GGG201B Class Notes: From Information Theory to Amino Acid Similarity

What is information?: We all have an intuitive idea about what information is. If we ask a kid what their favorite food is and they say 'chocolate' or 'ice cream', it's not very informative. We expected an answer like that. If they say 'broccoli and cheese' we would probably remember that weird kid. Information is a degree of surprise. The more surprising the **message**, the informative the message. The mathematical definition of information assumes there is an information **source** that emits messages. The information of any particular message is given in equation 1. Information is almost always calculated in base 2 and therefore given the unit **bits** (binary digits). Sometimes you may see **nats**, which corresponds to the log base e. For our purposes, we will usually use base 2 for log, and bits will be the units. As an example, let's say that the answer to an average kid's favorite food is 'chocolate' 50% of the time. The information of 'chocolate' is simply the -log(0.5) or 1.0. Similarly, if the probability of 'ice cream' is 0.25, this is 2.0 bits. Any easy way to think about this is as powers of two: 0.5 is 2^{-1} and 0.25 is 2^{-2}. A message occurring ~1/1000 times has 10 bits of information (2^{10} = 1024).

Equation 1

$$I(m) = -\log_2 P(m)$$

I: information
m: message
P(m): probability of message

Most programming languages have a log function that returns values in log base e. To convert to base 2, simply divide by log(2).

What does this have to do with genomics?: In order to determine if two sequences are related to each other, we need (a) some way to align sequences (b) some way to determine if the alignments are significant. It is common to use information theory to determine the significance.

Information content: Generally, we are more interested in the information content of a source rather than the information of any particular message. A rich source of information provides you with a lot of surprise *on average*. Information content is simply the average information per message (equation 2). Information content is also called **entropy** or Shannon's Entropy because it was invented by Claude Shannon. An information source can be thought of as a frequency distribution (histogram). Some histograms are more *predictable* than others. For example, consider two coins, one is fair, the other a trick coin. The fair coin comes up heads 50% of the time. The trick coin is heads 90% of the time. Which one is more predictable? As an information source, which one has higher information content? What about a fair vs loaded die? What about DNA? What about DNA with highly biased composition? Try working some examples.

Equation 2

$$H = -\sum P_i \log_2 P_i$$

H: information content
Pi: probability of message i

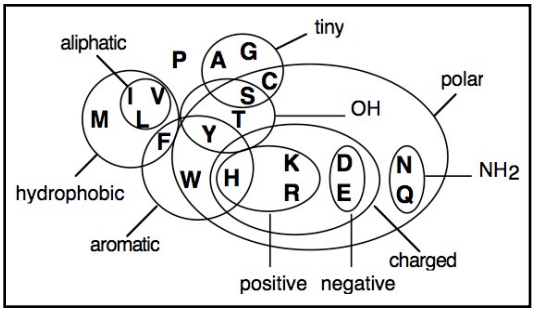
Relative entropy: Let's say you have nucleotide frequencies from several different genomes and you want to know which ones are the most similar to each other. How might you compare them? If you consider the sequences to be information sources and nucleotides to be messages, then you can use relative entropy to measure the similarity. This is also called **Kullback-Leibler distance** (equation 3). Strictly speaking, D(P||Q) is not always D(Q||P) but it's usually close. Note that if P or Q contains any zero probability values, you may get numerical errors (divide by zero or log zero).

Equation 3

$$D(P \,||\, Q) = \sum P_i \log_2 \left(\frac{P_i}{Q_i} \right)$$

D: distance
P: some frequency distribution
Q: other frequency distribution
Pi: probability of message i in P
Qi: probability of message i in Q

Codon bias: In the genetic code, some triplets code for the same amino acid, but not all codons are used with the same frequency. In different genomes the biased codons may not be the same. Codon bias probably exists because of translational efficiency. Not all tRNAs are expressed at the same level. As a result, highly expressed genes are optimized to use abundant tRNAs so they don't have to "wait" for rare tRNAs. Given that the codon bias of an organism is a kind of signature, if you found a gene with very different codon usage, you might expect horizontal gene transfer. You could use K-L distance to find outliers in a genome. In such an experiment, the source is the codon usage, and each message is a triplet.



