Chapter 1

Introduction

Our everyday experiences can be summarised as a series of decisions to take actions which manipulate our environment in some way or other. We base our decisions on the results of predictions or inferences of quantities that have some bearing on our quality of life, and we come to arrive at these inferences based on *models* of what we expect to observe. Models are designed to capture salient trends or regularities in the observed data with a view to predicting future events. Sometimes the models can be constructed with existing expertise, but for the majority of real applications the data are far too complex or the underlying processes not nearly well enough understood for the modeller to design a perfectly accurate model. If this is the case, we can hope only to design models that are simplifying approximations of the true processes that generated the data.

For example, the data might be a time series of the price of stock recorded every day for the last six months, and we would like to know whether to buy or sell stock today. This decision, and its particulars, depend on what the price of the stock is likely to be a week from now. There are obviously a very large number of factors that influence the price and these do so to varying degrees and in convoluted and complex ways. Even in the unlikely scenario that we knew exactly how all these factors affected the price, we would still have to gather every piece of data for each one and process it all in a short enough time to decide our course of action. Another example is trying to predict the best location to drill for oil, knowing the positions of existing drill sites in the region and their yields. Since we are unable to probe deep beneath the Earth's surface, we need to rely on a model of the geological processes that gave rise to the yields in those sites for which we have data, in order to be able to predict the best location.

The *machine learning* approach to modelling data constructs models by beginning with a flexible model specified by a set of *parameters* and then finds the setting of these model parameters that explains or fits the data best. The idea is that if we can explain our observations well, then we should also be confident that we can predict future observations well. We might also hope

that the particular setting of the best-fit parameters provides us with some understanding of the underlying processes. The procedure of fitting model parameters to observed data is termed *learning* a model.

Since our models are simplifications of reality there will inevitably be aspects of the data which cannot be modelled exactly, and these are considered noise. Unfortunately it is often difficult to know which aspects of the data are relevant for our inference or prediction tasks, and which aspects should be regarded as noise. With a sufficiently complex model, parameters can be found to fit the observed data exactly, but any predictions using this best-fit model will be sub-optimal as it has erroneously fitted the noise instead of the trends. Conversely, too simple a model will fail to capture the underlying regularities in the data and so will also produce sub-optimal inferences and predictions. This trade-off between the complexity of the model and its generalisation performance is well studied, and we return to it in section 1.2.

The above ideas can be formalised using the concept of probability and the rules of Bayesian inference. Let us denote the data set by $\mathbf{y}$, which may be made up of several variables indexed by j: $\mathbf{y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_j, \ldots, \mathbf{y}_J\}$. For example, $\mathbf{y}$ could be the data from an oil well for which the variables might be measurements of the type of oil found, the geographical location of the well, its average monthly yield, its operational age, and a host of other measurable quantities regarding its local geological characteristics. Generally each variable can be real-valued or discrete. Machine learning approaches define a *generative model* of the data through a set of parameters $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_K\}$ which define a probability distribution over data, $p(\mathbf{y} \,|\, \boldsymbol{\theta})$. One approach to learning the model then involves finding the parameters $\boldsymbol{\theta}^*$ such that

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \; p(\mathbf{y} \,|\, \boldsymbol{\theta}) \; . \tag{1.1}$$

This process is often called *maximum likelihood* learning as the parameters $\boldsymbol{\theta}^*$ are set to maximise the likelihood of $\boldsymbol{\theta}$, which is probability of the observed data under the model. The generative model may also include *latent* or *hidden* variables, which are unobserved yet interact through the parameters to generate the data. We denote the hidden variables by $\mathbf{x}$, and the probability of the data can then be written by summing over the possible settings of the hidden states:

$$p(\mathbf{y} \,|\, \boldsymbol{\theta}) = \sum_{\mathbf{x}} p(\mathbf{x} \,|\, \boldsymbol{\theta}) p(\mathbf{y} \,|\, \mathbf{x}, \boldsymbol{\theta}) \; , \tag{1.2}$$

where the summation is replaced by an integral for those hidden variables that are real-valued. The quantity (1.2) is often called the *incomplete-data likelihood*, and the summand in (1.2) correspondingly called the *complete-data likelihood*. The interpretation is that with hidden variables in the model, the observed data is an incomplete account of all the players in the model.

For a particular parameter setting, it is possible to infer the states of the hidden variables of the model, having observed data, using Bayes' rule:

$$p(\mathbf{x} \mid \mathbf{y}, \boldsymbol{\theta}) = \frac{p(\mathbf{x} \mid \boldsymbol{\theta}) p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta})}{p(\mathbf{y} \mid \boldsymbol{\theta})} \,. \tag{1.3}$$

This quantity is known as the *posterior* distribution over the hidden variables. In the oil well example we might have a hidden variable for the amount of oil remaining in the reserve, and this can be inferred based on observed measurements such as the operational age, monthly yield and geological characteristics, through the generative model with parameters $\boldsymbol{\theta}$. The term $p(\mathbf{x} \mid \boldsymbol{\theta})$ is a *prior* probability of the hidden variables, which could be set by the modeller to reflect the distribution of amounts of oil in wells that he or she would expect. Note that the probability of the data in (1.2) appears in the denominator of (1.3). Since the hidden variables are by definition unknown, finding $\boldsymbol{\theta}^*$ becomes more difficult, and the model is learnt by alternating between estimating the posterior distribution over hidden variables for a particular setting of the parameters and then re-estimating the best-fit parameters given that distribution over the hidden variables. This procedure is the well-known expectation-maximisation (EM) algorithm and is discussed in more detail in section 2.2.

Given that the parameters themselves are unknown quantities we can treat them as random variables. This is the *Bayesian* approach to uncertainty, which treats all uncertain quantities as random variables and uses the laws of probability to manipulate those uncertain quantities. The proper Bayesian approach attempts to integrate over the possible settings of all uncertain quantities rather than optimise them as in (1.1). The quantity that results from integrating out both the hidden variables and the parameters is termed the *marginal likelihood*:

$$p(\mathbf{y}) = \int d\boldsymbol{\theta} \; p(\boldsymbol{\theta}) \sum_{\mathbf{x}} p(\mathbf{x} \mid \boldsymbol{\theta}) p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}) \,, \tag{1.4}$$

where $p(\boldsymbol{\theta})$ is a prior over the parameters of the model. We will see in section 1.2 that the marginal likelihood is a key quantity used to choose between different models in a Bayesian model selection task. Model selection is a necessary step in understanding and representing the data that we observe. The diversity of the data available to machine learners is ever increasing thanks to the advent of large computational power, networking capabilities and the technologies available to the scientific research communities. Furthermore, expertise and techniques of analysis are always improving, giving rise to ever more diverse and complicated models for representing this data. In order to 'understand' the data with a view to making predictions based on it, we need to whittle down our models to one (or a few) to which we can devote our limited computational and conceptual resources. We can use the rules of Bayesian probability theory to entertain several models and choose between them in the light of data. These steps necessarily involve managing the marginal likelihood.

Unfortunately the marginal likelihood, $p(\mathbf{y})$, is an intractable quantity to compute for almost all models of interest (we will discuss why this is so in section 1.2.1, and see several examples in the course of this thesis). Traditionally, the marginal likelihood has been approximated either using analytical methods, for example the Laplace approximation, or via sampling-based approaches such as Markov chain Monte Carlo. These methods are reviewed in section 1.3. This thesis is devoted to one particular method of approximation, *variational Bayes*, sometimes referred to as *ensemble learning*. The variational Bayesian method constructs a lower bound on the marginal likelihood, and attempts to optimise this bound using an iterative scheme that has intriguing similarities to the standard expectation-maximisation algorithm. There are other variational methods, for example those based on Bethe and Kikuchi free energies, which for the most part are approximations rather than bounds; these are briefly discussed in the final chapter.

Throughout this thesis we assume that the reader is familiar with the basic concepts of probability and integral and differential calculus. Included in the appendix are reference tables for some of the more commonly used probability distributions.

The rest of this chapter reviews some key methods relevant to Bayesian model inference and learning. Section 1.1 reviews the use of graphical models as a tool for visualising the probabilistic relationships between the variables in a model and explains how efficient algorithms for computing the posterior distributions of hidden variables as in (1.3) can be designed which exploit independence relationships amongst the variables. In section 1.2, we address the issue of model selection in a Bayesian framework, and explain why the marginal likelihood is the key quantity for this task, and how it is intractable to compute. Since all Bayesian reasoning needs to begin with some prior beliefs, we examine different schools of thought for expressing these priors in section 1.2.2, including *conjugate*, *reference*, and *hierarchical* priors. In section 1.3 we review several practical methods for approximating the marginal likelihood, which we shall be comparing to variational Bayes in the following chapters. Finally, section 1.4 briefly summarises the remaining chapters of this thesis.

1.1 Probabilistic inference

Bayesian probability theory provides a language for representing beliefs and a calculus for manipulating these beliefs in a coherent manner. It is an extension of the formal theory of logic which is based on axioms that involve propositions that are true or false. The rules of probability theory involve propositions which have *plausibilities* of being true or false, and can be arrived at on the basis of just three *desiderata*: (1) degrees of plausibility should be represented by real numbers; (2) plausibilities should have qualitative correspondence with common sense; (3) different routes to a conclusion should yield the same result. It is quite astonishing that from just these desiderata, the product and sum rules of probability can be mathematically derived

(Cox, 1946). Cox showed that plausibilities can be measured on any scale and it is possible to transform them onto the canonical scale of probabilities that sum to one. For good introductions to probability theory the reader is referred to Pearl (1988) and Jaynes (2003).

Statistical modelling problems often involve large numbers of interacting random variables and it is often convenient to express the dependencies between these variables graphically. In particular such graphical models are an intuitive tool for visualising *conditional independency* relationships between variables. A variable a is said to be conditionally independent of b, given c if and only if $p(a, b \,|\, c)$ can be written $p(a \,|\, c)p(b \,|\, c)$. By exploiting conditional independence relationships, graphical models provide a backbone upon which it has been possible to derive efficient message-propagating algorithms for conditioning and marginalising variables in the model given observation data (Pearl, 1988; Lauritzen and Spiegelhalter, 1988; Jensen, 1996; Heckerman, 1996; Cowell et al., 1999; Jordan, 1999). Many standard statistical models, especially Bayesian models with hierarchical priors (see section 1.2.2), can be expressed naturally using probabilistic graphical models. This representation can be helpful in developing both sampling methods (section 1.3.6) and exact inference methods such as the junction tree algorithm (section 1.1.2) for these models. All of the models used in this thesis have very simple graphical model descriptions, and the theoretical results derived in chapter 2 for variational Bayesian approximate inference are phrased to be readily applicable to general graphical models.

