Lecture 14 -Hidden Markov Models [HMMs]

[Assignment 4 due today]

Markov Chains

- $X(t)$ is the state of the system at time t [t is discrete]
- The **state-space** S is the set of values that $X(t)$ can take.
- The **transition function** $P(x_1,x_2)$ gives the probability that the next value of X is x_2, given that X is currently in state x_1.

- Note, by assumption: $P(X(t)|X(t-1),X(t-2),\ldots,X(0)) = P(X(t)|X(t-1))$.
  - So "the future is independent of the past, except through the present state".
- Example:
  - $Y(t) = -1$ with prob 1/2; +1 otherwise, for all t.
  - Define $X(0) = 0$; $X(t)=X(t-1)+Y(t)$ $[=Y(t)+Y(t-1)+\ldots+Y(0)]$
  - So $P(X(t)=x_2|X(t-1)=x_1)=1/2$ if $|x_2-x_1| =1$ (for integers x_1,x_2)
  - This is a "symmetric random walk".

Elements of Discrete-time Hidden Markov Models

1. A finite number, N, of (hidden) states in the model.

2. At each clock time, t, a new (hidden) state is entered. (*Note that the new state may be the same as the last state*). This is controlled by the transition probability matrix. [It is a Markov Chain.]

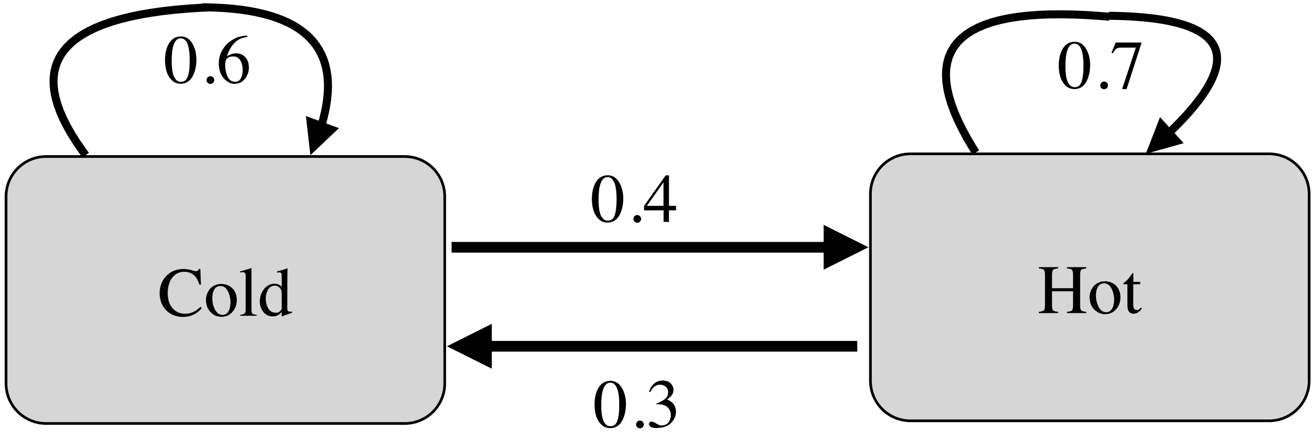
3. After each (potential) transition some observation is made. The observation is made according to a probability distribution that depends on the current state of the system. [Often called the “emission probability" or “confusion” matrix.] Note that the set of observations is typically **not** a Markov chain.

HMM example

- Motivating example: (Stamp, Mark. "A revealing introduction to hidden Markov models." Department of Computer Science San Jose State University (2004).)

- Wish to infer weather conditions over a series of years. We aim to categorize years as ‘hot’ or ‘cold’. Suppose that data records suggest the following:
  - the probability of a hot year being followed by another hot year is 0.7
  - the probability that a cold year is followed by another cold year is 0.6.

HMM example - hidden states



Described by a Markov chain:
X(t) is the type of weather in year t.

Transition matrix:

$$\begin{array}{c|cc} & H & C \\ \hline H & 0.7 & 0.3 \\ C & 0.4 & 0.6 \end{array}$$

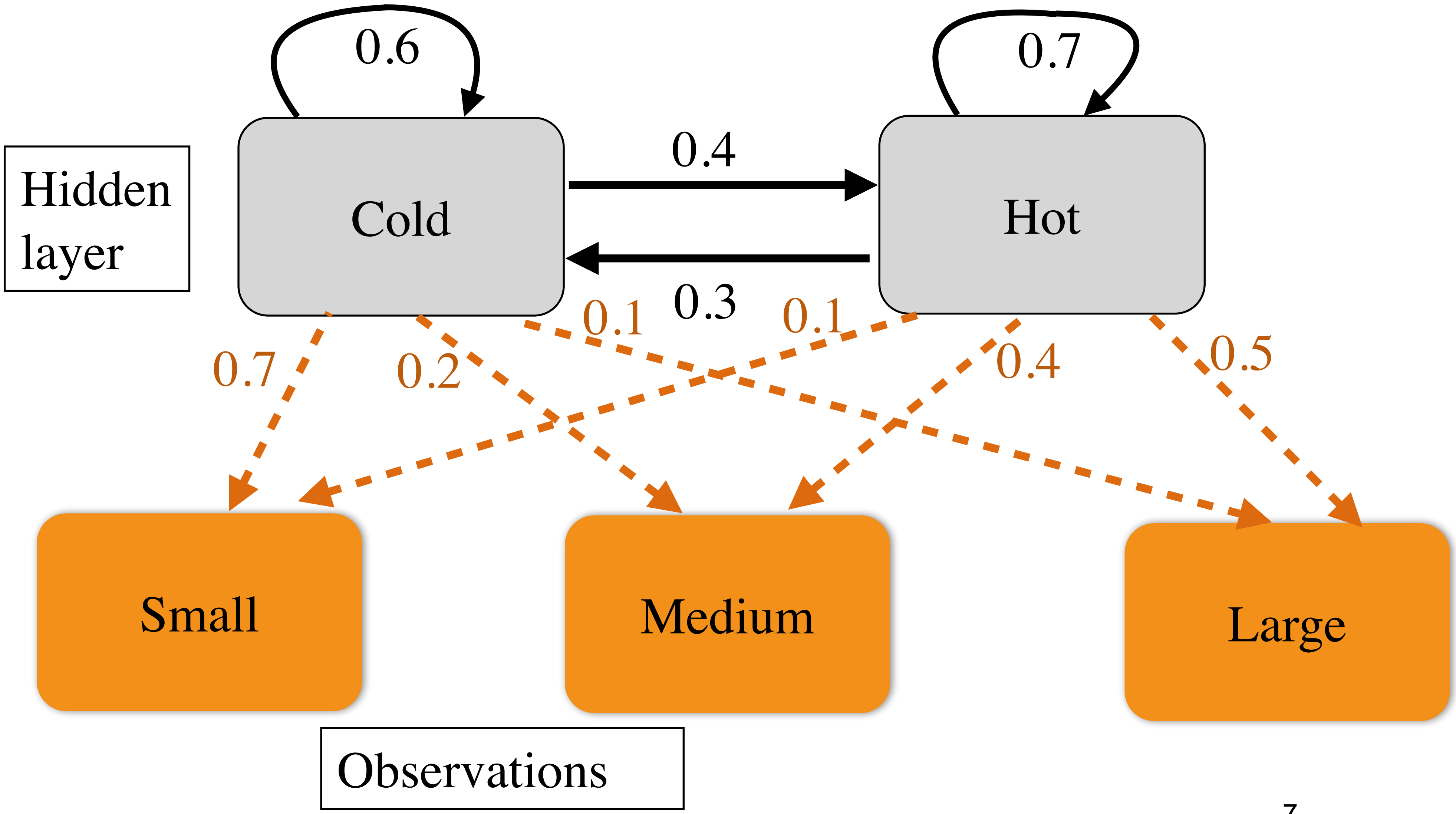
HMM example - observations

- But we can’t go back and observe the weather conditions. Instead we look at the width of tree rings:
  - Suppose there is a correlation between the size of tree growth rings and temperature.
  - We summarize this using three different tree ring sizes, small [S], medium [M] and large [L].
  - Suppose the probabilistic relationship between annual temperature and tree ring sizes is given by

$$\begin{array}{cc} & \begin{array}{ccc} S & M & L \end{array} \\ \begin{array}{c} H \\ C \end{array} & \left[\begin{array}{ccc} 0.1 & 0.4 & 0.5 \\ 0.7 & 0.2 & 0.1 \end{array}\right] \end{array}$$

Emission probabilities

HMM example - graph



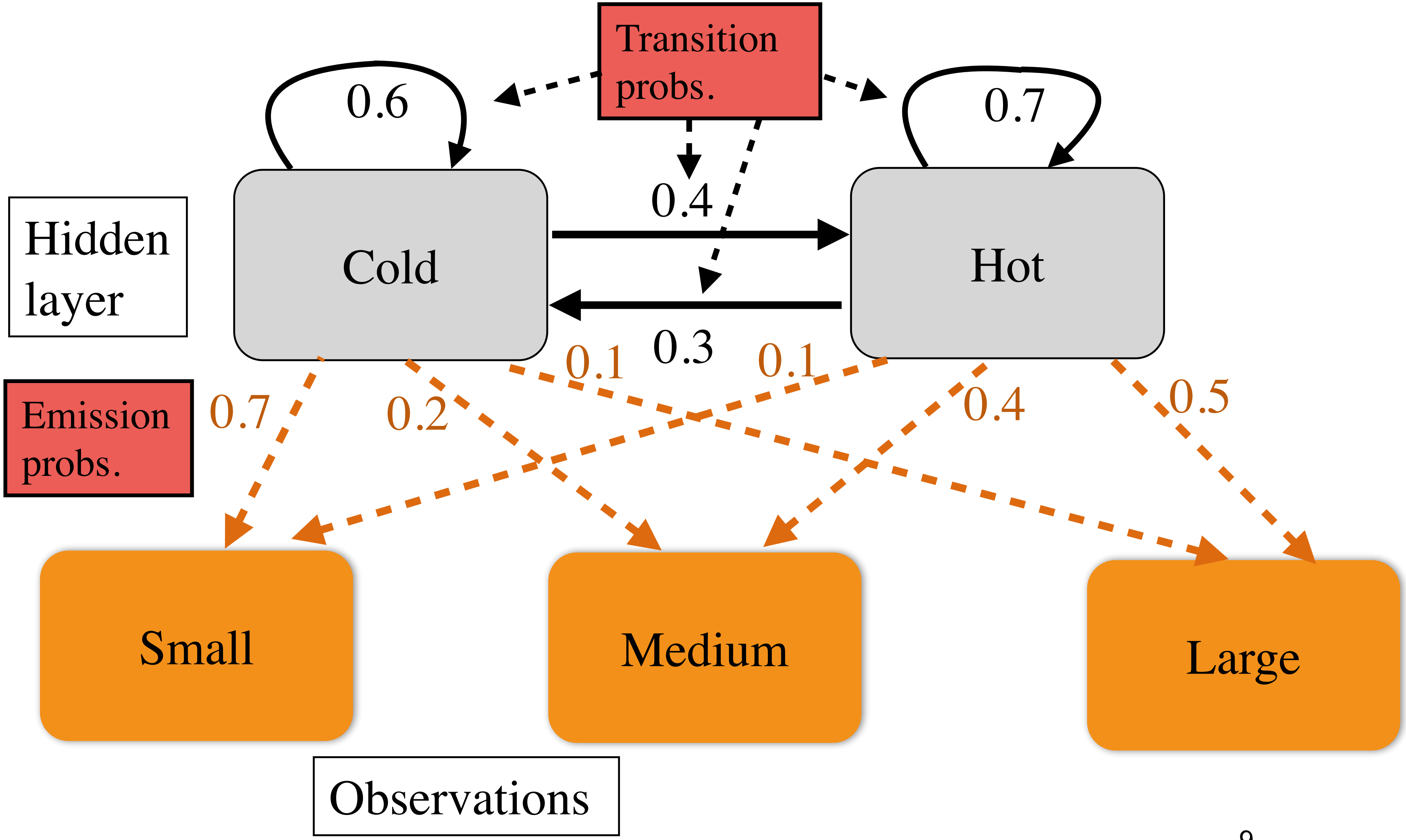
HMMs

- **Hidden states**: the underlying state of the process
- **Observations**: the thing that is actually measured

- **Transition matrix**: Describes the way the hidden states change.
- **Emission matrix** (or ‘observed probability matrix’, or ‘confusion matrix’): describes the conditional probability of seeing each observation as a function of the true underlying state of the process.

- Note: The sequence of hidden states is Markovian. The sequence of observed states **is typically not**.

HMM example - graph



HMMs

- Our goal:
  - Given a sequence of observed states that depend upon an underlying hidden process, we want to say something about that underlying process.
    - e.g., which is the most likely sequence of underlying states?
  - Note, we may or may not know both the transition matrix and emission probabilities, so sometimes we also need to infer one of those [much more difficult to do].

HMMs - example from decision theory

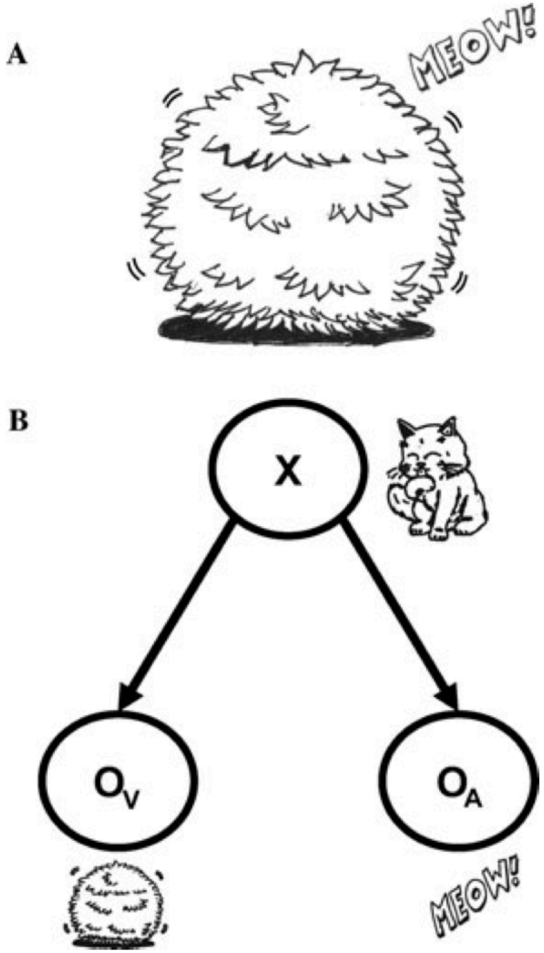
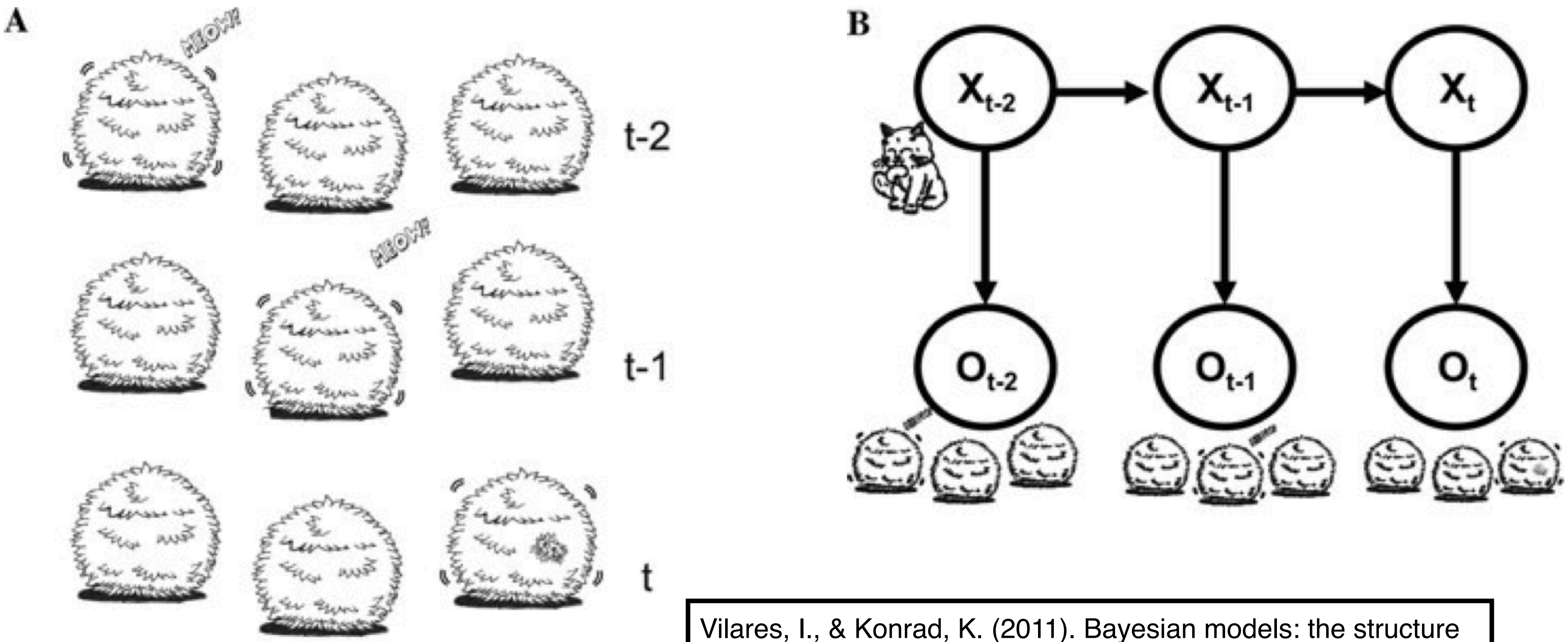


