Public Databases

Enormous amounts of biotechnological data are now archived in the World's three major cooperative public databases

- European Bioinformatics Institute (EBI) -UK
- National Center for Biotechnology Information (NCBI) of the National Library Of Medicine -USA
- Genome Net –Japan

The first two provide powerful servers and analytical tools

Databases
Protein Sequences

NonRedundant

These entries embrace only those sequences that are annotated; that is, they have been completely determined and have been proven to be a gene. Their function and homologies have been characterized. No genes (theoretically) are duplicated. Alleles→?

On May17, 2016 there were 87,545,396 sequences in the NR database

nt

nucleotide
All non-redundant GenBank+EMBL+DDBJ+PDB sequences (but no EST, STS, GSS, or HTGS sequences)
1.6 million sequences
/databases/blastdb/db1/ncbi

nr

peptide
All non-redundant GenBank CDS translations+PDB+Swissprot+PIR+PRF
4.7 million sequences
/databases/blastdb/db1/ncbi

swissprot

peptide
SWISS-PROT protein sequence database
237,000 sequences
/databases/blastdb/db1/ncbi

pataa

peptide
protein sequences derived from the Patent division of GenBank
380,000 sequences
/databases/blastdb/db1/ncbi

patnt

peptide
nucleotide sequences derived from the Patent division of GenBank
3.7 million sequences
/databases/blastdb/db1/ncbi

pdbaa

peptide
protein sequences derived from the 3-dimensional PDB
29,318 sequences
/databases/blastdb/db1/ncbi

pdbnt

nucleotide
nucleotide sequences derived from the 3-dimensional PDB
7,051 sequences
/databases/blastdb/db1/ncbi

est_human

nucleotide
Human subset of GenBank+EMBL+DDBJ sequences from EST div
~ 8 million sequences
/databases/blastdb/db1/ncbi

est_mouse

nucleotide
Mouse subset of GenBank+EMBL+DDBJ sequences from EST div
4.8 million sequences
/databases/blastdb/db1/ncbi

est_others

nucleotide
Non-redundant database of all other organisms GenBank+EMBL_DDBJ EST sequences
~ 11.9 million sequences
/databases/blastdb/db1/ncbi

gss

nucleotide
Genome Survey Sequence, includes single-pass genomic data, exon-trapped sequences, and Alu PCR sequences
~ 10.5 million sequences
/databases/blastdb/db1/ncbi

sts

nucleotide
Non-redundant database of GenBank+EMBL+DDBJ STS divisions
922,406 sequences
/databases/blastdb/db1/ncbi

month.aa

peptide
All new or revised GenBank CDS translations + PDB + SwissProt + PIR + PRF released in the last 30 days
200,216 sequences
/databases/blastdb/db1/ncbi

month.nt

nucleotide
All new or revised GenBank+EMBL+DDBJ+PDB sequences released in the last 30 days
114,786 sequences
/databases/blastdb/db1/ncbi

mito.aa

peptide
database of mitochondrial sequences
2,222 sequences
/databases/blastdb/db1/ncbi

mito.nt

nucleotide
database of mitochondrial sequences
129 sequences
/databases/blastdb/db1/ncbi

alu.a

peptide
translations of select Alu repeats from REPBASE, suitable for masking Alu repeats from query sequences
1,962 sequences
/databases/blastdb/db1/ncbi

alu.n

nucleotide
select Alu repeats from REPBASE, suitable for masking Alu repeats from query sequences
327 sequences
/databases/blastdb/db1/ncbi

vector

Vector subset of GenBank (R), NCBI
911 sequences
/databases/blastdb/db1/ncbi

yeast.aa

peptide
Yeast amino-acid sequences
6,298 sequences
/databases/blastdb/db1/ncbi

month.est_human

nucleotide
non-redundant database of Human GenBank+EMBL+DDBJ EST sequences
61,643 sequences
/databases/blastdb/db1/ncbi

month.est_mouse

nucleotide
non-redundant database of Mouse GenBank+EMBL+DDBJ EST sequences
4,132 sequences
/databases/blastdb/db1/ncbi

month.est_others

nucleotide
non-redundant database of all other organisms GenBank+EMBL+DDBJ EST sequences
211,077 sequences
/databases/blastdb/db1/ncbi

Search Engines

Searches can find

- local and global alignments of two pairs
- multiple sequence alignments.
- structure

  proteins are characterized by their 3-dimensional structure particularly by *motifs*. (Finding homologies of *motifs* is called *threading*)

Identity Search

If we have discovered some protein (or so we think), and we have sequenced all of its amino acids, we might want to know if we have indeed discovered this or if someone else has first

If we assume that all the known annotated proteins are registered in an annotated database, such as NR from NLM or from SP (Swiss Protein) databank, by searching the larger of the two or both, looking for an identity (exact match) we would have our answer.

The Content of the Databases

But there are reasons why we might not get the right answer regarding its existence:

1. If it *has* been discovered, perhaps it has not been entered into any database.
2. Perhaps it *has* been discovered and entered some other database, but it is not fully annotated and does not exist in these 2 annotated databases.
3. But we need to think past an identity search and consider a *homology* search. Suppose our protein has 1000 amino acids and varied by just one from a protein in the database. In that case an identity search would fail but a similarity search would score very, very high

Similarity Search
(Homology)

If we could address the similarity issue, that is, our protein sequence is “like” some other protein sequence, then we stand to learn a great deal more.

Of course we need to define the word “like” in such a way that we can actually put a number on it.

Similarity

We must define our goal a little more precisely

- Do we want to find other proteins that are like our protein in specific regions (a local alignment) ?
- Do we want to get “big picture” sense of the whole thing, by fiddling with our protein sequence a little bit (inserting gaps) so that we have a high scoring global alignment.

Dynamic Programming Algorithms are Polynomial

The S-W and N-W methods give the right answer but are exhaustive.

- The good news is that these run in polynomial time ***O*** (n^2). .
- The bad news is for a 500 million amino acid database, this could take a while.

Say the the average alignment length (protein) is n=1000. Then it would take $(5x10^8$ sequences$)((10^3)^2$length$)$ time

We Need a Strategy

The answer to this is to come up a strategy to do both similarity and identity faster than polynomial time.

- Remember that an heuristic is not guaranteed to give the best answer, but it will always give an answer.
- If we are smart, we set things up so that all errors accumulate in our favor.

As is always the case, if we want speed in a heuristic, we may have to give up sensitivity, and conversely.

Basic Local Alignment Search Tool
BLAST

Let us discuss a very robust and rapid way to search a database – BLAST

- BLAST is fast.

  At one extreme, BLAST opts for speed but relinquishes sensitivity, while at the other extreme, the Smith-Waterman is slow, but very sensitive. Other searching tools, such as FastA, are somewhere in between.

- BLAST can run proteins (BLASTp) or Bases (BLASTn)

BLAST Algorithm
CONCEPT

- Don’t waste time looking where there is no chance of getting a high alignment score; instead, find those locations in the database that have the potential to provide alignment
- Explore the edges of areas of small local alignments, seeking longer alignments until incorporating edges no longer improves the score

BLAST Algorithm

- Preprocessing
  - Index the query** string for all words
  - Maintain a table pointing to the locations of those words in the database.
- Run-time processing
  - Break up a query sequence into overlapping small words, find **acceptable** neighbor words
  - look up the locations of all 3-letter words and their neighbors.
  - Extend the word and neighbor words using local alignment and no gaps until no longer feasible
- Post Processing
  - Compute alignment statistics for all alignments within a certain confidence level

**Newer algorithms actually index the entire database (target string) .They make a hash table, which is much smaller and faster than a full table lookup. Typically the hash function is modulo some appropriate prime number

- BLAST-like Alignment Tool (BLAT). (originally suggested by Altschul et al)
- Sequence Search and Alignment by Hashing Algorithm (SSAHA)
- MegaBLAST

BLAST Algorithm
Preprocessing Details

- A “word” is selected of a specific length. The default length is 3 amino acids, but it is a user parameter
- Every overlapping query sequence of 3-letter words is indexed into a table.
- The target database is large (163 million overlapping words of width 3) but the preprocessing is still linear

RunTime Details

- Potential neighbors are defined as all of the words that arise from changing each of the letters in the word to another possible letter.

  Restated, if there are 20 amino acids and we have 3 letters, the number of possible neighbors is 20^3 or 8,000.

  We have studied many algorithms for efficient lookup of exact matches. We are looking for exact matches for our 3-mer neighbor words.

SCORING

Run Time Details-Nucleotide Search What does ‘Acceptable’ Mean?

If we are doing a nucleotide search, a basic scoring scheme : +1 for match, -1for mismatch suffices. No substitution matrix is used.

Run Time Details
Neighbors

We must define a "real neighbor" by determining whether a potential neighbor deviates from the original word* by an 'acceptable' amount θ.

* Typically the word length for amino acids is 3, for DNA is 11

Run Time Details
Neighbors

Suppose the query looks like this:

…AIHPFSQ…….

And the target (database) contains:

….ARHPFSTAHAFSQ…..

As we slide along the query string, we examine the 3-mer HPF

…AI**HPF**SQ…….

The idea is to find all the locations in the target that contain the selected 3-mer, but also, we wish to identify which other 3-mers of 8,000 possible 3-mers look similar to HPF. If they are suitably similar, we will also use them to query the database.

What does 'similar' mean? The substitution matrix gives us a cost for replacing one amino acid with another. Using a similarity matrix (substitution matrix) to test all 8,000 possible 3-mers against HPF, we will get 8,000 similarity scores. Those whose similarity to HPF is above an arbitrary threshold* will be selected. These are called neighbors, and it is these neighbors that are used to query the database in addition to the 3-mer HPF itself.

The actual value is defaulted in BLAST to13; however, this is a parameter (θ) that can be set by the power user.

Example of a matrix for determining similarities between 2 amino acids

A	Ala	4																			
R	Arg	-1	5																		
N	Asn	-2	0	6																	
D	Asp	-2	-2	1	6																
C	Cys	0	-3	-3	-3	9															
E	Gln	-1	1	0	0	-3	5														
Q	Glu	-1	0	0	2	-4	2	5													
G	Gly	0	-2	0	-1	-3	-2	-2	6												
H	His	-2	0	1	-1	-3	0	0	-2	8											
I	Ile	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
L	Leu	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
K	Lys	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
M	Met	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
F	Phe	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
P	Pro	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
S	Ser	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
T	Thr	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
W	Trp	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Y	Tyr	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
V	Val	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4
		Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val
		A	*R*	*N*	*D*	*C*	*E*	*Q*	*G*	*H*	*I*	*L*	*K*	*M*	*F*	*P*	*S*	*T*	*W*	*Y*	*V*

Finding Neighbors

Consider the 3-mer HPF:
Its similarity score, when compared to itself, using the matrix in the previous slide, is:

```
H  P  F
H  P  F   =21
8   7  6
```

Now look at some possible neighbors. Of the three 3-mers shown as possible neighbors, only HAF meets the threshold and is kept as a neighbor.

```
H  P  F              H  P  F              H  P  F
H  V  F   =12        H  L  F   =11        H  A  F   =13
8  -2  6             8  -3  6             8  -1  6
```

In addition to the one neighbor found (HPF) out of the 3 potential ones tested, there are likely to be several more, typically 20-50 in all.

Considerations Substitution Matrices

- The substitution matrix is the linchpin of the similarity search. Fundamentally it is a statement of how alike two amino acids are. Many hydrophilic amino acids can be swapped with other hydrophilic amino acids without unduly deleterious effects on the resultant protein structure.
- How do we know that? - observation.

Considerations-PAM Matrix

Dayhoff and **P**oint **A**ccepted **M**utations

Margaret Dayhoff did global alignments on proteins that modeled evolutionary rates. The various proteins chosen represented different points along the evolutionary scale.

- The number of mutations from one sequence to the next is called the evolutionary distance.
- The sequences are 1 PAM distant if s_1 is changed to s_2 with an average of 1 amino acid change/100 amino acids in the sequence

- PAM matrices are numbered by PAM distances-large number→more evolutionary distance

Considerations-BLOSUM Matrix

Henikoff and Henikoff

Block Substitution Matrix

BLOSUM

- *Prosite* database is organized by domains/families of proteins
  - ~1000 entries (hand curated) at that time*
- A ***block*** is an ungapped local MSA from a group of related proteins
  - BLOCKS database has ~1200 such blocks,derived from Prosite.

