Time Delay Estimation with Hidden Markov Models

Mehdi Azzouzi
azzouzim@aston.ac.uk

Ian T. Nabney
i.t.nabney@aston.ac.uk

Neural Computing Research Group
Aston University, BIRMINGHAM, B4 7ET, UK

Abstract

Most traditional methods for extracting the relationships between two time series are based on cross-correlation. In a non-linear non-stationary environment, these techniques are not sufficient. We show in this paper how to use hidden Markov models (HMMs) to identify the lag (or delay) between different variables for such data. Adopting an information-theoretic approach, we develop a procedure for training HMMs to maximise the mutual information (MMI) between delayed time series. The method is used to model the oil drilling process. We show that cross-correlation gives no information and that the MMI approach outperforms maximum likelihood.

1 Introduction

A key part of multivariate time series analysis is identifying the lags or delays between different variables. This differs from characterising the order or degree of freedom of a single time series, where the goal is to estimate the intrinsic dimensionality of the data in order to determine the window of past samples needed to map the deterministic component of the data generator. In the latter case, the correlation of past values usually tails off gradually, so that the most recent samples have the largest impact on the current value. This is not the case where two time series X_t and Y_t are related by a lag δ. There will be no relationship between X_{t-d} and Y_t for $d < \delta$.

Under the assumption of stationarity, cross-correlation is a powerful tool for measuring and modelling linear relationships between variables. They are often used as the basis for identifying the order of non-linear models as it is fast to compute. However, in many real-world applications the assumptions of linear dependencies and stationarity are not valid.

In this paper, we consider the problem of modelling processes which manifest a sequentially changing behaviour: the process variables usually remain constant, except for minor fluctuations, and then, at certain times, change to another set of values. Our approach is based on using hidden Markov models in order to model the distribution of the time series. More precisely, given two time series X_t and Y_t, related by a lag δ and generated from a non-stationary underlying process which exhibits different regimes, we show how HMMs can be used to estimate the value of δ. In [1] we proposed a procedure based on maximum likelihood estimation in order to estimate the lag. We adopt here an information-theoretic approach and develop a procedure for training HMMs to maximise the mutual information between X_t and Y_t.

We apply this approach to the analysis of the oil well drilling process, which exhibits complex time relationships between variables and a highly non-stationary behaviour. A fluid called 'mud' carries the drilling cuttings up the hole to the surface. The time it takes for the cuttings to come up to the surface is called the *lag for return* and is a crucial parameter for modelling the process. This time-varying parameter depends not only on the depth of the hole and the pressure of the drilling fluid, but also on the geology of the surrounding rock formation and the drilling mode. In section 4 we analyse drilling data and compare our results with numerical models based on fluid mechanics.

2 Modelling time series with HMMs

A multivariate continuous time series is a sequence of continuous m-dimensional random variable $\mathcal{O}$, such that at for each time t, O_t ranges over a continuous space. For simplicity, suppose that $m = 2$ so that at time T we have seen a sequence of such observations $o_1^T = [o_1, \ldots, o_t, \ldots, o_T]$ [1] where $o_t = (x_t, y_t)$ is the observed data at time t. Given a sequence, the task is to model the probability distribution from which the time series was generated.

Let $\mathcal{S}$ be a discrete random variable taking values in the set $\{q_1, \ldots, q_N\}$ and assume that the system at any time t is in one and only one of the N states $q_1, \ldots, q_N$. The random variable O_t can be considered to be a probabilistic function of the underlying states, i.e. o_t is an observed measurement from the system but the underlying states are not themselves directly observable. Assuming that the state variable S_t is a stationary discrete-time first-order Markov process, the resulting model is a doubly stochastic process and is called a first-order hidden Markov model (HMM). It is called hidden because the state of the underlying process is *not* observable, but can only be observed through another set of stochastic processes that produce the sequence of observations. Thus the model assumes two sets of conditional independence relations: that O_t is independent of all other random variables given S_t and that S_t is independent of $S_1, \ldots, S_{t-2}$ given S_{t-1} (the Markov property). Using these independence relations, the joint probability for the sequence of states and observations can be written as

$$P(O_1^T, S_1^T) = P(S_1)P(O_1|S_1) \prod_{t=2}^{T} P(S_t|S_{t-1})P(O_t|S_t) \quad (1)$$

In general, the parameters of a specific model are referred as $\Theta = \{A, B, \Pi\}$, where A denotes the state transition matrix, $B = \{b_i(o_t)\}$ the observation probability distributions in each state and Π the initial state distribution. For time series modelling, the probability distributions B are often chosen to be a finite mixture of Gaussians, as it can approximate, arbitrarily closely, any finite, continuous density function, provided that enough components are used[2]. HMMs have been successfully applied in speech recognition [8], cryptography, and more recently in other areas such computational biology [2].

