

Bioinformatics II

Hidden Markov Models

Dr. Konrad Schlude
schlude@inf.ethz.ch

Institute of Theoretical Computer Science
ETH Zürich

Contents

1. Introduction
2. Hidden Markov Model definition
3. Most Probable Path: Viterbi algorithm
4. Probability of a Sequence: Forward algorithm
5. HMM parameter estimation
6. HMM model structure
7. Conclusion

Teacher

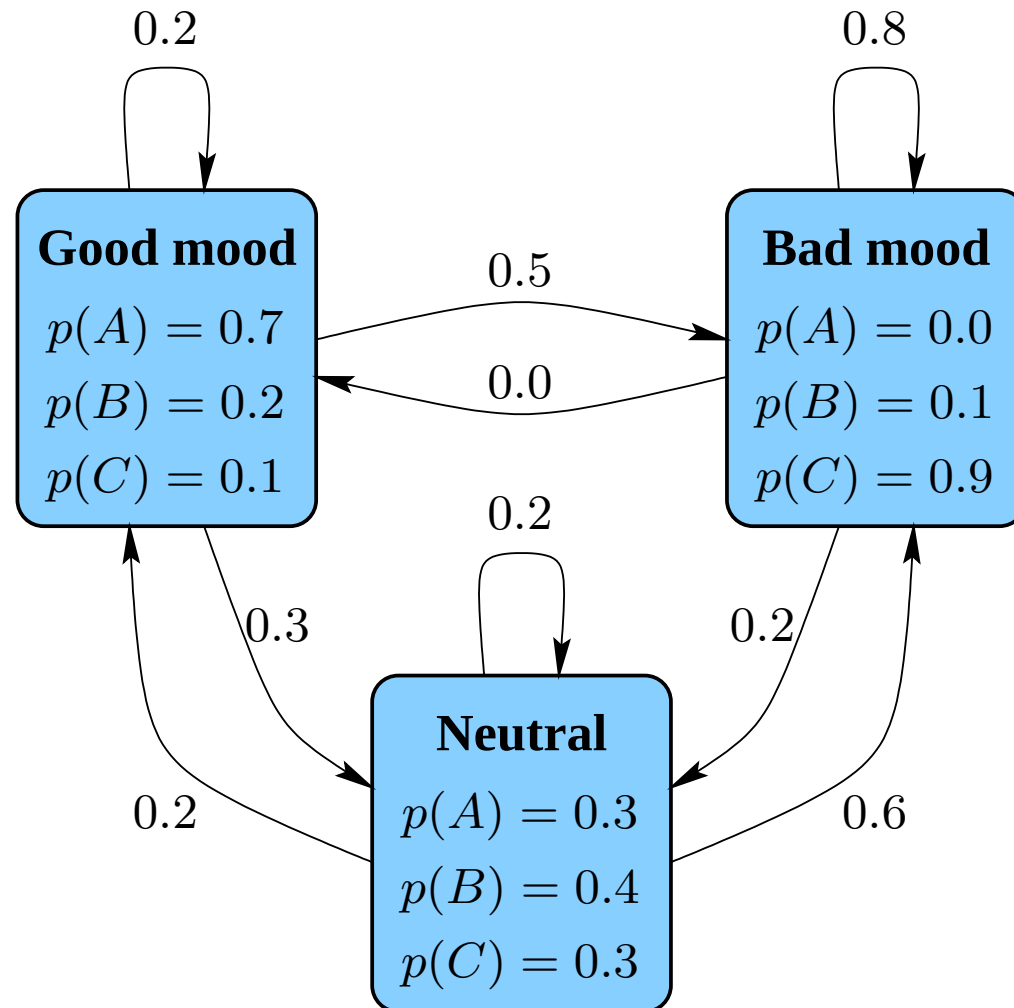
Situation:

- Your school teacher gave three different types of daily homework assignments:
 - A: took about 5 minutes to complete
 - B: took about 1 hour to complete
 - C: took about 3 hours to complete
- Your teacher did not reveal openly his mood to you daily, but you knew that your teacher was either in a bad, neutral, or a good mood for a whole day.
- Mood changes occurred only overnight.

