1

Inference and estimation in probabilistic time series models

David Barber, A. Taylan Cemgil and Silvia Chiappa

1.1 Time series

The term 'time series' refers to data that can be represented as a sequence. This includes for example financial data in which the sequence index indicates time, and genetic data (e.g. *ACATGC* ...) in which the sequence index has no temporal meaning. In this tutorial we give an overview of discrete-time probabilistic models, which are the subject of most chapters in this book, with continuous-time models being discussed separately in Chapters 4, 6, 11 and 17. Throughout our focus is on the basic algorithmic issues underlying time series, rather than on surveying the wide field of applications.

Defining a probabilistic model of a time series $y_{1:T} \equiv y_1, \ldots, y_T$ requires the specification of a joint distribution $p(y_{1:T})$.[1] In general, specifying all independent entries of $p(y_{1:T})$ is infeasible without making some statistical independence assumptions. For example, in the case of binary data, $y_t \in \{0, 1\}$, the joint distribution contains maximally $2^T - 1$ independent entries. Therefore, for time series of more than a few time steps, we need to introduce simplifications in order to ensure tractability. One way to introduce statistical independence is to use the probability of a conditioned on observed b

$$p(a|b) = \frac{p(a, b)}{p(b)}.$$

Replacing a with y_T and b with $y_{1:T-1}$ and rearranging we obtain $p(y_{1:T}) = p(y_T|y_{1:T-1})p(y_{1:T-1})$. Similarly, we can decompose $p(y_{1:T-1}) = p(y_{T-1}|y_{1:T-2})p(y_{1:T-2})$. By repeated application, we can then express the joint distribution as[2]

$$p(y_{1:T}) = \prod_{t=1}^{T} p(y_t|y_{1:t-1}).$$

This factorisation is consistent with the causal nature of time, since each factor represents a generative model of a variable conditioned on its past. To make the specification simpler, we can impose conditional independence by dropping variables in each factor conditioning set. For example, by imposing $p(y_t|y_{1:t-1}) = p(y_t|y_{t-m:t-1})$ we obtain the mth-order Markov model discussed in Section 1.2.

[1] To simplify the notation, throughout the tutorial we use lowercase to indicate both a random variable and its realisation.

[2] We use the convention that $y_{1:t-1} = \emptyset$ if $t < 2$. More generally, one may write $p_t(y_t|y_{1:t-1})$, as we generally have a different distribution at each time step. However, for notational simplicity we generally omit the time index.

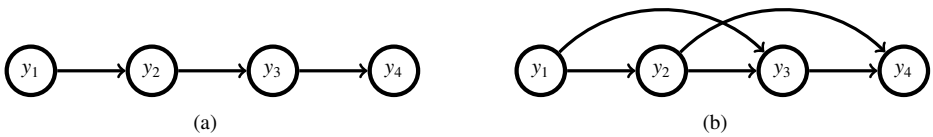


Figure 1.1 Belief network representations of two time series models. (a) First-order Markov model $p(y_{1:4}) = p(y_4|y_3)p(y_3|y_2)p(y_2|y_1)p(y_1)$. (b) Second-order Markov model $p(y_{1:4}) = p(y_4|y_3, y_2)p(y_3|y_2, y_1)p(y_2|y_1)p(y_1)$.

A useful way to express statistical independence assumptions is to use a belief network graphical model which is a directed acyclic graph[3] representing the joint distribution

$$p(y_{1:N}) = \prod_{i=1}^{N} p(y_i|\text{pa}\,(y_i))\,,$$

where $\text{pa}\,(y_i)$ denotes the parents of y_i, that is the variables with a directed link to y_i. By limiting the parental set of each variable we can reduce the burden of specification. In Fig. 1.1 we give two examples of belief networks corresponding to a first- and second-order Markov model respectively, see Section 1.2. For the model $p(y_{1:4})$ in Fig. 1.1(a) and binary variables $y_t \in \{0, 1\}$ we need to specify only $1 + 2 + 2 + 2 = 7$ entries,[4] compared to $2^4 - 1 = 15$ entries in the case that no independence assumptions are made.

