

bioinformatics 1 -- lecture 7

Probability and conditional probability

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

Erdoes & 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

How significant is that?

...a specific
inference. Thanks.

...as opposed to...

...how likely the data would not
have been the result of chance,...

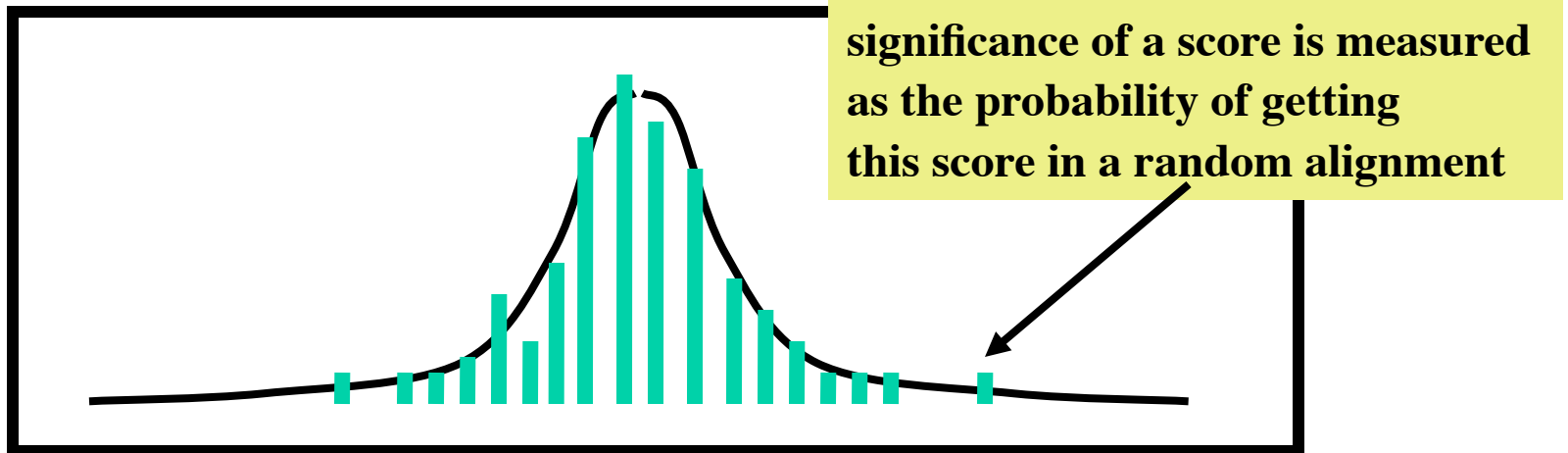
Please give me a number for...

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

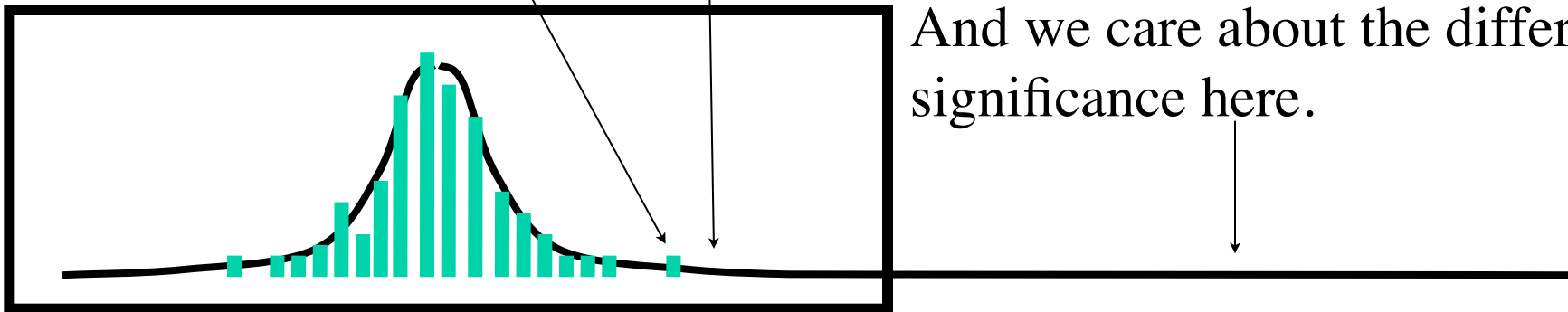


Why do we need a “model” for significance?

Because random scores end here.

And non-random scores start here.

And we care about the difference
significance here.



So we want to be able to calculate the significance.

Easy right? Just fit to a Gaussian (i.e. normal) distribution...?

