

91學年度上學期「生物資訊學」課程

Databank Search

張傳雄

國立陽明大學 遺傳學研究所

10-14-2002



Sequence Alignment vs. Similarity Searching

- Sequence alignment and similarity searching are different problems
- ☛ • Similarity searching (Blast, Fasta, Ssearch)
 - Searching for homologies to elucidate the function of an unknown protein
(The sequence itself is not informative; it must be analyzed by comparative methods against existing databases to develop hypothesis concerning relatives and function.)
 - Produces alignments, but the desired result is the score
- Sequence alignment
 - Searching for consensus sequence
 - Produces a score, but the desired result is the sequence

Why do you want to do a databank search?

- Early discovery of protein purification artifacts
- Identification of new/unknown proteins
 - to find out if a new DNA sequence already is deposited in the databanks
 - to find proteins homologous to a putative coding ORF
 - to find similar non-coding DNA stretches in the database (e.g., repeat elements, regulatory sequences)
- Look for the possible functions of an unknown protein (if you get a homologous protein)
- Collect sequences for other studies
 - phylogenetic analysis
 - primer design (e.g., locate false priming sites for a set of PCR oligonucleotides)
 - looking for new motifs

Steps in database similarity searching

- Choose a query sequence or a representation of a sequence alignment
- Search against every sequence in a sequence database to identify the most similar ones
- Show a list of the best-matched sequences
- Show an alignment of the query with the best matched sequences
- Evaluate the significance of the alignment score

Pairwise Alignment Methods

1. Dot matrix analysis
(Gibbs and McIntyre)

2. Dynamic programming algorithms Direct!
(Needleman-Wunsch, Smith-Waterman)

* Although dynamic programming is time-consuming, it finds the most rigorous solution & is most sensitive for detecting subtle similarities.

☛ 3. Heuristic Algorithms = Word or k -tuple methods
(BLAST, FASTA) Indirect!

* Computationally fast approach to find similar sequences by first finding identical stretches.

Finding efficient database searching methods

* Dynamic programming requires order N^2L computations!
(where N is size of the query sequence & L is the size of the database)

- Implement the dynamic programming algorithm in hardware to execute them faster
- Use parallel hardware to divide the problem to a number of processors and integrate the results later
- Use different algorithms to work faster than the original dynamic programming
 - too many calculations were “wasted” by comparing regions that have nothing in common

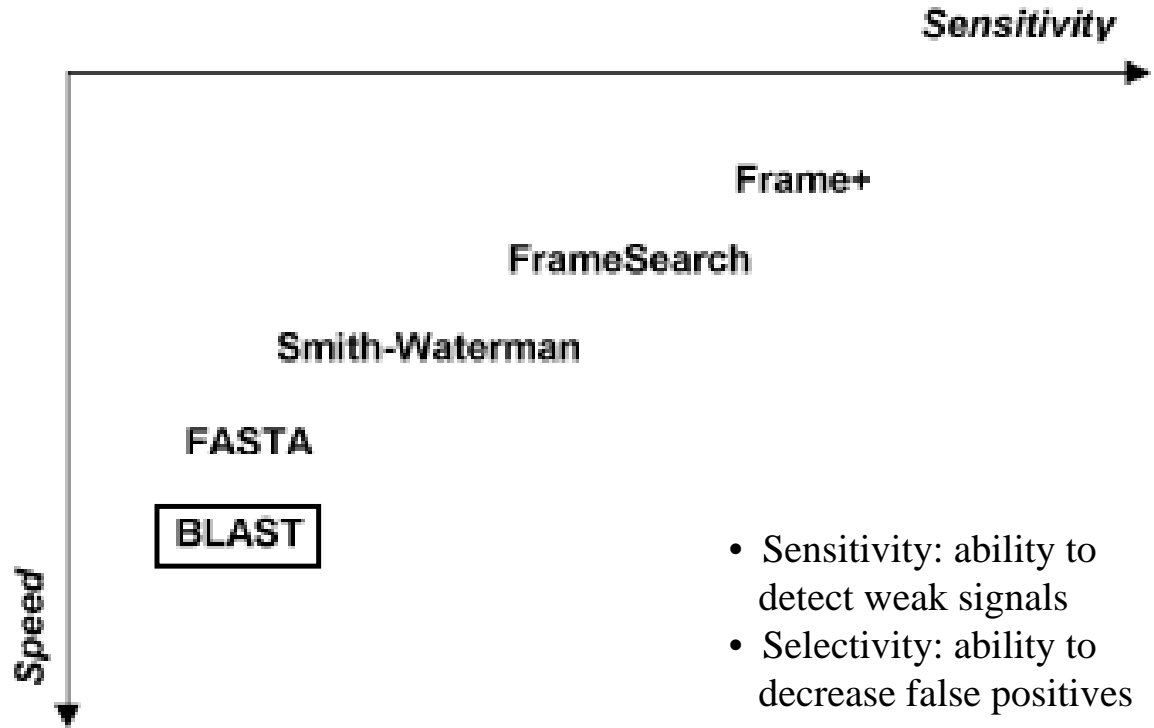
“Hit and extend” sequence searching

- Initial insight: Regions that are similar between two sequences are likely to share short stretches that are identical
- Basic method: Look for similar regions only near short stretches that match exactly
- Solutions: Use a precomputed table that lists where in the database each possible “word” occurs
 - a word size is the minimum number of exact “letter” matches that must occur before doing any further comparison or alignment
 - scan sequence a word at a time
 - generation of the table is of order L (size of database) but use of the table is of order N (size of query sequence)
 - computer science term for this approach is **hashing**

Difficulties in Similarity Searching & Solutions

- Slowness of dynamic programming alignment algorithm
 - use a word search, heuristic method
(nearly always works but is not a guaranteed algorithm)
- Low complexity regions in sequences give false high scores
 - filter out these regions in the query sequence
- How good is the match?
 - Produce an optimal local alignment
- How significant is the alignment score?
 - Provide an expect score (E value) for finding a match with an unrelated sequence in searching a sequence database of the same size

Popular Programs for database searching



- BLAST → better for proteins than for nucleotides
- FastA → better for nucleotides than for proteins
- Smith-Waterman → More sensitive than FastA or BLAST.

Specificity and Sensitivity

- **Sensitivity:** the ability to detect "true positive" matches.
The most sensitive search finds all true matches, but might have lots of false positives
- **Specificity:** the ability to reject "false positive" matches.
The most specific search will return only true matches, but might have lots of false negatives

- false positive
 - noise that has been taken as a signal
- false negative
 - signal that has been taken as a noise

Optimal Detection

- **High Specificity** - no/low false positives
high signal/background ratio:

$$\% \text{ Specificity} = \frac{\text{true negatives}}{\text{false positives} + \text{true negatives}} \times 100$$

$$\% \text{ Specificity} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}} \times 100$$

- **High Sensitivity** - no/low false negatives

$$\% \text{ Sensitivity} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}} \times 100$$

	Reality	
	TRUE	FALSE
Prediction	TRUE	TP FP
	FALSE	FN TN

FastA

Fast Alignment

Lipman DJ, Pearson WR.
Rapid and sensitive protein similarity searches.
Science 1985 Mar 22;227(4693):1435-1441.

Pearson WR, Lipman DJ.
Improved tools for biological sequence comparison.
Proc Natl Acad Sci U S A. 1988 Apr;85(8):2444-2448.

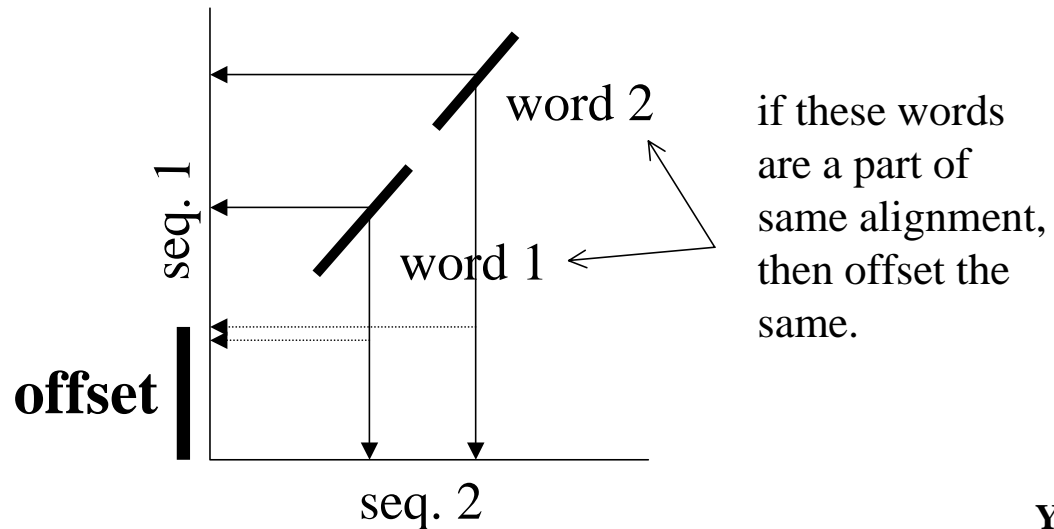
- FastA is a family of programs, which include:
 - FastA, TFastA, Ssearch, etc...

