PAM and BLAST

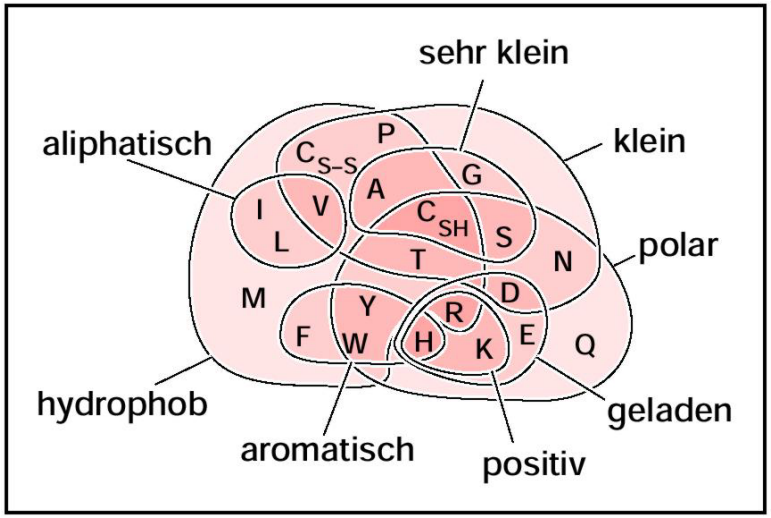
Ulf Leser

This Lecture

- Substitution Matrices
  - PAM distance
  - PAM matrices
- Scaling up Local Alignments
  - BLAST

Substitution Matrices

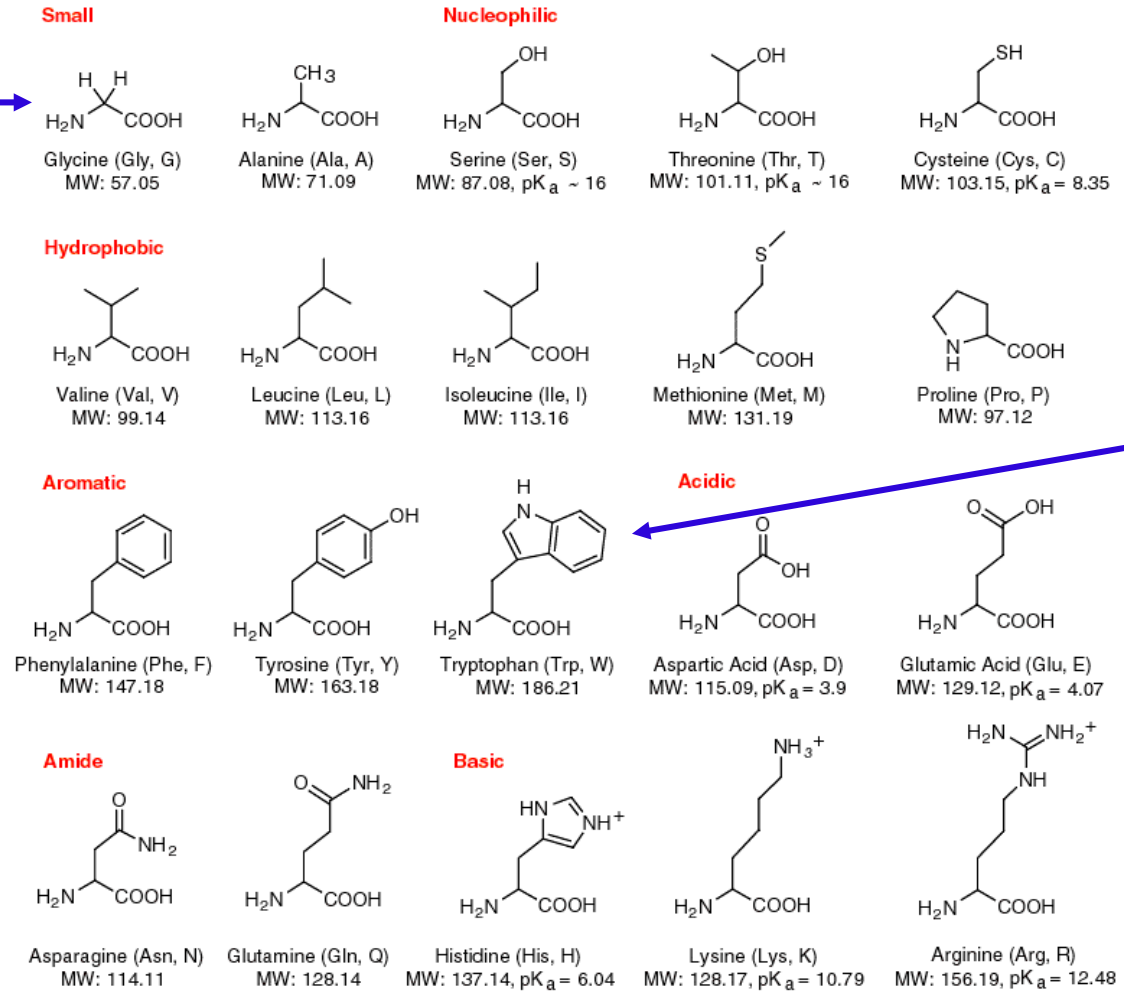
- Recall
  - A scoring function (substitution matrix) is a function s: $\Sigma' x \Sigma' \rightarrow$ int
- DNA: typically symmetric, simple matrices
- Protein sequences are different
  - Different AA have very different properties
  - Substitutions may change the 3D structure completely or just a little bit or not at all



