1 PROBABILISTIC MODELLING

Inference is the process of drawing conclusions from evidence. Much of our lives are spent making inferences about the world given our observations of it; in particular inference is a central aspect of the scientific process. Although deductive logic offers a framework for inferring conclusions from absolute statements of truth, it does not apply to the more typical real-world setting where the information we receive is subject to uncertainty.

The actual science of logic is conversant at present only with things either certain, impossible, or entirely doubtful, none of which (fortunately) we have to reason on. Therefore the true logic for this world is the calculus of probabilities
—James Clerk Maxwell

To make inferences under conditions of uncertainty, we must instead turn to probability theory. Probabilities offer a consistent framework for quantifying the uncertainty in our beliefs about the world and making inferences given these beliefs. The output of the inference process is itself probabilistic, reflecting that the conclusions we make given uncertain information will themselves be subject to uncertainty.

In this chapter we will first introduce the probability notation we will use in the rest of this work, and state some basic results which will be important in the later chapters. We will introduce graphical models as a compact way of visualising structure in probabilistic models. Finally we will give a concrete definition of the inference tasks that the methods presented in the rest of this thesis are aimed at computing (approximate) solutions to, and motivate why such approximate computational methods are needed.

Probability theory is nothing but common sense reduced to calculation.
— Pierre-Simon Laplace

1.1 PROBABILITY THEORY

A *probability space* is defined as a triplet $(S, \mathcal{E}, \mathrm{P})$ where

- S is the *sample space*, the set of all possible outcomes,
- $\mathcal{E}$ is the *event space*, a σ-algebra on S, defining all possible events (measurable subsets of S),
- P is the *probability measure*, a finite measure satisfying $\mathrm{P}(S) = 1$, which specifies the probabilities of events in $\mathcal{E}$.

A σ-algebra, $\mathcal{E}$, on a set S is set of subsets of S with $S \in \mathcal{E}$, $\emptyset \in \mathcal{E}$ and which is closed under complement and countable unions and intersections.

Kolmogorov's axioms:

1. *Non-negativity:* $\mathsf{P}(E) \geq 0 \; \forall E \in \mathcal{E}$,
2. *Normalisation:* $\mathsf{P}(S) = 1$,
3. *Countable additivity: for any countable set of disjoint events* $\{E_i\}_i : E_i \in \mathcal{F} \; \forall i$, $E_i \cap E_j = \emptyset \; \forall i \neq j$, $\mathsf{P}(\cup_i E_i) = \sum_i \mathsf{P}(E_i)$.

Given this definition of a probability space, Kolmogorov's axioms [20] can be used to derive a measure-theoretic formulation of probability theory. The probability of an event $E \in \mathcal{E}$ is defined as the measure of that event $\mathsf{P}(E)$. Two events $A, B \in \mathcal{E}$ are said to be *independent* if $\mathsf{P}(A \cap B) = \mathsf{P}(A)\mathsf{P}(B)$.

A measure-theoretic approach has the advantage of providing a unified treatment for describing probabilities on both finite and infinite sample spaces. Although alternative derivations of the laws of probability from different premises such as Cox's theorem [7, 8] have been proposed, modern extensions of this work result in a calculus of probabilities that is equivalent to Kolmogorov's [33], with the differences mainly being in the philosophical interpretations of probabilities.

1.1.1 Random variables

If $(X, \mathcal{F})$ and $(Y, \mathcal{G})$ are two measurable spaces, a function $f : X \to Y$ is measurable if $f^{-1}(E) \in \mathcal{F} \; \forall E \in \mathcal{G}$.

When modelling real-world processes, rather than considering events as subsets of an abstract sample space, it is usually more helpful to consider *random variables* which represent quantities in the model of interest. A random variable $\mathsf{x} : S \to X$ is defined as a measurable function from the sample space to a measurable space $(X, \mathcal{F})$.

The Borel σ-algebra $\mathscr{B}(\mathbb{R})$ is the smallest σ-algebra on $\mathbb{R}$ which contains all open real intervals.

Often X is the reals, $\mathbb{R}$, and $\mathcal{F}$ is the Borel σ-algebra on the reals, $\mathscr{B}(\mathbb{R})$, in which case we will refer to a *real random variable*. It is also common to consider cases where X is a real vector space, $\mathbb{R}^D$, and $\mathcal{F} = \mathscr{B}(\mathbb{R}^D)$ - in this case we will term the resulting random variable a *random vector* and use the notation $\mathbf{x} : S \to X$. A final special case is when X is countable and $\mathcal{F}$ is the power set $\mathscr{P}(X)$ in which case we will refer to x as a *discrete random variable*.

If $(X, \mathcal{F})$ and $(Y, \mathcal{G})$ are two measurable spaces, μ a measure on these spaces and $f : X \to Y$ a measurable function, the pushforward measure μ_f satisfies $\mu_f(A) = \mu \circ f^{-1}(A) \; \forall A \in \mathcal{G}$.

Due to the definition of a random variable as a measurable function, we can define a pushforward measure on a random variable x

$$\mathsf{P}_\mathsf{x}(A) = \mathsf{P} \circ \mathsf{x}^{-1}(A) = \mathsf{P}(\{s \in S : \mathsf{x}(s) \in A\}) \quad \forall A \in \mathcal{F}. \tag{1.1}$$

The measure P_x specifies that the probability of the event that the random variable x takes a value in a measurable set $A \in \mathcal{F}$ is $\mathsf{P}_\mathsf{x}(A)$.

1.1.2 Joint and conditional probability

Often we will jointly define multiple random variables on the same probability space. Let $(S, \mathcal{E}, \mathsf{P})$ be a probability space and $\mathsf{x} : S \to X$,

$\mathsf{y} : S \to Y$ be two random variables with corresponding σ-algebras $\mathcal{F}$ and $\mathcal{G}$. Then the *joint probability* of x and y is defined as

$$\mathsf{P}_{\mathsf{x},\mathsf{y}}(A,\, B) = \mathsf{P}\Big(\mathsf{x}^{-1}(A) \cap \mathsf{y}^{-1}(B)\Big) \quad \forall A \in \mathcal{F},\, B \in \mathcal{G}. \tag{1.2}$$

The joint probability is related to the probabilities P_x and P_y by

$$\mathsf{P}_{\mathsf{x},\mathsf{y}}(A,\, Y) = \mathsf{P}_\mathsf{x}(A),\ \mathsf{P}_{\mathsf{x},\mathsf{y}}(X,\, B) = \mathsf{P}_\mathsf{y}(B) \quad \forall A \in \mathcal{F},\, B \in \mathcal{G}. \tag{1.3}$$

In this context P_x and P_y are referred to as *marginals* of the joint.

The two random variables are said to be independent if and only if

$$\mathsf{P}_{\mathsf{x},\,\mathsf{y}}(A, B) = \mathsf{P}_\mathsf{x}(A)\mathsf{P}_\mathsf{y}(B) \quad \forall A \in \mathcal{F},\, B \in \mathcal{G}. \tag{1.4}$$

Also useful is the definition of *conditional probability*

$$\mathsf{P}(A \mid B) = \frac{\mathsf{P}(A \cap B)}{\mathsf{P}(B)} \quad \forall A \in \mathcal{E},\, B \in \mathcal{E} : \mathsf{P}(B) \neq 0. \tag{1.5}$$

In Kolmogorov's probability theory, (1.5) is given as an additional definition distinct from the basic axioms. In alternatives such as the work of Cox [7, 8] and de Finetti [11], conditional probabilities are instead viewed as a primitive.

Correspondingly, the conditional probabilities of random variables $\mathsf{P}_{\mathsf{x}|\mathsf{y}}$ and $\mathsf{P}_{\mathsf{x}|\mathsf{y}}$ can likewise be defined as satisfying

$$\begin{aligned}\mathsf{P}_{\mathsf{x},\mathsf{y}}(A,\, B) = \mathsf{P}_{\mathsf{x}|\mathsf{y}}(A \mid B)\, \mathsf{P}_\mathsf{y}(B) = \mathsf{P}_{\mathsf{y}|\mathsf{x}}(B \mid A)\, \mathsf{P}_\mathsf{x}(A) \\ \forall A \in \mathcal{F},\, B \in \mathcal{G} : \mathsf{P}_{\mathsf{x},\mathsf{y}}(A,\, B) \neq 0,\end{aligned} \tag{1.6}$$

which is sometimes referred to as the product rule.

An implication of (1.6) is what is often termed *Bayes' theorem*

$$\mathsf{P}_{\mathsf{x}|\mathsf{y}}(A \mid B) = \frac{\mathsf{P}_{\mathsf{y}|\mathsf{x}}(B \mid A)\, \mathsf{P}_\mathsf{x}(A)}{\mathsf{P}_\mathsf{y}(B)} \quad \forall A \in \mathcal{F},\, B \in \mathcal{G} : \mathsf{P}_\mathsf{y}(B) \neq 0, \tag{1.7}$$

which will be of key importance in the later discussion of inference.

The definition in (1.2) of the joint probability of a pair of random variables can be extended to arbitarily large collections of random variables. Similarly conditional probabilities can be defined for collections of multiple jointly dependent random variables, with the product rule given in (1.6) generalising to a combinatorial number of possible factorisations of the joint probability. Graphical models offer a convenient way of representing the dependencies between large collections of random variables and any resulting factorisation structure in their joint probability, and will be discussed later in this chapter in section 1.2 .

1.1.3 Probability densities

So far we have ignored how the probability measure P is defined and by consequence the probability of a random variable.

A measure on X is σ-finite if X is a countable union of finite measure sets.

The Radon–Nikodyn theorem guarantees that for a pair of $\sigma-$finite measures μ and ν on a measurable space $(X, \mathcal{F})$ where ν is absolutely continuous with respect to μ, then there is a unique (up to μ-null sets) measurable function $f : X \to [0, \infty)$ termed a *density* such that

$$\nu(A) = \int_A f \, \mathrm{d}\mu \quad \forall A \in \mathcal{F}. \tag{1.8}$$

If μ and ν are measures on a measurable space $(X, \mathcal{F})$ then ν has absolute continuity wrt to μ if $\forall A \in \mathcal{F}$, $\mu(A) = 0 \Rightarrow \nu(A) = 0$.

The density function f is also termed the *Radon-Nikodym derivative* of ν with respect to μ, denoted $\frac{\mathrm{d}\nu}{\mathrm{d}\mu}$. Density functions therefore represent a convenient way to define a probability measure with respect to an appropriate base measure. It can also be shown that if $f = \frac{\mathrm{d}\nu}{\mathrm{d}\mu}$ and g is a measurable function that

$$\int_X g \, \mathrm{d}\nu = \int_X g \, f \, \mathrm{d}\mu, \tag{1.9}$$

which we will use later when discussing calculation of expectations.

For real random variables, an appropriate base measure is usually the *Lebesgue measure*, λ, on $\mathbb{R}$. The probability P_x of a real random variable x can then be defined via a *probability density* $\mathsf{p}_\mathsf{x} : \mathbb{R} \to [0, \infty)$ by

$$\mathsf{P}_\mathsf{x}(A) = \int_A \mathsf{p}_\mathsf{x} \, \mathrm{d}\lambda = \int_A \mathsf{p}_\mathsf{x}(x) \, \mathrm{d}x \qquad \forall A \in \mathscr{B}(\mathbb{R}). \tag{1.10}$$

Analagously for a random vector $\mathbf{x}$ with density $\mathsf{p}_\mathbf{x} : \mathbb{R}^D \to [0, \infty)$ with respect to the D-dimensional Lebesgue measure λ^D

$$\mathsf{P}_\mathbf{x}(A) = \int_A \mathsf{p}_\mathbf{x} \, \mathrm{d}\lambda^D = \int_A \mathsf{p}_\mathbf{x}(\boldsymbol{x}) \, \mathrm{d}\boldsymbol{x} \qquad \forall A \in \mathscr{B}(\mathbb{R}^D). \tag{1.11}$$

The notation in the second equalities in (1.10) and (1.11) uses a convention that will be used throughout this thesis that integrals without an explicit measure are with respect to the Lebesgue measure.

The counting measure $\#$ is defined as $\#(A) = |A|$ for all finite A and $\#(A) = +\infty$ otherwise.

For discrete random variables, an appropriate base measure is instead the *counting measure*, $\#$. The probability of a discrete random variable

is then defined via a probability density $\mathsf{p}_\mathsf{x} : X \to [0, 1]$ by

$$\mathsf{P}_\mathsf{x}(A) = \int_A \mathsf{p}_\mathsf{x} \, \mathrm{d}\# = \sum_{x \in A} \mathsf{p}_\mathsf{x}(x) \qquad \forall A \in \mathscr{P}(X). \tag{1.12}$$

The co-domain of a probability density p_x for a discrete random variable is restricted to $[0, 1]$ due to the non-negativity and normalisation requirements for the probability measure P_x, with $\sum_{x \in X} \mathsf{p}_\mathsf{x}(x) = 1$. Commonly for the case of a discrete random variable, the density p_x is instead referred to as a *probability mass function*, with density reserved for real random variables. We will however use *probability density* in both cases in keeping with the earlier definition of a density with respect to a base measure, this avoiding difficulties when defininining joint probabilities on a mixture of real and discrete random variables.

The joint probability $\mathsf{P}_{\mathsf{x},\mathsf{y}}$ of a pair of random variables x and y with co-domains the measurable spaces $(X, \mathcal{F})$ and $(Y, \mathcal{G})$ respectively, can be defined via a joint probability density $\mathsf{p}_{\mathsf{x},\mathsf{y}} : X \times Y \to [0, \infty)$ by

$$\mathsf{P}_{\mathsf{x},\mathsf{y}}(A, B) = \int_{A \times B} \mathsf{p}_{\mathsf{x},\mathsf{y}} \, \mathrm{d}(\mu_\mathsf{x} \times \mu_\mathsf{y}) \quad \forall A \in \mathcal{F}, \, B \in \mathcal{G}, \tag{1.13}$$

where $\mu_\mathsf{x} \times \mu_\mathsf{y}$ represents the product measure of two appropriate base measures μ_x and μ_y, e.g. $\mu_\mathsf{x} = \lambda$ and $\mu_\mathsf{y} = \#$ if x is a real random variable and y is a discrete random variable.

If $(X_1, \mathcal{F}_1, \mu_1)$ and $(X_2, \mathcal{F}_2, \mu_2)$ are two measure spaces, the product measure $\mu_1 \times \mu_2$ on a measurable space $(X_1 \times X_2, \mathcal{F}_1 \otimes \mathcal{F}_2)$ is defined as satisfying $(\mu_1 \times \mu_2)(A_1 \times A_2) = \mu_1(A_1)\mu_2(A_2)$ $\forall A_1 \in \mathcal{F}_1, \, A_2 \in \mathcal{F}_2$.

When dealing with random variables, we will often only specify the co-domain of the random variable(s) and a (joint) probability density, with the base measure being implicitly defined as the Lebesgue measure for real random variables (or vectors), counting measure for discrete random variables and an appropriate product measure for a mix of random variables. Similarly we will usually neglect to explicitly define the probability space $(S, \mathcal{E}, \mathsf{P})$ which the random variable(s) map from. In this case we will typically use the loose notation $\mathsf{x} \in X$ to mean a random variable x with co-domain X.

This less explicit but more succinct probability notation in terms of random variables and densities is common in the machine learning and computational statitistics literature and will generally be preferred to improve readability. Tables 1.1, 1.2 and 1.3 give definitions of the densities and shorthand notation for some common parametric probability distributions that we will use in this thesis.

Name	Parameters	Shorthand	Density	Support
Bernoulli	$\pi \in [0,1]$	$\mathrm{Ber}(x \mid \pi)$	$\pi^x(1-\pi)^{(1-x)}$	$x \in \{0,1\}$
Categorical	$\boldsymbol{\pi} \in \mathbb{S}^K$	$\mathrm{Cat}(x \mid \boldsymbol{\pi})$	$\sum_{k=1}^{K}(\mathbb{1}_{\{k\}}(x)\pi_k)$	$x \in \{1 \dots K\}$

Table 1.1: Definitions of densities of parameteric distributions for discrete random variables that will be used in this thesis.