Question:

- How were his moods related to the homework type assigned that day?

Teacher



Questions Teacher

One week, your teacher gave the following homework assignments:

Monday: A

Tuesday: C

Wednesday: B

Thursday: A

Friday: C

Questions:

- What did his mood curve look like most likely that week?
- What is the probability that he would assign this order of homework assignments?
- What is the probability that he was in a good mood on Thursday?

Contents

1. Introduction
2. Hidden Markov Model definition
3. Most Probable Path: Viterbi algorithm
4. Probability of a Sequence: Forward algorithm
5. HMM parameter estimation
6. HMM model structure
7. Conclusion

Hidden Markov Model

Definition

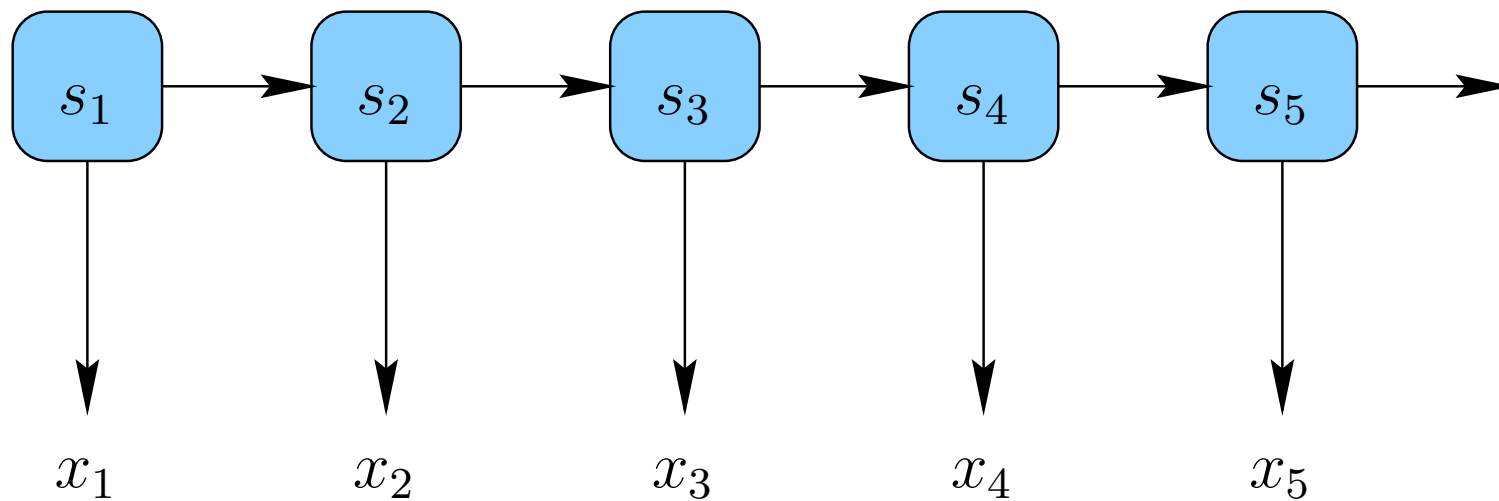
- States $S = \{s_1, s_2, \dots, s_m\}$
- a_{kl} = probability that state l follows state k (Markov chain)
- Alphabet Σ
- $e_k(x)$ = probability that state k emits symbol $x \in \Sigma$

Teacher

- $S = \{\text{Good mood, Bad mood, Neutral}\}$
- a_{kl}
- $\Sigma = \{A, B, C\}$
- $e_k(x)$

