

# **Personal notes – Bayesian machine learning**

Tuan Anh Le

October 21, 2014

# Contents

<b>1. Notation</b>	<b>4</b>
<b>2. Basics</b>	<b>5</b>
2.1. Probability distributions . . . . .	5
2.2. Stats . . . . .	5
2.2.1. Kolmogorov-Smirnov test . . . . .	5
2.2.2. Kullback-Leibler divergence . . . . .	5
2.3. Gaussian distribution . . . . .	7
2.3.1. Linear Gaussian model . . . . .	7
<b>3. Bayesian parameter estimation</b>	<b>8</b>
3.1. Beta-Bernoulli model . . . . .	8
3.2. Beta-Binomial model . . . . .	10
3.3. Poisson-Gamma model . . . . .	11
3.4. Dirichlet-Categorical model . . . . .	12
3.5. Dirichlet-Multinomial model . . . . .	13
<b>4. Advanced models</b>	<b>15</b>
4.1. Mixture models . . . . .	15
4.1.1. EM algorithm . . . . .	16
4.1.2. Gaussian mixture model . . . . .	19
4.2. Hidden Markov model . . . . .	21
4.3. Linear regression . . . . .	21
4.4. Logistic regression . . . . .	21
4.5. Latent Dirichlet allocation . . . . .	21
4.6. Linear dynamical systems . . . . .	21
4.7. Principal components analysis . . . . .	21
4.7.1. Classical PCA . . . . .	21
4.7.2. Probabilistic PCA . . . . .	23
4.8. Factor analysis . . . . .	25
4.9. Independent components analysis . . . . .	25
<b>5. Sampling algorithms</b>	<b>26</b>
5.1. Introduction . . . . .	26
5.2. Rejection sampling . . . . .	26
5.2.1. Why it works? . . . . .	26

5.3.	Importance sampling . . . . .	27
5.3.1.	Convergence of estimator as $R$ increases . . . . .	28
5.3.2.	Optimal proposal distribution . . . . .	28
5.4.	Sampling importance resampling . . . . .	29
5.4.1.	Why it works? . . . . .	29
5.5.	Particle filtering . . . . .	30
5.5.1.	Sequential importance sampling (SIS) . . . . .	30
5.5.2.	The degeneracy problem . . . . .	32
5.5.3.	The resampling step . . . . .	33
5.5.4.	The proposal distribution . . . . .	34
5.6.	Sequential Monte Carlo . . . . .	35
5.7.	Markov chain Monte Carlo methods . . . . .	36
5.7.1.	Definitions . . . . .	36
5.7.2.	Metropolis Hastings algorithm . . . . .	39
5.7.3.	Gibbs sampling . . . . .	40
5.8.	Particle Markov Chain Monte Carlo . . . . .	41
5.8.1.	Particle independent Metropolis Hastings (PIMH) sampler . . . . .	41
5.8.2.	Particle marginal Metropolis Hastings (PMMH) sampler . . . . .	42
5.8.3.	Particle Gibbs (PG) sampler . . . . .	43
<b>6.</b>	<b>Nonparametric Bayesian models</b>	<b>45</b>
6.1.	Gaussian process . . . . .	45
6.2.	Dirichlet processes . . . . .	45
6.2.1.	Definitions . . . . .	45
6.2.2.	Posterior measure . . . . .	46
6.2.3.	Stick-breaking construction . . . . .	47
6.2.4.	Pólya urn construction . . . . .	48
6.2.5.	Chinese restaurant process . . . . .	49
6.2.6.	Dirichlet process mixtures . . . . .	51
<b>7.</b>	<b>Probabilistic programming</b>	<b>52</b>
7.1.	Testing . . . . .	52
7.1.1.	Unit and measure tests . . . . .	52
7.1.2.	Conditional measure tests . . . . .	52
<b>8.</b>	<b>Weekly meetings for 4yp</b>	<b>54</b>
8.1.	MT14 – Week 1 . . . . .	54
<b>A.</b>	<b>Particle filter animation</b>	<b>55</b>

# 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).
$\text{diag}(x_1, \dots, x_N)$	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.
$\delta_X$	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 $\mathbb{I}(x = y)$ .

## 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

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

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

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

$$D_N(x) \triangleq \sup_x |F_N(x) - F(x)| \quad (2.2)$$

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

#### 2.2.2. Kullback-Leibler divergence

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

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

Distribution	Parameters	Support	PDF/PMF	Mean	Variance
Bernoulli (Ber)	$\theta \in [0, 1]$	$x \in \{0, 1\}$	$\begin{cases} \theta & \text{if } x = 1 \\ 1 - \theta & \text{if } x = 0 \end{cases}$	$\theta$	$\theta(1 - \theta)$
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)	$\lambda > 0$	$x \in \{0, 1, 2, \dots\}$	$\frac{\lambda^x}{x!} \exp(-\lambda)$	$\lambda$	$\lambda$
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)	$\boldsymbol{\theta} \in [0, 1]^K, \sum_k \theta_k = 1$	$x \in \{1, \dots, K\}$	$\theta_x$	Mean- ingless.	Meaningless.
Dirichlet (Dir)	$\boldsymbol{\alpha} \in (0, \infty)^K$	$\mathbf{x} \in [0, 1]^K, \sum_k x_k = 1$	$\frac{\Gamma(\sum_k \alpha_k)}{\prod_k \Gamma(\alpha_k)} \prod_k x_k^{\alpha_k - 1}$	$\sum_k \alpha_k$	$\text{var}[x_k] = \frac{\alpha_k(\alpha_0 - \alpha_k)}{\alpha_0^2(\alpha_0 + 1)}$
Multinomial (Mult)	$N \in \mathbb{N}, \boldsymbol{\theta} \in [0, 1]^K, \sum_k \theta_k = 1$	$\mathbf{x} \in \{0, \dots, N\}^K, \sum_k x_k = N$	$\frac{N!}{x_1! \dots x_K!} \theta_1^{x_1} \dots \theta_K^{x_K}$	$N\boldsymbol{\theta}$	$\text{var}[x_k] = N\theta_k(1 - \theta_k)$
Dirichlet- Multinomial (DirMult)	$N \in \mathbb{N}, \boldsymbol{\alpha} \in (0, \infty)^K$	$\mathbf{x} \in \{0, \dots, N\}^K, \sum_k x_k = N$	$\frac{\Gamma(N + 1)}{\prod_k \Gamma(\alpha_k + 1)} \prod_k \frac{\Gamma(\alpha_k + x_k)}{\Gamma(\alpha_k)}$		