Name	Parameters	Shorthand	Density
Normal	$\mu \in \mathbb{R}$: mean $\sigma > 0$: standard deviation	$\mathcal{N}\left(x \mid \mu, \sigma^2\right)$	$\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$
Multivariate normal	$\boldsymbol{\mu} \in \mathbb{R}^D$: mean vector $\boldsymbol{\Sigma} \in \mathcal{S}_{++}^D$: covariance matrix	$\mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})$	$\frac{1}{\sqrt{(2\pi)^D\lvert\boldsymbol{\Sigma}\rvert}}\exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\mathsf{T}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$
Student's t	$\nu > 0$: degrees of freedom $\mu \in \mathbb{R}$: location $\sigma > 0$: scale	$\mathrm{StT}(x \mid \nu, \mu, \sigma)$	$\frac{\Gamma\left(\frac{\nu+1}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)\sqrt{\pi\nu}\sigma}\left(1+\frac{1}{\nu}\left(\frac{x-\mu}{\sigma}\right)^2\right)^{-\frac{\nu+1}{2}}$
Logistic	$\mu \in \mathbb{R}$: location $\sigma > 0$: scale	$\mathrm{Logistic}(x \mid \mu, \sigma)$	$\frac{1}{4\sigma}\cosh\left(\frac{x-\mu}{2\sigma}\right)^{-2}$
Inverse cosh	$\mu \in \mathbb{R}$: location $\sigma > 0$: scale	$\mathrm{InvCosh}(x \mid \mu, \sigma)$	$\frac{1}{2\sigma}\cosh\left(\frac{\pi(x-\mu)}{2\sigma}\right)^{-1}$

Table 1.2: Definitions of densities of parameteric distributions for unbounded real random variables that will be used in this thesis.

Name	Parameters	Shorthand	Density	Support
Log-normal	$\mu \in \mathbb{R}$: log mean $\sigma > 0$: log standard deviation	$\mathrm{LogNorm}(x \mid \mu, \sigma^2)$	$\frac{1}{x\sqrt{2\pi}\sigma} \exp\left(-\frac{(\log x - \mu)^2}{2\sigma^2}\right)$	$x > 0$
Multivariate log-normal	$\boldsymbol{\mu} \in \mathbb{R}^D$: log mean $\boldsymbol{\Sigma} \in \mathcal{S}_{++}^D$: log covariance	$\mathrm{LogNorm}(\boldsymbol{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})$	$\frac{\exp\left(-\frac{1}{2}(\log \boldsymbol{x} - \boldsymbol{\mu})^\intercal \boldsymbol{\Sigma}^{-1} (\log \boldsymbol{x} - \boldsymbol{\mu})\right)}{\prod_{d=1}^D (x_d)\sqrt{(2\pi)^D \lvert\boldsymbol{\Sigma}\rvert}}$	$\boldsymbol{x} \in [0, \infty)^D$
Exponential	$\lambda > 0$: rate	$\mathrm{Exp}(x \mid \lambda)$	$\lambda \exp(-\lambda x)$	$x \geq 0$
Uniform	$a \in \mathbb{R}$: minimum $b \in \mathbb{R}$: maximum, $b > a$	$\mathcal{U}(x \mid a, b)$	$\frac{1}{b-a} \mathbb{1}_{[a,b]}(x)$	$a \leq x \leq b$
Half-Cauchy	$\gamma > 0$: scale	$C_{\geq 0}(x \mid \gamma)$	$\frac{2}{\pi\gamma}\left(1 + \frac{x^2}{\gamma^2}\right)^{-1}$	$x \geq 0$
Gamma	$\alpha > 0$: shape $\beta > 0$: rate	$\mathrm{Gamma}(x \mid \alpha, \beta)$	$\frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} \exp(-\beta x)$	$x \geq 0$
Beta	$\alpha > 0$: shape $\beta > 0$: shape	$\mathrm{Beta}(x \mid \alpha, \beta)$	$\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}$	$0 \leq x \leq 1$
Dirichlet	$\boldsymbol{\alpha} \in (0, \infty)^K$: concentration	$\mathrm{Dir}(\boldsymbol{x} \mid \boldsymbol{\alpha})$	$\frac{\Gamma\left(\sum_{k=1}^K \alpha_k\right)}{\prod_{k=1}^K \Gamma(\alpha_k)} \prod_{k=1}^K x_i^{\alpha_k - 1}$	$\boldsymbol{x} \in \mathbb{S}^K$
Lomax	$\alpha > 0$: shape $\beta > 0$: scale	$\mathrm{Lomax}(x \mid \alpha, \beta)$	$\frac{\alpha\beta^\alpha}{(\beta+x)^{\alpha+1}}$	$x \geq 0$

Table 1.3: Definitions of densities of parameteric distributions for bounded real random variables that will be used in this thesis.

1.1.4 Transforms of random variables

It is common to define a random variable via a transform of another. Let x be a random variable with co-domain the measurable space $(X, \mathcal{F})$. Further let $(Y, \mathcal{G})$ be a second measurable space and $\phi : X \to Y$ a measurable function between the two spaces. If we define $\mathsf{y} = \phi \circ \mathsf{x}$ then analagously to our original definition of P_x as the pushforward measure of P under the measurable function defining x, we can define P_y in terms of P_x as

$$\mathsf{P}_\mathsf{y}(A) = \mathsf{P}_\mathsf{x} \circ \phi^{-1}(A) = \mathsf{P}_\mathsf{x}(\{x \in X : \phi(x) \in A\}) \quad \forall A \in \mathcal{G}, \tag{1.14}$$

i.e. the probability of the event $\mathsf{y} \in A$ is equal to the probability of x being in the pre-image under ϕ of A. To calculate probabilities of transformed random variables therefore we will therefore need to be able to find the pre-images of values of the transformed variable.

If the probability P_x is defined by a probability density p_x with respect to a measure μ_x, we can also in some cases find a density p_y on the transformed variable $\mathsf{y} = \phi(\mathsf{x})$ with respect to a (potentially different) measure μ_y which can be used to calculate the probability P_y,

$$\mathsf{P}_\mathsf{y}(A) = \int_{\phi^{-1}(A)} \mathsf{p}_\mathsf{x} \, \mathrm{d}\mu_\mathsf{x} = \int_A \mathsf{p}_\mathsf{y} \, \mathrm{d}\mu_\mathsf{y} \quad \forall A \in \mathcal{G}. \tag{1.15}$$

For random variables with countable co-domains where the integral in (1.15) corresponds to a sum, a p_y satisfying (1.15) is simple to identify. If x is a discrete random variable with probability density p_x with respect to the counting measure, then $\mathsf{y} = \phi(\mathsf{x})$ will necessarily also be a discrete random variable. Applying (1.15) for $\mathsf{p}_\mathsf{x} = \frac{\mathrm{d}\mathsf{P}_\mathsf{x}}{\mathrm{d}\#}$ we have that

$$\begin{aligned}\int_{\phi^{-1}(A)} \mathsf{p}_\mathsf{x}(x) \, \mathrm{d}\#(x) &= \sum_{x \in \phi^{-1}(A)} \mathsf{p}_\mathsf{x}(x) = \sum_{y \in A} \sum_{x \in \phi^{-1}(y)} \mathsf{p}_\mathsf{x}(x) \\ &= \int_A \sum_{x \in \phi^{-1}(y)} \mathsf{p}_\mathsf{x}(x) \, \mathrm{d}\#(y) \quad \forall A \in \mathcal{G}. \end{aligned} \tag{1.16}$$

We can therefore define $\mathsf{p}_\mathsf{y} = \frac{\mathrm{d}\mathsf{P}_\mathsf{y}}{\mathrm{d}\#}$ in terms of p_x as

$$\mathsf{p}_\mathsf{y}(y) = \sum_{x \in \phi^{-1}(y)} \mathsf{p}_\mathsf{x}(x) \quad \forall y \in Y. \tag{1.17}$$

In the special case that ϕ is bijective we have that

$$\mathrm{p}_{\mathsf{y}}(y) = \mathrm{p}_{\mathsf{x}} \circ \phi^{-1}(y) \quad \forall y \in Y. \tag{1.18}$$

For transformations of real random variables and vectors, the situation is more complicated as we need to account for any local contraction or expansion of space by the map ϕ. Let $X = \mathbb{R}^M$ and $Y = \mathbb{R}^N$ with $N \leq M$, $N, M \in \mathbb{N}$. We will need a result from geometric measure theory, the *co-area formula* [10]. Let g be an L^1 integrable function and $\boldsymbol{\phi} : X \to Y$ a Lipschitz map. Then the co-area formula states that

$$\int_X g(\boldsymbol{x})\, J_{\boldsymbol{\phi}}(\boldsymbol{x})\, \mathrm{d}\lambda^M(\boldsymbol{x}) = \int_Y \int_{\boldsymbol{\phi}^{-1}(\boldsymbol{y})} g(\boldsymbol{x})\, \mathrm{d}\mathcal{H}^{M-N}(\boldsymbol{x})\, \mathrm{d}\lambda^N(\boldsymbol{y}) \tag{1.19}$$

where $\mathcal{H}^D$ is the D-dimensional *Hausdorff measure* and $J_{\boldsymbol{\phi}} : X \to [0, \infty)$ is the *Jacobian determinant* defined as

$$J_{\boldsymbol{\phi}}(\boldsymbol{x}) = \left| \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{x}} \frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{x}}^{\mathsf{T}} \right|^{\frac{1}{2}} \quad \forall \boldsymbol{x} \in X. \tag{1.20}$$

The D-dimensional Hausdorff measure $\mathcal{H}^D$ on $\mathbb{R}^N$ for $D \in \mathbb{N}$, $0 < D < N$ formalises a measure of the 'volume' of D-dimensional submanifolds of $\mathbb{R}^N$ - e.g. for $D = 1$ it corresponds to the length of a curve in $\mathbb{R}^N$. Additionally $\mathcal{H}^N = \lambda^N$ and $\mathcal{H}^0 = \#$.

Now let $\mathbf{x}$ be a random vector with co-domain the measurable space $(X, \mathscr{B}(\mathbb{R}^M))$ and define $\mathbf{y} = \boldsymbol{\phi} \circ \mathbf{x}$ as a random vector with co-domain the measurable space $(Y, \mathscr{B}(\mathbb{R}^N))$ with $\boldsymbol{\phi} : X \to Y$ a Lipschitz map as above. Let $Z = \left\{ \boldsymbol{x} \in X : J_{\boldsymbol{\phi}}(\boldsymbol{x}) = 0 \right\}$ and require that $\mathrm{P}_{\mathsf{x}}(Z) = 0$. Then for $A \in \mathscr{B}(\mathbb{R}^N)$ define an L^1 integrable function g as

$$g(\boldsymbol{x}) = \begin{cases} \mathbb{1}_A \circ \boldsymbol{\phi}(\boldsymbol{x})\, \mathrm{p}_{\mathbf{x}}(\boldsymbol{x})\, J_{\boldsymbol{\phi}}(\boldsymbol{x})^{-1} & \forall \boldsymbol{x} \in X \setminus Z \\ 0 & \forall \boldsymbol{x} \in Z \end{cases}. \tag{1.21}$$

Integrating $g(\boldsymbol{x})\, J_{\boldsymbol{\phi}}(\boldsymbol{x})$ over $\boldsymbol{x} \in X$ we have that

$$\begin{aligned} \int_X g(\boldsymbol{x})\, J_{\boldsymbol{\phi}}(\boldsymbol{x})\, \mathrm{d}\lambda^M(\boldsymbol{x}) &= \int_{X \setminus Z} \mathbb{1}_A \circ \boldsymbol{\phi}(\boldsymbol{x})\, \mathrm{p}_{\mathbf{x}}(\boldsymbol{x})\, \mathrm{d}\lambda^M(\boldsymbol{x}) && (1.22) \\ &= \int_X \mathbb{1}_A \circ \boldsymbol{\phi}(\boldsymbol{x})\, \mathrm{dP}_{\mathsf{x}}(\boldsymbol{x}) && (1.23) \\ &= \int_{\phi^{-1}(A)} \mathrm{dP}_{\mathsf{x}}(\boldsymbol{x}) = \mathrm{P}_{\mathsf{y}}(A). && (1.24) \end{aligned}$$

The equality between first and second lines comes from the requirement $\mathrm{P}_{\mathsf{x}}(Z) = 0$, with the Lebesgue integrals of a function over two

sets which differ by only a zero-measure set equal. Now applying the co-area formula (1.19) to the left-hand side gives

$$\int_Y \int_{\boldsymbol{\phi}^{-1}(\boldsymbol{y})} g(\boldsymbol{x})\, \mathrm{d}\mathcal{H}^{M-N}(\boldsymbol{x})\, \mathrm{d}\lambda^N(\boldsymbol{y}) = \mathrm{P}_{\mathsf{y}}(A). \tag{1.25}$$

Therefore we can define a density $\mathrm{p}_{\mathsf{y}} = \frac{\mathrm{dP}_{\mathsf{y}}}{\mathrm{d}\lambda^N}$ satisfying (1.15) as

$$\mathrm{p}_{\mathsf{y}}(\boldsymbol{y}) = \int_{\boldsymbol{\phi}^{-1}(\boldsymbol{y})} \mathrm{p}_{\mathsf{x}}(\boldsymbol{x})\, J_{\boldsymbol{\phi}}(\boldsymbol{x})^{-1}\, \mathrm{d}\mathcal{H}^{M-N}(\boldsymbol{x}) \quad \forall \boldsymbol{y} \notin \boldsymbol{\phi}(Z). \tag{1.26}$$

For the special case of a dimension-preserving map $\boldsymbol{\phi}$ with $N = M$ the integral in (1.26) is with respect to $\mathcal{H}^0$ which is equivalent to the counting measure #. In this case $J_{\boldsymbol{\phi}}(\boldsymbol{x}) = \left|\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{x}}\right|$ and we therefore get

$$\mathrm{p}_{\mathsf{y}}(\boldsymbol{y}) = \sum_{\boldsymbol{x} \in \boldsymbol{\phi}^{-1}(\boldsymbol{y})} \mathrm{p}_{\mathsf{x}}(\boldsymbol{x}) \left|\frac{\partial \boldsymbol{\phi}}{\partial \boldsymbol{x}}\right|^{-1} \quad \forall \boldsymbol{y} \notin \boldsymbol{\phi}(Z). \tag{1.27}$$

Under the further restriction that $\boldsymbol{\phi}$ is bi-Lipschitz, i.e. it is bijective and Lipschitz in both directions, we recover the more commonly presented multidimensional change of variables formula

$$\mathrm{p}_{\mathsf{y}}(\boldsymbol{y}) = \mathrm{p}_{\mathsf{x}} \circ \boldsymbol{\phi}^{-1}(\boldsymbol{y}) \left|\frac{\partial \boldsymbol{\phi}^{-1}}{\partial \boldsymbol{y}}\right| \quad \forall \boldsymbol{y} \in Y. \tag{1.28}$$

In both of the cases considered, we have seen that if the function ϕ the random variable x is mapped through is bijective, the resulting expression for the density on the mapped random variable y is simpler in the sense that the pre-image $\phi^{-1}(y)$ of a point $y \in Y$ is itself a point and so we do not need to integrate or sum over points in the pre-image which will often be difficult to do analytically.

Bijectivity is a very limiting condition however, with many models involving non-bijective transformations of random variables. Later in this thesis we will see that methods used for defining the more general forms for calculating the density of a transformed variable are key to proposed methods for performing inference in generative models defined by complex, non-dimension preserving and non-bijective transformations of random variables.

1.1.5 Expectations

A fundamental operation when working with probabilistic models is computing expectations of random variables. Let $(S, \mathcal{E}, \mathsf{P})$ be a probability space, and $\mathsf{x} : S \to X$ a random variable on this space. Then the *expected value of* x is defined as

$$\mathbb{E}[\mathsf{x}] = \int_S \mathsf{x}(s)\, \mathrm{dP}(s). \tag{1.29}$$

Often it will be more convenient to express expectations in terms of the probability P_x instead. If $f : S \to X$ is a measurable function and μ a measure on S then the integral with respect to the pushforward measure μ_f of an integrable function g satisfies

$$\int_X g(x)\, \mathrm{d}\mu_f(x) = \int_S g \circ f(s)\, \mathrm{d}\mu(s). \tag{1.30}$$

If we take g as the identity map we therefore have that

$$\mathbb{E}[\mathsf{x}] = \int_X x\, \mathrm{dP}_\mathsf{x}(x). \tag{1.31}$$

If P_x is given by a density $\mathsf{p}_\mathsf{x} = \frac{\mathrm{dP}_\mathsf{x}}{\mathrm{d}\mu}$ then using (1.9) we also have

$$\mathbb{E}[\mathsf{x}] = \int_X x\, \mathsf{p}_\mathsf{x}(x)\, \mathrm{d}\mu(x), \tag{1.32}$$

which is often the form used for computation.

A further useful implication of (1.30) is what is sometimes termed the *Law of the unconscious statistician*. Let $\mathsf{x} : S \to X$ be a random variable, $\phi : X \to Y$ a measurable function and define $\mathsf{y} = \phi \circ \mathsf{x}$. Then the expected value of y is

$$\mathbb{E}[\mathsf{y}] = \int_S \mathsf{y}(s)\, \mathrm{dP}(s) = \int_S \phi \circ \mathsf{x}(s)\, \mathrm{dP}(s) = \int_X \phi(x)\, \mathrm{dP}_\mathsf{x}(x), \tag{1.33}$$

i.e. it can be calculated by integrating ϕ with respect to P_x. This means we can calculate expectations of a transformed random variable $\mathsf{y} = \phi(\mathsf{x})$ without needing to use the change of variables formulae from Section 1.1.4 to explicitly calculate the probability P_y (or density p_y) and with a relatively weak condition of measurability on ϕ.

