CS 559: Machine Learning Fundamentals and Applications 11th Set of Notes

Instructor: Philippos Mordohai
Webpage: www.cs.stevens.edu/~mordohai
E-mail: Philippos.Mordohai@stevens.edu
Office: Lieb 215

Project Progress Reports

- Due on Tuesday Nov. 22 (see pdf)
  1. A description of the dataset and the available classes. Mention any unusual properties of the data.
  2. Short descriptions of the classification methods that will be used. There is no need to duplicate material from the slides.
  3. A description of pre-processing if it is important for your problem. This includes scaling the data if the features are inhomogeneous and the method requires it.
  4. Preliminary results using the simplest classifier. This step should help identify potential difficulties with the project.

- Homework 4 due on Monday Nov. 28

Schedule Changes

- Week 12: deep learning
- Week 13: unsupervised learning
- Week 14 (Dec. 7 and 9): final project presentations

Project Presentations

- Present project in class on December 7 and 9
  - Send me PPT/PDF file by 5pm
  - 37 projects * 8 min = 296 minutes
  - 6 min presentation + 2 min Q&A
- Counts for 10% of total grade

Project Presentations

- Target audience: fellow classmates
- Content:
  - Define problem
  - Show connection to class material
    - What is being classified, what are the classes etc.
  - Describe data
    - Train/test splits etc.
  - Show results
    - If additional experiments are in progress, describe them

Final Report

- Due December 12 (23:59)
- 6-10 pages including figures, tables and references
- Counts for 15% of total grade
- NO LATE SUBMISSIONS

Hidden Markov Models

Markov Chains

- Goal: make a sequence of decisions
  - Processes that unfold in time, states at time t are influenced by a state at time t-1
  - Applications: speech recognition, gesture recognition, parts of speech tagging and DNA sequencing,
  - Any temporal process without memory
    $\omega^T = \{\omega(1), \omega(2), \omega(3), ..., \omega(T)\}$ sequence of states
    We might have $\omega^6 = \{\omega_1, \omega_4, \omega_2, \omega_2, \omega_1, \omega_4\}$
  - The system can revisit a state at different steps and not every state needs to be visited

