BIOS 477/877
Bioinformatics and Molecular Evolution

Instructor: Etsuko Moriyama
(School of Biological Sciences)

Spring 2026 **Lecture 10**

1

Today's topics

- Amino Acid Substitution Matrix
  - Information Theory
- Statistical Significance of Alignment Scores
- Assignment 5

2

Log odds matrix

- PAM matrix

  $S(i,j) = 10\log_{10}\{M(i,j)/f(i)\}$

  $M(i,j)$: Mutation probability from AA_j to AA_i
  $f(i)$: Frequency of AA_i (number of AA_i / total number of residues)
  → Probability to find AA_i by chance

- BLOSUM matrix

  $S(i,j) = 2\log_2(q_{ij}/e_{ij})$

  q_{ij}: Observed frequency of AA_i, AA_j pairs
  e_{ij}: Expected frequencies of AA_i, AA_j pairs [pairing by chance]

- General form

  $S(i,j) = 1/\lambda \log_2(q_{ij}/p_ip_j)$ [in bit unit]
  $S(i,j) = 1/\lambda \log_e(q_{ij}/p_ip_j)$ [in nat unit]

3

Relative entropy (H)

	AA_1	AA_2
AA_1	S_{11}	S_{21}
AA_2	S_{12}	S_{22}

- **Expected score (E)**

  $E = \Sigma_i\Sigma_j\, p_ip_jS(i,j)$ [p_i, p_j: expected freq. of AA_i, AA_j]

- **Relative entropy (H)**

  $H = \Sigma_i\Sigma_j\, q_{ij}\, \lambda S(i,j)$

  [q_{ij}: observed freq. of AA_i, AA_j pair]

  Entropy in Information Theory: completely unrelated to "entropy" in thermodynamics!!

  - The average information per residue pair
  - Summarizes the behavior of the scoring matrix
  - Measures the ability of the matrix to discriminate related from unrelated (nonrandom from random matching)
    - → $H = 0$ when target distribution equals to background distribution
      [If $q_{ij}=p_ip_j \rightarrow S(i,j) = 1/\lambda\log_e(q_{ij}/p_ip_j) = 1/\lambda\log_e(1) = 0$]
    - → H increases when the two distributions become more distinguishable
  - Can be used to compare scoring matrices

4

Introduction to information theory

- **Information: a decrease in uncertainty**
  (unpredictability, a degree of surprise)
- If you are asking questions to somebody...
  - if you can guess every answer correctly
    → there is no surprise
    → you cannot gain any new information
  - but if you have no idea what answer you get
    → every answer is a surprise
    → you gain a lot of information

See "Information Theory Primer" by Tom Schneider (on CANVAS)

5

How to quantify information

- **Information: a decrease in uncertainty**
  (unpredictability, a degree of surprise)

Device ⟹ A, A, A, A, A, A, A, A, ...

Prob(A) = 1
Only one possible symbol

No surprise
No information

6

How to quantify information

- **Information: a decrease in uncertainty**
  (unpredictability, a degree of surprise)

Device ⟹ A, B!, A, B!, A, B, A, B, ...

Prob(A) = 0.5
Prob(B) = 0.5

Two possible symbols
A little surprise
A small amount of information

7

How to quantify information

- **Information: a decrease in uncertainty**
  (unpredictability, a degree of surprise)

Device ⟹ A, B!, C!!, A, B!, C!!, A, B, ...

Prob(A) = 0.33
Prob(B) = 0.33
Prob(C) = 0.33

Three possible symbols
More surprise, More information

8

How to quantify information

- **Information: a decrease in uncertainty**
  (unpredictability, a degree of surprise)

Device ⟹ A, A, A, A, A, B!!!, A, A, ...

Prob(A) = 7/8
Prob(B) = 1/8

Two possible symbols
Big surprise! A lot more information...?

But not much surprise in getting the symbol A's

9

How to quantify information

- **Information: a decrease in uncertainty**
  - Surprising answers convey more information!
  - → If each symbol is equally likely, the amount of information increases with the number of different symbols.
  - → The amount of information, or surprise of an answer, is inversely proportional to its probability.