*Release 20.129 of 26-Jul-2016 contains 1766 documentation entries, 1309 patterns, 1165 profiles and 1180 ProRule

Considerations-BLOSUM

Henikoff and Henikoff

Block Substitution Matrix

BLOSUM

- Henikoff and Henikoff calculated the frequency of mutation from one amino acid to the next in this BLOCKS evolutionary conserved system and compared the frequencies to a background rate
- Came up with the *log-odds ratio*

Substitution Matrices in General

All substitution matrices used in BLAST are based on the log-odds of a substitution in relation to the background frequencies of the query and target amino acids. The score of the substitution from amino acid *i* to amino acid *j*, $s_{i,j}$, is computed as

$$s_{i,j} = \frac{\ln \frac{t_{i,j}}{p_i p_j}}{\lambda}$$

where *t* is the transition frequency and p_i and p_j are the background frequencies. λ is a scale factor whose value does not affect the overall relationships of scores but whose actual value causes the normalization such that $\sum p_i p_j e^{t_{ij}\lambda} = 1$

λ figures prominently in interpreting the significance of matrix substitution scores

Considerations-BLOSUM

There are number of BLOSUM matrices provided to BLAST users.

- They are numbered according to sensitivity/generality
- The default is BLOSUM-62.

BLOSUM62 MATRIX

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

Considerations
Which Matrix?

- BLOSUM-62 is the default BLAST substitution matrix, however one can choose from many other PAM or BLOSUM matrices.
- Alternatively, one can pick some other structurally related matrices rather than evolutionary related ones
  - RISLER matrix
  - Identity matrix.

Run Time Processing Details
Finding Hits

Given a query string

- For each sliding query word, find its neighbors.
- The word and its neighbor words are used to do a database lookup, finding the locations within target sequences in the database where these words occur.

Hits

Here is an alignment of HPF in the query with the target database. Note that there is also an alignment of suitable similarity with one of HPF's neighbors, HAF

…AI**HPF**SQ…….

….AR**HPF**STA**HAF**SQ…..

Extending the local alignment to the right increases the score by 4, so the alignment score is now 25

…AI**HPF**SQ…….

….AR**HPF**STA**HAF**SQ…..

Extending the local alignment to the left decreases the score by 3, so the alignment score is now 22

…AI**HPF**SQ…….

….AR**HPF**STA**HAF**SQ…..

Run Time Processing Details

Extending Hits

Extension:

- Score the query word (or a neighbor) lined up with the target word (again using the substitution matrix)
- Extend the width of both the query and the target by 1 residue and recompute the score. Gaps are not allowed in this example.
- Continue the extension, now on the opposite side and recompute

Extending hits

Extending the local alignment to the right increases the score by 4, so the alignment score is now 25

…AI**HPF**SQ…….
….AR**HPF**STA**HAF**SQ…..

Extending the local alignment to the left decreases the score by 3, so the alignment score is now 22

…AI**HPF**SQ…….
….AR**HPF**STA**HAF**SQ…..

Extending the local alignment to the right decreases the score by 1, so the alignment score is now 21

…AI**HPF**SQ…….
….AR**HPF**STA**HAF**SQ…..

Extending the local alignment to the left increases the score by 4, so the alignment score is now 25

…AI**HPF**SQ…….
….AR**HPF**STA**HAF**SQ…..

The process continues until extension degrades the local alignment score

Run Time Processing Details
When to Stop Extending

- Define the **M**aximal **S**egment-**P**air (MSP) as the alignment* in which neither extension nor contraction can improve the score. Informally also called a ***hit***.

  A user parameter can limit how far the extension continues to be tested in the face of diminishing score

- Keep this MSP for later consideration

*Smith-Waterman local alignment, *e.g.*

Post Processing Details

The Karlin-Altschul-Dembo Statistics

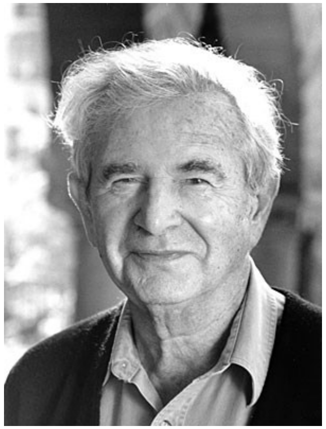
We get an alignment or many alignments for the entire query sequence, each associated with a score.

Now what?

Alignment Score– What does it mean?

Could the score of this hit have happened if the sequence were to appear randomly, with no biological significance?

The lower the probability of a random match, the more we believe in a biological relationship



Karlin-Altschul

A sophisticated analysis of this very question was put forth by Karlin and Altschul*

The analysis is explained masterfully in Mount’s book

Bioinformatics, Sequence and Genome Analysis Second Edition, David W. Mount, Cold Springs Harbor Laboratory Press, NY 2004

with excellent mathematical framework from

Introduction to Computational Biology, Maps, Sequences and Genomes Michael S Waterman, Chapman and Hall/CRC Press, Boca Raton 1995

*Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes

Proc. Natl. Acad. Sci. USA
Vol 8, pp 2264-2268, March 1990
Evolution

Significance of a Hit

To figure out whether an extended hit is of any importance, compare its score to what the best possible score would have been if the same sequence, given the same base (or amino acid) composition probabilities, and of the same length, were to occur with random bases (or amino acids)

If the score of the extended hit is better than the best possible random score*, chances are the hit is not random, but rather, is *significant*. The better the score beyond that, the lower the chances of the hit being random (the tail of the CDF describing the probability is approaching zero asymptotically)

*All else being equal

Significance of a Hit

In order to accept that proposition, we need to know

- How likely a random match is in the context of the search environment[*]. Restated: What are the considerations that account for the size of the query string to be matched and the target string(s) in the database against which the query string will be matched, and the scoring scheme employed?
- How maximal scores[**] are distributed in the context of the above.

[*] *i.e.,* all things NOT being equal

[**] The theory begins with the length of a match. *Via* a scoring scheme, the length of the match becomes the score of the alignment

Random or Real?

Our strategy to find the significance of a hit involves:

- Develop the theoretical basis for assessing the probability of a run of matching letters (hence a score) into a concrete statement of probability
- Draw on preexisting knowledge about probability distributions to create an formal expression for the probability distribution for database MSPs

Remember, in BLAST, we need to consider this question:

Are there any scores of alignments arising from naturally occurring sequences that are significantly higher than those that are artificially generated? If so, what is the probability of such an alignment ?

Jumping to the *denouement*..

.

- We convert the idea of a match-length into match-score, using some sort of scoring scheme. Each scoring scheme will yield a normalizing parameter for use in the final expression for an expected score (Karlin-Altschul theorem).

- By making a simplifying assumption, we derive the Karlin-Altschul expression for the probability of finding some score that exceeds a specified score. This is what we ultimately seek to answer the question 'how probable is it that the query sequence could find a match of some score *s* or higher, purely by chance in the database?'

- Using the modal and decay parameters from the Erdös and Rényi model for matches, we substitute into the expression for the Gumbel survival distribution to arrive at an expression for probability of a match at random.

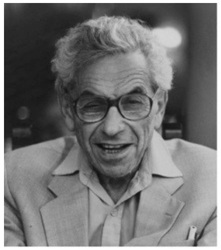
OUTLINE* Significance of an Alignment in BLAST

- Determination of the expected length of the longest identity alignment (match) in a set of trials if nucleotides were produced at random at the same frequency as occurs naturally
- Change in parameter convention to make expectation of longest match in matching trials relevant to nucleotide database searches
- Surrogacy of alignment score for match length: the expected (mean) extreme score
- Changing expectation of maximum length (now maximum score) into a probability using Poisson distribution as the intermediary
- Distribution of that probability
  - Extreme values: empirical evidence for probability distribution: the Gumbel distribution
  - Evident Gumbel distribution for the expression for probability theoretically derived
  - Tying Gumbel distribution mathematically to the observed probability distribution and the expression of probability in the context of a Gumbel distribution
- Back to expectation of HSPs and E scores

*The following development follows Mount's development of this complicated issue very closely.
Bioinformatics, Sequence and Genome Analysis Second Edition, David W. Mount, Cold Springs Harbor Laboratory Press, NY 2004

Runs of matches

Consider matching two strings, say, nucleotides. The probability of a match is p in each position. Therefore, it stands to reason that the joint probability for a run of R positions, is p^R



Erdös and Rényi

On a New Law of Large Numbers

The expectation of the number of consecutive matches (runs) R in *n* possible match-up trials is given by

$$E(\#\, runs\ of\ \ length\, R) \cong np^{R}$$

where *p* is the probability of a single letter match.

If the longest run is unique, then $1 = np^{R_n}$

Solving for R_n, $R_n = \log_{1/p}(n)$

If the match-ups were Heads/Tails with a fair coin, *p* would be .5 and the answer would be $R = \log_2 n$. For match-ups of nucleotides that are equally probably (not the case in nature) *p* would be .25 and our expression would be $R = \log_4 n$.

Sidebar: Moving between Number Systems in the Log World

Very Handy Mathematical Manipulation

$$\log_N x = \log_e x / \log_e N$$

So, for example, what is the $\log_2$ of 25?

Answer: $\log_2 25$= ln(25) / ln(2)=3.219 / 0.693 = 4.64

This makes sense because we know in our heads that $\log_2$ of 16 is 4 and $\log_2$ of 32 is 5

The Probability of a Hit Being a Random Sequence

- The theorem of Erdös and Rényi estimates the number of consecutive same outcomes (*sci* matches), given a sequence length and the probability of the outcome.
- In our case, the outcome would be a letter in the query sequence matching* a letter in the target sequence

*In the context of a definition of a ‘match’ (similarity, for example)

The Probability of a Hit Being a Random Sequence

So, how long a run (R) of pre-specified letters in a string of length n might we expect?

Answer: $R = \log_{1/p}(n)$

But we can slide the query (length m) along the target length n, so our space to find a longer run of matches is extended.

So our revised answer, accounting for the sliding*, is

$$R = \log_{1/p}(mn)$$

*We are not going to bog down on the problem of sliding off the ends- perhaps in a higher level course

Search Space

- m is the query length
- m is the target length
- mn is the search space

What about the ends, particularly if the query is very long?

L is the average length of an alignment

So, the effective search space is (m-L)(n-L)

Many single letter matches occur here, particularly with sliding, but there is only one match-up of R (run length) 4, seen after sliding the query string to the right by 3 letters

A	*C*	*G*	*G*	*A*	*C*	*T*	*T*	*T*	*G*	*T*	*G*	*A*	*A*	*G*	*C*	*T*	*A*
C	*C*	*A*	*T*	*C*	*G*	*G*	*A*	*C*	*T*	*T*	*G*	*C*	*T*	*C*	*A*	*A*	*T*

In fact, this is a ***local alignment***. For this alignment, in this search space, Erdös and Rényi's theorem would predict an expected length of 4.17. Actually, adjusting for end-effects, the expected value is 3.9

Waterman

Arratia ,Gordon and Waterman *et. al.* refined this formula to account for the expectation of a non-match and some other tweaks

$$R = \log_{1/p}(mn) + \log_{1/p}(q) + .577\log(e) - 0.5$$

The values of *p* and *q* depend on base composition (for example, ~0.25 in DNA), reflected in K. The expression is then simplified to

$$E(R) = \log_{1/p}(Kmn)$$

$$Let\ \ \lambda = \ln(\tfrac{1}{p})$$

Because $\log_N x = \log_e x / \log_e N$

$$then\ \ E(R) = \frac{\ln(Kmn)}{\lambda}$$

The above then relates the expectation of R (the longest match) to the ln of the product of query and target lengths. Restated, this is the **mode.** The formula needs to include another term if it is to consider mismatches.

Behavior of Score

Now, taking this result $E(R) = \frac{\ln(Kmn)}{\lambda}$

and recognizing that the expectation of the longest match length R is **directly related** to the expected maximum score S, it is ok to write

$$E(S) = \frac{\ln(Kmn)}{\lambda}$$

where the value of K may again have been tweaked based on the scoring rules or scoring matrix.

E(S) is the expected score, or the **mean** score.

