CSI5126. Algorithms in bioinformatics

Substitution Score

Marcel Turcotte

uOttawa

School of Electrical Engineering and Computer Science (EECS)
University of Ottawa

Version October 2, 2018

Summary

In this lecture, we consider **probabilistic models** for biological sequences. First, we review at a very high level approaches to determine if a given sequence alignment is statistically significant. Next, we look at simple models for one **biological sequence**, as well as a pairwise alignment. Finally, we introduce the concept of **Markov chain** and its application to derive a **substitution score**.

General objective

- Explain in your own terms the probabilistic models for biological sequences.

Reading

- Warren J. Ewens, Gregory R. Grant (2001) *Statistical Methods in Bioinformatics: An Introduction*. Springer. Pages: 238-249.

What is a **significant** score?

One approach consist in generating random sequences. (say 100 or more)

- Monte Carlo
- Shuffling
- (Or by simply reading sequences backwards)

and computing the optimal score for the alignment of those random sequences. **Assuming** the distribution of the scores follows a **normal distribution**, a simple test such as the **Z score**, would allow to distinguish the alignments of homologues from those of random pairs:

$$Z = (x - \mu)/\sigma$$

Empirical studies suggest that a Z score greater than 6 (3 standard deviations) is significant for the comparison of biological sequences.

Remarks

- Here, using actual (randomized) sequences ensures that the **frequency of the amino acids** is 1) biological and 2) comparable to the sequences under studies. It is also important that the randomized sequences being of approximately the same length as the sequences to be tested.

Remarks (continued)

- Very little is known about the distribution of global alignments scores. In particular, one cannot assume a normal distribution.
- Much more is known about the distribution of local alignment scores. For the case of ungapped local alignment it has been shown that the scores follows an **extreme value distribution** (EVD). Computational experiments suggests that gapped local alignments also follow an EVD.
- Based on EVD, it's possible to calculate what is called an **E value**, which depends on the score, the size of the query, as well as the size of the database.

Remarks (continued)

- "[An E-value] represents the number of distinct alignments with equivalent or superior score that might have been expected to have occurred purely by chance" Altschul 1998.
- An E-value of 10 is not statistically significant, whereas an E value of 10^{-5} is.

Probabilistic Framework

- Recall that a sequence alignment should answer the question: “**are the two sequences (evolutionary) related**?”

Probabilistic Framework

- Recall that a sequence alignment should answer the question: "**are the two sequences (evolutionary) related**?"
- In other words, is the observed sequence alignment the result of:

Probabilistic Framework

- Recall that a sequence alignment should answer the question: "**are the two sequences (evolutionary) related**?"
- In other words, is the observed sequence alignment the result of:
  1. an evolutionary process, where both sequences have evolved independtly from a **common ancestry**, or

Probabilistic Framework

- Recall that a sequence alignment should answer the question: "**are the two sequences (evolutionary) related**?"
- In other words, is the observed sequence alignment the result of:
  1. an evolutionary process, where both sequences have evolved independtly from a **common ancestry**, or
  2. can it be attributable to chance alone; randomly selecting two **unrelated sequences** could produce a similar alignment score.

Protein sequence probabilities

It's useful to consider a **simple** probabilistic model of a protein sequence, **given p_a, the probability of observing the amino acid** a, such that,

$$p_a > 0$$

$$\sum_{a=1}^{20} p_a = 1$$

Let's define the **probability of a sequence** $S(1)S(2)\ldots S(n)$ as,

$$p_{S(1)}p_{S(2)}\cdots p_{S(n)} = \prod_{i=1}^{n} p_{S(i)}$$

Remarks

- This model is **simple** in the sense that it **assumes that all proteins are n residues long**.
    - A more realistic models should account for **all possible lengths** and the **sum over all possible sequences** should be **1**.

Amino acids probabilities

A common practice consists of estimating the amino acid probabilities using the **observed frequencies** in a large database.

```
> GetAaFrequency(DB);
             Alanine  7.62 %
            Arginine  5.19 %
          Asparagine  4.40 %
       Aspartic acid  5.27 %
            Cysteine  1.64 %
           Glutamine  3.94 %
       Glutamic acid  6.40 %
             Glycine  6.87 %
           Histidine  2.24 %
          Isoleucine  5.84 %
             Leucine  9.47 %
              Lysine  5.96 %
          Methionine  2.38 %
       Phenylalanine  4.10 %
             Proline  4.91 %
              Serine  7.09 %
           Threonine  5.64 %
          Tryptophan  1.23 %
            Tyrosine  3.18 %
              Valine  6.62 %
```

Here are the amino acid frequencies observed for the database **Swiss-Prot version 39**.

Probabilistic Interpretation of a **Sequence Alignment**

Consider two **aligned** sequences, S_1 and S_2. For simplicity, ungaped alignments are considered.

$$\begin{array}{cccc} S_1(1) & S_1(2) & \ldots & S_1(n) \\ S_2(1) & S_2(2) & \ldots & S_2(n) \end{array}$$

The interpretation requires weighting **two outcomes**.

Probabilistic Interpretation of a **Sequence Alignment**

Consider two **aligned** sequences, S_1 and S_2. For simplicity, ungaped alignments are considered.

$$\begin{array}{cccc} S_1(1) & S_1(2) & \ldots & S_1(n) \\ S_2(1) & S_2(2) & \ldots & S_2(n) \end{array}$$

The interpretation requires weighting **two outcomes**.

1. Sequences are **related (Match** Model – M)

Probabilistic Interpretation of a **Sequence Alignment**

Consider two **aligned** sequences, S_1 and S_2. For simplicity, ungaped alignments are considered.

$$\begin{array}{llll} S_1(1) & S_1(2) & \ldots & S_1(n) \\ S_2(1) & S_2(2) & \ldots & S_2(n) \end{array}$$

The interpretation requires weighting **two outcomes**.

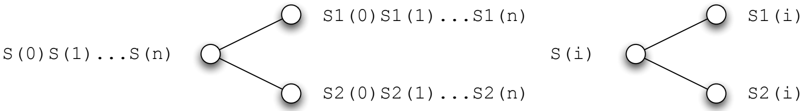
1. Sequences are **related (Match** Model – M)
2. Sequences are **unrelated** (**Random** Model – R)

Match model

In the **match model**, we have,

$$P(S_1, S_2 | M) = \prod_i q(S_1(i), S_2(i))$$

where $q(a, b)$ represents the probability that both residues a and b have both been **derived independently from an ancestral residue** c.