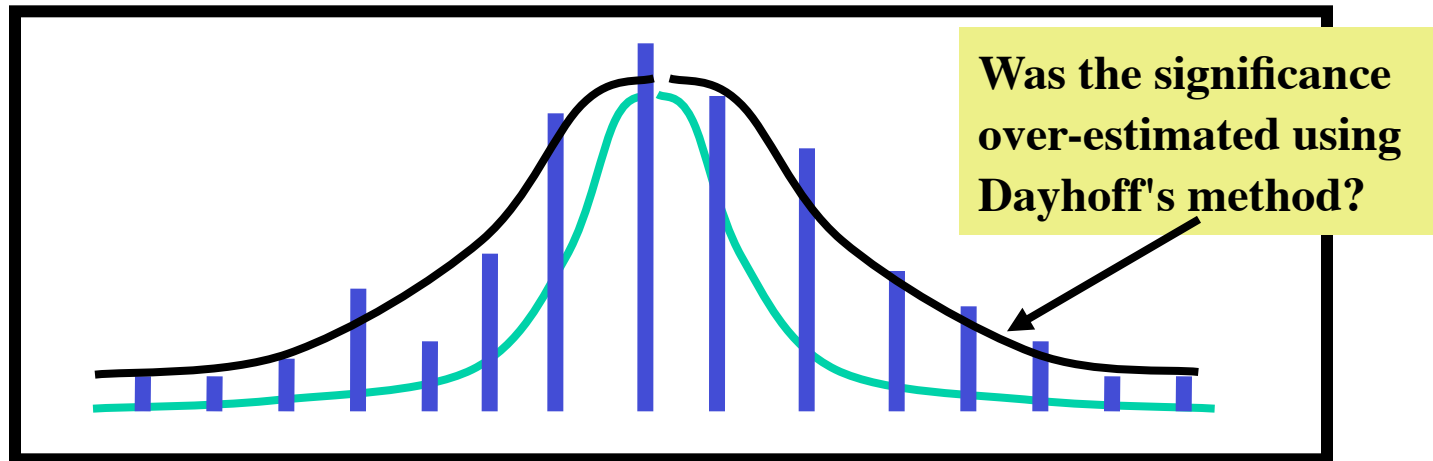
Lippman's randomization experiment

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.



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

Why was Lippman's distribution wider?

low complexity

complexity = sequence heterogeneity

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

AAAAAAAAAHAAAAAAAAAKAAAAAEAA

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.

Why was Lippman's distribution wider? (part 2)

Local patterns (words).

The three-letter sequence PGD occurs more often than expected by chance because PGD occurs in beta-turns.

The amphipathic helix pattern *pnnppnn* (p=polar, n=non-polar) occurs more often than expected because it folds into an alpha helix

- Even sequences that are non-homologous (Lippman's exp) have word matches. But scrambled sequences (Dayhoff's exp) do not have recurrent words.

Dayhoff & Lippman assumed a normal distribution for random alignment scores.

Should it really be a normal distribution?

I guess we need a *theoretical foundation*

Probability

Susan B Anthony



The Queen



Ireland



coin S



coin Q



coin I

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

$$0 \leq P \leq 1$$

Unconditional probabilities

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

$$P(\text{HHT}) = P(\text{H}) * P(\text{H}) * P(\text{T})$$

Conditional probabilities

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

$$P(\text{S,H}) = P(\text{S}) * P(\text{H}|\text{S})$$

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:

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

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

Log space is more convenient

$$\log \prod_i P(s_1[i]|s_2[i])/P(s_1[i]) = \sum_i S(s_1[i]|s_2[i])$$

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”).

The point is...

- Alignment scores are like coin tosses.
- ...with 20 “unfair” coins...
- (...and each coin actually has 20 sides...)

To establish the theoretical foundation, we first go to the simple case.

Theoretical distribution of coin toss alignments.

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

HTHTHTTTHHHTHTTHTHHHHHTH

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 ?

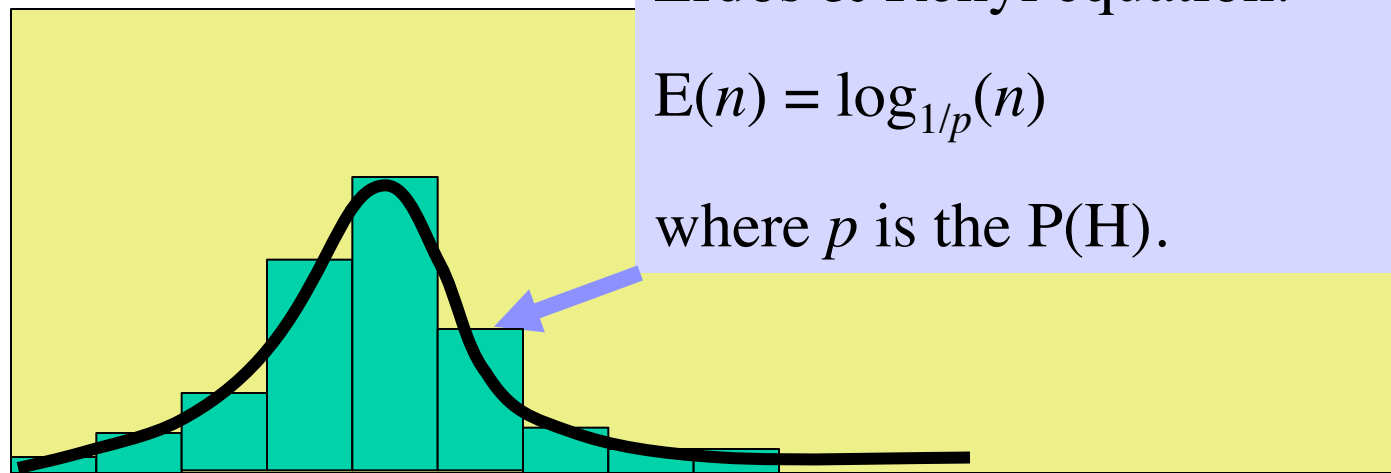
Erdos & Renyi equation:

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

where p is the $P(H)$.

number of times it occurred

Hey! Score never goes below zero.



length of longest sequence of H

Heads = *match*, tails = *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):

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

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

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

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 $\log_{1/p}(K) = \text{constant terms}$.