Amino Acids

Klein, leicht →

← Groß, schwer



Example

Where do all these numbers come from?

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

Is it Really Necessary?

Code	Häufig-keit	Mutier-barkeit
L	0.091	54
A	0.077	100
G	0.074	50
S	0.069	117
V	0.066	98
E	0.062	77
K	0.059	72
T	0.059	107
I	0.053	103
D	0.052	86
P	0.051	58
R	0.051	83
N	0.043	104
Q	0.041	84
F	0.040	51
Y	0.032	50
M	0.024	93
H	0.023	91
C	0.020	44
W	0.014	25

- We count how often a particular AA was replaced by any other AA
  - Using "real" sequence alignments
- Replacement rate of Alanin (A) := 100%
- Obviously no equal distribution
- Even if we assume that mutations happen more or the less at the same rate, they obviously don't survive at the same rate
  - Mutations are suppressed to different degrees
  - W (Tryptophan): Strong suppression
  - S (Serin): Little suppression

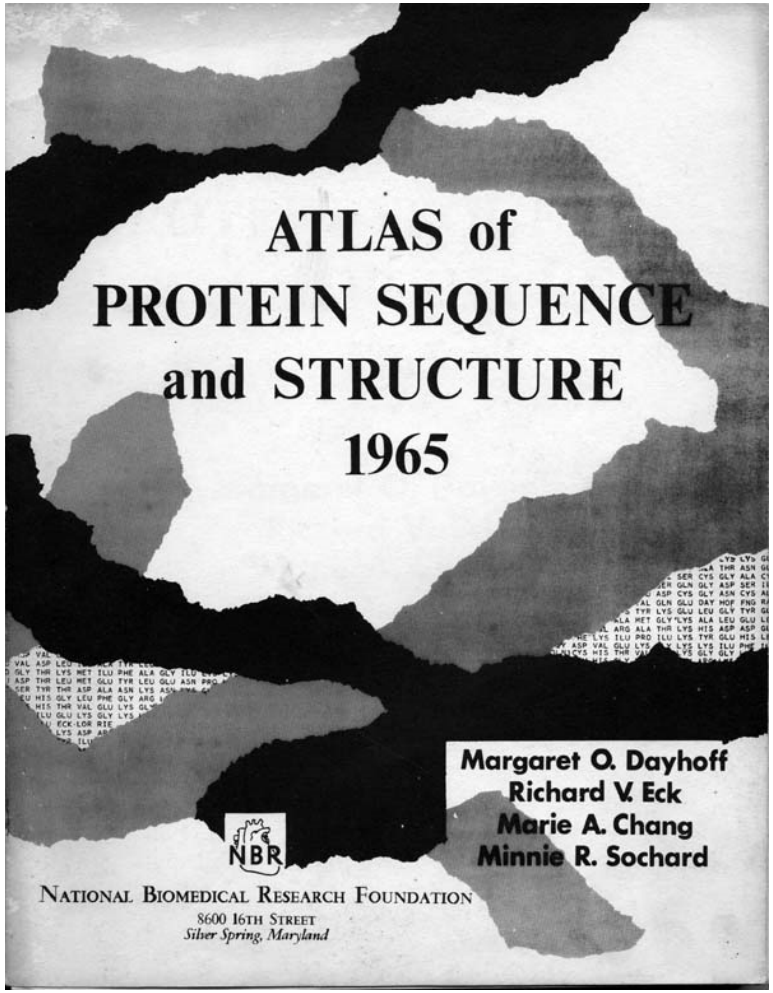
Quelle: Jones, Taylor, Thornton (1991). The rapid generation of mutation data matrices from protein sequences. CABIAS