Figure 1. Cue combination. (A) Example of an indirect observation of a cat's position. You can see the bush moving and hear a "meow" sound, but you cannot directly observe the cat. (B) Graphical model of what is seen in A. The variable X (position of the cat) is unobserved, but it produces two observed variables: the moving bush, which provides a visual cue (O_v), and the "meow" sound, which provides an auditory cue (O_A). Cartoons made by Hugo M. Martins.

Vilares, I., & Konrad, K. (2011). Bayesian models: the structure of the world, uncertainty, behavior, and the brain. *Annals of the New York Academy of Sciences*, *1224*(1), 22–39.

HMMs - example from decision theory



Vilares, I., & Konrad, K. (2011). Bayesian models: the structure of the world, uncertainty, behavior, and the brain. *Annals of the New York Academy of Sciences*, *1224*(1), 22–39.

Figure 3. Combining information across time. (A) Example of an indirect observation of a cat's position, at different points of time (t−2, t−1, and t, t being the present time). (B) Graphical model of what is seen in A. The hidden variable X (position of the cat) at each point of time produces a variable O that is observed.

HMMs - Notation

T = length of the observation sequence

N = number of states in the model

M = number of observation symbols

X_t = the state of the hidden process at time t [A Markov chain]

O_t = the observation made at time t [Not a Markov chain]

Q = $\{q_1, q_2, \ldots, q_N\}$, the distinct states of the Markov process [The state-space of X]

V = $\{v_1, v_2, \ldots, v_M\}$, the set of possible observations [The state-space of O]

A = $\{a_{ij}\}$, the (hidden) state transition probabilities [$a_{ij}=P(X_{t+1}=q_j \mid X_t=q_i)$]

B = $\{b_{ij}\}$, the observation probability matrix [$b_{ij}=P(O_t=v_j \mid X_t=q_i)$]

π = initial (hidden) state distribution.

Notation taken from - “A Revealing Introduction to Hidden Markov Models” - M. Stamp

Example - Fair/unfair dice

- A casino has two dice:
  - Die 1: Each outcome (1,2,3,4,5,6) has prob. 1/6.
  - Die 2: Rolls a 6 with prob. 1/2; rolls a 1,2,3,4 or 5 with prob. 1/10.
- If Die 1 was used last time, then the casino uses Die 2 next time with prob. 0.01; otherwise it uses Die 1 again.
- If Die 2 was used last time, then the casino uses Die 1 next time with prob. 0.02; otherwise it uses Die 2 again.
- You cannot tell which of the two dice is being used for any given roll.
- You observe the sequence of die-rolls.
- Goal: To infer which die is being used when.

Example - Fair/unfair dice

- A casino has two dice:
- If Die 1 was used last time, then the casino uses Die 2 next time with prob. 0.01; otherwise it uses Die 1 again.
- If Die 2 was used last time, then the casino uses Die 1 next time with prob. 0.02; otherwise it uses Die 2 again.

- Index the rolls with t=1,2,3,…
- Define X_t to be the die used at time t.
- X_t=1 if Die 1 was used; X_t=2 if Die 2 was used.
- Transition matrix, A:

Hidden layer

$$\begin{bmatrix} 0.99 & 0.01 \\ 0.02 & 0.98 \end{bmatrix}$$

[$a_{ij}=P(X_{t+1}=q_j \mid X_t=q_i)$]

Example - Fair/unfair dice

- A casino has two dice:
  - Die 1: Each outcome (1,2,3,4,5,6) has prob. 1/6.
  - Die 2: Rolls a 6 with prob. 1/2; rolls a 1,2,3,4 or 5 with prob. 1/10.

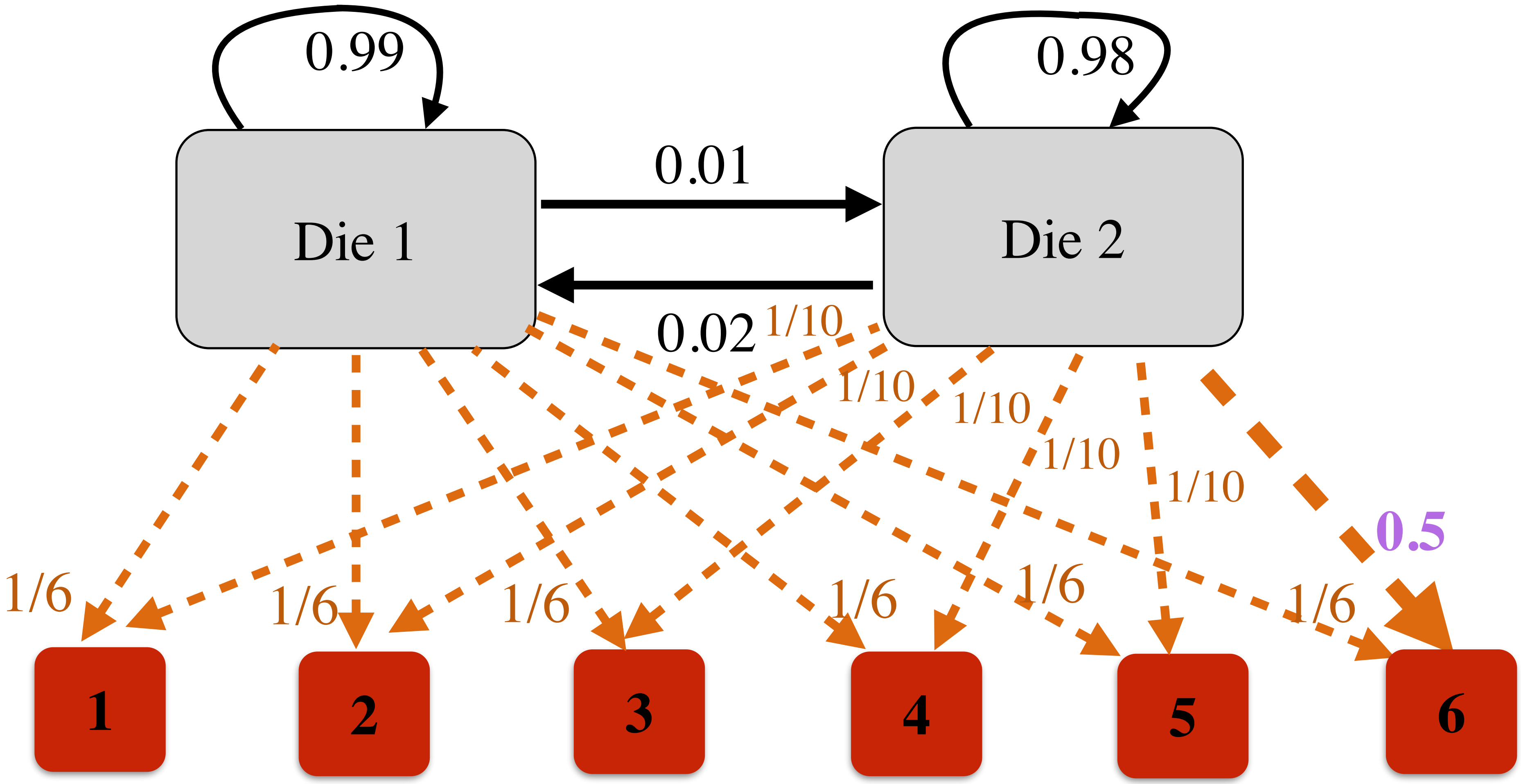
- Index the rolls with t=1,2,3,…
- Define O_t to be the die-roll observed at time t.
- Emission (confusion) matrix, B:

Observed layer

$$\begin{array}{cc} & \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{array}{c} 1 \\ 2 \end{array} & \left[\begin{array}{cccccc} 1/6 & 1/6 & 1/6 & 1/6 & 1/6 & 1/6 \\ 1/10 & 1/10 & 1/10 & 1/10 & 1/10 & 1/2 \end{array}\right] \end{array}$$

[$b_{ij}=P(O_t=v_j \mid X_t=q_i)$]

HMM example - graph



3 key HMM questions

1. Given the observation sequence **O** = (O_1, O_2, **…** , O_T), and a model λ=(A,B,π) how do we efficiently compute P(**O** I A,B,π), the probability of the observation sequence, given the model? (Note, this is not conditioned on *X*, the state of the hidden layer.) This can also be used to assess model fit. - Forward-backward algorithm

2. Given the observation sequence O = (O_1, O_2, **…** , O_T), and a model λ=(A,B,π) how do we choose a corresponding state sequence **X**=X_1,X_2,…,X_T that is optimal in some meaningful sense (e.g., maximum likelihood). (i.e., what sequence of hidden states best “explains” the observed data?) - Viterbi algorithm

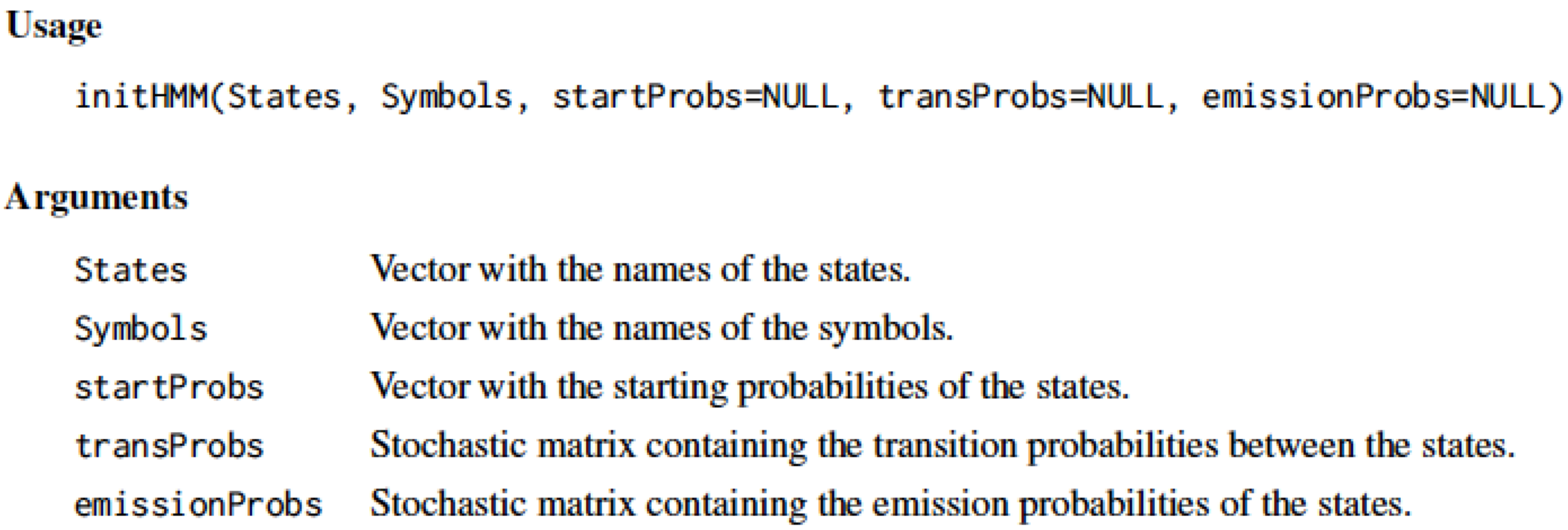
3. How do we adjust the model parameters (A,B,π) to maximize P(OIA,B,π)? Parameter estimation. - Baum-Welch algorithm

R package: HMM - Dice example

```
#install.packages(‘HMM')
library('HMM')
set.seed(985)
nSim = 2000
States = c("Fair", "Unfair")
Symbols = 1:6
# Define the transition matrix
transProbs = matrix(c(0.99, 0.01, 0.02, 0.98), c(length(States),length(States)), byrow = TRUE)
# Define the emission probabilities
emissionProbs = matrix(c(rep(1/6, 6), c(rep(0.1, 5), 0.5)),c(length(States), length(Symbols)), byrow = TRUE)
#Set up the HMM
hmm = initHMM(States, Symbols, transProbs = transProbs, emissionProbs = emissionProbs)
# Simulate some test data to play with
sim = simHMM(hmm, nSim)
# the resulting die-rolls are stored in sim$observation
plot(sim$observation,pch='.')
# the sequence recording which die was rolled is stored in sim$states. Let’s look at the first few values
sim$states[1:20]
```

[‘HMMs.R’ on Github in Week13_HiddenMarkovModels]

Initialization



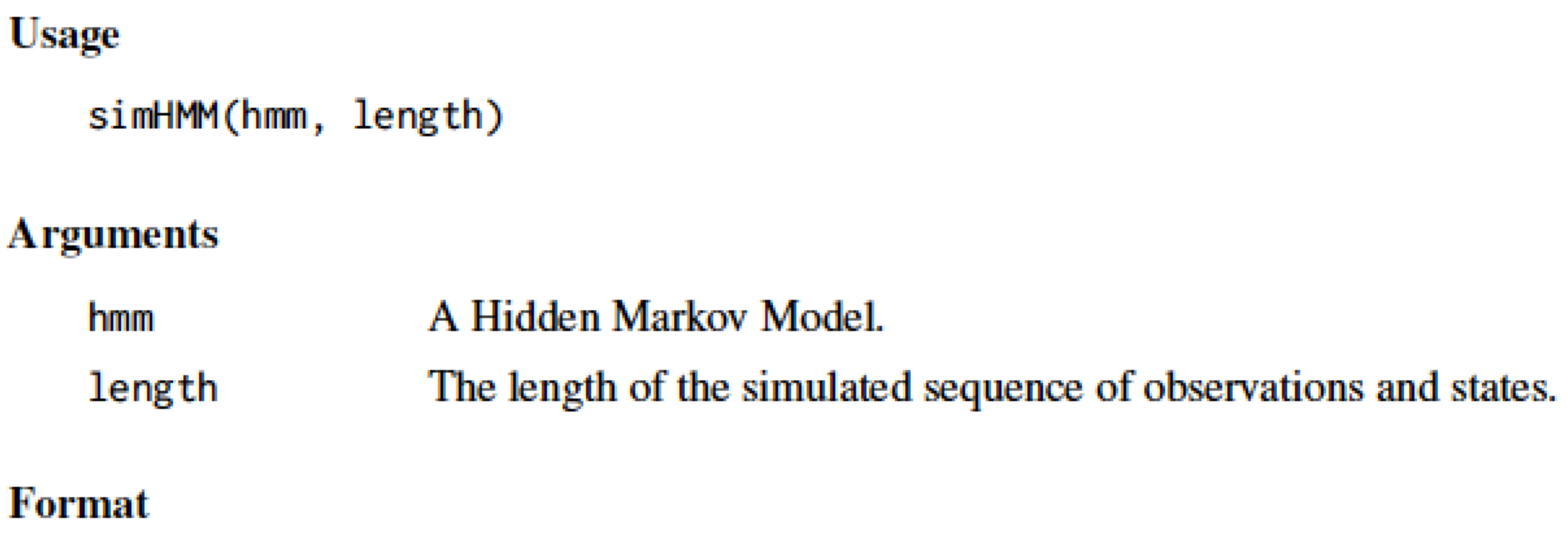
Usage

```
initHMM(States, Symbols, startProbs=NULL, transProbs=NULL, emissionProbs=NULL)
```

Arguments

`States`	Vector with the names of the states.
`Symbols`	Vector with the names of the symbols.
`startProbs`	Vector with the starting probabilities of the states.
`transProbs`	Stochastic matrix containing the transition probabilities between the states.
`emissionProbs`	Stochastic matrix containing the emission probabilities of the states.