Random model

Whilst the **random model** is simply,

$$P(S_1, S_2|R) = \prod_i p_{S_1(i)} \prod_j p_{S_2(j)}$$

but since we assumed that $|S_1| = |S_2|$,

$$P(S_1, S_2|R) = \prod_i p_{S_1(i)} p_{S_2(i)}$$

The ratio of the two **likelihoods** is called an **odds-ratio** (or **likelihood-ratio**),

$$\frac{P(S_1, S_2|M)}{P(S_1, S_2|R)} = \prod_i \frac{q(S_1(i), S_2(i))}{p_{S_1(i)} p_{S_2(i)}}$$

The ratio of the two **likelihoods** is called an **odds-ratio** (or **likelihood-ratio**),

$$\frac{P(S_1, S_2|M)}{P(S_1, S_2|R)} = \prod_i \frac{q(S_1(i), S_2(i))}{p_{S_1(i)} p_{S_2(i)}}$$

taking the **logarithm** leads to a quantity known as the **log-odds ratio**,

$$S(S_1, S_2) = \sum_i \log(\frac{q(S_1(i), S_2(i))}{p_{S_1(i)} p_{S_2(i)}})$$

where each,

$$s(a, b) = \log(\frac{q(a, b)}{p_a p_b})$$

represents the log-likelihood ratio that the residue pair (a, b) will occur as an aligned pair, as opposed to unaligned.

The ratio of the two **likelihoods** is called an **odds-ratio** (or **likelihood-ratio**),

$$\frac{P(S_1, S_2|M)}{P(S_1, S_2|R)} = \prod_i \frac{q(S_1(i), S_2(i))}{p_{S_1(i)} p_{S_2(i)}}$$

taking the **logarithm** leads to a quantity known as the **log-odds ratio**,

$$S(S_1, S_2) = \sum_i \log(\frac{q(S_1(i), S_2(i))}{p_{S_1(i)} p_{S_2(i)}})$$

where each,

$$s(a, b) = \log(\frac{q(a, b)}{p_a p_b})$$

represents the log-likelihood ratio that the residue pair (a, b) will occur as an aligned pair, as opposed to unaligned.
In the case of proteins $s(a, b)$ represents a 20×20 matrix, known as **score matrix** or **substitution matrix**.

- In this view, the **total score** of alignment is the sum of all the terms for the aligned pairs of residues and gaps.

- In this view, the **total score** of alignment is the sum of all the terms for the aligned pairs of residues and gaps.
- The score is interpreted as the logarithm of the relative likelihood that the sequences are **related** vs **not related**.

- In this view, the **total score** of alignment is the sum of all the terms for the aligned pairs of residues and gaps.
- The score is interpreted as the logarithm of the relative likelihood that the sequences are **related** vs **not related**.
- **Positive terms** represent substitutions are more **likely** than would be expected by chance.

- In this view, the **total score** of alignment is the sum of all the terms for the aligned pairs of residues and gaps.
- The score is interpreted as the logarithm of the relative likelihood that the sequences are **related** vs **not related**.
- **Positive terms** represent substitutions are more **likely** than would be expected by chance.
- **Negative terms** represent **unfavorable** substitutions.

- In this view, the **total score** of alignment is the sum of all the terms for the aligned pairs of residues and gaps.
- The score is interpreted as the logarithm of the relative likelihood that the sequences are **related** vs **not related**.
- **Positive terms** represent substitutions are more **likely** than would be expected by chance.
- **Negative terms** represent **unfavorable** substitutions.
- Finally, when the two hypotheses are **equally likely** the log-likelihood ratio will be **zero**.

- In this view, the **total score** of alignment is the sum of all the terms for the aligned pairs of residues and gaps.
- The score is interpreted as the logarithm of the relative likelihood that the sequences are **related** vs **not related**.
- **Positive terms** represent substitutions are more **likely** than would be expected by chance.
- **Negative terms** represent **unfavorable** substitutions.
- Finally, when the two hypotheses are **equally likely** the log-likelihood ratio will be **zero**.
- We see that such substitution matrix can be used for calculating **local sequence alignments**, since likely alignments will have a positive score and unlikely alignment will have a negative score.

- In this view, the **total score** of alignment is the sum of all the terms for the aligned pairs of residues and gaps.
- The score is interpreted as the logarithm of the relative likelihood that the sequences are **related** vs **not related**.
- **Positive terms** represent substitutions are more **likely** than would be expected by chance.
- **Negative terms** represent **unfavorable** substitutions.
- Finally, when the two hypotheses are **equally likely** the log-likelihood ratio will be **zero**.
- We see that such substitution matrix can be used for calculating **local sequence alignments**, since likely alignments will have a positive score and unlikely alignment will have a negative score.
- **Additive scoring scheme** means that **positions** along the sequence are considered **independent** from one another, i.e. mutations at different sites have occurred independently. It's a **working hypothesis**.

What about the **Substitution Scores**?

The substitution scores that we used were rather **arbitrary**, either the identity matrix or some hand made matrix.

What about the **Substitution Scores**?

The substitution scores that we used were rather **arbitrary**, either the identity matrix or some hand made matrix.

Let's have a look at scoring schemes that are appropriate for **protein sequences**.

What about the **Substitution Scores**?

The substitution scores that we used were rather **arbitrary**, either the identity matrix or some hand made matrix.

Let’s have a look at scoring schemes that are appropriate for **protein sequences**.

- Certain **amino acids** have similar properties (structure, volume, charge, hydrophobicity, etc.)

What about the **Substitution Scores**?

The substitution scores that we used were rather **arbitrary**, either the identity matrix or some hand made matrix.

Let's have a look at scoring schemes that are appropriate for **protein sequences**.

- Certain **amino acids** have similar properties (structure, volume, charge, hydrophobicity, etc.)
- Looking at the **genetic code**, you can see that certain pairs of amino acids are such that the minimum number of mutations at the codon level to change the encoding from one amino acid type to another is only one (Ala and Asp, GCC and GAC), there are pairs that need a minimum of two mutations (Ala and Arg, CGA and GCA) or even three (Asn and Trp, AAC or AAU and UGG).

What about the **Substitution Scores**?

The substitution scores that we used were rather **arbitrary**, either the identity matrix or some hand made matrix.

Let's have a look at scoring schemes that are appropriate for **protein sequences**.

- Certain **amino acids** have similar properties (structure, volume, charge, hydrophobicity, etc.)
- Looking at the **genetic code**, you can see that certain pairs of amino acids are such that the minimum number of mutations at the codon level to change the encoding from one amino acid type to another is only one (Ala and Asp, GCC and GAC), there are pairs that need a minimum of two mutations (Ala and Arg, CGA and GCA) or even three (Asn and Trp, AAC or AAU and UGG).
- The substitution score is expected to **reflect both** of these effects.