First-order Markov Models

- The production of any sequence is described by the transition probabilities

  $P(\omega_j(t + 1) \mid \omega_i(t)) = a_{ij}$

- Notes
  - a_{ij} does not have to be symmetric
  - a_{ii} is not 0 in general

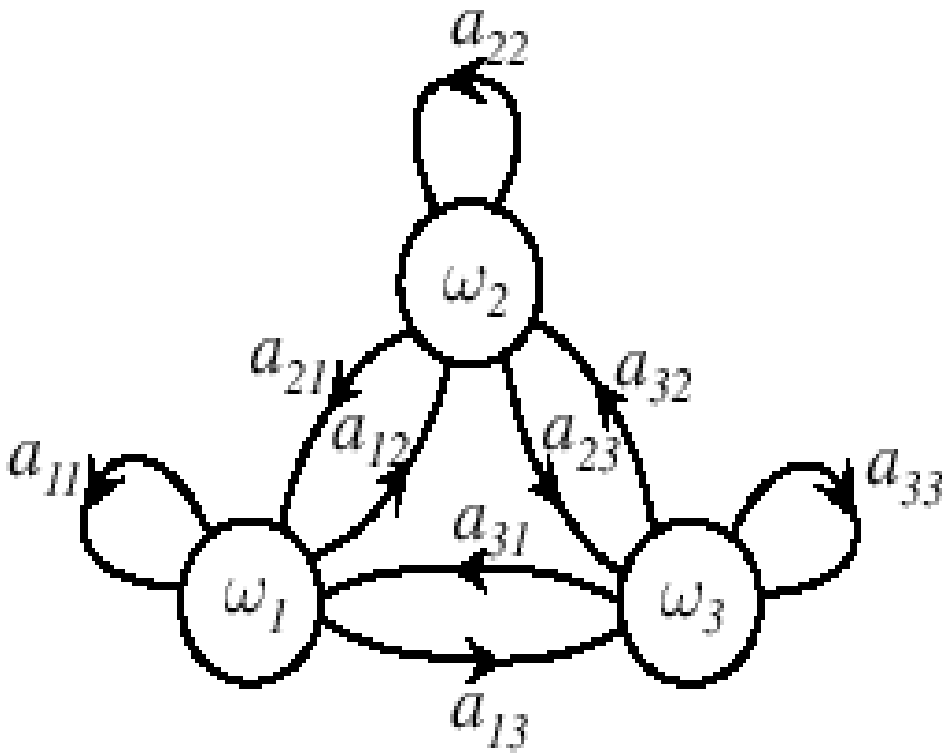


FIGURE 3.8. The discrete states, ω_i, in a basic Markov model are represented by nodes, and the transition probabilities, a_{ij}, are represented by links. In a first-order discrete-time Markov model, at any step t the full system is in a particular state $\omega(t)$. The state at step $t+1$ is a random function that depends solely on the state at step t and the transition probabilities. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Speech Analysis

Production of the word: “pattern” represented by phonemes

/p/ /a/ /tt/ /er/ /n/ // (// = silent state)

Transitions from /p/ to /a/, /a/ to /tt/, /tt/ to er/, /er/ to /n/ and /n/ to a silent state

Example: 3 Coins

Assume there are 3 coins:

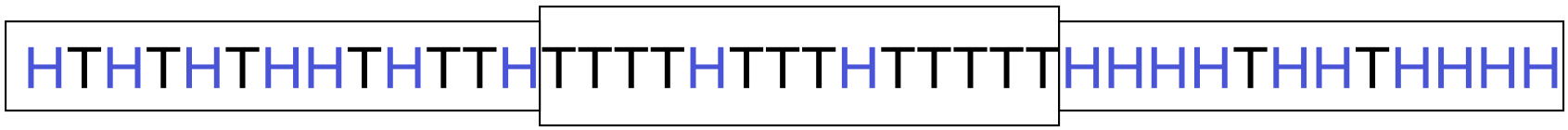
- one biased towards Heads
- one biased towards Tails
- one non-biased

The person behind the curtain tosses one coin repeatedly, then switches to another and tosses that ...

Example: 3 Coins

- Assume you see these observations:
- HTHTHTHHTHTTHTTTTHTTTHTTTTTTHHHHTHHTHHHH

What would you think is the most likely explanation as to which coins he is using?



Definition

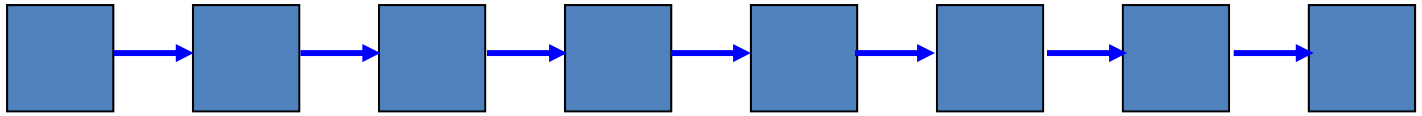
Doubly stochastic process with an underlying stochastic process that is not observable (hidden), but can only be observed through another set of stochastic processes that produce the sequence of observed symbols.

- The observations are the outcomes of the tosses
- The biased coins are the hidden states