Inference

Inference is the task of using a distribution to answer questions of interest. For example, given a set of observations $y_{1:T}$, a common inference problem in time series analysis is the use of the posterior distribution $p(y_{T+1}|y_{1:T})$ for the prediction of an unseen future variable y_{T+1}. One of the challenges in time series modelling is to develop computationally efficient algorithms for computing such posterior distributions by exploiting the independence assumptions of the model.

Estimation

Estimation is the task of determining a parameter θ of a model based on observations $y_{1:T}$. This can be considered as a form of inference in which we wish to compute $p(\theta|y_{1:T})$. Specifically, if $p(\theta)$ is a distribution quantifying our beliefs in the parameter values before having seen the data, we can use Bayes' rule to combine this prior with the observations to form a posterior distribution

$$\underbrace{p(\theta|y_{1:T})}_{\text{posterior}} = \frac{\underbrace{p(y_{1:T}|\theta)}_{\text{likelihood}}\underbrace{p(\theta)}_{\text{prior}}}{\underbrace{p(y_{1:T})}_{\text{marginal likelihood}}}.$$

The posterior distribution is often summarised by the maximum a posteriori (MAP) point estimate, given by the mode

$$\theta^{\text{MAP}} = \underset{\theta}{\text{argmax}}\; p(y_{1:T}|\theta)p(\theta).$$

[3] A directed graph is acyclic if, by following the direction of the arrows, a node will never be visited more than once.

[4] For example, we need one specification for $p(y_1 = 0)$, with $p(y_1 = 1) = 1 - p(y_1 = 0)$ determined by normalisation. Similarly, we need to specify two entries for $p(y_2|y_1)$.

It can be computationally more convenient to use the log posterior,

$$\theta^{\text{MAP}} = \underset{\theta}{\text{argmax}} \log \left(p(y_{1:T}|\theta) p(\theta) \right),$$

where the equivalence follows from the monotonicity of the log function. When using a 'flat prior' $p(\theta) = \text{const.}$, the MAP solution coincides with the maximum likelihood (ML) solution

$$\theta^{\text{ML}} = \underset{\theta}{\text{argmax}}\, p(y_{1:T}|\theta) = \underset{\theta}{\text{argmax}} \log p(y_{1:T}|\theta).$$

In the following sections we introduce some popular time series models and describe associated inference and parameter estimation routines.

1.2 Markov models

Markov models (or Markov chains) are of fundamental importance and underpin many time series models [21]. In an mth order Markov model the joint distribution factorises as

$$p(y_{1:T}) = \prod_{t=1}^{T} p(y_t|y_{t-m:t-1}),$$

expressing the fact that only the previous m observations $y_{t-m:t-1}$ directly influence y_t. In a time-homogeneous model, the transition probabilities $p(y_t|y_{t-m:t-1})$ are time-independent.

1.2.1 Estimation in discrete Markov models

In a time-homogeneous first-order Markov model with discrete scalar observations $y_t \in \{1, \ldots, S\}$, the transition from y_{t-1} to y_t can be parameterised using a matrix θ, that is

$$\theta_{ji} \equiv p(y_t = j|y_{t-1} = i, \theta), \quad i, j \in \{1, \ldots, S\}.$$

Given observations $y_{1:T}$, maximum likelihood sets this matrix according to

$$\theta^{ML} = \underset{\theta}{\text{argmax}} \log p(y_{1:T}|\theta) = \underset{\theta}{\text{argmax}} \sum_t \log p(y_t|y_{t-1}, \theta).$$

Under the probability constraints $0 \leq \theta_{ji} \leq 1$ and $\sum_j \theta_{ji} = 1$, the optimal solution is given by the intuitive setting

$$\theta^{ML}_{ji} = \frac{n_{ji}}{T-1},$$

where n_{ji} is the number of transitions from i to j observed in $y_{1:T}$.

