Bayesian machine learning
Personal notes

Tuan Anh Le

April 9, 2015

Contents

1. Notation

$\{a_n\}$	Same as $\{a_n\}_{n=1}^N$ and $\{a_1, \ldots, a_N\}$ – denotes a set of sequence.
$\mathbf{x} \in R^D$	D-dimensional real-valued vector.
$\sum_k f(\cdot)$	Shorthand for $\sum_{k=1}^K f(\cdot)$ (for an arbitrary index letter).
$\prod_k f(\cdot)$	Shorthand for $\prod_{k=1}^K f(\cdot)$ (for an arbitrary index letter).
$\mathrm{diag}(x_1, \ldots, x_N)$	Diagonal matrix formed from the elements $x_1, \ldots, x_N$.
$\mathbb{I}(\cdot)$	Indicator function, equal to 1 if the argument is true, 0 otherwise.
δ_X	Dirac measure on a set X. Defined for a given $x \in X$ and any measurable set $A \subseteq X$ by $\delta_x(A) = 0$ if $x \notin A$ and 1 if $x \in A$.
$\delta_x(A)$	Dirac measure (above).
$\delta(x, y)$	Indicator, same as $\mathbb{I}(x = y)$.
$\mathbb{N}$	Natural numbers, i.e. positive integers, $\{1, 2, 3, \ldots\}$
$\mathbb{N}_0$	Natural numbers including zero, $\{0, 1, 2, \ldots\}$

2. Basics

2.1. Probability distributions

Summarised in Table 2.1

2.2. Directed graphical models

2.3. Undirected graphical models

2.4. Gaussian distribution

The density of $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \mathbf{x} \in \mathbb{R}^D$ is

$$p(\mathbf{x}) = \frac{1}{\sqrt{|2\pi\boldsymbol{\Sigma}|}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right] \tag{2.1}$$

$$= (2\pi)^{-D/2}|\boldsymbol{\Sigma}|^{-1/2} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right] \tag{2.2}$$

2.4.1. Linear Gaussian model

Given the marginal and conditional distributions to be

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Lambda}^{-1}) \tag{2.3}$$

$$p(\mathbf{y} \mid \mathbf{x}) = \mathcal{N}(\mathbf{y}; \mathbf{A}\mathbf{x} + \mathbf{b}, \mathbf{L}^{-1}) \tag{2.4}$$

the marginal distribution of $\mathbf{y}$ and the conditional distribution of $\mathbf{x}$ given $\mathbf{y}$ are given by

$$p(\mathbf{y}) = \mathcal{N}\left(\mathbf{y}; \mathbf{A}\boldsymbol{\mu} + \mathbf{b}, \mathbf{L}^{-1} + \mathbf{A}\boldsymbol{\Lambda}^{-1}\mathbf{A}^T\right) \tag{2.5}$$

$$p(\mathbf{x} \mid \mathbf{y}) = \mathcal{N}\left(\mathbf{x}; \boldsymbol{\Sigma}\left\{\mathbf{A}^T\mathbf{L}(\mathbf{y}-\mathbf{b}) + \boldsymbol{\Lambda}\boldsymbol{\mu}\right\}, \boldsymbol{\Sigma}\right) \tag{2.6}$$

where

$$\boldsymbol{\Sigma} = \left(\boldsymbol{\Lambda} + \mathbf{A}^T\mathbf{L}\mathbf{A}\right)^{-1} \tag{2.7}$$

Why it works

2.4.2. Joint Gaussians

Distribution	Parameters	Support	PDF/PMF	Mean	Variance
Bernoulli (Ber)	$\theta \in [0,1]$	$x \in \{0,1\}$	$\begin{cases}\theta & \text{if } x=1\\ 1-\theta & \text{if } x=0\end{cases}$	θ	$\theta(1-\theta)$
Beta (Beta)	$\alpha,\beta>0$	$x \in [0,1]$	$\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}x^{\alpha-1}(1-x)^{\beta-1}$	$\frac{\alpha}{\alpha+\beta}$	$\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$
Binomial (Bin)	$N \in \mathbb{N}, \theta \in [0,1]$	$x \in \{0,\dots,N\}$	$\binom{N}{x}\theta^x(1-\theta)^{N-x}$	$N\theta$	$N\theta(1-\theta)$
Beta-Binomial (BetaBin)	$N \in \mathbb{N}, \alpha,\beta>0$	$x \in \{0,\dots,N\}$	$\binom{N}{x}\frac{B(x+\alpha,N-x+\beta)}{B(\alpha,\beta)}$	$\frac{N\alpha}{\alpha+\beta}$	$\frac{N\alpha\beta(\alpha+\beta+N)}{(\alpha+\beta)^2(\alpha+\beta+1)}$
Poisson (Poi)	$\lambda>0$	$x \in \{0,1,2,\dots\}$	$\frac{\lambda^x}{x!}\exp(-\lambda)$	λ	λ
Gamma (Gamma)	$\alpha,\beta>0$	$x>0$	$\frac{\beta^\alpha}{\Gamma(\alpha)}x^{\alpha-1}\exp(-\beta x)$	$\frac{\alpha}{\beta}$	$\frac{\alpha}{\beta^2}$
Negative-Binomial (NB)	$r>0, p\in(0,1)$	$x \in \{0,1,2,\dots\}$	$\frac{\Gamma(x+r)}{x!\,\Gamma(r)}(1-p)^r p^x$	$\frac{pr}{1-p}$	$\frac{pr}{(1-p)^2}$
Categorical (Cat)	$\boldsymbol{\theta} \in [0,1]^K, \sum_k \theta_k = 1$	$x \in \{1,\dots,K\}$	θ_x	Meaningless.	Meaningless.
Dirichlet (Dir)	$\boldsymbol{\alpha} \in (0,\infty)^K$	$\mathbf{x} \in [0,1]^K, \sum_k x_k = 1$	$\frac{\Gamma\left(\sum_k \alpha_k\right)}{\prod_k \Gamma(\alpha_k)}\prod_k x_k^{\alpha_k-1}$	$\frac{\boldsymbol{\alpha}}{\sum_k \alpha_k}$	$\text{var}[x_k] = \frac{\alpha_k(\alpha_0-\alpha_k)}{\alpha_0^2(\alpha_0+1)}$
Multinomial (Mult)	$N \in \mathbb{N}, \boldsymbol{\theta} \in [0,1]^K, \sum_k \theta_k = 1$	$\mathbf{x} \in \{0,\dots,N\}^K, \sum_k x_k = N$	$\frac{N!}{x_1!\cdots x_K!}\theta_1^{x_1}\cdots\theta_K^{x_K}$	$N\boldsymbol{\theta}$	$\text{var}[x_k] = N\theta_k(1-\theta_k)$
Dirichlet-Multinomial (DirMult)	$N \in \mathbb{N}, \boldsymbol{\alpha} \in (0,\infty)^K$	$\mathbf{x} \in \{0,\dots,N\}^K, \sum_k x_k = N$	$\frac{\Gamma(N+1)}{\prod_k \Gamma(x_k+1)}\cdot \frac{\Gamma\left(\sum_k \alpha_k\right)}{\Gamma\left(N+\sum_k \alpha_k\right)}\prod_k \frac{\Gamma(\alpha_k+x_k)}{\Gamma(\alpha_k)}$		

Table 2.1.: Summary of common probability distributions

[a] where $B(\alpha,\beta)$ is the normalisation constant for a Beta distribution, Beta(α,β), which is $\int_x x^{\alpha-1}(1-x)^{\beta-1}\,\mathrm{d}x$ or $\frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$.

3. Bayesian parameter estimation

Given a set of data $\mathcal{D} = \{\mathbf{x}_n\}$, we impose a probability distribution f with parameters $\boldsymbol{\theta}$, which we call the model parameters, on each data point, $\mathbf{x}_n \sim f(\boldsymbol{\theta}), n = 1, \ldots, N$, so that the likelihood becomes $p(\mathcal{D} \mid \boldsymbol{\theta}) = \prod_n f(\mathbf{x}_n \mid \boldsymbol{\theta})$. We also impose a distribution g on $\boldsymbol{\theta}$ with parameters $\boldsymbol{\alpha}$ which we call the hyperparameters. We call this distribution the prior distribution over $\boldsymbol{\theta}$. Bayesian parameter estimation evaluates the posterior distribution, $p(\boldsymbol{\theta} \mid \mathcal{D})$, and the posterior predictive distribution, $p(\tilde{\mathbf{x}} \mid \mathcal{D})$, where $\tilde{\mathbf{x}}$ is a new data point we want to predict.

When the prior $g(\boldsymbol{\theta} \mid \boldsymbol{\alpha})$ is a conjugate prior for a given likelihood distribution $f(\cdot \mid \boldsymbol{\theta})$, the posterior has the same distribution as g, just with different parameters. We call these updated hyperparameters and denote them by adding an apostrophe: $\boldsymbol{\alpha}'$. In other words, the posterior becomes $g(\boldsymbol{\theta} \mid \boldsymbol{\alpha}')$. Table 3 summarises the quantities of interest for several conjugate pairs, followed by the derivations.

3.1. Beta-Bernoulli model

$\mathcal{D} = \{x_n : x_n \sim \mathrm{Ber}(\theta)\}, \theta \sim \mathrm{Beta}(\alpha, \beta)$.

Likelihood.

$$p(\mathcal{D} \mid \theta) = \theta^{N_1}(1-\theta)^{N_0}$$

where $N_1 = \sum_n \mathbb{I}(x_n = 1)$ and $N_0 = \sum_n \mathbb{I}(x_n = 0)$.

Posterior.

$$\begin{aligned} p(\theta \mid \mathcal{D}) &\propto p(\mathcal{D} \mid \theta)p(\theta) \\ &\propto \theta^{N_1}(1-\theta)^{N_0}\theta^{\alpha-1}(1-\theta)^{\beta-1} \\ &= \theta^{\alpha+N_1-1}(1-\theta)^{\beta+N_0-1} \\ &\propto \mathrm{Beta}(\theta \mid \alpha + N_1, \beta + N_0) \end{aligned}$$

Posterior predictive.

$$\begin{aligned} p(\tilde{x} = 1 \mid \mathcal{D}) &= \int_\theta p(\tilde{x}, \theta \mid \mathcal{D}) \, \mathrm{d}\theta \\ &= \int_\theta p(\tilde{x} \mid \theta, \mathcal{D}) p(\theta \mid \mathcal{D}) \, \mathrm{d}\theta \end{aligned}$$

Likelihood	Model parameters	Prior	Hyperparameters	Posterior Hyperparameters	Posterior predictive
Bernoulli	θ	Beta	α, β	$\alpha + \sum_n \mathbb{I}(x_n = 1), \beta + \sum_n \mathbb{I}(x_n = 0)$	$\text{Ber}\left(\tilde{x} \mid \frac{\alpha'}{\alpha' + \beta'}\right)$
Binomial	θ	Beta	α, β	$\alpha + \sum_n x_n, \beta + \sum_n (T_n - x_n)$	$\text{BetaBin}(\tilde{x} \mid \alpha', \beta')$
Poisson	λ	Gamma	α, β	$\alpha + \sum_n x_n, \beta + N$	$\text{NB}\left(\tilde{x} \mid \alpha', \frac{1}{1+\beta'}\right)$
Categorical	$\boldsymbol{\theta} \in \mathbb{R}^K$	Dirichlet	$\boldsymbol{\alpha} \in \mathbb{R}^K$	$\boldsymbol{\alpha} + (n_1, \ldots, n_K)^T$	$\text{Ber}\left(\tilde{x} \mid \frac{\alpha'_{\tilde{x}}}{\sum_k \alpha'_k}\right)$
Multinomial	$\boldsymbol{\theta} \in \mathbb{R}^K$	Dirichlet	$\boldsymbol{\alpha} \in \mathbb{R}^K$	$\boldsymbol{\alpha} + \sum_n \mathbf{x}_n$	$\text{DirMult}(\tilde{\mathbf{x}} \mid \alpha', \tilde{T})$

Table 3.1.: Summary of Bayesian parameter estimation for conjugate pairs

$$
\begin{aligned}
&= \int_\theta p(\tilde{x} \mid \theta) p(\theta \mid \mathcal{D}) \, \mathrm{d}\theta \\
&= \int_\theta \theta \, \mathrm{Beta}(\theta, \alpha', \beta') \, \mathrm{d}\theta \\
&= \mathbb{E}_{\theta \sim \mathrm{Beta}(\alpha', \beta')}[\theta] \\
&= \frac{\alpha'}{\alpha' + \beta'} \\
\implies \tilde{x} &\sim \mathrm{Ber}\left(\frac{\alpha'}{\alpha' + \beta'}\right)
\end{aligned}
$$

3.2. Beta-Binomial model

$\mathcal{D} = \{x_n : x_n \sim \mathrm{Bin}(T_n, \theta)\}$ for some fixed total counts $\{T_n\}$, $\theta \sim \mathrm{Beta}(\alpha, \beta)$.

Likelihood.

$$
\begin{aligned}
p(\mathcal{D} \mid \theta) &= \prod_n \mathrm{Bin}(x_n \mid T_n, \theta) \\
&\propto \prod_n \theta^{x_n} (1-\theta)^{T_n - x_n} \\
&= \theta^{\sum_n x_n} (1-\theta)^{\sum_n T_n - x_n} \\
&= \theta^x (1-\theta)^{T-x} \\
&\propto \mathrm{Bin}(x \mid T, \theta)
\end{aligned}
$$

where $x = \sum_n x_n$ and $T = \sum_n T_n$.

Posterior.

$$
\begin{aligned}
p(\theta \mid \mathcal{D}) &\propto p(\mathcal{D} \mid \theta) p(\theta) \\
&= \mathrm{Bin}(x \mid T, \theta) \, \mathrm{Beta}(\theta \mid \alpha, \beta) \\
&\propto \theta^x (1-\theta)^{T-x} \theta^{\alpha-1} (1-\theta)^{\beta-1} \\
&= \theta^{\alpha+x-1} (1-\theta)^{\beta+T-x-1} \\
&\propto \mathrm{Beta}(\theta \mid \alpha + x, \beta + T - x) \\
&= \mathrm{Beta}\left(\theta \mid \alpha + \sum_n x_n, \beta + \sum_n (T_n - x_n)\right)
\end{aligned}
$$

Posterior predictive. (New data point $\tilde{x}$ for some fixed total count $\tilde{T}$).

$$
\begin{aligned}
p(\tilde{x} \mid \mathcal{D}, \tilde{T}) &= \int_\theta p(\tilde{x}, \theta \mid \mathcal{D}, \tilde{T}) \, \mathrm{d}\theta \\
&= \int_\theta p(\tilde{x} \mid \theta, \mathcal{D}, \tilde{T}) p(\theta \mid \mathcal{D}, \tilde{T}) \, \mathrm{d}\theta
\end{aligned}
$$

$$
\begin{aligned}
&= \int_\theta p(\tilde{x} \mid \theta, \tilde{T}) p(\theta \mid \mathcal{D}) \,\mathrm{d}\theta \\
&= \int_\theta \mathrm{Bin}(\tilde{x} \mid \tilde{T}, \theta) \,\mathrm{Beta}(\theta \mid \alpha', \beta') \,\mathrm{d}\theta \\
&= \int_\theta \left[\binom{\tilde{T}}{\tilde{x}} \theta^{\tilde{x}} (1-\theta)^{\tilde{T}-\tilde{x}} \right] \left[\frac{1}{B(\alpha', \beta')} \theta^{\alpha'-1} (1-\theta)^{\beta'-1} \right] \mathrm{d}\theta \\
&= \binom{\tilde{T}}{\tilde{x}} \frac{1}{B(\alpha', \beta')} \int_\theta \theta^{\tilde{x}+\alpha'-1} (1-\theta)^{\tilde{T}-\tilde{x}+\beta'-1} \,\mathrm{d}\theta \\
&= \binom{\tilde{T}}{\tilde{x}} \frac{B(\alpha' + \tilde{x}, \beta' + \tilde{T} - \tilde{x})}{B(\alpha', \beta')} \\
&= \mathrm{BetaBin}(\tilde{x} \mid \tilde{T}, \alpha', \beta')
\end{aligned}
$$

where $B(\alpha, \beta)$ is the normalisation constant for a Beta distribution, $\mathrm{Beta}(\alpha, \beta)$, which is $\int_x x^{\alpha-1}(1-x)^{\beta-1} \,\mathrm{d}x$ or $\frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$.

3.3. Poisson-Gamma model

$\mathcal{D} = \{x_n : x_n \sim \mathrm{Poi}(\lambda)\}$, $\lambda \sim \mathrm{Gamma}(\alpha, \beta)$.

Likelihood.

$$
p(\mathcal{D} \mid \lambda) = \prod_n \mathrm{Poi}(x_n \mid \lambda)
$$

Posterior.

$$
\begin{aligned}
p(\lambda \mid \mathcal{D}) &\propto p(\mathcal{D} \mid \lambda) p(\lambda) \\
&= \left(\prod_n \mathrm{Poi}(x_n \mid \lambda) \right) \mathrm{Gamma}(\lambda \mid \alpha, \beta) \\
&\propto \left[\prod_n \frac{\lambda^{x_n}}{x_n!} \exp(-\lambda) \right] \left[\lambda^{\alpha-1} \exp(-\lambda\beta) \right] \\
&\propto \lambda^{\alpha + \sum_n x_n - 1} \exp(-\lambda(\beta + N)) \\
&\propto \mathrm{Gamma}\left(\lambda \mid \alpha + \sum_n x_n, \beta + N \right)
\end{aligned}
$$

Posterior predictive.

$$
\begin{aligned}
p(\tilde{x} \mid \mathcal{D}) &= \int_\lambda p(\tilde{x}, \lambda \mid \mathcal{D}) \,\mathrm{d}\lambda \\
&= \int_\lambda p(\tilde{x} \mid \lambda, \mathcal{D}) p(\lambda \mid \mathcal{D}) \,\mathrm{d}\lambda
\end{aligned}
$$

$$
\begin{aligned}
&= \int_\lambda p(\tilde{x} \mid \lambda) p(\lambda \mid \mathcal{D}) \, \mathrm{d}\lambda \\
&= \int_\lambda \mathrm{Poi}(\tilde{x} \mid \lambda) \, \mathrm{Gamma}(\lambda \mid \alpha', \beta') \, \mathrm{d}\lambda \\
&= \int_\lambda \frac{\lambda^{\tilde{x}}}{\tilde{x}!} \exp(-\lambda) \frac{1}{G(\alpha', \beta')} \lambda^{\alpha'-1} \exp(-\beta'\lambda) \, \mathrm{d}\lambda \\
&= \frac{1}{\tilde{x}! G(\alpha', \beta')} \int_\lambda \lambda^{x+\alpha'-1} \exp(-\lambda(\beta'+1)) \, \mathrm{d}\lambda \\
&= \frac{G(\alpha'+x, \beta+1)}{\tilde{x}! G(\alpha', \beta')} \\
&= \frac{\Gamma(\alpha'+\tilde{x})}{\tilde{x}! \Gamma(\alpha')} \cdot \frac{\beta'^{\alpha'}}{(\beta'+1)^{\alpha'+\tilde{x}}} \\
&= \frac{\Gamma(\alpha'+\tilde{x})}{\tilde{x}! \Gamma(\alpha')} \left(1 - \frac{1}{1+\beta'}\right)^{\alpha'} \left(\frac{1}{1+\beta'}\right)^{\tilde{x}} \\
&= \mathrm{NB}\left(\tilde{x} \mid \alpha', \frac{1}{1+\beta'}\right)
\end{aligned}
$$

where $G(\alpha, \beta)$ is the normalisation constant for a Gamma distribution, $\mathrm{Gamma}(\alpha, \beta)$, which is $\int_x x^{\alpha-1} \exp(-\beta x) \, \mathrm{d}x$ or $\frac{\Gamma(\alpha)}{\beta^\alpha}$.

3.4. Dirichlet-Categorical model

$\mathcal{D} = \{x_n : x_n \sim \mathrm{Cat}(\boldsymbol{\theta}), \boldsymbol{\theta} \in \mathbb{R}^K\}$, $\boldsymbol{\theta} \sim \mathrm{Dir}(\boldsymbol{\alpha}), \alpha \in \mathbb{R}^K$.

Likelihood.

$$
p(\mathcal{D} \mid \boldsymbol{\theta}) = \prod_k \theta_k^{n_k}
$$

where $n_k = \sum_n \mathbb{I}(x_n = k)$.

Posterior.

$$
\begin{aligned}
p(\boldsymbol{\theta} \mid \mathcal{D}) &\propto p(\mathcal{D} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta}) \\
&= \prod_k \theta_k^{n_k} \, \mathrm{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) \\
&\propto \prod_k \theta_k^{n_k} \prod_k \theta_k^{\alpha_k - 1} \\
&= \prod_k \theta_k^{\alpha_k + n_k - 1} \\
&\propto \mathrm{Dir}\left(\boldsymbol{\theta} \mid \boldsymbol{\alpha} + (n_1, \ldots, n_K)^T\right)
\end{aligned}
$$

Posterior predictive.

$$\begin{aligned}
p(\tilde{x} \mid \mathcal{D}) &= \int_{\boldsymbol{\theta}} p(\tilde{x}, \boldsymbol{\theta} \mid \mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} \\
&= \int_{\boldsymbol{\theta}} p(\tilde{x} \mid \boldsymbol{\theta}, \mathcal{D}) p(\boldsymbol{\theta} \mid \mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} \\
&= \int_{\boldsymbol{\theta}} p(\tilde{x} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta} \mid \mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} \\
&= \int_{\boldsymbol{\theta}} \mathrm{Cat}(\tilde{x} \mid \boldsymbol{\theta}) \, \mathrm{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\alpha}') \\
&= \int_{\boldsymbol{\theta}} \theta_{\tilde{x}} \, \mathrm{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\alpha}') \, \mathrm{d}\boldsymbol{\theta} \\
&= \mathbb{E}_{\boldsymbol{\theta} \sim \mathrm{Dir}(\boldsymbol{\alpha}')}[\theta_{\tilde{x}}] \\
&= \frac{\alpha'_{\tilde{x}}}{\sum_k \alpha'_k}
\end{aligned}$$

Evidence.

$$\begin{aligned}
p(\mathcal{D}) &= \frac{p(\mathcal{D} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(\boldsymbol{\theta} \mid \mathcal{D})} \\
&= \frac{\left(\prod_k \theta_k^{n_k}\right) \left(\mathrm{Dir}(\boldsymbol{\theta} \mid \alpha)\right)}{\mathrm{Dir}\left(\boldsymbol{\theta} \mid \alpha + (n_1, \ldots, n_K)^T\right)} \\
&= \frac{\left(\prod_k \theta_k^{n_k}\right) \frac{\Gamma\left(\sum_k \alpha_k\right)}{\prod_k \Gamma(\alpha_k)} \left(\prod_k \theta_k^{\alpha_k - 1}\right)}{\frac{\Gamma\left(N + \sum_k \alpha_k\right)}{\prod_k \Gamma(\alpha_k + n_k)} \left(\prod_k \theta_k^{\alpha_k + n_k - 1}\right)} \\
&= \frac{\Gamma\left(\sum_k \alpha_k\right) \prod_k \Gamma(\alpha_k + n_k)}{\Gamma\left(N + \sum_k \alpha_k\right) \prod_k \Gamma(\alpha_k)}
\end{aligned} \tag{3.1}$$

3.5. Dirichlet-Multinomial model

$\mathcal{D} = \left\{\mathbf{x}_n : \mathbf{x}_n \sim \mathrm{Mult}(T_n, \boldsymbol{\theta}), \mathbf{x}_n, \boldsymbol{\theta} \in \mathbb{R}^K\right\}$ for fixed total counts $\{T_n\}$; $\boldsymbol{\theta} \sim \mathrm{Dir}(\boldsymbol{\alpha}), \boldsymbol{\alpha} \in \mathbb{R}^K$.

Likelihood.

$$\begin{aligned}
p(\mathcal{D} \mid \boldsymbol{\theta}) &= \prod_n \mathrm{Mult}(\mathbf{x}_n \mid T_n, \boldsymbol{\theta}) \\
&\propto \prod_n \left(\theta_1^{x_{n,1}} \cdots \theta_K^{x_{n,K}}\right) \\
&= \theta_1^{n_1} \cdots \theta_K^{n_K} \\
&\propto \mathrm{Mult}(\mathbf{x} \mid T, \boldsymbol{\theta})
\end{aligned}$$

where $n_k = \sum_n x_{n,k}, k = 1, \ldots, K$ are the total counts for the side k of the die, $\mathbf{x} = \sum_n \mathbf{x}_n$, and $T = \sum_n T_n$.

Posterior.

$$\begin{aligned}
p(\boldsymbol{\theta} \mid \mathcal{D}) &\propto p(\mathcal{D} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta}) \\
&= \operatorname{Mult}(\mathbf{x} \mid T, \boldsymbol{\theta}) \operatorname{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) \\
&\propto (\theta_1^{n_1} \cdots \theta_K^{n_K}) \left(\theta_1^{\alpha_1 - 1} \cdots \theta_K^{\alpha_K - 1}\right) \\
&= \theta_1^{n_1 + \alpha_1 - 1} \cdots \theta_K^{n_K + \alpha_K - 1} \\
&\propto \operatorname{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\alpha} + \mathbf{x}) \\
&= \operatorname{Dir}\left(\boldsymbol{\theta} \mid \boldsymbol{\alpha} + \sum_n \mathbf{x}_n\right) \qquad (3.2)
\end{aligned}$$

Posterior predictive. (New data point $\tilde{\mathbf{x}}$ for a given total count $\tilde{T} = \sum_k \tilde{x}_k$).

$$\begin{aligned}
p(\tilde{\mathbf{x}} \mid \mathcal{D}) &= \int_{\boldsymbol{\theta}} p(\tilde{\mathbf{x}}, \boldsymbol{\theta} \mid \mathcal{D}, \tilde{T}) \, \mathrm{d}\boldsymbol{\theta} \\
&= \int_{\boldsymbol{\theta}} p(\tilde{\mathbf{x}} \mid \boldsymbol{\theta}, \mathcal{D}, \tilde{T}) p(\boldsymbol{\theta} \mid \mathcal{D}, \tilde{T}) \, \mathrm{d}\boldsymbol{\theta} \\
&= \int_{\boldsymbol{\theta}} p(\tilde{\mathbf{x}} \mid \boldsymbol{\theta}, \tilde{T}) p(\boldsymbol{\theta} \mid \mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} \\
&= \int_{\boldsymbol{\theta}} \operatorname{Mult}(\tilde{\mathbf{x}} \mid \tilde{T}, \boldsymbol{\theta}) \operatorname{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\alpha}') \, \mathrm{d}\boldsymbol{\theta} \\
&= \int_{\boldsymbol{\theta}} \left[\frac{\tilde{T}!}{\prod_k \tilde{x}_k!} \prod_k \theta_k^{\tilde{x}_k}\right] \left[\frac{1}{D(\boldsymbol{\alpha}')} \prod_k \theta_k^{\alpha'_k - 1}\right] \mathrm{d}\boldsymbol{\theta} \\
&= \frac{\tilde{T}!}{\prod_k \tilde{x}_k!} \cdot \frac{1}{D(\boldsymbol{\alpha}')} \int_{\boldsymbol{\theta}} \prod_k \theta_k^{\alpha'_k + \tilde{x}_k - 1} \, \mathrm{d}\boldsymbol{\theta} \\
&= \frac{\tilde{T}!}{\prod_k \tilde{x}_k!} \cdot \frac{D(\boldsymbol{\alpha}' + \tilde{\mathbf{x}})}{D(\boldsymbol{\alpha}')} \\
&= \frac{\tilde{T}!}{\prod_k \tilde{x}_k!} \cdot \frac{\prod_k \Gamma(\alpha'_k + \tilde{x}_k)}{\Gamma(\sum_k \alpha'_k + \tilde{x}_k)} \cdot \frac{\Gamma(\sum_k \alpha'_k)}{\prod_k \Gamma(\alpha'_k)} \\
&= \frac{\Gamma(\tilde{T} + 1)}{\prod_k \Gamma(\tilde{x}_k + 1)} \cdot \frac{\Gamma\left(\sum_k \alpha'_k\right)}{\Gamma\left(\tilde{T} + \sum_k \alpha'_k\right)} \prod_k \frac{\Gamma(\alpha'_k + \tilde{x}_k)}{\Gamma(\alpha'_k)} \\
&= \operatorname{DirMult}(\tilde{\mathbf{x}} \mid \boldsymbol{\alpha}', \tilde{T})
\end{aligned}$$

where $D(\boldsymbol{\alpha})$ is the normalisation constant for the Dirichlet distribution, $\operatorname{Dir}(\boldsymbol{\alpha})$, which is $\int_{\mathbf{x}} \prod_k x_k^{\alpha_k - 1} \, \mathrm{d}\mathbf{x}$ or $\frac{\prod_k \Gamma(\alpha_k)}{\Gamma(\sum_k \alpha_k)}$.

