Hidden Markov Models

CISC 5800
Professor Daniel Leeds

Representing sequence data

- Spoken language
- DNA sequences
- Daily stock values

Example: spoken language

F?r plu? fi?e is nine
- Between F and r expect a vowel: “aw”, “ee”, “ah”; NOT “oh”, “uh”
- At end of “plu” expect consonant: “g”, “m”, “s”; NOT “d”, “p”

Markov Models

Start with:
- $n$ states: $s_1, ..., s_n$
- Probability of initial start states: $\Pi_1, ..., \Pi_n$
- Probability of transition between states: $A_{i,j} = P(q_t = s_j | q_{t-1} = s_i)$

A dice-y example

- Two colored die

What is the probability we start at $s_A$? $0.3$

What is the probability we have the sequence of die choices: $s_A, s_A, s_A, s_A$? $0.3 \times 0.8 = 0.24$

What is the probability we have the sequence of die choices: $s_B, s_A, s_B, s_A$? $0.7 \times 0.2 \times 0.2 \times 0.2 = 0.0056$
A dice-y example

- What is the probability we have the die choices $s_b$ at time $t=5$.

$$\Pi_A = 0.3, \Pi_B = 0.7$$

- Dynamic programming: find answer for $q_t$, then compute $q_{t+1}$.

| State | Time
|-------|------|
|       | $t_1$ | $t_2$ | $t_3$
| $s_A$ | 0.3   | 0.38  | 0.428
| $s_B$ | 0.7   | 0.62  | 0.572

$$p_t(i) = \sum_j p(q_t = s_j | q_{t-1} = s_i) p_{t-1}(j)$$

$$p_t(i) = P(q_t = s_i) \quad \text{-- Probability state } i \text{ at time } t$$

Hidden Markov Models

- Actual state $q$ “hidden”
- State produces visible data $o$: $\phi_{i,j} = P(o_t = x_i | q_t = s_j)$
- Compute

$$P(O, Q | \theta) = P(q_1 | \pi) \prod_{t=2}^{T} p(q_t | q_{t-1}, A) \prod_{t=1}^{T} p(o_t | q_t, \phi)$$

Probability observe value $x_i$ when state is $s_j$.

Intuition – balance transition and emission probabilities

Deducing die based on observed “emissions”

Each color is biased

| $\alpha$ | $P(o | s_A)$ | $P(o | s_B)$ |
|---------|-------------|-------------|
| 1       | 0.3         | 0.1         |
| 2       | 0.2         | 0.2         |
| 3       | 0.2         | 0.2         |
| 4       | 0.1         | 0.1         |
| 5       | 0.1         | 0.1         |
| 6       | 0.1         | 0.1         |

Observed numbers: 554565254556 – the 2 is probably from $s_B$

Observed numbers: 554565213321 – the 2 is probably from $s_A$

Deducing die based on observed “emissions”

Each color is biased

| $\alpha$ | $P(o | s_A)$ | $P(o | s_B)$ |
|---------|-------------|-------------|
| 1       | 0.3         | 0.1         |
| 2       | 0.2         | 0.2         |
| 3       | 0.2         | 0.2         |
| 4       | 0.1         | 0.1         |
| 5       | 0.1         | 0.1         |
| 6       | 0.1         | 0.1         |

- We see: 5
  What is probability of $o=5$, $q=B$ (blue)

$$\Pi_B \phi_{5,B} = 0.7 \times 0.2 = 0.14$$

- We see: 5, 3
  What is probability of $o=5,3$, $q=B, B$?

$$\Pi_B \phi_{5,B} A_{B,B} \phi_{3,B} = 0.7 \times 0.2 \times 0.8 \times 0.1 = 0.0112$$
Goal: calculate most likely states given observable data

Define and use $\delta_t(i)$

$\delta_t(i) = \max_{q_1 \cdots q_{t-1}} p(q_1 \cdots q_{t-1} \land q_t = s_i \land O_1 \cdots O_t)$

$\delta_t(i)$: max possible value of $P(q_1, \ldots, q_t, o_1, \ldots, o_t)$ given we insist $q_t = s_i$

Find the most likely path from $q_1$ to $q_t$ that

- $q_t = s_i$
- Outputs are $o_1, \ldots, o_t$

Viterbi algorithm: bigger picture

Compute all $\delta_t(i)$'s

- At time $t=1$ compute $\delta_1(i)$ for every state $i$
- At time $t=2$ compute $\delta_2(i)$ for every state $i$ (based on $\delta_1(i)$ values)
- ... At time $t=T$ compute $\delta_T(i)$ for every state $i$ (based on $\delta_{T-1}(i)$ values)

Find states going from $t=T$ back to $t=1$ to lead to max $\delta_T(i)$

- Now find state $j$ that gives maximum value for $\delta_T(j)$
- Find state $k$ at time $T-1$ used to maximize $\delta_T(j)$
- ...
- Find state $z$ at time 1 used to maximize $\delta_2(y)$

