Biological sequences analysis
(with exercices)

Catherine Matias

CNRS - Laboratoire de Probabilités et Modèles Aléatoires, Paris
catherine.matias@math.cnrs.fr
http://cmatias.perso.math.cnrs.fr/

LNCC, Petrópolis
October 2014

Outline of this course

- Part I: Introduction to sequence analysis
- Part II: Motifs detection
- Part III: Sequence evolution and alignment
- Part IV: Introduction to phylogeny

Part I

Introduction to sequence analysis

Biological sequences

What kind of sequences?

- DNA sequences (genes, regions, genomes, . . .) with alphabet $\mathcal{A} = \{A, C, G, T\}$.
- Protein sequences, with alphabet $\mathcal{A} = \{20 \text{ amino acids}\} = \{\text{Ala, Cys, Asp, Glu } \ldots\}$.
- RNA sequences, with alphabet $\mathcal{A} = \{A, C, G, U\}$.
- Obtained from different sequencing technologies.

Examples of repositories

- Primary sequences: GenBank
- Genome databases (with annotation): Ensembl (human, mouse, other vertebrates, eukaryotes . . .) and Ensembl Genomes (bacteria, fungi, plants,. . .)
- Protein sequences: UniProt, Swiss-Prot, PROSITE (protein families and domains)

Why do we need sequence analysis?

- Once the sequences are obtained, what do we learn from a biological point of view?
- Need of statistical and computational tools to extract biological information from these sequences.

Some of the oldest issues

- Where are the functional motifs: cross-over hotspot instigators (chi), restriction sites, regulation motifs, binding sites, active sites in proteins, etc.
  → Motif discovery issues.
- How do we explain differences between two genome species? → Sequence evolution models.
- How can we compare genomes of neighbour species?
  → Sequence alignment problem.
- How do we infer the ancestral relationships between sequences/species? → Phylogenies reconstruction.

Goals and tools

Some examples of Biological issues, Statistical answers and Corresponding tools

- Search for motifs, *i.e.* short sequences with unexpected occurrence behaviour
  - a) too rare or too frequent
  - or b) with a different distribution from background

  Define a "null model" (=what you expect, from already known information) and test if
  - a) the number of occurrences of a word is too large or too small w.r.t. this model
  - or b) the distribution of letters in this word is different from the model

  Markov chains or hidden Markov chains
- Understand differences between 2 copies of a gene in neighbour species, Models of sequence mutation, Markov processes (=time continuous Markov chains)

Biological models: constraints and usefulness

- A model is never true, it only has to be useful.
- That means that it should remain simple (for mathematical and computational issues) but also realistic: these two properties are in contradiction and one must find a balance.
- Understanding the model, its limitations and underlying assumptions is mandatory for correct biological interpretation.

Recap on probability

Formulas you need for this course

Conditional probability. For any events A, B,

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)}$$

Marginalization. For any discrete r.v. $X \in \mathcal{X}, Y \in \mathcal{Y}$,

$$\mathbb{P}(X = x) = \sum_{y \in \mathcal{Y}} \mathbb{P}(X = x, Y = y)$$

Expectation of an indicator function. For any sets A, B, any r.v. X with distribution $\mathbb{P}$ and any other r.v. Y,

$$\mathbb{E}(1_A) = \mathbb{P}(X \in A) \quad \text{and} \quad \mathbb{E}(1_A|B) = \mathbb{P}(X \in A|B).$$

Books references I


R. Durbin, S. Eddy, A. Krogh, and G. Mitchison.
Biological sequence analysis: probabilistic models of proteins and nucleic acids.
Cambridge University Press, Cambridge, UK, 1998.


Z. Yang.
Computational Molecular Evolution.
Oxford Series in Ecology and Evolution. Oxford University Press, 2006.


J. Felsenstein.
Inferring phylogenies.
Sinauer Associates, 2004.

Books references II

In French

G. Deléage and M. Gouy
Bioinformatique - Cours et cas pratique.
Dunod, 2013.

G. Perrière and C. Brochier-Armanet
Concepts et méthodes en phylogénie moléculaire.
Collection IRIS, Springer, 2010.

Chapters in books

E. Allman and J. Rhodes
Mathematical models in Biology. An introduction.
Chapters 4 and 5.
Cambridge University Press, 2004.

Part II

Motifs detection

Outline: Motifs detection

Markov chains (order 1)

Higher order Markov chains

Motifs detection with Markov chains

Hidden Markov models (HMMs)

Parameter estimation in HMM

Sequence segmentation with HMM

Motifs detection with HMMs

Motifs detection

Under this name, we group different biological problems

- Find functional motifs, such as cross-over hotspot instigators (chi), restriction sites, regulation motifs, binding sites, active sites in proteins, *etc*
- Identify and annotate genes in a sequence
- Browsing all words with small specified length, find those that behave abnormally (statistically) (for further biological investigation)
- ...

Outline Part 2

Markov chains (order 1)

Higher order Markov chains

Motifs detection with Markov chains

Hidden Markov models (HMMs)

Parameter estimation in HMM

Sequence segmentation with HMM

Motifs detection with HMMs

Modeling a sequence I

A biological sequence may be viewed as a sequence of random variables $X_1, \ldots, X_n$ (also denoted $X_{1:n}$) with values in a finite alphabet $\mathcal{A}$.

- The simplest model on these r.v. is i.i.d. model.

$\rightarrow$ Each site X_i behaves independently from the other sites and takes values in $\mathcal{A}$ with same distribution $\pi = (\pi(x), x \in \mathcal{A})$.
Here, $\pi(x) \geq 0$ and $\sum_{x \in \mathcal{A}} \pi(x) = 1$.
Exercise: $\mathcal{A} = \{A, C, G, T\}$, observe sequence $AACTTTGAC$.
Estimate the probabilities $\pi(A), \pi(C), \pi(G), \pi(T)$.

Modeling a sequence II

- However, it is easily seen from real biological data that the occurrence frequency of dinucleotides differs from the product of corresponding nucleotides frequencies, *i.e.* for any two letters $a, b \in \mathcal{A}$, we have

$$f_{ab} = \frac{N(ab)}{n-1} \neq f_a f_b = \frac{N(a)}{n} \frac{N(b)}{n}$$

  where $N(ab)$ = number of dinucleotides ab, while this should be (approximately) the case for long iid sequences.

- It seems natural to assume that the letters occurrences are dependent. Ex: in CpG islands (= regions with high frequency of dinucleotide CG), the probability of observing a G coming after a C is higher than after a A.
  $\rightarrow$ gives rise to Markov chain model.

Markov chains: definition

Principle

A (homogeneous) Markov chain is a sequence of dependent random variables such that the future state depends on the past observations only through the present state.

Mathematical formulation

Let $\{X_n\}_{n\geq 1}$ be a sequence of random variables with values in finite or countable space $\mathcal{A}$, s.t. $\forall i \geq 1, \forall x_{1:i+1} \in \mathcal{A}^{i+1}$,

$$\mathbb{P}(X_{i+1} = x_{i+1} | X_{1:i} = x_{1:i}) = \mathbb{P}(X_{i+1} = x_{i+1} | X_i = x_i) := p(x_i, x_{i+1})$$

p is the transition of the chain. When $\mathcal{A}$ is finite, this is a stochastic matrix: it has non-negative entries $p(x, x') \geq 0$ and its rows sum to one $\sum_{x' \in \mathcal{A}} p(x, x') = 1$ for all $x \in \mathcal{A}$.

Example I

Example of a transition matrix on state space $\mathcal{A} = \{A, C, G, T\}$.

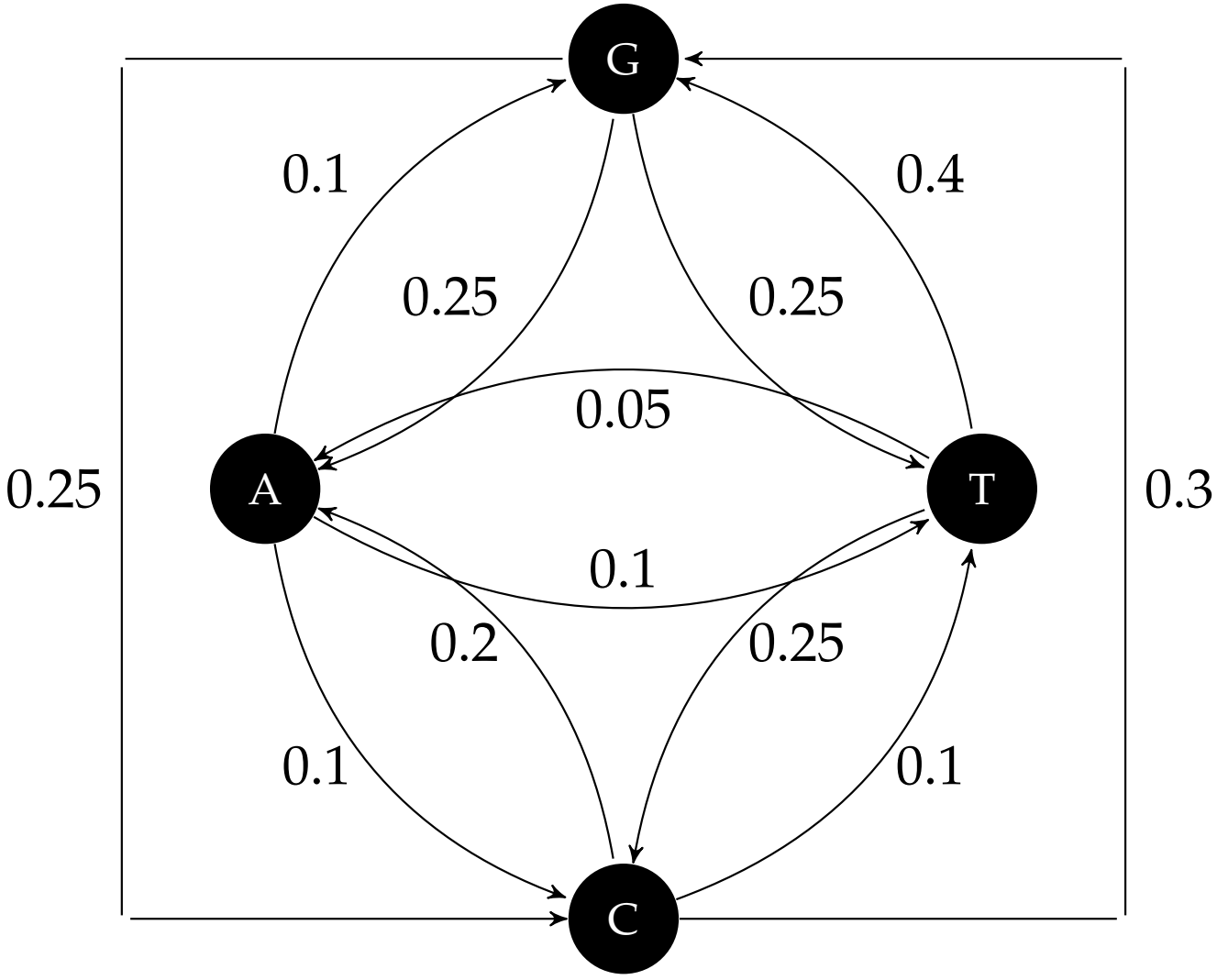
$$p = \begin{pmatrix} 0.7 & 0.1 & 0.1 & 0.1 \\ 0.2 & 0.4 & 0.3 & 0.1 \\ 0.25 & 0.25 & 0.25 & 0.25 \\ 0.05 & 0.25 & 0.4 & 0.3 \end{pmatrix}. \tag{1}$$

In particular,

- $p(2,3) = \mathbb{P}(X_{k+1} = G | X_k = C) = 0.3$.
- When $X_k = A$ then $X_{k+1} = \begin{cases} A & \text{with prob. } 0.7 \\ C, G \text{ or } T & \text{with prob. } 0.1 \end{cases}$.
- When $X_k = G$, then X_{k+1} is drawn uniformly on $\mathcal{A}$.

Example II

Automaton description



Example III

Remarks

- In the automaton, we do not draw the self-loops, but these jumps exist.
- Exercise: What's the transition matrix for an i.i.d. process with distribution π?

Probability of observing a sequence I

Distribution of a Markov chain

- Need to specify distr. of X_1, called initial distribution $\pi = \{\pi(x), x \in \mathcal{A}\}$ s.t. $\pi(x) \geq 0$ and $\sum_{x \in \mathcal{A}} \pi(x) = 1$,
- e.g. $\pi = (1/4, 1/4, 1/4, 1/4)$ gives uniform probability on $\mathcal{A} = \{A, C, G, T\}$, while $\pi = (0, 0, 1, 0)$ gives $X_1 = G$ almost surely.
- From initial distribution + transition, the distribution of the chain is completely specified (see below).

Probability of observing a sequence II

Probability of a sequence

For any $n \geq 1, \forall (x_1, \ldots, x_n) \in \mathcal{A}^n$, we get

$$\mathbb{P}(X_{1:n} = x_{1:n}) = \pi(x_1) \prod_{i=2}^{n} p(x_{i-1}, x_i). \tag{2}$$

The likelihood of an observed Markov chain is given as a product of transitions probabilities + initial term.

Exercise: Prove (2). Hint: proceed recursively on n.

Exercise: Apply it on the sequence *AACTTTGAC*.

Probability of observing a sequence III

Proof.

$$
\begin{aligned}
&\mathbb{P}(X_{1:n} = x_{1:n}) \\
&= \mathbb{P}(X_n = x_n | X_{1:n-1} = x_{1:n-1}) \mathbb{P}(X_{1:n-1} = x_{1:n-1}) && \text{(cond. prob. formula)} \\
&= \mathbb{P}(X_n = x_n | X_{n-1} = x_{n-1}) \mathbb{P}(X_{1:n-1} = x_{1:n-1}) && \text{(Markov property)} \\
&= p(x_{n-1}, x_n) \mathbb{P}(X_{1:n-1} = x_{1:n-1}) \\
&= \ldots && \text{(induction)} \\
&= p(x_{n-1}, x_n) \ldots p(x_1, x_2) \mathbb{P}(X_1 = x_1) \\
&= \pi(x_1) \prod_{i=2}^{n} p(x_{i-1}, x_i)
\end{aligned}
$$

□

Consequence: log-likelihood of an observation

Consider a sequence of observations $X_{1:n}$ following a Markov chain with initial distribution π and transition p. Then the log-likelihood of the sequence is

$$\ell_n(\pi, p) = \log \mathbb{P}(X_{1:n}) = \sum_{x \in \mathcal{A}} 1_{\{X_1 = x\}} \log \pi(x) + \sum_{x,y \in \mathcal{A}} N(xy) \log p(x, y), \tag{3}$$

where $N(xy)$ is the number of occurrences of dinucleotide xy in the sequence $X_{1:n}$.

Exercise: Prove this. Apply it on the sequence *AACTTTGAC*.
Hint: try first to apply (2) on the example.

Proof.

According to (2)

$$\begin{aligned} \log \mathbb{P}(X_{1:n} = x_{1:n}) &= \log \pi(x_1) + \sum_{i=2}^{n} \log p(x_{i-1}, x_i) \\ &= \sum_{x \in \mathcal{A}} 1_{\{X_1 = x\}} \log \pi(x) + \sum_{x,y \in \mathcal{A}} N(xy) \log p(x, y). \end{aligned}$$

Probability of state X_n

Let $\mathcal{A} = \{1, \ldots, Q\}$, $\pi = (\pi(1), \ldots, \pi(Q))$ viewed as row vector and $p = (p(i,j))_{1 \le i,j \le Q}$ the transition matrix. Then

$$\mathbb{P}(X_n = x) = (\pi p^n)(x), \quad \forall x \in \mathcal{A},$$

where p^n is a matrix power and πp^n is a vector times matrix product.

Proof.

By induction, let π_n be the row vector containing the probabilities $\mathbb{P}(X_n = x)$. Then

$$\begin{aligned}
\pi_n(x) = \mathbb{P}(X_n = x) &= \sum_{y \in \mathcal{A}} \mathbb{P}(X_{n-1} = y, X_n = x) \\
&= \sum_{y \in \mathcal{A}} \mathbb{P}(X_{n-1} = y) \mathbb{P}(X_n = x | X_{n-1} = y) \\
&= \sum_{y \in \mathcal{A}} \pi_{n-1}(y) p(y,x) = (\pi_{n-1} p)(x).
\end{aligned}$$

□

Markov chains: other computations

In the same way,

$$p^n(x, y) = \mathbb{P}(X_n = y | X_1 = x)$$

Exercise: Take matrix p given by (1) and compute $\mathbb{P}(X_7 = C | X_5 = T)$.

Markov chains: Stationarity I

- A sequence is stationary if each random variable X_i has same distribution $\pi^\star$.
- If it exists, a stationary distr. $\pi^\star$ must satisfy

$$\pi^\star p = \pi^\star,$$

i.e. $\pi^\star$ is a left eigenvector of matrix p associated with eigenvalue 1.

Exercise: Explain where this relation comes from.

Theorem

For finite state space $\mathcal{A}$, whenever there exist some $m \geq 1$ such that $\forall x, y \in \mathcal{A}, p^m(x,y) > 0$, then a stationary distr. $\pi^\star$ exists and is unique. Moreover, we have the convergence,

$$\forall x, y \in \mathcal{A}, \quad p^n(x,y) \underset{n \to +\infty}{\to} \pi^\star(y).$$

Markov chains: Stationarity II

- Consequence: Long Markov sequences forget their initial distribution and behave in the limit as stationary Markov seq.
- Remark: this property is at the core of MCMC techniques.

Exercise: Consider matrix p given by (1) and compute its stationary distribution.

Parameter estimation I

Consider a sequence of observations $X_{1:n}$ following a Markov chain. We want to fit a transition matrix on this sequence.

Maximum likelihood estimator

From (3), the maximum likelihood estimator of transition $p(x, y)$ is

$$\hat{p}(x, y) = \frac{N(xy)}{N(x\bullet)},$$

where $N(x\bullet) = \sum_{y \in \mathcal{A}} N(xy)$. Note that π may not be consistently estimated from the sequence (only one observation X_1). Often assume stationary regime and estimate $\hat{\pi}(x) = N(x)/n$.

Consequence: the dinucleotides counts in the observed sequence give estimators for the transition probabilities.

Parameter estimation II

Proof.

According to (3), we want to maximise $\sum_{x,y\in\mathcal{A}} N(xy)\log p(x,y)$ with respect to $\{p(x,y), x,y\in\mathcal{A}\}$ under the constraint $\sum_{y\in\mathcal{A}} p(x,y) = 1$. Introducing Lagrange multipliers λ_x for each constraint $\sum_{y\in\mathcal{A}} p(x,y) - 1 = 0$, we want

$$\sup_{\{\lambda_x, p(x,y)\}_{x,y\in\mathcal{A}}} \sum_{x,y\in\mathcal{A}} N(xy)\log p(x,y) + \sum_{x\in\mathcal{A}} \lambda_x \Big(\sum_{y\in\mathcal{A}} p(x,y) - 1 \Big).$$

By deriving, we obtain the set of equations

$$\begin{cases} \frac{N(xy)}{p(x,y)} + \lambda_x = 0, & \forall (x,y)\in\mathcal{A}^2 \\ \sum_{y\in\mathcal{A}} p(x,y) - 1 = 0, & \forall x\in\mathcal{A} \end{cases}$$

which gives the result. □

Example

Exercise:
1) Consider the observation
$X_{1:20}$ = CCCACGACGTATATTTCGAC
assume a Markov model and compute the estimator $\hat{p}$ of the transition matrix p.
2) Write an `R` function in order to do this on any (character) sequence (with alphabet $\mathcal{A} = \{A, C, G, T\}$ or any finite alphabet).

Outline Part 2

Markov chains (order 1)

Higher order Markov chains

Motifs detection with Markov chains

Hidden Markov models (HMMs)

Parameter estimation in HMM

Sequence segmentation with HMM

Motifs detection with HMMs

Higher order Markov chains

Motivation and underlying idea

- In coding sequences, nucleotides are organised into codons: the frequency of third letter strongly depends on two previous ones.
- Generalize Markov chains to case where the future state depends on past r states, called r-order Markov chains.
- Case $r = 1$ is ordinary Markov chain.
- r is the length of the memory of the process.

r-order (homogeneous) Markov chain

Mathematical formulation

Let $\{X_n\}_{n\geq 1}$ be a sequence of random variables with values in finite or countable space $\mathcal{A}$, s.t. $\forall i \geq r+1, \forall x_{1:i+1} \in \mathcal{A}^{i+1}$,

$$\begin{aligned}\mathbb{P}(X_{i+1} = x_{i+1} | X_{1:i} = x_{1:i}) &= \mathbb{P}(X_{i+1} = x_{i+1} | X_{i-r+1:i} = x_{i-r+1:i}) \\ &= p(x_{i-r+1:i}, x_{i+1})\end{aligned}$$

p is the transition of the chain. When $\mathcal{A}$ is finite, this is a stochastic matrix with dimension $|\mathcal{A}|^r \times |\mathcal{A}|$.

Distribution

- Need to specify distr. of $X_{1:r}$, called initial distribution $\pi = \{\pi(x_{1:r}), x_{1:r} \in \mathcal{A}^r\}$ s.t. $\pi(x_{1:r}) \geq 0$ and $\sum_{x_{1:r} \in \mathcal{A}^r} \pi(x_{1:r}) = 1$,
- From initial distribution + transition, the distribution of the chain is completely specified.

