6.891 Machine learning and neural networks

Tommi Jaakkola
MIT AI LAB
tommi@ai.mit.edu

LECTURE 14 (October 31, 2000)

Mid-term solutions are available
You'll get the exam back on Thursday

Topics

- Markov models
  - definition, properties
  - prediction, estimation
- Hidden markov models
  - definition, examples
  - forward-backward algorithm
  - dynamic programming
  - estimation via EM

Markov chains and Hidden Markov models

- Often we want to capture/model dynamical systems, not just static distributions
  1. Speech/language processing
  2. Human behavior (e.g., user modeling)
  3. Modeling physical/biological processes
  4. Stock market etc.
- Uncertainty $\Rightarrow$ probabilistic dynamical systems
- We need to define a class of probability models that we can estimate from observed behavior of the dynamical system

Markov chain: definition

- We define here a finite state Markov chain (stochastic finite state machine)
  1. **States**: $s \in \{1, \ldots, m\}$, where m is finite.
  2. **Starting state:** The starting state s_0 may be fixed or drawn from some a priori distribution $P_0(s_0)$.
  3. **Transitions**: we define how the system transitions from the current state s_t to the next state s_{t+1}

     The transitions satisfy the first order **Markov property:**

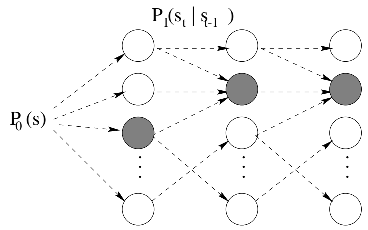
$$P(s_{t+1}|s_t, s_{t-1}, \ldots, s_0) = P_1(s_{t+1}|s_t) \quad (1)$$

- The resulting stochastic system generates a sequence of states

$$s_0 \to s_1 \to s_2 \to \ldots \quad (2)$$

where s_0 is drawn from $P_0(s_0)$ and s_{t+1} from $P_1(s_{t+1}|s_t)$ for all t

Markov chain: state diagram



- The initial state s_0 is drawn form $P_0(s_0)$.
- There are a set of possible transtions from each state. These are marked with dashed arrows and correspond to transitions for which $P_1(s'|s_t) > 0$.
- Given s_{t-1} we draw a new state s_t from $P_1(s_t|s_{t-1})$

$$s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow \ldots \tag{3}$$

- This is a *homogeneous Markov chain* where the transition probability only depends on the current state (and not on time t)

Markov chain: example

- The states correspond to words in a sentence
- The transitions are defined in terms of successive words in a sentence

  Example: a particular realization of the state sequence

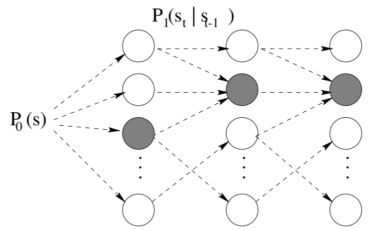
$$s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow \ldots \tag{4}$$

  might be

$$\text{This} \rightarrow \text{is} \rightarrow \text{a} \rightarrow \text{boring} \rightarrow \ldots \tag{5}$$

- Is a Markov chain an appropriate model in this context?

Markov chain: properties



$$s_0 \rightarrow s_1 \rightarrow s_2 \rightarrow \ldots$$

- Can we get from any one state to any other state in a finite number of transitions?
- If we can, the markov chain is *ergodic*

  (is the markov chain in the figure ergodic?)

Markov chains

- Problems we have to solve
  1. Prediction
  2. Estimation
- **Prediction**: Given that the system is in state $s_t = i$ at time t, what is the probability distribution over the possible states s_{t+k} at time $t + k$?

$$\begin{aligned}
&P_1(s_{t+1}|s_t = i) \\
&P_2(s_{t+2}|s_t = i) &=& \sum_{s_{t+1}} P_1(s_{t+1}|s_t = i)\, P_1(s_{t+2}|s_{t+1}) \\
&P_3(s_{t+3}|s_t = i) &=& \sum_{s_{t+2}} P_2(s_{t+2}|s_t = i)\, P_1(s_{t+3}|s_{t+2}) \\
& & \ldots & \\
&P_k(s_{t+k}|s_t = i) &=& \sum_{s_{t+k-1}} P_{k-1}(s_{t+k-1}|s_t = i)\, P_1(s_{t+k}|s_{t+k-1})
\end{aligned}$$

  where $P_k(s'|s)$ is the k-step transition probability matrix.

