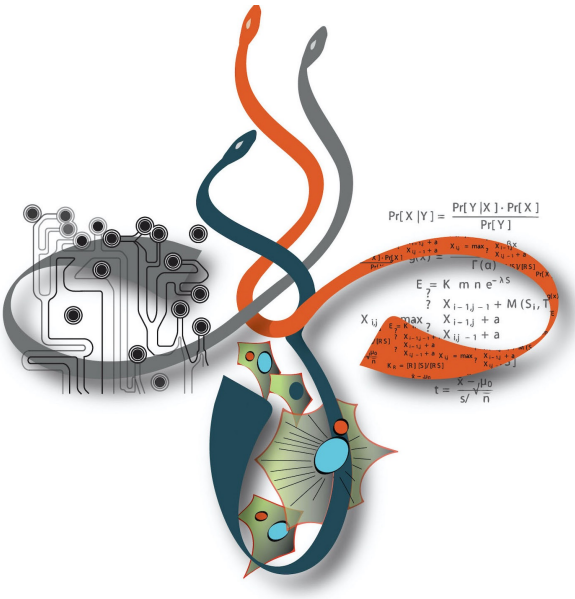
Hidden Markov Models (HMMs)

Some theory, some games, some applications

Mike Hallett
hallett.mike.t@gmail.com
http://mikehallett.science



The full course is available here

Lecture 14

BIOCHEM 3xxxA: Data science for the Life Sciences, Sept 16 2021

This booklet: https://hallett-biology-datascience.netlify.app/

R — you can work along with the examples.

Plan for the day

1. Markov Models (15 mins)

IMHO one of the top 3 concepts/tools for all life scientists.

Andrei Andreyevich Markov
1856-1922

Mathematician
Models of stochastic processes

2. Prokaryotic gene finding (10 mins)

3. Hidden Markov Models (15 mins)

It would be more appropriate to call them "partially hidden" or "noisely observable", but that's awkward.

Hidden Andrei Andreyevich Markov

4. Puzzles, exercises, and points of reflection (5 mins)

1. Markov Models

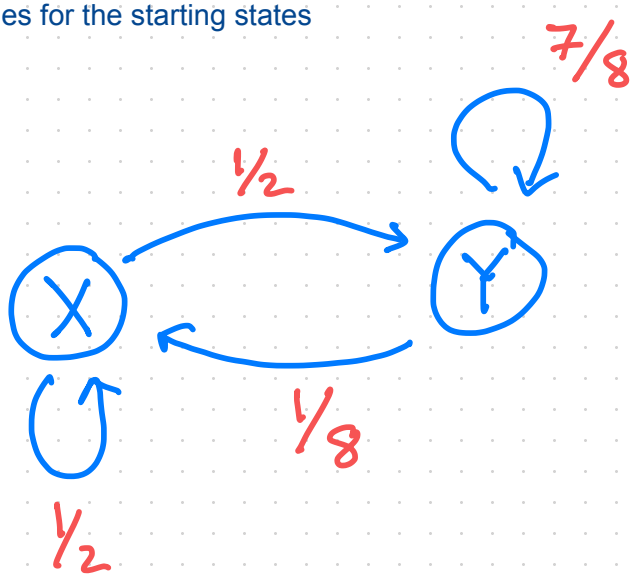
A Markov Model consists of 3 things

(1) A set of states

(2) Transition probabilities between states

(3) Probabilities for the starting states

Boring Example



Prob(starting in X)=4/5 Prob(starting in Y)=1/5

1. Markov Models

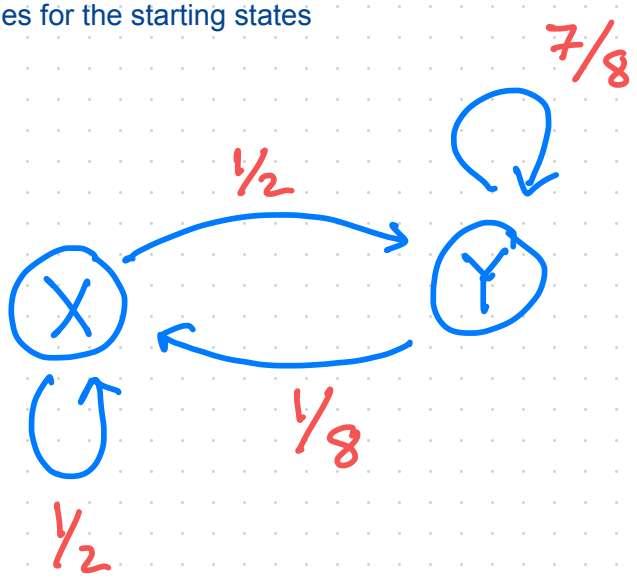
A Markov Model consists of 3 things

(1) A set of states

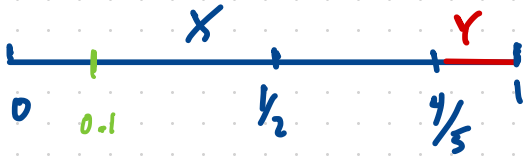
(2) Transition probabilities between states

(3) Probabilities for the starting states

Boring Example



Prob(starting in X)=4/5 Prob(starting in Y)=1/5



We can generate "walks" through the Markov model by choosing random numbers between 0 and 1.

Suppose the random number is 0.1.

walk

state: X

1. Markov Models

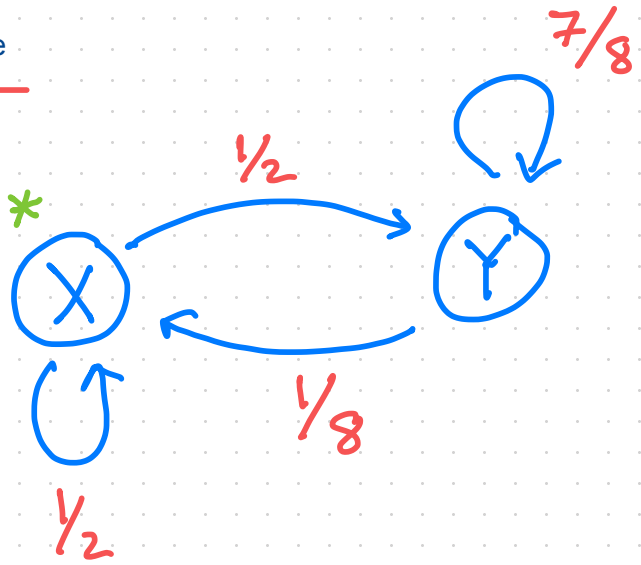
A Markov Model consists of 3 things

(1) A set of states

(2) Transition probabilities between states

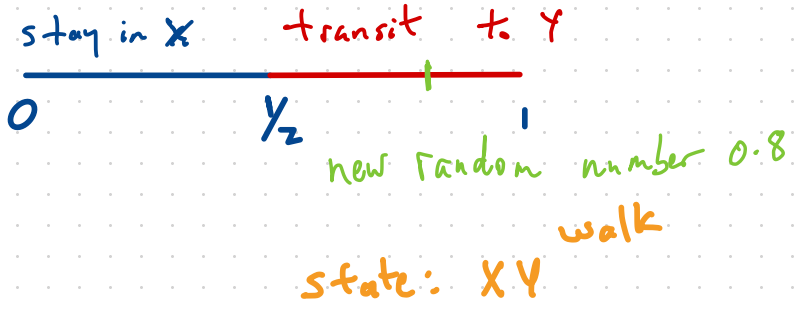
(3) Probabilities for the starting states

Boring Example



Prob(starting in X)=4/5 Prob(starting in Y)=1/5

We are in state X. We pick a random number to determine where next. ,'



1. Markov Models

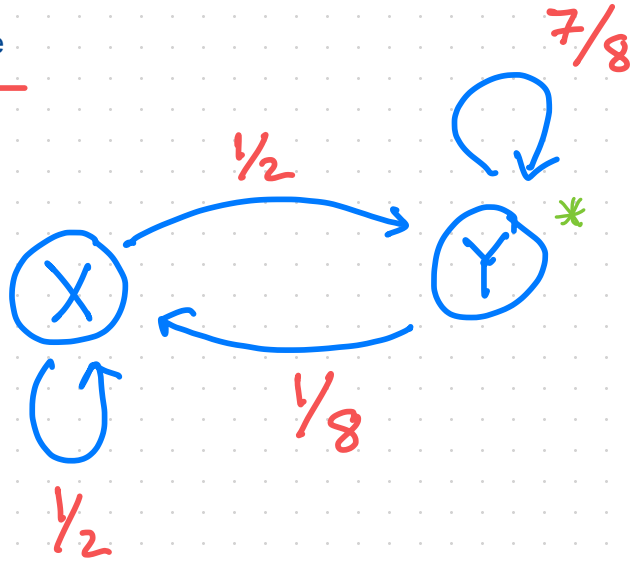
A Markov Model consists of 3 things

(1) A set of states

(2) Transition probabilities between states

(3) Probabilities for the starting states

Boring Example



Prob(starting in X)=4/5 Prob(starting in Y)=1/5

We repeat for as long as we want, each time picking a random number and using the transition probabilities to dictate the next step in our walk,

walk.

