Markov chains and Hidden Markov Models
Markov chains and HMMs

We will discuss:

- Markov chains
- Hidden Markov Models (HMMs)
This chapter is based on:


- An earlier version of this lecture by Daniel Huson.
As an introduction to Markov chains, we consider the problem of finding \( \text{CpG-islands} \) in the human genome.

A piece of double stranded DNA:

\[
\ldots | | | | | | | | | | | | | | | | | | | | | | | | | | | | | \ldots \\
\]

The \( \text{C} \) in a \( \text{CpG} \) pair is often modified by \textit{methylation} (that is, an \( H \)-atom is replaced by a \( CH_3 \)-group). There is a relatively high chance that the methyl-C will mutate to a T. Hence, \( \text{CpG} \)-pairs are underrepresented in the human genome.

Methylation plays an important role in transsscription regulation. Upstream of a gene, the methylation process is suppressed in a short region of length 100-5000. These areas are called \textit{CpG-islands}. They are characterized by the fact that we see more \( \text{CpG} \)-pairs in them than elsewhere.
More specifically, we can ask the following questions.

1. Given a (short) segment of genomic sequence. How to decide whether this segment is from a \( \text{CpG}\)-island or not?

2. Given a (long) segment of genomic sequence. How to find all \( \text{CpG}\)-islands contained in it?
Our goal is to come up with a probabilistic model for CpG-islands. Because pairs of consecutive nucleotides are important in this context, we need a model in which the probability of one symbol depends on the probability of its predecessor. This dependency is captured by the concept of a Markov chain.
Example.

- Circles = *states*, e.g., with names A, C, G, and T.

- Arrows = possible *transitions*, each labeled with a *transition probability* $a_{st}$. Let $x_i$ denote the state at time $i$. Then

$$a_{st} := P(x_{i+1} = t \mid x_i = s)$$

is the conditional probability to go to state $t$ in the next step, given that the current state is $s$. 


**Markov chains**

Definition.

A (time-homogeneous) *Markov chain* (of order 1) is a system \((Q, A)\) consisting of a finite set of *states* \(Q = \{s_1, s_2, \ldots, s_n\}\) and a *transition matrix* \(A = \{a_{st}\}\) with \(\sum_{t \in Q} a_{st} = 1\) for all \(s \in Q\) that determines the probability of the transition \(s \rightarrow t\) by

\[
P(x_{i+1} = t \mid x_i = s) = a_{st}.
\]

At any time \(i\) the Markov chain is in a specific state \(x_i\), and at the tick of a clock the chain changes to state \(x_{i+1}\) according to the given transition probabilities.
Remarks on terminology.

- **Order 1** means that the transition probabilities of the Markov chain can only “remember” one state of its history. Beyond this, it is memoryless. The “memorylessness” condition is very important. It is called the Markov property.

- The Markov chain is **time-homogenous** because the transition probability

\[
P(x_{i+1} = t \mid x_i = s) = a_{st}.
\]

does not depend on the time parameter \(i\).
Example.
Weather in Dahlem, daily at midday: Possible states are “rain”, “sun”, or “clouds”.

Transition probabilities:
\[
\begin{array}{ccc}
  & r & s & c \\
  r & .5 & .1 & .4 \\
  s & .2 & .6 & .2 \\
  c & .3 & .3 & .4 \\
\end{array}
\]

Note that all rows add up to 1.

Weather: ...rrrrrrccssssssscscscscrcrrcrrrrrcssssss...
Given a sequence of states $s_1, s_2, s_3, \ldots, s_L$. What is the probability that a Markov chain $x = x_1, x_2, x_3, \ldots, x_L$ will step through precisely this sequence of states? We have

$$P(x_L = s_L, x_{L-1} = s_{L-1}, \ldots, x_1 = s_1)$$

$$= P(x_L = s_L \mid x_{L-1} = s_{L-1}, \ldots, x_1 = s_1)$$

$$\cdot P(x_{L-1} = s_{L-1} \mid x_{L-2} = s_{L-2}, \ldots, x_1 = s_1)$$

$$\cdots P(x_2 = s_2 \mid x_1 = s_1) \cdot P(x_1 = s_1)$$

using the “expansion”