1.1.6 Conditional expectations and densities

A related concept, and one which will be key in our discussion of inference, is conditional expectation. Let $(S, \mathcal{E}, \mathsf{P})$ be a probability space, $(X, \mathcal{F})$ and $(Y, \mathcal{G})$ two measurable spaces and $\mathsf{x} : S \to X$ and $\mathsf{y} : S \to Y$ two random variables. Then the *conditional expectation of* x *given* y, is defined as a measurable function $\mathbb{E}[\mathsf{x} \mid \mathsf{y}] : Y \to X$ satisfying

$$\int_{\mathsf{y}^{-1}(A)} \mathsf{x}(s) \, \mathrm{dP}(s) = \int_A \mathbb{E}[\mathsf{x} \mid \mathsf{y}](y) \, \mathrm{dP}_{\mathsf{y}}(y) \quad \forall A \in \mathcal{G}. \tag{1.34}$$

$\mathbb{E}[\mathsf{x} \mid \mathsf{y}]$ is guaranteed to be uniquely defined almost everywhere in Y by (1.34), i.e. up to P_{y}-null sets. As a particular case where $A = Y$ we recover what is sometimes termed the *Law of total expectation*

$$\int_S \mathsf{x} \, \mathrm{dP} = \int_S \mathbb{E}[\mathsf{x} \mid \mathsf{y}] \circ \mathsf{y} \, \mathrm{dP} \implies \mathbb{E}[\mathsf{x}] = \mathbb{E}[\mathbb{E}[\mathsf{x} \mid \mathsf{y}] \circ \mathsf{y}]. \tag{1.35}$$

We will also use an alternative notation for the conditional expectation evaluated at point $\mathbb{E}[\mathsf{x} \mid \mathsf{y} = y] \equiv \mathbb{E}[\mathsf{x} \mid \mathsf{y}](y)$ but use the latter in this section to stress its definition as a measurable function.

We can use conditional expectation to motivate the definition of a conditional density. Assume a joint density $\mathsf{p}_{\mathsf{x},\mathsf{y}} = \frac{\mathrm{dP}_{\mathsf{x},\mathsf{y}}}{\mathrm{d}(\mu_{\mathsf{x}} \times \mu_{\mathsf{y}})}$ exists and has marginal density $\mathsf{p}_{\mathsf{y}} = \frac{\mathrm{dP}_{\mathsf{y}}}{\mathrm{d}\mu_{\mathsf{y}}}$. Then for all $A \in \mathcal{G}$

$$\begin{aligned}
\int_{\mathsf{y}^{-1}(A)} \mathsf{x}(s) \, \mathrm{dP}(s) &= \int_S \mathsf{x}(s) \, \mathbb{1}_A \circ \mathsf{y}(s) \, \mathrm{dP}(s) && (1.36)\\
&= \int_{X \times Y} x \, \mathbb{1}_A(y) \, \mathrm{dP}_{\mathsf{x},\mathsf{y}}(x, y) && (1.37)\\
&= \int_A \int_X x \, \mathsf{p}_{\mathsf{x},\mathsf{y}}(x, y) \, \mathrm{d}\mu_{\mathsf{x}}(x) \, \mathrm{d}\mu_{\mathsf{y}}(y). && (1.38)
\end{aligned}$$

Define $g : Y \to X$ as

$$g(y) = \begin{cases} \int_X x \, \frac{\mathsf{p}_{\mathsf{x},\mathsf{y}}(x,y)}{\mathsf{p}_{\mathsf{y}}(y)} \, \mathrm{d}\mu_{\mathsf{x}}(x) & \forall y \in Y : \mathsf{p}_{\mathsf{y}}(y) > 0 \\ 0 & \forall y \in Y : \mathsf{p}_{\mathsf{y}}(y) = 0. \end{cases} \tag{1.39}$$

Then from (1.38) we have that for all $A \in \mathcal{G}$

$$\int_{\mathsf{y}^{-1}(A)} \mathsf{x}(s) \, \mathrm{dP}(s) = \int_A g(y) \, \mathsf{p}_{\mathsf{y}}(y) \, \mathrm{d}\mu_{\mathsf{y}}(y) = \int_A g(y) \, \mathrm{dP}_{\mathsf{y}}(y). \tag{1.40}$$

The definition of g in (1.39) therefore satisfies the definition of conditional expectation in (1.34) and is uniquely defined up to a P_y-null set. Therefore if $\mathsf{p}_{\mathsf{x},\mathsf{y}}$ and p_y can be defined we have that

$$\mathbb{E}[\mathsf{x} \mid \mathsf{y}](y) = \int_X x \, \mathsf{p}_{\mathsf{x}|\mathsf{y}}(x \mid y) \, \mathrm{d}\mu_\mathsf{x}(x) \quad \forall y \in Y : \mathsf{p}_\mathsf{y}(y) > 0 \tag{1.41}$$

where the *conditional density of* x *given* y, $\mathsf{p}_{\mathsf{x}|\mathsf{y}}$, is defined as

$$\mathsf{p}_{\mathsf{x}|\mathsf{y}}(x \mid y) = \frac{\mathsf{p}_{\mathsf{x},\mathsf{y}}(x, y)}{\mathsf{p}_\mathsf{y}(y)} \quad \forall x \in X, \, y \in Y : \mathsf{p}_\mathsf{y}(y) > 0 \tag{1.42}$$

which can be seen to be analagous to the definition of conditional probability in (1.5). Note the definition of conditional expectation in (1.34) was not dependent on a joint density $\mathsf{p}_{\mathsf{x},\mathsf{y}}$ being defined and so is more general than (1.41).

1.2 GRAPHICAL MODELS

Graphical models = statistics × graph theory × computer science
—Zoubin Ghahramani

When working with probabilistic models involving large numbers of random variables, it will often be the case that not all the variables are jointly dependent on each other but that instead there are more local conditional relationships between them. Graphical models, which use graphs to describe the dependencies between random variables, are a useful framework for visualising the structure in complex probabilistic models and for giving a graph-theoretic basis for establishing the dependence between sets of random variables.

Central to all graphical models is the concept of conditional independence. Let $(S, \mathcal{E}, \mathsf{P})$ be a probability space and $\mathsf{x} : S \to X$, $\mathsf{y} : S \to Y$ and $\mathsf{z} : S \to Z$ be three random variables with corresponding σ-algebras, $\mathcal{F}_\mathsf{x}$, $\mathcal{F}_\mathsf{y}$ and $\mathcal{F}_\mathsf{z}$ respectively. Following from our earlier definition of (unconditional) independence of random variables in (1.4), we say that x *and* y *are conditionally independent given* z, denoted $\mathsf{x} \perp \mathsf{y} \mid \mathsf{z}$, if

$$\mathbb{E}[\mathbb{1}_A \circ \mathsf{x} \; \mathbb{1}_B \circ \mathsf{y} \mid \mathsf{z}] = \mathbb{E}[\mathbb{1}_A \circ \mathsf{x} \mid \mathsf{z}] \, \mathbb{E}[\mathbb{1}_B \circ \mathsf{y} \mid \mathsf{z}] \;\; \forall A \in \mathcal{F}_\mathsf{x}, \, B \in \mathcal{F}_\mathsf{y}, \tag{1.43}$$

holds almost everywhere with respect to P_z. If a joint density on the random variables exists, a sufficient condition for $\mathsf{x} \perp \mathsf{y} \mid \mathsf{z}$ is that the conditional density $\mathsf{p}_{\mathsf{x},\mathsf{y}|\mathsf{z}}$ factorises as

$$\mathsf{p}_{\mathsf{x},\mathsf{y}|\mathsf{z}}(x, y \mid z) = \mathsf{p}_{\mathsf{x}|\mathsf{z}}(x \mid z) \mathsf{p}_{\mathsf{y}|\mathsf{z}}(y \mid z) \quad \forall x \in X, \, y \in Y, \, z \in Z. \tag{1.44}$$

(a) Directed graphical model. (b) Undirected graphical model.

Figure 1.1: Examples of directed and undirected graphical models. Circular nodes represent random variables in the model, with edges between them indicating dependencies between variables.

This definition can be naturally extended to conditional independence when conditioning on more than one random variable, for example

$$\mathsf{v} \perp \mathsf{x} \mid \mathsf{y},\, \mathsf{z} \implies \mathsf{p}_{\mathsf{v},\mathsf{x}\mid\mathsf{y},\mathsf{z}}(v, x \mid y, z) = \mathsf{p}_{\mathsf{v}\mid\mathsf{y},\mathsf{z}}(v \mid y, z)\mathsf{p}_{\mathsf{x}\mid\mathsf{y},\mathsf{z}}(x \mid y, z) \quad (1.45)$$

1.2.1 Directed and undirected graphical models

Several different graphical frameworks have been proposed for representing conditional independency relationships (and other information) in probabilistic models.

Directed graphical models [28], also known as *Bayesian networks*, represent probabilistic models as *directed acyclic graphs* (i.e. a directed graph in which there are no directed cycles), with the nodes in the graph representing random variables in the model and the edges of the graph defining a factorisation of the joint density over these variables into a product of conditional and marginal densities. In particular a conditional density factor is included for each node with parents (on the node random variable value given the parent variable values) and a marginal density factor for each root node without any parents.

An example directed graphical model for three random variables, x_1, x_2 and x_3, is shown in Figure 1.1a. The graph implies that the joint density can be factorised as

$$\mathsf{p}_{\mathsf{x}_1,\mathsf{x}_2,\mathsf{x}_3}(x_1, x_2, x_3) = \mathsf{p}_{\mathsf{x}_3\mid\mathsf{x}_1,\mathsf{x}_2}(x_3 \mid x_1, x_2)\, \mathsf{p}_{\mathsf{x}_1}(x_1)\, \mathsf{p}_{\mathsf{x}_2}(x_2). \quad (1.46)$$

Note that this factorisation would not be valid for all joint densities on the three variables; in particular we have that x_1 and x_2 are (unconditionally) independent and so that the joint density $\mathsf{p}_{\mathsf{x}_1,\mathsf{x}_2}$ can be written as the product of the two marginals $\mathsf{p}_{\mathsf{x}_1}$ and $\mathsf{p}_{\mathsf{x}_2}$.

Directed graphical models are a natural way of specifying *generative models* - i.e. probabilistic models which can be used to generate simulated observable quantities. Typically the factorisation specified by a directed graphical model gives a straightforward method to generate values from the joint density via *ancestral sampling*.

Ancestral sampling in a directed graphical model corresponds to first sampling values from all the root nodes from their marginal densities, then iteratively sampling from the conditional densities on each node for which all the parents nodes already have sampled values to condition on.

An alternative formalism for graphically representing probabilistic models is that of *undirected graphical models* [19], which are also known as *Markov random fields*. As with directed graphical models, each node in the graph represents a random variable, but here the edges connecting nodes are undirected. Rather than describing a factorisation of a joint density into conditional and marginal densities, an undirected graphical model indicates the factorisation of a joint density into a product of clique potentials on each of the maximal cliques in the graph.

A *clique* is a fully connected component of the graph - i.e. a subset of nodes in the graph such that all pairs of nodes in the subset are connected by an edge. A *maximal clique* is a clique which is not a strict subset of any other clique. A *clique potential* is a non-negative function of the values of the random variables in the clique; it does necessarily correspond to any conditional or marginal probabilty density.

An example undirected graphical model on four random variables, x_1, x_2, x_3 and x_4, is shown in Figure 1.1b. Here the (maximal) cliques correspond to all the connected pairs of nodes. If $\psi_{a,b}$ denotes the clique potential on the pair (a, b) then the graphical model implies the joint density can be factorised as

$$\begin{aligned} \mathrm{p}_{\mathrm{x}_1,\mathrm{x}_2,\mathrm{x}_3,\mathrm{x}_4}(x_1, x_2, x_3, x_4) = \\ \frac{1}{Z}\psi_{\mathrm{x}_1,\mathrm{x}_2}(x_1, x_2)\psi_{\mathrm{x}_1,\mathrm{x}_3}(x_1, x_3)\psi_{\mathrm{x}_2,\mathrm{x}_4}(x_2, x_4)\psi_{\mathrm{x}_3,\mathrm{x}_4}(x_3, x_4), \end{aligned} \tag{1.47}$$

with Z a normalising constant such that the density integrates to 1 and so defines a valid probability measure.

Undirected graphical models are a natural representation for models of systems of mutually interacting components. For example they are commonly used in models of images to represent dependencies between pixel values and models of ferromagnetism to represent interactions between lattices of particles.

Unlike directed models, generating joint configurations of the random variables in an undirected graphical model from the implied joint distri-

bution is typically a non-trivial task, with no general equivalent to ancestral sampling. Further the joint density can typically only be evaluated up to an unknown normalising constant, with the integral needed to evaluate this constant often intractable for models involving a large number of variables or complex potentials. As we will see, these properties mean that inference in distributions defined by undirected graphical models is often particularly challenging.

D-separation: $x \perp y \mid C \iff$ *all paths in the graph between* x *and* y *are blocked. A path is blocked if at least one of the following holds: 1. The path includes a* $\rightarrow\bigcirc\rightarrow$ *node or a* $\leftarrow\bigcirc\rightarrow$ *node in* C*. 2. The path includes a* $\rightarrow\bigcirc\leftarrow$ *node and neither the node or its descendants are in* C*.*

U-separation: x *and* y *in the model and a conditioning set of random variables* C*,* $x \perp y \mid C \iff$ *at least one random variable node on every path between* x *and* y *is in* C*.*

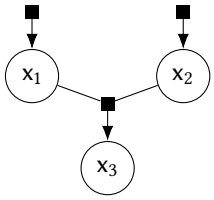
As suggested at the start of this section, both directed and undirected graphical models encode conditional independence properties of probabilistic models. In particular the rules of *D-separation* for directed graphical models and *U-separation* for undirected model give graph-based algorithmic descriptions of how to determine whether a pair of random variables are conditionally independent for a given conditioning set of random variables.

For example the directed graphical model in Figure 1.1a encodes the (un)conditional independence property $x_1 \perp x_2 \mid \emptyset = x_1 \perp x_2$ i.e. that x_1 and x_2 are independent if the value of x_3 is *not* conditioned on. The undirected graphical model in Figure 1.1b encodes the conditional independence properties $x_1 \perp x_4 \mid x_2, x_3$ and $x_2 \perp x_3 \mid x_1, x_4$.

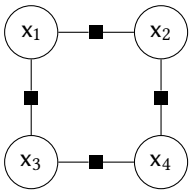
Although there are methods to convert a directed graphical model to an undirected one and vice versa, in general these transformations are lossy - not all of the conditional independence relationships encoded in the original graph will necessarily be maintained in the transformed graph. For example there is no undirected graphical model which will represent the exact set of conditional independence properties represented by the directed graphical model in Figure 1.1a. Likewise there is no directed graphical model which will represent the exact set of conditional independence properties represented by the undirected graphical model in Figure 1.1b. Further there are distributions with dependency structures and factorisations which cannot be uniquely represented by either directed or undirected graphical models [12].

1.2.2 Factor graphs

An alternative graphical model formalism which overcomes some of the limitations of directed and undirected graphical models is that of factor graphs [12, 13]. In factor graphs, in addition to nodes representing random variables, represented as in directed and undirected graphical



(a) A factor graph equivalent of the directed model in Figure 1.1a.



(b) A factor graph equivalent of the undirected model in Figure 1.1b.

Figure 1.2: Examples of factor graphs corresponding to the directed and undirected graphical models in Figure 1.1. Square black nodes correspond to individual factors depending on the connected variables (represented by circular nodes) in the joint density.

models by circular nodes, a second class of nodes, denoted by filled squares (■), are introduced which represent individual factors in the joint density across the random variables represented in the model.

Factors may be either directed or undirected. Undirected factors, denoted by factor nodes in which all edges connecting to variable nodes are undirected, correspond to a factor in the joint density which depends on all of the variables with nodes connected to the factor, but without any requirement that the factor corresponds to a conditional or marginal probability density. Directed factors, denoted by factor nodes in which at least one edge from the factor node to a variable node is directed, correspond to a conditional density on the variables pointed to by directed edges given the values of the variables connected to the the factor node by undirected edges (if there are no such variables then the factor instead corresponds to a marginal density).

Edges between nodes in a factor graph are always between nodes of disparate types i.e. between factor and variable nodes, but never between factor and factor or variable and variable nodes. As with directed graphical models, factor graphs with directed factors must not contain any directed cycles (i.e. a connected loop of edges in which one of every pair of edges connected to a factor on the loop is directed and all of the directed edges point in the same sense around the loop).