Markov chain: estimation

- We need to estimate the initial state distribution $P_0(s_0)$ and the transition probabilities $P_1(s'|s)$
- Estimation from L observed sequences of different lengths

$$s_0^{(1)} \to s_1^{(1)} \to s_2^{(1)} \to \ldots \to s_{n_1}^{(1)} \tag{6}$$
$$\ldots$$
$$s_0^{(L)} \to s_1^{(L)} \to s_2^{(L)} \to \ldots \to s_{n_L}^{(L)} \tag{7}$$

Maximum likelihood estimates (observed fractions)

$$\hat{P}_0(s_0 = i) \;=\; \frac{1}{L}\sum_{l=1}^{L} \delta(s_0^{(l)}, i) \tag{8}$$

where $\delta(x, y) = 1$ if $x = y$ and zero otherwise

Markov chain: estimation

$$s_0^{(1)} \to s_1^{(1)} \to s_2^{(1)} \to \ldots \to s_{n_1}^{(1)}$$
$$\ldots$$
$$s_0^{(L)} \to s_1^{(L)} \to s_2^{(L)} \to \ldots \to s_{n_L}^{(L)}$$

- The transition probabilities are obtained as observed fractions of transitions out of a specific state

  Joint estimate over successive states

$$\hat{P}(i, j) \;=\; \frac{1}{(\sum_{l=1}^{L} n_l)} \sum_{l=1}^{L} \sum_{t=0}^{n_l - 1} \delta(s_t^{(l)}, i)\delta(s_{t+1}^{(l)}, j) \tag{9}$$

  and the transition probability estimates

$$\hat{P}_1(s' = j | s = i) \;=\; \frac{\hat{P}(i, j)}{\sum_k \hat{P}(i, k)} \tag{10}$$

Markov chain: estimation

- Can we simply estimate Markov chains from a single long sequence?

$$s_0 \to s_1 \to s_2 \to \ldots \to s_n \tag{11}$$

  - What would our estimate for the initial state distribution $\hat{P}_0(s_0)$ be?
  - Ergodicity?

Hidden Markov models

- A hidden Markov model (HMM) is model where we generate a sequence of outputs in addition to the Markov state sequence

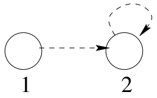
$$\begin{array}{ccccccc} s_0 & \to & s_1 & \to & s_2 & \to & \ldots \\ \downarrow & & \downarrow & & \downarrow & & \\ O_0 & & O_1 & & O_2 & & \end{array} \tag{12}$$

  A HMM is defined by
  1. number of states m
  2. initial state distribution $P_0(s_0)$
  3. state transition model $P_1(s_{t+1}|s_t)$
  4. output model $P_o(O_t|s_t)$ (discrete or continuous)
- This is a *latent variable model* in the sense that we will only observe the outputs $\{O_0, O_1, \ldots, O_n\}$; the state sequence remains "hidden"

HMM example

- Two states 1 and 2; observations are tosses of unbiased coins

$$P_0(s=1) = 0.5,\;\; P_0(s=2) = 0.5$$
$$P_1(s'=1|s=1) = 0,\;\; P_1(s'=2|s=1) = 1$$
$$P_1(s'=1|s=2) = 0,\;\; P_1(s'=2|s=2) = 1$$
$$P_o(O=\text{heads}|s=1) = 0.5,\;\; P_o(O=\text{tails}|s=1) = 0.5$$
$$P_o(O=\text{heads}|s=2) = 0.5,\;\; P_o(O=\text{tails}|s=2) = 0.5$$

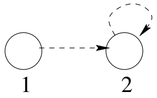


- This model is *unidentifiable* in the sense that the particular hidden state Markov chain has no effect on the observations

HMM example: biased coins

- Two states 1 and 2; outputs are tosses of *biased* coins

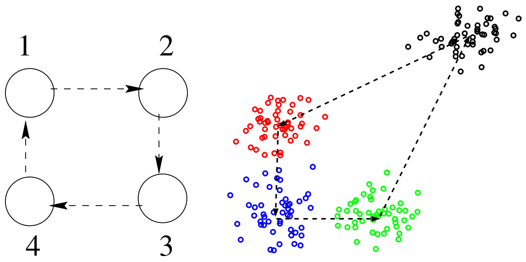
$$P_0(s=1) = 0.5,\;\; P_0(s=2) = 0.5$$
$$P_1(s'=1|s=1) = 0,\;\; P_1(s'=2|s=1) = 1$$
$$P_1(s'=1|s=2) = 0,\;\; P_1(s'=2|s=2) = 1$$
$$P_o(O=\text{heads}|s=1) = 0.25,\;\; P_o(O=\text{tails}|s=1) = 0.75$$
$$P_o(O=\text{heads}|s=2) = 0.75,\;\; P_o(O=\text{tails}|s=2) = 0.25$$



- What type of output sequences do we get from this HMM model?

HMM example

- Continuous output model: $O = [x_1, x_2]$, $P_o(O|s)$ is a Gaussian with mean and covariance depending on the underlying state s. Each state is initially equally likely.



- How does this compare to a mixture of four Gaussians model?

HMMs in practice

- HMMs have been widely used in various contexts
- Speech recognition (single word recognition)
  - words correspond to sequences of observations
  - we estimate a HMM for each word
  - the output model is a mixture of Gaussians over spectral features
- Biosequence analysis
  - a single HMM model for each type of protein (sequence of amino acids)
  - gene identification (parsing the genome)

  etc.

