8: Hidden Markov Models

Machine Learning and Real-world Data

Simone Teufel (some slides by Helen Yannakoudakis)

Department of Computer Science and Technology
University of Cambridge

- So far we've looked at (statistical) classification.
- Experimented with different ideas for sentiment detection.
- Let us now talk about …

- So far we've looked at (statistical) classification.
- Experimented with different ideas for sentiment detection.
- Let us now talk about . . . the weather!

Weather prediction

- Two types of weather: rainy and cloudy
- The weather doesn't change within the day

Weather prediction

- Two types of weather: rainy and cloudy
- The weather doesn't change within the day
- Can we guess what the weather will be like tomorrow?

Weather prediction

- Two types of weather: rainy and cloudy
- The weather doesn't change within the day
- Can we guess what the weather will be like tomorrow?
- We can use a history of weather observations:
$$P(w_t = Rainy \,|\, w_{t-1} = Rainy, w_{t-2} = Cloudy, \\ w_{t-3} = Cloudy, w_{t-4} = Rainy)$$

Weather prediction

- Two types of weather: rainy and cloudy
- The weather doesn't change within the day
- Can we guess what the weather will be like tomorrow?
- We can use a history of weather observations:
  $$P(w_t = Rainy \,|\, w_{t-1} = Rainy, w_{t-2} = Cloudy, w_{t-3} = Cloudy, w_{t-4} = Rainy)$$
- Markov Assumption (first order):
  $$P(w_t \,|\, w_{t-1}, w_{t-2}, \ldots, w_1) \approx P(w_t \,|\, w_{t-1})$$

Weather prediction

- Two types of weather: rainy and cloudy
- The weather doesn't change within the day
- Can we guess what the weather will be like tomorrow?
- We can use a history of weather observations:
  $$P(w_t = Rainy \,|\, w_{t-1} = Rainy, w_{t-2} = Cloudy, w_{t-3} = Cloudy, w_{t-4} = Rainy)$$
- Markov Assumption (first order):
  $$P(w_t \,|\, w_{t-1}, w_{t-2}, \ldots, w_1) \approx P(w_t \,|\, w_{t-1})$$
- The joint probability of a sequence of observations / events is then:
  $$P(w_1, w_2, \ldots, w_t) = \prod_{t=1}^{n} P(w_t \,|\, w_{t-1})$$

Markov Chains

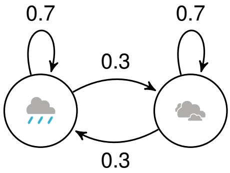
$$
\begin{array}{cc}
 & \text{Tomorrow} \\
\text{Today} &
\begin{array}{cc}
 & \begin{array}{cc} Rainy & Cloudy \end{array} \\
\begin{array}{c} Rainy \\ Cloudy \end{array} & \left[\begin{array}{cc} 0.7 & 0.3 \\ 0.3 & 0.7 \end{array}\right]
\end{array}
\end{array}
$$

Transition probability matrix

Markov Chains

$$
\text{Today} \quad \begin{array}{c} \\ Rainy \\ Cloudy \end{array}
\begin{array}{c}
\text{Tomorrow} \\
\begin{array}{cc} Rainy & Cloudy \end{array} \\
\begin{bmatrix} 0.7 & 0.3 \\ 0.3 & 0.7 \end{bmatrix}
\end{array}
$$

Transition probability matrix

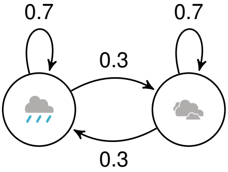


Two states: rainy and cloudy

Markov Chains

$$
\text{Today}\quad
\begin{array}{c}
\\
\mathit{Rainy} \\
\mathit{Cloudy}
\end{array}
\begin{array}{c}
\text{Tomorrow} \\
\begin{array}{cc}
\mathit{Rainy} & \mathit{Cloudy}
\end{array} \\
\begin{bmatrix}
0.7 & 0.3 \\
0.3 & 0.7
\end{bmatrix}
\end{array}
$$