1.1.1 Probabilistic graphical models: directed and undirected networks

A graphical model expresses a family of probability distributions on sets of variables in a model. Here and for the rest of the thesis we use the variable $\mathbf{z}$ to denote all the variables in the model, be they observed or unobserved (hidden). To differentiate between observed and unobserved variables we partition $\mathbf{z}$ into $\mathbf{z} = \{\mathbf{x}, \mathbf{y}\}$ where $\mathbf{x}$ and $\mathbf{y}$ are the sets of unobserved and observed variables, respectively. Alternatively, the variables are indexed by the subscript j, with $j \in \mathcal{H}$ the set of indices for unobserved (hidden) variables and $j \in \mathcal{V}$ the set of indices for observed variables. We will later introduce a further subscript, i, which will denote which data point out of a data set of size n is being referred to, but for the purposes of the present exposition we consider just a single data point and omit this further subscript.

Each arc between two nodes in the graphical model represents a probabilistic connection between two variables. We use the terms 'node' and 'variable' interchangeably. Depending on the pattern of arcs in the graph and their type, different independence relations can be represented between variables. The pattern of arcs is commonly referred to as the *structure* of the model.

The arcs between variables can be all *directed* or all *undirected*. There is a class of graphs in which some arcs are directed and some are undirected, commonly called *chain graphs*, but these are not reviewed here. Undirected graphical models, also called *Markov networks* or *Markov*

random fields, express the probability distribution over variables as a product over *clique potentials*:

$$p(\mathbf{z}) = \frac{1}{\mathcal{Z}} \prod_{j=1}^{J} \psi_j(C_j(\mathbf{z})) \,, \tag{1.5}$$

where $\mathbf{z}$ is the set of variables in the model, $\{C_j\}_{j=1}^{J}$ are *cliques* of the graph, and $\{\psi_j\}_{j=1}^{J}$ are a set of clique potential functions each of which returns a non-negative real value for every possible configuration of settings of the variables in the clique. Each clique is defined to be a fully connected subgraph (that is to say each clique C_j selects a subset of the variables in $\mathbf{z}$), and is usually *maximal* in the sense that there are no other variables whose inclusion preserves its fully connected property. The cliques can be overlapping, and between them cover all variables such that $\{C_1(\mathbf{z}) \cup \cdots \cup C_J(\mathbf{z})\} = \mathbf{z}$. Here we have written a normalisation constant, $\mathcal{Z}$, into the expression (1.5) to ensure that the total probability of all possible configurations sums to one. Alternatively, this normalisation can be absorbed into the definition of one or more of the potential functions. Markov networks can express a very simple form of independence relationship: two sets of nodes A and B are conditionally independent from each other given a third set of nodes C, if all paths connecting any node in A to any node in B via a sequence of arcs are separated by any node (or group of nodes) in C. Then C is said to *separate* A from B. The *Markov blanket* for the node (or set of nodes) A is defined as the smallest set of nodes C, such that A is conditionally independent of all other variables not in C, given C.

Directed graphical models, also called *Directed Acyclic Graphs* (DAGs), or *Bayesian networks*, express the probability distribution over J variables, $\mathbf{z} = \{\mathbf{z}_j\}_{j=1}^{J}$, as a product of conditional probability distributions on each variable:

$$p(\mathbf{z}) = \prod_{j=1}^{J} p(\mathbf{z}_j \,|\, \mathbf{z}_{\mathbf{pa}(j)}) \,, \tag{1.6}$$

where $\mathbf{z}_{\mathbf{pa}(j)}$ is the set of variables that are *parents* of the node j in the graph. A node a is said to be a parent of a node b if there is a directed arc from a to b, and in which case b is said to be a *child* of a. In necessarily recursive definitions: the *descendents* of a node are defined to include its children and its childrens' descendents; and the *ancestors* of a node are its parents and those parents' ancestors. Note that there is no need for a normalisation constant in (1.6) because by the definition of the conditional probabilities it is equal to one. A *directed path* between two nodes a and b is a sequence of variables such that every node is a parent of the following node in the sequence. An *undirected path* from a to b is any sequence of nodes such that every node is a parent or child of the following node. An *acyclic* graph is a graphical model in which there exist no directed paths including the same variable more than once. The semantics of a Bayesian network can be summarised as: each node is conditionally independent from its non-descendents given its parents.

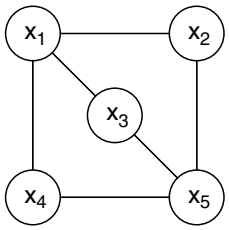
More generally, we have the following representation of independence in Bayesian networks: two sets of nodes A and B are conditionally independent given the set of nodes C if they are *d-separated* by C (here the *d-* prefix stands for *directed*). The nodes A and B are d-separated by C if, along every undirected path from A to B, there exists a node d which satisfies *either* of the following conditions: either (i) d has converging arrows (i.e. d is the child of the previous node and the parent of the following node in the path) *and* neither d nor its descendents are in C; or (ii) d does not have converging arrows and is in C. From the above definition of the Markov blanket, we find that for Bayesian networks the minimal Markov blanket for a node is given by the union of its parents, its children, *and* the parents of its children. A more simple rule for d-separation can be obtained using the idea of the 'Bayes ball' (Shachter, 1998). Two sets of nodes A and B are conditionally dependent given C if there exists a path by which the Bayes ball can reach a node in B from a node in A (or vice-versa), where the ball can move according to the following rules: it can pass through a node in the conditioning set C provided the entry and exit arcs are a pair of arrows converging on that node; similarly, it can only pass through every node in the remainder of the graph provided it does so on non-converging arrows. If there exist no such linking paths, then the sets of nodes A and B are conditionally independent given C.

Undirected models tend to be used in the physics and vision communities, where the systems under study can often be simply expressed in terms of many localised potential functions. The nature of the interactions often lack causal or direct probabilistic interpretations, and instead express degrees of agreement, compatibility, constraint or frustration between nodes. In the artificial intelligence and statistics communities directed graphs are more popular as they can more easily express underlying causal generative processes that give rise to our observations. For more detailed examinations of directed and undirected graphs see Pearl (1988).

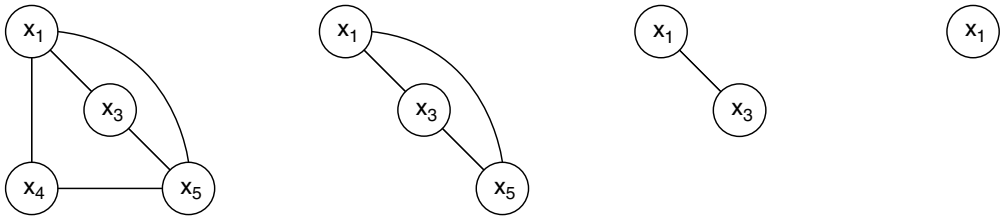
1.1.2 Propagation algorithms

The conditional independence relationships discussed in the previous subsection can be exploited to design efficient message-passing algorithms for obtaining the posterior distributions over hidden variables given the observations of some other variables, which is called inference. In this section we briefly present an inference algorithm for Markov networks, called the *junction tree* algorithm. We will explain at the end of this subsection why it suffices to present the inference algorithm for the undirected network case, since the inference algorithm for a directed network is just a special case.

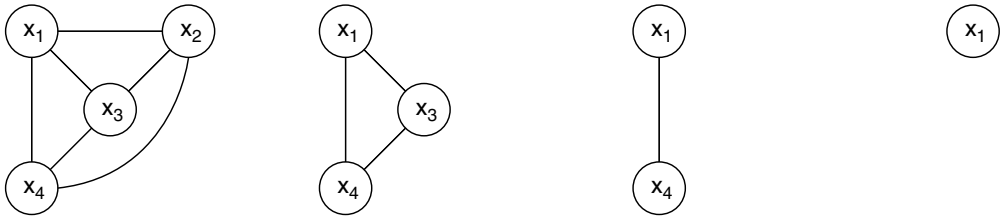
For data in which every variable is observed there is no inference problem for hidden variables, and learning for example the maximum likelihood (ML) parameters for the model using (1.1) often consists of a straightforward optimisation procedure. However, as we will see in chapter 2, if some of the variables are hidden this complicates finding the ML parameters. The common



(a) Original Markov network.



(b) One possible elimination ordering: (x_2, x_4, x_5, x_3)



(c) Another possible elimination ordering: (x_5, x_2, x_3, x_4).

Figure 1.1: **(a)** The original Markov network; **(b)** The sequence of intermediate graphs resulting from eliminating (integrating out) nodes to obtain the marginal on x_1 — see equations (1.9–1.14); **(c)** Another sequence of graphs resulting from a different elimination ordering, which results in a suboptimal inference algorithm.

practice in these cases is to utilise expectation-maximisation (EM) algorithms, which in their E step require the computation of at least certain properties of the posterior distribution over the hidden variables.

We illustrate the basics of inference using a simple example adapted from Jordan and Weiss (2002). Figure 1.1(a) shows a Markov network for five variables $\mathbf{x} = \{x_1, \ldots, x_5\}$, each of which is discrete and takes on k possible states. Using the Markov network factorisation given by (1.5), the probability distribution over the variables can be written as a product of potentials defined over five cliques:

$$p(\mathbf{x}) = p(x_1, \ldots, x_5) = \frac{1}{\mathcal{Z}} \psi(x_1, x_2)\psi(x_1, x_3)\psi(x_1, x_4)\psi(x_2, x_5)\psi(x_3, x_5)\psi(x_4, x_5) \ , \quad (1.7)$$

where we have included a normalisation constant $\mathcal{Z}$ to allow for arbitrary clique potentials. Note that in this graph 1.1(a) the maximal cliques are all pairs of nodes connected by an arc, and therefore the potential functions are defined over these same pairs of nodes. Suppose we wanted to obtain the marginal distribution $p(x_1)$, given by

$$p(x_1) = \frac{1}{\mathcal{Z}} \sum_{x_2} \sum_{x_3} \sum_{x_4} \sum_{x_5} \psi(x_1, x_2)\psi(x_1, x_3)\psi(x_1, x_4)\psi(x_2, x_5)\psi(x_3, x_5)\psi(x_4, x_5) \; . \tag{1.8}$$

At first glance this requires k^5 computations, since there are k^4 summands to be computed for each of the k settings of the variable x_1. However this complexity can be reduced by exploiting the conditional independence structure in the graph. For example, we can rewrite (1.8) as

$$p(x_1) = \frac{1}{\mathcal{Z}} \sum_{x_2} \sum_{x_3} \sum_{x_4} \sum_{x_5} \psi(x_1, x_3)\psi(x_3, x_5)\psi(x_1, x_4)\psi(x_4, x_5)\psi(x_1, x_2)\psi(x_2, x_5) \tag{1.9}$$

$$= \frac{1}{\mathcal{Z}} \sum_{x_3} \psi(x_1, x_3) \sum_{x_5} \psi(x_3, x_5) \sum_{x_4} \psi(x_1, x_4)\psi(x_4, x_5) \sum_{x_2} \psi(x_1, x_2)\psi(x_2, x_5) \tag{1.10}$$

$$= \frac{1}{\mathcal{Z}} \sum_{x_3} \psi(x_1, x_3) \sum_{x_5} \psi(x_3, x_5) \sum_{x_4} \psi(x_1, x_4)\psi(x_4, x_5) m_2(x_1, x_5) \tag{1.11}$$