state: XYYYXXYX ···

A nice set of animations

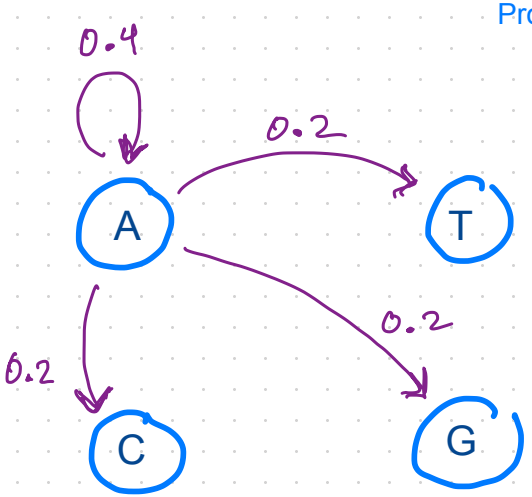
1. Markov Models

A Markov Model consists of 3 things

(1) A set of states — A, C, G, T

(2) Transition probabilities between states

(3) Probabilities for the starting states — Prob(start in A) = 1/4
Prob(start in T) = 1/4
Prob(start in C) = 0
Prob(start in G) =



The transition probabilities must sum to 1 for each node.
(otherwise they wouldn't be probabilities.)

Prob(Heads) + Prob(Tails) = 1
Prob(win lottery) + Prob(don't win lottery) = 1
Prob(dice is 1) + Prob(dice is 2) + … + Prob(dice is 6) = 1

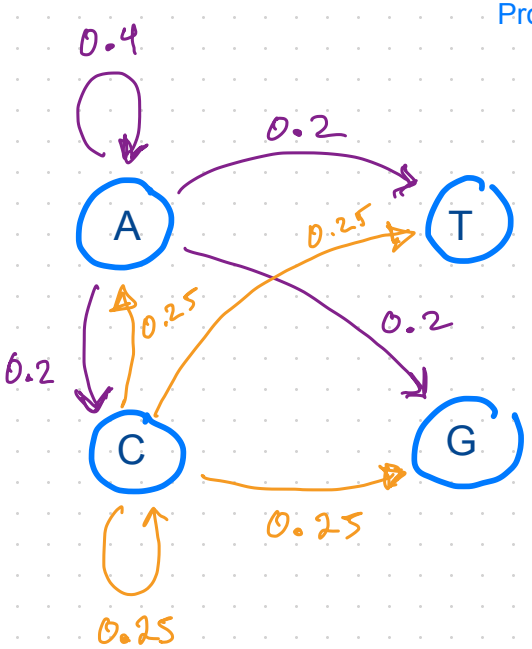
1. Markov Models

A Markov Model consists of 3 things

(1) A set of states — A, C, G, T

(2) Transition probabilities between states

(3) Probabilities for the starting states — Prob(start in A) = 1/4
Prob(start in T) = 1/4
Prob(start in C) = 0
Prob(start in G) =



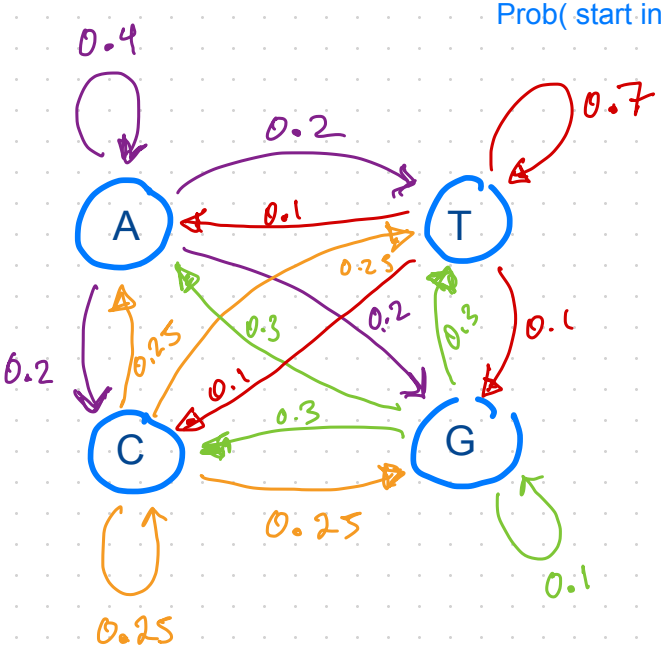
1. Markov Models

A Markov Model consists of 3 things

(1) A set of states — A, C, G, T

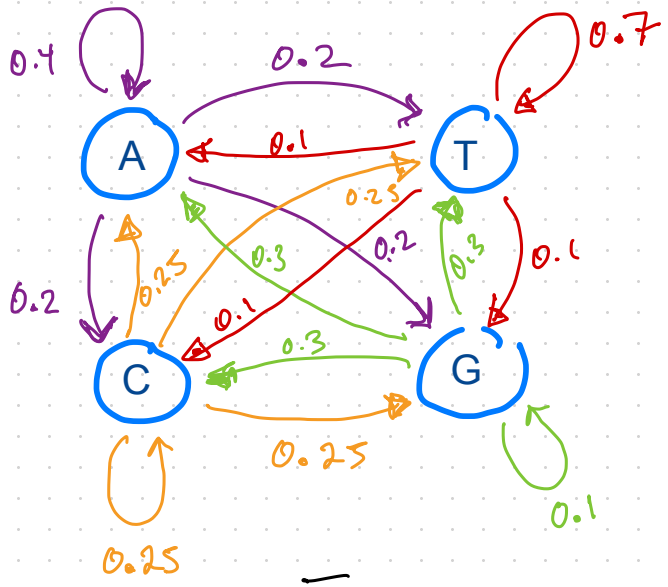
(2) Transition probabilities between states

(3) Probabilities for the starting states — Prob(start in A) = 1/4
Prob(start in T) = 1/4
Prob(start in C) = 0
Prob(start in G) =



1. Markov Models

A network can be a bit messy so sometimes we use a transition matrix (and this gets us ready to dig out all that old linear algebra).



=

to

from	A	C	G	T
A	0.4	0.2	0.2	0.2
C	0.25	0.25	0.25	0.25
G	0.3	0.3	0.1	0.3
T	0.1	0.1	0.1	0.7

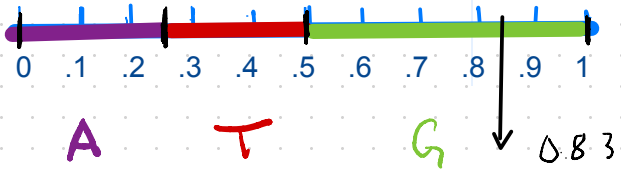
1. Markov Models

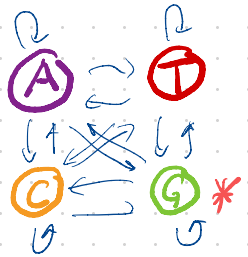
from \ to	A	C	G	T
A	0.4	0.2	0.2	0.2
C	0.25	0.25	0.25	0.25
G	0.3	0.3	0.1	0.3
T	0.1	0.1	0.1	0.7

Prob(start in A) = 1/4
Prob(start in T) = 1/4
Prob(start in C) = 0
Prob(start in G) =1/2

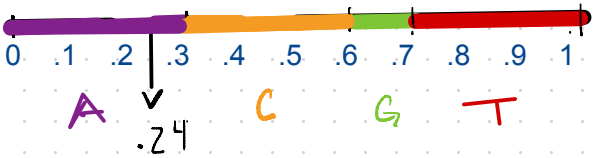
Let's create a random chromosome by walking through the Markov model.