Sensible Substitution Matrices

- For proteins, we need app. (20*20)/2=200 values
  - Scoring functions should be symmetric
- Possibility 1: Analytical
  - Capture weight, polarity, size, …
  - Find a scoring scheme to measure the difference between two AA
  - Needs to produce a single value per AA pair
  - Not used in practice
- Possibility 2: Empirical
  - Count which substitutions survived at which frequency in reality
  - Needs true alignments: Pairs of homologues and aligned sequences
  - Popular option: PAM – also considering evolutionary distance

Margaret O. Dayhoff

- Goal: "Deduce evolutionary relationships of the biological kingdoms, phyla, and other taxa from sequence evidence"
- Collection of all known protein sequences
  - First edition: 65 proteins
  - Several releases followed
  - Resulted in the Protein Information Resource (PIR)
  - … which was later merged with UniProt / SwissProt



Thanks to Antje Krause

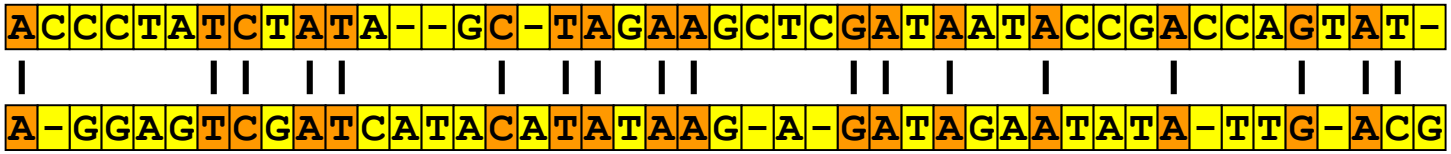
PAM: Point-Accepted Mutations

- Dayhoff, M. O., R. V. Eck, C. M. Park. (1972)
  A model of evolutionary change in proteins.
  in M. O. Dayhoff (ed.), Atlas of Protein Sequence and Structure Vol. 5.

- PAM has two meanings
  - 1 PAM – Unit for measuring the similarity of two AA sequences
  - PAM-X matrix – Substitution matrix to use when aligning two sequences that are X PAM distant
- Why having diff. scoring matrices for diff evol. distances?
  - Small distance – all chances work in isolation, only local effects
  - Larger distances – changes start interfering, combined effects
  - Note: The concrete distance is not important