(20) Amino Acids

A (ALA)

D (Asp)

E (Glu)

K (Lys)

P (Pro)

W (Trp)

V (Val)

R (Arg)

C (Cys)

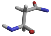
G (Gly)

I (Ile)

M (Met)

S (Ser)

Y (Tyr)


N (Asn)

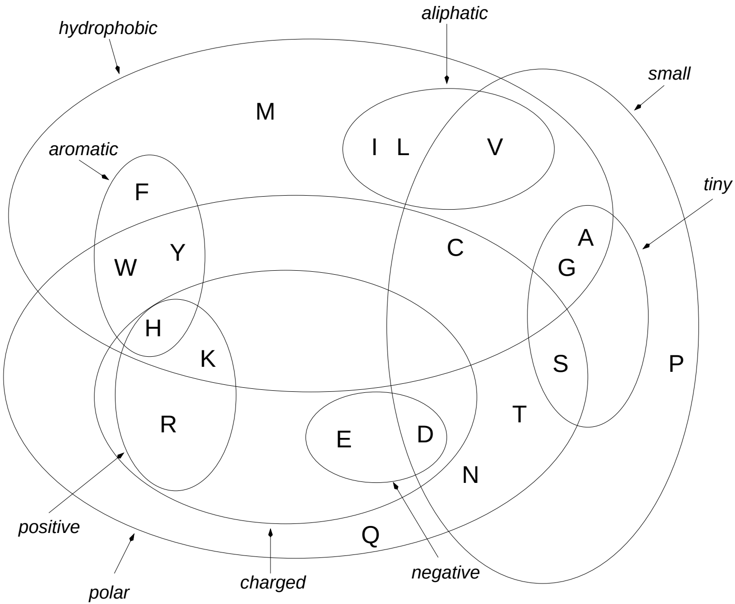
Q (Gln)

H (His)

L (Leu)

F (Phe)

T (Thr)


hydrophobic
aliphatic
small
M
I
L
V
aromatic
F
tiny
A
C
G
W
Y
H
K
S
P
T
R
E
D
N
positive
Q
polar
charged
negative

Genetic Code

	U		C		A		G		
U	UUU	Phe	UCU	Ser	UAU	Tyr	UGU	Cys	U
U	UUC	Phe	UCC	Ser	UAC	Tyr	UGC	Cys	C
U	UUA	Leu	UCA	Ser	UAA	*Stop*	UGA	*Stop*	G
U	UUG	Leu	UCG	Ser	UAG	*Stop*	UGG	Trp	A
C	CUU	Leu	CCU	Pro	CAU	His	CGU	Arg	U
C	CUC	Leu	CCC	Pro	CAC	His	CGC	Arg	C
C	CUA	Leu	CCA	Pro	CAA	Gln	CGA	Arg	A
C	CUG	Leu	CCG	Pro	CAG	Gln	CGG	Arg	G
A	AUU	Ile	ACU	Thr	AAU	Asn	AGU	Ser	U
A	AUC	Ile	ACC	Thr	AAC	Asn	AGC	Ser	C
A	AUA	Ile	ACA	Thr	AAA	Lys	AGA	Arg	A
A	AUG	Met	ACG	Thr	AAG	Lys	AGG	Arg	G
G	GUU	Val	GCU	Ala	GAU	Asp	GGU	Gly	U
G	GUC	Val	GCC	Ala	GAC	Asp	GGC	Gly	C
G	GUA	Val	GCA	Ala	GAA	Glu	GGA	Gly	A
G	GUG	Val	GCG	Ala	GAG	Glu	GGG	Gly	G

Deriving Scores

- Could be derived from **first principles** (chemical properties, etc.)
- Could be **estimated from the data**

Pitfalls

- **Sampling problem**: sequences come into families
- **Time dependent**: for distant sequences, we'd expect the probability of a substitution to be large, and low if the two sequences are close homologues
    - For short time periods, the influence of the genetic code is expected to be stronger than the chemical properties, the trend should be reversed for longer intervals.

```
86.5% identity;         Global alignment score: 786

             10        20        30        40        50        60
A    VLSAADKGNVKAAWGKVGGHAAEYGAEALERMFLSFPTTKTYFPHFDLSHGSAQVKGHGA
      :::::::.:::::::::::.:. .::::::::::.:::::::::::.::::: :::.::
B    VLSAADKANVKAAWGKVGGQAGAHGAEALERMFLGFPTTKTYFPHFNLSHGSDQVKAHGQ
             10        20        30        40        50        60
```

```
24.8% identity;         Global alignment score: 46

               10        20        30        40         50
A    VLSAADKGNVKAAWGKVGGHAAEYGAEALERMFLSFPTTKTYFPHFD-LSHGSAQ--VKG
       ::::.: :::..:.:.   .: .:. . : ..  : .   : .:. :  :.:.  ::.
B    SLSAAQKDNVKSSWAKA---SAAWGTAGPEFFMALFDAHDDVFAKFSGLFSGAAKGTVKN
               10           20        30        40        50
```

⇒ Consider the subtitution s(Gly,Ala) at position 8 of the first alignment and the same substitution at position 15 in the second alignment, are those two substitutions equally likely?

Markov Chains

- We need a **framework** to model substitutions.
    - **Discrete-time homogeneous finite Markov chain models**

Our presentation will be informal. An entire course could be taught on Markov chains and stochastic processes.

- **MAT 4374 Modern Computational Statistics**
  Simulation including the rejection method and importance sampling; applications to Monte Carlo Markov chains. Resampling methods such as the bootstrap and jackknife, with applications. Smoothing methods in curve estimation.
- **MAT 5198 Stochastic Models**
  Markov systems, stochastic networks, queuing networks, spatial processes, approximation methods in stochastic processes and queuing theory. Applications to the modelling and analysis of computer-communications systems and other distributed networks.

Markov Chains

- Like **finite state automata** (FSA):

Markov Chains

- Like **finite state automata** (FSA):
  - Finite Markov chains allow to model processes which can be represented by a **finite number of states**.

Markov Chains

- Like **finite state automata** (FSA):
  - Finite Markov chains allow to model processes which can be represented by a **finite number of states**.
  - A **process can be in any of these states at a given time**; for some **discrete units of time** $t = 0, 1, 2, \ldots$.

Markov Chains

- Like **finite state automata** (FSA):
  - Finite Markov chains allow to model processes which can be represented by a **finite number of states**.
  - A **process can be in any of these states at a given time**; for some **discrete units of time** $t = 0, 1, 2, \ldots$.
  - E.g. the amino acid type for a given sequence position at time t.

Markov Chains

- Unlike FSAs:

Markov Chains

- Unlike FSAs:
  - The **transitions from one state to another are stochastic** (not deterministic).

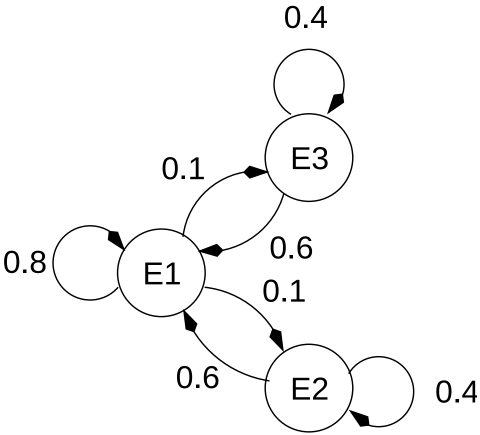
Markov Chains

- Unlike FSAs:
  - The **transitions from one state to another are stochastic** (not deterministic).
  - If the current state of the process at time t is E_i then at time $t+1$ either the process stays in E_i or move to E_j, for some j, according to a well defined probability.

