FWY shows the broadest spread of distances, with YEAST and ECOLI showing a marked difference from the expected distribution. At the organism level, HUMAN, MOUSE and RAT all demonstrate broadly similar biases, as do ECOLI and YEAST.

Although some organism specific biases are evident the underlying causes are unclear.

Codon Substitution Groups

Following on from the organism-specific amino acid substitution groups, it was decided that an investigation at the codon level might provide further clues as to the cause of organism specific biases. This section of the report describes how the original approach to amino acid substitution groups described by [WU96] was modified to examine codons instead.

The production of Codon Substitution groups was also inspired by an examination of the *amino acid* substitution groups in the context of codon usage, which had revealed the conservation of physical properties of the genetic code. As predicted by Dayhoff et al, in addition to having similar amino acid properties noted by [WU96], each substitution group also had the property of requiring a minimal number of mutational changes in going from one member to another. For example, it was observed that all 9 doublet substitution groups identified by the substitution group software are separated by a single point mutation. Moreover, of the 6 conserved triplet groups, 3 of the groups can substitute via a single point mutation and 2 consist of doublets each separated by a single point mutation. The effect is also visible within larger groups, for example all of the amino acids in the FILMV group can substitute via a single point mutation at codon position 1.

Method

The method builds upon the amino acid substitution group work done by [WU96], plus software tools provided by [HUANG01]. The latest version of the BLOCKS database [HENIKOFF99] (blocksplus-01Aug01) was used to derive the groups.

A cross-index was built using part of the data from the Codon Usage Database [NAKAMURA00], as this provides both the SWISS-PROT protein accession and the NCBI accession for the complete nucleotide coding sequence. A web spider was created so that the complete CDS for each blocks protein could be obtained via the NCBI getfeat CGI service. The getfeat service requires the NCBI gi as an argument, so this was derived from the NCBI FASTA-format blast nucleotide flatfile DB.

The convert-blocks-to-columns.pl script [HUANG01] was modified so that codon columns were produced instead of amino acid columns. This was achieved by multiplying the amino acid offset present with each block record by three to create an offset into the DNA sequence. The conversion was validated by translating the resultant DNA sequence back to the amino acid sequence using the standard genetic code, which led to a small number of block records being rejected.

The blocks were pre-filtered to exclude SPTREMBL proteins as no index was easily available to allow getfeat lookups with these proteins. Additionally, proteins created using an alternative genetic code (e.g. mitochondrial proteins) were also excluded to ensure that the backtranslation validation operated correctly. (It would have been possible to isolate the translation table using a more sophisticated getfeat spider , however time did not allow this.)

The C based amino acids columns processor produced by [HUANG01] is a highly optimised program which makes use of bitmaps and large arrays for maximum speed.

It was initially supposed that the program could be modified simply to use bitmaps of length 64 to accommodate all possible codon substitution groups. The bitmap assumption was correct, however the program also makes use of several large fixed arrays of the order of 2^{20} (1048576), which would need to increase to size 2^{64} to accommodate codon data. It was clear that a more substantial modification would be required in order to run the program on available hardware, specifically to convert the arrays handling to use sparse arrays.

The software was recoded using Java as this provides a simple mechanism for creating sparse arrays (HashMap) and also arbitrary size bitmaps (BitSet). This modification made the software far less efficient, but at least was able to accommodate the creation of codon subsitution groups within certain constraints.

Specifically, the number of alignments (columns) that could be processed had to be limited (to 50,000 out of a possible 201187). More significantly, the maximum group size that could be processed had to be limited to 8. It would have been desirable to increase this to at least 25, so that the larger amino acid substitution groups (e.g. FILVY) could be accommodated in their codon form, however this was not possible with the available hardware and time to optimise the software. A more optimal solution would probably be to convert the original C version to use sparse arrays.

Results

Pairs

As would be expected, the most highly conserved codon substitution groups are those that code for the same amino acid. There are 19 significant substitution groups containing two codons, coding for 15 out of the 20 possible amino acids.

The first 9 substitution groups all code for amino acids that are represented by only 2 codons (CHDYEFNKQ). TGT/TGC (Cysteine) is the most significant group. Of the remaining 10 substitution pairs, 7 are from amino acids represented by 6 codons (SRLLRSL). One pair (ATT/ATC) is also present for Isoleucine, which is actually represented by three codons in the standard genetic code.