Example of a 2-order Markov chain

Example of a transition matrix of a 2-order Markov chain on state space $\mathcal{A} = \{A, C, G, T\}$. The order of the rows is {AA, AC, AG, AT, CA, CC, CG, CT, GA, GC, GG, GT, TA, TC, TG, TT }.

$$p = \left(\begin{array}{cccc}
0.7 & 0.1 & 0.1 & 0.1 \\
0.2 & 0.4 & 0.3 & 0.1 \\
0.25 & 0.25 & 0.25 & 0.25 \\
0.05 & 0.25 & 0.4 & 0.3 \\
\hline
0.75 & 0.05 & 0.1 & 0.1 \\
0.4 & 0.1 & 0.4 & 0.1 \\
0.2 & 0.1 & 0.6 & 0.1 \\
0.05 & 0 & 0 & 0.95 \\
\hline
0.7 & 0.1 & 0.1 & 0.1 \\
0.2 & 0.4 & 0.3 & 0.1 \\
0.25 & 0.25 & 0.25 & 0.25 \\
0.05 & 0.25 & 0.4 & 0.3 \\
\hline
0.9 & 0.01 & 0.01 & 0.08 \\
0 & 0.65 & 0.3 & 0.05 \\
0.2 & 0.2 & 0.55 & 0.05 \\
0.15 & 0.25 & 0.45 & 0.15
\end{array}\right)$$

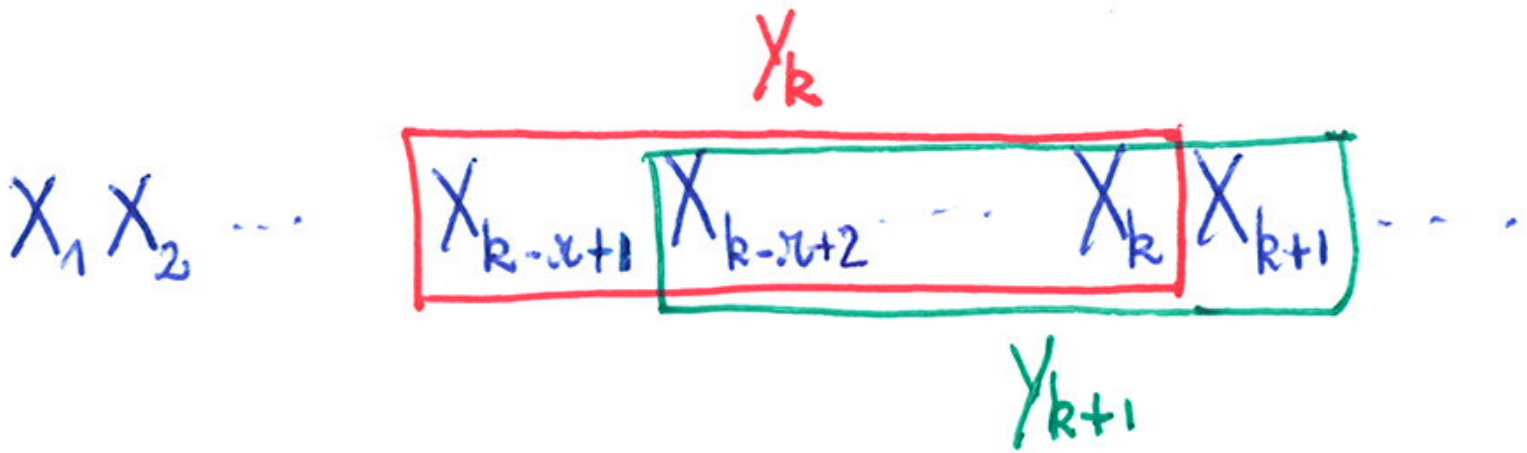
Exercise:

- What represents $p(7, 3)$?
- Comment on equality between first and third blocks.

Initial distribution $\pi = (1/16, \ldots, 1/16)$.

Remarks

- An r-order Markov chain may also be viewed as an $(r+k)$-order Markov chain for any $k \geq 0$, *i.e.* the r-order Markov chain models are embedded.
- When $\{X_k\}_{k\geq 1}$ is a r-order Markov chain, the sequence $\{Y_k\}_{k\geq r}$ defined by $Y_k = X_{k-r+1:k}$ is an order-1 Markov chain.



r-order Markov chain: transition estimation

Consider a sequence of observations $X_{1:n}$ and assume it follows a r-order Markov chain.

Maximum likelihood estimator

The maximum likelihood estimator of transition $p(x_{1:r}, y)$ is

$$\hat{p}(x_{1:r}, y) = \frac{N(x_{1:r}y)}{N(x_{1:r}\bullet)},$$

where $N(x_{1:r}y)$ counts the number of occurrences of word $x_{1:r}$ followed by letter y in $X_{1:n}$ and $N(x_{1:r}\bullet) = \sum_{y \in \mathcal{A}} N(x_{1:r}y)$.

Consequence: the counts of $(r+1)$-nucleotides (words of size $r+1$) in the observed sequence give estimators for the transition probabilities.

Exercise: Modify the previous R function for estimating transition matrices of r-order Markov chains.

Modeling through Markov chains

- Modeling a sequence through a r-order Markov chain is equivalent to saying that the sequence is characterised by the frequencies of size $(r+1)$-words.
  Ex: two sequences with same frequencies of di-nucleotides are identical from modeling through a (order 1) Markov chain point of view.
- Next issue: how to choose the value r?
  - Maximum likelihood w.r.t. r does not make sense: since the Markov chains models are embedded (*i.e.* a r-order MC is a particular case of a $r+1$- order MC), the larger the value r, the larger the value of the likelihood

$$\sup_{r \geq 1} \ell_n(r, \pi_r, p_r) = \sup_{r \geq 1} \log \mathbb{P}_{r-\text{order Markov}}(X_{1:n}) = +\infty.$$

  - However, too large values of r are not desirable because induces many parameters and thus large variance in estimation.
  - A penalty term is needed to compensate for the model size.

Order estimation: BIC I

The Bayesian Information Criterion (BIC) of a Markov chain model is defined as

$$BIC(r) = \log \hat{\mathbb{P}}_r(X_{1:n}) - \frac{N_r}{2} \log n,$$

where $\hat{\mathbb{P}}_r(X_{1:n})$ is the maximum likelihood of the sequence under a r-order Markov chain model

$$\log \hat{\mathbb{P}}_r(X_{1:n}) = \sum_{x_{1:r} \in \mathcal{A}^r, y \in \mathcal{A}} N(x_{1:r}y) \log \frac{N(x_{1:r}y)}{N(x_{1:r}\bullet)}$$

and $N_r = |\mathcal{A}|^r(|A| - 1)$ is the number of parameters (transitions) for this model.

Order estimation: BIC II

Theorem ([CS00])

Let $X_{1:n}$ be a sequence following a $r^\star$-order Markov chain, where $r^\star$ is (minimal and) unknown. Then,

$$\hat{r}_n = \sup_{r \geq 1} BIC(r) = \sup_{r \geq 1} \log \hat{\mathbb{P}}_r(X_{1:n}) - \frac{N_r}{2} \log n,$$

is a consistent estimate of r, namely $\lim_{n \to +\infty} \hat{r}_n = r^\star$ almost surely.

Exercise: Write an `R` function for computing the BIC criterion on a sequence.

Markov smoothing I

Zero counts

- As r increases, the number $|\mathcal{A}|^r$ of size-r words becomes huge. It often happens that in a finite sequence $X_{1:n}$, a word $x_{1:r}$ has zero occurrence.
- As a consequence
  - $N(x_{1:r}\bullet) = 0$ or/and $N(x_{1:r}y) = 0$ which causes pbm of dividing by zero or/and taking the logarithm of zero when computing maximum likelihood. Solution: be careful while implementing your likelihood computation and impose things like $0 \log(0/0) = 0$.
  - Putting $\hat{p}(x_{1:r}, y) = 0$ is obviously an underestimate of the transition probability $p(x_{1:r}, y)$. Solution: Markov smoothing.

Markov smoothing II

Markov smoothing

Different strategies have been developed

- Pseudo-counts: artificially add 1 to every count. Thus

$$\hat{p}(x_{1:r}, y) = \frac{1 + N(x_{1:r}y)}{\sum_{y \in \mathcal{A}} 1 + N(x_{1:r}y)} = \frac{1 + N(x_{1:r}y)}{|\mathcal{A}| + \sum_{y \in \mathcal{A}} N(x_{1:r}y)}.$$

  See page 9 in [DEKM98]. Widely used but not the wisest.
- A review of more elaborate strategies is given in [CG98].
- A performant approach is the one by Kneser-Ney [KN95].

Exercise: Include Markov smoothing into R functions for transition matrix estimation and BIC criterion.

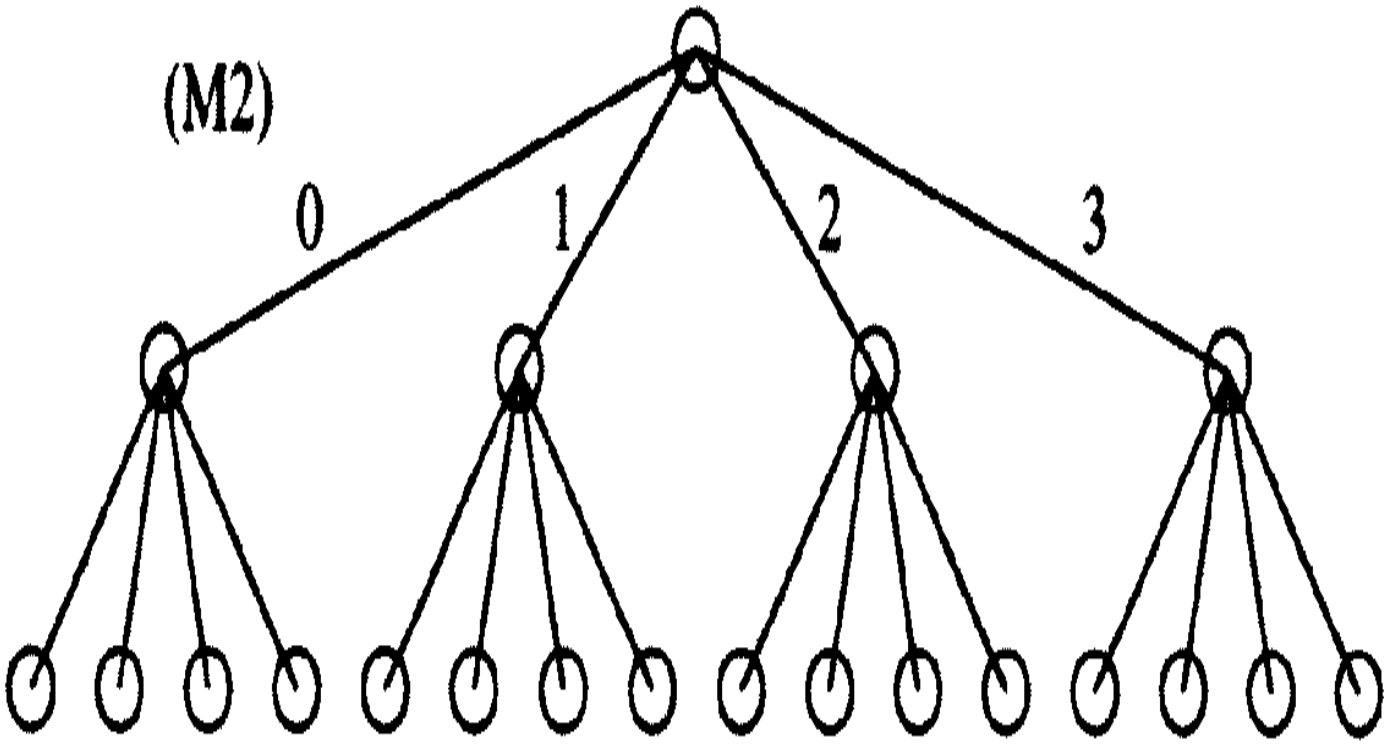
Variable length Markov chains [BW99] I

VLMC principle

- When the order r increases, the number of parameters in the r-order Markov chain model increases exponentially: $|\mathcal{A}|^r(|\mathcal{A}| - 1)$.
- For parsimony reasons, it is interesting to reduce the number of parameters, while keeping the possibility of looking at large memory values r.
- In VLMC, this is realised by letting the memory of the chain vary according to the context.

Variable length Markov chains [BW99] II

Context tree representation of a Markov chain with 4 states and order 2



Variable length Markov chains [BW99] III

Context tree representation of a VLMC

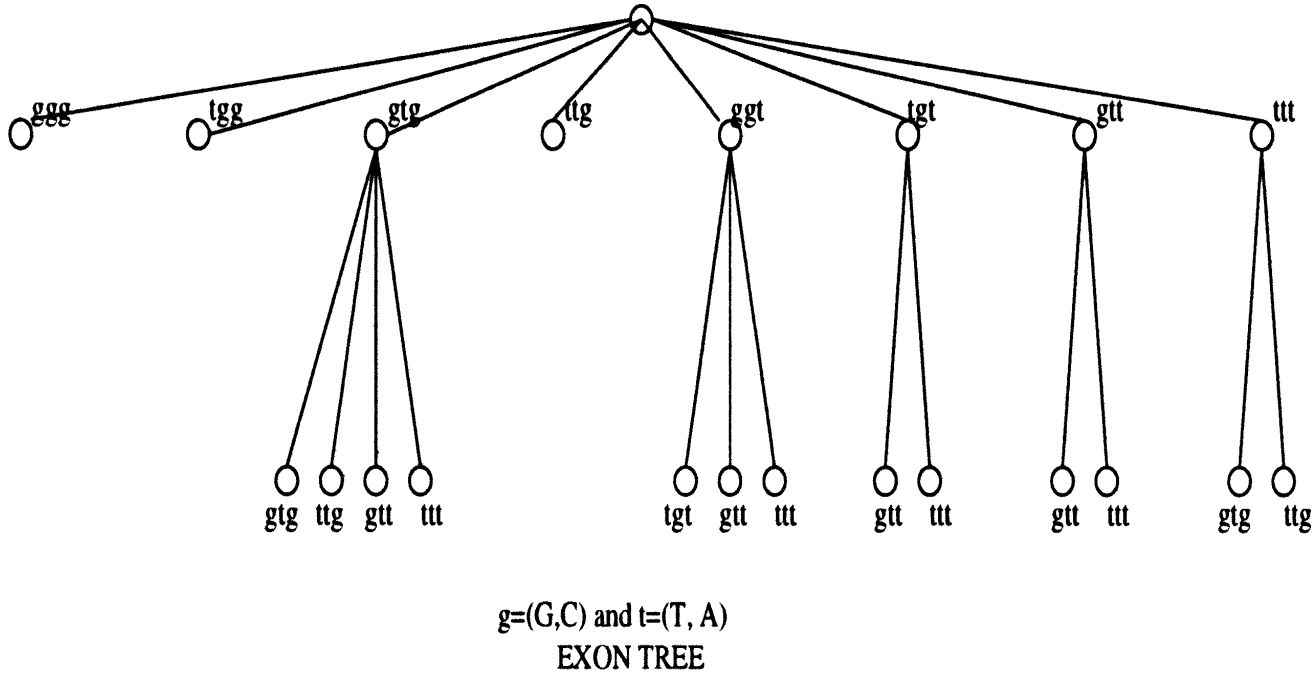


FIG. 2. *Triplet tree representation of the estimated minimal state space for exon sequence. The triplets are denoted in reverse order, for example, the terminal node with concatenation* (ggt)(gtt) *describes the context* $x_0 = g,\ x_{-1} = g,\ x_{-2} = t,\ x_{-3} = g,\ x_{-4} = t,\ x_{-5} = t$ *for the variable* x_1.

Outline Part 2

Markov chains (order 1)

Higher order Markov chains

Motifs detection with Markov chains

Hidden Markov models (HMMs)

Parameter estimation in HMM

Sequence segmentation with HMM

Motifs detection with HMMs

Detecting rare or frequent words

Principle and method

- Due to evolution pressure, functional motifs are likely to be more conserved than non-functional motifs.
- A natural strategy is to search for motifs which are statistically exceptional (ex: over- or under-represented).
- Browsing all possible words $w = w_{1:l} \in \mathcal{A}^l$ of a given length l, say if w is statistically too rare or too frequent.
- Method has two steps
  - Sequence scan: Count the number $N(w)$ of occurrences of w in the sequence. Efficient algorithms are required. See for e.g. [Nue11].
  - Statistical test: Define a "null-model" (what is expected, or already known) and look for deviations from this null model, *i.e.* counts too large or too small with respect to expected value under this null model.

Statistical test: details I

- As already mentioned, working with a r-order Markov chain model allows to take into account the sequence composition bias in $(r+1)$-mers.
- Null model $\mathcal{M}_0$: Choose a r-order Markov model with $r+1 \leq |w|-1$ (otherwise the count of w is automatically included in the model and may not be exceptional).
- It is then necessary to approximate the distribution of $N(w)$ under model $\mathcal{M}_0$. Different approximations have been proposed
  - Poisson or compound Poisson approximations;
  - Gaussian or near Gaussian approximations.
- Compare the observed value $N(w)$ to its theoretical distribution under model $\mathcal{M}_0$: if the value is below the 5%-quantile (too rare) or above the 95%- quantile (too frequent), the word is declared statistically exceptional.

Statistical test: details II

- The simplest approximation is Gaussian and computes the z-score

$$Z = \frac{N(w) - \mathbb{E}_r(N(w))}{\sqrt{\mathbb{V}\mathrm{ar}_r N(w)}} \approx \mathcal{N}(0,1)$$

  where $\mathbb{E}_r$ and $\mathbb{V}\mathrm{ar}_r$ are expectation and variance under Markov model of order r.

- Expectation easy to compute
  $\mathbb{E}_r(N(w)) = (n-l)\mathbb{P}_r(w_1 \dots w_l) = (n-l)\pi(w_1)\prod_{i=1}^{l-1} p(w_i, w_{i+1})$
- Variance is more tricky, especially if w can overlap itself !

Illustration: *E. coli*'s chi I

Context

- A "chi" is a cross-over hotspot instigator.
- RecBCD is an enzyme in *E. coli* that degrades every linear DNA strand it encounters and thus every phage.
- Remember *E. coli*'s DNA is circular thus has no end. However it sometimes opens, exposing the cell to lethal degradation.
- Whenever RecBCD encounters the chi motif, it recognises *E. coli*'s DNA and stops degradation; DNA repair may start.

Illustration: *E. coli*'s chi II

As a consequence,

- The chi motif is exceptionally frequent in *E. coli*.
- Searching for frequent motifs may help identifying chi motifs in other organisms.

Exercise: Simple application on sequences

- Use the package `seqinr` and rely on function `count` to obtain the counts of words in a given sequence.
- Use the function `zscore` to identify over- or under-represented di-nucleotides.
- Apply this on the following sequences:

```
> # install.packages('seqinr')
> library('seqinr')
> choosebank("emblTP")
> query("myseqs", "sp=felis catus AND t=cds AND
o=mitochondrion") # get a list of sequences names,
here all coding seqs in the cat's mitochondria
> seq1 <- getSequence(myseqs$req[[1]])
> closebank()
```

Some more complex problems

Issues to carefully deal with

- When a word is exceptional, its complement reverse sequence is also exceptional;
- Self-overlapping words are not easy to handle, see [RS07];
- Very often, functional motifs are formed by consensus sequences;

More complex problems

- Search for motifs composed of consensus words separated through some varying distance: PROSITE signatures, gapped motifs, etc
  e.g.
  W.(9-11)[VFY][FYW].(6-7)[GSTNE][GSTQCR][FYM]{R}{SA}P
- Take into account heterogeneity in the sequence through HMM.

Some avalaible tools

- `R'Mes`, is a tool for studying word frequencies in biological sequences. Available at `https://mulcyber.toulouse.inra.fr/projects/rmes/`
- `SPatt`, Statistics for patterns. Available at `stat.genopole.cnrs.fr/spatt`
- PROSITE is a database of protein domains, families and functional sites `http://www.expasy.org/prosite/`
- Regulatory Sequence Analysis Tools is a set of methods for finding motifs in regulatory regions `http://rsat.ulb.ac.be/rsat/`

Exercise: More elaborate example with R'mes

- Use R'mes to find oligonucleotides over- or under-represented.
- Apply this to the previous sequences.

First write a fasta file containing the previous sequence

```
> write.fasta(seq1,names=getName(myseqs$req[[1]]),
  file.out="seq1.fasta")
```

then run rmes in a terminal

```
$ rmes --gauss -s seq1.fasta -l 4 -m 2 -o seq1_res
$ rmes.format < seq1_res.0 > seq1_res_tab
```

Some more references on motifs counts

G. Nuel and B. Prum.
Analyse Statistique des Séquences Biologiques.
Hermes Sciences, 2007.

S. Schbath and S. Robin.
How can pattern statistics be useful for DNA motif discovery?
In Joseph Glaz, Vladimir Pozdnyakov, and Sylvan Wallenstein, editors, *Scan Statistics*, Statistics for Industry and Technology, pages 319–350. Birkhäuser Boston, 2009.

Outline Part 2

Markov chains (order 1)

Higher order Markov chains

Motifs detection with Markov chains

Hidden Markov models (HMMs)

Parameter estimation in HMM

Sequence segmentation with HMM

Motifs detection with HMMs

Heterogeneity and how to deal with it

Heterogeneity in sequences

- For long sequences, a Markov chain model is not adapted: for *e.g.* genes, intergenic regions, CpG islands, *etc*, may not be modeled with the same transition probabilities.
- The usual way to deal with heterogeneity in statistics is to rely on mixtures: assume the observations come from a mixture of say Q different homogeneous groups, but the group of each observation is unknown.
- Hidden Markov models are a generalization of mixtures, where the groups are temporally organised and dependent.

Finite mixture models

Definition

- Finite family of densities $\{f_q; q \in \{1, \ldots, Q\}\}$ (w.r.t. either Lebesgue or counting measure),
- Groups proportions $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_Q)$, such that $\pi_q \geq 0$ and $\sum_{q=1}^{Q} \pi_q = 1$,

The mixture distribution is given by $\sum_{q=1}^{Q} \pi_q f_q$.

Advantages

- Enable modeling heterogeneity in observations: these come from Q unobserved different groups, each group being homogeneous (same distribution f_q)
- parameters π_q represent the unknown groups proportions
- parameters f_q are the distribution within each homogeneous group.

Finite mixture models: an illustration

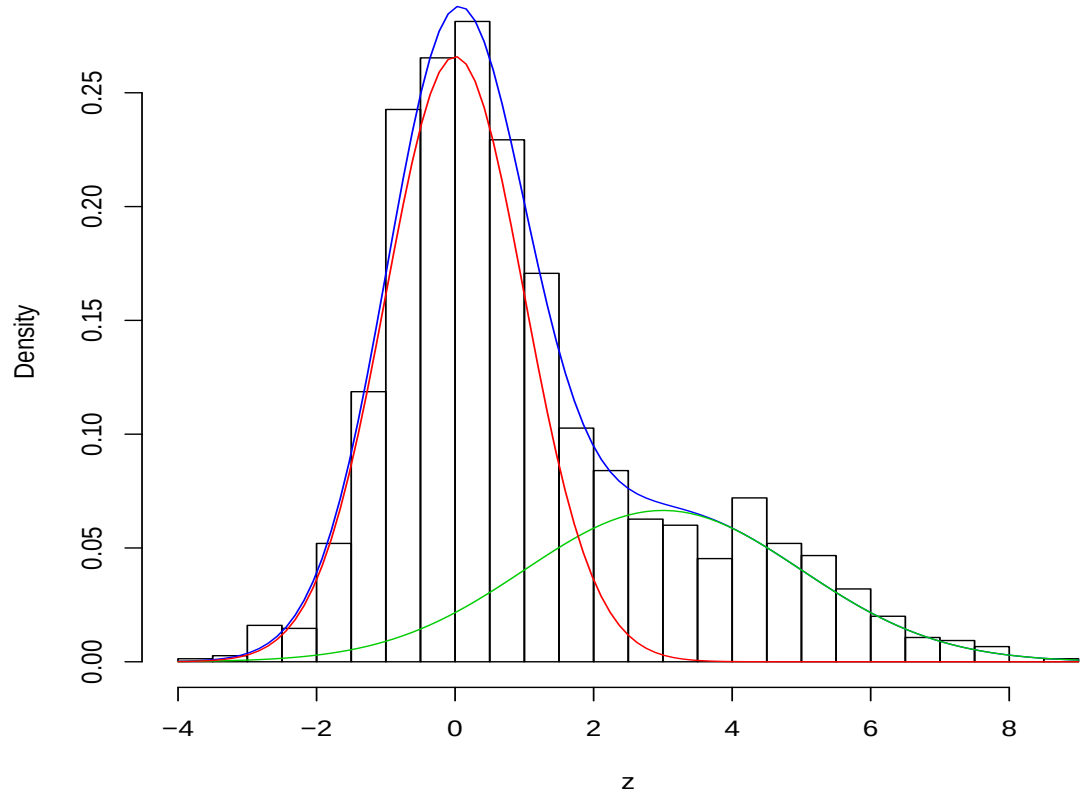


Figure : Histogram of a size $n = 1500$ sample distributed as the mixture $\frac{2}{3}\mathcal{N}(0,1) + \frac{1}{3}\mathcal{N}(3,2)$. Mixture density in blue, group densities appear respectively in red and green.

