AALBORG UNIVERSITY

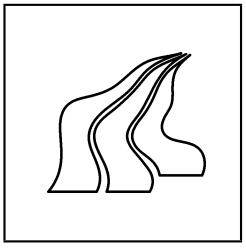
Learning dynamic Bayesian networks with mixed variables

by

Susanne G. Bøttcher

R-2005-23 June 2005

Department of Mathematical Sciences
Aalborg University
Fredrik Bajers Vej 7 G ▪ DK - 9220 Aalborg Øst ▪ Denmark
Phone: +45 96 35 80 80 ▪ Telefax: +45 98 15 81 29
URL: http://www.math.aau.dk



ISSN 1399–2503 ▪ On-line version ISSN 1601–7811

Learning Dynamic Bayesian Networks with Mixed Variables

Susanne G. Bøttcher
Aalborg University, Denmark

Abstract

This paper considers dynamic Bayesian networks for discrete and continuous variables. We only treat the case, where the distribution of the variables is conditional Gaussian. We show how to learn the parameters and structure of a dynamic Bayesian network and also how the Markov order can be learned. An automated procedure for specifying prior distributions for the parameters in a dynamic Bayesian network is presented. It is a simple extension of the procedure for the ordinary Bayesian networks. Finally the Wölfer's sunspot numbers are analyzed.

1 Introduction

In this paper we consider dynamic Bayesian networks (DBNs) for discrete and continuous variables. A DBN is an extension of an ordinary Bayesian network and is applied in the modeling of time series.

DBNs for first order Markov time series are described in Dean & Kanazawa (1989). In Murphy (2002), a thorough treatment of these models is presented and in Friedman, Murphy & Russell (1998) learning these networks in the case with only discrete variables is described.

Here we consider DBNs with both discrete and continuous variables. In these networks we also allow some of the variables to be static, *i.e.* some of the variables do not change over time. We only treat the case where the distribution of the variables is conditional Gaussian (CG) and show how to learn the parameters and structure of the DBN when data is complete. Further we present an automated method for specifying prior parameter distributions for the parameters in a DBN. These methods are simple extensions of the ones used for ordinary Bayesian networks with mixed variables, described in Bøttcher (2001).

We consider time series, where the Markov order can be higher than one and show how the Markov order can be learned.

In Section 2, DBNs with static and time varying variables are defined. Section 3 presents these DBNs for the mixed case and Section 4 gives some examples of some well known models that can be represented as DBNs. Section 5 shows how to learn the parameters and structure of a DBN with mixed variables. Further,

it shows how the Markov order can be learned. Section 6 presents a method for specifying prior distributions of the parameters in the DBN. In Section 7 Wölfer's sunspot numbers are analyzed using a DBN.

2 Dynamic Bayesian Networks

A *Bayesian network* is a graphical model that encodes the joint probability distribution for a set of variables. For terminology and theoretical aspects on graphical models, see Lauritzen (1996). We define it as a *directed acyclic graph* (DAG) $D = (V, E)$, where V is a finite set of nodes and E is a finite set of directed edges between the nodes. The DAG defines the structure of the Bayesian network. To each node $v \in V$ in the graph corresponds a random variable X_v. The set of variables associated with the graph D is then $X = (X_v)_{v \in V}$.

To each vertex v with parents $\text{pa}(v)$, there is attached a local probability distribution, $p(x_v|x_{\text{pa}(v)})$. The possible lack of directed edges in D encodes conditional independencies between the random variables X through the factorization of the joint probability distribution,

$$p(x) = \prod_{v \in V} p(x_v|x_{\text{pa}(v)}).$$

In a Bayesian network, the set of random variables X is fixed. To model a multivariate time series we need a framework, where we allow the set of random variables to vary with time. For this we use dynamic Bayesian networks, defined as below. This definition is consistent with the exposition in Murphy (2002), but here we also allow for static variables and Markov orders higher than one.

Let X^t be a set of time varying random variables, that is X^t can take on the values $X^0, X^1, \ldots, X^T$. We index the time varying variables by the non-negative integers to indicate that the observations are taken at discrete time points. The corresponding nodes in the graph are denoted V_t, so $X^t = (X^t_v)_{v \in V_t}$ for each time point t. Note however that V_t is "the same" for all time points t, so formally $V_t = \{(v, t), v \in V\}$. Further, let X^s be a set of static random variables, *i.e.* variables that do not change over time. The nodes corresponding to X^s are denoted V_s. The set of variables associated with a DBN is then $X = ((X^t)_{t=0}^T, X^s)$ and the set of nodes is $V = ((V_t)_{t=0}^T, V_s)$.

We refer to the time varying variables at one time point as a *time slice* or just a slice. We let the static variables X^s belong to the time slice at time $t = 0$ and refer to this as the initial time slice. So the initial time slice includes the variables X^0 and X^s and, for $t = 1, \ldots, T$, the time slice at time t includes the variables X^t.

We will mostly consider the variables in the initial time slice jointly, so to ease later notation we define $X^{\tilde{0}} = (X^0, X^s)$ and $V_{\tilde{0}} = (V_0, V_s)$.

The joint probability distribution of the variables in a dynamic Bayesian network can be very complex, as the number of variables grows over time. Therefore we assume that the time series we are dealing with, is *mth order Markov, i.e.*

$$p(x^t|x^{t-1},\dots,x^0) = p(x^t|x^{t-1},\dots,x^{t-m}),$$

for all time points $t = m,\dots,T$.

Further, we assume that the time series has *stationary dynamics*, so

$$p(x^t|x^{t-1},\dots,x^{t-m}) = p(x^m|x^{m-1},\dots,x^0),$$

for all $t = m,\dots,T$. Stationary dynamics refers to the fact that the conditional distributions are time independent, while the marginal distributions may be time dependent.

We will first introduce DBNs for time series that are first order Markov. With the above assumptions, a DBN for a first order Markov time series can be defined to be the pair $(B_{\tilde{0}}, B_{\rightarrow})$, where $B_{\tilde{0}}$ is a Bayesian network defining the probability distribution of $X^{\tilde{0}}$ as

$$p(x^{\tilde{0}}) = \prod_{v\in V_{\tilde{0}}} p(x_v^{\tilde{0}}|x^{\tilde{0}}_{\mathrm{pa}(v)}),$$

and $B_{\rightarrow}$ is a 2-slice temporal Bayesian network defining the conditional distribution of X^t as

$$p(x^t|x^{t-1},x^s) = \prod_{v\in V_t} p(x_v^t|x^t_{\mathrm{pa}(v)}, x^{t-1}_{\mathrm{pa}(v)}, x^s_{\mathrm{pa}(v)}).$$

The joint probability distribution for a DBN with $T+1$ time points is given as

$$p(x^0,\dots,x^T,x^s) = p(x^{\tilde{0}})\prod_{t=1}^{T} p(x^t|x^{t-1},x^s).$$

As we assumed that the time series has stationary dynamics, the DBN is completely specified through $B_{\tilde{0}}$ and $B_{\rightarrow}$.

For the dependency relations between the time slices we assume that arrows point forward in time, so the variables in time slice t can have parents in the time slices to time t and $t-1$. Further, they can have parents from X^s. Due to stationary dynamics, the dependency relations between the time slices are the same for all time points. This also means that if a time varying variable X_v^t has a static variable X_w^s as a parent, then X_w^s is also a parent of $X_v^1,\dots,X_v^T$. The variables in the initial time slice can have parents from the initial time slice and therefore also from X^s, as X^s is included in the initial time slice.

Within a time slice, there are no restrictions of the dependency relations between the variables, as long as the structure is a DAG. Due to stationary dynamics, the

dependency relations within a time slice are the same for the time slices to time $t = 1, \ldots, T$. They are however not necessarily the same as for the time varying variables in the initial time slice.

So the structure of the DBN repeats itself over time, except for $B_{\tilde{0}}$, where the time series is initialized.

Figure 1 shows an example of the structure of a a first order Markov DBN, $(B_{\tilde{0}}, B_{\rightarrow})$, with two time varying variables Y^t and Z^t and one static variable X^s. Because of the first order Markov property, the structure is completely specified through the first two time points and the structure of the DBN can therefore be represented by the DAG in Figure 2.

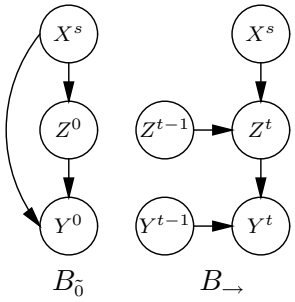


Figure 1: Example of a first order Markov DBN $(B_{\tilde{0}}, B_{\rightarrow})$.

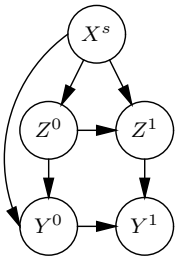


Figure 2: A first order Markov DBN $(B_{\tilde{0}}, B_{\rightarrow})$ represented by the first two time points.

For time series with higher Markov order properties, we need to extend the definition.

Consider an mth order Markov time series. The joint probability distribution for $T + 1$ time points can be written as

$$\begin{aligned} p(x^0, \ldots, x^T, x^s) &= p(x^{\tilde{0}}, x^1, \ldots, x^{m-1}) \prod_{t=m}^{T} p(x^t | x^{t-1}, \ldots, x^{t-m}, x^s) \\ &= p(x^{\tilde{0}}) p(x^1 | x^{\tilde{0}}) \cdots, p(x^{m-1} | x^{m-2}, \ldots, x^{\tilde{0}}) \\ &\times \prod_{t=m}^{T} p(x^t | x^{t-1}, \ldots, x^{t-m}, x^s). \end{aligned}$$

Following the definition for first order Markov time series, we let $B_{\rightarrow}$ be a $m+1$-slice temporal Bayesian network defining the conditional distribution of X^t,

$$p(x^t|x^{t-1},\ldots,x^{t-m},x^s) = \prod_{v\in V_t} p(x_v^t|x_{\mathrm{pa}(v)}^t,\ldots,x_{\mathrm{pa}(v)}^{t-m},x_{\mathrm{pa}(v)}^s),$$

for $t = m,\ldots,T$.