$I(p) = \log_2(1/p)$ or $I(p) = -\log_2 p$

I: information, p: probability

10

Unit of information: bit

- **Bits**: the unit for values converted to base 2 logarithms (**nats**: the unit if base e is used)

$I(p) = -\log_2 p$

- If an answer is highly unexpected (*e.g.*, $p = 0.1$),
  → $I(0.1) = -\log_2 0.1 = 3.3$ bits (more information)
- For a very much expected answer (*e.g.*, $p = 0.9$),
  → $I(0.9) = -\log_2 0.9 = 0.15$ bits (less information)
- If there is only one possible answer (symbol):
  → $p = 1$, $I(p) =$?

11

Shannon entropy

- Information can be represented by a series of symbols each with a certain probability
  - **Shannon Entropy**: the average information per symbol

$H = -\sum p_i(\log_2 p_i)$

  - If all n symbols are equally possible ($p_i = p = 1/n$),
    → $H = -\sum p(\log_2 p) = -(np \times \log_2 p)$
    $= -\log_2 p$, since $np = 1$
    $= -\log_2(1/n)$, since $p = 1/n$
    $= \log_2(n)$

e.g., $H(1)=\log_2(1)=0$ bit, $H(2)=\log_2(2)=1$, $H(4)=\log_2(2^2)=2$
$H(n)$: n is the number of equally possible letters

13

Shannon entropy

- Information can be represented by a series of symbols each with a certain probability:
  - Shannon Entropy: the average information per symbol

$$H = -\sum p_i(\log_2 p_i)$$

[Examples]

- For a random DNA sequence: ATGC (p = 0.25 for all)
  H = -4 x 0.25 x $\log_2(0.25)$ or $\log_2(4)$ = 2 bits
- For a AT-rich DNA sequence: $p_A = p_T$ = 0.45 and $p_G = p_C$ = 0.05
  H = {-2 x 0.45 x ($\log_2$0.45)} + {-2 x 0.05 x ($\log_2$0.05)}
  = {-2 x 0.45 x (-1.15)} + {-2 x 0.05 x (-4.32)} = 1.47 bits

14

Relative entropy (H)

- Expected score (E)
  $E = \Sigma_i\Sigma_j\, p_i p_j S(i,j)$ [p_i, p_j: expected freq. of AA_i, AA_j]
- Relative entropy (H)
  $H = \Sigma_i\Sigma_j\, q_{ij}\, \lambda S(i,j)$ [q_{ij}: observed freq. of AA_i, AA_j pair]

Since $S(i,j) = 1/\lambda \log_2(q_{ij}/p_i p_j)$ or $1/\lambda \log_e(q_{ij}/p_i p_j)$,

$H = \Sigma_i\Sigma_j\, q_{ij} \log_2(q_{ij}/p_i p_j)$ or $\Sigma_i\Sigma_j\, q_{ij} \log_e(q_{ij}/p_i p_j)$

$= \Sigma_i\Sigma_j\{q_{ij}\log_2(q_{ij}) - q_{ij}\log_2\{p_i p_j)\}$

Connection to Shannon entropy: $H = -\Sigma p_i(\log_2 p_i)$

Note: Both expected score and relative entropy have their units in bit (or nat).

15

Comparing scoring matrices

- Relative entropy (H) of a scoring matrix
  $H = \Sigma_i\Sigma_j\, q_{ij}\, \lambda S(i,j)$ [q_{ij}: observed freq. of AA_i, AA_j pair]
  - The average information per residue pair for a scoring matrix
  - Summarizes the behavior of the scoring matrix
  - Measures the ability of the matrix to discriminate related from unrelated (nonrandom from random matching)
    - → H = 0 when target distribution equals to background distribution
      [If $q_{ij}=p_i p_j \rightarrow S(i,j) = 1/\lambda\log_e(q_{ij}/p_i p_j) = 1/\lambda\log_e(1) = 0$]
    - → H increases when the two distributions become more distinguishable
  - Can be used to compare scoring matrices

16

Comparing scoring matrices

- Relative entropy (H)

H(PAM1) = 4.17 bits
H(PAM50) = 2.00
H(PAM120) = 0.98
H(PAM160) = 0.70
H(PAM250) = 0.36

from Altschul (1991)

- H decreases with increasing PAM#
- H increases with increasing BLOSUM#

- Higher BLOSUM is generated with sequences that are more similar to one another → More amino acid pairs are used

