

# **Personal notes – Bayesian machine learning**

Tuan Anh Le

September 10, 2014

# Contents

<b>1</b>	<b>Probability distributions</b>	<b>4</b>
1.1	Uniform distribution . . . . .	4
1.2	Beta distribution . . . . .	4
1.3	Bernoulli distribution . . . . .	4
1.4	Binomial distribution . . . . .	4
1.5	Beta-binomial distribution . . . . .	4
1.6	Categorical distribution . . . . .	4
1.7	Dirichlet distribution . . . . .	4
1.8	Multinomial distribution . . . . .	4
1.9	Pareto distribution . . . . .	4
<b>2</b>	<b>Bayesian parameter estimation</b>	<b>5</b>
2.1	Beta-Bernoulli model . . . . .	5
2.1.1	Summary . . . . .	5
2.1.2	Derivations . . . . .	5
2.2	Beta-binomial model . . . . .	5
2.2.1	Summary . . . . .	5
2.2.2	Derivations . . . . .	6
2.3	Dirichlet-categorical model . . . . .	6
2.3.1	Summary . . . . .	6
2.3.2	Derivations . . . . .	7
2.4	Dirichlet-multinomial model . . . . .	7
2.4.1	Summary . . . . .	7
2.4.2	Derivations . . . . .	8
2.5	Poisson-gamma model . . . . .	8
2.5.1	Summary . . . . .	8
2.5.2	Derivations . . . . .	8
<b>3</b>	<b>Generative models</b>	<b>9</b>
<b>4</b>	<b>Sampling algorithms</b>	<b>10</b>
4.1	Introduction . . . . .	10
4.2	Rejection sampling . . . . .	10
4.2.1	Why it works? . . . . .	10
4.3	Importance sampling . . . . .	11
4.3.1	Convergence of estimator as $R$ increases . . . . .	12
4.3.2	Optimal proposal distribution . . . . .	12

4.4	Sampling importance resampling . . . . .	13
4.4.1	Why it works? . . . . .	13
4.5	Particle filtering . . . . .	14
4.5.1	Sequential importance sampling (SIS) . . . . .	14
4.5.2	The degeneracy problem . . . . .	16
4.5.3	The resampling step . . . . .	17
4.5.4	Particle filter animation . . . . .	19
4.5.5	The proposal distribution . . . . .	20
4.6	Sequential Monte Carlo . . . . .	21
4.7	Markov chain Monte Carlo methods . . . . .	22
4.7.1	Definitions . . . . .	22
4.7.2	Metropolis Hastings algorithm . . . . .	25
4.7.3	Gibbs sampling . . . . .	26
4.8	Particle Markov Chain Monte Carlo . . . . .	26
4.8.1	Particle independent Metropolis Hastings (PIMH) sampler . . . . .	26
4.8.2	Particle marginal Metropolis Hastings (PMMH) sampler . . . . .	27
4.8.3	Particle Gibbs (PG) sampler . . . . .	29
<b>5</b>	<b>Nonparametric Bayesian models</b>	<b>31</b>

# **1 Probability distributions**

**1.1 Uniform distribution**

**1.2 Beta distribution**

**1.3 Bernoulli distribution**

**1.4 Binomial distribution**

**1.5 Beta-binomial distribution**

**1.6 Categorical distribution**

**1.7 Dirichlet distribution**

**1.8 Multinomial distribution**

**1.9 Pareto distribution**

## 2 Bayesian parameter estimation

### 2.1 Beta-Bernoulli model

#### 2.1.1 Summary

The model

$$X_i \sim \text{Ber}(\theta), \text{ for } i \in \{1, \dots, N\} \quad (2.1)$$

$$\mathcal{D} = \{x_1, \dots, x_N\} \quad (2.2)$$

$$N_1 = \sum_{i=1}^N \mathbb{I}(x_i = 1) \quad (2.3)$$

$$N_0 = \sum_{i=1}^N \mathbb{I}(x_i = 0) \quad (2.4)$$

Likelihood

$$p(\mathcal{D}|\theta) = \theta^{N_1} (1 - \theta)^{N_0} \quad (2.5)$$

Prior

$$p(\theta) = \text{Beta}(\theta|a, b) \quad (2.6)$$

Posterior

$$p(\theta|\mathcal{D}) = \text{Beta}(\theta|a' = N_1 + a, b' = N_0 + b) \quad (2.7)$$

