bioinformatics 1 -- lecture 7

Probability and conditional probability

Random sequences and significance (real sequences are *not* random)

Erdos & Renyi: theoretical basis for the significance of an alignment given its length and score

Extreme value distribution, better than a Gaussian for estimating significance.

E-values

Probability

Susan B Anthony

coin S

The **Q**ueen

coin Q

Ireland

coin I

"P(H)" means the probability of H, heads.

$0 \le P \le 1$

Unconditional probabilities

Joint probability of a sequence of 3 flips, given any one (un)fair coin, is the product.

$$P(HHT) = P(H)*P(H)*P(T)$$

Conditional probabilities

If the coins are "unfair" (not 50:50), then P(H) depends on the coin you choose (S,Q or I). P(H) is "conditional" on the choice of coin, which may have its own odds.

$$P(S,H) = P(S)*P(H|S)$$

Conditional probabilities

"P(A|B)" means the *probability of A given B*, where A is the **result** or **observation**, B is the **condition**. (The *condition* may be a *result/observation* of a previous *condition*.)

P(H|S) is the probability of H (heads) given that the coin is S.

In general, the probability of two things together, A and B, is

$$P(A,B) = P(A|B)P(B) = P(B|A)P(A)$$

Divide by P(B), you get Bayes' rule:

$$P(A|B)= P(B|A)* P(A) /P(B)$$

To reverse the order of conditional probabilities, multiply by the ratio of the probabilities of the conditions.

Scoring alignments using P

For each aligned position (match), we get P(A|B) , which is the **substitution probability**. Ignoring all but the first letters, the probability of these two sequences being homologs is:

$P(s_1[1]|s_2[1])$

substitution of $s_2[1]$ for $s_1[1]$

Ignoring all but the first two letters, it is:

$P(s_1[1]|s_2[1]) \times P(s_1[2]|s_2[2])$

Counting all aligned positions:

$\Pi_i P(s_1[i]|s_2[i])$

Each position is treated as a different coin.
(An independent stochastic process).

Log space is more convenient

$$\log \Pi_i P(s_1[i]|s_2[i])/P(s_1[i]) = \Sigma_i S(s_1[i]|s_2[i])$$

where $S(A|B) = 2*\log_2(P(A|B)/P(A))$ = BLOSUM score

This is the form of the substitution score, Log-likelihood ratios (alias LLRs, log-odds, lods). Usually "2 times $\log_2$ of the probability ratio" (or "half-bits").

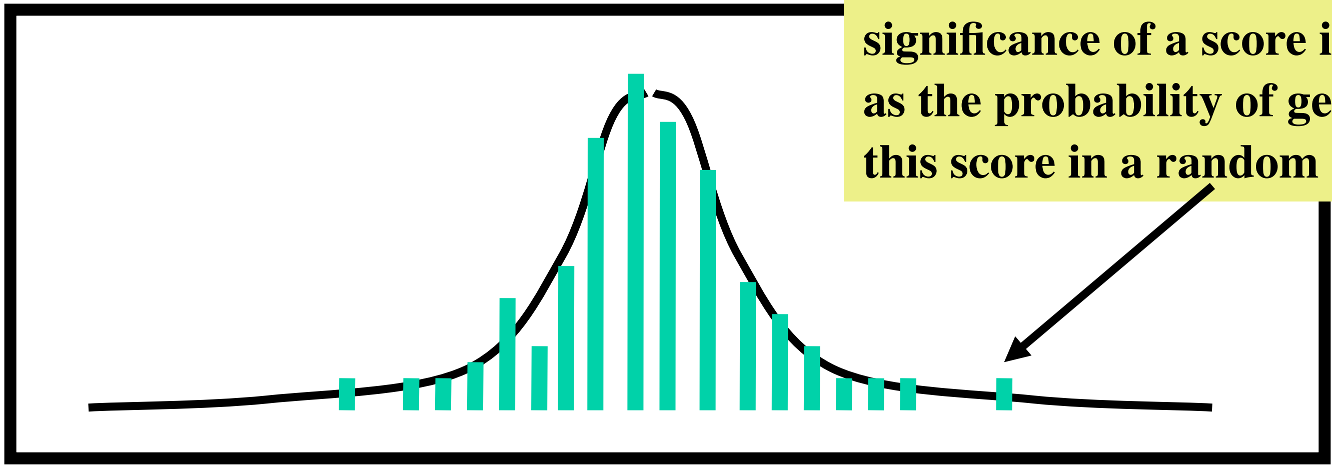
Dayhoff's randomization experiment

Aligned scrambled Protein A versus scrambled Protein B

100 times (re-scrambling each time).

NOTE: scrambling does not change the AA composition!

Results: A Normal Distribution



significance of a score is measured as the probability of getting this score in a random alignment

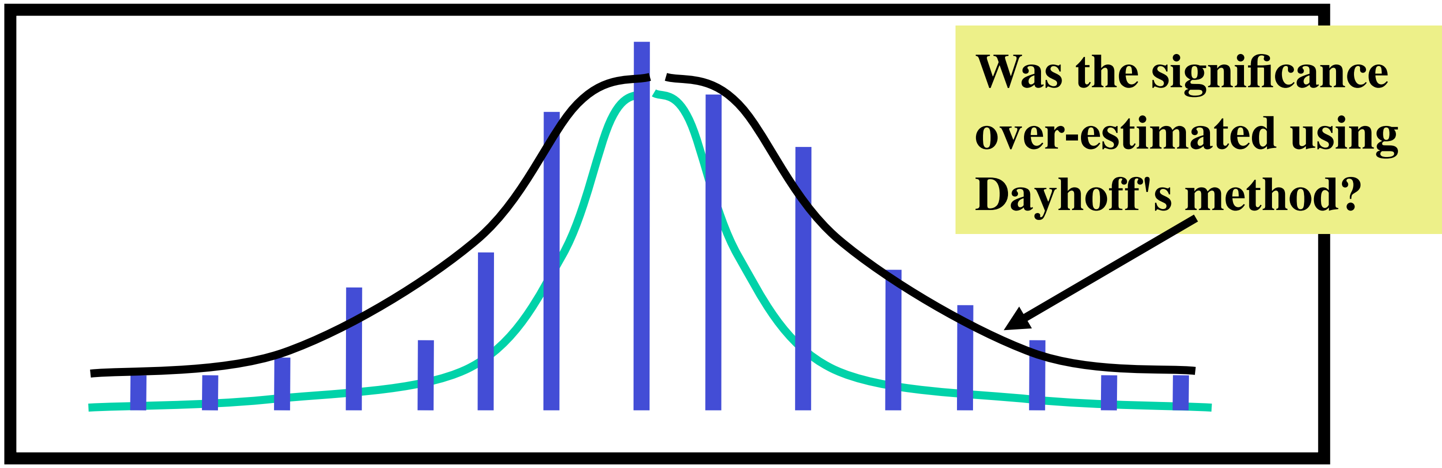
Lippman's randomization experiment

Aligned Protein A to 100 natural sequences, not scrambled.