Table 2.1.: Summary of common probability distributions

<sup>a</sup>

<sup>a</sup>where  $B(\alpha, \beta)$  is the normalisation constant for a Beta distribution,  $\text{Beta}(\alpha, \beta)$ , which is  $\int_x x^{\alpha-1} (1-x)^{\beta-1} dx$  or  $\frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$ .

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

Note that  $\text{KL}(p \parallel q) \neq \text{KL}(q \parallel p)$ .

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

*Proof.* asdf □

## 2.3. Gaussian distribution

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

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

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

### 2.3.1. Linear Gaussian model

Given the marginal and conditional distributions to be

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

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

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

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

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

where

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

**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

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

**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.**

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

**Posterior predictive.**

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



Likelihood	Model parameters	Prior	Hyperparameters	Posterior Hyperparameters	Posterior predictive
Bernoulli	$\theta$	Beta	$\alpha, \beta$	$\alpha + \sum_n \mathbb{I}(x_n = 1), \beta + \sum_n \mathbb{I}(x_n = 0)$	$\text{Ber}(\tilde{x} \mid \frac{\alpha'}{\alpha' + \beta'})$
Binomial	$\theta$	Beta	$\alpha, \beta$	$\alpha + \sum_n x_n, \beta + \sum_n (T_n - x_n)$	$\text{BetaBin}(\tilde{x} \mid \alpha', \beta')$
Poisson	$\lambda$	Gamma	$\alpha, \beta$	$\alpha + \sum_n x_n, \beta + N$	$\text{NB}(\tilde{x} \mid \alpha', \frac{1}{1 + \beta'})$
Categorical	$\boldsymbol{\theta} \in \mathbb{R}^K$	Dirichlet	$\boldsymbol{\alpha} \in \mathbb{R}^K$	$\boldsymbol{\alpha} + (n_1, \dots, n_K)^T$	$\text{Ber}(\tilde{x} \mid \frac{\alpha'_x}{\sum_k \alpha'_k})$
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

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

### 3.2. Beta-Binomial model

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

**Likelihood.**

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

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

**Posterior.**

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

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

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

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

where  $B(\alpha, \beta)$  is the normalisation constant for a Beta distribution,  $\text{Beta}(\alpha, \beta)$ , which is  $\int_0^1 x^{\alpha-1} (1-x)^{\beta-1} dx$  or  $\frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$ .

### 3.3. Poisson-Gamma model

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

**Likelihood.**

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

**Posterior.**

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

**Posterior predictive.**

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

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

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

### 3.4. Dirichlet-Categorical model

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

**Likelihood.**

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

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

**Posterior.**

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

**Posterior predictive.**

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

### 3.5. Dirichlet-Multinomial model

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

**Likelihood.**

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

where  $n_k = \sum_n x_{n,k}, k = 1, \dots, K$  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.**

$$\begin{aligned}
p(\boldsymbol{\theta} \mid \mathcal{D}) &\propto p(\mathcal{D} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta}) \\
&= \text{Mult}(\mathbf{x} \mid T, \boldsymbol{\theta}) \text{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) \\
&\propto (\theta_1^{n_1} \cdots \theta_K^{n_K}) (\theta_1^{\alpha_1-1} \cdots \theta_K^{\alpha_K-1}) \\
&= \theta_1^{n_1+\alpha_1-1} \cdots \theta_K^{n_K+\alpha_K-1} \\
&\propto \text{Dir}(\boldsymbol{\theta} \mid \boldsymbol{\alpha} + \mathbf{x}) \\
&= \text{Dir}\left(\boldsymbol{\theta} \mid \boldsymbol{\alpha} + \sum_n \mathbf{x}_n\right) \tag{3.1}
\end{aligned}$$

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

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

where  $D(\boldsymbol{\alpha})$  is the normalisation constant for the Dirichlet distribution,  $\text{Dir}(\boldsymbol{\alpha})$ , which is  $\int_{\mathbf{x}} \prod_k x_k^{\alpha_k - 1} d\mathbf{x}$  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  $\mathbf{Z} = \{z_n, z_n \in \{1, \dots, K\}\}, n = 1, \dots, N$  and observed states  $\mathbf{X} = \{\mathbf{x}_n, \mathbf{x}_n \in \mathbb{R}^D\}, n = 1, \dots, N$ . We set the priors and the class conditional likelihoods to be  $p(z_n) = \text{Cat}(\boldsymbol{\pi}), \boldsymbol{\pi} = (\pi_1, \dots, \pi_K)$  and  $p(\mathbf{x}_n | z_n = k; \boldsymbol{\theta}) = p_k(\mathbf{x}_n | \boldsymbol{\theta})$ . We can thus express the likelihood of the observed variables to be:

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

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:

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

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

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

*Unidentifiability* refers to the fact that the posterior distribution for the parameter  $p(\boldsymbol{\theta} | \mathcal{D})$  can be multimodal (with equal peaks) and hence can't 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:

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

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

$$\begin{aligned}
\ell_c(\boldsymbol{\theta}) &\triangleq \log p(\{\mathbf{x}_n, z_n\} \mid \boldsymbol{\theta}) \\
&= \log \prod_n p(\mathbf{x}_n, z_n \mid \boldsymbol{\theta}) \\
&= \sum_n \log p(\mathbf{x}_n, z_n \mid \boldsymbol{\theta}) \tag{4.5}
\end{aligned}$$

