I529: Machine Learning in Bioinformatics (Spring 2017)

Markov Models

Yuzhen Ye
School of Informatics and Computing
Indiana University, Bloomington
Spring 2017

Outline

- Simple model (frequency & profile) review
- Markov chain
- CpG island question 1
  - Model comparison by log likelihood ratio test
- Markov chain variants
  - Kth order
  - Inhomogeneous Markov chains
  - Interpolated Markov models (IMM)
- Applications
  - Gene finding (Genemark & Glimmer)
  - Taxonomic assignment in metagenomics (Phymm)

A DNA profile (matrix)

TATAAA
TATAAT
TATAAA
TATAAA
TATAAA
TATTAA
TTAAAA
TAGAAA

	1	2	3	4	5	6
T	8	1	6	1	0	1
C	0	0	0	0	0	0
A	0	7	1	7	8	7
G	0	0	1	0	0	0

Sparse data → pseudo-counts

	1	2	3	4	5	6
T	9	2	7	2	1	2
C	1	1	1	1	1	1
A	1	8	2	8	9	8
G	1	1	2	1	1	1

Frequency & profile model

- Frequency model: the order of nucleotides in the training sequences is ignored;
- Profile model: the training sequences are aligned → the order of nucleotides in the training sequences is fully preserved
- Markov chain model: orders are partially incorporated

Markov chain model

- Sometimes we need to model dependencies between adjacent positions in the sequence
  - There are certain regions in the genome, like TATA within the regulatory area, upstream a gene.
  - The pattern CG is less common than expected for random sampling.
- Such dependencies can be modeled by Markov chains.

Markov chains

- A Markov chain is a sequence of random variables with Markov property, i.e., given the present state, the future and the past are independent.
- A famous example of Markov chain is the "drunkard's walk"—at each step, the position may change by +1 or −1 with equal probability.
  - Pr(5→4) = Pr(5→6) = 0.5, all other transition probabilities from 5 are 0.
  - these probabilities are independent of whether the system was previously in step 4 or 6.

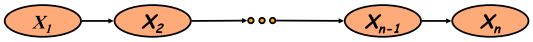
1st order Markov chain

*An **integer time stochastic process**, consisting of a set of **m>1** states $\{s_1,\ldots,s_m\}$ and*

1. *An **m** dimensional **initial distribution vector** ($p(s_1)$,..., $p(s_m)$)*
2. *An **m×m transition probabilities matrix M=** ($a_{s_i s_j}$)*

For example, for DNA sequence:
the states are {A, C, T, G} (m=4)
p(A) the probability of A to be the 1st letter
a_{AG} the probability that G follows A in a sequence.

1st order Markov chain



- *For each integer n, a Markov Chain assigns probability to sequences $(x_1 \ldots x_n)$ as follows:*

$$p((x_1,x_2,\ldots x_n)) = p(X_1 = x_1)\prod_{i=2}^{n} p(X_i = x_i \mid X_{i-1} = x_{i-1})$$

$$= p(x_1)\prod_{i=2}^{n} a_{x_{i-1}x_i}$$

Matrix representation

$$\begin{array}{c|cccc} & A & B & C & D \\ \hline A & 0.95 & 0 & 0.05 & 0 \\ B & 0.2 & 0.5 & 0 & 0.3 \\ C & 0 & 0.2 & 0 & 0.8 \\ D & 0 & 0 & 1 & 0 \end{array}$$

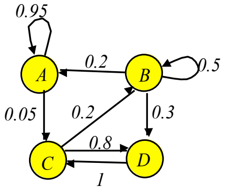
*The transition probabilities matrix **M** =(a_{st})*

***M** is a stochastic matrix:*

$$\sum_t a_{st} = 1$$

*The initial **distribution vector** $(u_1 \ldots u_m)$ defines the distribution of X_1 $(p(X_1 = s_i) = u_i)$.*

Digraph (directed graph) representation



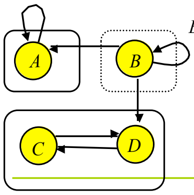
	A	B	C	D
A	0.95	0	0.05	0
B	0.2	0.5	0	0.3
C	0	0.2	0	0.8
D	0	0	1	0