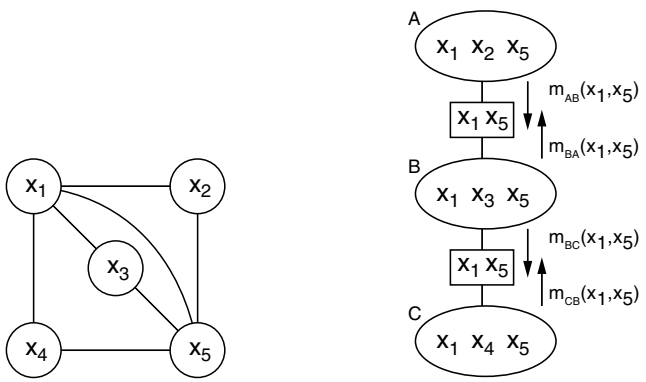
$$= \frac{1}{\mathcal{Z}} \sum_{x_3} \psi(x_1, x_3) \sum_{x_5} \psi(x_3, x_5) m_4(x_1, x_5) m_2(x_1, x_5) \tag{1.12}$$

$$= \frac{1}{\mathcal{Z}} \sum_{x_3} \psi(x_1, x_3) m_5(x_1, x_3) \tag{1.13}$$

$$= \frac{1}{\mathcal{Z}} m_1(x_1) \tag{1.14}$$

where each 'message' $m_j(x_\cdot, \dots)$ is a new potential obtained by *eliminating* the jth variable, and is a function of all the variables linked to that variable. By choosing this ordering (x_2, x_4, x_5, x_3) for summing over the variables, the most number of variables in any summand is three, meaning that the complexity has been reduced to $\mathcal{O}(k^3)$ for each possible setting of x_1, which results in an overall complexity of $\mathcal{O}(k^4)$.

This process can be described by the sequence of graphs resulting from the repeated application of a *triangulation* algorithm (see figure 1.1(b)) following these four steps: (i) choose a node x_j to eliminate; (ii) find all potentials ψ and any messages m that may reference this node; (iii) define a new potential m_j that is the sum with respect to x_j of the product of these potentials; (iv) remove the node x_j and *replace it with edges* connecting each of its neighbours — these represent the dependencies from the new potentials. This process is repeated until only the variables of interest remain, as shown in the above example. In this way marginal probabilities of single variables or joint probabilities over several variables can be obtained. Note that the second elimination step in figure 1.1(b)), that of marginalising out x_4, introduces a new message $m_4(x_1, x_5)$ but since there is already an arc connecting x_1 and x_5 we need not add a further one.



(a) (b)

Figure 1.2: **(a)** The triangulated graph corresponding to the elimination ordering in figure 1.1(b); **(b)** the corresponding junction tree including maximal cliques (ovals), separators (rectangles), and the messages produced in belief propagation.

The ordering chosen for this example is optimal; different orderings of elimination may result in suboptimal complexity. For example, figure 1.1(c) shows the process of an elimination ordering (x_5, x_2, x_3, x_4) which results in a complexity $\mathcal{O}(k^5)$. In general though, it is an NP-hard problem to find the optimal ordering of elimination that minimises the complexity. If all the nodes have the same cardinality, the optimal elimination ordering is independent of the functional forms on the nodes and is purely a graph-theoretic property.

We could use the above elimination algorithm repeatedly to find marginal probabilities for each and every node, but we would find that we had needlessly computed certain messages several times over. We can use the junction tree algorithm to compute all the messages we might need just once. Consider the graph shown in figure 1.2(a) which results from retaining all edges that were either initially present or added during the elimination algorithm (using the ordering in our worked example). Alongside in figure 1.2(b) is the junction tree for this graph, formed by linking the maximal cliques of the graph, of which there are three, labelled A, B and C. In between the clique nodes are *separators* for the junction tree, which contain nodes that are common to both the cliques attached to the separator, that is to say $\mathcal{S}_{AB} = \mathcal{C}_A \cap \mathcal{C}_B$. Here we use calligraphic $\mathcal{C}$ to distinguish these cliques from the original maximal cliques in the network 1.1(a). For a triangulated graph it is always possible to obtain such a singly-connected graph, or tree (to be more specific, it is always then possible to obtain a tree that satisfies the *running intersection property*, which states that if a variable appears in two different cliques, then it should also appear in every clique in the path between the two cliques). The so-called 'messages' in the elimination algorithm can now be considered as messages sent from one clique to another in the junction tree. For example, the message $m_2(x_1, x_5)$ produced in equation (1.11) as a result of summing over x_2 can be identified with the message $m_{AB}(x_1, x_5)$ that clique A sends to clique B. Similarly, the message $m_4(x_1, x_5)$ in (1.12) resulting from summing over x_4 is identified

with the message $m_{CB}(x_1, x_5)$ that C passes on to B. To complete the marginalisation to obtain $p(x_1)$, the clique B *absorbs* the incoming messages to obtain a joint distribution over its variables (x_1, x_3, x_5), and then marginalises out x_3 and x_5 in either order. Included in figure 1.2(b) are two other messages, $m_{BA}(x_1, x_5)$ and $m_{BC}(x_1, x_5)$, which would be needed if we wanted the marginal over x_2 or x_4, respectively.

For general junction trees it can be shown that the message that clique r sends to clique s is a function of the variables in their separator, $\mathcal{S}_{rs}(\mathbf{x})$, and is given by

$$m_{rs}(\mathcal{S}_{rs}(\mathbf{x})) = \sum_{\mathcal{C}_r(\mathbf{x})\setminus\mathcal{S}_{rs}(\mathbf{x})} \psi_r(\mathcal{C}_r(\mathbf{x})) \prod_{t\in\mathcal{N}(r)\setminus s} m_{tr}(\mathcal{S}_{tr}(\mathbf{x})) \,, \tag{1.15}$$

where $\mathcal{N}(r)$ are the set of neighbouring cliques of clique r. In words, the message from r to s is formed by: taking the product of all messages r has received from elsewhere other than s, multiplying in the potential ψ_r, and then summing out all those variables in r which are not in s.

The joint probability of the variables within clique r is obtained by combining messages into clique r with its potential:

$$p(\mathcal{C}_r(\mathbf{x})) \propto \psi_r(\mathcal{C}_r(\mathbf{x})) \prod_{t\in\mathcal{N}(r)} m_{tr}(\mathcal{S}_{tr}(\mathbf{x})) \,. \tag{1.16}$$

Note that from definition (1.15) a clique is unable to send a message until it has received messages from all other cliques except the receiving one. This means that the message-passing protocol must begin at the leaves of the junction tree and move inwards, and then naturally the message-passing moves back outwards to the leaves. In our example problem the junction tree has a very trivial structure and happens to have both separators containing the same variables (x_1, x_5).

Here we have explained how inference in a Markov network is possible: (i) through a process of triangulation the junction tree is formed; (ii) messages (1.15) are then propagated between junction tree cliques until all cliques have received and sent all their messages; (iii) clique marginals (1.16) can then be computed; (iv) individual variable marginals can be obtained by summing out other variables in the clique. The algorithm used for inference in a Bayesian network (which is directed) depends on whether it is singly- or multiply-connected (a graph is said to be singly-connected if it includes no pairs of nodes with more than one path between them, and multiply-connected otherwise). For singly-connected networks, an exactly analogous algorithm can be used, and is called *belief propagation*. For multiply-connected networks, we first require a process to convert the Bayesian network into a Markov network, called *moralisation*. We can then form the junction tree after a triangulation process and perform the same message-passing algorithm. The process of moralisation involves adding an arc between any variables sharing the same child (i.e. co-parents), and then dropping the directionality of all arcs.

Moralisation does not introduce any further conditional independence relationships into the graph, and in this sense the resulting Markov network is able to represent a superset of the probability distributions representable by the Bayesian network. Therefore, having derived the inference procedure for the more general Markov network, we already have the result for the Bayesian network as a special case.

1.2 Bayesian model selection

In this thesis we are primarily concerned with the task of model selection, or structure discovery. We use the term 'model' and 'model structure' to denote a variety of things, some already mentioned in the previous sections. A few particular examples of model selection tasks are given below:

Structure learning In probabilistic graphical models, each graph implies a set of conditional independence statements between the variables in the graph. The model structure learning problem is inferring the conditional independence relationships that hold given a set of (complete or incomplete) observations of the variables. Another related problem is learning the direction of the dependencies, i.e. the causal relationships between variables ($A \to B$, or $B \to A$).

Input dependence A special case of this problem is input variable selection in regression. Selecting which input (i.e. explanatory) variables are needed to predict the output (i.e. response) variable in the regression can be equivalently cast as deciding whether each input variable is a parent (or, more accurately, an ancestor) of the output variable in the corresponding directed graph.

Cardinality Many statistical models contain discrete nominal latent variables. A model structure learning problem of interest is then choosing the cardinality of each discrete latent variable. Examples of this problem include deciding how many mixture components are required in a finite mixture model, or how many hidden states are needed in a hidden Markov model.

Dimensionality Other statistical models contain real-valued vectors of latent variables. The dimensionality of this latent vector is usually unknown and needs to be inferred. Examples include choosing the intrinsic dimensionality in a probabilistic principal components analysis (PCA), or factor analysis (FA) model, or in a linear-Gaussian state-space model.

In the course of this thesis we tackle several of the above model selection problems using Bayesian learning. The machinery and tools for Bayesian model selection are presented in the following subsection.

1.2.1 Marginal likelihood and Occam's razor

An obvious problem with using maximum likelihood methods (1.1) to learn the parameters of models such as those described above is that the probability of the data will generally be greater for more complex model structures, leading to overfitting. Such methods fail to take into account model complexity. For example, inserting an arc between two variables in a graphical model can only help the model give higher probability to the data. Common ways for avoiding overfitting have included early stopping, regularisation, and cross-validation. Whilst it is possible to use cross-validation for simple searches over model size and structures — for example, if the search is limited to a single parameter that controls the model complexity — for more general searches over many parameters cross-validation is computationally prohibitive.

A Bayesian approach to learning starts with some prior knowledge or assumptions about the model structure — for example the set of arcs in the Bayesian network. This initial knowledge is represented in the form of a prior probability distribution over model structures. Each model structure has a set of parameters which have prior probability distributions. In the light of observed data, these are updated to obtain a posterior distribution over models and parameters. More formally, assuming a prior distribution over models structures $p(m)$ and a prior distribution over the parameters for each model structure $p(\boldsymbol{\theta} \,|\, m)$, observing the data set $\mathbf{y}$ induces a posterior distribution over models given by Bayes' rule:

$$p(m \,|\, \mathbf{y}) = \frac{p(m)p(\mathbf{y} \,|\, m)}{p(\mathbf{y})} \,. \tag{1.17}$$

The most probable model or model structure is the one that maximises $p(m \,|\, \mathbf{y})$. For a given model structure, we can also compute the posterior distribution over the parameters:

$$p(\boldsymbol{\theta} \,|\, \mathbf{y}, m) = \frac{p(\mathbf{y} \,|\, \boldsymbol{\theta}, m)p(\boldsymbol{\theta} \,|\, m)}{p(\mathbf{y} \,|\, m)} \,, \tag{1.18}$$

which allows us to quantify our uncertainty about parameter values after observing the data. We can also compute the density at a new data point $\mathbf{y}'$, obtained by averaging over both the uncertainty in the model structure and in the parameters,

$$p(\mathbf{y}' \,|\, \mathbf{y}) = \sum_m \int d\boldsymbol{\theta} \; p(\mathbf{y}' \,|\, \boldsymbol{\theta}, m, \mathbf{y}) p(\boldsymbol{\theta} \,|\, m, \mathbf{y}) p(m \,|\, \mathbf{y}) \,, \tag{1.19}$$

which is known as the *predictive distribution*.