One more possible step…

The theory developed so far estimates the mean ***score*** in a search space of size *mn* with parameter K

But we are interested in the ***probability*** of a score, not the score itself. Specifically we would like to know how probable would hits be where the hits have a score that would exceed some score *S*.

The key is in the Poisson distribution. It is the ‘counting’ distribution. It tell us that, given some mean number of events in a specified time period, what the probability is that there will be *c* occurrences in that time period.

integer

Poisson Density

Note that the value of the random variable in this Poisson density is an integer. Accordingly, we signify this by writing *p(c),* rather than *p(x)* as we do for a continuous r.v., usually denoted by *x*

$$p(c) = e^{-\mu} \frac{\mu^c}{c!}$$

The is the parameter μ and the variance is likewisemean μ

What if we asked the probability that **no** score *x* would exceed some given score *S*

We can model this in a Poisson distribution : $p(c) = e^{-\mu} \frac{\mu^c}{c!}$

where c=0: $p(0) = e^{-\mu} \frac{\mu^0}{0!} = \boxed{e^{-\mu}}$

But we know what μ is; we just now derived it from developing the Erdös and Rényi theorem: $E(S) = \mu = \frac{\ln(Kmn)}{\lambda}$

So, the probability that no score would exceed *S*, then, is:

$$p(S < x) = e^{-E(S)} = e^{-Kmne^{-\lambda x}}$$

And the probability that ***some*** score would exceed *S*, then, is:

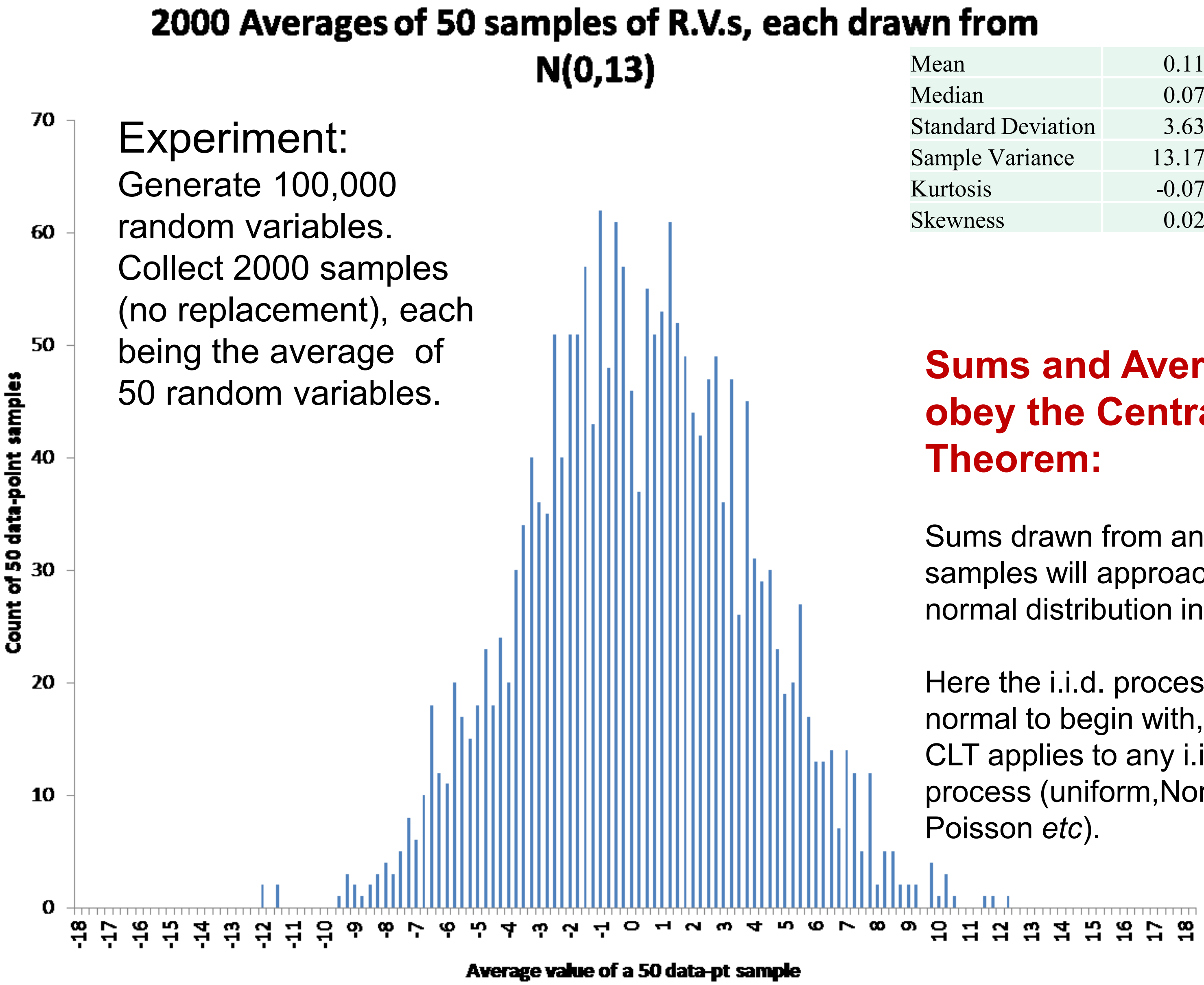
$$\boxed{p(S > x) = 1 - e^{-Kmne^{-\lambda x}}}$$

The above equation is a major result; we have now traveled the road from expected run-length to the ***probability*** of one or more scores exceeding a threshold.

A Different Perspective: Extreme Values

Keep in mind that we have culled out the highest scoring pairs from all matches. This changes the statistical framework. We note that these scores do not appear to distribute normally, but instead appears to exhibit an extreme value distribution. This is consistent with the nature of the random variables being distributed (just high sores, not all scores).

A NUMBER EXPERIMENT



Mean	0.11
Median	0.07
Standard Deviation	3.63
Sample Variance	13.17
Kurtosis	-0.07
Skewness	0.02

Experiment:

Generate 100,000 random variables. Collect 2000 samples (no replacement), each being the average of 50 random variables.

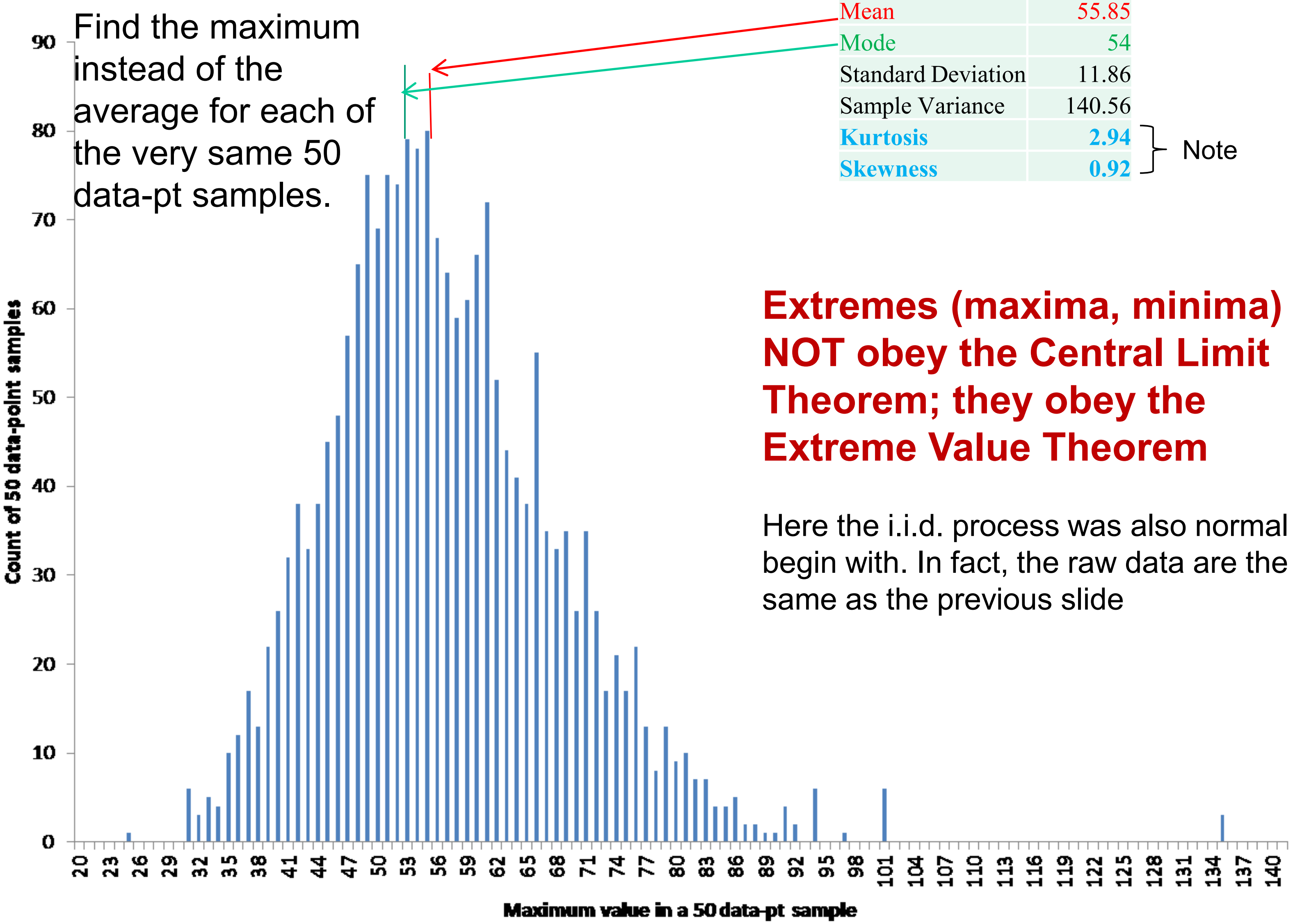
Sums and Averages obey the Central Limit Theorem:

Sums drawn from any i.i.d. samples will approach a normal distribution in the limit.

Here the i.i.d. process was normal to begin with, but the CLT applies to any i.i.d. process (uniform,Normal, Poisson *etc*).

A NUMBER EXPERIMENT continued

Find the maximum instead of the average for each of the very same 50 data-pt samples.



2000 Maxima drawn from 50 samples of R.V.s, each drawn from the same data

Mean	55.85
Mode	54
Standard Deviation	11.86
Sample Variance	140.56
Kurtosis	**2.94**
Skewness	**0.92**

Note

Extremes (maxima, minima) do NOT obey the Central Limit Theorem; they obey the Extreme Value Theorem