Hidden Markov Model

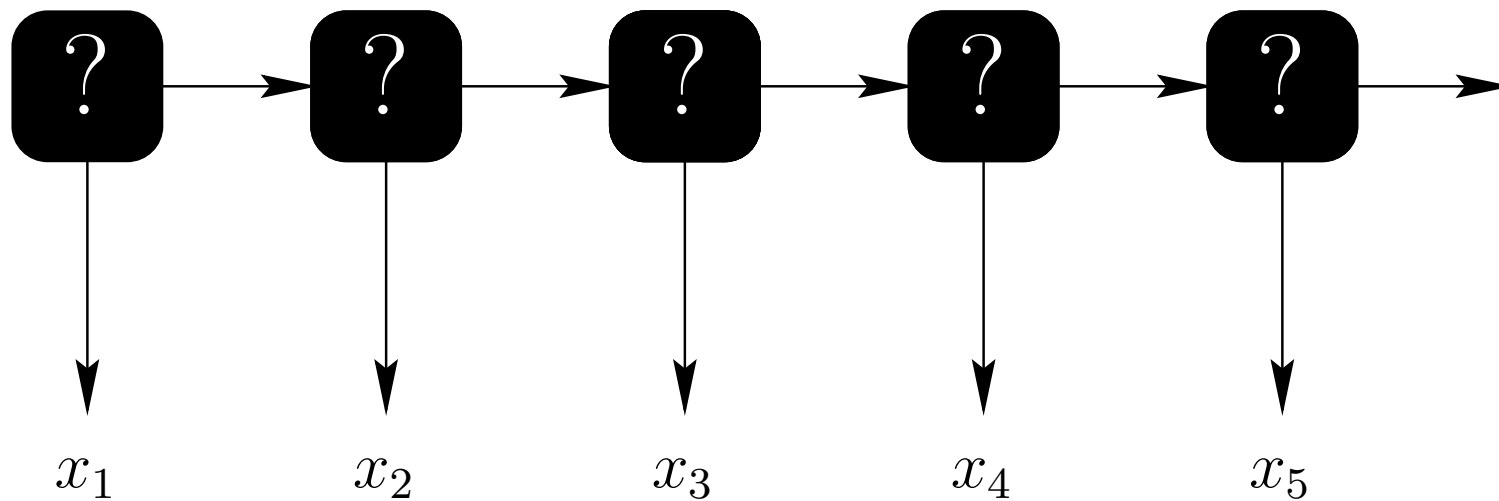
Markov chain



Emitted (observed) symbols

Hidden Markov Model

Markov chain is hidden



Emitted (observed) symbols

Sequence Analysis

- CpG islands in human genome:
CpG dinucleotides are rarer in the genome than could be expected from the independent probabilities of C and G, but CpG dinucleotides occur at a higher rate around promoter regions of genes than elsewhere. These regions are called *CpG islands*.
Given a long sequence, how can we find the CpG islands in it?
- Speech Recognition:
blumentopferde $\rightsquigarrow$ Blu-men-to-pfer-de

Contents

1. Introduction
2. Hidden Markov Model definition
3. **Most Probable Path: Viterbi algorithm**
4. Probability of a Sequence: Forward algorithm
5. HMM parameter estimation
6. HMM model structure
7. Conclusion

Most Probable Path

Given:

- Hidden Markov model: $S, a_{kl}, \Sigma, e_l(x)$.
- Observed symbol sequence $E = x_1, \dots, x_n$.

Most probable path of states that resulted in symbol sequence E .

Let $v_k(i)$ be the probability of the most probable path of the symbol sequence $x_1, x_2, \dots, x_i$ ending in state k . Then:

$$\begin{aligned} v_l(i+1) &= \max_k (e_l(x_{i+1}) * v_k(i) * a_{kl}) \\ &= e_l(x_{i+1}) * \max_k (v_k(i) * a_{kl}) \end{aligned}$$

Viterbi Algorithm

Matrix $v_k(i)$, where $k \in S$ and $1 \leq i \leq n$.

- Initialization: $v_k(1) = e_k(x_1)/\#states$ for all states $k \in S$.
- $v_l(i) = e_l(x_i) \max_k (v_k(i-1)a_{kl})$ for all states $k \in S$, $i \geq 2$.

Algorithm

- Iteratively build up matrix $v_k(i)$.
- Store pointers to chosen path.
- Probability of most probable path in maximum entry in last column.
- Reconstruct path along pointers.