Similarity: One of the most fundamental concepts in biology is that similarity is an indicator of common evolutionary history. Many phylogenetic trees are based on proteins. Before we can compare proteins, we need to be able to compare amino acids. How similar are any two amino acids? One could look at size, shape, charge, hydrophobicity, functional groups, etc. You can imagine a lot of different ways. The image at the right summarizes some of these ideas. Another way to determine similarity is by asking *how often can one amino acid substitute for another in a protein?* How would you design such an experiment? Replace an amino acid with another and perform some kind of assay for protein function? That would be great but it's a lot of work. Luckily, these experiments have already been performed billions and billions of times... in nature

Margaret Dayhoff: The 'mother of bioinformatics' aligned orthologous proteins by hand and published the multiple alignments in the Atlas of Protein Sequence and Structure. She and her colleagues produced multiple volumes of the atlas. Using known phylogenetic relationships, she was able to observe the rate at which one amino acid changes to another, which is called the **substitution frequency**. These changes are not symmetrical. That is, changing from G -> V ≠ V -> G. This is the truth, but we generally ignore this and average them. Today, there are too many proteins for a print publication. Databases like SwissProt, GenBank, and TrEMBL take place of the Atlas. Similarly, aligning all the sequences by hand would not be possible. There are several computer programs for creating multiple alignments (ClustalW, Dialign, T-Coffee).

```
HM40_CAEEL/188-247      RRKRRNFSKTSTEILNEYFLANIN..HPYPSEEVKQALAMQC.....NISVAQVSNWFGNKRIRYKK
PBX1_HUMAN/234-293      RRKRRNFNKQATEILNEYFYSHLS..NPYPSEEAKEELAKKC.....GITVSQVSNWFGNKRIRYKK
CUT_DROME/1746-1802     KKQRVLFSEEQKEALRLAFA...L..DPYPNVGTIEFLANEL.....GLATRTITNWFHNHRMRLKQ
CUTL2_MOUSE/1114-1170   KKPRVVLAPAEKEALRKAYQ...L..EPYPSQQTIELLSFQL.....NLKTNTVINWFHNYRSRMRR
CUTL1_MOUSE/1240-1296   KKPRVVLAPEEKEALKRAYQ...Q..KPYPSPKTIEELATQL.....NLKTSTVINWFHNYRSRIRR
Q22810_CAEEL/212-268    KKTKSPFTEHEIAVMMALFE...I..NKSPNHEEVQKLAVQL.....NLGYRSVANFFMNKRAKERK
Q9FNM1_ARATH/51-113     PKPEWKPNQHQAQILEELFI...G.GTVNPSLTSIKQITIKLQSYGEEVDDADVYKWFHNRKYSRKP
WOX9_ARATH/52-113       PKPRWNPKPEQIRILEAIFN...S.GMVNPPREEIRRIRAQLQE.YGQVGDANVFYWFQNRKSRSKH
WUS_PETHY/44-106        NSTRWTPTTDQIRILKDLYY...NNGVRSPTAEQIQRISAKLRQ.YGKIEGKNVFYWFQNHKARERQ
WOX6_ARATH/58-119       ATLRWNPTPEQITTLEELYR...S.GTRTPTTEQIQQIASKLRK.YGRIEGKNVFYWFQNHKARERL
WOX1_ARATH/73-134       VSSRWNPTPDQLRVLEELYR...Q.GTRTPSADHIQQITAQLRR.YGKIEGKNVFYWFQNHKARERQ
WOX2_ARATH/11-72        SSSRWNPTKDQITLLENLYK...E.GIRTPSADQIQQITGRLRA.YGHIEGKNVFYWFQNHKARQRQ
Q8LR86_ORYSA/41-102     ANARWTPTKEQIAVLEGLYR...Q.GLRTPTAEQIQQITARLRE.HGHIEGKNVFYWFQNHKARQRQ
Q9LIX7_ORYSA/24-85      STTRWCPTPEQLMMLEEMYR...G.GLRTPNAAQIQQITAHLST.YGRIEGKNVFYWFQNHKARDRQ
WOX4_ARATH/87-148       GGTRWNPTQEQIGILEMLYK...G.GMRTPNAQQIEHITLQLGK.YGKIEGKNVFYWFQNHKARERQ
WOX5_ARATH/21-82        KCGRWNPTVEQLKILTDLFR...A.GLRTPTTDQIQKISTELSF.YGKIESKNVFYWFQNHKARERQ
WOX5_ORYSA/11-72        KCGRWNPTAEQVKVLTELFR...A.GLRTPSTEQIQRISTHLSA.FGKVESKNVFYWFQNHKARERH
```

The **score for pairing amino acids** is shown in Equation 4. The score, *S*, for any two amino acids *i* and *j* is the log of the observed substitution frequency (Q_{ij}) divided by the expected substitution frequency. The observed frequency comes from counting occurrences in multiple alignments. The expected frequency is simply the chance that any two amino acids would be selected at random, so this is the product of the probabilities of the individual amino acid frequencies P_i and P_j.