Results: A wider normal distribution (Std dev = ~3 times larger)

WHY? Because natural sequences are different than random.

Even unrelated sequences have similar local patterns, and uneven amino acid composition.



Was the significance over-estimated using Dayhoff's method?

Lippman got a similar result if he randomized the sequences by **words** instead of letters.

complexity

= sequence heterogeneity

A *low complexity* sequence is homogeneous in its composition. For example:

`AAAAAAAHAAAAAAAAKAAAAAEAA`

is a *low-complexity* sequence.

Compared to other sequences, there are relatively few ways to make a 26-residue sequence that has 23 A's, 1 H, 1 K and 1 E.

What is the effect of low-complexity regions on the score distribution?

(1) .... from a Dayhoff-type randomization experiment?

(2) .... from a Lipmann-type randomization experiment?

What is the effect on significance?

Wider score distribution ---> lower significance of a given score.

Narrower score distribution ---> higher significance.

Local patterns (words) increase the standard deviation of random alignments scores.

The two-letter sequence "PG" occurs more often than expected by chance, perhaps because PG occurs in beta-turns.

If non-homolog sequences are actually made of of **small words**, instead of letters, then how will the score distribution be effected? Narrower? Wider?

- Whole word matches have higher scores.
- Whole word mismatches have lower scores.
- Total score of an alignment is the sum of word scores, which are more variable.

Expected

Expectation

Expectation value

Expectation value for coin tosses

Consider a fair coin, tossed n=25 times. The sequence is, let’s say:

HTHTHTTTHHHTHTTHHTHHHHHTH

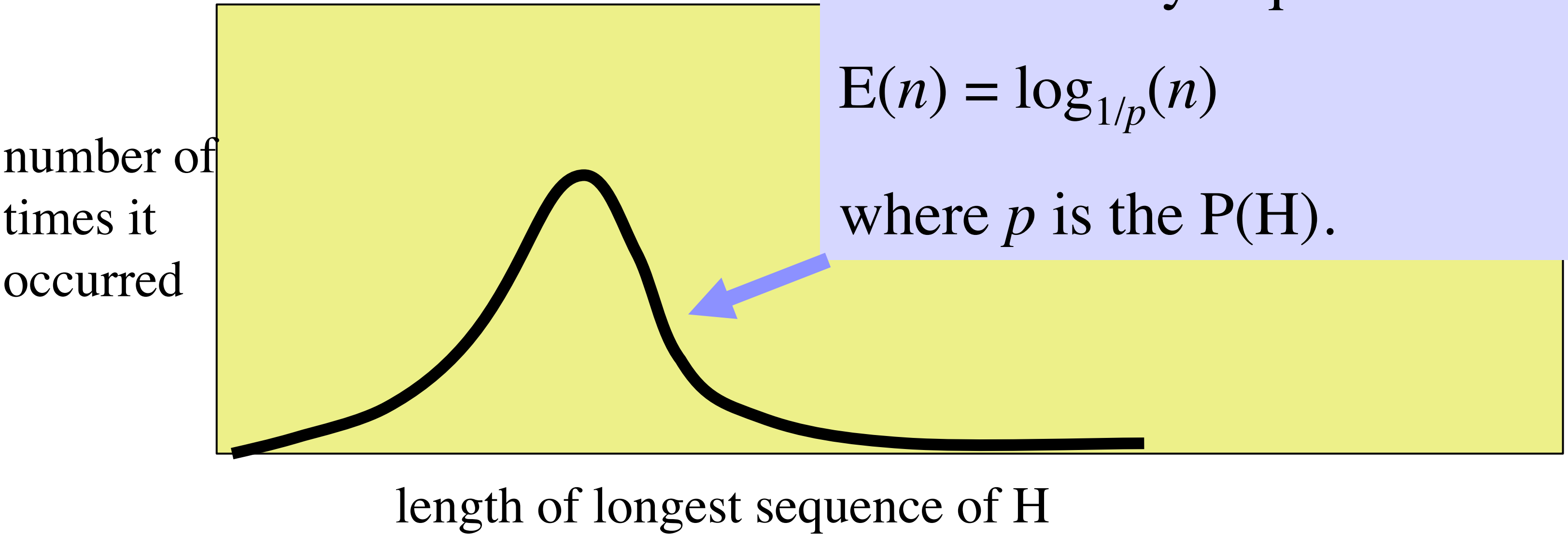
The longest row of H’s is 5 in this case.

What is the *expected length of the longest row* of H's given n?

Erdos & Renyi equation:

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

where p is the P(H).



Heads is like *match*, tails is like *mismatch*

Similarly, we can define an **expectation value, E(M),** for the **longest row of matches** in an alignment of length *n*. E(M) is calculated similar to the heads/tails way, using the Erdos & Renyi equation (*p* is the odds of a match, *1-p* is the odds of a mismatch):

$E(M) = \log_{1/p}(M)$ ← expectation given an alignment of length *M*

But over all possible alignments of two sequences of length *n*, the number is

$E(M) = \log_{1/p}(n*n) = 2 \log_{1/p}(n)$

If the two sequences are length *n* and length *m*, it is

$E(M) = \log_{1/p}(mn)$ [+ *some constant terms that don't depend on m and n*]

Heads/tails = match/mismatch

Theoretically derived equation for the *expectation value* for M, the longest block of Matches.

$E(M) = \log_{1/p}(mn) + \log_{1/p}(1-p) + 0.577\log(e) - 1/2$

Note that we can define a number K such that $\log_{1/p}(K)$ = constant terms.

$E(M) = \log_e(Kmn)/\lambda$

...where $\lambda = \log_e(1/p)$

In class exercise: empirical expectation value

Open Geneious.

Using DNA from any viral genome, extract **100 bases at random**, twice. (*make sure these two sequences non-overlapping!*)

Align the two extracted segments (default DNA parameters)

Find the **longest string of identity matches**. Write it down.

Delete the extracted sequences, and do it again. As many times as you can.

Plot the frequency vs lengths on the board.