*Each directed edge A→B is associated with the **positive** transition probability from A to B.*

Classification of Markov chain states

States of Markov chains are classified by the digraph representation (omitting the actual probability values)

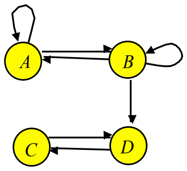
*A, C and D are **recurrent** states: they are in strongly connected components which are **sinks** in the graph.*



*B is not recurrent – it is a **transient** state*

<u>Alternative definitions:</u>
*A state **s** is **recurrent** if it can be reached from any state reachable from **s;** otherwise it is **transient**.*

Another example of recurrent and transient states



***A** and **B** are **transient** states, **C** and **D** are **recurrent** states.*

*Once the process moves from **B** to **D**, it will never come back.*

A 3-state Markov model of the weather

- Assume the weather can be: rain or snow (state 1), cloudy (state 2), or sunny (state 3)
- Assume the weather of any day *t* is characterized by one of the three states
- The transition probabilities between the three states

$$A = \{a_{ij}\} = \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = \begin{vmatrix} 0.4 & 0.3 & 0.3 \\ 0.2 & 0.6 & 0.2 \\ 0.1 & 0.1 & 0.8 \end{vmatrix}$$

- Questions
  - Given the first day is sunny, what is the probability that the weather for the following 7 days will be "sun-sun-rain-rain-sun-cloudy-sun"?
  - The probability of the weather staying in a state for *d* days?

Rabiner (1989)

CpG island modeling

- In mammalian genomes, **the dinucleotide CG** often transforms to (methyl-C)G which **often subsequently mutates to TG.**
- Hence **CG appears less than expected** from what is expected from the independent frequencies of C and G alone.
- Due to biological reasons, **this process is sometimes suppressed** in short stretches of genomes such as in the upstream regions of many genes.
- These areas are called **CpG islands.**

Questions about *CpG* islands

We consider two questions (and some variants):

Question 1: *Given a short stretch of genomic data, does it come from a CpG island ?*

Question 2: *Given a long piece of genomic data, does it contain CpG islands in it, where, and how long?*

We "solve" the first question by modeling sequences with and without CpG islands as Markov Chains over the same states {A,C,G,T} but different transition probabilities.

Markov models for (non) *CpG* islands

The "+" model: Use transition matrix $A^+ = (a^+{}_{st})$,
$a^+{}_{st}$ = (the probability that t follows s in a CpG island)
→ positive samples
The "-" model: Use transition matrix $A^- = (a^-{}_{st})$,
$a^-{}_{st}$ = (the probability that t follows s in a non CpG island sequence) → negative samples

*With these two models, to solve Question 1 we need to decide whether a given **short** sequence is more likely to come from the "+" model or from the "–" model. This is done by using the definitions of Markov Chain, in which the parameters are determined by training data.*

Matrices of the transition probabilities

A^+ (CpG islands):

$p_+(x_i \mid x_{i-1})$

(rows sum to 1)

X_{i-1} \ X_i	A	C	G	T
A	0.180	0.274	0.426	0.120
C	0.171	0.368	0.274	0.188
G	0.161	0.339	0.375	0.125
T	0.079	0.355	0.384	0.182

A^- (non-CpG islands):

X_{i-1} \ X_i	A	C	G	T
A	0.300	0.205	0.285	0.210
C	0.322	0.298	0.078	0.302
G	0.248	0.246	0.298	0.208
T	0.177	0.239	0.292	0.292

Model comparison

*Given a sequence **x**=(x_1....x_L), now compute the likelihood ratio*

$$\text{RATIO} = \frac{p(\boldsymbol{x} \mid + \text{model})}{p(\boldsymbol{x} \mid - \text{model})} = \frac{\prod_{i=0}^{L-1} p_+(x_{i+1} \mid x_i)}{\prod_{i=0}^{L-1} p_-(x_{i+1} \mid x_i)}$$

If RATIO>1, CpG island is more likely.
Actually – the log of this ratio is computed.

