

Time Warping Hidden Markov Models



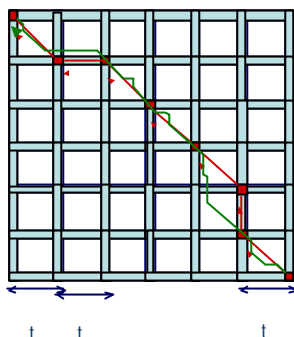
Lecture 2, Thursday April 3, 2003

Review of Last Lecture



Lecture 2, Thursday April 3, 2003

The Four-Russian Algorithm



Lecture 4, Thursday April 10, 2003

BLAST — Original Version

Dictionary:

All words of length k (~ 11)

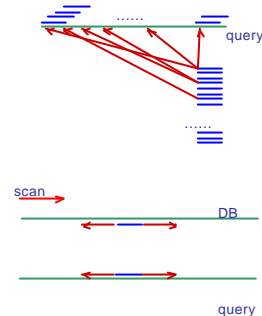
Alignment initiated between words of alignment score $\geq T$ (typically $T = k$)

Alignment:

Ungapped extensions until score below statistical threshold

Output:

All local alignments with score $>$ statistical threshold

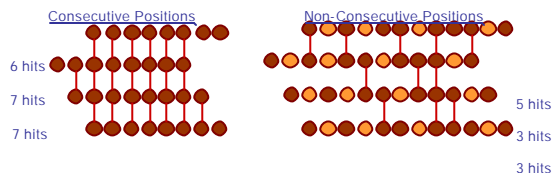


Lecture 4, Thursday April 10, 2003

PatternHunter

Main features:

- Non-consecutive position words
- Highly optimized



On a 70% conserved region:

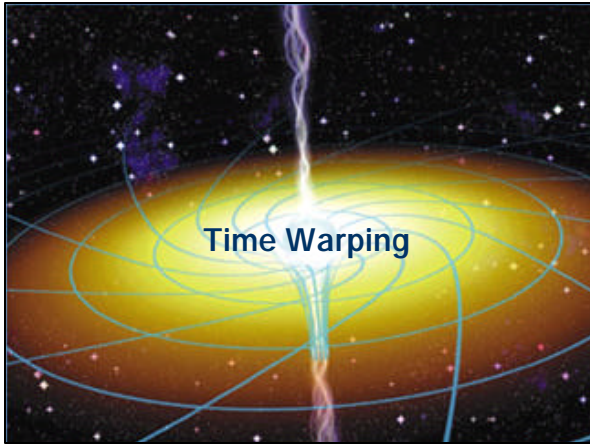
| | Consecutive | Non-consecutive |
|-------------------------|-------------|-----------------|
| Expected # hits: | 1.07 | 0.97 |
| Prob[at least one hit]: | 0.30 | 0.47 |

Lecture 4, Thursday April 10, 2003

Today

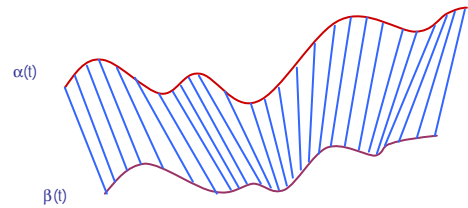
- Time Warping
- Hidden Markov models

Lecture 4, Thursday April 10, 2003



Time Warping

Align and compare two trajectories in multi-D space



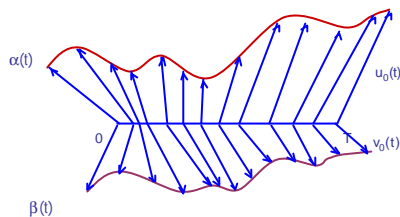
- Additive random error
- Variations in speed from one segment to another

Lecture 4, Thursday April 10, 2003

Time Warping

Definition: $\alpha(u)$, $\beta(v)$ are connected by an approximate continuous time warping (u_0, v_0) , if:

u_0, v_0 are strictly increasing functions on $[0, T]$, and
 $\alpha(u_0(t)) \equiv \beta(v_0(t))$ for $0 \leq t \leq T$



Lecture 4, Thursday April 10, 2003

Time Warping

How do we measure how "good" a time warping is?

