# Bioinformatics Algorithms

## (Fundamental Algorithms, module 2)

**Zsuzsanna Lipták**

Masters in Medical Bioinformatics
academic year 2017/18, spring term

## Scoring Matrices

# More complex scoring functions

**Until now:**

- match, mismatch, gap (linear gap functions)
- match, mismatch, gap open, gap extend (affine gap functions)
- i.e. $f(a, b)$ depends only on $a = b$ or $a \neq b$

# More complex scoring functions

**Until now:**

- match, mismatch, gap (linear gap functions)
- match, mismatch, gap open, gap extend (affine gap functions)
- i.e. $f(a, b)$ depends only on $a = b$ or $a \neq b$

**But:**

- For protein sequences, better to differentiate between different pairs of AAs $a$ and $b$, i.e. depending on how close / how different they are.
- Reason: homologous proteins often have different AAs in same position. If only match/mismatch are evaluated, then many homologous proteins are not found.

# More complex scoring functions

**Until now:**

- match, mismatch, gap (linear gap functions)
- match, mismatch, gap open, gap extend (affine gap functions)
- i.e. $f(a, b)$ depends only on $a = b$ or $a \neq b$

**But:**

- For protein sequences, better to differentiate between different pairs of AAs $a$ and $b$, i.e. depending on how close / how different they are.
- Reason: homologous proteins often have different AAs in same position. If only match/mismatch are evaluated, then many homologous proteins are not found.

**So now:**

- $f(a, b)$ depends on $a$ and $b$
- necessarily: $f(a, b) = f(b, a)$ (symmetry)

# Scoring matrices

- Scoring matrix $S$ of dimension $20 \times 20$ (for protein), also possible: dim. $4 \times 4$ (for DNA)
- $S_{ab}$ gives the similarity of $a$ and $b$
- Similarity could be defined by
  1. similarity of codon (DNA-level), e.g. $\min\{dist_{Hamming}(xyz, uvw) : xyz \text{ codon for } a \text{ and } uvw \text{ codon for } b\}$
  2. physico-chemical properties (hydrophobicity, size, basic/acidic, ...)
  3. based on empirical data: How frequently do we observe this change?
- PAM matrices: Scoring matrices based on empirical data (Margret Dayhoff, 1978)
- PAM = Point Accepted Mutation (or: Percent Accepted Mutation)

**Basic idea:**

- $S_{ab} > 0$ : probability that $b$ has mutated into $a$ at this evolutionary distance is greater than chance
- $S_{ab} = 0$ : the two probabilities are equal (we cannot say anything)
- $S_{ab} < 0$ : probability that $b$ has been aligned to $a$ by chance is greater than the probability that this is a true mutation

**Basic idea:**

- $S_{ab} > 0$ : probability that *b* has mutated into *a* at this evolutionary distance is greater than chance
- $S_{ab} = 0$ : the two probabilities are equal (we cannot say anything)
- $S_{ab} < 0$ : probability that *b* has been aligned to *a* by chance is greater than the probability that this is a true mutation

**Meaning of "by chance":**

- We are comparing two probabilities
- prob1: that *a* and *b* are aligned together because there has been a series of mutations changing *b* into *a*
- prob2: that *a* and *b* have been aligned together by chance (e.g. if in the database all sequences consist only of *a*'s, then the probability that *a* is there in a random alignment is 1)

# PAM scoring matrices

- family of matrices: PAM$k$ (for any $k \geq 1$), common are PAM40, PAM120, PAM250
- PAM$k$: $k$ is the evolutionary distance between the sequences to be scored; needs to be guessed *before* scoring
- higher $k$: applied to more distant / less closely related sequences / species
- the scoring matrix PAM$k$ is not a probability matrix
- it is based on a probability matrix

# Mutation probability matrix

- Dayhoff et al. generated mutation probability matrix $M$ (PAM1 mutation matrix) based on empirical data: a large set of aligned sequences which are known to be homologous (trusted alignments)
- $M_{ab}$ = probability that AA $b$ will change into AA $a$ in one time step[1]
- this probability is only estimated, based on observed data
- one time step = 1 PAM unit evolutionary distance = 1 mutation every 100 AAs on average
- sum over each column = 1: $\sum_{a \in \Sigma} M_{ab} = 1$ ($M$ is the transpose of a probability transition matrix)

[1] a bit unusual that they put the original AA in the columns and not in the rows, as is common in probability transition matrices

# Mutation probability at higher distances: $M^k$

- How about the probability that $b$ changes into $a$ in 2 steps?

[2] and not the real number $M_{ab}$ squared!

# Mutation probability at higher distances: $M^k$

- How about the probability that $b$ changes into $a$ in 2 steps?
- possibilities are:

| time step 1 | time step 2 |
|---|---|
| $b \rightarrow a$ | $a$ unchanged |
| $b$ unchanged | $b \rightarrow a$ |
| $c \neq a, b$: $b \rightarrow c$ | $c \rightarrow a$ |

[2] and not the real number $M_{ab}$ squared!