The second term in the numerator of (1.17) is called the *marginal likelihood*, and results from integrating the likelihood of the data over all possible parameter settings under the prior:

$$p(\mathbf{y} \,|\, m) = \int d\boldsymbol{\theta} \; p(\mathbf{y} \,|\, \boldsymbol{\theta}, m) p(\boldsymbol{\theta} \,|\, m) \,. \tag{1.20}$$

In the machine learning community this quantity is sometimes referred to as the *evidence* for model m, as it constitutes the data-dependent factor in the posterior distribution over models (1.17). In the absence of an informative prior $p(m)$ over possible model structures, this term alone will drive our model inference process. Note that this term also appears as the normalisation constant in the denominator of (1.18). We can think of the marginal likelihood as the average probability of the data, where the average is taken with respect to the model parameters drawn from the prior $p(\boldsymbol{\theta})$.

Integrating out the parameters penalises models with more degrees of freedom since these models can *a priori* model a larger range of data sets. This property of Bayesian integration has been called *Occam's razor*, since it favours simpler explanations (models) for the data over complex ones (Jefferys and Berger, 1992; MacKay, 1995). Having more parameters may impart an advantage in terms of the ability to model the data, but this is offset by the cost of having to code those extra parameters under the prior (Hinton and van Camp, 1993). The overfitting problem is avoided simply because no parameter in the pure Bayesian approach is actually *fit* to the data. A caricature of Occam's razor is given in figure 1.3, where the horizontal axis denotes all possible data sets to be modelled, and the vertical axis is the marginal probability $p(\mathbf{y} \mid m)$ under each of three models of increasing complexity. We can relate the complexity of a model to the range of data sets it can capture. Thus for a simple model the probability is concentrated over a small range of data sets, and conversely a complex model has the ability to model a wide range of data sets.

Since the marginal likelihood as a function of the data $\mathbf{y}$ should integrate to one, the simple model can give a higher marginal likelihood to those data sets it can model, whilst the complex model gives only small marginal likelihoods to a wide range of data sets. Therefore, given a data set, $\mathbf{y}$, on the basis of the marginal likelihood it is possible to discard both models that are too complex and those that are too simple. In these arguments it is tempting, but not correct, to associate the complexity of a model with the number of parameters it has: it is easy to come up with a model with many parameters that can model only a limited range of data sets, and also to design a model capable of capturing a huge range of data sets with just a single parameter (specified to high precision).

We have seen how the marginal likelihood is an important quantity in Bayesian learning, for computing quantities such as Bayes factors (the ratio of two marginal likelihoods, Kass and Raftery, 1995), or the normalising constant of a posterior distribution (known in statistical physics as the 'partition function' and in machine learning as the 'evidence'). Unfortunately the marginal likelihood is a very difficult quantity to compute because it involves integrating over all parameters and latent variables, which is usually such a high dimensional and complicated integral that most simple approximations fail catastrophically. We will see in section 1.3 some of the approximations to the marginal likelihood and will investigate variational Bayesian approximations in the following chapter.

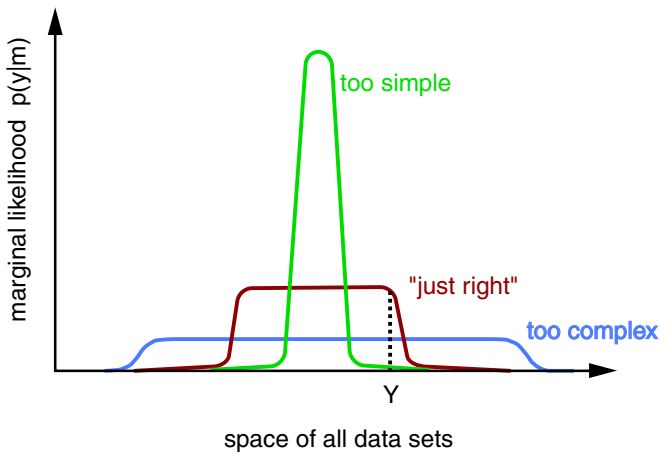


Figure 1.3: Caricature depicting Occam's razor (adapted from MacKay, 1995). The horizontal axis denotes all possible data sets of a particular size and the vertical axis is the marginal likelihood for three different model structures of differing complexity. Simple model structures can model certain data sets well but cannot model a wide range of data sets; complex model structures can model many different data sets but, since the marginal likelihood has to integrate to one, will necessarily not be able to model all simple data sets as well as the simple model structure. Given a particular data set (labelled Y), model selection is possible because model structures that are too simple are unlikely to generate the data set in question, while model structures that are too complex can generate many possible data sets, but again, are unlikely to generate that particular data set at random.

It is important to keep in mind that a realistic model of the data might need to be complex. It is therefore often advisable to use the most 'complex' model for which it is possible to do inference, ideally setting up priors that allow the limit of infinitely many parameters to be taken, rather than to artificially limit the number of parameters in the model (Neal, 1996; Rasmussen and Ghahramani, 2001). Although we do not examine any such infinite models in this thesis, we do return to them in the concluding comments of chapter 7.

Bayes' theorem provides us with the posterior over different models (1.17), and we can combine predictions by weighting them according to the posterior probabilities (1.19). Although in theory we should average over all possible model structures, in practice computational or representational constraints may make it necessary to select a single most probable structure by maximising $p(m \,|\, \mathbf{y})$. In most problems we may also have good reason to believe that the marginal likelihood is strongly peaked, and so the task of model selection is then justified.

1.2.2 Choice of priors

Bayesian model inference relies on the marginal likelihood, which has at its core a set of prior distributions over the parameters of each possible structure, $p(\boldsymbol{\theta} \,|\, m)$. Specification of parameter priors is obviously a key element of the Bayesian machinery, and there are several diverse

schools of thought when it comes to assigning priors; these can be loosely categorised into *subjective*, *objective*, and *empirical* approaches. We should point out that all Bayesian approaches are necessarily subjective in the sense that any Bayesian inference first requires some expression of prior knowledge $p(\boldsymbol{\theta})$. Here the emphasis is not on whether we use a prior or not, but rather *what* knowledge (if any) is conveyed in $p(\boldsymbol{\theta})$. We expand on these three types of prior design in the following paragraphs.

Subjective priors

The subjective Bayesian attempts to encapsulate prior knowledge as fully as possible, be it in the form of previous experimental data or expert knowledge. It is often difficult to articulate qualitative experience or beliefs in mathematical form, but one very convenient and analytically favourable class of subjective priors are *conjugate* priors in the *exponential family*. Generally speaking, a prior is conjugate if the posterior distribution resulting from multiplying the likelihood and prior terms is of the same form as the prior. Expressed mathematically:

$$f(\boldsymbol{\theta} \,|\, \tilde{\boldsymbol{\mu}}) = p(\boldsymbol{\theta} \,|\, \mathbf{y}) \propto f(\boldsymbol{\theta} \,|\, \boldsymbol{\mu}) p(\mathbf{y} \,|\, \boldsymbol{\theta}) \,, \tag{1.21}$$

where $f(\boldsymbol{\theta} \,|\, \boldsymbol{\mu})$ is some probability distribution specified by a parameter (or set of parameters) $\boldsymbol{\mu}$. Conjugate priors have at least three advantages: first, they often lead to analytically tractable Bayesian integrals; second, if computing the posterior in (1.21) is tractable, then the modeller can be assured that subsequent inferences, based on using the posterior as prior, will also be tractable; third, conjugate priors have an intuitive interpretation as expressing the results of previous (or indeed imaginary) observations under the model. The latter two advantages are somewhat related, and can be understood by observing that the only likelihood functions $p(\mathbf{y} \,|\, \boldsymbol{\theta})$ for which conjugate prior families exist are those belonging to general *exponential family* models. The definition of an exponential family model is one that has a likelihood function of the form

$$p(\mathbf{y}_i \,|\, \boldsymbol{\theta}) = g(\boldsymbol{\theta})\, f(\mathbf{y}_i)\, e^{\boldsymbol{\phi}(\boldsymbol{\theta})^\top \mathbf{u}(\mathbf{y}_i)} \,, \tag{1.22}$$

where $g(\boldsymbol{\theta})$ is a normalisation constant:

$$g(\boldsymbol{\theta})^{-1} = \int d\mathbf{y}_i \; f(\mathbf{y}_i)\, e^{\boldsymbol{\phi}(\boldsymbol{\theta})^\top \mathbf{u}(\mathbf{y}_i)} \,, \tag{1.23}$$

and we have used the subscript notation $\mathbf{y}_i$ to denote each data point (not each variable!). We assume that n data points arrive independent and identically distributed (i.i.d.) such that the probability of the data $\mathbf{y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$ under this model is given by $p(\mathbf{y} \,|\, \boldsymbol{\theta}) = \prod_{i=1}^n p(\mathbf{y}_i \,|\, \boldsymbol{\theta})$.