Preliminaries

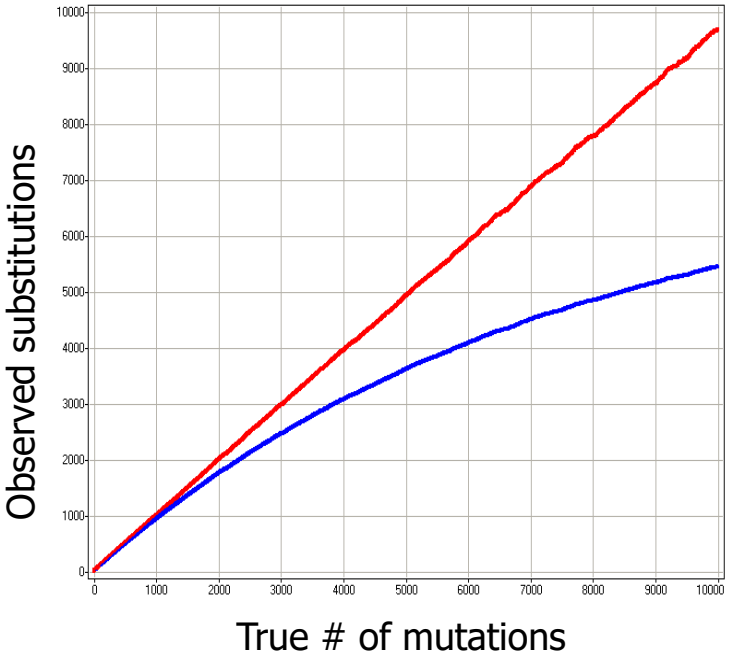
- We show alignments in columns



- The percentage of M-columns in the optimal alignment is the %-sequence-identity
  - In the example: 17/45 ~ 36%
  - Directly dependent on alignment score, but scaled to [0…1]
- %-sequence-diversity = 1 - %-sequence-identity

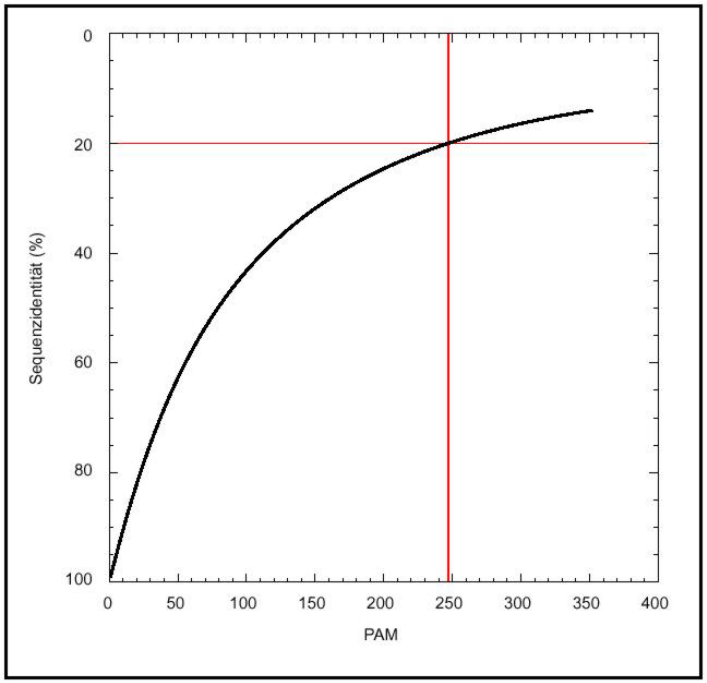
PAM as Distance Measure

- Definition
  Let *S_1, S_2 be two protein sequences with $|S_1|=|S_2|$. We say S_1 and S_2 are x PAM distant, iff S_1 most probably was produced from S_2 with x replacements per 100 AAs*
- Remarks
  - PAM is motivated by evolution
  - Assumptions: Mutations happen with the same rate at every position of a sequence
  - If mutation rate is high or time is long, mutations will occur at the same positions
  - PAM ≠ %-sequence-identity



PAM as Distance Measure

- No InDels, only replacements
- The PAM distance d of two DNA sequences can be derived analytically from their %-sequence-diversity p
  - d = -3/4*ln(1-4/3*p) (Jukes-Cantor model)
  - Derivation skipped
- Pairs with PAM >250 are probably not homologues
  - %-sequence-identity < 20%
  - Twilight zone
  - Which %-sequence-identity will two random protein sequences have?



PAM Matrices

- The PAM-X matrix contains measures for the probability that a AA (column) was replaced by another AA (row) in two sequences that are x PAM distant
- Estimated from data
  - Let $(S_{1,1}, S_{2,1})$, ..., $(S_{1,n}, S_{2,n})$ be n x-PAM distant pairs of aligned sequences (without InDels)
  - Compute f(i), the relative frequency of AA A_i in all pairs
  - Compute f(i,j), the relative substitution frequency of A_i and A_j
    - Number of positions k in any of the aligned pairs with $S_{1,z}$'[k]=A_i and $S_{2,z}$'[k]=A_j or vice versa
  - Then

$$M_x(i,j) = \log\left(\frac{f(i,j)}{f(i) * f(j)}\right)$$

Some Explanations

- Log-likelihood ratio combining
  - Observation: observed frequency of this mutation
  - Expectation: chances to generate this mutation by chance given the relative frequencies of the two involved AAs

$$M_x(i,j) = \log\left(\frac{f(i,j)}{f(i) * f(j)}\right)$$

- Interpretation
  - M(i,j) = 0: No selection
  - M(i,j) < 0: Negative selection, suppression of mutation
  - M(i,j) > 0: Positive selection, mutation is favored