Theoretical expectation value

$$E(M) = \log_{1/p}(mn) + \log_{1/p}(1-p) + 0.577\log(e) - 1/2$$

$$E(M) = \log_e(Kmn)/\lambda$$

Solving, using $p=0.25$, we get $K=0.6237$, $\lambda = \log_e(4) = 1.386$, $m=n=100$

$$E(M) = \log_e(Kmn)/\lambda$$
$$= 6.3$$

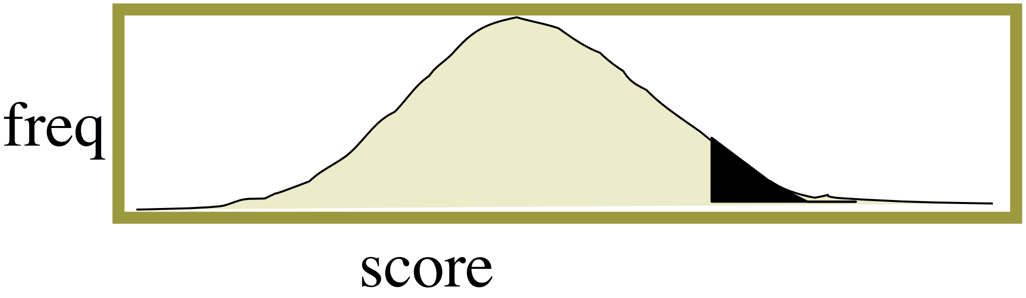
Is this what we found for the most probable longest string of identity matches?

$P(S > x)$

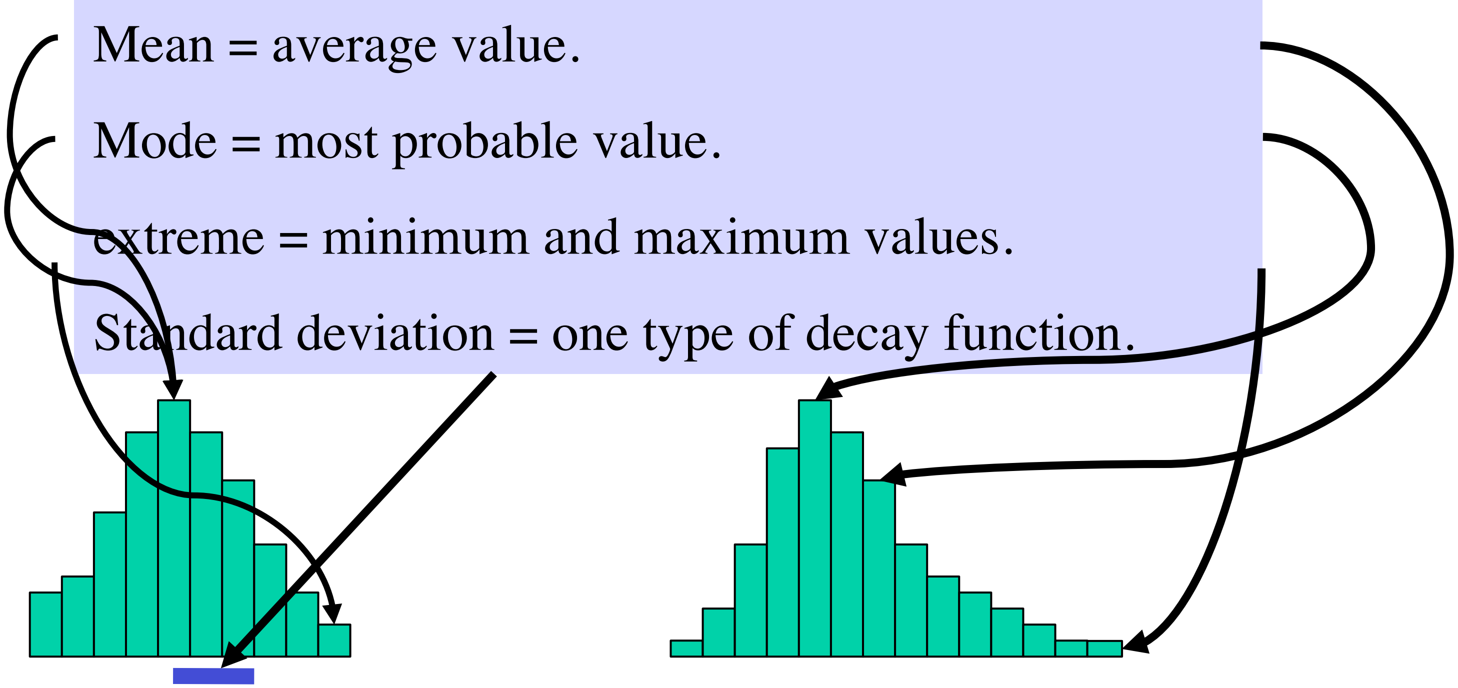
E(M) gives us the expected length of the longest number of matches in a row. But, what we really want is the answer to this question:

How *good* is the score *x*? (i.e. how significant)

So, we need to model the whole distribution of chance scores, then ask how likely is it that my score or greater comes from that model.



Distribution Definitions



Mean = average value.

Mode = most probable value.

extreme = minimum and maximum values.

Standard deviation = one type of decay function.

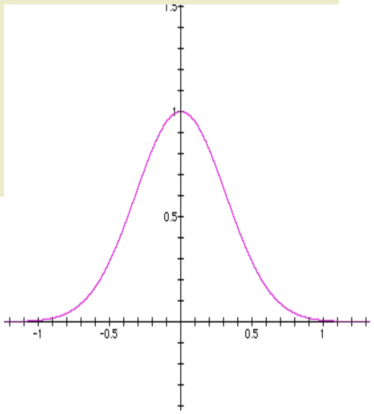
For a variable whose distribution comes from *extreme value*, such as random sequence alignment scores, the score must be greater than expected from a normal distribution to achieve the same level of significance.

