Applied Mathematical Sciences, Vol. 13, 2019, no. 4, 151 - 162
HIKARI Ltd, www.m-hikari.com
https://doi.org/10.12988/ams.2019.812195

Using Excel to Visualize State Identification in Hidden Markov Models Using the Forward and Backward Algorithms

William H. Laverty

Department of Mathematics and Statistics
University of Saskatchewan, Saskatoon, Canada

Ivan W. Kelly

Department of Educational Psychology and Special Education
University of Saskatchewan, Saskatoon, Canada

Abstract

In our earlier article (Laverty, Miket, & Kelly, 2002) [4] we used Excel to simulate Hidden Markov models and showed different realizations of hidden states (with differing means and standard deviations) in data. In this follow-up, we show how Excel can be used to simulate the states and calculate the probabilities of the unknown states and visually show the data generating the states. Two-state HMM will be considered in this article. More than two state simulations are possible. We leave this as an exercise to the reader.

Keywords: Hidden Markov Model, Forward algorithm, Backward algorithm, Excel

1 Introduction

Hidden Markov models (HMMs) are a widely used collection of statistical models. These models are applicable when studying a process that goes through a sequence of states. These states are unseen (hidden) but what is observed is data from each state. For example HMMs have been used to model heart rate variability (Walker II 2011) [9], to model financial data (Mamon, Elliott (2007) [7], Genon-Catalot, Jeantheau and Laredo (2000) [3], and to model residuals in regression (Laverty, Miket and Kelly (2002) [5]. Many applications of Hidden Markov Models are possible whenever we have data collected over time. The traditional time series models (AR, MA, ARMA etc.) assume that the process generating the data is constant over a single state. However, if the data is observed over longer periods of time it is likely that there are changes in the states that are generating the data.

An important problem is to identify the hidden states that have generated the observed data. One such solution to this problem is the Viterbi algorithm. It attempts to find the sequence of states for which the likelihood of the observed data is maximized. Another approach is to calculate the probability of the HMM being in a certain state at a certain time given either the data up to that time or the complete set of data. The model is used to make this calculation. If the parameters of the model are unknown they can be estimated using the techniques described in Rabiner (1989) [8]. To calculate these probabilities one uses the iterative procedures of the forward-backward algorithm described in Rabiner.

2. Hidden Markov models

A hidden Markov model will consist of a sequence of states $X_1, X_2, \ldots, X_T$, together with a sequence of observations $Y_1, Y_2, \ldots, Y_T$. We assume that the number of states (possible values of each X_i) is a finite number, m. The states can be represented by the integers $1, 2, \ldots, m$. The states are not observed. The observations $Y_1, Y_2, \ldots, Y_T$ are observed and could be vectors of dimension k. In this paper $k = 1$. The distribution of Y_t depends on the state X_t, the state the Markov process is in at time t. In this paper we are assuming that there are only two states and that the distribution of Y_t if $X_t = i$ $(i = 1, 2)$ is the Normal distribution with mean μ_i and standard deviation σ_i. The parameters of the hidden Markov model are the initial state probabilities

$$\pi_i = \Pr(X_1 = i) \;\; i = ,2, \ldots, m$$

and the transition probability matrix $\Gamma = (\gamma_{ij})$. This is an m x m matrix, with element γ_{ij} being the probability of a transition into state j starting from state i. i.e.

$$\gamma_{ij} = \Pr(X_t = j \mathrm{l}\, X_{t-1} = i),$$

where t denotes time. These two choices allow us to construct a sequence of states (known also as the Markov chain) $X_1, X_2, \ldots, X_T$ constituting the hidden part of a hidden Markov model.

When the Markov chain is in state i, at time t, it emits an observed signal Y_t, which is either a discrete or a continuous random variable (or random vector) with distribution conditional on the current state i.

In the discrete case

$$\Pr[Y_t = y | X_t = i] = p_i(y; \theta_i)$$

where p_i is probability mass function with parameters θ_i.