# Mutation probability at higher distances: $M^k$

- How about the probability that $b$ changes into $a$ in 2 steps?
- possibilities are:

| time step 1 | time step 2 |
|---|---|
| $b \rightarrow a$ | $a$ unchanged |
| $b$ unchanged | $b \rightarrow a$ |
| $c \neq a, b$: $b \rightarrow c$ | $c \rightarrow a$ |

- Prob($b$ changes into $a$ in 2 steps)
  $= M_{ab} \cdot M_{aa} + M_{bb} \cdot M_{ab} + \sum_{c \neq a,b} M_{cb} M_{ac} = \sum_{c \in \Sigma} M_{ac} M_{cb} = M^2_{ab}$

[2] and not the real number $M_{ab}$ squared!

# Mutation probability at higher distances: $M^k$

- How about the probability that $b$ changes into $a$ in 2 steps?
- possibilities are:

| time step 1 | time step 2 |
|---|---|
| $b \to a$ | $a$ unchanged |
| $b$ unchanged | $b \to a$ |
| $c \neq a, b$: $b \to c$ | $c \to a$ |

- Prob($b$ changes into $a$ in 2 steps) $= M_{ab} \cdot M_{aa} + M_{bb} \cdot M_{ab} + \sum_{c \neq a,b} M_{cb} M_{ac} = \sum_{c \in \Sigma} M_{ac} M_{cb} = M^2_{ab}$
- $M^2_{ab}$ is just the entry $a, b$, i.e. row $a$ and column $b$, of the product matrix $M^2 = M \cdot M$ (matrix multiplication)[2]

[2] and not the real number $M_{ab}$ squared!

# Mutation probability at higher distances: $M^k$

- How about the probability that $b$ changes into $a$ in 2 steps?
- possibilities are:

| time step 1 | time step 2 |
|---|---|
| $b \to a$ | $a$ unchanged |
| $b$ unchanged | $b \to a$ |
| $c \neq a, b$: $b \to c$ | $c \to a$ |

- Prob($b$ changes into $a$ in 2 steps) $= M_{ab} \cdot M_{aa} + M_{bb} \cdot M_{ab} + \sum_{c \neq a,b} M_{cb} M_{ac} = \sum_{c \in \Sigma} M_{ac} M_{cb} = M^2_{ab}$
- $M^2_{ab}$ is just the entry $a, b$, i.e. row $a$ and column $b$, of the product matrix $M^2 = M \cdot M$ (matrix multiplication)[2]
- in general: $M^k$ contains the probabilities for $k$ steps, i.e. $M^k_{ab} =$ prob. that $b$ has mutated into $a$ after $k$ steps

[2] and not the real number $M_{ab}$ squared!

# Computation of the scoring matrices

- the PAM scoring matrices are "log-odds" matrices
  - odds: compare two probabilities
  - log: take the logarithm (product → sum)

# Computation of the scoring matrices

- the PAM scoring matrices are "log-odds" matrices
  - odds: compare two probabilities
  - log: take the logarithm (product → sum)
- PAM$k$ scoring matrix:
  - take $M^k$
  - ${M^k}_{ab} = \text{Prob}(b \text{ changed into } a \text{ in } k \text{ steps})$
  - compare to: $\text{Prob}(a \text{ is there by chance}) = p_a$
    $p_a$ = relative frequency of $a$,
    e.g. if the DB is: $\{aabc, abca\}$, then $p_a = 1/2, p_b, p_c = 1/4$

# Computation of the scoring matrices

- the PAM scoring matrices are "log-odds" matrices
  - odds: compare two probabilities
  - log: take the logarithm (product → sum)
- PAM$k$ scoring matrix:
  - take $M^k$
  - $M^k_{ab} = \text{Prob}(b \text{ changed into } a \text{ in } k \text{ steps})$
  - compare to: $\text{Prob}(a \text{ is there by chance}) = p_a$
    $p_a =$ relative frequency of $a$,
    e.g. if the DB is: $\{aabc, abca\}$, then $p_a = 1/2, p_b, p_c = 1/4$
- take log (base 10), multiply by 10 (for nicer numbers), round to nearest integer:

$$S_{ab} = 10 \cdot \log_{10}(\frac{M^k_{ab}}{p_a}) \qquad \text{rounded to nearest int.}$$

# Computation of the scoring matrices

$$\boxed{S_{ab} = 10 \cdot \log_{10}(\frac{M^{k}_{ab}}{p_a})}$$

$$\frac{M^{k}_{ab}}{p_a} \qquad \begin{cases} > 1 & \text{if} \\ & \\ & \end{cases}$$