Simulating test data



Usage

```
simHMM(hmm, length)
```

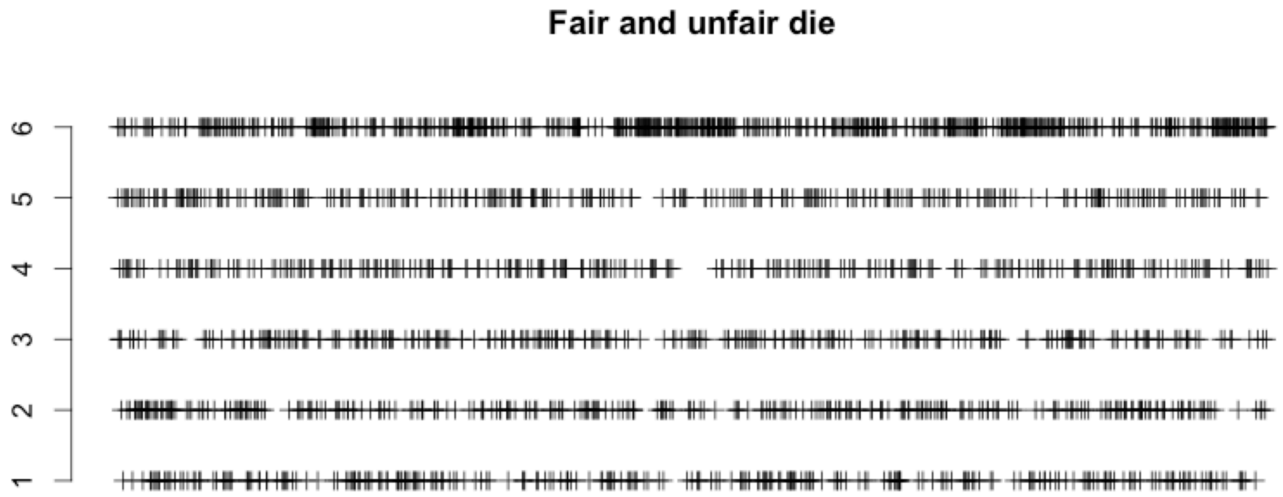
Arguments

`hmm` A Hidden Markov Model.

`length` The length of the simulated sequence of observations and states.

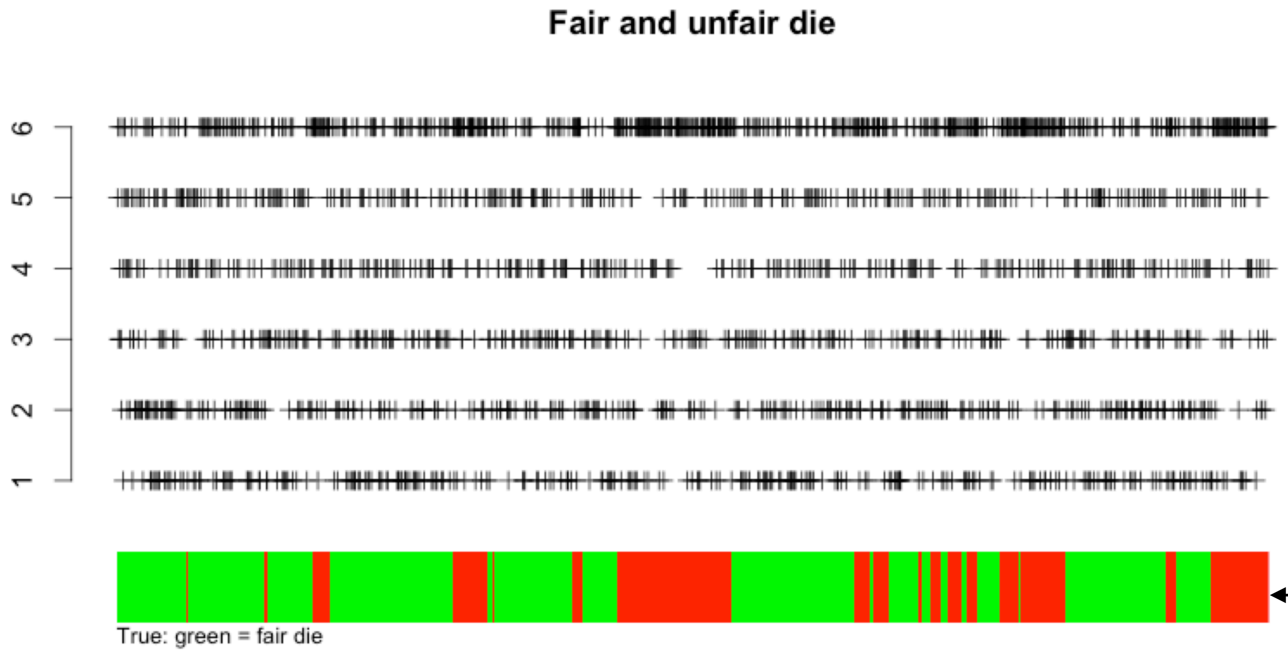
Format

Dice - Example output



Simulated die-rolls

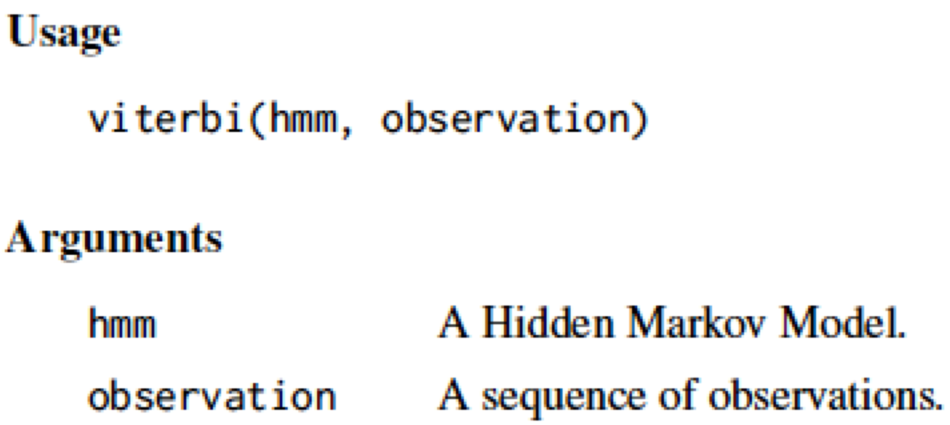
Dice - Example output



Simulated die-rolls

Hidden layer - true state

Viterbi algorithm to find most likely path



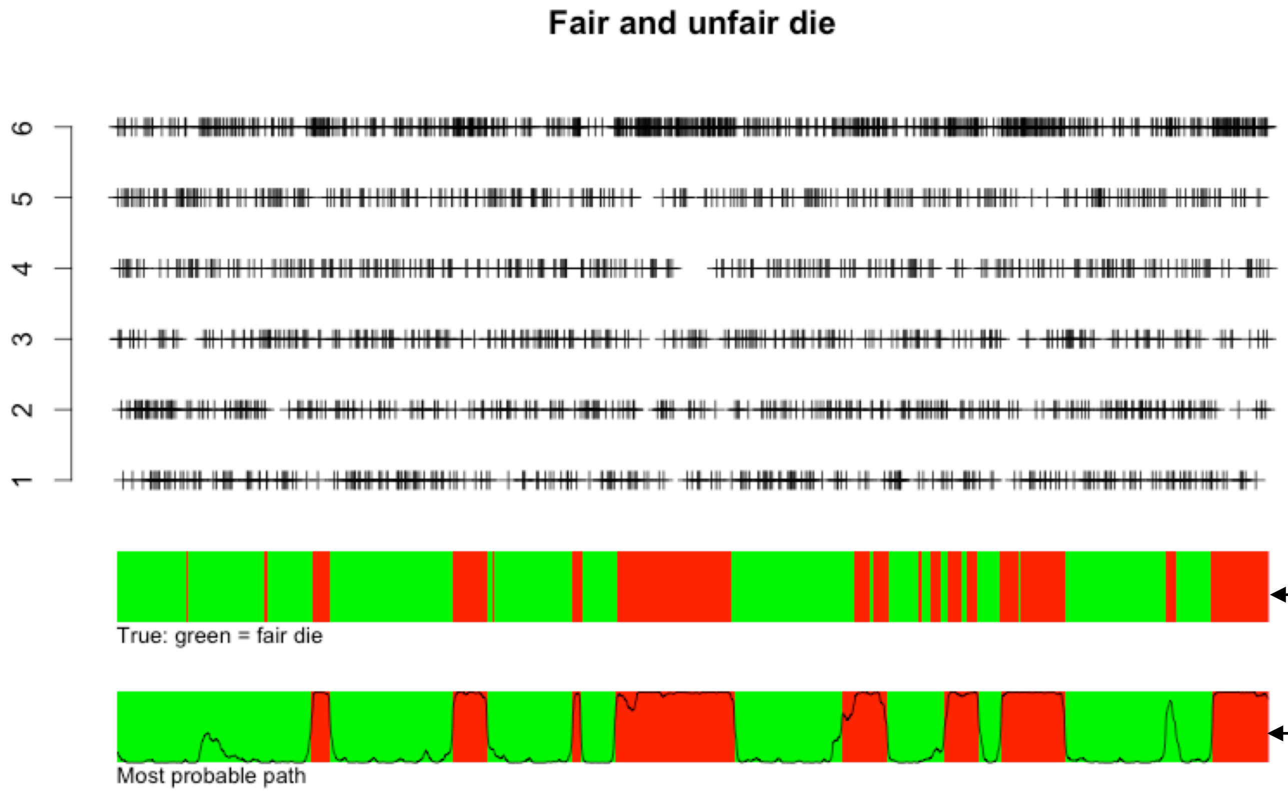
Usage

```
viterbi(hmm, observation)
```

Arguments

`hmm`	A Hidden Markov Model.
`observation`	A sequence of observations.

Dice - Example output

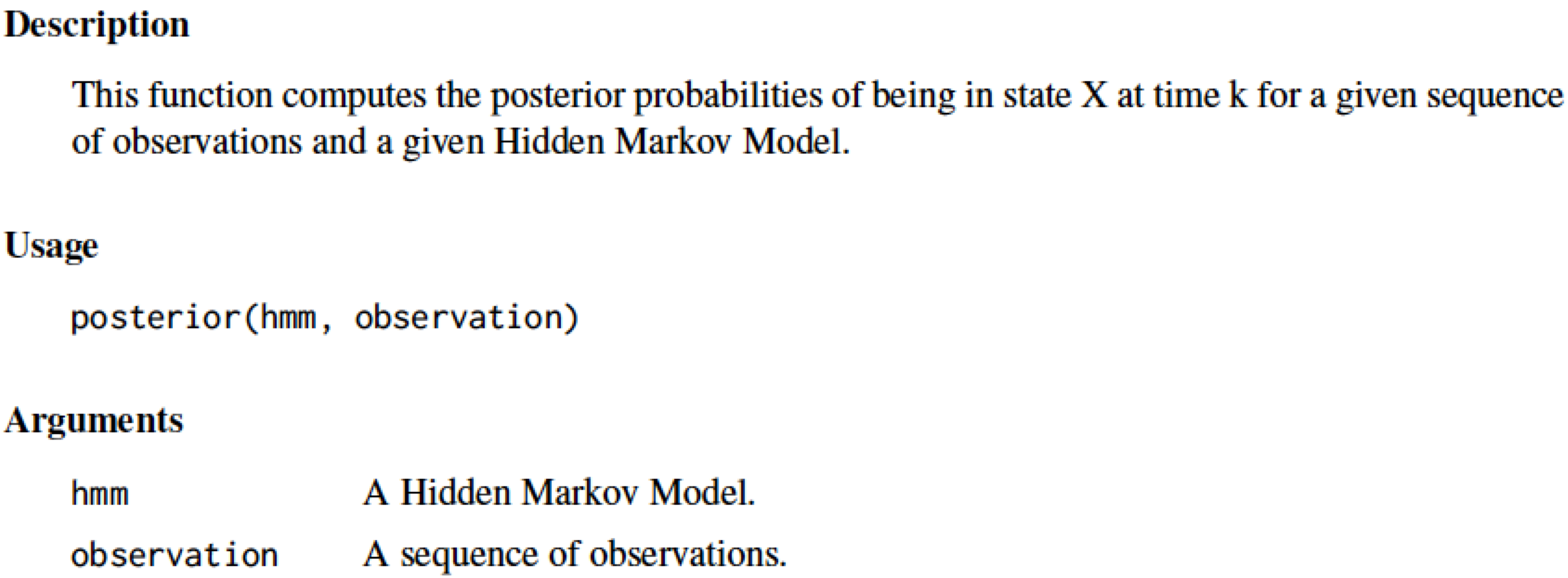


Simulated die-rolls

Hidden layer - true state

Hidden layer - inferred (most likely) state

Posterior state probs



Description

This function computes the posterior probabilities of being in state X at time k for a given sequence of observations and a given Hidden Markov Model.