In the continuous case the conditional density of $Y_t = y$ given $X_t = i$ is $f_i(y;\theta_i)$ In this paper we will assume that $f_i(y;\theta_i)$ is the Normal distribution with mean μ_i and standard deviation σ_i.

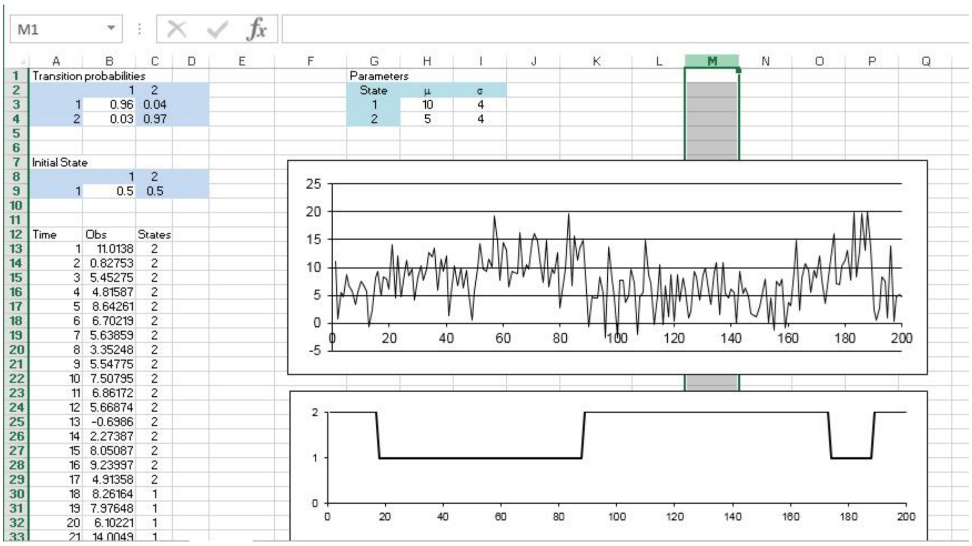
3. Simulation of a Hidden Markov Model with Normal Observations in Excel

Uniform random variates on [0, 1] can be generated in Excel with the function "RAND()". The generation of random variates with a Normal distribution with mean μ and standard deviation σ, can be carried out using the inverse-transform method (Fishman (8)). Namely $Y = F^{-1}(U)$ where $F(u)$ is the desired cumulative distribution of Y and U has a uniform distribution on [0,1]. In Excel this is achieved for the Normal distribution (mean μ, standard deviation σ) with the function "NORMINV(RAND(),μ,σ)."

To simulate a Hidden Markov model with m = 2 states and normal observations with mean μ_i and standard deviation σ_i when the Markov process is in state i we to first need to determine the sequence of states then generate the observations from those states.

Initially we will store the parameters of the model in various cells of the excel spread sheet. For example the transition probabilities γ_{ij} (i = 1,2; j = 1,2) will be stored in the cells B3:C4, the initial probabilities π_i (i = 1,2) will stored in cells B9:C9, and the parameters of the normal distribution (μ_i, σ_i) i = 1, 2 will be stored in cells H3:I4.

The next step is to generate the sequence of states. We generate the first state by determining if a uniform random variate U is above or below π_1. This is achieved by placing the formula "IF(RAND()<B9,1,2)" in cell C13. We now generate the following sequence of states determining if a uniform random variate U is above or below γ_{i1}. This is achieved by placing the formula "IF(OR(AND(C13=1,RAND()<B$3),AND(C13=2,RAND()<B$4)),1,2)" in cell C14. This formula can now be copied down to generate as many states as desired (In this paper we generate 200 states and observations). The final step is to generate normal observations with mean μ_i and standard deviation σ_i at each time point when the process is in state i. This is achieved by pasting the formula "IF(C13=1,H$3+I$3*NORMSINV(RAND()),H$4+I$4*NORMSINV(RAND()))" into cell B13. Again this formula can now be copied down to generate the complete set of data. Below is a copy of the spreadsheet with graphs of the data sequence and the state sequence.