In the original extension of undirected factor graphs [13] to include directivity [12], it was proposed to allow multiple directed factors to connect via directed edges to the same variable node, representing multiple factors in a conditional density on that varaiable. This generalisation introduces extra normalisation requirements and looses the interpreta-

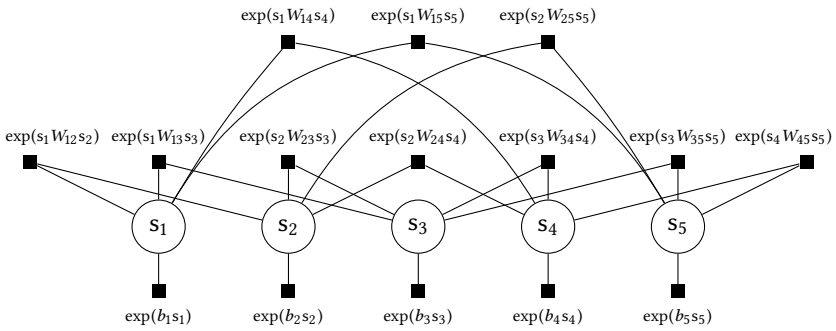


Figure 1.3: Five unit Boltzmann machine factor graph showing explicit factorisation of distribution into pairwise and single variable potentials.

tion of a directed factor as directly representing a conditional density, and so we will here only use directed factor graphs in which there is at most one directed edge connecting from a factor to a node.

Whether two variables are conditionally independent given a set of other variables can be checked from a factor graph by checking if all paths (i.e. connected series of edges and nodes) between the two corresponding variables nodes in the factor graph are *blocked*. A path is blocked if at least one of the following conditions is satisfied [12]

1. One of the variable nodes in the path is in the conditioning set.
2. One of the directed factor nodes in the path has two connected undirected edges in the path and there is no second directed path from the node to a variable node in the conditioning set.

Both directed and undirected graphical models can always be losslessly converted to a factor graph, i.e. such that by applying the above blocking rules after the transformation we obtain exactly the same set of conditional independency properties as present in the original graph, and thus they have a superset of the capacity to represent conditional indendence properties as either of these two alternative frameworks. For example, factor graph equivalents of the directed and undirected graphical model examples in Figure 1.1 are shown in Figure 1.2.

As well as allowing representations of mixed graphs with both directed and undirected factors which cannot be represented with either directed or undirected graphical models, factor graphs are also able to include finer-grained information about the factorisation of the joint density than either of the other two model types by explicitly indicating the presence of individual factors. For instance Figure 1.3 shows the

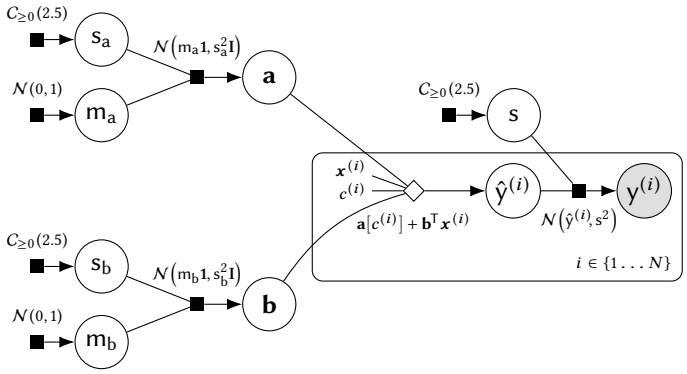


Figure 1.4: Hierarchical linear regression model factor graph showing examples of extended factor graph notation.

factor graph for a *Boltzmann machine* distribution, sometimes called a *pairwise binary Markov random field* or *Ising model*, on five binary random variables $\{\mathrm{s}_i\}_{i=1}^5$. A Boltzmann machine distribution can be factored in to a product of pairwise weighted interactions $\exp(\mathrm{s}_i W_{ij} \mathrm{s}_j)$ and single variable bias potentials $\exp(b_i \mathrm{s}_i)$, each of which are explicitly represented by labelled factors in Figure 1.3. A corresponding undirected graphical model representation would have a single clique involving all five variables, and so would not indicate any information about the factorisation of the joint density.

In Figure 1.4 we illustrate some additional useful factor graph notation we will use in this thesis. We use a factor graph corresponding to a hierarchical linear regression model which will be discussed in more detail later in the thesis as a motivating example. The exact meaning of the model and its various factors are unimportant to the discussion of notation here so will be skipped for now.

It will often be useful to be able to explicitly represent deterministic functions applied to the random variables in a factor graph. For this purpose we introduce an additional node type denoted by an unfilled diamond (◇). The semantics of this node type are very similar to standard directed factor nodes. Variables acting as inputs to the function are connected to the node by undirected edges and the variable corresponding to the function output indicated by a directed edge from the node to the relevant variable. Like standard factor nodes, the deterministic factor nodes only ever connect to variable nodes. The operations performed by the function on the inputs will usually be included as a label adjacent to the node as illustrated by the example in Figure 1.4.

A deterministic factor node can informally[1] be considered equivalent to a directed factor node with a degenerate Dirac delta conditional density on the output variable which concentrates all the probability mass at the output of the function applied to the inputs variables. The previously discussed rules for evaluating conditional independency properties in factor graphs can be directly extended to account for the new node type by just considering it as a directed factor node.

Optionally constant values used in a model may be included in a factor graph as plain nodes indicated only by a label. The $\boldsymbol{x}^{(i)}$ and $c^{(i)}$ nodes in Figure 1.4 are an example of this notation.

A commonly used convention in factor graphs (and other graphical models) is *plate notation* [6], with an example of a plate shown by the rounded rectangle bounding some of the nodes in Figure 1.4. Plates are used to indicate a subgraph in the model which is replicated multiple times (with the replications being indexed over a set which is typically indicated in the lower right corner of the plate as in Figure 1.4). The subgraph entirely contained on the plate is assumed to be replicated the relevant number of times, with any edges crossing into the plate from variable nodes outside of the plate being repeated once for each subgraph replication. Plates are commonly used to represent a model component repeated across multiple data items.

Each of the factors in Figure 1.4 is labelled with a shorthand for a probability density function corresponding to the conditional or marginal density factor associated with the node. Definitions for the shorthand notations that are used for densities in this thesis are given in Tables 1.2 and 1.3. The dependence of the factors on the value of the random variable the density is defined on is omitted in the labels for brevity.

A final additional notation used in Figure 1.4 is the use of a shaded variable node (corresponding to $\mathrm{y}^{(i)}$) to indicate a random variable corresponding to an observable quantity in the model.

1.2.3 Computation graphs

A final graph based tool we will make use of in this theis is that of *computation graphs* [2]. In particular computation graphs (via associated

1 A Dirac delta cannot strictly define a density as it is not the Radon–Nikodyn derivative of an absolutely continuous measure, however it can be informally treated as the density of a singular Dirac measure $f(0) = \int f(x)\,\mathrm{d}\delta(x) \simeq \int f(x)\delta(x)\,\mathrm{d}x$.

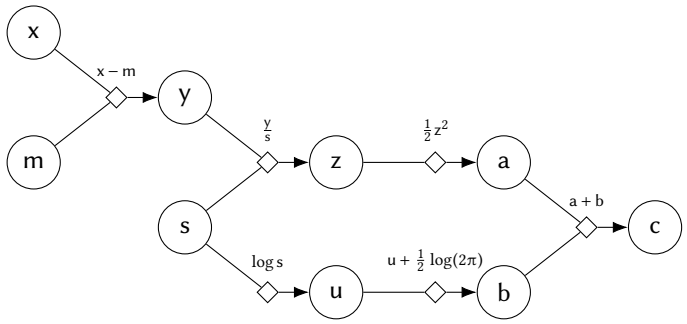


Figure 1.5: Example computation graph corresponding to calculation of the negative log density of a univariate normal distribution.

software frameworks [32]) will be used to allow automatic differentiation of complex probabilistic models used in later chapters. Computation graphs are not typically considered in the context of probabilistic graphical models, but they share many of the same features and as we will see are closely related to directed factor graphs.

A *computation graph*, sometimes insead termed a *computational graph* or *data flow graph*, represents the computations involved in evaluating a mathematical expression. In this thesis we will distinguish between two types of nodes in a computation graph. *Variable nodes* correspond to variables which hold either inputs to the computation or intermediate results corresponding to the outputs of sub-expressions. *Operation nodes* describe how non-input variable nodes are computed as functions of other variable nodes. In other presentations of computation graphs often the operation nodes are instead implicitly represented by directed edges between variable nodes. However analagously to the more explicit factorisation afforded by directed factor graphs compared to directed graphical models, directly representing operations as nodes allows finer grained information about the decomposition of the operations associated with a computation graph to be included.

As with directed graphical models and directed factor graphs, computation graphs cannot contain directed cycles. This does not preclude recursive and recurrent computations however as these can always be unrolled to form a directed acyclic graph. The 'mathematical expressions' a computation graph is constructed to evaluate can be arbitarily complex - a computation graph corresponding to the evaluation of any numerical algorithm can always be constructed including use of arbitrary nested flow control and branching statements.

An example of a computation graph representing the calculation of the negative log density of a univariate normal distribution, i.e.

$$c = \frac{1}{2}\left(\frac{x - m}{s}\right)^2 + \log s + \frac{1}{2}\log(2\pi) \tag{1.48}$$

is shown in Figure 1.5. The graph inputs have chosen to be the value of the random variable (x) to evaluate the density at and the mean (m) and the standard deviation (s) parameters of the density.

Variable nodes in the computation graph have been represented by labelled circles and operation nodes with labelled diamonds. Undirected edges connecting from a variable node to an operation node correspond to the inputs to the operation, and directed edges from an operation node to variable nodes to the outputs of the operation.

The computation graph associated with a given expression is not uniquely defined. There will usually be multiple possible orderings in which operations can be applied to achieve the same result (up to differences due to non-exact floating point computation). Similarly what should considered a single operation to be represented by a node in the computation graph as opposed to being split up into a sub-graph of multiple operations is a matter of choice. For example in Figure 1.5 the addition of the constant $\frac{1}{2}\log(2\pi)$ could have been included at various other points in the graph and the operation $\frac{1}{2}z^2$ could have been split in to separate multiplication and exponentation operations.

The main motivation for representing expressions as computation graphs is to formalise an efficient general procedure for automatically calculating derivatives of the output of an expression with respect to its inputs termed automatic differentiation [4, 27]. The key ideas in automatic differentiation are to use the chain rule to decompose the derivatives into products and sums of the partial derivatives of the output of each individual operation in the expression with respect to its input, and to use an efficient recursive accumulation of these partial derivative sum-products corresponding to a traversal of the computation graph such that multiple derivatives can be efficiently calculated together.

Depending on how the computation graph is traversed to accumulate the derivative terms, different modes of automatic differentiation are possible. Of most use in this thesis will be *reverse-mode accumulation* [30], in which the derivatives of an output node with respect to all in-

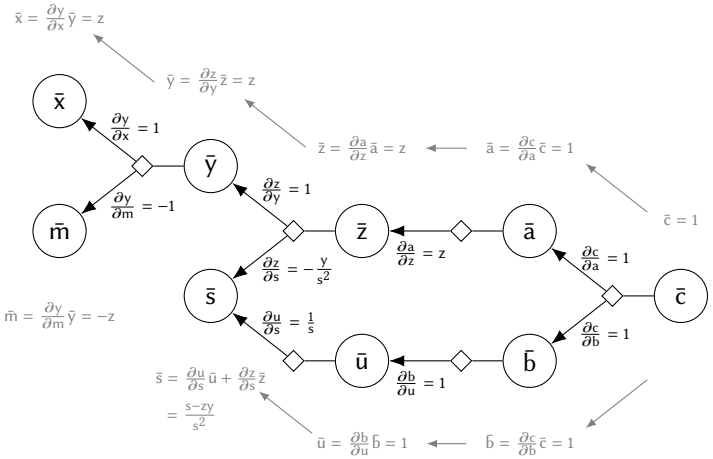


Figure 1.6: Visualisation of applying reverse-mode automatic differentiation to the computation graph in Figure 1.5 to calculate the derivatives of the negative log density of a univariate normal distribution.

put nodes are accumulated by a reverse pass through the computation graph from the output node to inputs.

As an example the partial derivatives of the expression for univariate normal log density given in (1.48) with respect to x, m and s can be decomposed using the chain rule in terms of the intermediate variables in the computation graph shown in Figure 1.5 as

$$\frac{\partial c}{\partial x} = \frac{\partial c}{\partial a}\frac{\partial a}{\partial z}\frac{\partial z}{\partial y}\frac{\partial y}{\partial x}, \tag{1.49}$$

$$\frac{\partial c}{\partial m} = \frac{\partial c}{\partial a}\frac{\partial a}{\partial z}\frac{\partial z}{\partial y}\frac{\partial y}{\partial m}, \tag{1.50}$$

$$\frac{\partial c}{\partial s} = \frac{\partial c}{\partial a}\frac{\partial a}{\partial z}\frac{\partial z}{\partial s} + \frac{\partial c}{\partial b}\frac{\partial b}{\partial u}\frac{\partial u}{\partial s}. \tag{1.51}$$

We can immediately see that some of the chains of products of partial derivatives are repeated in the different derivative expressions - for example $\frac{\partial c}{\partial a}\frac{\partial a}{\partial z}$ appears in the expressions for all three derivatives. Reverse-mode accumulation is effectively an automatic way of exploiting these possibilities for reusing calculations.

Figure 1.6 shows a visualisation of reverse-mode accumulation applied to the computation graph in Figure 1.5. The first step is for a *forward pass* through the graph to be performed, i.e. values are provided for each of the input variables and then each of the intermediate and output variables calculated from the incoming operation applied to their

parent values. Importantly the values of all variables in the graph calculated during the forward pass must be maintained in memory.

The *reverse pass* recursively calculates the values of the partial derivatives of the relevant output node with respect to each variable node in the graph - we will term these intermediate derivatives *accumulators* denoted with barred symbols in Figure 1.6 e.g. $\bar{a} = \frac{\partial c}{\partial a}$. The reverse pass begins by seeding an accumulator for the output node to one (i.e. $\bar{c} = \frac{\partial c}{\partial c} = 1$ in Figure 1.6). Accumulators for the input variables of an operation are calculated by multiplying the accumulator for the operation output by the partial derivatives of the operation output with respect to each input variable. For non-linear operations multiplying by the operator partial derivatives will require access to the value of the input variables calculated in the forward pass. If a variable is an input to multiple operations, the derivative terms from each operation are added together in the relevant accumulator, as for example shown for $\bar{s}$ in Figure 1.6. By recursively applying these product and sum operations, the derivatives of the output with respect to all variables in the graph can be calculated.

This reverse accumulation method allows computation of numerically exact (up to floating point error) derivatives of a single output variable in a computation graph with respect to *all input variables* with a computational cost, in terms of the number of atomic operations which need to be performed, that is a constant factor of the cost of the evaluation of the original expression represented by the computation graph in the forward pass. The constant factor is typically two to three and at most six [3]. This efficient computational cost is balanced by the requirement that the values of all intermediate variables in the computation graph evaluated in the forward pass through the graph must be stored in memory for the derivative accumulation in a reverse pass, which for large computational graphs can become a bottleneck.

To calculate the full Jacobian from a computation graph representing a function with M inputs $\{x_i\}_{i=1}^{M}$ and N outputs $\{y_i\}_{i=1}^{N}$, i.e. the $N \times M$ matrix $\boldsymbol{J}$ with entries $J_{i,j} = \frac{\partial y_i}{\partial x_j}$, we can do a single forward pass and N reverse passes each time accumulating the derivatives of one output variable with respect to all inputs. This leads to an overall computational cost that is $O(N)$ times the cost of a single (forward) function evalaution to evaluate the full Jacobian. As each of the reverse passes can trivially be run in parallel (in addition to any parallelisation of the

operations in the forward and reverse passes themselves), this $O(N)$ factor in the operation count need not corresponds to an equivalent increase in compute time.

An alternative to reverse-mode accumulation is *forward-mode accumulation* [34], which insteads accumulates partial derivatives with respect to a single input variable alongside the forward pass through the graph. In contrast to reverse-mode, this allows calculation of the partial derivatives of all output variables with respect to a single input variable at a computational cost that is a constant factor of the cost of the evaluation of the original expression in the forward pass. Forward-mode accumulation therefore allows evaluation of the Jacobian of a function with M inputs and N outputs at an overall computational cost that is $O(M)$ times the cost of a single function evaluation.

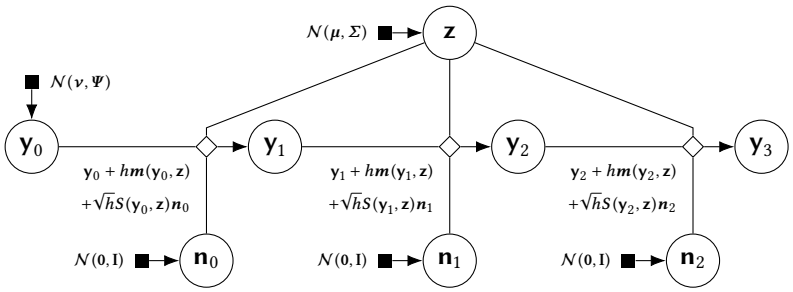
For functions with $M \gg N$, e.g. scalar valued functions of multiple inputs, reverse-mode accumulation is generally therefore signficantly more efficient at computing the Jacobian. Forward-mode accumulation is however useful for evaluating the Jacobian of functions with $N \gg M$, and also has the advantage over reverse-mode accumulation of avoiding the requirement to store the values of intermediate variables from the forward pass for the reverse pass(es).