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

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


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 $K=0.6237$, $\lambda = \log_e(4) = 1.386$, $m=n=100$

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



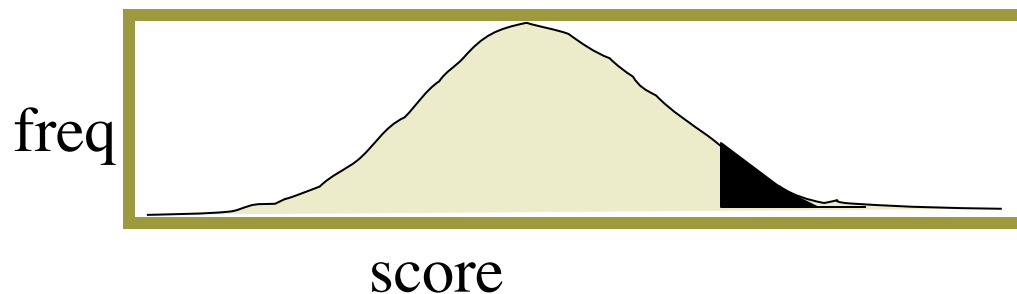
This would be the most likely number of matches in an alignment of two random DNA sequences of length 100 and 100. You can try this! Align randomly selected 100bp DNA using local DP alignment (i.e. LALIGN server). Count the longest string of matches. It is rarely less than 4, rarely greater than 9.

$$P(S > x)$$

$E(M)$ 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.



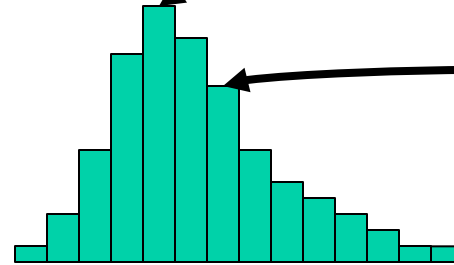
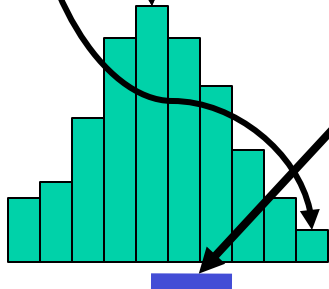
Distribution Definitions

Mean = average value.

Mode = most probable value.

extreme = minimum and maximum values.

Standard deviation = one type of decay function.



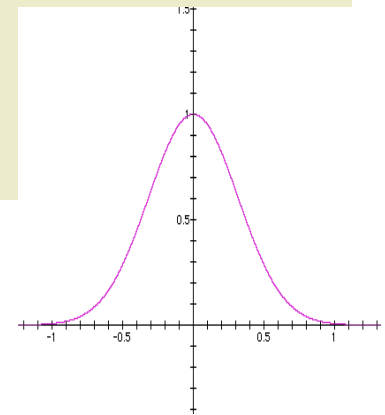
For a variable whose distribution comes from *extreme value*, such as random sequence alignment scores, the score must be greater than expected from a normal distribution to achieve the same level of significance.