4. Advanced models

4.1. Linear regression

We have training data $\left\{\mathbf{x}_n \in \mathbb{R}^{D_x}, y_n \in \mathbb{R}\right\}_{n=1}^{N}$, a mapping to feature space $\boldsymbol{\phi} : \mathbb{R}^{D_x} \to \mathbb{R}^D$ which gives us the features $\boldsymbol{\phi}_n = \boldsymbol{\phi}(\mathbf{x}_n), n = 1, \ldots, N$. We also group the variables into outputs $\mathbf{y} = [y_1, \ldots, y_N]^T$, inputs $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^T$ and features $\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_N]^T$. We also collectively notate data $\mathcal{D} = \{\mathbf{X}, \mathbf{y}\}$. The model is:

$$y_n = \mathbf{w}^T \boldsymbol{\phi}_n + \epsilon, n = 1, \ldots, N \tag{4.1}$$

$$\epsilon \sim \mathcal{N}(0, \sigma^2) \tag{4.2}$$

i.e.

$$y_n \sim \mathcal{N}(\mathbf{w}^T \boldsymbol{\phi}_n, \sigma^2), n = 1, \ldots, N \tag{4.3}$$

We can write the likelihood and prior as follows:

$$\begin{aligned} p(\mathcal{D} \mid \mathbf{w}) &= \prod_n \mathcal{N}(y_n \mid \mathbf{w}^T \boldsymbol{\phi}_n, \sigma^2) \\ &= \mathcal{N}(\mathbf{y} \mid \boldsymbol{\Phi}\mathbf{w}, \sigma^2 \mathbf{I}) \end{aligned} \tag{4.4}$$

$$p(\mathbf{w}) = \mathcal{N}(\mathbf{w} \mid \mathbf{m}_0, \mathbf{S}_0) \tag{4.5}$$

4.1.1. Posterior

The posterior can be evaluated using the results from 2.4.1 with the following mappings: $\mathbf{x} \to \mathbf{w}, \mathbf{y} \to \mathbf{y}, \boldsymbol{\mu} \to \mathbf{m}_0, \boldsymbol{\Lambda}^{-1} \to \mathbf{S}_0, \mathbf{A} \to \boldsymbol{\Phi}, \mathbf{b} \to \mathbf{0}, \mathbf{L}^{-1} \to \sigma^2 \mathbf{I}$ to get

$$p(\mathbf{w} \mid \mathcal{D}) = \mathcal{N}(\mathbf{w} \mid \mathbf{m}_N, \mathbf{S}_N) \tag{4.6}$$

$$\mathbf{m}_N = \mathbf{S}_N \left(\mathbf{S}_0^{-1} \mathbf{m}_0 + \frac{1}{\sigma^2} \boldsymbol{\Phi}^T \mathbf{y} \right) \tag{4.7}$$

$$\mathbf{S}_N^{-1} = \mathbf{S}_0^{-1} + \frac{1}{\sigma^2} \boldsymbol{\Phi}^T \boldsymbol{\Phi} \tag{4.8}$$

4.1.2. Posterior predictive

The posterior predictive for a new data point $\mathbf{x}^*$ ($\boldsymbol{\phi}^* = \boldsymbol{\phi}(\mathbf{x})$) can be evaluated as

$$p(\mathbf{y}^* \mid \mathcal{D}, \mathbf{x}^*) = \int p(y^*, \mathbf{w} \mid \mathcal{D}, \mathbf{x}^*) \, \mathrm{d}\mathbf{w} \tag{4.9}$$

$$= \int p(y^* \mid \mathbf{w}, \mathcal{D}, \mathbf{x}^*) p(\mathbf{w} \mid \mathcal{D}, \mathbf{x}^*) \, \mathrm{d}\mathbf{w} \tag{4.10}$$

$$= \int p(y^* \mid \mathbf{w}, \mathbf{x}^*) p(\mathbf{w} \mid \mathcal{D}) \, \mathrm{d}\mathbf{w} \tag{4.11}$$

$$= \int \mathcal{N}(y^* \mid \mathbf{w}^T \boldsymbol{\phi}^*, \sigma^2) \, \mathcal{N}(\mathbf{w} \mid \mathbf{m}_N, \mathbf{S}_N) \tag{4.12}$$

Using the results from 2.4.1 with the following mappings: $\mathbf{x} \to \mathbf{w} \mid \mathcal{D}, \mathbf{y} \to y^* \mid \mathcal{D}, \mathbf{x}^*, \boldsymbol{\mu} \to \mathbf{m}_N, \boldsymbol{\Lambda}^{-1} \to \mathbf{S}_N, \mathbf{A} \to \boldsymbol{\phi}^{*T}, \mathbf{b} \to \mathbf{0}, \mathbf{L}^{-1} \to \sigma^2$, we get

$$p(y^* \mid \mathcal{D}, \mathbf{x}^*) = \mathcal{N}\left(y^* \mid \boldsymbol{\phi}^{*T} \mathbf{m}_N, \sigma^2 + \boldsymbol{\phi}^{*T} \mathbf{S}_N \boldsymbol{\phi}^*\right) \tag{4.13}$$

4.1.3. Sequential learning

We can learn the posterior of $\mathbf{w}$ sequentially, i.e. by considering yesterday's posterior to be today's prior, which turns out to be equivalent to learning in batch. We need to prove that we get the same posterior for $\mathbf{w}$ in both cases for any number of old data points $N \in \mathbb{N}_0$ and any number of new data points to predict $K \in \mathbb{N}$. We introduce the notation $\mathcal{D}_{i:j} = \{(\mathbf{x}_i, y_i), \ldots, (\mathbf{x}_j, y_j)\}, 0 \leq i \leq j \leq N$. Learning sequentially means treating $p\left(\mathcal{D}_{(N+1):(N+K)} \mid \mathbf{w}\right)$ as the likelihood, $p\left(\mathbf{w} \mid \mathcal{D}_{1:N}\right)$ as the prior, $p\left(\mathbf{w} \mid \mathcal{D}_{(N+1):(N+K)}\right)$ as the posterior and $p\left(\mathcal{D}_{(N+1):(N+K)}\right)$ as the evidence. The posterior of $\mathbf{w}$ when learning in batch can be expressed as

$$\begin{aligned}
p\left(\mathbf{w} \mid \mathcal{D}_{1:(N+K)}\right) &= \frac{p\left(\mathbf{w}, \mathcal{D}_{1:(N+K)}\right)}{p\left(\mathcal{D}_{1:(N+K)}\right)} \\
&= \frac{p\left(\mathbf{w} \mid \mathcal{D}_{1:N}\right) p\left(\mathcal{D}_{1:N}\right) p\left(\mathcal{D}_{(N+1):(N+K)} \mid \mathbf{w}, \mathcal{D}_{1:N}\right)}{p\left(\mathcal{D}_{1:(N+K)}\right)} \\
&= \frac{p\left(\mathbf{w} \mid \mathcal{D}_{1:N}\right) p\left(\mathcal{D}_{(N+1):(N+K)} \mid \mathbf{w}, \mathcal{D}_{1:N}\right)}{p\left(\mathcal{D}_{(N+1):(N+K)} \mid \mathcal{D}_{1:N}\right)} \\
&= \frac{p\left(\mathbf{w} \mid \mathcal{D}_{1:N}\right) p\left(\mathcal{D}_{(N+1):(N+K)} \mid \mathbf{w}\right)}{p\left(\mathcal{D}_{(N+1):(N+K)}\right)}
\end{aligned}$$

which is in the same form as if $\mathbf{w}$ was learnt sequentially, i.e. $\mathbf{w} \mid \mathcal{D}_{(N+1):(N+K)}$ was evaluated with the aforementioned modifications to the likelihood, prior and evidence.

4.1.4. Relationship to ML, MAP and least-squares estimation

Assume $\mathbf{m}_0 = \mathbf{0}, \mathbf{S}_0 = \alpha^{-1}\mathbf{I}$. Then the log of the posterior becomes

$$\begin{aligned}
\ln p(\mathbf{w} \mid \mathcal{D}) &= \ln p(\mathcal{D} \mid \mathbf{w}) + \ln p(\mathbf{w}) + \text{const.} \\
&= -\frac{1}{2\sigma^2} \sum_n (y_n - \mathbf{w}^T \boldsymbol{\phi}_n)^2 - \frac{\alpha}{2} \mathbf{w}^T \mathbf{w} + \text{const.}
\end{aligned}$$

Therefore MAP estimation is equivalent to least-squares estimation with squared regularisation term $\frac{\alpha}{2}\|\mathbf{w}\|^2$. MLE arises when the prior is flat, i.e. $\alpha \to 0$, in which case it is equivalent to least-squares estimation without regularisation.

4.2. Logistic regression

4.3. Mixture models

In mixture models, we have discrete latent states $\mathbf{Z} = \{z_n, z_n \in \{1, \ldots, K\}\}, n = 1, \ldots, N$ and observed states $\mathbf{X} = \left\{\mathbf{x}_n, \mathbf{x}_n \in \mathbb{R}^D\right\}, n = 1, \ldots, N$. We set the priors and the class conditional likelihoods to be $p(z_n) = \text{Cat}(\boldsymbol{\pi}), \boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)$ and $p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta}) = p_k(\mathbf{x}_n \mid \boldsymbol{\theta})$. We can thus express the likelihood of the observed variables to be:

$$\begin{aligned} p(\mathbf{x}_n \mid \boldsymbol{\theta}) &= \sum_{k=1}^{K} p(\mathbf{x}_n, z_n = k; \boldsymbol{\theta}) \\ &= \sum_{k=1}^{K} p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta}) p(z_n = k \mid \boldsymbol{\theta}) \\ &= \sum_{k=1}^{K} \pi_k p_k(\mathbf{x}_n \mid \boldsymbol{\theta}) \end{aligned} \tag{4.14}$$

We can also express the posterior probability that point n belongs to cluster k, or the *responsibility* $r_{nk}(\boldsymbol{\theta})$ (often abbreviated as r_{nk}) of cluster k for point n to be:

$$\begin{aligned} r_{nk}(\boldsymbol{\theta}) &\triangleq p(z_n = k \mid \mathbf{x}_n; \boldsymbol{\theta}) \\ &= \frac{p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta}) p(z_n = k \mid \boldsymbol{\theta})}{\sum_{k'=1}^{K} p(\mathbf{x}_n \mid z_n = k'; \boldsymbol{\theta}) p(z_n = k' \mid \boldsymbol{\theta})} \end{aligned} \tag{4.15}$$

Evaluating the above is called *soft clustering*. *Hard clustering* finds the MAP estimate as follows:

$$\begin{aligned} z_n^* &= \arg\max_k r_{nk} \\ &= \arg\max_k \left\{\log p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta}) + \log(z_n = k \mid \boldsymbol{\theta})\right\} \end{aligned} \tag{4.16}$$

Unidentifiability refers to the fact that the posterior distribution for the parameter $p(\boldsymbol{\theta} \mid \mathcal{D})$ can be multimodal (with equal peaks) and hence cant find a unique ML/MAP estimate.

We distinguish between two log likelihoods – log likelihood for the observed data, denoted by $\ell(\boldsymbol{\theta})$ and log likelihood for complete data, denoted by $\ell_c(\boldsymbol{\theta})$. These two quantities can be expressed as:

$$\ell(\boldsymbol{\theta}) \triangleq \log p(\mathcal{D} \mid \boldsymbol{\theta})$$

$$
\begin{aligned}
&= \log \prod_{n=1}^{N} p(\mathbf{x}_n \mid \boldsymbol{\theta}) \\
&= \log \left\{ \prod_{n=1}^{N} \sum_{k=1}^{K} p(\mathbf{x}_n, z_n = k \mid \boldsymbol{\theta}) \right\} \\
&= \sum_{n=1}^{N} \log \sum_{k=1}^{K} p(\mathbf{x}_n, z_n = k \mid \boldsymbol{\theta}) \qquad (4.17) \\
\ell_c(\boldsymbol{\theta}) &\triangleq \log p\left(\{\mathbf{x}_n, z_n\} \mid \boldsymbol{\theta}\right) \\
&= \log \prod_{n} p(\mathbf{x}_n, z_n \mid \boldsymbol{\theta}) \\
&= \sum_{n} \log p(\mathbf{x}_n, z_n \mid \boldsymbol{\theta}) \qquad (4.18)
\end{aligned}
$$

The log likelihood for observed data, $\ell(\boldsymbol{\theta})$ can't be guaranteed to be convex so it might be intractable to find ML/MAP estimates. Alternatively, we just express these terms as $\ell(\boldsymbol{\theta}) = \log p(\mathbf{X} \mid \boldsymbol{\theta})$ and $\ell_c(\boldsymbol{\theta}) = \log p(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta})$.

4.3.1. EM algorithm

Maximise the likelihood

Goal is to maximise

$$p(\mathbf{X} \mid \boldsymbol{\theta})$$

Assume it's easy to maximise the *auxiliary function*

$$\mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right) \triangleq \mathbb{E}_{\mathbf{Z} \sim \cdot \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}}} \left[\ell_c(\boldsymbol{\theta})\right] \qquad (4.19)$$

w.r.t. $\boldsymbol{\theta}$. Note that this function can be rewritten as either

$$\mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right) = \sum_{\mathbf{Z}} p\left(\mathbf{Z} \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}}\right) \ln p\left(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta}\right) \qquad (4.20)$$

or

$$
\begin{aligned}
\mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right) &= \mathbb{E}_{\mathbf{Z} \sim \cdot \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}}} \left[\ln p\left(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta}\right)\right] \qquad (4.21) \\
&= \mathbb{E}_{\mathbf{Z} \sim \cdot \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}}} \left[\sum_{n} \ln p\left(\mathbf{x}_n, z_n \mid \boldsymbol{\theta}\right)\right] \qquad (4.22) \\
&= \sum_{n} \mathbb{E}_{z_n \sim \cdot \mid \mathbf{x}_n; \boldsymbol{\theta}^{\text{old}}} \left[\ln p(\mathbf{x}_n, z_n \mid \boldsymbol{\theta})\right] \qquad (4.23) \\
&= \sum_{n} \sum_{k} p\left(z_n = k \mid \mathbf{x}_n; \boldsymbol{\theta}^{\text{old}}\right) \ln p(\mathbf{x}_n, z_n = k \mid \boldsymbol{\theta}) \qquad (4.24) \\
&= \sum_{n} \sum_{k} r_{nk}\left(\boldsymbol{\theta}^{\text{old}}\right) \ln\left(\pi_k p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta})\right) \qquad (4.25) \\
&= \sum_{n} \sum_{k} r_{nk}\left(\boldsymbol{\theta}^{\text{old}}\right) \left(\ln \pi_k + \ln p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta})\right) \qquad (4.26)
\end{aligned}
$$

We can express $\ln p(\mathbf{X} \mid \boldsymbol{\theta})$ as

$$\ln p(\mathbf{X} \mid \boldsymbol{\theta}) = \mathcal{L}(q, \boldsymbol{\theta}) + \mathrm{KL}(q \parallel p) \tag{4.27}$$

where

$$\mathcal{L}(q, \boldsymbol{\theta}) = \sum_{\mathbf{Z}} q(\mathbf{Z}) \ln \frac{p(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta})}{q(\mathbf{Z})} \tag{4.28}$$

$$\mathrm{KL}(q \parallel p) = -\sum_{\mathbf{Z}} q(\mathbf{Z}) \ln \frac{p(\mathbf{Z} \mid \mathbf{X}, \boldsymbol{\theta})}{q(\mathbf{Z})} \tag{4.29}$$

because

$$\begin{aligned}
\mathrm{RHS} &= \mathcal{L}(q, \boldsymbol{\theta}) + \mathrm{KL}(q \parallel p) \\
&= \sum_{\mathbf{Z}} q(\mathbf{Z}) \ln \frac{p(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta})}{q(\mathbf{Z})} - \sum_{\mathbf{Z}} q(\mathbf{Z}) \ln \frac{p(\mathbf{Z} \mid \mathbf{X}, \boldsymbol{\theta})}{q(\mathbf{Z})} \\
&= \sum_{\mathbf{Z}} q(\mathbf{Z}) \ln \frac{p(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta})}{p(\mathbf{Z} \mid \mathbf{X}; \boldsymbol{\theta})} \\
&= \sum_{\mathbf{Z}} q(\mathbf{Z}) \ln p(\mathbf{X} \mid \boldsymbol{\theta}) \\
&= \ln p(\mathbf{X} \mid \boldsymbol{\theta}) \\
&= \mathrm{LHS}
\end{aligned}$$

The actual algorithm is as follows

Algorithm 1 EM algorithm for maximising the likelihood

1: Initialise $\boldsymbol{\theta}^{\mathrm{new}}$.
2: **repeat**
3: $\quad \boldsymbol{\theta}^{\mathrm{old}} \leftarrow \boldsymbol{\theta}^{\mathrm{new}}$
4: $\quad$ E step: Set $q(\mathbf{Z}) = p\left(\mathbf{Z} \mid \mathbf{X}, \boldsymbol{\theta}^{\mathrm{old}}\right)$.
5: $\quad$ M step: Hold $q(\mathbf{Z})$ fixed and set $\boldsymbol{\theta}^{\mathrm{new}} = \arg\max_{\boldsymbol{\theta}} \mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\mathrm{old}}\right)$.
6: **until** convergence.

E step. Hold $\boldsymbol{\theta}^{\mathrm{old}}$, maximise $\mathcal{L}\left(q, \boldsymbol{\theta}^{\mathrm{old}}\right)$ w.r.t. q. Since the quantity $\ln p(\mathbf{X} \mid \boldsymbol{\theta})$ in (4.27) is constant w.r.t. q, we can maximise $\mathcal{L}\left(q, \boldsymbol{\theta}^{\mathrm{old}}\right)$ by minimising $\mathrm{KL}(q \parallel p)$. This can be done by setting the KL to 0 by setting $q(\mathbf{Z}) = p(\mathbf{Z} \mid \mathbf{X}; \boldsymbol{\theta}^{\mathrm{old}})$.

M step. Hold $q(\mathbf{Z}) = p(\mathbf{Z} \mid \mathbf{X}; \boldsymbol{\theta}^{\mathrm{old}})$ fixed, maximise $\mathcal{L}(q, \boldsymbol{\theta})$ w.r.t. $\boldsymbol{\theta}$ to get $\boldsymbol{\theta}^{\mathrm{new}}$. We can rewrite $\mathcal{L}(q, \boldsymbol{\theta})$ as

$$\mathcal{L}(q, \boldsymbol{\theta}) = \sum_{\mathbf{Z}} q(\mathbf{Z}) \ln \frac{p(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta})}{q(\mathbf{Z})}$$

$$
\begin{aligned}
&= \sum_{\mathbf{Z}} q(\mathbf{Z}) \ln p\left(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta}\right) - \sum_{\mathbf{Z}} q(\mathbf{Z}) \ln q(\mathbf{Z}) \\
&= \sum_{\mathbf{Z}} p\left(\mathbf{Z} \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}}\right) \ln p\left(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta}\right) - \sum_{\mathbf{Z}} p\left(\mathbf{Z} \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}}\right) \ln p\left(\mathbf{Z} \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}}\right) \\
&= \mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right) + \text{constant w.r.t. } \boldsymbol{\theta}
\end{aligned}
$$

from which we can see that we should maximise $\mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right)$. In both steps, the value of $\mathcal{L}(q, \boldsymbol{\theta})$ increases.

Maximising the posterior

Goal is to maximise

$$p(\boldsymbol{\theta} \mid \mathbf{X})$$

Assume it's easy to maximise

$$\mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right) + \ln p(\boldsymbol{\theta}) \tag{4.30}$$

w.r.t. $\boldsymbol{\theta}$.

We can express $\ln p(\boldsymbol{\theta} \mid \mathbf{X})$ as

$$
\begin{aligned}
\ln p(\boldsymbol{\theta} \mid \mathbf{X}) &= \ln p(\mathbf{X} \mid \boldsymbol{\theta}) + \ln p(\boldsymbol{\theta}) - \ln p(\mathbf{X}) \\
&= \mathcal{L}(q, \boldsymbol{\theta}) + \mathrm{KL}(q \parallel p) + \ln p(\boldsymbol{\theta}) - \ln p(\mathbf{X})
\end{aligned} \tag{4.31}
$$

E step. Here, we perform the same thing as in maximising the likelihood, with the same reasons.

M step. Hold $q(\mathbf{Z}) = p(\mathbf{Z} \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}})$ fixed, maximise $\mathcal{L}\left(q, \boldsymbol{\theta}\right) + \ln p(\boldsymbol{\theta})$ w.r.t. $\boldsymbol{\theta}$ to get $\boldsymbol{\theta}^{\text{new}}$. We can rewrite $\mathcal{L}\left(q, \boldsymbol{\theta}\right) + \ln p(\boldsymbol{\theta})$ as

$$\mathcal{L}\left(q, \boldsymbol{\theta}\right) + \ln p(\boldsymbol{\theta}) = \mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right) + \ln p(\boldsymbol{\theta}) + \text{constant w.r.t. } \boldsymbol{\theta}$$

from which we can see that we should maximise $\mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right) + \ln p(\boldsymbol{\theta})$. In both steps, the value of $\mathcal{L}(q, \boldsymbol{\theta}) + \ln p(\boldsymbol{\theta})$ increases.

The actual algorithm is as follows

Algorithm 2 EM algorithm for maximising the posterior

1: Initialise $\boldsymbol{\theta}^{\text{new}}$.
2: **repeat**
3: $\quad \boldsymbol{\theta}^{\text{old}} \leftarrow \boldsymbol{\theta}^{\text{new}}$
4: $\quad$ E step: Set $q(\mathbf{Z}) = p\left(\mathbf{Z} \mid \mathbf{X}, \boldsymbol{\theta}^{\text{old}}\right)$.
5: $\quad$ M step: Hold $q(\mathbf{Z})$ fixed and set $\boldsymbol{\theta}^{\text{new}} = \arg\max_{\boldsymbol{\theta}} \left\{\mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right) + \ln(\boldsymbol{\theta})\right\}$.
6: **until** convergence.

4.4. Gaussian mixture model

Gaussian mixture model, a.k.a. GMM, or mixture of Gaussians is a mixture model where

$$\boldsymbol{\alpha} \in (0, \infty)^K \tag{4.32}$$
$$\boldsymbol{\pi} \mid \alpha \sim \mathrm{Dir}(\boldsymbol{\alpha}) \tag{4.33}$$
$$\boldsymbol{\mu}_k \mid \mathbf{m}_0, \mathbf{V}_0 \sim \mathcal{N}(\mathbf{m}_0, \mathbf{V}_0), k = 1, \ldots, K \tag{4.34}$$
$$\boldsymbol{\Sigma}_k \mid \mathbf{S}_0, \nu_0 \sim \text{Inverse-Wishart}(\mathbf{S}_0, \nu_0), k = 1, \ldots, K \tag{4.35}$$
$$\mathbf{x}_n \mid z_n, \boldsymbol{\theta} \sim \mathcal{N}\left(\boldsymbol{\mu}_{z_n}, \boldsymbol{\Sigma}_{z_n}\right), n = 1, \ldots, N \tag{4.36}$$

We adopt the following grouping of random variables:

$$\mathcal{G}_0 \coloneqq (\mathbf{m}_0, \mathbf{V}_0, \mathbf{S}_0, \nu_0) \tag{4.37}$$
$$\boldsymbol{\theta}_k \coloneqq (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), k = 1, \ldots, K \tag{4.38}$$
$$\boldsymbol{\theta} \coloneqq (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K). \tag{4.39}$$

The graphical model can be seen in Figure 4.4 below.

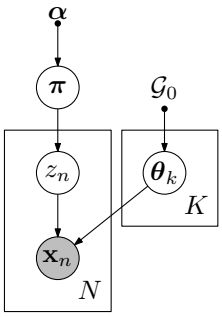

Figure 4.1.: Graphical model for the Gaussian mixture model.

4.4.1. Gibbs sampling for GMMs

4.4.2. Collapsed Gibbs sampling for GMMs

4.4.3. EM algorithm for GMM

Algorithm 3 EM algorithm for GMM

1: Initialise $\boldsymbol{\theta}^{\text{new}} = (\{\boldsymbol{\pi}_k^{\text{new}}, \boldsymbol{\mu}_k^{\text{new}}, \boldsymbol{\Sigma}_k^{\text{new}}\}, k = 1, \ldots, K)$.
2: **repeat**
3: $\quad \boldsymbol{\theta}^{\text{old}} \leftarrow \boldsymbol{\theta}^{\text{new}}$
4: $\quad$ Set $r_{nk} = p\left(z_n = k \mid \mathbf{x}_n; \boldsymbol{\theta}^{\text{old}}\right)$ for $k = 1, \ldots, K, n = 1, \ldots, N$. ▷ E step
5: $\quad$ Set ▷ M step

$$\pi_k^{\text{new}} = \frac{\sum_n r_{nk}}{N}$$

$$\boldsymbol{\mu}_k^{\text{new}} = \frac{\sum_n r_{nk}\mathbf{x}_n}{\sum_n r_{nk}}$$
$$\boldsymbol{\Sigma}_k^{\text{new}} = \frac{\sum_n r_{nk}(\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_n r_{nk}}$$

for $k = 1, \ldots, K$.

6: **until** convergence.

The analysis of the algorithm follows.

E step. We can express $q(\mathbf{Z} = \mathbf{K}) = p(\mathbf{Z} = \mathbf{K} \mid \mathbf{X}, \boldsymbol{\theta}^{\text{old}})$ where $\mathbf{K} = (k_1, \ldots, k_N), k_n \in \{1, \ldots, K\}$ for $n = 1, \ldots, N$ as

$$\begin{aligned} p(\mathbf{Z} = \mathbf{K} \mid \mathbf{X}, \boldsymbol{\theta}^{\text{old}}) &= \prod_n p\left(z_n = k_n \mid \mathbf{x}_n; \boldsymbol{\theta}^{\text{old}}\right) \\ &= \prod_n r_{nk_n}\left(\boldsymbol{\theta}^{\text{old}}\right) \end{aligned}$$

Therefore, in the E step, we set

$$r_{nk_n}\left(\boldsymbol{\theta}^{\text{old}}\right) = p\left(z_n = k_n \mid \mathbf{x}_n; \boldsymbol{\theta}^{\text{old}}\right) \tag{4.40}$$

for $n = 1, \ldots, N$ for all $\mathbf{K}$ and hold it fixed in the M step. This is effectively holding $r_{nk}\left(\boldsymbol{\theta}^{\text{old}}\right)$ (which we will abbreviate as r_{nk} in this section) fixed for $n = 1, \ldots, N$ and $k = 1, \ldots, K$.

M step. We want to find $\boldsymbol{\theta}^{\text{new}} = \arg\max_{\boldsymbol{\theta}} \mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right)$, where

$$\mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right) = \sum_n \sum_k r_{nk} \left(\ln \pi_k + \ln p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta})\right)$$

To maximise this expression, we use Langrange multipliers because we have a constraint $\sum_k \pi_k = 1$. The Lagrangian is

$$\mathcal{L}_{\mathcal{Q}}(\boldsymbol{\theta}, \lambda) = \mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right) + \lambda\left(1 - \sum_k \pi_k\right)$$

Now, we find the derivatives and set them to zero.

For π_k,

$$\begin{aligned} \frac{\partial \mathcal{L}_{\mathcal{Q}}}{\partial \pi_k} &= \frac{\partial}{\partial \pi_k}\left\{\lambda\left(1 - \sum_j \pi_j\right) + \sum_n \sum_j r_{nj} \ln \pi_j\right\} \\ &= -\lambda + \frac{\sum_n r_{nk}}{\pi_k} \end{aligned}$$

Setting this to zero, we get

$$\pi_k = \frac{\sum_n r_{nk}}{\lambda}$$

but since $\sum_k \pi_k = 1$, we have $\sum_k \frac{\sum_n r_{nk}}{\lambda} = 1$, hence $\lambda = \sum_n \sum_k r_{nk} = \sum_n 1 = N$. Hence

$$\pi_k = \frac{\sum_n r_{nk}}{N} \tag{4.41}$$

for $k = 1, \ldots, K$.

For $\boldsymbol{\mu}_k$,

$$\begin{aligned}
\operatorname{grad}_{\boldsymbol{\mu}_k} \mathcal{L}_{\mathcal{Q}} &= \operatorname{grad}_{\boldsymbol{\mu}_k} \left\{ \sum_n \sum_j r_{nj} \left(\ln \pi_j + \ln p(\mathbf{x}_n \mid z_n = j; \boldsymbol{\theta}) \right) + \lambda \left(1 - \sum_j \pi_j \right) \right\} \\
&= \operatorname{grad}_{\boldsymbol{\mu}_k} \left\{ \sum_n \sum_j r_{nj} \ln \mathcal{N}\left(\mathbf{x}_n \mid \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j \right) \right\} \\
&= \operatorname{grad}_{\boldsymbol{\mu}_k} \left\{ \sum_n r_{nk} \ln \mathcal{N}\left(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k \right) \right\} \\
&= \operatorname{grad}_{\boldsymbol{\mu}_k} \left\{ \sum_n r_{nk} \ln \left[(2\pi)^{-D/2} |\boldsymbol{\Sigma}_k|^{-1/2} \exp\left(-\frac{1}{2} (\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right) \right] \right\} \\
&= \operatorname{grad}_{\boldsymbol{\mu}_k} \left\{ \sum_n r_{nk} \left[-\frac{1}{2} \ln |\boldsymbol{\Sigma}_k| - \frac{1}{2} (\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right] \right\} \\
&= -\sum_n r_{nk} \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k)
\end{aligned}$$

Setting this to zero, we get

$$\boldsymbol{\mu}_k = \frac{\sum_n r_{nk} \mathbf{x}_n}{\sum_n r_{nk}} \tag{4.42}$$

for $k = 1, \ldots, K$.

For $\boldsymbol{\Sigma}_k$,

$$\begin{aligned}
\operatorname{grad}_{\boldsymbol{\Sigma}_k} \mathcal{L}_{\mathcal{Q}} &= \operatorname{grad}_{\boldsymbol{\Sigma}_k} \left\{ \sum_n r_{nk} \left[-\frac{1}{2} \ln |\boldsymbol{\Sigma}_k| - \frac{1}{2} (\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right] \right\} \\
&= -\frac{1}{2} \sum_n r_{nk} \left[\boldsymbol{\Sigma}_k^{-T} - \boldsymbol{\Sigma}_k^{-T} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-T} \right] \\
&= -\frac{1}{2} \boldsymbol{\Sigma}^{-1} \sum_n r_{nk} \left[\mathbf{I} - (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} \right]
\end{aligned}$$

Setting this to zero, we get

$$\boldsymbol{\Sigma}_k = \frac{\sum_n r_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_n r_{nk}} \tag{4.43}$$

4.5. Latent Dirichlet allocation

The model is described as follows:

$$\boldsymbol{\alpha} \in (0,\infty)^T, \text{ usually } = \alpha\mathbf{1} \tag{4.44}$$

$$\boldsymbol{\gamma} \in (0,\infty)^W, \text{ usually } = \gamma\mathbf{1} \tag{4.45}$$

$$\boldsymbol{\pi}_d \mid \alpha \sim \mathrm{Dir}(\boldsymbol{\alpha}), d = 1,\dots,D \tag{4.46}$$

$$\boldsymbol{\beta}_t \mid \boldsymbol{\gamma} \sim \mathrm{Dir}(\boldsymbol{\gamma}), t = 1,\dots,T \tag{4.47}$$

$$z_{n,d} \mid \{\boldsymbol{\pi}_\delta\} \sim \mathrm{Cat}(\boldsymbol{\pi}_d), n = 1,\dots,N_d, d = 1,\dots,D \tag{4.48}$$

$$w_{n,d} \mid z_{n,d}, \{\boldsymbol{\beta}_t\} \sim \mathrm{Cat}\left(\boldsymbol{\beta}_{z_{n,d}}\right), n = 1,\dots,N_d, d = 1,\dots,D. \tag{4.49}$$

where we use the following indexing scheme:

- d or δ for document,
- n or η for word in a document,
- t or τ for topic.