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*

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

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

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

or

$$\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}) = \mathbb{E}_{\mathbf{Z} \sim \cdot \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}}} [\ln p(\mathbf{X}, \mathbf{Z} \mid \boldsymbol{\theta})] \tag{4.8}$$

$$= \mathbb{E}_{\mathbf{Z} \sim \cdot \mid \mathbf{X}; \boldsymbol{\theta}^{\text{old}}} \left[ \sum_n \ln p(\mathbf{x}_n, z_n \mid \boldsymbol{\theta}) \right] \tag{4.9}$$

$$= \sum_n \mathbb{E}_{z_n \sim \cdot \mid \mathbf{x}_n; \boldsymbol{\theta}^{\text{old}}} [\ln p(\mathbf{x}_n, z_n \mid \boldsymbol{\theta})] \tag{4.10}$$

$$= \sum_n \sum_k p(z_n = k \mid \mathbf{x}_n; \boldsymbol{\theta}^{\text{old}}) \ln p(\mathbf{x}_n, z_n = k \mid \boldsymbol{\theta}) \tag{4.11}$$

$$= \sum_n \sum_k r_{nk}(\boldsymbol{\theta}^{\text{old}}) \ln (\pi_k p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta})) \tag{4.12}$$

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



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

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

where

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

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

because

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

The actual algorithm is as follows

---

**Algorithm 1** EM algorithm for maximising the likelihood

---

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

**E step.** Hold  $\boldsymbol{\theta}^{\text{old}}$ , maximise  $\mathcal{L}(q, \boldsymbol{\theta}^{\text{old}})$  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  $\mathcal{L}(q, \boldsymbol{\theta}^{\text{old}})$  by minimising  $\text{KL}(q \parallel p)$ . 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

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

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

from which we can see that we should maximise  $\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}})$ . 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

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

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

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

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

**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  $\mathcal{L}(q, \boldsymbol{\theta}) + \ln p(\boldsymbol{\theta})$  w.r.t.  $\boldsymbol{\theta}$  to get  $\boldsymbol{\theta}^{\text{new}}$ . We can rewrite  $\mathcal{L}(q, \boldsymbol{\theta}) + \ln p(\boldsymbol{\theta})$  as

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

from which we can see that we should maximise  $\mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}) + \ln p(\boldsymbol{\theta})$ . 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:    $\boldsymbol{\theta}^{\text{old}} \leftarrow \boldsymbol{\theta}^{\text{new}}$
  - 4:   E step: Set  $q(\mathbf{Z}) = p(\mathbf{Z} \mid \mathbf{X}, \boldsymbol{\theta}^{\text{old}})$ .
  - 5:   M step: Hold  $q(\mathbf{Z})$  fixed and set  $\boldsymbol{\theta}^{\text{new}} = \arg \max_{\boldsymbol{\theta}} \{ \mathcal{Q}(\boldsymbol{\theta}, \boldsymbol{\theta}^{\text{old}}) + \ln p(\boldsymbol{\theta}) \}$ .
  - 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

$$p(z_n = k) = \pi_k \quad (4.19)$$

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

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

#### EM algorithm for GMM

---

##### Algorithm 3 EM algorithm for GMM

---

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

$$\begin{aligned} \pi_k^{\text{new}} &= \frac{\sum_n r_{nk}}{N} \\ \boldsymbol{\mu}_k^{\text{new}} &= \frac{\sum_n r_{nk} \mathbf{x}_n}{\sum_n r_{nk}} \\ \boldsymbol{\Sigma}_k^{\text{new}} &= \frac{\sum_n r_{nk} (\mathbf{x}_n - \boldsymbol{\mu}_k)(\mathbf{x}_n - \boldsymbol{\mu}_k)^T}{\sum_n r_{nk}} \end{aligned}$$

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

- 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  $\mathbf{K} = (k_1, \dots, k_N)$ ,  $k_n \in \{1, \dots, K\}$  for  $n = 1, \dots, N$  as

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

Therefore, in the E step, we set

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

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

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

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

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

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

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

For  $\pi_k$ ,

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

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

$$\pi_k = \frac{\sum_n r_{nk}}{N} \quad (4.22)$$

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

For  $\boldsymbol{\mu}_k$ ,

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

Setting this to zero, we get

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

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

For  $\boldsymbol{\Sigma}_k$ ,

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

Setting this to zero, we get

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

## 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  $\{\mathbf{x}_n, \mathbf{x}_n \in \mathbb{R}^D\}, n = 1, \dots, N$ . 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*,  $\{\mathbf{z}_n, \mathbf{z}_n \in \mathbb{R}^M\}, n = 1, \dots, N$ . Let the *principal components* be  $\{\mathbf{u}_m, \mathbf{u}_m \in \mathbb{R}^D, \|\mathbf{u}_m\| = 1\}, m = 1, \dots, M$ . 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  $n = 1, \dots, N$  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

$$\begin{aligned}
V &= \sum_{m=1}^M \text{var}(\text{dimension } m) \\
&= \sum_{m=1}^M \frac{1}{N} \sum_{n=1}^N (z_{nm} - \bar{z}_m)^2 \\
&\quad \left( \text{where } \bar{z}_m = \frac{1}{N} \sum_{n=1}^N z_{nm} \right) \tag{4.25}
\end{aligned}$$

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

$$\left( \text{where } \mathbf{S} = \frac{1}{N} \sum_{n=1}^N (\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})^T \right) \tag{4.27}$$

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:

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

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

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

$$\text{grad}_{\mathbf{u}_m} \mathcal{L} = 2\mathbf{S}\mathbf{u}_m - 2\lambda_m \mathbf{u}_m \quad (4.30)$$

$$= 0 \quad (4.31)$$

$$\implies \mathbf{S}\mathbf{u}_m = \lambda_m \mathbf{u}_m \quad (4.32)$$

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  $\mathbf{Z} = \{\mathbf{z}_n, \mathbf{z}_n \in \mathbb{R}^M\}$ ,  $n = 1, \dots, N$  are the latent variables and  $\mathbf{X} = \{\mathbf{x}_n, \mathbf{x}_n \in \mathbb{R}^D\}$ ,  $n = 1, \dots, N$  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:

$$p(\mathbf{z}) = \mathcal{N}(\mathbf{z}; \mathbf{0}, \mathbf{I})$$

$$p(\mathbf{x} | \mathbf{z}) = \mathcal{N}(\mathbf{x}; \mathbf{W}\mathbf{z} + \boldsymbol{\mu}, \sigma^2 \mathbf{I})$$

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 2.3.1, we can express the remaining marginal and conditional as

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \mathbf{C})$$

$$p(\mathbf{z} | \mathbf{x}) = \mathcal{N}(\mathbf{z}; \mathbf{M}^{-1} \mathbf{W}^T (\mathbf{x} - \boldsymbol{\mu}), \sigma^2 \mathbf{M}^{-1})$$

where

$$\mathbf{C} = \mathbf{W}\mathbf{W}^T + \sigma^2 \mathbf{I}$$

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

#### MLE for probabilistic PCA

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

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

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

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

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

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

$$\begin{aligned} \mathbf{U}_M &= [\mathbf{u}_1, \dots, \mathbf{u}_M] \\ \mathbf{L}_M &= \text{diag}(\lambda_1, \dots, \lambda_M) \end{aligned}$$

where  $\mathbf{u}_1, \dots, \mathbf{u}_D$  and  $\lambda_1, \dots, \lambda_D$  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.** fsdaf 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:    $\boldsymbol{\theta}^{\text{old}} \leftarrow \boldsymbol{\theta}^{\text{new}}$

4:   Set

▷ E step

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

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



5: Set

▷ M step

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

6: **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  $Z_p = \int_{\mathcal{X}} p^*(\mathbf{x}) d\mathbf{x}$ ). We want to achieve the following:

**Problem 1** Generate samples  $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(R)}\}$ ,  $R \in \mathbb{N}$  (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

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

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  $u \sim \text{Unif}(0, cq^*(\mathbf{x}))$ .
  - 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

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

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

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

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

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

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

$$= \frac{1}{cZ_q} \quad (5.7)$$

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

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

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

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

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

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

which means

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

Working backwards

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

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

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

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  $E[f]$ .
-

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

$$E_q[\text{numer}] = E_q \left[ \sum_r w_r f(\mathbf{x}^{(r)}) \right] \quad (5.15)$$

$$= \sum_r E_q \left[ w_r f(\mathbf{x}^{(r)}) \right] \quad (5.16)$$

$$= \sum_r E_q \left[ \frac{p^*(\mathbf{x}^{(r)})}{q^*(\mathbf{x}^{(r)})} f(\mathbf{x}^{(r)}) \right] \quad (5.17)$$

$$= \sum_r E_q \left[ \frac{Z_p p(\mathbf{x}^{(r)})}{Z_q q(\mathbf{x}^{(r)})} f(\mathbf{x}^{(r)}) \right] \quad (5.18)$$

$$= \frac{Z_p}{Z_q} \sum_r \int_{\mathbf{x}^{(r)}} p(\mathbf{x}^{(r)}) f(\mathbf{x}^{(r)}) d\mathbf{x}^{(r)} \quad (5.19)$$

$$= \frac{Z_p}{Z_q} \sum_r E_p \left[ f(\mathbf{x}^{(r)}) \right] \quad (5.20)$$

$$= \frac{Z_p}{Z_q} R E_p [f(\mathbf{x})] \quad (5.21)$$

$$E_q[\text{denom}] = E_q \left[ \sum_r w_r \right] \quad (5.22)$$

$$= \sum_r E_q \left[ \frac{p^*(\mathbf{x}^{(r)})}{q^*(\mathbf{x}^{(r)})} \right] \quad (5.23)$$

$$= \sum_r E_q \left[ \frac{Z_p p(\mathbf{x}^{(r)})}{Z_q q(\mathbf{x}^{(r)})} \right] \quad (5.24)$$

$$= \frac{Z_p}{Z_q} \sum_r \int_{\mathbf{x}^{(r)}} p(\mathbf{x}^{(r)}) d\mathbf{x}^{(r)} \quad (5.25)$$

$$= \frac{Z_p}{Z_q} R \quad (5.26)$$

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

### 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

$$\text{var}_q \left[ \frac{p(\mathbf{x})}{q(\mathbf{x})} f(\mathbf{x}) \right] = E_q \left[ \frac{p^2(\mathbf{x})}{q^2(\mathbf{x})} f^2(\mathbf{x}) \right] - \left( E_q \left[ \frac{p(\mathbf{x})}{q(\mathbf{x})} f(\mathbf{x}) \right] \right)^2 \quad (5.27)$$

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

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

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

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

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

## 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  $\{\mathbf{x}^{(r)}\}$  from  $q$ .
- 2: Calculate importance weights  $\{w_r = \frac{p^*(\mathbf{z}^{(r)})}{q^*(\mathbf{z}^{(r)})}\}$ .
- 3: Calculate the normalised importance weights  $\{\hat{w}_r = \frac{w_r}{\sum_{r'} w_{r'}}\}$ . Note that  $\sum_r \hat{w}_r = 1$ .
- 4: We can resample from

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

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)  $R \rightarrow \infty$ . We can express the number of proposal samples that are in the interval  $\lim_{\delta x \rightarrow 0} [x, x + \delta x]$ ,  $N(x)$ , to be

$$N(x) = \lim_{\delta x \rightarrow 0} Rq(x)\delta x \quad (5.32)$$

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

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

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

$$\propto \lim_{\delta x \rightarrow 0} p(x) \delta x \quad (5.35)$$

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

$$\Pr(a \leq x^{(r)} \leq b) \propto \int_a^b p(x) dx \quad (5.36)$$

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

## 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  $t, t = 1, \dots, T$ .
- The initial state is characterised by  $\mathbf{x}_1 \sim \mu(\cdot | \boldsymbol{\theta})$  for some known parameter  $\boldsymbol{\theta} \in \Theta$ .
- The transitions are characterised by  $\mathbf{x}_t | \mathbf{x}_{t-1} \sim f(\cdot | \mathbf{x}_{t-1}; \boldsymbol{\theta})$ .
- The emissions are characterised by  $\mathbf{y}_t | \mathbf{x}_t \sim g(\cdot | \mathbf{x}_t; \boldsymbol{\theta})$ .