Here $\boldsymbol{\phi}(\boldsymbol{\theta})$ is a vector of so-called *natural parameters*, and $\mathbf{u}(\mathbf{y}_i)$ and $f(\mathbf{y}_i)$ are functions defining the exponential family. Now consider the conjugate prior:

$$p(\boldsymbol{\theta} \,|\, \eta, \boldsymbol{\nu}) = h(\eta, \boldsymbol{\nu})\, g(\boldsymbol{\theta})^{\eta}\, e^{\boldsymbol{\phi}(\boldsymbol{\theta})^{\top}\boldsymbol{\nu}} \,, \tag{1.24}$$

where η and $\boldsymbol{\nu}$ are parameters of the prior, and $h(\eta, \boldsymbol{\nu})$ is an appropriate normalisation constant. The conjugate prior contains the same functions $g(\boldsymbol{\theta})$ and $\boldsymbol{\phi}(\boldsymbol{\theta})$ as in (1.22), and the result of using a conjugate prior can then be seen by substituting (1.22) and (1.24) into (1.21), resulting in:

$$p(\boldsymbol{\theta} \,|\, \mathbf{y}) \propto p(\boldsymbol{\theta} \,|\, \eta, \boldsymbol{\nu}) p(\mathbf{y} \,|\, \boldsymbol{\theta}) \propto p(\boldsymbol{\theta} \,|\, \tilde{\eta}, \tilde{\boldsymbol{\nu}}) \,, \tag{1.25}$$

where $\tilde{\eta} = \eta + n$ and $\tilde{\boldsymbol{\nu}} = \boldsymbol{\nu} + \sum_{i=1}^{n} \mathbf{u}(\mathbf{y}_i)$ are the new parameters for the posterior distribution which has the *same functional form* as the prior. We have omitted some of the details, as a more general approach will be described in the following chapter (section 2.4). The important point to note is that the parameters of the prior can be viewed as the number (or amount), η, and the 'value', $\boldsymbol{\nu}$, of imaginary data observed prior to the experiment (by 'value' we in fact refer to the vector of sufficient statistics of the data). This correspondence is often apparent in the expressions for predictive densities and other quantities which result from integrating over the posterior distribution, where statistics gathered from the data are simply augmented with prior quantities. Therefore the knowledge conveyed by the conjugate prior is specific and clearly interpretable. On a more mathematical note, the attraction of the conjugate exponential family of models is that they can represent probability densities with a finite number of sufficient statistics, and are closed under the operation of Bayesian inference. Unfortunately, a conjugate analysis becomes difficult, and for the majority of interesting problems impossible, for models containing hidden variables $\mathbf{x}_i$.

Objective priors

The objective Bayesian's goal is in stark contrast to a subjectivist's approach. Instead of attempting to encapsulate rich knowledge into the prior, the objective Bayesian tries to impart as *little* information as possible in an attempt to allow the data to carry as much weight as possible in the posterior distribution. This is often called 'letting the data speak for themselves' or 'prior ignorance'. There are several reasons why a modeller may want to resort to the use of objective priors (sometimes called non-informative priors): often the modeller has little expertise and does not want to sway the inference process in any particular direction unknowingly; it may be difficult or impossible to elicit expert advice or translate expert opinions into a mathematical form for the prior; also, the modeller may want the inference to be robust to misspecifications of the prior. It turns out that expressing such vagueness or ignorance is in fact quite difficult, partly because the very concept of 'vagueness' is itself vague. Any prior expressed on the parameters

has to follow through and be manifest in the posterior distribution in some way or other, so this quest for uninformativeness needs to be more precisely defined.

One such class of noninformative priors are *reference priors*. These originate from an information theoretic argument which asks the question: "which prior should I use such that I maximise the expected amount of information about a parameter that is provided by observing the data?". This expected information can be written as a function of $p(\theta)$ (we assume θ is one-dimensional):

$$I(p(\theta), n) = \int d\mathbf{y}^{(n)}\; p(\mathbf{y}^{(n)}) \int d\theta\; p(\theta \,|\, \mathbf{y}^{(n)}) \ln \frac{p(\theta \,|\, \mathbf{y}^{(n)})}{p(\theta)} \,, \tag{1.26}$$

where we use $\mathbf{y}^{(n)}$ to make it obvious that the data set is of size n. This quantity is strictly positive as it is an expected Kullback-Leibler (KL) divergence between the parameter posterior and parameter prior, where the expectation is taken with respect to the underlying distribution of the data $\mathbf{y}^{(n)}$. Here we assume, as before, that the data arrive i.i.d. such that $\mathbf{y}^{(n)} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$ and $p(\mathbf{y}^{(n)} \,|\, \theta) = \prod_{i=1}^{n} p(\mathbf{y}_i \,|\, \theta)$. Then the n-reference prior is defined as the prior that maximises this expected information from n data points:

$$p_n(\theta) = \arg\max_{p(\theta)}\; I(p(\theta), n) \,. \tag{1.27}$$

Equation (1.26) can be rewritten directly as a KL divergence:

$$I(p(\theta), \mathbf{y}^{(n)}) = \int d\theta\; p(\theta) \ln \frac{f_n(\theta)}{p(\theta)} \,, \tag{1.28}$$

where the function $f_n(\theta)$ is given by

$$f_n(\theta) = \exp\left[\int d\mathbf{y}^{(n)}\; p(\mathbf{y}^{(n)} \,|\, \theta) \ln p(\theta \,|\, \mathbf{y}^{(n)}) \right] \,, \tag{1.29}$$

and n is the size of the data set $\mathbf{y}$. A naive solution that maximises (1.28) is

$$p_n(\theta) \propto f_n(\theta) \,, \tag{1.30}$$

but unfortunately this is only an implicit solution for the n-reference prior as $f_n(\theta)$ (1.29) is a function of the prior through the term $p(\theta \,|\, \mathbf{y}^{(n)})$. Instead, we make the approximation for large n that the posterior distribution $p(\theta \,|\, \mathbf{y}^{(n)}) \propto p(\theta) \prod_{i=1}^{n} p(\mathbf{y}_i \,|\, \theta)$ is given by $p^*(\theta \,|\, \mathbf{y}^{(n)}) \propto \prod_{i=1}^{n} p(\mathbf{y}_i \,|\, \theta)$, and write the reference prior as:

$$p(\theta) \propto \lim_{n\to\infty} \frac{f_n^*(\theta)}{f_n^*(\theta_0)} \,, \tag{1.31}$$

where $f_n^*(\theta)$ is the expression (1.29) using the approximation to the posterior $p^*(\theta \,|\, \mathbf{y}^{(n)})$ in place of $p(\theta \,|\, \mathbf{y}^{(n)})$, and θ_0 is a fixed parameter (or subset of parameters) used to normalise the

limiting expression. For discrete parameter spaces, it can be shown that the reference prior is uniform. More interesting is the case of real-valued parameters that exhibit asymptotic normality in their posterior (see section 1.3.2), where it can be shown that the reference prior coincides with Jeffreys' prior (see Jeffreys, 1946),

$$p(\theta) \propto h(\theta)^{1/2} \,, \tag{1.32}$$

where $h(\boldsymbol{\theta})$ is the Fisher information

$$h(\theta) = \int d\mathbf{y}_i \; p(\mathbf{y}_i \,|\, \theta) \left[-\frac{\partial^2}{\partial \theta^2} \ln p(\mathbf{y}_i \,|\, \theta) \right] \,. \tag{1.33}$$

Jeffreys' priors are motivated by requiring that the prior is invariant to one-to-one reparameterisations, so this equivalence is intriguing. Unfortunately, the multivariate extensions of reference and Jeffreys' priors are fraught with complications. For example, the form of the reference prior for one parameter can be different depending on the order in which the remaining parameters' reference priors are calculated. Also multivariate Jeffreys' priors are not consistent with their univariate equivalents. As an example, consider the mean and standard deviation parameters of a Gaussian, (μ, σ). If μ is known, both Jeffreys' and reference priors are given by $p(\sigma) \propto \sigma^{-1}$. If the standard deviation is known, again both Jeffreys' and reference priors over the mean are given by $p(\mu) \propto 1$. However, if neither the mean nor the standard deviation are known, the Jeffreys' prior is given by $p(\mu, \sigma) \propto \sigma^{-2}$, which does not agree with the reference prior $p(\mu, \sigma) \propto \sigma^{-1}$ (here the reference prior happens not to depend on the ordering of the parameters in the derivation). This type of ambiguity is often a problem in defining priors over multiple parameters, and it is often easier to consider other ways of specifying priors, such as hierarchically. A more in depth analysis of reference and Jeffreys' priors can be found in Bernardo and Smith (1994, section 5.4).

Empirical Bayes and hierarchical priors

When there are many common parameters in the vector $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_K)$, it often makes sense to consider each parameter as being drawn from the same prior distribution. An example of this would be the prior specification of the means of each of the Gaussian components in a mixture model — there is generally no a priori reason to expect any particular component to be different from another. The parameter prior is then formed from integrating with respect to a hyperprior with hyperparameter γ:

$$p(\boldsymbol{\theta} \,|\, \gamma) = \int d\gamma \; p(\gamma) \prod_{k=1}^{K} p(\theta_k \,|\, \gamma) \,. \tag{1.34}$$

Therefore, each parameter is independent *given* the hyperparameter, although they are dependent marginally. Hierarchical priors are useful even when applied only to a single parameter,

often offering a more intuitive interpretation for the parameter's role. For example, the precision parameter ν for a Gaussian variable is often given a (conjugate) gamma prior, which itself has two hyperparameters (a_γ, b_γ) corresponding to the shape and scale of the prior. Interpreting the marginal distribution of the variable in this generative sense is often more intuitively appealing than simply enforcing a Student-t prior. Hierarchical priors are often designed using conjugate forms (described above), both for analytical ease and also because previous knowledge can be readily expressed.

Hierarchical priors can be easily visualised using directed graphical models, and there will be many examples in the following chapters. The phrase *empirical Bayes* refers to the practice of optimising the hyperparameters (e.g. γ) of the priors, so as to maximise the marginal likelihood of a data set $p(\mathbf{y} \mid \gamma)$. In this way Bayesian learning can be seen as maximum marginal likelihood learning, where there are always distributions over the parameters, but the hyperparameters are optimised just as in maximum likelihood learning. This practice is somewhat suboptimal as it ignores the uncertainty in the hyperparameter γ. Alternatively, a more coherent approach is to define priors over the hyperparameters and priors on the parameters of those priors, etc., to the point where at the top level the modeller is content to leave those parameters unoptimised. With sufficiently vague priors at the top level, the posterior distributions over intermediate parameters should be determined principally by the data. In this fashion, no parameters are actually ever fit to the data, and all predictions and inferences are based on the posterior distributions over the parameters.

1.3 Practical Bayesian approaches

Bayes' rule provides a means of updating the distribution over parameters from the prior to the posterior distribution in light of observed data. In theory, the posterior distribution captures all information inferred from the data about the parameters. This posterior is then used to make optimal decisions or predictions, or to select between models. For almost all interesting applications these integrals are analytically intractable, and are inaccessible to numerical integration techniques — not only do the computations involve very high dimensional integrals, but for models with parameter symmetries (such as mixture models) the integrand can have exponentially many modes.

There are various ways we can tackle this problem. At one extreme we can restrict ourselves only to models and prior distributions that lead to tractable posterior distributions and integrals for the marginal likelihoods and predictive densities. This is highly undesirable since it inevitably leads us to lose prior knowledge and modelling power. More realistically, we can approximate the exact answer.

1.3.1 Maximum a posteriori (MAP) parameter estimates

The simplest approximation to the posterior distribution is to use a point estimate, such as the maximum a posteriori (MAP) parameter estimate,

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} \; p(\boldsymbol{\theta})p(\mathbf{y} \,|\, \boldsymbol{\theta}) \;, \tag{1.35}$$

which chooses the model with highest posterior probability density (the mode). Whilst this estimate does contain information from the prior, it is by no means completely Bayesian (although it is often erroneously claimed to be so) since the mode of the posterior may not be representative of the posterior distribution at all. In particular, we are likely (in typical models) to be over-confident of predictions made with the MAP model, since by definition *all* the posterior probability mass is contained in models which give poorer likelihood to the data (modulo the prior influence). In some cases it might be argued that instead of the MAP estimate it is sufficient to specify instead a set of *credible regions* or *ranges* in which most of the probability mass for the parameter lies (connected credible regions are called credible ranges). However, both point estimates and credible regions (which are simply a collection of point estimates) have the drawback that they are not unique: it is always possible to find a one-to-one monotonic mapping of the parameters such that any particular parameter setting is at the mode of the posterior probability density in that mapped space (provided of course that that value has non-zero probability density under the prior). This means that two modellers with identical priors and likelihood functions will in general find different MAP estimates if their parameterisations of the model differ.