Alternatively, a Bayesian treatment would compute the parameter posterior distribution

$$p(\theta|y_{1:T}) \propto p(\theta) p(y_{1:T}|\theta) = p(\theta) \prod_{i,j} \theta_{ji}^{n_{ji}}.$$

In this case a convenient prior for θ is a Dirichlet distribution on each column $\theta_{:i}$ with hyperparameter vector $\alpha_{:i}$

$$p(\theta) = \prod_i \mathcal{DI}(\theta_{:i}|\alpha_{:i}) = \prod_i \frac{1}{Z(\alpha_{:i})} \prod_j \theta_{ji}^{\alpha_{ji}-1},$$

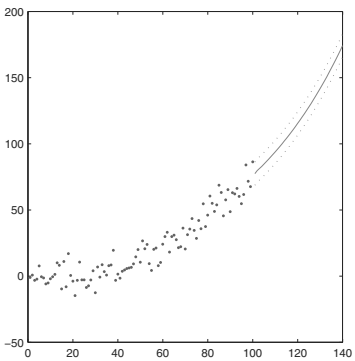


Figure 1.2 Maximum likelihood fit of a third-order AR model. The horizontal axis represents time, whilst the vertical axis the value of the time series. The dots represent the 100 observations $y_{1:100}$. The solid line indicates the mean predictions $\langle y \rangle_t, t > 100$, and the dashed lines $\langle y \rangle_t \pm \sqrt{r}$.

where $Z(\alpha_{:i}) = \int_0^1 \prod_j \theta_{ij}^{\alpha_{ji}-1} d\theta$. The convenience of this 'conjugate' prior is that it gives a parameter posterior that is also a Dirichlet distribution [15]

$$p(\theta|y_{1:T}) = \prod_i \mathcal{D}\mathcal{I}(\theta_{:i}|\alpha_{:i} + n_{:i}).$$

This Bayesian approach differs from maximum likelihood in that it treats the parameters as random variables and yields distributional information. This is motivated from the understanding that for a finite number of observations there is not necessarily a 'single best' parameter estimate, but rather a distribution of parameters weighted both by how well they fit the data and how well they match our prior assumptions.

1.2.2 Autoregressive models

A widely used Markov model of continuous scalar observations is the autoregressive (AR) model [2, 4]. An mth-order AR model assumes that y_t is a noisy linear combination of the previous m observations, that is

$$y_t = a_1 y_{t-1} + a_2 y_{t-2} + \cdots + a_m y_{t-m} + \epsilon_t,$$

where $a_{1:m}$ are called the AR coefficients, and ϵ_t is an independent noise term commonly assumed to be zero-mean Gaussian with variance r (indicated with $\mathcal{N}(\epsilon_t|0, r)$). A so-called *generative form* for the AR model with Gaussian noise is given by[5]

$$p(y_{1:T}|y_{1:m}) = \prod_{t=m+1}^{T} p(y_t|y_{t-m:t-1}), \qquad p(y_t|y_{t-m:t-1}) = \mathcal{N}\Big(y_t \Big| \sum_{i=1}^{m} a_i y_{t-i}, r\Big).$$

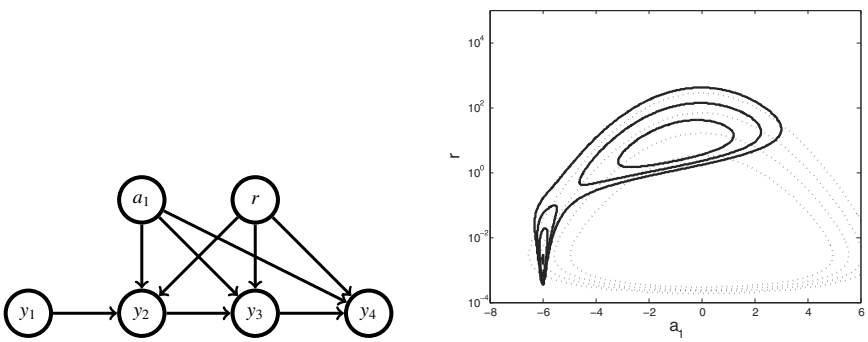
Given observations $y_{1:T}$, the maximum likelihood estimate for the parameters $a_{1:m}$ and r is obtained by maximising with respect to a and r the log likelihood

$$\log p(y_{1:T}|y_{1:m}) = -\frac{1}{2r} \sum_{t=m+1}^{T} \Big(y_t - \sum_{i=1}^{m} a_i y_{t-i}\Big)^2 - \frac{T-m}{2} \log(2\pi r).$$

The optimal $a_{1:m}$ are given by solving the linear system

$$\sum_i a_i \sum_{t=m+1}^{T} y_{t-i} y_{t-j} = \sum_{t=m+1}^{T} y_t y_{t-j} \qquad \forall j,$$

[5] Note that the first m variables are not modelled.