Viterbi Algorithm / Teacher

Empty table

	A	C	B	A	C
good					
neutral					
bad					

Viterbi Algorithm / Teacher

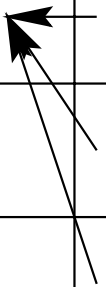
Initialization: $v_l(1) = e_l(A)/\#states$

	A	C	B	A	C
good	0.23				
neutral	0.1				
bad	0.0				

Viterbi Algorithm / Teacher

$$v_l(2) = e_l(C) \max_k (v_k(1) a_{kl})$$

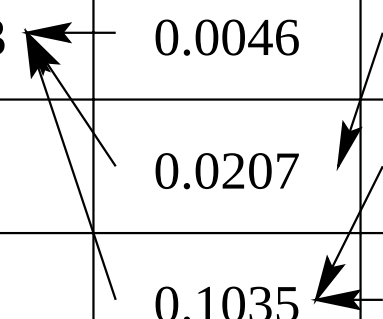
	A	C	B	A	C
good	0.23	0.0046			
neutral	0.1	0.0207			
bad	0.0	0.1035			



Viterbi Algorithm / Teacher

$$v_l(3) = e_l(B) \max_k (v_k(2) a_{kl})$$

	A	C	B	A	C
good	0.23	0.0046	.000828		
neutral	0.1	0.0207	.00828		
bad	0.0	0.1035	.00828		



Viterbi Algorithm / Teacher

$$v_l(4) = e_l(A) \max_k (v_k(3) a_{kl})$$

	A	C	B	A	C
good	0.23	0.0046	.000828	.0011592	
neutral	0.1	0.0207	.00828	.006624	
bad	0.0	0.1035	.00828	0	

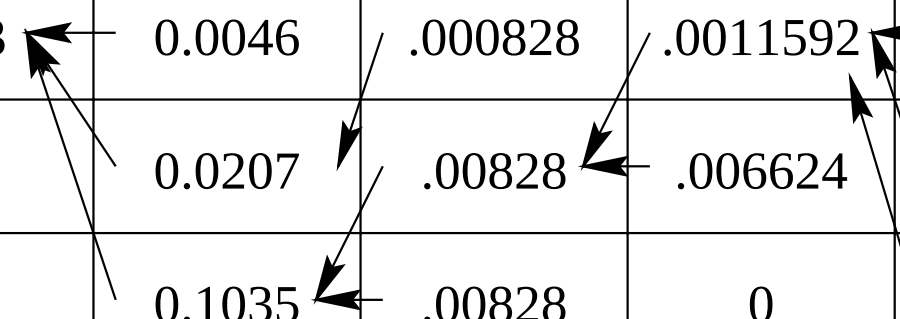
Diagram illustrating the Viterbi algorithm's backpointers for the sequence "good", "neutral", "bad". The arrows indicate the path taken to reach the current state:

- From "good" to "A" (0.23)
- From "neutral" to "B" (.00828)
- From "bad" to "C" (0.1035)

Viterbi Algorithm / Teacher

$$v_l(5) = e_l(C) \max_k (v_k(4) a_{kl})$$

	A	C	B	A	C
good	0.23	0.0046	.000828	.0011592	.0000231
neutral	0.1	0.0207	.00828	.006624	.0001043
bad	0.0	0.1035	.00828	0	.0005216



Viterbi Algorithm / Teacher

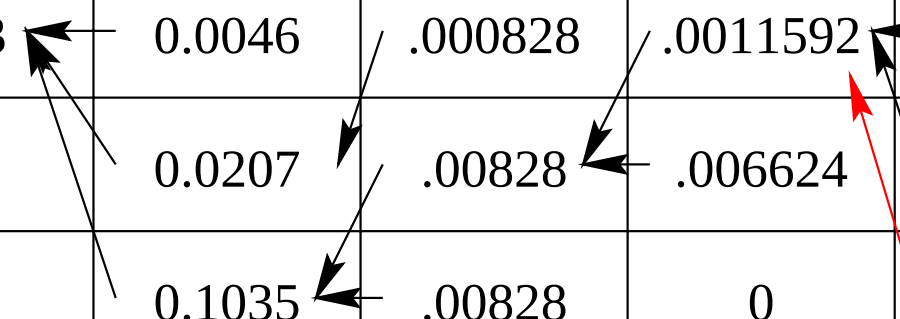
Maximum entry in last column

	A	C	B	A	C
good	0.23	0.0046	.000828	.0011592	.0000231
neutral	0.1	0.0207	.00828	.006624	.0001043
bad	0.0	0.1035	.00828	0	.0005216

