Maximum Likelihood Estimation in Hidden Markov Models

The Viterbi Algorithm

Jose Torres Postigo

2023-11-03

1 Objectives

The goal of this lab excercise is to practise with dynamic programming, a technique where an algorithmic problem is broken down into sub-problems, the results are saved an then the sub-problems are optimized to find the optimal solution.

For this project, it will be implemented the Viterbi's algorithm for maximum likelihood estimation of the internal states of a hidden Markov Model. This algorithm is a dynamic programming algorithm for decoding convolutional codes, proposed by A.J. Viterbi.

A Hidden Markov Model $HMM(S,\ O,\ \pi,\ \rho,\ \omega)$ is a probabilistic model of a system with hidden variables, based on the Markov Model, where: - $S = \{s_1,\ ...,\ s_n\}$ are the **states** (values of the random variable).

- $\pi = \{\pi_1,\ ...,\ \pi_n\}$ are the **marginal probabilities** of each state.
- $\rho = \{\rho\ _{ij} \mid 1 \leq i, j \leq n\}$ are the **transition probabilities** among the states.
- $O = \{o_1,\ ...,\ o_n\}$ are the possible observable **outputs** of the system.
- $\omega = \{\omega\ _{ij} \mid 1 \leq i \leq n, 1 \leq j \leq m\}$ are the **output probabilities**, i.e. $\omega\ _{ij}$ is the probability of observing output o_j if the system is in state s_i.

After the implementation, it will be a subject matter the results obtained: number of states, number of observations and time taken for each execution.

2 Experimental Setup

For this experiment, the Viterbi's algorithm will be run through a class given by the lecturer, which is going to run tests of it using hidden Markov models of increasing number of states and sequences of observations of increasing length. The initial values for these tests are 8 states and 256 observations, up to 202 states and 6561 observations.

Table 1: Computational environment considered.

CPU	Intel® Core™ i5 11600KF, 16GB
OS	Windows 11 Home 22H2
Java	openjdk 17.0.7 2023-04-18

3 Empirical Results

A summary of the experimental results is provided in Table 2 in the Appendix.

It can be seen in this graph that the execution time increases with the number of states. Because the Viterbi's algorithm needs to consider all possible sequences of states in order to find the most likely sequence, the more states are there, the more possible sequences exists and the longer the algorithm takes to compute the result.

The graph shows that the execution time increases also with the number of possible observable outputs. This is because the Viterbi's algorithm needs to consider all possible sequences of states for the entire observation sequence. The greater the number of observations, the greater is the number of possible sequences, and it will take longer to the algorithm to find the most likely one.

Also, The graph suggests that the Viterbi algorithm can be used efficiently for sequences of observations with a moderate number of states, such as 05 or 100 states. However, the algorithm may become computationally expensive for a large number of observations and a large number of states, such as 6,000 observations with 200 states, which takes almost a second to compute.