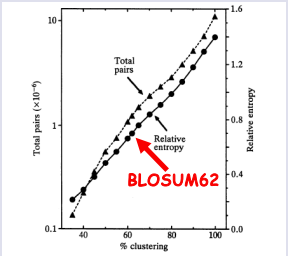


FIG. 1. Relationship between percentage clustering and total amino acid pair counts plotted on a logarithmic scale and relative entropy.

from Henikoff and Henikoff (1992)

17

Comparing scoring matrices

- Relative entropy (H) of a scoring matrix
  $H = \Sigma_i\Sigma_j\, q_{ij}\, \lambda S(i,j)$
  - The average information per residue pair for a scoring matrix
  - Decreases with increasing PAMn: H(PAM1) > H(PAM120)
    - All PAMn is extrapolated from PAM1
    - Higher PAMn is less specific, contains less information, → has a lower H
  - Increases with increasing BLOSUMn:
    H(BLOSUM45) < H(BLOSUM80)
    - Higher BLOSUMn is generated using more data (fewer information is eliminated) → has a higher H
      [*e.g.*, BLOSUM100 is generated using the threshold 100%; only identical sequences are down-weighted for calculation]

18

BLOSUM and PAM matrices

(default in BLAST)

BLOSUM80	BLOSUM62	BLOSUM45
Less divergent	⟷	More divergent
PAM120	PAM160	PAM250
H = 0.98 bits	0.7 bits	0.36 bits

19

BLOSUM62

```
#  Matrix made by matblas from blosum62.iij
#  * column uses minimum score
#  BLOSUM Clustered Scoring Matrix in 1/2 Bit Units
#  Blocks Database = /data/blocks_5.0/blocks.dat
#  Cluster Percentage: >= 62
#  Entropy =   0.6979, Expected =  -0.5209
   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  . . .
```

$S(i,j) = 2\log_2(q_{ij}/e_{ij})$

Can be downloaded from NCBI ftp site

20

How to select a right scoring matrix

Selecting the Right Similarity-Scoring Matrix
Pearson (2013)

Relative entropy (H)

Table 3.5.1 Scoring Matrix Target Identity, Information Content, and Alignment Length[a]

Matrix	Gap penalty[b]	% Identity	Bits/position	Random alignment length	50-bit length
SSEARCH version 36.3.6					
BLOSUM50[c]	10/2	25.3	0.21	160	238
BLOSUM62	11/1	28.9	0.40	86	125
VTML 160[c,d]	12/2	23.9	0.25	139	200
VTML 140	10/1	28.4	0.44	82	114
VTML 120	11/1	32.1	0.54	62	93
VTML 80	10/1	40.5	0.74	47	68
VTML 40	13/1	64.7	1.92	18	26
VTML 20	15/2	86.1	3.30	11	15
VTML 10	16/2	90.9	3.87	9	13
BLAST version 2.2.27+					
BLOSUM50[c]	13/2	29.4	0.39	85	128
BLOSUM62	11/1	29.6	0.41	82	122
BLOSUM80	10/1	32.0	0.48	69	104
PAM70	10/1	33.9	0.58	56	86
PAM30	9/1	45.9	0.90	34	56

(VTML PAM type)

- Default matrices (*e.g.*, BLOSUM62) are good for identifying ≳25% identity.
- Deep scoring matrices (*e.g.*, BLOSUM62, PAM250) require long sequence alignment to achieve significant scores (*e.g.*, >50 bits).
- They are more likely to extend alignments outside of homologous region.

21

How to select a right scoring matrix

Selecting the Right Similarity-Scoring Matrix
Pearson (2013)

VTML 20

	A	R	N	D	C	Q	E
A	7						
R	-7	8					
N	-6	-5	8				
D	-6	-12	-1	8			
C	-3	-7	-8	-14	12		
Q	-5	-2	-4	-4	-13	9	
E	-5	-10	-5	-1	-14	-1	7

BLOSUM62

	A	R	N	D	C	Q	E
A	4						
R	-1	5					
N	-2	0	6				
D	-2	-2	1	6			
C	0	-3	-3	-3	9		
Q	-1	1	0	0	-3	5	
E	-1	0	0	2	-4	2	5

Figure 3.5.2 Comparison of a "shallow" (VTML 20) and "deep" (BLOSUM62) scoring matrix. Both matrices are scaled in 1/2-bits. For the small part of the matrices shown here, the VTML20 matrix produces an average 2.80 half-bit identity score, and an average −0.59 nonidentical score (weighted by amino-acid abundance). In contrast, BLOSUM62 produces 1.86 for identities but only −0.06 for nonidentities. Thus, VTML20 targets shorter, higher-identity alignments, because it penalizes nonidentities much more strongly.