Example

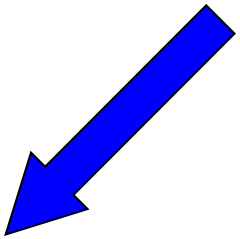
$S_{1,1}$: ACGTGAC
$S_{2,1}$: AGGTGCC

$S_{1,2}$: GTTAGTA
$S_{2,2}$: TTTAGTA

$S_{1,3}$: GGTCA
$S_{2,3}$: AGTCA

Relative frequencies

A: 10/38	C: 6/38	G: 11/38	T: 11/38



Mutation rates

	A	C	G	T
A	4/19	1/19	1/19	0/19
C		2/19	1/19	0/19
G			4/19	1/19
T				5/19

Matrix

	A	C	G	T
A	0,48	0,10	-0,16	-
C		0,63	0,06	-
G			0,40	-0,20
T				0,50

Problems

- Depends on predefined alignments
- We need a substitution matrix to find optimal alignments
  - A hen-egg problem
  - Alternative: Do it manually using experience, 3D-structure, ..
- Makes several assumptions
  - Mutations are equally likely at every position in a sequence
  - Mutations are equally likely independent from AA neighbors
  - …

Real Substitution Matrices

- PAM requires large n for each evolutionary distance X to adequately capture rare mutations
- Dirty trick: Molecular clock assumption
  - Assume that mutations appear with equal rate over time
  - Then the frequencies of PAM-x mutations depend on the frequencies of PAM-1 mutations
  - PAM-x matrices are computed by repeated matrix multiplication of PAM-1 with itself (assuming a linear relationship)
- The complete (highly heuristic) procedure
  - Choose set of n pairs with small PAM distance and align manually
  - Use these alignments to compute M_1
  - Compute $M_x = (M_1)^x$

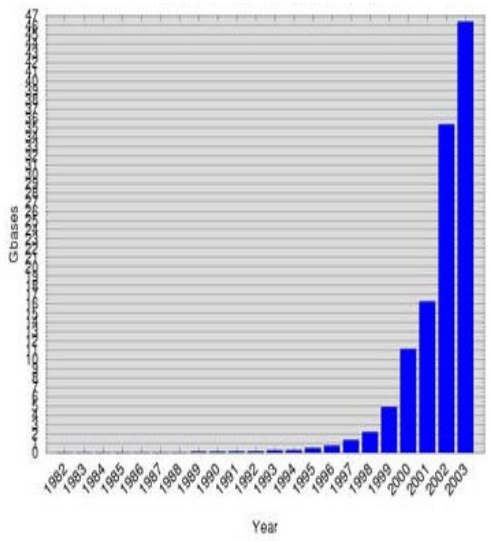
BLOSUM

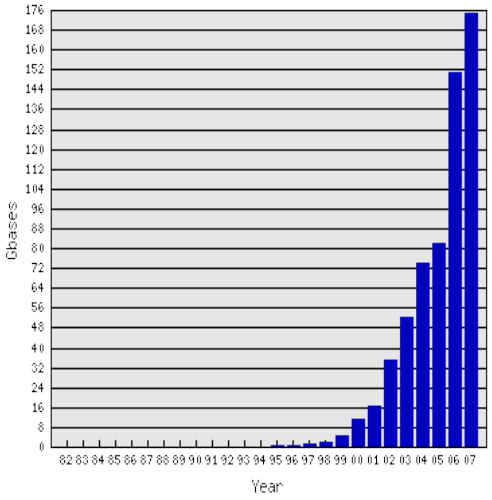
- PAM is a bit old-fashioned
- BLOSUM: BLOcks SUbstitution Matrix
  - Henikoff and Henikoff, 1993
  - Removes assumption of equal mutation rates across each sequence position by considering conserved blocks
  - Direct estimation for different PAM distances instead of error-propagating self multiplication