A Normal Distribution

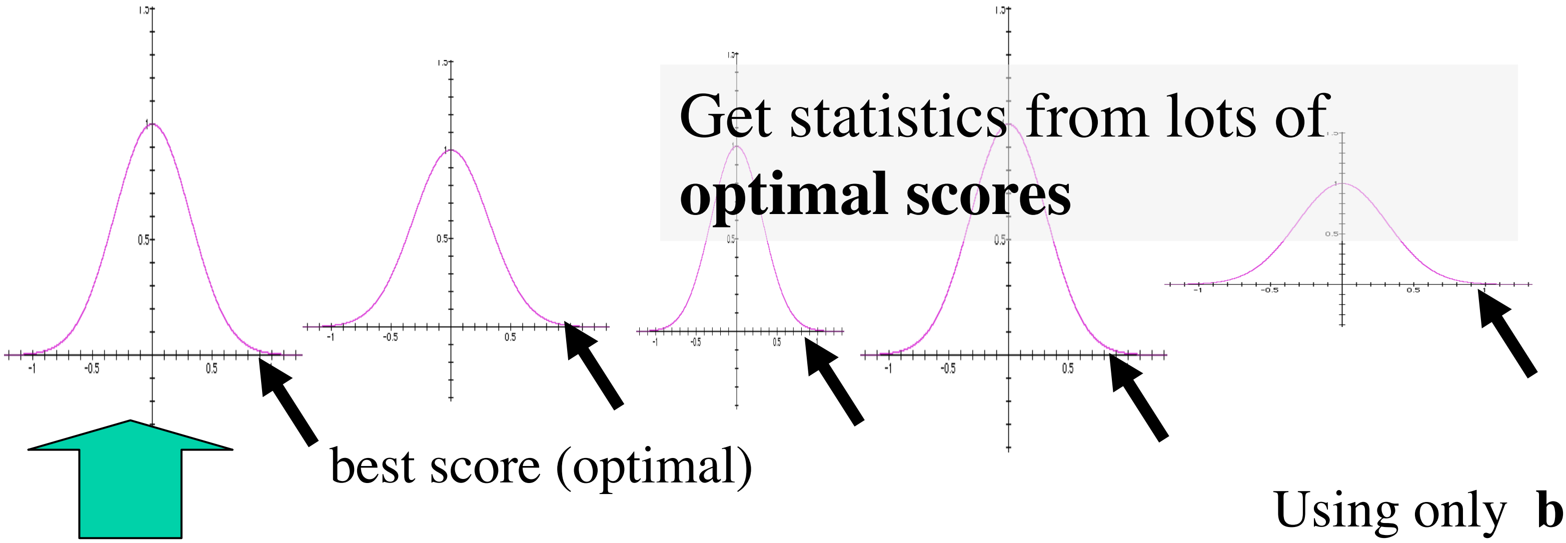
Usually, we suppose the likelihood of deviating from the mean by *x* in the positive direction is the same as the likelihood of deviating by *x* in the negative direction, and the likelihood of devating by *x* decreases as the power of *x*.

Why? Because multiplying probabilities gives this type of curve.

This is called a Normal, or Gaussian distribution.



Extreme value distribution, a distribution derived from extreme values

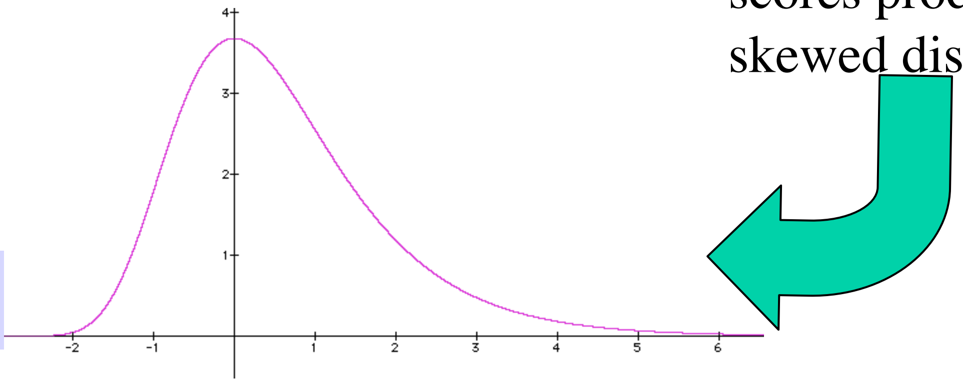


Get statistics from lots of **optimal scores**

best score (optimal)

all possible scores
for two sequences =
Normal distrib.

Using only **best**
scores produces a
skewed distrib.

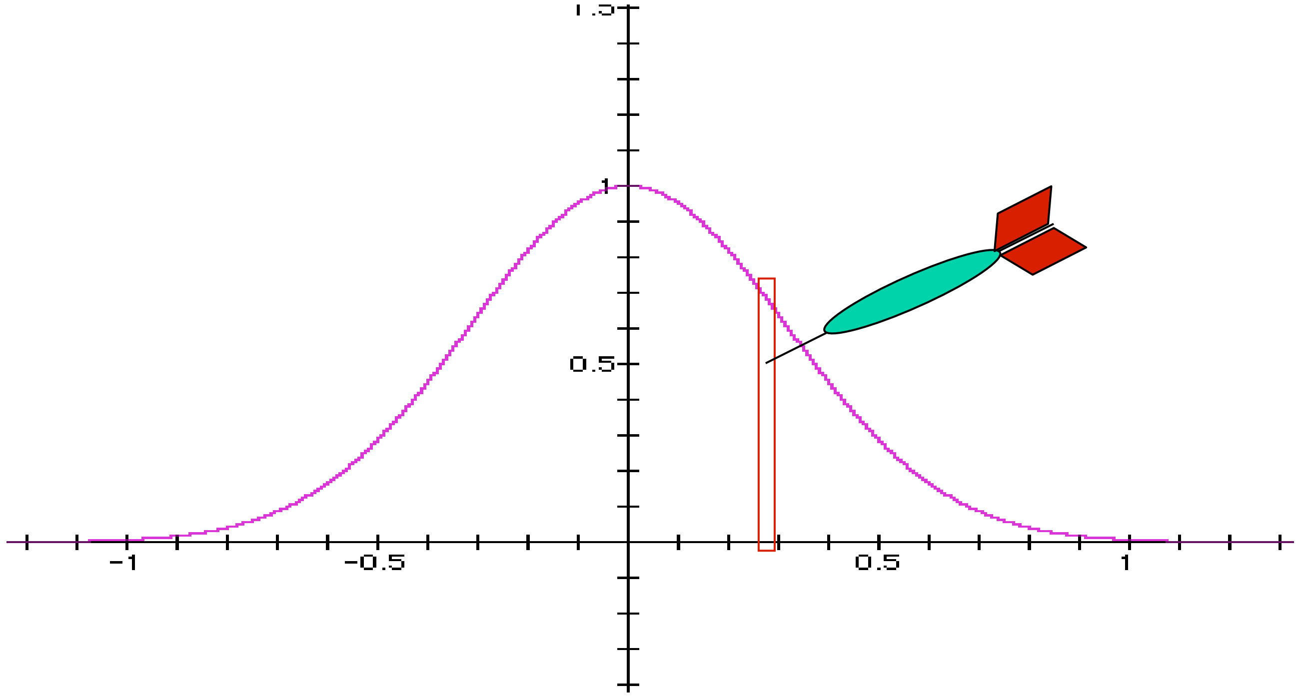


Extreme value distribution

$$y = \exp(-x - e^{-x})$$

EVD has this shape. But the Mode and decay parameters depend on the data.