[1] We use the notation O_i^j to denote the sequence of random variables from time i to time j, i.e. $O_i^j = [O_i, O_{i+1}, \ldots, O_j]$. A sequence of observations will be denoted $o_i^j = [o_i, o_{i+1}, \ldots, o_j]$.

3 Delay estimation

Consider the following problem: a sequence of observations $o_1^T = [(x_1, y_1), \ldots, (x_T, y_T)]$ is being generated by an underlying system. Unfortunately, we do not see the true sequence o_1^T but a modified version where one variable is delayed: $o_1^T(\delta) = [(x_{1-\delta}, y_1), \ldots, (x_{T-\delta}, y_T)]$. Our task is to estimate the value of δ. Given a two dimensional time series vector $O_1^T(\delta) = (X_{t-\delta}, Y_t)_{t=1\ldots T}$, we say that Y_t leads X_t by an unknown lag δ. For convenience and clarity, we define $X^d \equiv X_1^T(d) = [X_{1-d}, \ldots, X_{T-d}]$ to be the time series X_t delayed by d steps, omitting time indexes and $O^d \equiv (X^d, Y)$. The problem can be viewed as a *synchronisation* problem, where the goal is to recover the *correct* sequence of hidden states. Figure 1 shows the underlying sequence $S(d)$ corresponding to different delayed time series (in this example, the system can switch between two states q_1 and q_2). The value δ corresponds to a *true* sequence of hidden states and the task is to recover this sequence by identifying the corresponding observation sequence O^δ.

3.1 Maximum likelihood

The usual procedure for training HMMs is to find the parameter values of Θ that maximise the likelihood or the log-likelihood of the observed sequence of training data o_1^T

$$\Theta^* = \arg\max_{\Theta} \log p(o_1^T \mid \Theta) \quad (2)$$

The standard approach for doing this is to use the Baum-Welch algorithm, which is the relevant version of the EM algorithm [3]. In

[2] Although continuous density HMMs are applicable to a large number of problems, autoregressive HMMs, where the observation vectors are drawn from a state-dependent autoregressive process, have been investigated for time series modelling [6].

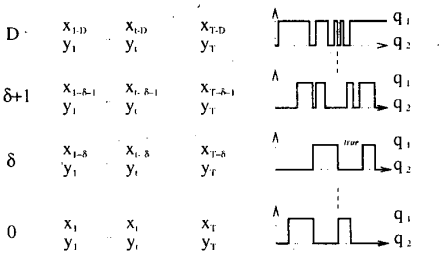

Figure 1: The synchronisation problem: we assume an underlying *true* state sequence S_{true}. This sequence is not observable but can be recovered by identifying the corresponding observation sequence o^{δ}, i.e. when x_t and y_t are properly synchronised.

[1], we proposed a delay estimation procedure based on MLE. First, an HMM Θ^d is trained with the delayed sequence o^d and the likelihood of each model Θ^d is estimated. The lag is then obtained by finding the most likely model,

$$\delta = \arg\max_{d} \log \mathcal{L}^d \tag{3}$$

where $\mathcal{L}^d = p(o^{\delta})$ denotes the likelihood of the delayed sequence o^d given the model Θ^d. The approach is motivated by the fact that the sequence o^d corresponds to a specific sequence of hidden states representing the dynamics of the process (Figure 1). Intuitively, we expect that $\mathcal{L}^d$ will always be less than $\mathcal{L}^{\delta}$ $(d \neq \delta)$. Indeed, assuming that for each time step t, $X_{t-\delta}$ and Y_t have been generated by a specific state S_t^*, the system will not be able to enter that *true* state if X_t and Y_t are not properly synchronised.