The key ingredient in the Bayesian approach is then not just the use of a prior but the fact that all variables that are unknown are averaged over, i.e. that uncertainty is handled in a coherent way. In this way is it not important which parameterisation we adopt because the parameters are integrated out.

In the rest of this section we review some of the existing methods for approximating marginal likelihoods. The first three methods are analytical approximations: the Laplace method (Kass and Raftery, 1995), the Bayesian Information Criterion (BIC; Schwarz, 1978), and the criterion due to Cheeseman and Stutz (1996). All these methods make use of the MAP estimate (1.35), and in some way or other try to account for the probability mass about the mode of the posterior density. These methods are attractive because finding the MAP estimate is usually a straightforward procedure. To almost complete the toolbox of practical methods for Bayesian learning, there follows a brief survey of sampling-based approximations, such as importance sampling and Markov chain Monte Carlo methods. We leave the topic of variational Bayesian learning until the next chapter, where we will look back to these approximations for comparison.

1.3.2 Laplace's method

By Bayes' rule, the posterior over parameters $\boldsymbol{\theta}$ of a model m is

$$p(\boldsymbol{\theta} \,|\, \mathbf{y}, m) = \frac{p(\boldsymbol{\theta} \,|\, m)\, p(\mathbf{y} \,|\, \boldsymbol{\theta}, m)}{p(\mathbf{y} \,|\, m)} \,. \tag{1.36}$$

Defining the logarithm of the numerator as

$$t(\boldsymbol{\theta}) \equiv \ln \left[p(\boldsymbol{\theta} \,|\, m)\, p(\mathbf{y} \,|\, \boldsymbol{\theta}, m) \right] = \ln p(\boldsymbol{\theta} \,|\, m) + \sum_{i=1}^{n} \ln p(\mathbf{y}_i \,|\, \boldsymbol{\theta}, m) \,, \tag{1.37}$$

the *Laplace approximation* (Kass and Raftery, 1995; MacKay, 1995) makes a local Gaussian approximation around a MAP parameter estimate $\hat{\boldsymbol{\theta}}$ (1.35). The validity of this approximation is based on the large data limit and some regularity conditions which are discussed below. We expand $t(\boldsymbol{\theta})$ to second order as a Taylor series about this point:

$$t(\boldsymbol{\theta}) = t(\hat{\boldsymbol{\theta}}) + (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^\top \left. \frac{\partial t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} + \frac{1}{2!} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^\top \left. \frac{\partial^2 t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^\top} \right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) + \ldots \tag{1.38}$$

$$\approx t(\hat{\boldsymbol{\theta}}) + \frac{1}{2} (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})^\top H(\hat{\boldsymbol{\theta}}) (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}) \,, \tag{1.39}$$

where $H(\hat{\boldsymbol{\theta}})$ is the Hessian of the log posterior (matrix of the second derivatives of (1.37)), evaluated at $\hat{\boldsymbol{\theta}}$,

$$H(\hat{\boldsymbol{\theta}}) = \left. \frac{\partial^2 \ln p(\boldsymbol{\theta} \,|\, \mathbf{y}, m)}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^\top} \right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} = \left. \frac{\partial^2 t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^\top} \right|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}} \,, \tag{1.40}$$

and the linear term has vanished as the gradient of the posterior $\frac{\partial t(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}$ at $\hat{\boldsymbol{\theta}}$ is zero as this is the MAP setting (or a local maximum). Substituting (1.39) into the log marginal likelihood and integrating yields

$$\ln p(\mathbf{y} \,|\, m) = \ln \int d\boldsymbol{\theta}\; p(\boldsymbol{\theta} \,|\, m)\, p(\mathbf{y} \,|\, \boldsymbol{\theta}, m) \tag{1.41}$$

$$= \ln \int d\boldsymbol{\theta}\; \exp\left[t(\boldsymbol{\theta}) \right] \,, \tag{1.42}$$

$$\approx t(\hat{\boldsymbol{\theta}}) + \frac{1}{2} \ln \left| 2\pi H^{-1} \right| \tag{1.43}$$

$$= \ln p(\hat{\boldsymbol{\theta}} \,|\, m) + \ln p(\mathbf{y} \,|\, \hat{\boldsymbol{\theta}}, m) + \frac{d}{2} \ln 2\pi - \frac{1}{2} \ln |H| \,, \tag{1.44}$$

where d is the dimensionality of the parameter space. Equation (1.44) can be written

$$p(\mathbf{y} \,|\, m)_{\text{Laplace}} = p(\hat{\boldsymbol{\theta}} \,|\, m)\, p(\mathbf{y} \,|\, \hat{\boldsymbol{\theta}}, m) \left| 2\pi H^{-1} \right|^{1/2} \,. \tag{1.45}$$

Thus the Laplace approximation to the marginal likelihood consists of a term for the data likelihood at the MAP setting, a penalty term from the prior, and a volume term calculated from the local curvature.

Approximation (1.45) has several shortcomings. The Gaussian assumption is based on the large data limit, and will represent the posterior poorly for small data sets for which, in principle, the advantages of Bayesian integration over ML or MAP are largest. The Gaussian approximation is also poorly suited to bounded, constrained, or positive parameters, such as mixing proportions or precisions, since it assigns non-zero probability mass outside of the parameter domain. Of course, this can often be alleviated by a change of parameter basis (see for example, MacKay, 1998); however there remains the undesirable fact that in the non-asymptotic regime the approximation is still not invariant to reparameterisation. Moreover, the posterior may not be log quadratic for likelihoods with hidden variables, due to problems of identifiability discussed in the next subsection. In these cases the regularity conditions required for convergence do not hold. Even if the exact posterior is unimodal the resulting approximation may well be a poor representation of the nearby probability *mass*, as the approximation is made about a locally maximum probability *density*. The volume term requires the calculation of $|H|$: this takes $\mathcal{O}(nd^2)$ operations to compute the derivatives in the Hessian, and then a further $\mathcal{O}(d^3)$ operations to calculate the determinant; this becomes burdensome for high dimensions, so approximations to this calculation usually ignore off-diagonal elements or assume a block-diagonal structure for the Hessian, which correspond to neglecting dependencies between parameters. Finally, the second derivatives themselves may be intractable to compute.

1.3.3 Identifiability: aliasing and degeneracy

The convergence to Gaussian of the posterior holds only if the model is *identifiable*. Therefore the Laplace approximation may be inaccurate if this is not the case. A model is not identifiable if there is *aliasing* or *degeneracy* in the parameter posterior.

Aliasing arises in models with symmetries, where the assumption that there exists a single mode in the posterior becomes incorrect. As an example of symmetry, take the model containing a discrete hidden variable $\mathbf{x}_i$ with k possible settings (e.g. the indicator variable in a mixture model). Since the variable is hidden these settings can be arbitrarily labelled $k!$ ways. If the likelihood is invariant to these permutations, and if the prior over parameters is also invariant to these permutations, then the landscape for the posterior parameter distribution will be made up of $k!$ identical aliases. For example the posterior for HMMs converges to a mixture of Gaussians, not a single mode, corresponding to the possible permutations of the hidden states. If the aliases are sufficiently distinct, corresponding to well defined peaks in the posterior as a result of large amounts of data, the error in the Laplace method can be corrected by multiplying the marginal likelihood by a factor of $k!$. In practice it is difficult to ascertain the degree of separation of the aliases, and so a simple modification of this sort is not possible. Although corrections have been devised to account for this problem, for example estimating the *permanent* of the model, they are complicated and computationally burdensome. The interested reader is referred

to Barvinok (1999) for a description of a polynomial randomised approximation scheme for estimating permanents, and to Jerrum et al. (2001) for a review of permanent calculations.

Parameter degeneracy arises when there is some redundancy in the choice of parameterisation for the model. For example, consider a model that has two parameters $\boldsymbol{\theta} = (\boldsymbol{\nu}_1, \boldsymbol{\nu}_2)$, whose difference specifies the noise precision of an observed Gaussian variable $\mathbf{y}_i$ with mean $\mathbf{0}$, say, $\mathbf{y}_i \sim \mathrm{N}(\mathbf{y}_i \,|\, \mathbf{0},\, \boldsymbol{\nu}_1 - \boldsymbol{\nu}_2)$. If the prior over parameters does not disambiguate $\boldsymbol{\nu}_1$ from $\boldsymbol{\nu}_2$, the posterior over $\boldsymbol{\theta}$ will contain an infinity of distinct configurations of $(\boldsymbol{\nu}_1, \boldsymbol{\nu}_2)$, all of which give the same likelihood to the data; this degeneracy causes the volume element $\propto \left|H^{-1}\right|$ to be infinite and renders the marginal likelihood estimate (1.45) useless. Parameter degeneracy can be thought of as a continuous form of aliasing in parameter space, in which there are infinitely many aliases.

1.3.4 BIC and MDL

The Bayesian Information Criterion (BIC) (Schwarz, 1978) can be obtained from the Laplace approximation by retaining only those terms that grow with n. From (1.45), we have

$$\ln p(\mathbf{y} \,|\, m)_{\text{Laplace}} = \underbrace{\ln p(\hat{\boldsymbol{\theta}} \,|\, m)}_{\mathcal{O}(1)} + \underbrace{\ln p(\mathbf{y} \,|\, \hat{\boldsymbol{\theta}}, m)}_{\mathcal{O}(n)} + \underbrace{\frac{d}{2} \ln 2\pi}_{\mathcal{O}(1)} - \underbrace{\frac{1}{2} \ln |H|}_{\mathcal{O}(d \ln n)} , \tag{1.46}$$

where each term's dependence on n has been annotated. Retaining $\mathcal{O}(n)$ and $\mathcal{O}(\ln n)$ terms yields

$$\ln p(\mathbf{y} \,|\, m)_{\text{Laplace}} = \ln p(\mathbf{y} \,|\, \hat{\boldsymbol{\theta}}, m) - \frac{1}{2} \ln |H| + \mathcal{O}(1) \,. \tag{1.47}$$

Using the fact that the entries of the Hessian scale linearly with n (see (1.37) and (1.40)), we can write

$$\lim_{n \to \infty} \frac{1}{2} \ln |H| = \frac{1}{2} \ln |n H_0| = \frac{d}{2} \ln n + \underbrace{\frac{1}{2} \ln |H_0|}_{\mathcal{O}(1)} , \tag{1.48}$$

and then assuming that the prior is non-zero at $\hat{\boldsymbol{\theta}}$, in the limit of large n equation (1.47) becomes the BIC score:

$$\ln p(\mathbf{y} \,|\, m)_{\text{BIC}} = \ln p(\mathbf{y} \,|\, \hat{\boldsymbol{\theta}}, m) - \frac{d}{2} \ln n \,. \tag{1.49}$$

The BIC approximation is interesting for two reasons: first, it does not depend on the prior $p(\boldsymbol{\theta} \,|\, m)$; second, it does not take into account the local geometry of the parameter space and hence is invariant to reparameterisations of the model. A Bayesian would obviously baulk at the first of these features, but the second feature of reparameterisation invariance is appealing because this should fall out of an exact Bayesian treatment in any case. In practice the dimension of the model d that is used is equal to the number of *well-determined* parameters, or the number

of *effective* parameters, after any potential parameter degeneracies have been removed. In the example mentioned above the reparameterisation $\boldsymbol{\nu}^* = \boldsymbol{\nu}_1 - \boldsymbol{\nu}_2$ is sufficient, yielding $d = |\boldsymbol{\nu}|$. The BIC is in fact exactly minus the minimum description length (MDL) penalty used in Rissanen (1987). However, the minimum message length (MML) framework of Wallace and Freeman (1987) is closer in spirit to Bayesian integration over parameters. We will be revisiting the BIC in the following chapters as a comparison to our variational Bayesian method for approximating the marginal likelihood.