A Normal Distribution

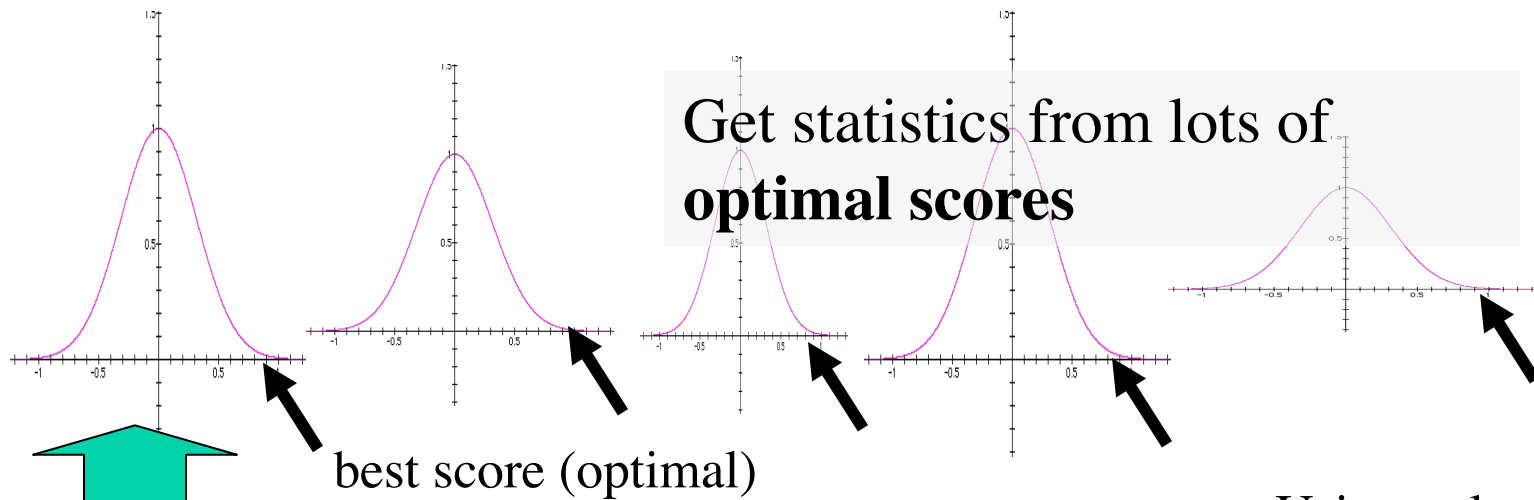
Usually, we suppose the likelihood of deviating from the mean by x in the positive direction is the same as the likelihood of deviating by x in the negative direction, and the likelihood of deviating by x decreases as the power of x .

Why? Because multiplying probabilities gives this type of curve.

This is called a Normal, or Gaussian distribution.



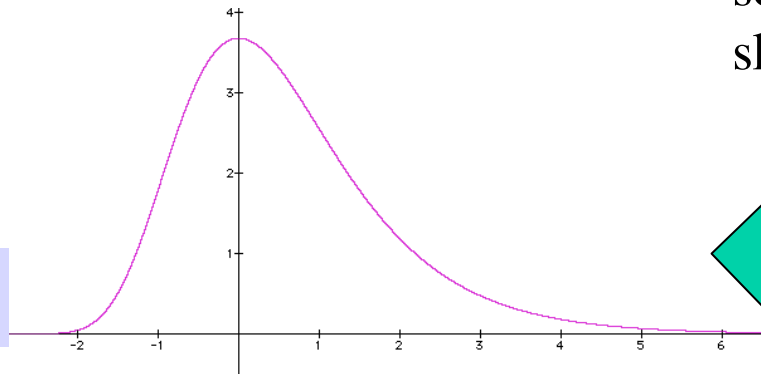
Extreme value distribution, a distribution derived from extreme values



all possible scores
for two sequences =
Normal distrib.