GCT/GCC are present for Alanine, however the other possible pair (GCA/GCG) are not significantly conserved. Similarly, GGT/GGC are present for Glycine, but not GGA/GGG.

Not all amino acids are represented as significant two codon substitution groups; Valine, Proline and Threonine are all absent. These three amino acids are coded for by 4 codons and so are conserved as higher order substitution groups.

As expected, Tryptophan is also absent as this is only coded for by a single codon.

It was noted that the first 12 significant substitution pairs are all separated by a transversion at the third position.

Triplets

The most significant triplet is ATT/ATC/ATA and contains all codons for Isoleucine from the standard genetic code. Isoleucine also features as a doublet (codon ATA

absent), however it is not clear why this doublet group should be conserved in preference to the triplet.

The next most significant substitution group (TAT/TAC/TGG) features codons from more than one amino acid, specifically Tryptophan and Tyrosine (WY). This pairing was not identified as a pair amino acid substitution group in [WU96], however did feature as part of larger conserved groups (FWY, FLWY). The pairing is also supported by a score of 2 in the BLOSUM62 substitution matrix.

Quads

The 5 most significant groups of four codons represent each of the five amino acids coded for by 4 codons (GPTAV). The compactness of the groups shows an inverse correlation with the relative mutability of amino acids as reported in [DAYHOFF78]. The exception to this is Valine, however this has a relatively high interference score.

TTT/TTC/TAT/TAC (FY) was the most significant amino acid substitution group in [WU96] and also scores highly in the BLOSUM62 substitution matrix. As both amino acids are coded from by two codons each, the group contains all possible codons for this amino acid pair. At the codon level, F may mutate to Y (and vice versa) through a transversion at the second position.

GAT/GAC/GAA/GAG (DE) was only the third most significant amino acid substitution group in [WU96]. The group contains all possible codons for this amino acid pair.

GAT/GAC/AAT/AAC (DN), CAT/CAC/TAT/TAC (HY) and GAA/GAG/CAA/CAG (EQ) are also conserved as both codon subsitution groups and amino acid substitution groups.

IV (the second most significant pair in [WU96]) is only ranked 16 in the codon quads. This probably because I and V are coded for by 3 and 4 codons respectively and would therefore be conserved in the higher order codon subsitution groups if the current version of the software was able to support them.

Further work

Once the software has been updated to cope with larger codon subsitution groups, the intention is to further divide the codon alignments along organism level boundaries to see if certain organisms favoured a certain type of codon at a conserved position. Conserved groups should provide a far more sensitive measure of biases at the DNA level, as the degrees of freedom are limited in a controlled manner.

Conclusion

The original goal of the project was to determine some organism level statistic that could be used to optimise amino acid substitution matrices for search operations involving that organism.

Initial experiments involving a neural network were not successful. The failure can probably be attributed to the use of base composition and codon usage data aggregated at the organism level, rather than at the level of individual aligned

proteins. This is because certain groups of organisms (e.g. mammals) show little variation in codon usage at the genome level, but broad variations for each gene. In the case of mammals, therefore, organism level base composition statistics are not an accurate predictor of compositional bias at the level of an alignment. The failure was exacerbated by insufficient training data given the high dimensionality of the conditioning inputs.

The results derived from examining organism specific relative amino acid abundance within amino acid substitution groups are far more encouraging, as some organism specific biases are evident within the lower dimensionality of each substitution group. It seems unlikely, however, that these substitution group biases could be generalised into organism specific substitution matrices, as they will be masked by other constraints (e.g. structure, function). The demonstration of organism-specific amino acid bias within substitution groups does suggest the possibility of optimising position specific substitution matrices and Markov models with organism specific probabilities.

The codon substitution group program has also shown some promise, however further work is required in order to handle larger substitution groups and organism biases. The work done in obtaining the DNA sequences for each BLOCKS alignment also provides a good starting point for evaluating the possible effect of dinucleotide Markov processes on amino acid substitution group bias.