Versions of FastA

- 1) FASTA (version 3) – compares a query protein (or DNA) sequence to protein sequence library to find similar sequences
- 2) TFASTA – compares query protein sequence to a DNA sequence library after translating the DNA library in all six reading frames
- 3) FASTF/TFASTF and FASTS/TFASTS– compares a set of short peptide fragments against a protein sequence database. Fragments could come from cleavage and sequencing or protein bands resolved by electrophoresis or from mass spectrometry analysis of a protein

The FastA method

- For the query sequence and each database sequence, perform a hash of words 1-2 aa length in proteins and 4-6 bp in DNA sequences [input: k (word or ktuple) size]
- Find matching sequence words based on similar sequence offsets (position in seq. 1 - position in sequence 2)



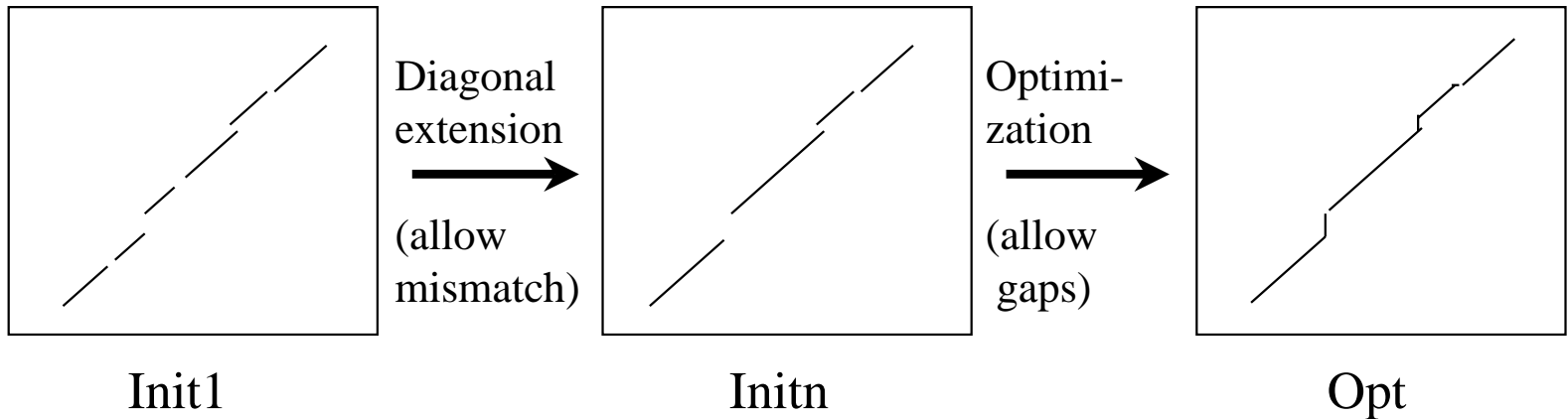
The FastA method

- Find diagonals (paired pieces from each sequence without gaps) that have the highest density of common words
- Rescore these using a scoring (similarity) matrix and trim ends that do not contribute to the highest score
 - Result: partial alignments without gaps
 - Reported as the “init1” score
- Join regions together, including penalties for gaps
 - Results: : unoptimized alignment with gaps
 - Reported as the “initn” score
- Use dynamic programming in a band 32 residues wide around the best “initn” score
 - Results: : optimized alignment with gaps
 - Reported as the “opt” score

The FastA method

- The 10 highest-scoring regions are rescored using a scoring matrix. The score of the highest scoring initial region is saved as the **init1** score.
- FastA determines if any of the initial regions from different diagonals may be joined together to form an approximate alignment with gaps. Only non-overlapping regions may be joined. The score for the joined regions is the sum of the scores of the initial regions minus a joining penalty for each gap. The score of the highest scoring region, at the end of this step, is saved as the **initn** score.
- After computing the initial scores, FastA determines the best segment of similarity between the query sequence and the search set sequence, using a variation of the Smith-Waterman algorithm. The score for this alignment is the **opt** score.

Major Steps in FastA



- Word search & diagonal extension
- Segment joining
- optimization

allow alignments to shift frames

The FastA method

- FastA uses a simple linear regression against the natural log of the search set sequence length to calculate a normalized **z-score** for the sequence pair. z-score is normalized by sequence length and is a measure (in standard deviations) of how far the score falls away from the mean.
- Using the distribution of the z-score, the program can estimate the number of sequences that would be expected to produce, purely by chance, a z-score greater than or equal to the z-score obtained in the search. This is reported as the **E() score**. E() score is an estimate of the likelihood of a similar match occurring by chance. Obviously, the lower the E() score, the more unlikely it is that the match is random. Generally, a figure of 0.01 or below is statistically very significant, and a figure of between 0.01 and 0.05 is borderline.

* the expectation value E() does not represent a measure of similarity between the two sequences!

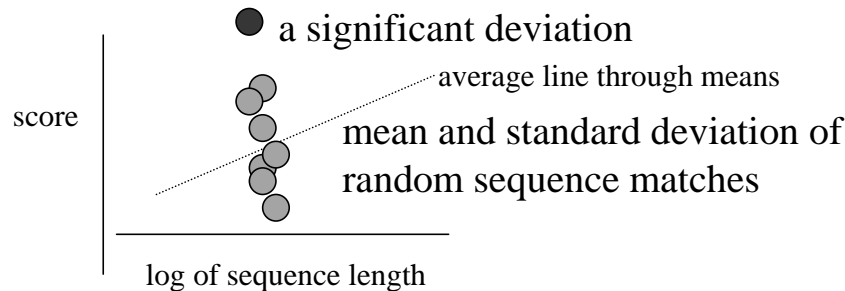
The FastA method

- The statistical significance of each score (i.e. the probability that this score could be generated by ‘hits’ between unrelated sequences, i.e. by chance is evaluated according to the extreme value distribution (EVD) in a measure called “z”
- The “z” score is the number of standard deviations away from a fitted line determined by linear regression of average scores plotted against the log of average sequence length in each length range

NOTE: high scoring presumably related sequences are removed before fitting the line.

The FastA method

- Produce local alignment of sequences with matching words and calculate score
- Use alignment scores of non-matching sequences to estimate the no. of standard deviations (s) of each score (z score) from the mean score for the unrelated sequences of the same length

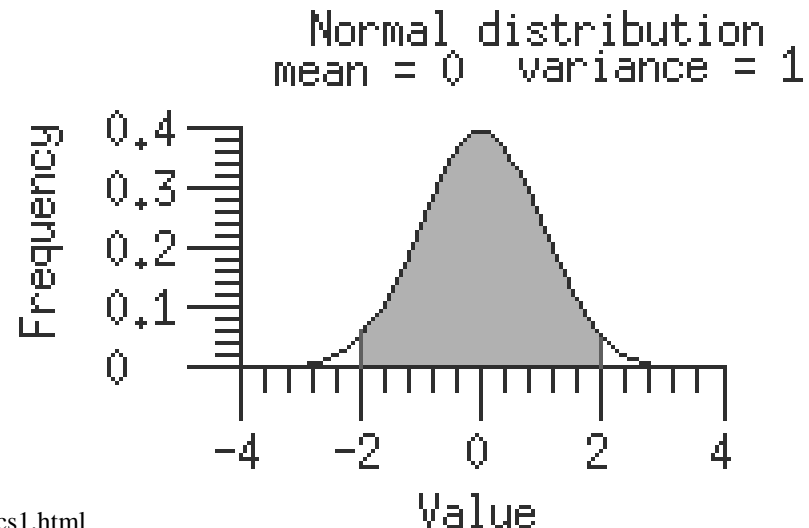


- normalize the z score to a new one of Z (or z') = $50z + 10s$
- also estimate K and l for the extreme value distribution (EVD)
- calculate $P(Z)$ for obtaining Z score and convert to an E value, usually by multiplying by the no. of sequences searched
- identify significant alignment score
($E < 0.01$ at least but the lower, the better)

The Gaussian distribution

- We are used to working with 'normal' (Gaussian) distributions
 - symmetry, a mean value and a standard deviation
 - Z-score of a value is the number of std's away from the mean
 - Z-score > 2 is considered statistically significant
($>95\%$ of surface lies between -2 and 2 standard deviations)
- When comparing all possible segments of two sequences, without gaps, the distribution of scores is normal
- The score of the MSP (maximal scoring pair) is naturally the maximum of that distribution

The distribution of random scores can be studied analytically. If we assume a normal distribution, we overestimate the significance!