Equation 4

$$S_{ij} = \log\left(\frac{Q_{ij}}{P_i P_j}\right)$$

Amino acid score examples

Given: $P_M = 0.02$, $P_L = 0.1$, $P_E = 0.04$

$Q_{ML} = 0.004$, $Q_{ME} = 0.001$, $Q_{LE} = 0.002$

Calculate:

S_{ML}, S_{LE}, S_{ME}

S_{ML} = log(0.004 / (0.02)(0.1)) = 1.0 bit

S_{LE} = log(0.002 / (0.04)(0.1)) = -1.0 bit

S_{ME} = log(0.001 / (0.02)(0.04)) = 0.32 bits

Scoring matrices: A **scoring matrix** is simply a table of all pairwise scores. The matrix produced by Dayhoff is called the PAM matrix (a rearrangement of acceptable point mutations). If you look at the scores in a matrix, you will note that they are all integers. What happened to values like 0.32 bits? They were scaled and rounded off. For example, one might scale 0.32 by a factor of 2 and then round off 0.64 to +1. Why? Historically, computers were slow and had little memory, so people used integers. There is no reason to do this now (floating point calculations are actually faster than integer today), but the practice of using integers for scoring matrices continues. Once the scores in a matrix are scaled and rounded off, the units are no longer bits.

BLOSUM62 Scoring Matrix

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	4	-1	-2	-2	0	-1	-1	0	-2	-1	-1	-1	-1	-2	-1	1	0	-3	-2	0
R	-1	5	0	-2	-3	1	0	-2	0	-3	-2	2	-1	-3	-2	-1	-1	-3	-2	-3
N	-2	0	6	1	-3	0	0	0	1	-3	-3	0	-2	-3	-2	1	0	-4	-2	-3
D	-2	-2	1	6	-3	0	2	-1	-1	-3	-4	-1	-3	-3	-1	0	-1	-4	-3	-3
C	0	-3	-3	-3	9	-3	-4	-3	-3	-1	-1	-3	-1	-2	-3	-1	-1	-2	-2	-1
Q	-1	1	0	0	-3	5	2	-2	0	-3	-2	1	0	-3	-1	0	-1	-2	-1	-2
E	-1	0	0	2	-4	2	5	-2	0	-3	-3	1	-2	-3	-1	0	-1	-3	-2	-2
G	0	-2	0	-1	-3	-2	-2	6	-2	-4	-4	-2	-3	-3	-2	0	-2	-2	-3	-3
H	-2	0	1	-1	-3	0	0	-2	8	-3	-3	-1	-2	-1	-2	-1	-2	-2	2	-3
I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4	2	-3	1	0	-3	-2	-1	-3	-1	3
L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4	-2	2	0	-3	-2	-1	-2	-1	1
K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5	-1	-3	-1	0	-1	-3	-2	-2
M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5	0	-2	-1	-1	-1	-1	1
F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6	-4	-2	-2	1	3	-1
P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7	-1	-1	-4	-3	-2
S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4	1	-3	-2	-2
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5	-2	-2	0
W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11	2	-3
Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	-1

Expected score: An important property of a matrix is its expected score (equation 5). To calculate this, one sums up the score contribution of each pairing (the contribution depends on the score and the expected frequencies of the individual amino acids). In general, the expected score of a matrix is negative.

Equation 5

$$Exp = \sum_i \sum_j P_i P_j S_{ij}$$

Relative entropy: The most important property of a scoring matrix is its relative entropy (equation 6). This is the bits per *aligned* pair of amino acids. To gain some intuition for this, imagine if the observed pairing (Q_{ij}) is equal to expected (P_iP_j). In this case, H = 0. That is, the scoring system reflects the random expectation. This is not so different from K-L distance if you compare to identical histograms. The distance is zero. H is maximum when what is observed is very different from what is expected. When does this happen? Continuing from the previous example where P_M = 0.04 and P_L = 0.1, the expectation is 0.004. If M is rarely observed to align with L, then Q_{ML} will be different from P_MP_L. If you create a scoring matrix from proteins that are all very similar to each other, there will be few substitutions, and Q_{ij} will be very different from P_iP_j. In biological terms, a scoring matrix from highly conserved orthologous proteins will result in a matrix with high H whereas a matrix derived from less similar proteins will have low H. If the alignments are random sequences with no real relationship, H will be zero.

Equation 6

$$H = \sum_i \sum_j Q_{ij} \log\left(\frac{Q_{ij}}{P_i P_j}\right)$$

BLOSUM matrices: Henikoff & Henikoff created their scoring matrices automatically. They did not restrict themselves to proteins with known phylogenetic relationships. To calculate the various Q_{ij} values, they assumed all pairings were possible. For any column in a multiple alignment, the counts of different amino acids is N_iN_j and the counts for the same amino acid is N choose 2. N! / 2! (N - 2)!