Let's try:

$$\int_0^T w(\alpha(u_0(t)), \beta(v_0(t))) dt$$

However, an equivalent time warping $(u_1(s), v_1(s))$, is given by:

$$s = f(t); \quad f: [0, T] \rightarrow [0, S]$$

has score

$$\int_0^S w(\alpha(u_1(s)), \beta(v_1(s))) ds = \int_0^T w(\alpha(u_0(t)), \beta(v_0(t))) f'(t) dt$$

This is arbitrarily different

Lecture 4, Thursday April 10, 2003

Time Warping

This one works:

$$d(u_0, v_0) = \int_0^T w(\alpha(u_0(t)), \beta(v_0(t))) [(u_0'(t) + v_0'(t))/2] dt$$

Now, if $s = f(t)$; $t = g(s)$, and $g = f^{-1}$,

$$\int_0^S w(\alpha(u_1(s)), \beta(v_1(s))) (u_1'(s) + v_1'(s))/2 ds =$$

$$f'(t) = f'(g(s)) = s;$$

$$f'(t) = f'(g(s)) g'(s) = 1, \text{ therefore } g'(s) = 1/f'(t)$$

$$u_0(t) = u_0(g(s)), \text{ therefore } u_0(t) = u_0(g(s)) g'(s)$$

$$\int_0^T w(\alpha(u_0(t)), \beta(v_0(t))) (u_0'(t) + v_0'(t))/2 g'(s) f'(t) dt =$$

$$\int_0^T w(\alpha(u_0(t)), \beta(v_0(t))) [(u_0'(t) + v_0'(t))/2] dt$$

Lecture 4, Thursday April 10, 2003

Time Warping

From continuous to discrete:

Let's **discretize** the signals:

$$\alpha(t): \quad a = a_0, \dots, a_M$$

$$\beta(t): \quad b = b_0, \dots, b_N$$

Definition:

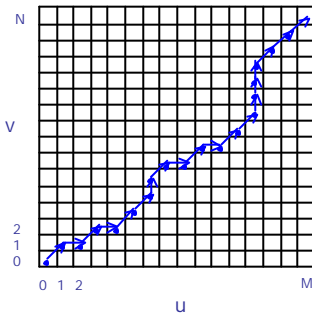
a, b are connected by an approximate discrete time warping (u, v) , if u and v are weakly increasing integer functions on $1 \leq h \leq H$, such that

$$a_{u[h]} \equiv b_{v[h]} \text{ for all } h = 1, \dots, H$$

$$\text{Moreover, we require } \begin{aligned} u[0] &= v[0] = 0; \\ u[H] &= M; \\ v[H] &= N \end{aligned}$$

Lecture 4, Thursday April 10, 2003

Time Warping



Define possible steps:
 $(\Delta u, \Delta v)$ is the possible difference of u and v
 between steps $h-1$ and h
 $(\Delta u, \Delta v) = \begin{cases} (1, 0) \\ (1, 1) \\ (0, 1) \end{cases}$

Lecture 4, Thursday April 10, 2003

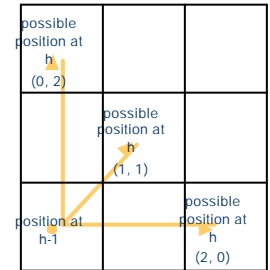
Time Warping

Alternatively:

$$(\Delta u, \Delta v) = \begin{cases} (2, 0) \\ (1, 1) \\ (0, 2) \end{cases}$$

Advantage:

Every time warp has the same number of steps



Lecture 4, Thursday April 10, 2003

Time Warping

Discrete objective function:

For $0 \leq i = u[h] \leq M$; $0 \leq j = v[h] \leq N$,
 Define $w(i, j) = w(a_{u[h]}, b_{v[h]})$

Then,

$$D(u, v) = \sum_h w(u[h], v[h]) (\Delta u + \Delta v) / 2$$

In the case where we allow $(2, 0)$, $(1, 1)$, and $(0, 2)$ steps,

$$D(u, v) = \sum_h w(u[h], v[h])$$

Lecture 4, Thursday April 10, 2003

Time Warping

Algorithm for optimal discrete time warping:

Initialization:

$$D(i, 0) = \frac{1}{2} \sum_{j=1}^i w(i, 0)$$

$$D(0, j) = \frac{1}{2} \sum_{i=1}^j w(0, j)$$

$$D(1, j) = D(i, 1) = w(i, j) + w(i-1, j-1)$$

Iteration:

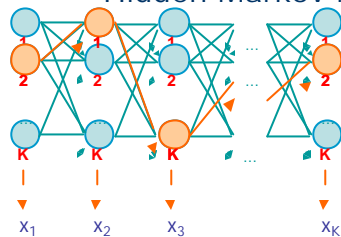
For $i = 2, \dots, M$

For $j = 2, \dots, N$

$$D(i, j) = \min \begin{cases} D(i-2, j) + w(i, j) \\ D(i-1, j-1) + w(i, j) \\ D(i-2, j) + w(i, j) \end{cases}$$