3.2 Mutual information

There are many very important properties of the Maximum Likelihood Estimate (MLE) but most of them stem from an implicit assumption of model correctness. The justifications for using MLE to estimate the mean and the variance of a Gaussian distribution, for example, assume that the sample has indeed been generated by a Gaussian. If, however, we do not know the 'correct' model which has generated the data and if there is no reason to believe that the sample has been generated from any particular model then we can ask ourselves whether the use of MLE is appropriate.

In previous work [4], [8], the maximum mutual information (MMI) criterion for discriminative training of multiple HMMs has been introduced in order to alleviate problems that may occur when several HMMs are to be designed at the same time. This leads to an algorithm where the goal is to choose a correct model m amongst a set of M models that maximises the following expression:

$$I(O, \Theta_m) = \left[\log \frac{P(o_1^T|\Theta_m)}{\sum_{n=1}^{M} P(o_1^T|\Theta_n)} \right] \tag{4}$$

In contrast, the motivation of our approach is not to maximise the mutual information between observation sequence and a *complete* set of models $\Theta = (\Theta_1, \ldots, \Theta_M)$, instead we are trying to estimate the parameters of a single HMM that maximises the mutual information between two random variables.

In order to measure the amount of information with respect to Y we may expect to obtain by observing X^d, it is useful to introduce the concept of mutual information $I(X^d, Y \mid \Theta)$ between X^d and Y:

$$I(X^d, Y) = H(X^d) + H(Y) - H(X^d, Y) \tag{5}$$

where $H(X) = -E_X[\log P(X)]$ is the entropy of the random variable X. As the joint probability $P(X^d, Y)$ can be rewritten as $P(X^d, Y) = P(Y \mid X^d)P(X^d)$, we have

$$I(X^d, Y) = H(Y) - H(Y \mid X^d) \tag{6}$$

In our problem, the goal is to maximise the information with respect to Y we may get by observing a delayed time series X^d. We notice however that the first term of Equation 6 does not depend on d and can be discarded in the optimisation procedure. Indeed, for two different values of d, i.e. for two different time series X^{d_1} and X^{d_2}, the entropy of Y does not affect the change in the mutual information $I(X^{d_1}, Y) - I(X^{d_2}, Y)$. Thus, maximising the conditional distribution $P(Y \mid X^d)$ is equivalent to maximising the mutual information between Y and X^d.

$$\arg\max_{d} I(X^d, Y) = \arg\max_{d} \log P(Y \mid X^d) \tag{7}$$

Comparing Equation 7 to Equation 3, we see that MLE and MMIE differ in the objective function. In MLE, we are interested in estimating parameters that maximise the joint

probability. The MMIE approach leads to maximising the conditional probability. In terms of previous work, our approach resembles that of [9] who used mutual information for image matching.

3.3 Learning algorithm

The Baum-Welch algorithm is a hill-climbing algorithm for maximum likelihood estimation, which does not require the model gradient. The procedure iterates between a step that fixes the current parameters and computes the posterior probabilities over the hidden states (E step) and a step that re-estimates the parameters given these conditional distributions over the hidden states (M step) [8]. Unfortunately, no such method is known for MMI estimation and we must therefore resort to the use of traditional minimisation techniques, with the objective function $E = -\log P(Y \mid X, \Theta)$. Suppose that the sample vector O_t is composed of two vectors $O_t = (X_t, Y_t)$ with mean $\mu = (\mu_1, \mu_2)$ and covariance matrix $\Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}$ then the conditional density $f(y_t \mid x_t)$ is Gaussian with mean $\mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(x_t - \mu_1)$ and covariance $\Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}$.

For an HMM with Gaussian observation densities, the output density of each hidden state i is parameterised by a mean vector and a covariance matrix. The derivatives of the density function with respect to the parameters of each hidden state can be easily found. The derivatives of the objective function E with respect to the parameters $\Theta = \{A, B, \Pi\}$ are obtained using the forward and backward variables of the Baum-Welch algorithm (E step). They can then be used either in a simple gradient descent algorithm or a nonlinear optimisation algorithm like conjugate gradients, which uses the gradient of the objective function. Such methods may require a line search which involves many evaluations of the objective function. Evaluating the objective function requires the computation of the forward variables, whereas the derivative of the function needs both forward and backward variables. Each forward and backward recursion requires on the order of N^2T calculations, where N is the number of hidden states and T is the length of the sequence. This can lead to a computationally expensive algorithm, especially if the HMM contains a large number of parameters. This is not a big issue for the problem we are interested in as the models we consider are relatively small.