The extrem value distribution (EVD)

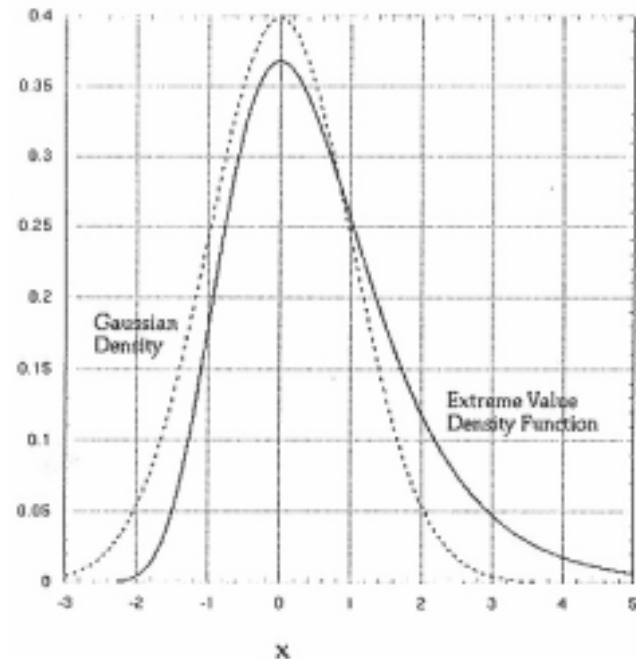
- The distribution of MSP scores of the comparisons of many random sequences is not 'normal'
it approximates an 'extreme value distribution'
- How does it compare with the normal distribution?
- In this distribution, the probability of a score being higher than x is given by:

$$P(S > x) = 1 - \lambda e^{-\lambda(x-u)}$$

- What are the parameters?
 - the characteristic value u (0 in the example)
 - the decay constant λ (1 in the example)
- The characteristic value u can be calculated as:

$$u = \frac{Kmn}{\lambda}$$

- m and n are the lengths of the sequences compared
- K and λ can be calculated from the data in the matrix used and from the relative frequencies of the amino acids (or nucleotides)



FastA Output

The output from FASTA is divided into 4 sections:

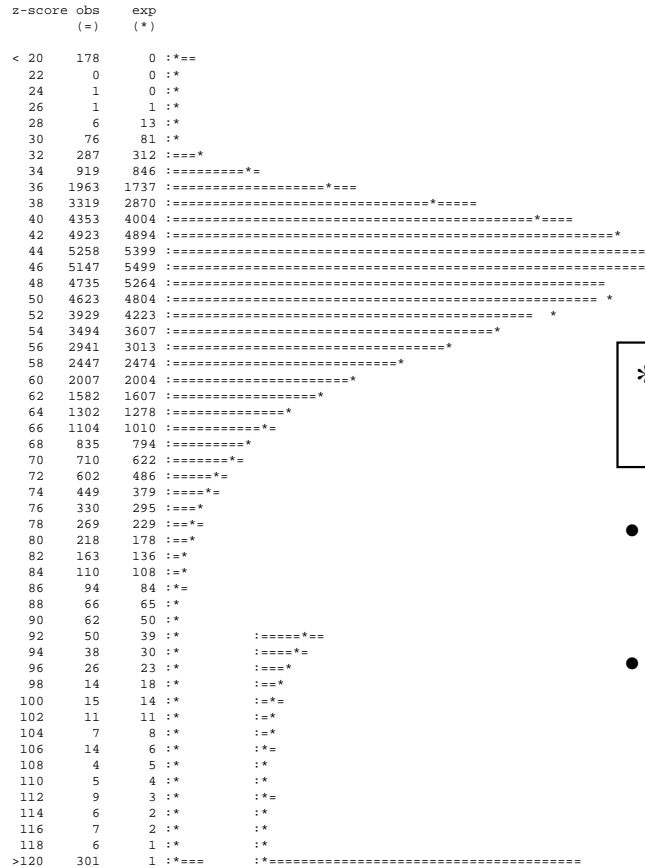
- information on the query sequence and the database searched.
- a histogram that shows graphically the observed score distribution.
- a list of the sequences matched with some statistical information about the strength of the match.
- the alignments themselves are shown.

* The histogram gives a graphical representation of the distribution of the scores. It should be expected that these scores would fall approximately into a normal distribution, and that any significant matches will fall outside the normal curve. You can see that at the bottom of the histogram there are 27 sequences that fall outside the curve (represented by the asterisks “*”).

(<http://www.bham.ac.uk/MBUG/fastaoutput.html>)

The FastA Histogram

(<http://www.bham.ac.uk/MBUG/fastaoutput.html>)



* → expected curve according to the extreme value distribution (EVD)

- the theoretic curve should be similar to the observed results
- deviations indicate that the fitting parameters are wrong
 - too weak gap penalties
 - compositional biases

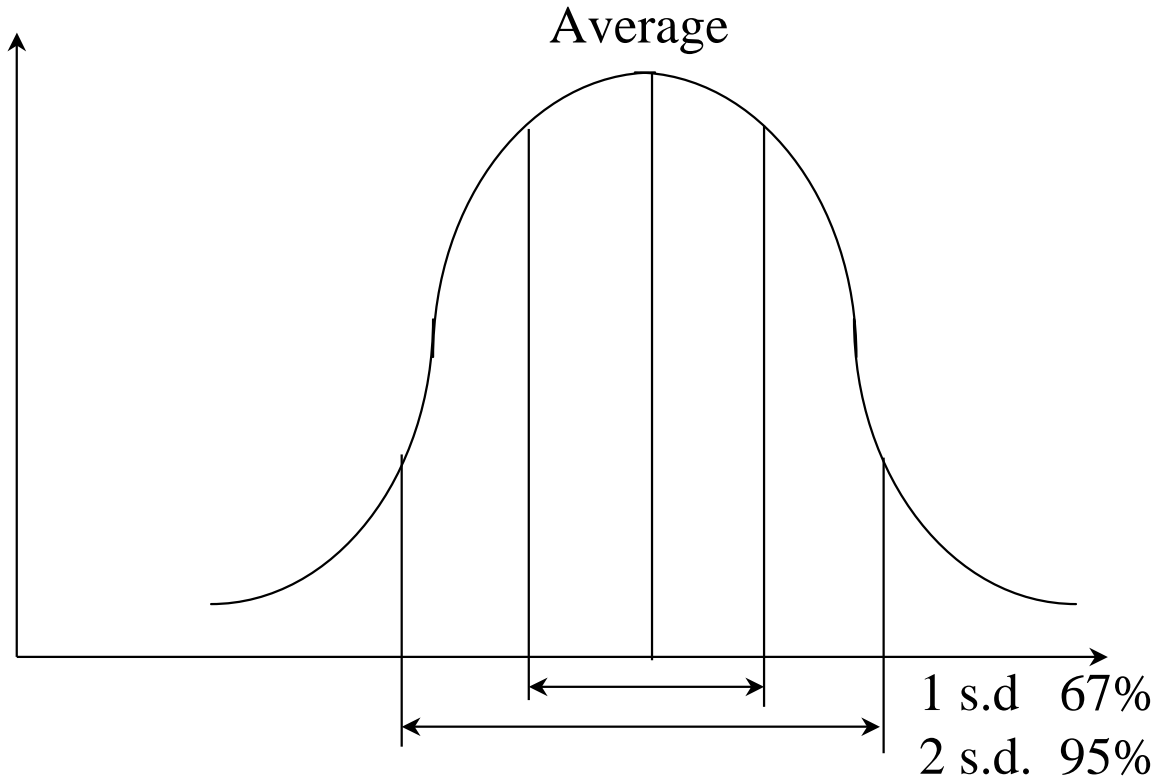
```

opt      E()
< 20  188    0:==
      22    0    0:          one = represents 109 library sequences
      24    0    0:
      26    2    1:*
      28    7    15:*
      30   28   91:*
      32  200  353:== *
      34  841  958:=====*
      36 2217 1968:=====*==
      38 3746 3253:=====*====
      40 5360 4538:=====*====
      42 6055 5547:=====*====
      44 6496 6119:=====*====
      46 5820 6232:=====*
      48 5469 5966:=====*
      50 4820 5444:=====*
      52 4202 4787:=====*
      54 3815 4089:=====*
      56 3271 3415:=====*
      58 2755 2804:=====*
      60 2268 2271:=====*
      62 1813 1821:=====*
      64 1500 1448:=====*
      66 1233 1145:=====*
      68  951  900:=====*
      70  746  706:=====*
      72  699  551:=====*

