Personal notes – Bayesian machine learning

Tuan Anh Le

October 16, 2014

Contents

1. Notation

$\{a_n\}$	Same as $\{a_n\}_{n=1}^N$ and $\{a_1, \dots, a_N\}$ – 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).
$\mathrm{diag}(x_1, \dots, x_N)$	Diagonal matrix formed from the elements $x_1, \dots, x_N$.
$\mathbb{I}(\cdot)$	Indicator function, equal to 1 if the argument is true, 0 otherwise.

2. Basics

2.1. Probability distributions

Summarised in Table 2.1

2.2. Stats

2.2.1. Kolmogorov-Smirnov test

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

$$F_N(x) \triangleq \frac{1}{N} \sum_n \mathbb{I}(x_n \leq x) \tag{2.1}$$

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

$$D_N(x) \triangleq \sup_x |F_N(x) - F(x)| \tag{2.2}$$

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

2.2.2. 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

$$\mathrm{KL}(p \parallel q) = -\int_{\mathcal{X}} p(\mathbf{x}) \ln q(\mathbf{x})\,\mathrm{d}\mathbf{x} - \left(-\int_{\mathcal{X}} p(\mathbf{x}) \ln p(\mathbf{x})\,\mathrm{d}\mathbf{x}\right)$$

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)	$\boldsymbol{\alpha} \in (0,\infty)^K$	$\mathbf{x} \in [0,1]^K, \sum_k x_k = 1$	$\frac{\Gamma\left(\sum_k \alpha_k\right)}{\prod_k \Gamma(\alpha_k)} \prod_k x_k^{\alpha_k - 1}$	$\frac{\boldsymbol{\alpha}}{\sum_k \alpha_k}$	$\mathrm{var}[x_k] = \frac{\alpha_k(\alpha_0-\alpha_k)}{\alpha_0^2(\alpha_0+1)}$
Multinomial (Mult)	$N \in \mathbb{N}, \boldsymbol{\theta} \in [0,1]^K, \sum_k \theta_k = 1$	$\mathbf{x} \in \{0,\dots,N\}^K, \sum_k x_k = N$	$\frac{N!}{x_1! \cdots x_K!}\theta_1^{x_1}\cdots\theta_K^{x_K}$	$N\boldsymbol{\theta}$	$\mathrm{var}[x_k] = N\theta_k(1-\theta_k)$
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]

[a] 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 $\frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$.

$$= -\int_{\mathcal{X}} p(\mathbf{x}) \ln \frac{q(\mathbf{x})}{p(\mathbf{x})} \,\mathrm{d}\mathbf{x} \tag{2.3}$$

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

Claim 2.2.1. $\mathrm{KL}(p \parallel q) \geq 0$ *with equality if and only if* $p(\mathbf{x}) = q(\mathbf{x})$.

Proof. asdf □

2.3. Gaussian distribution

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

$$p(\mathbf{x}) = \frac{1}{\sqrt{|2\pi\boldsymbol{\Sigma}|}} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right] \tag{2.4}$$

$$= (2\pi)^{-D/2}|\boldsymbol{\Sigma}|^{-1/2} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right] \tag{2.5}$$

2.3.1. Linear Gaussian model

Given the marginal and conditional distributions to be

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Lambda}^{-1}) \tag{2.6}$$

$$p(\mathbf{y} \mid \mathbf{x}) = \mathcal{N}(\mathbf{y}; \mathbf{A}\mathbf{x}+\mathbf{b}, \mathbf{L}^{-1}) \tag{2.7}$$

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

$$p(\mathbf{y}) = \mathcal{N}\left(\mathbf{y}; \mathbf{A}\boldsymbol{\mu}+\mathbf{b}, \mathbf{L}^{-1} + \mathbf{A}\boldsymbol{\Lambda}^{-1}\mathbf{A}^T\right) \tag{2.8}$$

$$p(\mathbf{x} \mid \mathbf{y}) = \mathcal{N}\left(\mathbf{x}; \boldsymbol{\Sigma}\left\{\mathbf{A}^T\mathbf{L}(\mathbf{y}-\mathbf{b}) + \boldsymbol{\Lambda}\boldsymbol{\mu}\right\}, \boldsymbol{\Sigma}\right) \tag{2.9}$$

where

$$\boldsymbol{\Sigma} = \left(\boldsymbol{\Lambda} + \mathbf{A}^T\mathbf{L}\mathbf{A}\right)^{-1} \tag{2.10}$$