Finite mixture models: a sub-case of HMM

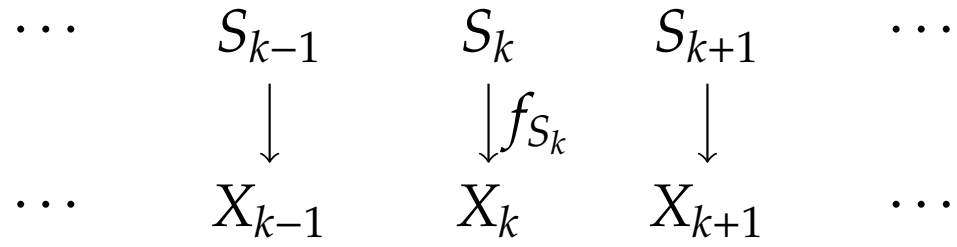
Notation

Let $\{S_k\}_{k\geq 1}$ i.i.d. with values in $\mathcal{S} = \{1, \ldots, Q\}$ with $\mathbb{P}(S_k = q) = \pi_q$ and $\{X_k\}_{k\geq 1}$ s.t., conditional on $S_1, \ldots, S_n$, observations $X_1, \ldots, X_n$ are independent and distribution of each X_k only depends on S_k

$$\mathbb{P}(X_{1:n}|S_{1:n}) = \prod_{k=1}^{n} \mathbb{P}(X_k|S_k), \text{ with density } f_{S_k}.$$

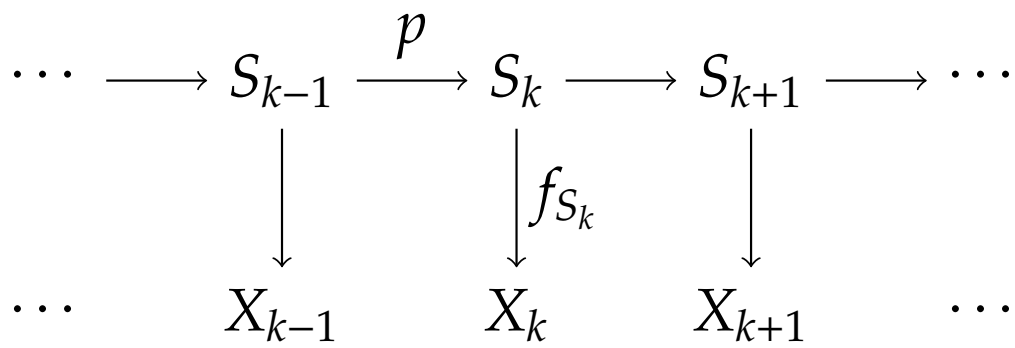
Then, $\{X_k\}_{k\geq 1}$ are i.i.d. with distribution $\sum_{q=1}^{Q} \pi_q f_q$.

Graphical representation



Hidden Markov models (HMMs)

Let us now introduce some dependency between hidden states



(i) $\{S_k\}$ unobserved Markov chain, with values in $\mathcal{S} = \{1, \ldots, Q\}$, transition matrix p and initial distribution π. It is the sequence of regimes,

(ii) $\{X_k\}$ is the sequence of observations, with values in $\mathcal{X}$,

(iii) Conditional on the regimes $S_1, \ldots, S_n$, the observations $X_1, \ldots, X_n$ are independent, with distribution of each X_k depending only on S_k :

$$\mathbb{P}(X_{1:n}|S_{1:n}) = \prod_{k=1}^{n} \mathbb{P}(X_k|S_k), \text{ with density } f_{S_k}.$$

Mixtures vs HMMs

Similarities/Differences

- In HMM, random variables $\{X_k\}_{k\geq 1}$ are not independent anymore (comparing with mixtures).
  $\{X_k\}_{k\geq 1}$ is not a Markov chain either! We say that the sequence has long range dependencies.
- Observations are globally heterogeneous, but they are temporally ordered and the model induces homogeneously distributed zones.
- Estimating hidden states provides a segmentation of the sequence into homogeneously distributed parts.

HMM for analysing sequences

Goals

- Sequence segmentation into different regimes
- For this, it is necessary to fit the model: *i.e.* estimate the parameters $(p, \{f_q\}_{1 \leq q \leq Q})$.

Methods

- Parameter estimation: through maximum likelihood estimation (MLE), leading to expectation-maximization (`EM`) algorithm.
- Sequence segmentation: Viterbi algorithm (widely used but not recommended by me) or stochastic versions of `EM`.

Exercise: Mixtures and HMM data generation

On state space $\mathbb{R}$

- Write an `R` function to generate a mixture model of Q distributions, with group proportions given by parameter $\boldsymbol{\pi}$ and conditional distributions are Gaussian with means $(m_1, \ldots, m_Q)$ and variance 1.
- Do the same but for HMM with Q hidden states, transition matrix $\boldsymbol{p}$ and same conditional distributions.

Finite state space

- Same exercise but with observation state space $\{A, C, G, T\}$.

NB: pay attention to the order of the observations.

Outline Part 2

Markov chains (order 1)

Higher order Markov chains

Motifs detection with Markov chains

Hidden Markov models (HMMs)

Parameter estimation in HMM

Sequence segmentation with HMM

Motifs detection with HMMs

HMM likelihood

Likelihood of the observations

Model parameter $\theta = (\pi, p, \{f_q\}_{1 \leq q \leq Q})$.

$$\ell_n(\theta) := \log \mathbb{P}_\theta(X_{1:n}) = \log \left(\sum_{s_1, \ldots, s_n} \mathbb{P}_\theta(X_{1:n}, S_{1:n} = s_{1:n}) \right).$$

- Computation requires summation over Q^n terms: impossible as soon as n is not small.
- Need to develop another strategy to compute MLE.

Models with incomplete data

- Expectation-Maximization (`EM`) algorithm [DLR77] is an iterative algo. that enables maximising (locally) the likelihood of models with incomplete data when complete data likelihood is simple.

Expectation-Maximization (EM) algorithm I

Let $X_{1:n}$ be observed data and $S_{1:n}$ missing data. We call complete data the set $(S_{1:n}, X_{1:n})$.
We assume that the complete data likelihood $\log \mathbb{P}_\theta(S_{1:n}, X_{1:n})$ is easy to compute.

Principle

- Start with initial parameter value θ^0,
- At k-th iteration, do
  - Expectation-step compute $Q(\theta, \theta^k) := \mathbb{E}_{\theta^k}(\log \mathbb{P}_\theta(S_{1:n}, X_{1:n})|X_{1:n})$.
  - Maximization-step compute $\theta^{k+1} := \text{Argmax}_\theta \, Q(\theta, \theta^k)$.
- Stop whenever $\delta := \|\theta^{k+1} - \theta^k\|/\|\theta^k\| \leq \epsilon$ or some maximal number of iterations is attained.

Expectation-Maximization (EM) algorithm II

Consequences

- At each iteration, the observed data likelihood (not complete data likelihood) increases (proof based on Jensen's Inequality).
- Using many different initialisations, the algorithm will eventually find the global maximiser, *i.e.* MLE.

Heuristics

- The complete data likelihood $\log \mathbb{P}_\theta(S_{1:n}, X_{1:n})$ is unknown because $S_{1:n}$ are not observed.
- At E-step, the quantity $\mathbb{E}_{\theta^k}(\log \mathbb{P}_\theta(S_{1:n}, X_{1:n})|X_{1:n})$ is the conditional expectation of the complete data likelihood, under current parameter value θ^k: this is the best knowledge we have on this complete data likelihood, according to the observations.

EM algo: increase of (observed data) log-likelihood

Proof.

Write that $Q(\theta^{k+1}, \theta^k) \geq Q(\theta^k, \theta^k)$, *i.e* :

$$
\begin{aligned}
0 &\leq \mathbb{E}_{\theta^k}\left[\log \frac{\mathbb{P}_{\theta^{k+1}}(S_{1:n}, X_{1:n})}{\mathbb{P}_{\theta^k}(S_{1:n}, X_{1:n})}\middle| X_{1:n}\right] \\
&\underset{\text{Jensen}}{\leq} \log \mathbb{E}_{\theta^k}\left[\frac{\mathbb{P}_{\theta^{k+1}}(S_{1:n}, X_{1:n})}{\mathbb{P}_{\theta^k}(S_{1:n}, X_{1:n})}\middle| X_{1:n}\right] \\
&= \log \int_{\mathcal{S}^n} \frac{\mathbb{P}_{\theta^{k+1}}(s_{1:n}, X_{1:n})}{\mathbb{P}_{\theta^k}(s_{1:n}, X_{1:n})} \mathbb{P}_{\theta^k}(s_{1:n}|X_{1:n}) ds_1 \dots ds_n \\
&= \log \int_{\mathcal{S}^n} \frac{\mathbb{P}_{\theta^{k+1}}(s_{1:n}, X_{1:n})}{\mathbb{P}_{\theta^k}(X_{1:n})} ds_1 \dots ds_n = \log \frac{\mathbb{P}_{\theta^{k+1}}(X_{1:n})}{\mathbb{P}_{\theta^k}(X_{1:n})}.
\end{aligned}
$$

Thus, $\mathbb{P}_{\theta^{k+1}}(X_{1:n}) \geq \mathbb{P}_{\theta^k}(X_{1:n})$. □

EM algo. in practice

In practice

- Need to perform E-step: compute the complete data log-likelihood $\log \mathbb{P}_\theta(S_{1:n}, X_{1:n})$ and take its conditional expectation w.r.t. observations.
- Need to perform M-step: maximisation of $Q(\theta, \theta^k)$ w.r.t. θ, either analytically (when possible) or numerically (grid search for e.g.).

Exercise: EM algo. for mixture models

- Write the likelihood of a sequence on alphabet $\{A, C, G, T\}$ under a mixture model with Q hidden states.
- Write the e-step of the algorithm and equations necessary to perform it. Take the example of a mixture of Gaussian distributions or discrete r.v. on $\{A, C, G, T\}$.
- Write the m-step of the algorithm.
- Implement it.

EM algo for HMM (Baum-Welch algorithm) I

Complete data likelihood

$$\log \mathbb{P}_\theta(S_{1:n}, X_{1:n}) = \sum_{q=1}^{Q} 1_{S_1=q} \log \pi_q$$

$$+ \sum_{i=2}^{n} \sum_{1 \le q,l \le Q} 1_{S_{i-1}=q, S_i=l} \log p(q,l) + \sum_{i=1}^{n} \sum_{q=1}^{Q} 1_{S_i=q} \log f_q(X_i).$$

Cond. expectation under parameter value θ^k

$$Q(\theta, \theta^k) = \sum_{q=1}^{Q} \mathbb{P}_{\theta^k}(S_1 = q | X_{1:n}) \log \pi_q$$

$$+ \sum_{i=2}^{n} \sum_{1 \le q,l \le Q} \mathbb{P}_{\theta^k}(S_{i-1} = q, S_i = l | X_{1:n}) \log p(q,l)$$

$$+ \sum_{i=1}^{n} \sum_{q=1}^{Q} \mathbb{P}_{\theta^k}(S_i = q | X_{1:n}) \log f_q(X_i).$$

EM algo for HMM (Baum-Welch algorithm) II

Algorithm

- E-step: Need to compute $\mathbb{P}_{\theta^k}(S_i|X_{1:n})$ and $\mathbb{P}_{\theta^k}(S_{i-1}, S_i|X_{1:n})$: done through the forward-backward equations. These are recursive formulas.
- M-step: analytical solution is straightforward: exactly as for MLE for Markov chains, because the complete data $\{(S_k, X_k)\}$ forms a Markov chain.

E-step for HMMs: forward-backward equations

Forward equations: computation of $\alpha_k(\cdot) := \mathbb{P}_\theta(S_k = \cdot, X_{1:k})$

- Initialisation $\forall q,\ \alpha_1(q) := \mathbb{P}_\theta(S_1 = q, X_1) = f_q(X_1)\mu(q)$,
- For any $k = 2, \dots, n$ and any l, $\alpha_k(l) = [\sum_{q=1}^{Q} \alpha_{k-1}(q)p(q,l)]f_l(X_k)$.

Rmk: One may obtain the observations' likelihood as
$\mathbb{P}_\theta(X_{1:n}) = \sum_{q=1}^{Q} \alpha_n(q)$, but then non trivial maximisation step!

Backward equations: computation of $\beta_k(\cdot) := \mathbb{P}_\theta(X_{k+1:n}|S_k = \cdot)$

- Initialisation $\beta_n(\cdot) := 1$,
- For any $k = n, \dots, 2$ and for any q, $\beta_{k-1}(q) = \sum_{l=1}^{Q} f_l(X_k)\beta_k(l)p(q,l)$.

E-step quantities

$\mathbb{P}(S_k = q|X_{1:n}) \propto \alpha_k(q)\beta_k(q)$
and $\mathbb{P}(S_{k-1} = q, S_k = l|X_{1:n}) \propto \alpha_{k-1}(q)p(q,l)f_l(X_k)\beta_k(l)$.

Tool: Directed acyclic graphs (DAGs, [Lau96])

The key to correctly handle conditional expectations is understanding directed acyclic graphs (DAG).

Factorized distributions

Let $\mathcal{V} = \{V_i\}_{1 \leq i \leq N}$ be a set of random variables and $\mathcal{G} = (\mathcal{V}, E)$ a DAG. Distribution $\mathbb{P}$ on $\mathcal{V}$ factorizes according to G if $\mathbb{P}(\mathcal{V}) = \mathbb{P}(V_{1:N}) = \prod_{i=1}^{N} \mathbb{P}(V_i | pa(V_i, \mathcal{G}))$, where $pa(V_i, \mathcal{G})$ is the set of parents of V_i in $\mathcal{G}$.

ex: HMM

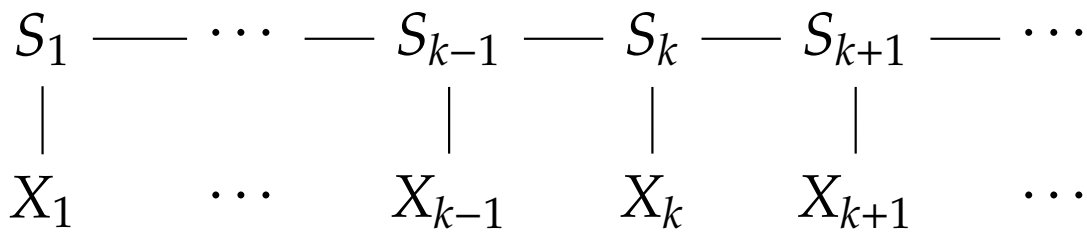
$$\begin{array}{ccccccccc} S_1 & \longrightarrow & \cdots & \longrightarrow & S_{k-1} & \longrightarrow & S_k & \longrightarrow & S_{k+1} & \longrightarrow & \cdots \\ \downarrow & & & & \downarrow & & \downarrow & & \downarrow & & \\ X_1 & & \cdots & & X_{k-1} & & X_k & & X_{k+1} & & \cdots \end{array}$$

$$\mathbb{P}(\{S_i, X_i\}_{1 \leq i \leq n}) = \mathbb{P}(S_1) \times \prod_{i=2}^{n} \mathbb{P}(S_i | S_{i-1}) \times \prod_{i=1}^{n} \mathbb{P}(X_i | S_i).$$

Properties of distributions factorized on graphs

Moral graph

The *moral graph* of a DAG $\mathcal{G}$ is obtained from $\mathcal{G}$ by "marrying" the parents and withdraw directions.
ex : Moral graph associated to a HMM



Independence properties

Let I, J, K subsets of $\{1, \dots N\}$,

- In a DAG $\mathcal{G}$, conditional on its parents, a variable is independent from its non-descendants.
- In the moral graph associated to $\mathcal{G}$, if all paths from I to J go through K, then $\{V_i\}_{i \in I} \perp\!\!\!\perp \{V_j\}_{j \in J} \mid \{V_k\}_{k \in K}$.

Example of application: proof of forward recurrence formula

Forward equations

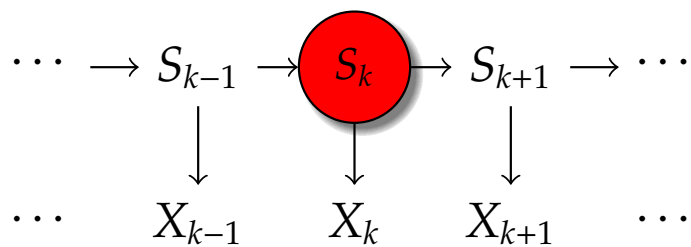
$$\begin{aligned}
\alpha_k(l) &= \mathbb{P}_\theta(S_k = l, X_{1:k}) = \sum_{q=1}^{Q} \mathbb{P}_\theta(S_{k-1} = q, S_k = l, X_{1:k}) \\
&= \sum_{q=1}^{Q} \mathbb{P}_\theta(X_k | S_{k-1} = q, S_k = l, X_{1:k-1}) \mathbb{P}_\theta(S_k = l | S_{k-1} = q, X_{1:k-1}) \mathbb{P}_\theta(S_{k-1} = q, X_{1:k-1}) \\
&= \sum_{q=1}^{Q} f_l(X_k) p(q, l) \alpha_{k-1}(q).
\end{aligned}$$

Example of application: proof of forward recurrence formula

Forward equations

$$\begin{aligned}
\alpha_k(l) &= \mathbb{P}_\theta(S_k = l, X_{1:k}) = \sum_{q=1}^{Q} \mathbb{P}_\theta(S_{k-1} = q, S_k = l, X_{1:k}) \\
&= \sum_{q=1}^{Q} \mathbb{P}_\theta(X_k | S_{k-1} = q, S_k = l, X_{1:k-1}) \mathbb{P}_\theta(S_k = l | S_{k-1} = q, X_{1:k-1}) \mathbb{P}_\theta(S_{k-1} = q, X_{1:k-1}) \\
&= \sum_{q=1}^{Q} f_l(X_k) p(q, l) \alpha_{k-1}(q).
\end{aligned}$$

DAG

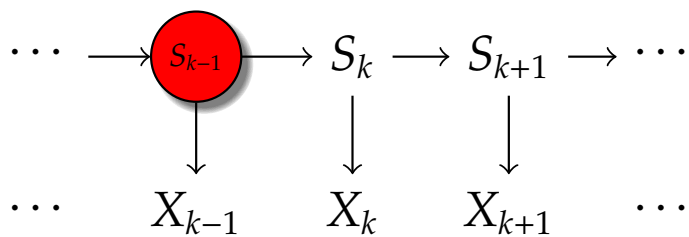


Example of application: proof of forward recurrence formula

Forward equations

$$\begin{aligned}
\alpha_k(l) &= \mathbb{P}_\theta(S_k = l, X_{1:k}) = \sum_{q=1}^{Q} \mathbb{P}_\theta(S_{k-1} = q, S_k = l, X_{1:k}) \\
&= \sum_{q=1}^{Q} \mathbb{P}_\theta(X_k | S_{k-1} = q, S_k = l, X_{1:k-1}) \mathbb{P}_\theta(S_k = l | S_{k-1} = q, X_{1:k-1}) \mathbb{P}_\theta(S_{k-1} = q, X_{1:k-1}) \\
&= \sum_{q=1}^{Q} f_l(X_k) p(q, l) \alpha_{k-1}(q).
\end{aligned}$$

DAG



M-step for HMMs: analytical solution

We want to find

$$\theta^{k+1} = \operatorname*{Argmax}_{\theta} Q(\theta, \theta^k)$$

A maximisation under constraints gives

$$\begin{aligned} p(q,l)^{k+1} &\propto \sum_{i=2}^{n} \mathbb{P}_{\theta^k}(S_{i-1} = q, S_i = l | X_{1:n}) \\ f_q^{k+1}(x) &\propto \sum_{i=1}^{n} \mathbb{P}_{\theta^k}(S_i = q | X_{1:n}) 1_{X_i = x} \end{aligned}$$

Assuming stationarity, one may moreover take

$$\pi^{k+1}(q) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{P}_{\theta^k}(S_i = q | X_{1:n}).$$

EM algo and multiple initialisations

- In practice, it is necessary to run EM with many different starting values θ^0,
- At the end of each EM run, one may obtain the (observed data) log-likelihood as

$$\ell_n(\hat{\theta}) := \log \mathbb{P}_{\hat{\theta}}(X_{1:n}) = \sum_{l=1}^{Q} \hat{f}_l(X_1)\hat{\beta}_1(l)\mathbb{P}_{\hat{\theta}}(S_1 = l).$$

- One finally selects the value $\hat{\theta}$ giving the largest log-likelihood through the different runs.

Exercise: Implement EM algorithm for HMM on a sequence.

Outline Part 2

Markov chains (order 1)

Higher order Markov chains

Motifs detection with Markov chains

Hidden Markov models (HMMs)

Parameter estimation in HMM

Sequence segmentation with HMM

Motifs detection with HMMs

Sequence segmentation I

We now want to reconstruct the sequence of regimes $\{S_k\}$.

Viterbi algorithm

- The most popular method. It consists in finding the maximum a posteriori path

$$\hat{S}_{1:n} = \underset{s_{1:n} \in \mathcal{S}^n}{\text{Argmax}}\, \mathbb{P}_{\hat{\theta}}(X_{1:n}, S_{1:n} = s_{1:n}), \tag{4}$$

  where $\hat{\theta}$ is the solution of `EM`-algorithm.
- Viterbi is an exact recursive algorithm for solving (4).
- Main drawback: unstable w.r.t. sequence length. E.g. remove the last observation, then $\hat{S}_{1:n}$ is completely changed.
- Exercise: More on Viterbi algorithm: in Section 3.2 of [DEKM98] for e.g.

Sequence segmentation II

Alternative solution

At the end of `EM` algorithm, one has access to $\hat{\mathbb{P}}(S_k = q|X_{1:n}) \propto \hat{\alpha}_k(q)\hat{\beta}_k(q)$. Thus, one may consider $\hat{S}_k = \text{Argmax}_{1\leq q\leq Q} \hat{\mathbb{P}}(S_k = q|X_{1:n})$

`SEM` (stochastic `EM`)

An `EM` variant, with 3 steps

- E-step: Compute joint distribution of $\{S_i\}_{i\geq 1}$ conditional on the obs. $\{X_i\}_{i\geq 1}$,under current param. value θ^k, *cf.* Forward-backward equations.
- S-step: Independently draw each $s_i \sim \mathbb{P}_{\theta^k}(S_i = \cdot|X_{1:n})$
- M-step: $\theta^{k+1} = \text{Argmax}_\theta \log \mathbb{P}_\theta(S_{1:n} = s^k_{1:n}, X_{1:n})$

Sequence segmentation III

Consequences

At the end of algo, one recovers an estimate of $\mathbb{P}_{\theta^k}(S_i = \cdot | X_{1:n})$: either consider MAP (maximum a posteriori), or simulate var. under this distribution.