The variables in time slice t can have parents in the time slices to times $t,\ldots,t-m$ and they can have parents from X^s. Again, due to stationary dynamics, the dependency relations between and within the time slices are the same for all time points $t = m,\ldots,T$. Further, if a time varying variable X_v^t has a static variable X_w^s as a parent, then X_w^s is also a parent of $X_v^m,\ldots,X_v^T$.

The question is now how to initialize the time series. The probability distribution $p(x^{\tilde{0}},x^1,\ldots,x^{m-1})$ can be written as

$$p(x^{\tilde{0}},x^1,\ldots,x^{m-1}) = p(x^{\tilde{0}})p(x^1|x^{\tilde{0}})\cdots p(x^{m-1}|x^{m-2},\ldots,x^{\tilde{0}}). \qquad (1)$$

As arrows point forward in time, this factorization defines the possible dependency relations between the variables $X^{\tilde{0}},\ldots,X^{m-1}$. As before we let $B_{\tilde{0}}$ be a Bayesian network defining the probability distribution of $X^{\tilde{0}}$ as

$$p(x^{\tilde{0}}) = \prod_{v\in V_{\tilde{0}}} p(x_v^{\tilde{0}}|x_{\mathrm{pa}(v)}^{\tilde{0}}).$$

Now we also define Bayesian networks for the rest of the conditional distributions in (1). We let B_1 be a 2-slice Bayesian network defining the conditional distribution of X^1 given $X^{\tilde{0}}$ as

$$p(x^1|x^{\tilde{0}}) = \prod_{v\in V_1} p(x_v^1|x_{\mathrm{pa}(v)}^1,x_{\mathrm{pa}(v)}^{\tilde{0}}),$$

and likewise for $B_2,\ldots,B_{m-1}$, where B_{m-1} is an m-slice Bayesian network defining the conditional distribution of X^{m-1} given $X^{m-2},\ldots,X^{\tilde{0}}$ as

$$p(x^{m-1}|x^{m-2},\ldots,x^{\tilde{0}}) = \prod_{v\in V_{m-1}} p(x_v^{m-1}|x_{\mathrm{pa}(v)}^{m-1},\ldots,x_{\mathrm{pa}(v)}^{\tilde{0}}).$$

So the variables in the time slice to time $t=1$ can have parents from the time slice to time $t=1$ and $t=0$. The variables in time slice $m-1$ can have parents from the time slices to time $t=0,\ldots,m-1$. The dependency relations between the time slices to time $t=0,\ldots,m-1$ are obviously not the same and the dependency relations within these time slices are not necessarily the same.

The tuple $(B_{\tilde{0}},B_1,\ldots,B_{m-1},B_{\rightarrow})$ is thus a DBN for an mth order Markov time series, where the different Bayesian networks in the tuple defines the corresponding probability distributions as above. Notice that we could also just have specified the networks $B_{\tilde{0}},B_1,\ldots,B_{m-1}$ as one large network, with the necessary restrictions on the arrows.

3 Dynamic Bayesian Networks for Mixed Variables

In this section we consider DBNs with *mixed variables*, *i.e.* the variables in the network can be of discrete and continuous type. We let $V = \Delta \cup \Gamma$, where Δ and Γ are the sets of discrete and continuous variables, respectively. The corresponding random variables X can then be denoted $X = (X_v)_{v \in V} = (I, Y) = ((I_\delta)_{\delta \in \Delta}, (Y_\gamma)_{\gamma \in \Gamma})$. Again, we index the sets of nodes and the random variables with t for time varying variables, s for static variables and $\tilde{0}$ for the variables in the initial time slice.

To ensure availability of exact local computation methods, we do not allow continuous parents of discrete nodes, so the probability distributions factorize into a discrete part and a mixed part as presented below. To simplify notation, we present the theory for first order Markov time series and comment on how to extend it to higher order Markov assumptions by following the definitions introduced in the previous section.

We consider $B_{\tilde{0}}$ and $B_{\rightarrow}$ separately, and the joint probability distribution is obtained as specified in the previous section.

For $B_{\tilde{0}}$ we have that

$$\begin{aligned} p(x^{\tilde{0}}) &= \prod_{v \in V_{\tilde{0}}} p(x_v^{\tilde{0}} | x_{\mathrm{pa}(v)}^{\tilde{0}}) \\ &= \prod_{\delta \in \Delta_{\tilde{0}}} p(i_\delta^{\tilde{0}} | i_{\mathrm{pa}(\delta)}^{\tilde{0}}) \prod_{\gamma \in \Gamma_{\tilde{0}}} p(y_\gamma^{\tilde{0}} | i_{\mathrm{pa}(\gamma)}^{\tilde{0}}, y_{\mathrm{pa}(\gamma)}^{\tilde{0}}) \end{aligned} \tag{2}$$

and for $B_{\rightarrow}$

$$\begin{aligned} p(x^t | x^{t-1}, x^s) &= \prod_{v \in V_t} p(x_v^t | x_{\mathrm{pa}(v)}^t, x_{\mathrm{pa}(v)}^{t-1}, x_{\mathrm{pa}(v)}^s) \\ &= \prod_{\delta \in \Delta_t} p(i_\delta^t | i_{\mathrm{pa}(\delta)}^t, i_{\mathrm{pa}(\delta)}^{t-1}, i_{\mathrm{pa}(\delta)}^s) \\ &\times \prod_{\gamma \in \Gamma_t} p(y_\gamma^t | i_{\mathrm{pa}(\gamma)}^t, i_{\mathrm{pa}(\gamma)}^{t-1}, i_{\mathrm{pa}(\gamma)}^s, y_{\mathrm{pa}(\gamma)}^t, y_{\mathrm{pa}(\gamma)}^{t-1}, y_{\mathrm{pa}(\gamma)}^s). \end{aligned} \tag{3}$$

To account for higher order Markov assumptions, we would just have to specify the probability distributions for the intervening networks accordingly.

To simplify notation for $B_{\rightarrow}$, we use the following notation, where the possible parent configurations are not explicitly defined. They must be specified in the given context and according to (3).

$$\begin{aligned} p(x^t | x^{t-1}, x^s) &= \prod_{v \in V_t} p(x_v^t | x_{\mathrm{pa}(v)}^{\rightarrow}) \\ &= \prod_{\delta \in \Delta_t} p(i_\delta^t | i_{\mathrm{pa}(\delta)}^{\rightarrow}) \prod_{\gamma \in \Gamma_t} p(y_\gamma^t | i_{\mathrm{pa}(\gamma)}^{\rightarrow}, y_{\mathrm{pa}(\gamma)}^{\rightarrow}). \end{aligned}$$

So for example, $i^{\rightarrow}_{\text{pa}(\delta)}$ contains the variables $i^{t}_{\text{pa}(\delta)}$, $i^{t-1}_{\text{pa}(\delta)}$ and $i^{s}_{\text{pa}(\delta)}$.

In this paper we only consider networks, where the joint distribution of the variables is conditional Gaussian. The local probability distributions are therefore defined as in the following two sections. In these sections, we do not distinguish between the variables in $B_{\tilde{0}}$ and $B_{\rightarrow}$, as the distribution of these variables is of the same type. The possible parent set differ however between variables in $B_{\tilde{0}}$ and variables in $B_{\rightarrow}$. In the following we therefore just denote the parents of a variable x_v by $x_{\text{pa}(v)}$ and $x_{\text{pa}(v)}$ must be specified according to (2) or (3).

3.1 Distribution for Discrete Variables

When the joint distribution is conditional Gaussian, the local probability distributions for the discrete variables are just unrestricted discrete distributions with

$$p(i_\delta | i_{\text{pa}(\delta)}) \geq 0 \quad \forall \quad \delta \in \Delta.$$

We parameterize this as

$$\theta_{i_\delta | i_{\text{pa}(\delta)}} = p(i_\delta | i_{\text{pa}(\delta)}, \theta_{\delta | i_{\text{pa}(\delta)}}),$$

where $\theta_{\delta | i_{\text{pa}(\delta)}} = (\theta_{i_\delta | i_{\text{pa}(\delta)}})_{i_\delta \in \mathcal{I}_\delta}$.

Furthermore $\sum_{i_\delta \in \mathcal{I}_\delta} \theta_{i_\delta | i_{\text{pa}(\delta)}} = 1$ and $0 \leq \theta_{i_\delta | i_{\text{pa}(\delta)}} \leq 1$. All parameters associated with a node δ is denoted by θ_δ, so $\theta_\delta = (\theta_{\delta | i_{\text{pa}(\delta)}})_{i_{\text{pa}(\delta)} \in \mathcal{I}_{\text{pa}(\delta)}}$.

3.2 Distribution for Continuous Variables

For the continuous variables, the local probability distributions are Gaussian linear regressions with parameters depending on the configuration of the discrete parents. So let the parameters be given by $\theta_{\gamma | i_{\text{pa}(\gamma)}} = (m_{\gamma | i_{\text{pa}(\gamma)}}, \beta_{\gamma | i_{\text{pa}(\gamma)}}, \sigma^2_{\gamma | i_{\text{pa}(\gamma)}})$. Then

$$(Y_\gamma | y_{\text{pa}(\gamma)}, i_{\text{pa}(\gamma)}, \theta_{\gamma | i_{\text{pa}(\gamma)}}) \sim \mathcal{N}(m_{\gamma | i_{\text{pa}(\gamma)}} + \beta_{\gamma | i_{\text{pa}(\gamma)}} y_{\text{pa}(\gamma)}, \sigma^2_{\gamma | i_{\text{pa}(\gamma)}}), \qquad (4)$$

where $\beta_{\gamma | i_{\text{pa}(\gamma)}}$ are the regression coefficients, $m_{\gamma | i_{\text{pa}(\gamma)}}$ is the regression intercept, and $\sigma^2_{\gamma | i_{\text{pa}(\gamma)}}$ is the conditional variance. Thus for each configuration of the discrete parents of γ the distribution of Y_γ is Gaussian with mean and variance given as in (4). The parameters associated with a node γ is then $\theta_\gamma = (\theta_{\gamma | i_{\text{pa}(\gamma)}})_{i_{\text{pa}(\gamma)} \in \mathcal{I}_{\text{pa}(\gamma)}}$.

3.3 The Parameterized Distributions

With the above distributional assumptions, we can specify the parameterized DBN as follows.

Let $\theta^{\tilde{0}} = ((\theta^{\tilde{0}}_{\delta})_{\delta\in\Delta_{\tilde{0}}}, (\theta^{\tilde{0}}_{\gamma})_{\gamma\in\Gamma_{\tilde{0}}})$ and $\theta^{\rightarrow} = ((\theta^{\rightarrow}_{\delta})_{\delta\in\Delta_t}, (\theta^{\rightarrow}_{\gamma})_{\gamma\in\Gamma_t})$. Further, let $\theta = (\theta^{\tilde{0}}, \theta^{\rightarrow})$. Then $B_{\tilde{0}}$ is given as

$$p(x^{\tilde{0}}|\theta^{\tilde{0}}) = \prod_{\delta\in\Delta_{\tilde{0}}} p(i^{\tilde{0}}_{\delta}|i^{\tilde{0}}_{\mathrm{pa}(\delta)}, \theta^{\tilde{0}}_{\delta|i_{\mathrm{pa}(\delta)}}) \prod_{\gamma\in\Gamma_{\tilde{0}}} p(y^{\tilde{0}}_{\gamma}|i^{\tilde{0}}_{\mathrm{pa}(\gamma)}, y^{\tilde{0}}_{\mathrm{pa}(\gamma)}, \theta^{\tilde{0}}_{\gamma|i_{\mathrm{pa}(\gamma)}}),$$

and $B_{\rightarrow}$ as

$$\begin{aligned} p(x^t|x^{t-1}, x^s, \theta^{\rightarrow}) &= \prod_{\delta\in\Delta_t} p(i^t_{\delta}|i^{\rightarrow}_{\mathrm{pa}(\delta)}, \theta^{\rightarrow}_{\delta|i_{\mathrm{pa}(\delta)}}) \\ &\times \prod_{\gamma\in\Gamma_t} p(y^t_{\gamma}|y^{\rightarrow}_{\mathrm{pa}(\gamma)}, i^{\rightarrow}_{\mathrm{pa}(\gamma)}, \theta^{\rightarrow}_{\gamma|i_{\mathrm{pa}(\gamma)}}). \end{aligned}$$

The joint distribution for $T+1$ time points is given as

$$p(x^0, \ldots, x^T, x^s, \theta) = p(x^{\tilde{0}}|\theta^{\tilde{0}}) \prod_{t=1}^{T} p(x^t|x^{t-1}, x^s, \theta^{\rightarrow}).$$

Notice that, due to stationarity, $\theta^{\rightarrow}$ is the parameter in the conditional distribution of x^t for *every* time point $t = 1, \ldots, T$.

4 Examples of DBNs

We will now give some examples of some well known models that can be represented as DBNs. In the figures, shaded nodes represent discrete variables and clear nodes represent continuous variables.

4.1 Hidden Markov Models

A Hidden Markov Model (HMM) is a stochastic automaton, where each state generates an observation. Figure 3 shows a HMM, where the hidden states are first order Markov.

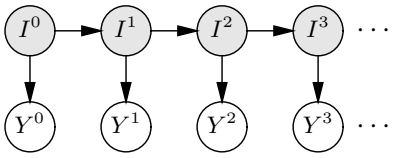


Figure 3: A Hidden Markov Model.

The hidden states, *i.e.* the discrete hidden variables, are denoted by I and the observations by Y. We have represented the observed variables as continuous, but they can also all be discrete. In this HMM, I^{t+1} is conditionally independent of I^{t-1}, given I^t. Further, Y^t is conditionally independent of the rest of the variables in the network, given I^t. A model like this is used in situations, where the observations do not follow the same model all the time, but can follow different models at different times. This gives for example the possibility to account for outliers.

When a HMM is represented as a DBN, we assume that the time series has stationary dynamics. So, together with the first order Markov property, we can specify the joint probability distribution for the variables in this network by just specifying the initial prior probabilities $p(i^0)$, the transition probabilities $p(i^t|i^{t-1})$ and the conditional Gaussian distributions $p(y^t|i^t)$ (or, if the observed variables are discrete, the conditional multinomial distributions $p(j^t|i^t)$).

There are many variants of this basic HMM, *e.g.* Buried Markov Model, Mixed-memory Markov Model and Hierarchical HMM, see Murphy (2002) for a presentation of these models represented as DBNs and their application within speech recognition.

4.2 Kalman Filter Models

A Kalman Filter Model (KFM), introduced by Harrison & Stevens (1976) as a state space model, models the dynamic behavior of a time series. In such a model, the continuous observations Y are indirect measurements of a latent Markov process Z.

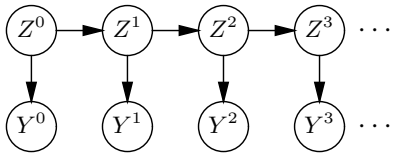


Figure 4: A Kalman Filter Model.

In Figure 4, a KFM is shown. The structure is the same as for the HMM, since the two models assume the same set of conditional independencies. The probability distributions to be specified is the Gaussian distribution $p(z^0)$, the Gaussian linear regression $p(z^t|z^{t-1})$ and the Gaussian linear regression $p(y^t|z^t)$. For a comprehensive treatment of KFMs and their applications, see West & Harrison (1989).

4.3 Multiprocess Kalman Filter Models

Multiprocess Kalman Filter Models (MKFMs), also known as switching state space Markov models, are an extension of the KFMs, see Harrison & Stevens (1976),

where the aim is to discriminate between different KFMs.

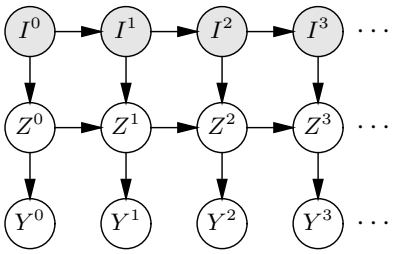


Figure 5: A Multiprocess Kalman Filter Model.

Figure 5 shows a MKFM. Again we see that the continuous observations Y are indirect measurements of a latent continuous Markov process Z, *i.e.* this part of the network represents a KFM. In addition, the process Z depends on the hidden states I, which in our example are first order Markov. Like the HMM, this model can be used in situations, where the observations do not follow the same model all the time, but can follow different models at different times, but here the models are KFMs. Applications include modeling piece-wise linear time series, which for example can be used for monitoring purposes, see *e.g.* Bøttcher, Milsgaard & Mortensen (1995).

Notice that because of the first order Markov property assumed for HMMs, KFMs and MKFMs, these models could have been represented by using only the first two time points, as the structure repeats over time.

4.4 Vector Autoregressive processes

Another classical time series model is the Vector Autoregressive process (VAR) of Markov order p. This model is equivalent to a DBN of Markov order p, in which all the variables are continuous and observed. So the local probability distributions in this model are Gaussian linear regressions on the continuous parents.

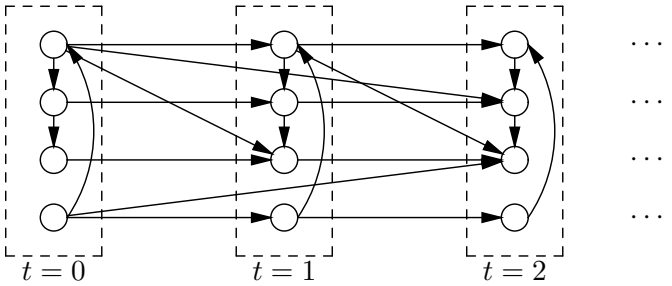


Figure 6: A Vector Autoregressive process.

In Figure 6, an example of a VAR process of order 2 is given. Because of the second order Markov property, this model can be represented by the first three time points.

In the next section, we will developed a method for learning the parameters and structure of a DBN. In this paper we assume that data are complete, so we can not learn networks with hidden variables. Therefore, the HMM, the KFM and the MKFM can only be learned with these methods, if a training dataset with complete data is available.

5 Learning DBNs with Mixed Variables

Learning first order Markov DBNs in the purely discrete case with no static variables is described in Friedman et al. (1998). Here we will consider learning DBNs with mixed variables for the case with both time varying and static variables. Further, we will also illustrate how to learn DBNs with higher Markov order and how to learn this order.

As noted in Murphy (2002), learning DBNs is, because of the way DBNs are defined, just a simple extension of learning BNs. This also applies for DBNs with mixed variables, so we will use the theory for learning Bayesian networks with mixed variables, described in Bøttcher (2001).

5.1 Parameter Learning

To learn the parameters for a given DAG, we use a Bayesian approach. We specify a prior distribution of a parameter θ, use a random sample d from the probability distribution $p(x|\theta)$ and obtain the posterior distribution by using Bayes' theorem

$$p(\theta|d) \propto p(d|\theta)p(\theta).$$

The proportionality constant is determined by the relation $\int_\Theta p(\theta|d)d\theta = 1$, where Θ is the parameter space.

To obtain closed formed expressions, we use conjugate distributions of the parameters.

We assume that the parameters associated with $B_{\tilde{0}}$ and $B_{\rightarrow}$ are independent. Further, for the parameters in respectively $B_{\tilde{0}}$ and $B_{\rightarrow}$, we assume that the parameters associated with one variable is independent of the parameters associated with the other variables and that the parameters are independent for each configuration of the discrete parents, *i.e.*

$$\begin{aligned} p(\theta) &= p(\theta^{\tilde{0}})p(\theta^{\rightarrow}) \\ &= \prod_{\delta\in\Delta_{\tilde{0}}}\prod_{i_{\mathrm{pa}(\delta)}\in\mathcal{I}_{\mathrm{pa}(\delta)}} p(\theta^{\tilde{0}}_{\delta|i_{\mathrm{pa}(\delta)}}) \prod_{\gamma\in\Gamma_{\tilde{0}}}\prod_{i_{\mathrm{pa}(\gamma)}\in\mathcal{I}_{\mathrm{pa}(\gamma)}} p(\theta^{\tilde{0}}_{\gamma|i_{\mathrm{pa}(\gamma)}}) \\ &\times \prod_{\delta\in\Delta_t}\prod_{i_{\mathrm{pa}(\delta)}\in\mathcal{I}_{\mathrm{pa}(\delta)}} p(\theta^{\rightarrow}_{\delta|i_{\mathrm{pa}(\delta)}}) \prod_{\gamma\in\Gamma_t}\prod_{i_{\mathrm{pa}(\gamma)}\in\mathcal{I}_{\mathrm{pa}(\gamma)}} p(\theta^{\rightarrow}_{\gamma|i_{\mathrm{pa}(\gamma)}}). \end{aligned} \tag{5}$$

We refer to this as *parameter independence*. Notice though that it is slightly different than parameter independence for ordinary Bayesian networks, as we here assume that the parameters in $B_{\rightarrow}$ are the same for each time point $t = 1, \ldots, T$.

In the case with higher order Markov properties, parameter independence is also valid for the parameters in the networks $B_1, \ldots, B_{m-1}$.

We also assume *complete data, i.e.* each case ${}^{c}x$ in a dataset d contains one instance of every random variable in the network. With this we can show posterior parameter independence. The likelihood $p(d|\theta)$ can be written as follows.

$$\begin{aligned} p(d|\theta) &= \prod_{c \in d} p({}^{c}x^0, \ldots, {}^{c}x^T, {}^{c}x^s|\theta) \\ &= \prod_{c \in d} \left(p({}^{c}x^{\tilde{0}}|\theta^{\tilde{0}}) \prod_{t=1}^{T} p({}^{c}x^t|{}^{c}x^{t-1}, {}^{c}x^s, \theta^{\rightarrow}) \right). \end{aligned}$$

As the time series has stationary dynamics, we see that for each observations of the variables in B_0, there are T observations of the variables in $B_{\rightarrow}$.

To simplify the expressions, we consider the likelihood terms for $B_{\tilde{0}}$ and $B_{\rightarrow}$ separately. For $B_{\tilde{0}}$ we have that

$$\prod_{c \in d} p({}^{c}x^{\tilde{0}}|\theta^{\tilde{0}}) = \prod_{c \in d} \prod_{\delta \in \Delta_{\tilde{0}}} p({}^{c}i_{\delta}^{\tilde{0}}|{}^{c}i_{\mathrm{pa}(\delta)}^{\tilde{0}}, \theta_{\delta|i_{\mathrm{pa}(\delta)}}^{\tilde{0}}) \prod_{\gamma \in \Gamma_{\tilde{0}}} p({}^{c}y_{\gamma}^{\tilde{0}}|{}^{c}y_{\mathrm{pa}(\gamma)}^{\tilde{0}}, {}^{c}i_{\mathrm{pa}(\gamma)}^{\tilde{0}}, \theta_{\gamma|i_{\mathrm{pa}(\gamma)}}^{\tilde{0}}),$$

where ${}^{c}i$ and ${}^{c}y$ respectively denotes the discrete part and the continuous part of a case ${}^{c}x$. Our goal is to show posterior parameter independence, so we must show that the likelihood, like the parameters, factorizes into a product over nodes and a product over the configuration of the discrete parents of a node. Therefore we write this part of the likelihood as

$$\begin{aligned} \prod_{c \in d} p({}^{c}x^{\tilde{0}}|\theta^{\tilde{0}}) = &\prod_{\delta \in \Delta} \prod_{i_{\mathrm{pa}(\delta)} \in \mathcal{I}_{\mathrm{pa}(\delta)}} \prod_{c : {}^{c}i_{\mathrm{pa}(\delta)}^{\tilde{0}} = i_{\mathrm{pa}(\delta)}^{\tilde{0}}} p({}^{c}i_{\delta}^{\tilde{0}}|i_{\mathrm{pa}(\delta)}^{\tilde{0}}, \theta_{\delta|i_{\mathrm{pa}(\delta)}}^{\tilde{0}}) \\ &\times \prod_{\gamma \in \Gamma} \prod_{i_{\mathrm{pa}(\gamma)} \in \mathcal{I}_{\mathrm{pa}(\gamma)}} \prod_{c : {}^{c}i_{\mathrm{pa}(\gamma)}^{\tilde{0}} = i_{\mathrm{pa}(\gamma)}^{\tilde{0}}} p({}^{c}y_{\gamma}^{\tilde{0}}|{}^{c}y_{\mathrm{pa}(\gamma)}^{\tilde{0}}, i_{\mathrm{pa}(\gamma)}^{\tilde{0}}, \theta_{\gamma|i_{\mathrm{pa}(\gamma)}}^{\tilde{0}}). \end{aligned} \tag{6}$$

We see that the product over cases is split up into a product over the configurations of the discrete parents and a product over those cases, where the configuration of the discrete parents is the same as the currently processed configuration. Notice however that some of the parent configurations might not be represented in the database, in which case the product over cases with this parent configuration just adds nothing to the overall product.

In the case with mth order Markov properties, the likelihood terms for all the networks $B_1, \ldots, B_{m-1}$, can be written as in (6).

The likelihood part from $B_{\rightarrow}$ is given as,

$$\begin{aligned}
&\prod_{c\in d}\prod_{t=1}^{T} p({}^{c}x^{t}|{}^{c}x^{t-1}, {}^{c}x^{s}, \theta^{\rightarrow}) \\
&= \prod_{c\in d}\prod_{t=1}^{T}\left(\prod_{\delta\in\Delta_t} p({}^{c}i_{\delta}^{t}|{}^{c}i_{\mathrm{pa}(\delta)}^{\rightarrow}, \theta_{\delta|i_{\mathrm{pa}(\delta)}}^{\rightarrow})\prod_{\gamma\in\Gamma_t} p({}^{c}y_{\gamma}^{t}|{}^{c}y_{\mathrm{pa}(\gamma)}^{\rightarrow}, {}^{c}i_{\mathrm{pa}(\gamma)}^{\rightarrow}, \theta_{\gamma|i_{\mathrm{pa}(\gamma)}}^{\rightarrow})\right) \\
&= \prod_{\delta\in\Delta_t}\prod_{i_{\mathrm{pa}(\delta)}^{\rightarrow}\in\mathcal{I}_{\mathrm{pa}(\delta)}^{\rightarrow}}\prod_{t=1}^{T}\prod_{c:{}^{c}i_{\mathrm{pa}(\delta)}^{\rightarrow}=i_{\mathrm{pa}(\delta)}^{\rightarrow}} p({}^{c}i_{\delta}^{t}|i_{\mathrm{pa}(\delta)}^{\rightarrow}, \theta_{\delta|i_{\mathrm{pa}(\delta)}}^{\rightarrow}) \qquad (7) \\
&\times \prod_{\gamma\in\Gamma}\prod_{i_{\mathrm{pa}(\gamma)}^{\rightarrow}\in\mathcal{I}_{\mathrm{pa}(\gamma)}^{\rightarrow}}\prod_{t=1}^{T}\prod_{c:{}^{c}i_{\mathrm{pa}(\gamma)}^{\rightarrow}=i_{\mathrm{pa}(\gamma)}^{\rightarrow}} p({}^{c}y_{\gamma}^{t}|{}^{c}y_{\mathrm{pa}(\gamma)}^{\rightarrow}, i_{\mathrm{pa}(\gamma)}^{\rightarrow}, \theta_{\gamma|i_{\mathrm{pa}(\gamma)}}^{\rightarrow})
\end{aligned}$$

The product over cases is split up as before. Further, this is also a product over time points, so for each time point t, we take the product over cases with a specific configuration of the discrete parents.

Posterior parameter independence now follows from (5), (6) and (7),

$$\begin{aligned}
p(\theta|d) &= p(\theta^{\tilde{0}}|d)p(\theta^{\rightarrow}|d) \\
&= \prod_{\delta\in\Delta_{\tilde{0}}}\prod_{i_{\mathrm{pa}(\delta)}\in\mathcal{I}_{\mathrm{pa}(\delta)}} p(\theta_{\delta|i_{\mathrm{pa}(\delta)}}^{\tilde{0}}|d)\prod_{\gamma\in\Gamma_{\tilde{0}}}\prod_{i_{\mathrm{pa}(\gamma)}\in\mathcal{I}_{\mathrm{pa}(\gamma)}} p(\theta_{\gamma|i_{\mathrm{pa}(\gamma)}}^{\tilde{0}}|d) \\
&\times \prod_{\delta\in\Delta_t}\prod_{i_{\mathrm{pa}(\delta)}\in\mathcal{I}_{\mathrm{pa}(\delta)}} p(\theta_{\delta|i_{\mathrm{pa}(\delta)}}^{\rightarrow}|d)\prod_{\gamma\in\Gamma_t}\prod_{i_{\mathrm{pa}(\gamma)}\in\mathcal{I}_{\mathrm{pa}(\gamma)}} p(\theta_{\gamma|i_{\mathrm{pa}(\gamma)}}^{\rightarrow}|d).
\end{aligned}$$

So due to parameter independence and complete data, the parameters stay independent given data. This means that we can learn the parameters in the local distributions independently and also that the parameters in $B_{\tilde{0}}$ and $B_{\rightarrow}$ can be learned independently. Again, if the time series is mth order Markov, posterior parameter independence also follows and we can learn the parameters in $B_{\tilde{0}}, \ldots, B_{m-1}$ and $B_{\rightarrow}$ independently.

Consider for example in $B_{\tilde{0}}$ a parameter for a discrete node δ, with a specific configuration of the discrete parents, $i_{\mathrm{pa}(\delta)}$. The posterior distribution of $\theta_{\delta|i_{\mathrm{pa}(\delta)}}^{\tilde{0}}$ is by Bayes' theorem found as

$$p(\theta_{\delta|i_{\mathrm{pa}(\delta)}}^{\tilde{0}}|d) \quad \propto \prod_{c:{}^{c}i_{\mathrm{pa}(\delta)}^{\tilde{0}}=i_{\mathrm{pa}(\delta)}^{\tilde{0}}} p({}^{c}i_{\delta}^{\tilde{0}}|i_{\mathrm{pa}(\delta)}^{\tilde{0}}, \theta_{\delta|i_{\mathrm{pa}(\delta)}}^{\tilde{0}})p(\theta_{\delta|i_{\mathrm{pa}(\delta)}}^{\tilde{0}}).$$

Thus $\theta^{\tilde{0}}_{\delta|i_{\mathrm{pa}(\delta)}}$ is updated with the cases in the database, where the configuration of the parents of δ is $i^{\tilde{0}}_{\mathrm{pa}(\delta)}$.

Likewise with a parameter $\theta^{\rightarrow}_{\delta|i_{\mathrm{pa}(\delta)}}$ in $B_{\rightarrow}$,

$$p(\theta^{\rightarrow}_{\delta|i_{\mathrm{pa}(\delta)}}|d) \quad \propto \quad \prod_{t=1}^{T} \prod_{c:c^{\rightarrow}_{i_{\mathrm{pa}(\delta)}}=i^{\rightarrow}_{\mathrm{pa}(\delta)}} p(c^{t}_{i_\delta}|i^{\rightarrow}_{\mathrm{pa}(\delta)}, \theta^{\rightarrow}_{\delta|i_{\mathrm{pa}(\delta)}}) p(\theta^{\rightarrow}_{\delta|i_{\mathrm{pa}(\delta)}}).$$

Here $\theta^{\rightarrow}_{\delta|i_{\mathrm{pa}(\delta)}}$ is, for each time point t, updated with the cases in the database for which the configuration of the parents of δ is $i^{\rightarrow}_{\mathrm{pa}(\delta)}$.

In the next sections we will introduce the conjugate distributions of the parameters and show how these are learned. The only difference in how the parameters in $B_{\tilde{0}}$ and $B_{\rightarrow}$ are learned, is the set of cases used to learn them. So in the following we do not differentiate between the parameters in $B_{\tilde{0}}$ and $B_{\rightarrow}$.

5.2 Learning the Discrete Variables

As described in DeGroot (1970), a conjugate family for multinomial observations is the family of Dirichlet distributions. Let the prior distribution of $\theta_{\delta|i_{\mathrm{pa}(\delta)}}$ be a Dirichlet distribution, $\mathcal{D}$, with hyperparameters $\alpha_{\delta|i_{\mathrm{pa}(\delta)}} = (\alpha_{i_\delta|i_{\mathrm{pa}(\delta)}})_{i_\delta \in \mathcal{I}_\delta}$, also written as

$$(\theta_{\delta|i_{\mathrm{pa}(\delta)}}|\alpha_{\delta|i_{\mathrm{pa}(\delta)}}) \sim \mathcal{D}(\alpha_{\delta|i_{\mathrm{pa}(\delta)}}).$$

The posterior distribution is then given as

$$(\theta_{\delta|i_{\mathrm{pa}(\delta)}}|d) \sim \mathcal{D}(\alpha_{\delta|i_{\mathrm{pa}(\delta)}} + n_{\delta|i_{\mathrm{pa}(\delta)}}),$$

where the vector $n_{\delta|i_{\mathrm{pa}(\delta)}} = (n_{i_\delta|i_{\mathrm{pa}(\delta)}})_{i_\delta \in \mathcal{I}_\delta}$, also called the counts, denotes the number of observations in d where δ and $\mathrm{pa}(\delta)$ have that specific configuration.

Again $\alpha_{\delta|i_{\mathrm{pa}(\delta)}}$ and $n_{\delta|i_{\mathrm{pa}(\delta)}}$ can be indexed by $\tilde{0}$ and $\rightarrow$, according to $B_{\tilde{0}}$ and $B_{\rightarrow}$. So for $B_{\tilde{0}}$ we have that $n^{\tilde{0}}_{i_\delta|i_{\mathrm{pa}(\delta)}}$ is the number of cases in d with a given configuration of δ and $\mathrm{pa}(\delta)$. Likewise for $B_{\rightarrow}$, where $n^{\rightarrow}_{i_\delta|i_{\mathrm{pa}(\delta)}}$ is the number of cases in d and for every time point $t = 1, \ldots, T$, with this configuration of δ and $\mathrm{pa}(\delta)$.

5.3 Learning the Continuous Variables

For the continuous variables we can write the local probability distributions as

$$(Y_\gamma|y_{\mathrm{pa}(\gamma)}, i_{\mathrm{pa}(\gamma)}, \theta_{\gamma|i_{\mathrm{pa}(\gamma)}}) \sim \mathcal{N}(z_{\mathrm{pa}(\gamma)}(m_{\gamma|i_{\mathrm{pa}(\gamma)}}, \beta_{\gamma|i_{\mathrm{pa}(\gamma)}})^{\mathrm{T}}, \sigma^2_{\gamma|i_{\mathrm{pa}(\gamma)}}),$$

where $z_{\text{pa}(\gamma)} = (1, y_{\text{pa}(\gamma)})$. A standard conjugate family for these observations is the family of Gaussian-inverse gamma distributions. Let the prior joint distribution of $(m_{\gamma|i_{\text{pa}(\gamma)}}, \beta_{\gamma|i_{\text{pa}(\gamma)}})$ and $\sigma^2_{\gamma|i_{\text{pa}(\gamma)}}$ be as follows.

$$\begin{aligned}(m_{\gamma|i_{\text{pa}(\gamma)}}, \beta_{\gamma|i_{\text{pa}(\gamma)}}|\sigma^2_{\gamma|i_{\text{pa}(\gamma)}}) &\sim \mathcal{N}_{k+1}(\mu_{\gamma|i_{\text{pa}(\gamma)}}, \sigma^2_{\gamma|i_{\text{pa}(\gamma)}}\tau^{-1}_{\gamma|i_{\text{pa}(\gamma)}}),\\ (\sigma^2_{\gamma|i_{\text{pa}(\gamma)}}) &\sim \mathcal{I}\Gamma\left(\frac{\rho_{\gamma|i_{\text{pa}(\gamma)}}}{2}, \frac{\phi_{\gamma|i_{\text{pa}(\gamma)}}}{2}\right).\end{aligned}$$

If $\theta_{\gamma|i_{\text{pa}(\gamma)}}$ is a parameter in $B_{\tilde{0}}$, the posterior distribution is found by

$$p(\theta^{\tilde{0}}_{\gamma|i_{\text{pa}(\gamma)}}|d) \propto \prod_{c:{}^c i^{\tilde{0}}_{\text{pa}(\gamma)} = i^{\tilde{0}}_{\text{pa}(\gamma)}} p({}^c y^{\tilde{0}}_{\gamma}|{}^c y^{\tilde{0}}_{\text{pa}(\gamma)}, i^{\tilde{0}}_{\text{pa}(\gamma)}, \theta^{\tilde{0}}_{\gamma|i_{\text{pa}(\gamma)}}) p(\theta^{\tilde{0}}_{\gamma|i_{\text{pa}(\gamma)}}).$$

We now join all the observations ${}^c y^{\tilde{0}}_{\gamma}$ for which ${}^c i^{\tilde{0}}_{\text{pa}(\gamma)} = i^{\tilde{0}}_{\text{pa}(\gamma)}$ in a vector ${}^b y^{\tilde{0}}_{\gamma}$, *i.e.* ${}^b y^{\tilde{0}}_{\gamma} = ({}^c y^{\tilde{0}}_{\gamma})_{{}^c i^{\tilde{0}}_{\text{pa}(\gamma)} = i^{\tilde{0}}_{\text{pa}(\gamma)}}$.

The same is done with the observations of the continuous parents of γ, *i.e.* ${}^b y^{\tilde{0}}_{\text{pa}(\gamma)} = ({}^c y^{\tilde{0}}_{\text{pa}(\gamma)})_{{}^c i^{\tilde{0}}_{\text{pa}(\gamma)} = i^{\tilde{0}}_{\text{pa}(\gamma)}}$. The posterior distribution of $\theta_{\gamma|i_{\text{pa}(\gamma)}}$ can now be written as

$$p(\theta^{\tilde{0}}_{\gamma|i_{\text{pa}(\gamma)}}|d) \propto p({}^b y^{\tilde{0}}_{\gamma}|{}^b y^{\tilde{0}}_{\text{pa}(\gamma)}, i^{\tilde{0}}_{\text{pa}(\gamma)}, \theta^{\tilde{0}}_{\gamma|i_{\text{pa}(\gamma)}}) p(\theta^{\tilde{0}}_{\gamma|i_{\text{pa}(\gamma)}}).$$

As the distribution, $p({}^c y^{\tilde{0}}_{\gamma}|{}^c y^{\tilde{0}}_{\text{pa}(\gamma)}, i^{\tilde{0}}_{\text{pa}(\gamma)}, \theta^{\tilde{0}}_{\gamma|i_{\text{pa}(\gamma)}})$, is a Gaussian distribution, then $p({}^b y^{\tilde{0}}_{\gamma}|{}^b y^{\tilde{0}}_{\text{pa}(\gamma)}, i^{\tilde{0}}_{\text{pa}(\gamma)}, \theta^{\tilde{0}}_{\gamma|i_{\text{pa}(\gamma)}})$ is a multivariate Gaussian distribution. The covariance matrix is diagonal as all the cases in the database are independent. This way we consider all the cases in a *batch*.

The same formulation applies for parameters in $B_{\rightarrow}$. Notice that the observations included in ${}^b y^{\rightarrow}_{\gamma}$ and ${}^b y^{\rightarrow}_{\text{pa}(\gamma)}$ are taken for each time point $t = 1, \ldots, T$.

The posterior distribution is found to be

$$\begin{aligned}(m_{\gamma|i_{\text{pa}(\gamma)}}, \beta_{\gamma|i_{\text{pa}(\gamma)}}|\sigma^2_{\gamma|i_{\text{pa}(\gamma)}}, d) &\sim \mathcal{N}_{k+1}(\mu'_{\gamma|i_{\text{pa}(\gamma)}}, \sigma^2_{\gamma|i_{\text{pa}(\gamma)}}(\tau^{-1}_{\gamma|i_{\text{pa}(\gamma)}})')\\ (\sigma^2_{\gamma|i_{\text{pa}(\gamma)}}|d) &\sim \mathcal{I}\Gamma\left(\frac{\rho'_{\gamma|i_{\text{pa}(\gamma)}}}{2}, \frac{\phi'_{\gamma|i_{\text{pa}(\gamma)}}}{2}\right),\end{aligned}$$

where

$$\begin{aligned}\tau'_{\gamma|i_{\text{pa}(\gamma)}} &= \tau_{\gamma|i_{\text{pa}(\gamma)}} + (z^b_{\text{pa}(\gamma)})^{\mathrm{T}} z^b_{\text{pa}(\gamma)}\\ \mu'_{\gamma|i_{\text{pa}(\gamma)}} &= (\tau'_{\gamma|i_{\text{pa}(\gamma)}})^{-1}(\tau_{\gamma|i_{\text{pa}(\gamma)}}\mu_{\gamma|i_{\text{pa}(\gamma)}} + (z^b_{\text{pa}(\gamma)})^{\mathrm{T}} y^b_{\gamma})\\ \rho'_{\gamma|i_{\text{pa}(\gamma)}} &= \rho_{\gamma|i_{\text{pa}(\gamma)}} + |b|\\ \phi'_{\gamma|i_{\text{pa}(\gamma)}} &= \phi_{\gamma|i_{\text{pa}(\gamma)}} + (y^b_{\gamma} - z^b_{\text{pa}(\gamma)}\mu'_{\gamma|i_{\text{pa}(\gamma)}})^{\mathrm{T}} y^b_{\gamma}\\ &\quad + (\mu_{\gamma|i_{\text{pa}(\gamma)}} - \mu'_{\gamma|i_{\text{pa}(\gamma)}})^{\mathrm{T}} \tau_{\gamma|i_{\text{pa}(\gamma)}}\mu_{\gamma|i_{\text{pa}(\gamma)}},\end{aligned}$$

where $|b|$ denotes the number of observations in y_γ^b.

5.4 Structure Learning

To learn the structure of a DBN, we again use a Bayesian approach and calculate the posterior probability of a DAG D given data d,

$$p(D|d) \propto p(d|D)p(D), \tag{8}$$

where $p(d|D)$ is the marginal likelihood of D and $p(D)$ is the prior probability of D.

In this paper we choose, for simplicity, to let all DAGs be equally likely a priori and therefore we use the measure

$$p(D|d) \propto p(d|D).$$

We refer to the above measure as a *network score*. We can, in principle, calculate the network score for all possible DAGs and then select the one with the highest score (or, if using model averaging, select a few with high score). In most situations however, there are too many different DAGs to evaluate and some kind of search strategy must be employed, see *e.g.* Cooper & Herskovits (1992).

The marginal likelihood $p(d|D)$ is given as follows.

$$\begin{aligned}
p(d|D) &= \int_{\theta\in\Theta} p(d|\theta, D)p(\theta|D)d\theta \\
&= \prod_{\delta\in\Delta_{\tilde{0}}} \prod_{i_{\mathrm{pa}(\delta)}\in\mathcal{I}_{\mathrm{pa}(\delta)}} \int \prod_{c:{}^c i^{\tilde{0}}_{\mathrm{pa}(\delta)}=i^{\tilde{0}}_{\mathrm{pa}(\delta)}} p({}^c i^{\tilde{0}}_\delta | i^{\tilde{0}}_{\mathrm{pa}(\delta)}, \theta^{\tilde{0}}_{\delta|i_{\mathrm{pa}(\delta)}}, D)p(\theta^{\tilde{0}}_{\delta|i_{\mathrm{pa}(\delta)}}|D)d\theta^{\tilde{0}}_{\delta|i_{\mathrm{pa}(\delta)}} \times \\
&\prod_{\gamma\in\Gamma_{\tilde{0}}} \prod_{i_{\mathrm{pa}(\gamma)}\in\mathcal{I}_{\mathrm{pa}(\gamma)}} \int \prod_{c:{}^c i^{\tilde{0}}_{\mathrm{pa}(\gamma)}=i^{\tilde{0}}_{\mathrm{pa}(\gamma)}} p({}^c y^{\tilde{0}}_\gamma | {}^c y^{\tilde{0}}_{\mathrm{pa}(\gamma)}, i^{\tilde{0}}_{\mathrm{pa}(\gamma)}, \theta^{\tilde{0}}_{\gamma|i_{\mathrm{pa}(\gamma)}}, D)p(\theta^{\tilde{0}}_{\gamma|i_{\mathrm{pa}(\gamma)}}|D)d\theta^{\tilde{0}}_{\gamma|i_{\mathrm{pa}(\gamma)}} \times \\
&\prod_{\delta\in\Delta_t} \prod_{i^{\rightarrow}_{\mathrm{pa}(\delta)}\in\mathcal{I}^{\rightarrow}_{\mathrm{pa}(\delta)}} \int \prod_{t=1}^{T} \prod_{c:{}^c i^{\rightarrow}_{\mathrm{pa}(\delta)}=i^{\rightarrow}_{\mathrm{pa}(\delta)}} p({}^c i^t_\delta | i^{\rightarrow}_{\mathrm{pa}(\delta)}, \theta^{\rightarrow}_{\delta|i_{\mathrm{pa}(\delta)}}, D)p(\theta^{\rightarrow}_{\delta|i_{\mathrm{pa}(\delta)}}|D)d\theta^{\rightarrow}_{\delta|i_{\mathrm{pa}(\delta)}} \times \\
&\prod_{\gamma\in\Gamma_t} \prod_{i^{\rightarrow}_{\mathrm{pa}(\gamma)}\in\mathcal{I}^{\rightarrow}_{\mathrm{pa}(\gamma)}} \int \prod_{t=1}^{T} \prod_{c:{}^c i^{\rightarrow}_{\mathrm{pa}(\gamma)}=i^{\rightarrow}_{\mathrm{pa}(\gamma)}} p({}^c y^t_\gamma | {}^c y^{\rightarrow}_{\mathrm{pa}(\gamma)}, i^{\rightarrow}_{\mathrm{pa}(\gamma)}, \theta^{\rightarrow}_{\gamma|i_{\mathrm{pa}(\gamma)}}, D)p(\theta^{\rightarrow}_{\gamma|i_{\mathrm{pa}(\gamma)}}|D)d\theta^{\rightarrow}_{\gamma|i_{\mathrm{pa}(\gamma)}}
\end{aligned}$$

We see that the marginal likelihood $p(d|D)$ factorizes into a product over terms involving only one node and its parents, called local marginal likelihoods, so the network score is *decomposable*. This also means that the likelihood factorizes into terms related to $B_{\tilde{0}}$ and terms related to $B_{\rightarrow}$. For mth order Markov time series, the likelihood factorizes in a similar manner into terms related to $B_{\tilde{0}}, \ldots, B_{m-1}$ and $B_{\rightarrow}$.

Because of the way we specified the possible parent sets of variables in $B_{\tilde{0}}$ and in $B_{\rightarrow}$, we can find the best DAG (the one with the highest network score) by finding the best DAG for $B_{\tilde{0}}$ and the best DAG for $B_{\rightarrow}$. So we can learn the structure of $B_{\tilde{0}}$ and $B_{\rightarrow}$ independently and we can learn them just as we learn ordinary Bayesian networks with mixed variables as described in Bøttcher (2001). This also applies for mth order Markov time series in which we can learn the structure of $B_{\tilde{0}}, \ldots, B_{m-1}$ and $B_{\rightarrow}$ independently.

In the following we do not distinguish between variables in $B_{\tilde{0}}$ and $B_{\rightarrow}$, as the terms presented apply for both $B_{\tilde{0}}$ and $B_{\rightarrow}$.

The network score contribution from the discrete variables in a network is given by

$$\prod_{\delta\in\Delta}\prod_{i_{\mathrm{pa}(\delta)}\in\mathcal{I}_{\mathrm{pa}(\delta)}}\frac{\Gamma(\alpha_{+\delta|i_{\mathrm{pa}(\delta)}})}{\Gamma(\alpha_{+\delta|i_{\mathrm{pa}(\delta)}}+n_{+\delta|i_{\mathrm{pa}(\delta)}})}\prod_{i_\delta\in\mathcal{I}_\delta}\frac{\Gamma(\alpha_{i_\delta|i_{\mathrm{pa}(\delta)}}+n_{i_\delta|i_{\mathrm{pa}(\delta)}})}{\Gamma(\alpha_{i_\delta|i_{\mathrm{pa}(\delta)}})}. \tag{9}$$

For the continuous variables, the local marginal likelihoods are non-central t distributions with $\rho_{\gamma|i_{\mathrm{pa}(\gamma)}}$ degrees of freedom, location vector $z^b_{\mathrm{pa}(\gamma)}\mu_{\gamma|i_{\mathrm{pa}(\gamma)}}$ and scale parameter $s_{\gamma|i_{\mathrm{pa}(\gamma)}} = \frac{\phi_{\gamma|i_{\mathrm{pa}(\gamma)}}}{\rho_{\gamma|i_{\mathrm{pa}(\gamma)}}}(I + (z^b_{\mathrm{pa}(\gamma)})\tau^{-1}_{\gamma|i_{\mathrm{pa}(\gamma)}}(z^b_{\mathrm{pa}(\gamma)})^{\mathrm{T}})$. The index b is defined as in Section 5.3.