Why it works

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, $\mathbf{x}_n \sim f(\boldsymbol{\theta}), n = 1, \dots, N$, 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 a dash, $\boldsymbol{\alpha}'$. 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	$\alpha + \sum_n \mathbb{I}(x_n = 1), \beta + \sum_n \mathbb{I}(x_n = 0)$	$\mathrm{Ber}\left(\tilde{x} \mid \frac{\alpha'}{\alpha' + \beta'}\right)$
Binomial	θ	Beta	α, β	$\alpha + \sum_n x_n, \beta + \sum_n (T_n - x_n)$	$\mathrm{BetaBin}(\tilde{x} \mid \alpha', \beta')$
Poisson	λ	Gamma	α, β	$\alpha + \sum_n x_n, \beta + N$	$\mathrm{NB}\left(\tilde{x} \mid \alpha', \frac{1}{1+\beta'}\right)$
Categorical	$\boldsymbol{\theta} \in \mathbb{R}^K$	Dirichlet	$\boldsymbol{\alpha} \in \mathbb{R}^K$	$\boldsymbol{\alpha} + (n_1, \ldots, n_K)^T$	$\mathrm{Ber}\left(\tilde{x} \mid \frac{\alpha'_{\tilde{x}}}{\sum_k \alpha'_k}\right)$
Multinomial	$\boldsymbol{\theta} \in \mathbb{R}^K$	Dirichlet	$\boldsymbol{\alpha} \in \mathbb{R}^K$	$\boldsymbol{\alpha} + \sum_n \mathbf{x}_n$	$\mathrm{DirMult}(\tilde{\mathbf{x}} \mid \alpha', \tilde{T})$

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

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

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.

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

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.

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

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

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

where $D(\boldsymbol{\alpha})$ is the normalisation constant for the Dirichlet distribution, $\mathrm{Dir}(\boldsymbol{\alpha})$, 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. 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 $p(z_n) = \mathrm{Cat}(\boldsymbol{\pi}), \boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)$ 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:

$$\begin{aligned} p(\mathbf{x}_n \mid \boldsymbol{\theta}) &= \sum_{k=1}^{K} p(\mathbf{x}_n, z_n = k; \boldsymbol{\theta}) \\ &= \sum_{k=1}^{K} p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta}) p(z_n = k \mid \boldsymbol{\theta}) \\ &= \sum_{k=1}^{K} \pi_k p_k(\mathbf{x}_n \mid \boldsymbol{\theta}) \end{aligned} \tag{4.1}$$

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:

$$\begin{aligned} r_{nk}(\boldsymbol{\theta}) &\triangleq p(z_n = k \mid \mathbf{x}_n; \boldsymbol{\theta}) \\ &= \frac{p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta}) p(z_n = k \mid \boldsymbol{\theta})}{\sum_{k'=1}^{K} p(\mathbf{x}_n \mid z_n = k'; \boldsymbol{\theta}) p(z_n = k' \mid \boldsymbol{\theta})} \end{aligned} \tag{4.2}$$

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

$$\begin{aligned} z_n^* &= \arg\max_k r_{nk} \\ &= \arg\max_k \left\{\log p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta}) + \log(z_n = k \mid \boldsymbol{\theta})\right\} \end{aligned} \tag{4.3}$$

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:

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

$$
\begin{aligned}
&= \log \prod_{n=1}^{N} p(\mathbf{x}_n \mid \boldsymbol{\theta}) \\
&= \log \left\{ \prod_{n=1}^{N} \sum_{k=1}^{K} p(\mathbf{x}_n, z_n = k \mid \boldsymbol{\theta}) \right\} \\
&= \sum_{n=1}^{N} \log \sum_{k=1}^{K} p(\mathbf{x}_n, z_n = k \mid \boldsymbol{\theta}) \qquad (4.4)
\end{aligned}
$$

$$
\begin{aligned}
\ell_c(\boldsymbol{\theta}) &\triangleq \log p\left(\{\mathbf{x}_n, z_n\} \mid \boldsymbol{\theta}\right) \\
&= \log \prod_{n} p(\mathbf{x}_n, z_n \mid \boldsymbol{\theta}) \\
&= \sum_{n} \log p(\mathbf{x}_n, z_n \mid \boldsymbol{\theta}) \qquad (4.5)
\end{aligned}
$$

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 $\ell_c(\boldsymbol{\theta}) = \log p(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta})$.