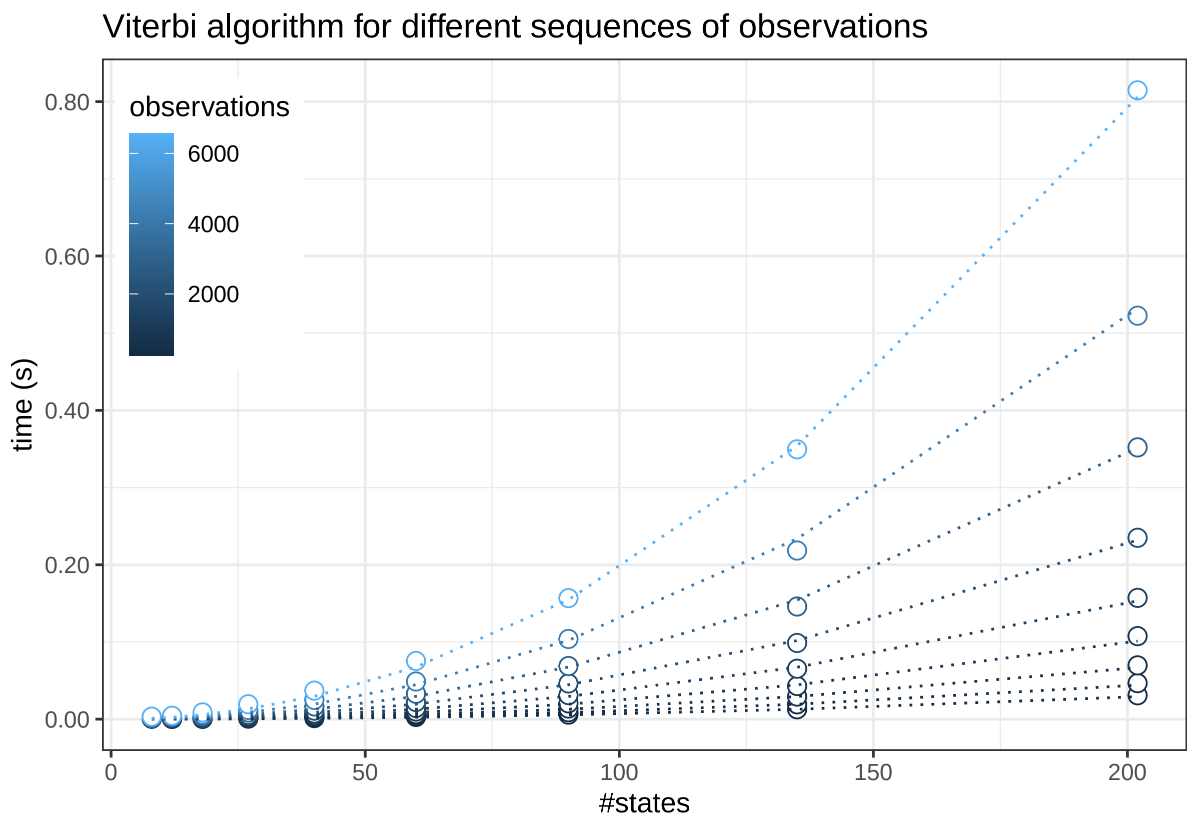

Figure 1: Time as a function of the number of states for different number of observations. The dotted lines show a fit to a power model $t = a \cdot \text{observations}^b \cdot \text{states}^c$.

4 Discussion

The empirical results I obtained highly match the theoretical predictions in general as the standard error is very low. There are some measurements which are not as fitted to the model as it should be, but this does not necessarily mean that there is a problem with the model nor with the implementation, but with the Hardware used.

The Viterbi algorithm seems to be a a powerful tool for finding the most likely sequence of hidden states for a wide range of sequences of observations. Also it seems to be computationally expensive with a large number of states and observations, so it should be taken into consideration using a more efficient algorithm or reducing the number of states and/or observations in these cases.

A Appendix

A.1 Data Summary

Table 2: Summary of the experimental results

#states	#observations	time (s)
8	256	0.0011236
8	384	0.0005438
8	576	0.0005700
8	864	0.0007007
8	1296	0.0012257
8	1944	0.0009905
8	2916	0.0013679
8	4374	0.0029341
8	6561	0.0032015
12	256	0.0004072
12	384	0.0003677
12	576	0.0004515
12	864	0.0008652
12	1296	0.0012245
12	1944	0.0019042
12	2916	0.0032884
12	4374	0.0038818
12	6561	0.0043984
18	256	0.0003711
18	384	0.0004878
18	576	0.0007440
18	864	0.0011125
18	1296	0.0017991
18	1944	0.0027119
18	2916	0.0037831
18	4374	0.0058467
18	6561	0.0087750
27	256	0.0006820
27	384	0.0010063
27	576	0.0016003
27	864	0.0023429
27	1296	0.0035550
27	1944	0.0053918
27	2916	0.0086268

#states	#observations	time (s)
27	4374	0.0119012
27	6561	0.0193271
40	256	0.0013557
40	384	0.0020484
40	576	0.0032662
40	864	0.0048635
40	1296	0.0072812
40	1944	0.0109898
40	2916	0.0165141
40	4374	0.0249419
40	6561	0.0369973
60	256	0.0028236
60	384	0.0042462
60	576	0.0064826
60	864	0.0098865
60	1296	0.0144765
60	1944	0.0222062
60	2916	0.0328669
60	4374	0.0487825
60	6561	0.0753955
90	256	0.0058874
90	384	0.0089772
90	576	0.0137309
90	864	0.0204807
90	1296	0.0311424
90	1944	0.0462752
90	2916	0.0688167
90	4374	0.1038559
90	6561	0.1567934
135	256	0.0127581
135	384	0.0192062
135	576	0.0292451
135	864	0.0429112
135	1296	0.0653461
135	1944	0.0987057
135	2916	0.1459384
135	4374	0.2184448
135	6561	0.3495404
202	256	0.0309171
202	384	0.0466361
202	576	0.0697062
202	864	0.1075412
202	1296	0.1571431
202	1944	0.2349715
202	2916	0.3520929
202	4374	0.5226516
202	6561	0.8146216

A.2 Model Fitting

```
##
## Formula: tavg ~ a * observations^b * states^c
##
## Parameters:
##    Estimate Std. Error t value Pr(>|t|)
## a 1.956e-09  1.690e-10   11.57   <2e-16 ***
## b 1.024e+00  6.182e-03  165.61   <2e-16 ***
## c 2.042e+00  1.318e-02  154.97   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.003443 on 78 degrees of freedom
##
## Number of iterations to convergence: 10
## Achieved convergence tolerance: 9.949e-07
```