(a) (b)

Figure 1.3 (a) Belief network representation of a first-order AR model with parameters a_1, r (first four time steps). (b) Parameter prior $p(a_1, r)$ (light grey, dotted) and posterior $p(a_1, r|y_1 = 1, y_2 = -6)$ (black). The posterior describes two plausible explanations of the data: (i) the noise r is low and $a_1 \approx -6$, (ii) the noise r is high with a set of possible values for a_1 centred around zero.

which is readily solved using Gaussian elimination. The linear system has a Toeplitz form that can be more efficiently solved, if required, using the Levinson-Durbin method [9]. The optimal variance is then given by

$$r = \frac{1}{T-m} \sum_{t=m+1}^{T} \left(y_t - \sum_{i=1}^{m} a_i y_{t-i} \right)^2 .$$

The case in which y_t is multivariate can be handled by assuming that a_i is a matrix and ϵ_t is a vector. This generalisation is known as vector autoregression.

Example 1 We illustrate with a simple example how AR models can be used to estimate trends underlying time series data. A third-order AR model was fit to the set of 100 observations shown in Fig. 1.2 using maximum likelihood. A prediction for the mean $\langle y \rangle_t$ was then recursively generated as

$$\langle y \rangle_t = \begin{cases} \sum_{i=1}^{3} a_i \langle y \rangle_{t-i} & \text{for} \quad t > 100, \\ y_t & \text{for} \quad t \leq 100 . \end{cases}$$

As we can see (solid line in Fig. 1.2), the predicted means for time $t > 100$ capture an underlying trend in the time series.

Example 2 In a MAP and Bayesian approach, a prior on the AR coefficients can be used to define physical constraints (if any) or to regularise the system. Similarly, a prior on the variance r can be used to specify any knowledge about or constraint on the noise. As an example, consider a Bayesian approach to a first-order AR model in which the following Gaussian prior for a_1 and inverse Gamma prior for r are defined:

$$p(a_1) = \mathcal{N}(a_1 | 0, q) ,$$
$$p(r) = \mathcal{IG}(r|\nu, \nu/\beta) = \exp\left[-(\nu+1)\log r - \frac{\nu}{\beta r} - \log\Gamma(\nu) + \nu \log \frac{\nu}{\beta} \right] .$$

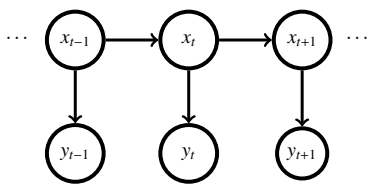


Figure 1.4 A first-order latent Markov model. In a hidden Markov model the latent variables $x_{1:T}$ are discrete and the observed variables $y_{1:T}$ can be either discrete or continuous.

Assuming that a_1 and r are a priori independent, the parameter posterior is given by

$$p(a_1, r|y_{1:T}) \propto p(a_1)p(r)\prod_{t=2}^{T} p(y_t|y_{t-1}, a_1, r).$$

The belief network representation of this model is given in Fig. 1.3(a). For a numerical example, consider $T = 2$ and observations and hyperparameters given by

$$y_1 = 1, \quad y_2 = -6, \quad q = 1.2, \quad \nu = 0.4, \quad \beta = 100.$$

The parameter posterior, Fig. 1.3(b), takes the form

$$p(a_1, r|y_{1:2}) \propto \exp\left[-\left(\frac{\nu}{\beta} + \frac{y_2^2}{2}\right)\frac{1}{r} + y_1 y_2 \frac{a_1}{r} - \frac{1}{2}\left(\frac{y_1^2}{r} + \frac{1}{q}\right)a_1^2 - (\nu + 3/2)\log r\right].$$

As we can see, Fig. 1.3(b), the posterior is multimodal, with each mode corresponding to a different interpretation: (i) The regression coefficient a_1 is approximately -6 and the noise is low. This solution gives a small prediction error. (ii) Since the prior for a_1 has zero mean, an alternative interpretation is that a_1 is centred around zero and the noise is high.