Lecture 4, Thursday April 10, 2003

Hidden Markov Models



Outline for our next topic

- Hidden Markov models – the theory
- Probabilistic interpretation of alignments using HMMs

Later in the course:

- Applications of HMMs to biological sequence modeling and discovery of features such as genes

Lecture 4, Thursday April 10, 2003

Example: The Dishonest Casino

A casino has two dice:

- Fair die:
 $P(1) = P(2) = P(3) = P(5) = P(6) = 1/6$
- Loaded die:
 $P(1) = P(2) = P(3) = P(5) = 1/10$
 $P(6) = 1/2$

Casino player switches back-&-forth between fair and loaded die once every 20 turns

Game:

- You bet \$1
- You roll (always with a fair die)
- Casino player rolls (maybe with fair die, maybe with loaded die)
- Highest number wins \$2



Lecture 4, Thursday April 10, 2003

Question # 1 – Evaluation

GIVEN

A sequence of rolls by the casino player

1245526462146146136136661664661636616366163616515615115146123562344

QUESTION

How likely is this sequence, given our model of how the casino works?

This is the **EVALUATION** problem in HMMs

Lecture 4, Thursday April 10, 2003

Question # 2 – Decoding

GIVEN

A sequence of rolls by the casino player

1245526462146146136136661664661636616366163616515615115146123562344

QUESTION

What portion of the sequence was generated with the fair die, and what portion with the loaded die?

This is the **DECODING** question in HMMs

Lecture 4, Thursday April 10, 2003

Question # 3 – Learning

GIVEN

A sequence of rolls by the casino player

1245526462146146136136661664661636616366163616515615115146123562344

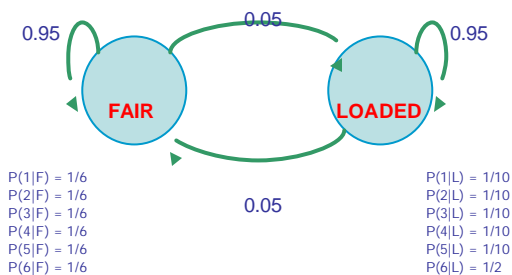
QUESTION

How "loaded" is the loaded die? How "fair" is the fair die? How often does the casino player change from fair to loaded, and back?

This is the **LEARNING** question in HMMs

Lecture 4, Thursday April 10, 2003

The dishonest casino model



Lecture 4, Thursday April 10, 2003

Definition of a hidden Markov model

Definition: A hidden Markov model (HMM)

- Alphabet $\Sigma = \{b_1, b_2, \dots, b_M\}$
- Set of states $Q = \{1, \dots, K\}$
- Transition probabilities between any two states

a_{ij} = transition prob from state i to state j

$a_{i1} + \dots + a_{iK} = 1$, for all states $i = 1 \dots K$

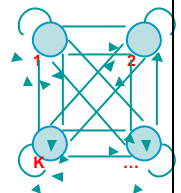
- Start probabilities a_{0i}

$a_{01} + \dots + a_{0K} = 1$

- Emission probabilities within each state

$e_i(b) = P(x_i = b \mid \pi_i = k)$

$e_i(b_1) + \dots + e_i(b_M) = 1$, for all states $i = 1 \dots K$

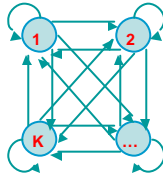


Lecture 4, Thursday April 10, 2003

A Hidden Markov Model is memory-less

At each time step t ,
the only thing that affects future states
is the current state π_t

$$\begin{aligned} P(\pi_{t+1} = k \mid \text{"whatever happened so far"}) &= \\ P(\pi_{t+1} = k \mid \pi_1, \pi_2, \dots, \pi_t, x_1, x_2, \dots, x_t) &= \\ P(\pi_{t+1} = k \mid \pi_t) \end{aligned}$$

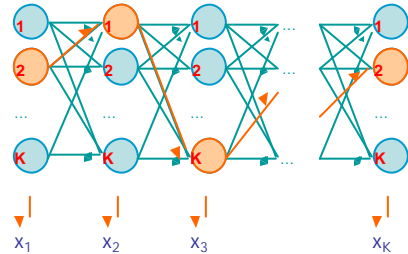


Lecture 4, Thursday April 10, 2003

A parse of a sequence

Given a sequence $x = x_1, \dots, x_N$,

A parse of x is a sequence of states $\pi = \pi_1, \dots, \pi_N$



Lecture 4, Thursday April 10, 2003

Likelihood of a parse

Given a sequence $x = x_1, \dots, x_N$
and a parse $\pi = \pi_1, \dots, \pi_N$,

To find how likely is the parse:
(given our HMM)