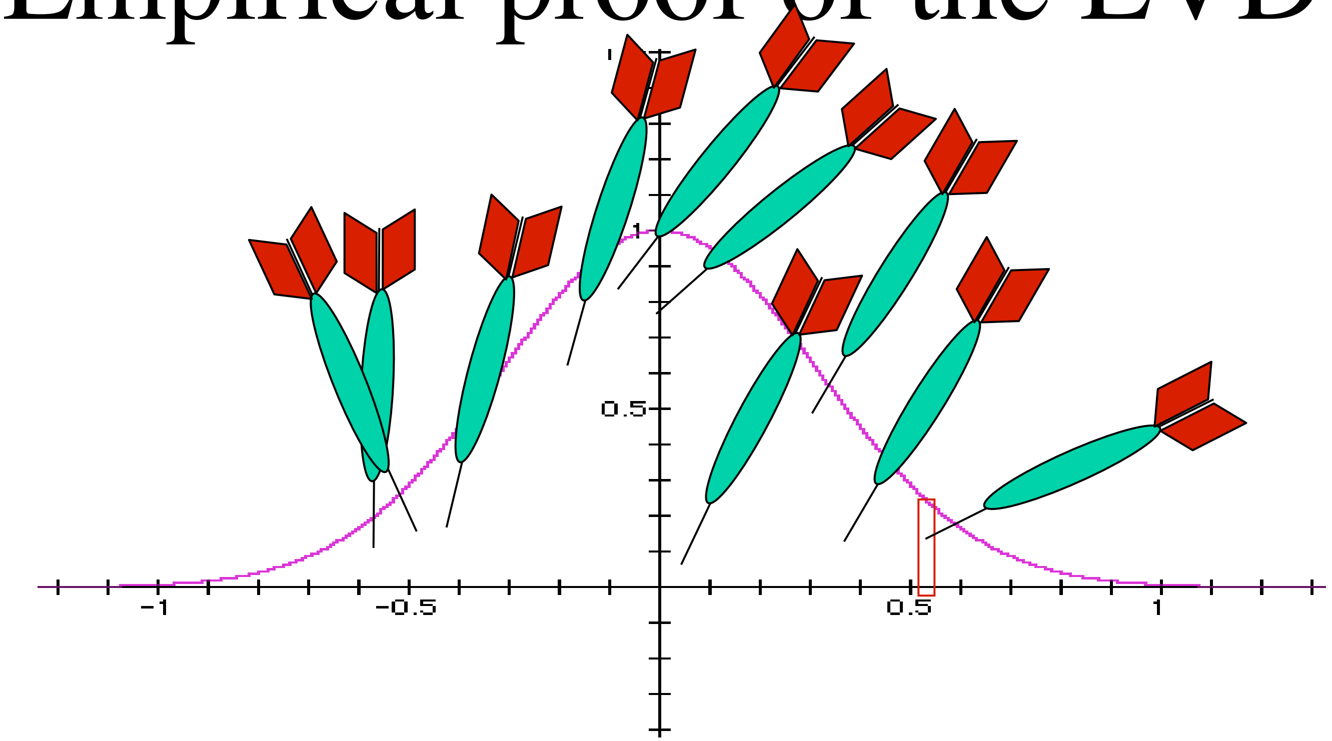
Empirical proof of the EVD



Suppose you had a Gaussian distribution “dart-board”. You throw 1000 darts randomly. Score your darts according the number on the X-axis where it lands. What is the probability distribution of scores?

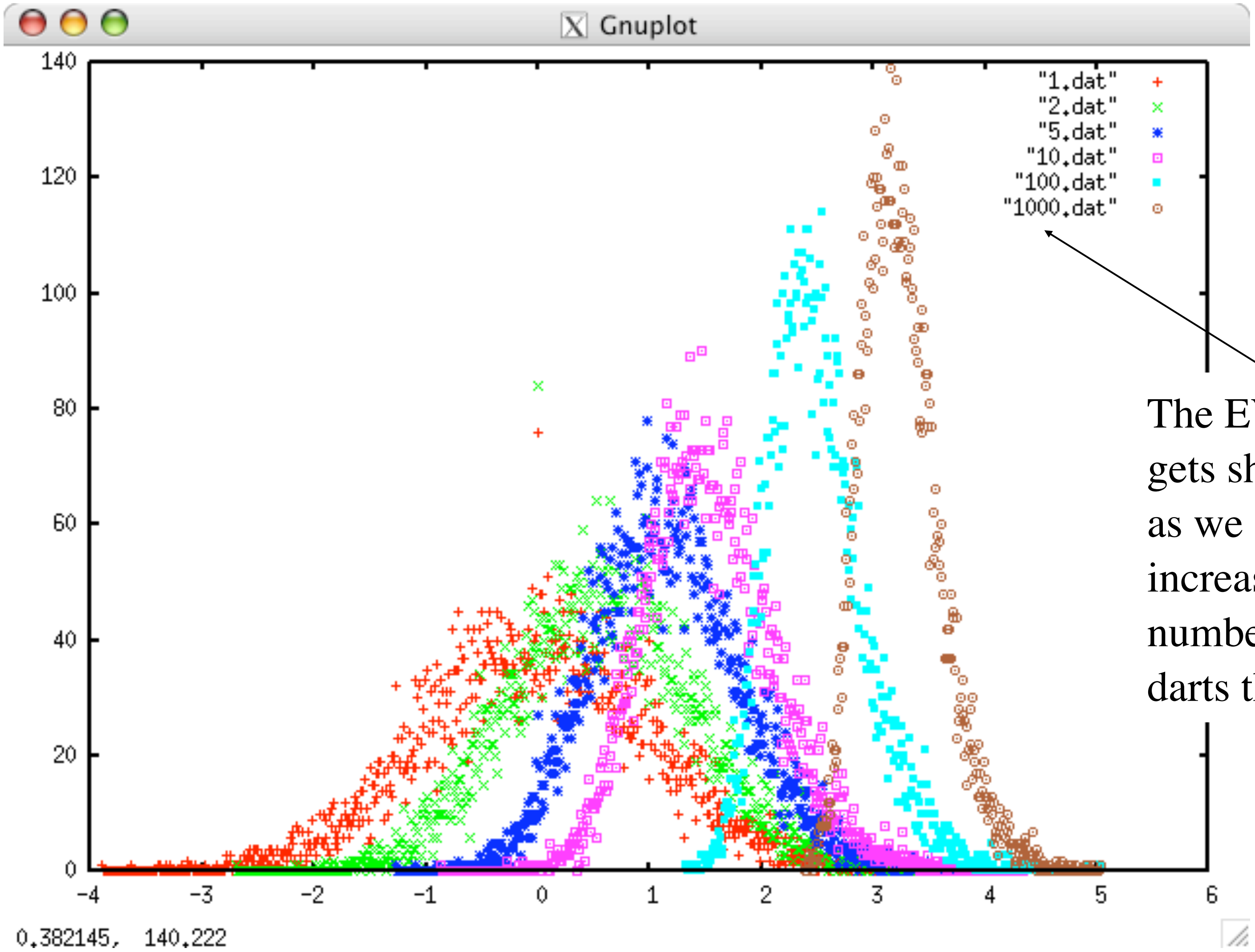
Answer:The same Gaussian distribution! (duh)