From this example we can make the following observations:

- Point estimates such as ML or MAP are not always representative of the solution.
- Even very simple models can lead to complicated posterior distributions.
- Variables that are independent *a priori* may become dependent *a posteriori*.
- Ambiguous data usually leads to a multimodal parameter posterior, with each mode corresponding to one plausible explanation.

1.3 Latent Markov models

In a latent Markov model, the observations $y_{1:T}$ are generated by a set of unobserved or 'latent' variables $x_{1:T}$. Typically, the latent variables are first-order Markovian and each observed variable y_t is independent from all other variables given x_t. The joint distribution thus factorises as[6]

$$p(y_{1:T}, x_{1:T}) = p(x_1)\prod_{t=2}^{T} p(y_t|x_t)p(x_t|x_{t-1}),$$

where $p(x_t|x_{t-1})$ is called the 'transition' model and $p(y_t|x_t)$ the 'emission' model. The belief network representation of this latent Markov model is given in Fig. 1.4.

[6] This general form is also known as a state space model.

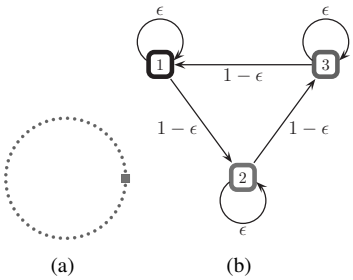


Figure 1.5 (a) Robot (square) moving sporadically with probability $1-\epsilon$ counter-clockwise in a circular corridor one location at a time. Small circles denote the S possible locations. (b) The state transition diagram for a corridor with $S = 3$ possible locations.

1.3.1 Discrete state latent Markov models

A well-known latent Markov model is the hidden Markov model[7] (HMM) [23] in which x_t is a scalar discrete variable ($x_t \in \{1, \ldots, S\}$).

Example Consider the following toy tracking problem. A robot is moving around a circular corridor and at any time occupies one of S possible locations. At each time step t, the robot stays where it is with probability ϵ, or moves to the next point in a counter-clockwise direction with probability $1-\epsilon$. This scenario, illustrated in Fig. 1.5, can be conveniently represented by an $S \times S$ matrix A with elements $A_{ji} = p(x_t = j | x_{t-1} = i)$. For example, for $S = 3$, we have

$$A = \epsilon \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} + (1-\epsilon) \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}. \tag{1.1}$$

At each time step t, the robot sensors measure its position, obtaining either the correct location with probability w or a uniformly random location with probability $1-w$. This can be expressed formally as

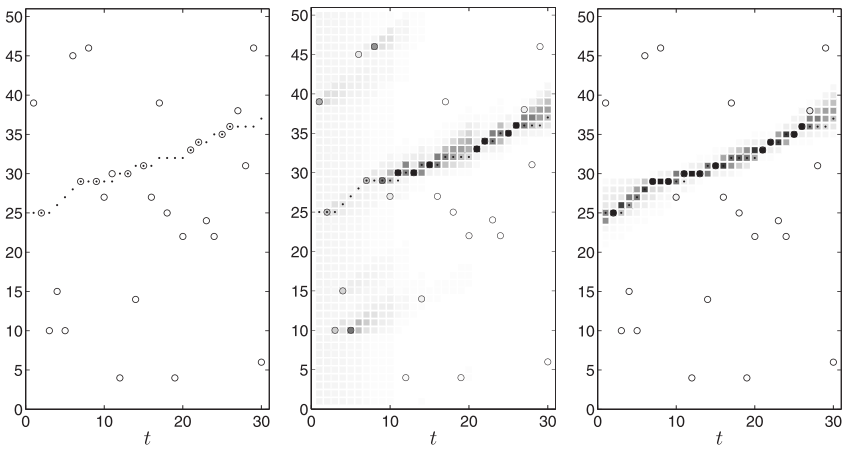
$$y_t | x_t \sim w\delta(y_t - x_t) + (1-w)\mathcal{U}(y_t | 1, \ldots, S),$$

where δ is the Kronecker delta function and $\mathcal{U}(y|1, \ldots, S)$ denotes the uniform distribution over the set of possible locations. We may parameterise $p(y_t|x_t)$ using an $S \times S$ matrix C with elements $C_{ui} = p(y_t = u | x_t = i)$. For $S = 3$, we have

$$C = w \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} + \frac{(1-w)}{3} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}.$$