The direct overlap in our notation to represent variable and operation nodes in computation graphs and that used to represent (random) variable nodes and deterministic factor nodes in factor graphs is intentional. Although often the operations associated with a deterministic node in a factor graph will be more complex than the operations usually represented by nodes in a computation graph, this is only a matter of granularity of reprensentation - fundamentally they perform the same role. Importantly this means we can treat subgraphs of a factor graphs consisting of only variable and deterministic factor nodes as computation graphs and if the operations performed by the deterministic nodes are differentiable, use reverse-mode automatic differentiation to efficiently propagate derivatives through these sub-graphs.

Like directed graphical models, a directed factor graph naturally specifies a generative process via ancestral sampling, with values for the random variables in the graph successively calculated in a forward pass consisting of a combination of deterministic and stochastic operations on the values of parent variables. A computation graph likewise spe-

$\mathbf{y}_0 \leftarrow \mathcal{N}(\boldsymbol{\nu}, \boldsymbol{\Psi})$
$\mathbf{z} \leftarrow \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
for $t \in \{1 \dots T\}$ **do**
    $\mathbf{n}_{t-1} \leftarrow \mathcal{N}(\mathbf{0}, \mathbf{I})$
    $\mathbf{y}_t \leftarrow \mathbf{y}_{t-1} + h\,\boldsymbol{m}(\mathbf{y}_{t-1}, \mathbf{z}) + \sqrt{h}\,S(\mathbf{y}_{t-1}, \mathbf{z})\mathbf{n}_{t-1}$

(a) Pseudo-code for Euler-Maruyama simulation of SDE model.



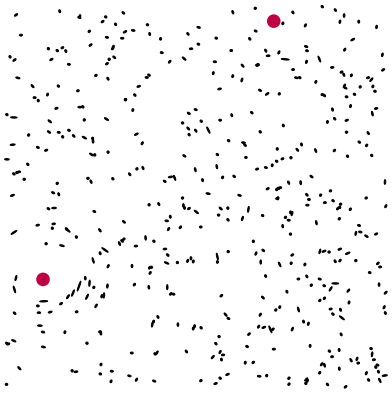
(b) Directed factor graph of 3 time steps of SDE simulation.

Figure 1.7: Example of a simulator model corresponding to Euler-Maruyma integration of a set of stochastic differential equations (SDEs), $\mathrm{d}\mathbf{y}(t) = \boldsymbol{m}(\mathbf{y}(t), \mathbf{z})\,\mathrm{d}t + S(\mathbf{y}(t), \mathbf{z})\,\mathrm{d}\mathbf{n}(t)$, specified as pseudo-code in (a) and a directed factor graph in (b). In the pseudo-code the notation $\leftarrow$ followed by a distribution shorthand represents generating a value from the associated distribution and assigning it to a variable.

cifies a generative process, how to compute the expression outputs given inputs, computed via a forward pass through the graph with the main differences being here that the inputs to that process are assumed to be given rather than sampled from marginal densities and the intermediate operations are all deterministic.

Rather than specifying a generative model via a directed factor graph (or graphical model), it is common for complex models to instead be specified procedurally in code as a *simulator*. Often such simulators may involve a mechanistic model of a physical process for example described by a set of stochastic differential equations (SDEs). Any stochasticity in a simulator model will be introduced via draws from a (pseudo-)random number generator in the programming language used to specify the model. Given these random inputs, the output of the simulator is then calculated as a series of determinstic operations performed to the inputs and so can be described by a computation graph. The overall composition of directed factor nodes specifying the generation of random inputs from known densities by the random number generator and computation graph describing the operations performed by the simulator code together therefore define a directed factor graph from which

(a) Hubble Space Telescope image of galaxy cluster MACS J1206 showing visible distortion due to gravitational lensing. Image credit: ESA/Hubble.



(b) Simulated galaxy cluster image from *Observing Dark Worlds* data [17] showing ellipticity of galaxies (black ellipses) distorted by dark matter halos (red circles).

Figure 1.8: Gravitational lensing in real and simulated galaxy cluster images.

we can extract a joint density on all the variables in the models as the product of all factors (with implicit Dirac delta terms on the outputs of deterministic factors). An example of a simulator model corresponding to Euler-Maruyma approximate integration of a set of SDEs is shown as both pseudo-code and a corresponding factor graph in Figure 1.7.

A key difference of simulator models from more typical probabilistic models is that the variables corresponding to observables in the factor graph of a simulator model may be the output of deterministic factors rather than probabilistic directed factors. As we will see later in the thesis this can complicate inference in such models.

1.3 INFERENCE

Having now introduced the tools we use to construct probabilistic models, we will now describe more concretely the task of inference. To help motivate our exposition we will discuss inference in the context of a specific problem: inferring the location of dark matter halos from the observed gravitational lensing of light emitted by surrounding galaxies. This task was the inspiration for a Kaggle[2] competition, *Observing Dark Worlds* [15, 17] and we will use the simplified formulation of the task used in the competition as the basis for our discussion here.

2 An online platform for predictive modelling competitions `https://www.kaggle.com`.

1.3.1 Example problem: Observing Dark Worlds

We will begin with a brief review of the motivation for the *Observing Dark Worlds* inference problem. Dark matter is a hypothesised form of matter which does not emit or interact with electromagnetic radiation. This prevents direct observation of dark matter by telescopes as it produces no signal in any part of the electromagnetic spectrum [24]. General relativity however predicts that all objects with mass locally distort spacetime. If a very large concentration of mass lies between an observer and a light source the strong distortion of spacetime by the mass significantly alters the paths of photons emitted by the source and causes the apparent image of the source to the observer to be visibly distorted [1]. This effect is analagous to placing a lens between the light source and observer and this motivates the term *gravitational lensing* to describe the phenonemon. It is believed that large concentrations of dark matter around galaxy clusters termed *dark matter halos*, cause gravitational lensing of the light emitted from background galaxies observed in telescope imagery of galaxy clusters.

In the context of gravitational lensing ellipticity is defined as a complex quantity $\epsilon = \frac{a-b}{a+b}\exp(2i\phi)$ *where* a *is the length of the ellipse major axis,* b *the minor axis length and* ϕ *the orientation of the major axis. Here we will parameterise ellipticity terms of* $e_1 = \mathrm{Re}(\epsilon)$ *and* $e_2 = \mathrm{Im}(\epsilon)$.

Galaxies typically have approximately elliptical shapes in telescope images. It is assumed that the *ellipticities* of observed galaxy images not subject to gravitational lensing are isotropically distributed [1]: there is no preferred orientation of the galaxies to an observer on Earth. The presence of a large mass concentration such as a dark matter halo between background galaxies and a telescope however locally distorts the distribution of the ellipticities of the observed galaxy images. An example of this effect in a galaxy cluster image from the Hubble Space Telescope is shown in Figure 1.8a with the galaxies showing a bias towards being oriented tangentially to the bright region at the centre of the image. Local biases in the spatial distribution of galaxy ellipticities can therefore be used to infer the location of dark matter halos.

The *Observing Dark Worlds* inference task is formulated as follows. The observed data consists of the sky co-ordinates $\{u_g, v_g\}_{g=1}^{G}$ and ellipticity components $\{e_{1,g}, e_{2,g}\}_{g=1}^{G}$ of a set of G galaxies. The task is given this data to infer the coordinates $\{\mathsf{x}_h, \mathsf{y}_h\}_{h=1}^{H}$ of the centers of a known number of dark matter halos present in the sky field of view. Galaxy coordinate and ellipticity data are provided for multiple cluster images each with a known number of halos present and the cluster image data are assumed to be independently and identically distributed (iid).

The *Observing Dark Worlds* data is simulated thus the ground truth positions of the dark matter halos are known in reality which was required for the purposes of evaluation in the competition. An example visualisation of one of the simulated galaxy cluster images in the data set is shown in Figure 1.8b. The known positions of the dark matter halos in this image are shown by red circles[3]. As can be seen the halo in the lower left of this image produces a strong visible distortion in the ellipticities, however the second halo at the top right has a much less visible effect on the surrounding galaxies.

The use of simulated data reduces the difficulty of the *Observing Dark Worlds* inference task compared to working with real gravitational lensing data, however for the purposes of our discussion the task is still sufficiently complex to highlight the computational challenges involved in inference problems. Further the probabilistic approach described here is similar to that used (albeit with significantly more complex models) in methods and tools used in practice to analyse gravitational lensing data to infer dark matter properties [16, 23].

1.3.2 Defining a model

You cannot do inference without making assumptions
—David Mackay

Our starting point for tackling inference problems will be to define a probabilistic model specifying proposed relationships between the observed and non-observed quantities to be inferred. The model codifies the assumptions we make about the problem and any prior beliefs we have. In virtually all real inference problems the model will be a significantly simplified description of a much more complex underlying process, usually motivated by prior empirical observations that the behaviour proposed by the model is a reasonable description of reality. For now we will consider the model as a singular fixed object we will perform inference with. In reality probabilistic modelling and inference are an iterative process with model criticism a key part of the loop [5, 14]. We will discuss some of the (computational) issues involved in probabilistic model evaluation and comparison at the end of this chapter.

3 A *training set* of cluster image data was provided to competition participants for which the associated true dark matter halo positions were given. This data was distinct from the *test set* cluster data were the halo positions are unknown and which is the basis of the described inference problem and scoring for the competition, with the training data intended to aid initial model exploration and evaluation.

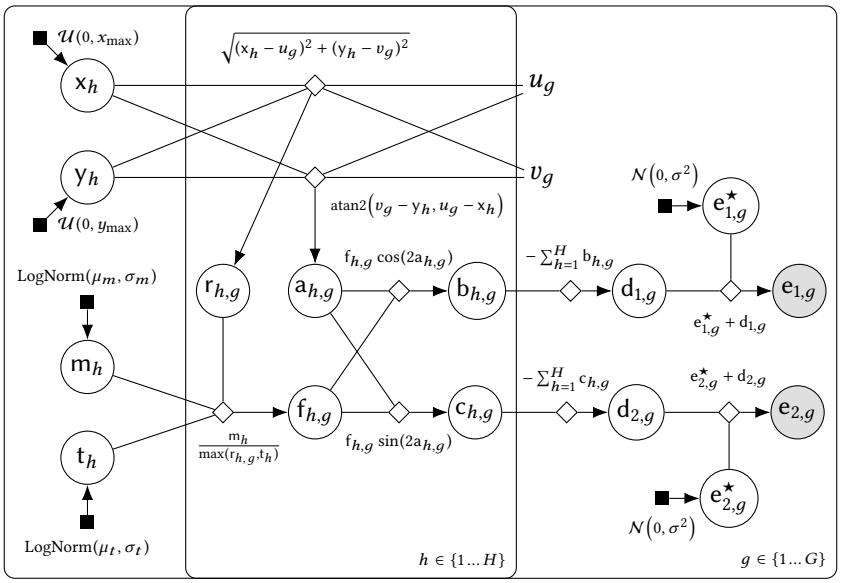


Figure 1.9: Factor graph of proposed model for *Observing Dark Worlds* gravitational lensing inference problem for a single cluster image.

For the *Observing Dark Worlds* problem, a proposed probabilistic model is shown as a directed factor graph in Figure 1.9[4]. This model assumes a simple, physically motivated relationship between the dark matter halo positions and the observed spatial distribution of galaxy ellipticities. As well as random variables for the halo coordinates $\{x_h, y_h\}_{h=1}^H$ and galaxy ellipticity components $\{e_{1,g}, e_{2,g}\}_{g=1}^G$, the model also introduces additional latent (non-observed) random variables $\{m_h, t_h\}_{h=1}^H$, which correpond respectively to the unknown total masses of each halo (multiplied by an arbitrary scaling factor) and an unknown *core radius* for each halo, which specifies a radial distance within which the distortion by the halo is of constant magnitude. The halo coordinates are assumed to be marginally uniformly distributed across the image extents. The halo mass and core radius latent variables are both positive quanties and are assumed to have log normal marginal distributions.

The model proposes that at sky coordinate (u, v) each halo produces a shear distortion of magnitude

$$f_h(u, v) = \frac{m_h}{\max(r_h, t_h)} \quad \text{where} \quad r_h = \sqrt{(x_h - u)^2 + (y_h - v)^2} \tag{1.52}$$

4 The factor graph in Figure 1.9 is based on the models used by the participants with the top-two winning entries in the Kaggle competition, Iain Murray and Tim Salimans, and described in their personal reports on their competition entries [26, 29] and an article evaluating the competition outcomes [15].

and acting in a tangential direction to the radial vector from the halo centre $(\mathsf{x}_h, \mathsf{y}_h)$ to (u, v). This functional relationship is just one possibility among many. The post-competition review article [15] revealed that the simulated data actually used dark matter halos with a mixture of two different radial density profiles, neither of which correspond to the distortion model assumed in (1.52). This mismatch between a model and the actual process by which observations were generated will virtually always be the case in real inference problems. We can still make inferences which are consistent with our modelling assumptions, however we should as far as possible also critically review those modelling assumptions to check the validity of the inferences made.

For galaxies where the variation of the magnitude of the distorting effect of a halo across the extent of the galaxy's image is small, a reasonable approximation of the gravitational lensing effect of a single halo on the observed ellipticity $(\mathsf{e}_{1,g}, \mathsf{e}_{2,g})$ of a galaxy image with intrinsic (prior to any gravitational lensing effect) ellipticity $(\mathsf{e}^\star_{1_g}, \mathsf{e}^\star_2)$ is that

$$\mathsf{e}_{1,g} = \mathsf{e}^\star_{1,g} - \mathsf{f}_{h,g}\cos(2\mathsf{a}_{h,g}), \quad \mathsf{e}_{2,g} = \mathsf{e}^\star_{2,g} - \mathsf{f}_{h,g}\sin(2\mathsf{a}_{h,g}), \tag{1.53}$$

where $\mathsf{a}_{h,g} = \operatorname{atan2}\big(v_g - \mathsf{y}_h, u_g - \mathsf{x}_g\big)$ is the angle the line from the centre of halo h to galaxy g makes to the horizontal axis and $\mathsf{f}_{h,g} = f_h(u_g, v_g)$ is the magnitude of the shear distortion according to the proposed relationship in Equation (1.52) evaluated at the galaxy image centre [1, 23]. For clusters with multiple halos, a further simple linearity assumption is made, that the shear distortions due to the different halos act additively on the ellipticity components

$$\mathsf{e}_{1,g} = \mathsf{e}^\star_{1,g} - \sum_{h=1}^{H} \mathsf{f}_{h,g}\cos(2\mathsf{a}_{h,g}), \; \mathsf{e}_{2,g} = \mathsf{e}^\star_{2,g} - \sum_{h=1}^{H} \mathsf{f}_{h,g}\sin(2\mathsf{a}_{h,g}). \tag{1.54}$$

The intrinsic ellipticities of the galaxies are assumed to have a isotropic, zero-mean normal distribution, with standard deviation σ, with normal assumption corresponding well to empirical observations from large scale surveys [1, 23]. For now we will assume the standard deviation σ of the intrinsic ellipticities, and the parameters μ_m, σ_m, μ_t, σ_t of the halo mass and core radius marginal distributions have somehow been set to reasonable values, for example based on prior beliefs about the typical ranges of the variables. We will discuss possible strategies for choosing (or inferring) these parameters in more detail later.

1.3.3 Making predictions

Having now defined a model for the *Observing Dark Worlds* problem, we now consider how to use this model to make predictions about the unobserved dark matter halo positions. The downstream task we are using the inference output for will generally determine what the exact inferential query we wish to evaluate is. In general however, any prediction output which takes into account all of the information we have about the unobserved variables given the assumed model and observed data will be computed as a conditional expectation.