1.3.5 Cheeseman & Stutz's method

If the *complete-data* marginal likelihood defined as

$$p(\mathbf{x}, \mathbf{y} \,|\, m) = \int d\boldsymbol{\theta}\; p(\boldsymbol{\theta} \,|\, m) \prod_{i=1}^{n} p(\mathbf{x}_i, \mathbf{y}_i \,|\, \boldsymbol{\theta}, m) \tag{1.50}$$

can be computed efficiently then the method proposed in Cheeseman and Stutz (1996) can be used to approximate the marginal likelihood of incomplete data. For any completion of the data $\hat{\mathbf{x}}$, the following identity holds

$$p(\mathbf{y} \,|\, m) = p(\hat{\mathbf{x}}, \mathbf{y} \,|\, m) \frac{p(\mathbf{y} \,|\, m)}{p(\hat{\mathbf{x}}, \mathbf{y} \,|\, m)} \tag{1.51}$$

$$= p(\hat{\mathbf{x}}, \mathbf{y} \,|\, m) \frac{\int d\boldsymbol{\theta}\; p(\boldsymbol{\theta} \,|\, m) p(\mathbf{y} \,|\, \boldsymbol{\theta}, m)}{\int d\boldsymbol{\theta}'\; p(\boldsymbol{\theta}' \,|\, m) p(\hat{\mathbf{x}}, \mathbf{y} \,|\, \boldsymbol{\theta}', m)} \,. \tag{1.52}$$

If we now apply Laplace approximations (1.45) to both numerator and denominator we obtain

$$p(\mathbf{y} \,|\, m) \approx p(\hat{\mathbf{x}}, \mathbf{y} \,|\, m) \frac{p(\hat{\boldsymbol{\theta}} \,|\, m) p(\mathbf{y} \,|\, \hat{\boldsymbol{\theta}}, m) \left|2\pi H^{-1}\right|^{1/2}}{p(\hat{\boldsymbol{\theta}}' \,|\, m) p(\hat{\mathbf{x}}, \mathbf{y} \,|\, \hat{\boldsymbol{\theta}}', m) \left|2\pi H'^{-1}\right|^{1/2}} \,. \tag{1.53}$$

If the approximations are made about the same point $\hat{\boldsymbol{\theta}}' = \hat{\boldsymbol{\theta}}$, then the hope is that errors in each Laplace approximation will tend to cancel one another out. If the completion $\hat{\mathbf{x}}$ is set to be the expected sufficient statistics calculated from an E step of the EM algorithm (discussed in more detail in chapter 2), then the ML/MAP setting $\hat{\boldsymbol{\theta}}'$ will be at the same point as $\hat{\boldsymbol{\theta}}$. The final part of the Cheeseman-Stutz approximation is to form the BIC asymptotic limit of each of the Laplace approximations (1.49). In the original *Autoclass* application (Cheeseman and Stutz, 1996) the dimensionalities of the parameter spaces for the incomplete and complete-data integrals were assumed equal so the terms scaling as $\ln n$ cancel. Since $\hat{\boldsymbol{\theta}}' = \hat{\boldsymbol{\theta}}$, the terms relating to the prior probability of $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\theta}}'$ also cancel (although these are $\mathcal{O}(1)$ in any case), and we obtain:

$$p(\mathbf{y} \,|\, m)_{\mathrm{CS}} = p(\hat{\mathbf{x}}, \mathbf{y} \,|\, m) \frac{p(\mathbf{y} \,|\, \hat{\boldsymbol{\theta}}, m)}{p(\hat{\mathbf{x}}, \mathbf{y} \,|\, \hat{\boldsymbol{\theta}}, m)} \,. \tag{1.54}$$

where $\hat{\boldsymbol{\theta}}$ is the MAP estimate. In chapter 2 we see how the Cheeseman-Stutz approximation is related to the variational Bayesian lower bound. In chapter 6 we compare its performance empirically to variational Bayesian methods on a hard problem, and discuss the situation in which the dimensionalities of the complete and incomplete-data parameters are different.

1.3.6 Monte Carlo methods

Unfortunately the large data limit approximations discussed in the previous section are limited in their ability to trade-off computation time to improve their accuracy. For example, even if the Hessian determinant were calculated exactly (costing $\mathcal{O}(nd^2)$ operations to find the Hessian and then $\mathcal{O}(d^3)$ to find its determinant), the Laplace approximation may still be very inaccurate. Numerical integration methods hold the answer to more accurate, but computationally intensive solutions.

The Monte Carlo integration method estimates the expectation of a function $\phi(\mathbf{x})$ under a probability distribution $f(\mathbf{x})$, by taking samples $\{\mathbf{x}^{(i)}\}_{i=1}^{N} \;:\; \mathbf{x}^{(i)} \sim f(\mathbf{x})$. An unbiased estimate, $\hat{\Phi}$, of the expectation of $\phi(\mathbf{x})$ under $f(\mathbf{x})$, using N samples is given by:

$$\Phi = \int d\mathbf{x}\; f(\mathbf{x})\phi(\mathbf{x}) \simeq \hat{\Phi} = \frac{1}{N}\sum_{i=1}^{N} \phi(\mathbf{x}^{(i)}) \;. \tag{1.55}$$

Expectations such as the predictive density, the marginal likelihood, posterior distributions over hidden variables etc. can be obtained using such estimates. Most importantly, the Monte Carlo method returns more accurate and reliable estimates the more samples are taken, and scales well with the dimensionality of $\mathbf{x}$.

In situations where $f(x)$ is hard to sample from, one can use samples from a different auxiliary distribution $g(x)$ and then correct for this by weighting the samples accordingly. This method is called *importance sampling* and it constructs the following estimator using N samples, $\{\mathbf{x}^{(i)}\}_{i=1}^{N}$, generated such that each $\mathbf{x}^{(i)} \sim g(\mathbf{x})$:

$$\Phi = \int d\mathbf{x}\; g(\mathbf{x})\frac{f(\mathbf{x})}{g(\mathbf{x})}\phi(\mathbf{x}) \simeq \hat{\Phi} = \frac{1}{N}\sum_{i=1}^{N} w^{(i)}\phi(\mathbf{x}^{(i)}) \;, \tag{1.56}$$

$$\text{where} \quad w^{(i)} = \frac{f(\mathbf{x}^{(i)})}{g(\mathbf{x}^{(i)})} \tag{1.57}$$

are known as the *importance weights*. Note that the estimator in (1.56) is unbiased just as that in (1.55). It is also possible to estimate Φ even if $p(\mathbf{x})$ and $g(\mathbf{x})$ can be computed only up to

multiplicative constant factors, that is to say: $f(\mathbf{x}) = f^*(\mathbf{x})/\mathcal{Z}_f$ and $g(\mathbf{x}) = g^*(\mathbf{x})/\mathcal{Z}_g$. In such cases it is straightforward to show that an estimator for Φ is given by:

$$\Phi = \int d\mathbf{x}\; g(\mathbf{x}) \frac{f(\mathbf{x})}{g(\mathbf{x})} \phi(\mathbf{x}) \simeq \hat{\Phi} = \frac{\sum_{i=1}^{N} w^{(i)} \phi(\mathbf{x}^{(i)})}{\sum_{i=1}^{N} w^{(i)}} \;, \tag{1.58}$$

$$\text{where} \quad w^{(i)} = \frac{f^*(\mathbf{x}^{(i)})}{g^*(\mathbf{x}^{(i)})} \tag{1.59}$$

are a slightly different set of importance weights. Unfortunately this estimate is now biased as it is really the ratio of two estimates, and the ratio of two unbiased estimates is in general not an unbiased estimate of the ratio. Although importance sampling is simple, $\hat{\Phi}$ can often have very high variance. Indeed, even in some simple models it can be shown that the variance of the weights $w^{(i)}$, and therefore of $\hat{\Phi}$ also, are unbounded. These and related problems are discussed in section 4.7 of chapter 4 where importance sampling is used to estimate the exact marginal likelihood of a mixture of factor analysers model trained with variational Bayesian EM. We use this analysis to provide an assessment of the tightness of the variational lower bound, which indicates how much we are conceding when using such an approximation (see section 4.7.2).

A method related to importance sampling is *rejection sampling*. It avoids the use of a set of weights $\{w^{(i)}\}_{i=1}^{N}$ by stochastically deciding whether or not to include each sample from $g(\mathbf{x})$. The procedure requires the existence of a constant c such that $c\,g(\mathbf{x}) > f(\mathbf{x})$ for all $\mathbf{x}$, that is to say $c\,g(\mathbf{x})$ envelopes the probability density $f(\mathbf{x})$. Samples are obtained from $f(\mathbf{x})$ by drawing samples from $g(\mathbf{x})$, and then accepting or rejecting each stochastically based on the ratio of its densities under $f(\mathbf{x})$ and $g(\mathbf{x})$. That is to say, for each sample an auxiliary variable $u^{(i)} \sim U(0,1)$ is drawn, and the sample under $g(\mathbf{x})$ accepted only if

$$f(\mathbf{x}^{(i)}) > u^{(i)} c\, g(\mathbf{x}^{(i)}) \;. \tag{1.60}$$

Unfortunately, this becomes impractical in high dimensions and with complex functions since it is hard to find a simple choice of $g(\mathbf{x})$ such that c is small enough to allow the rejection rate to remain reasonable across the whole space. Even in simple examples the acceptance rate falls exponentially with the dimensionality of $\mathbf{x}$.