Transition probability matrix



Two states: rainy and cloudy

- A Markov Chain is a stochastic process that embodies the Markov Assumption.
- Can be viewed as a probabilistic finite-state automaton.
- States are fully observable, finite and discrete; transitions are labelled with transition probabilities.
- Models sequential problems – your current situation depends on what happened in the past

Markov Chains

- Useful for modeling the probability of a sequence of events
    - Valid phone sequences in speech recognition
    - Sequences of speech acts in dialog systems (answering, ordering, opposing)
    - Predictive texting

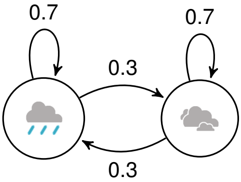
Markov Chains

- Useful for modeling the probability of a sequence of events that can be unambiguously observed
  - Valid phone sequences in speech recognition
  - Sequences of speech acts in dialog systems (answering, ordering, opposing)
  - Predictive texting

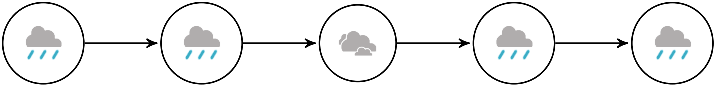
Markov Chains

- Useful for modeling the probability of a sequence of events that can be unambiguously observed
  - Valid phone sequences in speech recognition
  - Sequences of speech acts in dialog systems (answering, ordering, opposing)
  - Predictive texting
- What if we are interested in events that are not unambiguously observed?

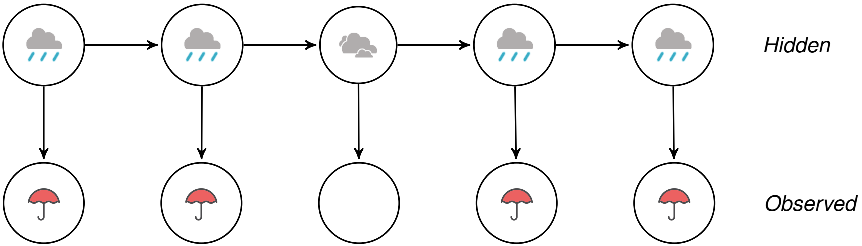
Markov Model



Markov Model: A Time-elapsed view

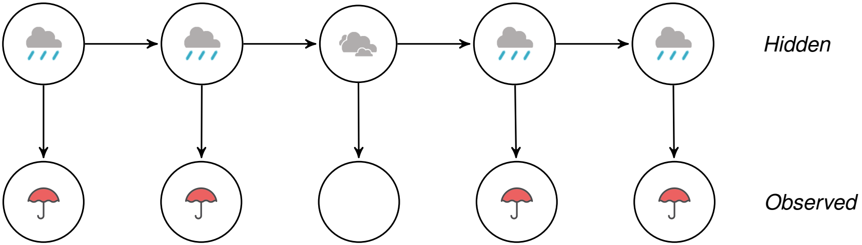


Hidden Markov Model: A Time-elapsed view



- Underlying Markov Chain over hidden states.
- We only have access to the observations at each time step.
- There is no 1:1 mapping between observations and hidden states.
- A number of hidden states can be associated with a particular observation, but the association of states and observations is governed by statistical behaviour.
- We now have to *infer* the sequence of hidden states that correspond to a sequence of observations.

Hidden Markov Model: A Time-elapsed view



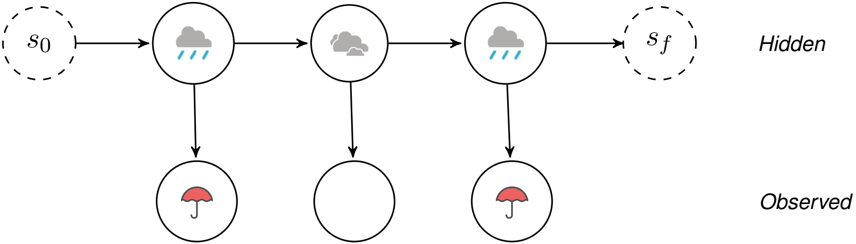
$$\begin{array}{c} \\ Rainy \\ Cloudy \end{array} \begin{array}{c} \begin{array}{cc} Rainy & Cloudy \end{array} \\ \begin{bmatrix} 0.7 & 0.3 \\ 0.3 & 0.7 \end{bmatrix} \end{array}$$