Viterbi Algorithm / Teacher

Reconstruct path along pointers

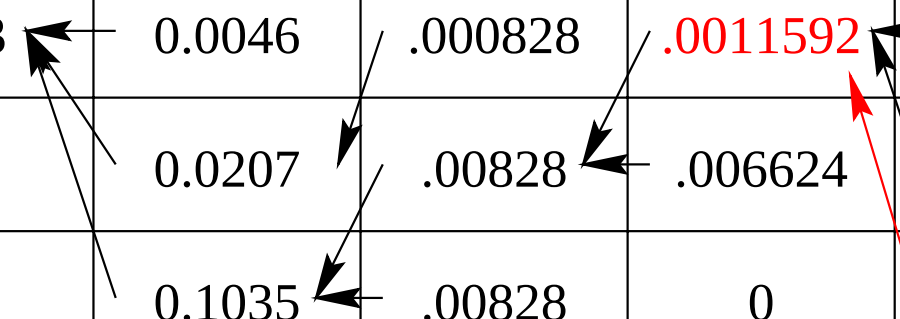
	A	C	B	A	C
good	0.23	0.0046	.000828	.0011592	.0000231
neutral	0.1	0.0207	.00828	.006624	.0001043
bad	0.0	0.1035	.00828	0	.0005216



Viterbi Algorithm / Teacher

Reconstruct path along pointers

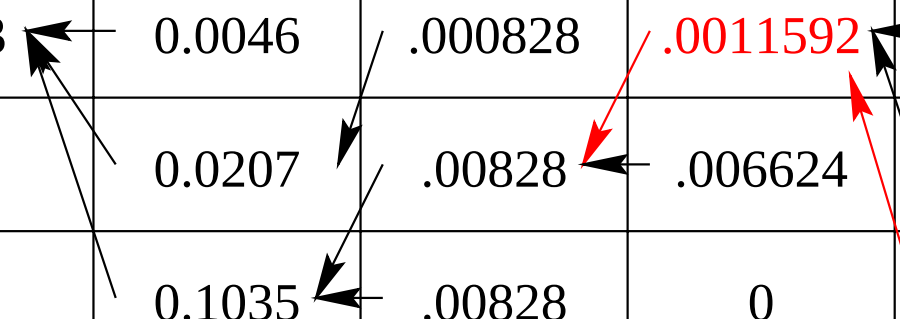
	A	C	B	A	C
good	0.23	0.0046	.000828	.0011592	.0000231
neutral	0.1	0.0207	.00828	.006624	.0001043
bad	0.0	0.1035	.00828	0	.0005216



Viterbi Algorithm / Teacher

Reconstruct path along pointers

	A	C	B	A	C
good	0.23	0.0046	.000828	.0011592	.0000231
neutral	0.1	0.0207	.00828	.006624	.0001043
bad	0.0	0.1035	.00828	0	.0005216



Viterbi Algorithm / Teacher

Reconstruct path along pointers

	A	C	B	A	C
good	0.23	0.0046	.000828	.0011592	.0000231
neutral	0.1	0.0207	.00828	.006624	.0001043
bad	0.0	0.1035	.00828	0	.0005216

The diagram illustrates the reconstruction of the Viterbi path. Black arrows show the path from the final state (good) back to the initial state (good), passing through neutral and bad. Red arrows highlight the path from the final state (good) back to the initial state (good), passing through neutral and bad, which is the path with the highest probability.