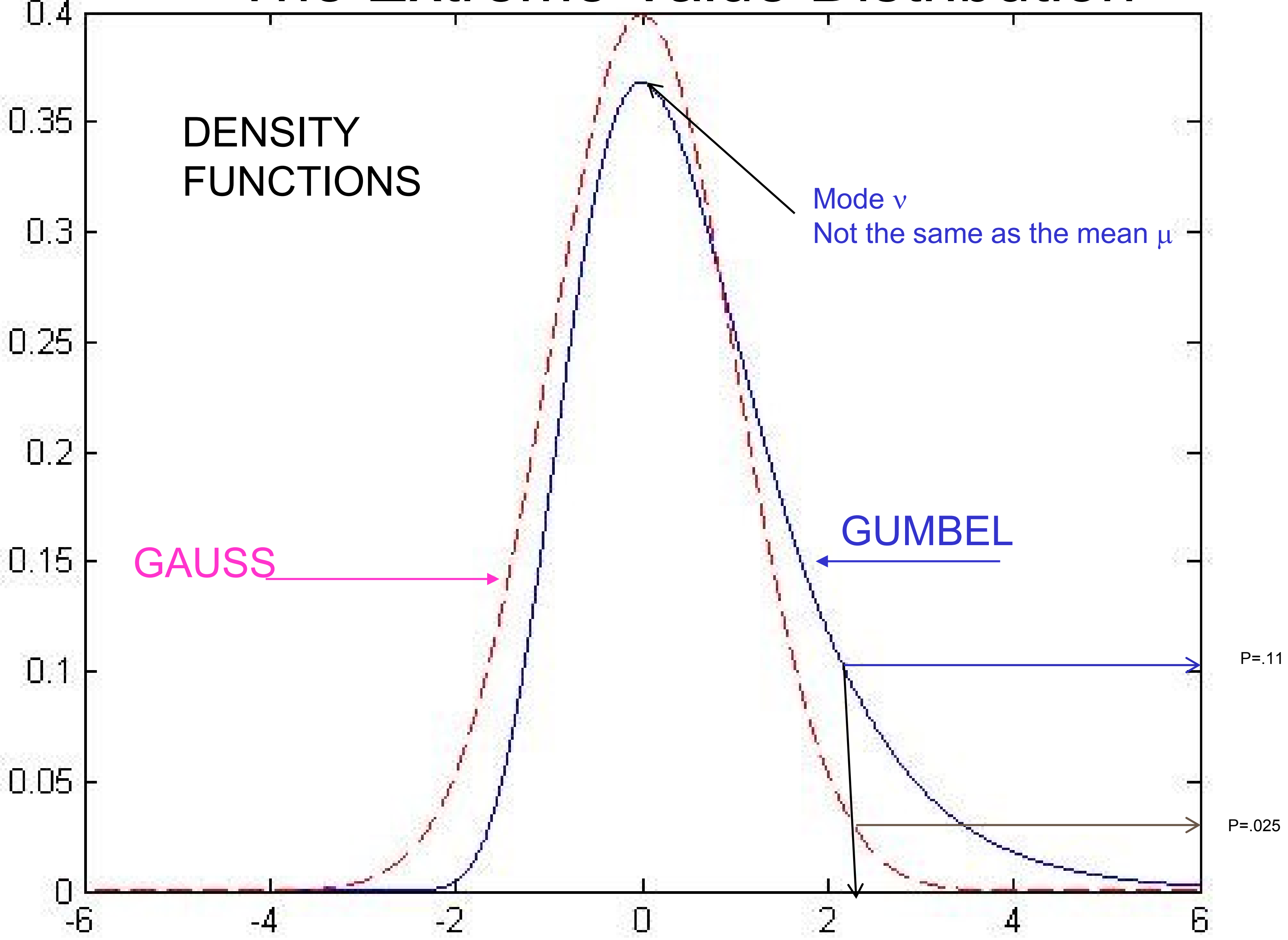
Here the i.i.d. process was also normal to begin with. In fact, the raw data are the same as the previous slide

HSPs are maxima

The Extreme Value Theory tells us that the distribution of HSP scores must be convergent to one of three extreme value distributions: Fréchet, Weibull, or Gumbel

The most likely is a ***Gumbel distribution*** because, unlike the others, it is not constrained on the *x*-axis.

The Extreme Value Distribution



DENSITY FUNCTIONS

Mode ν
Not the same as the mean μ

GUMBEL

GAUSS

P=.11

P=.025

Behavior of extreme numbers The Extreme Value Distribution

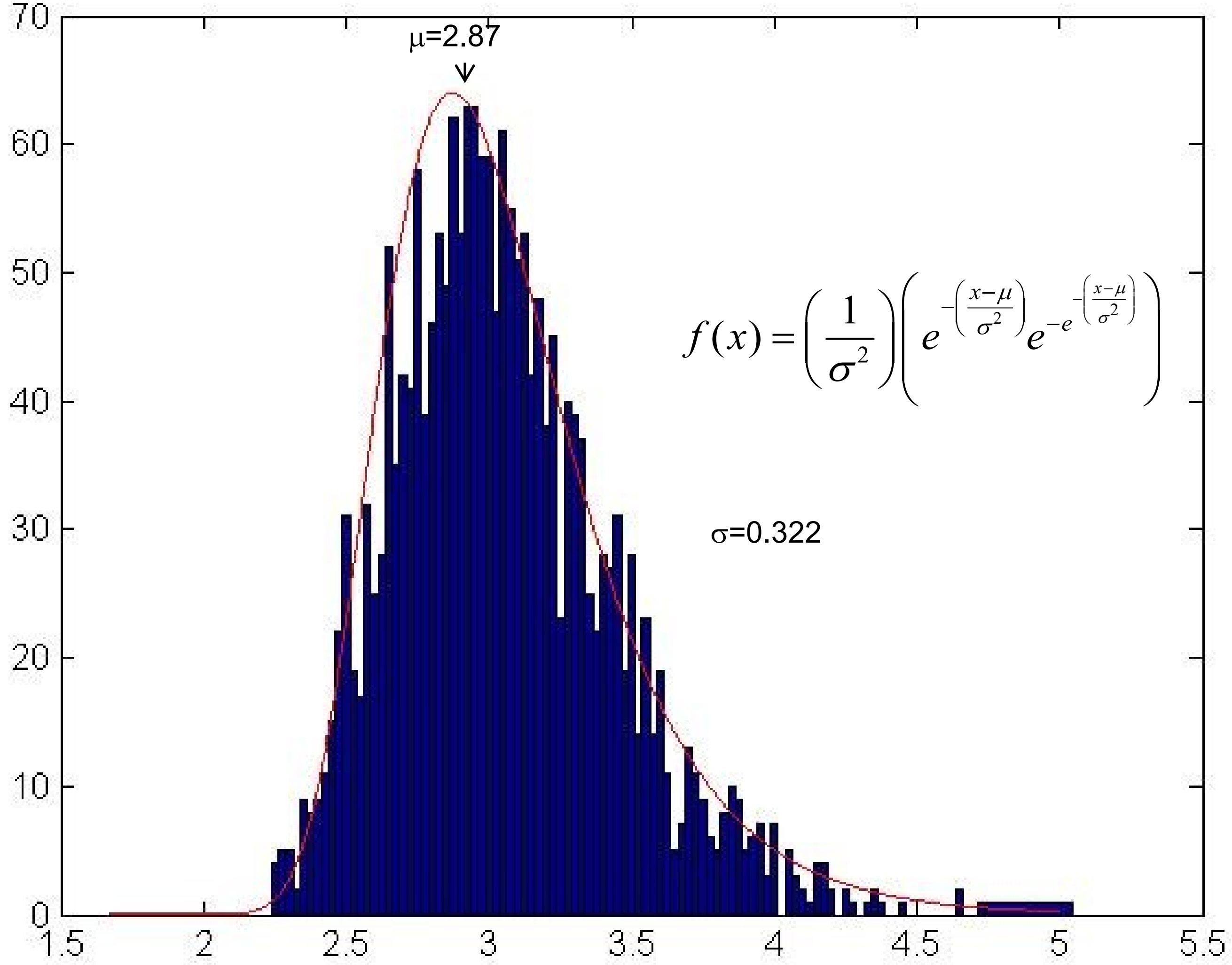
- The ***density*** is ***precisely specified*** by the Gumbel distribution
  - Developed by Gumbel for extreme statistics

$$f(x) = \left(\frac{1}{\sigma^2}\right)\left(e^{-\left(\frac{x-\mu}{\sigma^2}\right)} e^{-e^{-\left(\frac{x-\mu}{\sigma^2}\right)}}\right) \quad \textit{for}\ \text{maxima}$$

$$f(x) = e^{-x} e^{-e^{-x}} \quad (standardized\ \ form)$$

$$= f(x) = e^{-x-e^{-x}}$$

Experimental data fitted by a Gumbel EVD



μ=2.87

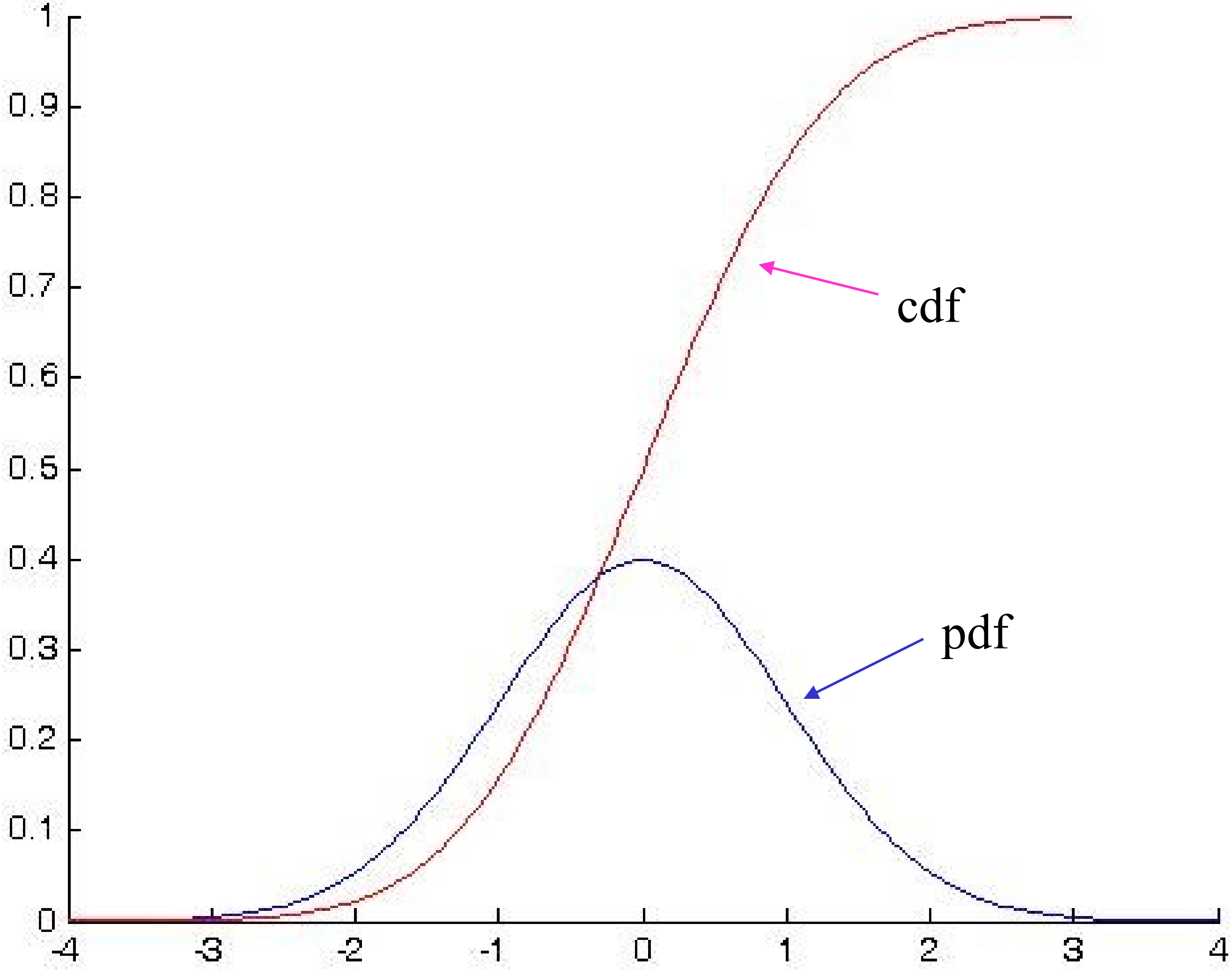
$$f(x) = \left(\frac{1}{\sigma^2}\right)\left(e^{-\left(\frac{x-\mu}{\sigma^2}\right)} e^{-e^{-\left(\frac{x-\mu}{\sigma^2}\right)}}\right)$$