Step 0: Pick a number at random to determine where to start



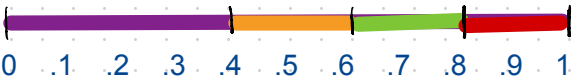


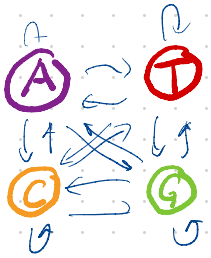
Step 1: Pick a random number to determine where to go from G



Step 2: Transit to A; Goto to Step 1.

Step 1: Pick a random number to determine where to go from A





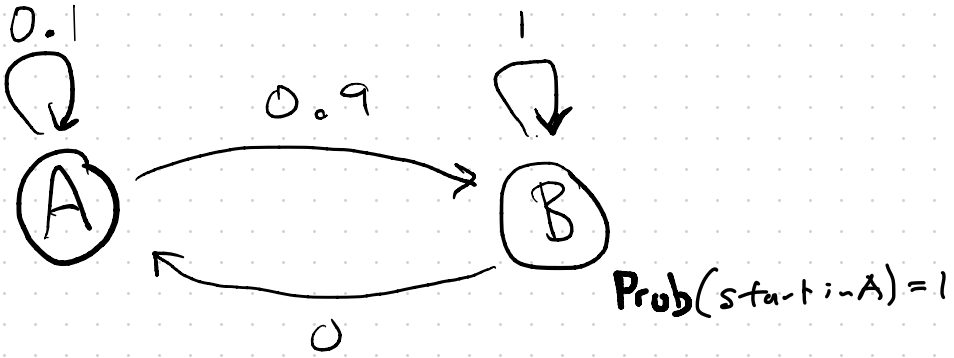
GA

1. Markov Models

Ok, so we could keep iterating like this and create a random gene, or chromosome or genome …

Here are some challenges to help your understanding of Markov models

Challenge 1: What does a random walk look like in this Markov model?



Challenge 2: Create a Markov model that repeats ABC an arbitrary number of times (could be 1 or more times) but the last one ends in X

ABCX

ABCABCABCX

ABCABCABCABCX

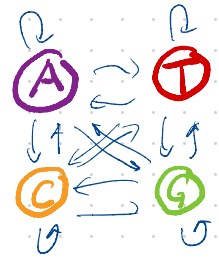
Only that sequence is allowed!
All other patterns are disallowed.

1. Inverting the Markov Models

If I give you a Markov Model as before and a gene,
how do you figure out the probability of that gene?

from \ to	A	C	G	T
A	0.4	0.2	0.2	0.2
C	0.25	0.25	0.25	0.25
G	0.3	0.3	0.1	0.3
T	0.1	0.1	0.1	0.7

Same model as before.



Prob(start in A) = .25
Prob(start in T) = .25
Prob(start in C) = 0
Prob(start in G) = .5

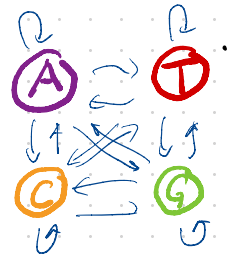
Bit small but good enough.

Gene of interest: TGCTCAAA

1. Inverting the Markov Models

If I give you a Markov Model as before and a gene,
how do you figure out the probability of that gene?

from \ to	A	C	G	T
A	0.4	0.2	0.2	0.2
C	0.25	0.25	0.25	0.25
G	0.3	0.3	0.1	0.3
T	0.1	0.1	0.1	0.7



Prob(start in A) = .25
Prob(start in T) = .25
Prob(start in C) = 0
Prob(start in G) = .5

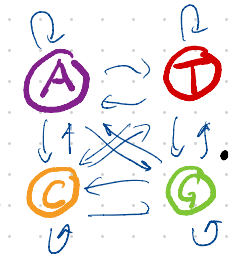
Gene of interest: TGCTCAAA

What is the probability of starting with state/nucleotide T? 1/4

1. Inverting the Markov Models

If I give you a Markov Model as before and a gene,
how do you figure out the probability of that gene?

from \ to	A	C	G	T
A	0.4	0.2	0.2	0.2
C	0.25	0.25	0.25	0.25
G	0.3	0.3	0.1	0.3
T	0.1	0.1	0.1	0.7



Prob(start in A) = .25
Prob(start in T) = .25
Prob(start in C) = 0
Prob(start in G) = .5

Gene of interest: TGCTCAAA

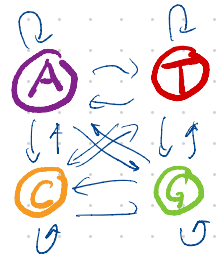
What is the probability of starting with state/nucleotide T? 0.25

We are in state T; what is the probability of transiting to G? 0.1

1. Inverting the Markov Models

If I give you a Markov Model as before and a gene,
how do you figure out the probability of that gene?

from \ to	A	C	G	T
A	0.4	0.2	0.2	0.2
C	0.25	0.25	0.25	0.25
G	0.3	0.3	0.1	0.3
T	0.1	0.1	0.1	0.7



Prob(start in A) = .25
Prob(start in T) = .25
Prob(start in C) = 0
Prob(start in G) = .5

Gene of interest: TGCTCAAA

What is the probability of starting with state/nucleotide T? 0.25

We are in state T; what is the probability of transiting to G? 0.1

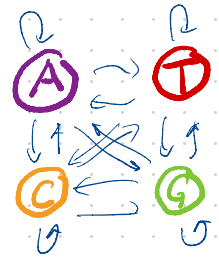
We are in state G; what is the probability of transiting to C? 0.3

(And so on and so forth for the remainder of our baby gene)

1. Inverting the Markov Models

If I give you a Markov Model as before and a gene,
how do you figure out the probability of that gene?

from \ to	A	C	G	T
A	0.4	0.2	0.2	0.2
C	0.25	0.25	0.25	0.25
G	0.3	0.3	0.1	0.3
T	0.1	0.1	0.1	0.7



Prob(start in A) = .25
Prob(start in T) = .25
Prob(start in C) = 0
Prob(start in G) = .5

Gene of interest: TGCTCAAA

What is the probability of starting with state/nucleotide T? 0.25
We are in state T; what is the probability of transiting to G? 0.1
We are in state G; what is the probability of transiting to C? 0.3
We are in state C; what is the probability of transiting to T? 0.25
We are in state T; what is the probability of transiting to C? 0.1
We are in state C; what is the probability of transiting to A? 0.25
We are in state A; what is the probability of transiting to A? 0.4
We are in state A; what is the probability of transiting to A? 0.4

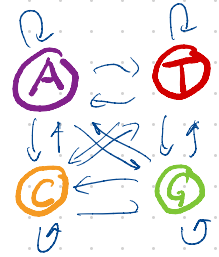
So we want that probability that all these things happen. This is the joint probability.

Prob(T) * Prob(T to G) * Prob(G to C) * Prob(C to T) * Prob(T to C)
* Prob(C to A) * Prob(A to A) * Prob(A to A)

= 0.25 * 0.1 * 0.3 * 0.25 * 0.1 * 0.25 * 0.4 * 0.4 = 0.0000075

Challenge #3: Using the same Markov Model, calculate the probability of the following sequence: GCAACTAG

from \ to	A	C	G	T
A	0.4	0.2	0.2	0.2
C	0.25	0.25	0.25	0.25
G	0.3	0.3	0.1	0.3
T	0.1	0.1	0.1	0.7



Prob(start in A) = .25
Prob(start in T) = .25
Prob(start in C) = 0
Prob(start in G) = .5

1. Inverting the Markov Models

If I give you a Markov Model as before and a gene,
how do you figure out the probability of that gene?

Prob(Gene of Interest TGCTCAAA) = 0.0000075

Why in the world is this even remotely interesting or important?

Fair question. First, it's true. We typically don't care about the probably of 0.0000075 itself. But. However.

Usually the Markov Model is built in a way that it captures some salient aspect of biology.

For example, we could build a Markov Model to capture the essence of "coding DNA"

Challenge #4: How would you build such a Markov model for coding DNA of Baker's yeast? That is, how would you determine the transition probabilities and the initial probabilities for coding DNA in Baker's Yeast?

So that probability measures to some extent how "realistic" a nucleic acid sequence is and how likely it would actually occur in nature.

This is at the heart of today's example of using Hidden Markov Models to find genes in genomes.

Challenge #4 corresponds to Assignment 3, Question #1 where you are asked to do this in R for Chromosome 1 of Baker's Yeast.