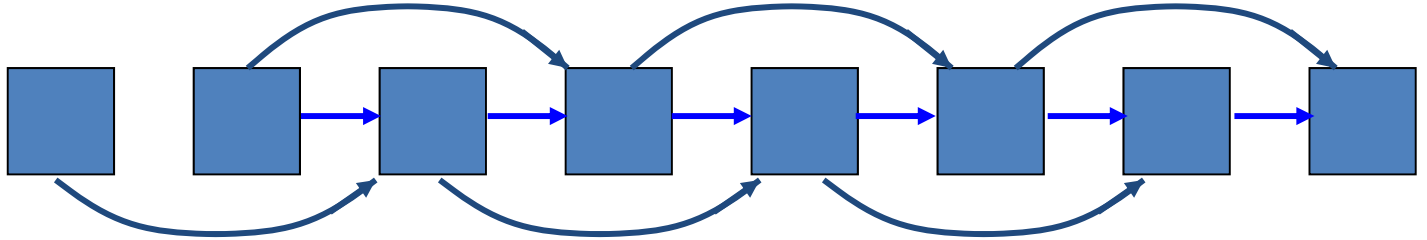
Markov Chains

$$\mathbf{x} = (x_1, x_2, \ldots, x_t, \ldots, x_T)$$

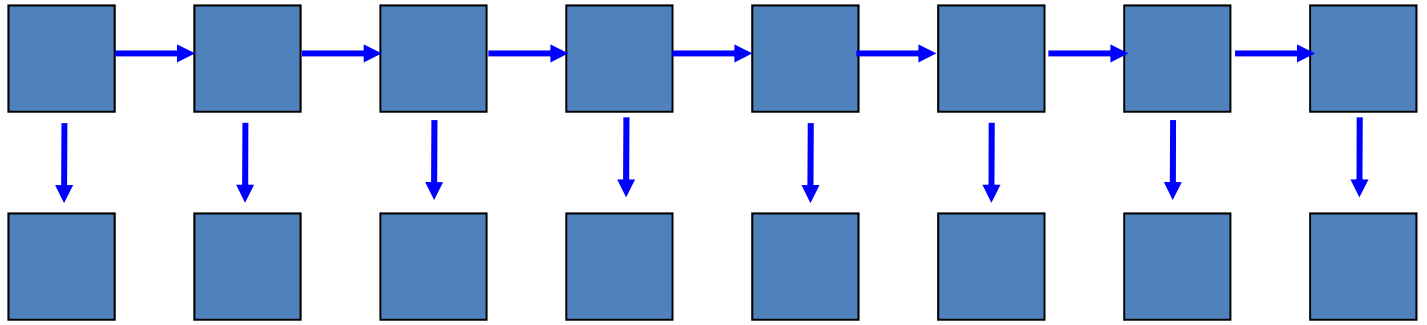
1st order Markov chain



2nd order Markov chain



1st order with stochastic observations -- HMM



Hidden Markov Model (HMM)

- Probabilities for emission of visible states:
  (a.k.a. emission probabilities)

$$b_{jk}=P(V_k(t) \mid \omega_j(t))$$

$$\sum_k b_{jk}= 1$$

- Probabilities for transitions between hidden states:
  (a.k.a. transition probabilities)

$$a_{ij}=P(\omega_j(t+1)\mid \omega_i(t))$$

$$\sum_j a_{ij}= 1$$

- Probabilities for the starting state being ω_i:

$$\pi_i$$

Hidden Markov Model Applications

- Three problems are associated with this model
  - The evaluation (likelihood) problem
  - The decoding problem (most likely hidden path)
  - The learning problem

The Evaluation Problem

The probability that the model produces a sequence V^T of visible states is:

$$P(V^T) = \sum_{r=1}^{r_{\max}} P(V^T \mid \omega_r^T) P(\omega_r^T)$$

where each r indexes a particular sequence $\omega_r^T = \{\omega(1), \omega(2), ..., \omega(T)\}$ of T hidden states.

$$(1) \qquad P(V^T \mid \omega_r^T) = \prod_{t=1}^{t=T} P(v(t) \mid \omega(t))$$

$$(2) \qquad \mathrm{P}(\omega_{\mathrm{r}}^{\mathrm{T}}) = \prod_{t=1}^{t=T} P(\omega(t) \mid \omega(t-1))$$

Using equations (1) and (2), we can write:

$$P(V^T) = \sum_{r=1}^{r_{\max}} \prod_{t=1}^{t=T} P(v(t) \mid \omega(t))\, P(\omega(t) \mid \omega(t-1))$$