To overcome the limitations of rejection sampling it is possible to adapt the density $c\,g(\mathbf{x})$ so that it envelopes $f(\mathbf{x})$ more tightly, but only in cases where $f(\mathbf{x})$ is log-concave. This method is called *adaptive rejection sampling* (Gilks and Wild, 1992): the envelope function $c\,g(\mathbf{x})$ is piecewise exponential and is updated to more tightly fit the density $f(\mathbf{x})$ after each sample is drawn. The result is that the probability of rejection monotonically decreases with each sample evaluation. However it is only designed for log-concave $f(\mathbf{x})$ and relies on gradient information to construct tangents which upper bound the density $f(\mathbf{x})$. An interesting extension (Gilks, 1992) to this constructs a *lower* bound $b\,l(\mathbf{x})$ as well (where b is a constant) which is updated in a similar fashion using chords between evaluations of $f(\mathbf{x})$. The advantage of also

using a piecewise exponential lower bound is that the method can become very computationally efficient by not having to evaluate densities under $f(\mathbf{x})$ (which we presume is costly) for some samples. To see how this is possible, consider drawing a sample $\mathbf{x}^{(i)}$ which satisfies

$$b\, l(\mathbf{x}^{(i)}) > u^{(i)} c\, g(\mathbf{x}^{(i)}) \,. \tag{1.61}$$

This sample can be automatically accepted *without* evaluation of $f(\mathbf{x}^{(i)})$, since if inequality (1.61) is satisfied then automatically inequality (1.60) is also satisfied. If the sample does not satisfy (1.61), then of course $f(\mathbf{x}^{(i)})$ needs to be computed, but this can then be used to tighten the bound further. Gilks and Wild (1992) report that the number of density evaluations required to sample N points from $f(\mathbf{x})$ increases as $\sqrt[3]{N}$, even for quite non-Gaussian densities. Their example obtains 100 samples from the standard univariate Gaussian with approximately 15 evaluations, and a further 900 samples with only 15 further evaluations. Moreover, in cases where the log density is close to but not log concave, the adaptive rejection sampling algorithm can still be used with Metropolis methods (see below) to correct for this (Gilks et al., 1995).

Markov chain Monte Carlo (MCMC) methods (as reviewed in Neal, 1992) can be used to generate a chain of samples, starting from $\mathbf{x}^{(1)}$, such that the next sample is a non-deterministic function of the previous sample: $\mathbf{x}^{(i)} \overset{\mathcal{P}}{\leftarrow} \mathbf{x}^{(i-1)}$, where we define $\mathcal{P}(\mathbf{x}', \mathbf{x})$ as the probability of transition from $\mathbf{x}'$ to $\mathbf{x}$. If $\mathcal{P}$ has $f(\mathbf{x})$ as its stationary (equilibrium) distribution, i.e. $f(\mathbf{x}) = \int d\mathbf{x}'\ f(\mathbf{x}')\mathcal{P}(\mathbf{x}', \mathbf{x})$, then the set $\{\mathbf{x}^{(i)}\}_{i=1}^{N}$ can be used to obtain an unbiased estimate of Φ as in (1.55) in the limit of a large number of samples. The set of samples have to drawn from the equilibrium distribution, so it is advisable to discard all samples visited at the beginning of the chain. In general $\mathcal{P}$ is implemented using a proposal density $\mathbf{x}^{(i)} \sim g(\mathbf{x}, \mathbf{x}^{(i-1)})$ about the previous sample. In order to ensure *reversibility* of the Markov chain, the probability of accepting the proposal needs to take into account the probability of a reverse transition. This gives rise to the the Metropolis-Hastings (Metropolis et al., 1953; Hastings, 1970) acceptance function $a(\cdot, \cdot)$:

$$a(\mathbf{x}^{(i)}, \mathbf{x}^{(i-1)}) = \frac{f^*(\mathbf{x}^{(i)}) g(\mathbf{x}^{(i-1)}, \mathbf{x}^{(i)})}{f^*(\mathbf{x}^{(i-1)}) g(\mathbf{x}^{(i)}, \mathbf{x}^{(i-1)})} \,. \tag{1.62}$$

If $a(\mathbf{x}^{(i)}, \mathbf{x}^{(i-1)}) \geq 1$ the sample is accepted, otherwise it is accepted according to the probability $a(\mathbf{x}^{(i)}, \mathbf{x}^{(i-1)})$. Several extensions to the MCMC method have been proposed including over-relaxation (Adler, 1981), hybrid MCMC (Neal, 1993), and reversible-jump MCMC (Green, 1995). These and many others can be found at the MCMC preprint service (Brooks).

Whilst MCMC sampling methods are guaranteed to yield exact estimates in the limit of a large number of samples, even for well-designed procedures the number of samples required for accurate estimates can be infeasibly large. There is a large amount of active research dedicated to constructing measures to ascertain whether the Markov chain has reached equilibrium, whether the samples it generates are independent, and analysing the reliability of the estimates. This

thesis is concerned with fast, reliable, deterministic alternatives to MCMC. Long MCMC runs can then be used to check the accuracy of these deterministic methods.

In contrast to MCMC methods, a new class of sampling methods has been recently devised in which samples from exactly the equilibrium distribution are generated in a finite number of steps of a Markov chain. These are termed *exact sampling* methods, and make use of trajectory *coupling* and *coalescence* via pseudorandom transitions, and is sometimes referred to as *coupling from the past* (Propp and Wilson, 1996). Variations on exact sampling include interruptible algorithms (Fill, 1998) and continuous state-space versions (Murdoch and Green, 1998). Such methods have been applied to graphical models for machine learning problems in the contexts of mixture modelling (Casella et al., 2000), and noisy-or belief networks (Harvey and Neal, 2000).

Finally, one important role of MCMC methods is to compute partition functions. One such powerful method for computing normalisation constants, such as $\mathcal{Z}_f$ used above, is called *annealed importance sampling* (Neal, 2001). It is based on methods such as thermodynamic integration for estimating the free energy of systems at different temperatures, and work on tempered transitions (Neal, 1996). It estimates the ratio of two normalisation constants $\mathcal{Z}_t$ and $\mathcal{Z}_0$, which we can think of for our purposes as the ratio of marginal likelihoods of two models, by collating the results of a chain of intermediate likelihood ratios of 'close' models,

$$\frac{\mathcal{Z}_t}{\mathcal{Z}_0} = \frac{\mathcal{Z}_1}{\mathcal{Z}_0} \cdots \frac{\mathcal{Z}_{t-2}}{\mathcal{Z}_{t-3}} \frac{\mathcal{Z}_{t-1}}{\mathcal{Z}_{t-2}} \frac{\mathcal{Z}_t}{\mathcal{Z}_{t-1}} \,. \tag{1.63}$$

Each of the ratios is estimated using samples from a Markov chain Monte Carlo method. We will look at this method in much more detail in Chapter 6, where it will be used as a gold standard against which we test the ability of the variational Bayesian EM algorithm to approximate the marginal likelihoods of a large set of models.

To conclude this section we note that Monte Carlo is a purely frequentist procedure and in the words of O'Hagan (1987) is 'fundamentally unsound'. The objections raised therein can be summarised as follows. First, the estimate $\hat{\Phi}$ depends on the sampling density $g(\mathbf{x})$, even though $g(\mathbf{x})$ itself is ancillary to the estimation. Put another way, the same set of samples $\{\mathbf{x}^{(i)}\}_{i=1}^n$, conveying exactly the same information about $p(\mathbf{x})$, but generated under a different $g(\mathbf{x})$ would produce a different estimate $\hat{\Phi}$. Of course, the density $g(\mathbf{x})$ is often tailored to the problem at hand and so we would expect it to contain some of the essence of the estimate. Second, the estimate does not depend on the location of the $\mathbf{x}^{(i)}$s, but only on function evaluations at those points, e.g. $f(\mathbf{x}^{(i)})$. This is surely suboptimal, as the spatial distribution of the function evaluations provides information on the integrand $f(\mathbf{x})\phi(\mathbf{x})$ as a whole. To summarise, classical Monte Carlo bases its estimate on irrelevant information, $g(\mathbf{x})$, and also discards relevant information from the location of the samples. Bayesian variants of Monte Carlo integration procedures have been devised to address these objections using Gaussian process models (O'Hagan, 1991; Rasmussen and Ghahramani, 2003), and there is much future work to do in this direction.

1.4 Summary of the remaining chapters

Chapter 2 Forms the theoretical core of the thesis, and examines the use of variational methods for obtaining lower bounds on the likelihood (for point-parameter learning) and the marginal likelihood (in the case of Bayesian learning). The implications of VB applied to the large family of *conjugate-exponential* graphical models are investigated, for both directed and undirected representations. In particular, a general algorithm for conjugate-exponential models is derived and it is shown that existing propagation algorithms can be employed for inference, with approximately the same complexity as for point-parameters.

In addition, the relations of VB to a number of other commonly used approximations are covered. In particular, it is shown that the Cheeseman-Stutz (CS) score is in fact a looser lower bound on the marginal likelihood than the VB score.

Chapter 3 Applies the results of chapter 2 to hidden Markov models (HMMs). It is shown that it is possible to recover the number of hidden states required to model a synthetic data set, and that the variational Bayesian algorithm can outperform maximum likelihood and maximum a posteriori parameter learning algorithms on real data in terms of generalisation.

Chapter 4 Applies the variational Bayesian method to a mixtures of factor analysers (MFA) problem, where it is shown that the procedure can automatically determine the optimal number of components and the local dimensionality of each component (i.e. the number of factors in each analyser). Through a stochastic procedure for adding components to the model, it is possible to perform the variational optimisation incrementally and avoid local maxima. The algorithm is shown to perform well on a variety of synthetic data sets, and is compared to a BIC-penalised maximum likelihood algorithm on a real-world data set of hand-written digits.

This chapter also investigates the generally applicable method of drawing importance samples from the variational approximation to estimate the marginal likelihood and the KL divergence between the approximate and exact posterior. Specific results applying variants of this procedure to the MFA model are analysed.

Chapter 5 Presents an application of the theorems presented in chapter 2 to linear dynamical systems (LDSs). The result is the derivation of a variational Bayesian input-dependent Rauch-Tung-Striebel smoother, such that it is possible to infer the posterior hidden state trajectory whilst integrating over all model parameters. Experiments on synthetic data show that it is possible to infer the dimensionality of the hidden state space and determine which dimensions of the inputs and the data are relevant. Also presented are preliminary experiments for elucidating gene-gene interactions in a well-studied human immune response mechanism.

Chapter 6 Investigates a novel application of the VB framework to approximating the marginal likelihood of discrete-variable directed acyclic graphs (DAGs) that contain hidden variables. The VB lower bound is compared to MAP, BIC, CS, and annealed importance sampling (AIS), on a simple (yet non-trivial) model selection task of determining which of all possible structures within a class generated a data set.

The chapter also discusses extensions and improvements to the particular form of AIS used, and suggests related approximations which may be of interest.

Chapter 7 Concludes the thesis with a discussion on some topics closely related to the ideas already investigated. These include: Bethe and Kikuchi approximations, infinite models, inferring causality using the marginal likelihood, and automated algorithm derivation. The chapter then concludes with a summary of the main contributions of the thesis.