The graphical model can be seen in Figure 4.5 below.

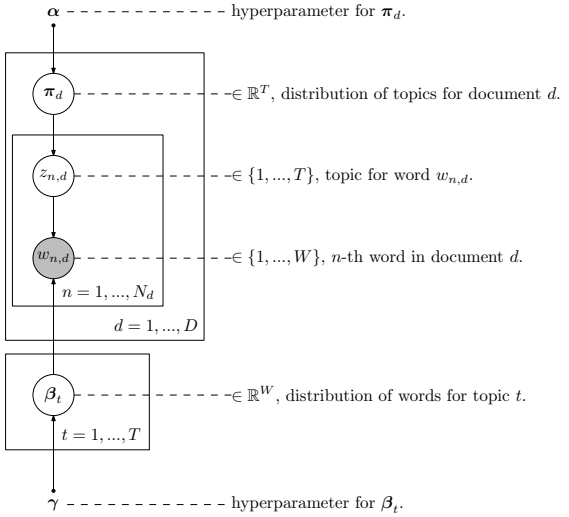

Figure 4.2.: Graphical model for Latent Dirichlet allocation.

The joint probability is

$$p\left(\{\boldsymbol{\pi}_d\}, \{z_{n,d}\}, \{w_{n,d}\}, \{\beta_t\} \mid \alpha, \gamma\right) \tag{4.50}$$

$$= \left(\prod_d p(\boldsymbol{\pi}_d \mid \alpha)\right)\left(\prod_n \prod_d p(z_{n,d} \mid \boldsymbol{\pi}_d)\right)\left(\prod_n \prod_d p(w_{n,d} \mid z_{n,d}, \{\beta_t\})\right)\left(\prod_t p(\beta_t \mid \gamma)\right) \tag{4.51}$$

$$= \left(\prod_d \mathrm{Dir}(\boldsymbol{\pi}_d \mid \boldsymbol{\alpha})\right)\left(\prod_n \prod_d \mathrm{Cat}(z_{n,d} \mid \boldsymbol{\pi}_d)\,\mathrm{Cat}(w_{n,d} \mid \boldsymbol{\beta}_{z_{n,d}})\right)\left(\prod_t \mathrm{Dir}(\boldsymbol{\beta}_t \mid \boldsymbol{\gamma})\right) \tag{4.52}$$

4.5.1. Gibbs sampling for LDA

Although we might only be interested in the quantity $\{z_{n,d}\} \mid \{w_{\eta,\delta}\}; \boldsymbol{\alpha}, \boldsymbol{\gamma}$, the vanilla Gibbs sampler will give us samples from the extended state $\{z_{n,d}\}, \{\boldsymbol{\pi}_d\}, \{\boldsymbol{\beta}_t\} \mid \{w_{\eta,\delta}\}; \boldsymbol{\alpha}, \boldsymbol{\gamma}$ from which if we discard $\{\boldsymbol{\pi}_d\}$ and $\{\boldsymbol{\beta}_t\}$, we get the samples we are interested in.

Word topics $z_{n,d}$

$$
\begin{aligned}
&p\left(z_{n,d} = t \mid \{\boldsymbol{\pi}_\delta\}, \{w_{\eta,\delta}\}, \{\boldsymbol{\beta}_t\}, \{z_{\eta,\delta}\} \setminus z_{\eta,\delta}; \boldsymbol{\alpha}, \boldsymbol{\gamma}\right) \\
&\quad \propto \prod_\eta \prod_\delta \mathrm{Cat}\left(z_{\eta,\delta} \mid \boldsymbol{\pi}_\delta\right) \mathrm{Cat}\left(w_{\eta,\delta} \mid \beta_{z_{\eta,\delta}}\right) && \text{(proportional to the joint)} \\
&\quad \propto \mathrm{Cat}\left(z_{n,d} \mid \boldsymbol{\pi}_d\right) \mathrm{Cat}\left(w_{n,d} \mid \boldsymbol{\beta}_{z_{n,d}}\right) && \text{(discard terms which don't contain } z_{n,d}\text{)} \\
&\quad \propto \mathrm{Cat}\left(t \mid \boldsymbol{\pi}_d\right) \mathrm{Cat}\left(w_{n,d} \mid \boldsymbol{\beta}_t\right) \\
&\quad \propto \pi_{d,t} \beta_{t,w_{n,d}} && (4.53)
\end{aligned}
$$

Document specific parameters $\boldsymbol{\pi}_d$

$$
\begin{aligned}
&p\left(\boldsymbol{\pi}_d \mid \{\boldsymbol{\pi}_\delta\} \setminus \boldsymbol{\pi}_d, \{z_{\eta,\delta}\}, \{w_{\eta,\delta}\}, \{\boldsymbol{\beta}_t\}; \boldsymbol{\alpha}, \gamma\right) \\
&\quad \propto \left(\prod_\delta \mathrm{Dir}(\boldsymbol{\pi}_\delta \mid \boldsymbol{\alpha})\right) \left(\prod_\eta \prod_\delta \mathrm{Cat}\left(z_{\eta,\delta|\boldsymbol{\pi}_\delta}\right)\right) \\
&\quad \propto \mathrm{Dir}(\boldsymbol{\pi}_d \mid \boldsymbol{\alpha}) \prod_\eta \mathrm{Cat}\left(z_{\eta,d} \mid \boldsymbol{\pi}_d\right) \\
&\quad \propto \mathrm{Dir}\left(\boldsymbol{\pi}_d \mid \boldsymbol{\alpha} + (\xi_{d,1}, \ldots, \xi_{d,T})^T\right) && (4.54)
\end{aligned}
$$

where

$$
\begin{aligned}
\xi_{d,t} &= \sum_\eta \mathbb{I}(z_{\eta,d} = t) && (4.55) \\
&= \text{number of words of topic } t \text{ in document } d
\end{aligned}
$$

Topic specific parameters $\boldsymbol{\beta}_t$

$$
\begin{aligned}
&p\left(\boldsymbol{\beta}_t \mid \{\boldsymbol{\pi}_d\}, \{z_{n,d}\}, \{w_{n,d}\}, \{\boldsymbol{\beta}_\tau\} \setminus \boldsymbol{\beta}_t; \boldsymbol{\alpha}, \boldsymbol{\gamma}\right) \\
&\quad \propto \left(\prod_\eta \prod_\delta \mathrm{Cat}\left(w_{\eta,\delta} \mid \boldsymbol{\beta}_{z_{\eta,\delta}}\right)\right) \left(\prod_\tau \mathrm{Dir}(\boldsymbol{\beta}_\tau \mid \boldsymbol{\gamma})\right) \\
&\quad \propto \mathrm{Dir}(\boldsymbol{\beta}_t \mid \boldsymbol{\gamma}) \prod_{\substack{\eta,\delta \\ z_{\eta,\delta}=t}} \mathrm{Cat}(w_{\eta,\delta} \mid \boldsymbol{\beta}_t) \\
&\quad \propto \mathrm{Dir}\left(\boldsymbol{\beta}_t \mid \boldsymbol{\gamma} + (\zeta_{1,t}, \ldots, \zeta_{W,t})^T\right) && (4.56)
\end{aligned}
$$

where

$$\begin{aligned}\zeta_{w,t} &= \sum_{\substack{\eta,\delta \\ z_{\eta,\delta}=t}} \mathbb{I}(w_{\eta,\delta} = w) \\ &= \sum_{\eta,\delta} \mathbb{I}(w_{\eta,\delta} = w, z_{\eta,\delta} = t) \\ &= \text{number words } w \text{ that are assigned to topic } t\end{aligned} \tag{4.57}$$

4.5.2. Collapsed Gibbs sampling for LDA

In the Collapsed Gibbs sampler for LDA, we sample directly from $\{z_{n,d}\} \mid \{w_{\eta,\delta}\}; \boldsymbol{\alpha}, \boldsymbol{\gamma}$. We need to marginalise out $\{\boldsymbol{\beta}_t\}, \{\boldsymbol{\pi}_d\}$ from the original joint distribution.

Marginalising out $\{\boldsymbol{\beta}_t\}, \{\boldsymbol{\pi}_d\}$

$$\begin{aligned}&p\left(\{z_{n,d}\}, \{w_{n,d}\} \mid \boldsymbol{\alpha}, \boldsymbol{\gamma}\right) \\ &\quad= \int_{\{\boldsymbol{\pi}_d\}} \int_{\{\boldsymbol{\beta}_t\}} \left(\prod_d \mathrm{Dir}(\boldsymbol{\pi}_d \mid \boldsymbol{\alpha})\right) \left(\prod_n \prod_d \mathrm{Cat}(z_{n,d} \mid \boldsymbol{\pi}_d)\, \mathrm{Cat}(w_{n,d} \mid \boldsymbol{\beta}_{z_{n,d}})\right) \\ &\quad\left(\prod_t \mathrm{Dir}(\boldsymbol{\beta}_t \mid \boldsymbol{\gamma})\right) \mathrm{d}\{\boldsymbol{\beta}_t\} \mathrm{d}\{\boldsymbol{\pi}_d\} \\ &\quad= \int_{\{\boldsymbol{\pi}_d\}} \left(\prod_d \mathrm{Dir}(\boldsymbol{\pi}_d \mid \boldsymbol{\alpha})\right) \left(\prod_n \prod_d \mathrm{Cat}(z_{n,d} \mid \boldsymbol{\pi}_d)\right) \\ &\quad\int_{\{\boldsymbol{\beta}_t\}} \left(\prod_t \mathrm{Dir}(\boldsymbol{\beta}_t \mid \boldsymbol{\gamma})\right) \left(\prod_n \prod_d \mathrm{Cat}(w_{n,d} \mid \boldsymbol{\beta}_{z_{n,d}})\right) \mathrm{d}\{\boldsymbol{\beta}_t\} \\ &\quad\mathrm{d}\{\boldsymbol{\pi}_d\} \\ &\quad= \left(\frac{\Gamma(W\gamma)}{\Gamma(\gamma)^W}\right)^T \prod_t \frac{\prod_w \Gamma\left(\gamma + \zeta_{w,t}\right)}{\Gamma\left(C_t + W\gamma\right)} \\ &\quad\left(\frac{\Gamma(T\alpha)}{\Gamma(\alpha)^T}\right)^D \prod_d \frac{\prod_t \Gamma\left(\alpha + \xi_{d,t}\right)}{\Gamma\left(N_d + T\alpha\right)} \qquad \text{(see next two subsections)}\end{aligned} \tag{4.58}$$

The $\{\boldsymbol{\beta}_t\}$ integral

$$\begin{aligned}&\int_{\{\boldsymbol{\beta}_t\}} \left(\prod_t \mathrm{Dir}(\boldsymbol{\beta}_t \mid \boldsymbol{\gamma})\right) \left(\prod_n \prod_d \mathrm{Cat}(w_{n,d} \mid \boldsymbol{\beta}_{z_{n,d}})\right) \mathrm{d}\{\boldsymbol{\beta}_t\} \quad \text{(integrals separable)} \\ &\quad= \prod_t \int_{\boldsymbol{\beta}_t} \mathrm{Dir}(\boldsymbol{\beta}_t \mid \boldsymbol{\gamma}) \prod_{\substack{n,d \\ z_{n,d}=t}} \mathrm{Cat}(w_{n,d} \mid \boldsymbol{\beta}_{z_{n,d}}) \mathrm{d}\boldsymbol{\beta}_t \qquad \text{(integrand is prior multiplied by likelihood)}\end{aligned}$$

$$
\begin{aligned}
&= \prod_t \frac{\Gamma\left(\sum_w \gamma_w\right) \prod_w \Gamma\left(\gamma_w + \zeta_{w,t}\right)}{\Gamma\left(C_t + \sum_w \gamma_w\right) \prod_w \Gamma\left(\gamma_w\right)} && \text{(hence the integral is the evidence (see (3.1)))} \\
&= \prod_t \frac{\Gamma\left(W\gamma\right) \prod_w \Gamma\left(\gamma + \zeta_{w,t}\right)}{\Gamma\left(C_t + W\gamma\right) \Gamma\left(\gamma\right)^W} && \text{(assume } \boldsymbol{\gamma} = \gamma\mathbf{1}\text{)} \\
&= \left(\frac{\Gamma(W\gamma)}{\Gamma(\gamma)^W}\right)^T \prod_t \frac{\prod_w \Gamma\left(\gamma + \zeta_{w,t}\right)}{\Gamma\left(C_t + W\gamma\right)} && (4.59)
\end{aligned}
$$

where

$$
\begin{aligned}
\zeta_{w,t} &= \sum_{\substack{\eta,\delta \\ z_{\eta,\delta}=t}} \mathbb{I}(w_{\eta,\delta} = w) \\
&= \sum_{\eta,\delta} \mathbb{I}(w_{\eta,\delta} = w, z_{\eta,\delta} = t) && (4.60) \\
&= \text{number words } w \text{ that are assigned to topic } t && \text{(same as in (4.57))}
\end{aligned}
$$

and

$$
\begin{aligned}
C_t &= \sum_{n,d} \mathbb{I}(z_{n,d} = t) && (4.61) \\
&= \text{number of words that are assigned the topic } t.
\end{aligned}
$$

The $\{\boldsymbol{\pi}_d\}$ integral

$$
\begin{aligned}
&\int_{\{\boldsymbol{\pi}_d\}} \left(\prod_d \mathrm{Dir}(\boldsymbol{\pi}_d \mid \boldsymbol{\alpha})\right) \left(\prod_n \prod_d \mathrm{Cat}(z_{n,d} \mid \boldsymbol{\pi}_d)\right) \mathrm{d}\{\boldsymbol{\pi}_d\} \\
&= \int_{\{\boldsymbol{\pi}_d\}} \prod_d \left((\mathrm{Dir}(\boldsymbol{\pi}_d \mid \boldsymbol{\alpha})) \left(\prod_n \mathrm{Cat}(z_{n,d} \mid \boldsymbol{\pi}_d)\right)\right) \mathrm{d}\{\boldsymbol{\pi}_d\} && \text{(integrals separable)} \\
&= \prod_d \int_{\boldsymbol{\pi}_d} \mathrm{Dir}(\boldsymbol{\pi}_d \mid \boldsymbol{\alpha}) \prod_n \mathrm{Cat}(z_{n,d} \mid \boldsymbol{\pi}_d) \mathrm{d}\boldsymbol{\pi}_d && \text{(integrand is prior multiplied by likelihood)} \\
&= \prod_d \frac{\Gamma\left(\sum_t \alpha_t\right) \prod_t \Gamma\left(\alpha_t + \xi_{d,t}\right)}{\Gamma\left(N_d + \sum_t \alpha_t\right) \prod_t \Gamma\left(\alpha_t\right)} && \text{(hence the integral is the evidence (see (3.1)} \\
&= \prod_d \frac{\Gamma\left(T\alpha\right) \prod_w \Gamma\left(\alpha + \xi_{d,t}\right)}{\Gamma\left(N_d + T\alpha\right) \Gamma\left(\alpha\right)^T} && \text{(assume } \boldsymbol{\alpha} = \alpha\mathbf{1}\text{)} \\
&= \left(\frac{\Gamma(T\alpha)}{\Gamma(\alpha)^T}\right)^D \prod_d \frac{\prod_t \Gamma\left(\alpha + \xi_{d,t}\right)}{\Gamma\left(N_d + T\alpha\right)} && (4.62)
\end{aligned}
$$

where

$$
\begin{aligned}
\xi_{d,t} &= \sum_{\eta} \mathbb{I}(z_{\eta,d} = t) && (4.63) \\
&= \text{number of words of topic } t \text{ in document } d && \text{(same as in (4.55))}
\end{aligned}
$$

and

$$N_d = \text{number of words in document } d \qquad \text{(from the model (Fig. 4.5))} \qquad (4.64)$$

Gibbs updates

4.6. Hidden Markov model

4.6.1. The model

4.6.2. Filtering

4.6.3. Smooting

4.6.4. Posterior sampling

4.7. State space models

4.7.1. The model

4.7.2. Filtering

4.7.3. Smoothing

4.7.4. Extended Kalman filter

4.8. Robotics

4.8.1. Localisation

4.8.2. Mapping

4.8.3. Simultaneous Localisation and Mapping (SLAM)

4.9. Kalman Filters

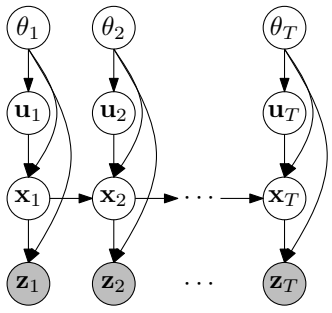

Figure 4.3.: Probabilistic Graphical Model for the Kalman Filter.

4.9.1. Linear Kalman Filter

The model is, for $t = 1, \ldots, T$:

$$\mathbf{x}_t = \mathbf{A}_t \mathbf{x}_{t-1} + \mathbf{B}_t \mathbf{u}_t + \boldsymbol{\epsilon}_t \qquad \in \mathbb{R}^n \tag{4.65}$$
$$\mathbf{z}_t = \mathbf{C}_t \mathbf{x}_t + \mathbf{D}_t \mathbf{u}_t + \boldsymbol{\delta}_t \qquad \in \mathbb{R}^m \tag{4.66}$$
$$\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_t) \qquad \in \mathbb{R}^n \tag{4.67}$$
$$\boldsymbol{\delta}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_t) \qquad \in \mathbb{R}^m \tag{4.68}$$
$$\boldsymbol{\theta}_t = \{\mathbf{A}_t, \mathbf{B}_t, \mathbf{C}_t, \mathbf{D}_t, \mathbf{Q}_t, \mathbf{R}_t\} \tag{4.69}$$

The posteriors of interest are (we drop the conditional dependence on $\boldsymbol{\theta}_t$'s):

$$p(\mathbf{x}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) = \mathcal{N}\left(\mathbf{x}_t \mid \boldsymbol{\mu}_{t|t-1}, \boldsymbol{\Sigma}_{t|t-1}\right) \tag{4.70}$$
$$\boldsymbol{\mu}_{t|t-1} = \mathbf{A}_t \boldsymbol{\mu}_{t-1|t-1} + \mathbf{B}_t \mathbf{u}_t \tag{4.71}$$
$$\boldsymbol{\Sigma}_{t|t-1} = \mathbf{A}_t \boldsymbol{\Sigma}_{t-1|t-1} \mathbf{A}_t^T + \mathbf{Q}_t \tag{4.72}$$
$$p(\mathbf{x}_t \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) = \mathcal{N}\left(\mathbf{x}_t \mid \boldsymbol{\mu}_{t|t}, \boldsymbol{\Sigma}_{t|t}\right) \tag{4.73}$$
$$\boldsymbol{\mu}_{t|t} = \boldsymbol{\mu}_{t|t-1} + \mathbf{K}_t \mathbf{r}_t \tag{4.74}$$
$$\boldsymbol{\Sigma}_{t|t} = (\mathbf{I} - \mathbf{K}_t \mathbf{C}_t) \boldsymbol{\Sigma}_{t|t-1} \tag{4.75}$$
$$\mathbf{r}_t = \mathbf{z}_t - (\mathbf{C}_t \boldsymbol{\mu}_{t|t-1} + \mathbf{D}_t \mathbf{u}_t) \tag{4.76}$$
$$\mathbf{K}_t = \boldsymbol{\Sigma}_{t|t-1} \mathbf{C}_t^T \mathbf{S}_t^{-1} \tag{4.77}$$
$$\mathbf{S}_t = \mathbf{C}_t \boldsymbol{\Sigma}_{t|t-1} \mathbf{C}_t^T + \mathbf{R}_t \tag{4.78}$$

Derivations

4.9.2. Extended Kalman Filter

The model is , for $t = 1, \ldots, T$:

$$\mathbf{x}_t = \mathbf{g}(\mathbf{x}_{t-1}, \mathbf{u}_t) + \boldsymbol{\epsilon}_t \qquad \in \mathbb{R}^n \tag{4.79}$$
$$\mathbf{z}_t = \mathbf{h}(\mathbf{x}_t, \mathbf{u}_t) + \boldsymbol{\delta}_t \qquad \in \mathbb{R}^m \tag{4.80}$$
$$\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_t) \qquad \in \mathbb{R}^n \tag{4.81}$$
$$\boldsymbol{\delta}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_t) \qquad \in \mathbb{R}^m \tag{4.82}$$

The posteriors of interest are (we drop the conditional dependence on $\boldsymbol{\theta}_t$'s):

$$p(\mathbf{x}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \approx \mathcal{N}\left(\mathbf{x}_t \mid \boldsymbol{\mu}_{t|t-1}, \boldsymbol{\Sigma}_{t|t-1}\right) \tag{4.83}$$
$$\boldsymbol{\mu}_{t|t-1} \approx \mathbf{g}(\boldsymbol{\mu}_{t-1|t-1}, \mathbf{u}_t) \tag{4.84}$$
$$\boldsymbol{\Sigma}_{t|t-1} \approx \mathbf{G}_t \boldsymbol{\Sigma}_{t-1|t-1} \mathbf{G}_t^T + \mathbf{Q}_t \tag{4.85}$$
$$p(\mathbf{x}_t \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) \approx \mathcal{N}\left(\mathbf{x}_t \mid \boldsymbol{\mu}_{t|t}, \boldsymbol{\Sigma}_{t|t}\right) \tag{4.86}$$
$$\boldsymbol{\mu}_{t|t} \approx \boldsymbol{\mu}_{t|t-1} + \mathbf{W}_t \left(\mathbf{z}_t - \mathbf{h}(\boldsymbol{\mu}_{t|t-1}, \mathbf{u}_t)\right) \tag{4.87}$$

$$\boldsymbol{\Sigma}_{t|t} \approx \boldsymbol{\Sigma}_{t|t-1} - \mathbf{W}_t \mathbf{S}_t \mathbf{W}_t^T \tag{4.88}$$

Note that we make two types of approximations: (1) we assume the posteriors are Gaussians (which they are not) and (2) we calculate the moments using linearised versions of random variables of interest. We define previously undefined variables below.

Derivations

The derivation of the prediction equations is as follows:

$$\begin{aligned}
\boldsymbol{\mu}_{t|t-1} &= \mathbb{E}\left[\mathbf{x}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}\right] && (4.89)\\
&= \mathbb{E}\left[\mathbf{g}(\mathbf{x}_{t-1}, \mathbf{u}_t) + \boldsymbol{\epsilon}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}\right] && (4.90)\\
&= \mathbb{E}\left[\mathbf{g}(\boldsymbol{\mu}_{t-1|t-1}, \mathbf{u}_t) + \mathbf{G}_t(\mathbf{x}_{t-1} - \boldsymbol{\mu}_{t-1|t-1}) + \cdots + \boldsymbol{\epsilon}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}\right] && (4.91)
\end{aligned}$$

We have linearised around $\boldsymbol{\mu}_{t-1|t-1}$ where $\mathbf{G}_t$ is the Jacobian of $\mathbf{g}(\mathbf{x}, \mathbf{u})$ w.r.t. $\mathbf{x}$ evaluated at $\boldsymbol{\mu}_{t-1|t-1}$:

$$\mathbf{G}_t \triangleq \left.\frac{\partial \mathbf{g}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\boldsymbol{\mu}_{t-1|t-1}, \mathbf{u}=\mathbf{u}_t} \tag{4.92}$$

i.e. $\frac{\partial \mathbf{g}(\mathbf{x},\mathbf{u})}{\partial \mathbf{x}} \in \mathbb{R}^{n \times n}$ and the $(i,j)^{\text{th}}$ element is $\frac{\partial g_i(\mathbf{x},\mathbf{u})}{\partial x_j}$. Taking only the first two terms of the Taylor expansion and continuing with the derivation:

$$\begin{aligned}
\boldsymbol{\mu}_{t|t-1} &\approx \mathbb{E}\left[\mathbf{g}(\boldsymbol{\mu}_{t-1|t-1}, \mathbf{u}_t) + \mathbf{G}_t(\mathbf{x}_{t-1} - \boldsymbol{\mu}_{t-1|t-1}) + \boldsymbol{\epsilon}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}\right] && (4.93)\\
&\vdots && (4.94)\\
&= \mathbf{g}(\boldsymbol{\mu}_{t-1|t-1}, \mathbf{u}_t) && (4.95)
\end{aligned}$$

Similarly, the prediction equation for the covariance can be derived as follows:

$$\begin{aligned}
\boldsymbol{\Sigma}_{t|t-1} &= \operatorname{var}\left[\mathbf{x}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}\right] && (4.96)\\
&= \operatorname{var}\left[\mathbf{g}(\mathbf{x}_{t-1}, \mathbf{u}_t) + \boldsymbol{\epsilon}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}\right] && (4.97)\\
&= \operatorname{var}\left[\mathbf{g}(\boldsymbol{\mu}_{t-1|t-1}, \mathbf{u}_t) + \mathbf{G}_t(\mathbf{x}_{t-1} - \boldsymbol{\mu}_{t-1|t-1}) + \cdots + \boldsymbol{\epsilon}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}\right] && (4.98)\\
&\approx \operatorname{var}\left[\mathbf{g}(\boldsymbol{\mu}_{t-1|t-1}, \mathbf{u}_t) + \mathbf{G}_t(\mathbf{x}_{t-1} - \boldsymbol{\mu}_{t-1|t-1}) + \boldsymbol{\epsilon}_t \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}\right] && (4.99)\\
&\vdots && (4.100)\\
&= \mathbf{G}_t \boldsymbol{\Sigma}_{t-1|t-1} \mathbf{G}_t^T + \mathbf{Q}_t && (4.101)
\end{aligned}$$

The derivation of the update equations is as follows:

$$\begin{aligned}
\boldsymbol{\mu}_{t|t} &= \mathbb{E}\left[\mathbf{x}_t \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t}\right] && (4.102)\\
&\vdots && (4.103)
\end{aligned}$$

$$= \boldsymbol{\mu}_{t|t-1} + \mathbf{W}_t \left(\mathbf{z}_t - \mathbf{h}(\boldsymbol{\mu}_{t|t-1}, \mathbf{u}_t) \right) \tag{4.104}$$

$$\boldsymbol{\Sigma}_{t|t} = \text{var}\left[\mathbf{x}_t \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t} \right] \tag{4.105}$$

$$= \mathbb{E}\left[(\boldsymbol{\mu}_{t|t} - \mathbf{x}_t)^T (\boldsymbol{\mu}_{t|t} - \mathbf{x}_t) \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t} \right] \tag{4.106}$$

$$\vdots \tag{4.107}$$

$$= \boldsymbol{\Sigma}_{t|t-1} - \mathbf{W}_t \mathbf{S}_t \mathbf{W}_t^T \tag{4.108}$$

where

$$\mathbf{W}_t = \boldsymbol{\Sigma}_{t|t-1} \mathbf{H}_t \mathbf{S}_t \tag{4.109}$$

$$\mathbf{S}_t = \mathbf{H}_t \boldsymbol{\Sigma}_{t|t-1} \mathbf{H}_t + \mathbf{R}_t \tag{4.110}$$

We have linearised around $\boldsymbol{\mu}_{t|t-1}$ where $\mathbf{H}_t$ is the Jacobian of $\mathbf{h}(\mathbf{x}, \mathbf{u})$ w.r.t. $\mathbf{x}$ evaluated at $\boldsymbol{\mu}_{t|t-1}$:

$$\mathbf{H}_t \triangleq \left. \frac{\partial \mathbf{h}(\mathbf{x}, \mathbf{u})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\boldsymbol{\mu}_{t|t-1}, \mathbf{u}=\mathbf{u}_t} \tag{4.111}$$

i.e. $\frac{\partial \mathbf{h}(\mathbf{x},\mathbf{u})}{\partial \mathbf{x}} \in \mathbb{R}^{m \times n}$ and the $(i,j)^{\text{th}}$ element is $\frac{\partial h_i(\mathbf{x},\mathbf{u})}{\partial x_j}$.

4.9.3. Localisation

We introduce the map vector $\mathbf{M}$ which contains position and feature vectors of K landmarks in our environment.

$$\mathbf{M} = \begin{bmatrix} \mathbf{m}_1 \\ \vdots \\ \mathbf{m}_K \end{bmatrix} \tag{4.112}$$

where $\mathbf{m}_k$ contains the location and features of the k^{th} landmark, $\mathbf{m}_k \in \mathbb{R}^d$ and $\mathbf{M} \in \mathbb{R}^{Kd}$. In the localisation problem, we assume $\mathbf{M}$ is known and so the graphical model looks as follows:

The model can be described, for $t = 1, \ldots, T$, by the following equations:

$$\mathbf{x}_t = \mathbf{g}(\mathbf{x}_{t-1}, \mathbf{u}_t) + \boldsymbol{\epsilon}_t \tag{4.113}$$

$$\mathbf{z}_t = \mathbf{h}(\mathbf{x}_t, \mathbf{u}_t, \mathbf{M}) + \boldsymbol{\delta}_t \tag{4.114}$$

$$\boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_t) \tag{4.115}$$

$$\boldsymbol{\delta}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_t) \tag{4.116}$$

This model almost exactly resembles the Extended Kalman filter. The update and prediction equations only differ in that $\mathbf{H}_t \triangleq \frac{\partial \mathbf{h}(\mathbf{x},\mathbf{u},\mathbf{M})}{\partial \mathbf{x}}$ instead of $\mathbf{H}_t \triangleq \frac{\partial \mathbf{h}(\mathbf{x},\mathbf{u})}{\partial \mathbf{x}}$ (evaluated at $\mathbf{x} = \boldsymbol{\mu}_{t|t-1}$). We need to characterise $\mathbf{g}$ and $\mathbf{h}$ to fully describe the EKF algorithm.

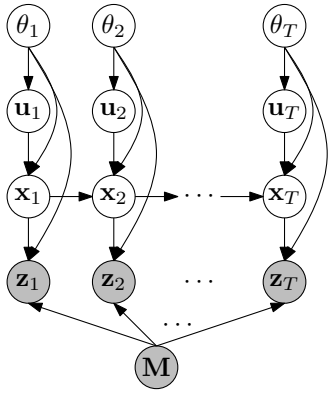

Figure 4.4.: Probabilistic Graphical Model for the Kalman Filter for Localisation.