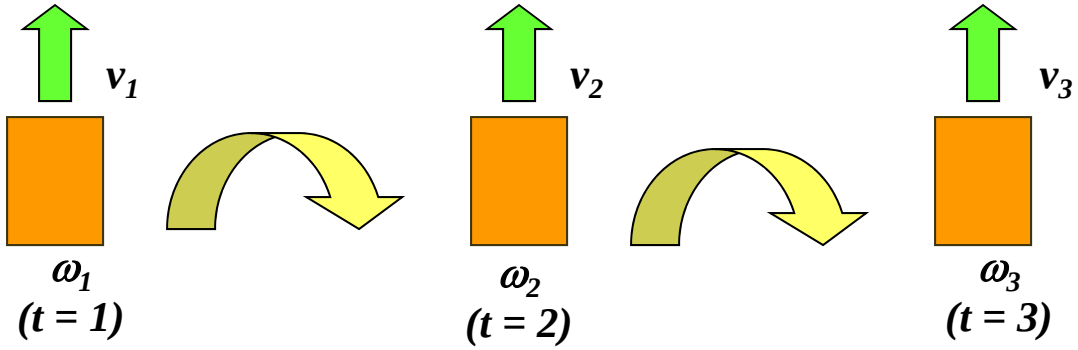
Interpretation: The probability that we observe the particular sequence of T visible states V^T is equal to the sum over all r_{max} possible sequences of hidden states of the conditional probability that the system has made a particular transition multiplied by the probability that it then emitted the visible symbol in our target sequence.

Example: Let $\omega_1, \omega_2, \omega_3$ be the hidden states; v_1, v_2, v_3 be the visible states

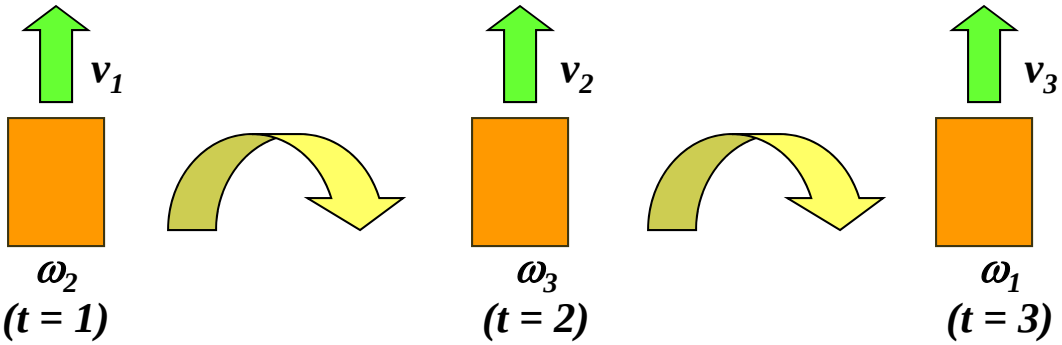
and $V^3 = \{v_1, v_2, v_3\}$ is the sequence of visible states

$P(\{v_1, v_2, v_3\}) = P(\omega_1)P(v_1 \mid \omega_1)P(\omega_2 \mid \omega_1)P(v_2 \mid \omega_2)P(\omega_3 \mid \omega_2)P(v_3 \mid \omega_3)$
+…+ (possible terms in the sum= all possible (3^3= 27) cases !)
In general $r_{max}=c^T$, where c is the number of states

First possibility:



Second Possibility:



$P(\{v_1, v_2, v_3\}) = P(\omega_2)P(v_1 \mid \omega_2)P(\omega_3 \mid \omega_2)P(v_2 \mid \omega_3)P(\omega_1 \mid \omega_3)P(v_3 \mid \omega_1) + \ldots +$

Therefore:

$$P(\{v_1, v_2, v_3\}) = \sum_{\substack{\textit{possible}\ \text{sequence} \\ \text{of hidden states}}} \prod_{t=1}^{t=3} P(v(t) \mid \omega(t))P(\omega(t) \mid \omega(t-1))$$