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

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

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

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

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

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

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

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

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

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

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

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

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

$$w_t^{(r)} \propto \frac{p(\mathbf{x}_{1:t}^{(r)} | \mathbf{y}_{1:t})}{q(\mathbf{x}_{1:t}^{(r)} | \mathbf{y}_{1:t})} \quad (5.48)$$

$$\propto \frac{p(\mathbf{y}_t | \mathbf{x}_t^{(r)}) p(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}) p(\mathbf{x}_{1:t-1}^{(r)} | \mathbf{y}_{1:t-1})}{q(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t) q(\mathbf{x}_{1:t-1}^{(r)} | \mathbf{y}_{1:t-1})} \quad (5.49)$$

$$= w_{t-1}^{(r)} \frac{p(\mathbf{y}_t | \mathbf{x}_t^{(r)}) p(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)})}{q(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t)} \quad (5.50)$$

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

For  $t = 1$

$$w_1^{(r)} \propto \frac{p(\mathbf{x}_1^{(r)} | \mathbf{y}_1)}{q(\mathbf{x}_1^{(r)} | \mathbf{y}_1)} \quad (5.52)$$

$$\propto \frac{p(\mathbf{x}_1^{(r)}, \mathbf{y}_1)}{q(\mathbf{x}_1^{(r)} | \mathbf{y}_1)} \quad (5.53)$$

$$\propto \frac{p(\mathbf{y}_1 | \mathbf{x}_1^{(r)}) p(\mathbf{x}_1^{(r)})}{q(\mathbf{x}_1^{(r)} | \mathbf{y}_1)} \quad (5.54)$$

$$= \frac{g(\mathbf{y}_1 | \mathbf{x}_1^{(r)}) \mu(\mathbf{x}_1^{(r)})}{q(\mathbf{x}_1^{(r)} | \mathbf{y}_1)} \quad (5.55)$$

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

$$\mathbf{x}_1^{(r)} \sim q(\cdot | \mathbf{y}_1^{(r)}; \boldsymbol{\theta}), r = 1, \dots, R \quad (5.56)$$

2: Compute weights

$$w_1^{(r)} \propto \frac{g(\mathbf{y}_1 | \mathbf{x}_1^{(r)}) \mu(\mathbf{x}_1^{(r)})}{q(\mathbf{x}_1^{(r)} | \mathbf{y}_1)}, r = 1, \dots, R \quad (5.57)$$

3: Normalise weights

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

4: We can resample from

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

to estimate

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

5: **for**  $t = 2, \dots, T$  **do**

▷ Main loop

6:     Sample from proposal

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

7:     Compute weights

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

8:     Normalise weights

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

9:     We can resample from

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

to estimate

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

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:

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



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

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

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

2: Compute weights

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

3: Normalise weights

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

4: We can resample from

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

to estimate

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

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

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

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

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

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

8: Compute weights

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

9: Normalise weights

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

10: We can resample from

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

to estimate

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

#### 5.5.4. The proposal distribution

It is common to use the following proposal distribution

$$q(\mathbf{x}_{1:t}^{(r)} | \mathbf{y}_{1:t}) = q(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t) \quad (5.79)$$

$$= p(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}) \quad (5.80)$$

$$= f(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}) \quad (5.81)$$

Hence the weight equation in (5.51) becomes

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

$$= w_{t-1}^{(r)} g(\mathbf{y}_t | \mathbf{x}_t^{(r)}) \quad (5.83)$$

This approach can be inefficient because the likelihood,  $p(\mathbf{y}_t | \mathbf{x}_t^{(r)})$ , can be very small at many places meaning many of the particles will be very small.

The optimal proposal distribution has the form

$$q(\mathbf{x}_{1:t}^{(r)} | \mathbf{y}_{1:t}) = q(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t) \quad (5.84)$$

$$= p(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t) \quad (5.85)$$

$$= \frac{p(\mathbf{y}_t | \mathbf{x}_t, \mathbf{x}_{t-1}^{(r)}) p(\mathbf{x}_t, \mathbf{x}_{t-1}^{(r)})}{p(\mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t)} \quad (5.86)$$

$$= \frac{p(\mathbf{y}_t | \mathbf{x}_t) p(\mathbf{x}_t | \mathbf{x}_{t-1}^{(r)})}{p(\mathbf{y}_t | \mathbf{x}_{t-1}^{(r)})} \quad (5.87)$$

$$= \frac{g(\mathbf{y}_t | \mathbf{x}_t) f(\mathbf{x}_t | \mathbf{x}_{t-1}^{(r)})}{p(\mathbf{y}_t | \mathbf{x}_{t-1}^{(r)})} \quad (5.88)$$

The weight equation in (5.51) becomes

$$w_t^{(r)} \propto w_{t-1}^{(r)} p(\mathbf{y}_t | \mathbf{x}_{t-1}^{(r)}) \quad (5.89)$$

$$= w_{t-1}^{(r)} \int p(\mathbf{y}_t, \mathbf{x}'_t | \mathbf{x}_{t-1}^{(r)}) d\mathbf{x}' \quad (5.90)$$

$$= w_{t-1}^{(r)} \int p(\mathbf{y}_t | \mathbf{x}'_t, \mathbf{x}_{t-1}^{(r)}) p(\mathbf{x}'_t | \mathbf{x}_{t-1}^{(r)}) d\mathbf{x}' \quad (5.91)$$

$$= w_{t-1}^{(r)} \int p(\mathbf{y}_t | \mathbf{x}'_t) p(\mathbf{x}'_t | \mathbf{x}_{t-1}^{(r)}) d\mathbf{x}' \quad (5.92)$$

$$= w_{t-1}^{(r)} \int g(\mathbf{y}_t | \mathbf{x}'_t) f(\mathbf{x}'_t | \mathbf{x}_{t-1}^{(r)}) d\mathbf{x}' \quad (5.93)$$

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  $M_t$ :

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

Also assume that

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

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**
- 5:   Calculate normalised weights  $\hat{w}_0^{(r)} \propto \frac{p_0(\tilde{x}_0^{(r)})}{q_0(\tilde{x}_0^{(r)})}$ , such that  $\sum_r \hat{w}_0^{(r)} = 1$ .