Exercise: Add to your `EM` implementation a sequence segmentation step.

Model selection: choosing the number of hidden states

- Number of hidden states Q may be motivated by the biological pbm. E.g.: gene detection in bacteria, select $Q = 2$ to model coding/non-coding regimes.
- The BIC (Bayesian Information Criterion) is consistent to select the number of hidden states of a HMM

$$\hat{Q} = \text{Argmin}_Q \Big\{ -\log \mathbb{P}_{\hat{\theta},Q}(X_{1:n}) + \frac{N_Q}{2} \log n \Big\},$$

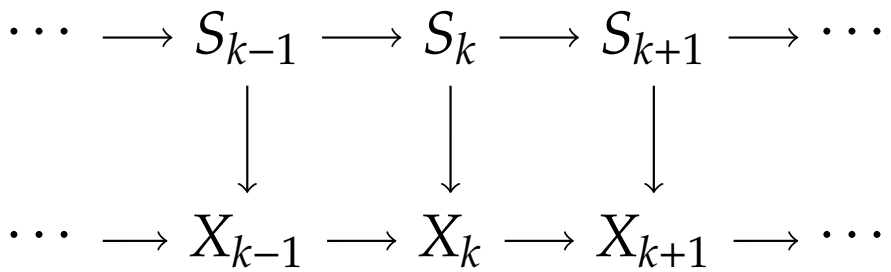
where $N_Q = Q(Q-1) + Q(|\mathcal{A}| - 1)$ is the number of parameters in a HMM with Q hidden states and $\mathbb{P}_{\hat{\theta},Q}(X_{1:n})$ is the corresponding likelihood obtained through `EM` algorithm.

Exercise: Implement a model selection step.

More general HMM

HMM

People regularly use Markov chains with Markov regimes (and call them HMM). Namely, conditional on $\{S_i\}_{i\geq 1}$, the sequence of observations $\{X_i\}_{i\geq 1}$ is an order-k Markov chain, and the distribution of each X_i depends on S_i and $X_{i-k:i-1}$.
Ex : $k = 1$



Outline Part 2

Markov chains (order 1)

Higher order Markov chains

Motifs detection with Markov chains

Hidden Markov models (HMMs)

Parameter estimation in HMM

Sequence segmentation with HMM

Motifs detection with HMMs

Genes detection in bacteria

Ex. from *B. subtilis*, [Nic03, NBM *et al.* 02].

Underlying idea

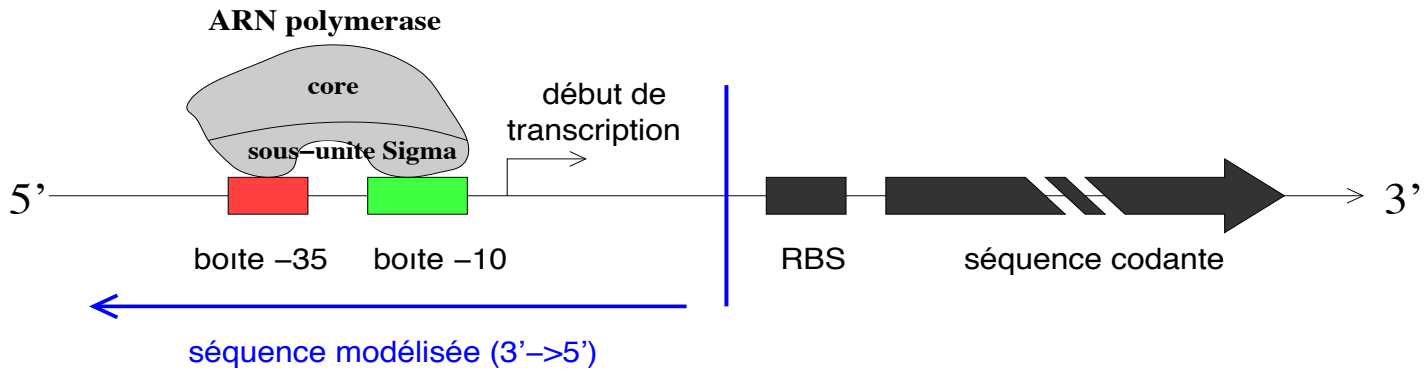
Coding sequences follow a letter distribution that should be different than in non coding sequences: thus, running a HMM with two states (coding/non coding) should enable to detect genes on a sequence.

Genes detection (*B. subtilis*, [Nic03])

Figure : Segmentation of a sequence from *B. subtilis* with 5 hidden states [Nic03]. Posterior distributions on hidden states are close to 0 or 1.GenBank annotation are super-imposed on the sequence.

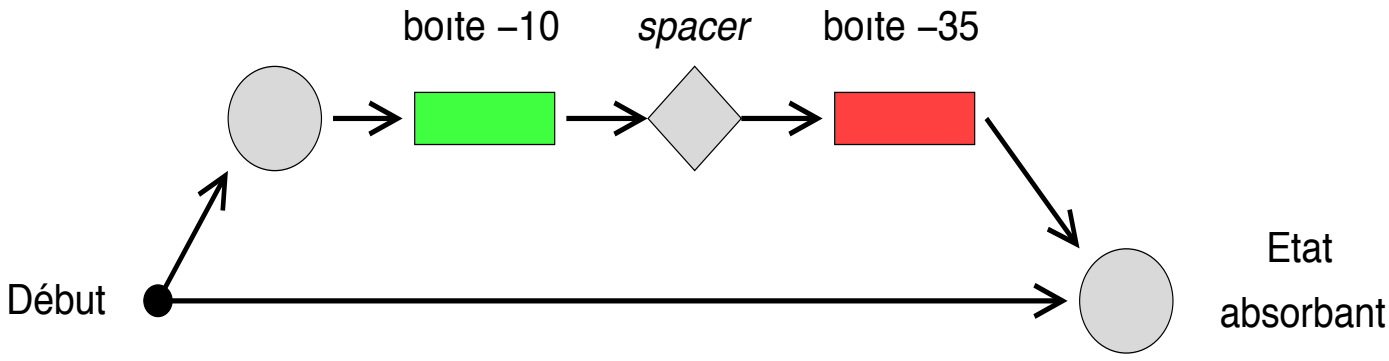
Motifs detection ([Nic03])

Ex: promoter sequence



Ideas

- Constrain your HMM so that it detects structures,
- Use hidden semi-Markov models (HSMM) that generalize HMM to case where homogeneous parts do not have geometric length (implied in HMM case).



Motifs detection ([Nic03])

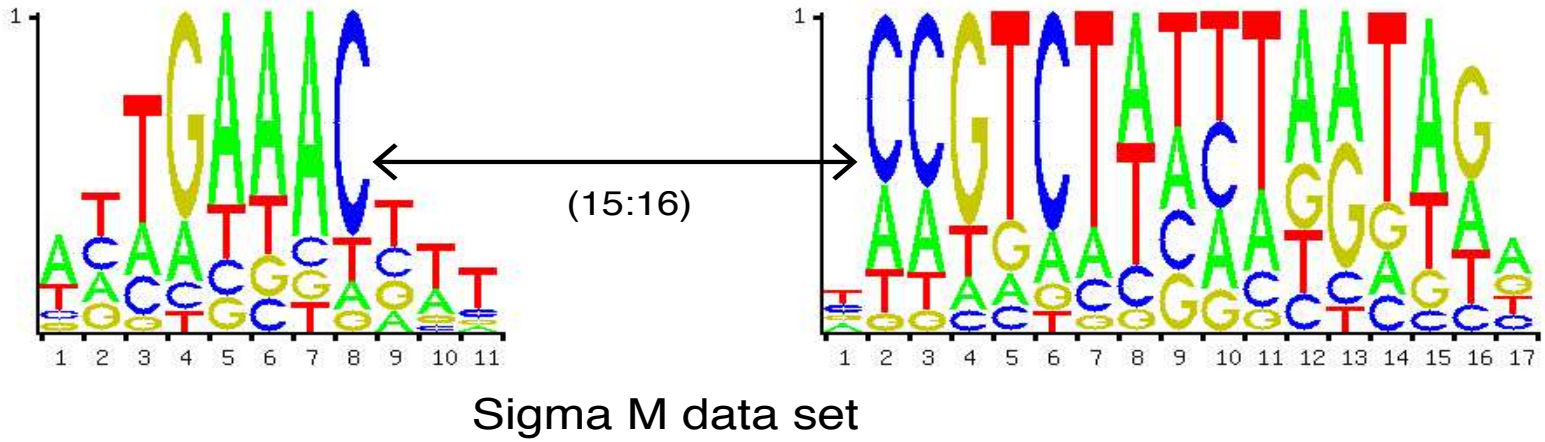


Figure : Exemple of a promoter motif estimated from a sequence of *B. subtilis*.

Part II - References I


[BW99] P. Bühlmann and A.J. Wyner.
Variable length Markov chains.
The Annals of Statistics, 27(2):480–513, 1999.


[CG98] S.F. Chen and J. Goodman.
An empirical study of smoothing techniques for language modeling.
Technical Report TR-10-98, Center for Research in Computing Technology (Harvard University), 1998.


[CS00] I. Csiszár and P. C. Shields.
The consistency of the BIC Markov order estimator.
Ann. Statist., 28(6):1601–1619, 2000.


[DEKM98] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison.
Biological sequence analysis: probabilistic models of proteins and nucleic acids.
Cambridge University Press, Cambridge, UK, 1998.

Part II - References II

- [DLR77] A. P. Dempster, N. M. Laird, and D. B. Rubin.
  Maximum likelihood from incomplete data via the EM algorithm.
  J. Roy. Statist. Soc. Ser. B, 39(1):1–38, 1977.

- [KN95] R. Kneser and H. Ney.
  Improved backing-off for m-gram language modeling.
  In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 181–184, 1995.

- [Lau96] S. L. Lauritzen.
  Graphical models, volume 17 of *Oxford Statistical Science Series*.
  The Clarendon Press Oxford University Press, New York, 1996.
  Oxford Science Publications.

Part II - References III

[NBM *et al.* 02] P. Nicolas, L. Bize, F. Muri, M. Hoebeke, F. Rodolphe, S.D. Ehrlich, B. Prum, and P. Bessières.
Mining bacillus subtilis chromosome heterogeneities using hidden Markov models.
Nucleic Acids Res., 30(6):1418–1426, 2002.

[Nic03] P. Nicolas.
Mise au point et utilisation de modèles de chaînes de Markov cachées pour l'étude des séquences d'ADN.
PhD Thesis, Université d'Évry, France, 2003.

[Nue11] G. Nuel.
Bioinformatics - Trends and Methodologies, chapter Significance Score of Motifs in Biological Sequences.
InTech., 2011.
Mahmood A. Mahdavi (ed.).

Part II - References IV

[RS07] E. Roquain and S. Schbath.
Improved compound Poisson approximation for the number of occurrences of any rare word family in a stationary Markov chain.
Adv. in Appl. Probab., 39(1):128–140, 2007.

Part III

Sequence evolution and alignment

Outline Part 3

Principles of comparative genomics

Sequence evolution
- The basics
- Towards more complex models

Sequence alignment
- Introduction to alignment
- Dotplots
- Alignment through scoring
- Alignment through HMMs (statistical alignment)
- Multiple alignment

Comparative genomics

Definition and procedures

- Measure similarity between sequences.
- Through many different methods
  - alignment (of genes, parts of genomes, complete genomes...). This is the focus of this course,
  - comparison of the order of genes (or domains),
  - comparison of words' sequence composition, ...

Usages

- identification of functional sites,
- functional prediction,
- proteins secondary structure prediction,
- phylogenetic reconstruction, ...

Preliminary remark

- Most of this part focuses on comparing 2 sequences first,
- Then find ways to compare sets of sequences.

What is an alignment? I

- Consider 2 (or more) sequences $X_{1:n}$ and $Y_{1:m}$ with values in the same finite alphabet $\mathcal{A}$.
- Question: are they similar?
- An alignment is a correspondence between the letters of each sequence, respecting the letters' order, and possibly authorizing gaps.

Example

$\mathcal{A} = \{T, C, A, G\}$, $X_{1:9} = GAATCTGAC$, $Y_{1:6} = CACGTA$, *and a (global) alignment of these two sequences*

$$\begin{array}{cccccccccc} G & A & A & T & C & - & T & G & A & C \\ C & A & - & - & C & G & T & - & A & - \end{array}$$

What is an alignment? II

Vocabulary

- Two facing letters are either called a match (if identical), or mismatch (if different), or indifferently (mis)-match,
- a letter facing a gap is called an indel (insertion-deletion) or simply gap.

First remarks

- When the sequences are highly similar, one may consider alignment without gaps.
- Two types of alignment exist
  - **global alignment**: sequence are entirely aligned;
  - **local alignment**: searching for similar portions in the sequences.

Alignment of sequences from *A. tumephaciens* and *M. loti*.

Source : Hobolth, Jensen, JCB, 2005

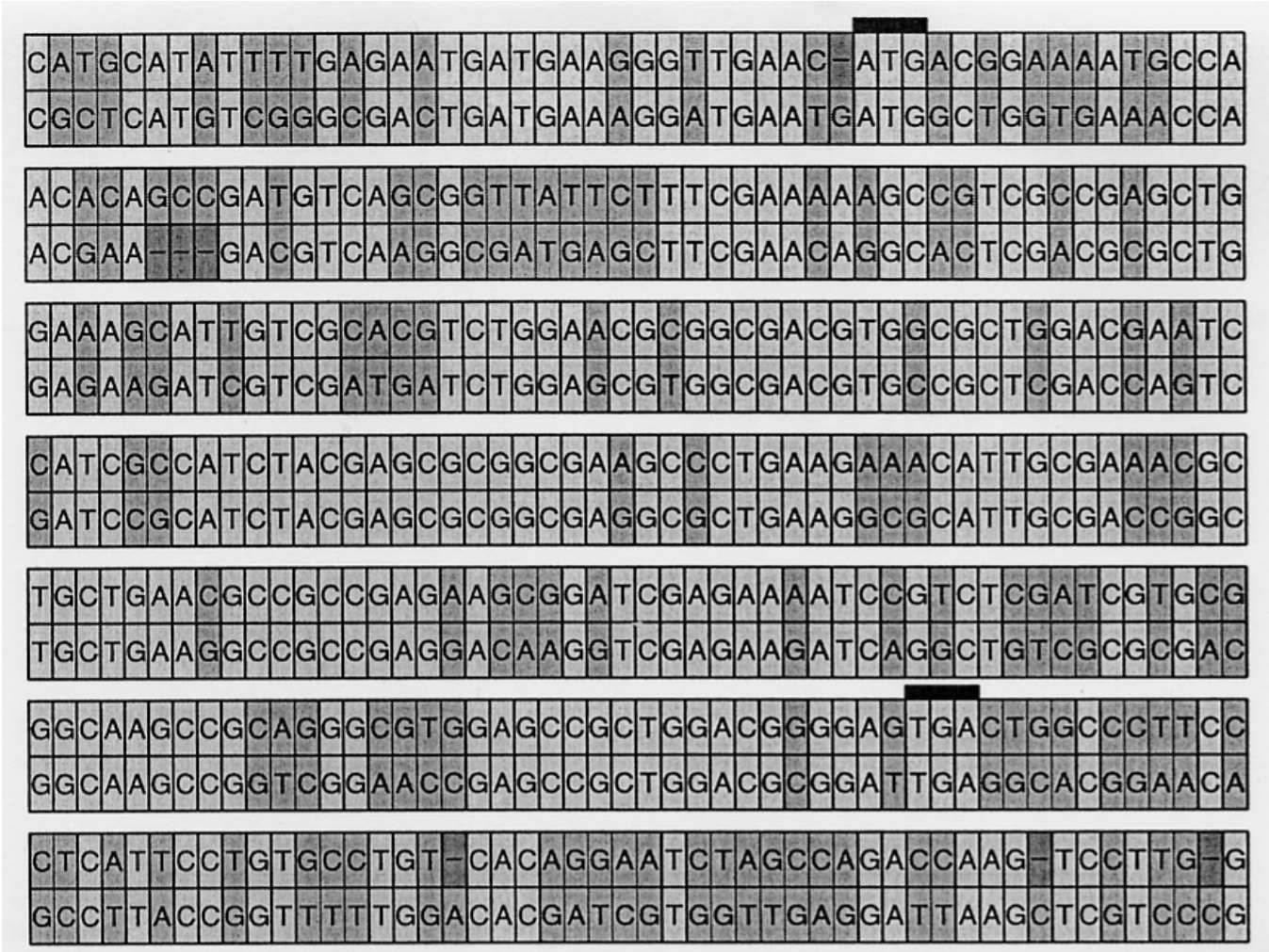


FIG. 3. Part of the pairwise alignment of *A.tumefaciens* and *M.loti*. Light gray color corresponds to conserved positions, and nonconserved positions and gaps are shown in dark gray. The two black bars on top of the alignment

What does an alignment stand for?

- Observed sequences evolved from a common ancestor through some evolutionary process.
- Sequence evolution comprises many different local modifications. Among the most studied one are
  - mutations: a nucleotide (*ie* a letter) is replaced by another,
  - insertions and deletions: one or many nucleotides are inserted or deleted from the sequence.
- There are many other phenomena (duplications, inversions, horizontal transfers, re-arrangements . . .) that we shall not consider here.

An alignment reflects the sequences evolution thus their underlying phylogeny. Alignment and phylogeny are highly intertwinned.

Outline Part 3

Principles of comparative genomics

Sequence evolution
- The basics
- Towards more complex models

Sequence alignment
- Introduction to alignment
- Dotplots
- Alignment through scoring
- Alignment through HMMs (statistical alignment)
- Multiple alignment

Outline Part 3

Principles of comparative genomics

Sequence evolution
- The basics
- Towards more complex models

Sequence alignment
- Introduction to alignment
- Dotplots
- Alignment through scoring
- Alignment through HMMs (statistical alignment)
- Multiple alignment

Some textbooks

O. Gascuel and M. A. Steel, editors.
Reconstructing evolution: new mathematical and computational advances.
Oxford university press, Oxford, 2007.

Z. Yang.
Computational Molecular Evolution.
Oxford Series in Ecology and Evolution. Oxford University Press, 2006.

Models of sequence evolution

Principles

- Only mutations are considered here (no indel, duplications, inversions,. . .).
- The vast majority of models assumes that each site (nucleotide) in the sequence evolves independently and identically to the other sites.
- Continuous time Markov models are used to describe the evolution of each site.
- Mutation parameter and (sometimes) evolutionary distances may be inferred from a set of aligned sequences.

Continuous time Markov models (on alphabet $\mathcal{A}$) I

Definition

A process $X = \{X(t)\}_{t \geq 0}$ is a continuous time (homogeneous) Markov process if for any $t_1 < t_2 < \ldots < t_k < t_{k+1}$ and any $i_1, \ldots i_k, i_{k+1} \in \mathcal{A}^{k+1}$ we have

$$\begin{aligned}\mathbb{P}(X(t_{k+1}) = i_{k+1} | X(t_1) = i_1, \ldots, X(t_k) = i_k) \\ = \mathbb{P}(X(t_{k+1}) = i_{k+1} | X(t_k) = i_k).\end{aligned}$$

Future state only depends on the past through the present.

Continuous time Markov models (on alphabet $\mathcal{A}$) II

Rate matrix

A rate matrix $Q = (q_{ij})_{i,j\in\mathcal{A}^2}$ satisfies

- For $i \neq j$, $q_{ij} \geq 0$ is the instantaneous substitution rate from nucleotide i to j. Thus $q_{ij}\Delta t$ is the probability that nucleotide i is substituted by j in small time interval Δt.
- $q_{ii} = -\sum_{j\neq i} q_{ij}$. The total substitution rate for i is $-q_{ii} > 0$.
- Note that each row of the matrix sums to 0.
- In the following, the states are ordered as T, C, A, G.

Consider an initial probability distribution π on $\mathcal{A}$.
Then, the process $X = \{X(t)\}_{t\geq 0}$ follows a continuous time (homogeneous) Markov distribution with parameters (π, Q) if we have $\mathbb{P}(X(0) = i) = \pi_i$ and $\mathbb{P}(X(t) = j|X(0) = i) = (e^{Qt})_{ij}$

Continuous time Markov models (on alphabet $\mathcal{A}$) III

Remarks

- Note that $P(t) = e^{Qt}$ is a matrix exponential. Its computation requires for e.g. diagonalization of Q.
- Also note that $P_{ij}(t) = (e^{Qt})_{ij}$ is not equal to $e^{Q_{ij}t}$.
- The state of the process at time t is given by $\mathbb{P}(X(t) = j) = \sum_{i \in \mathcal{A}} \pi(i) P_{ij}(t)$, so that

$$\mathbb{P}(X_t = \cdot) = \pi P(t) = \pi e^{Qt}$$

  in matrix notation, where $\mathbb{P}(X(t) = \cdot)$ and π are row vectors.
- Distribution of ancestor sequence may not be estimated, thus one often assumes that π is the stationary distribution associated to Q.
- Replacing Q by Q/λ and t by λt does not change the process. Sometimes Q is normalised s.t. $-\sum_i \pi_i q_{ii} = 1$.

Construction of a continuous time Markov process

It can be shown that the process may be generated on $[0, T]$ as follows

- Start with $t = 0, X(0) \sim \pi = (\pi_T, \pi_C, \pi_A, \pi_G)$,
- While $t \leq T$, iterate
  - Draw $U \sim \mathcal{E}xp(-q_{X(t)X(t)})$ (exponential distr. with mean $-1/q_{X(t)X(t)}$) and
  - update $t \leftarrow t + U$
  - For any $j \in \mathcal{A}$ and such that $j \neq X(t)$, replace $X(t)$ by j with probability $-q_{X(t)j}/q_{X(t)X(t)}$.

One obtains a sequence of mutation times (the t's) and a sequence of states of the process (the $X(t)$'s).

The Jukes Cantor model [JC69]

Jukes Cantor model

Every nucleotide has same rate λ of changing into any other and the stationary distribution π is uniform

$$\pi = \left(\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right) \text{ and } Q = \begin{pmatrix} -3\lambda & \lambda & \lambda & \lambda \\ \lambda & -3\lambda & \lambda & \lambda \\ \lambda & \lambda & -3\lambda & \lambda \\ \lambda & \lambda & \lambda & -3\lambda \end{pmatrix}$$

Exercise: Generate a continuous time Markov process from the Jukes Cantor model.