Extreme value distribution

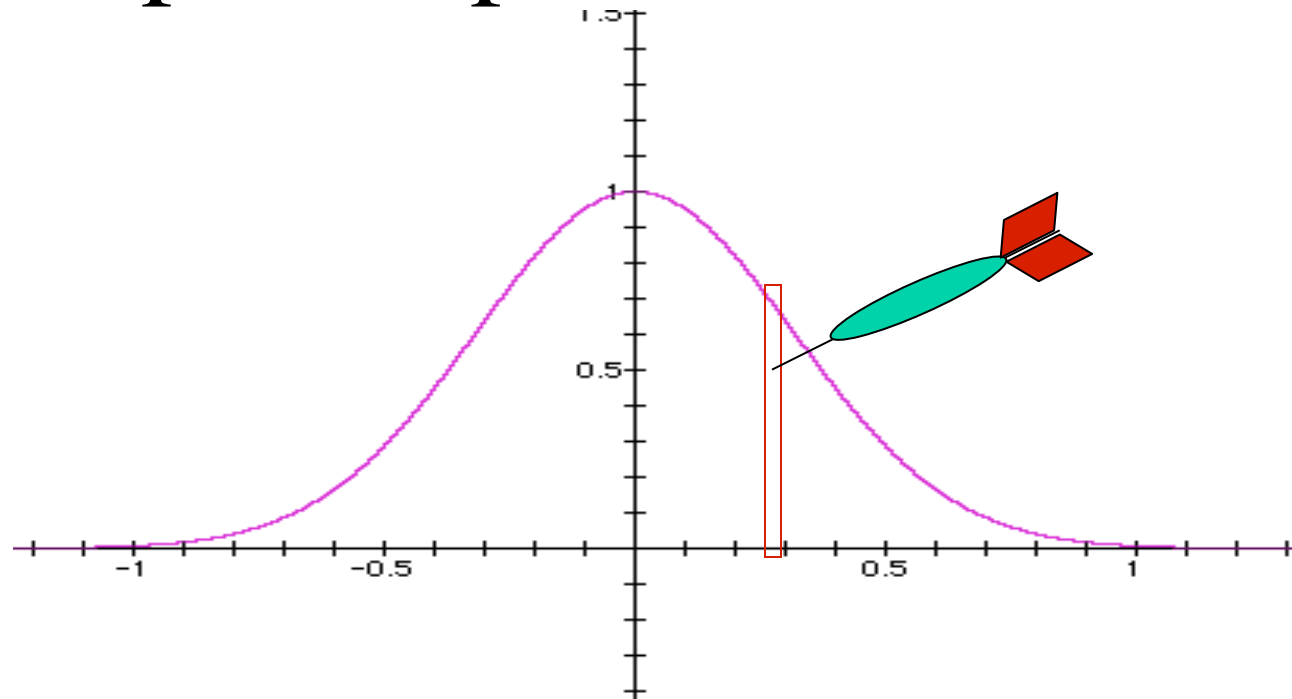
$$y = \exp(-x - e^{-x})$$



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

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

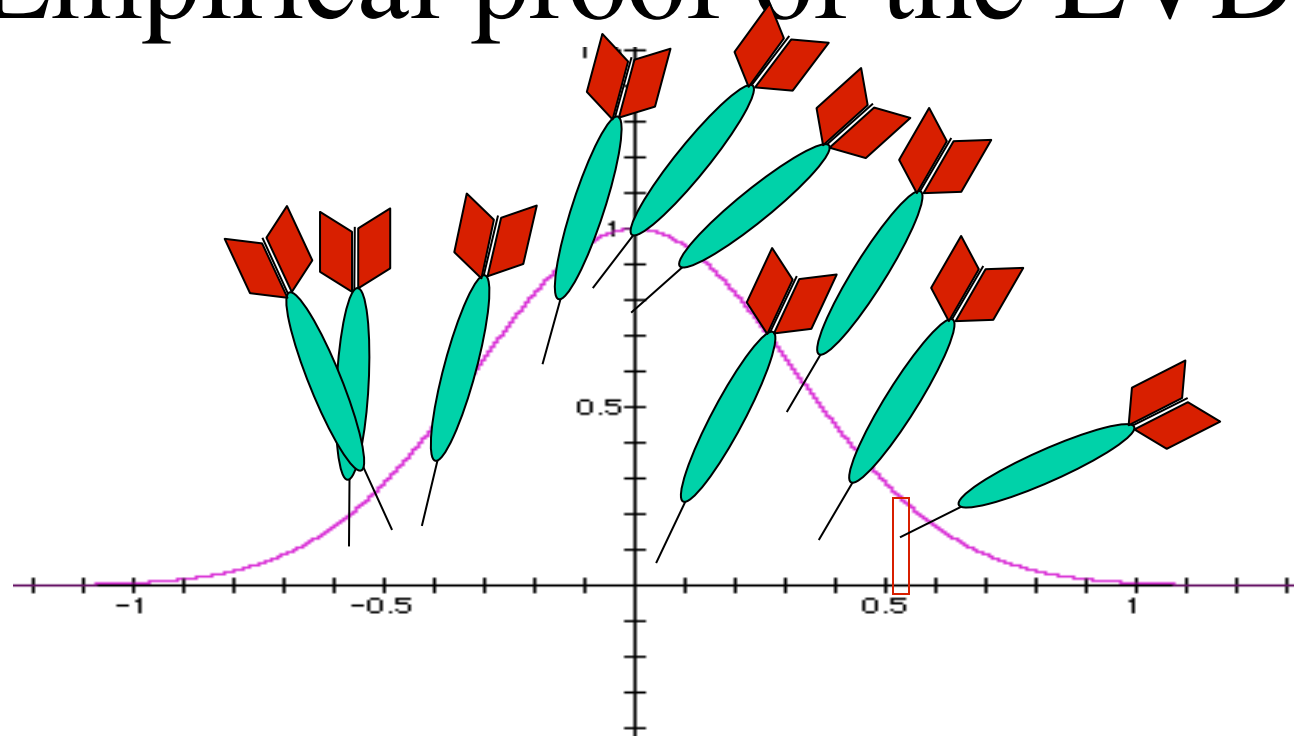
Empirical proof of the EVD



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?