$$P(A \mid B) = \frac{P(A \cap B)}{P(B)} \iff P(A \cap B) = P(A \mid B) \cdot P(B).$$
Now

\[ P(x_i = s_i \mid x_{i-1} = s_{i-1}, \ldots, x_1 = s_1) = P(x_i = s_i \mid x_{i-1} = s_{i-1}) \]

by the Markov property. Thus,

\[
\begin{align*}
P(x_{L} = s_{L}, x_{L-1} = s_{L-1}, \ldots, x_1 = s_1) &= P(x_{L} = s_{L} \mid x_{L-1} = s_{L-1}, \ldots, x_1 = s_1) \\
& \quad \cdot P(x_{L-1} = s_{L-1} \mid x_{L-2} = s_{L-2}, \ldots, x_1 = s_1) \\
& \quad \cdots P(x_2 = s_2 \mid x_1 = s_1) \cdot P(x_1 = s_1) \\
&= P(x_{L} = s_{L} \mid x_{L-1} = s_{L-1}) \\
& \quad \cdot P(x_{L-1} = s_{L-1} \mid x_{L-2} = s_{L-2}) \\
& \quad \cdots P(x_2 = s_2 \mid x_1 = s_1) \cdot P(x_1 = s_1) \\
&= P(x_1 = s_1) \prod_{i=2}^{L} a_{s_{i-1}s_i}.
\end{align*}
\]

Hence the probability of a path is the product of the probability of the initial state and the transition probabilities of its edges.
Modeling the begin and end states

A Markov chain *starts* in state $x_1$ with an initial probability of $P(x_1 = s)$. For simplicity we would like to model this as a transition, too.

Therefore we add a *begin state* to the model that is labeled '$b$'. We also impose the constraint that $x_0 = b$ holds. Then:

$$P(x_1 = s) = a_{bs}.$$ 

This way, we can store all probabilities in one matrix and the “first” state $x_1$ is no longer special:

$$P(x_L = s_L, x_{L-1} = s_{L-1}, \ldots, x_1 = s_1) = \prod_{i=1}^{L} a_{s_{i-1}s_i}.$$
Similarly, we explicitly model the end of the sequence of states using an end state ‘\(e\)’. Thus, the probability that the Markov chain stops is

\[
P(x_L = t) = a_{x_L e}.
\]

if the current state is \(t\).

We think of \(b\) and \(e\) as silent states, because they do not correspond to letters in the sequence.
Example:

# Markov chain that generates CpG islands
# (Source: DEMK98, p 50)
# Number of states:
6
# State labels:
# Transition matrix:
0.1795 0.2735 0.4255 0.1195 0 0.002
0.1705 0.3665 0.2735 0.1875 0 0.002
0.1605 0.3385 0.3745 0.1245 0 0.002
0.0785 0.3545 0.3835 0.1815 0 0.002
0.2495 0.2495 0.2495 0.2495 0 0.002
0.0000 0.0000 0.0000 0.0000 0 1.000
Determining the transition matrix

How do we find transition probabilities that explain a given set of sequences best? It can be proven that the following is optimal.

The transition matrix $A^+$ for DNA that comes from a $\text{CpG}$-island, is determined as follows:

$$a^+_{st} = \frac{c^+_{st}}{\sum_{t'} c^+_{st'}},$$

where $c^+_{st}$ is the number of positions in a training set of $\text{CpG}$-islands at which state $s$ is followed by state $t$. We can calculate these counts in a single pass over the sequences and store them in a $|\Sigma| \times |\Sigma|$ matrix.

We obtain the matrix $A^-$ for non-$\text{CpG}$-islands from empirical data in a similar way.

In general, the matrix of transition probabilities is not symmetric.
Determining the transition matrix

Two examples of Markov chains.