```

74	460	430:===*= 76	337	335:===* 78	287	260:==* 80	244	202:==*= 82	185	154:=* 84	115	122:=* 86	114	95:*= 88	75	73:*	inset = represents 1 library sequences	
90	70	57:*																
92	48	44:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
94	26	34:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
96	33	26:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
98	14	20:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
100	10	16:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
102	7	12:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
104	6	9:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
106	5	7:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
108	2	6:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
110	2	4:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
112	1	3:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
114	0	3:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
116	0	2:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
118	0	2:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
>120	27	1:*	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:

Review of Simple Statistics



Signal vs. Noise

z-score	obs (=)	exp (*)	
90	62	50	:*
92	50	39	:* :=====*==
94	38	30	:* :=====* =
96	26	23	:* :====*
98	14	18	:* :==*
100	15	14	:* :=* =
102	11	11	:* :=*
104	7	8	:* :=*
106	14	6	:* :* =
108	4	5	:* :*
110	5	4	:* :*
112	9	3	:* :* =
114	6	2	:* :*
116	7	2	:* :*
118	6	1	:* :*
>120	301	1	:* :*=====

Rules of Thumb (I)

- Improbable: < 3 s.d.
- Marginal: 3-5 s.d.
- Probable: 5-10 s.d.
- Certain: > 10 s.d.

When do you care about the statistical significance?

- Decide whether your protein is an unknown protein.
- Look for a distant member of a gene family.

The list of FastA hits

(<http://www.bham.ac.uk/MBUG/fastoutput.html>)

The last two numbers are a statistical measure of the significance of the match

The scores calculated at the various stages of the comparison



The best scores are:						initn	initl	opt	z-sc	E(66345)
MERR_PSEAE	mercuric	resistance	operon	regu	(144)	928	928	928	1129.8	0
MERR_SHIFL	mercuric	resistance	operon	regu	(144)	871	871	871	1061.3	0
MERR_SERMA	mercuric	resistance	operon	regu	(144)	810	810	810	988.1	0
MERR_STAAU	mercuric	resistance	operon	regu	(135)	292	172	298	373.6	3.5e-14
MERR_BACSR	(strain rc607).	mercuric	resist		(132)	241	198	289	363.0	1.4e-13
YHDM_ECOLI	hypothetical	transcriptional	re		(141)	175	175	276	347.0	1.1e-12



The database name, description, & (length) of the matched sequence

- Unlike BLAST, FASTA will only report a single match between your sequence and each database sequence, however it will allow gaps in the alignments that it generates.

Tips for FastA results

- When $\text{init1}=\text{init0}=\text{opt}$:
100 % homology over the matched stretch.
- When $\text{initn} > \text{init1}$:
more than 1 matching region in the database with poorly matching separating regions.
- When $\text{opt} > \text{initn}$:
the matching regions are greatly improved by adding gaps in one or both of the sequences.

FastA alignments

(<http://www.bham.ac.uk/MBUG/fastoutput.html>)

```
>>MERR_STAAU mercuric resistance operon regulatory protei (135 aa)
  initn: 292 initl1: 172 opt: 298 Z-score: 373.6 expect() 3.5e-14
Smith-Waterman score: 298; 36.923% identity in 130 aa overlap
```

```

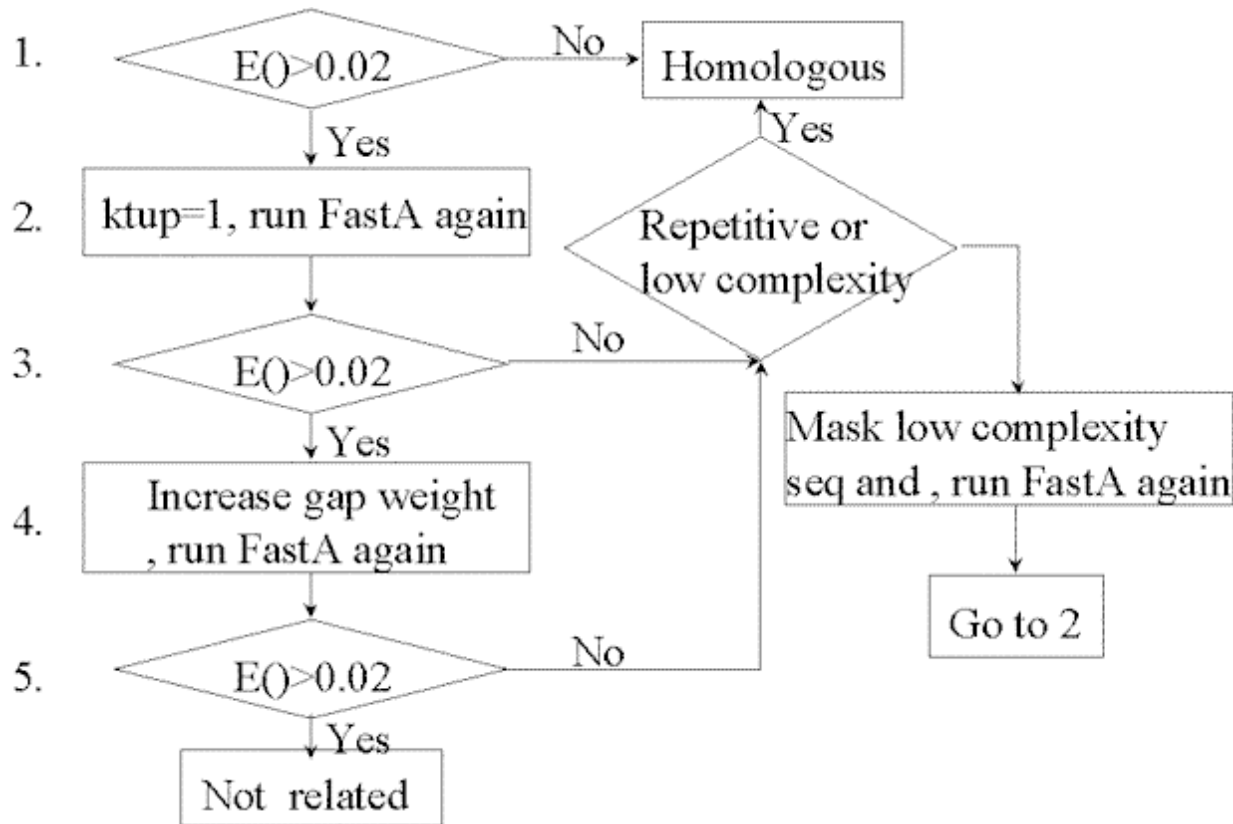
          10          20          30          40          50          60
MerR  MENNLENLTIGVFAKAAGVNVETIRFYQRKGLLLEPKPYGSIRRYGEADVTRVRFVKSA
      . . . : : : : : : : : : : : : : : : : : : : : : : : : : : : : :
MERR_S  MGMKISELAKACDVNKETVRYRERKGLIAGPPRNESEGYRIYSEETADRVRFIKRM
          10          20          30          40          50
          70          80          90          100         110
MerR  QRLGFSLDEIAELLRL--EDGTHCEEASSLAEHKLDVREKMDLARMEAVLSELVCACH
      . . : : : : : . . . : : . . . . . : : : : : : : : : : : :
MERR_S KELDFSLKEIHLLFGVVDQGERCKDMYAFTVQKTKEIERKVQGLLRIQRLLEELKEKCP
          60          70          80          90          100         110
          120         130         140
MerR  ARRGVSCPLIASLQGGASLAGSAMP
      . . . : : : : : : : :
MERR_S DEKAMYTCPIIETLMGGPDK
          120         130
```

Identities are shown with the : symbol, and similarities with the . symbol. Where FASTA has introduced gaps to optimize the alignment, these are shown with -- symbols in the sequence.

Rules of Thumb (II)

- Assumption: 100 a.a. in length
- Certain: $> 25\%$ identity
- Marginal: 15-25% identity
- Improbable: $< 15\%$ Identity

Pearson's suggestion for FastA parameter setting



What is k-tuple (ktup)?

- short for *k respective tuple*
- The number of alphabets in a word during hash coding.
- Speed and selectivity are controlled with the “ktup” (wordsize) parameter.
 - Tips for ktup:
 - For proteins, the default, ktup=2, ktup=1 is more sensitive but slower.
 - For DNA, ktup=6, the default, ktup=3 or ktup=4 more sensitivity, ktup=1 for oligonucleotides (length <20).
- Larger ktuple increases speed since fewer “hits” are found but it also decreases sensitivity for finding similar sequences since exact matches of this length are required

Effect of k-tuple (word size) on matching

- Large k-tuple will lose sensitivity
 - does not match
 - does not extend

	5	10	15	20	
	TCGGA	TTCGT	ACGGT	ACGGA	TC
	TCGAA	TTCGC	ACGGT	ACGGA	TC

If $k_{\text{tup}}=4$, you will not find TCGA and TCGC in this sequence.

Effect of k-tuple (word size) on initial scores

ktup=2

	60	70	80	90	100	110
tf3a_xenla	HLTRHSLTHTGEKNFTCDSD GCDLRF TTKANMKKHFNRFHNIKICVYVCHFENCGKAFKK					
	:: : : ::: : : : :::: : : : : :: :					
P43_XENLA	QILKHVKRHLALKKLSCPTAG GCKMTF STKKSLSRHKLYKHGEAVPLK-CFVPGCKRSFRK					

ktup=1

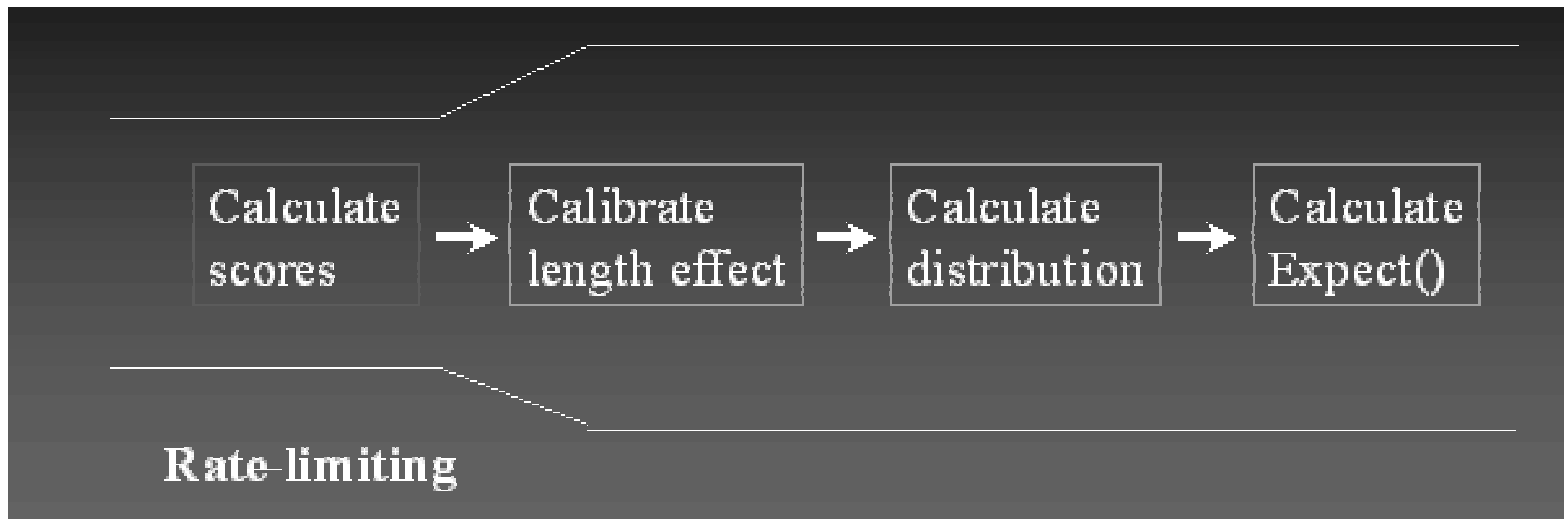
	60	70	80	90	100	110
tf3a_xenla	HLTRHSLT HT GEKNFTCDSD GCDLRF TTKANMKKHFNRFHNIKICVYVCHFENCGKAFKK					
	:: : : ::: : : : :::: : : : : :: :					
P43_XENLA	QILKHVKR HL LALKKLSCPTAG GCKMTF STKKSLSRHKLYKHGEAVPLK-CFVPGCKRSFRK					

- It is prudent to search using FastA with word size 1 if nothing interesting turns up at word size 2.

Effect of ktup on FastA output

Statistics		ktup=2	ktup=1
scores saved that exceeded 77		1555	1843
optimizations performed		49745	50877
Joining threshold		37	43
optimization threshold		25	31
opt. Width		16	32
.....			
p43	init1	1555	1843
	initn	49745	50877
	opt	37	43
	z-score	25	31
	E(58538)	16	32
	aa overlap	294	294

Summary: FastA Statistics



BLAST

Basic Local Align Search Tool

Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ.
Basic local alignment search tool.
J Mol Biol. 1990 Oct 5;215(3):403-10.

Altschul SF, Madden TL, Schaffer AA, Zhang J, Zhang Z,
Miller W, Lipman DJ.
Gapped BLAST and PSI-BLAST: a new generation of protein
database search programs.
Nucleic Acids Res. 1997 Sep 1;25(17):3389-402.

- Blast programs were designed for fast database searching, with minimal sacrifice of sensitivity to distant related sequences.

Blast method

- The motivation to the development of BLAST was the need to increase the speed of FastA by finding fewer and better hot spots (i.e., matching ktup-length substrings) during the algorithm.
- The idea was to integrate the substitution matrix in the first stage of finding the hot spots.

Blast method

- Compare query to each sequence in database
- Use heuristic to speed pairwise comparison
- Create 'sequence abstraction' by listing exact and similar words
 - on the fly for the query
 - in advance for the database
- Find similar words between query and each database sequence
- Extend such words to obtain high-scoring sequence pairs (HSPs)
- Calculate statistics analytically

HSP & MSP

HSP: High scoring segment pairs

MSP: Maximal segment pairs

- All segment pairs whose scores can not be improved by extension or trimming. These are called high-scoring segment pairs or HSPs. (local alignments with no gaps.)
- The statistical significance can be directly calculated from the score of HSP
 - the expected number of HSPs with score at least S is given by the formula:

$$E = Kmn e^{-\lambda S}$$

- Expectation (E) value is the number of different alignments with scores equivalent to or better than S that are expected to occur in a database search by chance.

The lower the E value, the more significant the score.

Blast method

- May filter input query sequence
- Also start with word search calculated based on blosum62 scoring matrix but find highest scoring words in query sequence

suppose that the query sequence is PQD SRD GYN

then evaluate each possible match with PQD, QDS etc

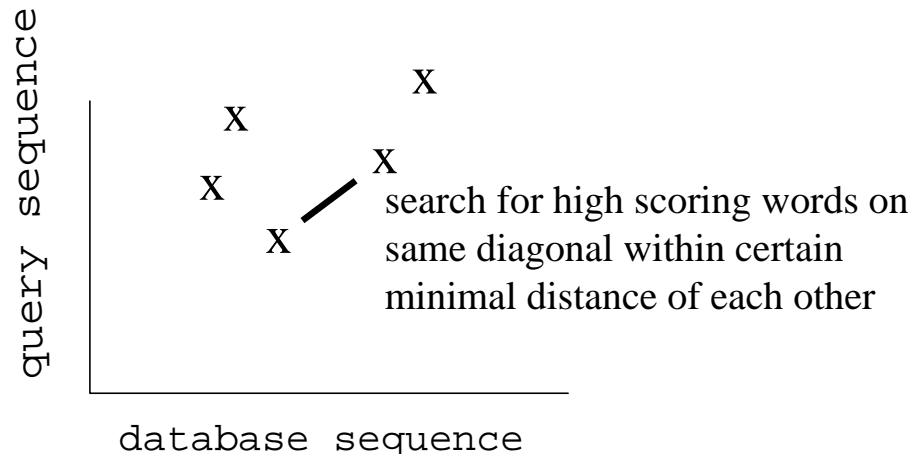
for example,	PQG	PQG	PQG	PQG
	PQG	PEG	PRG	PSG
score =	7+5+6 =18	15	14	12

choose a cutoff score so that about 50 possible matches could score

Now search the database sequence for these high scoring words.- there are about 12,500 per protein!!

Blast method

- Scan each sequence with word list for query sequence using a search tree for the word list



- Determine whether or not combined score exceeds a threshold score S
- Perform a local alignment of the qualifying database sequences and calculate E value for score

Blast method

- To calculate E values, use the K and λ calculated for random sequences of the same lengths (with length corrections) and the number of sequences in the current database
 - The parameters K and λ can be thought of simply as natural positive constant scaling factors for the search space size and the scoring system respectively
- List in order of score and significance, the matches found, then local alignments.

$$E = Kmn e^{-\lambda S}$$

The BLAST programs achieve much of their speed by avoiding the calculation of optimal alignment scores for all but a handful of unrelated sequences.