---

```

6: Resample from the pmf  $\sum_r \hat{w}_0^{(r)} \delta_{\tilde{x}_0^{(r)}}(\cdot)$  to get  $R$  samples  $\{x_0^{(r)}\}$ . ▷ Resample.
7:
8: Iterate,  $t = 1, \dots, T$ :
9: for  $t = 1, \dots, T$  do
10:   for  $r = 1, \dots, R$  do ▷ Sample.
11:     Set  $\tilde{x}_{0:t-1}^{(r)} = x_{0:t-1}^{(r)}$ .
12:     Sample  $\tilde{x}_t^{(r)} \sim M_t(\tilde{x}_{0:t-1}^{(r)}, \cdot)$ .
13:   for  $r = 1, \dots, R$  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  $\{x_t^{(r)}\}$ . 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  $\mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}')$  of going from  $\mathbf{x}$  to  $\mathbf{x}'$ , i.e.  $\mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') = \Pr(\mathbf{x}' \mid \mathbf{x})$ . This transition probability applies whenever the chain is in state  $\mathbf{x}$ .

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

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

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

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

If further, it is aperiodic, then

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

A MC following these conditions is ergodic

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

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

$$\pi(\mathbf{X} = \mathbf{x}') = \sum_{\mathbf{x} \in \mathcal{X}} \pi(\mathbf{X} = \mathbf{x}) \mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') \quad \forall \mathbf{x}' \quad (5.98)$$

**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  $\forall a \in \mathcal{X}$ ,

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

**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  $\pi$  such that, for all  $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$

$$\pi(\mathbf{x})\mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') = \pi(\mathbf{x}')\mathcal{T}(\mathbf{x}' \rightarrow \mathbf{x}). \quad (5.100)$$

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  $\pi$ , then  $\pi$  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  $\pi$  is a stationary distribution of  $\mathcal{T}$ . We have

$$\sum_{\mathbf{x} \in \mathcal{X}} \pi(\mathbf{x})\mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') = \sum_{\mathbf{x} \in \mathcal{X}} \pi(\mathbf{x}')\mathcal{T}(\mathbf{x}' \rightarrow \mathbf{x}) \quad (5.101)$$

$$= \sum_{\mathbf{x} \in \mathcal{X}} \pi(\mathbf{x}') \Pr(\mathbf{x} \mid \mathbf{x}') \quad (5.102)$$

$$= \pi(\mathbf{x}') \sum_{\mathbf{x} \in \mathcal{X}} \Pr(\mathbf{x} \mid \mathbf{x}') \quad (5.103)$$

$$= \pi(\mathbf{x}') \quad (5.104)$$

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

**Proposition 5.7.2.** Let  $\mathcal{T}_1, \dots, \mathcal{T}_K$  be a set of kernels each of which satisfies detailed balance w.r.t.  $\pi$ . Let  $p_1, \dots, p_K$  be any distribution over  $\{1, \dots, K\}$ . 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  $\pi$ .

*Proof.* The aggregate kernel can be written as

$$\mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') = \Pr(\mathbf{x}' \mid \mathbf{x}) \quad (5.105)$$

$$= \sum_k \Pr(\mathbf{x}', k \mid \mathbf{x}) \quad (5.106)$$

$$= \sum_k \Pr(\mathbf{x}' \mid k, \mathbf{x}) \Pr(k \mid \mathbf{x}) \quad (5.107)$$

$$= \sum_k \mathcal{T}_k(\mathbf{x} \rightarrow \mathbf{x}') p_k \quad (5.108)$$

Using this, we can prove the detailed balance as follows

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

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

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

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

$$= \pi(\mathbf{x}') \mathcal{T}(\mathbf{x}' \rightarrow \mathbf{x}) \quad (5.113)$$

□

**Proposition 5.7.3.** *Let  $\mathcal{T}_1, \dots, \mathcal{T}_K$  be a set of kernels each of which satisfies detailed balance w.r.t.  $\pi$ . 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  $\pi$  as its stationary distribution.*

*Proof.* The aggregate kernel can be written as

$$\mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') = \Pr(\mathbf{x}' \mid \mathbf{x}) \quad (5.114)$$

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

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

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

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

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

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

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

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

...

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

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

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

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

□

### 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  $q(\cdot \mid \mathbf{x}) \equiv q(\mathbf{x} \rightarrow \cdot)$ . Let  $p \equiv \pi$  be the required distribution (stationary distribution for this MCMC). Assume we can only evaluate  $q$  and  $\pi$  up to a multiplicative factor (i.e. we can only evaluate  $q^*(\mathbf{x} \rightarrow \mathbf{x}') = Z_q q(\mathbf{x} \rightarrow \mathbf{x}')$  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**  $t = 1, \dots, T$  **do**
- 3:   **repeat**
- 4:     Sample  $\mathbf{x}^{(t)} \sim q(\mathbf{x}^{(t-1)} \rightarrow \cdot)$ .
- 5:     Accept  $\mathbf{x}^{(t)}$  with the acceptance probability

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

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

#### Why it works?

We need to prove that  $\pi$  is the unique stationary distribution of this MCMC.

We can express the aggregate transition model to be

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

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