2. The Gene Finding Problem

Candida albicans SC5314 chromosome 1

```
GAGTCACGCCAATCACAAATTCCTTTGAAAAACTTGATTCGACCACATTCACAAGTTTGATTGATTTGAA
AAACTTGATTCGACACCATCCTGCTGTCCATCCGTGAGCCACACAGATTCAGAATTGAGTCGCTGACTAA
GCGGTTAGACATACGTGATATTCACCGACTTTGAGAGTCCCACTAATCGGCTAGACATACGTAAATTACA
TAGCTCCCTCCAATACACACCCTACTTACTATTGTCTTTTTTTAACTTTTTCGTAATCTCTACCCATAAA
AATACACTTTCCCTCCAAATCTCTAATTTACAACTCAACTGAACTTTAATTAACCTCTACTGCCTTAATT
TAAGCTTATTTCTTGTCTATCAGCTGTTTCTGTTTCACCATTTTCACAACTTCTCCCCTAGGTGACATTT
TTTTCTGCTGATTTTTTCTCAAATTCAGCCCAAAAAACTTAAACCAAAACTCAAAATTACAACGCAAACT
CTATTTAGAGTGCCCCTACTACCCCTACTGAGTCTTATTTTGAGTTTACCACCGATTTCTGTGCTCCTCC
TGTCTCCAGATTTCCGGTCTTCGTTCTTTTTTCGATCGAAAACTTTGTAAAACTAAACTAAAAAATTCAC
TCCATTTGACCAACAAASTGCTCAAAATCAGACCAGGCTCACTGCTTCTGCTTTGTCCCTAAAGATTACA
AAAGCTACGCTGCAAAAGAACTTAAAATTGCGTTCCATTATAATCTATACACACCCATCTCCTGCTATCA
CTTCACCTCACGTCCTCCCTGCGCTTGTCCATCCGTGAGTTCAACTACCGCCTCCCTCTTCCCTTGTCCA
CCCGTGATTCGCCAGTCCCTGGCTCTCCATCTTCCACAGATCCTTCACTTGCTTTCCATTGACTATCTTC
TTCTCTTGCCCTAGCTTTTGATTTCCATATTCCTTCAACCATTGTACTAACTCTCTCTTTACTCTGTGCT
TAACTACTATCTCTCTGATCACCTGGCCTGGCGTTATTCTATTTCCAGTTTTTTTTTTTTTCATTGATCC
AACACAACTTCAACTCCCATTCGCTCGGCTCTTGACCCCCTTATCCATTCTCTCAGTACTTCCCGATCCC
TTTTGTTCTTCATTACCCTTTTCTCTGTCTTGCCCTGCTACCCATCCGTGATTYTCCAGCRCTGTTCACT
CCCACGTCCCCGCTGTTGATTGACATTTCCAATTTCACTGACTTTGTTCCCCTACTTTTGCTCACATTTT
TCTGTTCTCAAACTCCTCTCTTGAATTCTCAGCTTGCTGTGTCTCCTTCTTGCCATTACAACTGCTTTTC
TTCACTTGCTTCCTTCTGCTTTGACAACACTGATCATTGACTTGATTTCATTACTTTTCACAAACCCAGT
TTCTAGCTCTATTGACTTCCTCTGCTATCCAGATTTCAAACTTCTTATTGTAACAGTTATAACTGCGTTC
TTCATCTCATCTAATTGATTGATTTGTTGTCGTTGAAGAAAAGTGATATTTTTTGACCAGCACATTTCTT
GTCCAACTTTTTTTCGATGWCTTCTCCACACTTTTCTGCCACGTTTTCCCTATTTTTTTTGCCACGTCAG
AAAAAAAAAAATTTTTTTCACCACTTTTCTTCCCACCGCCAACAACACCAATGATGTTCTACCTGCCAGA
GTGCCAGTTCTACATATGTTCCGATTTCCTAGCTCTTCAGATTCAGCAACTCCAACTACCAATTTTTGAA
TTCCCACAATCCAACTAATTCCCCGCCATCTTGCMAACTCAGTCCACAATTTCTGTCCAACYACAAATTT
TCAAACTGCAACAACTGTCACTGCCACATGCTATTCAACCGGCAAACAWACGAARCTGTAATGATTTCAA
CAACTGCCATTGATCACTCATTTATCAACCACCAAACACAGCAGCGCAACAGCTTCCACAGTTCTTGTTG
CCACGATTTCGGCAACTACGATTGACTAKTGATTTTTCAGCCAGCAAACACAACTGCTTTGACAACAGCA
AATACAACGAGATACACAACATGCATCGACAACTCCCTCCACAGTTCGTGTTGAATTTCCCATTGCCACT
ATGTTCAATTTTCGACACTGYCATTGACAACGAGATACACAACTGCTTCCACATTTCGTGTTGCATTTCC
CACTGCCATCAACTAGCAAGCACAACATGCATCGACAACACCCTCCACAGTTCGTGTTGCATTTCCCATT
GACATAGTTTATTTGCACTTGCCACAACCAGCAAGCACAACTGCATTGACWACACCCTCCTCATTTCGTG
TTGCATTCCWCCAGTTGTCATCAATCAKCCACGGGTTGTTCTACTTTTGATTGTTCAGCCAGCAAACACA
ACCACAACTGCTTTGACTACACCCTCTTCATTTCGTGTTGCAATTCCCAYTACCACTAGGTTCACATTTC
CCACCGCCATTGACTACTCAAACTACAAGTTGTTCTATCGTCCCTTCTCCAACYAGCAAGCACAACGAGA
TACATGTCTGGGCATTTACAATAGCTTCTACTCATCATTTTGCATCTGCCATGCAATCTGCCCACCACCC
ATCATCCAACCAGCAAACACAACCGCAACGGGCATTGACAACTGCTTCCACTGCTATGACACCACCACTG
ACTACATGTTGTTCACCCAGCAAACATAACACCTTGCACAGTTCAAGTTCAATTTCCCATTCTACAACTG
CAATTTCTACTGGGTCCCCGAGCAGTTTGACTTCCGTAAAATACACCACCCCACAGATCAACTATCCCCY
GCCGGCTTGACTTCCGTAAAATACACTACAAAGCCTACCCCTTGTCTGACTACCCCTCAGTCCCACAGAT
CAACTATCCCCYGCCGGCTTGACTTCCGTAA ...
```

etc. etc. etc. Yada Yada Yada

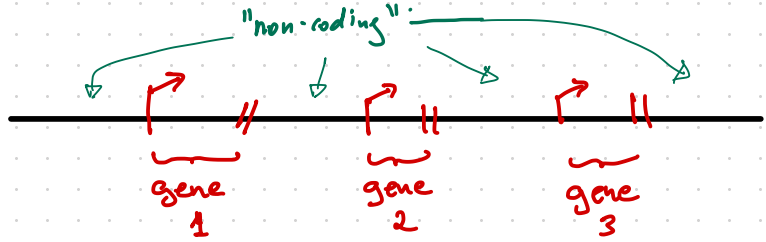
for 3.18 million base pairs

Let's simplify a bit here:

2. The Gene Finding Problem

We are given an unannoted genome. Think of it as a long linear chromosome.

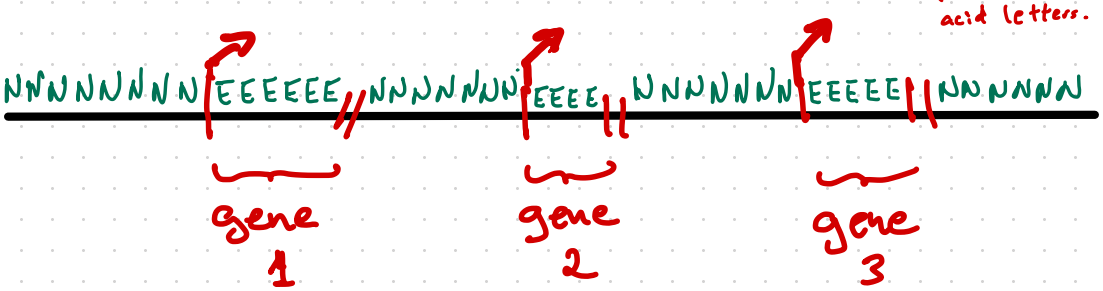
The goal is to find those regions that code for genes.



For simplicity of exposition, let's assume that genes are really simple (eg no introns)

We can think of walking along the chromosome, annotating each position as coding (E) or non-coding (N)..

I use E because G(gene) & C (coding) conflict with nucleic acid letters.