# Markov chain for CpG islands
# (Source: DEKM98, p 50)
# Number of states:
6
# State labels:
A  C  G  T  *  +
# Transition matrix:
.1795 .2735 .4255 .1195 0 0.002
.1705 .3665 .2735 .1875 0 0.002
.1605 .3385 .3745 .1245 0 0.002
.0785 .3545 .3835 .1815 0 0.002
.2495 .2495 .2495 .2495 0 0.002
.0000 .0000 .0000 .0000 0 1.000

# Markov chain for non-CpG islands
# (Source: DEKM98, p 50)
# Number of states:
6
# State labels:
A  C  G  T  *  +
# Transition matrix:
.2995 .2045 .2845 .2095 0 0.002
.3215 .2975 .0775 .3015 0 0.002
.2475 .2455 .2975 .2075 0 0.002
.1765 .2385 .2915 .2915 0 0.002
.2495 .2495 .2495 .2495 0 0.002
.0000 .0000 .0000 .0000 0 1.000

Note the different values for CpG: $a_{CG}^+ = 0.2735$ versus $a_{CG}^- = 0.0775$. 
Testing hypotheses

When we have two models, we can ask which one explains the observation better.

Given a (short) sequence $x = (x_1, x_2, \ldots, x_L)$. Does it come from a CpG-island (model$^+$)?

We have

$$P(x \mid \text{model}^+) = \prod_{i=0}^{L} a_{x_i x_{i+1}}^+,$$

with $x_0 = b$ and $x_{L+1} = e$. Similar for (model$^-$).
Testing hypotheses

To compare the models, we calculate the *log-odds ratio*:

\[
S(x) = \log \frac{P(x \mid \text{model}^+)}{P(x \mid \text{model}^-)} = \sum_{i=0}^{L} \log \frac{a_{x_{i-1}x_i}^+}{a_{x_{i-1}x_i}^-}.
\]

Then this ratio is normalized by the length of \(x\). This resulting *length-normalized log-odds score* \(S(x)/|x|\) can be used to classify \(x\). The higher this score is, the higher the probability is that \(x\) comes from a CpG-island.
The histogram of the length-normalized scores for the sequences from the training sets for $A^+$ and $A^-$ shows that $S(x)/|x|$ is indeed a good classifier for this data set. (Since the base two logarithm was used, the unit of measurement is called “bits”.)
Example.
Weather in Dahlem, daily at midday: Possible states are rain, sun or clouds.

Transition probabilities:
\[
\begin{array}{ccc}
\text{r} & \text{s} & \text{c} \\
\text{r} & .5 & .1 & .4 \\
\text{s} & .2 & .6 & .2 \\
\text{c} & .3 & .3 & .4 \\
\end{array}
\]

Types of questions that the Markov chain model can answer:

If it is sunny today, what is the probability that the sun will shine for the next seven days?

And what is more unlikely: 7 days sun or 8 days rain?
Hidden Markov Models

(HMMs)
Motivation: Question 2, how to find CpG-islands in a long sequence?

We could approach this using Markov Chains and a “window technique”: a window of width $w$ is moved along the sequence and the score (as defined above) is plotted. However the results are somewhat unsatisfactory: It is hard to determine the boundaries of CpG-islands, and which window size $w$ should one choose? . . .

Another approach is to merge the two Markov chains $\text{model}^+$ and $\text{model}^-$ to obtain a so-called Hidden Markov Model. (But HMMs are much more general than this.)
Hidden Markov Models

Definition.
An HMM is a system \( M = (\Sigma, Q, A, e) \) consisting of

- an alphabet \( \Sigma \),

- a set of states \( Q \),

- a matrix \( A = \{a_{kl}\} \) of transition probabilities \( a_{kl} \) for \( k, l \in Q \), and

- an emission probability \( e_k(b) \) for every \( k \in Q \) and \( b \in \Sigma \).
A HMM for CpG-islands:

Now we also have all transitions between states in either of the two sets that carry over from the two Markov chains \( \text{model}^+ \) and \( \text{model}^- \). The old edges within the two models are still there, but not shown here.
HMM for CpG-islands