4.1.1. EM algorithm

Maximise the likelihood

Goal is to maximise

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

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

$$\mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right) \triangleq \mathrm{E}_{\mathbf{Z} \sim \cdot \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}}}\left[\ell_c(\boldsymbol{\theta})\right] \qquad (4.6)$$

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

$$\mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right) = \sum_{\mathbf{Z}} p\left(\mathbf{Z} \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}}\right) \ln p\left(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta}\right) \qquad (4.7)$$

or

$$
\begin{aligned}
\mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right) &= \mathrm{E}_{\mathbf{Z} \sim \cdot \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}}}\left[\ln p\left(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta}\right)\right] \qquad (4.8) \\
&= \mathrm{E}_{\mathbf{Z} \sim \cdot \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}}}\left[\sum_{n} \ln p\left(\mathbf{x}_n, z_n \mid \boldsymbol{\theta}\right)\right] \qquad (4.9) \\
&= \sum_{n} \mathrm{E}_{z_n \sim \cdot \mid \mathbf{x}_n; \boldsymbol{\theta}^{\text{old}}}\left[\ln p(\mathbf{x}_n, z_n \mid \boldsymbol{\theta})\right] \qquad (4.10) \\
&= \sum_{n} \sum_{k} p\left(z_n = k \mid \mathbf{x}_n; \boldsymbol{\theta}^{\text{old}}\right) \ln p(\mathbf{x}_n, z_n = k \mid \boldsymbol{\theta}) \qquad (4.11) \\
&= \sum_{n} \sum_{k} r_{nk}\left(\boldsymbol{\theta}^{\text{old}}\right) \ln\left(\pi_k p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta})\right) \qquad (4.12)
\end{aligned}
$$

$$= \sum_n \sum_k r_{nk} \left(\boldsymbol{\theta}^{\text{old}}\right) (\ln \pi_k + \ln p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta})) \tag{4.13}$$

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

$$\ln p(\mathbf{X} \mid \boldsymbol{\theta}) = \mathcal{L}(q, \boldsymbol{\theta}) + \mathrm{KL}(q \parallel p) \tag{4.14}$$

where

$$\mathcal{L}(q, \boldsymbol{\theta}) = \sum_{\mathbf{Z}} q(\mathbf{Z}) \ln \frac{p(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta})}{q(\mathbf{Z})} \tag{4.15}$$

$$\mathrm{KL}(q \parallel p) = -\sum_{\mathbf{Z}} q(\mathbf{Z}) \ln \frac{p(\mathbf{Z} \mid \mathbf{X}, \boldsymbol{\theta})}{q(\mathbf{Z})} \tag{4.16}$$

because

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

The actual algorithm is as follows

Algorithm 1 EM algorithm for maximising the likelihood

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 $\boldsymbol{\theta}^{\text{new}} = \arg\max_{\boldsymbol{\theta}} \mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right)$.
6: **until** convergence.

E step. Hold $\boldsymbol{\theta}^{\text{old}}$, maximise $\mathcal{L}\left(q, \boldsymbol{\theta}^{\text{old}}\right)$ w.r.t. q. Since the quantity $\ln p(\mathbf{X} \mid \boldsymbol{\theta})$ in (4.14) is constant w.r.t. q, we can maximise $\mathcal{L}\left(q, \boldsymbol{\theta}^{\text{old}}\right)$ by minimising $\mathrm{KL}(q \parallel p)$. This can be done by setting the KL to 0 by setting $q(\mathbf{Z}) = p(\mathbf{Z} \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}})$.

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

$$
\mathcal{Q}\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}\right) + \ln p(\boldsymbol{\theta}) \tag{4.17}
$$

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

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

$$
\begin{aligned}
\ln p(\boldsymbol{\theta} \mid \mathbf{X}) &= \ln p(\mathbf{X} \mid \boldsymbol{\theta}) + \ln p(\boldsymbol{\theta}) - \ln p(\mathbf{X}) \\
&= \mathcal{L}(q, \boldsymbol{\theta}) + \mathrm{KL}(q \parallel p) + \ln p(\boldsymbol{\theta}) - \ln p(\mathbf{X})
\end{aligned} \tag{4.18}
$$

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.