4 Results on drilling data

A significant difficulty during exploration drilling is ensuring the drilling debris is effectively removed from the bore; this is known as the 'hole cleaning' problem [5, 7]. In the case of vertical wells, an adequate velocity in the mud circulation is generally sufficient to guarantee that most debris are brought to the surface. The problem is more complicated when drilling deviated wells[3] since gravity settlement can occur. The gradual build up of low gravity solids increases the torque required to turn the drill string. In extreme cases, the drill pipe may get stuck or even fracture off.

The time it takes for the cuttings to come to the surface is called the lag for return and is a crucial parameter in early stuck pipe detection and modelling the drilling process. The algorithms currently used on rigs to estimate the lag for return are physical models based on fluid mechanics but are believed to have a precision in the order of several minutes, mainly because of the poor understanding of downhole conditions. As our analysis will show, even this level of accuracy may be optimistic compared to reality.

At present, no equipment exists to monitor the status of hole cleaning. Recently a new device, capable of detecting fine particulate solids in drilling fluids, has been developed by Thule Rigtech Ltd. Our aim is to use this device to monitor trends in the volumes of drilled solids in order to obtain a better picture of downhole conditions with regard to drilled solids than has ever been possible before.

A drilling engineer has gathered a large dataset of drilling variables and low gravity solid information from a rig in Holland. As all the data are collected on the surface, if δ represents the lag, then Y_t, which is the amount of low gravity solids (LGS) measured at time t, is effectively the amount of solids that has been generated by the bit at

[3] Whenever possible, wells are drilled vertically, but sometimes, especially offshore, it is necessary to deviate from vertical in order to reach a wide spread of targets from a single platform.

time $t-\delta$. Thus assuming that Y_t is related to other drilling parameters $X_{t-\delta}$, it makes sense to use the procedure described in Section 3 in order to estimate the lag for return, as we believe that δ remains relatively constant over a 2 hour time scale. Typically X_t represents one relevant drilling parameter (although we have also considered models with more than one parameter): for instance, the pressure of the circulating fluid inside the pipe or the torque of the pipe, as the drill pipe is subjected to both torsion and tension. The total force applied (hook load) on the drilling system in order to hold the drill pipe in the rig and the rate of progress (ROP), are other important parameters.

Figure 2 shows our results on a data set representing 'normal' drilling conditions, as no special event was identified by the drilling engineers. The data contains 450 data points and represents a period of 4 hours of drilling. The numerical models suggest a value of 31 min for the lag for return. Figure 2a plots the cross-correlogram between LGS and two important drilling parameters, namely the pipe pressure and the pipe hook load. No correlation significantly different from zero can be detected. For each value of d, 100 HMMs with different initial parameters have been trained using MLE and MMI approaches. Figures 2b and 2c plot the mean and the two standard deviation error bars computed around the global maximum[4]. The MLE approach suggests a value between 36 and 40 min whereas the MMIE approach is more confident and suggests a sharper peak at 37 min, which is statistically significant. The results have been obtained by training 3 state HMMs using the pipe hook load for the time series X_t.

We have successfully applied our procedure to other datasets corresponding to different drilling situations. Table 1 reports the results of our simulations on different data sets and shows how they differ from MLE and the ones obtained from the fluid mechanics model. In no case was it possible to identify the lag from the cross-correlogram. The MMI approach always suggests a sharper peak than MLE. It should

[4]The likelihood may have different maxima. We have therefore discarded non interesting local maxima and computed the mean and the variance around the 'global' maximum we obtained over the 100 HMMs.