The transition model

The transition model is defined for the state

$$\mathbf{x}_t = \begin{bmatrix} x_t \\ y_t \\ \theta_t \end{bmatrix} \tag{4.117}$$

which contains the position and orientation of the vehicle (note the slightly overloaded notation of $\mathbf{x}_t$ and x_t). The transition model is then described by

$$\mathbf{g}(\mathbf{x}_{t-1}, \mathbf{u}_t) = \mathbf{x}_{t-1} + \begin{bmatrix} -\frac{v_t}{\omega_t}\sin(\theta_{t-1}) + \frac{v_t}{\omega_t}\sin(\theta_{t-1} + \omega_t \Delta t) \\ \frac{v_t}{\omega_t}\cos(\theta_{t-1}) - \frac{v_t}{\omega_t}\cos(\theta_{t-1} + \omega_t \Delta t) \\ \omega_t \Delta t \end{bmatrix} \tag{4.118}$$

where $\mathbf{u}_t = (v_t, \omega_t)^T$ contains translational and rotational velocities.

The observation model

The observation model is defined for the observation

$$\mathbf{z}_t = \begin{bmatrix} \mathbf{z}_{t,1} \\ \vdots \\ \mathbf{z}_{t,L} \end{bmatrix} \tag{4.119}$$

which contains $L \leq K$ observations $\mathbf{z}_{t,1}, \ldots, \mathbf{z}_{t,L}$ corresponding to the L landmarks observed (out of the total K landmarks) at time t. In the simplified case, $\mathbf{z}_{t,\ell} = (r_{t,\ell}, \phi_{t,\ell})^T$ contains the Euclidean and angular distance from the landmark. We assume that we know which one out of the K it is by introducing the correspondence variables $\mathbf{c}_t$, where if $c_{t,\ell} = j \in \{1, \ldots, K\}$ then the i-th feature observed at t, $\mathbf{z}_{t,\ell}$, corresponds to the j-th landmark, $\mathbf{m}_j$. We assume that a landmark $\mathbf{m}_j = (m_{j,x}, m_{j,y})^T$ just contains the

position of the landmark in this simplified case. Hence we can write, given that $c_{t,\ell} = j$,

$$\mathbf{z}_{t,\ell} = \begin{bmatrix} r_{t,\ell} \\ \phi_{t,\ell} \end{bmatrix} + \begin{bmatrix} \mathcal{N}(0, \sigma_r^2) \\ \mathcal{N}(0, \sigma_\phi^2) \end{bmatrix} \tag{4.120}$$

$$= \begin{bmatrix} \|\mathbf{m}_j - (x_t, y_t)^T\| \\ \arctan\left((m_{j,y} - y_t)/(m_{j,x} - x_t)\right) - \theta_t \end{bmatrix} + \begin{bmatrix} \mathcal{N}(0, \sigma_r^2) \\ \mathcal{N}(0, \sigma_\phi^2) \end{bmatrix} \tag{4.121}$$

The observation model is then described by

$$\mathbf{h}(\mathbf{x}_t, \mathbf{u}_t, \mathbf{M}) = \begin{bmatrix} \|\mathbf{m}_{c_{t,1}} - (x_t, y_t)^T\| \\ \arctan\left((m_{c_{t,1},y} - y_t)/(m_{c_{t,1},x} - x_t)\right) - \theta_t \\ \vdots \\ \|\mathbf{m}_{c_{t,\ell}} - (x_t, y_t)^T\| \\ \arctan\Big((m_{c_{t,\ell},y} - y_t)/(m_{c_{t,\ell},x} - x_t)\Big) - \theta_t \\ \vdots \\ \|\mathbf{m}_{c_{t,L}} - (x_t, y_t)^T\| \\ \arctan\Big((m_{c_{t,L},y} - y_t)/(m_{c_{t,L},x} - x_t)\Big) - \theta_t \end{bmatrix} \tag{4.122}$$

and

$$\mathbf{R}_t = \text{diag}(\sigma_r^2, \sigma_\phi^2, \ldots, \sigma_r^2, \sigma_\phi^2) \in \mathbb{R}^{2L \times 2L} \tag{4.123}$$

4.9.4. Mapping

In the mapping problem, we assume we don't have the knowledge of the map $\mathbf{M}$ (described in subsection 4.9.3), but know the vehicle states $\mathbf{x}_1, \ldots, \mathbf{x}_T$ exactly. We want to figure out $\mathbf{M}$. The graphical model is as follows

on the left, we shade $\mathbf{x}_1, \ldots, \mathbf{x}_T$ and $\mathbf{z}_1, \ldots, \mathbf{z}_T$ to signify that they are observed variables. Note that since both the observation and the states are now observed, the directed edges between them are now redundant (but are left there for clarity). On the right, we group the $\mathbf{x}$'s and $\mathbf{M}$ together. We call these random variables, for $t = 1, \ldots, T$

$$\mathbf{x}_t^* = \begin{bmatrix} \mathbf{x}_t \\ \mathbf{M} \end{bmatrix} \tag{4.124}$$

Since these random variables are only "half-observed", we don't shade them.

The model can be described, for $t = 1, \ldots, T$, by the following equations:

$$\mathbf{x}_t^* = \mathbf{g}^*(\mathbf{x}_{t-1}^*, \mathbf{u}_t) + \boldsymbol{\epsilon}_t^* \tag{4.125}$$

$$\mathbf{z}_t = \mathbf{h}^*(\mathbf{x}_t^*, \mathbf{u}_t) + \boldsymbol{\delta}_t^* \tag{4.126}$$

We need to characterise $\mathbf{g}^*(\mathbf{x}^*, \mathbf{u})$, $\boldsymbol{\epsilon}_t^*$, $\mathbf{h}^*(\mathbf{x}^*, \mathbf{u})$, and $\boldsymbol{\delta}_t^*$ to describe the EKF algorithm fully.

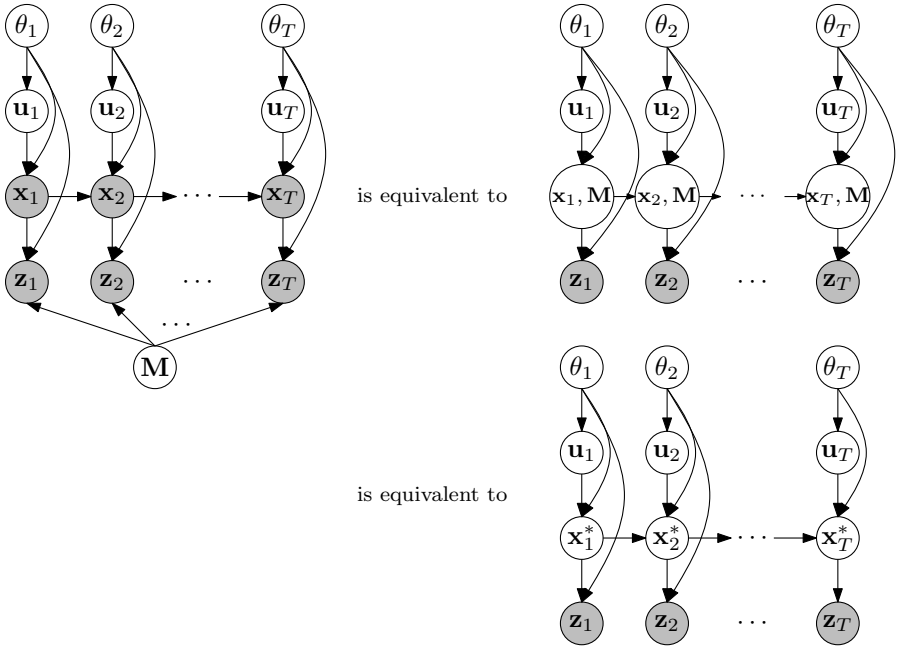

Figure 4.5.: Probabilistic Graphical Model for the Kalman Filter for Mapping.

The transition model

The transition model is described by

$$\mathbf{g}^*(\mathbf{x}^*_{t-1}, \mathbf{u}_t) = \begin{bmatrix} \mathbf{x}_t \\ \mathbf{M} \end{bmatrix} \tag{4.127}$$

and

$$\mathbf{Q}_t = \begin{bmatrix} \mathbf{0}_{n\times n} & \mathbf{0}_{n\times Kd} \\ \mathbf{0}_{Kd\times n} & \mathbf{Q}_{t,M} \end{bmatrix} \tag{4.128}$$

where $\mathbf{Q}_{t,M}$ is the covariance for $\mathbf{M}$ at time t.

The observation model

The observation model is defined for the observation variable is the same as the one in the Localisation case, described in the Subsubsection 4.9.3, in Equations (4.119) and

(4.121). The observation model is then

$$\mathbf{h}^*(\mathbf{x}_t^*, \mathbf{u}_t) = \begin{bmatrix} \|\mathbf{m}_{c_{t,1}} - (x_t, y_t)^T\| \\ \arctan\left((m_{c_{t,1},y} - y_t)/(m_{c_{t,1},x} - x_t)\right) - \theta_t \\ \vdots \\ \|\mathbf{m}_{c_{t,\ell}} - (x_t, y_t)^T\| \\ \arctan\Big((m_{c_{t,\ell},y} - y_t)/(m_{c_{t,\ell},x} - x_t)\Big) - \theta_t \\ \vdots \\ \|\mathbf{m}_{c_{t,L}} - (x_t, y_t)^T\| \\ \arctan\Big((m_{c_{t,L},y} - y_t)/(m_{c_{t,L},x} - x_t)\Big) - \theta_t \end{bmatrix} \tag{4.129}$$

and

$$\mathbf{R}_t = \text{diag}(\sigma_r^2, \sigma_\phi^2, \ldots, \sigma_r^2, \sigma_\phi^2) \in \mathbb{R}^{2L \times 2L} \tag{4.130}$$

Note that the observation model is very similar to the one in the Localisation case, the only differences being grouping of $\mathbf{x}, \mathbf{M}$ into $\mathbf{x}^*$.

4.9.5. Simultaneous Localisation and Mapping (SLAM)

In the SLAM problem, we assume we don't have the knowledge of neither the map $\mathbf{M}$, nor the vehicle states $\mathbf{x}_1, \ldots, \mathbf{x}_T$. We want to figure out $p(\mathbf{x}_t, \mathbf{M} \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t})$. The graphical model is as follows

The graphical model is almost similar to the one in the case of mapping (Figure 4.5), the only difference being that $\mathbf{x}_1, \ldots, \mathbf{x}_T$ are unobserved. We also group the $\mathbf{x}$'s and $\mathbf{M}$ together, for $t = 1, \ldots, T$:

$$\mathbf{x}_t^* = \begin{bmatrix} \mathbf{x}_t \\ \mathbf{M} \end{bmatrix} \tag{4.131}$$

Similarly to the mapping case, the model can be described, for $t = 1, \ldots, T$, by the following equations:

$$\mathbf{x}_t^* = \mathbf{g}^*(\mathbf{x}_{t-1}^*, \mathbf{u}_t) + \boldsymbol{\epsilon}_t^* \tag{4.132}$$

$$\mathbf{z}_t = \mathbf{h}^*(\mathbf{x}_t^*, \mathbf{u}_t) + \boldsymbol{\delta}_t^* \tag{4.133}$$

We need to characterise $\mathbf{g}^*(\mathbf{x}^*, \mathbf{u})$, $\boldsymbol{\epsilon}_t^*$, $\mathbf{h}^*(\mathbf{x}^*, \mathbf{u})$, and $\boldsymbol{\delta}_t^*$ to describe the EKF algorithm fully.

The transition model

The transition model is defined for (part of) the state

$$\mathbf{x}_t = \begin{bmatrix} x_t \\ y_t \\ \theta_t \end{bmatrix} \tag{4.134}$$

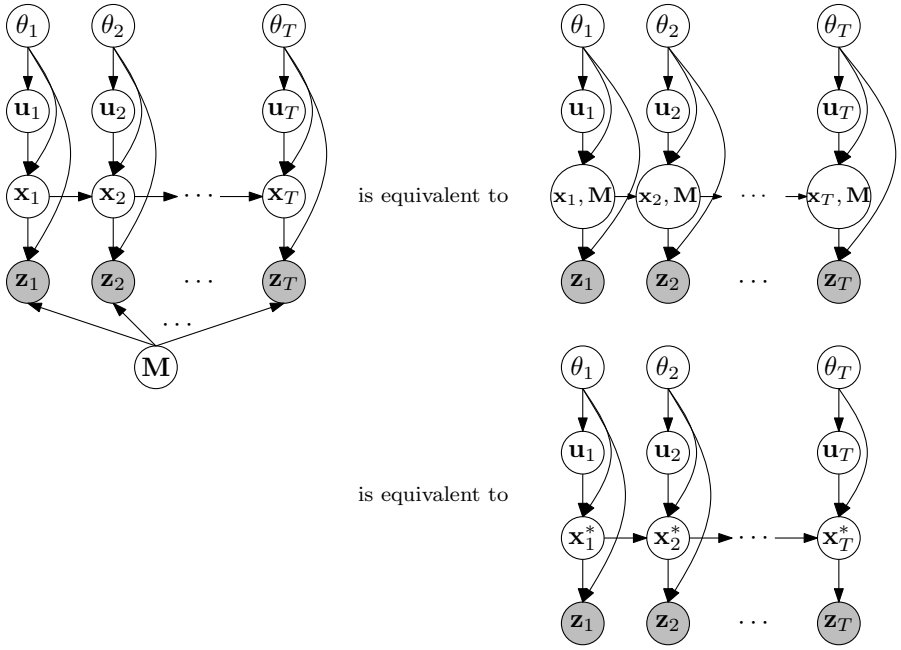

Figure 4.6.: Probabilistic Graphical Model for the Kalman Filter for SLAM.

which contains the position and orientation of the vehicle (note the slightly overloaded notation of $\mathbf{x}_t$ and x_t). The transition model is then described by

$$\mathbf{g}^*(\mathbf{x}^*_{t-1}, \mathbf{u}_t) = \mathbf{x}^*_{t-1} + \begin{bmatrix} -\frac{v_t}{\omega_t}\sin(\theta_{t-1}) + \frac{v_t}{\omega_t}\sin(\theta_{t-1} + \omega_t \Delta t) \\ \frac{v_t}{\omega_t}\cos(\theta_{t-1}) - \frac{v_t}{\omega_t}\cos(\theta_{t-1} + \omega_t \Delta t) \\ \omega_t \Delta t \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{4.135}$$

where $\mathbf{u}_t = (v_t, \omega_t)^T$ contains translational and rotational velocities. The zeros arise because the map doesn't change.

The observation model

The observation model is defined for the observation variable is the same as the one in the Localisation case, described in the Subsubsection 4.9.3, in Equations (4.119) and

(4.121). The observation model is then

$$\mathbf{h}^*(\mathbf{x}_t^*, \mathbf{u}_t) = \begin{bmatrix} \|\mathbf{m}_{c_{t,1}} - (x_t, y_t)^T\| \\ \arctan\left((m_{c_{t,1},y} - y_t)/(m_{c_{t,1},x} - x_t)\right) - \theta_t \\ \vdots \\ \|\mathbf{m}_{c_{t,\ell}} - (x_t, y_t)^T\| \\ \arctan\Big((m_{c_{t,\ell},y} - y_t)/(m_{c_{t,\ell},x} - x_t)\Big) - \theta_t \\ \vdots \\ \|\mathbf{m}_{c_{t,L}} - (x_t, y_t)^T\| \\ \arctan\Big((m_{c_{t,L},y} - y_t)/(m_{c_{t,L},x} - x_t)\Big) - \theta_t \end{bmatrix} \tag{4.136}$$

and

$$\mathbf{R}_t = \text{diag}(\sigma_r^2, \sigma_\phi^2, \ldots, \sigma_r^2, \sigma_\phi^2) \in \mathbb{R}^{2L \times 2L} \tag{4.137}$$

Note that the observation model is very similar to the one in the Localisation case, the only differences being grouping of $\mathbf{x}, \mathbf{M}$ into $\mathbf{x}^*$.

4.10. Principal components analysis

4.10.1. Classical PCA

We have data points $\left\{\mathbf{x}_n, \mathbf{x}_n \in \mathbb{R}^D\right\}, n = 1, \ldots, N$. The goal is to project to a lower dimensional space with dimension $M, M < D$, while maximising the variance to get data points in the *principal space*, $\left\{\mathbf{z}_n, \mathbf{z}_n \in \mathbb{R}^M\right\}, n = 1, \ldots, N$. Let the *principal components* be $\left\{\mathbf{u}_m, \mathbf{u}_m \in \mathbb{R}^D, \|\mathbf{u}_m\| = 1\right\}, m = 1, \ldots, M$. The projected data can be expressed as

$$\begin{aligned} \mathbf{z}_n &= \begin{bmatrix} \mathbf{u}_1^T \mathbf{x}_n \\ \vdots \\ \mathbf{u}_M^T \mathbf{x}_n \end{bmatrix} \\ &= \mathbf{U}^T \mathbf{x}_n \end{aligned}$$

for $n = 1, \ldots, N$ where $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_M]$.

The total variance we are trying to maximise, i.e. the sum of variances along the dimensions $\{\mathbf{u}_m\}$ is

$$\begin{aligned} V &= \sum_{m=1}^{M} \text{var(dimension } m) \\ &= \sum_{m=1}^{M} \frac{1}{N} \sum_{n=1}^{N} (z_{nm} - \bar{z}_m)^2 \\ &\left(\text{where } \bar{z}_m = \frac{1}{N} \sum_{n=1}^{N} z_{nm}\right) \end{aligned} \tag{4.138}$$

$$= \frac{1}{N}\sum_{m=1}^{M}\sum_{n=1}^{N}\left(z_{nm}^2 - 2z_{nm}\bar{z}_m + \bar{z}_m^2\right)$$

$$= \frac{1}{N}\sum_{m=1}^{M}\sum_{n=1}^{N}\left(\left(\mathbf{u}_m^T\mathbf{x}_n\right)^2 - 2\left(\mathbf{u}_m^T\mathbf{x}_n\right)\left(\mathbf{u}_m^T\bar{\mathbf{x}}\right) + \left(\mathbf{u}_m^T\bar{\mathbf{x}}\right)^2\right), \text{where } \bar{\mathbf{x}} = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n$$

$$= \sum_{m=1}^{M}\mathbf{u}_m^T\left(\frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^T - 2\mathbf{x}_n\bar{\mathbf{x}}^T + \bar{\mathbf{x}}\bar{\mathbf{x}}^T\right)\mathbf{u}_m$$

$$= \sum_{m=1}^{M}\mathbf{u}_m^T\left(\frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^T\right)\mathbf{u}_m$$

$$= \sum_{m=1}^{M}\mathbf{u}_m^T\mathbf{S}\mathbf{u}_m \tag{4.139}$$

$$\left(\text{where } \mathbf{S} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^T\right) \tag{4.140}$$

We want to maximise this with the constraint $\|\mathbf{u}_m\| = 1, m = 1, \ldots, M$ which is equivalent to $\mathbf{u}_m^T\mathbf{u}_m = 1, m = 1, \ldots, M$. We use Lagrange multipliers $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_M)$. Hence we need to maximise the following Lagrangian

$$\mathcal{L}(\boldsymbol{\lambda}, \mathbf{u}_1, \ldots, \mathbf{u}_M) = \sum_{m=1}^{M}\mathbf{u}_m^T\mathbf{S}\mathbf{u}_m + \boldsymbol{\lambda}^T\begin{bmatrix}1 - \mathbf{u}_1^T\mathbf{u}_1\\ \vdots \\ 1 - \mathbf{u}_M^T\mathbf{u}_M\end{bmatrix}$$

We know that $\mathbf{S}$ is positive semi-definite because it is a covariance matrix for $\{\mathbf{x}_n\}$. The term $\mathbf{u}_m^T\mathbf{S}\mathbf{u}_m$ is convex w.r.t. $\mathbf{u}_m$ because the Hessian $2\mathbf{S}$ is positive semi-definite. Hence $\sum_{m=1}^{M}\mathbf{u}_m^T\mathbf{S}\mathbf{u}_m$ must be convex w.r.t. $(\mathbf{u}_1, \ldots, \mathbf{u}_M)$. Also, the second term in the Lagrangian is convex w.r.t. the principal components. Hence, we can maximise the Lagrangian by setting the gradients to zero:

$$\text{grad}_{\boldsymbol{\lambda}}\,\mathcal{L} = \mathbf{0} \tag{4.141}$$

$$\text{grad}_{\mathbf{u}_m}\,\mathcal{L} = \mathbf{0}, m = 1, \ldots, M \tag{4.142}$$

From (4.141), we obtain $\mathbf{u}_m^T\mathbf{u}_m = 1, m = 1, \ldots, M$. From (4.142), we obtain

$$\text{grad}_{\mathbf{u}_m}\,\mathcal{L} = 2\mathbf{S}\mathbf{u}_m - 2\lambda_m\mathbf{u}_m \tag{4.143}$$

$$= 0 \tag{4.144}$$

$$\implies \mathbf{S}\mathbf{u}_m = \lambda_m\mathbf{u}_m \tag{4.145}$$

Thus we can see that $\{\mathbf{u}_m\}$ should be selected to be the eigenvectors corresponding to the eigenvalues $\{\lambda_m\}$ of $\mathbf{S}$. If we premultiply (4.145) by $\mathbf{u}_m^T$, we get $\lambda_m = \mathbf{u}_m^T\mathbf{S}\mathbf{u}_m$ which can be substituted back to total variance

$$V = \sum_{m=1}^{M}\lambda_m$$

from which we can see that to maximise, we set $\{\lambda_m\}$ to be the largest M eigenvalues of $\mathbf{S}$. The principal components $\{\mathbf{u}_m\}$ are the corresponding eigenvectors.

4.10.2. Probabilistic PCA

Following the mixture model, where $\mathbf{Z} = \left\{\mathbf{z}_n, \mathbf{z}_n \in \mathbb{R}^M\right\}$, $n = 1, \ldots, N$ are the latent variables and $\mathbf{X} = \left\{\mathbf{x}_n, \mathbf{x}_n \in \mathbb{R}^D\right\}$, $n = 1, \ldots, N$ are the observed variables, probabilistic PCA assumes $\mathbb{R}^M$ is the lower-dimensional space we want to project our data in $\mathbb{R}^D$ to. We have the following assumptions:

$$\begin{aligned} p(\mathbf{z}) &= \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I}) \\ p(\mathbf{x} \mid \mathbf{z}) &= \mathcal{N}(\mathbf{x}; \mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2\mathbf{I}) \end{aligned}$$

where $\mathbf{0}, \mathbf{I}, \mathbf{W}, \boldsymbol{\mu}, \mathbf{I}$ all have the appropriate dimensions. Note that the model is parameterised by $\boldsymbol{\theta} = (\mathbf{W}, \boldsymbol{\mu}, \sigma^2)$. Following Subsection 2.4.1, we can express the remaining marginal and conditional as

$$\begin{aligned} p(\mathbf{x}) &= \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \mathbf{C}) \\ p(\mathbf{z} \mid \mathbf{x}) &= \mathcal{N}(\mathbf{z}; \mathbf{M}^{-1}\mathbf{W}^T(\mathbf{x} - \boldsymbol{\mu}), \sigma^2\mathbf{M}^{-1}) \end{aligned}$$

where

$$\begin{aligned} \mathbf{C} &= \mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I} \\ \mathbf{M} &= \mathbf{W}^T\mathbf{W} + \sigma^2\mathbf{I} \end{aligned}$$

MLE for probabilistic PCA

To find ML estimates for our model, we want to maximise the following likelihood function:

$$\begin{aligned} p(\mathcal{D} \mid \boldsymbol{\theta}) &= \prod_{n=1}^{N} p(\mathbf{x}_n \mid \boldsymbol{\theta}) \\ &= \prod_{n=1}^{N} \mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}, \mathbf{C}) \end{aligned}$$

Maximising this w.r.t. the parameters $\mathbf{W}$ and σ^2, we get the following MLEs:

$$\begin{aligned} \mathbf{W}_{ML} &= \mathbf{U}_M \left(\mathbf{L}_M - \sigma^2\mathbf{I}\right)^{1/2} \mathbf{R} \\ \sigma^2_{ML} &= \frac{1}{D - M} \sum_{i=M+1}^{D} \lambda_i \end{aligned}$$

where $\mathbf{R}, \mathbf{R} \in \mathbb{R}^{M \times M}, \mathbf{R}\mathbf{R}^T = \mathbf{I}$ is an arbitrary orthogonal matrix and

$$\mathbf{U}_M = [\mathbf{u}_1, \ldots, \mathbf{u}_M]$$

$$\mathbf{L}_M = \text{diag}(\lambda_1, \ldots, \lambda_M)$$

where $\mathbf{u}_1, \ldots, \mathbf{u}_D$ and $\lambda_1, \ldots, \lambda_D$ are eigenvectors and eigenvalues of the data covariance matrix $\mathbf{S}$ (defined below in (4.140)), sorted in descending order.

Other stuff to note

Alternative view. fdsaf a

Intuitive view. fsda

Redundancy in parameterisation. f ds

Computational complexity. fsdaf

EM algorithm for probabilistic PCA

The EM algorithm to find MLE for probabilistic PCA is as follows

Algorithm 4 EM algorithm for probabilistic PCA

1: Initialise $\boldsymbol{\theta}^{\text{new}} = (\mathbf{W}^{\text{new}}, (\sigma^{\text{new}})^2)$. Set $\boldsymbol{\mu}_{MLE} = \bar{\mathbf{x}}$.
2: **repeat**
3: $\quad \boldsymbol{\theta}^{\text{old}} \leftarrow \boldsymbol{\theta}^{\text{new}}$
4: $\quad$ Set ▷ E step

$$\mathbb{E}[\mathbf{z}_n] = \left(\mathbf{M}^{\text{old}}\right)^{-1} \left(\mathbf{W}^{\text{old}}\right)^T (\mathbf{x}_n - \bar{\mathbf{x}})$$
$$\mathbb{E}\left[\mathbf{z}_n \mathbf{z}_n^T\right] = \left(\sigma^{\text{old}}\right)^2 \left(\mathbf{M}^{\text{old}}\right)^{-1} + \mathbb{E}[\mathbf{z}_n]\, \mathbb{E}[\mathbf{z}_n]^T$$

where $\mathbf{M} = \mathbf{W}^T\mathbf{W} + \sigma^2\mathbf{I}$.
5: $\quad$ Set ▷ M step

$$\begin{aligned} \mathbf{W}^{\text{new}} &= \left[\sum_n (\mathbf{x}_n - \bar{\mathbf{x}})\, \mathbb{E}[\mathbf{z}_n]^T\right] \left[\sum_n \mathbb{E}\left[\mathbf{z}_n \mathbf{z}_n^T\right]\right]^{-1} \\ (\sigma^{\text{new}})^2 &= \frac{1}{ND} \sum_n \|\mathbf{x}_n - \bar{\mathbf{x}}\|^2 - 2\, \mathbb{E}[\mathbf{z}_n]^T \left(\mathbf{W}^{\text{new}}\right)^T (\mathbf{x}_n - \bar{\mathbf{x}}) \\ &\quad + \text{Tr}\left(\mathbb{E}\left[\mathbf{z}_n \mathbf{z}_n^T\right] (\mathbf{W}^{\text{new}})^T \mathbf{W}^{\text{new}}\right) \end{aligned} \tag{4.146}$$

6: **until** convergence.

Bayesian PCA

4.11. Factor analysis

4.12. Independent components analysis

5. Sampling algorithms

5.1. Introduction

Let p be a probability distribution with a pdf $p(\mathbf{x}), \mathbf{x} \in \mathcal{X}$ (usually $\mathcal{X} = \mathbb{R}^D, D \in \mathbb{N}$), which we assume can be evaluated within a multiplicative factor (i.e. we can only evaluate $p^*(\mathbf{x}) = Z_p p(\mathbf{x})$, where $Z_p = \int_{\mathcal{X}} p^*(\mathbf{x})\,\mathrm{d}\mathbf{x}$). We want to achieve the following:

Problem 1 Generate samples $\{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(R)}\}, R \in \mathbb{N}$ (we will use the shorthand notation $\{\mathbf{x}^{(r)}\}$ from now) from the probability distribution p.

Problem 2 Estimate the expectation of an arbitrary function f given $\mathbf{x} \sim p$, $\mathbb{E}_{\mathbf{x}\sim p}[f(\mathbf{x})]$ (we will use the shorthand notation $\mathbb{E}[f]$ from now).

5.2. Rejection sampling

Assume we can sample from a proposal distribution q with a pdf $q(\mathbf{x})$, which can be evaluated within a multiplicative factor (i.e. we can only evaluate $q^*(\mathbf{x}) = Z_q q(\mathbf{x})$). Also assume we know the value of a constant c such that

$$cq^*(\mathbf{x}) > p^*(\mathbf{x}) \text{ for all } \mathbf{x} \tag{5.1}$$

The procedure that generates a sample $\mathbf{x} \sim p$ is described in Algorithm 5 below.

Algorithm 5 Rejection sampling

1: Generate $\mathbf{x} \sim q$.
2: Generate $u \sim \mathrm{Unif}(0, cq^*(\mathbf{x}))$.
3: If $u > p^*(\mathbf{x})$ it is rejected, otherwise it is accepted.

5.2.1. Why it works?

Assume $\mathbf{x} \in \mathbb{R}^D$. Define sets $\mathcal{X}$ and $\mathcal{X}'$ to be

$$\mathcal{X} = \left\{\boldsymbol{\alpha} \in \mathbb{R}^{d+1} : \alpha_{1:d} \in \mathbb{R}^d, \alpha_{d+1} \in [0, cq^*(\boldsymbol{\alpha})]\right\} \tag{5.2}$$

$$\mathcal{X}' = \left\{\boldsymbol{\alpha} \in \mathbb{R}^{d+1} : \alpha_{1:d} \in \mathbb{R}^d, \alpha_{d+1} \in [0, p^*(\boldsymbol{\alpha})]\right\} \tag{5.3}$$

Note that $\mathcal{X}' \subseteq \mathcal{X}$.