4.1.2. Gaussian mixture model

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

$$p(z_n = k) = \pi_k \tag{4.19}$$
$$p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta}) = \mathcal{N}(\mathbf{x}_n \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{4.20}$$

for $n = 1, \ldots, N$ and $k = 1, \ldots, K$, where $\boldsymbol{\theta} = (\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}, k = 1, \ldots, K)$.

EM algorithm for GMM

Algorithm 3 EM algorithm for GMM

Initialise $\boldsymbol{\theta}^{\text{new}} = (\{\boldsymbol{\pi}_k^{\text{new}}, \boldsymbol{\mu}_k^{\text{new}}, \boldsymbol{\Sigma}_k^{\text{new}}\}, k = 1, \ldots, K)$.
repeat
    $\boldsymbol{\theta}^{\text{old}} \leftarrow \boldsymbol{\theta}^{\text{new}}$
    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
    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 $k = 1, \ldots, K$.
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

$$r_{nk_n}\left(\boldsymbol{\theta}^{\text{old}}\right) = p\left(z_n = k_n \mid \mathbf{x}_n; \boldsymbol{\theta}^{\text{old}}\right) \tag{4.21}$$

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 subsection) 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

$$\pi_k = \frac{\sum_n r_{nk}}{N} \tag{4.22}$$

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

For $\boldsymbol{\mu}_k$,

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

$$= -\sum_n r_{nk}\boldsymbol{\Sigma}_k^{-1}(\mathbf{x}_n - \boldsymbol{\mu}_k)$$

Setting this to zero, we get

$$\boldsymbol{\mu}_k = \frac{\sum_n r_{nk}\mathbf{x}_n}{\sum_n r_{nk}} \tag{4.23}$$

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

$$\boldsymbol{\Sigma}_k = \frac{\sum_n r_{nk}(\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_n r_{nk}} \tag{4.24}$$

4.2. Hidden Markov model

4.3. Linear regression

4.4. Logistic regression

4.5. Latent Dirichlet allocation

4.6. Linear dynamical systems

4.7. Principal components analysis

4.7.1. 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

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

$$= \mathbf{U}^T \mathbf{x}_n$$

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

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

$$\begin{aligned}
V &= \sum_{m=1}^{M} \text{var}(\text{dimension } m) \\
&= \sum_{m=1}^{M} \frac{1}{N} \sum_{n=1}^{N} (z_{nm} - \bar{z}_m)^2 \\
&\left(\text{where } \bar{z}_m = \frac{1}{N} \sum_{n=1}^{N} z_{nm} \right) && (4.25) \\
&= \frac{1}{N} \sum_{m=1}^{M} \sum_{n=1}^{N} \left(z_{nm}^2 - 2 z_{nm} \bar{z}_m + \bar{z}_m^2 \right) \\
&= \frac{1}{N} \sum_{m=1}^{M} \sum_{n=1}^{N} \left(\left(\mathbf{u}_m^T \mathbf{x}_n \right)^2 - 2 \left(\mathbf{u}_m^T \mathbf{x}_n \right) \left(\mathbf{u}_m^T \bar{\mathbf{x}} \right) + \left(\mathbf{u}_m^T \bar{\mathbf{x}} \right)^2 \right), \text{where } \bar{\mathbf{x}} = \frac{1}{N} \sum_{n=1}^{N} \mathbf{x}_n \\
&= \sum_{m=1}^{M} \mathbf{u}_m^T \left(\frac{1}{N} \sum_{n=1}^{N} \mathbf{x}_n \mathbf{x}_n^T - 2 \mathbf{x}_n \bar{\mathbf{x}}^T + \bar{\mathbf{x}} \bar{\mathbf{x}}^T \right) \mathbf{u}_m \\
&= \sum_{m=1}^{M} \mathbf{u}_m^T \left(\frac{1}{N} \sum_{n=1}^{N} (\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^T \right) \mathbf{u}_m \\
&= \sum_{m=1}^{M} \mathbf{u}_m^T \mathbf{S} \mathbf{u}_m && (4.26) \\
&\left(\text{where } \mathbf{S} = \frac{1}{N} \sum_{n=1}^{N} (\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^T \right) && (4.27)
\end{aligned}$$

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

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

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. $(\mathbf{u}_1, \dots, \mathbf{u}_M)$. 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:

$$\text{grad}_{\boldsymbol{\lambda}} \mathcal{L} = \mathbf{0} \tag{4.28}$$

$$\operatorname{grad}_{\mathbf{u}_m} \mathcal{L} = \mathbf{0}, m = 1, \ldots, M \tag{4.29}$$

From (4.28), we obtain $\mathbf{u}_m^T \mathbf{u}_m = 1, m = 1, \ldots, M$. From (4.29), we obtain

$$\begin{aligned}
\operatorname{grad}_{\mathbf{u}_m} \mathcal{L} &= 2\mathbf{S}\mathbf{u}_m - 2\lambda_m \mathbf{u}_m &(4.30)\\
&= 0 &(4.31)\\
\implies \mathbf{S}\mathbf{u}_m &= \lambda_m \mathbf{u}_m &(4.32)
\end{aligned}$$

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 (4.32) 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.