Markov Chains

- Unlike FSAs:
  - The **transitions from one state to another are stochastic** (not deterministic).
  - If the current state of the process at time t is E_i then at time $t+1$ either the process stays in E_i or move to E_j, for some j, according to a well defined probability.
  - E.g. at time $t+1$ the amino acid type for a given sequence position either stays the same of is substituted by one of the remaining 19 amino acid types, according to a well defined probability, to be estimated.

Markov Chains



Properties

A (first-order) **Markovian process** must conform to the following 2 properties:

1. **Memory less**. If a process is in state E_i at time t then the probability that it will be in state E_j at time $t + 1$ only depends on E_i (and not on the previous states visited at time $t' < t$, no history). This is called a first-order Markovian process.

Properties

A (first-order) **Markovian process** must conform to the following 2 properties:

1. **Memory less.** If a process is in state E_i at time t then the probability that it will be in state E_j at time $t+1$ only depends on E_i (and not on the previous states visited at time $t' < t$, no history). This is called a first-order Markovian process.
2. **Homogeneity of time.** If a process is in state E_i at time t then the probability that it will be in state E_j at time $t+1$ is independent of t.

Mutations are often modeled as the result of a **Markovian process**. For a given protein, if the amino acid type found at a certain position is A at time t then:

Mutations are often modeled as the result of a **Markovian process**. For a given protein, if the amino acid type found at a certain position is A at time t then:

1. The probability that A is replaced by B at time $t+1$ depends only on the **current amino acid type** found at this position at time t, which is A, and the fact that C was previously found at this position for some $t' < t$ does not influence the probability of A being substituted by B.

Sometimes the concept of **time** is replaced by that of **space**. This allows to model dependencies along a protein or DNA sequence.

Mutations are often modeled as the result of a **Markovian process**. For a given protein, if the amino acid type found at a certain position is A at time t then:

1. The probability that A is replaced by B at time $t+1$ depends only on the **current amino acid type** found at this position at time t, which is A, and the fact that C was previously found at this position for some $t' < t$ does not influence the probability of A being substituted by B.
2. Also, the probability of A being replaced by B at $t+1$ is independent of t, i.e. the fact that this event is occuring **now** or **250 million years ago** does not affect the probability of A being substituted by B.

Sometimes the concept of **time** is replaced by that of **space**. This allows to model dependencies along a protein or DNA sequence.

Markov chain

A (first-order) **Markov chain** is a sequence of random variables

$$X_0, \ldots, X_{t-1}, X_t$$

that satisfies the following property

$$P(X_t = x_t | X_{t-1} = x_{t-1}, X_{t-2} = x_{t-2}, \ldots, X_0 = x_0) = P(X_t = x_t | X_{t-1} = x_{t-1})$$

Markov chain

More generally, a *m*-**order** Markov chain is a sequence of random variables:

$$X_0, \ldots, X_{t-1}, X_t$$

that satisfies the following property

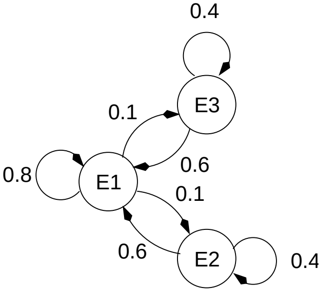
$$P(X_t = x_t | X_{t-1} = x_{t-1}, X_{t-2} = x_{t-2}, \ldots, X_0 = x_0)$$

$$= P(X_t = x_t | X_{t-1} = x_{t-1}, \ldots, X_{t-m} = x_m)$$

a 0-order model is known as a **Bernouilli model**. Markov chain models are denoted Mm, where m is the order of the model, e.g. $M0$, $M1$, $M2$, $M3$, etc.

Transition Probabilities

The **transition probabilities**, p_{ij}, can be represented graphically,



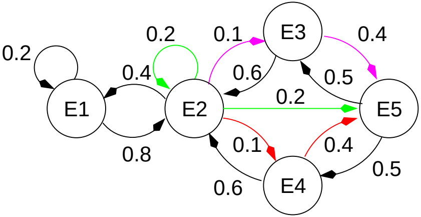
or as a **transition probability matrix**,

$$P = \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.6 & 0.4 & 0.0 \\ 0.6 & 0.0 & 0.4 \end{bmatrix}$$

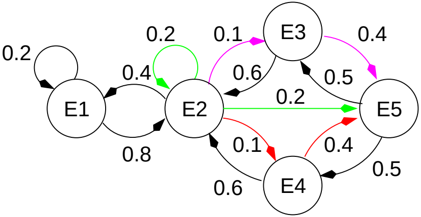
Transition Probabilities

$$P = \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.6 & 0.4 & 0.0 \\ 0.6 & 0.0 & 0.4 \end{bmatrix}$$

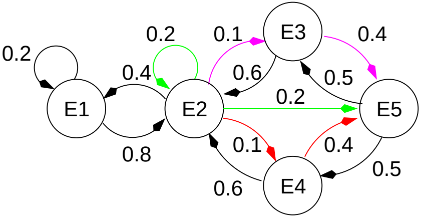
- where p_{ij} is understood as the probability of a transition from state i (row) to state j (column).
- The values in a row represent all the transitions from state i, i.e. all outgoing arcs, and therefore their **sum must be 1**.



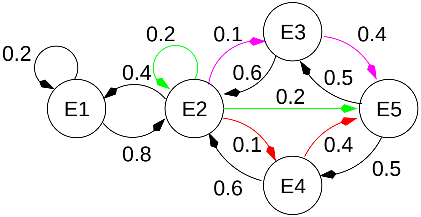
- The framework allows to answer elegantly questions such as this one, ‘‘**a Markovian random variable is in state** E_i **at time** t**, what is the probability that it will be in state** E_j **at** $t + 2$**?**''



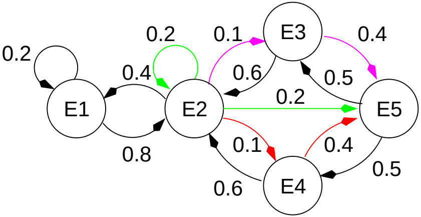
- The framework allows to answer elegantly questions such as this one, ‘‘**a Markovian random variable is in state E_i at time t, what is the probability that it will be in state E_j at $t+2$?**”
- For the Markovian process graphically depicted above, knowing that a random variable is in state E_2 at time t **what is the probability that it will be state E_5 at** $t+2$, i.e. after two transitions?



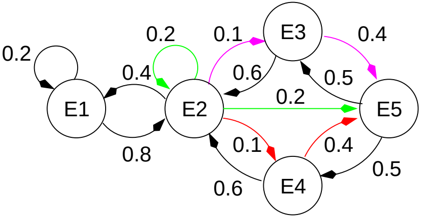
- There are exactly **3 paths of length 2** leading from E_2 to E_5: (E_2, E_2, E_5), (E_2, E_3, E_5) and (E_2, E_4, E_5).