Note: $p_+(x_1|x_0)$ is defined for convenience as $p_+(x_1)$.
$p_-(x_1|x_0)$ is defined for convenience as $p_-(x_1)$.

Log likelihood ratio test

Taking logarithm yields

$$\log\ Q = \log \frac{p(x_1 \ldots x_L | +)}{p(x_1 \ldots x_L | -)} = \sum_i \log \frac{p_+(x_i | x_{i-1})}{p_-(x_i | x_{i-1})}$$

If logQ > 0, then + is more likely (CpG island).
If logQ < 0, then - is more likely (non-CpG island).

A toy example

- Sequence: CGCG
- P(CGCG|+) = ?
- P(CGCG|-) = ?
- Log likelihood ratio?

Where do the parameters (transition probabilities) come from ?

Learning from training data.

Source: A collection of sequences from CpG islands, and a collection of sequences from non-CpG islands.

Input: Tuples of the form $(x_1, \ldots, x_L, h)$, where h is + or -

Output: Maximum Likelihood parameters (MLE)

Count all pairs ($X_i=a$, $X_{i-1}=b$) with label +, and with label -, say the numbers are $N_{ba,+}$ and $N_{ba,-}$.

CpG island: question 2

Question 2: Given a long piece of genomic data, does it contain CpG islands in it, and where?

*For this, we need to decide which parts of a given **long** sequence of letters is more likely to come from the "+" model, and which parts are more likely to come from the "–" model.*

We will define a Markov Chain over 8 states.

A^+ C^+ G^+ T^+

A^- C^- G^- T^-

*The problem is that we don't know the sequence of **states** (hidden) which are traversed, but just the sequence of **letters (observation)**.*

***Hidden Markov Model**!*

Markov model variations

- *k*th order Markov chains (Markov chains with memory)
- Inhomogeneous Markov chains (vs homogeneous Markov chains)
- Interpolated Markov chains

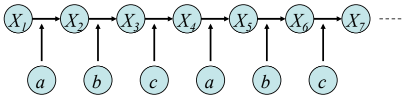
kth order Markov Chain (a Markov chain with memory *k*)

- *kth Markov Chain assigns probability to sequences $(x_1 \ldots x_n)$ as follows:*

$$p(x_1 \ldots x_n) = p(X_1 = x_1, \ldots, X_k = x_k) \cdot \prod_{i=k}^{n} p(X_i = x_i \mid X_{i-1} = x_{i-1}, X_{i-2} = x_{i-2}, \ldots, X_{i-k} = x_{i-k})$$

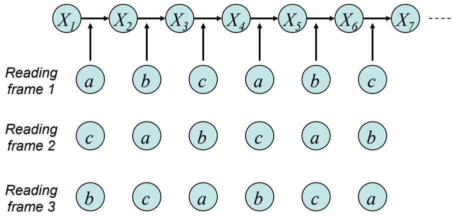
Initial distribution ***Transition probabilities***

Inhomogeneous Markov chain for gene finding



Again, the parameters (the transition probabilities, a, b, and c need to be learned from training samples)

Inhomogeneous Markov chain: prediction



Gene finding using inhomogeneous Markov chain

Consider sequence $x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_{9\ldots.}$
where x_i *is a nucleotide*

$$\text{let } p_1 = a_{x1x2} b_{x2x3} c_{x3x4} a_{x4x5} b_{x5x6} c_{x6x7} \ldots.$$
$$p_2 = c_{x1x2} a_{x2x3} b_{x3x4} c_{x4x5} a_{x5x6} b_{x6x7} \ldots.$$
$$p_3 = b_{x1x2} c_{x2x3} a_{x3x4} b_{x4x5} c_{x5x6} a_{x6x7} \ldots.$$

then probability that ith reading frame is the coding frame is:

$$P_i = \frac{p_i}{p_1 + p_2 + p_3}$$

Genemark (gene finder for bacterial genomes)

Selecting the order of a Markov chain

- For Markov models, what order to choose?
- Higher order, more "memory" (higher predictive value), but means more parameters to learn
- The higher the order, the less reliable the parameter estimates.
- E.g., we have a DNA sequence of 100 kbp
  - 2nd order Markov chain, 4^3=64 parameters, 1562 times on average for each history
  - 5th order, 4^6=4096 parameters, 24 times on average
  - 8th order, 4^9=65536 parameters, 1.5 times on average