Before gene finding let's start with a simple example.

3. Hidden Markov Models (HMMs)

An HMM is a Markov Model that emits symbols at each state with different probabilities.

Let's build one for this example:

They look identical!

- You are at a casino and the dealer has two coins.
- One coin is fair: 50% Heads and 50%Tails.
- One coin is biased: 90% Heads and 10% Tails.

MARKOV MADNESS

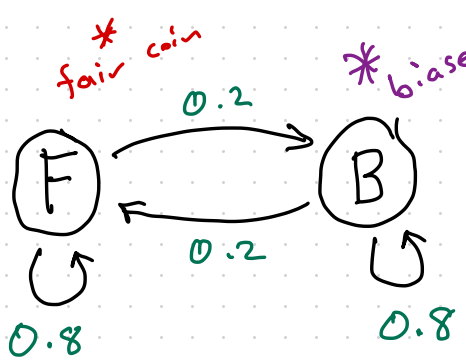
The dealer uses the following algorithm:

- 0. Pick the fair coin with 50% probability in secret.

Now repeat the following 10 times

1. Flip the coin in public and make the result visible.
2. In secret, keep the same coin with probability 80%; otherwise swap.
3. Go to Step 1.

GOAL: For each of the 10 coin tosses, guess which coin she used.



- Two states F, B = two coins.

- Prob(start in F)=0.5

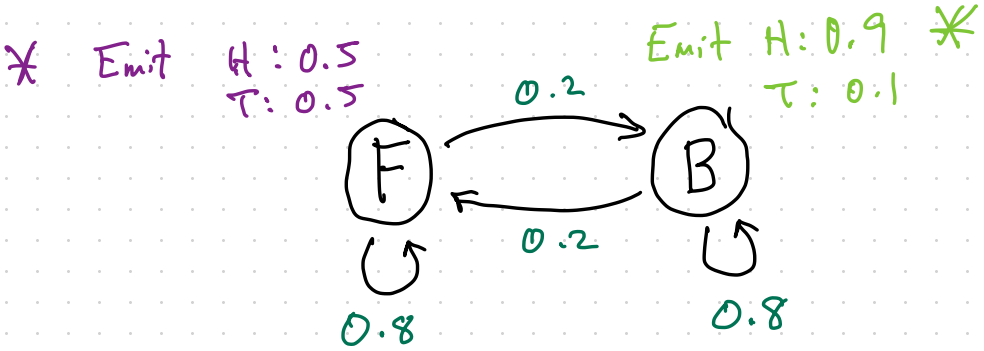
3. Hidden Markov Models (HMMs)

An HMM is a Markov Model that emits symbols at each state different probabilities.

You are at a casino and the dealer has two coins.
One coin is fair: 50% Heads and 50%Tails.
One coin is biased: 90% Heads and 10% Tails.

0. Pick the fair coin with 50% probability in secret.
Now repeat the following 10 times
1. Keep the coin in your hand with probability 80%; otherwise swap.
2. Flip the coin in public and make the result visible.
3. Go to Step 1.

GOAL: For each of the 10 coin tosses, guess which coin she used.



A walk in an HMM from the dealers perspective:

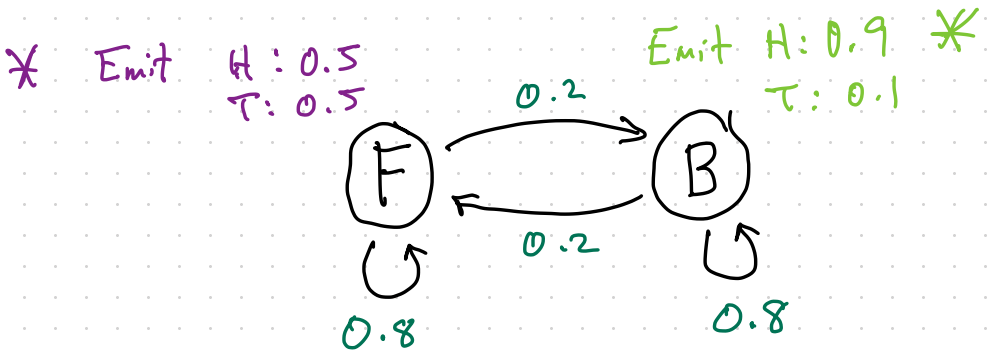
state:	F	F	B	B	B	B	F	F
emissions:	T	H	H	H	T	H	T	H

But the player sees only the emissions.
States are hidden.

The player's goal is to guess this.

state:	[redacted]							
emissions:.	T	H	H	H	T	H	T	H

3. Hidden Markov Models (HMMs)



Challenge #5:

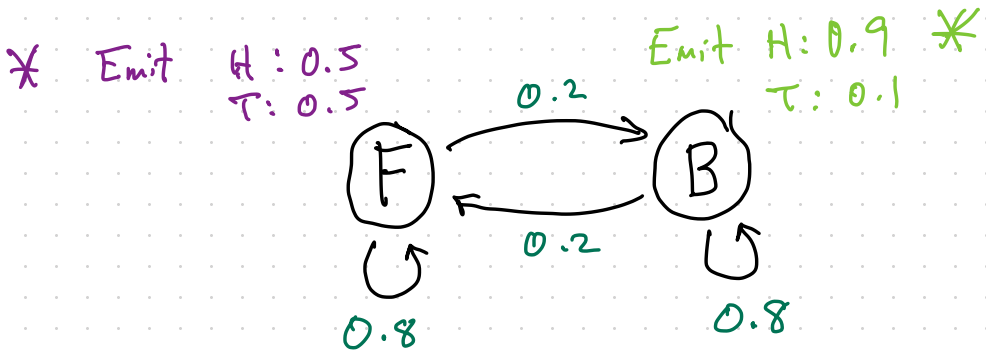
What would be your guess for states from the following emissions?

emissions : T H H H T H H H H T T H A H H

states : ?

What is the worst guess for states for the same sequence? Why did you chose it?..

3. Hidden Markov Models (HMMs)



It's easy for the dealer to compute the probability because they know both the states and the omissions

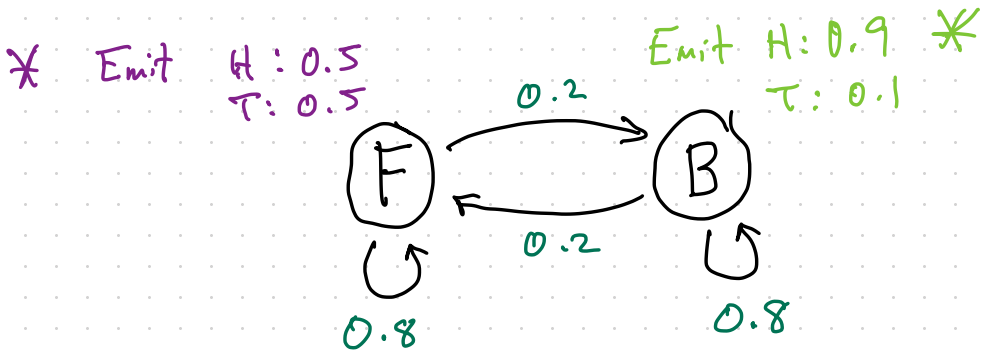
$$\text{Prob}\left(\begin{array}{rl}\text{state:} & F\ F\ B\ B\ B \\ \text{emissions:} & T\ H\ H\ H\ H\end{array}\right)$$

$$= \text{Prob}(\text{start in } F) \cdot \text{Prob}(\text{emit } T \text{ in } F) \cdot \text{Prob}(\text{stay state } F)$$
$$\cdot \text{Prob}(\text{emit } H \text{ in } F) \cdot \text{Prob}(F \text{ to } B)$$
$$\cdot \text{Prob}(\text{emit } H \text{ in } B) \cdot \text{Prob}(\text{stay in } B)$$
$$\cdot \text{Prob}(\text{emit } H \text{ in } B) \cdot \text{Prob}(\text{stay in } B)$$
$$\cdot \text{Prob}(\text{emit } H \text{ in } B)$$

$$= \frac{1}{2} \cdot \frac{1}{2} \cdot 0.8 \cdot \frac{1}{2} \cdot 0.2$$
$$\cdot 0.9 \cdot 0.8 \cdot 0.9 \cdot 0.8 \cdot 0.9$$

$$= 0.00933$$

3. Hidden Markov Models (HMMs)



But not easy for the player without knowing the states.....