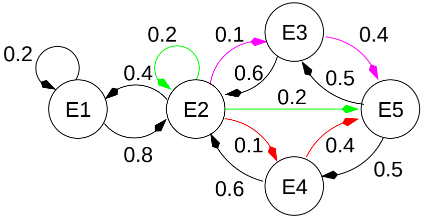
- There are exactly **3 paths of length 2** leading from E_2 to E_5: (E_2, E_2, E_5), (E_2, E_3, E_5) and (E_2, E_4, E_5).
  - The probability that (E_2, E_2, E_5) is followed is $0.2 \times 0.2 = 0.04$



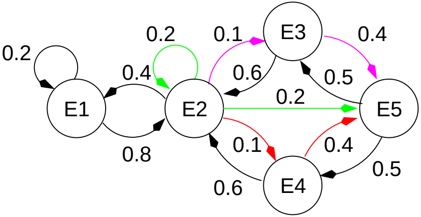
- There are exactly **3 paths of length 2** leading from E_2 to E_5: (E_2, E_2, E_5), (E_2, E_3, E_5) and (E_2, E_4, E_5).
    - The probability that (E_2, E_2, E_5) is followed is $0.2 \times 0.2 = 0.04$
    - The probability that (E_2, E_3, E_5) is followed is $0.1 \times 0.4 = 0.04$



- There are exactly **3 paths of length 2** leading from E_2 to E_5: (E_2, E_2, E_5), (E_2, E_3, E_5) and (E_2, E_4, E_5).
  - The probability that (E_2, E_2, E_5) is followed is $0.2 \times 0.2 = 0.04$
  - The probability that (E_2, E_3, E_5) is followed is $0.1 \times 0.4 = 0.04$
  - The probability that (E_2, E_4, E_5) is followed is $0.1 \times 0.4 = 0.04$



- There are exactly **3 paths of length 2** leading from E_2 to E_5: (E_2, E_2, E_5), (E_2, E_3, E_5) and (E_2, E_4, E_5).
  - The probability that (E_2, E_2, E_5) is followed is $0.2 \times 0.2 = 0.04$
  - The probability that (E_2, E_3, E_5) is followed is $0.1 \times 0.4 = 0.04$
  - The probability that (E_2, E_4, E_5) is followed is $0.1 \times 0.4 = 0.04$
  - Therefore, the probability that the random variable is found in E_5 at $t + 2$ knowing that it was in E_2 at t is $0.04 + 0.04 + 0.04 = 0.12$.



- **In general**, the probability that a random variable is found in state E_j at $t+2$ knowing that it was in E_i at t is,

$$p_{ij}^{(2)} = \sum_k p_{ik} p_{kj}$$

- ...which is **the product of row** i **by column** j of the transition probability matrix.

$$P = \begin{bmatrix} 0.2 & 0.8 & 0.0 & 0.0 & 0.0 \\ 0.4 & 0.2 & 0.1 & 0.1 & 0.2 \\ 0.0 & 0.6 & 0.0 & 0.0 & 0.4 \\ 0.0 & 0.6 & 0.0 & 0.0 & 0.4 \\ 0.0 & 0.0 & 0.5 & 0.5 & 0.0 \end{bmatrix}$$

$$P^2 = \begin{bmatrix} 0.36 & 0.32 & 0.08 & 0.08 & 0.16 \\ 0.16 & 0.48 & 0.12 & 0.12 & 0.12 \\ 0.24 & 0.12 & 0.26 & 0.26 & 0.12 \\ 0.24 & 0.12 & 0.26 & 0.26 & 0.12 \\ 0.00 & 0.60 & 0.00 & 0.00 & 0.40 \end{bmatrix}$$

- ...which is **the product of row** i **by column** j of the transition probability matrix.
- This is also the element (i,j) in the matrix P^2!

$$P = \begin{bmatrix} 0.2 & 0.8 & 0.0 & 0.0 & 0.0 \\ 0.4 & 0.2 & 0.1 & 0.1 & 0.2 \\ 0.0 & 0.6 & 0.0 & 0.0 & 0.4 \\ 0.0 & 0.6 & 0.0 & 0.0 & 0.4 \\ 0.0 & 0.0 & 0.5 & 0.5 & 0.0 \end{bmatrix}$$

$$P^2 = \begin{bmatrix} 0.36 & 0.32 & 0.08 & 0.08 & 0.16 \\ 0.16 & 0.48 & 0.12 & 0.12 & 0.12 \\ 0.24 & 0.12 & 0.26 & 0.26 & 0.12 \\ 0.24 & 0.12 & 0.26 & 0.26 & 0.12 \\ 0.00 & 0.60 & 0.00 & 0.00 & 0.40 \end{bmatrix}$$

- ...which is **the product of row** i **by column** j of the transition probability matrix.
- This is also the element (i, j) in the matrix P^2!
- Hence, P^2 gives **all** the transition probabilities moving from state E_i to E_j in **two units of time** (steps).

$$P = \begin{bmatrix} 0.2 & 0.8 & 0.0 & 0.0 & 0.0 \\ 0.4 & 0.2 & 0.1 & 0.1 & 0.2 \\ 0.0 & 0.6 & 0.0 & 0.0 & 0.4 \\ 0.0 & 0.6 & 0.0 & 0.0 & 0.4 \\ 0.0 & 0.0 & 0.5 & 0.5 & 0.0 \end{bmatrix}$$

$$P^2 = \begin{bmatrix} 0.36 & 0.32 & 0.08 & 0.08 & 0.16 \\ 0.16 & 0.48 & 0.12 & 0.12 & 0.12 \\ 0.24 & 0.12 & 0.26 & 0.26 & 0.12 \\ 0.24 & 0.12 & 0.26 & 0.26 & 0.12 \\ 0.00 & 0.60 & 0.00 & 0.00 & 0.40 \end{bmatrix}$$

- ...which is **the product of row i by column j** of the transition probability matrix.
- This is also the element (i, j) in the matrix P^2!
- Hence, P^2 gives **all** the transition probabilities moving from state E_i to E_j in **two units of time** (steps).

$$P = \begin{bmatrix} 0.2 & 0.8 & 0.0 & 0.0 & 0.0 \\ 0.4 & 0.2 & 0.1 & 0.1 & 0.2 \\ 0.0 & 0.6 & 0.0 & 0.0 & 0.4 \\ 0.0 & 0.6 & 0.0 & 0.0 & 0.4 \\ 0.0 & 0.0 & 0.5 & 0.5 & 0.0 \end{bmatrix}$$

$$P^2 = \begin{bmatrix} 0.36 & 0.32 & 0.08 & 0.08 & 0.16 \\ 0.16 & 0.48 & 0.12 & 0.12 & 0.12 \\ 0.24 & 0.12 & 0.26 & 0.26 & 0.12 \\ 0.24 & 0.12 & 0.26 & 0.26 & 0.12 \\ 0.00 & 0.60 & 0.00 & 0.00 & 0.40 \end{bmatrix}$$