Viterbi Algorithm / Teacher

Reconstruct path along pointers

	A	C	B	A	C
good	0.23	0.0046	.000828	.0011592	.0000231
neutral	0.1	0.0207	.00828	.006624	.0001043
bad	0.0	0.1035	.00828	0	.0005216

The diagram illustrates the reconstruction of the Viterbi path. Black arrows indicate the path from the final state (good) back to the initial state (good), following the highest probability path. Red arrows indicate the path from the final state (good) back to the initial state (good), following a different path. The path highlighted in red is the one that ends at the 'bad' state.

Viterbi Algorithm / Teacher

Reconstruct path along pointers

	A	C	B	A	C
good	0.23	0.0046	.000828	.0011592	.0000231
neutral	0.1	0.0207	.00828	.006624	.0001043
bad	0.0	0.1035	.00828	0	.0005216

The diagram illustrates the reconstruction of the Viterbi path. Black arrows show the path from the 'good' state in the second column (0.23) to the 'neutral' state in the first column (0.1), and from the 'neutral' state in the second column (0.0207) to the 'bad' state in the first column (0.0). Red arrows show the path from the 'neutral' state in the third column (.00828) to the 'good' state in the fourth column (.0011592), and from the 'bad' state in the third column (0.1035) to the 'good' state in the fourth column (.0011592). The values .0011592, .00828, 0.1035, and .0005216 are highlighted in red in the original image.

Viterbi Algorithm / Teacher

Reconstruct path along pointers

	A	C	B	A	C
good	0.23	0.0046	.000828	.0011592	.0000231
neutral	0.1	0.0207	.00828	.006624	.0001043
bad	0.0	0.1035	.00828	0	.0005216

The diagram illustrates the reconstruction of the Viterbi path. Red arrows indicate the sequence of states chosen at each step: from 'bad' to 'C' in the first step, then to 'A' in the second, then to 'B' in the third, then to 'A' in the fourth, and finally to 'C' in the fifth. Black arrows show the other possible transitions from each state.

Viterbi Algorithm / Teacher

Question:

- What did his mood curve look like most likely that week?

Answer:

- Most probable mood curve:

Day:	Mon	Tue	Wed	Thu	Fri
Assignment:	A	C	B	A	C
Mood:	good	bad	neutral	good	bad

Problem: Initialization

We said:

$$v_l(1) = e_l(A) / \# \text{states}$$

However, the probabilities of the states are different:

$$\Pr(\text{good}) \approx 0.05 \quad \Pr(\text{neutral}) \approx 0.2 \quad \Pr(\text{bad}) \approx 0.75$$

Alternative initialization:

$$v_l(1) = e_l(A) \Pr(l)$$

Viterbi Algorithm / Teacher

Initialization: $v_l(1) = e_l(A)\text{Pr}(l)$

	A	C	B	A	C
good	0.035				
neutral	0.06				
bad	0.0				

Viterbi Algorithm / Teacher

$$v_l(i+1) = e_l(x_{i+1}) \max_k (v_k(i) a_{kl})$$

	A	C	B	A	C
good	0.035	0.0012	.000144	.0003629	.0000073
neutral	0.06	0.0036	.002592	.0001556	.0000327
bad	0.0	0.0324	.002592	0	.0001633

Viterbi Algorithm / Teacher

Reconstruct path along pointers

	A	C	B	A	C
good	0.035	0.0012	.000144	.0003629	.0000073
neutral	0.06	0.0036	.002592	.0001556	.0000327
bad	0.0	0.0324	.002592	0	.0001633

The diagram illustrates the Viterbi algorithm's backtracking process. Red arrows trace the optimal path from the final state (good, A) back to the start (bad, A). Black arrows show the other possible transitions at each step.

Path reconstruction (red arrows):

- From (good, A) to (good, C)
- From (good, C) to (neutral, C)
- From (neutral, C) to (neutral, B)
- From (neutral, B) to (neutral, A)
- From (neutral, A) to (bad, A)

Other transitions (black arrows):

- From (good, A) to (good, B)
- From (good, C) to (good, A)
- From (neutral, C) to (neutral, A)
- From (neutral, B) to (neutral, C)
- From (neutral, A) to (neutral, C)
- From (bad, B) to (bad, A)

Different initializations = different results

Initialization: $v_l(1) = e_l(A)/\#\text{states}$

Day:	Mon	Tue	Wed	Thu	Fri
Assignment:	A	C	B	A	C
Mood:	good	bad	neutral	good	bad

Initialization: $v_l(1) = e_l(A)\text{Pr}(l)$

Day:	Mon	Tue	Wed	Thu	Fri
Assignment:	A	C	B	A	C
Mood:	neutral	bad	neutral	good	bad

Correct initialization depends on the problem.