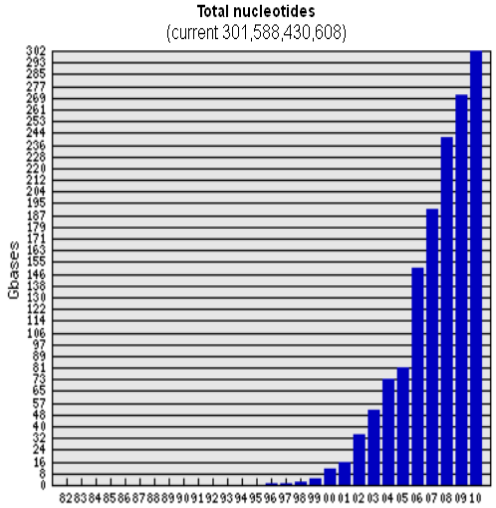
This Lecture

- Substitution Matrices
  - PAM distance
  - PAM matrices
- Scaling up Local Alignments
  - BLAST

Scaling Up Local Alignment







- Searching similar sequences (with a high local alignment score) is a fundamental operation in Bioinformatics
- Sequence databases grow exponentially
- We need faster algorithms, even if they sometimes fail

Similarity Search Problems and their Accuracy

- Task: Given a sequence s and a database D, find all sequences T in D that are sufficiently local-similar to s
  - Often, exactly computing T is not feasible and not necessary (think of the WWW and search engines)
- Assume a method that finds a set X of answers for s
- How good is this method?
  - Some sequences will be in X and T – true positives
  - Some will be in X but not T – false positives
    - Also called Type I error
  - Some will be in T but not X – false negatives
    - Also called Type II error
  - Some will be neither in X nor T – true negatives

Prediction \ Reality	+	-
+	TruePositive (TP)	FalsePositive (FP)
-	FalseNegative (FN)	TrueNegative (TN)

Precision and Recall

- Precision = TP/(TP+FP)
  - What is the fraction of correct answers in X?
  - Related to specificity

		Reality: +	Reality: -
Prediction	+	TruePositive (TP)	FalsePositive (FP)
	-	FalseNegative (FN)	TrueNegative (TN)

- Recall = TP/(TP+FN)
  - Which fraction of correct answers from T are also in X?
  - Also called sensitivity
- Trade-Offs
  - Usual methods compute a score per element of D
  - All sequences with a score above a threshold t are returned as X
  - Increasing t : higher precision, lower recall
  - Lowering t: lower precision, higher recall
  - … if the score correlates with correctness …

Example

- Let |DB| = 1000, |X|=15, |T|=20, |X∩T|=9

	Real: Positive	Real: Negative
Alg: Positive	TP = 9	FP = 6
Alg: Negative	FN = 11	TN= 974

  - Precision = TP/(TP+FP) = 9/15 = 60%
  - Recall = TP/(TP+FN) = 9/20 = 45%

- Assume we increase t: |X|=10, |X∩T|=7

	Real: Positive	Real: Negative
Alg: Positive	TP = 7	FP = 3
Alg: Negative	FN = 13	

  - Precision: 70%, recall = 35%

BLAST

- Altschul, Gish, Miller, Myers, Lipman: „Basic Local Alignment Search Tool", J Mol Bio, 1990
  - A heuristic algorithm for sequence similarity search
  - Very fast, high recall, not perfect
  - Very successful: You "blast" a sequence
  - NCBI runs thousands of BLAST searches every day
- A family of tools
  - Gapped-BLAST, PSI-BLAST, MegaBlast, BLAST-ALL, PATHBLAST, Name-BLAST, …
  - BLAST for DNA, protein, DNA-protein, protein-DNA, …
  - We only look at the simple DNA-DNA version
  - We skip several heuristic and domain-specific tricks