Identification of the State Sequence from the Data Sequence for a Hidden Markov Model

Efficient Methods for computing Likelihood

The Forward and Backward algorithm described below is a special case of Kalman Filtering (Eliott et al (1995) [1].

The *Forward Method*

Let $\mathbf{Y}^{(t)} = (Y_1, Y_2, \ldots, Y_t)'$ and $\mathbf{y}^{(t)} = (y_1, y_2, \ldots, y_t)'$

Consider

$$\alpha_t(i_t) = P\left[\mathbf{Y}^{(t)} = \mathbf{y}^{(t)}, X_t = i_t\right] = P[Y_1 = y_1, Y_2 = y_2, \ldots, Y_t = y_t, X_t = i_t]$$

Note $\alpha_1(i_1) = P\left[\mathbf{Y}^{(1)} = \mathbf{y}^{(1)}, X_1 = i_1\right] = P[Y_1 = y_1, X_1 = i_1] = \pi_{i_1}\beta_{i_1 y_1}$

where $\beta_{i_t y_t} = P[Y_t = y_t | X_t = i_t] = p(y_t | \theta_{i_t})$ in the discrete case

and $\alpha_{t+1}(i_{t+1}) = P\left[\mathbf{Y}^{(t+1)} = \mathbf{y}^{(t+1)}, X_{t+1} = i_{t+1}\right]$

$$= P\left[\mathbf{Y}^{(t)} = \mathbf{y}^{(t)}, Y_{t+1} = y_{t+1}, X_t = i_{t+1}\right]$$

$$= \sum_{i_t} P\left[\mathbf{Y}^{(t)} = \mathbf{y}^{(t)}, Y_{t+1} = y_{t+1}, X_t = i_t, X_{t+1} = i_{t+1}\right]$$

$$= \sum_{i_t} P\left[\mathbf{Y}^{(t)} = \mathbf{y}^{(t)}, X_t = i_t\right] \times P\left[Y_{t+1} = y_{t+1}, X_{t+1} = i_{t+1} \mid \mathbf{Y}^{(t)} = \mathbf{y}^{(t)}, X_t = i_t\right]$$

$$= \sum_{i_t} P\left[\mathbf{Y}^{(t)} = \mathbf{y}^{(t)}, X_t = i_t\right] \gamma_{i_t i_{t+1}} \beta_{i_{t+1} y_{t+1}}$$

$$= \sum_{i_t} \alpha_t(i_t) \gamma_{i_t i_{t+1}} \beta_{i_{t+1} y_{t+1}}$$

or $\boxed{\alpha_{t+1}(i_{t+1}) = \sum_{i_t} \alpha_t(i_t) \gamma_{i_t i_{t+1}} \beta_{i_{t+1} y_{t+1}}}$ with $\alpha_t(t_1) = \pi_{i_1} \beta_{i_1 y_1}$ (4)

This provides a recursion relationship for computing $\alpha_t(i_t)$.

From these values one can compute

$$P[\mathbf{Y} = \mathbf{y}] = P\left[\mathbf{Y}^{(T)} = \mathbf{y}^{(T)}\right]$$

$$= P[Y_1 = y_1, Y_2 = y_2, \ldots, Y_T = y_T]$$

$$= \sum_{i_T} P\left[\mathbf{Y}^{(T)} = \mathbf{y}^{(T)}, X_T = i_T\right]$$

$$= \sum_{i_T} P[Y_1 = y_1, Y_2 = y_2, \ldots, Y_T = y_T, X_T = i_T]$$

$$= \sum_{i_T} \alpha_T(i_T)$$

Hence $P[\mathbf{Y} = \mathbf{y}] = \sum_{i_T} \alpha_T(i_T)$

Computation of $P[\mathbf{Y} = \mathbf{y}]$ using the Backward procedure

Let $\mathbf{Y}^{*(t)} = (Y_{t+1}, Y_{t+2}, \ldots, Y_T)'$ and $\mathbf{y}^{*(t)} = (y_{t+1}, y_{t+2}, \ldots, y_T)'$