- ...which is **the product of row i by column j** of the transition probability matrix.
- This is also the element (i, j) in the matrix P^2!
- Hence, P^2 gives **all** the transition probabilities moving from state E_i to E_j in **two units of time** (steps).

$$P = \begin{bmatrix} 0.2 & 0.8 & 0.0 & 0.0 & 0.0 \\ 0.4 & 0.2 & 0.1 & 0.1 & 0.2 \\ 0.0 & 0.6 & 0.0 & 0.0 & 0.4 \\ 0.0 & 0.6 & 0.0 & 0.0 & 0.4 \\ 0.0 & 0.0 & 0.5 & 0.5 & 0.0 \end{bmatrix}$$

$$P^2 = \begin{bmatrix} 0.36 & 0.32 & 0.08 & 0.08 & 0.16 \\ 0.16 & 0.48 & 0.12 & 0.12 & 0.12 \\ 0.24 & 0.12 & 0.26 & 0.26 & 0.12 \\ 0.24 & 0.12 & 0.26 & 0.26 & 0.12 \\ 0.00 & 0.60 & 0.00 & 0.00 & 0.40 \end{bmatrix}$$

$\Rightarrow$ What are all those zeros?

- **In general**, P^n (P to the nth power) gives all the "n-steps" transition probabilities.

$$P^5 = \begin{bmatrix} 0.1974 & 0.3827 & 0.1280 & 0.1280 & 0.1638 \\ 0.1914 & 0.3894 & 0.1182 & 0.1182 & 0.1827 \\ 0.1536 & 0.4406 & 0.1085 & 0.1085 & 0.1888 \\ 0.1536 & 0.4406 & 0.1085 & 0.1085 & 0.1888 \\ 0.2304 & 0.2688 & 0.1688 & 0.1688 & 0.1632 \end{bmatrix}$$

$$P^{25} = \begin{bmatrix} 0.1899 & 0.3797 & 0.1266 & 0.1266 & 0.1772 \\ 0.1899 & 0.3797 & 0.1266 & 0.1266 & 0.1772 \\ 0.1899 & 0.3797 & 0.1266 & 0.1266 & 0.1772 \\ 0.1899 & 0.3797 & 0.1266 & 0.1266 & 0.1772 \\ 0.1899 & 0.3797 & 0.1266 & 0.1266 & 0.1772 \end{bmatrix}$$

In three steps, we have,

$$p_{ij}^{(3)} = \sum_k p_{ik} p_{kj}^{(2)}$$

In three steps, we have,

$$p_{ij}^{(3)} = \sum_k p_{ik} p_{kj}^{(2)}$$

and for *n* steps,

$$p_{ij}^{(n)} = \sum_k p_{ik} p_{kj}^{(n-1)}$$

In three steps, we have,

$$p_{ij}^{(3)} = \sum_k p_{ik} p_{kj}^{(2)}$$

and for n steps,

$$p_{ij}^{(n)} = \sum_k p_{ik} p_{kj}^{(n-1)}$$

In other words,

$$P^{(n)} = \underbrace{P \times P \times \ldots \times P}_{n \text{ times}}$$

PAM Matrices

- Dayhoff, M., Schwartz, R. and Orcutt, B. (1978). A model of evolutionary change in protein. In *Atlas of Protein Sequences and Structure*, **5**, 345–352.

PAM Matrices

- Dayhoff, M., Schwartz, R. and Orcutt, B. (1978). A model of evolutionary change in protein. In *Atlas of Protein Sequences and Structure*, **5**, 345–352.
- PAM stands for “Point Accepted Mutation”, which is a mutation which not only has occurred but it has also been **retained** and has **spread** to the entire population (species).

PAM Matrices

- Dayhoff, M., Schwartz, R. and Orcutt, B. (1978). A model of evolutionary change in protein. In *Atlas of Protein Sequences and Structure*, **5**, 345–352.
- PAM stands for "Point Accepted Mutation", which is a mutation which not only has occurred but it has also been **retained** and has **spread** to the entire population (species).
- The **PAM1** matrix is a Markov chain matrix corresponding to a period of time such that **1% of the amino acids have undergone a point accepted mutation**.

Margaret Dayhoff (1925–1983)

Georgetown University Medical Center Professor, and **Bioinformatics** pioneer!

PAM matrix: construction

- Just like for the **BLOSUM** matrix, which is another popular substitution scheme, the probabilities are **estimated from data**.

PAM matrix: **construction**

- Just like for the **BLOSUM** matrix, which is another popular substitution scheme, the probabilities are **estimated from data**.
- The starting point is a collection ungapped multiple alignments.

PAM matrix: **construction**

- Just like for the **BLOSUM** matrix, which is another popular substitution scheme, the probabilities are **estimated from data**.
- The starting point is a collection ungapped multiple alignments.
- The sequences have to be sufficiently **close** (homologues) that they can be reliably aligned (with a **trivial substitution matrix**). Dayhoff *et al.* decided that all the sequences in an alignment had to be **no more than 15% different from any other sequence**.

PAM matrix: construction

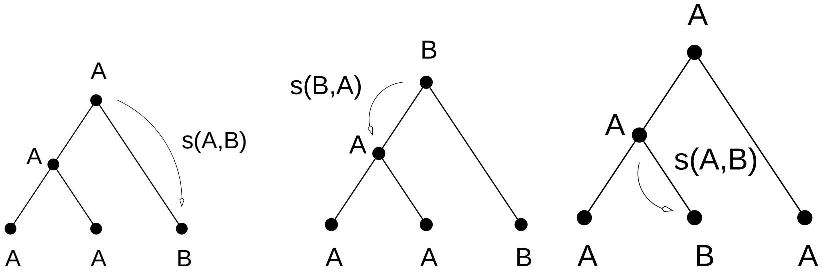
- Just like for the **BLOSUM** matrix, which is another popular substitution scheme, the probabilities are **estimated from data**.
- The starting point is a collection ungapped multiple alignments.
- The sequences have to be sufficiently **close** (homologues) that they can be reliably aligned (with a **trivial substitution matrix**). Dayhoff *et al.* decided that all the sequences in an alignment had to be **no more than 15% different from any other sequence**.
- The choice of the cutoff was also dictated by the fact that they wanted to avoid the possibility that **more than one mutation had occurred at a given site**, which is important since substitutions matrices for longer period of time will be derived from PAM1 by raising it the *n*th power.