Maximum likelihood estimation

- A continuous time stationary Markov model is parametrized by: the substitution rates $q_{ij}, i \neq j$ and evolutionary time t, with only the product Qt identifiable.
- With 2 homologous sequences $S^1_{1:n}$ and $S^2_{1:n}$ with same length and thus automatically aligned, the model parameters are estimated through maximum likelihood

$$\begin{aligned}\ell_n(Q,t) &= \sum_{i=1}^{n} \sum_{a,b\in\mathcal{A}} 1\{S_i^1 = a, S_i^2 = b\} \log[\pi_a P_{ab}(t)] \\ &= \sum_{a,b\in\mathcal{A}} N_{ab} \log[\pi_a (e^{Qt})_{ab}],\end{aligned}$$

  where N_{ab} is the number of pairs a, b in the alignment.
- In practice: align sequences and remove gaps from the alignment.

Jukes Cantor model (follow.)

Transition probabilities

It can be shown that

$$P_{ij}(t) = \mathbb{P}(X(t) = j | X(0) = i) = (e^{Qt})_{ij} = \begin{cases} \frac{1}{4} - \frac{1}{4}e^{-4\lambda t} & \text{for } i \neq j, \\ \frac{1}{4} + \frac{3}{4}e^{-4\lambda t} & \text{for } i = j. \end{cases}$$

Note that only the product λt may be estimated without additional information.

Maximum likelihood estimation

Exercise: Write the likelihood of a sequence under JC model and estimate the value λt.

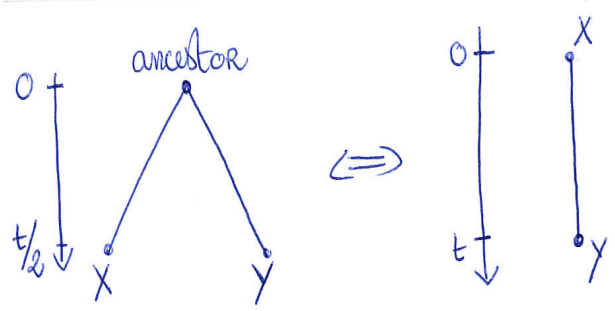
Reversibility of a Markov process

A Markov process is said to be reversible whenever for any $i, j \in \mathcal{A}$, and $t \geq 0$,

$$\pi(i)\mathbb{P}(X(t) = j | X(0) = i) = \pi(j)\mathbb{P}(X(t) = i | X(0) = j)$$
$$\iff \mathbb{P}((X(0), X(t)) = (i, j)) = \mathbb{P}((X(0), X(t)) = (j, i)).$$

Consequence

- The direction of time has no influence on the model
- If two sequences have a common ancestor some time $t/2$ ago it is equivalent to consider that one is the ancestor of the other after a time t of divergence.



Evolutionary distance between 2 sequences under JC I

- Consider 2 homologous sequences $S^1_{1:n}$ and $S^2_{1:n}$ with same length and thus automatically aligned.
- Since JC is reversible, it is equivalent to consider that the sequences have a common ancestor at time $t/2$ or that one evolved from the other with divergence time t.
- Substitution rate is the number of substitutions per time unit. Each nucleotide has total substitution rate $3\lambda = -q_{ii}$.
- Thus the total number of expected substitutions per site should be the evolutionary distance $d = 3\lambda t$.
- The probability that two nucleotides differ $S^1_i \neq S^2_i$ corresponds to

$$\begin{aligned}\mathbb{P}(X(t) \neq X(0)|X(0)) &= 3\mathbb{P}(X(t) = j|X(0) = i) \quad \forall i \neq j \\ &= \frac{3}{4} - \frac{3}{4}e^{-4\lambda t}\end{aligned}$$

Evolutionary distance between 2 sequences under JC II

- Let x be the number of mismatchs in the alignment of $S^1_{1:n}$ and $S^2_{1:n}$. The frequency x/n estimates $\mathbb{P}(X(t) \neq X(0)|X(0))$.
- Finally $x/n = \hat{\mathbb{P}}(X(t) \neq X(0)|X(0))$ gives

$$\widehat{\lambda t} = -\frac{1}{4}\log\left(1 - \frac{4x}{3n}\right)$$

$$\text{and thus} \quad 3\widehat{\lambda t} = \hat{d} = -\frac{3}{4}\log\left(1 - \frac{4x}{3n}\right).$$

NB: the "observed distance" x/n between the two sequences underestimates the "evolutionary distance" d.
We also recover the result from ML estimation!

Variance

Note that one may estimate the variance of $\hat{d}$ as
$\mathbb{V}\text{ar}(\hat{d}) = \frac{x/n(1-x/n)}{n} \times \frac{1}{[1-4x/(3n)]^2}$.

Distinguishing transitions and transversions I

Transitions and transversions

- Substitutions between pyrimidines (T↔C) or between purines (A↔G) are called transitions,
- Substitutions between a pyrimidine and a purine (T,C ↔ A,G) are called transversions.

Kimura (K80) [Kim80]

- α = rate for transitions; β = rate for transversions
- The rate matrix is given by (remember order T, C, A, G)

$$Q = \begin{pmatrix} -(\alpha+2\beta) & \alpha & \beta & \beta \\ \alpha & -(\alpha+2\beta) & \beta & \beta \\ \beta & \beta & -(\alpha+2\beta) & \alpha \\ \beta & \beta & \alpha & -(\alpha+2\beta) \end{pmatrix}$$

- And stationary distribution $\pi = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$

Distinguishing transitions and transversions II

K80 model properties

- Total substitution rate per nucleotide $\alpha + 2\beta$
- Evolutionary distance between sequences separated by time t is $d = (\alpha + 2\beta)t$
- The model is often parametrized through $(d, \kappa = \alpha/\beta)$ instead of $(\alpha t, \beta t)$.
- Let S= proportion of transitions between two aligned sequences and V= proportion of transversions. Then

$$\hat{d} = -\frac{1}{2}\log(1 - 2S - V) - \frac{1}{4}\log(1 - 2V)$$

$$\hat{\kappa} = \frac{2\log(1 - 2S - V)}{\log(1 - 2V)} - 1$$

- Formulas for variances can also be given.

Other famous models

- JC and K80 have symmetrical rates $q_{ij} = q_{ji}$ and thus stat. dist. π is uniform. This is unrealistic.
- [HKY85]: parameters are stationary distribution $\pi = (\pi_T, \pi_C, \pi_A, \pi_G)$, transition rate α and transversion rate β.
- Felsentein (F84), Tamura & Nei [TN93] are further generalisations
- ...

General Time Reversible model (GTR)

- All previous models are reversible
- The most general reversible Markov model has stationary distribution $\pi = (\pi_T, \pi_C, \pi_A, \pi_G)$ and rate matrix

$$Q = \begin{pmatrix} \star & a\pi_C & b\pi_A & c\pi_G \\ a\pi_T & \star & d\pi_A & e\pi_G \\ b\pi_T & d\pi_C & \star & f\pi_G \\ c\pi_T & e\pi_C & f\pi_A & \star \end{pmatrix}$$

  where $\star$ are terms such that rows sum to 0.
- This model has 6+3=9 parameters.
- Reversible models are useful as they simplify phylogeny computations. However they are not biologically funded.

More evolutionary distances I

- The analysis to derive evolutionary distance from model parameters is not always easy as for JC and K80 models.
- The general formula relying d and the model parameters for GTR is

$$d = -2\text{trace}(\Pi Q t),$$

  where $\Pi = \text{diag}(\pi)$.
- Let F be the $|\mathcal{A}| \times |\mathcal{A}|$ matrix containing the frequencies $F_{ij} = N_{ij}/N$ in the alignment. Here N_{ij} is the number of times i in first seq is aligned with j in second seq and $N = \sum_{ij} N_{ij}$. Then F estimates $\Pi \exp(Q2t)$ (note the factor 2!).
- A consequence is that one may estimate

$$\hat{d} = -\text{trace}(\hat{\Pi} \log(\hat{\Pi}^{-1} F)),$$

  where $\hat{\Pi}$ contains the observed nucleotides frequencies in the sequences. NB: there is a matrix logarithm!

More evolutionary distances II

Log-det and paralinear distances

- The log-det distance [Ste94] is given by

$$\hat{d} = -\log\det(F).$$

- The paralinear [Lak94] distance is given by

$$\hat{d} = -\frac{1}{4}\Big[\log\det(F) - \frac{1}{2}\log(\det(\hat{\Pi}_x\hat{\Pi}_y))\Big],$$

where $\hat{\Pi}_x$ (resp. $\hat{\Pi}_y$) is the frequency of letters in the first (resp. second) sequence.

NB: no matrix logarithm there!

Exercise: Simulate 2 sequences evolved from a GTR model. Compute their different distances (JC, K80, GTR formula, log-det, paralinear) and compare them.

Outline Part 3

Principles of comparative genomics

Sequence evolution
- The basics
- Towards more complex models

Sequence alignment
- Introduction to alignment
- Dotplots
- Alignment through scoring
- Alignment through HMMs (statistical alignment)
- Multiple alignment

Rates variation across sites I

Γ distributed rate heterogeneity [Yan94]

- Sites are heterogeneous, *e.g.* some sites are more conserved and evolve more slowly;
- Introduce a rate parameter per site r, such that instantaneous substitution rates are given by rQ (Q is a transition matrix common to all sites);
- Recall Gamma distribution: two parameters α (shape) and β (scale) with density

$$g(r;\alpha,\beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} e^{-\beta r} r^{\alpha-1}, \quad r > 0;$$

- Assume that $r \sim \Gamma(\alpha,\alpha)$ (set $\alpha = \beta$ otherwise time could be rescaled with no change). This induces one extra shape parameter α (besides parameters of Q).
- In practice, many implementations of the model use a discretized version of the Gamma distribution.

Rates variation across sites II

Invariant sites

- Some sites never vary (under some strong evolutionary constraints)
- Introduce a latent variable per site I, with values in $\{0, 1\}$ and such that if $I = 0$ then the site is fixed, otherwise it follows the substitution model;
- This corresponds to a mixture model with two groups: invariant and non-invariant sites;

GTR + Γ +I

- One of the most widely used models of nucleotide substitution.

Exercise: Generate sequences evolving under this model.

Relaxing independence between sites

- Different attempts have been made to relax the independence assumption between the sites,
- In practice, these models remain largely untractable at the moment,
- But this might change in the near future.
- A pretty good attempt is given by the model [BGP08]. See also [BG12, Fal10, FB12].

Main issue: cone of dependencies

When looking backwards in time, the dependencies at a specific site propagate along a cone.

Outline Part 3

Principles of comparative genomics

Sequence evolution
- The basics
- Towards more complex models

Sequence alignment
- Introduction to alignment
- Dotplots
- Alignment through scoring
- Alignment through HMMs (statistical alignment)
- Multiple alignment

Outline Part 3

Principles of comparative genomics

Sequence evolution
- The basics
- Towards more complex models

Sequence alignment
- Introduction to alignment
- Dotplots
- Alignment through scoring
- Alignment through HMMs (statistical alignment)
- Multiple alignment

Graphical representation of a pairwise alignment I

- An alignment between two sequences with length n and m = a path on the grid $[0,n] \times [0,m]$ constrained to three different steps types: $(1,1), (1,0)$ and $(0,1)$.
- steps $(1,1)$ correspond to *matchs* or *mismatchs*
- steps $(1,0)$ and $(0,1)$ correspond to *indels*

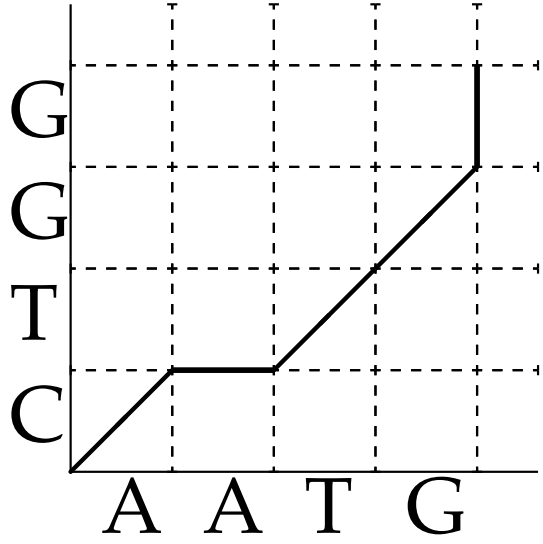


Figure : Graphical representation of an alignment between $X = AATG$ and $Y = CTGG$. This alignment corresponds to $\begin{matrix} A & A & T & G & - \\ C & - & T & G & G \end{matrix}$.

Graphical representation of a pairwise alignment II

Correspondence

- a global alignment = a path starting at $(0, 0)$ and ending at (n, m),
- local alignment = any constrained path included in $[0, n] \times [0, m]$.
- *Nota Bene*: the "best" global alignment does not necessarily contain the best local alignment.

Graphical representation of a pairwise alignment III

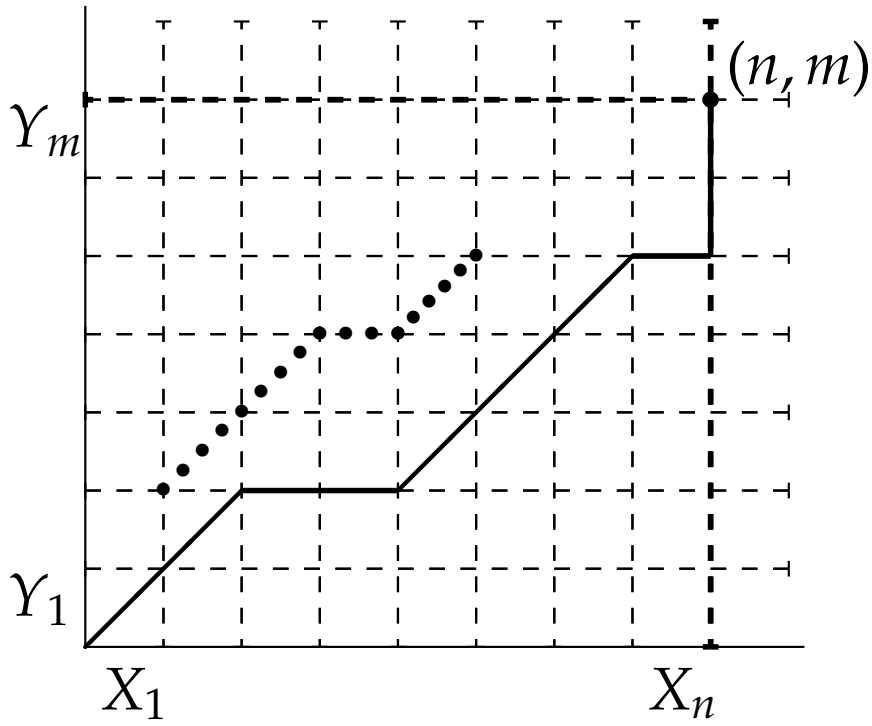


Figure : Graphical representation of best global (solid line) and best local (dashed line) alignments of $X_{1:n}$ and $Y_{1:m}$.

Outline Part 3

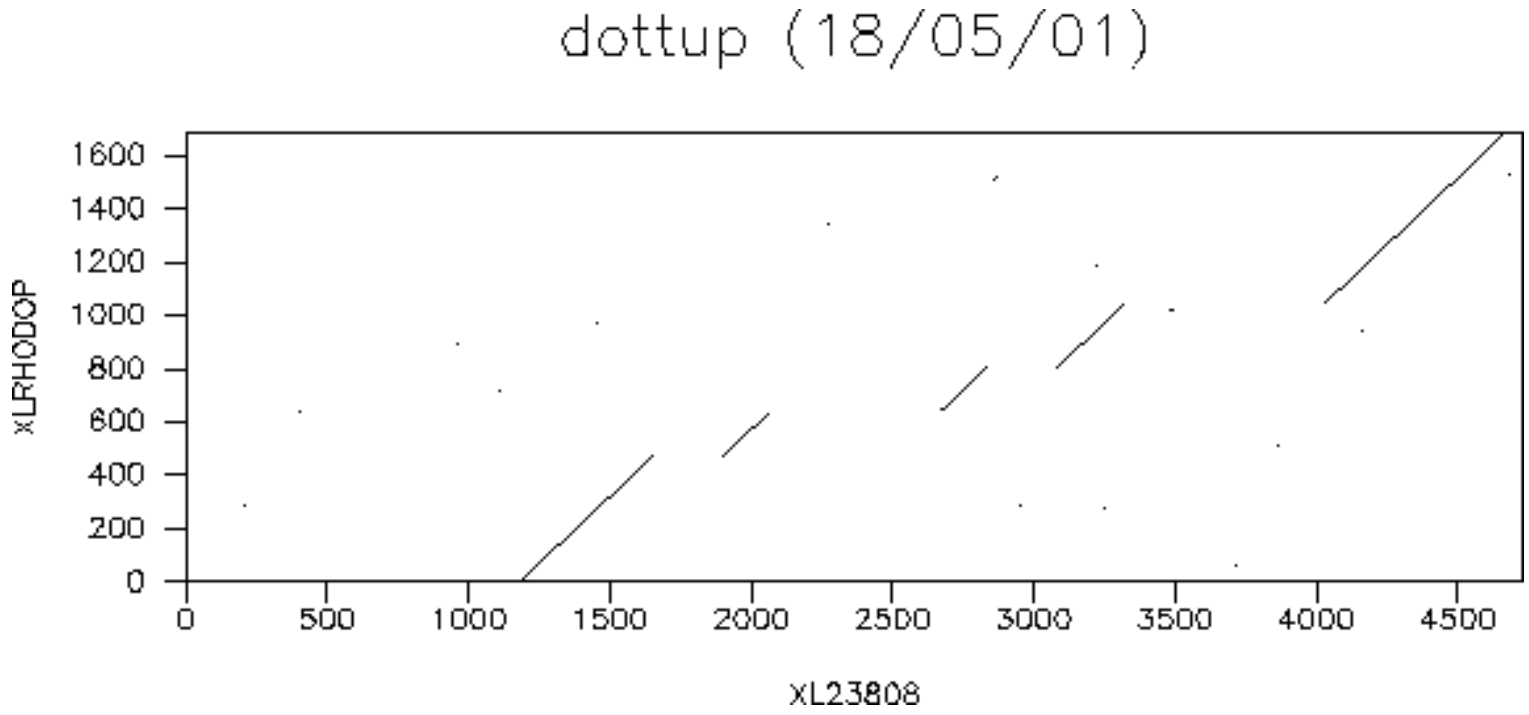
Principles of comparative genomics

Sequence evolution
- The basics
- Towards more complex models

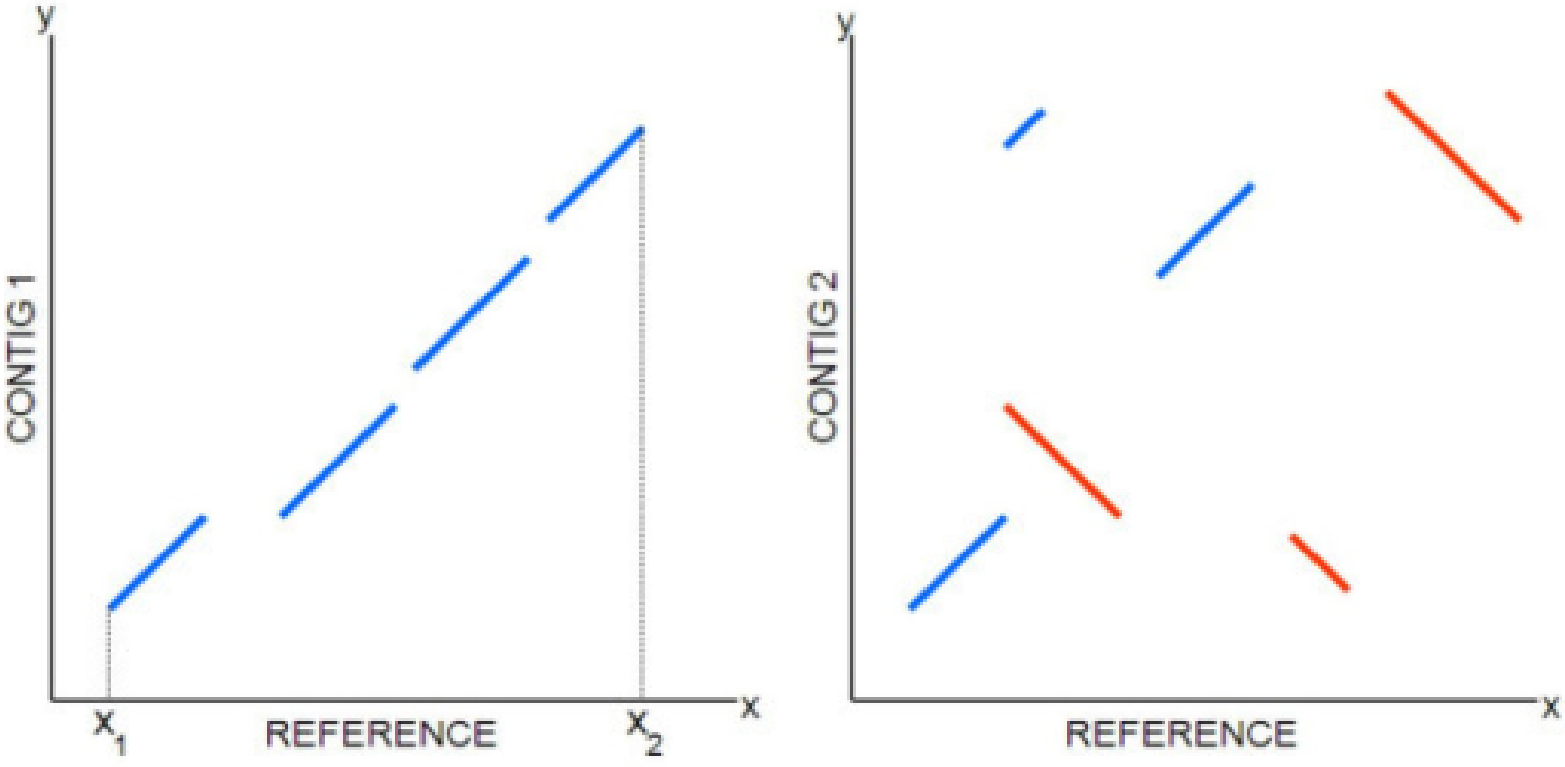
Sequence alignment
- Introduction to alignment
- Dotplots
- Alignment through scoring
- Alignment through HMMs (statistical alignment)
- Multiple alignment

Simplest comparison: the dotplot I

The two sequences $X_{1:n}$ and $Y_{1:m}$ are on the two axes, similarities are highlighted with dots.



Simplest comparison: the dotplot II



- Visual comparison only,
- Simplest dotplots show identities only.
- Dotmatcher tool from Emboss : Scores are computed over fixed size windows and thresholds are used.

Simplest comparison: the dotplot III

Exercise: Use the dotmatcher tool `http://emboss.bioinformatics.nl/cgi-bin/emboss/dotmatcher` on two chosen sequences (obtained for e.g on Genbank). Change the parameter values. Observe the results.

Outline Part 3

Principles of comparative genomics

Sequence evolution
- The basics
- Towards more complex models

Sequence alignment
- Introduction to alignment
- Dotplots
- Alignment through scoring
- Alignment through HMMs (statistical alignment)
- Multiple alignment

Alignment with scores

Principle

- associate a score to each alignment, high scores corresponding to most likely alignments,
- select the alignment with highest score.