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

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

$$= \min (\pi(\mathbf{x})q(\mathbf{x} \rightarrow \mathbf{x}'), \pi(\mathbf{x}')q(\mathbf{x}' \rightarrow \mathbf{x})) \quad (5.129)$$

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

$$= \pi(\mathbf{x}')\mathcal{T}(\mathbf{x}' \rightarrow \mathbf{x}) \quad (5.131)$$

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

Hence  $\pi$  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  $\pi$  is the unique stationary distribution of this MCMC.

### 5.7.3. Gibbs sampling

Assume we want to sample from  $p(\mathbf{x}) = p(x_1, \dots, x_D)$ . We can only sample from the conditionals  $p(x_i \mid \mathbf{x}_{-i})$  where  $\mathbf{x}_{-i}$  denotes  $\mathbf{x}$  with the  $i^{\text{th}}$  component omitted. 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**  $t = 1, \dots, T$  **do**
  - 3:   Sample  $x_1^{(t)} \sim p(\cdot \mid x_2^{(t-1)}, x_3^{(t-1)}, \dots, x_D^{(t-1)})$
  - 4:   Sample  $x_2^{(t)} \sim p(\cdot \mid x_1^{(t)}, x_3^{(t-1)}, \dots, x_D^{(t-1)})$
  - 5:    $\vdots$
  - 6:   Sample  $x_D^{(t)} \sim p(\cdot \mid x_1^{(t)}, x_2^{(t)}, \dots, x_{D-1}^{(t)})$
- 

#### 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  $p$ :

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

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

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

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

$$= p(\mathbf{x}')\mathcal{T}_i(\mathbf{x}' \rightarrow \mathbf{x}) \quad (5.136)$$

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  $q_i(\mathbf{x} \rightarrow \mathbf{x}')$  is set to be  $p(\mathbf{x}' | \mathbf{x}) = p(x'_i | \mathbf{x})$  the acceptance probability is one (shown below) and so it is equivalent to one sampling step in Gibbs sampling.

$$\mathcal{A}(\mathbf{x} \rightarrow \mathbf{x}') = \min \left( 1, \frac{p(\mathbf{x}')p(\mathbf{x} | \mathbf{x}')}{p(\mathbf{x})p(\mathbf{x}' | \mathbf{x})} \right) \quad (5.137)$$

$$= \min \left( 1, \frac{p(\mathbf{x}', \mathbf{x})}{p(\mathbf{x}', \mathbf{x})} \right) \quad (5.138)$$

$$= 1 \quad (5.139)$$

## 5.8. Particle Markov Chain Monte Carlo

### 5.8.1. Particle independent Metropolis Hastings (PIMH) sampler

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

---

**Algorithm 13** Particle independent Metropolis Hastings sampler

---

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

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

2: Sample

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

3: Let

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

denote the corresponding marginal likelihood estimate.

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

5:     Run SMC targeting

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

6:     Sample

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

7:     Let

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

denote the corresponding marginal likelihood estimate

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

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

9:     **if** success **then**

10:         Set

$$\mathbf{x}_{1:T}(s) = \mathbf{x}_{1:T}^*$$

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

11:     **else**  
 12:         Set

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

### 5.8.2. 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**  $s = 1, \dots, S$  **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 corresponding marginal likelihood estimate

- 10:     Sample from  $\text{Ber}(\cdot)$  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

$$\boldsymbol{\theta}(s) = \boldsymbol{\theta}^*$$

$$\mathbf{x}_{1:T}(s) = \mathbf{x}_{1:T}^*$$

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

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

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

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

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

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

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

- 4: Compute weights

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

- 5: Normalise weights

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

- 6: We can resample from

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

to estimate

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

- 7: **for**  $t = 2, \dots, T$  **do** ▷ Main loop  
 8:     For  $r \neq B_t$ , sample

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

- 9:     For  $r \neq B_t$ , sample

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

10: Compute weights

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

11: Normalise weights

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

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 $\boldsymbol{\theta}(0)$ , $\mathbf{x}_{1:T}(0)$ , $B_{1:T}(0)$ arbitrarily. | ▷ Initialisation, $s = 0$ |
| 2: <b>for</b> Sweep $s = 1, \dots, S$ <b>do</b>                                     | ▷ Main loop               |
| 3:     Sample parameter   |                           |

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

- 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 process

## 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  $\emptyset$ , 1 for the entire space,  $\Omega$ , and
- countable additivity:  $\forall$  countable collections  $\{E_i\}$  of pairwise disjoint sets of  $\Omega$ ,

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

**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  $X_t : \Omega \rightarrow S$  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  $\Theta$ , and  $\alpha$  a positive scalar. Consider a finite partition  $(T_1, \dots, T_K)$  of  $\Theta$ .*

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