4.7.2. 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 $\boldsymbol{\theta} = \left(\mathbf{W}, \boldsymbol{\mu}, \sigma^2\right)$. Following Subsection 2.3.1, 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:

$$p(\mathcal{D} \mid \boldsymbol{\theta}) = \prod_{n=1}^{N} p(\mathbf{x}_n \mid \boldsymbol{\theta})$$

$$= \prod_{n=1}^{N} \mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}, \mathbf{C})$$

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

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

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

$$\mathbf{U}_M = [\mathbf{u}_1, \ldots, \mathbf{u}_M]$$
$$\mathbf{L}_M = \mathrm{diag}(\lambda_1, \ldots, \lambda_M)$$

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 (4.27)), 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 $\boldsymbol{\theta}^{\text{new}} = \left(\mathbf{W}^{\text{new}}, (\sigma^{\text{new}})^2\right)$. 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

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

where $\mathbf{M} = \mathbf{W}^T\mathbf{W} + \sigma^2\mathbf{I}$.

Set ▷ M step

$$\begin{aligned}
\mathbf{W}^{\text{new}} &= \left[\sum_n (\mathbf{x}_n - \bar{\mathbf{x}})\,\mathrm{E}[\mathbf{z}_n]^T\right] \left[\sum_n \mathrm{E}\left[\mathbf{z}_n\mathbf{z}_n^T\right]\right]^{-1} \\
(\sigma^{\text{new}})^2 &= \frac{1}{ND}\sum_n \|\mathbf{x}_n - \bar{\mathbf{x}}\|^2 - 2\,\mathrm{E}[\mathbf{z}_n]^T\left(\mathbf{W}^{\text{new}}\right)^T(\mathbf{x}_n - \bar{\mathbf{x}}) \\
&\quad + \mathrm{Tr}\left(\mathrm{E}\left[\mathbf{z}_n\mathbf{z}_n^T\right]\left(\mathbf{W}^{\text{new}}\right)^T\mathbf{W}^{\text{new}}\right)
\end{aligned} \tag{4.33}$$

until convergence.

Bayesian PCA

4.8. Factor analysis

4.9. 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 $\mathbf{x} \sim p$, $\mathrm{E}_{\mathbf{x}\sim p}[f(\mathbf{x})]$ (we will use the shorthand notation $\mathrm{E}[f]$ 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 $\text{E}[f]$.

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 $\mathrm{E}[f]$, as R increases. We consider the the expectations of the numerator and denominator separately:

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

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

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

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

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

The second part is independent of q so we can ignore it. By Jensen's inequality, we have $\mathrm{E}\left[g(u(\mathbf{x}))\right] \geq g\left(\mathrm{E}\left[u(\mathbf{x})\right]\right)$ 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:

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

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

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

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

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

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

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

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

- $\mathbf{x}_t, \mathrm{x}_t \subset \mathcal{X}^D$ and $\mathbf{y}_t, \mathrm{y}_t \subset \mathcal{Y}^D$ 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

$$\begin{aligned}
q(\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}; \boldsymbol{\theta}) &= q(\mathbf{x}_t \mid \mathbf{x}_{1:t-1}, \mathbf{y}_{1:t}; \boldsymbol{\theta}) q(\mathbf{x}_{1:t-1} \mid \mathbf{y}_{1:t}; \boldsymbol{\theta}) && (5.38)\\
&= q(\mathbf{x}_t \mid \mathbf{x}_{1:t-1}, \mathbf{y}_{1:t}; \boldsymbol{\theta}) q(\mathbf{x}_{1:t-1} \mid \mathbf{y}_{1:t-1}; \boldsymbol{\theta}) && (5.39)\\
&= q(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{y}_t; \boldsymbol{\theta}) && (5.40)
\end{aligned}$$