Consider $\alpha_t^*(i_t) = P\left[\mathbf{Y}^{*(t)} = \mathbf{y}^{*(t)} \mid X_t = i_t\right]$

$$= P[Y_{t+1} = y_{t+1}, Y_{t+2} = y_{t+2}, \ldots, Y_T = y_T \mid X_t = i_t]$$

Note $\alpha_{T-1}^*(i_{T-1}) = P\left[\mathbf{Y}^{*(T-1)} = \mathbf{y}^{*(T-1)} \mid X_{T-1} = i_{T-1}\right]$

$$= P\left[Y_T = y_T \mid X_{T-1} = i_{T-1}\right] = \sum_{i_T} \gamma_{i_{T-1}i_T} \beta_{i_T y_T}$$

Now $\alpha^*_{t-1}(i_{t-1}) = P\left[\mathbf{Y}^{*(t-1)} = \mathbf{y}^{(t-1)} \mid X_{t-1} = i_{t-1}\right]$

$$= P\left[Y_t = y_t, \mathbf{Y}^{*(t)} = \mathbf{y}^{*(t)} \mid X_{t-1} = i_{t-1}\right]$$

$$= \sum_{i_t} P\left[Y_t = y_t, \mathbf{Y}^{*(t)} = \mathbf{y}^{*(t)}, X_t = i_t \mid X_{t-1} = i_{t-1}\right]$$

$$= \sum_{i_t} P\left[\mathbf{Y}^{*(t)} = \mathbf{y}^{*(t)} \mid X_t = i_t\right] P\left[Y_t = y_t \mid X_t = i_t\right] \times P\left[X_t = i_t \mid X_{t-1} = i_{t-1}\right]$$

$$= \sum_{i_t} \alpha^*_t(i_t) \pi_{i_{t-1}i_t} \beta_{i_{t-1}y_{t-1}}$$

Hence $\boxed{\alpha^*_{t-1}(i_{t-1}) = \sum_{i_t} \alpha^*_t(i_t) \gamma_{i_{t-1}i_t} \beta_{i_{t-1}y_{t-1}}}$ with $\alpha^*_{T-1}(i_{T-1}) = \sum_{i_T} \gamma_{i_{T-1}i_T} \beta_{i_T y_T}$

This defines a backward recursion relationship for computing

$\alpha^*_t(i_t)$ for $t = T$-1, T-2, …, 2, 1

Then $P\left[\mathbf{Y} = \mathbf{y}\right] = P\left[\mathbf{Y}^{*(0)} = \mathbf{y}^{(0)}\right]$

$$= P\left[Y_1 = y_1, Y_2 = y_2, \ldots, Y_T = y_T\right]$$

$$= \sum_{i_1} P\left[Y_1 = y_1, \mathbf{Y}^{*(1)} = \mathbf{y}^{*(1)}, X_1 = i_1\right]$$

$$= \sum_{i_1} P\left[\mathbf{Y}^{*(1)} = \mathbf{y}^{*(1)} \mid X_1 = i_1\right] P\left[X_1 = i_1, Y_1 = y_1\right]$$

$$= \sum_{i_1} P\left[\mathbf{Y}^{*(1)} = \mathbf{y}^{*(1)} \mid X_1 = i_1\right] \pi_{i_1} \beta_{y_1 i_1},$$

$$= \sum_{i_1} \alpha^*_1(i_1) \pi_{i_1} \beta_{y_1 i_1}$$

Hence $P\left[\mathbf{Y} = \mathbf{y}\right] = \sum_{i_1} \alpha^*_1(i_1) \pi_{i_1} \beta_{y_1 i_1}$

Prediction of states from the observations and the model:

Since the states are hidden, we are interested in determining the state at time t given the

observations up to

1. time t (the current time), i.e. $P\left[X_t = i_t \middle| \mathbf{Y}_t = \mathbf{y}_t\right]$,
2. or time $t-1$(the previous time), i.e. $P\left[X_t = i_t \middle| \mathbf{Y}_{t-1} = \mathbf{y}_{t-1}\right]$
3. time T (all the time) i.e. $P\left[X_t = i_t \middle| \mathbf{Y}_T = \mathbf{y}_T\right] = P\left[X_t = i_t \middle| \mathbf{Y} = \mathbf{y}\right]$. Note: All of these can be computed from the forward and backward probabilities, $\alpha_t(i)$ and $\alpha_t^{**}(i)$ and the parameters of the model

1. $$P\left[X_t = i_t \middle| \mathbf{Y}_t = \mathbf{y}_t\right] = P\left[X_t = i_t, \mathbf{Y}_t = \mathbf{y}_t\right] \Big/ P\left[\mathbf{Y}_t = \mathbf{y}_t\right]$$

$$= P\left[X_t = i_t, \mathbf{Y}_t = \mathbf{y}_t\right] \Big/ \sum_i P\left[X_t = i, \mathbf{Y}_t = \mathbf{y}_t\right] = \alpha_t(i_t) \Big/ \sum_i \alpha_t(i)$$

2. $$P\left[X_t = i_t \middle| \mathbf{Y}_{t-1} = \mathbf{y}_{t-1}\right] = P\left[X_t = i_t, \mathbf{Y}_{t-1} = \mathbf{y}_{t-1}\right] \Big/ P\left[\mathbf{Y}_{t-1} = \mathbf{y}_{t-1}\right]$$

$$= \sum_i P\left[X_t = i_t, X_{t-1} = i, \mathbf{Y}_{t-1} = \mathbf{y}_{t-1}\right] \Big/ P\left[\mathbf{Y}_{t-1} = \mathbf{y}_{t-1}\right]$$

$$= \sum_i P\left[X_{t-1} = i, \mathbf{Y}_{t-1} = \mathbf{y}_{t-1}\right] P\left[X_t = i_t \middle| X_{t-1} = i\right] \Big/ \sum_j P\left[X_{t-1} = j, \mathbf{Y}_{t-1} = \mathbf{y}_{t-1}\right]$$

$$= \sum_i \alpha_{t-1}(i)\gamma_{i,i_t} \Big/ \sum_j \alpha_{t-1}(j)$$

Finally

3. $$P\left[X_t = i_t \middle| \mathbf{Y} = \mathbf{y}\right] = \frac{P\left[\mathbf{Y} = \mathbf{y}, X_t = i_t\right]}{P\left[\mathbf{Y} = \mathbf{y}\right]}$$

$$= \frac{P\left[\mathbf{Y}^{(t)} = \mathbf{y}^{(t)}, X_t = i_t\right] P\left[\mathbf{Y}^{*(t)} = \mathbf{y}^{*(t)} \middle| X_t = i_t\right]}{P\left[\mathbf{Y} = \mathbf{y}\right]}$$

$$= \frac{\alpha_t(i_t)\alpha_t^*(i_t)}{P\left[\mathbf{Y} = \mathbf{y}\right]} = \frac{\alpha_t(i_t)\alpha_t^*(i_t)}{\sum_i \alpha_t(i)\alpha_t^*(i)}$$

Since $P\left[\mathbf{Y} = \mathbf{y}\right] = \sum_i P\left[\mathbf{Y} = \mathbf{y}, X_t = i\right] = \sum_i \alpha_t(i)\alpha_t^*(i)$

Consider the case when Y_1, Y_2, ... , Y_T are continuous random variables: Let $\beta_{iy} = f\left(y\middle|\theta_i\right)$

denote the conditional distribution of Y_t given $X_t = i$ and $\beta_{iy} = f\left(y\middle|\theta_i\right)$. Then all of the

calculations carry forward.