To motivate this statement for the *Observing Dark Worlds* example we will discuss some instances of outputs we might wish to compute. For notational convenience we define the following random vectors for the halo random variables

$$\mathbf{x} = \begin{bmatrix} \mathrm{x}_1 \\ \vdots \\ \mathrm{x}_H \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} \mathrm{y}_1 \\ \vdots \\ \mathrm{y}_H \end{bmatrix}, \quad \mathbf{m} = \begin{bmatrix} \mathrm{m}_1 \\ \vdots \\ \mathrm{m}_H \end{bmatrix}, \quad \mathbf{t} = \begin{bmatrix} \mathrm{t}_1 \\ \vdots \\ \mathrm{t}_H \end{bmatrix}, \tag{1.55}$$

and similarly for the observed and intrinsic galaxy ellipticities

$$\mathbf{e}_1 = \begin{bmatrix} \mathrm{e}_{1,1} \\ \vdots \\ \mathrm{e}_{1,G} \end{bmatrix}, \quad \mathbf{e}_2 = \begin{bmatrix} \mathrm{e}_{2,1} \\ \vdots \\ \mathrm{e}_{2,G} \end{bmatrix}, \quad \mathbf{e}_1^\star = \begin{bmatrix} \mathrm{e}_{1,1}^\star \\ \vdots \\ \mathrm{e}_{1,G}^\star \end{bmatrix}, \quad \mathbf{e}_2^\star = \begin{bmatrix} \mathrm{e}_{2,1}^\star \\ \vdots \\ \mathrm{e}_{2,G}^\star \end{bmatrix}. \tag{1.56}$$

An obvious output we may wish to compute is the expected (mean) values of the halo positions given the observed ellipticities. From a decision theoretic standpoint these values correspond to the position predictions which minimise the expected squared error loss from the true positions given the model and observed data. As conditonal expectations these are simply

$$\boldsymbol{m}_{\mathbf{x}} = \mathbb{E}\left[\mathbf{x} \,|\, \mathbf{e}_1 = \boldsymbol{e}_1, \, \mathbf{e}_2 = \boldsymbol{e}_2\right] \text{ and } \boldsymbol{m}_{\mathbf{y}} = \mathbb{E}\left[\mathbf{y} \,|\, \mathbf{e}_1 = \boldsymbol{e}_1, \, \mathbf{e}_2 = \boldsymbol{e}_2\right]. \tag{1.57}$$

We may also be interested in the covariances of the halo positions conditioned on the observations, these giving some idea of our remaining uncertainty in the positions and any correlations between them after

conditioning on the observed ellipticities. Again these can be expressed as conditional expectations

$$\begin{aligned} \boldsymbol{C}_{\mathbf{x},\mathbf{x}} &= \mathbb{E}\Big[(\mathbf{x}-\boldsymbol{m}_{\mathbf{x}})(\mathbf{x}-\boldsymbol{m}_{\mathbf{x}})^{\mathsf{T}} \mid \mathbf{e}_1 = \boldsymbol{e}_1,\, \mathbf{e}_2 = \boldsymbol{e}_2\Big], \\ \boldsymbol{C}_{\mathbf{y},\mathbf{y}} &= \mathbb{E}\Big[(\mathbf{y}-\boldsymbol{m}_{\mathbf{y}})(\mathbf{y}-\boldsymbol{m}_{\mathbf{y}})^{\mathsf{T}} \mid \mathbf{e}_1 = \boldsymbol{e}_1,\, \mathbf{e}_2 = \boldsymbol{e}_2\Big], \\ \text{and} \quad \boldsymbol{C}_{\mathbf{x},\mathbf{y}} &= \mathbb{E}\Big[(\mathbf{x}-\boldsymbol{m}_{\mathbf{x}})(\mathbf{y}-\boldsymbol{m}_{\mathbf{y}})^{\mathsf{T}} \mid \mathbf{e}_1 = \boldsymbol{e}_1,\, \mathbf{e}_2 = \boldsymbol{e}_2\Big]. \end{aligned} \tag{1.58}$$

In the *Observing Dark Worlds* competition, participants' halo position predictions were evaluated by comparing to the known true halo positions using a metric provided as part of the competition instructions. If we denote the metric $\ell(\boldsymbol{x}, \boldsymbol{y}; \hat{\boldsymbol{x}}, \hat{\boldsymbol{y}})$ where $(\boldsymbol{x}, \boldsymbol{y})$ are the true halo positions and $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}})$ the predicted positions, then the optimal predictions given the assumed model and observed data can be calculated by minimising the expected metric value conditioned on the observed data

$$\bar{\ell}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}) = \mathbb{E}\big[\ell(\mathbf{x}, \mathbf{y}; \hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}) \mid \mathbf{e}_1 = \boldsymbol{e}_1, \mathbf{e}_2 = \boldsymbol{e}_2\big]. \tag{1.59}$$

Due to linearity of the expectation operator, we can also calculate the derivatives of the expected metric with respect to the predictions as

$$\begin{aligned} \frac{\partial \bar{\ell}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}})}{\partial \hat{\boldsymbol{x}}} &= \mathbb{E}\left[\frac{\partial \ell(\mathbf{x}, \mathbf{y}; \hat{\boldsymbol{x}}, \hat{\boldsymbol{y}})}{\partial \hat{\boldsymbol{x}}} \,\middle|\, \mathbf{e}_1 = \boldsymbol{e}_1, \mathbf{e}_2 = \boldsymbol{e}_2\right] \\ \text{and} \quad \frac{\partial \bar{\ell}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}})}{\partial \hat{\boldsymbol{y}}} &= \mathbb{E}\left[\frac{\partial \ell(\mathbf{x}, \mathbf{y}; \hat{\boldsymbol{x}}, \hat{\boldsymbol{y}})}{\partial \hat{\boldsymbol{y}}} \,\middle|\, \mathbf{e}_1 = \boldsymbol{e}_1, \mathbf{e}_2 = \boldsymbol{e}_2\right]. \end{aligned} \tag{1.60}$$

For some metrics we will be able to analytically solve for the stationary points to find the predictions minimising $\bar{\ell}(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}})$ and more generally we can use the derivatives in an iterative optimisation scheme.

Evaluating conditional expectations of functions of the unobserved variables in a model given observed data is therefore the key computational task in making inferences about the variables in a probabilistic model. The *Observing Dark Worlds* model factor graph in Figure 1.9 defines a joint density across all the variables in the model. We can therefore use Equation (1.41) to write the conditional expectation of a measurable function f of the halo position random variables $\mathbf{x}$ and $\mathbf{y}$ as[5]

$$\begin{aligned} &\mathbb{E}\big[f(\mathbf{x}, \mathbf{y}) \mid \mathbf{e}_1 = \boldsymbol{e}_1, \mathbf{e}_2 = \boldsymbol{e}_2\big] = \\ &\qquad \iint f(\boldsymbol{x}, \boldsymbol{y})\, \mathrm{p}_{\mathbf{x},\mathbf{y}\mid\mathbf{e}_1,\mathbf{e}_2}(\boldsymbol{x}, \boldsymbol{y} \mid \boldsymbol{e}_1, \boldsymbol{e}_2)\, \mathrm{d}\boldsymbol{x}\, \mathrm{d}\boldsymbol{y}, \end{aligned} \tag{1.61}$$

5 For brevity the sets on which integrals are evaluated have been omitted in this section and should be assumed to be the full co-domain of the corresponding random vector.

where the conditional density $\mathsf{p}_{\mathbf{x},\mathbf{y}|\mathbf{e}_1,\mathbf{e}_2}$ is defined as

$$\mathsf{p}_{\mathbf{x},\mathbf{y}|\mathbf{e}_1,\mathbf{e}_2}(\boldsymbol{x},\boldsymbol{y} \mid \boldsymbol{e}_1,\boldsymbol{e}_2) = \frac{\mathsf{p}_{\mathbf{x},\boldsymbol{y},\mathbf{e}_1,\mathbf{e}_2}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{e}_1,\boldsymbol{e}_2)}{\mathsf{p}_{\mathbf{e}_1,\mathbf{e}_2}(\boldsymbol{e}_1,\boldsymbol{e}_2)}, \tag{1.62}$$

with $\mathsf{p}_{\mathbf{e}_1,\mathbf{e}_2}$ defined in terms of $\mathsf{p}_{\mathbf{x},\mathbf{y},\mathbf{e}_1,\mathbf{e}_2}$ by marginalising out $\mathbf{x}$ and $\mathbf{y}$

$$\mathsf{p}_{\mathbf{e}_1,\mathbf{e}_2}(\boldsymbol{e}_1,\boldsymbol{e}_2) = \iint \mathsf{p}_{\mathbf{x},\mathbf{y},\mathbf{e}_1,\mathbf{e}_2}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{e}_1,\boldsymbol{e}_2)\,\mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{y}. \tag{1.63}$$

We can express $\mathsf{p}_{\mathbf{x},\boldsymbol{y},\mathbf{e}_1,\mathbf{e}_2}$ itself as a marginal of the joint density over all random variables in the model (which we can read off as a product of factors from Figure 1.9) by marginalising out $\mathbf{m}$, $\mathbf{t}$, $\mathbf{e}_1^\star$ and $\mathbf{e}_2^\star$

$$\begin{aligned}\mathsf{p}_{\mathbf{x},\mathbf{y},\mathbf{e}_1,\mathbf{e}_2}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{e}_1,\boldsymbol{e}_2) = \iiiint\; & \mathsf{p}_{\mathbf{x}}(\boldsymbol{x})\,\mathsf{p}_{\mathbf{y}}(\boldsymbol{y})\,\mathsf{p}_{\mathbf{m}}(\boldsymbol{m})\,\mathsf{p}_{\mathbf{t}}(\boldsymbol{t}) \\ & \mathsf{p}_{\mathbf{e}_1|\mathbf{e}_1^\star,\mathbf{x},\mathbf{y},\mathbf{m},\mathbf{t}}(\boldsymbol{e}_1 \mid \boldsymbol{e}_1^\star,\boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t}) \\ & \mathsf{p}_{\mathbf{e}_2|\mathbf{e}_2^\star,\mathbf{x},\mathbf{y},\mathbf{m},\mathbf{t}}(\boldsymbol{e}_2 \mid \boldsymbol{e}_2^\star,\boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t}) \\ & \mathsf{p}_{\mathbf{e}_1^\star}(\boldsymbol{e}_1^\star)\,\mathsf{p}_{\mathbf{e}_2^\star}(\boldsymbol{e}_2^\star)\,\mathrm{d}\boldsymbol{e}_1^\star\,\mathrm{d}\boldsymbol{e}_2^\star\,\mathrm{d}\boldsymbol{t}\,\mathrm{d}\boldsymbol{m}.\end{aligned} \tag{1.64}$$

The integration with respect to $\boldsymbol{e}_1^\star$ and $\boldsymbol{e}_2^\star$ can be performed analytically as the conditional density factors $\mathsf{p}_{\mathbf{e}_2|\mathbf{e}_2^\star,\mathbf{x},\mathbf{y},\mathbf{m},\mathbf{t}}$ and $\mathsf{p}_{\mathbf{e}_2|\mathbf{e}_1^\star,\mathbf{x},\mathbf{y},\mathbf{m},\mathbf{t}}$ correspond to deterministic factors which depend linearly on $\mathbf{e}_1^\star$ and $\mathbf{e}_2^\star$. Expressing the deterministic factors as Dirac deltas and denoting the functions corresponding to the computational graphs in Figure 1.9 mapping from $\{\mathsf{x}_h, \mathsf{y}_h, \mathsf{m}_h, \mathsf{t}_h\}_{h=1}^H$ to $\{\mathsf{d}_{1,g}\}_{g=1}^G$ and $\{\mathsf{d}_{2,g}\}_{g=1}^G$, $\boldsymbol{d}_1$ and $\boldsymbol{d}_2$ respectively, we have that

$$\begin{aligned}&\mathsf{p}_{\mathbf{e}_1|\mathbf{x},\mathbf{y},\mathbf{m},\mathbf{t}}(\boldsymbol{e}_1 \mid \boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t}) \\ &\quad = \int \mathsf{p}_{\mathbf{e}_1|\mathbf{e}_1^\star,\mathbf{x},\mathbf{y},\mathbf{m},\mathbf{t}}(\boldsymbol{e}_1 \mid \boldsymbol{e}_1^\star,\boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t})\,\mathsf{p}_{\mathbf{e}_1^\star}(\boldsymbol{e}_1^\star)\,\mathrm{d}\boldsymbol{e}_1^\star \\ &\quad = \int \delta\Big(\boldsymbol{e}_1 - \boldsymbol{e}_1^\star - \boldsymbol{d}_1(\boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t})\Big)\,\mathcal{N}\Big(\boldsymbol{e}_1^\star \mid \mathbf{0}, \sigma^2\mathbf{I}\Big)\,\mathrm{d}\boldsymbol{e}_1^\star \\ &\quad = \mathcal{N}\Big(\boldsymbol{e}_1 \mid \boldsymbol{d}_1(\boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t}), \sigma^2\mathbf{I}\Big),\end{aligned} \tag{1.65}$$

and likewise for $\mathsf{p}_{\mathbf{e}_2|\mathbf{x},\mathbf{y},\mathbf{m},\mathbf{t}}$

$$\mathsf{p}_{\mathbf{e}_2|\mathbf{x},\mathbf{y},\mathbf{m},\mathbf{t}}(\boldsymbol{e}_2 \mid \boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t}) = \mathcal{N}\Big(\boldsymbol{e}_2 \mid \boldsymbol{d}_2(\boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t}), \sigma^2\mathbf{I}\Big). \tag{1.66}$$

Rewriting (1.64) in terms of (1.65) and (1.66) and substituting the definitions for the factors from Figure 1.9 we have that

$$\begin{aligned}
\mathrm{p}_{\mathbf{x},\mathbf{y},\mathbf{e}_1,\mathbf{e}_2}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{e}_1,\boldsymbol{e}_2) &= \iint \mathrm{p}_{\mathbf{x},\mathbf{y},\mathbf{m},\mathbf{t},\mathbf{e}_1,\mathbf{e}_2}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t},\boldsymbol{e}_1,\boldsymbol{e}_2)\,\mathrm{d}\boldsymbol{t}\,\mathrm{d}\boldsymbol{m} \\
&= \iint \prod_{h=1}^{H} \left(\mathcal{U}(x_h \mid 0, x_{\max})\,\mathcal{U}(y_h \mid 0, y_{\max})\right) \\
&\qquad \mathcal{N}\left(\boldsymbol{e}_1 \mid \boldsymbol{d}_1(\boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t}), \sigma^2\mathbf{I}\right)\mathcal{N}\left(\boldsymbol{e}_2 \mid \boldsymbol{d}_2(\boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t}), \sigma^2\mathbf{I}\right) \\
&\qquad \mathrm{LogNorm}\left(\boldsymbol{m} \mid \mu_m\mathbf{1}, \sigma_m^2\mathbf{I}\right)\mathrm{LogNorm}\left(\boldsymbol{t} \mid \mu_m\mathbf{1}, \sigma_m^2\mathbf{I}\right)\mathrm{d}\boldsymbol{t}\,\mathrm{d}\boldsymbol{m}.
\end{aligned} \tag{1.67}$$

We cannot simplify this integral any further analytically. Thefore the conditional expectation we wish to compute in terms of integrals of functions we can evaluate pointwise is

$$\mathbb{E}\left[f(\mathbf{x},\mathbf{y}) \mid \mathbf{e}_1 = \boldsymbol{e}_1, \mathbf{e}_2 = \boldsymbol{e}_2\right] = \frac{\iiiint f(\boldsymbol{x},\boldsymbol{y})\,\mathrm{p}_{\mathbf{x},\mathbf{y},\mathbf{m},\mathbf{t},\mathbf{e}_1,\mathbf{e}_2}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t},\boldsymbol{e}_1,\boldsymbol{e}_2)\,\mathrm{d}\boldsymbol{t}\,\mathrm{d}\boldsymbol{m}\,\mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{y}}{\iiiint \mathrm{p}_{\mathbf{x},\mathbf{y},\mathbf{m},\mathbf{t},\mathbf{e}_1,\mathbf{e}_2}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t},\boldsymbol{e}_1,\boldsymbol{e}_2)\,\mathrm{d}\boldsymbol{t}\,\mathrm{d}\boldsymbol{m}\,\mathrm{d}\boldsymbol{x}\,\mathrm{d}\boldsymbol{y}}. \tag{1.68}$$

Each of the vector integration variables in (1.68) is of dimension H and so the overall dimension of the space being integrated over in both the numerator and denominator is $4H$. In the *Observing Dark Worlds* data the number of halos per cluster image H is between one and three so to evaluate conditional expectations of the halo positions we need to compute integrals over spaces with four, eight or twelve dimensions.

For four or eight dimensions it may be feasible to use quadrature methods [9], which involve evaluating the integrand across a fixed grid of points and then computing a weighted sum of these values, to numerically approximate the integrals to reasonable accuracy. For a fixed grid resolution however the cost of quadrature scales exponentially with the dimension of the space being integrated over - if N points are used per dimension, using quadrature to evaluate (1.68) will involve N^{4H} evaluations of the integrand.