PAM matrix: construction

- Just like for the **BLOSUM** matrix, which is another popular substitution scheme, the probabilities are **estimated from data**.
- The starting point is a collection ungapped multiple alignments.
- The sequences have to be sufficiently **close** (homologues) that they can be reliably aligned (with a **trivial substitution matrix**). Dayhoff *et al.* decided that all the sequences in an alignment had to be **no more than 15% different from any other sequence**.
- The choice of the cutoff was also dictated by the fact that they wanted to avoid the possibility that **more than one mutation had occurred at a given site**, which is important since substitutions matrices for longer period of time will be derived from PAM1 by raising it the *n*th power.

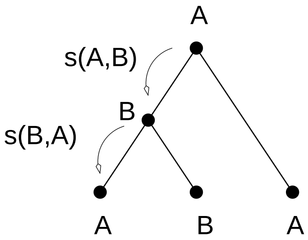
Phylogenetic trees

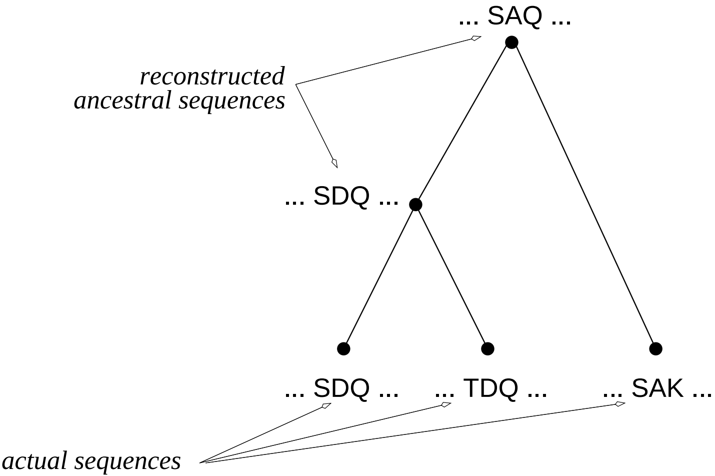
- From the sequences, phylogenetic trees are reconstructed. The method that they used is called **maximum parsimony**. It produces trees such that total number of substitutions across the whole tree is minimum.
- In the following trees, only one mutational event is necessary to explain the actual sequences:



Phylogenetic trees (continued)

On the other hand, the following tree necessitates 2 events, not minimum, therefore not the most parsimonious tree.





The trees are such that the **leaves** are labeled with the **actual (contemporary) sequences** and the **internal nodes** are labeled with **ancestral (reconstructed) sequences**. Therefore, contemporary sequences are never compared directly.

Estimation

- Pairs (i, j) are counted for **adjacent nodes** in all the trees and divided by the number of trees; if there are more than one “most parsimonious tree”.

Estimation

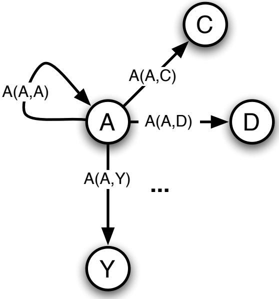
- Pairs (i, j) are counted for **adjacent nodes** in all the trees and divided by the number of trees; if there are more than one “most parsimonious tree”.
- The likelihood of a substitution i to j is assumed to be the **same as** the likelihood of a substitution j to i. Therefore, when counting the number of substitutions, cells $A_{i,j}$ and $A_{j,i}$ are both incremented.

Estimation

- Pairs (i, j) are counted for **adjacent nodes** in all the trees and divided by the number of trees; if there are more than one "most parsimonious tree".
- The likelihood of a substitution i to j is assumed to be the **same as** the likelihood of a substitution j to i. Therefore, when counting the number of substitutions, cells $A_{i,j}$ and $A_{j,i}$ are both incremented.
- The result is a matrix, A, such that A_{ij} **counts** the number of observed substitutions from/to the amino acid type i to/from the amino acid type j.

Our task is to estimate the **transition probabilities** of the Markov chain matrix, the following quantity moves us one step closer,

$$a_{ij} = \frac{A_{ij}}{\sum_k A_{ik}}$$



For reasons that will be explained in a moment, the a_{ij} are **scaled by a factor** c. For $i \neq j$, let,

$$p_{ij} = c \cdot a_{ij}$$

For reasons that will be explained in a moment, the a_{ij} are **scaled by a factor** c. For $i \neq j$, let,

$$p_{ij} = c \cdot a_{ij}$$

and

$$p_{ii} = 1 - \sum_{k \neq i} c \cdot a_{ik}$$

For reasons that will be explained in a moment, the a_{ij} are **scaled by a factor** c. For $i \neq j$, let,

$$p_{ij} = c \cdot a_{ij}$$

and

$$p_{ii} = 1 - \sum_{k \neq i} c \cdot a_{ik}$$

i.e.

$$p_{ii} = 1 - \sum_{k \neq i} p_{ik}$$

For reasons that will be explained in a moment, the a_{ij} are **scaled by a factor** c. For $i \neq j$, let,

$$p_{ij} = c \cdot a_{ij}$$

and

$$p_{ii} = 1 - \sum_{k \neq i} c \cdot a_{ik}$$

i.e.

$$p_{ii} = 1 - \sum_{k \neq i} p_{ik}$$

and $\sum_j p_{ij} = 1$ by definition.

The **expected proportion** of the amino acids that will change after one unit of time is given by,

$$\sum_i \sum_{j \neq i} p_i p_{ij}$$

where the frequency of occurrence of each amino acid type, p_i, is estimated from the observed distribution found in the original data.

The constant c is defined such that the **expected proportion** of amino acid changes, after one unit of time, is **1%**.

$$0.01 = \sum_{i} \sum_{j \neq i} p_i p_{ij}$$

The constant c is defined such that the **expected proportion** of amino acid changes, after one unit of time, is **1%**.

$$0.01 = \sum_{i} \sum_{j \neq i} p_i p_{ij}$$

$$0.01 = \sum_{i} \sum_{j \neq i} p_i \ c \ a_{ij}$$

The constant *c* is defined such that the **expected proportion** of amino acid changes, after one unit of time, is **1%**.

$$0.01 = \sum_{i} \sum_{j \neq i} p_i p_{ij}$$

$$0.01 = \sum_{i} \sum_{j \neq i} p_i \; c \; a_{ij}$$

$$0.01 = c \sum_{i} \sum_{j \neq i} p_i a_{ij}$$

The constant c is defined such that the **expected proportion** of amino acid changes, after one unit of time, is **1%**.

$$0.01 = \sum_{i} \sum_{j \neq i} p_i p_{ij}$$

$$0.01 = \sum_{i} \sum_{j \neq i} p_i \ c \ a_{ij}$$

$$0.01 = c \sum_{i} \sum_{j \neq i} p_i a_{ij}$$

i.e.,

$$c = \frac{0.01}{\sum_i \sum_{j \neq i} p_i a_{ij}}$$

- In the literature, the resulting matrix is often denoted M, rather than P, and so the p_{ij}s are referred to as m_{ij}s, and this constitutes PAM1 or M_1.