Viterbi algorithm: $\delta_t(i)$

$\delta_1(i) = \Pi_1 P(o_1 \mid q_1 = s_i) = \Pi_1 \phi_{o_1,i}$

$\delta_t(i) = P(o_t \mid q_t = s_i) \max_j \delta_{t-1}(j) P(q_t = s_i \mid q_{t-1} = s_j) = \phi_{o_t,i} \max_j \delta_{t-1}(j) A_{i,j}$

$P(Q^* \mid O) = \arg\max_Q P(Q \mid O) = \arg\max_i \delta_T(i)$

Viterbi in action: observe “5, 1”

$\Pi_A = 0.3, \Pi_B = 0.7$

<table>
<thead>
<tr>
<th></th>
<th>$P(o \mid s_A)$</th>
<th>$P(o \mid s_B)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>.3</td>
<td>.1</td>
</tr>
<tr>
<td>2</td>
<td>.2</td>
<td>.1</td>
</tr>
<tr>
<td>3</td>
<td>.2</td>
<td>.1</td>
</tr>
<tr>
<td>4</td>
<td>.1</td>
<td>.2</td>
</tr>
<tr>
<td>5</td>
<td>.1</td>
<td>.2</td>
</tr>
<tr>
<td>6</td>
<td>.1</td>
<td>.3</td>
</tr>
</tbody>
</table>

$\delta_2(A):$

$\delta_2(B):$

<table>
<thead>
<tr>
<th></th>
<th>$t=1$ ($o_1=5$)</th>
<th>$t=2$ ($o_2=1$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_1=s_A$</td>
<td>.3x.1 = .03</td>
<td>$q_1=s_B$</td>
</tr>
</tbody>
</table>
Parameters in HMM

Initial probabilities: \( \pi_i \)

Transition probabilities \( A_{i,j} \)

Emission probabilities \( \phi_{i,j} \)

How do we learn these values?

Learning HMM parameters: \( \pi_i \)

First, assume we know the states

Compute MLE for each parameter

\[
\pi^* = \arg \max_{\pi} \prod_{k=1}^{T} \pi(q_1) \prod_{t=2}^{T} p(q_t | q_{t-1}) \prod_{t=1}^{T} p(o_t | q_t, \phi)
\]

\[
\pi_A = \frac{\#D(q_1 = s_A)}{\#D}
\]

Note: we can add +1 to numerator (and number of states to denominator) to prevent \( \pi_A = 0 \)

\[
\pi_A = \frac{\#D(q_1 = s_A) + 1}{\#D + |Q|}
\]

Learning HMM parameters: \( A_{i,j} \)

Compute MLE for each parameter

\[
A^* = \arg \max_{A} \prod_{k} \pi(q_1) \prod_{t=2}^{T} p(q_t | q_{t-1}) \prod_{t=1}^{T} p(o_t | q_t, \phi)
\]

\[
A_{i,j} = \frac{\#D(\{q_t = s_i \land q_{t-1} = s_j\})}{\#D(\{q_{t-1} = s_j\})}
\]

Learning HMM parameters: \( \phi_{i,j} \)

Compute MLE for each parameter

\[
\phi^* = \arg \max_{\phi} \prod_{k} \pi(q_1) \prod_{t=2}^{T} p(q_t | q_{t-1}) \prod_{t=1}^{T} p(o_t | q_t, \phi)
\]

\[
\phi_{i,j} = \frac{\#D(o_t = i, q_t = s_j)}{\#D(q_t = s_j)}
\]

First, assume we know the states

Learning HMM parameters: \( x^1: A,B,A,A,B \)

Learning HMM parameters: \( x^2: B,B,B,A,A \)

Learning HMM parameters: \( x^3: A,A,B,A,B \)

Learning HMM parameters: \( x^4: A,B,A,A,B \)

Learning HMM parameters: \( x^5: B,B,B,A,A \)

Learning HMM parameters: \( x^6: A,A,B,A,B \)

Learning HMM parameters: \( x^7: A,B,A,A,B \)

Learning HMM parameters: \( x^8: B,B,B,A,A \)

Learning HMM parameters: \( x^9: A,A,B,A,B \)

Learning HMM parameters: \( x^{10}: A,B,A,A,B \)

Learning HMM parameters: \( x^{11}: B,B,B,A,A \)

Learning HMM parameters: \( x^{12}: A,A,B,A,B \)

Learning HMM parameters: \( x^{13}: A,B,A,A,B \)

Learning HMM parameters: \( x^{14}: B,B,B,A,A \)

Learning HMM parameters: \( x^{15}: A,A,B,A,B \)

Learning HMM parameters: \( x^{16}: A,B,A,A,B \)

Learning HMM parameters: \( x^{17}: B,B,B,A,A \)

Learning HMM parameters: \( x^{18}: A,A,B,A,B \)