Empirical proof of the EVD



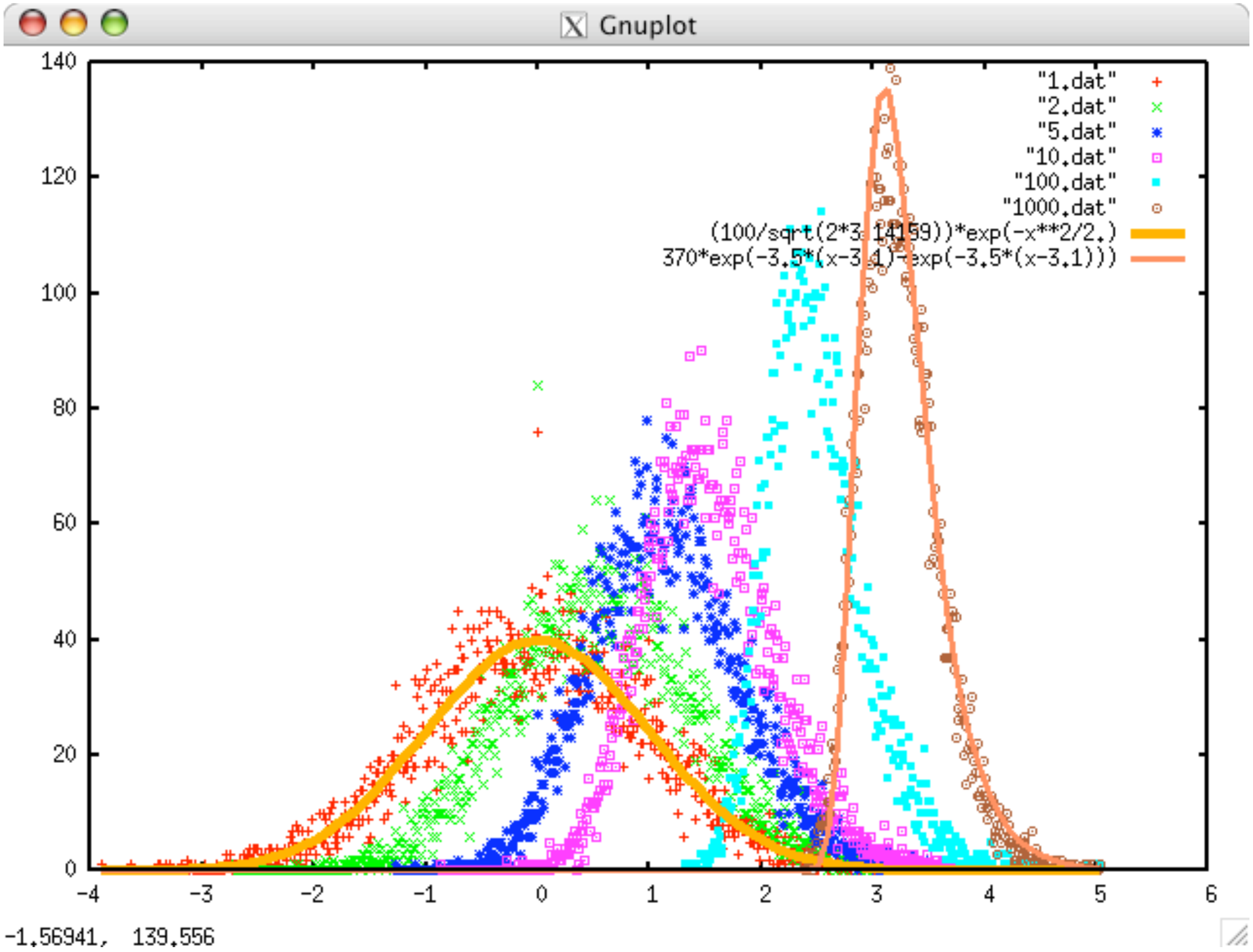
What if we throw 10 darts at a time and remove all but the highest-scoring dart. Do that 1000 times. What is the distribution of the scores?

Empirical proof of the EVD



The EVD gets sharper as we increase the number of darts thrown

Empirical proof of the EVD



Extreme value distribution for sequence scores

The EVD with mode $u\lambda$ and decay parameter λ:

$$y = \exp(-x - e^{-\lambda(x-u)})$$

The mode, from the Erdos & Renyi equation:

$$u = \log_e(Kmn)/\lambda$$

substituting gives:

$$P(x) = \exp(-x - e^{-\lambda(x-\ln(Kmn)/\lambda)})$$

Integrating from x to *infinity* gives:

$$P(S \geq x) = 1 - \exp(-Kmne^{-\lambda x})$$

the scoring function and λ

λ is calculated as the value of *x* that satisfies:

$$\Sigma p_i p_j e^{S_{ij}x} = 1$$

Substitution matrix values.

S_{ij} is the log-likelihood ratio, log[P(i->j)/$p_i p_j$]. So, $e^{S_{ij}}$ is the likelihood ratio, P(i->j)/$p_i p_j$. So $e^{S_{ij}x}$ is e^xP(i->j)/$p_i p_j$. If $e^x = p_i p_j$ (on average), then $e^{S_{ij}x}$ is approximately the observed P(i->j) the sum over all amino acid pairs of P(i->j) is one by definition. So λ = log(e^x) = the log of the average expected subsitution probability $p_i p_j$.