As a consequence, one needs to be able to

- compute the score of all possible alignments;
- explore the set of alignments in an efficient way so as to select the best one.

Which scoring functions?

- "Site by site" scoring functions, that attribute to an alignment the sum of individual score of each step in this alignment,
- e.g: $+1$ for a match, $-\mu$ for a mismatch and $-\delta$ for an indel ($\mu, \delta > 0$).
- More generally, consider a scoring matrix on $\mathcal{A} \times \mathcal{A}$ that gives individual score $s(a, b)$ to a position where a stands in front of b,
- Linear or affine penalisation of indel lengths is used: $-\Delta - \delta k$ with k equal to indel length. Here, $\Delta \geq 0$ is the gap opening penalty and $\delta > 0$ is the gap widening penalty.

Note that relying on an additive scoring function corresponds to assuming that sites evolution is independent (very rough assumption).

Remarks

- There is a balance to achieve between (mis)-match scores and indel scores. This has a strong influence on the resulting alignments.
- The optimal score naturally increases with sequence length: two phases appear, linear and logarithmic with respect to sequence length.
- The logarithmic regime is the interesting one.
- The space of alignments is huge thus searching for an optimal alignment is not easy. However, existence of dynamic programming algorithms solves the problem.

Exact algorithms I

- Needleman & Wunsch for global alignment [NW70], later improved by Gotoh [Got82].
- Smith & Waterman [SW81] for local alignment.
- Both are based on dynamic programming (thus rely on additive form of the score).

Principle

At each step in the alignment, 3 possibilities arise. Next step can either be

- a letter from X facing a letter from Y;
- a letter from X in front of an indel;
- a letter from Y in front of an indel.

From these 3 possibilities, keep the one that maximises the score (= preceding score + current cost) and go on.

Exact algorithms II

Dynamic programing - global alignement - linear penalty

Let $F(i,j)$, the optimal (global) alignment score between $X_{1:i}$ and $Y_{1:j}$. Construct matrix F recursively:

- $F(0,0) = 0$, $F(i,0) = -\delta i$ and $F(0,j) = -\delta j$,
- $$\begin{array}{ccc} F(i-1,j-1) & & F(i-1,j) \\ & \searrow & \downarrow \\ F(i,j-1) & \rightarrow & F(i,j) \end{array}$$
- $$F(i,j) = \max \begin{cases} F(i-1,j-1) + s(X_i, Y_j) \\ F(i-1,j) - \delta \\ F(i,j-1) - \delta \end{cases}$$

Complexity: $O(nm)$ (time and memory).

Exact algorithms III

Dynamic programing - local alignment - linear penalty

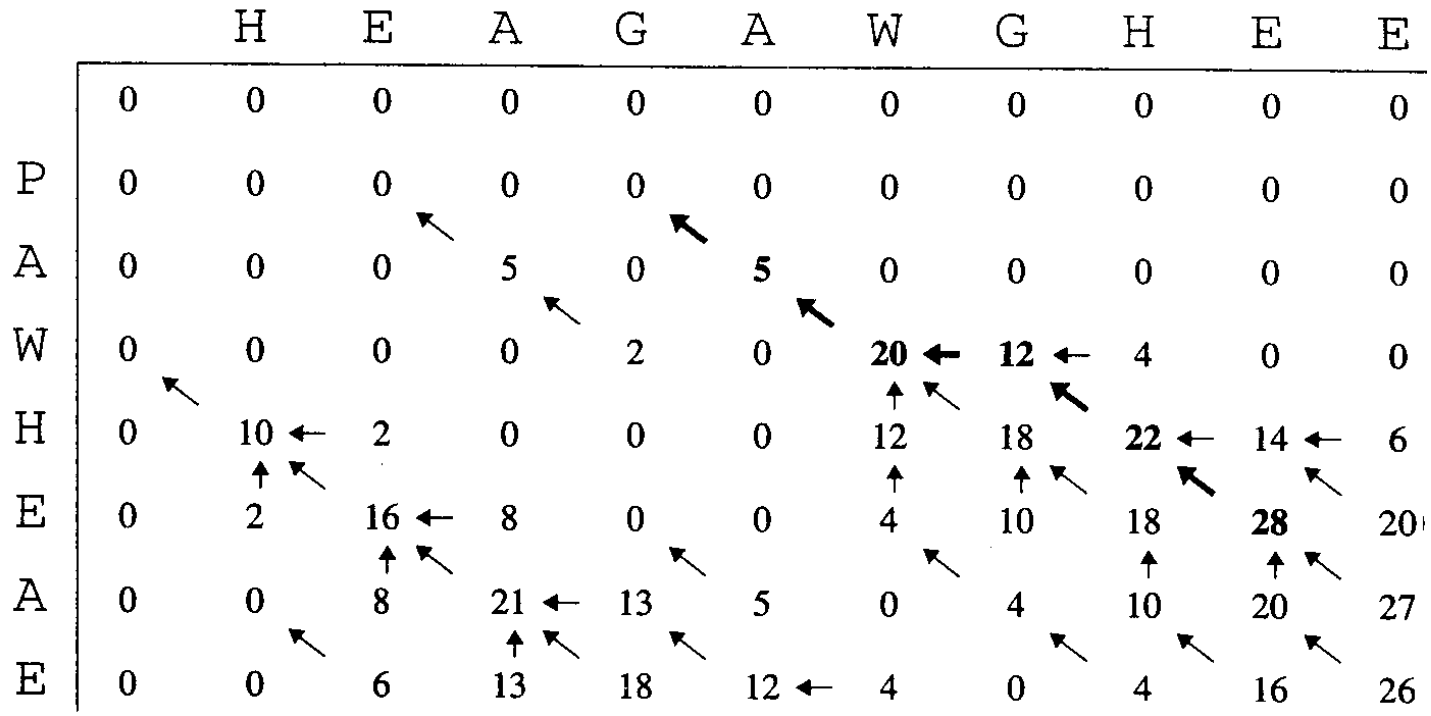
Let $F(i,j)$, the optimal (local) alignment score between $X_{1:i}$ and $Y_{1:j}$. Construct matrix F recursively:

- $F(0,0) = F(i,0) = F(0,j) = 0$,
- $$\begin{array}{ccc} F(i-1,j-1) & & F(i-1,j) \\ & \searrow & \downarrow \\ F(i,j-1) & \rightarrow & F(i,j) \end{array}$$
- $$F(i,j) = \max \begin{cases} 0 \\ F(i-1,j-1) + s(X_i, Y_j) \\ F(i-1,j) - \delta \\ F(i,j-1) - \delta \end{cases}$$

Complexity: $O(nm)$ (time and memory).
For more details, see [DEKM98].

Exact algorithms IV

(Source Durbin *et al.* [DEKM98])



		H	E	A	G	A	W	G	H	E	E
	0	0	0	0	0	0	0	0	0	0	0
P	0	0	0	0	0	0	0	0	0	0	0
A	0	0	0	5	0	**5**	0	0	0	0	0
W	0	0	0	0	2	0	**20**	**12**	4	0	0
H	0	10	2	0	0	0	12	18	**22**	14	6
E	0	2	16	8	0	0	4	10	18	**28**	20
A	0	0	8	21	13	5	0	4	10	20	27
E	0	0	6	13	18	12	4	0	4	16	26

AWGHE

AW-HE

Figure 2.6 *Above, the local dynamic programming matrix for the example sequences. Below, the optimal local alignment, with score* 28.

Approximate algorithms

- Smith & Waterman's algo is too slow to compare a sequence to a whole database.
- Heuristics have been developed to fasten these procedures, for instance by first searching for identical segments (anchor points) and extend the alignment from these parts;
- These heuristics are implemented in BLAST, FASTA...

Exercise: Write an R function for computing the local alignment between two sequences.

Substitution matrices I

- Choice of $s : \mathcal{A} \times \mathcal{A} \to \mathbb{R}$ is an issue. [It's also the case for indels costs, but current algo are limited to cost functions affine w.r.t. indel size].
- For $\mathcal{A} = \{A, T, G, C\}$, one often uses Identity matrix, or two different values: $s(X, X) = s(Y, Y) \neq s(X, Y)$ depending on functional groups purines $X = \{A, G\}$ / pyrimidines $Y = \{C, T\}$.
- For $\mathcal{A} = \{\text{amino acids}\}$ (size 20), there exist two main families of substitution matrices
  - PAM ("Percent Accepted Mutations"), see [DSO78].
  - BLOSUM ("Blocks Substitution Matrix"), see [HH92].
  - They both are based on log-odds ratios principle, but constructed on different datasets.

Substitution matrices II

Log-odds ratios

- Take a family of proteins that have been manually aligned, and whose evolutionary distance is rather well known.
- Obtain $s(a,b) = \log \frac{p_{ab}}{q_a q_b}$ where q_a is frequency of a in the dataset, and $p_{a,b}$ frequency of (a,b) in the alignment.
- A whole family of substitution matrices is then obtained by introducing a scale factor that accounts for different evolutionary distances between sequences.
- It works if the set of sequences under consideration has the "same characteristics" as the original dataset.

Alternative

An alternative to the choice of the scoring function is given by statistical alignment, that corresponds to select scoring functions from data through maximum likelihood.

Linear vs logarithmic regime

- For local alignments, it may be shown that a phase transition occurs when the parameters vary, between a linear increase of the maximal local score w.r.t. sequence lengths and a logarithmic increase;
- The logarithmic regime is the interesting one; otherwise long alignments would tend to have high score independently of whether the segments aligned were related;
- For local scores without indels, this is ensured as long as the expected score for aligning a random pair is negative; *i.e.* $\mathbb{E}(s(X, Y)) < 0$.

Statistical significance of an alignment I

Statistical context

- Test the null hypothesis H_0 : ”the two sequences are independent” versus the alternative H_1 : ”the two sequences evolved from a common ancestor”.

Hypothesis testing

- If the alignement score between two sequences is very large, then the sequences are thought to be highly similar and the null hypothesis is rejected: the alignment is considered to be significant.
- Relies on the knowledge of the tail distribution of the score under the null hypothesis.

Statistical significance of an alignment II

Pitfalls

- The distribution of optimal alignments under null hypothesis is not known;
- One may generate many independent sequences pairs with appropriate length and sequence composition and compute their optimal score and estimate mean value and standard deviation. Then compute a "z-score";
- However this does not give a p-value because the distribution of z-score is not Gaussian;
- There is a multiple testing issue: when testing 1000 hypotheses, an individual type I error of 10^{-4} is required to guarantee an overall type I error less than 0.1 (Bonferroni correction).

Distribution of the score under the null hypothesis I

The "without indel" case

- In this case, the distribution of the maximal local score is analytically well understood;
- It follows a Gumbel distribution (extreme value distribution), with parameters that may be estimated;
- E-value(S): is defined as the expected number of high-scoring segments pairs with score at least S (Often used by programs when p-values unknown);
- In this case, E-value$(S) = Kmne^{-\lambda S}$, where K, λ are parameters depending on the scoring values and m, n are the sequences lengths;

Distribution of the score under the null hypothesis II

General case (with indels)

- In general, the tail distribution of the maximal score (local or global) is unknown;
- *E*-values and *p*-values produced by alignment tools are based on roughs approximations;
- Moreover, a multiple testing issue arises: when searching a whole database for sequence similarity, one makes thousands of tests. Alignment tools have specific corrections of *E*-values and *p*-values w.r.t. database sizes.

Conclusions on alignment with scoring functions

- Highly dependent on the choice of the scoring function;
- Statistical significance is only roughly evaluated.

Developing alternatives

- with adaptive choice of the scores
- with better established significance statistics

is highly desirable.

Exercise: Varying the parameters of an alignment

Use the tools
`http://emboss.bioinformatics.nl/cgi-bin/emboss/needle`
and
`http://emboss.bioinformatics.nl/cgi-bin/emboss/water`
for (exact) global and local alignments of two sequences.
Modify the parameters and observe the results.

Outline Part 3

Principles of comparative genomics

Sequence evolution
- The basics
- Towards more complex models

Sequence alignment
- Introduction to alignment
- Dotplots
- Alignment through scoring
- Alignment through HMMs (statistical alignment)
- Multiple alignment

Context

Scoring alignment vs statistical alignment

- Good scoring functions should be derived from the knowledge of the evolutionary processes at stake. A priori choosing these induces a bias.
- Statistical alignment deals with this issue by achieving at the same time both sequence alignment and parameter estimation of the underlying evolutionary process.
- In practice, this relies on maximum likelihood estimation in a pair-hidden Markov model.

Introduction to statistical alignment

Principle

- We consider a specific evolutionary model (substitution + indel process) and observe 2 seqs.
- Try to reconstruct the homologous positions *i.e.* sites that evolve from a common ancestor, by maximising the likelihood of the sequences under the model.

Framework

- Models combining substitutions + indel processes where first introduced by Thorne, Kishino and Felsenstein [TKF91, TKF92], with many different generalisations (e.g. [MLH04, AGMP09] ...).
- This specific class of models is contained in the larger class of pair-HMM. Probabilistic framework.
- Many advantages: parameter inference is possible, but also hypothesis testing ...

TKF model I

Evolutionary model

- Each site evolves independently. Two independent processes apply on each site: a reversible substitution process (any of those previously described)+ an indel one.
- Each site may be deleted (with some rate μ) and an insertion may happen between two sites (with rate λ).
- The whole resulting process is reversible.

Consequences (1/2)

- Each alignment between two sequences may be coded through a sequence with values in $\{H, D, I\}$ indicating which positions are homologous H, i.e. matchs/mismatchs), deleted (D) in the first sequence or inserted in (I) the first sequence.

TKF model II

Consequences (2/2)

- Under the above setup, the sequence $W_{1:L}$ with $W_i \in \{H, D, I\}$ that encodes the evolution between the two sequences follows a Markov distribution. Here, L is the length of the 'true' alignment between the sequences.
- Conditional on this sequence $W_{1:L}$, the model draws independently the letters in the two sequences $\longrightarrow$ Pair-HMM.

Pair-hidden Markov model I

Reminder: Graphical representation of an alignment

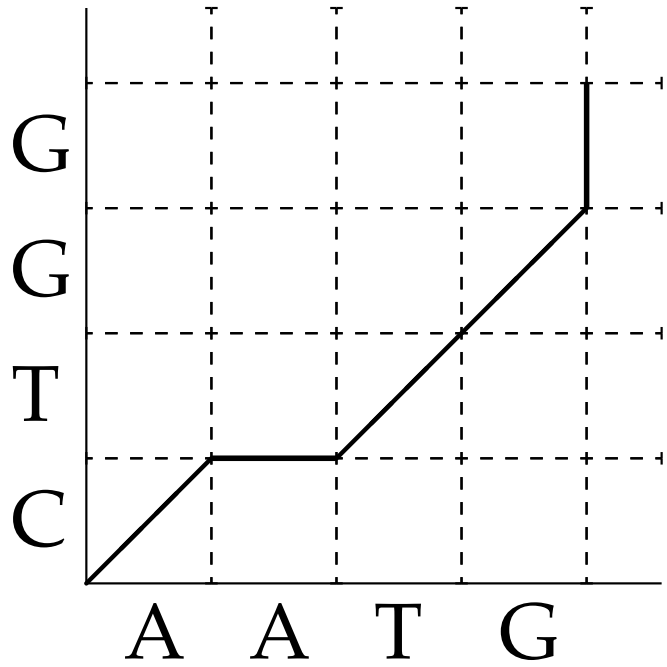


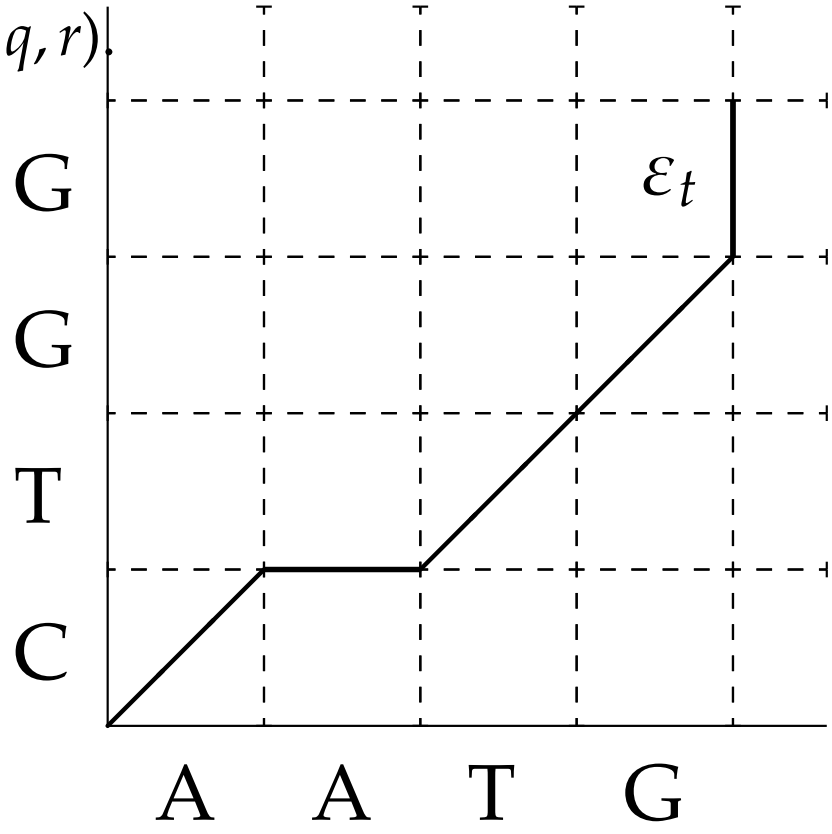
Figure : Graphical representation of an alignment between 2 sequences $X = AATG$ and $Y = CTGG$. The alignment corresponds to $\begin{matrix} A & A & T & G & - \\ C & - & T & G & G \end{matrix}$.

Pair-hidden Markov model II

Notation [AGGM06]

- $\mathcal{A}$ finite alphabet (e.g. $\{A, C, G, T\}$).
- $\{\varepsilon_t\}_{t\geq 1}$ stationary and ergodic Markov chain on state space $\mathcal{E} = \{(1,0);(0,1);(1,1)\}$, with transition matrix π and stationary distribution $\mu = (p, q, r)$.

- At time t, conditional on $\{\varepsilon_s, s \leq t\}$ draw independently
  - A pair (X, Y) with law h on $\mathcal{A} \times \mathcal{A}$, whenever $\varepsilon_t = (1,1)$,
  - A letter X with law f on $\mathcal{A}$ whenever $\varepsilon_t = (1,0)$,
  - A letter Y with law g on $\mathcal{A}$ whenever $\varepsilon_t = (0,1)$.



Pair-hidden Markov model III

- $\theta = (\pi, f, g, h) \in \Theta$ is the model parameter
- Let $Z_0 = (0,0)$ and $Z_t = (N_t, M_t) = \sum_{s=1}^{t} \varepsilon_s$, the random walk on $\mathbb{N} \times \mathbb{N}$.

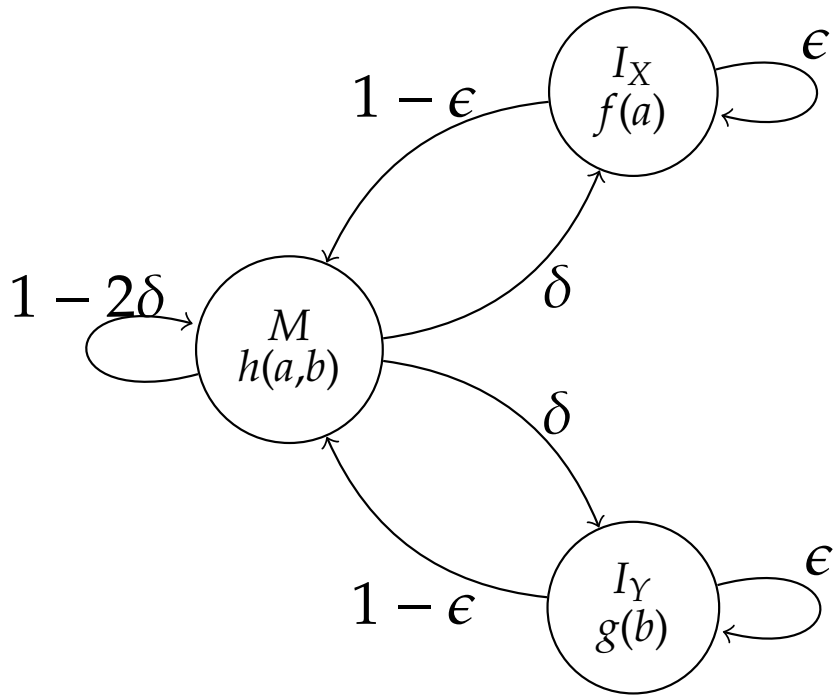
We have

$$\begin{aligned} & \mathbb{P}(X_{1:N_t}, Y_{1:M_t} | \varepsilon_{1:t}) \\ = & \prod_{s=1}^{t} f(X_{N_s})^{1\{\varepsilon_s=(1,0)\}} g(Y_{M_s})^{1\{\varepsilon_s=(0,1)\}} h(X_{N_s}, Y_{M_s})^{1\{\varepsilon_s=(1,1)\}} \end{aligned}$$

and $$\mathbb{P}(\varepsilon_{1:t}) = \mu_{\varepsilon_1} \prod_{s=1}^{t-1} \pi(\varepsilon_s, \varepsilon_{s+1}).$$

Pair-hidden Markov model IV

Representation as an automaton



$$\pi = \begin{pmatrix} \epsilon & 0 & 1-\epsilon \\ 0 & \epsilon & 1-\epsilon \\ \delta & \delta & 1-2\delta \end{pmatrix}$$

Likelihood

Observe two sequences $X_{1:n}$ and $Y_{1:m}$.

- The likelihood writes as

$$\mathbb{P}_\theta(X_{1:n}, Y_{1:m}) = \sum_{e \in \mathcal{E}_{n,m}} \mathbb{P}_\theta(\varepsilon_{1:|e|} = e_{1:|e|}, X_{1:n}, Y_{1:m})$$

  where $\mathcal{E}_{n,m}$is the set of paths from $(0,0)$ to (n,m).
- EM-algorithm applies to pair-HHM. The forward-backward equations may be generalised to this context to compute the E-step.
- It enables computing the MLE of θ.
- Moreover, one obtains a posterior distribution on the alignments.
- (One may also use Viterbi's algorithm to recover the optimal alignment).

Advantages of pairHMM over scoring methods

- Parameters are estimated. This corresponds to selecting the optimal score (from an evolutionary perspective) for these sequences.
- PairHMM provide a posterior distribution on the alignments.
- NB: [LDMH05] gives an interesting review about statistical alignment issues.