$$\begin{aligned} P(x, \pi) &= P(x_1, \dots, x_N, \pi_1, \dots, \pi_N) = \\ &P(x_N, \pi_N \mid \pi_{N-1}) P(x_{N-1}, \pi_{N-1} \mid \pi_{N-2}) \dots P(x_2, \pi_2 \mid \pi_1) P(x_1, \pi_1) = \\ &P(x_N \mid \pi_N) P(\pi_N \mid \pi_{N-1}) \dots P(x_2 \mid \pi_2) P(\pi_2 \mid \pi_1) P(x_1 \mid \pi_1) P(\pi_1) = \\ &a_{\pi_1} a_{x_1 \mid \pi_1} a_{\pi_2 \mid \pi_1} \dots a_{x_N \mid \pi_N} a_{\pi_N \mid \pi_{N-1}} e_{\pi_1}(x_1) \dots e_{\pi_N}(x_N) \end{aligned}$$

Lecture 4, Thursday April 10, 2003

Example: the dishonest casino

Let the sequence of rolls be:

$x = 1, 2, 1, 5, 6, 2, 1, 6, 2, 4$

Then, what is the likelihood of

$\pi = \text{Fair, Fair, Fair, Fair, Fair, Fair, Fair, Fair, Fair, Fair?}$

(say initial probs $a_{\text{Fair}} = 1/2$, $a_{\text{Loaded}} = 1/2$)

$$1/2 \times P(1 \mid \text{Fair}) P(\text{Fair} \mid \text{Fair}) P(2 \mid \text{Fair}) P(\text{Fair} \mid \text{Fair}) \dots P(4 \mid \text{Fair}) =$$

$$1/2 \times (1/6)^{10} \times (0.95)^9 = .00000000521158647211 = 0.5 \times 10^{-9}$$

Lecture 4, Thursday April 10, 2003



Example: the dishonest casino

So, the likelihood the die is fair in all this run
is just 0.521×10^{-9}

OK, but what is the likelihood of

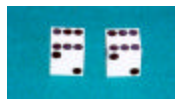
$\pi = \text{Loaded, Loaded, Loaded, Loaded, Loaded, Loaded, Loaded, Loaded, Loaded, Loaded?}$

$$1/2 \times P(1 \mid \text{Loaded}) P(\text{Loaded} \mid \text{Loaded}) \dots P(4 \mid \text{Loaded}) =$$

$$1/2 \times (1/10)^8 \times (1/2)^2 (0.95)^9 = .00000000078781176215 = 7.9 \times 10^{-10}$$

Therefore, it is after all 6.59 times more likely that the die is fair all the way,
than that it is loaded all the way.

Lecture 4, Thursday April 10, 2003



Example: the dishonest casino

Let the sequence of rolls be:

$x = 1, 6, 6, 5, 6, 2, 6, 6, 3, 6$

Now, what is the likelihood $\pi = F, F, \dots, F?$

$$1/2 \times (1/6)^{10} \times (0.95)^9 = 0.5 \times 10^{-9}, \text{ same as before}$$

What is the likelihood

$\pi = L, L, \dots, L?$

$$1/2 \times (1/10)^4 \times (1/2)^6 (0.95)^9 = .00000049238235134735 = 0.5 \times 10^{-7}$$

So, it is 100 times more likely the die is loaded

Lecture 4, Thursday April 10, 2003



The three main questions on HMMs

1. Evaluation

GIVEN a HMM M , and a sequence x ,
FIND $\text{Prob}[x | M]$

2. Decoding

GIVEN a HMM M , and a sequence x ,
FIND the sequence π of states that maximizes $P[x, \pi | M]$

3. Learning

GIVEN a HMM M , with unspecified transition/emission probs.,
and a sequence x ,

FIND parameters $\theta = (e_i(\cdot), a_{ij})$ that maximize $P[x | \theta]$

Lecture 4, Thursday April 10, 2003

Let's not be confused by notation

$P[x | M]$: The probability that sequence x was generated by the model

The model is: architecture (#states, etc)
+ parameters $\theta = a_{ij}, e_i(\cdot)$

So, $P[x | \theta]$, and $P[x]$ are the same, when the architecture, and the entire model, respectively, are implied

Similarly, $P[x, \pi | M]$ and $P[x, \pi]$ are the same

In the **LEARNING** problem we always write $P[x | \theta]$ to emphasize that we are seeking the θ that maximizes $P[x | \theta]$

Lecture 4, Thursday April 10, 2003