Blast method

- Define maximal segment pair (MSP) to be the highest scoring pair of identical length segments chosen from 2 sequences
(in FastA terms, highest init1 diagonal)
- Define a segment pair to be locally maximal if its score cannot be improved either by extending or by shortening both segments
- Approach: find segment pairs by first finding word pairs that score above a threshold, i.e., find word pairs of fixed length w with a score of at least T
 - Key concept: Seems similar to FASTA, but we are searching for words which score above T rather than that match exactly
 - The “ T ” parameter dictates the speed and sensitivity of the search

BLAST Statistical significance

- A key to the utility of BLAST is the ability to calculate expected probabilities of occurrence of Maximum Segment Pairs (MSPs) given w and T
- This allows BLAST to rank matching sequences in order of “significance” and to cut off listings at a user-specified probability
- The BLAST programs report E -value rather than P -value because it is easier to understand the difference between, e.g., E -value of 5 and 10 than P -value of 0.993 and 0.99995.
(However, when $E < 0.01$, P -values and E value are nearly identical.)
- E value = number of matches expected due to chance
- P value = probability of finding at least one match
with score $\geq S$

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

When does extension stop?

- When you hit the end of the sequence
- Or more likely when the “score” drops off by some number “X” from its optimal score
- The extension has no hope of achieving some minimal cut off score (~55-70, for BLOSUM 62)
- Note: in older versions of blast (prior to 2.0), there is no gapping. If there are multiple hits to a given gene that are not continuous, they are reported as “HSP”s. These HSP’s need to be manually assembled into an alignment.

The Results

- In addition for finding the “MSP” or “HSP” BLAST will identify other nearby well scoring segments for the same sequence
- Results may include a number of aligned pairs. The number of aligned pairs used to create the “SUM” statistics will be indicated in the “N” column of the result printout

The Statistics

- The score is literally the score of your alignment according to the chosen substitution matrix and gap penalty (Sum based on each pair of residues).
- Since different matrices will give different scores for the same sequence, a normalized “bit” score is provided that removes the effects of scoring matrix upon the score. The bigger the bit score, the better.
- The E value is the probability of observing the null hypothesis. The null hypothesis is that the observed database hit occurred by chance (for this given query, matrix and database [size]).

What is word size?

The number of alphabets in a word during hash coding.

String Search

Moving box

Hash coding

A lookup table (a hash) to store the position of each word

Dinucleotide Position			Dinucleotide Position		
1	GG	3,13,18	9	GA	4,19
2	TG	-	10	TA	10,15
3	AG	-	11	AA	-
4	CG	2,8,12,17	12	CA	-
5	GT	9,14	13	GA	-
6	TT	6	14	TC	1,7,21
7	AT	5,20	15	AC	11,16
8	CT	-	16	CC	-

Large word size will lose sensitivity

- Does not match
- Does not extend

Gapped Blast (v.2.0) vs. Blast 1.4

圖表 40. 以 p97 基因，用 BLAST 2.0 搜尋資料庫，和 AA285332 序列並列部分結果

```
gb|AA285332|AA285332 HTH148 HTC DL1 Homo sapiens cDNA 5'/3' similar to Mus. translation
initiation factor (Eif4g2)
Length = 650

Score = 912 bits (460), Expect = 0.0
Identities = 615/655 (93%), Positives = 615/655 (93%), Gaps = 12/655 (1%)

Query: 2884 aattagtgaagaagaagctttcttggcttggaaagaagatataacccaagagtttccgg 2942
          ||||| ||| || || ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct: 1 aattattgaanaaaaaactttcttggcttggaaagaanataataacccaanagtt-ccgg 59
```

圖表 41. 以 p97 基因，用 BLAST 1.4 搜尋資料庫，和 AA285332 序列並列部分結果

```
gb|AA285332|AA285332 HTH148 HTC DL1 Homo sapiens cDNA 5'/3' similar to
Mus. translation initiation factor (Eif4g2)
Length = 650

Plus Strand HSPs:

Score = 458 (126.6 bits), Expect = 3.6e-171, Sum P(4) = 3.6e-171
Identities = 106/127 (83%), Positives = 106/127 (83%), Strand = Plus / Plus

Query: 2884 AATTACTGAACAAGAAGCTTTCTTGGCTTGGCAARGAAGATATATACCCAGAGCTTTCCGGG 2943
          ||||| ||| || || ||||| ||||| ||||| || || | |||||
Sbjct: 1 AATTATTGAANAAAAAactttcttggcttggcaargaanatataacccaanagttccggg 60
```

Gapped BLAST (Blast2.0)

3 Changes to the Algorithm

- Threshold for neighborhood word generation was decreased.
- Criterion for extending word pairs modified, there must be two hits on the same diagonal within some distance X , (this gives an increase in speed)
- Smith-Waterman calculations are used to produce the final alignment on successful extensions (thus, it will contain gaps)

Word Extension

- In the older versions of BLAST, if a word pair with a score above T was encountered when screening the DB, it was extended.
- In the newer version, two non-overlapping words located at some distance X (the “hitdist”) from each other must hit the same sequence in the DB before an extension is performed.
- To maintain sensitivity, must lower the value of T . This yields more hits, but few are extended.

Gapped Alignment

- Original BLAST found many HSP's and used all to generate a SUM statistic
- If you gap then you only need to find only one rather than all ungapped alignments.
- T is lowered to achieve more hits on initial scan
- Only pairs of hits on the same diagonal within some distance "H" are extended
- Gapped alignments are achieved via dynamic programming to extend the pairs of aligned residues in both directions within some window of gap tolerance.

Gapped Blast (v.2.0) vs. Blast 1.4

- Detect all *word hits* (exact, or nearly identical matches) of a given length between the two sequences
 - $k=10$ for nucleotide sequences (exact word matches)
 - $k=3$ for protein sequences (nearly identical word matches)
- Extend the word hits in both directions to high-scoring *gap-free* segment pairs (HSPs)
 - retain only HSPs that score above a threshold
 - start from the center of the HSP (original BLAST, 1990), or from the center of a pair of HSPs located close to each other on the same diagonal (gapped BLAST, 1997)
- Extend the HSPs in both directions allowing for gaps
 - use dynamic programming, and stop when the alignment score falls more than a threshold X below the best score yet seen
- Report all statistically significant local alignments
 - E-value (starting with BLAST 2.0) is used to measure the statistical significance
 - *E-value* = the number of alignments with score equal to or higher than s one would expect to find by chance when searching the database

Gapped Blast (v.2.0) vs. Blast 1.4

- P value is the probability of an alignment occurring with the score in question or better. The p value is calculated by relating the observed alignment score, S , to the expected distribution of HSP scores from comparisons of random sequences of the same length and composition as the query to the database. The most highly significant P values will be those close to 0. P values and E values are different ways of representing the significance of the alignment.
- In Blast 2.0, an E value is used instead of a P value to report the significance of each hit.
- Unlike in FastA, where only the diagonal allowed to have local shifts, restricted to a band, in Gapped BLAST, local alignments from different diagonals are allowed to merge as long as the resulting alignment has a score above some threshold.

What is the Expect (E) value?

http://www.ncbi.nlm.nih.gov/BLAST/blast_FAQs.html#Expect

- The Expect value (E) is a parameter that describes the number of hits one can "expect" to see just by chance when searching a database of a particular size. It decreases exponentially with the Score (S) that is assigned to a match between two sequences. Essentially, the E value describes the random background noise that exists for matches between sequences. For example, an E value of 1 assigned to a hit can be interpreted as meaning that in a database of the current size one might expect to see 1 match with a similar score simply by chance. This means that the lower the E-value, or the closer it is to "0" the more "significant" the match is. However, keep in mind that searches with short sequences, can be virtually indistinguishable and have relatively high E-value. This is because the calculation of the E-value also takes into account the length of the Query sequence. This is because shorter sequences have a high probability of occurring in the database purely by chance.
- The Expect value can also be used as a convenient way to create a significance threshold for reporting results. You can change the Expect value threshold on most main BLAST search pages. When the Expect value is increased from the default value of 10, a larger list with more low-scoring hits can be reported.

Blast output

- The list of hits
- Database accession codes, name, description, general information about the hit
- Score in bits, the alignment score expressed in units of information. (Usually 30 bits are required for significance.)
 - bit scores have been normalized with respect to the scoring system so if you change scoring systems, you can still compare search results. They can be used to compare alignment scores from different searches.
- Expectation value $E()$, how many hits we expect to find by chance with this score, when comparing this query to the database.
(the $E()$ value does not represent a measure of similarity between the two sequences.)

Blast output

- The information for each hit
- A header including hit name, description, & length
- The same for all additional entries removed due to redundancy
- Composite expectation value
- Each hit may contain several HSPs
- score and expectation value
 - how many identical residues
 - how many residues contributing positively to the score
- The local alignment itself

List of Related Sequences of A Typical Blast Output

WARNING: -hspmax 100 was exceeded with 23 of the database sequences, with as many as 230 HSPs being found at one time.