By definition, $\mathcal{X}$ is the support of $(\mathbf{x}, u)$. The probability of $(\mathbf{x}, u)$ can be expressed as

$$\Pr(\mathbf{x}, u) = \Pr(\mathbf{x}) \Pr(u) \tag{5.4}$$

$$= q(\mathbf{x})\frac{1}{cq^*(\mathbf{x})} \tag{5.5}$$

$$= q(\mathbf{x})\frac{1}{cZ_q q(\mathbf{x})} \tag{5.6}$$

$$= \frac{1}{cZ_q} \tag{5.7}$$

which is constant w.r.t. $(\mathbf{x}, u)$, i.e.

$$(\mathbf{x}, u) \sim \text{Unif}(\mathcal{X}) \tag{5.8}$$

Let $(\mathbf{x}', u')$ be the value of $(\mathbf{x}, u)$ that gets accepted. By definition, $\mathcal{X}'$ is the support of $(\mathbf{x}', u')$:

$$(\mathbf{x}', u') = \begin{cases} (\mathbf{x}, u) & \text{if } (\mathbf{x}, u) \in \mathcal{X}' \\ \text{nothing} & \text{otherwise.} \end{cases} \tag{5.9}$$

The probability of $(\mathbf{x}', u')$ can be expressed as

$$\Pr(\mathbf{x}', u') = \begin{cases} \Pr(\mathbf{x}, u) & \text{if } (\mathbf{x}, u) \in \mathcal{X}' \\ 0 & \text{otherwise.} \end{cases} \tag{5.10}$$

which means

$$(\mathbf{x}', u') \sim \text{Unif}(\mathcal{X}') \tag{5.11}$$

Working backwards

$$\Pr(\mathbf{x}') = \frac{\Pr(\mathbf{x}', u')}{\Pr(u')} \tag{5.12}$$

$$\propto \frac{1}{1/p^*(\mathbf{x}')} \tag{5.13}$$

$$\propto p^*(\mathbf{x}') \tag{5.14}$$

Hence the accepted $\mathbf{x}$, $\mathbf{x}'$ is $\sim p$.

5.3. Importance sampling

Assume we can sample from a proposal distribution q with a pdf $q(\mathbf{x})$, which can be evaluated within a multiplicative factor (i.e. we can only evaluate $q^*(\mathbf{x}) = Z_q q(\mathbf{x})$). To solve problem 2, we follow Algorithm 6 below.

Algorithm 6 Importance sampling

1: Generate samples from q, $\{\mathbf{x}^{(r)}\}$.
2: Calculate importance weights $w_r = \frac{p^*(\mathbf{x}^{(r)})}{q^*(\mathbf{x}^{(r)})}$.
3: $\hat{\mathbf{y}} = \frac{\sum_r w_r f(\mathbf{x}^{(r)})}{\sum_r w_r}$ is the estimator of $\mathbb{E}[f]$.

5.3.1. Convergence of estimator as R increases

We want to prove that if $q(\mathbf{x})$ is non-zero for all $\mathbf{x}$ where $p(\mathbf{x})$ is non-zero, the estimator $\hat{\mathbf{y}}$ converges to $\mathbb{E}[f]$, as R increases. We consider the the expectations of the numerator and denominator separately:

$$\mathbb{E}_q[\text{numer}] = \mathbb{E}_q\left[\sum_r w_r f(\mathbf{x}^{(r)})\right] \tag{5.15}$$
$$= \sum_r \mathbb{E}_q\left[w_r f(\mathbf{x}^{(r)})\right] \tag{5.16}$$
$$= \sum_r \mathbb{E}_q\left[\frac{p^*(\mathbf{x}^{(r)})}{q^*(\mathbf{x}^{(r)})} f(\mathbf{x}^{(r)})\right] \tag{5.17}$$
$$= \sum_r \mathbb{E}_q\left[\frac{Z_p p(\mathbf{x}^{(r)})}{Z_q q(\mathbf{x}^{(r)})} f(\mathbf{x}^{(r)})\right] \tag{5.18}$$
$$= \frac{Z_p}{Z_q} \sum_r \int_{\mathbf{x}^{(r)}} p(\mathbf{x}^{(r)}) f(\mathbf{x}^{(r)}) \, \mathrm{d}\mathbf{x}^{(r)} \tag{5.19}$$
$$= \frac{Z_p}{Z_q} \sum_r \mathbb{E}_p\left[f(\mathbf{x}^{(r)})\right] \tag{5.20}$$
$$= \frac{Z_p}{Z_q} R \, \mathbb{E}_p\left[f(\mathbf{x})\right] \tag{5.21}$$
$$\mathbb{E}_q[\text{denom}] = \mathbb{E}_q\left[\sum_r w_r\right] \tag{5.22}$$
$$= \sum_r \mathbb{E}_q\left[\frac{p^*(\mathbf{x}^{(r)})}{q^*(\mathbf{x}^{(r)})}\right] \tag{5.23}$$
$$= \sum_r \mathbb{E}_q\left[\frac{Z_p p(\mathbf{x}^{(r)})}{Z_q q(\mathbf{x}^{(r)})}\right] \tag{5.24}$$
$$= \frac{Z_p}{Z_q} \sum_r \int_{\mathbf{x}^{(r)}} p(\mathbf{x}^{(r)}) \, \mathrm{d}\mathbf{x}^{(r)} \tag{5.25}$$
$$= \frac{Z_p}{Z_q} R \tag{5.26}$$

Hence $\hat{\mathbf{y}}$ converges to $\mathbb{E}_p[f]$ as R increases (but is not necessarily an unbiased estimator because $\mathbb{E}_q[\hat{\mathbf{y}}]$ is not necessarily $= \mathbb{E}_p[f]$).

5.3.2. Optimal proposal distribution

Assuming we can evaluate $p(\mathbf{x})$ and $q(\mathbf{x})$, we want to find a proposal distribution q to minimise the variance of the weighted samples

$$\mathrm{var}_q\left[\frac{p(\mathbf{x})}{q(\mathbf{x})} f(\mathbf{x})\right] = \mathbb{E}_q\left[\frac{p^2(\mathbf{x})}{q^2(\mathbf{x})} f^2(\mathbf{x})\right] - \left(\mathbb{E}_q\left[\frac{p(\mathbf{x})}{q(\mathbf{x})} f(\mathbf{x})\right]\right)^2 \tag{5.27}$$

$$= \mathbb{E}_q \left[\frac{p^2(\mathbf{x})}{q^2(\mathbf{x})} f^2(\mathbf{x}) \right] - \left(\mathbb{E}_p \left[f(\mathbf{x}) \right] \right)^2 \tag{5.28}$$

The second part is independent of q so we can ignore it. By Jensen's inequality, we have $\mathbb{E}\left[g(u(\mathbf{x}))\right] \geq g\left(\mathbb{E}\left[u(\mathbf{x})\right]\right)$ for $u(\mathbf{x}) \geq 0$ where $g : x \mapsto x^2$. Setting $u(\mathbf{x}) = p(\mathbf{x})|f(\mathbf{x})|/q(\mathbf{x})$, we have the following lower bound:

$$\mathbb{E}_q \left[\frac{p^2(\mathbf{x})}{q^2(\mathbf{x})} f^2(\mathbf{x}) \right] \geq \left(\mathbb{E}_q \left[\frac{p(\mathbf{x})}{q(\mathbf{x})} |f(\mathbf{x})| \right] \right)^2 = \left(\mathbb{E}_p [|f(\mathbf{x})|] \right)^2 \tag{5.29}$$

with the equality when $u(\mathbf{x}) = \text{const.} \implies q_{\text{optimal}}(\mathbf{x}) \propto |f(\mathbf{x})| p(\mathbf{x})$. Taking care of normalisation, we get

$$q_{\text{optimal}}(\mathbf{x}) = \frac{|f(\mathbf{x})| p(\mathbf{x})}{\int |f(\mathbf{x}')| p(\mathbf{x}') \, \mathrm{d}\mathbf{x}'} \tag{5.30}$$

5.4. Sampling importance resampling

In Sampling importance resampling (SIR), we approximate the pdf of p as point masses and resample from them to get samples approximately $\sim p$. The process is described in Algorithm 7 below.

Algorithm 7 Sampling importance resampling

1: Generate samples $\left\{ \mathbf{x}^{(r)} \right\}$ from q.

2: Calculate importance weights $\left\{ w_r = \frac{p^*(\mathbf{z}^{(r)})}{q^*(\mathbf{z}^{(r)})} \right\}$.

3: Calculate the normalised importance weights $\left\{ \hat{w}_r = \frac{w_r}{\sum_{r'} w_{r'}} \right\}$. Note that $\sum_r \hat{w}_r = 1$.

4: We can resample from

$$\hat{p}(\mathrm{d}\mathbf{x}) = \sum_r \hat{w}_r \delta_{\mathbf{x}^{(r)}}(\mathrm{d}\mathbf{x}) \tag{5.31}$$

to estimate sampling from $p(\mathbf{x})$.

5.4.1. Why it works?

We consider the univariate case (to do: general case) as the number of proposal samples (particles) $R \to \infty$. We can express the number of proposal samples that are in the interval $\lim_{\delta x \to 0}[x, x + \delta x]$, $N(x)$, to be

$$N(x) = \lim_{\delta x \to 0} R q(x) \delta x \tag{5.32}$$

We can express the probability of the one final sample, $x^{(r)}$ being in the interval $\lim_{\delta x \to 0}[x, x + \delta x]$ to be

$$\lim_{\delta x \to 0} \Pr(x \leq x^{(r)} \leq x + \delta x) = N(x) \hat{w}_r \tag{5.33}$$

$$\propto \lim_{\delta x \to 0} R q(x) \delta x \frac{p(x)}{q(x)} \tag{5.34}$$

$$\propto \lim_{\delta x \to 0} p(x)\delta x \tag{5.35}$$

Hence (to do: why exactly does that result in an integral)

$$\Pr(a \leq x^{(r)} \leq b) \propto \int_a^b p(x)\,\mathrm{d}x \tag{5.36}$$

$$\implies x^{(r)} \sim p \tag{5.37}$$

5.5. Particle filtering

5.5.1. Sequential importance sampling (SIS)

Assume the probabilistic graphical model similar to the one in HMMs, where

- $\mathbf{x}_t, \mathbf{x}_t \subset \mathcal{X}^D$ and $\mathbf{y}_t, \mathbf{y}_t \subset \mathcal{Y}^D$ are the hidden and observed random variables at time $t, t = 1, \ldots, T$.
- The initial state is characterised by $\mathbf{x}_1 \sim \mu(\cdot \mid \boldsymbol{\theta})$ for some known parameter $\boldsymbol{\theta} \subset \Theta$.
- The transitions are characterised by $\mathbf{x}_t \mid \mathbf{x}_{t-1} \sim f(\cdot \mid \mathbf{x}_{t-1}; \boldsymbol{\theta})$.
- The emmissions are characterised by $\mathbf{y}_t \mid \mathbf{x}_t \sim g(\cdot \mid \mathbf{x}_t; \boldsymbol{\theta})$.

We want to sample from the distribution $p(\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}; \boldsymbol{\theta})$. Assume we can sample from the probability distribution with the pdf of the following form

$$q(\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}; \boldsymbol{\theta}) = q(\mathbf{x}_t \mid \mathbf{x}_{1:t-1}, \mathbf{y}_{1:t}; \boldsymbol{\theta}) q(\mathbf{x}_{1:t-1} \mid \mathbf{y}_{1:t}; \boldsymbol{\theta}) \tag{5.38}$$

$$= q(\mathbf{x}_t \mid \mathbf{x}_{1:t-1}, \mathbf{y}_{1:t}; \boldsymbol{\theta}) q(\mathbf{x}_{1:t-1} \mid \mathbf{y}_{1:t-1}; \boldsymbol{\theta}) \tag{5.39}$$

$$= q(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{y}_t; \boldsymbol{\theta}) \tag{5.40}$$

If we express the pdf of p for $t = 1, \ldots, T$ in the form of (for convenience, we drop the conditional dependence on $\boldsymbol{\theta}$):

$$p(\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_{1:t} \mid \mathbf{x}_{1:t}) p(\mathbf{x}_{1:t})}{p(\mathbf{y}_{1:t})} \tag{5.41}$$

$$= \frac{p(\mathbf{y}_t \mid \mathbf{x}_{1:t}, \mathbf{y}_{1:t-1}) p(\mathbf{y}_{1:t-1} \mid \mathbf{x}_{1:t}) p(\mathbf{x}_{1:t})}{p(\mathbf{y}_t \mid \mathbf{y}_{1:t-1}) p(\mathbf{y}_{1:t-1})} \tag{5.42}$$

$$= \frac{p(\mathbf{y}_t \mid \mathbf{x}_{1:t}, \mathbf{y}_{1:t-1}) p(\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t-1})}{p(\mathbf{y}_t \mid \mathbf{y}_{1:t-1})} \tag{5.43}$$

$$= \frac{p(\mathbf{y}_t \mid \mathbf{x}_{1:t}, \mathbf{y}_{1:t-1}) p(\mathbf{x}_t \mid \mathbf{x}_{1:t-1}, \mathbf{y}_{1:t-1}) p(\mathbf{x}_{1:t-1} \mid \mathbf{y}_{1:t-1})}{p(\mathbf{y}_t \mid \mathbf{y}_{1:t-1})} \tag{5.44}$$

$$= \frac{p(\mathbf{y}_t \mid \mathbf{x}_t) p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) p(\mathbf{x}_{1:t-1} \mid \mathbf{y}_{1:t-1})}{p(\mathbf{y}_t \mid \mathbf{y}_{1:t-1})} \tag{5.45}$$

$$\propto p(\mathbf{y}_t \mid \mathbf{x}_t) p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) p(\mathbf{x}_{1:t-1} \mid \mathbf{y}_{1:t-1}) \tag{5.46}$$

$$= g(\mathbf{y}_t \mid \mathbf{x}_t) f(\mathbf{x}_t \mid \mathbf{x}_{t-1}) p(\mathbf{x}_{1:t-1} \mid \mathbf{y}_{1:t-1}) \tag{5.47}$$

we can write the weight of the sample $\mathbf{x}_{1:t}^{(r)}$ from the proposal q to be

$$w_t^{(r)} \propto \frac{p\left(\mathbf{x}_{1:t}^{(r)} \mid \mathbf{y}_{1:t}\right)}{q\left(\mathbf{x}_{1:t}^{(r)} \mid \mathbf{y}_{1:t}\right)} \tag{5.48}$$

$$\propto \frac{p\left(\mathbf{y}_t \mid \mathbf{x}_t^{(r)}\right) p\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}\right) p\left(\mathbf{x}_{1:t-1}^{(r)} \mid \mathbf{y}_{1:t-1}\right)}{q\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t\right) q\left(\mathbf{x}_{1:t-1}^{(r)} \mid \mathbf{y}_{1:t-1}\right)} \tag{5.49}$$

$$= w_{t-1}^{(r)} \frac{p\left(\mathbf{y}_t \mid \mathbf{x}_t^{(r)}\right) p\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}\right)}{q\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t\right)} \tag{5.50}$$

$$= w_{t-1}^{(r)} \frac{g\left(\mathbf{y}_t \mid \mathbf{x}_t^{(r)}\right) f\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}\right)}{q\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t\right)} \tag{5.51}$$

For $t = 1$

$$w_1^{(r)} \propto \frac{p\left(\mathbf{x}_1^{(r)} \mid \mathbf{y}_1\right)}{q\left(\mathbf{x}_1^{(r)} \mid \mathbf{y}_1\right)} \tag{5.52}$$

$$\propto \frac{p\left(\mathbf{x}_1^{(r)}, \mathbf{y}_1\right)}{q\left(\mathbf{x}_1^{(r)} \mid \mathbf{y}_1\right)} \tag{5.53}$$

$$\propto \frac{p\left(\mathbf{y}_1 \mid \mathbf{x}_1^{(r)}\right) p\left(\mathbf{x}_1^{(r)}\right)}{q\left(\mathbf{x}_1^{(r)} \mid \mathbf{y}_1\right)} \tag{5.54}$$

$$= \frac{g\left(\mathbf{y}_1 \mid \mathbf{x}_1^{(r)}\right) \mu\left(\mathbf{x}_1^{(r)}\right)}{q\left(\mathbf{x}_1^{(r)} \mid \mathbf{y}_1\right)} \tag{5.55}$$

Note that second line is proportional to the first line with respect to $p(\mathbf{y}_1)$ which is justifiable because the the constant of proportionality cancels out during the normalisation step. The algorithm for SIS is shown in Algorithm 8 below.

Algorithm 8 Sequential importance sampling

1: Sample from proposal ▷ Initialisation

$$\mathbf{x}_1^{(r)} \sim q\left(\cdot \mid \mathbf{y}_1^{(r)}; \boldsymbol{\theta}\right), r = 1, \ldots, R \tag{5.56}$$

2: Compute weights

$$w_1^{(r)} \propto \frac{g\left(\mathbf{y}_1 \mid \mathbf{x}_1^{(r)}\right) \mu\left(\mathbf{x}_1^{(r)}\right)}{q\left(\mathbf{x}_1^{(r)} \mid \mathbf{y}_1\right)}, r = 1, \ldots, R \tag{5.57}$$

3: Normalise weights

$$\hat{w}_1^{(r)} = \frac{w_1^{(r)}}{\sum_{r'} w_1^{(r')}}, r = 1, \ldots, R \tag{5.58}$$

4: We can resample from

$$\hat{p}(\mathrm{d}\mathbf{x}_1 \mid \mathbf{y}_1; \boldsymbol{\theta}) = \sum_r \hat{w}_1^{(r)} \delta_{\mathbf{x}_1^{(r)}}(\mathrm{d}\mathbf{x}_1) \tag{5.59}$$

to estimate

$$p(\mathbf{x}_1 \mid \mathbf{y}_1; \boldsymbol{\theta}) \tag{5.60}$$

5: **for** $t = 2, \ldots, T$ **do** ▷ Main loop

6: Sample from proposal

$$\mathbf{x}_t^{(r)} \sim q\left(\cdot \mid \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t; \boldsymbol{\theta}\right), r = 1, \ldots, R \tag{5.61}$$

7: Compute weights

$$w_t^{(r)} \propto w_{t-1}^{(r)} \frac{g\left(\mathbf{y}_t \mid \mathbf{x}_t^{(r)}; \boldsymbol{\theta}\right) f\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}; \boldsymbol{\theta}\right)}{q\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t; \boldsymbol{\theta}\right)}, r = 1, \ldots, R \tag{5.62}$$

8: Normalise weights

$$\hat{w}_t^{(r)} = \frac{w_t^{(r)}}{\sum_{r'} w_t^{(r')}}, r = 1, \ldots, R \tag{5.63}$$

9: We can resample from

$$\hat{p}(\mathrm{d}\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}; \boldsymbol{\theta}) = \sum_r \hat{w}_t^{(r)} \delta_{\mathbf{x}_{1:t}^{(r)}}(\mathrm{d}\mathbf{x}_{1:t}) \tag{5.64}$$

to estimate

$$p(\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}; \boldsymbol{\theta}) \tag{5.65}$$

The reason why it works is the same as in the case of Sampling importance resampling described in section 5.4.

5.5.2. The degeneracy problem

Because the support of the pdf we are approximating ($p(\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t})$) is growing, the constant number of weights we use (R) won't be sufficient after a while. This is because many weights will become very negligible, wasting our resources. An **effective sample size** is used to measure this degeneracy is defined to be and approximated by the following:

$$S_{\text{eff}} \triangleq \frac{S}{1 + \operatorname{var}\left[w_t^{(r)*}\right]} \tag{5.66}$$

$$\hat{S}_{\text{eff}} \approx \frac{1}{\sum_r \left(w_t^{(r)}\right)^2} \tag{5.67}$$

where $w_t^{(r)*} = p(\mathbf{x}_t^{(r)} \mid \mathbf{y}_{1:t}) / q(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t)$ is the "true weight" of particle r.

There are (among others) two solutions to this problem – introduce the resampling step, and using a good proposal distribution.

5.5.3. The resampling step

Whenever the effective sample size drops below some threshold, resample to get new R samples from the approximation of the pdf. This step is also called **rejuvenation**. The full algorithm for a generic particle filter is shown in Algorithm 9 below in which we resample during every step.

Algorithm 9 Generic particle filter

1: Sample from proposal ▷ Initialisation

$$\mathbf{x}_1^{(r)} \sim q\left(\cdot \mid \mathbf{y}_1^{(r)}; \boldsymbol{\theta}\right), r = 1, \ldots, R \tag{5.68}$$

2: Compute weights

$$w_1^{(r)} \propto \frac{p\left(\mathbf{x}_1^{(r)} \mid \mathbf{y}_1; \boldsymbol{\theta}\right)}{q\left(\mathbf{x}_1^{(r)} \mid \mathbf{y}_1; \boldsymbol{\theta}\right)}, r = 1, \ldots, R \tag{5.69}$$

3: Normalise weights

$$\hat{w}_1^{(r)} = \frac{w_1^{(r)}}{\sum_{r'} w_1^{(r')}}, r = 1, \ldots, R \tag{5.70}$$

4: We can resample from

$$\hat{p}(\mathrm{d}\mathbf{x}_1 \mid \mathbf{y}_1; \boldsymbol{\theta}) = \sum_r \hat{w}_1^{(r)} \delta_{\mathbf{x}_1^{(r)}}(\mathrm{d}\mathbf{x}_1) \tag{5.71}$$

to estimate

$$p(\mathbf{x}_1 \mid \mathbf{y}_1; \boldsymbol{\theta}) \tag{5.72}$$

5: **for** $t = 2, \ldots, T$ **do** ▷ Main loop

6: Sample parents' indices of t^{th} generation

$$A_{t-1}^{(r)} \sim \mathrm{Cat}\left(\hat{w}_{t-1}^{(1)}, \ldots, \hat{w}_{t-1}^{(R)}\right), r = 1, \ldots, R \tag{5.73}$$

7: Sample t^{th} generation using corresponding parents

$$\mathbf{x}_t^{(r)} \sim q\left(\cdot \mid \mathbf{x}_{t-1}^{A_{t-1}^{(r)}}, \mathbf{y}_t; \boldsymbol{\theta}\right), r = 1, \ldots, R \tag{5.74}$$

8: Compute weights

$$w_t^{(r)} \propto w_{t-1}^{(r)} \frac{g\left(\mathbf{y}_t \mid \mathbf{x}_t^{(r)}; \boldsymbol{\theta}\right) f\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{A_{t-1}^{(r)}}; \boldsymbol{\theta}\right)}{q\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{A_{t-1}^{(r)}}, \mathbf{y}_t; \boldsymbol{\theta}\right)}, r = 1, \ldots, R \tag{5.75}$$

9: Normalise weights

$$\hat{w}_t^{(r)} = \frac{w_t^{(r)}}{\sum_{r'} w_t^{(r')}}, r = 1, \ldots, R \tag{5.76}$$

10: We can resample from

$$\hat{p}(\mathrm{d}\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}; \boldsymbol{\theta}) = \sum_r \hat{w}_t^{(r)} \delta_{\mathbf{x}_{1:t}^{(r)}}(\mathrm{d}\mathbf{x}_{1:t}) \tag{5.77}$$

to estimate

$$p(\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}; \boldsymbol{\theta}) \tag{5.78}$$

5.5.4. The proposal distribution

It is common to use the following proposal distribution

$$q\left(\mathbf{x}_{1:t}^{(r)} \mid \mathbf{y}_{1:t}\right) = q\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t\right) \tag{5.79}$$

$$= p\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}\right) \tag{5.80}$$

$$= f\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}\right) \tag{5.81}$$

Hence the weight equation in (5.51) becomes

$$w_t^{(r)} \propto w_{t-1}^{(r)} \frac{g\left(\mathbf{y}_t \mid \mathbf{x}_t^{(r)}\right) f\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}\right)}{q\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t\right)} \tag{5.82}$$

$$= w_{t-1}^{(r)} g\left(\mathbf{y}_t \mid \mathbf{x}_t^{(r)}\right) \tag{5.83}$$

This approach can be **in**efficient because the likelihood, $p\left(\mathbf{y}_t \mid \mathbf{x}_t^{(r)}\right)$, can be very small at many places meaning many of the particles will be very small.

The optimal proposal distribution has the form

$$q\left(\mathbf{x}_{1:t}^{(r)} \mid \mathbf{y}_{1:t}\right) = q\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t\right) \tag{5.84}$$

$$= p\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t\right) \tag{5.85}$$

$$= \frac{p\left(\mathbf{y}_t \mid \mathbf{x}_t, \mathbf{x}_{t-1}^{(r)}\right) p\left(\mathbf{x}_t, \mathbf{x}_{t-1}^{(r)}\right)}{p\left(\mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t\right)} \tag{5.86}$$

$$= \frac{p\left(\mathbf{y}_t \mid \mathbf{x}_t\right) p\left(\mathbf{x}_t \mid \mathbf{x}_{t-1}^{(r)}\right)}{p\left(\mathbf{y}_t \mid \mathbf{x}_{t-1}^{(r)}\right)} \tag{5.87}$$

$$= \frac{g\left(\mathbf{y}_t \mid \mathbf{x}_t\right) f\left(\mathbf{x}_t \mid \mathbf{x}_{t-1}^{(r)}\right)}{p\left(\mathbf{y}_t \mid \mathbf{x}_{t-1}^{(r)}\right)} \tag{5.88}$$

The weight equation in (5.51) becomes

$$w_t^{(r)} \propto w_{t-1}^{(r)} p\left(\mathbf{y}_t \mid \mathbf{x}_{t-1}^{(r)}\right) \tag{5.89}$$

$$= w_{t-1}^{(r)} \int p\left(\mathbf{y}_t, \mathbf{x}'_t \mid \mathbf{x}_{t-1}^{(r)}\right) \mathrm{d}\mathbf{x}' \tag{5.90}$$

$$= w_{t-1}^{(r)} \int p\left(\mathbf{y}_t \mid \mathbf{x}'_t, \mathbf{x}_{t-1}^{(r)}\right) p\left(\mathbf{x}'_t \mid \mathbf{x}_{t-1}^{(r)}\right) \mathrm{d}\mathbf{x}' \tag{5.91}$$

$$= w_{t-1}^{(r)} \int p\left(\mathbf{y}_t \mid \mathbf{x}'_t\right) p\left(\mathbf{x}'_t \mid \mathbf{x}_{t-1}^{(r)}\right) \mathrm{d}\mathbf{x}' \tag{5.92}$$

$$= w_{t-1}^{(r)} \int g\left(\mathbf{y}_t \mid \mathbf{x}'_t\right) f\left(\mathbf{x}'_t \mid \mathbf{x}_{t-1}^{(r)}\right) \mathrm{d}\mathbf{x}' \tag{5.93}$$

The proposal distribution is optimal because for any fixed $\mathbf{x}_{t-1}^{(r)}$, the new weight $w_t^{(r)}$ takes the same value regardless of the value drawn for $\mathbf{x}_t^{(r)}$. Hence, conditional on the old values, the variance of true weights is zero.

5.6. Sequential Monte Carlo

TODO: REDO

Assume that at time t, we can extend a particle's path using a Markov kernel M_t:

$$p_t(x_t) = p_{t-1}(x_{t-1}) M_t(x_{t-1}, x_t) \tag{5.94}$$

Also assume that

$$\tilde{p}_t(x_{0:t}) = p_t(x_t) \sum_{k=1}^{t} L_k(x_k, x_{k-1}) \tag{5.95}$$

where $\{L_k\}$ is a sequence of auxiliary Markov transition kernels.

The generic algorithm for Sequential Monte Carlo (SMC) can be found in Algorithm 10.

Algorithm 10 Generic Sequential Monte Carlo

Initialisation, $t = 0$:
for $r = 1, \ldots, R$ **do** ▷ Sample.
    Sample $\tilde{x}_0^{(r)} \sim q_0(\cdot)$.
for $r = 1, \ldots, R$ **do**
    Calculate normalised weights $\hat{w}_0^{(r)} \propto \frac{p_0\left(\tilde{x}_0^{(r)}\right)}{q_0\left(\tilde{x}_0^{(r)}\right)}$, such that $\sum'_r \hat{w}_0^{(r')} = 1$.
Resample from the pmf $\sum_r \hat{w}_0^{(r)} \delta_{\tilde{x}_0^{(r)}}(\cdot)$ to get R samples $\left\{x_0^{(r)}\right\}$. ▷ Resample.

Iterate, $t = 1, \ldots, T$:
for $t = 1, \ldots, T$ **do**
    for $r = 1, \ldots, R$ **do** ▷ Sample.
        Set $\tilde{x}_{0:t-1}^{(r)} = x_{0:t-1}^{(r)}$.
        Sample $\tilde{x}_t^{(r)} \sim M_t\left(\tilde{x}_{0:t-1}^{(r)}, \cdot\right)$.
    for $r = 1, \ldots, R$ **do**

14: Calculate normalised weights $\hat{w}_t^{(r)} \propto \frac{p_t(x_t)L_t(x_t,x_{t-1})}{p_{t-1}(x_{t-1})M_t(x_{t-1},x_t)}$.

15: Resample from the pmf $\sum_r \hat{w}_t^{(r)} \delta_{\tilde{x}_t^{(r)}}(\cdot)$ to get R samples $\left\{x_t^{(r)}\right\}$. Reset the weights to $1/R$. ▷ Resample.

5.7. Markov chain Monte Carlo methods

5.7.1. Definitions

Definition 5.7.1. Markov chain *(MC) is defined via a state space $\mathcal{X}$ and a model that defines, for every state $\mathbf{x} \in \mathcal{X}$ a next-state distribution over $\mathcal{X}$. More precisely, the transition model $\mathcal{T}$ specifies for each pair of state $\mathbf{x}, \mathbf{x}'$ the probability $\mathcal{T}(\mathbf{x} \to \mathbf{x}')$ of going from $\mathbf{x}$ to $\mathbf{x}'$, i.e. $\mathcal{T}(\mathbf{x} \to \mathbf{x}') = \Pr(\mathbf{x}' \mid \mathbf{x})$. This transition probability applies whenever the chain is in state $\mathbf{x}$.*

If the MCMC generates a sequence of states $\mathbf{x}_0, \ldots, \mathbf{x}_T$, the state at time t, $\mathbf{x}_t$ can be viewed as a random variable $\mathbf{X}_t$ for $t = 1, \ldots, T$.

Theorem 5.7.1 (Ergodic Theorem for MC (simplified))**.** *If $(\mathbf{X}_0, \ldots, \mathbf{X}_T)$ is an* irreducible, time-homogeneous *discrete space MC with* stationary distribution *π, then*

$$\frac{1}{T}\sum_{t=1}^{T} f(\mathbf{X}_t) \xrightarrow[n\to\infty]{a.s.} \mathbb{E}[f(\mathbf{X})] \qquad \text{where } \mathbf{X} \sim \pi \tag{5.96}$$

for any bounded function $f : \mathcal{X} \mapsto \mathbb{R}$.