Usage

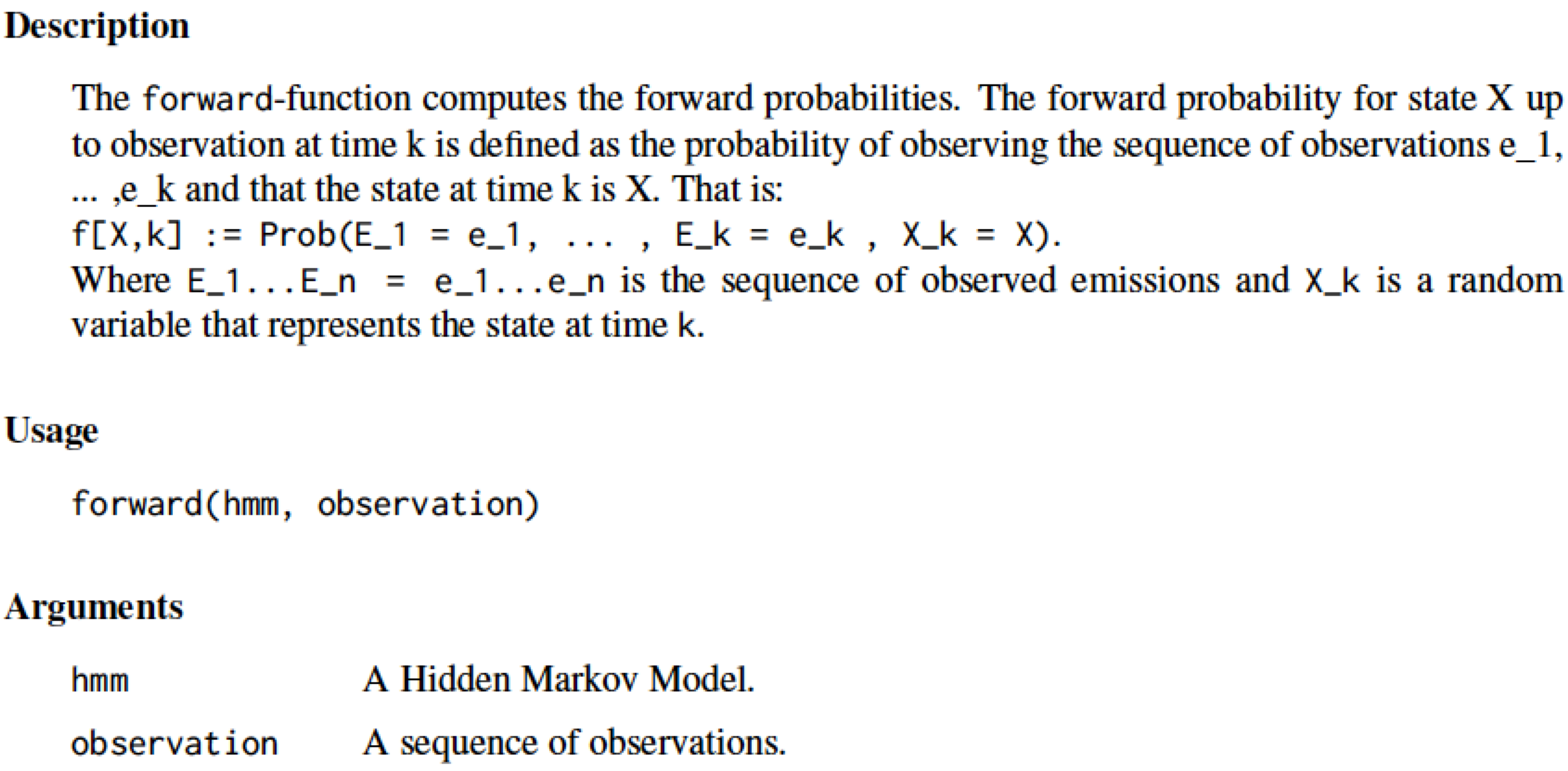
```
posterior(hmm, observation)
```

Arguments

`hmm`	A Hidden Markov Model.
`observation`	A sequence of observations.

This uses the output of the forwards (& backwards) algorithms for the probabilities of everything up to (after) a given time.
[It averages over all possible paths, rather than just taking the most likely path.]

Forward algorithm



Description

The `forward`-function computes the forward probabilities. The forward probability for state X up to observation at time k is defined as the probability of observing the sequence of observations e_1, ... ,e_k and that the state at time k is X. That is:
`f[X,k] := Prob(E_1 = e_1, ... , E_k = e_k , X_k = X).`
Where `E_1...E_n = e_1...e_n` is the sequence of observed emissions and `X_k` is a random variable that represents the state at time `k`.

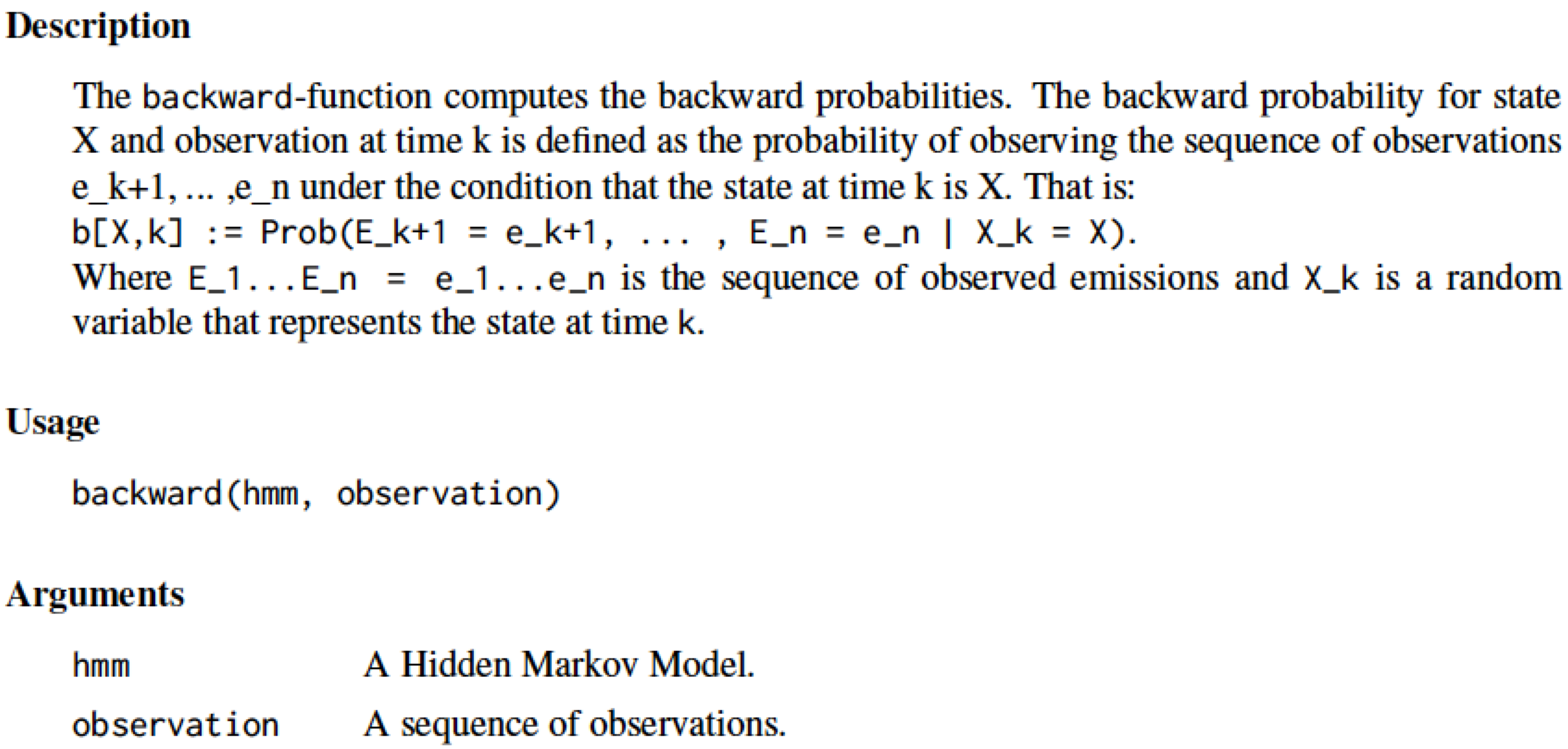
Usage

```
forward(hmm, observation)
```

Arguments

`hmm`	A Hidden Markov Model.
`observation`	A sequence of observations.

Backward algorithm



Description

The backward-function computes the backward probabilities. The backward probability for state X and observation at time k is defined as the probability of observing the sequence of observations e_k+1, ... ,e_n under the condition that the state at time k is X. That is:
b[X,k] := Prob(E_k+1 = e_k+1, ... , E_n = e_n | X_k = X).
Where E_1...E_n = e_1...e_n is the sequence of observed emissions and X_k is a random variable that represents the state at time k.

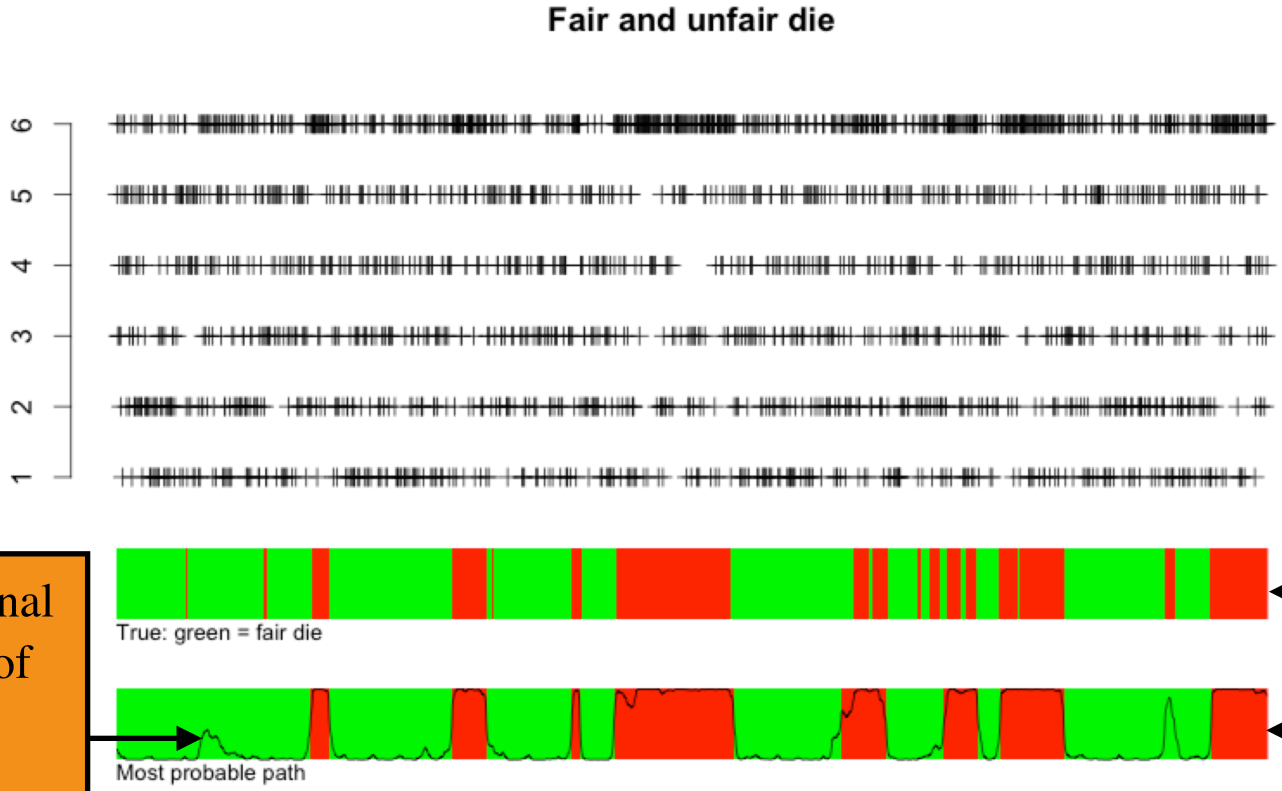
Usage

```
backward(hmm, observation)
```

Arguments

hmm	A Hidden Markov Model.
observation	A sequence of observations.

Dice - Example output



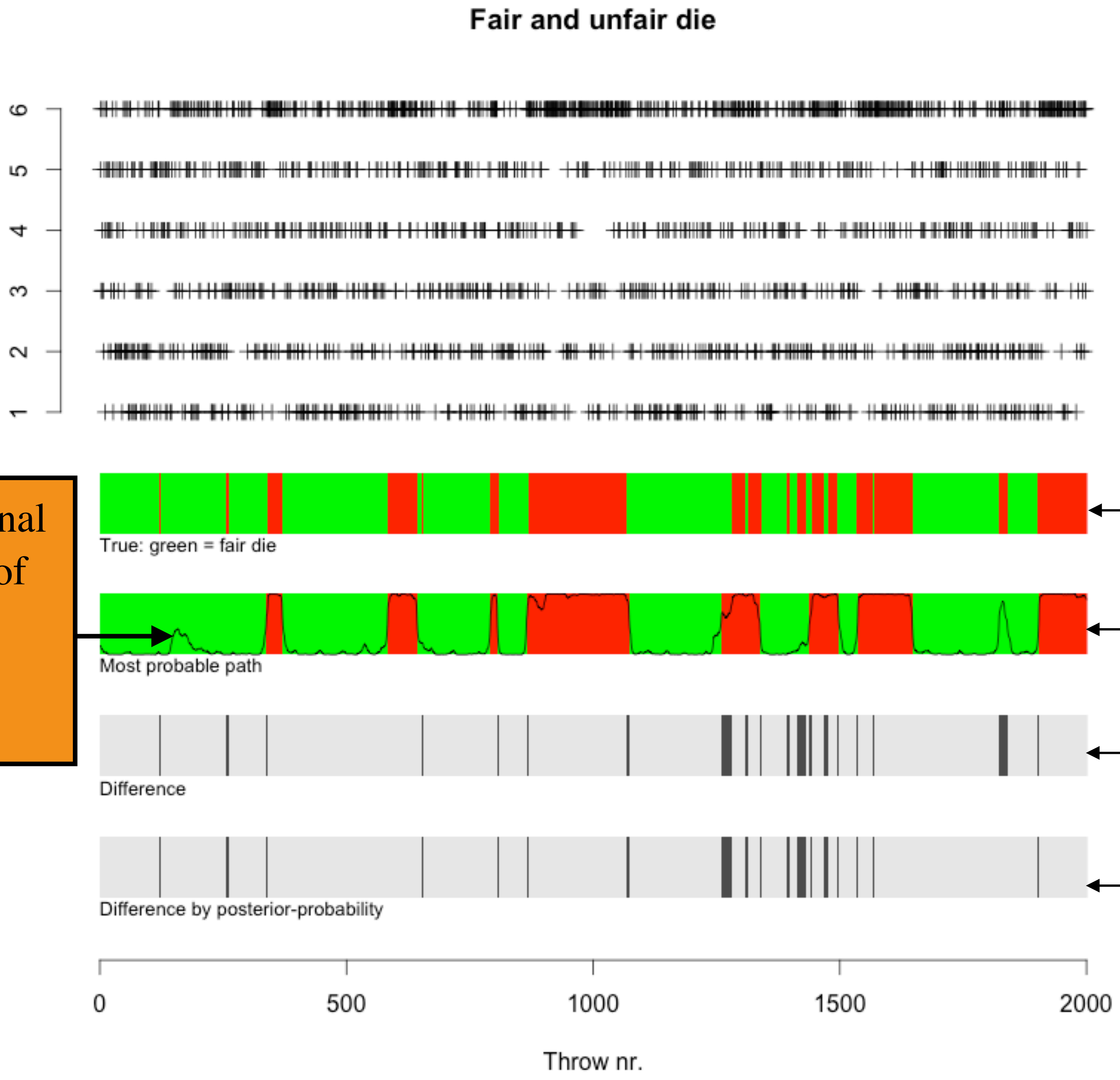
Simulated die-rolls

Hidden layer - true state

Hidden layer - inferred (most likely) state

Marginal prob. of using unfair Die

Dice - Example output



Simulated die-rolls

Marginal prob. of using unfair Die

Hidden layer - true state

Hidden layer - inferred (most likely) state

Difference between ‘true’ and ‘inferred’