- Short alignments require shallow scoring matrices to be more significant.
- Shallower scoring matrices (*e.g.*, PAM20) are more effective when searching over shorter evolutionary distances.

22

BLOSUM62, RBLOSUM, and CorBLOSUM

CORRESPONDENCE VOLUME 26 NUMBER 3 MARCH 2008 NATURE BIOTECHNOLOGY

BLOSUM62 miscalculations improve search performance

RBLOSUM

Styczynski *et al.*

Hess *et al. BMC Bioinformatics* (2016) 17:189
DOI 10.1186/s12859-016-1060-3

BMC Bioinformatics

RESEARCH ARTICLE

CorBLOSUM

Addressing inaccuracies in BLOSUM computation improves homology search performance

Martin Hess[1,2†], Frank Keul[2*†], Michael Goesele[1] and Kay Hamacher[2]

Govindarajan *et al. BMC Res Notes* (2018) 11:328
https://doi.org/10.1186/s13104-018-3415-5

BMC Research Notes

RESEARCH NOTE Open Access

RBLOSUM performs better than CorBLOSUM with lesser error per query

Renganayaki Govindarajan*, Biji Christopher Leela and Achuthsankar S. Nair

Still the original BLOSUM62 is used...

23

More specific substitution matrices

Keul *et al. BMC Bioinformatics* (2017) 18:293
DOI 10.1186/s12859-017-1703-2

BMC Bioinformatics

RESEARCH ARTICLE Open Access

PFASUM: a substitution matrix from Pfam structural alignments

Frank Keul[1†], Martin Hess[2*†], Michael Goesele[2] and Kay Hamacher[1]

BIOINFORMATICS

Vol. 27 ISMB 2011, pages i15–i23
doi:10.1093/bioinformatics/btr230

Environment specific substitution tables improve membrane protein alignment

Jamie R. Hill[1], Sebastian Kelm[1], Jiye Shi[2,3] and Charlotte M. Deane[1,*]

Published online 16 September 2021

NAR Genomics and Bioinformatics, 2021, Vol. 3, No. 3 1
https://doi.org/10.1093/nargab/lqab079

Evolutionary and functional lessons from human-specific amino acid substitution matrices

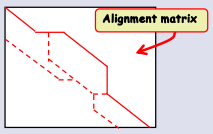
Tair Shaul[1], Nadav Brandes[1] and Michal Linial[2,*]

- See more on CANVAS (References)
- More substitution matrices reviewed in Trivedi & Nagarajaram (2020)
- EModelDB: a database of substitution matrix

24

Pairwise alignment summary

- Alignment score depends on:
  - Scoring matrix (match, mismatch, Ts/Tv, BLOSUM, PAM, *etc.*)
  - Gap penalty
  - Alignment method (*e.g.*, global or local)
- Alignment scores cannot be compared directly
  - if the scoring systems used are different
  - if sequences compared are different (*e.g.*, longer alignments tend to have higher scores)
- Alignment scores are used:
  - for searching optimal alignments
  - from the alignment matrix
  - for a given pair of sequences
  - based on a given scoring system

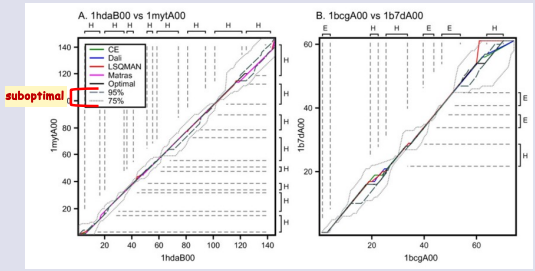


25

Optimal vs. suboptimal alignment

Improving pairwise sequence alignment accuracy using near-optimal protein sequence alignments