# Computation of the scoring matrices

$$\boxed{S_{ab} = 10 \cdot \log_{10}(\tfrac{M^{k}{}_{ab}}{p_a})}$$

$$\frac{M^{k}{}_{ab}}{p_a} \qquad \begin{cases} > 1 & \text{if } M^{k}{}_{ab} > p_a \\ = 1 & \text{if} \\ & \end{cases}$$

# Computation of the scoring matrices

$$\boxed{S_{ab} = 10 \cdot \log_{10}(\frac{M^k{}_{ab}}{p_a})}$$

$$\frac{M^k{}_{ab}}{p_a} \qquad \begin{cases} > 1 & \text{if } M^k{}_{ab} > p_a \\ = 1 & \text{if } M^k{}_{ab} = p_a \\ < 1 & \text{if } \end{cases}$$

# Computation of the scoring matrices

$$\boxed{S_{ab} = 10 \cdot \log_{10}(\tfrac{M^k{}_{ab}}{p_a})}$$

$$\frac{M^k{}_{ab}}{p_a} \qquad \begin{cases} > 1 & \text{if } M^k{}_{ab} > p_a \\ = 1 & \text{if } M^k{}_{ab} = p_a \\ < 1 & \text{if } M^k{}_{ab} < p_a \end{cases}$$

# Computation of the scoring matrices

$$\boxed{S_{ab} = 10 \cdot \log_{10}(\frac{M^k_{ab}}{p_a})}$$

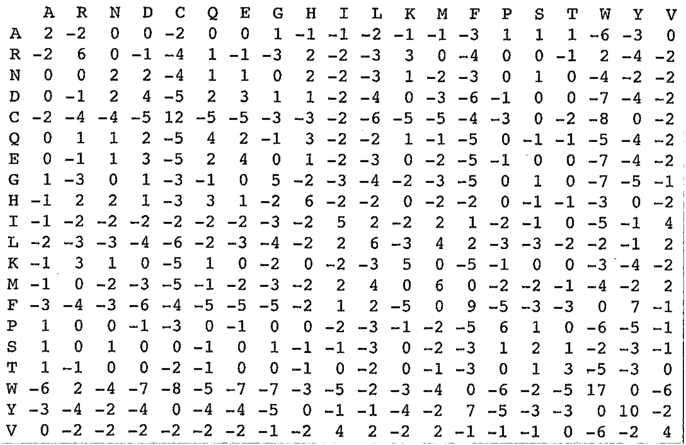
$$\frac{M^k_{ab}}{p_a} \qquad \begin{cases} > 1 & \text{if } M^k_{ab} > p_a \\ = 1 & \text{if } M^k_{ab} = p_a \\ < 1 & \text{if } M^k_{ab} < p_a \end{cases}$$

Therefore

$$S_{ab} \qquad \begin{cases} > 0 & \text{if } M^k_{ab} > p_a & \text{i.e. if prob1 is greater than prob2} \\ = 0 & \text{if } M^k_{ab} = p_a & \text{i.e. if they are equal} \\ < 0 & \text{if } M^k_{ab} < p_a & \text{i.e. if prob2 is greater than prob1} \end{cases}$$

Note that scoring matrices are symmetrical (but not the prob. matrices).

# PAM 250 Matrix



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

# Why use logarithm?

We use logarithms for computational reasons:

- since log is strictly monotonically increasing, one can replace all $x$ with $\log x$
- products of probs → sums of log-of-probs
- easier to compute sums than products of very small numbers (note that all probabilities are between 0 and 1): reduce rounding errors

# Two caveats

PAM matrices use two silent assumptions:

- mutations (changes) of AAs happen independently (i.e. independent of context): scoring by individual columns
- uses an evolutionary model: $k$ distance $=$ $k$ identical steps (i.e. with same probabilites)

# BLOSUM matrices

BLOSUM scoring matrices (Henikoff and Henikoff, 1992)

- other family of commonly used scoring matrices
- remedies second issue: uses no underlying evolutionary model
- same principle as PAM matrices, but:
- used different sets of aligned sequences for different distances
- BLOSUM $m$ = only used sequences that had $m$% identity or less
- higher number = closer related
- common: BLOSUM 45, 62, 80; BLOSUM62 $\sim$ PAM120

# Summary

**PAM matrices**

- allow scoring different AA pairs according to evolutionary relatedness
- different PAM$k$ acc. to evolutionary distance
- all modern AA scoring matrices are based on empirical data: observed frequencies in trusted alignment data
- the probabilities are estimated probabilites of AAs (from the data)
- mutation probability matrix $M$ (1 step = 1 PAM unit)
  $\rightsquigarrow$ $M^k$ mutation probability matrix for $k$ steps ($k$ PAM units)
  $\rightsquigarrow$ PAM$k$ scoring matrix $S$ (log-odds matrix)
- higher number = less related = more distant
- commonly used: PAM40, PAM120, PAM160, PAM250
- $k$ in PAM$k$ needs to be decided before scoring
- BLOSUM: similar to PAM but higher number = more related