σ=0.322

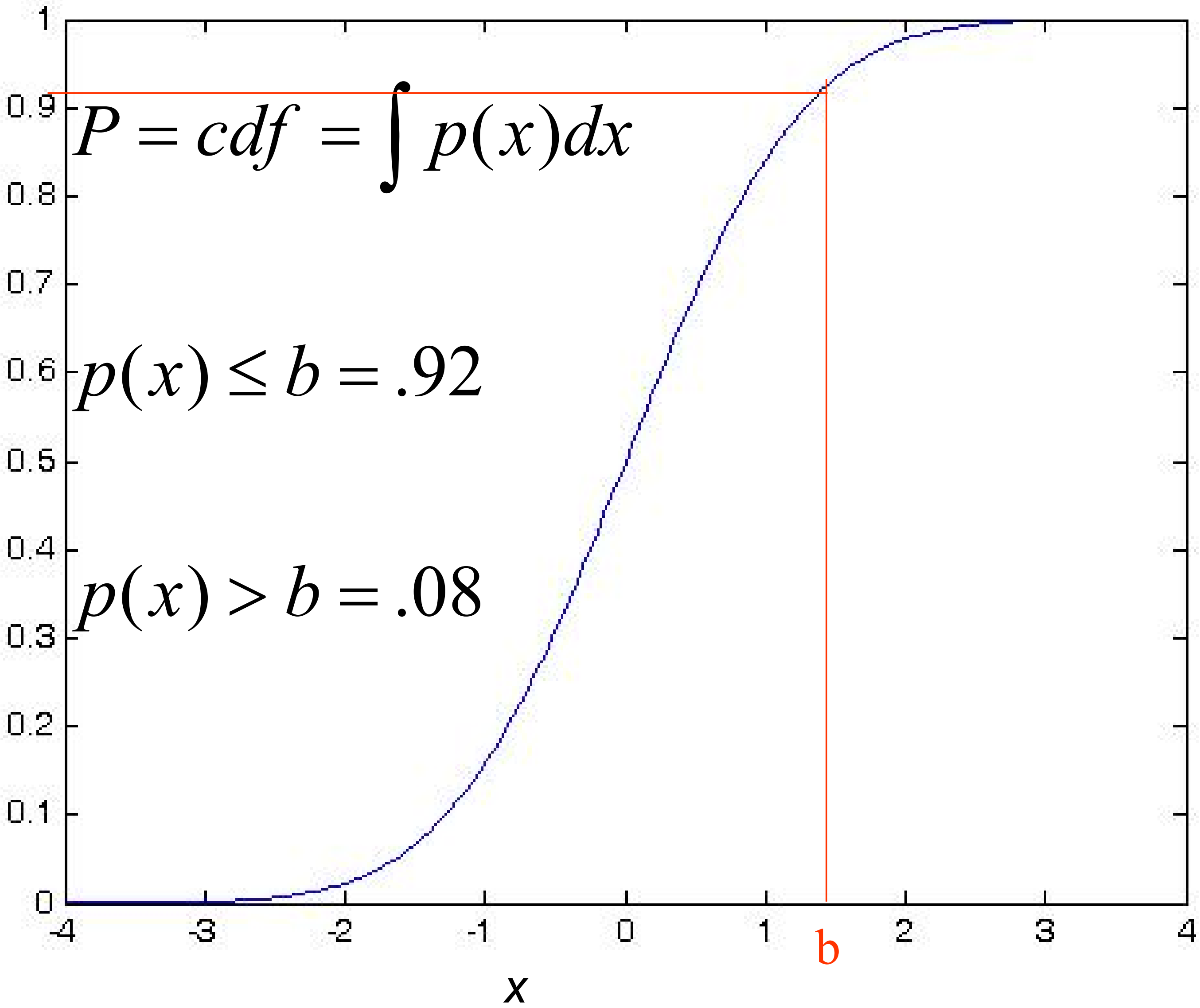
Cumulative Distributions Functions (CDF)

- When you need a total probability of all events leading up to an event of interest, you need a ***cumulative*** distribution, not a probability density function.

Gaussian Density and Cumulative Distribution



Gaussian Cumulative Distribution Function (cdf)



$$P = cdf = \int p(x)dx$$

$$p(x) \leq b = .92$$

$$p(x) > b = .08$$

The Extreme Value Cumulative Distribution Function

To obtain the cumulative distribution function, *i.e.* prob (score < x), we must integrate the standardized density function

$$\int e^{-x-e^{-x}} = e^{-e^{-x}}$$

and, as you would expect, the survival curve is

$$\boxed{1-e^{-e^{-x}}}$$

This expression is a major result, giving a formal structure to the result derived from Erdös and Rényi. It now remains to tie the two together

Gumbel Survival Curve Adjusted

The Gumbel distribution has moments just as most distributions do: a mean μ and a standard deviation σ. These moments can be directly tied to the parameters also characterizing the Gumbel distribution, the mode ν and the decay constant λ. This λ can also be derived from the substitution matrix as a normalizing factor.

So, to express the survival function in terms of experimental parameters, we can write

$$p(S \geq x) = 1 - e^{-e^{-\lambda(x-\nu)}}$$

where ν is the mean (actually the characteristic, or modal, value) and λ is a normalizing parameter (decay constant)

Simplification to get a final expression

Remember from Erdös and Rényi that the modal value ν is $\frac{\ln Kmn}{\lambda}$

$$p(S \geq x) = 1 - e^{-e^{-\lambda(x-\nu)}}$$

$$p(S \geq x) = 1 - e^{-e^{-\lambda\left(x-\frac{\ln Kmn}{\lambda}\right)}}$$

$$\Downarrow$$

$$p(S \geq x) = \boxed{1 - e^{-Kmne^{-\lambda x}}}$$

Compare this with the result from Erdös and Rényi !!!!

Finally, a very practical simplification to clear an exponent layer

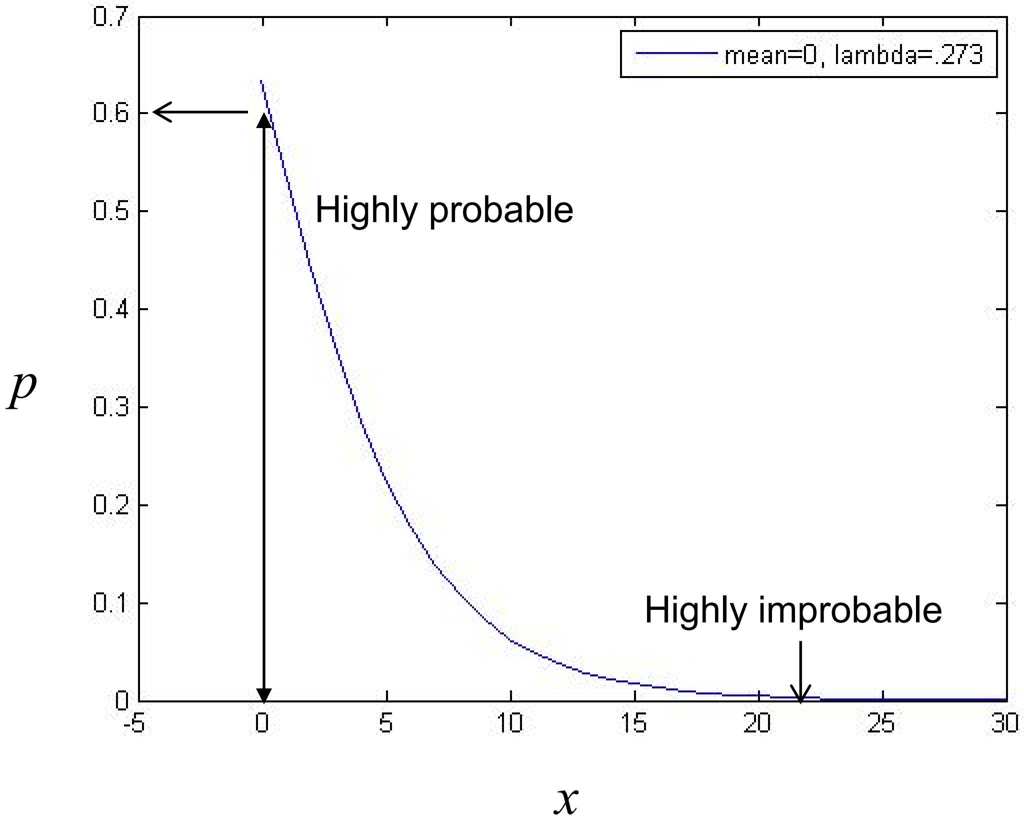
$$p(S \geq x) = \lim_{x \to \infty}\left(1 - e^{-Kmne^{-\lambda x}}\right) = \boxed{Kmne^{-\lambda x}}$$

The Extreme Value
Cumulative Gumbel Distribution Function
Improbability of Random High Scores (Survival)

$$x = score$$

$$p = 1 - e^{-e^{-\lambda(x-\nu)}}$$

p is the probability of any score ≥ *x*



Summary

- This last relationship gives us the probability that a **score** in excess of a certain value would happen randomly.
- The distribution of that probability is not normal, but is an EVD
- The larger search space, the larger the expectation of a match

The BLAST E-value

- This is different- it is the number of matches that exceed the mean extreme score
- We have already identified the mean extreme score as E(S). So the E-value is the that expected number of hits with score ≥ S but with database size D

$$E = 1 - e^{-p(S>x)D}$$

So, is E a Count or a Probability?

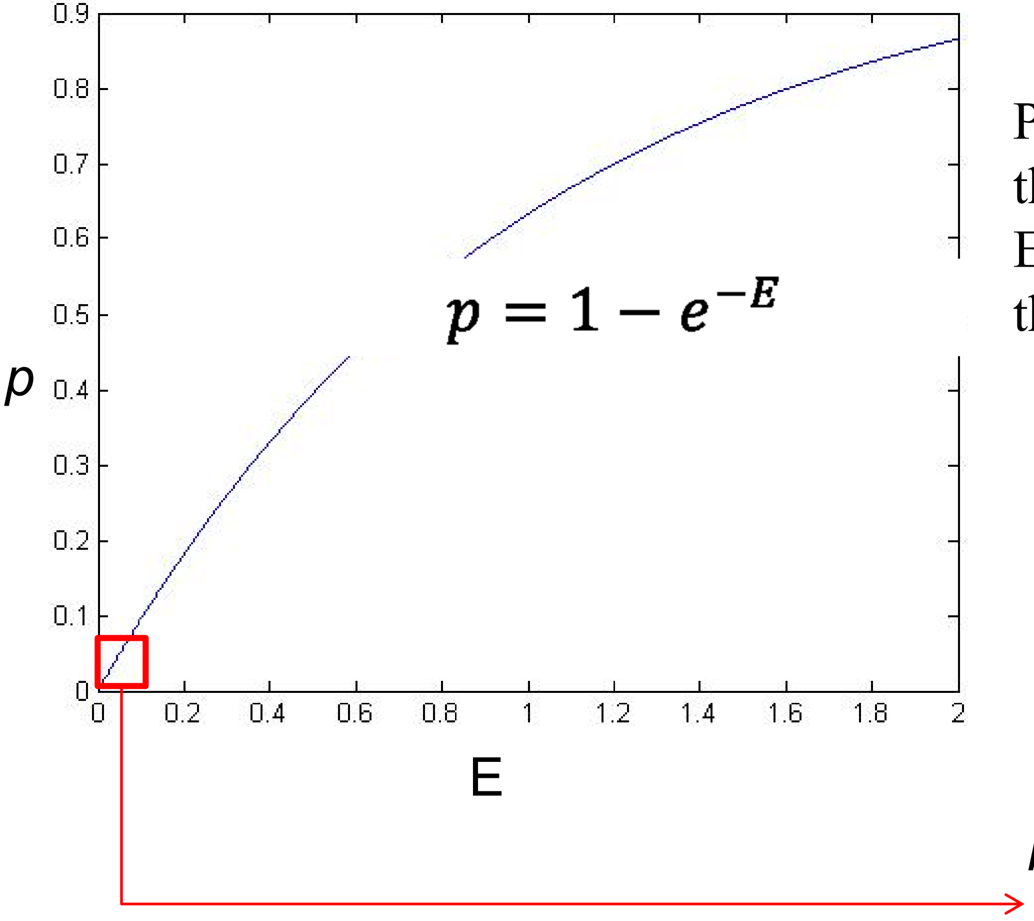
- The E-value is the expectation of a count; *i.e.,* the expected, or ‘average’, of the ***number*** of alignments that are expected to occur by chance that would exceed the expected score . It should be a very small number if the match is not random

- p is the probability that at least one alignment exceeds the expected score It is the Poisson probability c hits, with a scores $\geq$ S