- HMMs are closely related to Kalman filters

HMM problems

- There are several problems we have to solve
  1. How do we evaluate the probability that our model generated the observation sequence $\{O_0, O_1, \ldots, O_n\}$?
     - forward-backward algorithm
  2. How do we uncover the most likely hidden state sequence corresponding to these observations?
     - dynamic programming
  3. How do we adapt the parameters of the HMM to better account for the observations?
     - the EM-algorithm

Probability of observed data

- In principle computing the probability of the observed sequence involves summing over exponentially many possible hidden state sequences

$$P(O_0, \ldots, O_n) = \sum_{s_0, \ldots, s_n} \overbrace{P_0(s_0)P_1(O_0|s_0)\ldots P_1(s_n|s_{n-1})P_o(O_n|s_n)}^{\text{Prob. given a specific hidden state sequence}}$$

$$\begin{array}{ccccccccc} s_0 & \rightarrow & s_1 & \rightarrow & s_2 & \rightarrow \ldots \rightarrow & s_{n-1} & \rightarrow & s_n \\ \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\ O_0 & & O_1 & & O_2 & & O_{n-1} & & O_n \end{array}$$

- We can, however, exploit the structure of the model to evaluate the probability much more efficiently

Probability of observed data

$$\begin{array}{ccccccccc} s_0 & \rightarrow & s_1 & \rightarrow & s_2 & \rightarrow \ldots \rightarrow & s_{n-1} & \rightarrow & s_n \\ \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\ O_0 & & O_1 & & O_2 & & O_{n-1} & & O_n \end{array}$$

- Forward probabilities $\alpha_t(i)$:

$$\alpha_t(i) = P(O_0, \ldots, O_t, s_t = i) \tag{13}$$

These can be updated recursively

$$\begin{aligned} \alpha_0(i) &= P_0(s_0 = i)\, P_o(O_0|s_0 = i) \\ \alpha_1(i) &= \sum_j \alpha_0(j)\, P_1(s_1 = i|s_0 = j) P_o(O_1|s_1 = i) \\ &\ldots \\ \alpha_n(i) &= \sum_j \alpha_{n-1}(j)\, P_1(s_n = i|s_{n-1} = j) P_o(O_n|s_n = i) \end{aligned}$$

Forward-propagation

Recursive updates

$$\begin{aligned} \alpha_0(i) &= P_0(s_0 = i)\, P_o(O_0|s_0 = i) \\ \alpha_1(i) &= \sum_j \alpha_0(j)\, P_1(s_1 = i|s_0 = j) P_o(O_1|s_1 = i) \\ &\ldots \\ \alpha_n(i) &= \sum_j \alpha_{n-1}(j)\, P_1(s_n = i|s_{n-1} = j) P_o(O_n|s_n = i) \end{aligned}$$

- As a result of our recursive computation, we'll end up with

$$\alpha_n(i) = P(O_0, \ldots, O_n, s_n = i) \tag{14}$$

and thus the probability of the observed sequence is obtained easily as

$$P(O_0, \ldots, O_n) = \sum_i P(O_0, \ldots, O_n, s_n = i) = \sum_i \alpha_n(i) \tag{15}$$

Forward propagation cont'd

- Computational cost of the forward propagation algorithm
  - each propagation step costs about m^2 where m is the number of states

$$\alpha_{t+1}(i) \;=\; \sum_j \alpha_t(j)\, P_1(s_{t+1}=i|s_t=j) P_o(O_t|s_t=i)$$

  - there are n such updates before we get $\alpha_n(i)$
  - the computational cost of the forward-propagation algorithm is therefore $O(n\,m^2)$

- For later utility we also need an analogous backward propagation algorithm

Backward propagation

- Definition of backward propabilities $\beta_t(i)$:

$$\beta_n(i) = P(O_{t+1},\ldots,O_n|s_t=i) \tag{16}$$

Again, these probabilities can be updated recursively

$$\begin{aligned}
\beta_n(i) &= 1 \\
\beta_{n-1}(i) &= \sum_j P_1(s_n=j|s_{n-1}=i) P_o(O_n|s_n=j)\,\beta_n(j) \\
&\ldots \\
\beta_0(i) &= \sum_j P_1(s_1=j|s_0=i) P_o(O_1|s_1=j)\,\beta_1(j)
\end{aligned}$$

$$\begin{array}{ccccccccc}
s_0 & \rightarrow & s_1 & \rightarrow & s_2 & \rightarrow \ldots \rightarrow & s_{n-1} & \rightarrow & s_n \\
\downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\
O_0 & & O_1 & & O_2 & & O_{n-1} & & O_n
\end{array}$$

Backward propagation cont'd

- The probability of observed sequence is now

$$\begin{aligned}
P(O_0,\ldots,O_n) &= \sum_i P_0(s_0=i) P_o(O_0|s_0=i)\, P(O_1,\ldots,O_n|s_0=i) \\
&= \sum_i P_0(s_0=i) P_o(O_0|s_0=i)\, \beta_0(i)
\end{aligned} \tag{17}$$

$$\begin{array}{ccccccccc}
s_0 & \rightarrow & s_1 & \rightarrow & s_2 & \rightarrow \ldots \rightarrow & s_{n-1} & \rightarrow & s_n \\
\downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\
O_0 & & O_1 & & O_2 & & O_{n-1} & & O_n
\end{array}$$