If further, it is aperiodic*, then*

$$\Pr(\mathbf{X}_T = \mathbf{x} \mid \mathbf{X}_0 = \mathbf{x}_0) \xrightarrow[T\to\infty]{} \pi(\mathbf{x}) \qquad \forall\, \mathbf{x}, \mathbf{x}_0 \in \mathcal{X}. \tag{5.97}$$

A MC following these conditions is ergodic

Definition 5.7.2. *A MC $(\mathbf{X}_t)$ is* time-homogeneous *if $\Pr(\mathbf{X}_{t+1} = b \mid \mathbf{X}_t = a) = \mathcal{T}(a \to b)$ $\forall\, t \in \{1, \ldots, T-1\}$ $\forall\, a, b \in \mathcal{X}$ for some kernel function $\mathcal{T}$.*

Definition 5.7.3. *A pmf π on $\mathcal{X}$ is a* stationary (invariant) distribution *(w.r.t. $\mathcal{T}$) if*

$$\pi(\mathbf{X} = \mathbf{x}') = \sum_{\mathbf{x}\in\mathcal{X}} \pi(\mathbf{X} = \mathbf{x})\mathcal{T}(\mathbf{x} \to \mathbf{x}') \qquad \forall\, \mathbf{x}' \tag{5.98}$$

Definition 5.7.4. *A MC $(\mathbf{X}_t)$ is* irreducible *if $\forall\, a, b \in \mathcal{X}$ $\exists\, t \geq 0$ s.t. $\Pr(\mathbf{X}_t = b \mid \mathbf{X}_0 = a) > 0$.*

Definition 5.7.5. *An* irreducible *MC $(\mathbf{X}_t)$ is* aperiodic *if $\forall\, a \in \mathcal{X}$,*

$$\gcd\{t : \Pr(\mathbf{X}_t = a \mid \mathbf{X}_0 = a) > 0\} = 1. \tag{5.99}$$

Definition 5.7.6. *A MC is* regular *if there exists some number k such that, for every $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$, the probability of getting from $\mathbf{x}$ to $\mathbf{x}'$ in exactly k steps is > 0.*

Theorem 5.7.2. *If a finite state MC described by $\mathcal{T}$ is* regular*, then it has a unique* stationary distribution*.*

A MC being *ergodic* is equivalent to it being *regular* [1, p. 510].

Definition 5.7.7. *A finite state MC described by $\mathcal{T}$ is* reversible *if there exists a unique distribution π such that, for all $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$*

$$\pi(\mathbf{x})\mathcal{T}(\mathbf{x} \to \mathbf{x}') = \pi(\mathbf{x}')\mathcal{T}(\mathbf{x}' \to \mathbf{x}). \tag{5.100}$$

This equation is called the detailed balance *(DB).*

Proposition 5.7.1. *If a finite state MC described by $\mathcal{T}$ is* regular *and satisfies the* detailed balance *equation relative to π, then π is the* unique stationary distribution *of $\mathcal{T}$.*

Proof. Assuming the DB equation (5.100), we want to prove the stationarity equation (5.98) to ensure π is a stationary distribution of $\mathcal{T}$. We have

$$\begin{aligned}
\sum_{\mathbf{x}\in\mathcal{X}} \pi(\mathbf{x})\mathcal{T}(\mathbf{x} \to \mathbf{x}') &= \sum_{\mathbf{x}\in\mathcal{X}} \pi(\mathbf{x}')\mathcal{T}(\mathbf{x}' \to \mathbf{x}) && (5.101)\\
&= \sum_{\mathbf{x}\in\mathcal{X}} \pi(\mathbf{x}') \Pr(\mathbf{x} \mid \mathbf{x}') && (5.102)\\
&= \pi(\mathbf{x}') \sum_{\mathbf{x}\in\mathcal{X}} \Pr(\mathbf{x} \mid \mathbf{x}') && (5.103)\\
&= \pi(\mathbf{x}') && (5.104)
\end{aligned}$$

which proves the equation (5.98). π is the unique stationary distribution of $\mathcal{T}$ because of Theorem 5.7.2. □

Proposition 5.7.2. *Let $\mathcal{T}_1, \ldots, \mathcal{T}_K$ be a set of kernels each of which satisfies detailed balance w.r.t. π. Let $p_1, \ldots, p_K$ be any distribution over $\{1, \ldots, K\}$. The mixture MC $\mathcal{T}$, which at each step takes a step sampled from $\mathcal{T}_k$ with probability p_k also satisfies the detailed balance equation relative to π.*

Proof. The aggregate kernel can be written as

$$\begin{aligned}
\mathcal{T}(\mathbf{x} \to \mathbf{x}') &= \Pr(\mathbf{x}' \mid \mathbf{x}) && (5.105)\\
&= \sum_k \Pr(\mathbf{x}', k \mid \mathbf{x}) && (5.106)\\
&= \sum_k \Pr(\mathbf{x}' \mid k, \mathbf{x}) \Pr(k \mid \mathbf{x}) && (5.107)\\
&= \sum_k \mathcal{T}_k(\mathbf{x} \to \mathbf{x}')p_k && (5.108)
\end{aligned}$$

Using this, we can prove the detailed balance as follows

$$\pi(\mathbf{x})\mathcal{T}(\mathbf{x} \to \mathbf{x}') = \pi(\mathbf{x}) \sum_k \mathcal{T}_k(\mathbf{x} \to \mathbf{x}')p_k \tag{5.109}$$

$$= \sum_k \pi(\mathbf{x}) \mathcal{T}_k(\mathbf{x} \to \mathbf{x}') p_k \tag{5.110}$$

$$= \sum_k \pi(\mathbf{x}') \mathcal{T}_k(\mathbf{x}' \to \mathbf{x}) p_k \tag{5.111}$$

$$= \pi(\mathbf{x}') \sum_k \mathcal{T}_k(\mathbf{x}' \to \mathbf{x}) p_k \tag{5.112}$$

$$= \pi(\mathbf{x}') \mathcal{T}(\mathbf{x}' \to \mathbf{x}) \tag{5.113}$$

□

Proposition 5.7.3. *Let $\mathcal{T}_1, \ldots, \mathcal{T}_K$ be a set of kernels each of which satisfies detailed balance w.r.t. π. The aggregate MC, $\mathcal{T}$, where each step consists of a sequence of K steps, with step k being sampled from $\mathcal{T}_k$ has π as its stationary distribution.*

Proof. The aggregate kernel can be written as

$$\mathcal{T}(\mathbf{x} \to \mathbf{x}') = \Pr(\mathbf{x}' \mid \mathbf{x}) \tag{5.114}$$

$$= \sum_{\mathbf{x}_{1:K-1}} \Pr(\mathbf{x}', \mathbf{x}_{K-1}, \ldots, \mathbf{x}_1 \mid \mathbf{x}) \tag{5.115}$$

$$= \sum_{\mathbf{x}_{1:K-1}} \Pr(\mathbf{x}_K, \ldots, \mathbf{x}_1 \mid \mathbf{x}_0) \tag{5.116}$$

$$= \sum_{\mathbf{x}_{1:K-1}} \Pr(\mathbf{x}_1 \mid \mathbf{x}_0) \cdots \Pr(\mathbf{x}_K \mid \mathbf{x}_{K-1}) \tag{5.117}$$

$$= \sum_{\mathbf{x}_{1:K-1}} \mathcal{T}_1(\mathbf{x}_0 \to \mathbf{x}_1) \cdots \mathcal{T}_K(\mathbf{x}_{K-1} \to \mathbf{x}_K) \tag{5.118}$$

where we've used the substitution $\mathbf{x} = \mathbf{x}_0$ and $\mathbf{x}' = \mathbf{x}_K$. Using this, we can prove that π is the stationary distribution as follows

$$\sum_{\mathbf{x} \in \mathcal{X}} \pi(\mathbf{x}) \mathcal{T}(\mathbf{x} \to \mathbf{x}') = \sum_{\mathbf{x}_0} \pi(\mathbf{x}_0) \sum_{\mathbf{x}_{1:K-1}} \mathcal{T}_1(\mathbf{x}_0 \to \mathbf{x}_1) \cdots \mathcal{T}_K(\mathbf{x}_{K-1} \to \mathbf{x}_K) \tag{5.119}$$

$$= \sum_{\mathbf{x}_{0:K-1}} \pi(\mathbf{x}_0) \mathcal{T}_1(\mathbf{x}_0 \to \mathbf{x}_1) \cdots \mathcal{T}_K(\mathbf{x}_{K-1} \to \mathbf{x}_K) \tag{5.120}$$

$$= \sum_{\mathbf{x}_{0:K-1}} \mathcal{T}_1(\mathbf{x}_1 \to \mathbf{x}_0) \pi(\mathbf{x}_1) \cdots \mathcal{T}_K(\mathbf{x}_{K-1} \to \mathbf{x}_K) \tag{5.121}$$

$$\cdots$$

$$= \sum_{\mathbf{x}_{0:K-1}} \mathcal{T}_1(\mathbf{x}_1 \to \mathbf{x}_0) \cdots \mathcal{T}_K(\mathbf{x}_K \to \mathbf{x}_{K-1}) \pi(\mathbf{x}_K) \tag{5.122}$$

$$= \pi(\mathbf{x}_K) \sum_{\mathbf{x}_{0:K-1}} \mathcal{T}_K(\mathbf{x}_K \to \mathbf{x}_{K-1}) \cdots \mathcal{T}_1(\mathbf{x}_1 \to \mathbf{x}_0) \tag{5.123}$$

$$= \pi(\mathbf{x}_K) \sum_{\mathbf{x}_{0:K-1}} \Pr(\mathbf{x}_{0:K-1} \mid \mathbf{x}_K) \tag{5.124}$$

$$= \pi(\mathbf{x}_K). \tag{5.125}$$

□

5.7.2. Metropolis Hastings algorithm

The Metropolis Hastings (MH) algorithm is a recipe to create a MCMC with a particular stationary distribution. Assume we can sample from a proposal distribution $q(\cdot \mid \mathbf{x}) \equiv q(\mathbf{x} \to \cdot)$. Let $p \equiv \pi$ be the required distribution (stationary distribution for this MCMC). Assume we can only evaluate q and π up to a multiplicative factor (i.e. we can only evaluate $q^*(\mathbf{x} \to \mathbf{x}') = Z_q q(\mathbf{x} \to \mathbf{x}')$ and $\pi^*(\mathbf{x}) = Z_p \pi(\mathbf{x})$). The MH algorithm is outlined in Algorithm 11.

Algorithm 11 Metropolis Hastings algorithm

1: Sample $\mathbf{x}^{(0)}$ from an arbitrary probability distribution over $\mathcal{X}$.
2: **for** $t = 1, \ldots, T$ **do**
3: **repeat**
4: Sample $\mathbf{x}^{(t)} \sim q(\mathbf{x}^{(t-1)} \to \cdot)$.
5: Accept $\mathbf{x}^{(t)}$ with the acceptance probability

$$\mathcal{A}(\mathbf{x}^{(t-1)} \to \mathbf{x}^{(t)}) = \min\left(1, \frac{\pi^*(\mathbf{x}^{(t)}) q^*(\mathbf{x}^{(t)} \to \mathbf{x}^{(t-1)})}{\pi^*(\mathbf{x}^{(t-1)}) q^*(\mathbf{x}^{(t-1)} \to \mathbf{x}^{(t)})}\right) \tag{5.126}$$

6: **until** $\mathbf{x}^{(t)}$ is accepted.

Why it works?

We need to prove that π is the unique stationary distribution of this MCMC.

We can express the aggregate transition model to be

$$\mathcal{T}(\mathbf{x} \to \mathbf{x}') = \begin{cases} q(\mathbf{x} \to \mathbf{x}') \mathcal{A}(\mathbf{x} \to \mathbf{x}') & \text{if } \mathbf{x} \neq \mathbf{x}' \\ q(\mathbf{x} \to \mathbf{x}) + \sum_{\mathbf{x}', \mathbf{x}' \neq \mathbf{x}} q(\mathbf{x} \to \mathbf{x}')(1 - \mathcal{A}(\mathbf{x} \to \mathbf{x}')) & \text{if } \mathbf{x} = \mathbf{x}' \end{cases} \tag{5.127}$$

To prove that π is a stationary distribution of this MCMC, we make sure the DB equation holds.

For $\mathbf{x} \neq \mathbf{x}'$, we have

$$\begin{aligned} \pi(\mathbf{x}) \mathcal{T}(\mathbf{x} \to \mathbf{x}') &= \pi(\mathbf{x}) q(\mathbf{x} \to \mathbf{x}') \min\left(1, \frac{\pi(\mathbf{x}') q(\mathbf{x}' \to \mathbf{x})}{\pi(\mathbf{x}) q(\mathbf{x} \to \mathbf{x}')}\right) && (5.128) \\ &= \min\left(\pi(\mathbf{x}) q(\mathbf{x} \to \mathbf{x}'), \pi(\mathbf{x}') q(\mathbf{x}' \to \mathbf{x})\right) && (5.129) \\ &= \pi(\mathbf{x}') q(\mathbf{x}' \to \mathbf{x}) \min\left(1, \frac{\pi(\mathbf{x}) q(\mathbf{x} \to \mathbf{x}')}{\pi(\mathbf{x}') q(\mathbf{x}' \to \mathbf{x})}\right) && (5.130) \\ &= \pi(\mathbf{x}') \mathcal{T}(\mathbf{x}' \to \mathbf{x}) && (5.131) \end{aligned}$$

For $\mathbf{x} = \mathbf{x}'$, the DB equation $\pi(\mathbf{x}) \mathcal{T}(\mathbf{x} \to \mathbf{x}') = \pi(\mathbf{x}') \mathcal{T}(\mathbf{x}' \to \mathbf{x})$ obviously holds.

Hence π is a stationary distribution of the MCMC described via $\mathcal{T}$. Unfortunately, regularity doesn't hold in general. We need to make sure our created MCMC is regular before we can claim that π is the unique stationary distribution of this MCMC.

5.7.3. Gibbs sampling

Assume we want to sample from $p(\mathbf{x}) = p(x_1, \ldots, x_D)$. We can only sample from the conditionals $p(x_i \mid \mathbf{x}_{-i})$ where $\mathbf{x}_{-i}$ denotes $\mathbf{x}$ with the i^{th} component ommited. The Gibbs sampling algorithm (12) is given below.

Algorithm 12 Gibbs sampling algorithm

1: Sample $\mathbf{x}^{(0)}$ from an arbitrary probability distribution over $\mathcal{X}$.
2: **for** $t = 1, \ldots, T$ **do**
3: Sample $x_1^{(t)} \sim p\left(\cdot \mid x_2^{(t-1)}, x_3^{(t-1)}, \ldots, x_D^{(t-1)}\right)$
4: Sample $x_2^{(t)} \sim p\left(\cdot \mid x_1^{(t)}, x_3^{(t-1)}, \ldots, x_D^{(t-1)}\right)$
5: $\vdots$
6: Sample $x_D^{(t)} \sim p\left(\cdot \mid x_1^{(t)}, x_2^{(t)}, \ldots, x_{D-1}^{(t)}\right)$

Why it works?

Each of the sampling steps can be viewed to be governed by a different kernel with the whole process being governed by the aggregate kernel. We prove that the single kernels follow the DB equation with respect to p:

$$
\begin{aligned}
p(\mathbf{x})\mathcal{T}_i(\mathbf{x} \to \mathbf{x}') &= p(\mathbf{x})p(\mathbf{x}_{-i}, x_i' \mid \mathbf{x}) && (5.132)\\
&= p(\mathbf{x}_{-i}, x_i', \mathbf{x}) && (5.133)\\
&= p(\mathbf{x}, x_i', \mathbf{x}_{-i}) && (5.134)\\
&= p(\mathbf{x}')p(\mathbf{x} \mid x_i', \mathbf{x}_{-i}) && (5.135)\\
&= p(\mathbf{x}')\mathcal{T}_i(\mathbf{x}' \to \mathbf{x}) && (5.136)
\end{aligned}
$$

This is the premise of Proposition 5.7.3, hence the aggregate kernel $\mathcal{T}$ has p as its stationary distribution.

We can also view Gibbs sampling as an instance of the MH algorithm. If the proposal of MH $q_i(\mathbf{x} \to \mathbf{x}')$ is set to be $p(\mathbf{x}' \mid \mathbf{x}) = p(x_i' \mid \mathbf{x})$ the acceptance probability is one (shown below) and so it is equivalent to one sampling step in Gibbs sampling.

$$
\begin{aligned}
\mathcal{A}(\mathbf{x} \to \mathbf{x}') &= \min\left(1, \frac{p(\mathbf{x}')p(\mathbf{x} \mid \mathbf{x}')}{p(\mathbf{x})p(\mathbf{x}' \mid \mathbf{x})}\right) && (5.137)\\
&= \min\left(1, \frac{p(\mathbf{x}', \mathbf{x})}{p(\mathbf{x}', \mathbf{x})}\right) && (5.138)\\
&= 1 && (5.139)
\end{aligned}
$$

5.8. Particle Markov Chain Monte Carlo

5.8.1. Particle independent Metropolis Hastings (PIMH) sampler

We want to sample from $p(\mathbf{x}_{1:T} \mid \mathbf{y}_{1:T}, \boldsymbol{\theta})$.

Algorithm 13 Particle independent Metropolis Hastings sampler

1: Run SMC targetting ▷ Initial sweep $s = 0$

$$p(\mathbf{x}_{1:T} \mid \mathbf{y}_{1:T}; \boldsymbol{\theta})$$

2: Sample

$$\mathbf{x}_{1:T}(0) \sim \hat{p}(\cdot \mid \mathbf{y}_{1:T}; \boldsymbol{\theta})$$

3: Let

$$\hat{p}(\mathbf{y}_{1:T} \mid \boldsymbol{\theta})$$

denote the corresponding marginal likelihood estimate.

4: **for** $s = 1, \ldots, S$ **do** ▷ Main loop

5: Run SMC targeting

$$p(\mathbf{x}_{1:T} \mid \mathbf{y}_{1:T}; \boldsymbol{\theta})$$

6: Sample

$$\mathbf{x}^*_{1:T} \sim \hat{p}(\cdot \mid \mathbf{y}_{1:T}; \boldsymbol{\theta})$$

7: Let

$$\hat{p}(\mathbf{y}_{1:T}; \boldsymbol{\theta})^*$$

denote the coresponding marginal likelihood estimate

8: Sample from $\mathrm{Ber}(\cdot)$ with the success probability

$$\min\left(1, \frac{\hat{p}(\mathbf{y}_{1:T} \mid \boldsymbol{\theta})^*}{\hat{p}(\mathbf{y}_{1:T}; \boldsymbol{\theta})(s-1)}\right)$$

9: **if** success **then**

10: Set

$$\mathbf{x}_{1:T}(s) = \mathbf{x}^*_{1:T}$$
$$\hat{p}(\mathbf{y}_{1:T} \mid \boldsymbol{\theta})(s) = \hat{p}(\mathbf{y}_{1:T} \mid \boldsymbol{\theta})^*$$

11: **else**

12: Set

$$\mathbf{x}_{1:T}(s) = \mathbf{x}_{1:T}(s-1)$$
$$\hat{p}(\mathbf{y}_{1:T} \mid \boldsymbol{\theta})(s) = \hat{p}(\mathbf{y}_{1:T} \mid \boldsymbol{\theta})(s-1)$$

5.8.2. Particle marginal Metropolis Hastings (PMMH) sampler

We want to sample from $p(\boldsymbol{\theta}, \mathbf{x}_{1:T} \mid \mathbf{y}_{1:T}) \propto p(\mathbf{x}_{1:T} \mid \mathbf{y}_{1:T;\boldsymbol{\theta}})p(\boldsymbol{\theta})$.

Algorithm 14 Particle marginal Metropolis Hastings sampler

1: Set $\boldsymbol{\theta}(0)$ arbitrarily.

2: Run SMC targetting ▷ Initial sweep $s = 0$

$$p(\mathbf{x}_{1:T} \mid \mathbf{y}_{1:T}; \boldsymbol{\theta}(0))$$

3: Sample

$$\mathbf{x}_{1:T}(0) \sim \hat{p}(\cdot \mid \mathbf{y}_{1:T}; \boldsymbol{\theta}(0))$$

4: Let

$$\hat{p}(\mathbf{y}_{1:T} \mid \boldsymbol{\theta}(0))$$

denote the corresponding marginal likelihood estimate.

5: **for** $s = 1, \ldots, S$ **do** ▷ Main loop

6: Sample

$$\boldsymbol{\theta}^* \sim q(\cdot \mid \boldsymbol{\theta}(s-1))$$

7: Run SMC targeting

$$p(\mathbf{x}_{1:T} \mid \mathbf{y}_{1:T}; \boldsymbol{\theta}^*)$$

8: Sample

$$\mathbf{x}^*_{1:T} \sim \hat{p}(\cdot \mid \mathbf{y}_{1:T}; \boldsymbol{\theta}^*)$$

9: Let

$$\hat{p}(\mathbf{y}_{1:T}; \boldsymbol{\theta}^*)$$

denote the coresponding marginal likelihood estimate

10: Sample from $\mathrm{Ber}(\cdot)$ with the success probability

$$\min\left(1, \frac{\hat{p}(\mathbf{y}_{1:T} \mid \boldsymbol{\theta}^*) p(\boldsymbol{\theta}^*) q(\boldsymbol{\theta}(s-1) \mid \boldsymbol{\theta}^*)}{\hat{p}(\mathbf{y}_{1:T}; \boldsymbol{\theta}(s-1)) p(\boldsymbol{\theta}(s-1)) q(\boldsymbol{\theta}^* \mid \boldsymbol{\theta}(s-1))}\right)$$

11: **if** success **then**

12: Set

$$\begin{aligned} \boldsymbol{\theta}(s) &= \boldsymbol{\theta}^* \\ \mathbf{x}_{1:T}(s) &= \mathbf{x}^*_{1:T} \\ \hat{p}(\mathbf{y}_{1:T} \mid \boldsymbol{\theta})(s) &= \hat{p}(\mathbf{y}_{1:T} \mid \boldsymbol{\theta}^*) \end{aligned}$$

13: **else**

14: Set

$$\begin{aligned} \boldsymbol{\theta}(s) &= \boldsymbol{\theta}(s-1) \\ \mathbf{x}_{1:T}(s) &= \mathbf{x}_{1:T}(s-1) \\ \hat{p}(\mathbf{y}_{1:T} \mid \boldsymbol{\theta})(s) &= \hat{p}(\mathbf{y}_{1:T} \mid \boldsymbol{\theta})(s-1) \end{aligned}$$

5.8.3. Particle Gibbs (PG) sampler

Conditional SMC update

We want to sample from $p(\mathbf{x}_{1:T} \mid \mathbf{y}_{1:T}; \boldsymbol{\theta})$.

Algorithm 15 Conditional SMC update

Choose a fixed ancestral lineage $B_{1:T}$ arbitrarily. ▷ Initialise fixed path

Let

$$\mathbf{x}_{1:T} = \left(\mathbf{x}_1^{(B_1)}, \ldots, \mathbf{x}_T^{(B_T)}\right)$$

be a path associated with the ancestral lineage $B_{1:T}$.

For $r \neq B_1$, sample ▷ Time $t = 1$

$$\mathbf{x}_1^{(r)} \sim q(\cdot \mid \mathbf{y}_1, \boldsymbol{\theta})$$

Compute weights

$$w_1^{(r)} \propto \frac{p\left(\mathbf{x}_1^{(r)}, \mathbf{y}_1\right)}{q\left(\mathbf{x}_1^{(r)} \mid \mathbf{y}_1\right)}$$

Normalise weights

$$\hat{w}_1^{(r)} = \frac{w_1^{(r)}}{\sum_{r'} w_1^{(r')}}$$

We can resample from

$$\hat{p}(\mathrm{d}\mathbf{x}_1 \mid \mathbf{y}_1, \boldsymbol{\theta}) = \sum_r \hat{w}_1^{(r)} \delta_{\mathbf{x}_1^{(r)}}(\mathrm{d}\mathbf{x}_1)$$

to estimate

$$p(\mathbf{x}_1 \mid \mathbf{y}_1, \boldsymbol{\theta})$$

for $t = 2, \ldots, T$ **do** ▷ Main loop

  For $r \neq B_t$, sample

$$A_{t-1}^{(r)} \sim \mathrm{Cat}\left(\hat{w}_{t-1}^{(1)}, \ldots, \hat{w}_{t-1}^{(R)}\right)$$

  For $r \neq B_t$, sample

$$\mathbf{x}_t^{(r)} \sim q\left(\cdot \mid \mathbf{y}_t, \mathbf{x}_{t-1}^{(A_{t-1}^{(r)})}\right)$$

  Compute weights

$$w_t^{(r)} = \frac{p\left(\mathbf{x}_{1:t}^{(r)}, \mathbf{y}_{1:t}; \boldsymbol{\theta}\right)}{p\left(\mathbf{x}_{1:t-1}^{\left(A_{t-1}^{(r)}\right)}, \mathbf{y}_{1:t-1}; \boldsymbol{\theta}\right) q\left(\mathbf{x}_n^{(r)} \mid \mathbf{y}_t, \mathbf{x}_{t-1}^{\left(A_{t-1}^{(r)}\right)}; \boldsymbol{\theta}\right)}$$

11: Normalise weights

$$\hat{w}_t^{(r)} = \frac{w_t^{(r)}}{\sum_{r'} w_t^{(r')}}$$

12: We can resample from

$$\hat{p}(\mathrm{d}\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}, \boldsymbol{\theta}) = \sum_r \hat{w}_t^{(r)} \delta_{\mathbf{x}_{1:t}^{(r)}}(\mathrm{d}\mathbf{x}_{1:t})$$

to estimate

$$p(\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}, \boldsymbol{\theta})$$

Particle Gibbs sampler

We want to sample from $p(\boldsymbol{\theta}, \mathbf{x}_{1:T} \mid \mathbf{y}_{1:T})$.

Algorithm 16 Particle Gibbs sampler

1: Set $\theta(0)$, $\mathbf{x}_{1:T}(0)$, $B_{1:T}(0)$ arbitrarily. ▷ Initialisation, $s = 0$

2: **for** Sweep $s = 1, \ldots, S$ **do** ▷ Main loop

3: Sample parameter

$$\boldsymbol{\theta}(s) \sim p\left(\cdot \mid \mathbf{y}_{1:T}, \mathbf{x}_{1:T}(s-1)\right)$$

4: Run conditional SMC (Algorithm 15) targetting

$$p(\mathbf{x}_{1:T} \mid \mathbf{y}_{1:T}; \boldsymbol{\theta}(s))$$

conditional on

- $\mathbf{x}_{1:T}(s-1)$, and
- $B_{1:T}(s-1)$.

5: Sample

$$\mathbf{x}_{1:T}(s) \sim \hat{p}(\cdot \mid \mathbf{y}_{1:T}; \boldsymbol{\theta}(s))$$

---

6. Nonparametric Bayesian models

6.1. Gaussian processes

Gaussian Process is a collection of random variables, any finite number of which have a joint Gaussian distribution.

Let $f : \mathcal{X} \to \mathbb{R}$ be a function and

$$\begin{aligned} m(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})] \\ K(\mathbf{x}, \mathbf{x}') &= \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))] \end{aligned}$$

be the mean function and covariance functions mapping from the input space $\mathcal{X}$ to $\mathbb{R}$.

Then we write

$$f \sim \mathrm{GP}(m, K)$$

if for any N points from the input space $\mathbf{x}_1, \ldots, \mathbf{x}_N \in \mathcal{X}$

$$\mathbf{f} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{K})$$

where

$$\begin{aligned} \mathbf{f} &= [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)]^T \\ \boldsymbol{\mu} &= [m(\mathbf{x}_1), \ldots, m(\mathbf{x}_N)]^T \\ \mathbf{K} &= \begin{bmatrix} K(\mathbf{x}_1, \mathbf{x}_1) & \cdots & K(\mathbf{x}_1, \mathbf{x}_N) \\ \vdots & \ddots & \vdots \\ K(\mathbf{x}_N, \mathbf{x}_1) & \cdots & K(\mathbf{x}_N, \mathbf{x}_N) \end{bmatrix} \end{aligned}$$

6.1.1. Predictions

Consider data points $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^T, \mathbf{f} = [f_1, \ldots, f_N]^T$ and input points $\mathbf{X}^* = [\mathbf{x}_1^*, \ldots, \mathbf{x}_{N^*}^*]^T$ for which we want to predict $\mathbf{f}^* = [f(\mathbf{x}_1^*), \ldots, f(\mathbf{x}_{N^*}^*)]^T$. Then we can write

$$\begin{bmatrix} \mathbf{f} \\ \mathbf{f}^* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\mu}^* \end{bmatrix}, \begin{bmatrix} \mathbf{K}(\mathbf{X}, \mathbf{X}) & \mathbf{K}(\mathbf{X}, \mathbf{X}^*) \\ \mathbf{K}(\mathbf{X}^*, \mathbf{X}) & \mathbf{K}(\mathbf{X}^*, \mathbf{X}^*) \end{bmatrix} \right)$$

where

$$\begin{aligned} \boldsymbol{\mu}^* &= [m(\mathbf{x}_1^*), \ldots, m(\mathbf{x}_{N^*}^*)]^T \\ \mathbf{K}(\mathbf{X}, \mathbf{X}) &= \begin{bmatrix} K(\mathbf{x}_1, \mathbf{x}_1) & \cdots & K(\mathbf{x}_1, \mathbf{x}_N) \\ \vdots & \ddots & \vdots \\ K(\mathbf{x}_N, \mathbf{x}_1) & \cdots & K(\mathbf{x}_N, \mathbf{x}_N) \end{bmatrix} \end{aligned}$$

$$\mathbf{K}(\mathbf{X}, \mathbf{X}^*) = \begin{bmatrix} K(\mathbf{x}_1, \mathbf{x}_1^*) & \cdots & K(\mathbf{x}_1, \mathbf{x}_{N^*}^*) \\ \vdots & \ddots & \vdots \\ K(\mathbf{x}_N, \mathbf{x}_1^*) & \cdots & K(\mathbf{x}_N, \mathbf{x}_{N^*}^*) \end{bmatrix}$$
$$\mathbf{K}(\mathbf{X}^*, \mathbf{X}) = \mathbf{K}(\mathbf{X}, \mathbf{X}^*)^T$$

Hence, according to Subsection 2.4.2, the predictions can be made as follows

$$\mathbf{f}^* \mid \mathbf{X}^*, \mathbf{X}, \mathbf{f} \sim \mathcal{N}(\boldsymbol{\mu}^{*|\mathcal{D}}, \mathbf{K}^{*|\mathcal{D}})$$

where

$$\boldsymbol{\mu}^{*|\mathcal{D}} = \boldsymbol{\mu}^* + \mathbf{K}(\mathbf{X}^*, \mathbf{X})\mathbf{K}(\mathbf{X}, \mathbf{X})^{-1}(\mathbf{f} - \boldsymbol{\mu})$$
$$\mathbf{K}^{*|\mathcal{D}} = \mathbf{K}(\mathbf{X}^*, \mathbf{X}^*) - \mathbf{K}(\mathbf{X}^*, \mathbf{X})K(\mathbf{X}, \mathbf{X})^{-1}\mathbf{K}(\mathbf{X}, \mathbf{X}^*)$$

6.2. Dirichlet processes

Notes made from Erik Sudderth's PhD.