References

[ALTSCHUL91]	Altschul, S. F. (1991). "Amino acid substitution matrices from an information theoretic perspective." J Mol Biol 219(3): 555-65.
[BERMAN00]	H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, P.E. Bourne: The Protein Data Bank. *Nucleic Acids Research*, **28** pp. 235-242 (2000)
[DAYHOFF69]	Dayhoff, M. O. (1969). "Computer analysis of protein evolution." Sci Am 221(1): 86-95.
[DAYHOFF79]	Dayhoff, M. O., R. M. Schwartz, et al. (1979). "A Model of Evolutionary Change in Proteins." Atlas of Protein Structure 5(Suppl. 3): 345-352.
[GENTLES01]	1: Gentles AJ, Karlin S. Genome-scale compositional comparisons in eukaryotes. Genome Res. 2001 Apr;11(4):540-6. PMID: 11282969 [PubMed - indexed for MEDLINE]
[GONNET92]	Gonnet, G. H., M. A. Cohen, et al. (1992). "Exhaustive Matching of the Entire Protein Sequence Database." Science 256(5062): 1443-5.
[HENIKOFF92]	Henikoff, S. and J. G. Henikoff (1992). "Amino acid substitution matrices from protein blocks." Proc Natl Acad Sci U S A 89(22): 10915-9.
[HENIKOFF96]	Henikoff, J. G. and S. Henikoff (1996). "Using substitution probabilities to improve position-specific scoring matrices." Comput Appl Biosci 12(2): 135-43.
[HENIKOFF99]	S. Henikoff, J.G. Henikoff & S. Pietrokovski, "Blocks+: A non-redundant database of protein alignment blocks dervied from multiple compilations", Bioinformatics 15(6):471-479 (1999).
[HUANG01]	"alphabet" software supplied by Jimmy Yi-Ming Huang, Stanford University.
[JONES92]	Jones, D. T., W. R. Taylor, et al. (1992). "The Rapid Generation of Mutation Data Matrices from Protein Sequences." Comput Appl Biosci 8(3): 275-82.
[KNIGHT01]	1: Knight RD, Freeland SJ, Landweber LF. A simple model based on mutation and selection explains trends in codon and amino-acid usage and GC composition within and across genomes. Genome Biol. 2001;2(4):RESEARCH0010. PMID: 11305938
[KOSHI95]	1: Koshi JM, Goldstein RA. Context-dependent optimal substitution matrices. Protein Eng. 1995 Jul;8(7):641-5.PMID: 8577693
[LIN01]	1: Lin K, May AC, Taylor WR. Amino acid substitution matrices from an artificial neural network model. J Comput Biol. 2001;8(5):471-81. PMID: 11694178 [PubMed - in process]
[LUTHY91]	Luthy, R., A. D. McLachlan, et al. (1991). "Secondary structure-based profiles: use of structure-conserving scoring tables in searching protein sequence databases for structural similarities." Proteins 10(3): 229-239.
[NAKAMURA00]	Codon usage tabulated from the international DNA sequence databases: status

	for the year 2000. Nakamura, Y., Gojobori, T. and Ikemura, T. (2000) *Nucl. Acids Res.* 28, 292
[ORENGO97]	Orengo, C.A., Michie, A.D., Jones, S., Jones, D.T., Swindells, M.B., and Thornton, J.M. (1997) *CATH- A Hierarchic Classification of Protein Domain Structures.* Structure. Vol 5. No 8. p.1093-1108.
[SUEOKA62]	Sueoka, N. On the genetic basis of variation and heterogeneity of DNA base composition. Proc Natl Acad Sci USA 1962, 48:582-592
[WILBUR85]	Wilbur, W. J. (1985). "On the PAM matrix model of protein evolution." Mol Biol Evol 2(5): 434-47.
[WU96]	Wu, T. D. and D. L. Brutlag (1996). "Discovering Empirically Conserved Amino Acid Substitution Groups in Databases of Protein Families." ISMB-96 3: 230-240.