Michael L Sierk[1], Michael E Smoot[2], Ellen J Bass[3], William R Pearson[4*]



Suboptimal alignments substantially overlap with structure-based alignments

26

Pairwise alignment summary (continued)

- Optimal alignments and biologically meaningful alignments may not be the same
- Depending on the scoring system, unreasonable alignments can become optimal
  - → We need to choose a better (biologically reasonable) scoring system considering:
    - level of divergence (scoring matrices)
    - gap penalty (affine, etc.)
    - algorithm (local, global, or semi-global)
  - → Manual adjustment may be necessary
  - → Need to test statistical significance of the alignment (is the alignment possible just by chance?)

27

Hypothesis testing

- For general tests
  - Two hypotheses
    - Null-hypothesis
      H_0: The previous (original) belief is true
    - Alternative hypothesis
      H_1: The previous (original) belief is false; The new theory is true
  - S: Test statistic
  - Significance level is chosen *a priori* (*e.g.*, 0.05)
  - P-value: $P(S|H_0 \text{ is true})$
    → Probability of getting S if H_0 is true
  - If P < Significance level, reject H_0

28

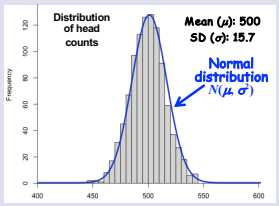
How to calculate P-values

- P-value: $P(S|H_0 \text{ is true})$
  - Calculated from the test statistic S
    → Need to know the probability distribution of the test statistic S under the null hypothesis, H_0

Central Limit Theorem:
If the sample size is large enough, the sampling distribution of the mean of any independent, random variables will be normal or nearly normal.

[Example experiment]
- Toss a coin 1000 times
- Count the number of heads
- Repeat 1000 times

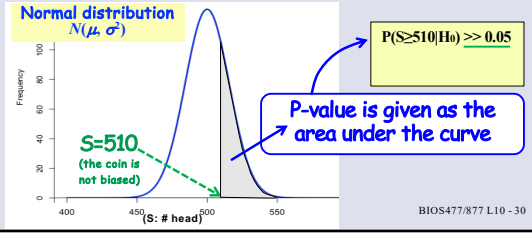
(Expect to see 500 heads/experiment)



29

How to calculate P-values

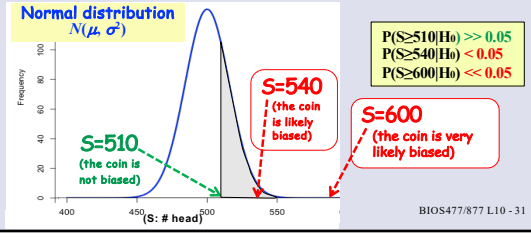
- P-value: $P(S|H_0 \text{ is true})$
  - Calculated from the test statistic S
    → Need to know the probability distribution of the test statistic S under the null hypothesis, H_0



30

How to calculate P-values

- P-value: $P(S|H_0 \text{ is true})$
  - Calculated from the test statistic S
    → Need to know the probability distribution of the test statistic S under the null hypothesis, H_0



31

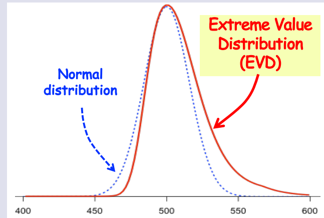
How to test alignment scores

- Hypothesis testing for sequence alignment
  - Two hypotheses
    - Null-hypothesis
      H_0: Two sequences are not related (random)
    - Alternative hypothesis
      H_1: Two sequences are related
  - Test statistic: alignment score (S)
  - Significance level is chosen a priori (*e.g.*, 0.05)
  - P-value: $P(S|H_0$ is true$)$
    → Probability of getting the alignment score S even if unrelated randomly selected sequences are aligned
  - If P < Significance level, reject H_0
    (The score cannot be obtained just by aligning unrelated sequences)