Transition probabilities
$P(w_t|w_{t-1})$

$$\begin{array}{c} \\ Rainy \\ Cloudy \end{array} \begin{array}{c} \begin{array}{cc} Umbrella & No\ umbrella \end{array} \\ \begin{bmatrix} 0.9 & 0.1 \\ 0.2 & 0.8 \end{bmatrix} \end{array}$$

Emission probabilities $P(o_t|w_t)$
(Observation likelihoods)

Hidden Markov Model: A Time-elapsed view – start and end states



- Could use initial probability distribution over hidden states.
- Instead, for simplicity, we will also model this probability as a transition, and we will explicitly add a special start state.
- Similarly, we will add a special end state to explicitly model the end of the sequence.
- Special start and end states not associated with “real” observations.

More formal definition of Hidden Markov Models; States and Observations

$S_e = \{s_1, \ldots, s_N\}$	a set of N emitting hidden states,
s_0	a special start state,
s_f	a special end state.
$K = \{k_1, \ldots k_M\}$	an output alphabet of M observations ("vocabulary").
k_0	a special start symbol,
k_f	a special end symbol.
$O = O_1 \ldots O_T$	a sequence of T observations, each one drawn from K.
$X = X_1 \ldots X_T$	a sequence of T states, each one drawn from S_e.

More formal definition of Hidden Markov Models; First-order Hidden Markov Model

1. **Markov Assumption (Limited Horizon):** Transitions depend only on current state:

$$P(X_t|X_1...X_{t-1}) \approx P(X_t|X_{t-1})$$

2. **Output Independence:** Probability of an output observation depends only on the current state and not on any other states or any other observations:

$$P(O_t|X_1...X_t, ..., X_T, O_1, ..., O_t, ..., O_T) \approx P(O_t|X_t)$$

More formal definition of Hidden Markov Models; State Transition Probabilities

a_{ij} is the probability of moving from state s_i to state s_j:

$$a_{ij} = P(X_t = s_j | X_{t-1} = s_i)$$

$$\forall_i \sum_{j=0}^{N+1} a_{ij} = 1$$

Special Start state s_0 and end state s_f:

- Not associated with “real” observations.
- a_{0i} describe transition probabilities out of the start state into state s_i.
- a_{if} describe transition probabilities into the end state.
- Transitions into start state (a_{i0}) and out of end state (a_{fi}) undefined.

More formal definition of Hidden Markov Models; State Transition Probabilities

A: a state transition probability matrix of size $(N+2)\times(N+2)$.

$$A = \begin{bmatrix} - & a_{01} & a_{02} & a_{03} & . & . & . & a_{0N} & - \\ - & a_{11} & a_{12} & a_{13} & . & . & . & a_{1N} & a_{1f} \\ - & a_{21} & a_{22} & a_{23} & . & . & . & a_{2N} & a_{2f} \\ - & . & . & . & & & & . & . \\ - & . & . & . & & & & . & . \\ - & . & . & . & & & & . & . \\ - & a_{N1} & a_{N2} & a_{N3} & . & . & . & a_{NN} & a_{Nf} \\ - & - & - & - & - & - & - & - & - \end{bmatrix}$$

a_{ij} is the probability of moving from state s_i to state s_j:

$$a_{ij} = P(X_t = s_j | X_{t-1} = s_i)$$

$$\forall_i \sum_{j=0}^{N+1} a_{ij} = 1$$

More formal definition of Hidden Markov Models; State Transition Probabilities

A: a state transition probability matrix of size $(N+2) \times (N+2)$.

$$A = \begin{bmatrix} - & a_{01} & a_{02} & a_{03} & . & . & . & a_{0N} & - \\ - & a_{11} & a_{12} & a_{13} & . & . & . & a_{1N} & a_{1f} \\ - & a_{21} & a_{22} & a_{23} & . & . & . & a_{2N} & a_{2f} \\ - & . & . & . & & & & . & . \\ - & . & . & . & & & & . & . \\ - & . & . & . & & & & . & . \\ - & a_{N1} & a_{N2} & a_{N3} & . & . & . & a_{NN} & a_{Nf} \\ - & - & - & - & - & - & - & - & - \end{bmatrix}$$

a_{ij} is the probability of moving from state s_i to state s_j:

$$a_{ij} = P(X_t = s_j | X_{t-1} = s_i)$$

$$\forall_i \sum_{j=0}^{N+1} a_{ij} = 1$$

More formal definition of Hidden Markov Models; State Transition Probabilities

A: a state transition probability matrix of size $(N+2) \times (N+2)$.