# Number of states:
9
# Names of states (begin/end, A+, C+, G+, T+, A-, C-, G- and T-):
0 A+ C+ G+ T+ A- C- G- T-
# Number of symbols:
4
# Names of symbols:
 a c g t
# Transition matrix, probability to change from +island to -island (and vice versa) is 10E-4
#
# O  A+  C+  G+  T+  A-  C-  G-  T-
0  0.000000  0.0725193  0.1637630  0.1788242  0.0754545  0.1322050  0.1267006  0.1226380  0.1278950
A+  0.0010000  0.1762237  0.2682517  0.4170629  0.1174825  0.0035964  0.0054745  0.0085104  0.0023976
C+  0.0010000  0.1672435  0.3599201  0.2679840  0.1838722  0.0034131  0.0073453  0.0054690  0.0037524
G+  0.0010000  0.1576223  0.3318881  0.3671328  0.1223776  0.0032167  0.0067732  0.0074915  0.0024975
T+  0.0010000  0.0773426  0.3475514  0.3759440  0.1223776  0.0032167  0.0073453  0.0054690  0.0037524
A-  0.0010000  0.0002997  0.0002047  0.0002837  0.0002097  0.2994005  0.2045904  0.2844305  0.2095804
C-  0.0010000  0.0003216  0.0002977  0.0000769  0.0003016  0.3213566  0.2974045  0.0778441  0.3013966
G-  0.0010000  0.0001768  0.0002387  0.0002917  0.0002917  0.1766463  0.2385224  0.2914165  0.2914155
T-  0.0010000  0.0002477  0.0002457  0.0002977  0.0002077  0.2475044  0.2455084  0.2974035  0.2075844

# Emission probabilities:
#
# O  a  c  g  t
0  1  0  0  0
A+  1  0  0  0
C+  0  1  0  0
G+  0  0  1  0
T+  0  0  1  0
A-  1  0  0  0
C-  0  1  0  0
G-  0  0  1  0
T-  0  0  1  0

26
HMM for \textit{CpG}-islands \textsuperscript{(2)}

Note the emission probabilities: The model emits the letters A, C, G, T, but for each letter there are two states where the letter can come from. Thus we cannot reconstruct the path the HMM has taken from the sequence alone.

In general, the emission probabilities need not be zero-one. But in the HMM for \textit{CpG}-islands every state emits a unique letter. Thus the emitted symbol is not really “random” for a given state.

Next we look at an example where the states and the emitted symbols are associated in a looser way.
An *occasionally dishonest casino* uses two dice, a *fair* and a *loaded* one:

A casino guest only observes the *numbers* rolled:

6 4 3 2 3 4 6 5 1 2 3 4 5 6 6 6 6 3 2 1 2 6 3 4 2 1 6 6 ...  

However, which *die* was used remains hidden:

F F F F F F F F F F F F F F F F F F F F F F F F F F F F F F ...
HMMs, like Markov chains, are related to stochastic regular grammars and finite automata. Here is how we can generate a random sequence using an HMM:

**Algorithm: Random sequence from HMM**

- Start in state 0.
- While we have not re-entered state 0:
  - Choose a new *state* according to the transition probabilities.
  - Choose a *symbol* using the emission probabilities, and report it.
Example.

Here the fair/loaded HMM was used to generate a sequence of states and symbols:

States : FFFFFFFFFFFFFFFFFFUUUUUUUUUUUUUUUUUFFFFFFFFFUUUUUUUUUUUUUUUUUU
Symbols: 24335642611341666666526562426612134635535566462666636664253

States : FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFUUUUUUUFFUUUUUUUUUUUUUUUUFFFF
Symbols: 35246363252521655615445656366366511145445656621261532516435

States : FFUUUUUUU
Symbols: 514652666
Algorithmic tasks for HMMs

Let $M$ be an HMM and $x$ be a sequence of symbols.

1. Determine the sequence of states through $M$ which has the highest probability to generate $x$: **Viterbi algorithm**

2. Determine the probability $P(x)$ that $M$ generates $x$: **forward algorithm** or **backward algorithm**

3. Assuming that $x$ was generated by $M$, determine the conditional probability that $M$ was in state $s$ when it generated the $i$-th symbol $x_i$: **Posterior decoding**, done by a combination of the **forward and backward algorithm**

4. Given $x$ and perhaps some additional sequences of symbols, adjust the transition and emission probabilities of $M$ such that it explains best these observations: E.g., **Baum-Welch algorithm**