- In the literature, the resulting matrix is often denoted M, rather than P, and so the p_{ij}s are referred to as m_{ij}s, and this constitutes PAM1 or M_1.
- The element (i, j) of M_n, $m_{ij}^{(n)}$, is the probability to observe the amino type j at a given position knowing that i occurred at that same position n units of time ago.

The **transition probability matrix** is transformed into a **scoring matrix** as follows:

$$C \cdot \log\left(\frac{m_{ij}^{(n)}}{p_j}\right)$$

The **transition probability matrix** is transformed into a **scoring matrix** as follows:

$$C \cdot \log\left(\frac{m_{ij}^{(n)}}{p_j}\right)$$

Let $q(i,j)$ be the join probability that i occurred at a given position at time 0, and to observe j after n units of time, at the same position. The quantities, $q(i,j)$ and p_{ij} are related as follows,

$$q(i,j) = p_i m_{ij}^{(n)}$$

i.e.

$$m_{ij}^{(n)} = \frac{q(i,j)}{p_i}$$

Therefore, elements of the scoring matrix represent,

$$C \cdot \log\left(\frac{q(i,j)}{p_i p_j}\right)$$

which brings us back to our probabilistic interpretation of a sequence alignment:

$$S(S_1, S_2) = \sum_i \log\left(\frac{q_{S_1(i)S_2(i)}}{p_{S_1(i)} p_{S_2(i)}}\right)$$

where S_1 and S_2 are two aligned sequences.

$\Rightarrow$ PAM250 is the most frequently used matrix.

```
DayMatrix(Peptide, pam=250, Sim: max=14.152, min=-5.161, max offdiag=5.080, del=-19.814-1.396*(k-1))

C  11.5
S   0.1  2.2
T  -0.5  1.5  2.5
P  -3.1  0.4  0.1  7.6
A   0.5  1.1  0.6  0.3  2.4
G  -2.0  0.4 -1.1 -1.6  0.5  6.6
N  -1.8  0.9  0.5 -0.9 -0.3  0.4  3.8
D  -3.2  0.5 -0.0 -0.7 -0.3  0.1  2.2  4.7
E  -3.0  0.2 -0.1 -0.5 -0.0 -0.8  0.9  2.7  3.6
Q  -2.4  0.2  0.0 -0.2 -0.2 -1.0  0.7  0.9  1.7  2.7
H  -1.3 -0.2 -0.3 -1.1 -0.8 -1.4  1.2  0.4  0.4  1.2  6.0
R  -2.2 -0.2 -0.2 -0.9 -0.6 -1.0  0.3 -0.3  0.4  1.5  0.6  4.7
K  -2.8  0.1  0.1 -0.6 -0.4 -1.1  0.8  0.5  1.2  1.5  0.6  2.7  3.2
M  -0.9 -1.4 -0.6 -2.4 -0.7 -3.5 -2.2 -3.0 -2.0 -1.0 -1.3 -1.7 -1.4  4.3
I  -1.1 -1.8 -0.6 -2.6 -0.8 -4.5 -2.8 -3.8 -2.7 -1.9 -2.2 -2.4 -2.1  2.5  4.0
L  -1.5 -2.1 -1.3 -2.3 -1.2 -4.4 -3.0 -4.0 -2.8 -1.6 -1.9 -2.2 -2.1  2.8  2.8  4.0
V  -0.0 -1.0  0.0 -1.8  0.1 -3.3 -2.2 -2.9 -1.9 -1.5 -2.0 -2.0 -1.7  1.6  3.1  1.8  3.4
F  -0.8 -2.8 -2.2 -3.8 -2.3 -5.2 -3.1 -4.5 -3.9 -2.6 -0.1 -3.2 -3.3  1.6  1.0  2.0  0.1  7.0
Y  -0.5 -1.9 -1.9 -3.1 -2.2 -4.0 -1.4 -2.8 -2.7 -1.7  2.2 -1.8 -2.1 -0.2 -0.7 -0.0 -1.1  5.1  7.8
W  -1.0 -3.3 -3.5 -5.0 -3.6 -4.0 -3.6 -5.2 -4.3 -2.7 -0.8 -1.6 -3.5 -1.0 -1.8 -0.7 -2.6  3.6  4.1 14.2
      C    S    T    P    A    G    N    D    E    Q    H    R    K    M    I    L    V    F    Y    W
```

Remarks

- One of the problems with the PAM matrix, as calculated by Dayhoff et al., is that **higher values of PAM are derived from smaller values of PAM**.

Remarks

- One of the problems with the PAM matrix, as calculated by Dayhoff et al., is that **higher values of PAM are derived from smaller values of PAM**.
- For short period of times, **one would expect the substitutions to be dominated by the constraints of the genetic code**; substitutions that require a single mutation at the codon level.

Remarks

- One of the problems with the PAM matrix, as calculated by Dayhoff et al., is that **higher values of PAM are derived from smaller values of PAM**.
- For short period of times, **one would expect the substitutions to be dominated by the constraints of the genetic code**; substitutions that require a single mutation at the codon level.
- For longer period of time, **one would expect to observe substitutions that reflect the chemical properties of the amino acids**.

Remarks

- One of the problems with the PAM matrix, as calculated by Dayhoff et al., is that **higher values of PAM are derived from smaller values of PAM**.
- For short period of times, **one would expect the substitutions to be dominated by the constraints of the genetic code**; substitutions that require a single mutation at the codon level.
- For longer period of time, **one would expect to observe substitutions that reflect the chemical properties of the amino acids**.
- To overcome this problem, Henikoff & Henikoff 1991, have constructed a set of matrices, **BLOSUM**, derived from (ungapped) alignments at various percentage of identities.

Remarks (continued)

- Substitution scores are **average scores**. They do not account for the context: **n-term**, **c-term**, **exposed**, **buried**, **helix**, **strands**, etc.
- The cost of a substitution, say Ala to Trp, remains the same no matter where along the sequence the substitution occurs. Later, we will consider models where the cost of a substitution varies along the sequence; position specific scoring matrices and **Hidden Markov Models**.

References

- W. J. Ewens and G.R. Grant (2001) Statistical Methods in Bioinformatics. Springer. pp. 199–210.
- Kosiol, C., & Gojobori, T. (2005). Different versions of the dayhoff rate matrix. *Molecular Biology and Evolution*, 22(2), 193–199.
- Ortet, P., & Bastien, O. (2010). Where does the alignment score distribution shape come from? Evolutionary Bioinformatics Online, 6(6), 159–187. http://doi.org/10.4137/EBO.S5875
- Dan Gusfield (1997) Algorithms on strings, trees, and sequences: computer science and computational biology. Cambridge Press, §11 and 15.
- A. Isaev (2006) Introduction to Mathematical Methods in Bioinformatics. Springer, §3 (Markov chains/models), §6 (Probability theory), §8 (Statistics), §7 (Significance of an alignment score) and **§9** (Substitution matrices).

References

Pensez-y!

L'impression de ces notes n'est probablement pas nécessaire!