voodoo mathematics

For values of *x* greater than 1, we can make this approximation:

$$1\text{-}\exp[-e^{-x}] \approx e^{-x}$$

That means, $P(S \geq x) = 1 - \exp(-Kmne^{-\lambda x})$

becomes, $P(S \geq x) \approx Kmne^{-\lambda x}$

taking the log of both sides, $\log(P(S \geq x)) \approx \log(Kmn) - \lambda x$

We can plot $log(P(S \geq x))$ versus x

(using a large number of known false alignment scores x),

and fit it. The slope is $-\lambda$, the intercept is $log(Kmn)$

Finding the EVD parameters

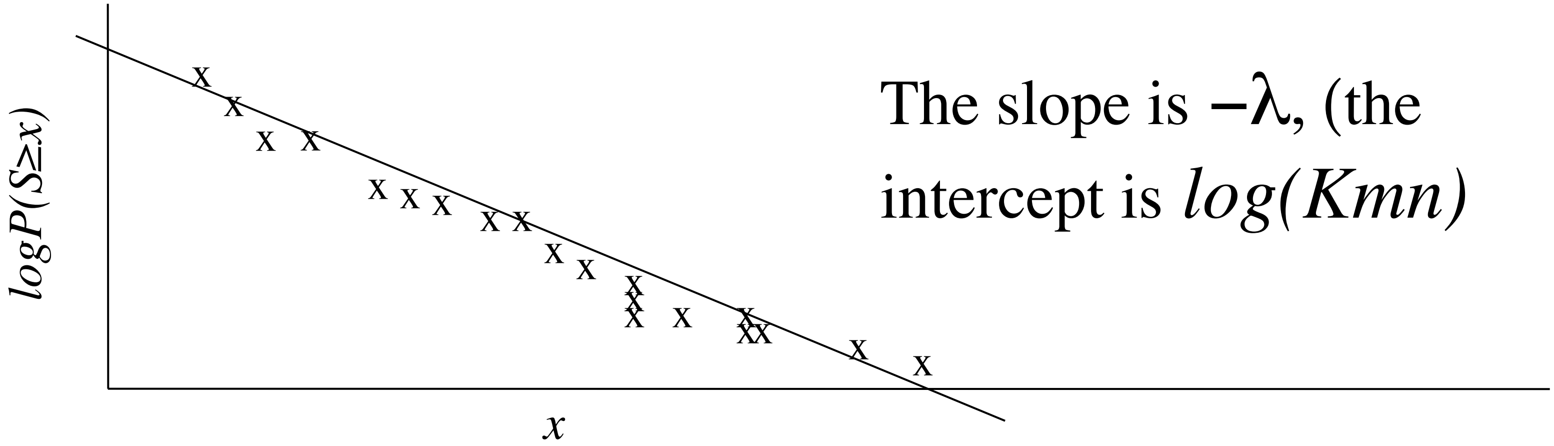
Estimated P(score x or higher) given random alignments:

$$P(S \geq x) \approx K m n e^{-\lambda x}$$

Taking the log of both sides,

$$\log(P(S \geq x)) \approx \log(K m n) - \lambda x$$

We generate a large number of known false alignment scores *S*, (all alignments with the same two lengths *m* and *n*), plot *log(P(S≥x))* versus *x* , and fit the data to a line!



The slope is $-\lambda$, (the intercept is *log(Kmn)*

Geneious exercise: fitting EVD to false scores

1. Scramble a protein sequence “by words”. (cut and paste 5-10 residue segments about 20 times)
2. Submit to BLAST. Set max e-value = 1000, max hits=200, gapcost 12 , uncheck “fully annotate..”
3. Sort by “bit-score” (= alignment score) high to low.
4. Save *n* (row number) and S (bit score) on a list. Randomly choose from your list.
5. Write log(*n*) next to S.
6. Plot S on the x-axis, log(n) on the y-axis. Find the slope and intercept.
7. Find the p-value for S=50. Find the e-value, assuming the size of the database was 1,000,000.

e-values in BLAST

- Every BLAST "hit" has a bit-score, *x*, derived from the substitution matrix.
- Parameters for the EVD have been previously calculated for *m* and *n*, the lengths of the database and query.
- Applying the EVD to *x* we get $P(S \geq x)$, which is our "*p-value*"
- To get the "*e-value*" (expected number of times this score will occur over the whole database) we multiply by the size of the database *m*.

$$\mathit{e\text{-}value}(x) = P(S \geq x) * m$$

where x is the alignment score, m is the size of the database, and P is calculated from false alignment scores.

Matrix bias in local alignment

In Local Alignment we take a MAX over *zero (0) and three other scores (diagonal, across, down)*. **Matrix Bias is added to all match scores**, so the average match score,and the extremes, can be adjusted.

What happens if match scores are....?

all negative? :	Best alignment is always **no alignment**.
all positive? :	Best alignment is **gapless**, **global-local**.
average positive? :	Best alignment is **local** (longer). Typical random alignment is **local**.
average negative? :	Best alignment is **local** (shorter). Typical random alignment is **no alignment**.