	High	Smallest	
	Score	Sum	
Sequences producing High-scoring Segment Pairs:		Probability	N
		P(N)	
..			
SW:TF3A_XENLA ! P03001 xenopus laevis (african clawed fro...	1930	1.2e-268	1
SW:TF3A_XENBO ! P17842 xenopus borealis (kenyan clawed fr...	1564	2.4e-231	2
SW:TF3A_RANPI ! P34695 rana pipiens (northern leopard fro...	1173	2.7e-172	2
SW:TF3A_BUFAM ! P34694 bufo americanus (american toad). t...	741	8.2e-161	3
SP_HUM:Q13097 ! Q13097 homo sapiens (human). dna/rna-bind...	544	4.8e-149	3
SP_HUM:Q92664 ! Q92664 homo sapiens (human). xenopus tran...	537	2.2e-141	3
SP_HUM:Q12963 ! Q12963 homo sapiens (human). transcriptio...	454	5.5e-134	3
SP_OV:P79797 ! P79797 ictalurus punctatus (channel catfis...	510	5.4e-86	2
SW:P43_XENLA ! P25456 xenopus laevis (african clawed frog...	231	2.8e-55	4
SW:P43_XENBO ! P25066 xenopus borealis (kenyan clawed fro...	224	2.5e-52	4
SP_HUM:Q14590 ! Q14590 homo sapiens (human). zinc finger ...	103	1.2e-41	6
...			

Blast Alignments

>SW:P43_XENLA P25456 xenopus laevis (african clawed frog). p43 5s rna binding protein (42s p43) (thesaurin b). 2/94
Length = 365

Score = 231 (108.4 bits), Expect = 2.8e-55, Sum P(4) = 2.8e-55
Identities = 45/115 (39%), Positives = 64/115 (55%)

Query: 194 CDVCNRKFRHKDYLRDHQKTHEKERTVYLCPRDGCDSYTTAFNLRSHIQSFHEEQRPFV 253
C C + F+ LR H+ TH K+ CPR CD++++ FNL H++ H +
Sbjct: 193 CAACKKPFKKASALRRHKATHAKKPLQLPCPRQCDKTFSSVFNLTHHVRKLLHLCLQTHR 252

Query: 254 CEHAGCGKCFAMKKSLEHRSVVDPEKRLKEKCPRPKRSLASRLTGYIPPKSKE 308
C H+GC + FAM++SL RH VVDPE++KLG K R R T P +E
Sbjct: 253 CPHSGCTRSFAMRESLLRHLVVDPERKLLKLFVIRGPSKFLGRGTRCRTPVVEE 307

Score = 156 (73.2 bits), Expect = 2.8e-55, Sum P(4) = 2.8e-55
Identities = 27/79 (34%), Positives = 42/79 (53%)

Query: 15 CSFADCGAAYNKNWKLQAHLCCKHTGEKPFPCKEEGCEKGFSTLHHLTRHSLTHTGEKNFT 74
C A C A Y K KLQ H+ H+ +KP+ C + C+K F + +H H K +
Sbjct: 17 CPAAGCKAFYRKEGKLDHMHAGHSEQKPWKCGIKDCDKVFAKQRQILKHVVKRHLALKKLS 76

Query: 75 CDSGCDLRF'TTKANMKKH 93
C + GC + F+TK ++ +H
Sbjct: 77 CPTAGCKMTFSTKKSLSRH 95

Bit score, with a given E, is significant if it is greater than N/E , N the size of the search space.

Types of Blast programs

The image shows a screenshot of the NCBI BLAST website. At the top, the NCBI logo is on the left, and the word "BLAST" is prominently displayed in the center. Below the logo, there are navigation tabs for PubMed, Entrez, BLAST, OMIM, Taxonomy, and Structure. The main content area is titled "What's NEW in BLAST®" and features a news item dated March 5th 2002 about new database linkouts. Below this, there are several sections with headings and lists of links:

- Nucleotide BLAST**
 - Standard nucleotide-nucleotide BLAST [blastn]
 - MEGABLAST
 - Search for short nearly exact matches
- Protein BLAST**
 - Standard protein-protein BLAST [blastp]
 - PSI- and PHI-BLAST
 - Search for short nearly exact matches
- Translated BLAST Searches**
 - Nucleotide query - Protein db [blastx]
 - Protein query - Translated db [tblastx]
 - Nucleotide query - Translated db [tblastn]
- Search for conserved domains**
 - Search the Conserved Domain Databases using RPS-BLAST
 - Search by domain architecture [CDART]
- Pairwise BLAST**
 - BLAST 2 Sequences
- Genomic BLAST pages**
 - Human Genome
 - Mouse Genome
 - Rat Genome
 - Fugu rubripes
 - Zebrafish Genome
 - Anopheles gambiae
 - Arabidopsis thaliana
 - Oryza sativa
 - Other eukaryotes
 - Microbial Genomes
- Specialized BLAST pages**
 - VecScreen - BLAST-based detection of vector contamination
 - IgBLAST - Analysis of immunoglobulin sequences in GenBank
 - Trace BLAST - A page optimized for cross-species comparisons
- Retrieve results for an existing Request ID**
 - Retrieve results with a Request ID

On the left side of the page, there is a vertical navigation menu with links for NCBI, SITE MAP, BLAST info, BLAST overview, Frequently Asked Questions, BLAST Program Selection Guide, Description of BLAST Services, Subscribe to BLAST-Announce, New/Noteworthy, BLAST course, BLAST tutorial, BLAST references, URL API documentation, HTML format, and PDF format. On the right side, there are additional links for PostScript format, FTP, BLAST FTP site, Credits, BLAST Credits, Mail, BLAST Help Desk, and NCBI Info Service.

<http://www.ncbi.nlm.nih.gov/BLAST/>

Parameter comparison for nucleotide blast

Blastn

Choose filter Low complexity Human repeats Mask for lookup table only Mask lower case

Expect

Word Size

Short nearly exact matches

Choose filter Low complexity Human repeats Mask for lookup table only Mask lower case

Expect

Word Size

Protein blast

- BLASTP
- PSI-BLAST (position specific iterated)
& PHI-BLAST (pattern hit initiated)
for searching short nearly exact matches
- RPS (reverse position specific)-BLAST
for conserved domain search

Parameter comparison for protein blast

Composition-based statistics

Blastp

Choose filter Low complexity Mask for lookup table only Mask lower case

Expect

Word Size

Matrix Gap Costs

Composition-based statistics

Short nearly exact matches

Choose filter Low complexity Mask for lookup table only Mask lower case

Expect

Word Size

Matrix Gap Costs

Assumption of Program Blast

The probability of finding an amino acid at a given position is proportional to the composition of that amino acid in a protein.

Two Ways to Eliminate the Violations of Assumption

x-filter: mask short repeats

s-filter: mask low abundance seq.

Effect of Filters

1 MAAKIFCLIMXXXXXXXXXXXXXIFPQCSQAPIASLLPPYLSPAMSSVCENPILLPYRIQQ 60
1 MAAKIFCLIM**LLGLSASAATAS**IFPQCSQAPIASLLPPYLSPAMSSVCENPILLPYRIQQ 60

61 AIAAGIXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXNIRXXXXXXXXXXXXXXXXXXYSQQQQFLPFN 120
61 AIAAGI**LPLSPLFLQQSSALLQQPLVHLLAQ**NIR**AQQQLQQVLANLAA**YSQQQQFLPFN 120