Posterior probabilities on alignments

(Source Metzler *et al.*, J. Mol. Evol. 2001)

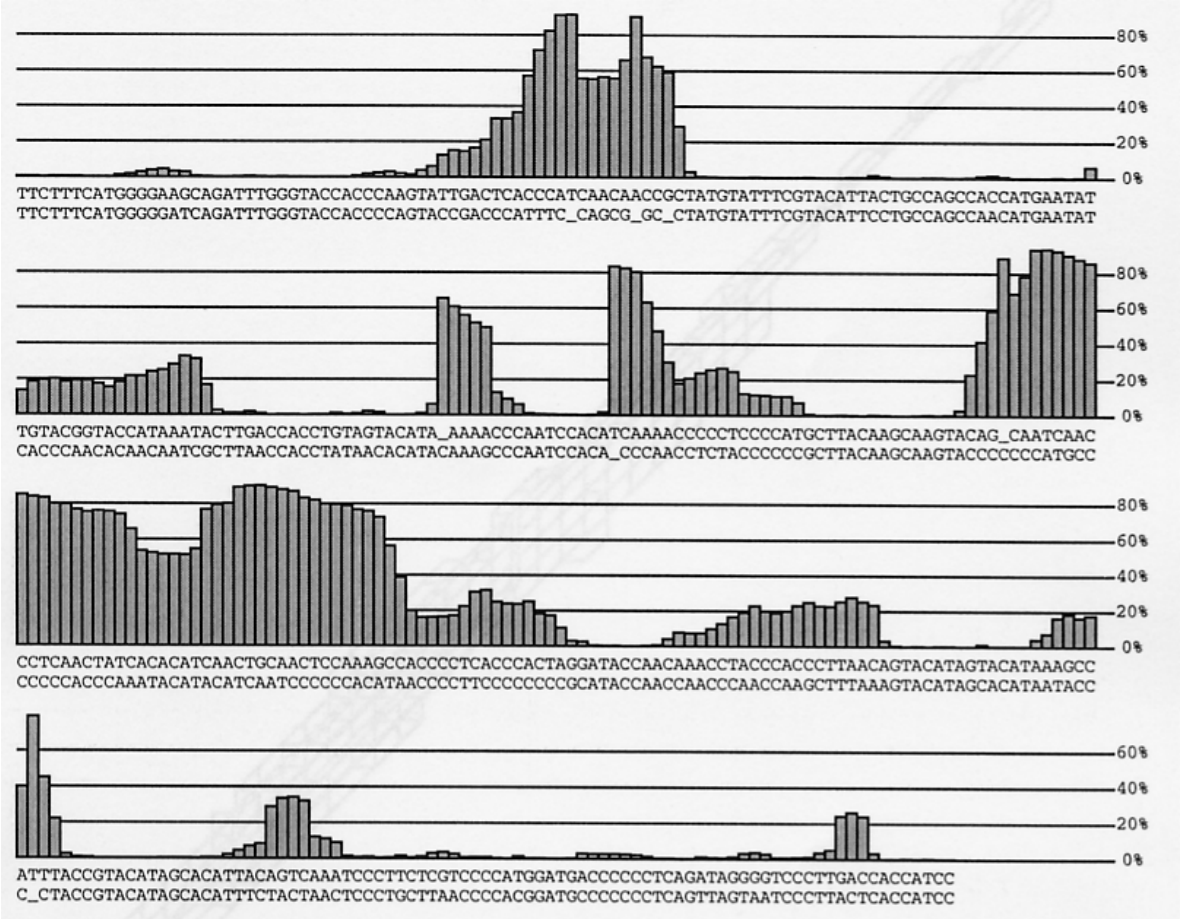


Figure 3: The most probable of the sampled alignments of a human and an orangutan HVR-1 sequence and the percentages of sampled alignments that differ from it in each position.

Exercise: Running pair-HMM with `fsa`

- Download the two alpha-1 globin sequences of human and mouse at
  `http://www.ncbi.nlm.nih.gov/nuccore/NM_000558.4`
  and
  `http://www.ncbi.nlm.nih.gov/nuccore/NM_008218.2`
- Create one fasta file *my_seqs.fa* including those two sequences
- Run `fsa` with the commands

```
fsa --gui my_seqs.fa > result.mfa
java -jar path_to_fsa/fsa/display/mad.jar my_seqs.fa
```

Relaxing independence between sites

As for evolutionary models, people have tried to relax the "independence between sites" assumption that underlines alignment procedures.

Context-dependent scoring alignments

- Some attempts have been made [WL84, Hua94, GTT06, GW07];
- However the choice of these scoring parameters becomes even more problematic !

Context-dependent statistical alignment

Two different frameworks:

- [AGM12] generalise pair-HMM to handle a Markov process conditional on the latent alignment;
- [HB11] use *tree adjoining grammars (TAG)*.

Outline Part 3

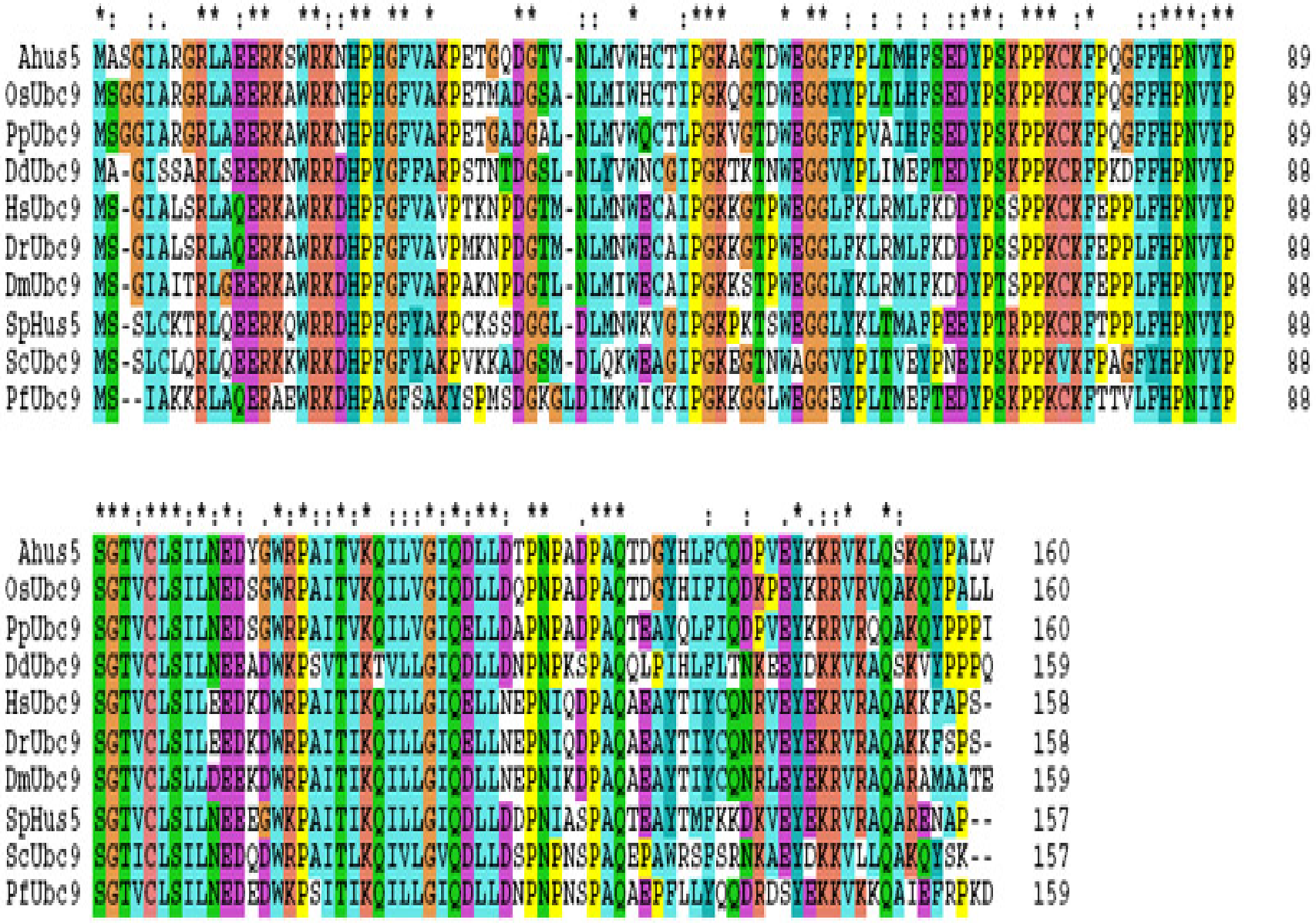
Principles of comparative genomics

Sequence evolution
- The basics
- Towards more complex models

Sequence alignment
- Introduction to alignment
- Dotplots
- Alignment through scoring
- Alignment through HMMs (statistical alignment)
- Multiple alignment

Multiple alignment of sequences

Alignment of Hus5/Ubc9 proteins in a set of organisms



Introduction to multiple alignment I

General remarks

- When more than 3 sequences, each site is either
  - a *homologous site* (i.e. present in the ancestral sequence),
  - or *deleted* (w.r.t. ancestral sequence);
  - or *inserted* (w.r.t. ancestral sequence).
- With more than 3 sequences, the space of possible alignments is huge. Complexity drastically increases.

Scoring alignment algorithms

Mainly 2 different types of strategies

- progressive strategies, based on pairs of aligned sequences (Clustal W). Strong dependency on the order of the sequences.
- with multiple anchor points (DIALIGN2, MUSCLE).

Introduction to multiple alignment II

Specific Issues

- Insertions between homologous positions are often inserted to the right, as a convention but do not represent homologies within clades for e.g.
- Homology may not be handled at the clade level,
- When using progressive alignment, indels inserted at an early stage may have a big impact on the result.

Which sequences to align?

- Be careful to the heterogeneity of the sequences;
- If there is a subset of sequences that are too close, this will induce a bias in the alignment.
- Some software weight the sequences pairs according to their similarity.

Multiple statistical alignment

Principle

- Generalising pair-HMM to more than 2 sequences is non trivial;
- Requires a phylogeny of the sequences to compute the likelihood under an evolutionary model;
- Algorithms suffer the same computational problems as for scoring-based alignment.

Some recent developments

- [FMvH05] or BaliPhy [RS05] propose to simultaneously reconstruct the phylogeny and the sequence alignments;
- FSA: fast statistical alignment [BRS09] relies on pair-HMM (thus on pairs of sequences);

Profile HMMs I

References [Edd98, KBM94]

A profile is a description of the consensus of a multiple sequence alignment.

Principle

- A number of homologous positions L is a priori fixed. A Markov chain (the profile chain) describes the succession of states *homologous, deleted or inserted*.
- Conditional to the profile, the sequences are supposed to be independent;
- Two different usages of profileHMM are made
  - Training from unaligned sequences The parameters (and underlying alignment) are estimated from the set of unaligned sequences, through a `em` algorithm.
  - Starting from an alignement: the parameters are adjusted through simple counts of symbol emission and state transitions.

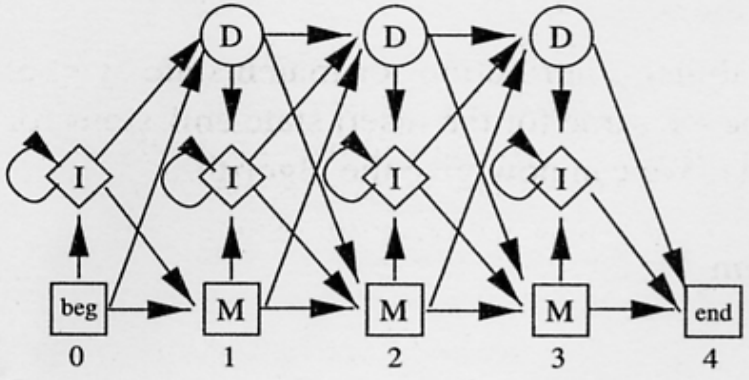
Profile HMMs II

References [Edd98, KBM94]

(a) Multiple alignment:

	x	x	.	.	.	x
bat	A	G	-	-	-	C
rat	A	-	A	G	-	C
cat	A	G	-	A	A	-
gnat	-	-	A	A	A	C
goat	A	G	-	-	-	C
	1	2	.	.	.	3

(b) Profile-HMM architecture:



(c) Observed emission/transition counts

		model position			
		0	1	2	3
match emissions	A	-	4	0	0
	C	-	0	0	4
	G	-	0	3	0
	T	-	0	0	0
insert emissions	A	0	0	6	0
	C	0	0	0	0
	G	0	0	1	0
	T	0	0	0	0
state transitions	M-M	4	3	2	4
	M-D	1	1	0	0
	M-I	0	0	1	0
	I-M	0	0	2	0
	I-D	0	0	1	0
	I-I	0	0	4	0
	D-M	-	0	0	1
	D-D	-	1	0	0
	D-I	-	0	2	0

Figure 5.7 *As an example of model construction from an alignment, a small DNA multiple alignment is given (a), with three columns marked above with x's. These three columns are assigned to positions 1–3 in the model architecture (b). The assignment of columns to model positions determines the symbol emission and state transition counts (c) from which probability parameters would be estimated.*

Profile HMMs III

References [Edd98, KBM94]

Additional remarks

- L is often chosen as the mean length value of the sequences;
- May be viewed as position-specific scoring alignment;
- Generalising pairHMM to more than 2 sequences is different from profileHMM (because in latter case, sequences are independent conditional on profile).

Exercise: Use the software `Hmmer` on sequences. Follow the tutorial from the user guide `ftp://selab.janelia.org/pub/software/hmmer3/3.1b1/Userguide.pdf`

Part III - References I

- [AGGM06] A. Arribas-Gil, E. Gassiat, and C. Matias.
  Parameter estimation in pair-hidden Markov models.
  Scand. J. Statist., 33(4):651–671, 2006.

- [AGM12] A. Arribas-Gil and C. Matias.
  A context dependent pair hidden Markov model for statistical alignment.
  Statistical applications in genetics and molecular biology, 11(1):Article 5, 2012.

- [AGMP09] A. Arribas-Gil, D. Metzler, and J.-L. Plouhinec.
  Statistical alignment with a sequence evolution model allowing rate heterogeneity along the sequence.
  IEEE/ACM Transactions on Computational Biology and Bioinformatics, 6(2):281–295, 2009.

Part III - References II


[BG12] J. Bérard and L. Guéguen.
Accurate estimation of substitution rates with neighbor-dependent models in a phylogenetic context.
Systematic Biology, 61(3):510–521, 2012.


[BGP08] J. Bérard, J.-B. Gouéré, and D. Piau.
Solvable models of neighbor-dependent nucleotide substitution processes.
Mathematical Biosciences, 211:56–88, 2008.


[BRS *et al.*09] R. K. Bradley, A. Roberts, M. Smoot, S. Juvekar, J. Do, C. Dewey, I. Holmes, and L. Pachter.
Fast statistical alignment.
PLoS Comput Biol, 5(5):e1000392, 05 2009.

Part III - References III


[DKEM98] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison.
Biological sequence analysis: probabilistic models of proteins and nucleic acids.
Cambridge University Press, Cambridge, UK, 1998.


[DSO78] M. Dayhoff, R. Schwartz, and B. Orcutt.
A model of evolutionary change in proteins.
In *Atlas of Protein sequence and structure*, volume 5, Supplement 3, pages 345–352, Washington DC, 1978. National Biomedical Research Foundation.


[Edd98] S. R. Eddy.
Profile hidden Markov models.
Bioinformatics Review, 14(9):755–763, 1998.

Part III - References IV

[Fal10] M. Falconnet.
Phylogenetic distances for neighbour dependent substitution processes.
Mathematical Biosciences, 224(2):101–108, 2010.

[FB12] M. Falconnet and S. Behrens.
Accurate estimations of evolutionary times in the context of strong CpG hypermutability.
J Comput Biol, 19(5):519–531, 2012.

[FMvH05] R. Fleissner, D. Metzler, and A. von Haeseler.
Simultaneous statistical multiple alignment and phylogeny reconstruction.
Systematic Biology, 54(4):548–561, 2005.

[Got82] O. Gotoh.
An improved algorithm for matching biological sequences.
J. Mol. Biol., 162(3):705–8, 1982.

Part III - References V

- [GTT06] A. Gambin, J. Tiuryn, and J. Tyszkiewicz.
  Alignment with context dependent scoring function.
  J Comput Biol, 13(1):81–101, 2006.
- [GW07] A. Gambin and P. Wojtalewicz.
  CTX-BLAST: context sensitive version of protein BLAST.
  Bioinformatics, 23(13):1686–1688, 2007.
- [HB11] G. Hickey and M. Blanchette.
  A probabilistic model for sequence alignment with context-sensitive indels.
  J Comput Biol., 18(11):1449–1464, 2011.
- [HH92] S. Henikoff and J. Henikoff.
  Amino acid substitution matrices from protein blocks.
  Proc Natl Acad Sci U S A., 89(22):10915–9, 1992.

Part III - References VI


[HKY85] M. Hasegawa, H. Kishino, and T. Yano.
Dating of the human-ape splitting by a molecular clock of mitochondrial DNA.
J Mol Evol., 22(2):160–174, 1985.


[Hua94] X. Huang.
A context dependent method for comparing sequences.
In *Combinatorial pattern matching (Asilomar, CA, 1994)*, volume 807 of *Lecture Notes in Comput. Sci.*, pages 54–63. Springer, Berlin, 1994.


[JC69] T. H. Jukes and C. R. Cantor.
Evolution of Protein Molecules.
In H. N. Munro, editor, *Mammalian Protein Metabolism*, pages 21–132. Academic Press, 1969.

Part III - References VII


[KBM *et al.*94] A. Krogh, M. Brown, I. Mian, K. Sjolander, and D. Haussler.
Hidden Markov models in computational biology : Applications to protein modelling.
J. Mol. Biol., 235:1501–1531, 1994.


[Kim80] M. Kimura.
A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences.
J Mol Evol, 16(2):111–120, 1980.


J. A. Lake.
Reconstructing evolutionary trees from dna and protein sequences: paralinear distances.
Proceedings of the National Academy of Sciences, 91(4):1455–1459, 1994.

Part III - References VIII

- [LDMH05] G. Lunter, A. J. Drummond, I. Miklós, and J. Hein.
  Statistical alignment: recent progress, new applications, and challenges.
  In *Statistical methods in molecular evolution*, Stat. Biol. Health, pages 375–405. Springer, New York, 2005.

- [MLH04] I. Miklos, G. A. Lunter, and I. Holmes.
  A "Long Indel" Model For Evolutionary Sequence Alignment.
  Molecular Biology and Evolution, 21(3):529–540, 2004.

- [NW70] S. Needleman and C. Wunsch.
  A general method applicable to the search for similarities in the amino acid sequence of two proteins.
  J. Mol. Biol., 48(3):443–53, 1970.

Part III - References IX


[RS05] B. D. Redelings and M. A. Suchard.
Joint bayesian estimation of alignment and phylogeny.
Systematic Biology, 54(3):401–418, 2005.


[SW81] T. Smith and M. Waterman.
Identification of common molecular subsequences.
J. Mol. Biol., 147(1):195–7, 1981.


M. Steel.
Recovering a tree from the leaf colourations it generates under a markov model.
Applied Mathematics Letters, 7(2):19 – 23, 1994.


[TKF91] J. Thorne, H. Kishino, and J. Felsenstein.
An evolutionary model for maximum likelihood alignment of DNA sequences.
J. Mol. Evol., 33:114–124, 1991.

Part III - References X


[TKF92] J. Thorne, H. Kishino, and J. Felsenstein.
Inching toward reality: an improved likelihood model of sequence evolution.
Journal of Molecular Evolution, 34:3–16, 1992.


[TN93] K. Tamura and M. Nei.
Estimation of the number of nucleotide substitutions in the control region of mitochondrial DNA in humans and chimpanzees.
Mol. Biol. Evol., 10(3):512–526, 1993.


[WL84] W. Wilbur and D. Lipman.
The context dependent comparison of biological sequences.
SIAM J. Appl. Math., 44(3):557–567, 1984.

Part III - References XI

[Yan94] Z. Yang.
Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: Approximate methods.
Journal of Molecular Evolution, 39(3):306–314, 1994.

Part IV

Introduction to phylogeny

Outline Part 4

Trees

Phylogenies of sequences
- Introduction to sequences phylogenies
- Model based phylogenies
- Bootstrap support
- Extensions

Species phylogenies

Trees: some generalities I

Definitions

- In graphs, vertices or nodes are connected through edges or branches. The degree of a node is the number of edges connecting this node. Trees are graphs with no cycles;
- We consider binary trees, where each internal node has degree 3 and the leaves have degree 1 (root has degree 2);
- The leaves represent extant species, while internal nodes represent ancestral species;
- The tree may be rooted or unrooted: the root is the most recent common ancestor (MRCA) of the set of extant species;
- The molecular clock assumption states that the evolutionary rate is constant along the tree (often violated);

Trees: some generalities II

- To root the tree, methods either use
  - the molecular clock assumption (distance and ML methods);
  - or an outgroup.
- The tree contains two type of information: a topology and branch lengths;
- A tree without branch lengths is called a cladogram;
- Branch lengths may either represent the amount of sequence divergence or a time period;
- The number of trees on n taxons is huge: it is equal to $3 \times 5 \times 7 \times \cdots \times (2n-5)$ denoted by $(2n-5)!!$
  e.g. $n = 10$ gives about 2 million and $n = 20$ gives 2.2×10^{20}.
- Thus exhaustive search through the tree space is prohibitive unless n is small.

Trees: some generalities III

Gene trees and species trees

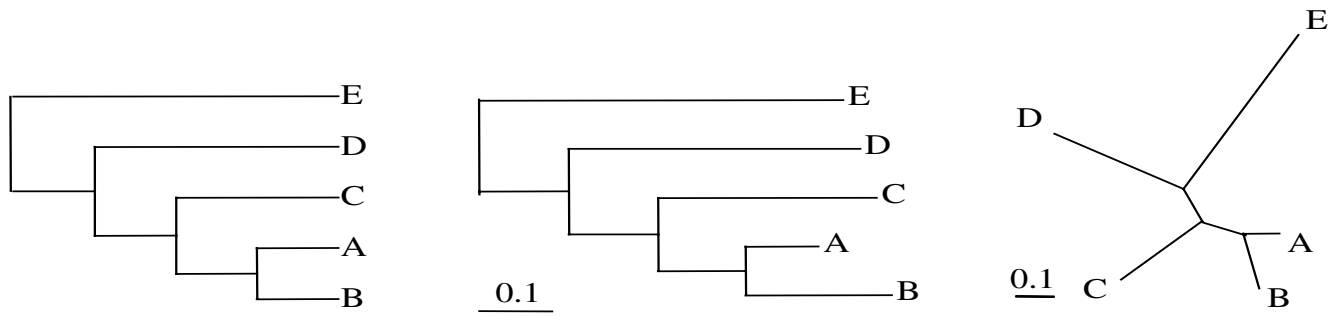
- The phylogeny of a set of genes or sequences is a gene tree;
- For many reasons, gene trees and species trees may not be identical: gene duplications (paralogues), losses, lateral gene transfers, lineage sorting and estimation errors ...

Searching the tree space

- Tree space is huge: exhaustive search is impossible and there is a need for heuristic algorithms for exploring the tree space;
- See [Yan06] for more details.