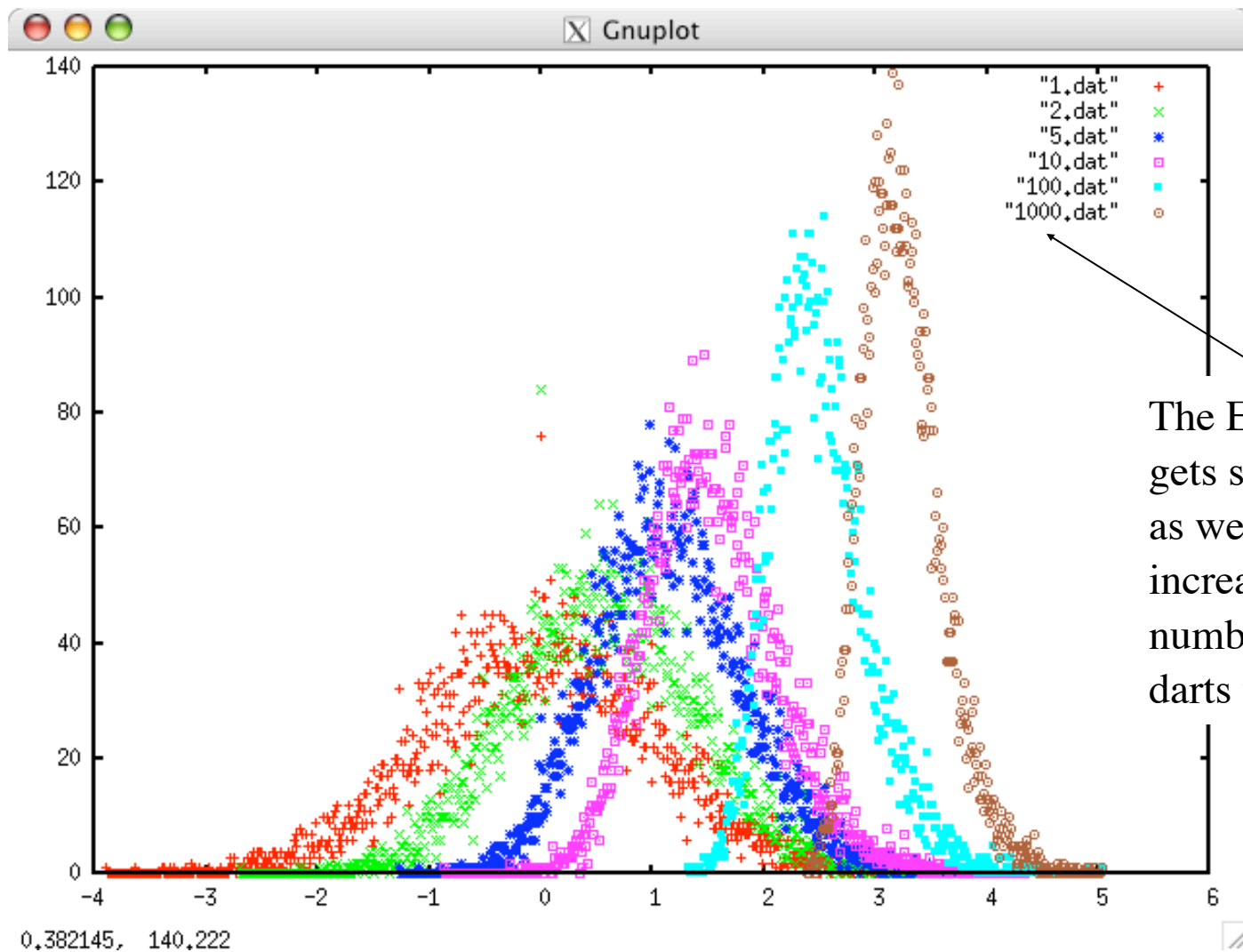
Answer: The same Gaussian distribution! (duh)