$$\text{Prob}\Big(\begin{array}{rl}\text{state:} & \\ \text{emissions:.} & T\ H\ H\ H\ H\end{array}\Big)$$

Already for a walk with 5 nucleotides, there are 2^5 different state combinations

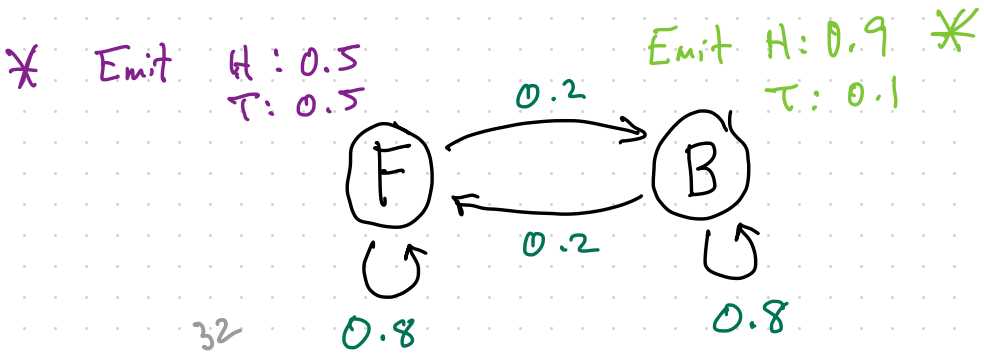
Case 1

$$\text{Prob}\Big(\begin{array}{rl}\text{state:} & F\ F\ F\ F\ F \\ \text{emissions:.} & T\ H\ H\ H\ H\end{array}\Big)$$

One of 32 possibilities is that the dealer always used the fair coin.

$$= \frac{1}{2}\cdot\frac{1}{2}\cdot\frac{4}{5}\cdot\frac{1}{2}\cdot\frac{4}{5}\cdot\frac{1}{2}\cdot\frac{4}{5}\cdot\frac{1}{2}\cdot\frac{4}{5}\cdot\frac{1}{2} = \frac{1}{2^6}\cdot\left(\frac{4}{5}\right)^4 = 0.0064$$

3. Hidden Markov Models (HMMs)



There are 2^5 different possibilities (just for 5 nucleotides!)

Case 1

$$\text{Prob}\left(\begin{array}{rl} \text{state:} & F\ F\ F\ F\ F \\ \text{emissions:.} & T\ H\ H\ H\ H \end{array}\right) = 0.0064$$

Case 2

$$\text{Prob}\left(\begin{array}{rl} \text{state:} & F\ F\ F\ F\ B \\ \text{emissions:.} & T\ H\ H\ H\ H \end{array}\right) = 0.0029$$

Case 2

$$\text{Prob}\left(\begin{array}{rl} \text{state:} & F\ F\ F\ B\ F \\ \text{emissions:.} & T\ H\ H\ H\ H \end{array}\right) = 0.0007$$

Case 2

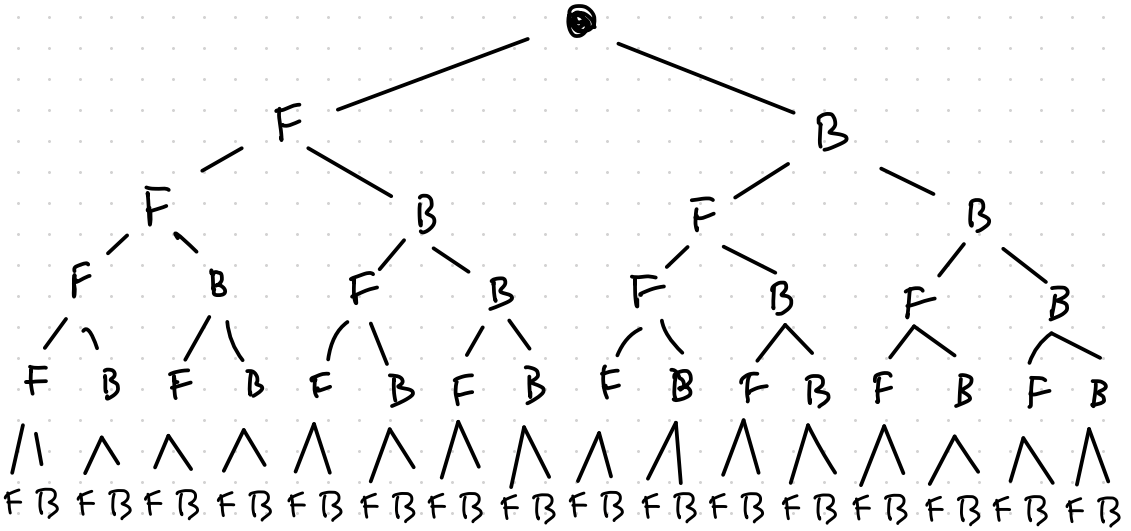
$$\text{Prob}\left(\begin{array}{rl} \text{state:} & B\ B\ B\ B\ B \\ \text{emissions:.} & T\ H\ H\ H\ H \end{array}\right) = 0.0134$$

3. Hidden Markov Models (HMMs)

Because the states are hidden from the player, the player has to consider all possibilities and choose the state sequence with the highest probability

This answer has the maximum likelihood of being correct

seq. of length 5



2^n possibilities.

$n=5$, 32

We want the one with the highest probability.

Which one has max prob?

Only 2^{250} molecules in the universe.

3. Hidden Markov Models (HMMs)

The Viterbi algorithm

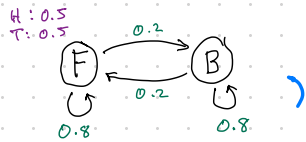
Beyond the scope of this course

Beautiful, elegant algorithm that finds the most likely state sequence

Input: a HMM and a emission sequence

Output: a state sequence with max probability

Really fast!! One of the important algorithms known

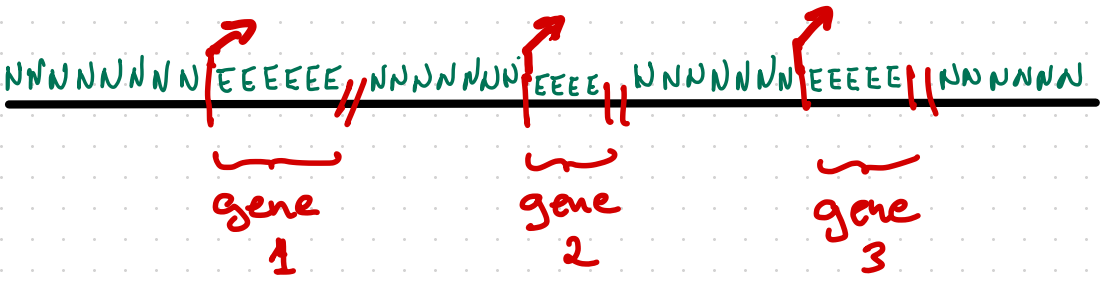
Input : 

, HHTH ··· HTTT

Output : states FFFB ··· FBBB

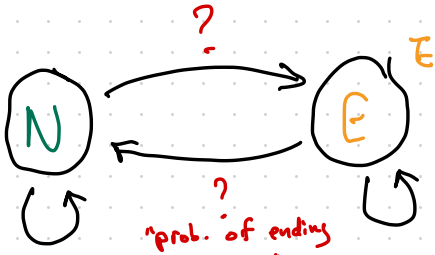
3. Hidden Markov Models (HMMs) and Gene Finding

How might we set up an HMM for gene finding?



"probability of starting a gene"

N = non-coding emissions



E = coding/exon emissions

A	?
C	?
G	?
T	?

Prob(start in coding E) = ?

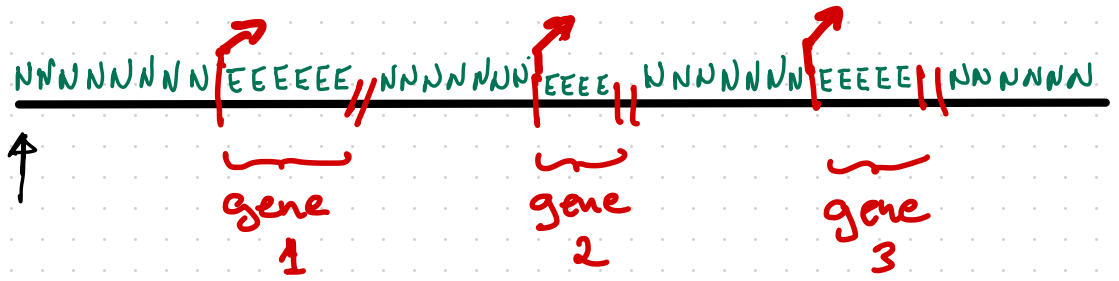
A	?
C	?
G	?
T	?