6.2.1. Definitions

Definition 6.2.1 (Probability measure)**.** *Probability measure is a real-valued function P defined on a set of events in a probability space $(\Omega, \mathcal{F}, P)$ that satisfies*

- *P must return results $\in [0, 1]$, returning 0 for $\varnothing$, 1 for the entire space, Ω, and*
- *countable additivity: $\forall$ countable collections $\{E_i\}$ of pairwise disjoint sets of Ω,*

$$P\left(\bigcup_{i \in I} E_i\right) = \sum_{i \in I} P(E_i)$$

Definition 6.2.2 (Stochastic process)**.** *Suppose that $(\Omega, \mathcal{F}, P)$ is a probability space, and that T ("time") is a totally ordered set. Suppose further that for each $t \in T$, there is a random variable $X_t : \Omega \to S$ defined on $(\Omega, \mathcal{F}, P)$. A stochastic process X is a collection $\{X_t : t \in T\}$. S is called the state space of the process.*

Theorem 6.2.1 (Dirichlet process)**.** *Let H be a probability distribution on a measurable space Θ, and α a positive scalar. Consider a finite partition $(T_1, \ldots, T_K)$ of Θ.*

A random probability distribution G on Θ is drawn from a Dirichlet process if its measure on every finite partition follows a Dirichlet distribution:

$$(G(T_1), \ldots, G(T_K)) \sim \mathrm{Dir}(\alpha H(T_1), \ldots, \alpha H(T_K)) \tag{6.1}$$

For any α, H, there exists a unique stochastic process satisfying these conditions, which we denote $\mathrm{DP}(\alpha, H)$.

Claim 6.2.1. *The base measure is the mean, i.e.*

$$\forall T \subset \Theta, \mathbb{E}[G(T)] = H(T) \tag{6.2}$$

Proof. Let $T \equiv T_k$ for some finite partition $(T_1, \ldots, T_k, \ldots, T_K)$ of Θ. Then since (6.1) we have

$$\mathbb{E}[G(T_k)] = \frac{\alpha H(T_k)}{\sum_j \alpha H(T_j)} = \frac{H(T_k)}{\sum_j H(T_j)} = H(T_k) \tag{6.3}$$

□

6.2.2. Posterior measure

Proposition 6.2.1 (Posterior measure)**.** *Let $G \sim \mathrm{DP}(\alpha, H)$ be a random measure distributed according to a Dirichlet process. Given N independent observations $\mathcal{D} = \{x_n : x_n \sim G\}_{n=1}^N$, the posterior measure also follows a Dirichlet process:*

$$G \mid \mathcal{D}, \alpha, H \sim \mathrm{DP}\left(\alpha + N, \frac{1}{\alpha + N}\left(\alpha H + \sum_n \delta_{x_n}\right)\right) \tag{6.4}$$

Proof. For any finite partition $(T_1, \ldots, T_K)$ of the sample space Θ, we have the following:

$$(G(T_1), \ldots, G(T_K)) \sim \mathrm{Dir}(\alpha H(T_1), \ldots, \alpha H(T_K))$$

We can represent the observations $\mathcal{D}$ as $\mathcal{D}'$, only caring about which partition T_k it comes from, in the following manner:

$$\mathcal{D}' = \{\mathbf{x}'_n : \mathbf{x}'_n = (\mathbb{I}(x_n \in T_1), \ldots, \mathbb{I}(x_n \in T_K)) \sim \mathrm{Mult}\,(1, (G(T_1), \ldots, G(T_K)))\}$$

The samples are indeed drawn from a given Multinomial distribution since $\Pr(x_n \in T_k) = G(T_k), k = 1, \ldots, K$ by definition.

From conjugacy in (3.2)

$$\begin{aligned}
(G(T_1), \ldots, G(T_K)) \mid \mathcal{D} &\sim \mathrm{Dir}\left((\alpha H(T_1), \ldots, \alpha H(T_K)) + \sum_n \mathbf{x}'_n\right) \\
&\equiv \mathrm{Dir}\left((\alpha H(T_1), \ldots, \alpha H(T_K)) + \sum_n (\mathbb{I}(x_n \in T_1), \ldots, \mathbb{I}(x_n \in T_K))\right) \\
&\equiv \mathrm{Dir}\left(\alpha H(T_1) + \sum_n \mathbb{I}(x_n \in T_1), \ldots, \alpha H(T_K) + \sum_n \mathbb{I}(x_n \in T_K)\right) \\
&\equiv \mathrm{Dir}\left(\alpha H(T_1) + \sum_n \delta_{x_n}(T_1), \ldots, \alpha H(T_K) + \sum_n \delta_{x_n}(T_K)\right)
\end{aligned}$$

Since this is true for any finite partition $(T_1, \ldots, T_K)$, it implies that

$$G \mid \mathcal{D} \sim \mathrm{DP}\left(Z, \frac{1}{Z}\left(\alpha H + \sum_n \delta_{x_n}\right)\right)$$

for some normalisation constant of the new base measure. Suppose that we now partition the space into $(T_1 = \{x_1\}, \ldots, T_N = \{x_N\}, T' = \Theta \setminus \{x_1, \ldots, x_N\})$, the normalisation constant Z can be evaluated as

$$\begin{aligned} Z &= \left(\alpha H(T') + \sum_{n=1}^{N} \delta_{x_n}(T')\right) + \sum_{m=1}^{N} \left(\alpha H(T_m) + \sum_{n=1}^{N} \delta_{x_n}(T_m)\right) \\ &= \alpha H(T') + \sum_{m=1}^{N} \alpha H(T_m) + \sum_{m=1}^{N} \sum_{n=1}^{N} \delta_{x_n}(T_m) \\ &= \alpha \left(H(T') + \sum_{m=1}^{N} H(T_m)\right) + N \\ &= \alpha + N \end{aligned}$$

We can write the final posterior as

$$G \mid \mathcal{D} \sim \mathrm{DP}\left(\alpha + N, \frac{1}{\alpha + N}\left(\alpha H + \sum_{n} \delta_{x_n}\right)\right)$$

□

Doksum and Fabius showed that for every measurable $T \subset \Theta$, and any N observations $\mathcal{D} = \{x_n : x_n \sim G\}$, the posterior distribution $p(G \mid \mathcal{D})$ depends only on the number of observations that fall within T (and not their particular locations). I.e. observations provide information only about those cells which directly contain them.

6.2.3. Stick-breaking construction

Given $G \sim \mathrm{DP}(\alpha, H)$ and $\mathcal{D} = \{x_n : x_n \sim G\}$. From (6.2) and (6.4) we know that for any $T \subset \Theta$

$$\mathbb{E}[G(T) \mid \mathcal{D}, \alpha, H] = \frac{1}{\alpha + N}\left(\alpha H + \sum_{n} \delta_{x_n}(T)\right) \tag{6.5}$$

For finite α

$$\begin{aligned} \lim_{N \to \infty} \mathbb{E}[G(T) \mid \mathcal{D}, \alpha, H] &= \lim_{N \to \infty} \frac{1}{N} \sum_{n} \delta_{x_n}(T) \\ &= \sum_{k=1}^{\infty} \pi_k \delta_{\bar{x}_k}(T) \end{aligned}$$

where $\{\bar{x}_k\}_{k=1}^{\infty}$ are the unique values of $\{x_n\}_{n=1}^{\infty}$ and $\pi_k = \lim_{N \to \infty} \frac{\sum_n \mathbb{I}(x_n = \bar{x}_k)}{N}$ is the limiting empirical frequency of $\bar{x}_k$.

Theorem 6.2.2 (Stick-breaking construction)**.** *Let $\boldsymbol{\pi} = \{\pi_k\}_{k=1}^{\infty}$ be an infinite sequence of mixture weights derived from the following stick-breaking process, with parameter $\alpha > 0$:*

$$\beta_k \sim \mathrm{Beta}(1, \alpha) \tag{6.6}$$

$$\pi_k = \beta_k \prod_{\ell=1}^{k-1}(1-\beta_\ell) \tag{6.7}$$

$$= \beta_k \left(1 - \sum_{\ell=1}^{k-1} \pi_\ell\right) \tag{6.8}$$

for $k = 1, 2, \ldots$. *Given a base measure* H *on* Θ*, consider the following discrete random measure:*

$$G(x) = \sum_{k=1}^{\infty} \pi_k \delta(x, x_k) \qquad\qquad x_k \sim H \tag{6.9}$$

The construction guarantees $G \sim \mathrm{DP}(\alpha, H)$*. Also, samples from a* DP *are discrete with probability* 1 *and have a representation as in* (6.9)*.*

Proof. TODO □

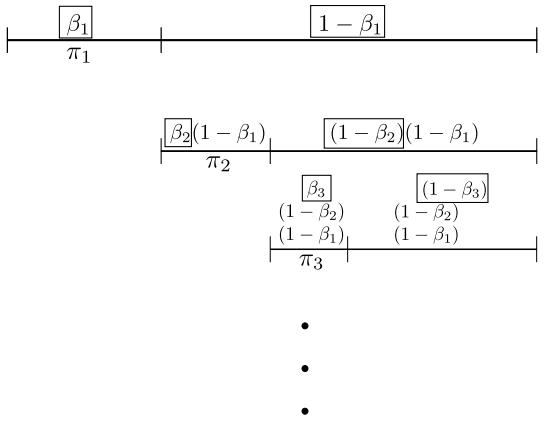

Figure 6.1.: Stick-breaking construction

We use $\boldsymbol{\pi} \sim \mathrm{GEM}(\alpha)$ to indicate a set of mixture weights sampled from this process. In short

$$\begin{aligned} \boldsymbol{\pi} &\sim \mathrm{GEM}(\alpha) & \boldsymbol{\pi} &= (\pi_1, \pi_2, \ldots) \\ x_k &\sim H & k &= 1, 2, \ldots \\ G(x) &= \sum_{k=1}^{\infty} \pi_k \delta(x, x_k) \\ \implies G &\sim \mathrm{DP}(\alpha, H) \end{aligned}$$

6.2.4. Pólya urn construction

The purpose is to generate samples from the posterior predictive, $p(\tilde{x} \mid \mathcal{D}, \alpha, H)$ where $\mathcal{D} = \{x_n : x_n \sim G, G \sim DP(\alpha, H)\}$.

Theorem 6.2.3. *Let $G \sim \mathrm{DP}(\alpha, H)$. Let $h(x)$ be the density of the base measure H. Consider a set of N observations $x_n \sim G$ taking $K \leq N$ distinct values $\{\bar{x}_k\}_{k=1}^K$. The predictive distribution of the next observation is*

$$p(\tilde{x} \mid x_1, \ldots, x_N, \alpha, H) = \frac{1}{\alpha + N} \left(\alpha h(\tilde{x}) + \sum_k N_k \delta(\tilde{x}, \bar{x}_k) \right) \tag{6.10}$$

where $N_k = \sum_n \delta(x_n, \bar{x}_k)$ is the count of observations that equal $\bar{x}_k$.

Proof. TODO □

We can get a sample from this distribution via the generalised Pólya urn model:

Algorithm 17 Pólya urn construction

1: Assume we have a bag with N identical balls of K different colours (our observations) with the probability of drawing each of them being $\frac{1}{\alpha+N}$. We also have a special black ball which can be drawn with probability $\frac{\alpha}{\alpha+N}$.
2: Draw a ball.
3: **if** it's not black **then**
4: Record colour.
5: (Put back and add one more ball with the same colour.)
6: **else**
7: Draw a ball from the bag of yet unseen colours, following H.
8: Record new colour.
9: (Put back both the black ball and the ball with a new colour.)
10: The recorded colour follows the posterior predictive in (6.10).

This follows (6.10) exactly.

6.2.5. Chinese restaurant process

Since $G \sim \mathrm{DP}(\alpha, H)$ is almost surely a discrete probability measure, if we draw N observations $x_n \sim G$, we will only have $K \leq N$ unique observations $\{\bar{x}_k\}_{k=1}^K$. We can view these as clusters. Let $\{z_n\}_{n=1}^N$ be cluster indicators, i.e. $z_n =$ the cluster number of x_n or equivalently $x_n = \bar{x}_{z_n}$. An equivalent version of (6.10) can be written down, caring only about the cluster numbers:

$$p(\tilde{z} \mid z_1, \ldots, z_N, \alpha, H) = \frac{1}{\alpha + N} \left(\alpha \delta(\tilde{z}, K+1) + \sum_{k=1}^{K} N_k \delta(\tilde{z}, k) \right) \tag{6.11}$$

We can get a sample from this distribution via the Chinese restaurant process, similar to the Pólya urn model in Algorithm 17:

Algorithm 18 Chinese restaurant process

1: Assume there are K occupied tables (clusters) at the restaurant numbered from 1 to K. The table k has N_k customers already sitting there (observations of cluster k), with the total of N customers. A new customer sits at an occupied table k with probability $\frac{N_k}{\alpha+N}$ and chooses a new table with probability $\frac{\alpha}{\alpha+N}$.
2: New customer comes.
3: The table number they choose follows the posterior predictive in (6.11) (with $K+1$ corresponding to choosing an unoccupied table).

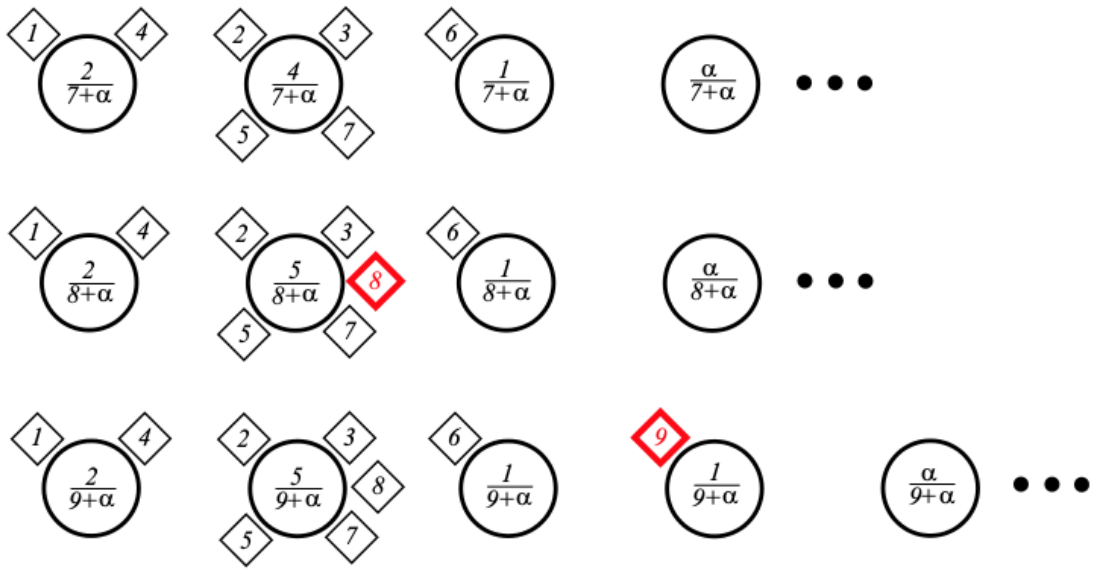

Figure 6.2.: (Figure from Erik Suddherth's PhD) Chinese restaurant process interpretation of the partitions induced by the Dirichlet process $\mathrm{DP}(\alpha, H)$. Tables (circles) are analogous to clusters, and customers (diamonds) to a series of observations. *Top row*: A starting configuration, in which seven customers occupy three tables. Each table is labeled with the probability that the next customer sits there. *Middle row*: New customers sit at occupied table k with probability proportional to the number of previously seated diners N_k. In this example, the eighth customer joins the most popular, and hence likely, table. *Bottom row*: Customers may also sit at one of the infinitely many unoccupied tables. The ninth diner does this.

The number of occupied tables K almost surely approaches $\alpha \log(N)$ as $N \to \infty$.

6.2.6. Dirichlet process mixtures

The purpose is to cluster observations. We can't model continuous observations directly using a Dirichlet processes because the samples from them are almost surely discrete

probability measures. Also, the posterior measure assigned to x_i would never beinfluenced by observations $x_j \neq x_i$, regardless of their proximity.

The Dirichlet process mixtures model is as follows:

$$\begin{aligned} G &\sim \mathrm{DP}(\alpha, H) \\ \bar{\theta}_n &\sim G & n = 1, \dots, N \\ x_n &\sim F(\bar{\theta}_n) \end{aligned}$$

where G is being sampled from $\mathrm{DP}(\alpha, H)$ via the stick-breaking construction:

$$\begin{aligned} \boldsymbol{\pi} &\sim \mathrm{GEM}(\alpha) & \boldsymbol{\pi} &= (\pi_1, \pi_2, \dots) \\ \theta_k &\sim H(\lambda) & k &= 1, 2, \dots \\ G(\theta) &= \sum_{k=1}^{\infty} \pi_k \delta(\theta, \theta_k) \end{aligned}$$

this solves the problem of inability of the DP to model the distribution of observations directly. Now two observations x_i, x_j are considered to be from the same cluster of $\bar{\theta}_n$ if both are $\sim F(\bar{\theta}_n)$.

7. Probabilistic programming (Anglican)

7.1. How it works

7.1.1. Notation

The syntax is as follows

```
[assume symbol <expr>]
[observe (<random proc> <arg> ... <arg>) <const>]
[predict <expr>]
```

where `assume`'s are either deterministic or random variables declarations, `observe`'s condition the distribution of the `assume`'d variables and `predict`'s give samples from the posteriors of the corresponding `<expr>`'s.

Probability of an execution trace is

$$\tilde{p}(\mathbf{y}, \mathbf{x}) = \prod_{n=1}^{N} p(y_n \mid \boldsymbol{\theta}_{t_n}, \mathbf{x}_n)\tilde{p}(\mathbf{x}_n \mid \mathbf{x}_{n-1}) \tag{7.1}$$

$$\tilde{p}(\mathbf{x}_n \mid \mathbf{x}_{n-1}) = \prod_{k=1}^{|\mathbf{x}_n \setminus \mathbf{x}_{n-1}|} p(x_{n,k} \mid \boldsymbol{\theta}_{t_{n,k}}, x_{n,1:(k-1)}, \mathbf{x}_{n-1}) \tag{7.2}$$

$$p(y_n \mid \boldsymbol{\theta}_{t_n}, \mathbf{x}_n) = \text{likelihood of observed output } y_n \tag{7.3}$$

$$\text{tilde} = \text{distributions we can only sample from} \tag{7.4}$$

$$y_n = n^{\text{th}} \texttt{ observe}\text{'d output} \tag{7.5}$$

$$t_n = \text{type of } n^{\text{th}} \texttt{ observe}\text{'d main } \texttt{random proc} \tag{7.6}$$

$$\boldsymbol{\theta}_{t_n} = \text{arguments of } t_n \tag{7.7}$$

$$\mathbf{x}_n = \text{set of all random procedure application results computed before } p(y_n \mid \boldsymbol{\theta}_{t_n}, \mathbf{x}_n) \text{ is evaluated. I.e. before the } n^{\text{th}} \text{ observe.} \tag{7.8}$$

Whevever a `predict` is called, we want to sample from $\tilde{p}(\mathbf{x} \mid \mathbf{y}) \propto \tilde{p}(\mathbf{y}, \mathbf{x})$. A general overview of this can be seen in Figure 7.1.

7.1.2. Random databse

This is an Metropolis-Hastings (see Subsection 5.7.2) approach to inference. The proposal step of the MH algorithm is illustrated in Figure 7.2. Following the Algorithm 11, the proposal step consists of these steps:

- Pick a single variable $x_{n,k}$ from the $|\mathbf{x}|$ random draws uniformly randomly.

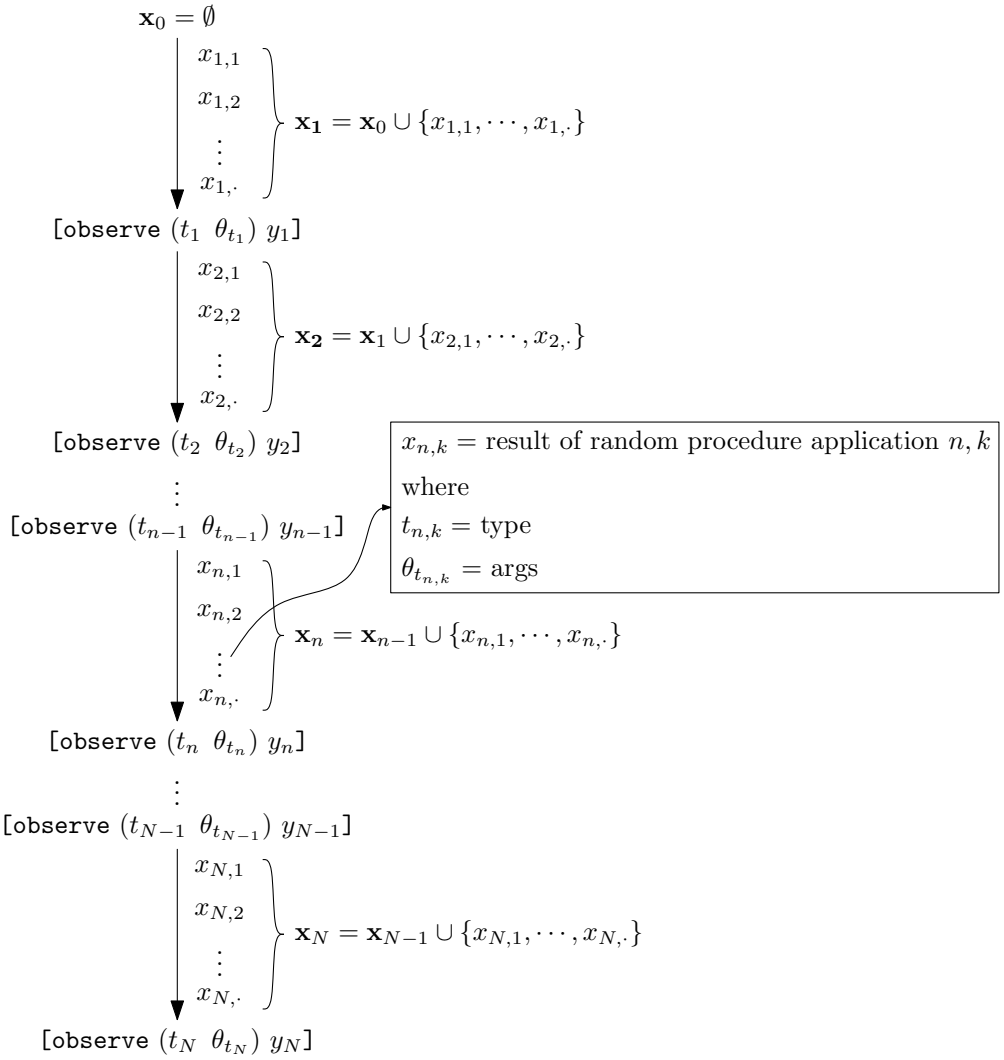

Figure 7.1.: A general overview of Anglican interpretation.

- Get a new random choice $x'_{n,k}$ by sampling from a kernel $x'_{n,k} \sim \kappa(\cdot \mid x_{n,k})$.
- Continue interpretation of program to get a new set of variables, $\mathbf{x}'$, that correspond to a new valid execution trace. (whenever a random procedure in the interpretation is the same as in $\mathbf{x}$, we reuse the existing value, only rescoring the conditional probability when necessary).
- $\mathbf{x}'$ is our MH proposal.

Following this procedure and notation in Figure 7.2, the proposal distribution can be expressed as

$$q(\mathbf{x}' \mid \mathbf{x}) = \frac{\kappa(x'_{n,k} \mid x_{n,k})}{|\mathbf{x}|} \frac{p(\mathbf{x}' \setminus \mathbf{x} \mid \mathbf{x}' \cap \mathbf{x})}{p(x'_{n,k} \mid \mathbf{x}' \cap \mathbf{x})} \tag{7.9}$$

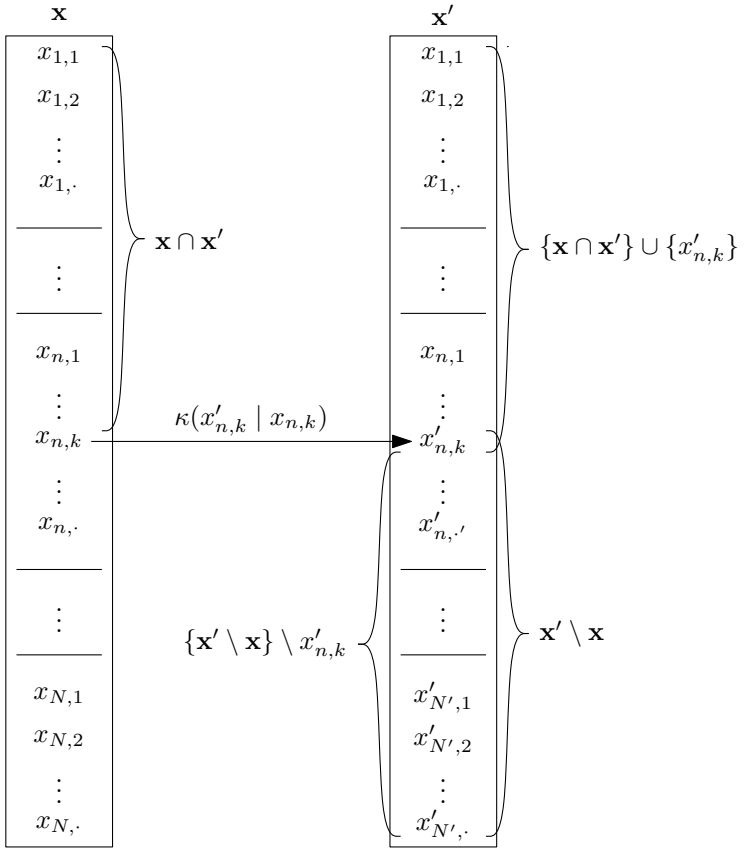

Figure 7.2.: Illustration of the RDB proposal step.

The $1/|\mathbf{x}|$ corresponds to randomly uniformly choosing a single variable. The $\kappa(x'_{n,k} \mid x_{n,k})$ corresponds to the proposal kernel. And finally,

$$\begin{aligned}\frac{p(\mathbf{x}' \setminus \mathbf{x} \mid \mathbf{x}' \cap \mathbf{x})}{p(x'_{n,k} \mid \mathbf{x}' \cap \mathbf{x})} &= p\left(\{\mathbf{x}' \setminus \mathbf{x}\} \setminus x'_{n,k} \mid x'_{n,k}, \mathbf{x} \cap \mathbf{x}'\right) \\ &= p\left(\{\mathbf{x}' \setminus \mathbf{x}\} \setminus x'_{n,k} \mid \{\mathbf{x} \cap \mathbf{x}'\} \cup \{x'_{n,k}\}\right)\end{aligned}$$

which corresponds to the probability of "the rest of execution given the past random choices of this proposed execution trace".

The acceptance probability can be written as

$$\begin{aligned}\mathcal{A}(\mathbf{x}' \mid \mathbf{x}) &= \min\left(1, \frac{p(\mathbf{y}', \mathbf{x}')q(\mathbf{y}, \mathbf{x} \mid \mathbf{y}', \mathbf{x}')}{p(\mathbf{y}, \mathbf{x})q(\mathbf{y}', \mathbf{x}' \mid \mathbf{y}, \mathbf{x})}\right) \\ &= \min\left(1, \frac{p(\mathbf{y} \mid \mathbf{x}')p(\mathbf{x}')q(\mathbf{x} \mid \mathbf{x}')}{p(\mathbf{y} \mid \mathbf{x})p(\mathbf{x})q(\mathbf{x}' \mid \mathbf{x})}\right) \qquad (7.10)\end{aligned}$$

In the *propose from prior* case, a new random choice $x'_{n,k}$ is obtained by continuing the interpretation from $x_{n,k-1}$ which means the expression of the kernel becomes $\kappa(x'_{n,k} \mid x_{n,k}) = p(x'_{n,k} \mid \mathbf{x}' \cap \mathbf{x})$. Note that the reverse kernel becomes $\kappa(x_{n,k} \mid x'_{n,k}) = p(x_{n,k} \mid \mathbf{x} \cap \mathbf{x}')$. Hence the expressions for the proposal distribution (in both ways) become

$$q(\mathbf{x}' \mid \mathbf{x}) = \frac{p(\mathbf{x}' \setminus \mathbf{x} \mid \mathbf{x}' \cap \mathbf{x})}{|\mathbf{x}|}$$
$$q(\mathbf{x} \mid \mathbf{x}') = \frac{p(\mathbf{x} \setminus \mathbf{x}' \mid \mathbf{x} \cap \mathbf{x}')}{|\mathbf{x}'|}$$

Substituting this to the acceptance probability in (7.10), we obtain

$$\mathcal{A}(\mathbf{x}' \mid \mathbf{x}) = \min\left(1, \frac{p(\mathbf{y} \mid \mathbf{x}')p(\mathbf{x}')|\mathbf{x}|p(\mathbf{x} \setminus \mathbf{x}' \mid \mathbf{x} \cap \mathbf{x}')}{p(\mathbf{y} \mid \mathbf{x})p(\mathbf{x})|\mathbf{x}'|p(\mathbf{x}' \setminus \mathbf{x} \mid \mathbf{x}' \cap \mathbf{x})}\right) \tag{7.11}$$

Summary: we just keep proposing and accepting and whenever a `predict` is needed, we just report the current (or the corresponding function of a subset of) $\mathbf{x}$.

