Chapter 5

Methods of Estimation of Dynamic Bayesian Networks

Trees are subclasses of Bayesian Networks (BNs) defined in Chapter 3. BNs are effective when there are no cyclic dependencies among data. Also, several networks with the same undirected graph structure, but different directions of some edges may represent the same distribution. And the dynamics of the system under study, could not be taken into account when using a BN to model an evolutionary pathway. Signaling pathways are dynamic events that take place over a given period of time. The limitations of BN may be over come by Dynamic Bayesian Networks (DBN), which model the stochastic evolution of a set of random variables over time. The advantages of DBNs include the ability to model stochasticity, to incorporate prior knowledge, and to handle hidden variables and missing data in a principled way. Moreover, in DBNs the acyclicity constraint is relaxed. For example, consider Figure 5.1. The network contains a cycle $X_1 \rightarrow X_2 \rightarrow X_4 \rightarrow X_5 \rightarrow X_1$. A BN model can not treat such a network. On the other hand, the DBN can construct a cycle by dividing the state space by time points, as in
5.1 Dynamic Bayesian Networks

A DBN, $N$ is a representation of stochastic evolution of a set of random variables $X = \{X_1, X_2, \ldots, X_n\}$, where $n$ is the number of variables over discretized time $t$. The represented temporal process is assumed to be Markovian, that is,

$$P(X^t \mid X^0, X^1, \ldots, X^{t-1}) = P(X^t \mid X^{t-1})$$

and time homogeneous, that is, $P(X^t \mid X^{t-1})$ are independent of $t$. The representation consists of
two components:

1. a directed graph $G = (X, E)$ encoding conditional independencies,

2. a family of conditional distributions $P(X_t \mid pa_t^{t-1})$ where $pa_i = \{X_j \in X \mid (X_j, X_i) \in E\}$.

By assumption, the joint distribution over all the possible trajectories of the process decomposes into the following product form

$$P(X^0, X^1, \ldots, X^T) = P(X^0) \prod_{t=1}^T P(X^t \mid X^{t-1}).$$

Consequently, the evolution of the random variables is given by,

$$P(X^1, X^2, \ldots, X^T \mid X^0) = \prod_{t=1}^T P(X^t \mid X^{t-1}) = \prod_{t=1}^T \prod_{i=1}^n P(X^t_i \mid pa_i^{t-1}) = \prod_{i=1}^n \prod_{t=1}^T P(X^t_i \mid pa_i^{t-1}).$$

### 5.1.1 Methods of Estimating DBNs in Genetics

1. To overcome the inefficiency of tree models to take into account, time, a factor of obvious importance to oncogenesis, Desper et al. (1998) introduced a more elaborate model, called *timed oncogenetic trees*. A timed oncogenetic tree is a labeled tree $T = (V, E, r, \lambda)$, together with a distribution $P_T$ on the positive reals. Timed oncogenetic trees represent the following sampling procedure. First, for each edge $e$, a random variable $t(e)$ exponentially distributed with mean $\frac{1}{\lambda(e)}$ is sampled and a real number $t_{tot}$ is sampled from distribution $P_T$. A node $v$ is included in the outcome if, and only if, there is a path from $r$ to $v$ in $T$, and the sum of all $t(e)s$ over all edges $e$ on this path is at most $t_{tot}$. Thus, in a
timed oncogenetic tree, those events are selected that have happened by
time $t_{tot}$ where $t_{tot}$ is sampled from $P_T$ which reflects the time, relative
to the oncogenetic process, at which a tumor is sampled from a patient.

2. Murphy et al (1999) introduced the use of DBN in modeling gene
expression data. Their paper provides a review of techniques for learning DBNs. They used the DBN of J. Pearl (1988), which was very much
in use in representing stochastic models in the uncertainty in Artificial
Intelligence community. They could show that most of the discrete
time models, namely, Boole network model, linear and non linear mod-
eels, hidden Markov models are all special cases of DBNs.  

models.
(a) In the discrete model, discretisation is applied to remove noise from
the microarray data. Let $U = \{u_1, u_2, \ldots, u_m\}$ be a finite set of discrete
values. An expression value $x_{ij}$ is then transformed to $u_l$, and $\theta_{jkl} =
\Pr(X_{ij} = u_l | P_{i-1,j} = u_{jk})$ where $u_{jk}$ is the $k^{th}$ entry of the state table of
parents of the $j^{th}$ gene. Then the density of the model, $f(x_{11}, \ldots, x_{np}; \theta)$
can be modeled as a multinomial distribution

$$f(x_{11}, \ldots, x_{np}; \theta) = \prod_{j=1}^{p} \prod_{k=1}^{Q_j} \prod_{l=1}^{m} \theta_{jkl}^{N_{jkl}}$$

where $\theta = (\theta_{111}, \ldots, \theta_{pQ_pm})'$, $N_{jkl}$ indicates the number of observations
satisfying $x_{ij} = u_l$ and $P_{i-1,j} = u_{jk}$ for $i = 2, 3 \ldots, n$, and $Q_j = m^{q_j}$
is the number of entries of the state table of parents of the $j^{th}$ gene.
$P_{i-1,j} = (P_{i-1,1}^{(j)}, \ldots, P_{i-1,q_j}^{(j)})'$ is a random variable vector of parent genes of $j^{th}$ gene at time $i - 1$. ; $p_{0j} = \phi$.

(b) In the continuous model,

$$f(x_{11}, \ldots, x_{np}; \theta) = \prod_{i=1}^{n} \prod_{j=1}^{p} g_j(x_{ij} \mid p_{i-1,j})$$

where $g_j(x_{ij} \mid p_{i-1,j}; \theta_j) = \frac{1}{\sqrt{2\pi}\sigma_j^2} \exp\left\{ \frac{- (x_{ij} - m(p_{i-1,j}))^2}{2\sigma_j^2} \right\}$, the normal density function. For capturing even non-linear relationships between genes, they used a non-parametric regression model based on B-splines:

$$m(p_{i-1,j}) = \sum_{m=1}^{M_{j1}} \gamma_{m1}(P_{i-1,1}^{(j)}) + \cdots + \sum_{m=1}^{M_{jq_j}} \gamma_{mq_j}(P_{i-1,q_j}^{(j)})$$

where $\gamma_{1k}, \ldots, \gamma_{M_{jk}k}$ are coefficient parameters and $\{b_1^{(j)}(\cdot), \ldots, b_{M_{jk}}^{(j)}(\cdot)\}$ is a prescribed set of B-splines. The criterion they used for selecting the network was defined by, $BNRC_{dynamic}(G) = \sum_{j=1}^{p} BNRC_{dynamic}^{(j)}$ where

$$BNRC_{dynamic}^{(j)} \equiv -2\log \pi^{(j)} - r_j \log \left( \frac{2\pi}{n} \right) + \log |J_{\hat{\lambda}_j}^{(j)}(\hat{\theta}_j) | - 2n l_{\lambda_j}^{(j)}(\hat{\theta}_j \mid X)$$

$r_j$ is the dimension of $\theta_j$,

$$l_{\lambda_j}^{(j)}(\hat{\theta}_j \mid X) = \sum_{i=1}^{n} \log g_j(x_{ij} \mid p_{i-1,j}; \theta_j)$$

$$J_{\lambda_j}^{(j)}(\hat{\theta}_j) = -\frac{\delta^2 \{ l_{\lambda_j}^{(j)}(\hat{\theta}_j \mid X) \}}{\delta \theta_j \delta \theta_j'}$$

$\hat{\theta}_j$ is the mode of $l_{\lambda_j}^{(j)}(\theta_j \mid X)$ and $\pi(j)$ are prior probabilities satisfying

$$\sum_{j=1}^{p} \log \pi(j) = \log \pi(G)$$. The DBN, $G$, for which this score is minimal is chosen. Graphical view of their network model is given in Figure 5.3.
4. Perrin et al (2003) combined kalman filter with the deterministic inertial model proposed by d’Alche et al (2004). The equations are of the form,

\[
\frac{d^2 E_i(t)}{dt^2} + 2\lambda_i \omega_i \frac{dE_i(t)}{dt} + \omega_i^2 E_i(t) = \sum_j \omega_{ij} E_j(t)
\]

where \( E_i(t) \) is the expression level of gene \( i \) at time \( t \), namely, the quantity of mRNA produced by the gene at this time. \( \lambda_i \) plays the role of an absorption coefficient specific to gene \( i \), while \( \omega_i \) acts as a natural frequency of gene \( i \). Following D’Alche et al (2004), this \( n \) second order system of equations can be discretised to be numerically implemented. Assuming that the measurement time unit is lower than the gene evolution characteristic time, continuous derivatives are replaced by

\[
\frac{\Delta E_i(t)}{\Delta t} = E_i(t + 1) - E_i(t),
\]

and,

\[
X_t = (E_1(t), \ldots, E_n(t), \frac{\Delta E_1(t)}{\Delta t}, \ldots, \frac{\Delta E_n(t)}{\Delta t})'.
\]
The evolution of the network of $n$ genes can be described by the Kalman filter model $X_{(t+1)} = AX_t + U$

$Y_t = CX_t + \mu_{obs} + V$, where $X_t$ is the hidden state of the gene network at instant $t$, while $Y_t$ is the observed state of the network, composed of all observations of gene expression levels.

$A$ is the transition matrix, $A = \begin{bmatrix} \text{identity} & \text{identity} \\ W - \Omega^2 & \text{identity} - 2\Omega\Lambda \end{bmatrix}$ identity

is the identity matrix of size $n \times n$, $W = (\omega_{ij})_{1 \leq i,j \leq n}$,

$\Omega = \text{diag}(\omega_1, \omega_2, \ldots, \omega_n)$, $\Lambda = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$;

$C$ is the projection matrix, $C_{n \times 2n} = \begin{bmatrix} \text{identity} & D_{n \times n} \end{bmatrix}$, $\mu_{obs}$ is a measurement adjustment vector, $U$ and $V$ are iid Gaussian random variables with zero mean and variances $\sigma_x^2$ and $\sigma_{obs}^2$. The Figure 5.4 illustrates the situation.

Figure 5.4: BN(top) Perrin’s DBN
5.2 Proposed IBMDBN

As is evident from the above models, DBNs are time-invariant models and the underlying network structures do not change over time. That is, the dependencies between variables in $X^{t-1}$ and $X^t$ are fixed, and $P(X^t \mid X^{t-1})$ is invariant over time. See Figures 5.3 and 5.4. Song and Xing (2009) proposed a time-varying DBN (TVDBN) for modeling the structurally varying directed dependency structures underlying non-stationary biological or neural time series. Under this model, let $G^t = (V, E^t)$ represent the conditional independence relations between the components of random vectors $X^{t-1}$ and $X^t$. The vertex set $V$ is a common set of variables underlying $X^{1:T}$, that is, each node in $V$ corresponds to a sequence of variables $X^{1:T}$, where $X^{1:T}$ is the set of variables $X^1, X^2, \ldots, X^T$. The edge set $E^t \subseteq V \times V$ contains directed edges from components of $X^{t-1}$ to those of $X^t$; an edge $(i, j)$ do not belong to $E^t$ if, and only if, $X^t_i$ is conditionally independent of $X^{t-1}_j$ given the rest of the variables in the model. Hence the density function over the model is,

$$P(X^1, X^2, \ldots, X^T) = P(X^1) \prod_{t=2}^{T} P^t(X^t \mid X^{t-1})$$

$$= P(X^1) \prod_{t=2}^{T} \prod_{i=1}^{p} P^t(X^t_i \mid X^{t-1}_{pa_i})$$

where $X^{t-1}_{pa_i} = \{X^{t-1}_j : X^{t-1}_j \rightarrow X^t_i \in E^T\}$ and the auto-regressive model suggested is, $X^t = A^t X^{t-1} + \epsilon, \epsilon \sim N(0, \sigma^2 I)$. Now, the estimation of the TVDBN is equivalent to the estimation of non-zero entries
in the sequence of time dependent transition matrices 
\{A^t\}, \ (t = 1, \ldots, T).

Our proposed method uses the formalisms of imprecise probabilities, 
set-based weighted graphs and Bayesian analysis, described in the pre-
vious chapter.

We consider \(N\) samples, observed over \(T\) time points. For \(k^{th}\) sam-
ple, we consider tables of the form given in figure 5.5. Let the \(\left(i, j\right)^{th}\)

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline
 & mut1 & mut2 & mut3 & \cdots & mutp \\
\hline
time & & & & & \\
1 & 0 & 1 & 0 & \cdots & 0 \\
2 & 0 & 1 & 0 & \cdots & 0 \\
3 & 1 & 1 & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
T-1 & 1 & 1 & 0 & \cdots & 0 \\
T & 1 & 1 & \cdots & \cdots & 0 \\
\hline
\end{tabular}
\caption{View of a sample}
\end{table}

entry in the table in Figure 5.5 be \(X_{ij}^{(k)}\), where,

\[ x_{ij}^{(k)} = \begin{cases} 
1 & \text{if } mut_j \text{ is present at time } i \text{ in sample } k \\
0 & \text{otherwise} 
\end{cases} \]

For sample \(k\) it is evident from table in Figure 5.5 that \(mut1\) follows 
\(mut2\) and \(mut3\) follows \(mut2\). Now, for all the \(N\) samples, we count
such evolutions and form the transition count matrix as,

$$ Y = \begin{pmatrix}
\text{mut}_1 & \text{mut}_2 & \text{mut}_3 & \ldots & \text{mut}_p \\
\text{mut}_1 & - & Y_{12} & Y_{13} & \ldots & Y_{1p} \\
\text{mut}_2 & Y_{21} & - & Y_{23} & \ldots & Y_{2p} \\
\text{mut}_3 & \ldots & \ldots & \ldots & \ldots & \ldots \\
\ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
\text{mut}_p & Y_{p1} & Y_{p2} & Y_{p3} & \ldots & -
\end{pmatrix} $$

The diagonal elements are not entered, because, it is assumed that when a mutation appears, that continues or disappears due to treatment. An entry that becomes one in the table will remain in that status.

$Y_{ij}$ = number of transitions from $\text{mut}_i$ to $\text{mut}_j$ in time $t-1$ to $t$; $i, j = 1, 2, \ldots, p; i \neq j; t = 2, \ldots, T$.

$$ Y_{ij} = \sum_{k=1}^{N} \sum_{t=2}^{T} \#(x_{ij}^{(k)} = 1 \mid x_{i-1}^{(k)} = 1, x_{j-1}^{(k)} = 0) $$

Let $\theta_{ij} = P(Y_{ij} \neq 0); i \neq j$

$$ \theta_{p \times p} = \begin{pmatrix}
- & \theta_{12} & \theta_{13} & \ldots & \theta_{1p} \\
\theta_{21} & - & \theta_{23} & \ldots & \theta_{2p} \\
\ldots & \ldots & \ldots & \ldots & \ldots \\
\theta_{p1} & \theta_{p2} & \theta_{p3} & \ldots & -
\end{pmatrix} $$

We have, $Y_{ij} \sim Binomial(\theta_{ij}, N); i, j = 1, 2, \ldots, p; i \neq j$. In the Bayesian analysis, the conjugate prior distribution for $\theta_{ij}$ is $\text{beta}(\alpha_{ij}, \beta_{ij})$. 
5.2.1 Credal Set Theory and Imprecise Beta Model

Credal set theory utilizes closed sets of probability measures, referred to as a credal set. Credal set theory can be seen as a generalization of precise Bayesian theory based on expected utility maximization procedures. This belongs to a family of theories referred to as imprecise probability where most theories utilize the same underlying mathematical structure (Coolen, F et al. 2010). A credal set is a closed convex set of probability mass functions. A credal set for a rv $X$ is denoted by $K(X)$. For any rv, the lower and upper probabilities according to the credal set $K(X)$ are $P(x) = \min_{p(X) \in K} p(x)$ and $\overline{P}(x) = \max_{p(X) \in K} p(x)$. Walley (1991) shows that inferences based on a credal set are equivalent to those based only on its vertices. A way of obtaining a credal set is to use a statistical model based on imprecise probabilities.

We use the imprecise Beta model of Walley (1991), to model the prior distributions of our parameters $\theta_{ij}$, which results in a set of posterior probability distributions for $\theta_{ij}$.

$$g(\theta_{ij}) \propto \theta_{ij}^{\alpha_{ij}-1}(1 - \theta_{ij})^{\beta_{ij}-1}; \quad 0 < \theta_{ij} < 1, \quad s > 0, \quad 0 < t_{ij} < 1,$$

and $\alpha_{ij} = st_{ij}$ and $\beta_{ij} = s(1 - t_{ij}); \quad i \neq j$. Value of $s$ indicates how fast the upper and lower probabilities converge to a single value. Higher value of $s$ indicates slow convergence. The posterior upper and lower expected values of the parameters are given by

$$\overline{E}(\theta_{ij}) = \sum_{i} \sum_{i \neq j} y_{ij} + s \overline{E}(\theta_{ij}) = \sum_{i} \sum_{i \neq j} y_{ij} + s$$
\[ E(\theta_{ij}) = \sum_{i} \sum_{i \neq j} \frac{y_{ij}}{y_{ij} + s} \]

We use the formalism of set-based weighted graphs as in previous chapter to find the dominant edges of our Imprecise Beta Model DBN (IBMDBN).

### 5.3 Proposed Algorithm

1. Enter number of samples \( N \), number of mutations \( p \), number of time points \( T \), and value of \( s \).

2. Enter state transition matrices for \( N \) samples.

3. Form transition count matrix \( Y_{p \times p} \), where,

\[ y_{ij} = \sum_{k} \sum_{t} \#(x_{tj}^{(k)} = 1 \mid x_{t-1i}^{(k)} = 1, x_{t-1j}^{(k)} = 0), \]

for \( i, j = 1, 2, \ldots, p \)

4. Assign names \( V_1, V_2, \ldots, V_p \) to mutations.

5. Total \( \leftarrow \sum_{i} \sum_{j} y_{ij} \).

6. \( i \leftarrow 1; \ max \ i \leftarrow p, \ j \leftarrow 1; \ max \ j \leftarrow p; \ i \neq j. \)

7. Compute \( \begin{bmatrix} \frac{y_{ij}}{total+s} & \frac{y_{ij}+s}{total+s} \end{bmatrix} \)

8. \( i = i + 1, \ j = j + 1. \)

9. Go to step \#7 if \( i, j < max \ i, max \ j. \)
10. Form interval of transition probability matrix \( D \), where

\[
D_{ij} = \left[ \frac{y_{ij}}{y_{ij} + y_{ij} + s} \right] \text{ for } i, j = 1, 2, \ldots, p; \ i \neq j.
\]

11. Compare intervals to find dominant edges.

12. Initialize \( D_t \) as a \( p \times p \) null matrix.

13. The \((ij)\text{th} \) entry in \( D_t \) is one if the edge \( V_i \rightarrow V_j \) is dominant; and zero otherwise.

14. Return \( D_t \).

### 5.4 Construction of IBMDBN

We use mutation data given in (N.Beerenwinkel 2006). Ten samples constructed as explained in Section 5.2, from N.Beerenwinkel (2006), are given in Figures 5.6, 5.7, 5.8. In the reference paper authors construct hidden Markov tree models for longitudinal clonal HIV sequence data. The mutations they considered are 100I, 101E, 101Q, 103N, 108I, 190S, and 225H. We assign numbers 1, 2, \ldots, 7 to these mutations.
Figure 5.6: Samples of longitudinal clonal HIV sequence data
Figure 5.7: Samples of longitudinal clonal HIV sequence data

Figure 5.8: Samples of longitudinal clonal HIV sequence data
The count matrix $Y$ for the data is

$$
Y = \begin{pmatrix}
V_1 & V_2 & V_3 & V_4 & V_5 & V_6 & V_7 \\
V_1 & - & 3 & 0 & 1 & 0 & 2 & 1 \\
V_2 & 1 & - & 2 & 2 & 1 & 0 & 2 \\
V_3 & 1 & 1 & - & 1 & 2 & 0 & 0 \\
V_4 & 2 & 2 & 1 & - & 3 & 1 & 2 \\
V_5 & 1 & 1 & 1 & 0 & - & 1 & 1 \\
V_6 & 1 & 1 & 1 & 2 & 0 & - & 1 \\
V_7 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & - \\
\end{pmatrix}
$$

The matrix of transition probability intervals $D$, for $s = 1$, is

$$
D = \begin{pmatrix}
V_1 & V_2 & V_3 & V_4 \\
V_1 & - & [0.068, 0.091] & [0, 0.023] & [0.023, 0.045] \\
V_2 & [0.023, 0.045] & - & [0.045, 0.068] & [0.045, 0.068] \\
V_3 & [0.023, 0.045] & [0.023, 0.045] & - & [0.023, 0.045] \\
V_4 & [0.045, 0.068] & [0.045, 0.068] & [0.023, 0.045] & - \\
V_5 & [0.023, 0.045] & [0.023, 0.045] & [0.023, 0.045] & [0, 0.023] \\
V_6 & [0.023, 0.045] & [0.023, 0.045] & [0.023, 0.045] & [0.045, 0.068] \\
V_7 & [0.023, 0.045] & [0.023, 0.045] & [0, 0.023] & [0, 0.023] \\
\end{pmatrix}
$$
The matrix $D_t$ after comparison for dominant edges, for $s = 1$, is

\[
D_t = \begin{pmatrix}
V_1 & V_2 & V_3 & V_4 & V_5 & V_6 & V_7 \\
V_1 & - & 1 & 0 & 0 & 0 & 1 & 0 \\
V_2 & 0 & - & 1 & 1 & 0 & 0 & 1 \\
V_3 & 0 & 0 & - & 0 & 1 & 0 & 0 \\
V_4 & 1 & 1 & 0 & - & 1 & 0 & 1 \\
V_5 & 0 & 0 & 0 & 0 & - & 0 & 0 \\
V_6 & 0 & 0 & 0 & 1 & 0 & - & 0 \\
V_7 & 0 & 0 & 0 & 0 & 0 & 0 & -
\end{pmatrix}
\]
Figure 5.9: IBMDBN for the data

5.5 Results

The Dynamic Bayesian Network constructed using the proposed algorithm for the longitudinal clonal sequence data is given in Figure 5.9. The constructed DBN based on Imprecise Beta Model allows cycles in the mutation evolution. For example, in our IBMDBN we have cycles, namely, $V_1 \rightarrow V_6 \rightarrow V_4 \rightarrow V_1$ and $V_1 \rightarrow V_2 \rightarrow V_4 \rightarrow V_1$. In Genetics, especially in Oncogenesis, it is difficult to get samples of large sizes. And when the data requires to be collected (observed) over a long period of time, working with large sized data would be infeasible. Proposed algorithm works with samples of small sizes. This makes it possible to observe a very small number of patients for a long period of time.