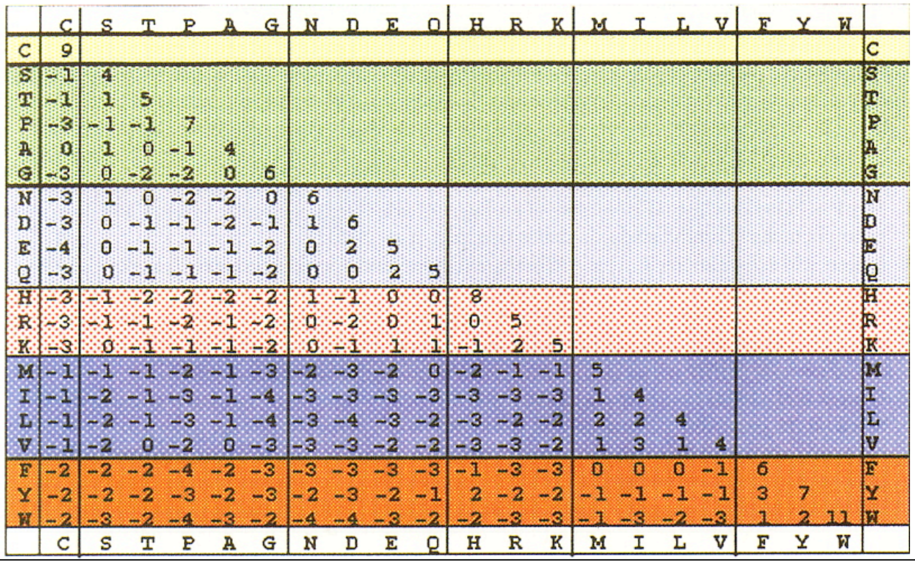
Altschul's Principle

For local DP alignment, the match (substitution) scores should be

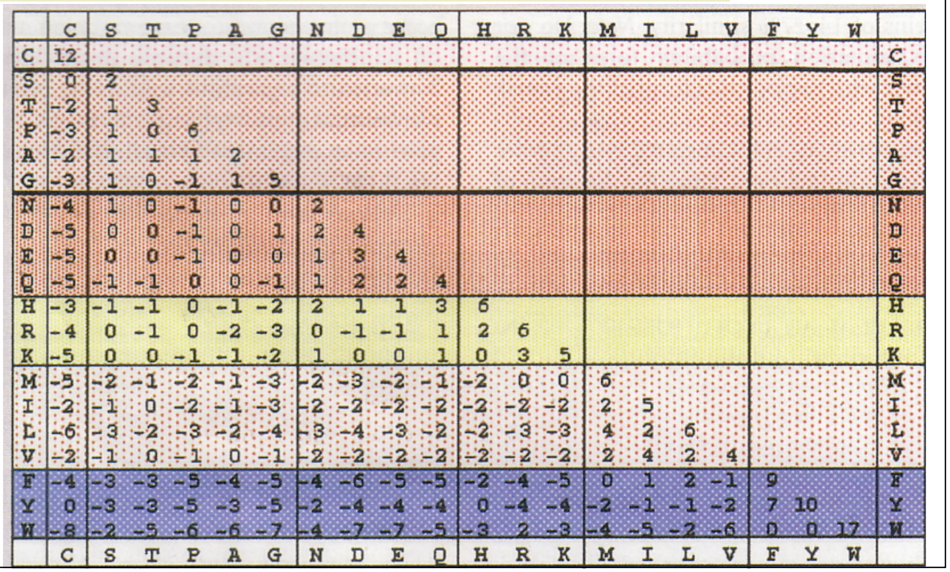
> zero for a match, and

< zero for a mismatch,

on average. (some mismatches may have a > 0 score)



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

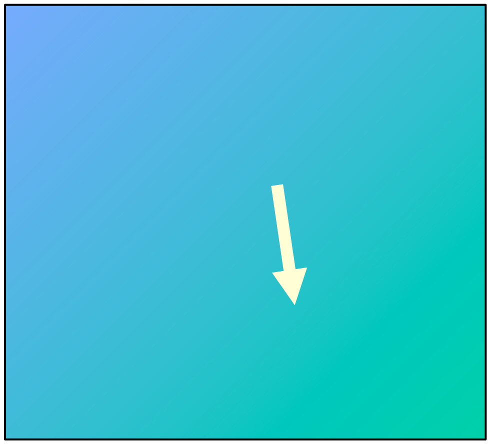


	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	
C	12																				C
S	0	2																			S
T	-2	1	3																		T
P	-3	1	0	6																	P
A	-2	1	1	1	2																A
G	-3	1	0	-1	1	5															G
N	-4	1	0	-1	0	0	2														N
D	-5	0	0	-1	0	1	2	4													D
E	-5	0	0	-1	0	0	1	3	4												E
Q	-5	-1	-1	0	0	-1	1	2	2	4											Q
H	-3	-1	-1	0	-1	-2	2	1	1	3	6										H
R	-4	0	-1	0	-2	-3	0	-1	-1	1	2	6									R
K	-5	0	0	-1	-1	-2	1	0	0	1	0	3	5								K
M	-5	-2	-1	-2	-1	-3	-2	-3	-2	-1	-2	0	0	6							M
I	-2	-1	0	-2	-1	-3	-2	-2	-2	-2	-2	-2	-2	2	5						I
L	-6	-3	-2	-3	-2	-4	-3	-4	-3	-2	-2	-3	-3	4	2	6					L
V	-2	-1	0	-1	0	-1	-2	-2	-2	-2	-2	-2	-2	2	4	2	4				V
F	-4	-3	-3	-5	-4	-5	-4	-6	-5	-5	-2	-4	-5	0	1	2	-1	9			F
Y	0	-3	-3	-5	-3	-5	-2	-4	-4	-4	0	-4	-4	-2	-1	-1	-2	7	10		Y
W	-8	-2	-5	-6	-6	-7	-4	-7	-7	-5	-3	2	-3	-4	-5	-2	-6	0	0	17	W
	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	

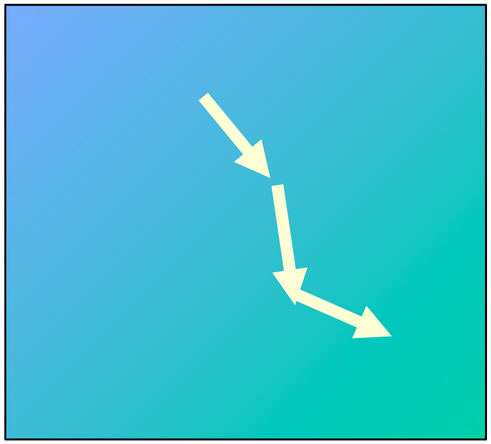
What happens with matrix bias?

If we add a constant to each value in the substitution matrix, it favors matches over gaps. As we increase matrix bias...

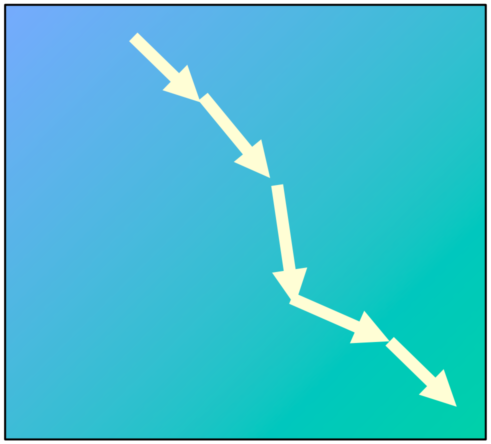
- Longer alignments are more common in random sets.
- Longer alignments are less significant.



Negative matrix bias



No matrix bias



Positive matrix bias

summary of significance

- Significance of a score is measured by the **probability of getting that score by chance**.
- History of modeling “chance” in alignments
  - 1970’s Dayhoff: Guassian fit to scrambled alignments
  - 1980’s Lipman: Gaussian fit to false alignments
  - 1990’s Altschul: EVD fit to false alignments

summary of significance

- The expectation value for the maximum length of a match between two sequences, lengths *n* and *m*, given the probability of a match *p*, has a theoretical solution. $\log_{(1/p)}(nm)$, the Erdos & Lenyi equation.
- The score of an alignment is roughly proportional to the number of matches (local alignments only). Therefore, the expectation value of alignment scores follows the same theoretical equation.

summary

- The Extreme Value Distribution = exp[-x-exp(-x)] models the distribution is over extreme random values (such as optimal, but false, alignment scores).

- The EVD models the length-dependence of the score.

- The parameters (λ,K) of the EVD are determined *empirically* by plotting false scores and fitting.

- Once λ and K have been found, the significance of a given score x is the probability of getting a higher score S from random alignments. This is approximated by integrating the EVD from x to infinity.

$$P(S \geq x) = 1 - \exp(-Kmne^{-\lambda x}) \approx Kmne^{-\lambda x}$$

Pop-quiz

You did a BLAST search using a sequence that has absolutely no homologs in the database. *Absolutely none*.

The BLAST search gave you false "hits" with the top *e-values* ranging from 0 to 20. You look at them and you notice a pattern in the e-values.

How many of your hits have e-value ≤ 10.?