Appendix A - Codon Substitution Matrix Raw Data

DNA	Protein	#	compact	interfere	separate	norm-sep
TGT TGC	C C	510.00	237.88	8.69	229.19	3.39E+08
CAT CAC	H H	520.00	186.81	20.17	166.64	2.47E+08
GAT GAC	D D	1293.00	201.23	41.77	159.45	2.36E+08
TAT TAC	Y Y	943.00	210.49	54.93	155.56	2.30E+08
GAA GAG	E E	1281.00	188.15	35.05	153.10	2.26E+08
TTT TTC	F F	1177.00	202.71	63.54	139.16	2.06E+08
AAT AAC	N N	867.00	162.92	24.17	138.75	2.05E+08
AAA AAG	K K	1075.00	160.70	23.28	137.42	2.03E+08
CAA CAG	Q Q	710.00	150.79	19.06	131.73	1.95E+08
AGT AGC	S S	314.00	87.76	30.98	56.78	8.40E+07
ATT ATC	I I	1096.00	133.39	91.37	42.02	6.22E+07
AGA AGG	R R	290.00	94.02	67.28	26.74	3.96E+07
TTG CTG	L L	839.00	91.63	72.59	19.04	2.82E+07
TTA TTG	L L	568.00	87.12	68.57	18.55	2.74E+07
CGT CGC	R R	378.00	113.71	99.56	14.15	2.09E+07
GCT GCC	A A	851.00	116.43	105.54	10.89	1.61E+07
TCT TCC	S S	445.00	99.29	88.44	10.85	1.61E+07
CTC CTG	L L	724.00	83.81	82.20	1.61	2.38E+06
GGT GGC	G G	1202.00	178.10	177.62	0.47	7.02E+05
ATT ATC ATA	I I I	253.00	91.37	16.81	74.56	1.10E+08
CCT CCC CCA	P P P	285.00	180.74	141.64	39.10	5.78E+07
TCT TCC TCA	S S S	145.00	88.44	60.84	27.60	4.08E+07
CGT CGC AGA	R R R	140.00	99.34	76.49	22.85	3.38E+07
TAT TAC TGG	Y Y W	34.00	18.37	10.70	7.67	1.13E+07
GCT GCC GCA	A A A	299.00	105.54	99.49	6.05	8.94E+06
GTT GTA GTG	V V V	209.00	87.42	83.61	3.81	5.64E+06
GTT GTC GTG	V V V	274.00	93.51	91.53	1.99	2.94E+06
TTA TTG CTG	L L L	232.00	68.57	67.29	1.28	1.90E+06
TTA TTG CTT	L L L	166.00	63.70	62.68	1.03	1.52E+06
GGT GGC GGA GGG	G G G G	296.00	200.49	5.16	195.33	2.89E+08
CCT CCC CCA CCG	P P P P	156.00	141.64	4.88	136.76	2.02E+08
ACT ACC ACA ACG	T T T T	89.00	107.49	6.91	100.58	1.49E+08
GCT GCC GCA GCG	A A A A	136.00	99.49	8.71	90.78	1.34E+08
GTT GTC GTA GTG	V V V V	102.00	83.61	22.62	60.99	9.02E+07
TTT TTC TAT TAC	F F Y Y	69.00	73.88	13.71	60.17	8.90E+07
GAT GAC GAA GAG	D D E E	64.00	45.79	7.18	38.61	5.71E+07
AAT AAC GAT GAC	N N D D	42.00	39.41	8.42	30.98	4.58E+07
TAT TAC CAT CAC	Y Y H H	12.00	31.81	10.30	21.51	3.18E+07
TCT TCC TCA TCG	S S S S	56.00	60.84	45.60	15.24	2.25E+07
TTG CTT CTC CTG	L L L L	107.00	77.68	64.12	13.57	2.01E+07
AAA AAG GAA GAG	K K E E	20.00	15.35	5.74	9.61	1.42E+07
CGT CGC AGA AGG	R R R R	55.00	69.28	65.01	4.28	6.33E+06
TTG CTG ATT ATC	L L I I	18.00	9.38	7.34	2.04	3.01E+06
CGT CGC CGA AGA	R R R R	55.00	76.49	74.79	1.71	2.52E+06
ATT ATC ATA GTT	I I I V	35.00	16.81	15.24	1.57	2.33E+06
CAA CAG GAA GAG	Q Q E E	17.00	12.72	11.76	0.96	1.43E+06
CTC CTG ATT ATC	L L I I	19.00	11.88	10.92	0.96	1.42E+06
ATT ATC ATA ATG	I I I M	19.00	8.43	8.25	0.17	2.59E+05
TTA TTG CTT CTC CTG	L L L L L	52.00	64.12	58.07	6.04	8.94E+06
ATT GTT GTC GTA GTG	I V V V V	29.00	22.62	20.36	2.27	3.35E+06
TCT GCT GCC GCA GCG	S A A A A	11.00	8.71	6.73	1.98	2.93E+06
CGT CGC CGA CGG AGA AGG	R R R R R R	13.00	67.66	0.00	67.66	1.00E+08
TTA TTG CTT CTC CTA CTG	L L L L L L	22.00	53.43	0.00	53.43	7.90E+07
ATT ATC GTT GTC GTA GTG	I I V V V V	12.00	20.36	0.00	20.36	3.01E+07