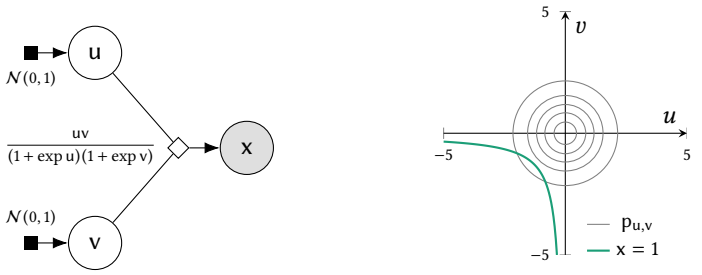
Assuming the computational cost of the function $f(\mathbf{x},\mathbf{y})$ the conditional expectation is being calculated of is negligible, the dominant cost in the evaluation of the integrands in (1.68) will be in evaluating $\boldsymbol{d}_1(\boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t})$ and $\boldsymbol{d}_2(\boldsymbol{x},\boldsymbol{y},\boldsymbol{m},\boldsymbol{t})$ which involve computing the distances and angles between all galaxy-halo pairs. Using a very simplistic (and

conservative) assumption that each arithmetic, square root, comparison and trigonometric operation has a unit floating point operation cost, the computation graph in Figure 1.9 which is present on the overlap between the two plates will involve 15 floating point operations in a single forward pass and is evaluated GH times for an overall cost per integrand evaluation of $15GH$ floating point operations.

A rough but conservative lower bound on the floating point operation count of evaluating the integrals in (1.68) using quadrature is therefore $15GHN^{4H}$. The number of galaxies G per cluster image in the *Observing Dark Worlds* data varies between 300 and 740. If we assume $G = 500$ and that we use only $N = 20$ grid points per dimension, for the $H = 2$ cases evaluating the conditional expectation will involve $\sim 1 \times 10^9$ floating point operations, for $H = 2$, $\sim 4 \times 10^{14}$ floating point operations and for $H = 3$, $\sim 9 \times 10^{19}$ floating point operations.

At the time of writing, the theoretical peak floating point operation performance of a top-end multi-core server Central Processing Unit (CPU) is around 5×10^{11} FLoating point OPerations per Second (FLOPS). Assuming this peak performance could be obtained when using quadrature to approximate (1.68) , our back-of-the-envelope calculation suggests a trivial two millisecond compute time for clusters with one halo, a more noticeable thirteen minute computation for clusters with two halos, and an impractical six year wait for clusters with three halos.

Current top-end Graphics Processing Units (GPUs) have a peak theoretical (single-precision) floating point performance of around 10^{13} FLOPS which at best would cut the computation time for a $H = 3$ case by a factor of 20 to around 100 days. With a cluster of CPU or GPU nodes, or faster individual nodes due to future growth in processing power, the compute time could potentially be brought down further to more reasonable timescales. However the key point in this example is the exponential growth with dimension - even moving from eight to twelve dimensions made compute time impractical. Clearly therefore approximating the integrals in expectations like (1.68) using quadrature methods is not viable when performing inference in models with large numbers of unobserved variables (with the example here showing that even integrals over what might initially seem a low dimensionality of twelve can be problematic).



(a) Directed factor graph defining model with two latent variables (u, v), and an observed variable x.

(b) Plot of density on latent variables (contours) and set of values for which x = 1 (green curve).

Figure 1.10: Simple example of a probabilistic model where the observed variable is a non-bijective function of two latent variables.

In the *Observing Dark Worlds* example it was possible to analytically integrate out the random vectors $\mathbf{e}_1^\star$ and $\mathbf{e}_2^\star$, which correspond to the intrinsic galaxy ellipticities, due to the observed ellipticities $\mathbf{e}_1$ and $\mathbf{e}_2$ being deterministic linear functions of $\mathbf{e}_1^\star$ and $\mathbf{e}_2^\star$ respectively (for fixed $\mathbf{x}$, $\mathbf{y}$, $\mathbf{m}$ and $\mathbf{t}$). Rather than calculating the conditional densities on $\mathbf{e}_1$ and $\mathbf{e}_2$ by integrating out $\mathbf{e}_1^\star$ and $\mathbf{e}_2^\star$, we could equivalently have applied the change of variables formula (1.28) for a bijective transformation; in this case the Jacobian determinant is simply one.

If the observed variables had instead been the output of a deterministic factor where there was no bijective dependence on any of the parent variables (inputs to the deterministic factor), the simplified form of the change of variables formula (1.28) would no longer have applied and instead the more general form (1.26) would have been required. Generally in such cases it will not be possible to analytically marginalise out a parent variable to give a well-defined conditional density on the observed variable(s).

An illustration of such a case for a simple three variable model is shown in Figure 1.10. Here the observed variable x is a deterministic function of two latent (unobserved) variables u and v. There is no analytic solution in terms of elementary functions for u as a function of x and v or for v as a function of x and u. This means the Dirac delta term corresponding to the deterministic factor cannot be integrated out. Due to the presence of the Dirac delta the joint density $\mathrm{p}_{\mathrm{x},\mathrm{u},\mathrm{v}}$ is not well defined (the joint probability $\mathrm{P}_{\mathrm{x},\mathrm{u},\mathrm{m}}$ is not absolutely continuous with respect to any measure) which complicates evaluations of conditional expectations such as $\mathbb{E}\left[f(\mathrm{u},\mathrm{v}) \,|\, \mathrm{x} = 1\right]$.

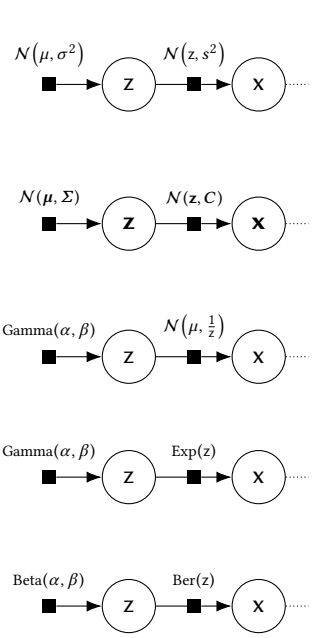
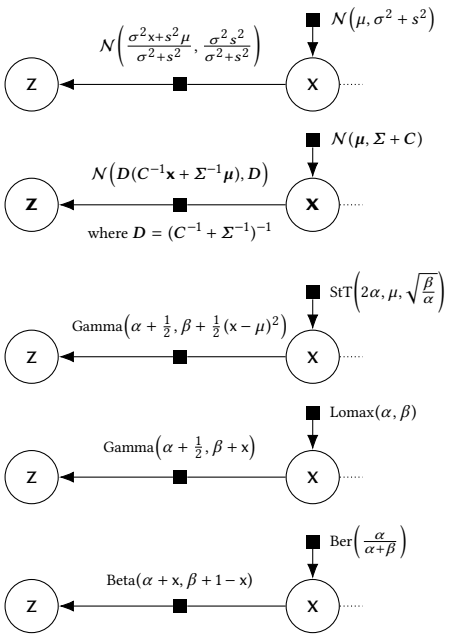
Original factorisation	Conjugate factorisation
	

Table 1.4: Factor graph illustrations of conjugate distributions.

In particular the set of u and v values corresponding to solutions to $x = 1$ (illustrated as the green curve in Figure 1.1a) has zero Lebesgue measure. Therefore even though the dimensionality is low in this case we can not use simple quadrature to evaluate conditional expectations without some further form of approximation. We will revisit methods for performing inference in such models later in the thesis.

Some densities have a *conjugacy* property that can simplify inference. If x and z are two random variables in a model then the joint density on the two variable can be factorised as

$$p_{x,z}(x, z) = p_{x|z}(x \mid z)p_z(z) = p_{z|x}(z \mid x)p_x(x). \tag{1.69}$$

A conditional density $p_{u|v}$ is from the exponential family if it can be written as $p_{u|v}(\boldsymbol{u} \mid \boldsymbol{v}) = \frac{h(\boldsymbol{u}) \exp(\boldsymbol{\eta}(\boldsymbol{v})^\top \boldsymbol{t}(\boldsymbol{u}))}{z(\boldsymbol{v})}$, with $\boldsymbol{\eta}(\boldsymbol{v})$ termed the natural parameters *and $\boldsymbol{t}(\boldsymbol{u})$ termed the* sufficient statistics.

For certain pairs of $p_{x|z}$ and p_z the corresponding $p_{z|x}$ and p_x have closed-form expressions. Some examples are shown in Table 1.4. In all of these examples the conditional density $p_{x|z}$ is a density for an *exponential family distribution*. For every exponential family distribution density $p_{x|z}$ there exists a density p_z such that $p_{z|x}$ is of the same parametric family as p_z.

If x corresponds to an observed variable in the model, then if evaluating conditional expectations $\mathbb{E}[f(z) \mid x]$ the conjugacy property means

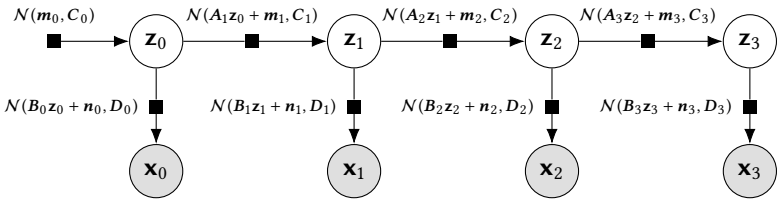


Figure 1.11: Factor graph for a latent linear dynamical system model.

that conditional density $p_{z|x}$ which f should be integrated against has a closed form expression. Often for simple f, e.g. $f(z) = z$ or $f(z) = z^2$, the conditional expectations will have closed form solutions in such cases (i.e. corresponding to moments of the distribution defined by $p_{z|x}$). Even when f is more complex, typically generating independent random samples from $p_{z|x}$ in such cases will be possible, simplifying the use of Monte Carlo methods (which will be introduced in the next chapter). If both z and x are latent variables then the conjugacy property is also useful as in this case z can be analytically marginalised out of conditional expectations of functions which do not depend on z.

A Markov process is a stochastic process such that future states are conditionally independent of past states given the current state.

Various algorithms have been developed to exploit conjugacy in restricted classes of probabilistic models to perform efficient exact inference. Inference in *latent linear dynamical systems*, an example of which is shown as a factor graph in Figure 1.11, can be efficiently performed with the recursive Kalman filtering and smoothing algorithms [18]. Closely related are the forward and backward updates for inference in *hidden Markov models* [31] which have the same factorisation structure as latent linear dynamical systems but use a discrete rather than real-valued latent state. The *junction tree algorithm* [21] is a general purpose algorithm for performing exact inference in undirected graphical models of discrete random variables which exploits conditional independency structure to efficiently decompose the inference into local computations; the computational cost of the algorithm scales exponentially with the number of nodes in the largest clique in the model and so is mainly relevant for graphs with relatively small maximal cliques.

1.3.4 A note on Bayesian terminology

The system of inference that we have described would often be identified as being *Bayesian*. This name arises because of the central importance of *Bayes' theorem* (1.7) in defining the conditional probability of unobserved variables given observations.

While often the inference problems we will discuss in this thesis will be motivated from a Bayesian standpoint, the use of probabilistic modelling and resulting need to deal with the computational challenges of computing integrals in high dimensional spaces are not unique to Bayesian statistics. For instance many of the approximate inference methods we will discuss in the next two chapters were originally developed for use in studying statistical physics problems such as the phase transitions of the *Ising spin model* (which we earlier encounted under the alternative name of the Boltzmann machine model). As our main interest in this thesis will be the computational aspects of inference rather than specific applications, we will therefore generally prefer to refer to probabilistic modelling and inference in general terms rather than Bayesian inference in particular.

For completeness we will briefly review the nomenclature typically used in Bayesian inference. For a model with observed variables $\mathbf{x} \in X$ and unobserved variables $\mathbf{z} \in Z$, often referred to as *model parameters* in a Bayesian setting, the standard presentation of Bayesian inference assumes that the joint probability is specified by a density $\mathrm{p}_{\mathrm{x,z}} = \frac{\mathrm{dP}_{\mathbf{x},\mathbf{z}}}{\mathrm{d}(\mu_{\mathbf{x}} \times \mu_{\mathbf{z}})}$ which naturally factorises as $\mathrm{p}_{\mathbf{x},\mathbf{z}}(\boldsymbol{x}, z) = \mathrm{p}_{\mathbf{x}|\mathbf{z}}(\boldsymbol{x} \mid z)\, \mathrm{p}_{\mathbf{z}}(z)$.

In this case the probability distribution defined by the marginal density on the model parameters $\mathrm{p}_{\mathbf{z}}$ is typically termed the *prior distribution*, with the intepretation that it captures our beliefs about the parameters before observing data. The conditional density on the observed variables given parameters $\mathrm{p}_{\mathbf{x}|\mathbf{z}}$ is often referred to as the *likelihood*.

The distribution defined by the conditional density on the parameters given the observations $\mathrm{p}_{\mathbf{z}|\mathbf{x}}$ is termed the *posterior distribution*, with this naming indicative of it representing our beliefs about the parameters after observing data. The posterior density can be calculated from the prior and likelihood using the definition for conditional density given in (1.42). Analagously to the definition for probabilities in (1.7) the resulting relationship is often also termed Bayes' theorem

$$\mathrm{p}_{\mathbf{z}|\mathbf{x}}(z \mid \boldsymbol{x}) = \frac{\mathrm{p}_{\mathbf{x}|\mathbf{z}}(\boldsymbol{x} \mid z)\, \mathrm{p}_{\mathbf{z}}(z)}{\mathrm{p}_{\mathbf{x}}(\boldsymbol{x})} \quad \forall z \in Z,\, \boldsymbol{x} \in X : \mathrm{p}_{\mathbf{x}}(\boldsymbol{x}) > 0. \tag{1.70}$$

Inference is then typically posed as the problem of computing the posterior distribution. This is typically not available in a closed form as

evaluating the denominator in (1.70), $\mathrm{p}_{\mathbf{x}}$, sometimes termed the *model evidence*, requires integrating the joint density over the parameters

$$\mathrm{p}_{\mathbf{x}}(\boldsymbol{x}) = \int_Z \mathrm{p}_{\mathbf{x},\mathbf{z}}(\boldsymbol{x}, z)\,\mathrm{d}z = \int_Z \mathrm{p}_{\mathbf{x}|\mathbf{z}}(\boldsymbol{x} \mid z)\,\mathrm{p}_{\mathbf{z}}(z)\,\mathrm{d}z. \tag{1.71}$$

Other than in special cases such as the conjugate pairs of densites shown in Table 1.4, this integral does not have a closed form solution and typically Z will be of a dimensionality which means numerical quadrature will be too expensive. As $\mathrm{p}_{\mathrm{x}}(\boldsymbol{x})$ is found by marginalising out $\mathbf{z}$ from the product of the prior and likelihood, it is also sometimes termed the *marginal likelihood*, though this name obscures that it is also dependent on the prior.

The usage of the term likelihood in Bayesian inference is somewhat overloaded: while it is often informally used to refer to the conditional density in $\mathrm{p}_{\mathbf{x}|\mathbf{z}}$ in Bayes' theorem (which is often summarised as *posterior is proportional to likelihood times prior*), the likelihood is usually formally defined as a function of the parameters (given fixed values for the observed variables) with notation such as $\ell(z \mid \boldsymbol{x}) = \mathrm{p}_{\mathbf{x}|\mathbf{z}}(\boldsymbol{x} \mid z)$ sometimes used to emphasise this. This usage is particularly common when discussing *maximum likelihood* methods which find values of the parameters which maximise the likelihood (given data). In this latter interpretation it makes sense to refer to the likelihood of the parameters, but not the likelihood of observed variables (or observations / data). This leads to recommendations to refer to 'the likelihood of the parameters given the [observed] data' [22], which given the construct being discussed is actually a density on the observed data given parameter values is, in our opinion, not particularly clear.

In this thesis we will generally avoid using the terms *likelihood* and *marginal likelihood*. We will instead prefer to refer to individual conditional and marginal densities (or probabilities) explicitly. We will also prefer to refer to unobserved or latent variables (which we will use interchangeably) rather than model parameters when these values are being inferred (as opposed to set to fixed values). Parameters and latent variables are sometimes used distinctly (e.g. defining parameters as unobserved variables which are global and of a fixed number and latent variables as unobserved variables that are linked with individual observed variables and increase in number with the number of observations), however from the computational perspective of inference both

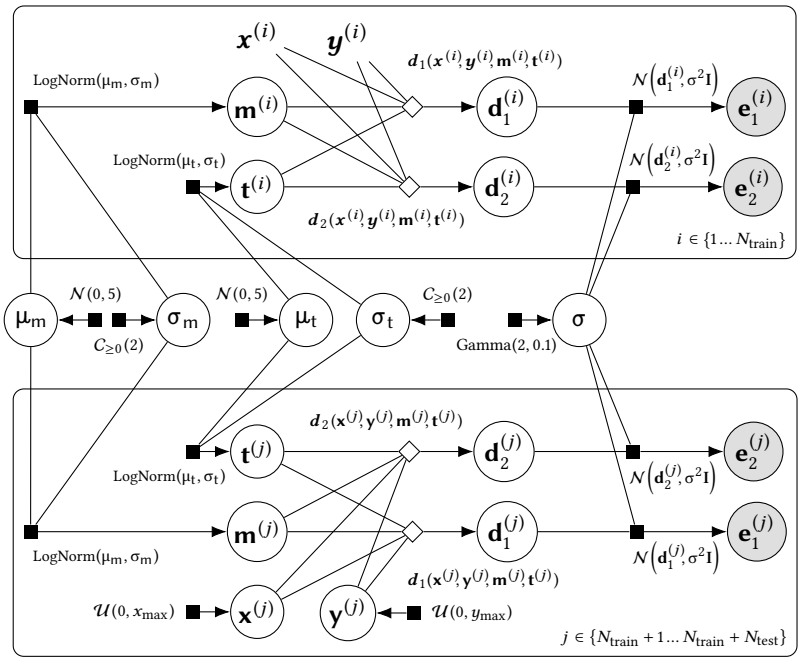


Figure 1.12: Observing Dark Worlds hierarchical model.

are equivalent - they are variables in the model which we assume we are uncertain about the value of as we do not observe their values directly. Although this point may seem minor, as we will see later a key idea of this thesis is that treating all unobserved variables in a probabilistic model jointly rather than differentially treating a subset considered as parameters can be of signficant computational benefit.