Difference between ‘true’ and ‘inferred’ marginally

Example 2 - DNA sequences

- Goal - to detect CpG regions (‘methylation islands’) in DNA sequence data.
- After http://web.stanford.edu/class/stats366/exs/HMM1.html (where you can find more details)

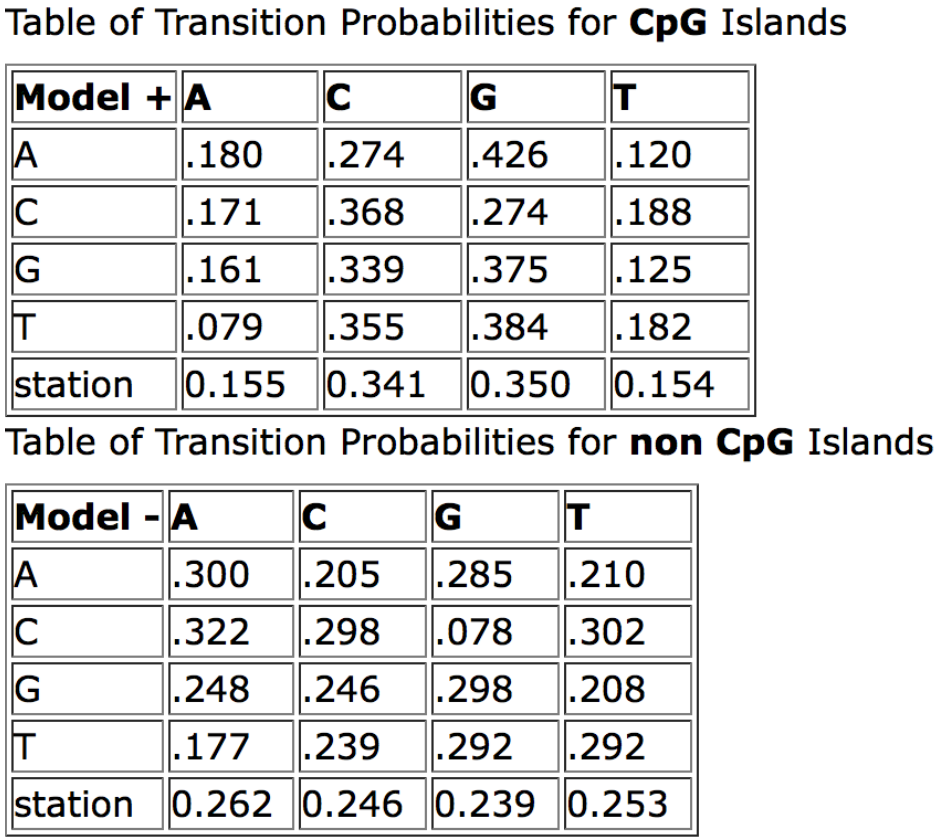


Table of Transition Probabilities for **CpG** Islands

Model +	**A**	**C**	**G**	**T**
A	.180	.274	.426	.120
C	.171	.368	.274	.188
G	.161	.339	.375	.125
T	.079	.355	.384	.182
station	0.155	0.341	0.350	0.154

Table of Transition Probabilities for **non CpG** Islands

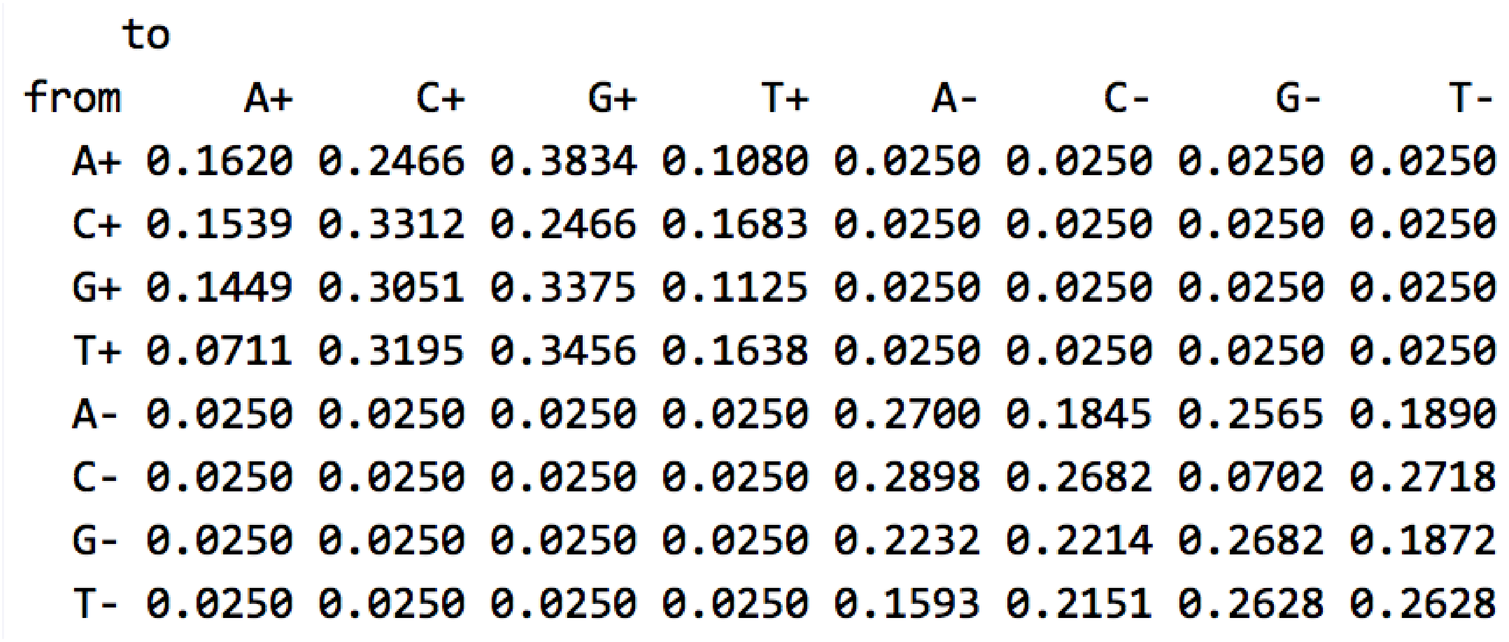
Model -	**A**	**C**	**G**	**T**
A	.300	.205	.285	.210
C	.322	.298	.078	.302
G	.248	.246	.298	.208
T	.177	.239	.292	.292
station	0.262	0.246	0.239	0.253

- How should we set this up as an HMM?
  - What is the hidden layer?
  - What are the states of the hidden layer?
  - What is the observed layer?
  - What are the states of the observed layer?

Suggestion

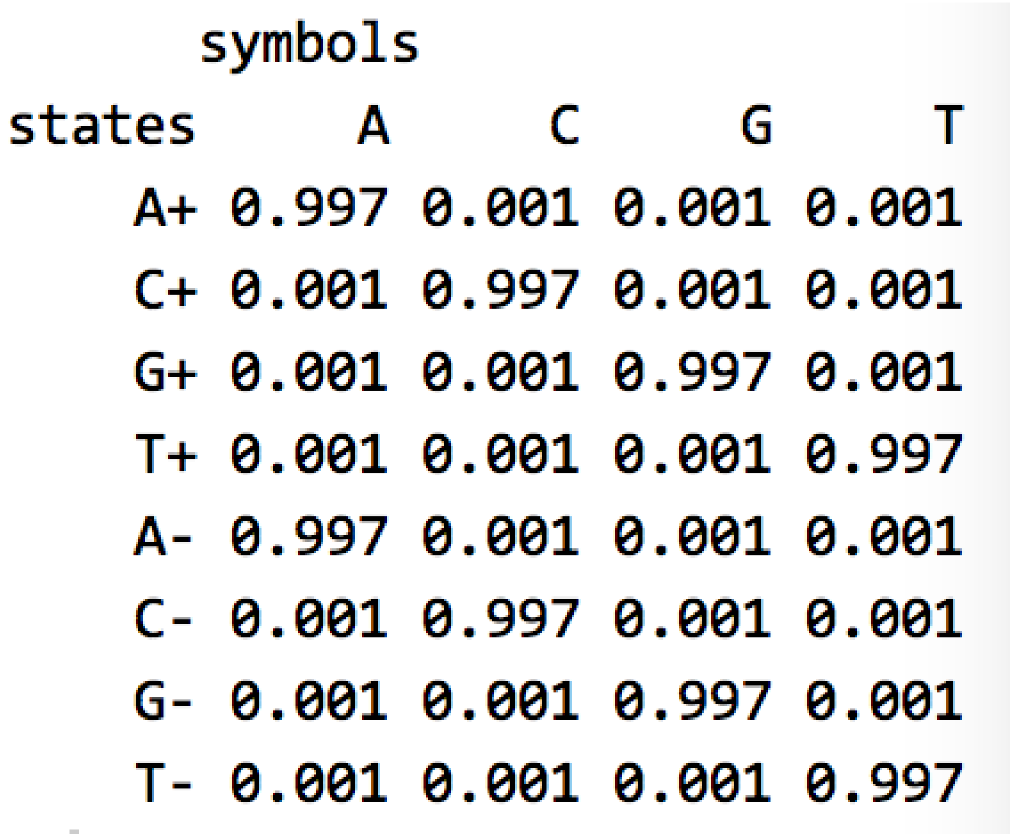
- Hidden layer contains both the nucleotide and a flag for whether or not we are in a CpG island:
  - A+,C+,G+,T+,A-,C-,G-,T-
- Observed layer contains just the nucleotide:
  - A,C,G,T
- Code: HMMs_CpGs.R on Github.

Transition mx



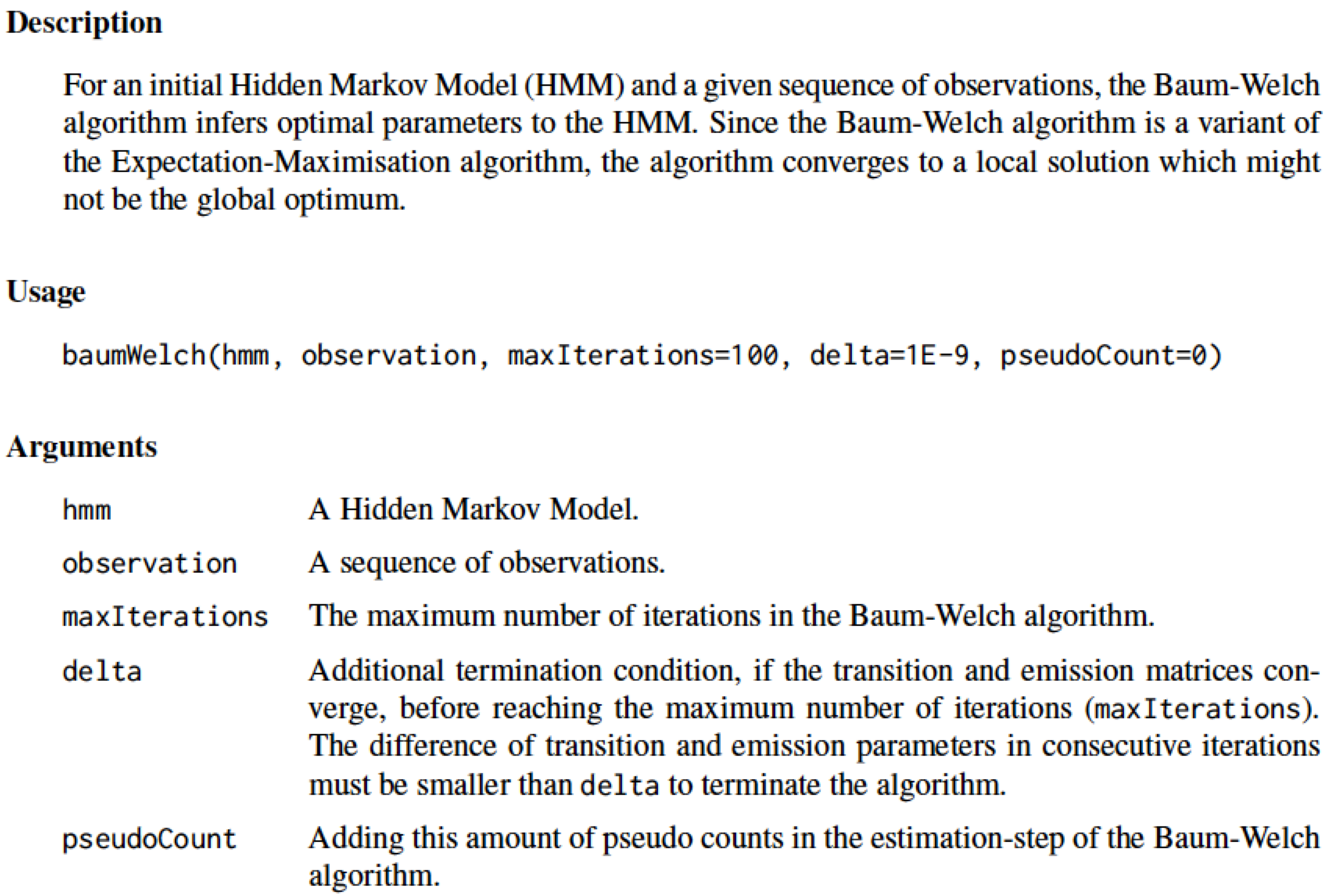
```
    to
from     A+     C+     G+     T+     A-     C-     G-     T-
  A+ 0.1620 0.2466 0.3834 0.1080 0.0250 0.0250 0.0250 0.0250
  C+ 0.1539 0.3312 0.2466 0.1683 0.0250 0.0250 0.0250 0.0250
  G+ 0.1449 0.3051 0.3375 0.1125 0.0250 0.0250 0.0250 0.0250
  T+ 0.0711 0.3195 0.3456 0.1638 0.0250 0.0250 0.0250 0.0250
  A- 0.0250 0.0250 0.0250 0.0250 0.2700 0.1845 0.2565 0.1890
  C- 0.0250 0.0250 0.0250 0.0250 0.2898 0.2682 0.0702 0.2718
  G- 0.0250 0.0250 0.0250 0.0250 0.2232 0.2214 0.2682 0.1872
  T- 0.0250 0.0250 0.0250 0.0250 0.1593 0.2151 0.2628 0.2628
```

Emission matrix



```
      symbols
states     A     C     G     T
    A+ 0.997 0.001 0.001 0.001
    C+ 0.001 0.997 0.001 0.001
    G+ 0.001 0.001 0.997 0.001
    T+ 0.001 0.001 0.001 0.997
    A- 0.997 0.001 0.001 0.001
    C- 0.001 0.997 0.001 0.001
    G- 0.001 0.001 0.997 0.001
    T- 0.001 0.001 0.001 0.997
```

Inferring of parameters - Baum-Welch algorithm



Description

For an initial Hidden Markov Model (HMM) and a given sequence of observations, the Baum-Welch algorithm infers optimal parameters to the HMM. Since the Baum-Welch algorithm is a variant of the Expectation-Maximisation algorithm, the algorithm converges to a local solution which might not be the global optimum.