Computing State Probabilities in Excel
Computation of Normal Densities

1. We first compute for each observation y_t the normal density for states 1 and 2 with the formula "NORMDIST($B13,$H$4,$I$4,FALSE)" for state 1 and NORMDIST($B13,$H$5,$I$5,FALSE) for state 2, place in cells E13 and F13. These are then copied down to the cells of the last time point (E212 and F212 with 200 data pts.)

Computation of Forward Probabilities $\alpha_t(t)$

1. Next we compute in cells G13 and H13 the starting values of Forward Algoritm $\alpha_1(1)$ (formula E13*B9) and $\alpha_1(2)$ (formula F13*C9)

2. Next we put in the recursion formulae in cells:
G14 (formula $G13*B$3*E14+$H13*B$4*E14) and
H14 (formula $G13*C$3*F14+$H13*C$4*F14)

3. The formulae are then copied down to Cells G212 and H212 to complete the calculation of $\alpha_t(1)$ and $\alpha_t(2)$ $t = 2, \ldots, 200$

Computation of Backward Probabilities $\alpha_t^*(i)$

1. Next we compute in cells I212 and J212 the starting values of Backward Algoritm $\alpha_{200}^*(1) = 1$ and $\alpha_{200}^*(2) = 1$

2. Next we put in the backward recursion formulae in cells:
I211 (formula $I212*$B$3*E211+$J212*C3*E211) and
J211 (formula $I212*$B$4*F211+$J212*C4*F211)

3. The formulae are then copied up to Cells I12 and J12 to complete the calculation of $\alpha_t^*(1)$ and $\alpha_t^*(2)$ t = 199, … , 1

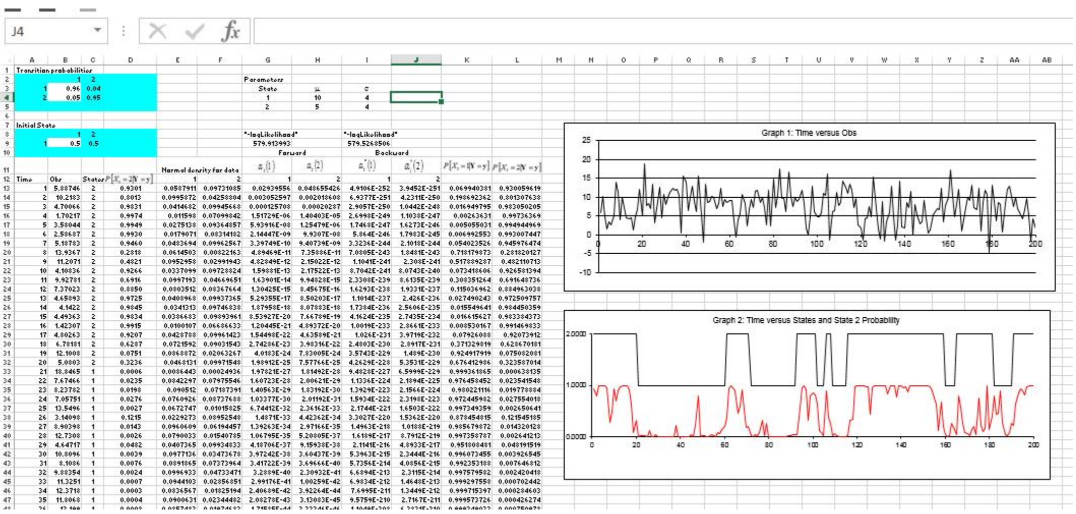
Computation of State Probabilities given all of data $P\left[X_t = i \middle| \mathbf{Y} = \mathbf{y}\right]$

1. We insert the following formulae:
- (G13*I13)/(G13*I13+H13*J13) in Cell K13
- (H13*J13)/(G13*I13+H13*J13) in Cell L13

2. Then we copy those formulae down to cells K212 and L212

Drawing graphs

For convenience the data in column L, generated by $P\left[X_t = 2 \middle| \mathbf{Y} = \mathbf{y}\right]$, is copied into column D. Then **Time** (column A) plotted against **Obs** (column B) in graph 1. Also **Time** (column A) plotted against **States** (column C) and $P\left[X_t = 2 \middle| \mathbf{Y} = \mathbf{y}\right]$ (Column D) in graph 2.