Fundamental Idea

- Fundamental idea : If two sequences have a good local alignment, then the matching area contains, with very high probability, a sub-area where the match is even better (or even exact)
- These sub-areas are called seeds

```
TTGACTCGATTATAGTCGCGGATATACTATCG
CCTATCACAAGAATATAGTCCCTGATCCAGC

   TTGACTC GATTATAGTCGCGGAT ATACTATCG
CCTATCACAA GAATATAGTCCCTGAT CCAGC

   TTGACTC GATTATAGTCGCGGAT ATACTATCG
CCTATCACAA GAATATAGTCCCTGAT CCAGC
```

Algorithm

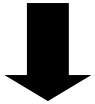
- Given query sequence s and sequence database D={d_i}
- 1. Compute all substrings s_i of s of length q
  - Also called q-grams
  - How many?
- 2. Find all approximate occurrences of all s_i in all d_j
  - Gap-free alignment with matrix; score must be above threshold t
  - Hits are called seeds – approx. occurrences of some s_i in d_j
- 3. Extend seeds to left and right in s_i and d_j until
  - [Constantly update the similarity score]
  - … the score drops sharply
  - … s_i or d_j ends
  - … the score gets too bad compared to other hits found earlier

Example

q=5, t= 3, Matrix: M=+1, R=-1
s=ACGTGATA
d=GATTGACGTGACTGCTAGTGATACTATAT

s_1=ACGTG
s_2=CGTGA
s_3=GTGAT
s_4=TGATA

GATTGACGTGACTGCTAGTGATACTATAT
GATTGACGTGACTGCTAGTGATACTATAT
GATTGACGTGACTGCTAGTGATACTATAT
GATTGACGTGACTGCTAGTGATACTATAT



GATTGACGTGACTGCAAGTGATACTATAT

ACGTGATA	5
ACGTGATA	5+1=6
ACGTGATA	6-1=5
...	...