Usage

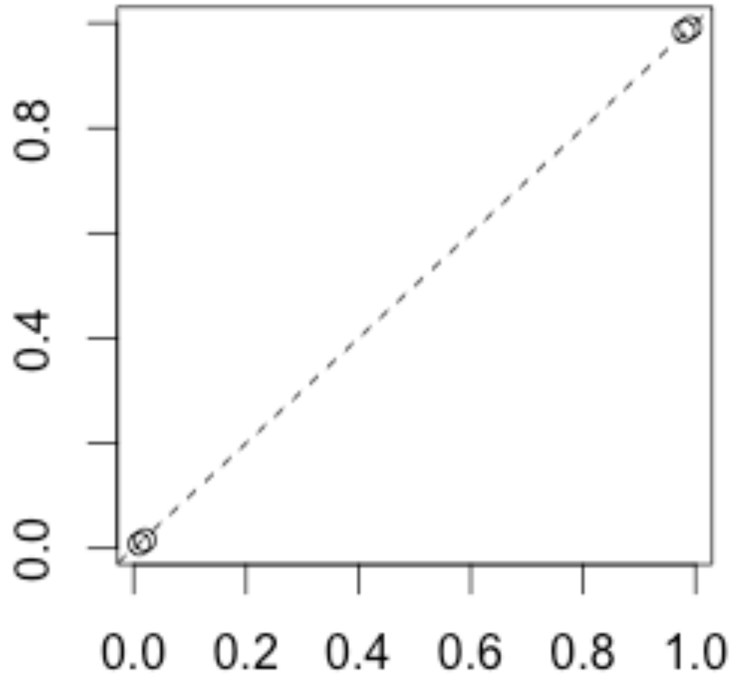
```
baumWelch(hmm, observation, maxIterations=100, delta=1E-9, pseudoCount=0)
```

Arguments

`hmm`	A Hidden Markov Model.
`observation`	A sequence of observations.
`maxIterations`	The maximum number of iterations in the Baum-Welch algorithm.
`delta`	Additional termination condition, if the transition and emission matrices converge, before reaching the maximum number of iterations (`maxIterations`). The difference of transition and emission parameters in consecutive iterations must be smaller than `delta` to terminate the algorithm.
`pseudoCount`	Adding this amount of pseudo counts in the estimation-step of the Baum-Welch algorithm.

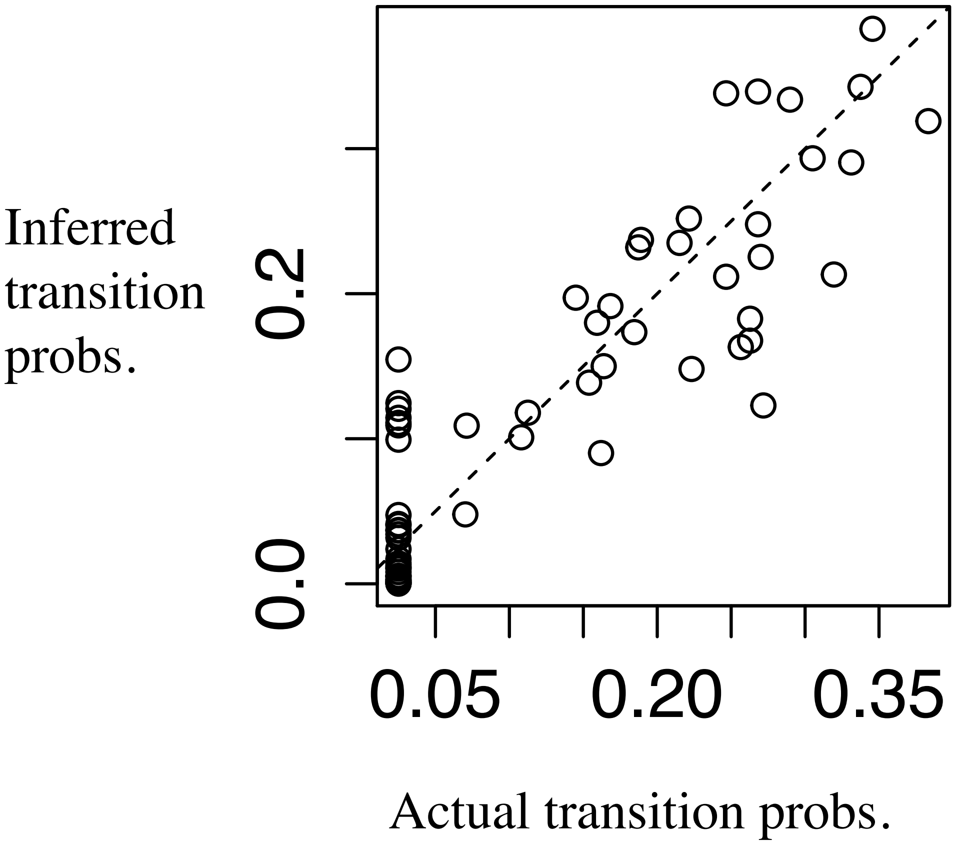
Application to Dice

Inferred transition probs.



Actual transition probs.

Application to CpG problem



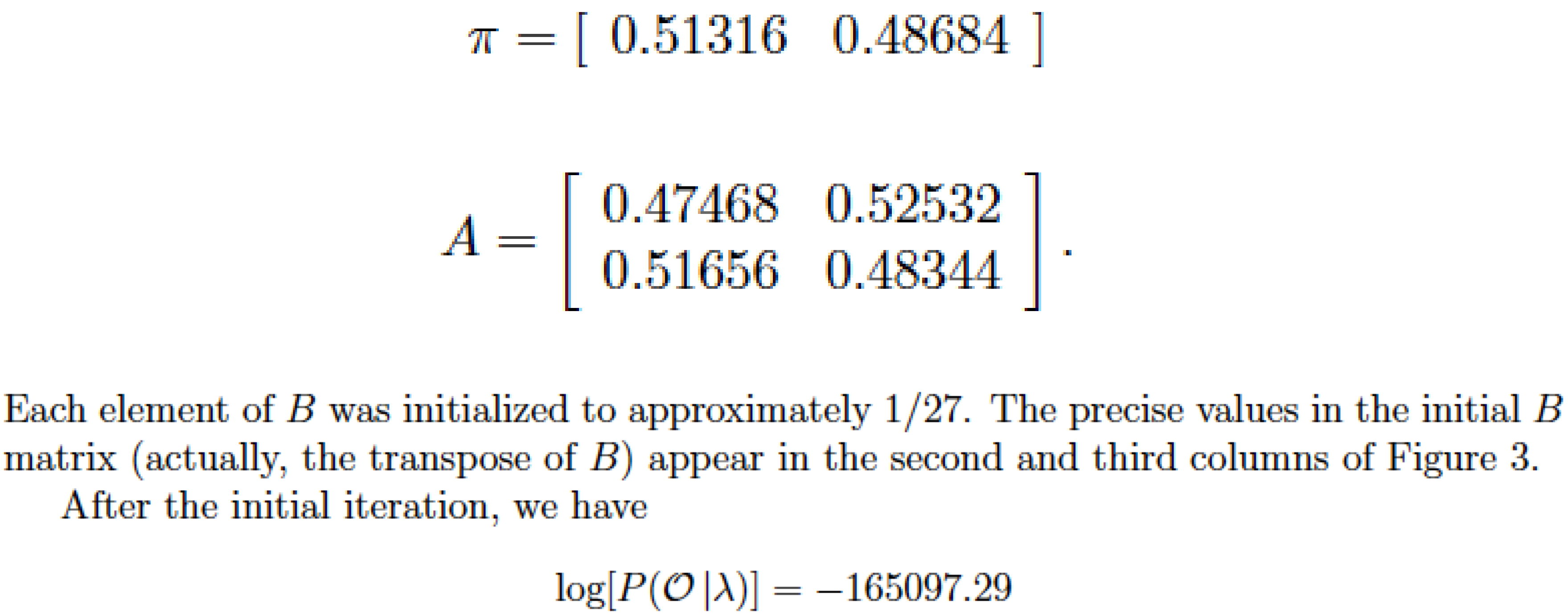
HMM application

- A Martian lands on earth and starts reading books.
- What can they understand about the English language using an HMM?

- We use a two-state model for the underlying hidden state, X.
- Assume that letter frequencies vary according to underlying state X.

R. L. Cave and L. P. Neuwirth, Hidden Markov models for English, in J. D. Ferguson, editor, Hidden Markov Models for Speech, IDA-CRD, Princeton, NJ, October 1980.

- Initial conditions



$$\pi = \begin{bmatrix} 0.51316 & 0.48684 \end{bmatrix}$$

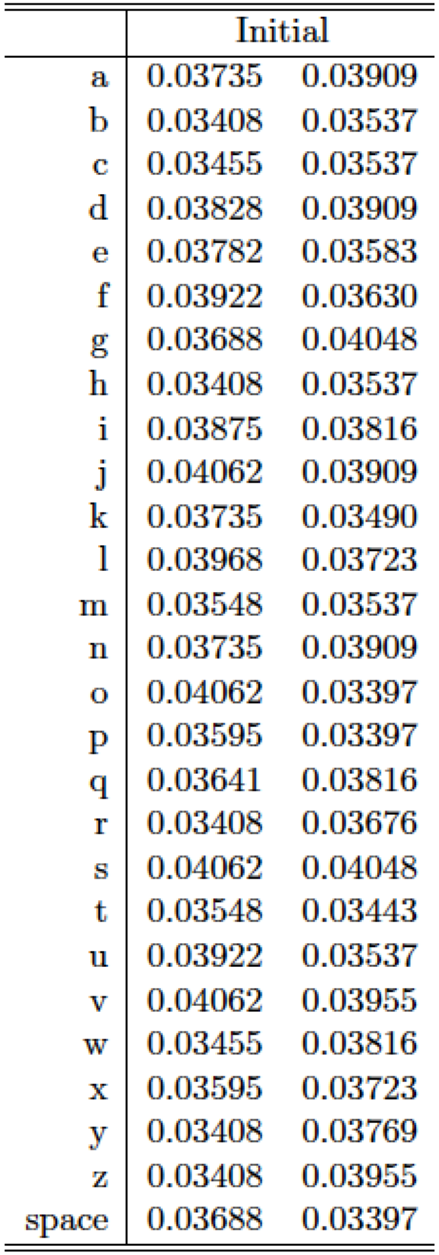
$$A = \begin{bmatrix} 0.47468 & 0.52532 \\ 0.51656 & 0.48344 \end{bmatrix}.$$

Each element of B was initialized to approximately $1/27$. The precise values in the initial B matrix (actually, the transpose of B) appear in the second and third columns of Figure 3.

After the initial iteration, we have

$$\log[P(\mathcal{O}\,|\,\lambda)] = -165097.29$$

“A Revealing Introduction to Hidden Markov Models” - M. Stamp, 2012

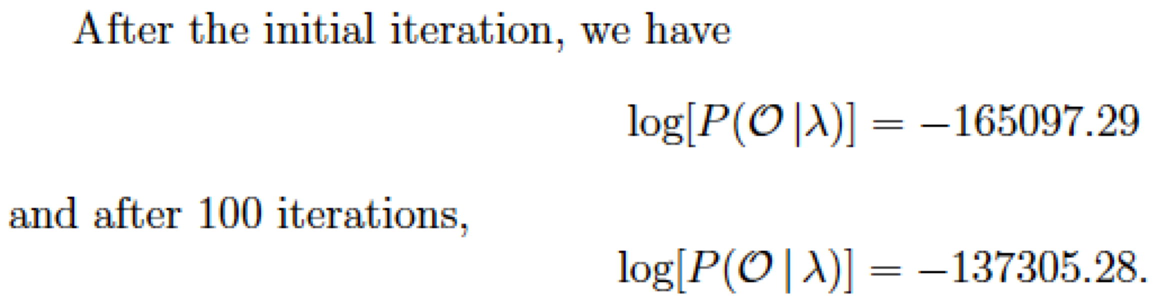


	Initial	
a	0.03735	0.03909
b	0.03408	0.03537
c	0.03455	0.03537
d	0.03828	0.03909
e	0.03782	0.03583
f	0.03922	0.03630
g	0.03688	0.04048
h	0.03408	0.03537
i	0.03875	0.03816
j	0.04062	0.03909
k	0.03735	0.03490
l	0.03968	0.03723
m	0.03548	0.03537
n	0.03735	0.03909
o	0.04062	0.03397
p	0.03595	0.03397
q	0.03641	0.03816
r	0.03408	0.03676
s	0.04062	0.04048
t	0.03548	0.03443
u	0.03922	0.03537
v	0.04062	0.03955
w	0.03455	0.03816
x	0.03595	0.03723
y	0.03408	0.03769
z	0.03408	0.03955
space	0.03688	0.03397

Initial emission probabilities

“A Revealing Introduction to Hidden Markov Models” - M. Stamp, 2012

Now run the Baum-Welch algorithm:

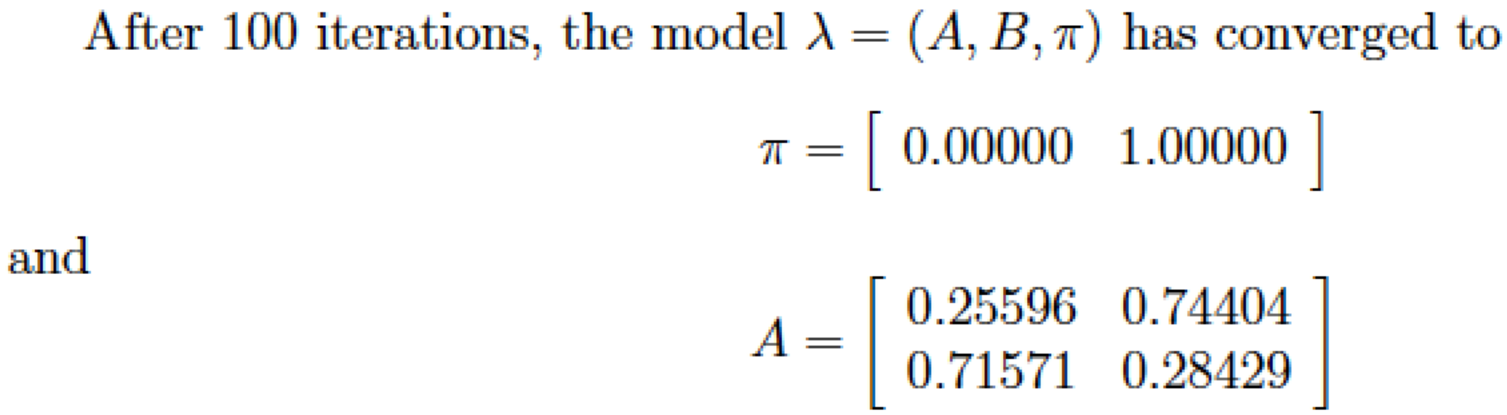


After the initial iteration, we have

$$\log[P(\mathcal{O}\,|\,\lambda)] = -165097.29$$

and after 100 iterations,

$$\log[P(\mathcal{O}\,|\,\lambda)] = -137305.28.$$



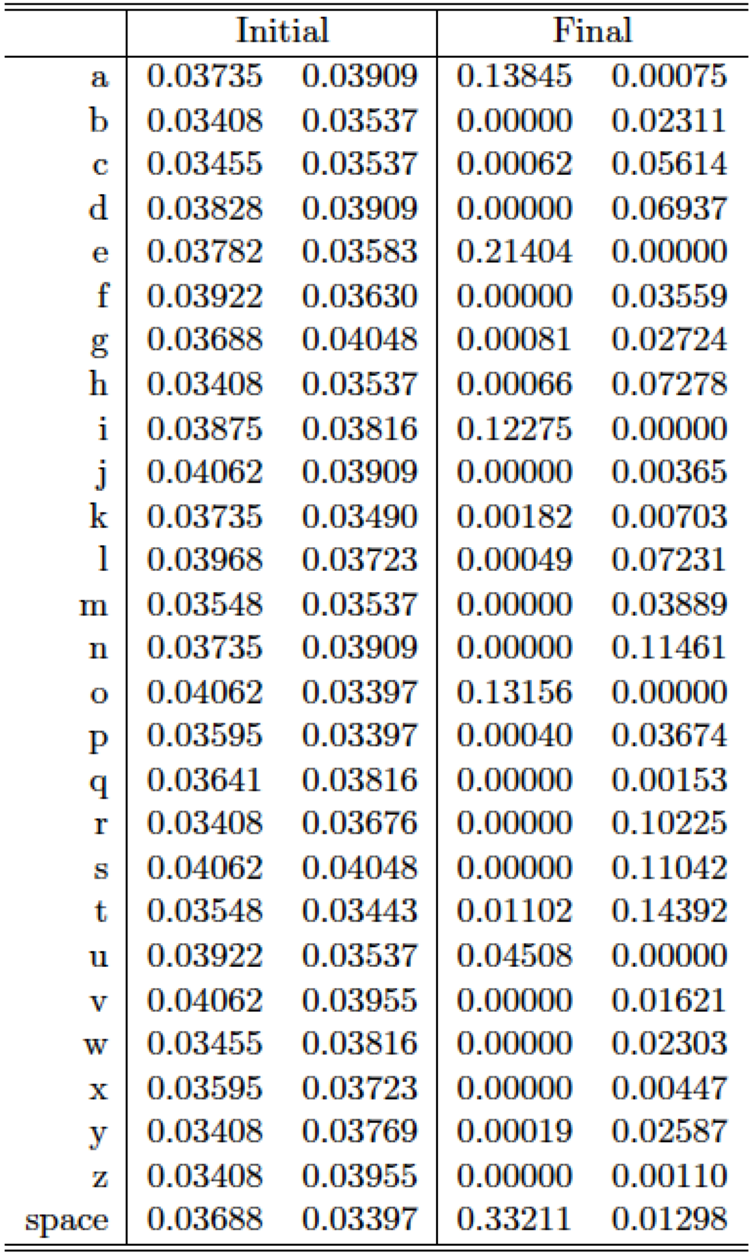
After 100 iterations, the model $\lambda = (A, B, \pi)$ has converged to

$$\pi = \begin{bmatrix} 0.00000 & 1.00000 \end{bmatrix}$$

and

$$A = \begin{bmatrix} 0.25596 & 0.74404 \\ 0.71571 & 0.28429 \end{bmatrix}$$

“A Revealing Introduction to Hidden Markov Models” - M. Stamp, 2012

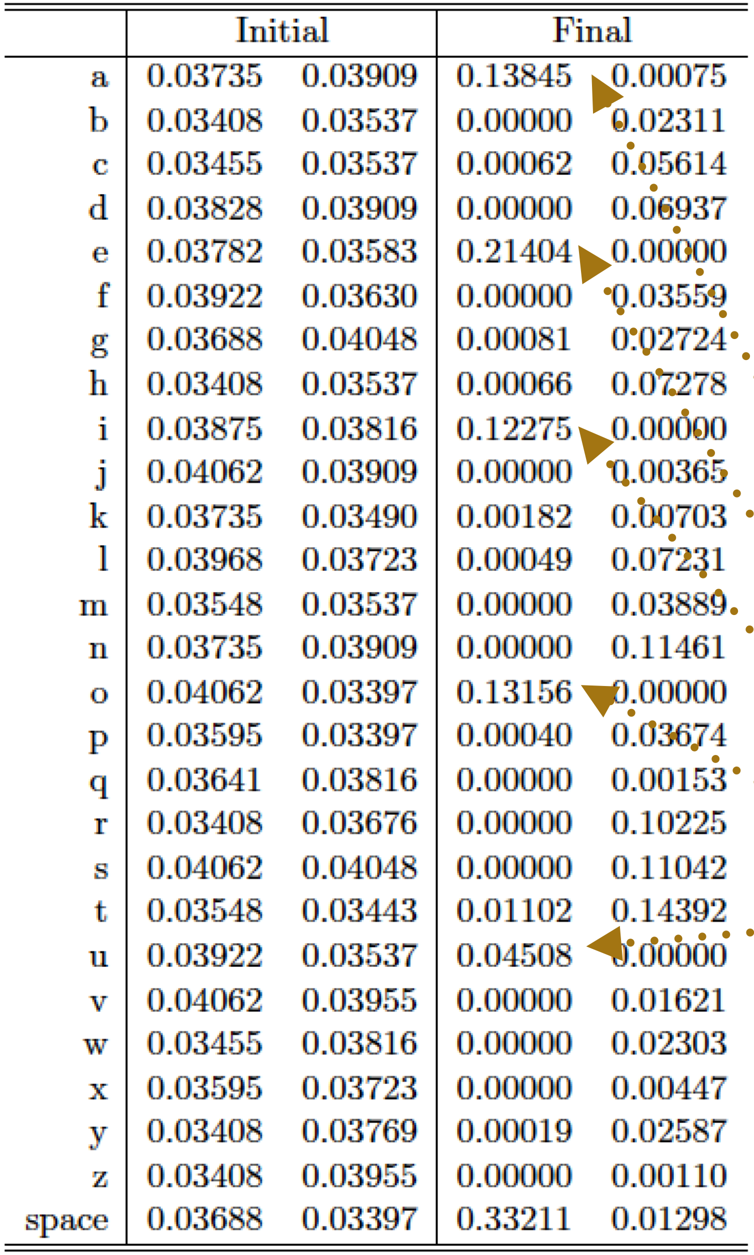


	Initial		Final	
a	0.03735	0.03909	0.13845	0.00075
b	0.03408	0.03537	0.00000	0.02311
c	0.03455	0.03537	0.00062	0.05614
d	0.03828	0.03909	0.00000	0.06937
e	0.03782	0.03583	0.21404	0.00000
f	0.03922	0.03630	0.00000	0.03559
g	0.03688	0.04048	0.00081	0.02724
h	0.03408	0.03537	0.00066	0.07278
i	0.03875	0.03816	0.12275	0.00000
j	0.04062	0.03909	0.00000	0.00365
k	0.03735	0.03490	0.00182	0.00703
l	0.03968	0.03723	0.00049	0.07231
m	0.03548	0.03537	0.00000	0.03889
n	0.03735	0.03909	0.00000	0.11461
o	0.04062	0.03397	0.13156	0.00000
p	0.03595	0.03397	0.00040	0.03674
q	0.03641	0.03816	0.00000	0.00153
r	0.03408	0.03676	0.00000	0.10225
s	0.04062	0.04048	0.00000	0.11042
t	0.03548	0.03443	0.01102	0.14392
u	0.03922	0.03537	0.04508	0.00000
v	0.04062	0.03955	0.00000	0.01621
w	0.03455	0.03816	0.00000	0.02303
x	0.03595	0.03723	0.00000	0.00447
y	0.03408	0.03769	0.00019	0.02587
z	0.03408	0.03955	0.00000	0.00110
space	0.03688	0.03397	0.33211	0.01298

Figure 3: Initial and final B transpose

"A Revealing Introduction to Hidden Markov Models" - M. Stamp, 2012

Initial and final emission probabilities



	Initial		Final	
a	0.03735	0.03909	0.13845	0.00075
b	0.03408	0.03537	0.00000	0.02311
c	0.03455	0.03537	0.00062	0.05614
d	0.03828	0.03909	0.00000	0.06937
e	0.03782	0.03583	0.21404	0.00000
f	0.03922	0.03630	0.00000	0.03559
g	0.03688	0.04048	0.00081	0.02724
h	0.03408	0.03537	0.00066	0.07278
i	0.03875	0.03816	0.12275	0.00000
j	0.04062	0.03909	0.00000	0.00365
k	0.03735	0.03490	0.00182	0.00703
l	0.03968	0.03723	0.00049	0.07231
m	0.03548	0.03537	0.00000	0.03889
n	0.03735	0.03909	0.00000	0.11461
o	0.04062	0.03397	0.13156	0.00000
p	0.03595	0.03397	0.00040	0.03674
q	0.03641	0.03816	0.00000	0.00153
r	0.03408	0.03676	0.00000	0.10225
s	0.04062	0.04048	0.00000	0.11042
t	0.03548	0.03443	0.01102	0.14392
u	0.03922	0.03537	0.04508	0.00000
v	0.04062	0.03955	0.00000	0.01621
w	0.03455	0.03816	0.00000	0.02303
x	0.03595	0.03723	0.00000	0.00447
y	0.03408	0.03769	0.00019	0.02587
z	0.03408	0.03955	0.00000	0.00110
space	0.03688	0.03397	0.33211	0.01298

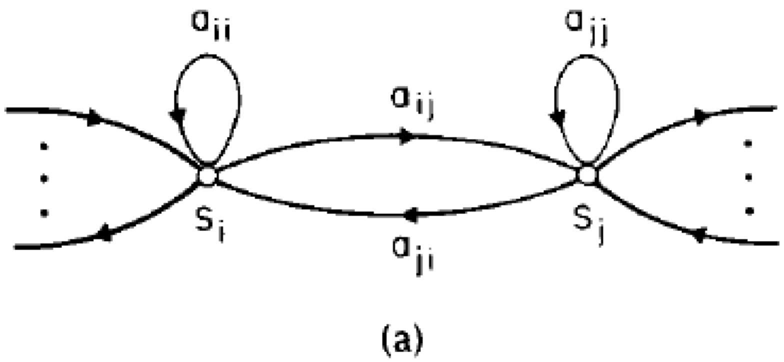
Initial and final emission probabilities

vowels

Figure 3: Initial and final B transpose

“A Revealing Introduction to Hidden Markov Models” - M. Stamp, 2012

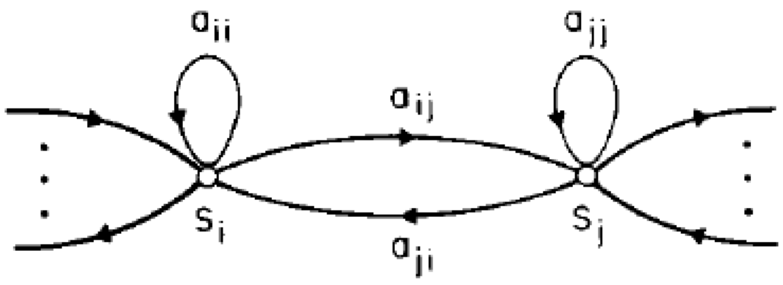
Duration in states



What is the distribution of time spent in each state?

A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition - L.R. RABINER

Duration in states

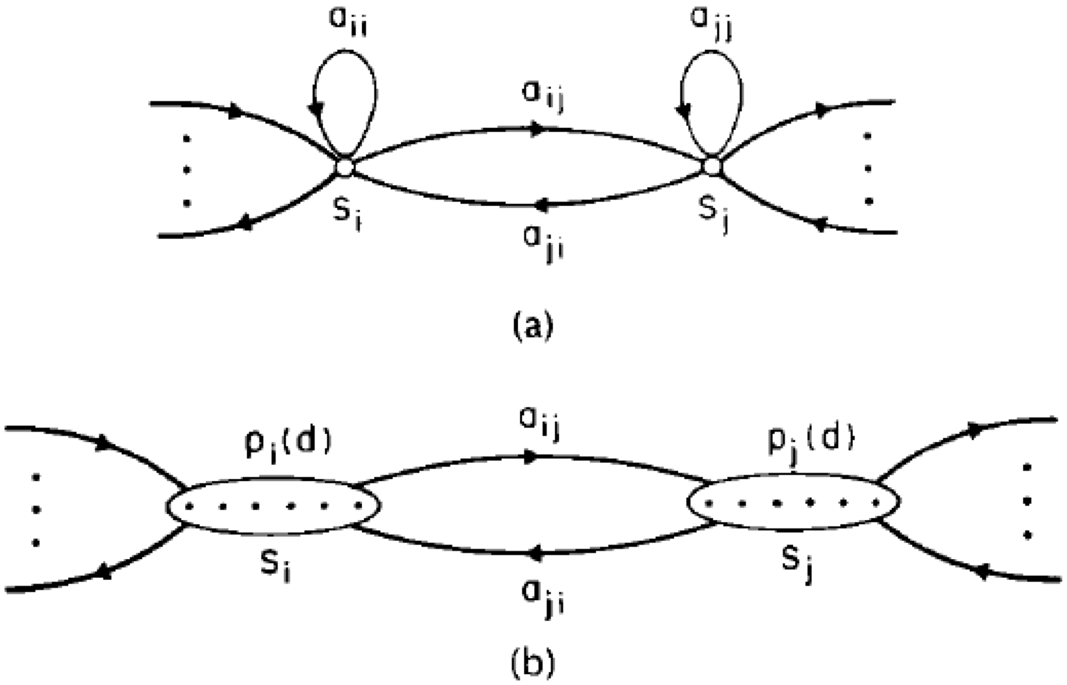


(a)

What is the distribution of time spent in each state?

Discrete time -> duration is geometrically distributed
Continuous time -> duration is exponentially distributed

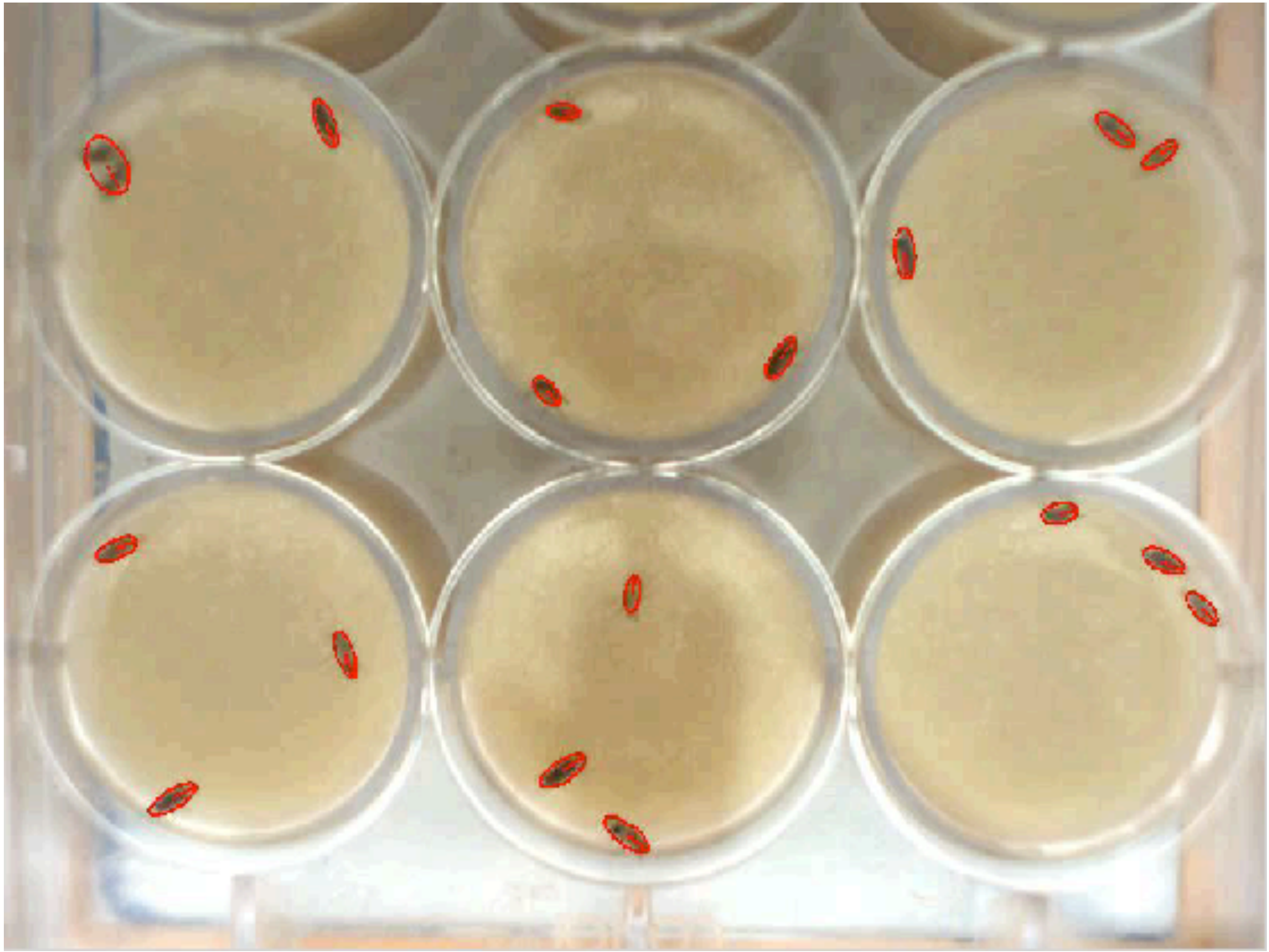
Duration in states



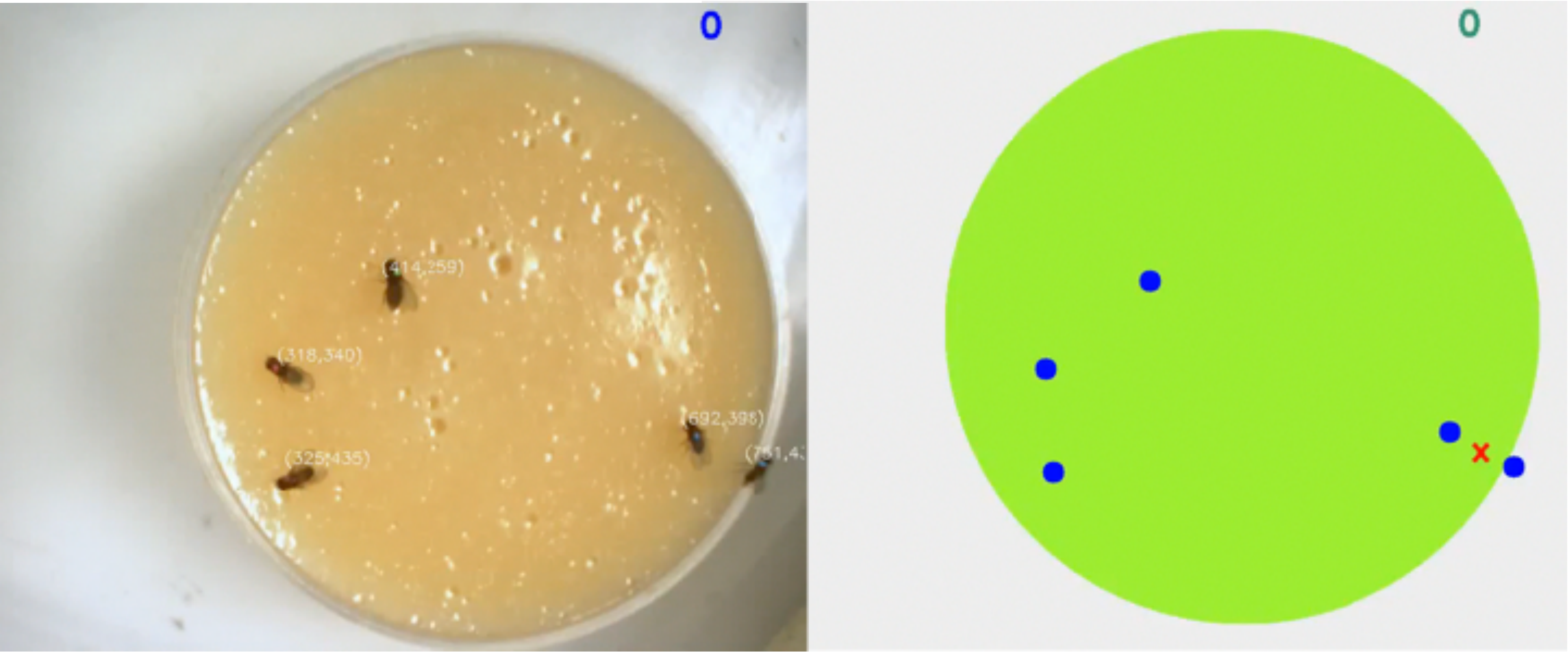
Generalized HMMs [(b) above] remove ‘self’ -> ‘self’ loops and instead define an arbitrary distribution for duration in each state.

A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition - L.R. RABINER

Application



Application extension



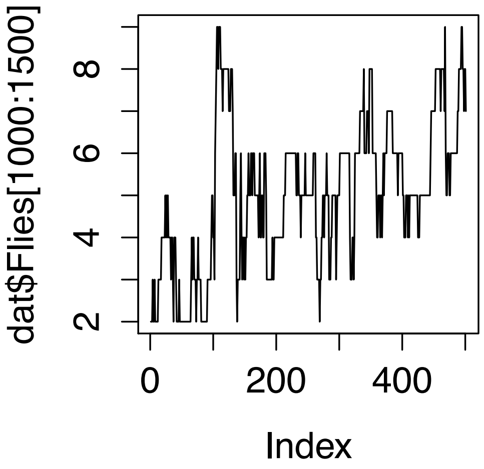
Goal: To automatically and accurately detect the number of flies on the food patch.

HMM elements for this problem

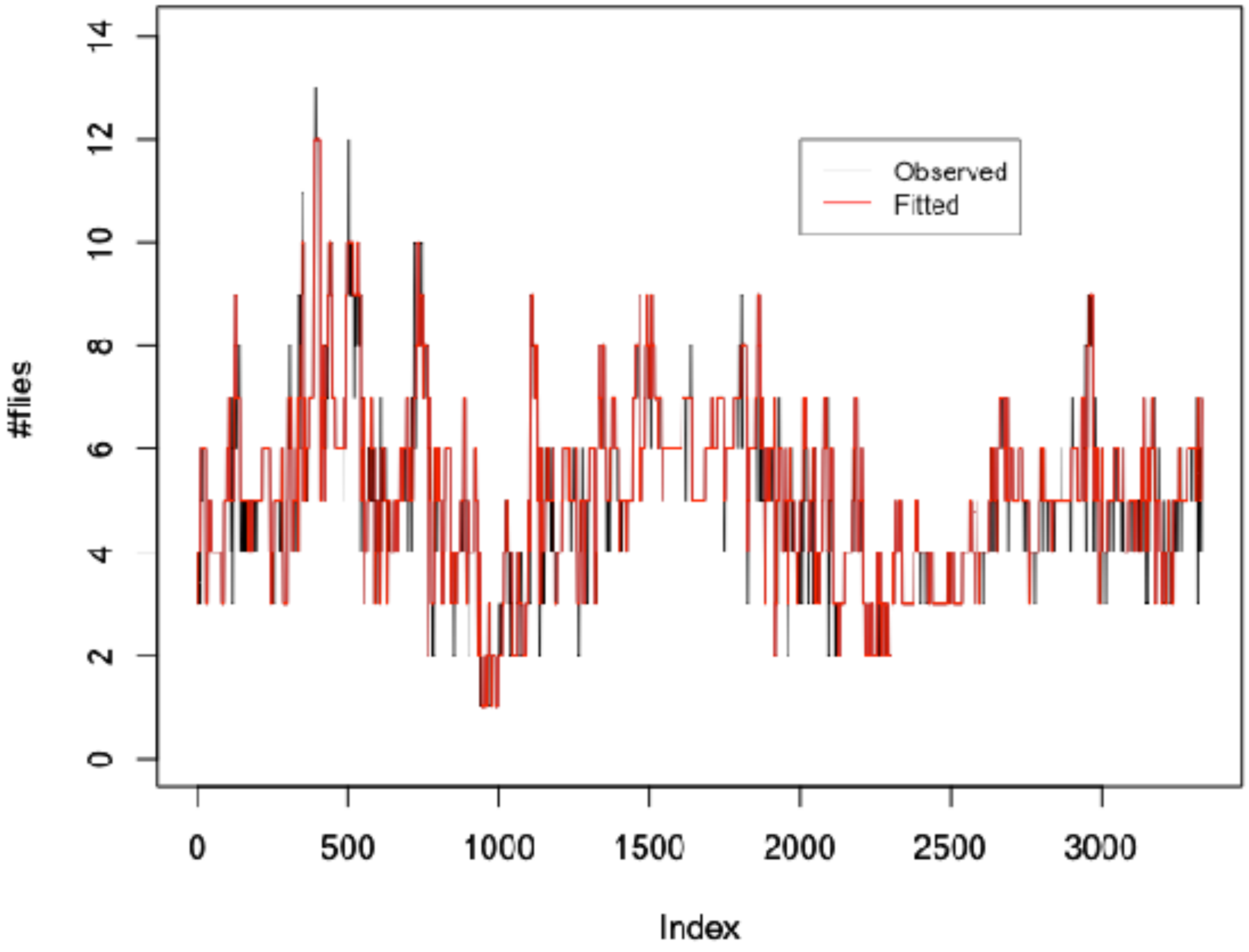
1. A finite number, N, of (hidden) states in the model. The number of flies on the food patch.

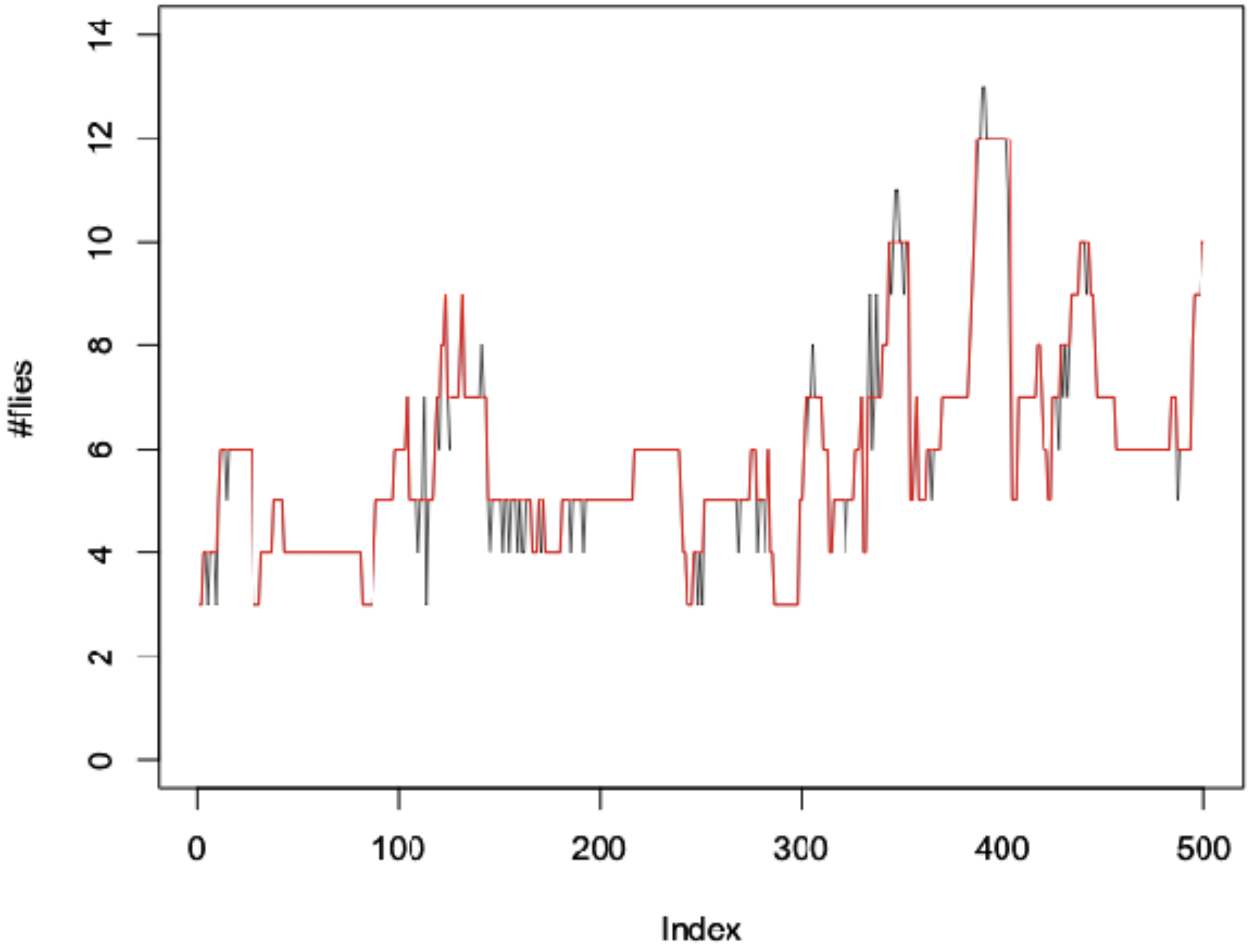
2. A Markov Chain that describes how the number of flies changes from frame-to-frame.

3. After each (potential) transition some observation is made. The observation is the number of 'flies' (i.e., dark patches) that are detected by the image processing software.



Can we use HMMs to smooth out the data (remove tracking noise)?


Observed
Fitted
#flies
0
2
4
6
8
10
12
14
0
500
1000
1500
2000
2500
3000
Index


#flies
0
2
4
6
8
10
12
14
0
100
200
300
400
500
Index

Further reading

Nice review paper:

Visser, I. (2011). “Seven things to remember about hidden Markov models: A tutorial on Markovian models for time series”. *Journal of Mathematical Psychology*, *55*(6), 403–415. doi:10.1016/j.jmp.2011.08.002

Course evaluation

1. Have your students access their evaluation forms by opening up an email from c-evals@usc.edu or following a link in Blackboard under "Course Evaluations" tab. They should **NOT** start their evaluation while the instructor is attending the session.

2. Instructors should verbally inform students of the evaluation process and may ask a student to proctor the evaluation. The following statement should be read to students:

"Learning Experience Evaluations are your opportunity to provide feedback to your instructor. USC and its faculty take these evaluations very seriously, as they provide valuable information that faculty and schools can use to improve teaching. It is important to remember that the learning process is collaborative and requires significant effort from the instructor, individual students, and the class as a whole. Please provide a thoughtful assessment of your experience, as well as of your own effort, with comments focused on specific aspects of instruction or the course. Comments on personal characteristics of the instructor are not appropriate and will not be considered. Evaluations should be completed individually with no undue influence by either a student or instructor. Should any inappropriate behavior occur, it should be reported to the Office of Institutional Research."

END