Complexity

Number of paths: $|S|^n$

Viterbi algorithm:

- Space: $O(|S|n)$
- Time: $O(|S|^2n)$

Example: $|S| = 4, n = 100$

- $|S|n = 400$
- $|S|^2n = 1600$
- $|S|^n \approx 1.61 * 10^{60}$

Problem: Numerical Stability

The algorithms multiply many small probabilities. E.g., genomic sequences of 100000 bases, probability ≈ 0.1 : $10^{-100000}$.

- Underflow error
- Program crash
- Arbitrary wrong numbers

Solution:

$$\prod_i p_i \quad \text{versus} \quad \log \left(\prod_i p_i \right) = \sum_i (\log(p_i))$$

Contents

1. Introduction
2. Hidden Markov Model definition
3. Most Probable Path: Viterbi algorithm
4. **Probability of a Sequence: Forward algorithm**
5. HMM parameter estimation
6. HMM model structure
7. Conclusion

Probability of a Sequence

Given:

- Hidden Markov model: $S, a_{kl}, \Sigma, e_l(x)$.
- Observed symbol sequence $E = x_1, \dots, x_n$.

What is the probability of E .

Let $f_k(i)$ be the probability of the symbol sequence $x_1, x_2, \dots, x_i$ ending in state k . Then:

$$\begin{aligned} f_l(i+1) &= \sum_k (e_l(x_{i+1}) * f_k(i) * a_{kl}) \\ &= e_l(x_{i+1}) * \sum_k (f_k(i) * a_{kl}) \end{aligned}$$

Forward algorithm

Matrix $f_k(i)$, where $k \in S$ and $1 \leq i \leq n$.

- Initialization: $f_k(1) = e_k(x_1)/\#\text{states}$ for all states $k \in S$.
- $f_l(i) = e_l(x_i) \sum_k (f_k(i-1)a_{kl})$ for all states $k \in S$, $i \geq 2$.

Algorithm

- Iteratively build up matrix $f_k(i)$.
- Probability of symbol sequence is sum of entries in last column.

Forward Algorithm / Teacher

Initialization: $f_l(1) = e_l(A)/\#\text{states}$

	A	C	B	A	C
good	0.23				
neutral	0.1				
bad	0.0				

Forward Algorithm / Teacher

$$f_l(i+1) = e_l(x_{i+1}) \sum_k (f_k(i) a_{kl})$$

	A	C	B	A	C
good	0.23	0.0066	.001332	.0023604	.0000857
neutral	0.1	0.0267	.015528	.0019235	.0003278
bad	0.0	0.1575	.014532	0	.0021009

Forward Algorithm / Teacher

Probability of symbol sequence is sum of entries in last column.

	A	C	B	A	C
good	0.23	0.0066	.001332	.0023604	.0000857
neutral	0.1	0.0267	.015528	.0019235	.0003278
bad	0.0	0.1575	.014532	0	.0021009

Σ .0025144

Forward Algorithm / Teacher

Initialization: $f_l(1) = e_l(A)\text{Pr}(l)$

	A	C	B	A	C
good	0.035				
neutral	0.06				
bad	0.0				

Forward Algorithm / Teacher

$$f_l(i+1) = e_l(x_{i+1}) \sum_k (f_k(i) a_{kl})$$

	A	C	B	A	C
good	0.035	0.0019	.000346	.0006952	.0000201
neutral	0.06	0.00675	.00462	.0003083	.0000811
bad	0.0	0.04815	.004352	0	.0006459

Forward Algorithm / Teacher

Probability of symbol sequence is sum of entries in last column.