Learning HMM parameters: \( x^{19}: A,B,A,A,B \)

Learning HMM parameters: \( x^{20}: B,B,B,A,A \)

Learning HMM parameters: \( x^{21}: A,A,B,A,B \)

Learning HMM parameters: \( x^{22}: A,B,A,A,B \)

Learning HMM parameters: \( x^{23}: B,B,B,A,A \)

Learning HMM parameters: \( x^{24}: A,A,B,A,B \)
Challenges in HMM learning

Learning parameters ($\pi, A, \phi$) with known states is not too hard

BUT usually states are unknown

If we had the parameters and the observations, we could figure out the states:

$$Viterbi \ P(Q^* | O) = \arg\max_Q P(Q | O)$$

Expectation-Maximization, or “EM”

Problem: Uncertain of $y^i$ (class), uncertain of $\theta^i$ (parameters)

Solution: Guess $y^i$, deduce $\theta^i$, re-compute $y^i$, re-compute $\theta^i$ ... etc.

OR: Guess $\theta^i$, deduce $y^i$, re-compute $\theta^i$, re-compute $y^i$

Will converge to a solution

E step: Fill in expected values for missing labels $y$

M step: Regular MLE for $\theta$ given known and filled-in variables

Also useful when there are holes in your data

Computing states $q_t$

Instead of picking one state: $q_t=s_p$ find $P(q_t=s_p|o)$

$$P(q_t = s_i | o_1, ..., o_T) = \frac{\alpha_t(i) \beta_t(i)}{\sum_j \alpha_t(j) \beta_t(j)}$$

Forward probability: $\alpha_t(i) = P(o_1 ... o_t \land q_t = s_i)$

Backward probability: $\beta_t(i) = P(o_{t+1} ... o_T | q_t = s_i)$

Details of forward probability

Forward probability: $\alpha_t(i) = P(o_1 ... o_t \land q_t = s_i)$

$$\alpha_1(i) = \phi_{o_1} \pi_i = P(o_1 | q_1 = s_i) P(q_1 = s_i)$$

$$\alpha_t(i) = \phi_{o_t} \sum_j A_{i,j} \alpha_{t-1}(j)$$

$$\alpha_t(i) = P(o_t | q_t = s_i) \sum_j P(q_t = s_i | q_{t-1} = s_j) \alpha_{t-1}(j)$$

Note: We presume we know number of possible class labels $y$ (or states $q$), we just don’t know which state occurs at which time
Details of backward probability

Backward probability:

\[ \beta_t(i) = P(o_{t+1} \ldots o_T | q_t = s_i) \]

\[ \beta_t(i) = \sum_j P(q_{t+1} = s_j | q_t = s_i) P(o_{t+1} | q_{t+1} = s_j) \beta_{t+1}(j) \]

Final \( \beta \):

\[ \beta_{T-1}(i) = \sum_j A_{j,i} \phi_{o_{T-1},j} \]

\[ = P(q_T = s_j | q_T = s_i) P(o_T | q_T = s_j) \]

E-step: State probabilities

One state:

\[ P(q_t = s_i | o_1, \ldots, o_T) = \frac{\alpha_t(i) \beta_t(i)}{\sum_j \alpha_t(j) \beta_t(j)} = S_t(i) \]

Two states in a row:

\[ P(q_t = s_j, q_{t+1} = s_i | o_1, \ldots, o_T) = \frac{\alpha_t(j) A_{i,j} \phi_{o_{t+1},i} \beta_{t+1}(i)}{\sum_f \sum_g \alpha_t(g) A_{f,g} \phi_{o_{t+1},f} \beta_{t+1}(f)} = S_t(i, j) \]

Recall: when states known

\[ \pi_A = \frac{\#D(q_1 = s_A)}{\#D} \]

\[ A_{i,j} = \frac{\#D(q_t = s_i, q_{t-1} = s_j)}{\#D} \]

\[ \phi_{i,j} = \frac{\#D(o_t = i)}{\#D(q_t = s_j)} \]

M-step

\[ A_{i,j} = \frac{\sum_t S_t(i,j)}{\sum_t S_t(j)} \]

\[ \phi_{i,j} = \frac{\sum_t |o_t = i \land q_t = s_j|}{\sum_t |q_t = s_j|} \]

\[ \pi_i = S_1(i) \]

Known states:

* \( \pi_A = \frac{\#D(q_1 = s_A)}{\#D} \)

* \( A_{i,j} = \frac{\#D(q_t = s_i, q_{t-1} = s_j)}{\#D(q_{t-1} = s_j)} \)

* \( \phi_{i,j} = \frac{\#D(o_t = i \land q_t = s_j)}{\#D(q_t = s_j)} \)
Review of HMMs in action

For classification, find highest probability class given features

Features for one sound:
* $[q_1, o_1, q_2, o_2, ..., q_T, o_T]$

Conclude word:

Generates states: