1 Elements of Graphical Models

1.1 An Example

The AR(1) SV HMM provides a nice set of examples.

Recall the model in centered form, defined by a set of conditional distributions that imply the full joint density over all states and parameters:

\[ p(y_{1:n}, z_{0:n}, \gamma_{1:n}, \mu, \phi) = \prod_{t=1}^{n} p(y_t|z_t, \gamma_t)p(z_t|z_{t-1}, \mu, \phi)v(\gamma_t)p(\gamma_0)p(\mu, \phi) \]

and, as a detail, \( p(\mu, \phi) = p(\mu)p(\phi) \). The strong set of conditional independencies implicit here are encoded in the graph

\[ \cdot \cdot \cdot \gamma_t-1 \downarrow \gamma_t \downarrow \gamma_t+1 \downarrow \cdot \cdot \cdot \]
\[ \cdots \gamma_t-1 \downarrow \gamma_t \downarrow \gamma_t+1 \downarrow \cdots \]
\[ \cdots \downarrow \gamma_t \downarrow \gamma_t \downarrow \gamma_t \downarrow \cdots \]
\[ \cdots \downarrow \gamma_t \downarrow \gamma_t \downarrow \gamma_t \downarrow \cdots \]
\[ \downarrow (\mu, \phi) \]

Variables and parameters are nodes of the graph, and edges imply dependency: a directed edge, or arrow, from node \( a \) to \( b \) is associated with a conditional dependence of \( b \) on \( a \) in the joint distribution defined by the set of conditional distributions the graph describes. The graph is directed since all edges have arrows, and it is acyclic since there are no cycles resulting from the defined set of arrows - this latter fact results from the specification of the graph from the set of conditional distributions defined by the full, proper joint distribution over

\[ X_n = \{ y_{1:n}, z_{0:n}, \gamma_{1:n}, \mu, \phi, v \} \]

Example. Consider the node for \( y_t \). The existence of arrows from each of \( z_t \) and \( \gamma_t \) to \( y_t \), coupled with the lack of arrows to \( y_t \) from any other nodes in the graph, imply that the conditional distribution for \( y_t \) conditional on all other variables depends on, but only on, \( z_t \) and \( \gamma_t \). That is,

\[ y_t \perp \perp \{X_n \setminus (z_t, \gamma_t)\} \mid (z_t, \gamma_t) \]

The variables \( (z_t, \gamma_t) \) are parents of \( y_t \) in the directed graph; \( y_t \) is a child of each of \( z_t \) and \( \gamma_t \).

Some of the implied dependencies among variables in graphs are defined through common children. For example, \( \gamma_t \) and \( z_t \) are parents of \( y_t \), so share a common “bond” that implies association in the overall joint distribution. We have already seen the relevance of this in the Gibbs sampling MCMC analysis: the conditional posterior for the \( z_t \) depends on the \( \gamma_t \), and vice-versa, though the original DAG representation - consistent with the model specification - has no edges between \( \gamma_t \) and \( z_t \). The undirected graph that the model implies is shown below, now with the full set of relevant edges related to dependencies in the overall
Any conditional distribution, and the relevant variables associated with a specific node, can be “read off” this graph. A node has edges linking to its neighbours, and that set of neighbours of any node represent the variables on which it depends; conditioning on the neighbours renders the variable at the target node conditionally independent of all other variables.

1.2 General Structure and Terminology for Directed Graphical Models

- A joint distribution for \( x = (x_1, \ldots, x_p)' \) has p.d.f. \( p(x) \) that may be factorized in, typically, several or many ways. Fixing the order of the variables as above, the usual compositional form of the p.d.f. is

\[
p(x) = \prod_{i=1}^{p} p(x_i|x_{(i+1):p}).
\]

- In the \( i^{th} \) term here, it may be that some of the variables \( x_j, \) for \( j \in (i + 1):p, \) do not in fact play a role. The parents of \( x_i \) are those variables that do appear and play roles in defining the compositional conditional; that is, knowledge of all \( x_j \) for \( j \in pa(i) \subseteq \{(i + 1):p\} \) are necessary and sufficient to render \( x_i \) conditionally independent of \( x_k \) for \( k \in \{(i + 1):p\} \) but \( k \notin pa(i). \)

- Generally, a joint distribution may factorize as

\[
p(x) = \prod_{i=1}^{p} p(x_i|x_{pa(i)})
\]

in a number of ways as the indices are permuted. Each such factorization is consistent with a graphical representation in terms of a directed, acyclic graph (DAG) in which the \( p \) nodes correspond to the variables \( x_i, \) and directed edges (arrows) are drawn from node \( x_j \) to node \( x_i \) if, and only if, \( j \in pa(i). \)

- Simulation of \( p(x) \) via compositional sampling is easy given any specification in terms of a joint density factored over a DAG.

- Reversing arrows and adding additional directed edges is the root to identifying specific conditional distributions in a DAG. We have already seen in the SV HMM example how to just “read” a DAG representation of a complex, multivariate distribution to identify the variables relevant for a particular conditional of interest. There, for example, the conditional distribution (conditional posterior) for \( z_t \) given all other variables depends on the evident parents \( \{z_{t-1}, z_{t+1}, (\mu, \phi, v)\}, \) but also \( \{y_t, \gamma_t\}. \) The dependence on \( y_t \) is implied since \( y_t \) is a child of \( z_t \) in the DAG; the additional dependence on \( \gamma_t \) arises since \( z_t \) shares parentage of \( y_t \) with \( \gamma_t. \) This kind of development is clearly defined through conversion of DAGs to undirected graphs, discussed below. Its relevance in developments such as Gibbs sampling in complicated statistical models is apparent.
1.2.1 Multivariate Normal Example: Exchange Rate Data

Exploratory regression analysis of international exchange rate returns for 12 currencies yields the following set of coupled regressions in “triangular” form. The data are daily returns relative to the US dollar (USD) over a period of about three years to the end of 1996. The returns are centered so that the distribution can be considered zero mean, and, for currencies labeled as below, regressions are specified for currency \( i \) conditional on subset of currencies \( j > i \):

<table>
<thead>
<tr>
<th>Index</th>
<th>Country</th>
<th>Currency</th>
<th>Parents</th>
<th>( pa(i) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Canada</td>
<td>CAD (dollar)</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>New Zealand</td>
<td>NZD (dollar)</td>
<td>AUD</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>Australia</td>
<td>AUD (dollar)</td>
<td>GBP, DEM</td>
<td>6,12</td>
</tr>
<tr>
<td>4</td>
<td>Japan</td>
<td>JPY (yen)</td>
<td>CHF, DEM</td>
<td>10,12</td>
</tr>
<tr>
<td>5</td>
<td>Sweden</td>
<td>SEK (krone)</td>
<td>GBP, FRF</td>
<td>6,9</td>
</tr>
<tr>
<td>6</td>
<td>Britain</td>
<td>GBP (pound)</td>
<td>DEM</td>
<td>12</td>
</tr>
<tr>
<td>7</td>
<td>Spain</td>
<td>ESP (peseta)</td>
<td>BEF, FRF, DEM</td>
<td>8,9,12</td>
</tr>
<tr>
<td>8</td>
<td>Belgium</td>
<td>BEF (franc)</td>
<td>FRF, NLG, DEM</td>
<td>9,11,12</td>
</tr>
<tr>
<td>9</td>
<td>France</td>
<td>FRF (franc)</td>
<td>NLG</td>
<td>11</td>
</tr>
<tr>
<td>10</td>
<td>Switzerland</td>
<td>CHF (franc)</td>
<td>NLG, DEM</td>
<td>11,12</td>
</tr>
<tr>
<td>11</td>
<td>Netherlands</td>
<td>NLG (guilder)</td>
<td>DEM</td>
<td>12</td>
</tr>
<tr>
<td>12</td>
<td>Germany</td>
<td>DEM (mark)</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

This defines as set of conditional distributions that cohere to give the joint distribution with density defined as the product. The corresponding DAG is below.

**Figure 1:** Acyclic directed graph corresponding to the set of conditional regressions defining a joint distribution for exchange rate returns.

See Matlab code and example on the course web site.
1.3 Joint Distributions and Undirected Graphs

1.3.1 Undirected Graphical Models

Undirected graphs are useful representations of the dependencies in a joint distribution via qualitative display of the full set of complete conditional distributions \( p(x_i|x_{-i}) \). The set of conditioning variables \( x_j \) that in fact play a role in defining \( p(x_i|x_{-i}) \) are the *neighbours* of \( x_i \), and we write \( ne(i) \) for the set of indices of these neighbours. Hence \( x_i \perp \perp x_k \mid \{x_j : j \in ne(i)\} \) for all \( k \neq i \) such that \( k \neq ne(i) \). The complete conditionals are then the set

\[
p(x_i|x_{-i}) \equiv p(x_i|x_{ne(i)}), \quad i = 1, \ldots, p.
\]

Some key facts are that:

- Given a specified joint density, it is easy to generate the complete conditionals and neighbor sets simply by inspection, viz.

\[
p(x_i|x_{ne(i)}) \propto p(x),
\]

and identifying the normalization constant (that depends generally on \( x_{ne(i)} \)).

- Neighbourhood membership is symmetric in the sense that \( j \in ne(k) \) if, and only if, \( k \in ne(j) \).

1.3.2 Graphs from DAGs

In some analyses an initial representation of a joint distribution in terms of a factorization over a DAG is the initial basis for an undirected graphical model. Starting with a DAG, note that:

1. A directed edge from a node \( x_j \) to node \( x_i \) implies a dependency in the joint distribution.

   This indicates that directed edges - arrows - in a DAG induce undirected edges representing conditional dependencies in the implied graph. That is, if \( x_j \in pa(i) \) then \( x_j \in ne(i) \).

2. If a pair of nodes \( x_j \) and \( x_k \) are parents of a node \( x_i \) in a DAG then they are, by association through the child node \( x_i \), dependent.

   This indicates that all parents of a given node will be associated under the joint distribution - associated through common children if not already directly linked in the DAG. That is, for any \( j \) and \( k \) such that \( \{j, k\} \in pa(i) \) we have \( j \in ne(k) \) and \( k \in ne(j) \) as well as, of course, \( \{j, k\} \in ne(i) \).

These two steps define the process to generate the unique undirected graph from a specified DAG. Note that, generally, an undirected graph can be consistent with more than one DAG, since the DAG relates to one specific form of compositional factorization of the joint density and there may be several or many others. The choice of ordering of variables is key in defining the DAG, whereas the graph represents the full joint distribution without regard to ordering.

Dropping arrow heads is then the first step towards generating a graph from a DAG. The second step - inserting an undirected edge between any nodes that share parentage but are not already connected in the DAG - is referred to as *moralization* of the graph; all parents of any child must be married.
1.3.3 Multivariate Normal Example: Exchange Rate Data

The graph corresponding to the model Figure 1.2.1 is below.

![Graph Image]

Figure 2: Undirected graph corresponding to the DAG Figure 1.2.1 for the fitted distribution of exchange rate returns.

1.4 Factorization of Graphical Models

1.4.1 General Decompositions of Graphs

A multivariate distribution over an undirected graph can usually be factorized in a number of different ways, and breaking down the joint density into components can aid in interpretation and computations with the distribution. Key factorizations relate to the graph theoretic decompositions of arbitrary planar graphs.

Consider an undirected graph \( G = (V_G, E_G) \) defined by a set of nodes \( V_G \) and a set of edges \( E_G \). If two nodes \( v, u \in V_G \) are neighbours, then \( E_G \) contains the edge \((v, u)\) and we write \( v \sim u \) in \( G \). Hence \( E_G = \{(i, j) : i \sim j\} \). The relation \( v \sim u \) is of course symmetric and equivalent to each of \( j \in \text{ne}(i) \) and \( i \in \text{ne}(j) \). In the context of graphical models for joint distributions of the \( p \)-vector random variable, \( V = \{x_1, \ldots, x_p\} \).

- Consider any subset of nodes \( V_A \subseteq V_G \), and write \( E_A \) for the corresponding edge set in \( G \). Then \( (V_A, E_A) \) defines a subgraph - an undirected graph on nodes in \( V_A \).
- Consider two subgraphs \( A = (V_A, E_A) \) and \( B = (V_B, E_B) \) of \( G \), and identify the intersection \( S = (V_S, E_S) \) where \( V_S = V_A \cap V_B \). Then the subgraph \( S \) separates \( A \) and \( B \) in \( G \). Simply, there are no nodes \( v \in V_A \setminus V_S \) and \( u \in V_B \setminus V_S \), such that \( v \sim u \). The intersection graph \( S \) is called a separator of \( A \) and \( B \) in \( G \).
• Any subgraph $A$ is \textit{complete} if it has all possible edges: $E_A = \{ (i, j) : \text{ for all } i, j \in V_A \}$. Every node in a complete graph $A$ is a neighbour of every other such node. The subgraph is \textit{fully connected}.

• A subgraph $S \subseteq G$ is a \textit{clique} of $G$ if it is a \textit{maximally complete subgraph} of $G$. That is, $S$ is complete and we cannot add a further node that shares an edge with each node of $S$. Proper subgraphs of $S$ (all subgraphs apart from $S$ itself) are complete but not maximal. (Also, for cliques, we denote the graph by $S \equiv V_S$ since the edge set is, by definition, full and so the notation is redundant.)

Graphs can be decomposed, often in many different ways, into sequences of interconnecting subgraphs separated by complete subgraphs. Such a decomposition is known as a \textit{junction tree} representation of the graph - a tree since it defines an ordered sequence of subgraphs with a tree structure. Based on a specified (usually quite arbitrary) ordering of the nodes, a junction tree decomposition has the form

$$G \rightarrow J_G = [C_1, S_2, C_2, S_3, \cdots, C_{k-1}, S_k, C_k]$$

where:

• the $C_i$, $(i \in 1 \ldots k)$, are proper subgraphs of $G$ (each with at least 2 nodes); each of the $C_i$, may or may not be complete;
• each $S_i$ is a \textit{complete} subgraph of $G$;
• $S_i$ is the intersection of $C_i$ with all the previous components $C_{1:i-1}$, so that $S_i$ \textit{separates} the next component from the previous set.

The junction tree is a set of the $k$ \textit{prime component} subgraphs $C_{1:k}$ of $G$ linked by the sequence of $k - 1$ separating subgraphs $S_{2:k}$, and has the \textit{running intersection property} of the last bullet above.

![Figure 3: A graph $G$ on $p = 9$ nodes.](image)

\subsection*{1.4.2 Example: A 9 node graph}

The $p = 9$ node graph $G$ in Figure 3 can be decomposed sequentially as follows (see Figure 4):

• $S_2 = (2, 5)$ separates component $C_1 = (1, 2, 5)$ from the rest;
• further separation comes from $S_3 = (4, 6)$;
• then $S_4 = (2, 4)$ and $S_5 = (6, 7)$, in sequence, provide additional separation.
• This results in a set of $k = 5$ components with these 4 separators and the corresponding junction tree representation (Figure 5).
Figure 4: 9 node example: (A) $S_2 = (2, 5)$ separates component $C_1$ from the graph. (B) $S_3 = (4, 6)$ defines further separation. (C) The graph $G$ is separated into $k = 5$ components via the 4 separators.
1.4.3 Decomposition of Distributions over Graphs

If the graph $G$ represents the conditional dependency structure in $p(x)$ where $x = (x_1, \ldots, x_p)'$ and the nodes represent the $x_i$ variables, the joint p.d.f. factorizes corresponding to any decomposition of the graph into a junction tree. This is a general and powerful result that is related to the Hammersley-Clifford characterization of joint densities. To avoid unnecessary complications, suppose $p(x) > 0$ everywhere. Then, based on a junction tree representation

$$G \rightarrow J_G = [C_1, S_2, C_2, S_3, \ldots, C_{k-1}, S_k, C_k]$$

we have the density decomposition

$$p(x) = \frac{\prod_{i=1}^{k} p(x_{C_i})}{\prod_{i=2}^{k} p(x_{S_i})}$$

where $x_{C_i} = \{x_j : j \in C_i\}$ is the set of variables in component $i$, and $x_{S_i} = \{x_j : j \in S_i\}$ is the set of variables in separator $i$.

- As a simpler, shorthand notation for the above factorisation we write

$$p(x) = \frac{\prod_{C} p(x_C)}{\prod_{S} p(x_S)}.$$  

- That is, the joint density factors as a product of joint densities of variables within each prime component divided by the product of joint densities of variables within each separating complete subgraph. This is quite general.
One nice, intuitive way to view this decomposition is that the joint density is a product over all component densities, but that implied “double counting” of variables in separators requires a “correction” in terms of “taking out” the contributions from the densities on separators via division.

That there may be several or many such decompositions simply represents alternative factorizations of the density.

1.4.4 Example: The 9 node graph

In the example note that four of the five prime components are themselves complete, whereas one is not a complete subgraph (an “incomplete” prime component). A joint density \( p(x) \) in the graph in Figure 3 then has the representation

\[
p(x) = \frac{p(x_1, x_2, x_5)p(x_2, x_3, x_4)p(x_2, x_4, x_5, x_6)p(x_4, x_6, x_7)p(x_6, x_7, x_8, x_9)}{p(x_2, x_5)p(x_2, x_4)p(x_4, x_6)p(x_6, x_7)}.
\]

Some insight is gained by noting that this can be written as

\[
p(x) = \frac{p(x_1, x_2, x_5) p(x_2, x_3, x_4) p(x_2, x_4, x_5, x_6) p(x_4, x_6, x_7)}{p(x_2, x_5) p(x_2, x_4) p(x_4, x_6) p(x_6, x_7)} p(x_6, x_7, x_8, x_9)
\]

\[
= p(x_1| x_2, x_5) p(x_3| x_2, x_4) p(x_2, x_5| x_4, x_6) p(x_4| x_6, x_7) p(x_6, x_7, x_8, x_9),
\]

i.e., one specific compositional form corresponding to a DAG on the implied elements. This shows how the general component-separator decomposition can naturally yield representations of joint densities that are useful for simulation of the joint distribution as well as for investigating dependency structure. It is an example of how we can construct DAGs from graphs.

1.5 Additional Comments

Marginalization induces complexity in graphical models. If \( G = (V, E) \) is the graph for \( p(x_{1:p}) \), and \( A \subset x_{1:p} \), then the graph representing the marginal density \( p(x_A) \) will generally have more edges than the subgraph \( (A, E_A) \) of \( G \). The variables removed by marginalization generally induce edges representing the now “hidden” dependencies.

A simple example is the “star” graph defined by the joint density \( p(x_{1:p}) = p(x_1) \prod_{i=2}^{p} p(x_i|x_1) \). This graph encodes the dependencies \( x_i \perp x_j|x_1 \) for all \( i, j > 1 \). Here \( x_1 \) is a key variable inducing dependencies among all the others. On marginalisation over \( x_1 \) the result is a complete graph on \( x_{2:p} \), a much less “sparse” graph in this \( p - 1 \) dimensional margin than we see for the subgraph in the full \( p \) dimensions. Node \( x_1 \) separates all other nodes in the full graph.

1.6 Special Cases of Decomposable Graphs

Decomposable graphs are special but key examples.

- A decomposable graph \( G \) is a graph in which there are no edge cycles of path length four or more - the graph is triangulated. The 9 node example above is non-decomposable since it has a four cycle; modifying that graph to add one edge - between nodes 4 and 5 or between nodes 2 and 6 - leads to a triangulated and hence decomposable graph.

- One of the relevant features of decomposable graphs is that all prime components are complete - that is, any junction tree representation defines a graph decomposition as a sequence of intersecting complete subgraphs.
2 Gaussian Graphical Models

2.1 Introductory Comments

In the Gaussian case, lack of an edge between nodes $i$ and $j$ in a graph implies a conditional independence consistent with the implied zero in the precision matrix of the multivariate normal distribution. Hence:

- Decomposition into a junction tree implies that, within each separator $S_i$, all variables are conditionally dependent since the separator is a complete subgraph.
- The precision matrix $\Omega$, having zeros corresponding to missing edges, is sparse; the variance matrix $\Sigma = \Omega^{-1}$ will usually not have off-diagonal zeros; non-zero pairwise correlations usually exist between conditionally independent variables, being induced by intervening neighbours, i.e., by paths of more than one edge in the graph. If $\Omega$ is block-diagonal (or its rows/columns can be permuted to make it so) then the same sparsity pattern is shared by $\Sigma$; in such cases we have marginal independence between components.
- If the graph is decomposable, we can view each component subgraph in the junction tree (all prime components and separators) as defining a set of subsets of jointly normal variables, within each of which there is a full set of conditional dependencies. Each component and each separator forms a clique.

A good reference to the Gaussian graphical models, hyper-Wishart models and computation – and a very nice entry point to the broader area of graphical modelling, is the paper by B. Jones, A. Dobra, C. Carvalho, C. Hans, C. Carter and M. West (2005), Experiments in stochastic computation for high-dimensional graphical models, Statistical Science 20, 388-400. This builds on the extensive general theory that is well laid-out in Lauritzen, 1996, Graphical Models (O.U.P).

For the remainder of the development here we focus on decomposable models. The framework extends to non-decomposable graphs, with subsequent computational challenges in understanding these distributions when $p$ is more than small integer value. Some major open questions - questions at the frontiers of research in statistics and computational science and the interfaces with machine learning - are those of exploring spaces of graphs themselves to make inferences on the dependence structure itself. The aim of this section is to introduce the ideas and some basic elements of inference on a specified graphical model that underlies any such broader investigation, and provides some idea of how structured multivariate analysis on graphs begins.

2.1.1 Example

Consider an example in $p = 6$ dimensions with multivariate normal distribution for $p(x|\Sigma)$ for $x = x_{1:6}$. Suppose the precision matrix $\Omega$ is exactly

$$\Omega = \begin{pmatrix}
0.80 & -0.32 & 0 & 0 & 0 & 0 \\
-0.32 & 0.54 & 0.09 & -0.08 & 0 & 0 \\
0 & 0.09 & 2.50 & -0.20 & -0.09 & 0.04 \\
0 & -0.08 & -0.20 & 0.58 & -0.06 & -0.05 \\
0 & 0 & -0.09 & -0.06 & 1.20 & -0.10 \\
0 & 0 & 0.04 & -0.05 & -0.10 & 0.32 \\
\end{pmatrix},$$
so that (with entries to only three decimal places) the variance matrix is
\[
\Sigma = \Omega^{-1} = \begin{pmatrix}
1.650 & 1.000 & -0.026 & 0.132 & 0.007 & 0.026 \\
1.000 & 2.510 & -0.065 & 0.331 & 0.017 & 0.065 \\
-0.026 & -0.065 & 0.418 & 0.137 & 0.036 & -0.020 \\
0.132 & 0.331 & 0.137 & 1.860 & 0.127 & 0.313 \\
0.007 & 0.017 & 0.036 & 0.127 & 0.852 & 0.282 \\
0.026 & 0.065 & -0.020 & 0.313 & 0.282 & 3.260
\end{pmatrix}.
\]

The graph, in Figure 6, is decomposable with prime components \(C_1 = (1 : 2), C_2 = (2 : 4)\) and \(C_3 = (3 : 6)\); the separators are \(S_2 = (2)\) and \(S_3 = (3 : 4)\). The variances of the cliques – namely \(\Sigma_{C_1} = V(x_{1:2})\), \(\Sigma_{C_2} = V(x_{2:4})\), \(\Sigma_{C_3} = V(x_{3:6})\), \(\Sigma_{S_2} = V(x_2)\), and \(\Sigma_{S_3} = V(x_{3:4})\), can just be read out of \(\Sigma\) directly. Thus
\[
\Sigma_{C_1} = \begin{pmatrix}
1.650 & 1.000 \\
1.000 & 2.510
\end{pmatrix}, \quad \Sigma_{C_2} = \begin{pmatrix}
2.510 & -0.065 & 0.331 \\
-0.065 & 0.418 & 0.137 \\
0.331 & 0.137 & 1.860
\end{pmatrix}
\]
and
\[
\Sigma_{C_3} = \begin{pmatrix}
0.418 & 0.137 & 0.036 & -0.020 \\
0.137 & 1.860 & 0.127 & 0.313 \\
0.036 & 0.127 & 0.852 & 0.282 \\
-0.020 & 0.313 & 0.282 & 3.260
\end{pmatrix},
\]
with
\[
\Sigma_{S_2} = 2.510 \quad \text{and} \quad \Sigma_{S_3} = \begin{pmatrix}
0.418 & 0.137 \\
0.137 & 1.860
\end{pmatrix}.
\]

The intersections of separators and components implies that the variance matrices of the separators are sub-matrices of prime components: \(\Sigma_{S_2}\) is the (upper left block – in this case, just a scalar) diagonal entry of \(\Sigma_{C_2}\) and a (lower right) diagonal entry of \(\Sigma_{C_1}\); similarly, \(\Sigma_{S_3}\) is a block diagonal component of each of \(\Sigma_{C_2}\) and \(\Sigma_{C_3}\).

The joint density for \(x\) decomposes on the graph as
\[
p(x|\Sigma) = p(x_{1:2}|\Sigma_{C_1})p(x_{2:4}|\Sigma_{C_2})p(x_{3:6}|\Sigma_{C_3})p(x_2|\Sigma_{S_2})p(x_{3:4}|\Sigma_{S_3}),
\]
where each term is the corresponding marginal normal density of \(p(x)\). One clear reduced representation is simply the compositional (DAG) form \(p(x_1|x_2)p(x_2|x_{3:4})p(x_{3:6})\).
2.2 Data and Likelihood Functions

Now consider the normal distribution on any decomposable graph \( G \) with the joint p.d.f.

\[
p(x|\Sigma) = \frac{\prod_{C} p(x_C|\Sigma_C)}{\prod_{S} p(x_S|\Sigma_S)}
\]

where we are recognising that each clique, whether a prime component or a separator, has its corresponding normal distribution. We should properly indicate the dependence on \( G \) in the conditioning, but leave that out for clarify in notation and because we are only considering one graph here.

- The \( \Sigma \) are the block diagonal components of the \( p \times p \) variance matrix \( \Sigma \) and they intersect: since \( S_i \subset C_i \), then \( \Sigma_{S_i} \) must be a (block) sub-matrix of \( \Sigma_{C_i} \) and will be a sub-matrix of at least one of \( \Sigma_{C_1}, \ldots, \Sigma_{C_{i-1}} \). The numerical example above demonstrates this.
- As earlier, write \( J_G \) for the full set of cliques, \( J_G = [C_1, S_2, C_2, \ldots, S_k, C_k] \). We will use \( H \in J_G \) to denote any one of the components or separators, and \( h \in 1 : p \) for the corresponding index set of variables.
- As with the full multivariate model it is more convenient to parametrise and work in terms of precisions, so define \( \Omega_H = \Sigma_H^{-1} \) for each for each clique \( H \in J_G \). Then

\[
p(x|\Omega) \equiv p(x|\Sigma) = c \frac{\prod_{C} |\Omega_C|^{n/2} \exp\{-\text{trace}(\Omega_C x_C x'_C)/2\}}{\prod_{S} |\Omega_S|^{n/2} \exp\{-\text{trace}(\Omega_S x_S x'_S)/2\}} \tag{1}
\]

where \( c > 0 \).
- Now assume we observe a random sample of size \( n \) from \( p(x) \); for notational convenience denote the full set of \( n \) observations by \( X \). The joint p.d.f. is then the product over samples \( i = 1 : n \) of terms given by equation (1). On any clique \( H \in J_G \) with variables \( h \in 1 : p \), extend the notation to denote by \( x_{i,h} \) the sample \( i \) values on these variables, and define the clique sample variance matrix \( V_H \) by the usual

\[
V_H = n^{-1} \sum_{i=1}^{n} x_{i,H} x'_{i,H}.
\]

It then follows trivially that the full likelihood function is

\[
p(X|\Omega) \propto \frac{\prod_{C} |\Omega_C|^{n/2} \exp\{-\text{trace}(\Omega_C (nV_C))/2\}}{\prod_{S} |\Omega_S|^{n/2} \exp\{-\text{trace}(\Omega_S (nV_S))/2\}} \tag{2}
\]

That is, the likelihood function for \( \Omega = \Sigma^{-1} \), factorises over the graph.
- For any clique \( H \in J_G \), the MLEs of \( \Sigma_H \) and \( \Omega_H \) based on the data only for the variables in \( H \) are the usual forms

\[
\hat{\Sigma}_H = V_H, \quad \hat{\Omega}_H = V_H^{-1}.
\]
- Maximising the full likelihood of equation (2) with respect to \( \Sigma \) (equivalently \( \Omega \)) can be done analytically; see Lauritzen, 1996, *Graphical Models* (O.U.P.). It can be shown that the MLE conditional on the graph \( G \) is

\[
\hat{\Omega} = \sum_{C} [\hat{\Omega}_A]^0 - \sum_{S} [\hat{\Omega}_S]^0
\]

where the notation \([\cdot]^0\) indicates the extension of the argument matrix to a \( p \times p \) by filling in other elements as zeros. That is, we construct the MLE \( \hat{\Omega} \) as follows:
– initialise as the zero matrix;
– visit each prime component $C$ and add in the contribution $\hat{\Omega}_C = V_C^{-1}$ to the entries corresponding to the elements in this component;
– visit each separator $S$ and subtract out the contribution $\hat{\Omega}_S = V_S^{-1}$ from the corresponding elements.

The subtraction steps account for the “double counting” implied by the intersection of components. The MLE $\hat{\Sigma} = \hat{\Omega}^{-1}$ may then be deduced. The construction makes clear that the constraints imposed by $G$ are respected by the MLE; that is, for any two variables not connected by an edge in $G$, the corresponding $(i, j)$ element of $\hat{\Omega}$ is zero. Some of the parsimony resulting in estimation by conditioning on the structure of a graph begins to be evident.

2.2.1 Hyper-Wishart Distributions over Graphs

The class of hyper-Wishart distributions for $\Omega$, and the corresponding/implied hyper-inverse Wishart distributions for $\Sigma$, extends the standard Wishart/inverse-Wishart theory to graphs. The major area of Gaussian graphical modelling and highly structured multivariate analysis – including methods for exploring the (high-dimensional) space of graphs to learn about relevant structure – i.e., posterior distributions over graphs as well as inference on any given graph – begins with the HW/HIW distributions. We close here, but simply note a couple of details that begin to give insight into the field.

- Under an HW prior on $\Omega$, the implied prior and posterior distribution for $\Omega$ – hence, equivalently, $\Sigma$ – factorise over $G$ in a manner that corresponds to the factorisation of the data density and resulting likelihood function.
- The corresponding prior and posterior densities for the full component matrices $\Omega_S$ and $\Omega_C$ on each clique of $G$ are themselves regular Wishart distributions. This shows how the standard normal/Wishart theory is implied on each subset of variables within any clique, and shows how the hyper-theory extends the standard theory in concept.
- As one limiting example, a reference prior distribution in the HW/HIW class on the above graph $G$ leads immediately to the MLE as described above as the posterior mean for $\Omega$. This is a neat, direct extension of the full normal/Wishart result, and rationalises the use of the MLEs $\hat{\Omega}$ and $\hat{\Sigma}$ on structured graphical models as Bayesian estimates.

B. Jones et al (2005), Experiments in stochastic computation for high-dimensional graphical models, *Statistical Science* 20, 388-400, is an excellent start to this important and new and exciting field.