MATH 285: Stochastic Processes

math-old.ucsd.edu/~ynemish/teaching/285

Today: HMM. Viterbi algorithm

- Homework 4 is due on Friday, February 11, 11:59 PM

Hidden Markov Model

(Y_n) is a MC on S and transition probabilities $p(i,j)$

(X_n) is a stochastic process (non necessarily Markov) with state space $\mathbb{R}$ and $\mathbb{P}[X_n = x \mid Y_n = y] = e_y(x)$

$Z_n = (X_n, Y_n)$ is a MC with transition probabilities

$$\mathbb{P}[Z_{n+1} = (x', y') \mid Z_n = (x, y)] = p(y, y')\, e_{y'}(x')$$

- $x = (x_0, x_1, \dots, x_N)$ the observed sequence
- $y = (y_0, y_1, \dots, y_N)$ the state sequence
- $\mathbb{P}[x] = \mathbb{P}[X_0 = x_0, \dots, X_N = x_N]$
- $\mathbb{P}[x, y] = \mathbb{P}[X_0 = x_0, \dots, X_N = x_N, Y_0 = y_0, \dots, Y_N = y_N]$

Q: What is the probability that the hidden states are $(y_0, y_1, \dots, y_N)$ given that we observe $(x_0, x_1, \dots, x_N)$?

The forward algorithm

$$\mathbb{P}[y \mid x] = \frac{\mathbb{P}[x, y]}{\mathbb{P}[x]}$$ How to efficiently compute $\mathbb{P}[x]$?

- Initialization:

  For $y \in S$, set $\alpha_0(y) = \mathbb{P}[X_0 = y_0, Y_0 = y_0] = \mathbb{P}[Y_0 = y_0]\, e_{y_0}(x_0)$

- Recursion:

  For $y' \in S$ and $0 < n < N$ set $\alpha_{n+1}(y') = e_{y'}(x_{n+1}) \sum_{y \in S} \alpha_n(y)\, p(y, y')$

- Termination:

  $$\mathbb{P}[x] = \sum_{y \in S} \alpha_N(y)$$

  Requires $O(N|S|^2)$ operations

Q: Given x, find y that maximizes $\mathbb{P}[y \mid x]$.

Most likely trajectory

Motivation: signal processing, speach recognition,
error correcting codes

Y_n – signal (uncontaminated)

X_n – signal with random noise

Receive the sequence $(x_0, x_1, \dots, x_N)$

What is the best guess for the values of $(y_0, y_1, \dots, y_N)$?

Mathematically: compute , so that

Computational complexity

Direct calculation:

- $\mathbb{P}[y|x]$ for fixed y
- Repeat for all $y \in S^N$
- Select the maximizer

In total operations, grows

Viterbi algorithm:

- Recursive algorith that allows to
- Complexity grows
- $\max\limits_{y} \mathbb{P}[y|x] =$
- Define $V_n(y) :=$

Viterbi algorithm

$$V_n(y) := \max_{y_0,\dots,y_{n-1}\in S} \mathbb{P}[X_0=x_0,\dots,X_n=x_n,\ Y_0=y_0,\dots,Y_{n-1}=y_{n-1},\ Y_n=y]$$

Then $\max\limits_{Y} \mathbb{P}[X,Y] =$

Idea
- compute $\max\limits_{Y} \mathbb{P}[X,Y]$ recursively
- backtrack to find the maximizing sequence

$$\mathbb{P}[X_0=x_0,\dots,X_n=x_n,X_{n+1}=x_{n+1},Y_0=y_0,\dots,Y_{n-1}=y_{n-1},Y_n=y,Y_{n+1}=y']$$

$=$

$=$

$=$

Viterbi algorithm

$$V_{n+1}(y') = \max_{y_0,\dots,y} \mathbb{P}[X_0 = x_0, \dots, X_{n+1} = x_{n+1}, Y_0 = y_0, \dots, Y_n = y, Y_{n+1} = y_{n+1}]$$

$$=$$

$$=$$

$$=$$

This allows to compute $\max_{y} \mathbb{P}[x, y]$ recursively

- $V_0(y) =$
- $V_{n+1}(y') =$
- $\max_{y} \mathbb{P}[x, y] =$

Viterbi algorithm

Backtracking: keep track of the element that maximizes $p(y, y') V_n(y)$:

- for $y, y' \in S$ define $W_{n+1}(y, y') :=$
- for all $y' \in S$ find y that maximizes $W_{n+1}(y, y')$

  $\varphi_n^*(y') :=$
- in particular $V_{n+1}(y') =$
- set
- then $\max_y \mathbb{P}[x, y] =$

  $=$

Viterbi algorithm

Initialization:

For $y \in S$, set $V_0(y) = \mathbb{P}[X_0 = x_0, Y_0 = y_0] = \mathbb{P}[Y_0 = y_0] e_{y_0}(x_0)$

Recursion:

For $y, y' \in S$ and $0 < n \leq N$ set $W_{n+1}(y, y') = V_n(y) p(y, y') e_{y'}(x_{n+1})$

Then compute $\psi_n^*(y') = \underset{y}{\operatorname{argmax}}\, W_{n+1}(y, y')$

Set $V_{n+1}(y') = W_{n+1}(\psi_n^*(y'), y')$

Termination:

$\max_{Y} \mathbb{P}[X, Y] = \max_{y} V_N(y)$, define $y_N^* = \underset{y}{\operatorname{argmax}}\, V_N(y)$

Backtracking

For $0 \leq k < N$, set $y_k^* = \psi_k^*(y_{k+1}^*)$