$$A = \begin{bmatrix} - & a_{01} & a_{02} & a_{03} & . & . & . & a_{0N} & - \\ - & a_{11} & a_{12} & a_{13} & . & . & . & a_{1N} & a_{1f} \\ - & a_{21} & a_{22} & a_{23} & . & . & . & a_{2N} & a_{2f} \\ - & . & . & . & & & & . & . \\ - & . & . & . & & & & . & . \\ - & . & . & . & & & & . & . \\ - & a_{N1} & a_{N2} & a_{N3} & . & . & . & a_{NN} & a_{Nf} \\ - & - & - & - & - & - & - & - & - \end{bmatrix}$$

a_{ij} is the probability of moving from state s_i to state s_j:

$$a_{ij} = P(X_t = s_j | X_{t-1} = s_i)$$

$$\forall_i \sum_{j=0}^{N+1} a_{ij} = 1$$

More formal definition of Hidden Markov Models; State Transition Probabilities

A: a state transition probability matrix of size $(N+2) \times (N+2)$.

$$A = \begin{bmatrix} - & a_{01} & a_{02} & a_{03} & . & . & . & a_{0N} & - \\ - & a_{11} & a_{12} & a_{13} & . & . & . & a_{1N} & a_{1f} \\ - & a_{21} & a_{22} & a_{23} & . & . & . & a_{2N} & a_{2f} \\ - & . & . & . & & & & . & . \\ - & . & . & . & & & & . & . \\ - & . & . & . & & & & . & . \\ - & a_{N1} & a_{N2} & a_{N3} & . & . & . & a_{NN} & a_{Nf} \\ - & - & - & - & - & - & - & - & - \end{bmatrix}$$

a_{ij} is the probability of moving from state s_i to state s_j:

$$a_{ij} = P(X_t = s_j | X_{t-1} = s_i)$$

$$\forall_i \sum_{j=0}^{N+1} a_{ij} = 1$$

More formal definition of Hidden Markov Models; Emission Probabilities

B: an emission probability matrix of size $(M+2) \times (N+2)$.

$$B = \begin{bmatrix} b_0(k_0) & - & - & - & - & - & - & - & - \\ - & b_1(k_1) & b_2(k_1) & b_3(k_1) & . & . & . & b_N(k_1) & - \\ - & b_1(k_2) & b_2(k_2) & b_3(k_2) & . & . & . & b_N(k_2) & - \\ - & . & . & . & & & & . & - \\ - & . & . & . & & & & . & - \\ - & . & . & . & & & & . & - \\ - & b_1(k_M) & b_2(k_M) & b_3(k_M) & . & . & . & b_N(k_M) & - \\ - & - & - & - & - & - & - & - & b_f(k_f) \end{bmatrix}$$

$b_i(k_j)$ is the probability of emitting vocabulary item k_j from state s_i:

$$b_i(k_j) = P(O_t = k_j | X_t = s_i)$$

Our HMM is defined by its parameters $\mu = (A, B)$.

More formal definition of Hidden Markov Models; Emission Probabilities

B: an emission probability matrix of size $(M+2) \times (N+2)$.

$$B = \begin{bmatrix} b_0(k_0) & - & - & - & - & - & - & - & - \\ - & b_1(k_1) & b_2(k_1) & b_3(k_1) & . & . & . & b_N(k_1) & - \\ - & b_1(k_2) & b_2(k_2) & b_3(k_2) & . & . & . & b_N(k_2) & - \\ - & . & . & . & & & & . & - \\ - & . & . & . & & & & . & - \\ - & . & . & . & & & & . & - \\ - & b_1(k_M) & b_2(k_M) & b_3(k_M) & . & . & . & b_N(k_M) & - \\ - & - & - & - & - & - & - & - & b_f(k_f) \end{bmatrix}$$

$b_i(k_j)$ is the probability of emitting vocabulary item k_j from state s_i:

$$b_i(k_j) = P(O_t = k_j | X_t = s_i)$$

Our HMM is defined by its parameters $\mu = (A, B)$.

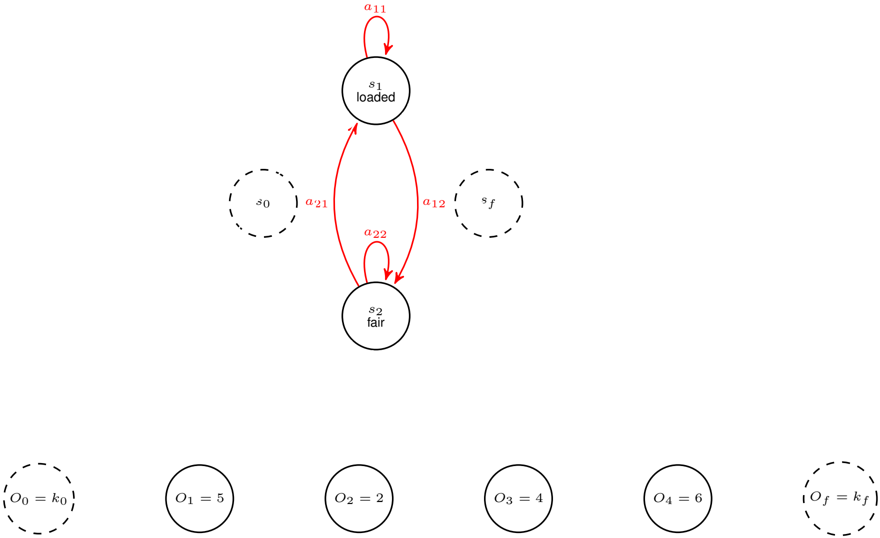
Examples where states are hidden

- Speech recognition
  - Observations: audio signal
  - States: phonemes
- Part-of-speech tagging (assigning tags like Noun and Verb to words)
  - Observations: words
  - States: part-of-speech tags
- Machine translation
  - Observations: target words
  - States: source words

Today's task: the dice HMM

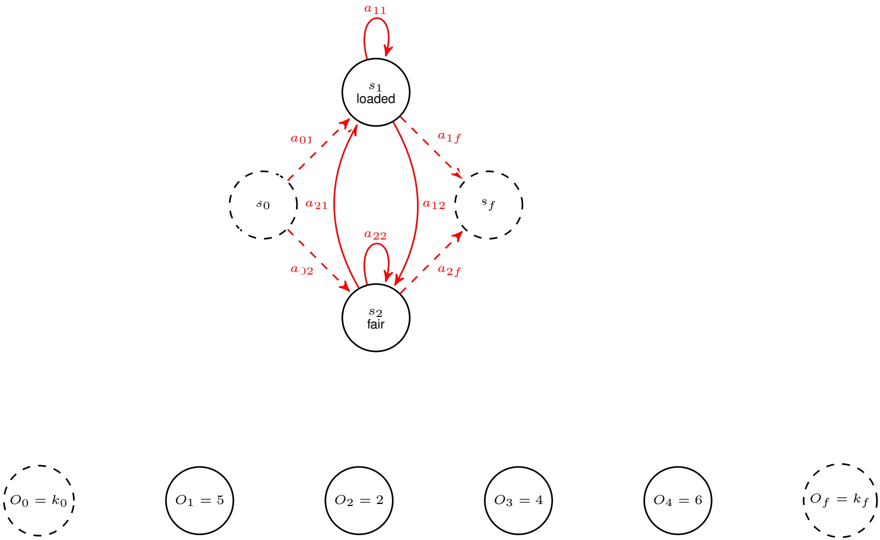
- Imagine a fraudulous croupier in a casino where customers bet on dice outcomes.
- She has two dice – a fair one and a loaded one.
- The fair one has the standard distribution of outcomes – $P(O) = \frac{1}{6}$ for each number 1 to 6.
- The loaded one has a different distribution.
- She secretly switches between the two dice.
- You don't know which dice is currently in use. You can only observe the numbers that are thrown.