Relating Probability to E-Value

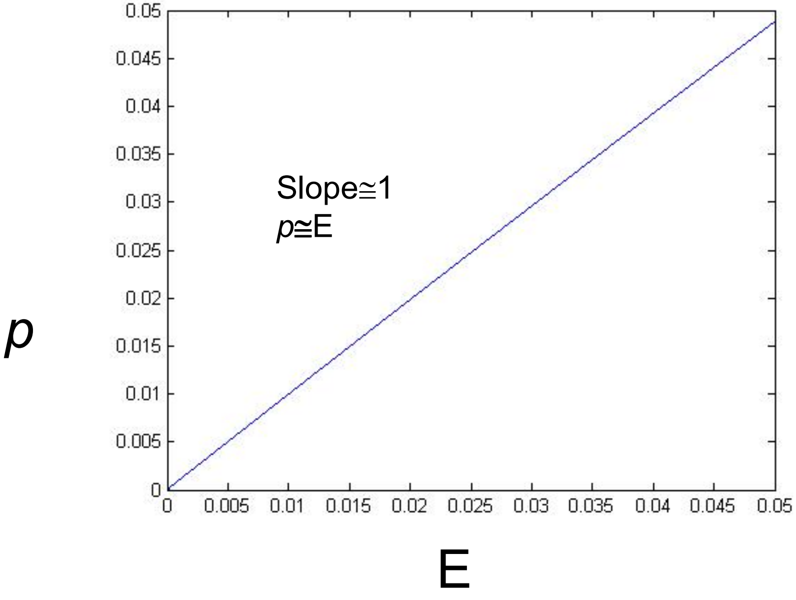
$$P = 1 - e^{-E}$$

Plotting the E-value vs p



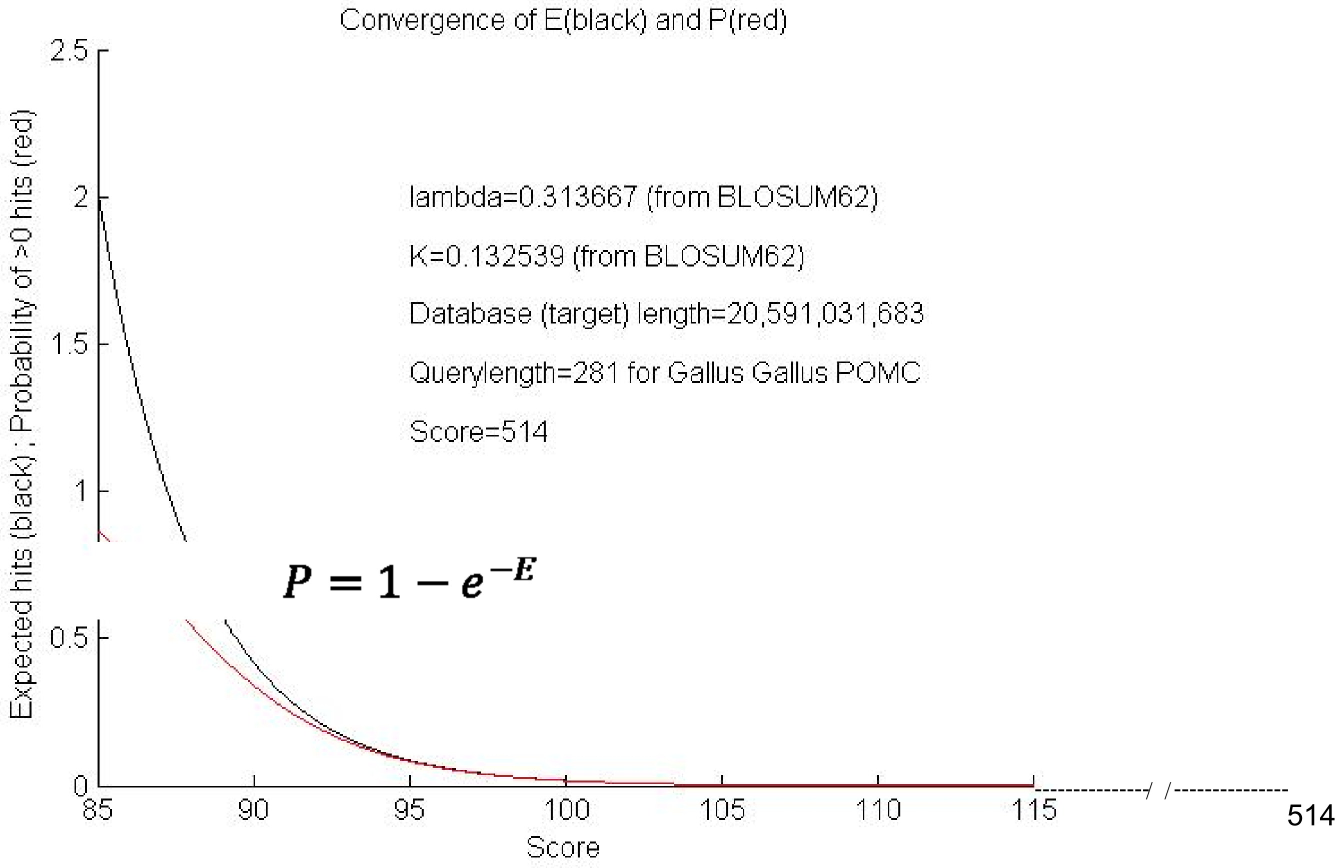
$$p = 1 - e^{-E}$$

Plugging in numbers, one can see that, for values of E ≤ ~.05, p and E are very close and below E=.01 they are essentially the **same**



Slope≅1
p≅E

E *vs* P



Convergence of E(black) and P(red)

lambda=0.313667 (from BLOSUM62)

K=0.132539 (from BLOSUM62)

Database (target) length=20,591,031,683

Querylength=281 for Gallus Gallus POMC

Score=514

$$P = 1 - e^{-E}$$

- The BLAST output lists E values. We are careless in thinking of them as probabilities; they are not! But any E value > .05 would be discarded anyway, so calling E values ‘probabilities’ is OK in that context
- Consider the POMC protein of the chicken. With a raw score of 514, in the context of the database parameters on Jan 11, 2015 listed above, The P-value and the E-value (<2e-276) are indistinguishable

The Score

- Raw score, S, comes right out of the alignment score computation, using the specific substitution matrix
- S can be normalized to S′ using database specific parameters λ and K
- It is expressed in bits; S′ is called the *bit-score*

$$S' = \frac{\lambda S - \ln K}{\ln 2}$$

The Bit Score

This normalized score can be used to compare alignments across databases of different content, substitution matrices, and database sizes

Given 2 bit scores, the respective E values could then be calculated then compared meaningfully

$$E = 2^{-S'} mn$$

The Message

The significance of a hit is based on

- The Extreme Value Distribution
- The score
  - Score depends on the length of the alignment
  - Score depends on substitution matrix
  - Score depends on how many low complexity matches there were
- The size of the query
- The size of the database
- Database specific parameters K and λ

Post Processing Details
E-score

For each hit, there is an E-value. It is the *expected number of hits* that would occur by random, given the query string, the database, and the scoring matrix. This is essentially (almost exactly, below .05) equivalent to the *probability* that the alignment achieved its score by chance. It is calculated by a very sophisticated process (Karlin, Altschul, Dembo) using the considerations just reviewed.

- Anything larger than 10^{-8} is suspicious
- The output is ordered by E-value

Considerations
About E-Values….

Remember!

You cannot compare E-values from different runs. Each run makes its own calculations based on the parameters as they exist at the time.

The nr database (for example) is updated frequently.

Turning on/off the low-complexity filter can change everything

Considerations About Bit Scores….

- Bit scores are normalized
  - The normalization takes into account the parameters of the database
  - So… you *can* compare bit scores among different runs and even among different databases (*eg* nr *vis à vis* E Coli)

Considerations GAPS

- In a large sequence, BLAST may find “islands” of high similarity. It may not be able to extend the alignment to bridge the void between them.
- As a consequence, each “island” will be considered a separate hit.
- It may be more meaningful to consider a longer alignment; this might be more consistent with the underlying phylogeny
- BLAST can allow gaps for that purpose (user parameter)

Considerations
Low Complexity Filters

- DNA, particularly Human DNA, has many regions of repeating, non coding sequence.
- Because the sequences are both numerous and irrelevant to the biological information sought, the inevitably high number of hits can skew the data interpretation.
- These regions can be excluded from the alignments by using the low-complexity filter.

Considerations
Hashing

Some very clever ideas can be implemented in hashing functions.

For example, if the word width gets larger than 3, the size of a flat hash table would grow rapidly.

All we really need to know up front is whether there is a table entry. The yes/no could be a pointer (or the absence thereof) to a linked list for that table entry.

It is possible to construct a hash table using a hashing function based on modulo ***p*** where ***p*** represents a prime number. In such a scheme the probability of never having a failure is greater than 1-1/n . Space increases only nominally with an exponential probability of having uncompromised success. A new variant of BLAST, the **B**last-**L**ike **A**lignment **T**ool (BLAT) preprocesses the database into a hash table.

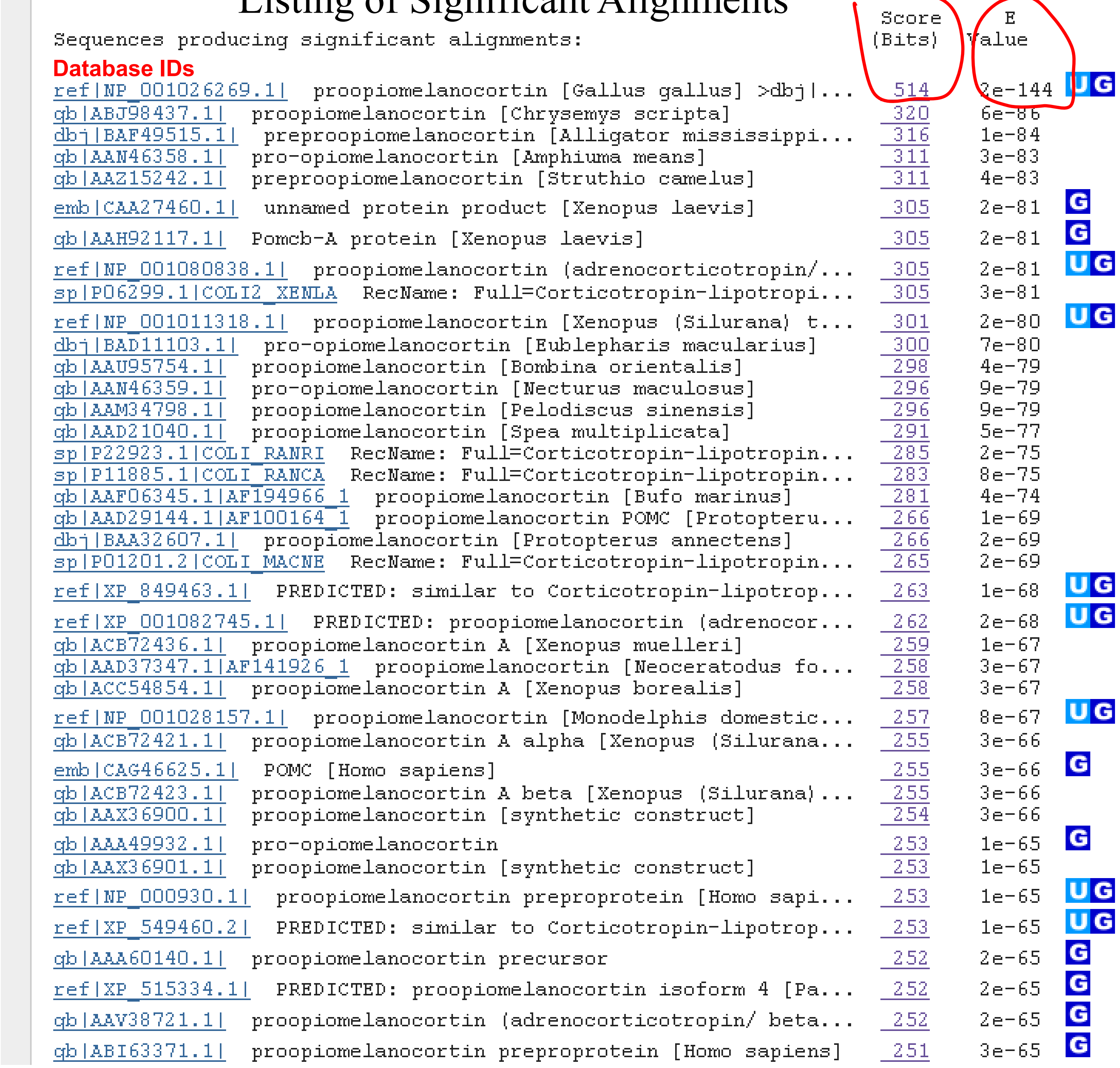
The NCBI Makes It Easy

You can follow up on any database alignment by clicking on its link. This gives details on the structure, function, classification, and references. In addition, the accession number, mother database ID, and (sometimes) the GID are provided for further analysis and classification.