32

Extreme value distribution

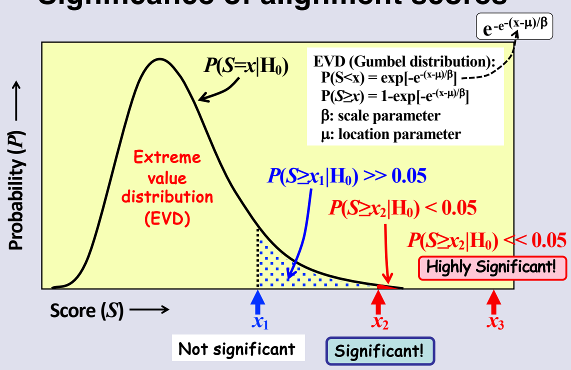
- P-value: $P(S|H_0$ is true$)$
  - Calculated from the test statistic S
    → Need to know the probability distribution of the test statistic S under the null hypothesis, H_0



Distribution of alignment scores follow
Extreme Value Distribution
(Gumbel distribution)
The probability distribution of highest values in an experiment (*e.g.*, optimal alignment scores)

33

Significance of alignment scores

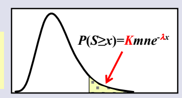


34

Significance of alignment scores

- $P(S\geq x|H_0)$: probability of getting the alignment score $S \geq x$ by chance

Karlin-Altschul equation (Karlin and Altschul 1990)
$P(S\geq x) = 1 - \exp[-Kmne^{-\lambda x}] \approx Kmne^{-\lambda x}$



- K and λ: parameters based on a given scoring matrix and the amino acid composition of the sequences
- m and n: lengths of sequences aligned

EVD (Gumbel distribution):
$P(S\geq x) = 1-\exp[-e^{-(x-\mu)/\beta}]$
$\lambda = 1/\beta$, $\mu = (\ln Kmn)/\lambda$

➜ Solved for ungapped local alignments
➜ Can be applied for gapped local alignments

- E-value = $P(S\geq x|H_0) \times N$
  N: the number of sequences in the dataset
  Expected number of randomly selected sequences in the dataset that have alignment scores $\geq x$

E-value ≠ P-value

35

Estimation of K, λ, and P-value

- Estimation of K and λ from empirical distribution of random alignment scores (used in LALIGN and PRSS)
  1) The second sequence is shuffled many times
     → Simulates random sequences
  2) Smith-Waterman local alignment score is calculated from each alignment: $P(S\geq x|H_0)$
  3) The distribution is fitted to an extreme value distribution to obtain estimates of K and λ
  4) P-value is estimated based on the K and λ, and the original alignment score x: $P(S\geq x) \approx Kmne^{-\lambda x}$

36

Simulation of random score distribution
(example)

[Input sequences]
- RECA_ECOLI (P0A7G6; 353 amino acids)
- RAD51_YEAST (P25454; 400 amino acids)

Smith-Waterman local alignment score = 293
(BLOSUM50, gap opening: -10, gap extension: -1)

```
RECA_ECOLI        3 IDENK-QKALAAALGQIEKQFGKGSIMRLGEDRSMDVETISTGSLSLDIA     51
                    |.|.| .|.|..|...:...|...:...:   |..::..::|||.:||..
RAD51_YEAST     124 ISEAKADKLLNEAARLVPMGFVTAADFHM---RRSELICLTTGSKNLDTL    170

RECA_ECOLI       52 LGAGGLPMGRIVEIYGPESSGKTTL------TLQV-IAAAQREGKTCAFI     94
                    || ||:..|.|.|::|...:||:.|      |.|: :.....||| |.:|
RAD51_YEAST     171 LG-GGVETGSITELFGEFRTGKSQLCHTLAVTCQIPLDIGGGEGK-CLYI    218

RECA_ECOLI       95 DAEHALDPI----YARKLGVDID----NLLCSQPDTGEQALEICDALAR-    135
                    |.|..:.|:    .|:::|:|.|    |:..:....::.|.:.||.|:
RAD51_YEAST     219 DTEGTFRPVRLVSIAQRFGLDPDDALNNVAYARAYNADHQLRLLDAAAQM    268

RECA_ECOLI      136 --SGAVDVIVVDSVAALTPKAEI--EGEIGDSHMGLAARMMSQAMRKLAG    181
                      .....:||||||.||. :.:.  .||:...:|.||..|  :|:::||
RAD51_YEAST     269 MSESRFSLIVVDSVMALY-RTDFSGRGELSARQMHLAKFM--RALQRLA-    314

RECA_ECOLI      182 NLKQSNTLLIFINQIRMKI--GVMFG-NPETTTGGNALKFYASVRLDIRR    228
                      .|....::..||:..::  |:.|. :|:...|||.:...::.||
RAD51_YEAST     315 --DQFGVAVVVTNQVVAQVDGGMAFNPDPKKPIGGNIMAHSSTTRL----    358

RECA_ECOLI      229 IGAVKEGENVVGSETRVKVVKNKIAAPFKQAEFQILYGEGI    269
                      ..|:|:   |.:...||| :....|..:..|.| |.:|:
RAD51_YEAST     359 --GFKKGK---GCQRLCKVV-DSPCLPEAECVFAI-YEDGV    392
```