1.3.5 Hierarchical modelling

When introducing the proposed model (Figure 1.9) for the *Observing Dark Worlds* problem we glossed over how the parameters σ, μ_m, σ_m, μ_t and σ_t were chosen, assuming for simplicity they had somehow been set to reasonable values. Rather than fixing the parameter values, a more Bayesian approach would be instead to also treat these parameters as further unobserved variables to infer.

In particular we can encode our beliefs about plausible values for σ, μ_m, σ_m, μ_t and σ_t by setting prior distributions on each of the variables and then jointly conditioning on *all* of the observed data together. For example, for the the prior on the intrinsic ellipticity scale variable σ we might use Gamma$(2, 0.1)$. This has support only for positive values as required for a scale variable, and reflects the known properties of the intrinsic ellipticities - that they have non-zero variability (with the

Gamma density tending to zero as $\sigma \to 0$) and that they are bounded to unit magnitude and hence the standard deviation would be expected to be significantly less than one. We might have less strong prior beliefs about the range of plausible values for the halo mass and core radii distribution, and so choose to use concordantly less constraining prior distributions on the scale and location variables for these distributions. For example a prior of $\mathcal{N}(0, 5)$ on the location variables μ_m and μ_t and a prior of $C_{\geq 0}(2)$ on the scale variables σ_m and σ_t, supports a wide range of values for these variables as being reasonable while still making a weak assumption that very extreme values are implausible.

In the *Observing Dark Worlds* competition, two distinct sets of data were provided - a training set of $N_{\text{train}} = 300$ of cluster images in which the observed ellipticities, galaxy coordinates and halo positions were all provided; and a test set of $N_{\text{test}} = 120$ images where only the galaxy observed ellipticities and coordinates are provided. Assuming both the training and test data are iid we can form a *hierarchical model* for the problem which specifies a joint density across the observed and latent variables for all of the training and test set clusters as well as the global variables σ, μ_m, σ_m, μ_t and σ_t. A factor graph for the proposed hierarchical model is shown in Figure 1.12. Compared to the factor graph in Figure 1.9 some detail in the model structure for each cluster has been hidden to make the higher level structure clearer. The vector notation introduced when discussing inference in the model has also been used to remove the need for plates indexed across galaxies and halos.

When making predictions using the hierarchical model, the latent variables associated with each cluster image are now dependent when not conditioning on the values of the variables σ, μ_m, σ_m, μ_t and σ_t. It is therefore necessary to calculate conditional expectations on all of the test cluster halo coordinates jointly given the observed data. Even considering only the halo position variables the expectations are being calculated over, the resulting integral is over a space of several hundred dimensions. Once all of the additional unobserved variables defined in the overall joint density are included, the resulting integral is over several thousand variables. Given the discussed infeasibility of using quadrature to compute conditional expectations of the halo positions for even a single cluster images, clearly in this case the need for computational methods for inference which scale to high dimensionalities is even more stark.

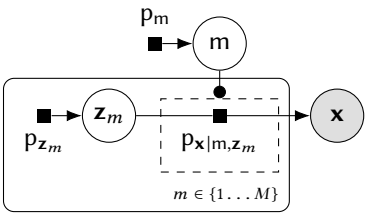


Figure 1.13: Factor graph for inference over multiple models.

1.3.6 Model comparison

So far we have discussed making inferences about the unobserved variables in a single fixed model. An important second level of inference is comparing between competing models for the same observed data. As we will see this can be treated consistently within a simple extension to the existing probabilistic framework we have discussed.

As a simple motivating example, we can consider comparing our proposed model for the *Observing Dark Worlds* problem, in which we assumed that the magnitude of the shear distortion had the functional dependence given in (1.52), to an alternative model which assumes

$$f_h(u, v) = \frac{m_h}{r_h} \quad \text{where} \quad r_h = \sqrt{(x_h - u)^2 + (y_h - v)^2}. \tag{1.72}$$

Ockham's Razor is a philoshopical principle, commonly attributed to the 14th century Franciscan friar William of Ockham, that states if there exist multiple explanations for observations, all else being equal we should prefer the simplest.

This model is simpler in the sense of requiring only one additional latent variable per halo, but produces a non-realistic infinite magnitude distortion at zero radial distances. Given the observed data, we would ideally like to be able to make a judgement as to which of the two proposed models better describes the data. To be useful this comparison must take into account the relative complexity of the models; a model with more free variables will generally be able to fit observed data more closely, however *Ockham's Razor*, and empricical evidence of the loss of predictive power of overly complex models, suggests we should prefer simpler models where possible. By marginalising over the unobserved variables in a model, the probabilistic model comparison framework we will describe automatically embodies Ockham's Razor [22].

The general set up will be that we have a finite set of M models which we index with the variable $m \in \{1 \dots M\}$. A common special case will be where $M = 2$ and we are performing a comparison between a pair of models. All models share the same observed random vector $\mathbf{x}$, and there are a set of per model vectors of unobserved variables $\{\mathbf{z}_m\}_{m=1}^{M}$

which are assumed to be independent (before conditioning on any observations). More complex structures could be assumed e.g. that the models share a set of common unobserved variables, however we only consider the simple case of independent models here. The joint density across the observations, model indicator and model latent variables is then assumed to factorise as

$$\begin{aligned} &\mathrm{p}_{\mathbf{x},\mathsf{m},\mathbf{z}_1,\ldots,\mathbf{z}_M}(\boldsymbol{x}, m, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_M) = \\ &\qquad \mathrm{p}_{\mathbf{x}|\mathsf{m},\mathbf{z}_m}(\boldsymbol{x} \mid m, \boldsymbol{z}_m)\, \mathrm{p}_{\mathsf{m}}(m) \prod_{n=1}^{M} \mathrm{p}_{\mathbf{z}_n}(\boldsymbol{z}_n). \end{aligned} \tag{1.73}$$

The marginal density on the model indicator p_{m} represents our prior beliefs about the relative probabilities of the models before observing data. Importantly the value of the model indicator variable m selects the relevant per model conditional density on the observed variables given latent variables $\mathrm{p}_{\mathbf{x}|\mathsf{m},\mathbf{z}_m}$; this represents the assumption that conditioned on the model indicator assuming a particular model index m the observed variables are conditionally independent of the latent variables of all other models $\mathbf{x} \perp \{\mathbf{z}_n\}_{n \neq m} \mid \mathsf{m} = m$. An extension to factor graphs, *gates* [25], can be used to represent such context-dependent conditional independency relationships. Figure 1.13 shows a gated factor graph representation of equation 1.73, with the gate indicated by the dashed box.

Given this computational set up, the task in model comparison is then to compute the relative probabilities of each of the models given observed data. These probabilities[6] are given by

$$\mathrm{p}_{\mathsf{m}|\mathbf{x}}(m \mid \boldsymbol{x}) = \frac{\mathrm{p}_{\mathbf{x}|\mathsf{m}}(\boldsymbol{x} \mid m)\, \mathrm{p}_{\mathsf{m}}(m)}{\sum_{n=1}^{M} (\mathrm{p}_{\mathbf{x}|\mathsf{m}}(\boldsymbol{x} \mid n)\, \mathrm{p}_{\mathsf{m}}(n))}, \tag{1.74}$$

which can be seen as a direct analogue to Bayes' theorem for the posterior density on unobserved random variables for a single model. The key quantities needed to evaluate the model posterior probabilities are the marginal densities $\mathrm{p}_{\mathbf{x}|\mathsf{m}}(\boldsymbol{x} \mid m)$ evaluated at the observed data. Computing these values requires marginalising out the unobserved variables from the per model joint densities on the observed and unobserved variables

$$\mathrm{p}_{\mathbf{x}|\mathsf{m}}(\boldsymbol{x} \mid m) = \int_{Z_m} \mathrm{p}_{\mathbf{x}|\mathsf{m},\mathbf{z}_m}(\boldsymbol{x} \mid m, z) \mathrm{p}_{\mathbf{z}_m}(z)\, \mathrm{d}z. \tag{1.75}$$

6 As m is a discrete random variable the probability of the event of m taking a value in the singleton set $\{m\}$ given that $\mathbf{x} = \boldsymbol{x}$ is equal to the density $\mathrm{p}_{\mathsf{m}|\mathbf{x}}(m \mid \boldsymbol{x})$.

This value is equal to the denominator in Bayes' theorem (1.70), this explaining the naming of this term as the *model evidence*.

As with evaluating conditional expectations of functions of the latent variables in a model given observations, evaluating the model evidence values requires integrating across the space of all latent variables. In most cases this integral will not have an analytic solution and the number of latent variables will be sufficiently large to make numerical quadrature methods impractical. The key computational challenge in being able to perform probabilistic model comparison with complex high dimensional models is therefore again being able to efficiently to compute integrals in high dimensional spaces.

1.4 SUMMARY

In this chapter we introduced the probabilistic modelling theory and tools that will form the basis for the methods we will introduce in the rest of the thesis. We illustrated the ability of factor graphs to efficiently communicate the structure of complex probabilistic models and discussed their close links to computation graphs which allow efficient calculation of the derivatives of the outputs of complex functions of a large numbers of variables with respect to their inputs. We concluded by motivating the computational challenges of performing inference in complex probabilistic models involving large numbers of unobserved variables, due to the need to be able to evaluate integrals across high-dimensional spaces. In the next chapter we will introduce some of the computational methods which have been proposed to address the challenges of finding solutions to inferences problems.

BIBLIOGRAPHY

[1] Matthias Bartelmann and Peter Schneider. 'Weak gravitational lensing'. In: *Physics Reports* 340.4 (2001), pp. 291–472.

[2] Friedrich L Bauer. 'Computational graphs and rounding error'. In: *SIAM Journal on Numerical Analysis* 11.1 (1974), pp. 87–96.

[3] Atilim Gunes Baydin, Barak A Pearlmutter, Alexey Andreyevich Radul and Jeffrey Mark Siskind. 'Automatic differentiation in machine learning: a survey'. In: *arXiv preprint arXiv:1502.05767* (2015).

[4] L. M. Beda, L. N. Korolev, N. V. Sukkikh and T. S. Frolova. *Programs for automatic differentiation for the machine BESM*. Technical Report. (In Russian). Moscow, USSR: Institute for Precise Mechanics and Computation Techniques, Academy of Science, 1959.

[5] George EP Box. 'Sampling and Bayes' inference in scientific modelling and robustness'. In: *Journal of the Royal Statistical Society. Series A (General)* (1980), pp. 383–430.

[6] Wray L Buntine. 'Operations for learning with graphical models'. In: *Journal of artificial intelligence research* (1994).

[7] Richard T Cox. 'Probability, frequency and reasonable expectation'. In: *American Journal of Physics* 14.1 (1946), pp. 1–13. URL: http://dx.doi.org/10.1119/1.1990764.

[8] Richard T Cox. 'The algebra of probable inference'. In: *American Journal of Physics* 31.1 (1963), pp. 66–67. URL: http://dx.doi.org/10.1119/1.1969248.

[9] Philip J. Davis and Philip Rabinowitz. *Numerical Integration*. Blaisdell Publishing Company, 1967.

[10] Herbert Federer. *Geometric measure theory*. Springer, 1969.

[11] Bruno de Finetti. 'Foresight: its logical laws, its subjective sources'. In: *Studies in Subjective Probability*. Ed. by H. E. Kyburg. English translation of original 1937 French article *La Prévision: ses lois logiques, ses sources subjectives*. Springer, 1992, pp. 134–174.

[12] Brendan J Frey. 'Extending factor graphs so as to unify directed and undirected graphical models'. In: *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc. 2002, pp. 257–264. URL: https://arxiv.org/abs/1212.2486.

[13] Brendan J Frey, Frank R Kschischang, Hans-Andrea Loeliger and Niclas Wiberg. 'Factor graphs and algorithms'. In: *Proceedings of the 35th Annual Allerton Conference on Communication Control and Computing*. 1997.

[14] Andrew Gelman and Cosma Rohilla Shalizi. 'Philosophy and the practice of Bayesian statistics'. In: *British Journal of Mathematical and Statistical Psychology* 66.1 (2013), pp. 8–38.

[15] David Harvey, Thomas D Kitching, Joyce Noah-Vanhoucke, Ben Hamner, Tim Salimans and AM Pires. 'Observing Dark Worlds: A crowdsourcing experiment for dark matter mapping'. In: *Astronomy and Computing* 5 (2014), pp. 35–44.

[16] Eric Jullo, Jean-Paul Kneib, Marceau Limousin, Ardis Eliasdottir, PJ Marshall and Tomas Verdugo. 'A Bayesian approach to strong lensing modelling of galaxy clusters'. In: *New Journal of Physics* 9.12 (2007), p. 447. URL: https://arxiv.org/abs/0706.0048.

[17] Kaggle. *Observing dark worlds*. https://www.kaggle.com/c/DarkWorlds. 2012.

[18] Rudolph Emil Kalman. 'A new approach to linear filtering and prediction problems'. In: *Journal of Basic Engineering* 82.1 (1960), pp. 35–45.

[19] Ross Kindermann and Laurie Snell. *Markov random fields and their applications*. American Mathematical Society, 1980.

[20] Andreĭ Nikolaevich Kolmogorov. *Foundations of the Theory of Probability*. Ed. by Nathan Morrison. 2nd English Edition. English translation of original 1933 German monograph, *Grundbegriffe der Wahrscheinlichkeitrechnung*. Chelsea Publishing Company, 1956. URL: https://pdfs.semanticscholar.org/c3e1/51f71168a5f348bdebfde11752ca603fa6d0.pdf.

[21] Steffen L Lauritzen and David J Spiegelhalter. 'Local computations with probabilities on graphical structures and their application to expert systems'. In: *Journal of the Royal Statistical Society. Series B (Methodological)* (1988), pp. 157–224. DOI: 10.2307/2345762.

[22] David JC MacKay. *Information theory, inference and learning algorithms*. Cambridge University Press, 2003.

[23] Phillip James Marshall, Michael Paul Hobson and Anže Slosar. 'Bayesian joint analysis of cluster weak lensing and Sunyaev–Zel'dovich effect data'. In: *Monthly Notices of the Royal Astronomical Society* 346.2 (2003), pp. 489–500.

[24] Richard Massey, Thomas Kitching and Johan Richard. 'The dark matter of gravitational lensing'. In: *Reports on Progress in Physics* 73.8 (2010), p. 086901.

[25] Tom Minka and John Winn. 'Gates: a graphical notation for mixture models'. In: *Advances in Neural Information Processing Systems*. 2009, pp. 1073–1080.

[26] Iain Murray. *A Bayesian approach to Observing Dark Worlds*. http://homepages.inf.ed.ac.uk/imurray2/pub/. 2012.

[27] John F Nolan. 'Analytical differentiation on a digital computer'. PhD thesis. Massachusetts Institute of Technology, 1953.

[28] Judea Pearl. *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Morgan Kaufmann, 1988.

[29] Tim Salimans. *Observing Dark Worlds*. http://timsalimans.com/observing-dark-worlds/. 2012.

[30] Bert Speelpenning. 'Compiling Fast Partial Derivatives of Functions Given by Algorithms'. PhD thesis. University of Illinois at Urbana-Champaign, 1980.

[31] R. L. Stratonovich. 'Conditional Markov Processes'. In: *Theory of Probability & Its Applications* 5.2 (1960), pp. 156–178. DOI: 10.1137/1105015. URL: http://dx.doi.org/10.1137/1105015.

[32] Theano Development Team et al. 'Theano: A Python framework for fast computation of mathematical expressions'. In: *arXiv e-prints* abs/1605.02688 (May 2016). URL: http://arxiv.org/abs/1605.02688.

[33] Alexander Terenin and David Draper. 'Cox's Theorem and the Jaynesian Interpretation of Probability'. arXiv preprint. 2015. URL: https://arxiv.org/abs/1507.06597v2.

[34] Robert Edwin Wengert. 'A simple automatic derivative evaluation program'. In: *Communications of the ACM* 7.8 (1964), pp. 463–464.