7.1.3. Sequential Monte Carlo

In this case, we follow the Algorithm 9. We show in Figure 7.3 an illustration for one SMC iteration, adopting the notation in this chapter.

- The proposal is done by just continuing interpretation, i.e. $q\left(\mathbf{x}_n^{(\ell)} \mid \mathbf{x}_{n-1}^{A_{n-1}^{(\ell)}}, \mathbf{y}_n; \boldsymbol{\theta}\right) = p\left(\mathbf{x}_n^{(\ell)} \mid \mathbf{x}_{n-1}^{A_{n-1}^{(\ell)}}\right)$.
- The weights calculation then simplifies to $p\left(y_n \mid \mathbf{x}_{n-1}^{A_{n-1}^{(\ell)}}\right)$. TODO: Verify.
- Whenever a `predict` is needed, we can resample from

  $$\hat{p}(\mathrm{d}\mathbf{x}_{1:n} \mid y_{1:n}; \boldsymbol{\theta}) = \sum_{\ell} \hat{w}_t^{(\ell)} \delta_{\mathbf{x}_{1:n}^{(\ell)}}(\mathrm{d}\mathbf{x}_{1:n})$$

  to get a sample from the posterior.

7.1.4. Particle Gibbs

Here, we follow the Particle Gibbs algorithm described in Algorithm 16. The illustrations are below in Figure 7.4, and Figure 7.5. Whenever a `predict` is needed, a sample of the posterior of the execution trace can be obtained by sampling from

$$\hat{p}(\mathrm{d}\mathbf{x}_n \mid y_{1:n}, \boldsymbol{\theta}) = \sum_{\ell} \hat{w}_n^{(\ell)} \delta_{\mathbf{x}_n^{(\ell)}}(\mathrm{d}\mathbf{x}_n)$$

(This is *with* the retained particle).

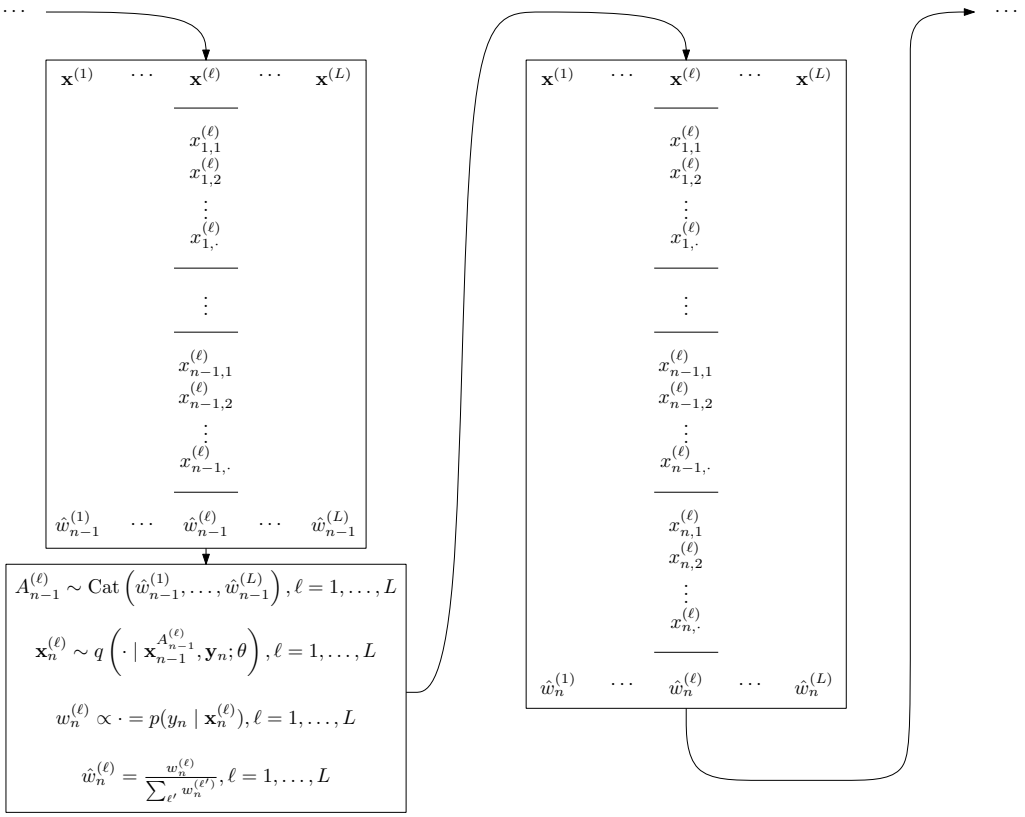

Figure 7.3.: Illustration of the SMC iteration n.

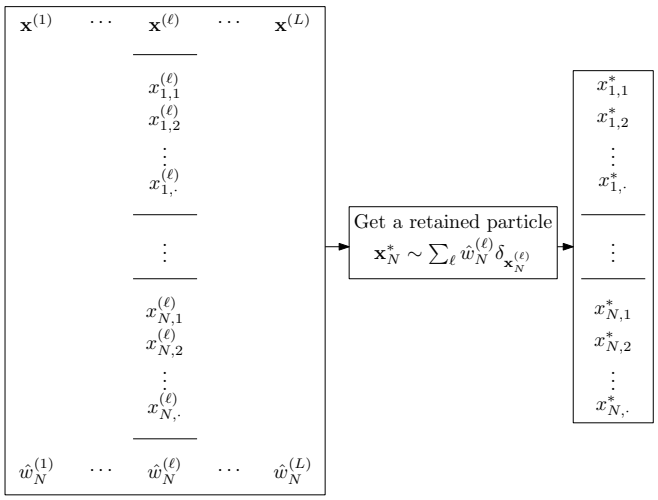

Figure 7.4.: Initialisation of the Particle Gibbs sampler.

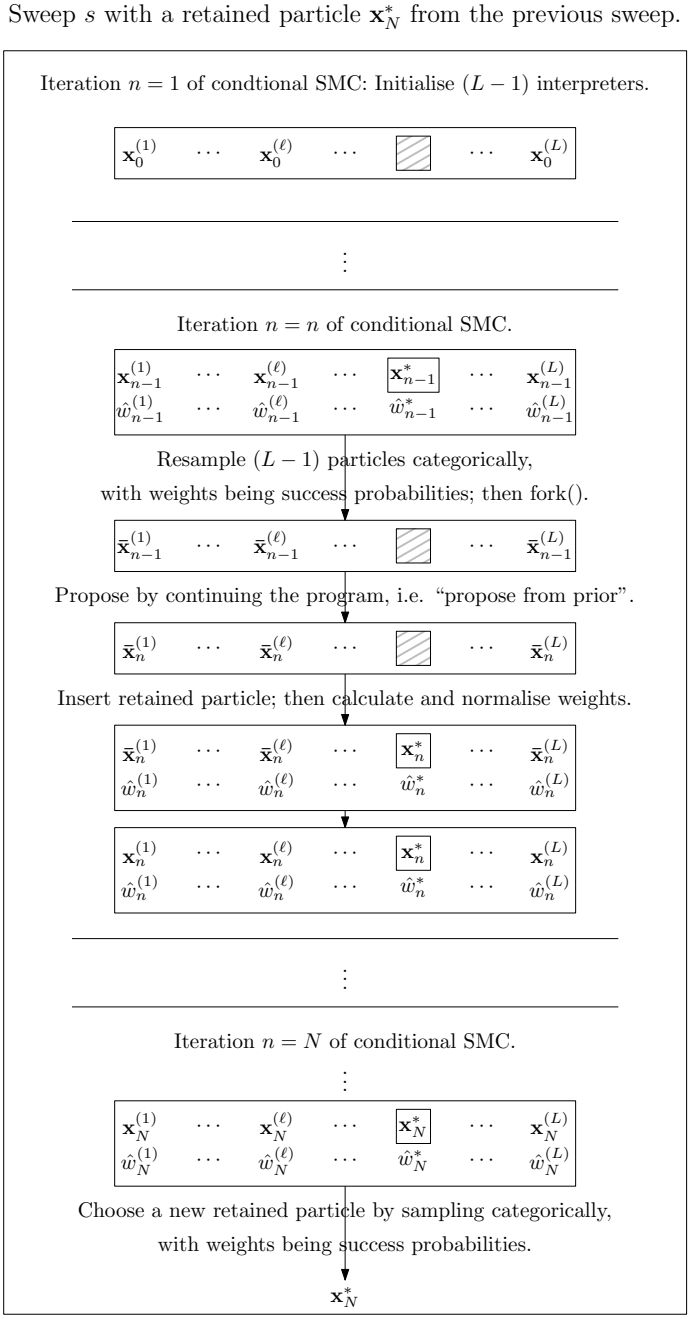

Figure 7.5.: Sweep s of the Particle Gibbs sampler.

7.2. Testing

7.2.1. Unit and measure tests

Calculate KL divergences for discrete sample spaces and KS test statistics for continuous sample spaces.

7.2.2. Conditional measure tests

ERPs

The purpose is to test whether the `*-lnpdf` functions work. For some distributions f and g, if we `assume` $\boldsymbol{\theta} \sim f$, then `observe` $\mathcal{D} = \{y_n : y_n \sim g(\ldots, \boldsymbol{\theta})\}_{n=1}^{N}$, and finally `predict` $\boldsymbol{\theta} \mid \mathcal{D}$, the inference engine will evaluate the `*-lnpdf` functions of g in order to characterise $\tilde{p}(\mathbf{x} \mid \mathbf{y}) \propto \tilde{p}(\mathbf{y}, \mathbf{x}) = \prod_n p(y_n \mid \theta_{t_n}, \mathbf{x}_n)\tilde{p}(\mathbf{x}_n \mid \mathbf{x}_{n-1})$. We can then test whether the `predict`'s follow the true distribution of $\boldsymbol{\theta} \mid \mathcal{D}$. Using this fact and taking advantage of conjugate pairs described in Chapter 3 and on Wikipedia, we can test the ERPs in the system as follows.

Bernoulli	
$\theta \sim \text{Beta}(\alpha, \beta)$	`[assume theta (beta a b)]`
$x \mid \theta \sim \text{Ber}(\theta)$	`[observe (flip theta) x1]` $\cdots$
$\mathcal{D} = \{x_n\}$	`[observe (flip theta) xN]`
$\theta \mid \mathcal{D} \sim \text{Beta}(\alpha + N_1, \beta + N_0)$	`[predict theta]`

Binomial	
$\theta \sim \text{Beta}(\alpha, \beta)$	`[assume theta (beta a b)]`
$x \mid \theta \sim \text{Bin}(T, \theta)$	`[observe (binomial theta T) x1]` $\cdots$
$\mathcal{D} = \{x_n\}$	`[observe (binomial theta T) xN]`
$\theta \mid \mathcal{D} \sim \text{Beta}(\alpha + \sum_n x_n, \beta + TN - \sum_n x_n)$	`[predict theta]`

Poisson	
$\lambda \sim \text{Gamma}(\alpha, \beta)$	`[assume l (gamma a b)]`
$x \mid \theta \sim \text{Poi}(\lambda)$	`[observe (poisson l) x1]` $\cdots$
$\mathcal{D} = \{x_n\}$	`[observe (poisson l) xN]`
$\lambda \mid \mathcal{D} \sim \text{Gamma}(\alpha + \sum_n x_n, \beta + N)$	`[predict l]`

Categorical	
$\boldsymbol{\theta} \sim \text{Dir}(\boldsymbol{\alpha}), \boldsymbol{\theta}, \boldsymbol{\alpha} \in \mathbb{R}^K$	`[assume ...]`
$x \mid \boldsymbol{\theta} \sim \text{Dir}(\boldsymbol{\theta})$	`[observe ...]` $\cdots$
$\mathcal{D} = \{x_n\}$	`[observe ...]`
$\boldsymbol{\theta} \mid \mathcal{D} \sim \text{Dir}(\boldsymbol{\alpha} + (n_1, \ldots, n_K)^T)$	`[predict ...]`

Univariate Normal with known variance	
Fix σ^2	`[assume var #var#]`
$\mu \sim \mathcal{N}(\mu_0, \sigma_0^2)$	`[assume mu (normal mu0 var0)]`
$x \mid \mu \sim \mathcal{N}(\mu, \sigma^2)$	`[observe (normal mu var) x1]` $\cdots$
$\mathcal{D} = \{x_n\}$	`[observe (normal mu var) xN]`
$\mu \mid \mathcal{D} \sim \mathcal{N}\left(\frac{\frac{\mu_0}{\sigma_0^2} + \frac{\sum_n x_n}{\sigma^2}}{\frac{1}{\sigma_0^2} + \frac{N}{\sigma^2}}, \left(\frac{1}{\sigma_0^2} + \frac{N}{\sigma^2}\right)^{-1}\right)$	`[predict mu]`

8. Neural networks

8.1. Feedforward neural networks

Not really Bayesian, but whatever.

8.1.1. Notation

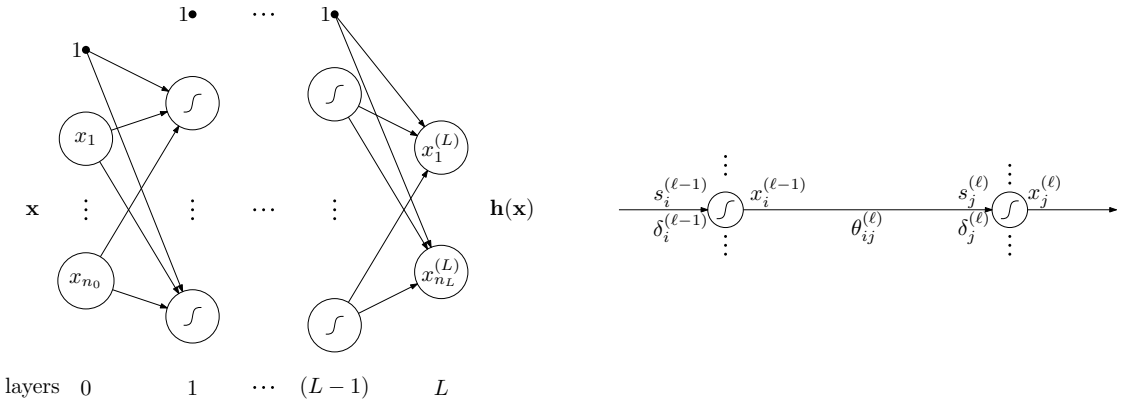

Figure 8.1.: Feedforward neural network.

There are L layers, n_ℓ at layer ℓ. $\theta_{ij}^{(\ell)}$ is the weight where

- layer $\ell = 1, \ldots, L$
- input $i = 0, \ldots, n_{\ell-1}$ (includes the intercept)
- output $j = 1, \ldots, n_\ell$.

The values of the neurons are

$$x_j^{(\ell)} = g\left(\sum_{i=0}^{n_{\ell-1}} \theta_{ij}^{(\ell)} x_i^{(\ell-1)}\right) \tag{8.1}$$

$$= g\left(\boldsymbol{\theta}_j^{(\ell)T} \mathbf{x}^{(\ell-1)}\right) \tag{8.2}$$

$$= g\left(s_j^{(\ell)}\right) \tag{8.3}$$

where

- $x_0^{(\ell)} = 0, \ell = 1, \ldots, L-1$
- $g : \mathbb{R} \to \mathbb{R}$ is a differentiable, nonlinear *activation function*

- $\boldsymbol{\theta}_j^{(\ell)} := (\theta_{0j}^{(\ell)}, \ldots, \theta_{n_{\ell-1},j}^{(\ell)})^T \in \mathbb{R}^{n_{\ell-1}+1}$ are the weights from layer $(\ell-1)$ to neuron j in the layer ℓ.
- $\mathbf{x} = (x_1, \ldots, x_{n_0})^T$ is the input.
- $\mathbf{x}^{(\ell)} := (x_0^{(\ell)}, \ldots, x_{n_\ell}^{(\ell)})^T \in \mathbb{R}^{n_\ell+1}$ are the neurons in layer $\ell = 0, \ldots L-1$. Note that $\mathbf{x}^{(0)} = \left(x_0^{(0)}, \mathbf{x}^T\right)^T$.
- $\mathbf{x}^{(L)} := (x_1^{(L)}, \ldots, x_{n_L}^{(L)})^T \in \mathbb{R}^{n_L}$ is the predicted output.
- $s_j^{(\ell)}$ is the *signal* for the j-th neuron in the ℓ-th layer. This gets fed into the *activation function* to get the (j,ℓ)-th neuron.

We can group the weights further, $\boldsymbol{\Theta}^{(\ell)} = (\boldsymbol{\theta}_1^{(\ell)}, \ldots, \boldsymbol{\theta}_{n_\ell}^{(\ell)}) \in \mathbb{R}^{(n_{\ell-1}+1)\times n_\ell}$, so that

$$\mathbf{x}^{(\ell)} = g\left(\boldsymbol{\Theta}^{(\ell)T}\mathbf{x}^{(\ell-1)}\right) \tag{8.4}$$

Note the slight inaccuracy of notation in (8.4): we are overloading g (for multivariable inputs and outputs) and also the left hand side should be $(x_1^{(\ell)}, \ldots, x_{n_\ell}^{(\ell)})^T$ (it should not include the intercept term). We can finally group $\boldsymbol{\Theta} := (\boldsymbol{\Theta}^{(1)}, \ldots, \boldsymbol{\Theta}^{(L)})$.

We define the mapping from input to output to be $\mathbf{h}$. The error on the example $(\mathbf{x}, \mathbf{y})$ is $e(\boldsymbol{\Theta})$ which is usually an analytic function of $\mathbf{y}$ and $\mathbf{x}^{(L)}$ (something like norm of the difference squared). Note that we subscript it with n if the error refers to the n-th example. The total error is $E(\boldsymbol{\Theta}) = \sum_n e_n(\boldsymbol{\Theta})$.

8.1.2. Backpropagation algorithm

We need to find $\operatorname{grad}_{\boldsymbol{\Theta}} e(\boldsymbol{\Theta})$, which is finding $\frac{\partial e(\boldsymbol{\Theta})}{\partial \theta_{ij}^{(\ell)}}$ for all i, j, ℓ (evaluated at the current values of neurons; we sometimes omit this for clarity). Since

$$\frac{\partial e(\boldsymbol{\Theta})}{\partial \theta_{ij}^{(\ell)}} = \frac{\partial e(\boldsymbol{\Theta})}{\partial s_j^{(\ell)}} \frac{\partial s_j^{(\ell)}}{\partial \theta_{ij}^{(\ell)}} \tag{8.5}$$

and

$$s_j^{(\ell)} = \sum_{i=0}^{n_{\ell-1}} \theta_{ij}^{(\ell)} x_i^{(\ell-1)} \tag{8.6}$$

$$\implies \frac{\partial s_j^{(\ell)}}{\partial \theta_{ij}^{(\ell)}} = x_i^{(\ell-1)} \tag{8.7}$$

we only need

$$\delta_j^{(\ell)} := \frac{\partial e(\boldsymbol{\Theta})}{\partial s_j^{(\ell)}} \tag{8.8}$$

to evaluate (8.5).

For the final layer. We can find

$$\delta_j^{(L)} = \frac{\partial e(\mathbf{\Theta})}{\partial s_j^{(L)}} \tag{8.9}$$

$$= \frac{\partial e(\mathbf{\Theta})}{\partial x_j^{(L)}} \frac{\partial x_j^{(L)}}{\partial s_j^{(L)}} \tag{8.10}$$

$$= \frac{\partial e(\mathbf{\Theta})}{\partial x_j^{(L)}} g'\left(s_j^{(L)}\right) \tag{8.11}$$

analytically for $j = 1, \ldots, n_L$.

For the layers before. We can find

$$\delta_i^{(\ell-1)} = \frac{\partial e(\mathbf{\Theta})}{\partial s_i^{(\ell-1)}}\Big|_{x_i^{(\ell-1)}} \tag{8.12}$$

$$= \sum_{j=1}^{n_\ell} \frac{\partial e(\mathbf{\Theta})}{\partial s_j^{(\ell)}} \frac{\partial s_j^{(\ell)}}{\partial s_i^{(\ell-1)}} \tag{8.13}$$

$$= \sum_{j=1}^{n_\ell} \frac{\partial e(\mathbf{\Theta})}{\partial s_j^{(\ell)}} \frac{\partial s_j^{(\ell)}}{\partial x_i^{(\ell-1)}} \frac{\partial x_i^{(\ell-1)}}{\partial s_i^{(\ell-1)}} \tag{8.14}$$

$$= \sum_{j=1}^{n_\ell} \delta_j^{(\ell)} \cdot \theta_{ij}^{(\ell)} \cdot g'\left(s_i^{(\ell-1)}\right) \tag{8.15}$$

recursively for $i = 1, \ldots, n_{\ell-1}$.

The algorithm becomes

Algorithm 19 Backpropagation algorithm (SGD)

1: Initialise $\mathbf{\Theta}$ at random.
2: **repeat**
3: Pick example $n \in \{1, \ldots, N\}$ at random.
4: Forward pass: compute all $x_j^{(\ell)}$'s using (8.1).
5: Backward pass: compute all $\delta_j^{(\ell)}$'s using (8.11) and (8.15).
6: Update all $\theta_{ij}^{(\ell)}$'s ($i = 0, \ldots, n_{\ell-1}$ and $j = 1, \ldots, n_\ell$):

$$\theta_{ij}^{(\ell)} \leftarrow \theta_{ij}^{(\ell)} - \eta \frac{\partial e_n(\mathbf{\Theta})}{\partial \theta_{ij}^{(\ell)}} \tag{8.16}$$

$$= \theta_{ij}^{(\ell)} - \eta x_i^{(\ell-1)} \delta_j^{(\ell)} \tag{8.17}$$

7: **until** it is time to stop.
8: Return the final weights $\mathbf{\Theta}$.

8.1.3. Full specifications

In order to fully specify the network and the backpropagation algorithm, we need to specify *activation functions* g, their derivatives g', the error of one example e and its derivatives $\frac{\partial e(\mathbf{\Theta})}{\partial x_j^{(L)}}$. Then we can arbitrarily mix and match activation functions and error functions to give us the required output, etc.

Sigmoid activation

For $[0, 1]$ output:

$$g(x) = \frac{1}{1 + \exp(-x)} \tag{8.18}$$

$$g'(x) = g(x)(1 - g(x)). \tag{8.19}$$

Identity activation

For $\mathbb{R}$ output:

$$g(x) = x \tag{8.20}$$

$$g'(x) = 1. \tag{8.21}$$

Softmax activation

For vector output that sums to one:

$$g(\mathbf{x})_i = \frac{\exp(x_i)}{\sum_j \exp(x_j)} \tag{8.22}$$

$$\frac{\partial g(\mathbf{x})_i}{\partial x_j} = g(\mathbf{x})_i(\delta_{ij} - g(\mathbf{x})_j). \tag{8.23}$$

Note that δ_{ij} is the Kronecker delta function and $= 1$ if $i = j$ and $= 0$ otherwise.

Exponential activation

For $\mathbb{R}^+$ output:

$$g(x) = \exp(x) \tag{8.24}$$

$$g'(x) = \exp(x). \tag{8.25}$$

Tanh activation

For $[0, 1]$ output:

$$g(x) = \tanh(x) \tag{8.26}$$

$$= \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)} \tag{8.27}$$

$$g'(x) = 1 - \tanh^2(x). \tag{8.28}$$

Rectifier activation

For R^+ output:

$$g(x) = \max(0, x) \tag{8.29}$$

$$g'(x) = \begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x > 0 \end{cases}. \tag{8.30}$$

Use whatever for $g'(0)$.

Squared norm error

$$e(\mathbf{\Theta}) = \frac{1}{2} \left\| \mathbf{x}^{(L)} - \mathbf{y} \right\|^2 \tag{8.31}$$

$$= \frac{1}{2} \left(\mathbf{x}^{(L)} - \mathbf{y} \right)^T \left(\mathbf{x}^{(L)} - \mathbf{y} \right) \tag{8.32}$$

$$\operatorname{grad}_{\mathbf{x}^{(L)}} e(\mathbf{\Theta}) = \left(\mathbf{x}^{(L)} - \mathbf{y} \right) \tag{8.33}$$

8.1.4. Feedforward neural networks for conditional density estimation

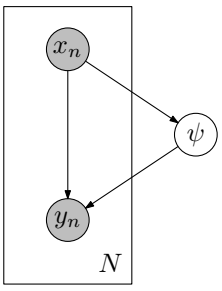

Figure 8.2.: Graphical model for conditional density estimation.

We are interested in estimating $y \mid x$ using a neural network using training data $\{x_n, y_n\}$. We can set up the following generative model

$$\psi := \eta(x_n) \tag{8.34}$$

$$y_n \mid x_n, \psi \sim F(x_n, \psi) \tag{8.35}$$

where η is our neural network and F is some generative model with some density function f. By maximising the likelihood

$$\mathcal{L}(\psi) = \prod_n f(y_n \mid x_n, \psi) \tag{8.36}$$

we can find the MLE estimate of ψ. This is equivalent to setting the error function of the neural network η to be the negative log-likelihood of one data point:

$$e_n(\mathbf{\Theta}) = -\log f(y_n \mid x_n, \psi) \tag{8.37}$$

where $\mathbf{\Theta}$ are the weights of the neural network. If we can design a generative model F in such a way that we can calculate the derivatives $\frac{\partial e_n(\mathbf{\Theta})}{\partial x_j^{(L)}}$, then we are done (SGD will minimise $\sum_n e_n$ which is total negative log-likelihood). We need to mix and match the activation functions for the last layer to match the support of ψ.

Example

Let the generative model be

$$f(y \mid x, \psi) = \mathcal{N}(y \mid x, \psi) \tag{8.38}$$
$$= \mathcal{N}(y \mid m, \sigma^2) \tag{8.39}$$

where

$$\psi = (m, \sigma). \tag{8.40}$$

Let the input to the a 2 layer (1 input layer, 1 hidden layer, 1 output layer) neural network be $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$ and the output be

- $x_1^{(2)} \in \mathbb{R}$ which is approximating m, and
- $x_2^{(2)} \in \mathbb{R}^+$ which is approximating σ.

Our negative log-likelihood for one data point a.k.a. loss function is

$$e(\mathbf{\Theta}) = -\log \mathcal{N}\left(y \mid x_1^{(2)}, \left(x_2^{(2)}\right)^2\right) \tag{8.41}$$
$$= -\log \frac{1}{x_2^{(2)}\sqrt{2\pi}} \exp -\frac{(x_1^{(2)} - y)^2}{2\left(x_2^{(2)}\right)^2} \tag{8.42}$$
$$= \log(\sqrt{2\pi}) + \log(x_2^{(2)}) + \frac{(x_1^{(2)} - y)^2}{2\left(x_2^{(2)}\right)^2} \tag{8.43}$$

and the derivatives are

$$\frac{\partial e}{\partial x_1^{(2)}} = \frac{\left(x_1^{(2)} - y\right)}{\left(x_2^{(2)}\right)^2} \tag{8.44}$$
$$\frac{\partial e}{\partial x_2^{(2)}} = \frac{1}{x_2^{(2)}} - \left(x_1^{(2)} - y\right)^2 \left(x_2^{(2)}\right)^{-3} \tag{8.45}$$
$$= \frac{1}{x_2^{(2)}} \left(1 - \frac{(x_1^{(2)} - y)^2}{\left(x_2^{(2)}\right)^2}\right). \tag{8.46}$$

We choose the identity and exponential activation functions for m and σ respectively.

8.2. Convolutional neural networks

8.3. Deep generative models

8.3.1. Deep directed networks

8.3.2. Deep Boltzmann machines

8.3.3. Deep belief networks

8.3.4. Deep autoencoders

9. Statistical hypothesis testing

9.1. Kullback-Leibler divergence

A.k.a. *KL divergence*, or *relative enropy*. KL divergence between the distributions $p(\mathbf{x})$ and $q(\mathbf{x})$ for $\mathbf{x} \in \mathcal{X}$, denoted $\mathrm{KL}(p \parallel q)$ or $\mathrm{KL}(p, q)$, is a measure of similarity between p and q and is given by

$$\begin{aligned}\mathrm{KL}(p \parallel q) &= -\int_{\mathcal{X}} p(\mathbf{x}) \ln q(\mathbf{x}) \, \mathrm{d}\mathbf{x} - \left(-\int_{\mathcal{X}} p(\mathbf{x}) \ln p(\mathbf{x}) \, \mathrm{d}\mathbf{x}\right) \\ &= -\int_{\mathcal{X}} p(\mathbf{x}) \ln \frac{q(\mathbf{x})}{p(\mathbf{x})} \, \mathrm{d}\mathbf{x} \end{aligned} \tag{9.1}$$

Note that $\mathrm{KL}(p \parallel q) \not\equiv \mathrm{KL}(q \parallel p)$.

Claim 9.1.1. $\mathrm{KL}(p \parallel q) \geq 0$ *with equality if and only if* $p(\mathbf{x}) = q(\mathbf{x})$.

Proof. asdf □

9.2. Kolmogorov-Smirnov test

9.2.1. Kolmogorov-Smirnov statistic

Null hypothesis, often denoted by H_0 is a general statement or a default position saying there is no relationship between two measured phenomena.

The Kolmogorov (KS) test quantifies a distance between

- The empirical distribution function (or the empirical cdf) and the cdf of the reference function (H_0 = sample is drawn from the reference distribution), or
- The empirical cdfs of two samples (H_0 = samples are drawn from the same distribution).

The empirical cdf F_N for N iid observations $\{x_n\}$ is

$$F_N(x) \triangleq \frac{1}{N} \sum_n \mathbb{I}(x_n \leq x) \tag{9.2}$$

basically $F_N(x) = \frac{1}{N} \times$ number of samples less than or equal to x.

The KS statistic for a given cdf $F(x)$ is

$$D_N(x) \triangleq \sup_x |F_N(x) - F(x)| \tag{9.3}$$

By Glivenko-Cantelli theorem, if $\{x_n\} \sim F$, then $D_N \to 0$ almost surely when $N \to \infty$.

A. Particle filter animation

Needs to be opened in Adobe Reader.

Bibliography

[1] Daphne Koller and Nir Friedman. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*. The MIT Press, 2009.