Using EMBOSS WATER @ EBI

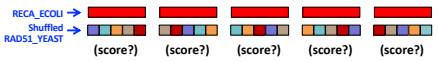
37

Simulation of random score distribution
(example)

[Input sequences]
- RECA_ECOLI (P0A7G6; 353 amino acids)
- RAD51_YEAST (P25454; 400 amino acids)

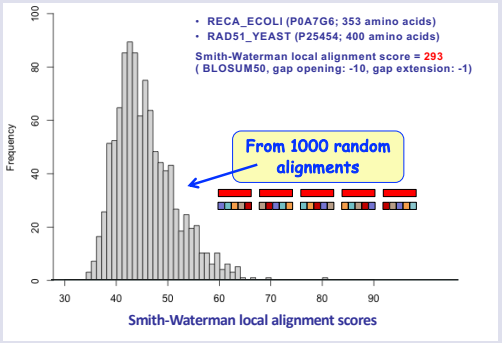
Smith-Waterman local alignment score = 293
(BLOSUM50, gap opening: -10, gap extension: -1)

1. Shuffle RAD51_YEAST 1000 times (generate 1000 random sequences)
2. Align with RECA_ECOLI (generate 1000 random local alignments)



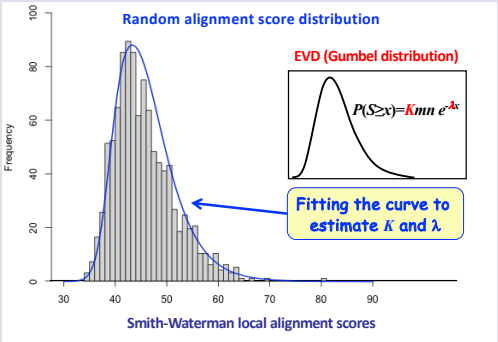
38

Simulation of random score distribution
(example)



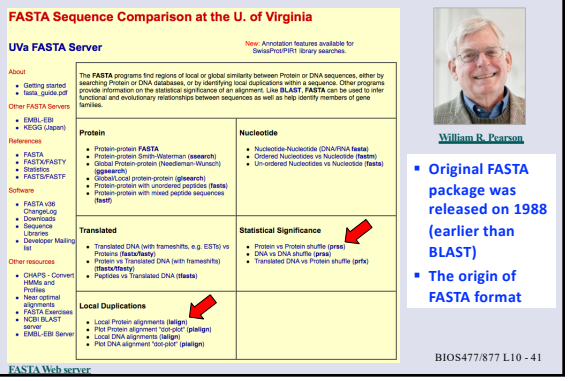
39

Simulation of random score distribution
(example)



40

FASTA Web server by William Pearson

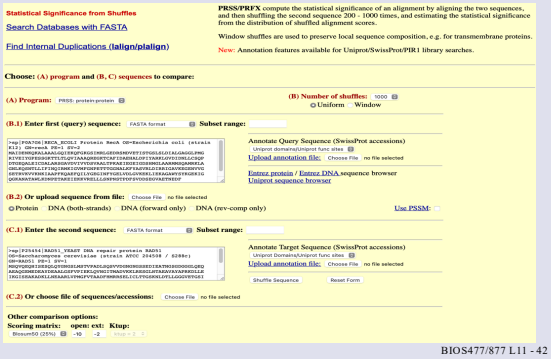


William R. Pearson

- Original FASTA package was released on 1988 (earlier than BLAST)
- The origin of FASTA format

41

Statistical analysis of alignments by PRSS



42