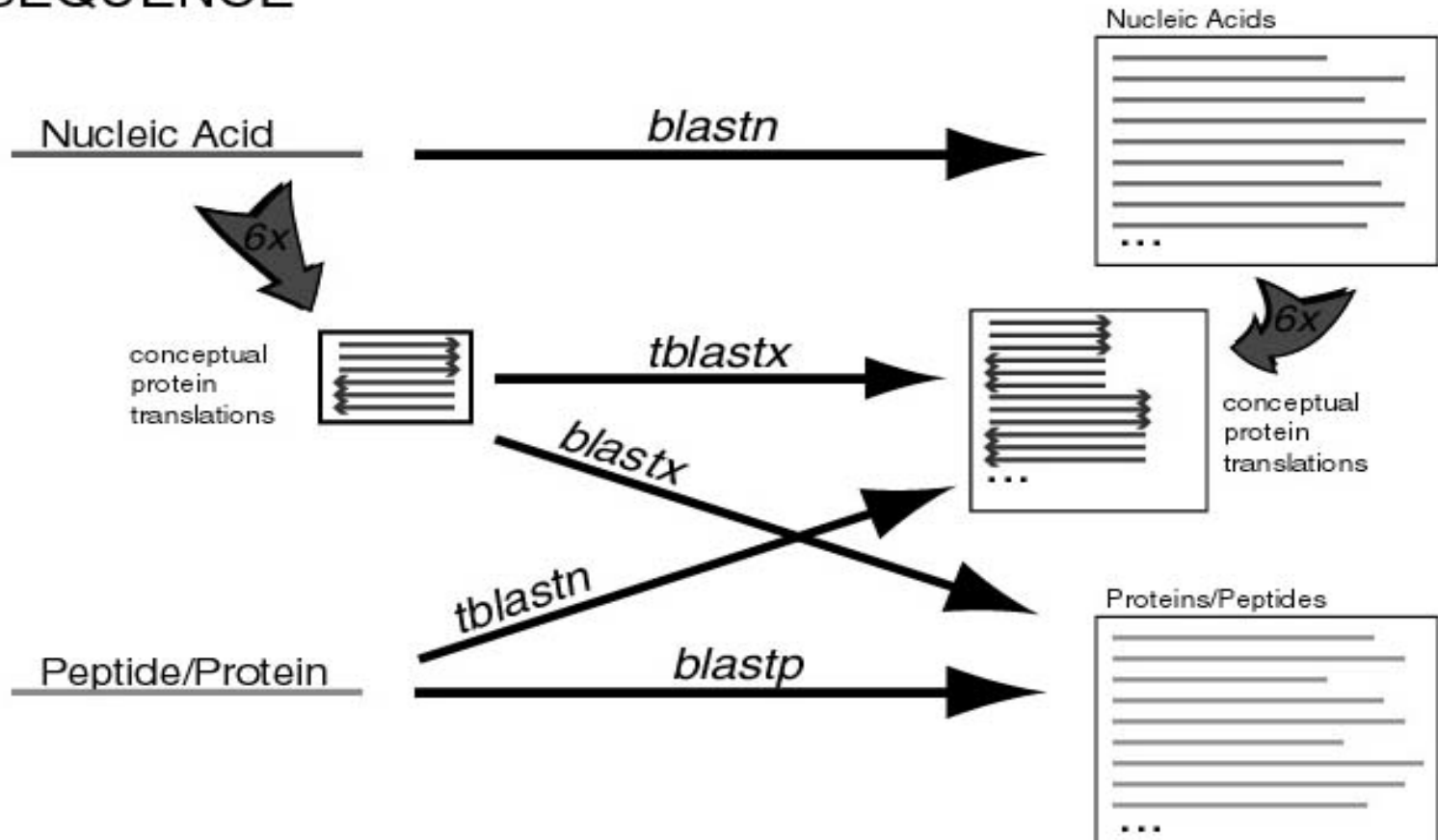
121 QXXXXXXXXXXXXXXXXXXPFSQLAAAYPRQFLPFNQLAALNSHAYVXXXXXXXXPFSQLAAVS 180
121 Q**LAALNSAAYLQQQQLL**PFSQLAAAYPRQFLPFNQLAALNSHAYV**QQQQLL**PFSQLAAVS 180

181 PAAFLTQQQLLPFYLHTAPNVGTXXXXXXXXXXXXXXXXXTNPAAFYQQPIIGGALF 235
181 PAAFLTQQQLLPFYLHTAPNVGT**LLQLQQLLPFDQLAL**TNPAAFYQQPIIGGALF 235

The different versions of BLAST

QUERY
SEQUENCE

DATABASE



Translated nucleic acid db has more information than protein db

FEATURES
source

Location/Qualifiers
1..1518
/organism="Xenopus laevis"
/db_xref="taxon:8355"

Genbank



GenPept

mRNA

<1..>1518
/product="TFIIIA mRNA"

EMBL



TrEMBL

CDS

42..1076
/note="5S RNA gene transcription factor (putative); putative"
/codon_start=1
/protein_id="AAA49967.1"
/db_xref="GI:214819"
/translation="MGEKALPVVYKRYICSFADCGAAYNKNWKLQAHLCCKHTGEKPFPCKEEGCEKGF'TSLHHLTRHSLTHTGEKNFTCDSDGCDLRF'TTKANMKKHFNRFHNIKICVYVCHFENC GKAFKKHNQLKVHQFSHTQQLPYECPHEGCDKRFSLPSRLKRHEKVHAGYPCKKDDSCSFVGKTWTLYLKHVAECHQDLAVCDVCNRKFRHKDYLRDHQKTHEKERTVYLCPRDGCDRSYTTAFNLRSHIQSFHEEQRPVCEHAGCGKCFAMKKSLEKHSVVHDPEKRKLKEKCPRPKRSLASRLTGYIPPKSKEKNASVSGTEKTDLSLVKNKPSGTETNGSLVLDKLTIQ"

DNA vs. Protein searches

- DNA is composed of 4 characters: A,G,C,T It is anticipated that on the average, at least 25% of the residues of any 2 unrelated aligned sequences, would be identical.
- Protein sequence is composed of 20 characters (aa). The sensitivity of the comparison is improved. It is accepted that convergence of proteins is rare, meaning that high similarity between 2 proteins always means homology.

Nucleic acid comparison has more noise!

Protein comparison has a higher signal strength

Same amino acid
seq. with
different nucleic
acid seq.

		2nd			
		T	C	A	G
1st					
T	TTT 0.43	TCT 0.18	TAT 0.42	TGT 0.42	
	TTC 0.57 ^{Phe}	TCC 0.23 ^{Ser}	TAC 0.58 ^{Tyr}	TGC 0.58 ^{Cys}	
	TTA 0.06 ^{Leu}	TCA 0.15	TAA 0.22 ^{TERM}	TGA 0.61 ^{TERM}	
	TTG 0.12	TCG 0.06	TAG 0.17	TGG 1.00 ^{Trp}	
C	CTT 0.12	CCT 0.29	CAT 0.41	CGT 0.09	
	CTC 0.20 ^{Leu}	CCC 0.33 ^{Pro}	CAC 0.59 ^{His}	CGC 0.19 ^{Arg}	
	CTA 0.07	CCA 0.27	CAA 0.27 ^{Gln}	CGA 0.10	
	CTG 0.43	CCG 0.11	CAG 0.73	CGG 0.19	
A	ATT 0.35	ACT 0.23	AAT 0.44	AGT 0.14	
	ATC 0.52 ^{Ile}	ACC 0.38 ^{Thr}	AAC 0.56 ^{Asn}	AGC 0.25 ^{Ser}	
	ATA 0.14	ACA 0.27	AAA 0.40 ^{Lys}	AGA 0.21 ^{Arg}	
	ATG 1.00 ^{Met}	ACG 0.12	AAG 0.60	AGG 0.22	
G	GTT 0.17	GCT 0.28	GAT 0.44	GGT 0.18	
	GTC 0.25 ^{Val}	GCC 0.40 ^{Ala}	GAC 0.56 ^{Asp}	GGC 0.33 ^{Gly}	
	GTA 0.10	GCA 0.22	GAA 0.41 ^{Glu}	GGA 0.26	
	GTG 0.48	GCG 0.10	GAG 0.59	GGG 0.23	

* Picture made from http://www.kazusa.or.jp/java/codon_table_java/

Protein comparison has a higher signal to noise ratio

- * What about very different DNA sequences that code for similar protein sequences?
We certainly do not want to miss those.

Protein comparison will be more sensitive
and have less false positives

Conclusion:

We should use proteins for database similarity searches when possible.

DNA vs. Protein searches

- The reasons for this conclusion are:
 - When comparing DNA sequences, we get significantly more random matches than we get with proteins.
 - The DNA databases are much larger, and grow faster than Protein databases. Bigger database means more random hits!
 - For DNA we usually use identity matrices, for protein more sensitive matrices like PAM and BLOSUM, which allow for better search results.
 - The conservation in evolution, protein are rarely mutated.

Specialized blast pages

- Human Genome
- Finished and Unfinished Microbial Genomes
- *P. falciparum*
- VecScreen - BLAST-based detection of vector contamination
- IgBLAST - Analysis of immunoglobulin sequences in GenBank

If you want more
specialized blast pages

Establish your own blast server
using information at
<ftp://ncbi.nlm.nih.gov/blast/>

Tips for database searches

- Use latest database version
- Run Blast first, then depending on your results run a finer tool (FastA, Ssearch, SW, blocks, etc..)
- Where possible use translated sequence.
- $E() < 0.05$ is statistically significant, usually biologically interesting. Check also $0.05 < E() < 10$ because you might find interesting hits.
- Pay attention to abnormal composition of the query sequence, it usually causes biased scoring.
- Split large query sequence (if >1000 for DNA, >200 for protein).