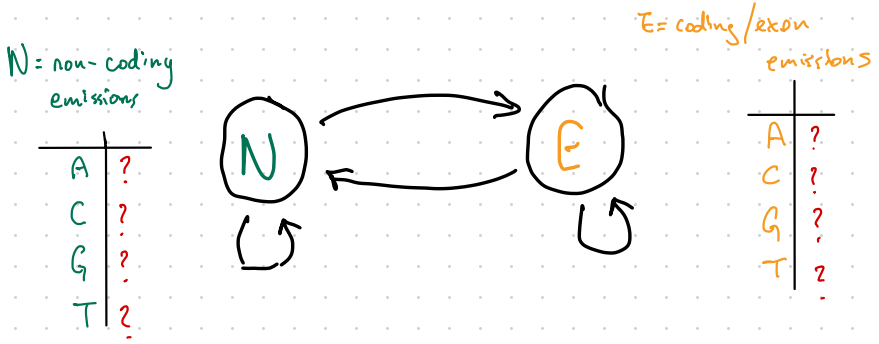
Does this initial probability even matter?

HINT: A chromosome might be millions of base pairs long.

3. Hidden Markov Models (HMMs) and Gene Finding

How might we set up an HMM for gene finding?



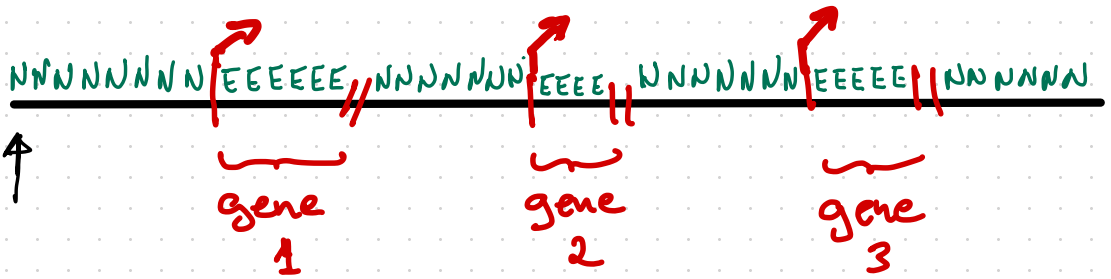


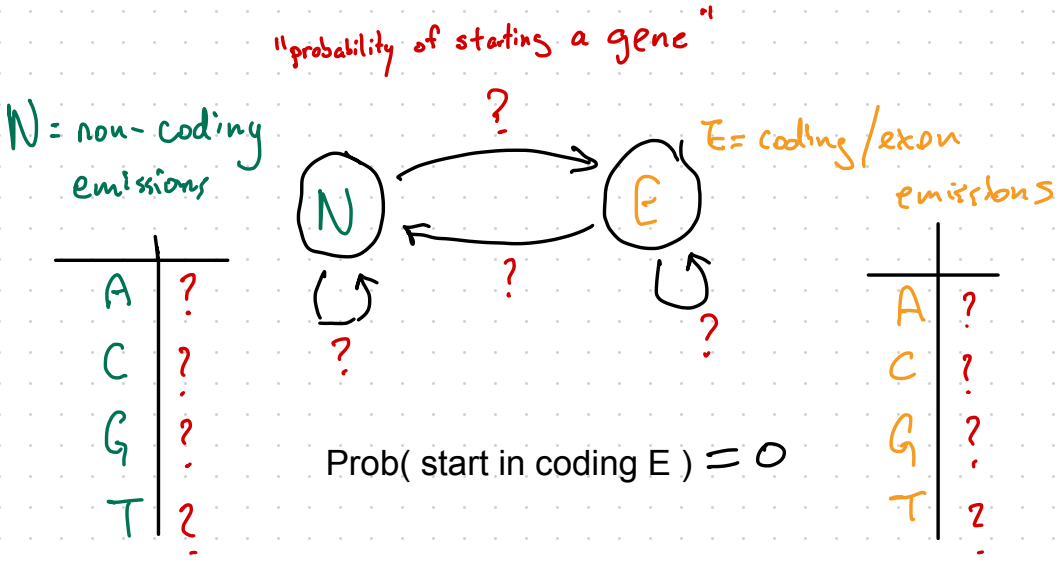
Prob(start in coding E) = 0

Nah. Start in non-coding.

3. Hidden Markov Models (HMMs) and Gene Finding

How might we set up an HMM for gene finding?





For the remaining transition probabilities we need training data.

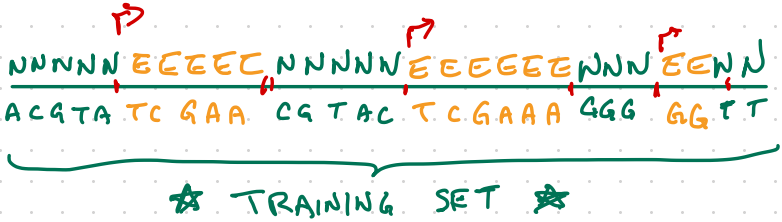
For example, if we are working with an obscure fungus,
we might use a well annotated genome like Baker's yeast
to estimate these parameters.

This is called a "learning set", a concept central in machine learning.

3. Hidden Markov Models (HMMs) and Gene Finding

So well studied we know where genes are

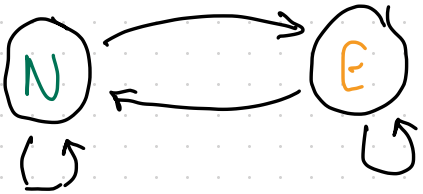
BAKER'S YEAST



How do we estimate the non-coding emissions?

N = non-coding emissions

A	$N_A/\#N = 3/15$
C	$N_C/\#N = 3/15$
G	$N_G/\#N = 5/15$
T	$N_T/\#N = 4/15$



$\#N$ = total number of non-coding nucleotides. (=15)

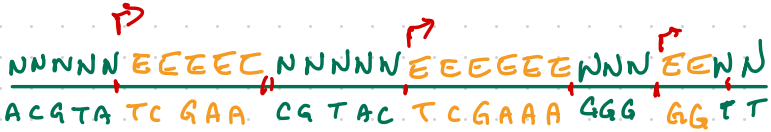
N_A = total number of non-coding A's (=3)

Same for N_T, N_C, N_G.

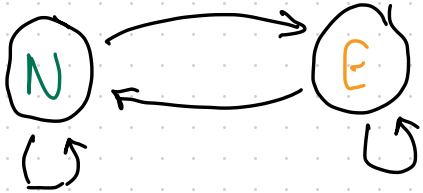
3. Hidden Markov Models (HMMs) and Gene Finding

So well studied we know where genes are

BAKER'S YEAST



Do the analogous for coding emissions,



A	5/13
C	2/13
G	4/13
T	2/13

$\#E$ = total number of coding nucleotides $(=13)$

E_A = total number of coding A's. $(=5)$

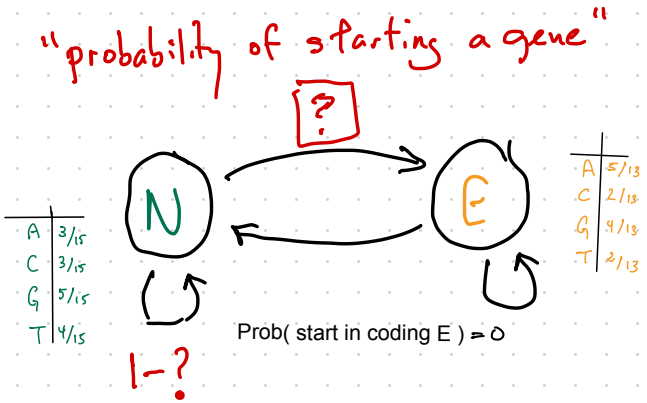
E_C, E_G, E_T analogous.

3. Hidden Markov Models (HMMs) and Gene Finding

BAKER'S YEAST

NNNNNEEEEENNNNNEEEGEENNNEENN

ACGTATCGAACGTACTCGAAAGGGGTT



$$\text{Prob}(\text{N to E}) = \frac{\#\text{ of genes}}{\text{length of genome}}$$

$$\left(= \frac{3}{28} \right)$$

In other words, of all positions in the genome, only 3 start a gene.