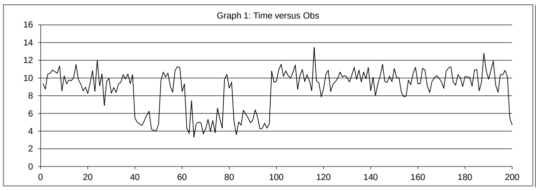


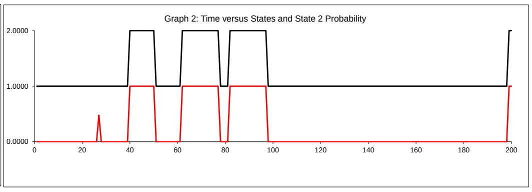
By pressing the refresh key (F9), the data is regenerated and the graphs are reproduced. This allows us to illustrate and assess the accuracy of the Backward and Forward algorithm of state identification. In the next section we will describe some exercises that will help in this endeavour.

4. Exercises that can be performed to illustrate the Performance of the Backward and Forward Algorithm

In these exercises we generate realizations using different sets of parameters to examine the performance of Forward and Backward algorithms in identifying the hidden states of a Hidden Markov model.

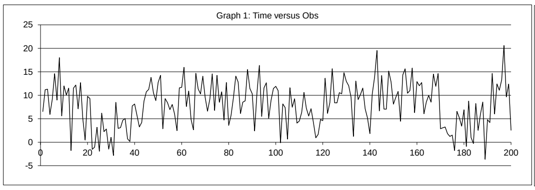
1. Different means ($\mu_1 = 10$, $\mu_2 = 5$), identical small standard deviations ($\sigma_1 = 1$, $\sigma_2 = 1$)

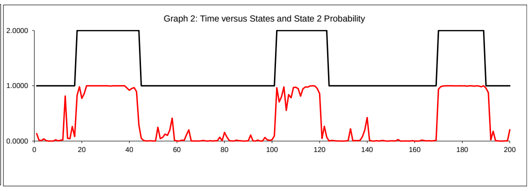




Comment: accurate prediction of states from data

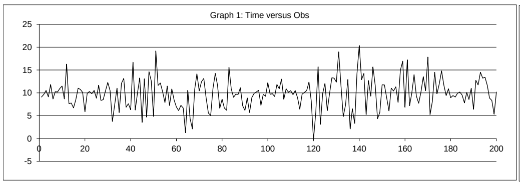
2. Different means ($\mu_1 = 10$, $\mu_2 = 5$), identical larger standard deviations ($\sigma_1 = 4$, $\sigma_2 = 4$)

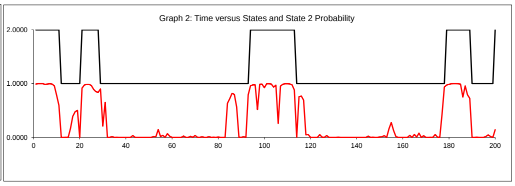




Comment: less accurate prediction of states from data.

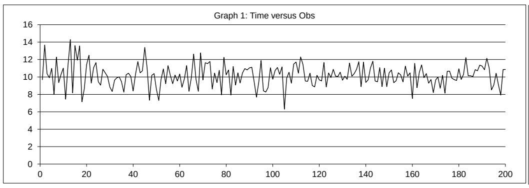
3. Same means ($\mu_1 = 10$, $\mu_2 = 10$), different standard deviations ($\sigma_1 = 4$, $\sigma_2 = 1$)

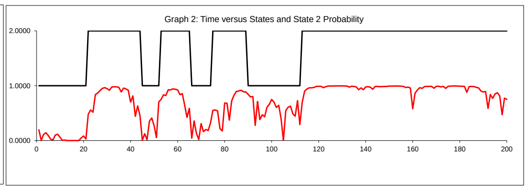




Comment: A small sequence of observations can sometimes have more (sometimes less) volatility, than predicted by the standard deviation. For this reason the identification of states differing in their standard deviations is somewhat less accurate than states that differ solely in their means.