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

$$\begin{aligned}
p(\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}) &= \frac{p(\mathbf{y}_{1:t} \mid \mathbf{x}_{1:t}) p(\mathbf{x}_{1:t})}{p(\mathbf{y}_{1:t})} && (5.41)\\
&= \frac{p(\mathbf{y}_t \mid \mathbf{x}_{1:t}, \mathbf{y}_{1:t-1}) p(\mathbf{y}_{1:t-1} \mid \mathbf{x}_{1:t}) p(\mathbf{x}_{1:t})}{p(\mathbf{y}_t \mid \mathbf{y}_{1:t-1}) p(\mathbf{y}_{1:t-1})} && (5.42)\\
&= \frac{p(\mathbf{y}_t \mid \mathbf{x}_{1:t}, \mathbf{y}_{1:t-1}) p(\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t-1})}{p(\mathbf{y}_t \mid \mathbf{y}_{1:t-1})} && (5.43)\\
&= \frac{p(\mathbf{y}_t \mid \mathbf{x}_{1:t}, \mathbf{y}_{1:t-1}) p(\mathbf{x}_t \mid \mathbf{x}_{1:t-1}, \mathbf{y}_{1:t-1}) p(\mathbf{x}_{1:t-1} \mid \mathbf{y}_{1:t-1})}{p(\mathbf{y}_t \mid \mathbf{y}_{1:t-1})} && (5.44)\\
&= \frac{p(\mathbf{y}_t \mid \mathbf{x}_t) p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) p(\mathbf{x}_{1:t-1} \mid \mathbf{y}_{1:t-1})}{p(\mathbf{y}_t \mid \mathbf{y}_{1:t-1})} && (5.45)\\
&\propto p(\mathbf{y}_t \mid \mathbf{x}_t) p(\mathbf{x}_t \mid \mathbf{x}_{t-1}) p(\mathbf{x}_{1:t-1} \mid \mathbf{y}_{1:t-1}) && (5.46)
\end{aligned}$$

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

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

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

Normalise weights

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

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

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

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

Sample from proposal

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

Compute weights

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

Normalise weights

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

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:

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

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

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

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

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

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

Compute weights

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

Normalise weights

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

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

$$\begin{aligned} q\left(\mathbf{x}_{1:t}^{(r)} \mid \mathbf{y}_{1:t}\right) &= q\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t\right) &(5.79)\\ &= p\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}\right) &(5.80)\\ &= f\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}\right) &(5.81) \end{aligned}$$

Hence the weight equation in (5.51) becomes

$$\begin{aligned} w_t^{(r)} &\propto w_{t-1}^{(r)} \frac{g\left(\mathbf{y}_t \mid \mathbf{x}_t^{(r)}\right) f\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}\right)}{q\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t\right)} &(5.82)\\ &= w_{t-1}^{(r)}\, g\left(\mathbf{y}_t \mid \mathbf{x}_t^{(r)}\right) &(5.83) \end{aligned}$$

This approach can be **i**nefficient 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

$$\begin{aligned} q\left(\mathbf{x}_{1:t}^{(r)} \mid \mathbf{y}_{1:t}\right) &= q\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t\right) &(5.84)\\ &= p\left(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t\right) &(5.85)\\ &= \frac{p\left(\mathbf{y}_t \mid \mathbf{x}_t, \mathbf{x}_{t-1}^{(r)}\right) p\left(\mathbf{x}_t, \mathbf{x}_{t-1}^{(r)}\right)}{p\left(\mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t\right)} &(5.86) \end{aligned}$$

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

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

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

(to do: improve to be more rigorous)

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

1: Initialisation, $t = 0$:
2: **for** $r = 1, \dots, R$ **do** ▷ Sample.
3: Sample $\tilde{x}_0^{(r)} \sim q_0(\cdot)$.
4: **for** $r = 1, \dots, R$ **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**