A typical realisation $y_{1:T}$ from the process defined by this HMM with $S = 50$, $\epsilon = 0.5$, $T = 30$ and $w = 0.3$ is depicted in Fig. 1.6(a). We are interested in inferring the true locations of the robot from the noisy measured locations $y_{1:T}$. At each time t, the true location can be inferred from the so-called 'filtered' posterior $p(x_t|y_{1:t})$ (Fig. 1.6(b)), which uses measurements up to t; or from the so-called 'smoothed' posterior $p(x_t|y_{1:T})$ (Fig. 1.6(c)), which uses both past and future observations and is therefore generally more accurate. These posterior marginals are obtained using the efficient inference routines outlined in Section 1.4.

[7] Some authors use the terms 'hidden Markov model' and 'state space model' as synonymous [4]. We use the term HMM in a more restricted sense to refer to a latent Markov model where $x_{1:T}$ are discrete. The observations $y_{1:T}$ can be discrete or continuous.



(a) (b) (c)

Figure 1.6 Filtering and smoothing for robot tracking using a HMM with $S = 50$. (a) A realisation from the HMM example described in the text. The dots indicate the true latent locations of the robot, whilst the open circles indicate the noisy measured locations. (b) The squares indicate the filtering distribution at each time step t, $p(x_t|y_{1:t})$. This probability is proportional to the grey level with black corresponding to 1 and white to 0. Note that the posterior for the first time steps is multimodal, therefore the true position cannot be accurately estimated. (c) The squares indicate the smoothing distribution at each time step t, $p(x_t|y_{1:T} = y_{1:T})$. Note that, for $t < T$, we estimate the position retrospectively and the uncertainty is significantly lower when compared to the filtered estimates.

1.3.2 Continuous state latent Markov models

In continuous state latent Markov models, x_t is a multivariate continuous variable, $x_t \in \mathbb{R}^H$. For high-dimensional continuous x_t, the set of models for which operations such as filtering and smoothing are computationally tractable is severely limited. Within this tractable class, the linear dynamical system plays a special role, and is essentially the continuous analogue of the HMM.

Linear dynamical systems

A linear dynamical system (LDS) on variables $x_{1:T}, y_{1:T}$ has the following form:

$$x_t = Ax_{t-1} + \bar{x}_t + \epsilon_t^x, \quad \epsilon_t^x \sim \mathcal{N}(\epsilon_t^x|0, Q), \quad x_1 \sim \mathcal{N}(x_1|\mu, P),$$

$$y_t = Cx_t + \bar{y}_t + \epsilon_t^y, \quad \epsilon_t^y \sim \mathcal{N}\left(\epsilon_t^y|0, R\right),$$

with transition matrix A and emission matrix C. The terms $\bar{x}_t, \bar{y}_t$ are often defined as $\bar{x}_t = Bz_t$ and $\bar{y}_t = Dz_t$, where z_t is a known input that can be used to control the system. The complete parameter set is therefore $\{A, B, C, D, Q, R, \mu, P\}$. As a generative model, the LDS is defined as

$$p(x_t|x_{t-1}) = \mathcal{N}(x_t|Ax_{t-1} + \bar{x}_t, Q), \quad p(y_t|x_t) = \mathcal{N}(y_t|Cx_t + \bar{y}_t, R).$$

Example As an example scenario that can be modelled using an LDS, consider a moving object with position, velocity and instantaneous acceleration at time t given respectively by q_t, v_t and a_t. A discrete time description of the object dynamics is given by Newton's laws (see for example [11])