Reading the BLAST Output

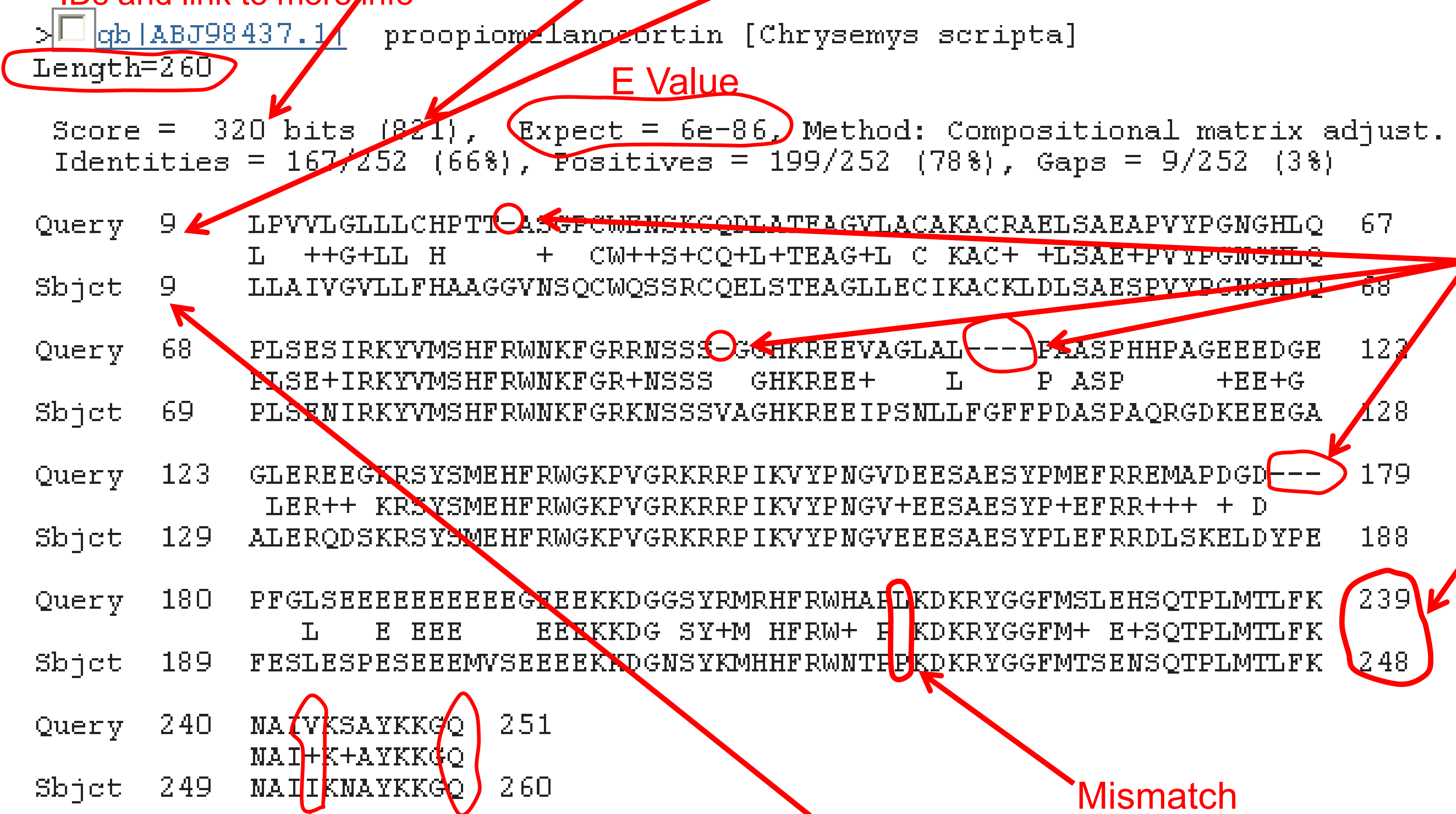
- E-value
- Coding length and raw score
- Normalized score in bits adjusted for the DB size and the substitution matrix
- The individual alignments
  - Identities
  - Positives
  - The graphical depiction
- At the very bottom of the last page…...

Listing of Significant Alignments



```
                                                                      Score      E
Sequences producing significant alignments:                           (Bits)   Value
Database IDs
ref|NP_001026269.1|  proopiomelanocortin [Gallus gallus] >dbj|...      514    2e-144  UG
gb|ABJ98437.1|  proopiomelanocortin [Chrysemys scripta]                320    6e-86
dbj|BAF49515.1|  preproopiomelanocortin [Alligator mississippi...      316    1e-84
gb|AAN46358.1|  pro-opiomelanocortin [Amphiuma means]                  311    3e-83
gb|AAZ15242.1|  preproopiomelanocortin [Struthio camelus]              311    4e-83
emb|CAA27460.1|  unnamed protein product [Xenopus laevis]              305    2e-81   G
gb|AAH92117.1|  Pomcb-A protein [Xenopus laevis]                       305    2e-81   G
ref|NP_001080838.1|  proopiomelanocortin (adrenocorticotropin/...      305    2e-81  UG
sp|P06299.1|COLI2_XENLA  RecName: Full=Corticotropin-lipotropi...      305    3e-81
ref|NP_001011318.1|  proopiomelanocortin [Xenopus (Silurana) t...      301    2e-80  UG
dbj|BAD11103.1|  pro-opiomelanocortin [Eublepharis macularius]         300    7e-80
gb|AAU95754.1|  proopiomelanocortin [Bombina orientalis]               298    4e-79
gb|AAN46359.1|  pro-opiomelanocortin [Necturus maculosus]              296    9e-79
gb|AAM34798.1|  proopiomelanocortin [Pelodiscus sinensis]              296    9e-79
gb|AAD21040.1|  proopiomelanocortin [Spea multiplicata]                291    5e-77
sp|P22923.1|COLI_RANRI  RecName: Full=Corticotropin-lipotropin...      285    2e-75
sp|P11885.1|COLI_RANCA  RecName: Full=Corticotropin-lipotropin...      283    8e-75
gb|AAF06345.1|AF194966_1  proopiomelanocortin [Bufo marinus]           281    4e-74
gb|AAD29144.1|AF100164_1  proopiomelanocortin POMC [Protopteru...      266    1e-69
dbj|BAA32607.1|  proopiomelanocortin [Protopterus annectens]           266    2e-69
sp|P01201.2|COLI_MACNE  RecName: Full=Corticotropin-lipotropin...      265    2e-69
ref|XP_849463.1|  PREDICTED: similar to Corticotropin-lipotrop...      263    1e-68  UG
ref|XP_001082745.1|  PREDICTED: proopiomelanocortin (adrenocor...      262    2e-68  UG
gb|ACB72436.1|  proopiomelanocortin A [Xenopus muelleri]               259    1e-67
gb|AAD37347.1|AF141926_1  proopiomelanocortin [Neoceratodus fo...      258    3e-67
gb|ACC54854.1|  proopiomelanocortin A [Xenopus borealis]               258    3e-67
ref|NP_001028157.1|  proopiomelanocortin [Monodelphis domestic...      257    8e-67  UG
gb|ACB72421.1|  proopiomelanocortin A alpha [Xenopus (Silurana...      255    3e-66
emb|CAG46625.1|  POMC [Homo sapiens]                                   255    3e-66   G
gb|ACB72423.1|  proopiomelanocortin A beta [Xenopus (Silurana)...      255    3e-66
gb|AAX36900.1|  proopiomelanocortin [synthetic construct]              254    3e-66
gb|AAA49932.1|  pro-opiomelanocortin                                   253    1e-65   G
gb|AAX36901.1|  proopiomelanocortin [synthetic construct]              253    1e-65
ref|NP_000930.1|  proopiomelanocortin preproprotein [Homo sapi...      253    1e-65  UG
ref|XP_549460.2|  PREDICTED: similar to Corticotropin-lipotrop...      253    1e-65  UG
gb|AAA60140.1|  proopiomelanocortin precursor                          252    2e-65   G
ref|XP_515334.1|  PREDICTED: proopiomelanocortin isoform 4 [Pa...      252    2e-65   G
gb|AAV38721.1|  proopiomelanocortin (adrenocorticotropin/ beta...      252    2e-65   G
gb|ABI63371.1|  proopiomelanocortin preproprotein [Homo sapiens]       251    3e-65   G
```

Details of One Alignment (Gapped Blast)



```
>□gb|ABJ98437.1|  proopiomelanocortin [Chrysemys scripta]
Length=260

 Score =  320 bits (821),  Expect = 6e-86, Method: Compositional matrix adjust.
 Identities = 167/252 (66%), Positives = 199/252 (78%), Gaps = 9/252 (3%)

Query  9    LPVVLGLLLCHPTT-ASGPCWENSKCQDLATEAGVLACAKACRAELSAEAPVYPGNGHLQ  67
            L   ++G+LL H     +  CW++S+CQ+L+TEAG+L C KAC+ +LSAE+PVYPGNGHLQ
Sbjct  9    LLAIVGVLLFHAAGGVNSQCWQSSRCQELSTEAGLLECIKACKLDLSAESPVYPGNGHLQ  68

Query  68   PLSESIRKYVMSHFRWNKFGRRNSSS-GGHKREEVAGLAL----PAASPHHPAGEEEDGE  122
            PLSE+IRKYVMSHFRWNKFGR+NSSS  GHKREE+     L     P ASP       +EE+G
Sbjct  69   PLSENIRKYVMSHFRWNKFGRKNSSSVAGHKREEIPSNLLFGFFPDASPAQRGDKEEEGA  128

Query  123  GLEREEGKRSYSMEHFRWGKPVGRKRRPIKVYPNGVDEESAESYPMEFRREMAPDGD---  179
             LER++ KRSYSMEHFRWGKPVGRKRRPIKVYPNGV+EESAESYP+EFRR+++ + D
Sbjct  129  ALERQDSKRSYSMEHFRWGKPVGRKRRPIKVYPNGVEEESAESYPLEFRRDLSKELDYPE  188

Query  180  PFGLSEEEEEEEEEEGEEEKKDGGSYRMRHFRWHAPLKDKRYGGFMSLEHSQTPLMTLFK  239
               L   E EEE    EEEKKDG SY+M HFRW+ P KDKRYGGFM+ E+SQTPLMTLFK
Sbjct  189  FESLESPESEEEMVSEEEEKKDGNSYKMHHFRWNTPPKDKRYGGFMTSENSQTPLMTLFK  248

Query  240  NAIVKSAYKKGQ  251
            NAI+K+AYKKGQ
Sbjct  249  NAIIKNAYKKGQ  260
```