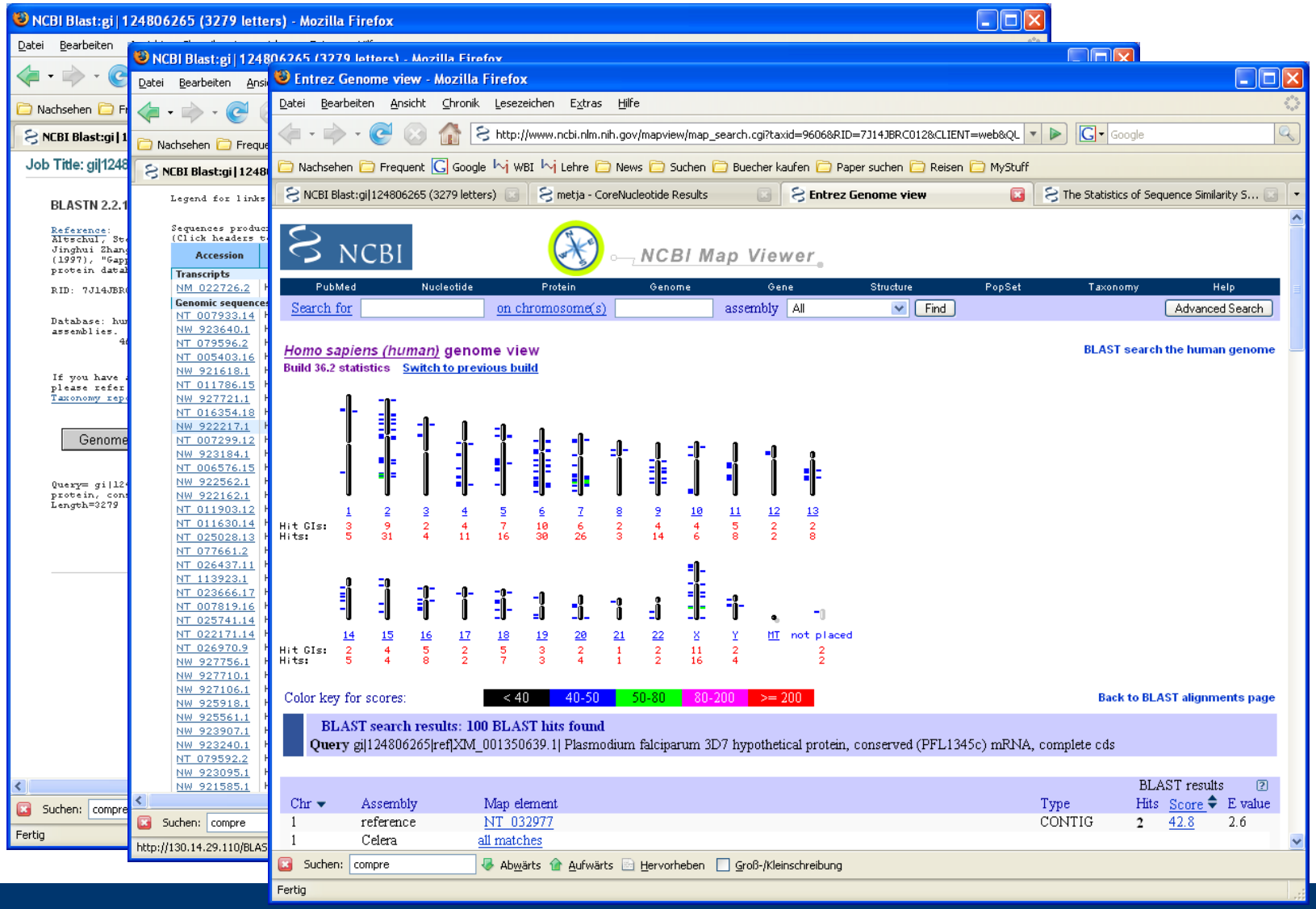
Properties

- Finding seeds efficiently requires more work
  - Pre-compute all q-grams of all d_i
  - Group by q-gram
  - Called a hash-index (should be kept in main memory)
  - Lookup: Given s, find all matching q-grams (as seeds)
- Exclusion method
  - Vast majority of all sequences in DB are never looked at because they do not contain a seed
  - The "seed" idea is the basis of nearly all fast alignment methods
- Where it fails
  - Sensitive to t: Too high – missing hits; too low – slow
  - Does not consider gaps

Speed – Precision - Recall

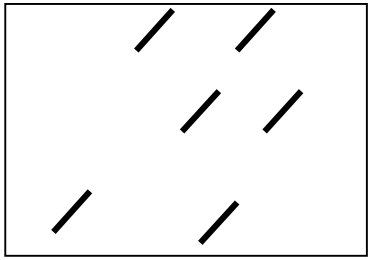
- Increasing t
  - Higher requirements for any seed
  - Less seeds, less extensions
  - Lower recall, higher speed, precision stays
- Increasing q (and adapting t)
  - Higher requirements for any seed
  - Less seeds, less extensions
  - Lower recall, higher speed, precision stays

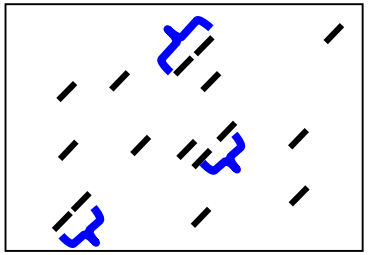
BLAST Screenshots

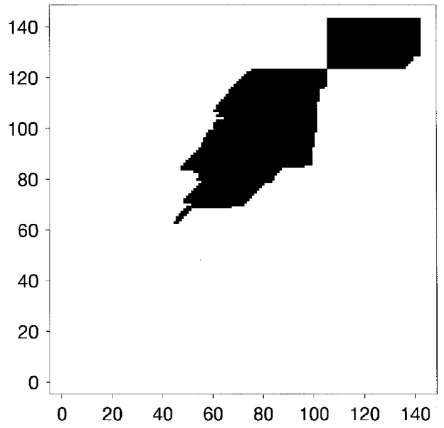


BLAST-2

- Altschul, Madden, Schaffer, Zhang, Zhang, Miller, Lipman: „Gapped BLAST and PSI-BLAST: a new generation of protein database search programs“, NAR, 1997
- Faster
  - BLAST: 90% of time spend in extensions
  - BLAST2: Two seeds in short distance
    - Needs a decrease in t
- Higher recall
  - BLAST didn't even consider gaps in the extension phase
  - BLAST2: Full local alignment starting from seeds
    - Allows an increase of t







Further Reading

- Substitution matrixes: Krane & Raymer, Chapter 3
- BLAST, BLAST2: Merkl & Waack, Chapter 12