$$(G(T_1), \dots, G(T_K)) \sim \text{Dir}(\alpha H(T_1), \dots, \alpha H(T_K)) \quad (6.1)$$

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

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

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

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

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

□

### 6.2.2. Posterior measure

**Proposition 6.2.1** (Posterior measure). *Let  $G \sim \text{DP}(\alpha, H)$  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:*

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

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

$$(G(T_1), \dots, G(T_K)) \sim \text{Dir}(\alpha H(T_1), \dots, \alpha H(T_K))$$

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

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

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

From conjugacy in (3.1)

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

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

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

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

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

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

We can write the final posterior as

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

□

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

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

For finite  $\alpha$

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

where  $\{\bar{x}_k\}_{k=1}^{\infty}$  are the unique values of  $\{x_n\}_{n=1}^{\infty}$  and  $\pi_k = \lim_{N \rightarrow \infty} \frac{\sum_n \mathbb{I}(x_n = \bar{x}_k)}{N}$  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  $\alpha > 0$ :*

$$\beta_k \sim \text{Beta}(1, \alpha) \quad (6.6)$$

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

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

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

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

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

*Proof.* TODO □

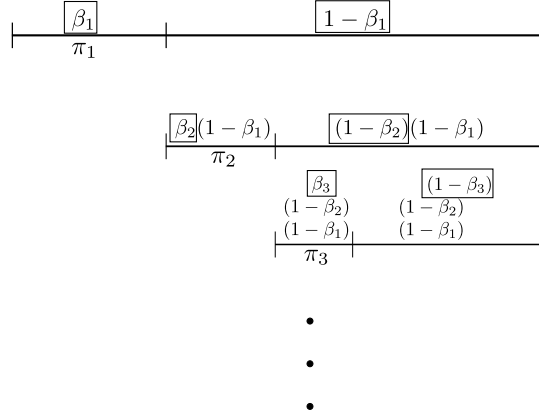


Figure 6.1.: Stick-breaking construction

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

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

#### 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  $\mathcal{D} = \{x_n : x_n \sim G, G \sim \text{DP}(\alpha, H)\}$ .

**Theorem 6.2.3.** *Let  $G \sim \text{DP}(\alpha, H)$ . 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*

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



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  $G \sim \text{DP}(\alpha, H)$  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:

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

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).
- 

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

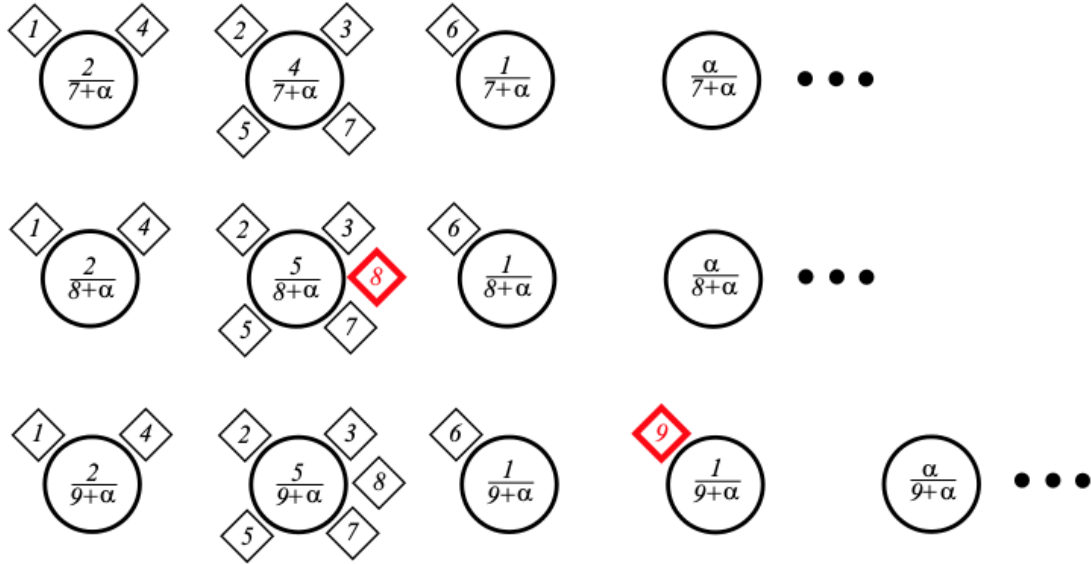


Figure 6.2.: (Figure from Erik Sudderth's PhD) Chinese restaurant process interpretation of the partitions induced by the Dirichlet process  $DP(\alpha, H)$ . Tables (circles) are analogous to clusters, and customers (diamonds) to a series of observations. *Top row:* A starting configuration, in which 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.

### 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 be influenced by observations  $x_j \neq x_i$ , regardless of their proximity.

The Dirichlet process mixtures model is as follows:

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

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

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

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

### 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`  $\theta \sim f$ , then `observe`  $\mathcal{D} = \{y_n : y_n \sim g(\dots, \theta)\}_{n=1}^N$ , and finally `predict`  $\theta \mid \mathcal{D}$ , 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  $\theta \mid \mathcal{D}$ . 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	
$\theta \sim \text{Beta}(\alpha, \beta)$	<code>[assume theta (beta a b)]</code>
$x \mid \theta \sim \text{Ber}(\theta)$	<code>[observe (flip theta) x1] ...</code>
$\mathcal{D} = \{x_n\}$	<code>[observe (flip theta) xN]</code>
$\theta \mid \mathcal{D} \sim \text{Beta}(\alpha + N_1, \beta + N_0)$	<code>[predict theta]</code>
Binomial	
$\theta \sim \text{Beta}(\alpha, \beta)$	<code>[assume theta (beta a b)]</code>
$x \mid \theta \sim \text{Bin}(T, \theta)$	<code>[observe (binomial theta T) x1] ...</code>
$\mathcal{D} = \{x_n\}$	<code>[observe (binomial theta T) xN]</code>
$\theta \mid \mathcal{D} \sim \text{Beta}(\alpha + \sum_n x_n, \beta + TN - \sum_n x_n)$	<code>[predict theta]</code>

---

Poisson	
$\lambda \sim \text{Gamma}(\alpha, \beta)$	[assume 1 (gamma a b)]
$x \mid \theta \sim \text{Poi}(\lambda)$	[observe (poisson 1) x1] ...
$\mathcal{D} = \{x_n\}$	[observe (poisson 1) xN]
$\lambda \mid \mathcal{D} \sim \text{Gamma}(\alpha + \sum_n x_n, \beta + N)$	[predict 1]

---



---

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 ...] ...
$\mathcal{D} = \{x_n\}$	[observe ...]
$\boldsymbol{\theta} \mid \mathcal{D} \sim \text{Dir}(\boldsymbol{\alpha} + (n_1, \dots, n_K)^T)$	[predict ...]

---



---

Univariate Normal with known variance	
Fix $\sigma^2$	[assume var #var#]
$\mu \sim \mathcal{N}(\mu_0, \sigma_0^2)$	[assume mu (normal mu0 var0)]
$x \mid \boldsymbol{\theta} \sim \mathcal{N}(\mu, \sigma^2)$	[observe (normal mu var) x1] ...
$\mathcal{D} = \{x_n\}$	[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. 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**



# Bibliography

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