Empirical proof of the EVD

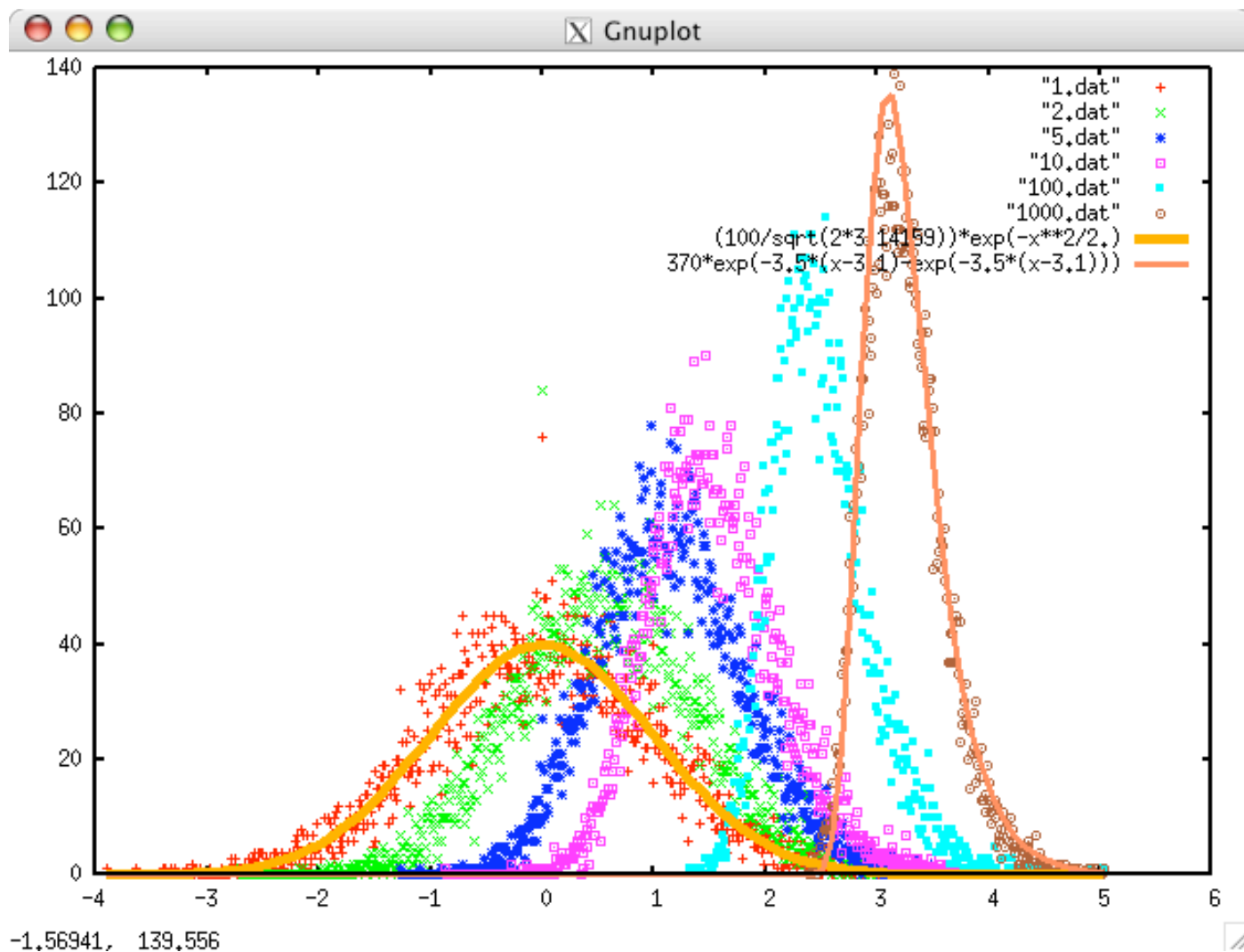


What if we throw 10 darts at a time and remove all but the highest-scoring dart. Do that 1000 times. What is the distribution of the scores?

Empirical proof of the EVD



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:

$$P(S \geq x) = 1 - \exp(-K m n e^{-\lambda x})$$

the scoring function and λ

λ is calculated as the value of x that satisfies:

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

Substitution matrix values.

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

voodoo mathematics

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

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

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

becomes,

$$P(S \geq x) \approx Kmn e^{-\lambda x}$$

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

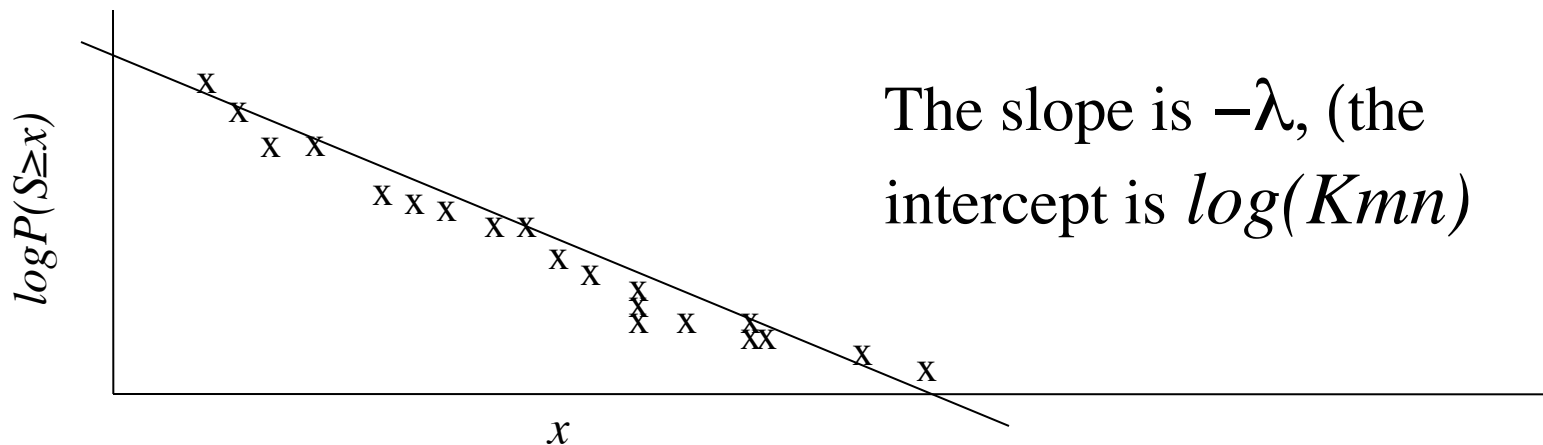
Estimated P(score x
or higher) given
random alignments:

$$P(S \geq x) \approx Kmn e^{-\lambda x}$$

Taking the log of
both sides,

$$\log(P(S \geq x)) \approx \log(Kmn) - \lambda x$$

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



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 .