Newick format

Trees are encoded through Newick format indicating the clades in the tree. Examples:
a and b: ((((A, B), C), D), E);
b: ((((A: 0.1, B: 0.2): 0.12, C: 0.3): 0.123, D: 0.4): 0.1234, E: 0.5);
c: (((A, B), C), D, E);
c: (((A: 0.1, B: 0.2): 0.12, C: 0.3): 0.123, D: 0.4, E: 0.6234);



(a) Cladogram (b) Phylogram (c) Unrooted tree

Fig. 3.2 The same tree shown in different styles. (a) The cladogram shows the tree topology without branch lengths or with branch lengths ignored. (b) In a phylogram, branches are drawn in proportion to their lengths. (c) In an unrooted tree, the location of the root is unknown or ignored.

(Source [Yan06])

Outline Part 4

Trees

Phylogenies of sequences

- Introduction to sequences phylogenies
- Model based phylogenies
- Bootstrap support
- Extensions

Species phylogenies

Outline Part 4

Trees

Phylogenies of sequences
- Introduction to sequences phylogenies
- Model based phylogenies
- Bootstrap support
- Extensions

Species phylogenies

Methods for (seqs) phylogeny reconstruction I

Principle

- Most of the methods start from a set of aligned sequences with no indel and infer their ancestral relationships (tree);
- This is somehow circular because the construction of an alignment should use the phylogeny between the seqs !

Different types of methods

- Parsimony: reconstruct the tree that explains the observed alignment with minimal number of mutations;
- Distance methods: clustering methods where most similar sequences are ”clustered” together;
- Model-based methods: infer the tree under some evolutionary model relating these sequences; either Maximum likelihood or Bayesian methods.

Methods for (seqs) phylogeny reconstruction II

Comments on methods

- Parsimony and model-based methods are character based contrary to distance methods (that rely on distance).
- Parsimony and model-based methods both search for a tree that optimizes a specific criterion. On the contrary, distance based methods construct a tree with no explicit criterion.

Parsimony methods I

Principle

- Find the tree that explains the sequences with the most parsimonious number of mutations;
- Possible thanks to algorithms developed in [Fit71, Har73].

Non-Informative sites

- A constant site (all seqs have same letter at this position) is non-informative for the phylogeny.
- A singleton: a site at which exactly 2 letters are observed and one is observed only once is non-informative.
- Other patterns appear to be non informative as any tree requires the same amount of mutations to obtain this pattern.
- For a site to be parsimony informative, at least 2 characters must be observed, each being obs. at least twice.
- This concept is relevant ONLY in parsimony.

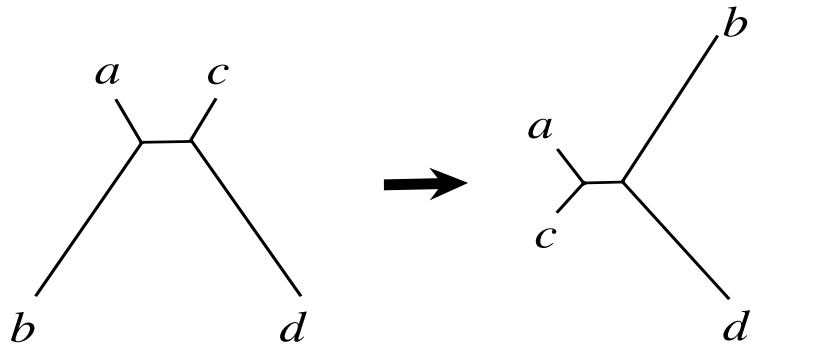
Parsimony methods II

Advantages/Inconvenients

- As for scoring alignment, requires the choice of a score for each event (often same score); thus depends on this choice.
- The method ignores branch lengths;
- Most parsimonious scenario is not the most likely in general. In fact the method is statistically inconsistent under certain scenarios [Fel78];
- Suffers from the long branch attraction problem.

It's an historical method that should'nt be used anymore.

Long branch attraction phenomenon



(a) Correct tree T_1 (b) Wrong tree T_2

(Source [Yan06]). Pattern: $xyxy$ has higher probability than $xxyy$.

Distance methods

Principle

- Compute pairwise distances between the sequences (thus a distance matrix);
- Use a clustering method to convert this matrix into a tree: *e.g.* UPGMA (unweighted pair-group method using arithmetic averages, [SS63]), Neighbor-Joining [SN87] and least-squares (LS).
- UPGMA is based on molecular clock assumption and generates rooted trees;

Distances

- Simplest case: distance = 1- percent identity;
- However some nucleotides or a.a. are ”closer” than others: thus use a similarity score and distance =1-similarity;
- There is a large literature on the choice of distances.

Distance: UPGMA

Start with a set of distances $d_{i,j}$ between the sequences.

Algorithm

- Initialisation: Assign each sequence i to its own cluster C_i ;
  Each sequence is a leaf of T, placed at height 0;
- Iteration: Take the 2 clusters i, j for which $d_{i,j}$ is minimal (pick at random if multiple choices);
  Define a new cluster k by union $C_k = C_i \cup C_j$ and set

$$d_{kl} = \frac{d_{il}|C_i| + d_{jl}|C_j|}{|C_i| + |C_j|} \quad \forall l,$$

  Define a node k, parent of i and j, placed at height $d_{ij}/2$;
  Add C_k to the set of clusters and remove C_i, C_j;
- Termination: When only 2 clusters i, j remain, place the root at height $d_{ij}/2$

Exercise: Try it by hand on a small set of 5 sequences. Compare the resulting and initial distances. What do you note?

Distance: Least-squares method

Start with a set of distances $d_{i,j}$ between the sequences.

Method's principle

- For any tree T, let $\hat{d}_{ij}(T)$ be the sum of branch lengths along the path between i and j;
- Minimize $\sum_{i,j}(d_{ij} - \hat{d}_{ij}(T))^2$ w.r.t branch lengths and denote by $S(T)$ the resulting tree length (sum of branch distances);
- Choose tree T with smallest value $S(T)$ (need to explore the tree space).

NB: Approximate algorithms may propose solutions with negative branch lengths.

Neighbor joining

- Method that minimizes an ”evolution criterion”: the sum of branch lengths;
- Divisive cluster algorithm: starting with the star tree and join two nodes, choosing the pair that achieves greatest reduction in tree length; Iterate the procedure.

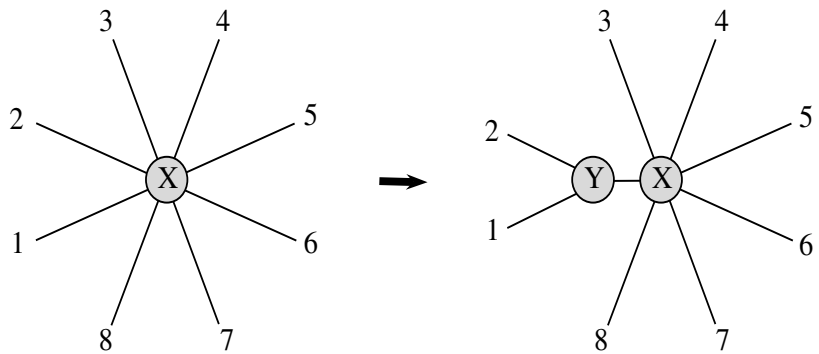


Fig. 3.17 The neighbour-joining method of tree reconstruction is a divisive cluster algorithm, dividing taxa successively into finer groups.

(Source [Yan06]).

Advantages/Inconvenients of distance-based methods

- Fast to compute;
- Applies whenever one can define a distance between the objects;
- Large distances are poorly estimated;

Outline Part 4

Trees

Phylogenies of sequences
- Introduction to sequences phylogenies
- Model based phylogenies
- Bootstrap support
- Extensions

Species phylogenies

Maximum likelihood

Principle: 2 main steps

- Step 1: For each possible tree topology T, compute the maximum likelihood $L(\theta|T)$ of the alignment conditional on this tree; *i.e.* find evolutionary parameters θ (= branch lengths + substitution rates) that maximize the likelihood;
- Step 2: Explore the set of trees to find one with maximum likelihood;

Step 1: computing the likelihood

- Markov evolution model *i.e.* $P_{xy}(t) = \mathbb{P}(X_t = y|X_0 = x)$;
- Sites in the alignment are assumed i.i.d. so that the likelihood is a product over all sites $L(\theta|T) = \prod_{i=1}^{n} L_i(\theta|T)$;
- The likelihood of each site is obtained through Felsenstein's pruning algorithm [Fel81] on rooted trees.
- Then, numerical optimization finds best parameter value.

Felsenstein's pruning algorithm (rooted trees) I

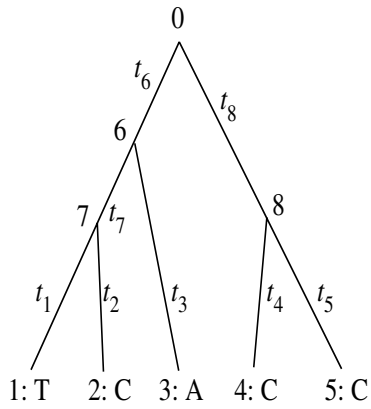


Fig. 4.1 A tree of five species used to demonstrate calculation of the liklihood function. The nucleotides observed at the tips at a site are shown. Branch lengths t_1–t_8 are measured by the expected number of nucleotide substitutions per site.

(Source [Yan06]).

Computation of the likelihood $L_i(\theta|T)$ at a site i requires summing over all possible values at the (here 4) internal nodes, which is prohibitive.

$$L_i(\theta|T) = \sum_{x_0}\sum_{x_6}\sum_{x_7}\sum_{x_8} \pi(x_0)P_{x_0x_6}(t_6)P_{x_6x_7}(t_7)P_{x_6A}(t_3)P_{x_7T}(t_1)P_{x_7C}(t_2)P_{x_0x_8}(t_8)P_{x_8C}(t_4)P_{x_8C}(t_5).$$

Felsenstein's pruning algorithm (rooted trees) II

Method (from leaves to root)

- $L_i(x_i)$ = probability of observing data at the tips that are descendants of node i, given that nucleotide at node i is x_i;
- At the leaves, $L_i(x_i)$ is 1 if x_i is observed, 0 otherwise;
- If i is an interior node with daughter nodes j and k and respective branch lengths t_j, t_k then

$$L_i(x_i) = \left(\sum_{x \in \mathcal{A}} P_{x_i x}(t_j) L_j(x)\right) \times \left(\sum_{y \in \mathcal{A}} P_{x_i y}(t_k) L_k(y)\right).$$

- Finally, the likelihood for a site is obtained as $L(\theta|T) = \sum_{x \in \mathcal{A}} \pi_x L_{root}(x)$, where π is stationary distribution of the Markov evolutionary process (estimated from extant sequences).

Felsenstein's pruning algorithm (rooted trees) III

Advantages

- For each site, the complexity is linear w.r.t. the number of species (while the number of possible states for internal nodes increases exponentially);
- The quantities $P_{xy}(t_j)$ are the same for all sites (only depend on branch length) and computed once for all sites;
- If two sites are identical, their likelihood is the same: sites are collapsed into site patterns;

Reversible models and the pulley principle [Fel81]

- For reversible models, neither direction of time nor root positioning are relevant;
- This induces potential simplifications in likelihood calculations;

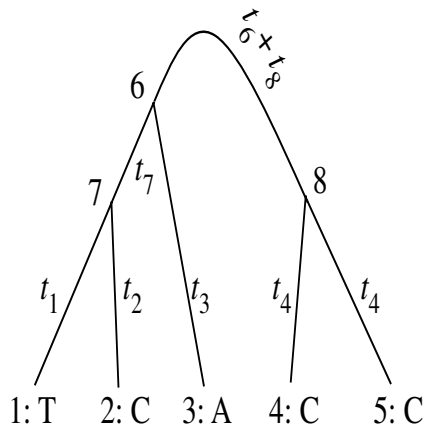


Fig. 4.3 The ensuing unrooted tree when the root is moved from node 0 to node 6 in the tree of Fig. 4.1.

(Source [Yan06]).

Extensions to more complex models

- One may assume different evolutionary models at the sites (still independent);
- For variable rates among sites (e.g. Γ model): assume K different (unknown) groups of sites (with homogeneous evolution), with proportions $\pi_1, \ldots, \pi_K$. Then the likelihood is a weighted average of the likelihood within each group $L(\theta, \pi) = \sum_{k=1}^{K} \pi_k L_{\text{model k}}(\theta)$;
- Correlation between sites may be introduced through HMM [Yan95, FC96].
- Covarion models: rates may vary along the branches of the tree and switch from ”on” (sites evolves with constant rate) to ”off”= invariable [Gal01, Hue02].

Some words on step 2

Recall: Step 2 consists in exploring the set of trees to find one with large likelihood.

- Felsenstein's pruning algorithm has been known from 1981 but likelihood methods have started to be used only very recently;
- This is due to efficient implementations of the tree space search [GG03];
- In particular, thanks to phyML [GDL10].

Bayesian methods

Bayesian paradigm

- Put a prior distribution π on the parameter θ of the model;
- Compute its posterior probability $\mathbb{P}(\theta|\text{data}) \propto \pi(\theta)\mathbb{P}(\text{data}|\theta)$, where $\mathbb{P}(\text{data}|\theta)$ is the model likelihood.
- See Chapter 5 in [Yan06] for more details.

Bayesian phylogenetics

- Put a prior on the set of tree topologies (e.g. uniform) and a prior on the set of branch lengths+mutation rates;
- Compute posterior through MCMC algorithms;
- This provides posterior probabilities for trees or clades (compare with bootstrap support values).
- But the method is criticised ...

Outline Part 4

Trees

Phylogenies of sequences
- Introduction to sequences phylogenies
- Model based phylogenies
- Bootstrap support
- Extensions

Species phylogenies

Bootstrap principle

Consider a sample $\mathbf{X} = (X_1, \ldots, X_n)$ of i.i.d. variables and a statistic $T(\mathbf{X})$.

- A bootstrap sample $\mathbf{X}^\star = (X_1^\star, \ldots, X_n^\star)$ is a sample of size n, obtained by random sampling with replacement from $\mathbf{X}$.
- For $1 \leq b \leq B$, draw $\mathbf{X}^b$ a bootstrap sample of $\mathbf{X}$ and compute $T^b = T(\mathbf{X}^b)$.
- Then $\mathbf{T} = (T^1, \ldots, T^B)$ is a sample of size B drawn from the conditional distribution of T given $\mathbf{X}$.
- This sample may be used to estimate quantities on the statistic T, such as bias, variance, etc. (valid only of B is large).
- Based on the idea that the distr. $\mathbb{P}$ of X_i's is approximated by the empirical distribution $\mathbb{P}_n$ (valid if n large).

Example: bootstrap estimation of the variance of EM estimators

Context

- Model with missing data or latent variables (ex: mixture, HMM ...).
- Estimator $\hat{\theta}^{MLE}$ of θ does not have an analytic expression. It is approximated through $\hat{\theta}^{EM}$ the result of an EM algorithm.
- But EM algo. does not provide an estimation of $\mathbb{V}\text{ar}(\hat{\theta}^{EM})$.

Standard error of $\hat{\theta}^{EM}$

- For each bootstrap sample $\mathbf{X}^b$, use EM algo. to obtain $\hat{\theta}^{EM,b}$.
- Then estimate $\text{sd}(\hat{\theta}^{EM})$ with

$$\widehat{\text{sd}}_{boot}(\hat{\theta}^{EM}) = \left\{ \frac{1}{B-1} \sum_{b=1}^{B} (\hat{\theta}^{EM,b} - \frac{1}{B} \sum_{b'} \hat{\theta}^{EM,b'})^2 \right\}^{1/2}.$$

Bootstrap support of a branch [F85]

Principle

Start with an alignment and tree T inferred by any method (ML or distance . . .). Consider the clades induced by the branches of this tree.

- For each $1 \leq b \leq B$, randomly sample with replacement among the columns of the alignment; estimate the corresponding tree T^b with your method;
- The bootstrap support of a branch of T is the percentage of times this branch appears in the bootstrapped trees T^b.

Approximate boostrap for ML

Many approximate methods have been proposed, to speed up computations

- `REEL`: resampling estimated likelihoods [Kis90]
- `RAxML`: rapid bootstrap [Sta06]
- . . .

Outline Part 4

Trees

Phylogenies of sequences
- Introduction to sequences phylogenies
- Model based phylogenies
- Bootstrap support
- Extensions

Species phylogenies

Attempts to infer phylogenies and alignments at the same time I

Setup

- Reconstructed phylogenies heavily depend on alignments; but alignments should be consequences of underlying phylogenies!!!
- Exactly as we noticed that alignments and evolutionary parameters should be inferred at the same time, alignments and phylogenies (and evolutionary parameters!) should be inferred directly from the sequences.

Attempts to infer phylogenies and alignments at the same time II

Current proposals

- SATÉ [Liu et al.09, Liu et al.12] may be one of the most promising methods to infer phylogenies directly from sequences ;
- It's an iterative method that iterates 2 steps:
  - compute an alignment of the sequences using a *guiding tree* ;
  - update the tree from the current alignment using maximum likelihood approach.
- SATÉ appears to be powerful. First step (alignment) relies on scoring alignment;

Exercise: Run SATÉ, following the tutorial at `http://phylo.bio.ku.edu/software/sate/sate_tutorial.pdf`

Attempts to infer phylogenies and alignments at the same time III

Current proposals (foll.)

- An earlier reference [Fleissner et al.05] proposes a similar approach with scoring alignment replaced by statistical alignment (=pair-HMM for more than 2 sequences) and tree ML search replaced by neighbor joining
- Currently less performing than SATÉ;
- Ideally, one should develop the same approach, mixing statistical alignment (step 1) with ML methods for tree reconstruction (step 2);

Outline Part 4

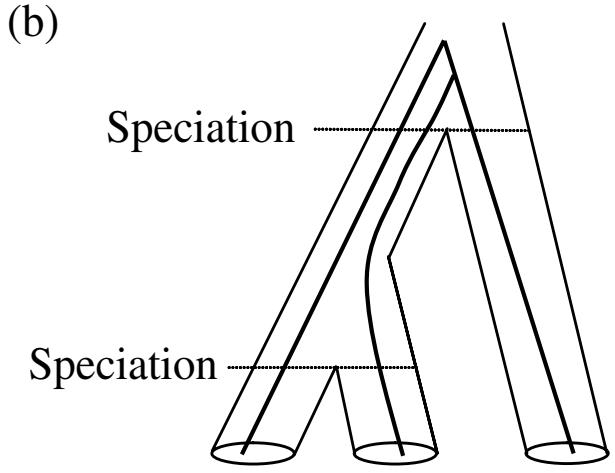
Trees

Phylogenies of sequences
- Introduction to sequences phylogenies
- Model based phylogenies
- Bootstrap support
- Extensions

Species phylogenies

Discrepancies between seqs/genes and species phylogenies

- Gene duplications: genes may duplicate independently of speciation. Pbm of distinguishing paralogs from orthologs;
- Transfers: horizontal gene transfer has to be taken into account;
- Losses: This covers many different situations (e.g. sampling errors, extinction . . .)
- Lineage sorting: happens when a polymorphism appears prior to speciation. The shortest the branch lengths, the more likely it is.



Methods for species phylogenies reconstruction

Many methods exist

- Super-matrices: concatenate all genes alignments and infer a global tree. The method may use different evolutionary models per gene;
- Consensus trees: construct one tree per gene and use a method to extract some ”consensus tree”;
- Coalescent-based methods: model the way genes (= individuals) evolve in species (= populations).
- Reconciliation methods: reconciliation is a mapping between nodes of gene trees and species trees. Those methods try to reconstruct one or both trees by taking into account evolutionary events such as duplication, transfers and/or losses.

References I


[F85] J. Felsenstein.
Confidence Limits on Phylogenies: An Approach Using the Bootstrap.
Evolution, 39:783–791, 1985.


[FC96] J. Felsenstein and G. A. Churchill.
A hidden Markov model approach to variation among sites in rate of evolution.
Mol. Biol. Evol., 13:93–104, 1996.


[Fel78] J. Felsenstein.
Cases in which parsimony and compatibility methods will be positively misleading.
Syst. Zool, 27:401–410, 1978.

References II


[Fel81] J. Felsenstein.
Evolutionary trees from DNA sequences: A maximum likelihood approach.
J. Mol. Evol., 17:368–376, 1981.


[Fit71] W. M. Fitch.
Toward defining the course of evolution: minimum change for a specific tree topology.
Syst. Zool, 20:406–416, 1971.


[Fleissner *et al.*05] R. Fleissner, D. Metzler and A. Von Haeseler.
Simultaneous Statistical Multiple Alignment and Phylogeny Reconstruction.
Syst. Biol. 54(4): 548–561, 2005.

References III


[Gal01] N. Galtier.
Maximum-likelihood phylogenetic analysis under a covarion-like model.
Mol. Biol. Evol., 18:866–873, 2001.


[GDL *et al.*10] S. Guindon, J. Dufayard, V. Lefort, M. Anisimova, W. Hordijk, and G. O.
New algorithms and methods to estimate maximum-likelihood phylogenies: Assessing the performance of phyml 3.0.
Systematic Biology, 59(3):307–21, 2010.


[GG03] S. Guindon and O. Gascuel.
A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood.
Systematic Biology, 52(5):696–704, 2003.

References IV


[Har73] J. A. Hartigan.
Minimum evolution fits to a given tree.
Biometrics, 29:53–65, 1973.


[Hue02] J. P. Huelsenbeck.
Testing a covariotide model of DNA substitution.
Mol. Biol. Evol., 19:698–707, 2002.


[Kis90] H. Kishino and T. Miyata and M. Hasegawa.
Maximum-likelihood inference of protein phylogeny and the origin of chloroplasts.
J. Mol. Evol., 31:151–160, 1990.


[Liu*et al.*09] K. Liu, S. Raghavan, S. Nelesen, C. Randal Linder, T. Warnow.
Rapid and Accurate Large-Scale Coestimation of Sequence Alignments and Phylogenetic Trees.
Science 324: 1561, 2009.

References V


[Liu*et al.*12] K. Liu, T.J. Warnow, M.T. Holder, S.M. Nelesen, J. Yu, A.P. Stamatakis and C. Randal Linder.
SATé-II: Very Fast and Accurate Simultaneous Estimation of Multiple Sequence Alignments and Phylogenetic Trees.
Syst Biol, 61(1): 90-106, 2012.


[SN87] N. Saitou and M. Nei.
The neighbor-joining method: a new method for reconstructing phylogenetic trees.
Mol. Biol. Evol, 4:406–425, 1987.


[SS63] R. R. Sokal and P. H. A. Sneath.
Numerical Taxonomy.
W.H. Freeman and Co., San Francisco, CA., 1963.

References VI


[Sta06] A. Stamatakis.
RAxML-VI-HPC: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models.
Bioinformatics, 22:2688–2690, 2006.


[Yan95] Z. Yang.
A space-time process model for the evolution of DNA sequences.
Genetics, 139:993–1005, 1995.


[Yan06] Z. Yang.
Computational Molecular Evolution.
Oxford Series in Ecology and Evolution. Oxford University Press, 2006.