The network score contribution from the continuous variables is given by

$$\prod_{\gamma\in\Gamma}\prod_{i_{\mathrm{pa}(\gamma)}\in\mathcal{I}_{\mathrm{pa}(\gamma)}}\frac{\Gamma((\rho_{\gamma|i_{\mathrm{pa}(\gamma)}}+|b|)/2)}{\Gamma(\rho_{\gamma|i_{\mathrm{pa}(\gamma)}}/2)[\det(\rho_{\gamma|i_{\mathrm{pa}(\gamma)}}s_{\gamma|i_{\mathrm{pa}(\gamma)}}\pi)]^{\frac{1}{2}}}\times$$

$$\left[1+\frac{1}{\rho_{\gamma|i_{\mathrm{pa}(\gamma)}}}(y^b_\gamma - z^b_{\mathrm{pa}(\gamma)}\mu_{\gamma|i_{\mathrm{pa}(\gamma)}})s^{-1}_{\gamma|i_{\mathrm{pa}(\gamma)}}(y^b_\gamma - z^b_{\mathrm{pa}(\gamma)}\mu_{\gamma|i_{\mathrm{pa}(\gamma)}})^{\mathrm{T}}\right]^{\frac{-(\rho_{\gamma|i_{\mathrm{pa}(\gamma)}}+|b|)}{2}}. \tag{10}$$

The network score is thus the product of (9) and (10).

So if the time series is first order Markov, we can find the best DAG by finding the best DAG for $B_{\tilde{0}}$ and the best DAG for $B_{\rightarrow}$. If it is mth order Markov, we find the best DAGs for $B_{\tilde{0}}, \ldots, B_{m-1}$ and $B_{\rightarrow}$.

5.5 Learning the Markov Order

If the Markov order of the time series is unknown, we can learn it by choosing a "prior" order and learn the DBN with this order. The learned order can then be read from the best DAG for $B_{\rightarrow}$, by determining which time slices X^t has parents from. The slice furthest back in time will give the order.

It is important that the prior order is chosen high enough to ensure that no order higher than this is better in describing the time series. How high this prior order in practice should be chosen, depends on any prior information available on the time series, but also of how large a dataset the network is learned from. The higher we choose the order, the more complex the possible DAGs are, with more parameters to estimate and fewer cases to learn them from.

To increase the stability of the search procedure, it could therefore be better to start by learning a DBN with a low Markov order. If the best DAG for $B_{\rightarrow}$ include dependencies up to the chosen order, a network with a higher order should be tried and this should be repeated until no dependencies of higher order reveal themselves. However, with this procedure there is a chance that the best Markov order will not be learned. If *e.g.* a prior order of three is chosen and the learned network only reveals second order Markov properties, we would with this procedure conclude that the time series is second order Markov, even though the best order could be higher than three. An example of this is shown in Section 7.

Situations can arise, where the Markov order in the initial DAGs is higher than in $B_{\rightarrow}$. For example, if we have assumed that the time series is third order Markov, we need to learn the structure of $B_{\tilde{0}}, B_1, B_2$ and $B_{\rightarrow}$. Consider now a situation where $B_{\rightarrow}$ is learned to be first order Markov, *i.e.* X^t has only parents in X^t and X^{t-1}, while B_2 is learned to be second order Markov, *i.e.* to have time varying parents from $B_{\tilde{0}}$. This is not necessarily a problem, but it should be noted that if we had assumed the first order Markov property, then there would have been more cases to learn the parameters in $B_{\rightarrow}$ by. In such situations, the importance of specifying the initialization of the time series correctly, must be compared to the loss of precision in the distribution of the parameters in $B_{\rightarrow}$.

6 Specifying Prior Distributions

To learn the structure of the DAG we need to specify prior parameter distributions for all possible DAGs under evaluation. An automated procedure for doing this has been developed for ordinary Bayesian networks. We call it the *master prior procedure*. The procedure is for the purely discrete case treated in Heckerman, Geiger & Chickering (1995), for the purely continuous case in Geiger & Heckerman (1994) and for the mixed case in Bøttcher (2001).

We will here give an outline of the procedure and show how it can be used for specifying prior parameter distributions for DBNs.

6.1 The Master Prior Procedure

The idea in the master prior procedure is that from a given Bayesian network, we can deduce parameter priors for any possible DAG. The user just has to specify a *prior Bayesian network*, which is the Bayesian network as he believes it to be. Also, he has to specify an *imaginary sample size*, N, which is a measure of how much confidence he has in the prior network. The procedure works as follows.

1. Specify an imaginary sample size.
2. Specify a prior Bayesian network, *i.e.* a prior DAG and prior local probability distributions. Calculate the joint prior distribution.
3. From the joint prior distribution and the imaginary sample size, the marginal distribution of all parameters in the family consisting of a node and its parents can be determined. We call this a *master prior*.
4. The local parameter priors are now determined by conditioning in these master prior distributions.

This procedure ensures parameter independence. Further, it has the property that if a node has the same set of parents in two different networks, then the local parameter prior for this node will be the same in the two networks. Therefore, we only have to deduce the local parameter prior for a node, given the same set of parents, once. This property is called *parameter modularity*. Finally, the procedure ensures *likelihood equivalence*, that is, if two DAGs represent the same set of conditional independencies, the network score for these two DAGs will be the same.

As an example, we will show how to deduce parameter priors for the discrete nodes.

Let $\Psi = (\Psi_i)_{i \in \mathcal{I}}$ be the parameters for the joint distribution of the discrete variables. The joint prior parameter distribution is assumed to be a Dirichlet distribution

$$p(\Psi) \sim \mathcal{D}(\alpha),$$

with hyperparameters $\alpha = (\alpha_i)_{i \in \mathcal{I}}$. To specify this Dirichlet distribution, we need to specify these hyperparameters. Consider the following relation for the Dirichlet distribution,

$$p(i) = \mathbb{E}(\Psi_i) = \frac{\alpha_i}{N},$$

with $N = \sum_{i \in \mathcal{I}} \alpha_i$. Now we let the probabilities in the prior network be an estimate of $\mathbb{E}(\Psi_i)$, so we only need to determine N in order to calculate the parameters α_i.

We determine N by using the notion of an imaginary data base. We imagine that we have a database of cases, from which we have updated the distribution of Ψ out of total ignorance. The *imaginary sample size* of this imaginary data base is

thus N. It expresses how much confidence we have in the dependency structure expressed in the prior network, see Heckerman et al. (1995).

We use this joint distribution to deduce the master prior distribution of the family $A = \delta \cup \text{pa}(\delta)$. Let

$$\alpha_{i_A} = \sum_{j:j_A=i_A} \alpha_j,$$

and let $\alpha_A = (\alpha_{i_A})_{i_A \in \mathcal{I}_A}$. Then the marginal distribution of Ψ_A is Dirichlet, $p(\Psi_A) \sim \mathcal{D}(\alpha_A)$. This is the master prior in the discrete case. Notice that the parameters in the master prior can also be found as

$$\alpha_{i_A} = Np(i_A),$$

where $p(i_A) = \sum_{j:j_A=i_A} p(i)$.

The local parameter priors can now be found by conditioning in these master prior distributions. The conditional distribution of $\Psi_{\delta|i_{\text{pa}(\delta)}}$ is

$$p(\Psi_{\delta|i_{\text{pa}(\delta)}}) \sim \mathcal{D}(\alpha_{\delta|i_{\text{pa}(\delta)}}),$$

with $\alpha_{i_\delta|i_{\text{pa}(\delta)}} = \alpha_{i_A}$.

6.2 The Master Prior Procedure for DBNs

For DBNs, the parameter priors can also be found by using the above procedure. Consider a DBN for a first order Markov time series (the procedure is directly extendible to time series with higher order Markov properties). As the DAG from time $t = 1$ and forward repeats itself, the structure of the overall DAG is completely specified by the structure of the first two time slices. So we can specify all the parameter priors we need from a prior network consisting of the variables $X^{\tilde{0}}$ and X^1. Notice that the parameter priors for $B_{\rightarrow}$ *are* the same as the parameter priors for the parameters in X^1, as this is the first time point in the time series.

We will also allow for different imaginary sample sizes for the parameters in $B_{\tilde{0}}$ and the parameters in $B_{\rightarrow}$. One reason for this is that the parameters in $B_{\rightarrow}$ are updated with more cases than the parameters in $B_{\tilde{0}}$ and therefore might need a stronger prior distribution.

The procedure works almost as the procedure for ordinary Bayesian networks, the only difference being the different imaginary sample sizes.

1. Specify an imaginary sample size, $N^{\tilde{0}}$, for $B_{\tilde{0}}$, and an imaginary sample size, $N^{\rightarrow}$, for $B_{\rightarrow}$, .
2. Specify a prior Bayesian network for the first two time slices. Calculate the joint prior distribution.

3. From the joint prior distribution and the imaginary sample size, the master prior for all parameters in a family can be determined. For families including only variables from $X^{\tilde{0}}$, the imaginary sample size for $B_{\tilde{0}}$ is used and for the other families, the imaginary sample size for $B_{\rightarrow}$ is used.

4. The local parameter priors are now determined by conditioning in the appropriate master prior distribution.

It is obvious that parameter independence and parameter modularity still applies as these properties are not influenced by the use of different imaginary sample sizes. Neither is likelihood equivalence, as variables in $X^{\tilde{0}}$ can not have parents from X^1. This means that parameter priors for two DAGs that represent the same set of conditional independencies, are calculated using the same imaginary sample sizes. So likelihood equivalence also still applies.

As a simple example of the master prior procedure for DBNs, consider a time series for a single discrete variable $I^0, \ldots, I^T$. Assume that the time series is first order Markov. The parameter priors for the DAG in Figure 7 are deduced as follows

$$\alpha^0_{i^0} = N^0 p(i^0),$$
$$\alpha^{\rightarrow}_{i^t|i^{t-1}} = N^{\rightarrow} p(i^0, i^1).$$

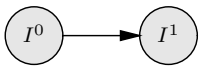


Figure 7: DAG for first order Markov time series.