The Forward Algorithm

- Direct computation prohibitively expensive
- Feasible recursive alternative

$$a_j(t) = b_{jk} \left[\Sigma_i a_i(t-1) a_{ij}\right]$$

- The only nonzero contribution at *t* is transmission probability corresponding to visible state
- $[\Sigma_i a_i(t-1)a_{ij}]$ is a predictor term based on past data
- b_{jk} is a corrector term since it relies on current observation
  - Recall that $b_{jk}=P(V_k(t) \mid \omega_j(t))$

Algorithm

1. Initialize: $t \leftarrow 0$, a_{ij}, b_{jk}, visible sequence V^T, $a_j(0)$*
2. for $t \leftarrow t+1$
3. $\quad a_j(t) \leftarrow b_{jk} \Sigma\, a_i(t-1)\, a_{ij}$
4. until $t=T$
5. return $P(V^T)$

$a_j(t)$: probability of being in state ω_j at step t, having generated first t elements of V^T

$a_0(T)$ is probability of sequence ending at known final state

* assumes that first hidden state is known, otherwise consider all options using π_i and b_{jk} for the first state

DHS Chapter 3: Example 3

- What is the probability of observing $V^4=\{v_1, v_3, v_2, v_0\}$, given that the initial state is ω_1

$$a_{ij} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0.2 & 0.3 & 0.1 & 0.4 \\ 0.2 & 0.5 & 0.2 & 0.1 \\ 0.8 & 0.1 & 0 & 0.1 \end{bmatrix} \qquad b_{jk} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0.3 & 0.4 & 0.1 & 0.2 \\ 0 & 0.1 & 0.1 & 0.7 & 0.1 \\ 0 & 0.5 & 0.2 & 0.1 & 0.2 \end{bmatrix}$$

HHMs for Classification

- Given an HMM per class with parameters θ_m
- Compute $P(\theta_m|V^T)$ for all classes
  - Equivalent to computing $P(V^T|\theta_m)\ P(\theta_m)$
  - Compute $P(V^T|\theta_m)$ using forward or backward algorithm
  - Often assume $P(\theta_m)$ uniform
- Classify input according to maximum $P(\theta_m|V^T)$

The Decoding Problem: Optimal State Sequence

Given a sequence of visible states V^T, the decoding problem is to find the most probable sequence of hidden states.

This problem can be expressed mathematically as:

find the single "best" state sequence (hidden states)

$$\hat{\omega}(1),\hat{\omega}(2),...,\hat{\omega}(T) \ \textit{such that :}$$

$$\hat{\omega}(1),\hat{\omega}(2),...,\hat{\omega}(T) = \underset{\omega(1),\omega(2),...,\omega(T)}{\arg\max} P[\omega(1),\omega(2),...,\omega(T),v(1),v(2),...,V(T)\,|\,\lambda]$$

Viterbi algorithm (requires Dynamic Programming)

The Learning Problem: Parameter Estimation

This third problem consists of determining a method to adjust the model parameters θ = [π,A,B] to satisfy a certain optimization criterion. We need to find the best model

$$\hat{\theta} = [\hat{\pi}, \hat{A}, \hat{B}]$$

Such that to maximize the probability of the observation sequence:

$$\underset{\theta}{Max}\, P(V^T \mid \theta)$$

Baum-Welch algorithm (Generalized EM)

Example

- Consider gesture recognition
- Hidden states
  - A0: neutral pose
  - A1: extend arm sideways
  - A2: raise arm upwards
- Visible states
  - N: neutral
  - H: horizontal arm
  - V: vertical arm

- Assume initial state always neutral
- Also assume following transition and emission probabilities

To

From $A = \begin{bmatrix} 0.7 & 0.3 & 0 \\ 0.2 & 0.6 & 0.2 \\ 0 & 0.4 & 0.6 \end{bmatrix}$

Visible States

Hidden States $B = \begin{bmatrix} 0.8 & 0.2 & 0 \\ 0.2 & 0.6 & 0.2 \\ 0 & 0.3 & 0.7 \end{bmatrix}$

- Draw HMM diagram
- What is the probability of observing:
  - NHH
  - NVHV