	A	C	B	A	C
good	0.035	0.0019	.000346	.0006952	.0000201
neutral	0.06	0.00675	.00462	.0003083	.0000811
bad	0.0	0.04815	.004352	0	.0006459

Σ .0007471

Different initializations = different results

Initialization: $v_l(1) = e_l(A)/\#\text{states}$

- $\Pr(A, C, B, A, C) = 0.0025144$

Initialization: $v_l(1) = e_l(A)\Pr(l)$

- $\Pr(A, C, B, A, C) = 0.0007471$

Correct initialization depends on the problem.

Problem: Numerical Stability

The algorithms multiply many small probabilities.

The log transformation cannot be used!

Solution: Rescale variables

For each i , define a scaling variable s_i and define new f variables

$$\tilde{f}_l(i+1) = \frac{f_l(i)}{\prod_{j \leq i} s_j}$$

Contents

1. Introduction
2. Hidden Markov Model definition
3. Most Probable Path: Viterbi algorithm
4. Probability of a Sequence: Forward algorithm
5. HMM parameter estimation
6. HMM model structure
7. Conclusion

Parameter Estimation

Question:

- How do we know the transition probabilities a_{kl} and the emission probabilities $e_k(x)$?

Answer:

- We use training sequences and construct maximum likelihood estimators.

Parameter Estimation

Case 1: State sequences of training sequences are known.

We construct maximum likelihood estimators.

Let A_{kl} be the number of transitions from state k into state l in training data (+ constant).

Let $E_k(x)$ be the number of emissions of symbol x from state k in training data (+ constant).

Transition estimator:

$$a_{kl} = \frac{A_{kl}}{\sum_{l'} A_{kl'}}$$

Emission estimator:

$$e_k(x) = \frac{E_k(x)}{\sum_{x'} E_k(x')}$$

Parameter Estimation

Case 2: State sequences of training sequences are not known.

Viterbi training: We iteratively use the Viterbi algorithm to compute the most probable paths and set the estimators (from case 1) according to this data.

Algorithm sketch:

Initialization: Pick arbitrary model parameters.

REPEAT

Set all A_{kl} and $E_k(x)$ to their constant value

FOR each training sequence

Compute most probable state paths (Viterbi)

Add contribution to A_{kl} and $E_k(x)$.

Update $a_{kl} = \frac{A_{kl}}{\sum_{l'} A_{kl'}}$ and $e_k(x) = \frac{E_k(x)}{\sum_{x'} E_k(x')}$.

UNTIL stopping criterion is reached.

Contents

1. Introduction
2. Hidden Markov Model definition
3. Most Probable Path: Viterbi algorithm
4. Probability of a Sequence: Forward algorithm
5. HMM parameter estimation
6. HMM model structure
7. Conclusion

HMM Model Structure

1. Define set of states.
2. Decide which transitions should be allowed.

In practice, allowing transitions between any two states leads to bad models, even with a lot of training data.

Use knowledge about the problem.

HMM Process Overview

1. Design

- Define set of states.
- Decide which transitions should be allowed.

2. Train

- Use training sequences to obtain transition and emission probabilities.

3. Analyze

- Analyze new sequences using the Viterbi algorithm, Forward algorithm, and Backward algorithm.

Contents

1. Introduction
2. Hidden Markov Model definition
3. Most Probable Path: Viterbi algorithm
4. Probability of a Sequence: Forward algorithm
5. HMM parameter estimation
6. HMM model structure
7. Conclusion

Conclusion

Hidden Markov Model is a powerful tool for sequences analysis.

Many applications:

- Bioinformatics
- Speech recognition
- Strange teacher examples

Further reading and reference

A. Krogh:

An Introduction to Hidden Markov Models for Biological Sequences

In *Computational Methods in Molecular Biology*, edited by

S.L. Salzberg, D.B. Searls and S. Kasif, pages 45-63. Elsevier, 1998.

R. Durbin , S.R. Eddy, A. Krogh, G. Mitchison:

Biological Sequence Analysis : Probabilistic Models of Proteins and

Nucleic Acids

Cambridge Univ. Pr., 1999