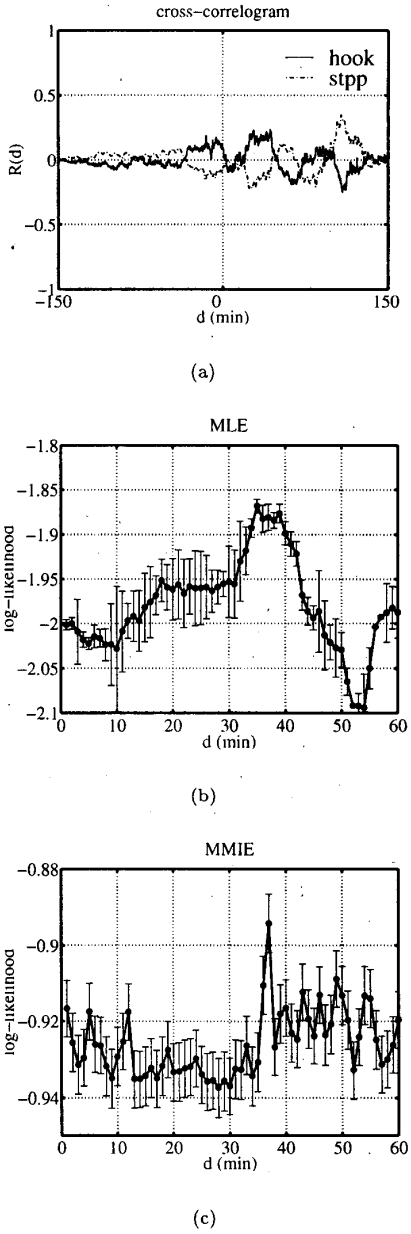

Figure 2: Cross-correlogram (top) and log-likelihood of the observation sequences o^d given the model HMM^d for MLE (middle) and MMI (bottom) approaches.

be noticed that for dataset D it is possible to estimate the lag for return by visually comparing two parameters, since a transition occurred between two geological formations which significantly affected ROP and LGS. This gave a value around 23 min, confirming the superiority of our HMM approach.

Data	fluid mech.	MLE	MMIE
A	68	72-75	74
B	31	36-40	37
C	36	40-43	42
D	43	22-25	23

Table 1: Our results (in min) compared to the ones obtained by fluid mechanics models.

5 Conclusion

In this paper, we have shown how hidden Markov models can be used to identify relationships between variables. We proposed a novel mutual information estimation approach, which maximises the conditional probability of one variable with respect to the other. The method was tested on data from a real-world process and it is clear that relationships between variables can be identified using HMMs. Our procedure also outperforms numerical models based on fluid mechanics used in the oil industry. When compared to MLE, the MMIE approach seems consistently to estimate the lag more precisely. However, the MMIE implementation is time consuming: a typical MMIE procedure needs roughly 20 times more forward-backward passes than MLE.

6 Acknowledgement

This work is funded by EPSRC (grant GR/L08632), Shell and Agip. The authors would like to thank Mike Affleck of Thule Rigtech Ltd. for helping in collecting and interpreting the data. Mehdi Azzouzi is grateful to Francesco Vivarelli for helpful discussions.

References

[1] M Azzouzi and I T Nabney. Analysing time series structure with hidden Markov models. In T Constantinides, S Y Kung, M Niranjan, and E Wilson, editors, *Neural Networks for Signal Processing*, volume 8, pages 402–408. IEEE, 1998.

[2] P Baldi, Y Chauvin, T Hunkapiller, and M A McClure. Hidden Markov models in molecular biology: New algorithms and application. In S J Hanson, J D Cowan, and C L Giles, editors, *Advances in Neural Information Processing Systems*, volume 5, pages 11–18. Morgan Kaufmann, San Mateo, 1993.

[3] L Baum, T Petrie, G Soules, and N Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*, 41:164–171, 1970.

[4] P F Brown. *The Acoustic-Modeling Problem in Automatic Speech Recognition.* PhD thesis, IBM Thomas J. Watson Research Center, 1987.

[5] G J Guild, I M Wallace, and M J Wassenborg. Hole cleaning program for extended reach wells. *Society of Petroleum Engineers*, pages 425–433, 1995. In SPE/IADC Drilling Conference.

[6] A B Poritz. Linear Predictive Hidden Markov Models and the Speech Signal. In *Proc. ICASSP*, pages 1291–1294, May 1982.

[7] H Rabia. *OilWell Drilling Engineering: Principles and Practice.* Graham & Trotman, 1985.

[8] L R Rabiner. A tutorial on hidden Markov models and selected application in speech recognition. In *Proceedings of the IEEE*, volume 77-2, pages 257–286, 1989.

[9] P Viola and N Wells. Alignment by maximization of mutual information. In I. C. S. Press, editor, *International Conference in Computer Vision*, pages 16–23, June 1995.