Interpolated Markov models (IMMs)

- IMMs are called variable-order Markov models
- A IMM uses a variable number of states to compute the probability of the next state

simple linear interpolation

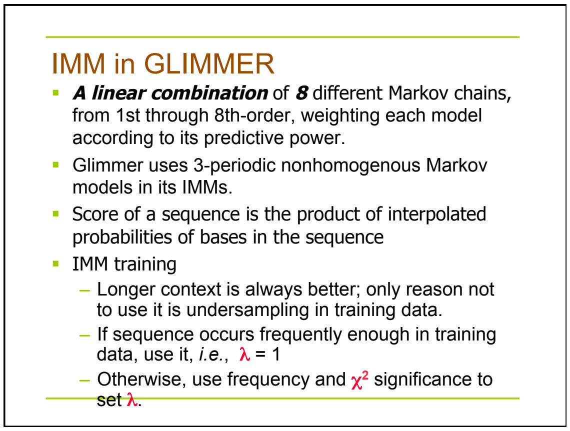
$$P(x_i|x_{i-n},\cdots,x_{i-1}) = \lambda_0 P(x_i) + \lambda_1 P(x_i|x_{i-1}) + \cdots + \lambda_n P(x_i|x_{i-n},\cdots,x_{i-1})$$

general linear interpolation

$$P(x_i|x_{i-n},\cdots,x_{i-1}) = \lambda_0 P(x_i) + \lambda_1(x_i) P(x_i|x_{i-1}) + \cdots + \lambda_n(x_{i-n},\cdots,x_{i-1}) P(x_i|x_{i-n},\cdots,x_{i-1})$$

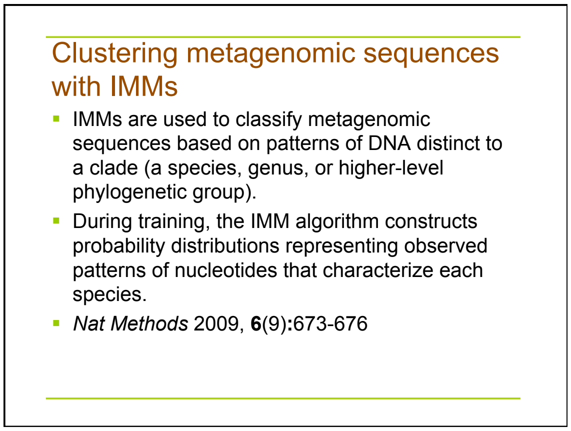
GLIMMER

- Glimmer is a system for finding genes in microbial DNA, especially the genomes of bacteria, archaea, and viruses
  - eukaryotic version of Glimmer: GlimmerHMM
- Glimmer (Gene Locator and Interpolated Markov ModelER) uses IMMs to identify the coding.
- Glimmer version 3.02 is the current version of the system (http://www.cbcb.umd.edu/software/glimmer/)
- Glimmer3 makes several algorithmic changes to reduce the number of false positive predictions and to improve the accuracy of start-site predictions



IMM in GLIMMER

- ***A linear combination*** of ***8*** different Markov chains, from 1st through 8th-order, weighting each model according to its predictive power.
- Glimmer uses 3-periodic nonhomogenous Markov models in its IMMs.
- Score of a sequence is the product of interpolated probabilities of bases in the sequence
- IMM training
  - Longer context is always better; only reason not to use it is undersampling in training data.
  - If sequence occurs frequently enough in training data, use it, *i.e.*, $\lambda = 1$
  - Otherwise, use frequency and χ^2 significance to set λ.



Clustering metagenomic sequences with IMMs

- IMMs are used to classify metagenomic sequences based on patterns of DNA distinct to a clade (a species, genus, or higher-level phylogenetic group).
- During training, the IMM algorithm constructs probability distributions representing observed patterns of nucleotides that characterize each species.
- *Nat Methods* 2009, **6**(9)**:**673-676