7 Example

In this section, we will analyze the Wölfer's sunspot numbers using a dynamic Bayesian network. The Wölfer's sunspot numbers are annual measures of sunspot activity, collected from 1700 to 1988. In statistical terms, the sunspot numbers is a univariate continuous time series $Y^0, \ldots, Y^{288}$. The dataset we use is from Tong (1996).

The sunspot numbers are shown in Figure 8.

Many statistical investigations of these numbers have been made. Anderson (1971) gives a short review of some of these studies. For example, for annual measures of sunspot activity from 1749 to 1924, Yule (1927) proposed the autoregressive process as a statistical model. He calculated the $AR(p)$ for $p = 2$ and $p = 5$ and

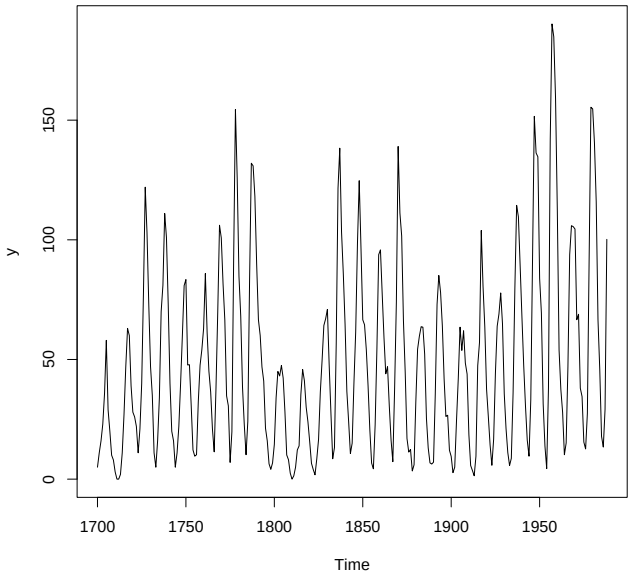


Figure 8: Wölfer's sunspot numbers.

found that an AR(2) was sufficient, *i.e.* he estimated the sequence to be second order Markov. Another example is found in Schaerf (1964). She fits an autoregressive model with lags 1, 2, and 9.

Here we will use a DBN as the statistical model and learn the Markov order by structural learning of the DBN. The software package `deal`, see Bøttcher & Dethlefsen (2003), is used for the analysis.

Our aim is to learn the Markov order, so we are only interested in learning the structure of $B_{\rightarrow}$. The structure of the initial networks is not of interest and are actually not likely to be determined by learning from the sunspot numbers. These numbers are namely represented by *one* time series, meaning that for the initial networks there are only one observation of each variable.

As the prior network we use the empty network, *i.e.* the one without any arrows. In order to get the right location and scale of the parameters, we estimate the prior probability distribution for the empty network from data, *i.e.* we use the sample mean and the sample variance as the mean and variance in the prior probability distribution.

As the number of observations in the sunspot series is relatively large, we can choose a rather high Markov order for the DBN. Anderson (1971) concludes that the order is not higher than 18. But to be absolutely sure that we capture the best

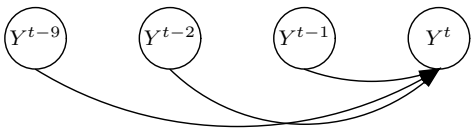


Figure 9: The learned network, $B_{\rightarrow}$, when an Markov order of 30 is assumed. The variables that do not influence Y^t, have been omitted.

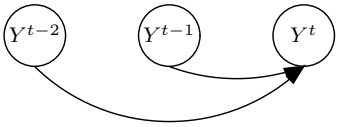


Figure 10: The learned network, $B_{\rightarrow}$, when the 3. order Markov property is assumed. The variable Y^{t-3} have been omitted as it does not influence Y^t.

order, we choose an order of 30. The result of the structural learning of $B_{\rightarrow}$ is shown in Figure 9. The variables that do not influence Y^t, have been omitted in the figure. From the result we see that the sunspot numbers can be described by a Markov process of order 9 with lags 1, 2 and 9, *i.e.*

$$Y^t = m + \beta_1 Y^{t-1} + \beta_2 Y^{t-2} + \beta_9 Y^{t-9} + \epsilon_t, \quad \epsilon_t \sim \mathcal{N}(0, \sigma^2),$$

with parameter estimates $m = 5.06$, $\beta_1 = 1.21$, $\beta_2 = 0.51$, $\beta_9 = 0.21$ and $\sigma^2 = 267.5$.

The result is in accordance with some of the previous studies, *e.g.* Schaerf (1964) as mentioned earlier. Other studies determine that an second order Markov process is sufficient, *e.g.* Yule (1927). But as mentioned, he only examines an order as high as 5.

We have also tried to learn $B_{\rightarrow}$ using lower Markov order properties. If we *e.g.* use a Markov order of 3, we reach the conclusion that the sunspot numbers are 2. order Markov, with lags 1 and 2. This result is shown in Figure 10. Similarly, if we learn $B_{\rightarrow}$ using the order $2, \ldots, 7$ or 8, we still reach the conclusion that the sunspot numbers are second order Markov, with lags 1 and 2. This is therefore an example of the importance of choosing the prior Markov order high enough.

As can be seen from Figure 8, the sunspot numbers are periodical with a period of between 10 and 11 years. To determine the period more precisely, we calculate the spectrum,

$$f(\omega) = \sigma^2 (1 - \sum_t \beta_t e^{-it\omega})^{-2},$$

see Venables & Ripley (1997), using the parameter estimates obtained from `deal`.

The spectrum is shown in Figure 11.

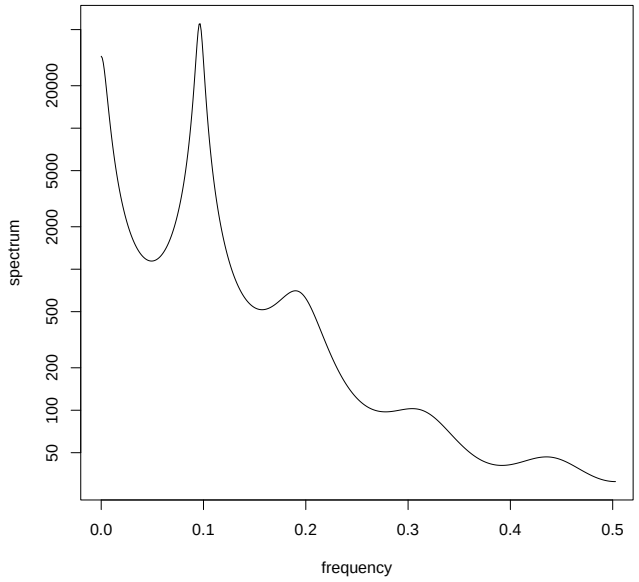


Figure 11: Spectrum of Wölfer's sunspot numbers.

There is a peak at frequency 0.096, which corresponds to a period of $1/0.096 = 10.40$ years. This result is also in accordance with previous studies.

Acknowledgements

This research was supported by Novo Nordisk A/S. Also I would like to thank Claus Dethlefsen for useful discussions and his help with the implementation of the example. Finally, I thank my supervisor Steffen L. Lauritzen for many valuable comments.

References

Anderson, T. W. (1971). *The Statistical Analysis of Time Series*, John Wiley and Sons, New York.

Bøttcher, S. G. (2001). Learning Bayesian Networks with Mixed Variables, *Artificial Intelligence and Statistics 2001*, Morgan Kaufmann, San Francisco, CA, USA, pp. 149–156.

Bøttcher, S. G. & Dethlefsen, C. (2003). `deal`: A Package for Learning Bayesian Networks, *Journal of Statistical Software* **8**(20): 1–40.

Bøttcher, S. G., Milsgaard, M. B. & Mortensen, R. S. (1995). *Monitoring by using dynamic linear models - illustrated by tumour markers for cancer*, Master's thesis, Aalborg University.

Cooper, G. & Herskovits, E. (1992). A Bayesian method for the induction of probabilistic networks from data, *Machine Learning* **9**: 309–347.

Dean, T. & Kanazawa, K. (1989). A model for reasoning about persistence and causation, *Computational Intelligence* **5**: 142–150.

DeGroot, M. H. (1970). *Optimal Statistical Decisions*, McGraw-Hill, New York.

Friedman, N., Murphy, K. P. & Russell, S. (1998). Learning the Structure of Dynamic Probabilistic Networks, *Proceedings of Fourteenth Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann, San Francisco, CA, USA, pp. 139–147.

Geiger, D. & Heckerman, D. (1994). Learning Gaussian Networks, *Proceedings of Tenth Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann, San Francisco, CA, USA, pp. 235–243.

Harrison, P. J. & Stevens, C. F. (1976). Bayesian forecasting, *Journal of Royal Statistics* **38**: 205–247.

Heckerman, D., Geiger, D. & Chickering, D. M. (1995). Learning Bayesian networks: The combination of knowledge and statistical data, *Machine Learning* **20**: 197–243.

Lauritzen, S. L. (1996). *Graphical Models*, Clarendon press, Oxford, New York.

Murphy, K. P. (2002). *Dynamic Bayesian Networks: Representation, Inference and Learning*, PhD thesis, University of California, Berkeley.

Schaerf, M. C. (1964). Estimation of the covariance and autoregressive structure of a stationary time series, *Technical report*, Department of Statistics, Stanford University.

Tong, H. (1996). *Non-Linear Time Series*, Clarendon Press, Oxford.

Venables, W. N. & Ripley, B. D. (1997). *Modern Applied Statistics with S-PLUS*, second edn, Springer-Verlag, New York.

West, M. & Harrison, J. (1989). *Bayesian Forecasting and Dynamic Models*, Springer-Verlag, New York.

Yule, G. U. (1927). On a method for investigating periodicities in disturbed series with special reference to Wölfer's sunspot numbers, *Philosophical Transactions of the Royal Society, Series A* **226**: 267–298.