Today's task: the dice HMM



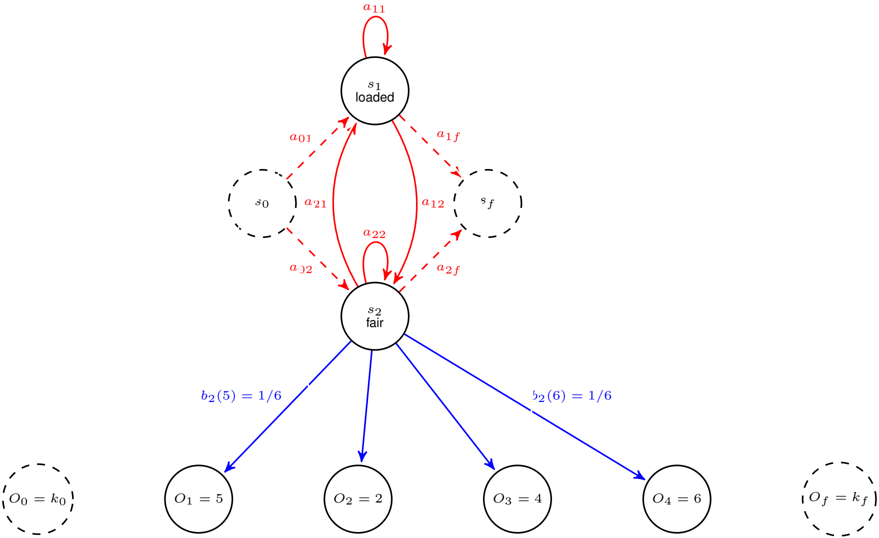
- States: fair and loaded, plus special states s_0 and s_f.
- Distribution of observations differs between the states.

Today's task: the dice HMM



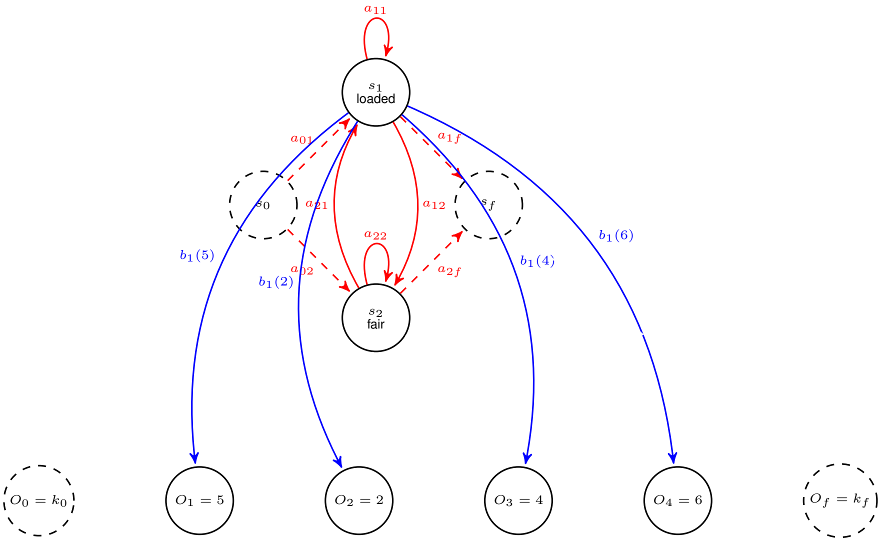
- States: fair and loaded, plus special states s_0 and s_f.
- Distribution of observations differs between the states.