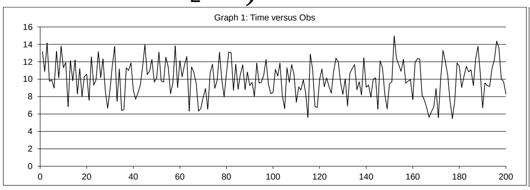
4. Same means (μ_1 = 10, μ_2 = 10) , different but close standard deviations (σ_1 = 2, σ_2 =1)

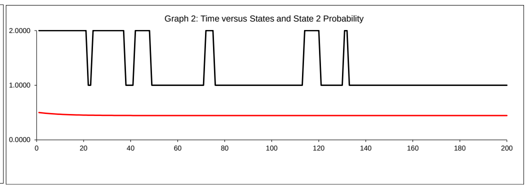




Comment: Accuracy of state identification decreases as standard deviation become similar in value.

5. Same means (μ_1 = 10, μ_2 = 10), same standard deviations (σ_1 = 2, σ_2 =2)





Comment: When there is no difference between then states the data provides no distinguishing information. The Forward and Backward algorithm gives the long run state distribution for the unseen Markov chain.

5. Conclusion

Hidden Markov models are increasingly being used by researchers in a variety of disciplines to identify underlying states contributing to patterns in observed data. These hidden states can provide new information about a process over time that may be missed by traditional statistical approaches. The purpose of this article was to demonstrate how the Forward and Backward algorithm performs in state identification using Excel. By using this article and the included exercises one can develop an applied understanding of underlying state identification in the Hidden Markov Model.

References

[1] R.J. Elliott, L. Aggoun and J. B. Moore, *Hidden Markov Models*, Springer, New York, 1995. https://doi.org/10.1007/978-0-387-84854-9

[2] G. S. Fishman, *Monte Carlo, Concepts, Algorithms and Applications*, Springer, New York, 1996. https://doi.org/10.1007/978-1-4757-2553-7

[3] V. Genon-Catalot, T. Jeantheau and C. Laredo, Stochastic volatility models as Hidden Markov Models and statistical applications, *Bernoulli*, **6** (2000), 1051-1079.

https://doi.org/10.2307/3318471

[4] W.H. Laverty, M.J. Miket and I.W. Kelly, Application of Hidden Markov models on residuals: an example using Canadian traffic accident data, *Perceptual and Motor Skills*, **94** (2002), 1151-1156.
https://doi.org/10.2466/pms.2002.94.3c.1151

[5] W.H. Laverty, M.J. Miket and I.W. Kelly, Examination of Residuals to Vancouver Crisis Call Data by using Hidden Markov Models, *Perceptual and Motor Skills*, **94** (2002), 548-550. https://doi.org/10.2466/pms.2002.94.2.548

[6] W.H. Laverty, M.J. Miket and I.W. Kelly, Simulation of Hidden Markov models with EXCEL, *Journal of Royal Statistical Society: Series D*, **51** (2002), 31-40. https://doi.org/10.1111/1467-9884.00296

[7] R. Mamon, R, R. Elliott, Hidden Markov Models in Finance. International Series n Operations Research and Management Science, Springer, New York, 2007. https://doi.org/10.1007/0-387-71163-5

[8] L. Rabiner, A tutorial on Hidden Markov Models and selected applications in speech recognition, *Proceedings of the IEEE*, **77** (1989), 257 –286
https://doi.org/10.1109/5.18626

[9] M. Walker II, *Hidden Markov Models for Heart Rate Variability with Biometric Applications*, PHD Thesis, All Theses and Dissertations (ETDs), 491, 2011.

Received: January 14, 2019; Published: February 4, 2019