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

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)

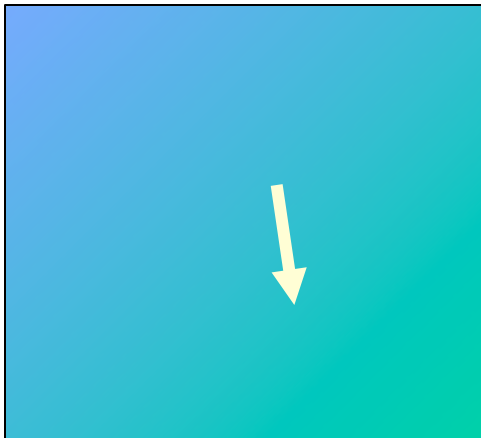
	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	
C	9																				C
S	-1	4																			
T	-1	1	5																		
P	-3	-1	-1	7																	
A	0	1	0	-1	4																
G	-3	0	-2	-2	0	6															
N	-3	1	0	-2	-2	0	6														
D	-3	0	-1	-1	-2	-1	1	6													
E	-4	0	-1	-1	-1	-2	0	2	5												
Q	-3	0	-1	-1	-1	-2	0	0	2	5											
H	-3	-1	-2	-2	-2	-2	1	-1	0	0	8										
R	-3	-1	-1	-2	-1	-2	0	-2	0	1	0	5									
K	-3	0	-1	-1	-1	-2	0	-1	1	1	-1	2	5								
M	-1	-1	-1	-2	-1	-3	-2	-3	-2	0	-2	-1	-1	5							
I	-1	-2	-1	-3	-1	-4	-3	-3	-3	-3	-3	-3	-3	1	4						
L	-1	-2	-1	-3	-1	-4	-3	-4	-3	-2	-3	-2	-2	2	2	4					
V	-1	-2	0	-2	0	-3	-3	-3	-2	-2	-3	-3	-2	1	3	1	4				
F	-2	-2	-2	-4	-2	-3	-3	-3	-3	-3	-1	-3	-3	0	0	0	-1	6			
Y	-2	-2	-2	-3	-2	-3	-2	-3	-2	-1	2	-2	-2	-1	-1	-1	-1	3	7		
W	-2	-3	-2	-4	-3	-2	-4	-4	-3	-2	-2	-3	-3	-1	-3	-2	-3	1	2	11	
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																				
S	0	2																			
T	-2	1	3																		
P	-3	1	0	6																	
A	-2	1	1	1	2																
G	-3	1	0	-1	1	3															
N	-4	1	0	-1	0	0	2														
D	-5	0	0	-1	0	1	2	4													
E	-5	0	0	-1	0	0	1	3	4												
Q	-5	-1	-1	0	0	-1	1	2	2	4											
H	-3	-1	-1	0	-1	-2	2	1	1	3	6										
R	-4	0	-1	0	-2	-3	0	-1	-1	1	2	6									
K	-5	0	0	-1	-1	-2	1	0	0	1	0	3	5								
M	-5	-2	-1	-2	-1	-3	-2	-3	-2	-1	-2	0	0	6							
I	-2	-1	0	-2	-1	-3	-2	-2	-2	-2	-2	-2	-2	2	5						
L	-6	-3	-2	-3	-2	-4	-3	-4	-3	-2	-2	-3	-3	4	2	6					
V	-2	-1	0	-1	0	-1	-2	-2	-2	-2	-2	-2	-2	2	4	2	4				
F	-4	-3	-3	-5	-4	-5	-4	-6	-5	-5	-2	-4	-5	0	1	2	-1	9			
Y	0	-3	-3	-5	-3	-5	-2	-4	-4	-4	0	-4	-4	-2	-1	-1	-2	7	10		
W	-8	-2	-5	-6	-6	-7	-4	-7	-7	-5	-3	2	-3	-4	-5	-2	-6	0	0	17	
C		S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	

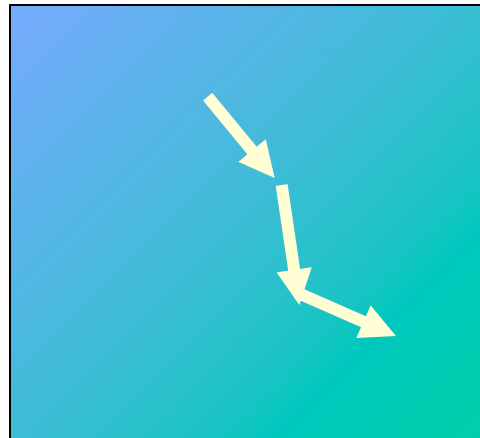
What happens with matrix bias*?

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

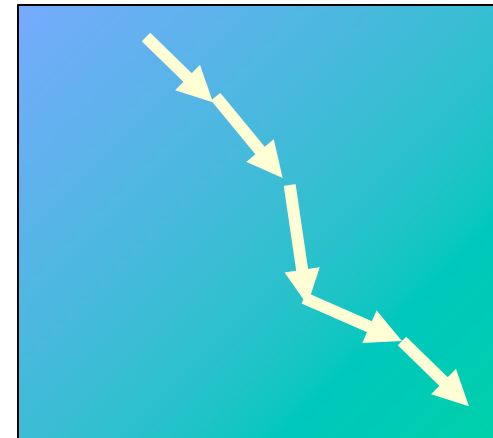
- Longer alignments are more common in random sets.
- Longer alignments are less significant.



Negative matrix bias



No matrix bias



Positive matrix bias

*aka: bonus

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: Gaussian 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.

$$P(S \geq x) = 1 - \exp(-K m n e^{-\lambda x}) \approx K m n e^{-\lambda x}$$

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 ?