Location of alignment start in query

Bit Score

Raw Score

IDs and link to more info

E Value

Note 9 gaps in query and offset of 9 in target position

Mismatch

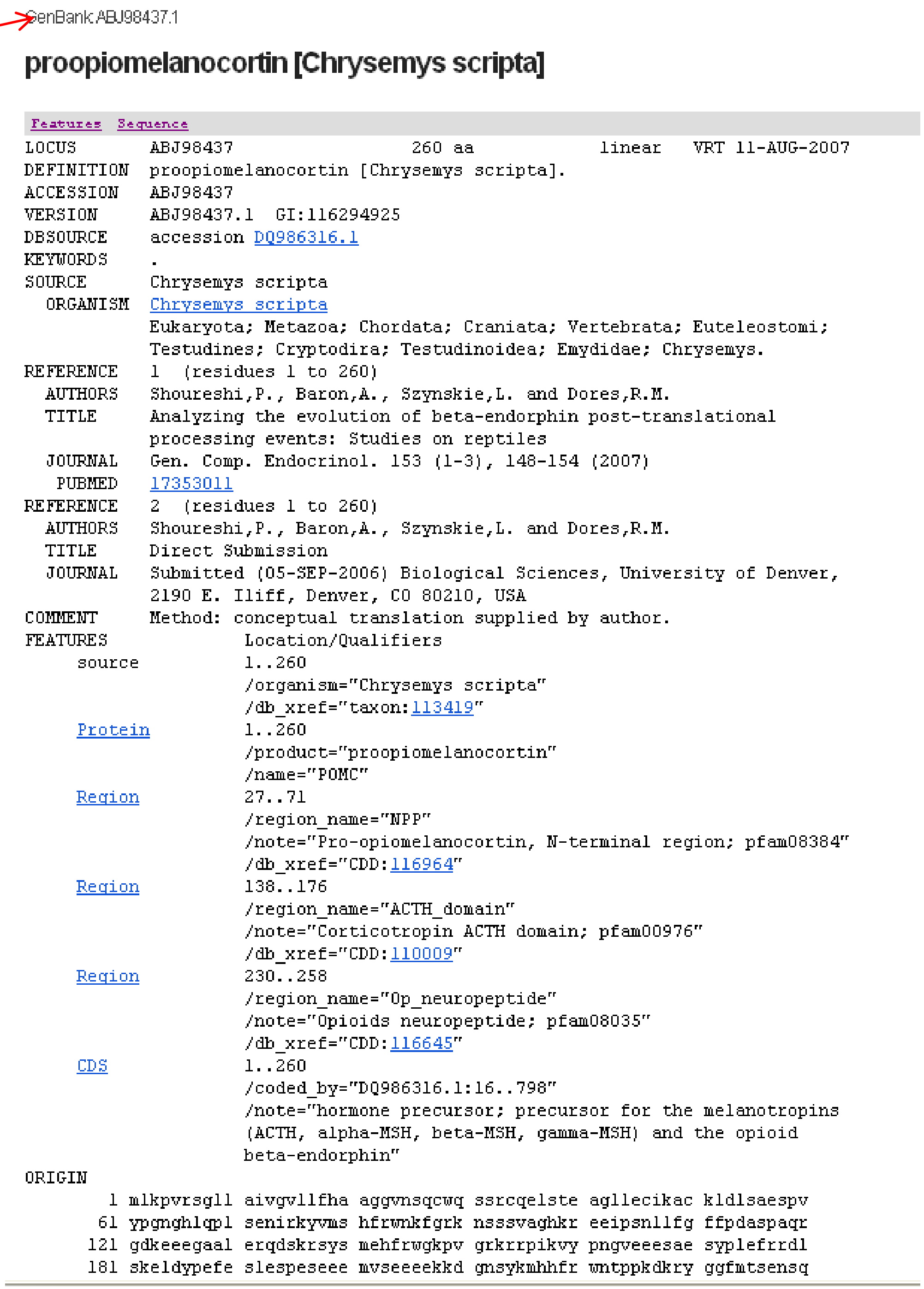
Positive

Identity

Location of alignment start in target

Following one of Many Useful Links

GenBank ID



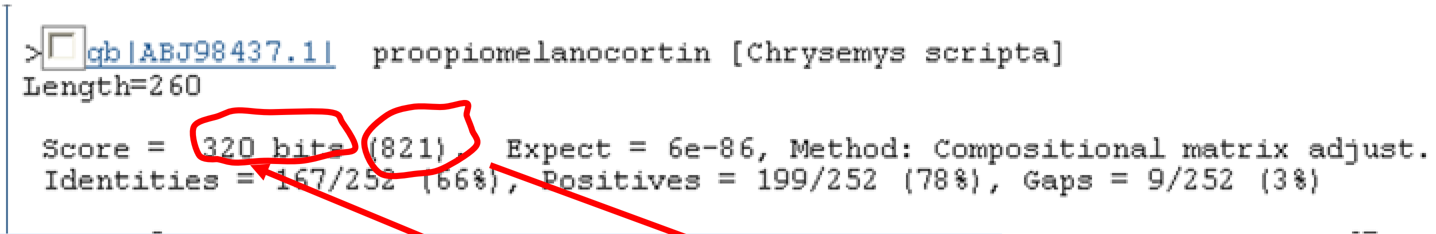
GenBank ABJ98437.1

proopiomelanocortin [Chrysemys scripta]

Features Sequence

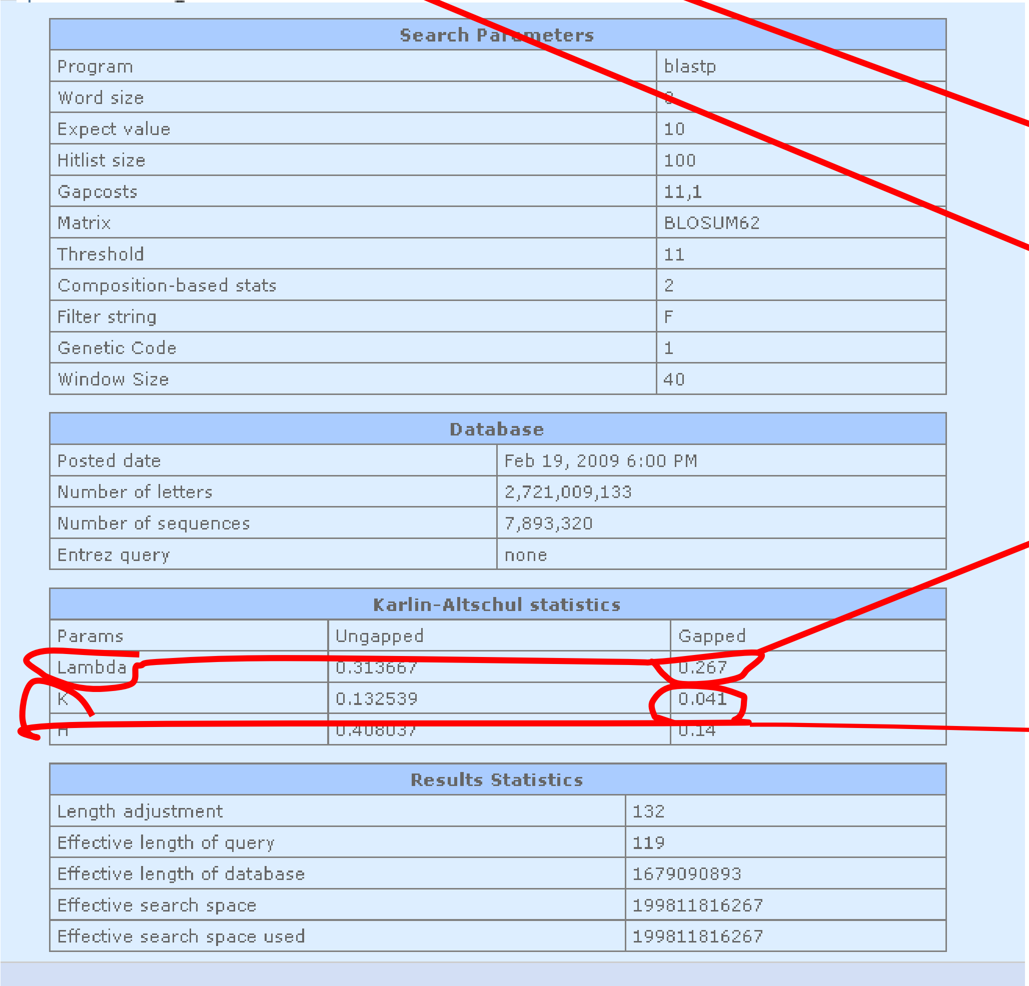
```
LOCUS       ABJ98437                 260 aa            linear   VRT 11-AUG-2007
DEFINITION  proopiomelanocortin [Chrysemys scripta].
ACCESSION   ABJ98437
VERSION     ABJ98437.1  GI:116294925
DBSOURCE    accession DQ986316.1
KEYWORDS    .
SOURCE      Chrysemys scripta
  ORGANISM  Chrysemys scripta
            Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi;
            Testudines; Cryptodira; Testudinoidea; Emydidae; Chrysemys.
REFERENCE   1  (residues 1 to 260)
  AUTHORS   Shoureshi,P., Baron,A., Szynskie,L. and Dores,R.M.
  TITLE     Analyzing the evolution of beta-endorphin post-translational
            processing events: Studies on reptiles
  JOURNAL   Gen. Comp. Endocrinol. 153 (1-3), 148-154 (2007)
   PUBMED   17353011
REFERENCE   2  (residues 1 to 260)
  AUTHORS   Shoureshi,P., Baron,A., Szynskie,L. and Dores,R.M.
  TITLE     Direct Submission
  JOURNAL   Submitted (05-SEP-2006) Biological Sciences, University of Denver,
            2190 E. Iliff, Denver, CO 80210, USA
COMMENT     Method: conceptual translation supplied by author.
FEATURES             Location/Qualifiers
     source          1..260
                     /organism="Chrysemys scripta"
                     /db_xref="taxon:113419"
     Protein         1..260
                     /product="proopiomelanocortin"
                     /name="POMC"
     Region          27..71
                     /region_name="NPP"
                     /note="Pro-opiomelanocortin, N-terminal region; pfam08384"
                     /db_xref="CDD:116964"
     Region          138..176
                     /region_name="ACTH_domain"
                     /note="Corticotropin ACTH domain; pfam00976"
                     /db_xref="CDD:110009"
     Region          230..258
                     /region_name="Op_neuropeptide"
                     /note="Opioids neuropeptide; pfam08035"
                     /db_xref="CDD:116645"
     CDS             1..260
                     /coded_by="DQ986316.1:16..798"
                     /note="hormone precursor; precursor for the melanotropins
                     (ACTH, alpha-MSH, beta-MSH, gamma-MSH) and the opioid
                     beta-endorphin"
ORIGIN
        1 mlkpvrsgll aivgvllfha aggvnsqcwq ssrcqelste agllecikac kldlsaespv
       61 ypgnghlqpl senirkyvms hfrwnkfgrk nsssvaghkr eeipsnllfg ffpdaspaqr
      121 gdkeeegaal erqdskrsys mehfrwgkpv grkrrpikvy pngveeesae syplefrrdl
      181 skeldypefe slespeseee mvseeeekkd gnsykmhhfr wntppkdkry ggfmtsensq
```

Search Summary



```
>□gb|ABJ98437.1|  proopiomelanocortin [Chrysemys scripta]
Length=260

 Score =  320 bits (821),  Expect = 6e-86, Method: Compositional matrix adjust.
 Identities = 167/252 (66%), Positives = 199/252 (78%), Gaps = 9/252 (3%)
```



Search Parameters	
Program	blastp
Word size	3
Expect value	10
Hitlist size	100
Gapcosts	11,1
Matrix	BLOSUM62
Threshold	11
Composition-based stats	2
Filter string	F
Genetic Code	1
Window Size	40

Database	
Posted date	Feb 19, 2009 6:00 PM
Number of letters	2,721,009,133
Number of sequences	7,893,320
Entrez query	none

Karlin-Altschul statistics		
Params	Ungapped	Gapped
Lambda	0.313667	0.267
K	0.132539	0.041
H	0.408037	0.14

Results Statistics	
Length adjustment	132
Effective length of query	119
Effective length of database	1679090893
Effective search space	199811816267
Effective search space used	199811816267

$$Normalized\ \ score\ base2 = \frac{\lambda S - \ln K}{\ln 2}$$

$$= \frac{.267(821) - \ln .041}{.693}$$

$$= \boxed{320.9}$$

With BLOSUM-62

Database

Database parameter name	Database parameter value
Posted date	Dec 25, 2014 12:36 PM
Number of letters	19,531,459,180
Number of sequences	54,183,042
Entrez query	none

Karlin-Altschul statistics

Params	Ungapped	Gapped
Lambda	**0.313667**	0.267
K	**0.132539**	0.041
H	0.408037	0.14
Alpha	0.7916	1.9
Alpha_v	4.96466	42.6028
Sigma		43.6362

With PAM250

Karlin-Altschul statistics

Params	Ungapped	Gapped
Lambda	**0.337579**	0.291
K	**0.230331**	0.091
H	1.09149	0.41
Alpha	0.325	0.71
Alpha_v	0.633439	6.00297
Sigma		6.71657

The **P**osition **S**pecific **S**coring **M**atrix (PSSM) – a concept

POSITION IN THE SEQUENCE

ELEMENT

	1	**2**	**3**	**4**	**…….**
A	.7	.6	.25	.3	…
C	.2	.15	.25	.2	…
G	.05	.15	.25	.2	…
T	.05	.1	.25	.3	…

Position **S**pecific **I**terated-BLAST

- Make an alignment with BLAST
- Use the highest scoring alignment as a seed
- Using all other alignments above some cutoff, complete the PSSM
- Pass the PSSM against the database as a substitution matrix
- Use the good hits to refine the PSSM
- Iterate until no new sequences are added

Psi-BLAST

- The converged PSSM discovers alignments that are further away than BLASTp