Today's task: the dice HMM



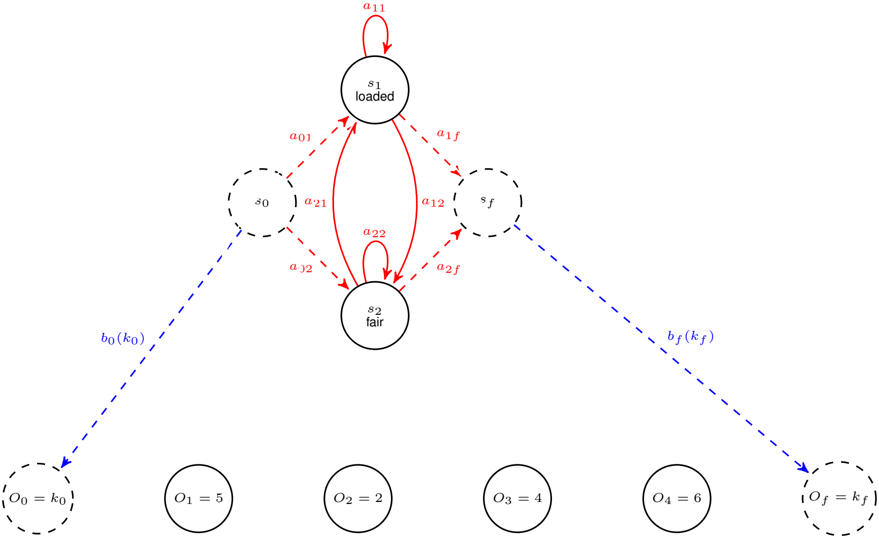
- States: fair and loaded, plus special states s_0 and s_f.
- Distribution of observations differs between the states.

Today's task: the dice HMM



- States: fair and loaded, plus special states s_0 and s_f.
- Distribution of observations differs between the states.

Today's task: the dice HMM



- States: fair and loaded, plus special states s_0 and s_f.
- Distribution of observations differs between the states.

Fundamental tasks with HMMs

- Problem 1 (Labelled Learning)
  - Given a parallel observation and state sequence O and X, learn the HMM parameters A and B. $\rightarrow$ today
- Problem 2 (Unlabelled Learning)
  - Given an observation sequence O (and only the set of emitting states S_e), learn the HMM parameters A and B.
- Problem 3 (Likelihood)
  - Given an HMM $\mu = (A, B)$ and an observation sequence O, determine the likelihood $P(O|\mu)$.
- Problem 4 (Decoding)
  - Given an observation sequence O and an HMM $\mu = (A, B)$, discover the best hidden state sequence X. $\rightarrow$ Task 8

Your Task today

Task 7:

- Your implementation performs labelled HMM learning, i.e. it has
  - Input: dual tape of state and observation (dice outcome) sequences X and O.

(s_0)	F	F	F	F	L	L	L	F	F	F	F	L	L	L	L	F	F	(s_f)
(k_0)	1	3	4	5	6	6	5	1	2	3	1	4	3	5	4	1	2	(k_f)

  - Output: HMM parameters A, B.
- Note: you will in a later task use your code for an HMM with more than two states. Either plan ahead now or modify your code later.

Parameter estimation of HMM parameters A, B

- Transition matrix A consists of transition probabilities a_{ij}

$$a_{ij} = P(X_{t+1} = s_j | X_t = s_i) \sim \frac{count_{trans}(X_t = s_i, X_{t+1} = s_j)}{count_{trans}(X_t = s_i)}$$

- Emission matrix B consists of emission probabilities $b_i(k_j)$

$$b_i(k_j) = P(O_t = k_j | X_t = s_i) \sim \frac{count_{emission}(O_t = k_j, X_t = s_i)}{count_{emission}(X_t = s_i)}$$

- (Add-one smoothed versions of these)

Literature

- Manning and Schutze (2000). Foundations of Statistical Natural Language Processing, MIT Press. Chapters 9.1, 9.2.
  - We use state-emission HMM instead of arc-emission HMM
  - We avoid initial state probability vector π by using explicit start and end states (s_0 and s_f) and incorporating the corresponding probabilities into the transition matrix A.
- (Jurafsky and Martin, 2nd Edition, Chapter 6.2 (but careful, notation!))
- Fosler-Lussier, Eric (1998). Markov Models and Hidden Markov Models: A Brief Tutorial. TR-98-041.
- Smith, Noah A. (2004). Hidden Markov Models: All the Glorious Gory Details.
- Bockmayr and Reinert (2011). Markov chains and Hidden Markov Models. Discrete Math for Bioinformatics WS 10/11.