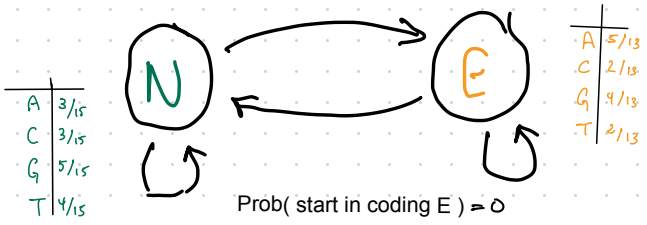
3. Hidden Markov Models (HMMs) and Gene Finding

BAKER'S YEAST

NNNNNEEEEENNNNNEEEGEENNNEENN

ACGTATCGAACGTACTCGAAAGGGGTT

"probability of ending a gene"

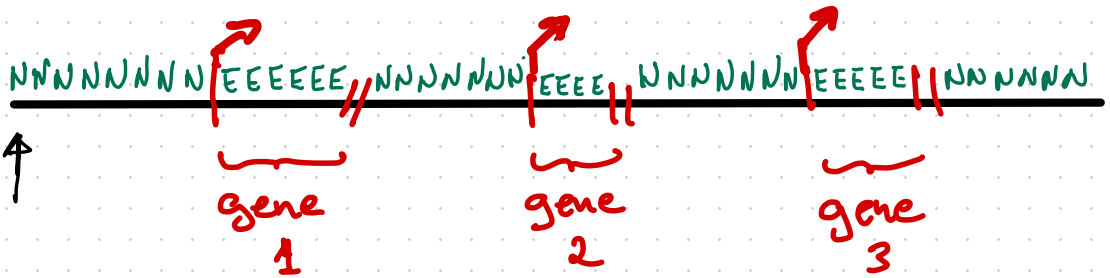


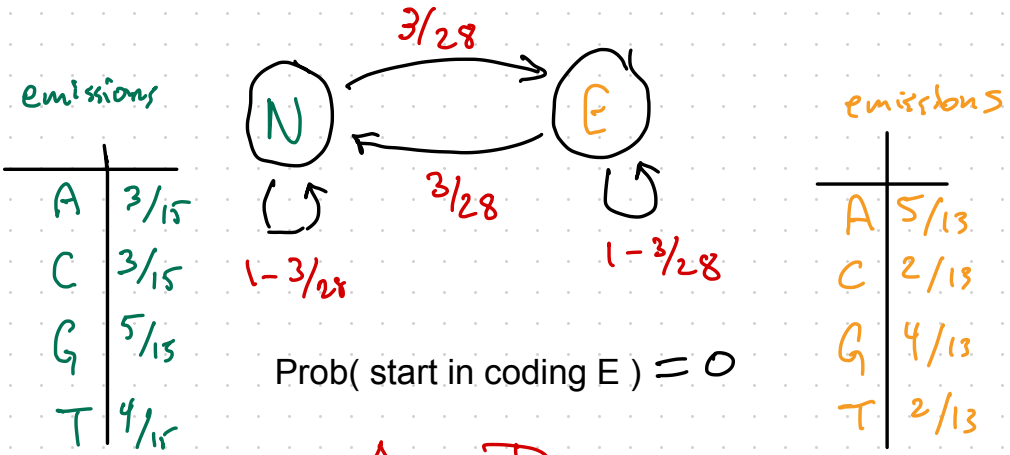
$$\text{Prob}(\text{E to N}) = \frac{\#\text{ of genes}}{\text{length of genome}}$$

$$\left(= \frac{3}{28}\right)$$

"For every start, there is an end and vice versa" ancient proverb

3. Hidden Markov Models (HMMs) and Gene Finding





Prob(start in coding E) = 0

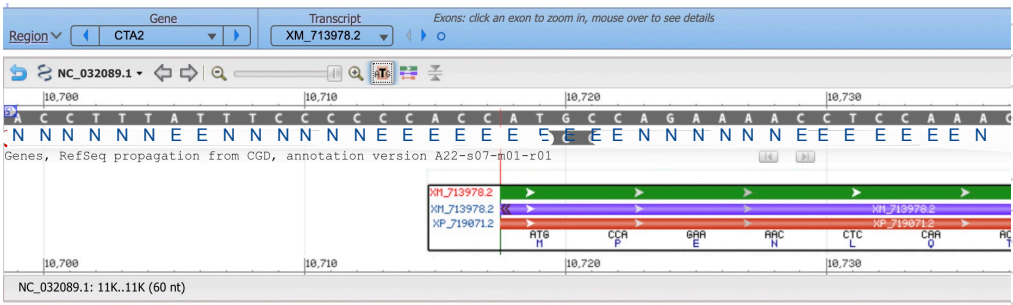
ALL DONE

etc. etc. etc. Yada Yada Yada

← Apply it to our new unannotated genome. (Homework exercise)

3. Hidden Markov Models (HMMs) and Gene Finding

Note how there are more Es in the gene region than outside. Noisy but …



Software and Resources

Toolkit for bio-sequence analysis using HMMs:
hmmer.org

The package rhmmer gives you access to it in R.
(My course uses the HMM package in R though.)

An alternative non-math and non-bio presentation for HMMs: Louis Serrano

More math-ee but still accessible:
https://towardsdatascience.com/markov-chains-and-hmms-ceaf2c854788

There are more mathematically rich HMM tools (not necessarily specific to bio):

R packages:	msm, depmixS4, momentuHMM
Python:	scikit-learn, HMMLearn
Julia:	HMMBase

RStudio learn Quiz (R, Python or Julia)

Hidden Markov Models

Start Over

The following questions should help you understand if you understood the lecture material:

Quiz

Which of the following statements are correct (Markov Models):

- [ ] The sum of the probabilities of all transitions to a node X must sum to one.
- [ ] The starting state probablities must be equal.
- [ ] Transition probabailities may be 0.
- [ ] The probability of any walk is greater than or equal to 0.

Submit Answer

Which of the following are correct (Hidden Markov Models):

- ○ The emission probabilities are not necessarily equal.
- ○ The same symbols must be emitted at each state
- ○ The sum of emission probabilities at each node must sum to 1.
- ○ The most likely walk found by Viterbi is always the correct true walk.

Submit Answer

Create a random walk with the following two-state HMM (use the runif function):

```
## [1] "Transition probs: "
```

```
##      X   Y
## X 0.3 0.2
## Y 0.7 0.8
```

```
## [1] "Emissions:"
```

```
##      X    Y
## A 0.1 0.25
```

Assignment #4

You might consider (but it is not mandatory) using R Markdown to write your answers.

50 total marks.

Question 1 [points 10] Using the S. cerevisiae (Baker's yeast) data that we imported into R in Lectures 13 and 14, show R code of how you would estimate the frequency of A, C, G, T nucleotides in *coding* regions only. Use only chromosome 1.

Question 2 [points 10] Using the S. cerevisiae (Baker's yeast) data that we imported into R in Lectures 13 and 14, show R code of how you would estimate the frequency of A, C, G, T nucleotides in *non-coding* regions only. Use only chromosome 1. Comment on the differences bewteen the two matrices? Do you believe any observed differences are significant? Comment on how you might test significance.

Question 3 [points 20] Using the HMM package in R , implement your model. The documentation for this package is here. Note that you might want to look at the dishonestCasino() function that I wrote to help you with the concepts here. Perhaps follow the viterbi function and the example there. Show your code. Apply it back to chromosome 1. Apply it chromosome 2 too.

Questiom 4 [points 10] Compute the specificity, sensitivity and accuracy on both chromosomes individually. Comment on your findings.

Good luck!

Points of Reflection

Make sure that you understand the concept of searching for the most probable walk in the HMM and why using that walk is a reasonable way to “guess” the correct answer. This is a good example of mathematical optimization.

Suppose I was really interested in some kind of strange Archaea that lives on the bottom of the ocean on the side of a volcano. In fact let’s suppose that it’s a completely newly discovered species. Explain some of the problems that might arise using a gene finding HMM for a species that’s very different from anything we’ve seen before.

Instead of gene finding, suppose you wanted to predict the secondary structural elements of a nascent amino acid chain. That is, do you want to be able to sub strains of the sequence the correspond to turns, helixes, and beta sheets. Describe how you would do that with an HMM. Specifically describe the structure of the HMM but also how you would learn the probabilities to parameterize the HMM.