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

  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*

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

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

If further, it is aperiodic, *then*

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

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. $\square$

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

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

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

$$\pi(\mathbf{x})\mathcal{T}(\mathbf{x} \to \mathbf{x}') = \pi(\mathbf{x})q(\mathbf{x} \to \mathbf{x}') \min\left(1, \frac{\pi(\mathbf{x}')q(\mathbf{x}' \to \mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x} \to \mathbf{x}')}\right) \tag{5.128}$$

$$= \min\left(\pi(\mathbf{x})q(\mathbf{x} \to \mathbf{x}'), \pi(\mathbf{x}')q(\mathbf{x}' \to \mathbf{x})\right) \tag{5.129}$$

$$= \pi(\mathbf{x}')q(\mathbf{x}' \to \mathbf{x}) \min\left(1, \frac{\pi(\mathbf{x})q(\mathbf{x} \to \mathbf{x}')}{\pi(\mathbf{x}')q(\mathbf{x}' \to \mathbf{x})}\right) \tag{5.130}$$

$$= \pi(\mathbf{x}')\mathcal{T}(\mathbf{x}' \to \mathbf{x}) \tag{5.131}$$

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:

$$p(\mathbf{x})\mathcal{T}_i(\mathbf{x} \to \mathbf{x}') = p(\mathbf{x})p(\mathbf{x}_{-i}, x_i' \mid \mathbf{x}) \tag{5.132}$$

$$= p(\mathbf{x}_{-i}, x_i', \mathbf{x}) \tag{5.133}$$

$$= p(\mathbf{x}, x_i', \mathbf{x}_{-i}) \tag{5.134}$$

$$= p(\mathbf{x}')p(\mathbf{x} \mid x_i', \mathbf{x}_{-i}) \tag{5.135}$$

$$= p(\mathbf{x}')\mathcal{T}_i(\mathbf{x}' \to \mathbf{x}) \tag{5.136}$$

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 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**

Set

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

else

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

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

Run SMC targetting ▷ Initial sweep $s = 0$

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

Sample

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

Let

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

denote the corresponding marginal likelihood estimate.

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

Sample

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

Run SMC targeting

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

Sample

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

Let

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

denote the coresponding marginal likelihood estimate

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

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

if success **then**

Set

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

else

Set

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

5.8.3. Particle Gibbs (PG) sampler

Conditional SMC update

We want to smple 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
$$A_{t-1}^{(r)} \sim \text{Cat}\left(\hat{w}_{t-1}^{(1)}, \ldots, \hat{w}_{t-1}^{(R)}\right)$$

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

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

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

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

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

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

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

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
- $B_{1:T}(s-1)$.

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 process

6.2. Dirichlet process

6.3. Chinese restaurant process

6.4. Hierarchical Dirichlet process

6.5. Hierarchical Dirichlet process

6.6. Indian buffet process

6.7. Dirichlet diffusion trees

6.8. Pitman-Yor process

7. Probabilistic programming

7.1. Testing

7.1.1. Unit and measure tests

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

7.1.2. 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` $\mathcal{D} = \{y_n : y_n \sim g(\ldots, \boldsymbol{\theta})\}$, 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	
$\theta \sim \mathrm{Beta}(\alpha, \beta)$	`[assume theta (beta a b)]`
$x \mid \theta \sim \mathrm{Ber}(\theta)$	`[assume x (flip theta)]`
$\mathcal{D} = \{x_n\}$	`[observe x x1]` $\cdots$ `[observe x xN]`
$\theta \mid \mathcal{D} \sim \mathrm{Beta}(\alpha + N_1, \beta + N_0)$	`[predict theta]`

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

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

Categorical	
$\boldsymbol{\theta} \sim \mathrm{Dir}(\boldsymbol{\alpha}), \boldsymbol{\theta}, \boldsymbol{\alpha} \in \mathbb{R}^K$	`[assume ...]`
$x \mid \boldsymbol{\theta} \sim \mathrm{Dir}(\boldsymbol{\theta})$	`[assume x ...]`
$\mathcal{D} = \{x_n\}$	`[observe x x1]` $\cdots$ `[observe x xN]`
$\boldsymbol{\theta} \mid \mathcal{D} \sim \mathrm{Dir}(\boldsymbol{\alpha} + (n_1, \ldots, n_K)^T)$	`[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)]`
$x \mid \boldsymbol{\theta} \sim \mathcal{N}(\mu, \sigma^2)$	`[assume x (normal mu var)]`
$\mathcal{D} = \{x_n\}$	`[observe x x1]` $\cdots$ `[observe x 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. Weekly meetings for 4yp

8.1. MT14 – Week 1

- Implement measure and conditional measure tests to test the ERPs.
- Research continuous integration (Jenkins, etc.)
- Study Dirichlet processes
- Research stuff
  - Improve RDB by sampling from ?? half of the time instead of sampling from the prior.
  - Sample ERPs (?) in a discretised manner in order to cover more of the sample space.

A. Particle filter animation

Particle filter

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.