Posterior predictive

$$p(\tilde{x} = 1|\mathcal{D}) = \frac{a'}{a' + b'} \quad (2.8)$$

Evidence

#### 2.1.2 Derivations

### 2.2 Beta-binomial model

#### 2.2.1 Summary

The model

$$N_1 \sim \text{Bin}(N, \theta) \quad (2.9)$$

$$\mathcal{D} = \{N_1, N\} \quad (2.10)$$

$$N_1 = \text{number of successes} \quad (2.11)$$

$$N = \text{total number of trials} \quad (2.12)$$

$$\tilde{\mathcal{D}} = \{\tilde{N}_1, \tilde{N}\} \quad (2.13)$$

$$\tilde{N}_1 = \text{number of successes in a new batch of data} \quad (2.14)$$

$$\tilde{N} = \text{total number of trials in a new batch of data} \quad (2.15)$$

#### Likelihood

$$p(\mathcal{D}|\theta) = \text{Bin}(N_1|N, \theta) \quad (2.16)$$

#### Prior

$$p(\theta) = \text{Beta}(\theta|a, b) \quad (2.17)$$

#### Posterior

$$p(\theta|\mathcal{D}) = \text{Beta}(\theta|a' = N_1 + a, b' = N_0 + b) \quad (2.18)$$

#### Posterior predictive

$$p(\tilde{\mathcal{D}}|\mathcal{D}) = \text{Bb}(\tilde{N}_1; a', b', \tilde{N}) \quad (2.19)$$

#### Evidence

### 2.2.2 Derivations

## 2.3 Dirichlet-categorical model

### 2.3.1 Summary

#### The model

$$X_i \sim \text{Cat}(\boldsymbol{\theta} = (\theta_1, \dots, \theta_K)^T), \text{ for } i \in \{1, \dots, N\} \quad (2.20)$$

$$\mathcal{D} = \{x_1, \dots, x_N\} \quad (2.21)$$

$$n_k = \sum_{i=1}^N \mathbb{I}(x_i = k) \quad (2.22)$$

#### Likelihood

$$p(\mathcal{D}|\theta) = \prod_{k=1}^K \theta_k^{n_k} \quad (2.23)$$

#### Prior

$$p(\theta) = \text{Dir}(\boldsymbol{\theta}; \boldsymbol{\alpha}) \quad (2.24)$$

### Posterior

$$p(\theta|\mathcal{D}) = \text{Dir}(\boldsymbol{\theta}; \boldsymbol{\alpha}' = \boldsymbol{\alpha} + (n_1, \dots, n_K)^T) \quad (2.25)$$

### Posterior predictive

$$p(\tilde{X} = j|\mathcal{D}) = \frac{\alpha'_j}{\sum_{k=1}^K \alpha'_k} \quad (2.26)$$

$$= \frac{\alpha_j + n_j}{\alpha_0 + N} \quad (2.27)$$

$$\text{where } \alpha_0 = \sum_{k=1}^K \alpha_k \quad (2.28)$$

### Evidence

#### 2.3.2 Derivations

## 2.4 Dirichlet-multinomial model

### 2.4.1 Summary

#### The model

$$\mathbf{N} \sim \text{Mult}(N, \boldsymbol{\theta}) \in \mathbb{R}^K \quad (2.29)$$

$$\mathcal{D} = \{\mathbf{n} = \text{vector of counts of successes}\} \quad (2.30)$$

$$N = \sum_{i=1}^K n_i \quad (2.31)$$

$$\tilde{\mathcal{D}} = \{\tilde{\mathbf{n}} = \text{vector of counts of successes in a new batch of data}\} \quad (2.32)$$

$$\tilde{N} = \sum_{i=1}^K \tilde{n}_i \quad (2.33)$$

#### Likelihood

$$p(\mathcal{D}|\theta) = \text{Mult}(\mathbf{n}; N, \boldsymbol{\theta}) \quad (2.34)$$

#### Prior

$$p(\theta) = \text{Dir}(\boldsymbol{\theta}; \boldsymbol{\alpha}) \quad (2.35)$$

#### Posterior

$$p(\theta|\mathcal{D}) = \text{Dir}(\boldsymbol{\theta}; \boldsymbol{\alpha}' = \boldsymbol{\alpha} + (n_1, \dots, n_K)^T) \quad (2.36)$$

### Posterior predictive

$$p(\tilde{\mathcal{D}}|\mathcal{D}) = \frac{\Gamma(\alpha_0 + N)}{\Gamma(\alpha_0 + N + \tilde{N})} \prod_{k=1}^K \frac{\Gamma(\alpha_k + n_k + \tilde{n}_k)}{\Gamma(\alpha_k + n_k)} \quad (2.37)$$

$$\text{where } \alpha_0 = \sum_{k=1}^K \alpha_k \quad (2.38)$$

### Evidence

$$p(\mathcal{D}|\boldsymbol{\alpha}) = \frac{\Gamma(\alpha_0)}{\Gamma(\alpha_0 + N)} \prod_{k=1}^K \frac{\Gamma(\alpha_k + n_k)}{\Gamma(\alpha_k)} \quad (2.39)$$

### 2.4.2 Derivations

## 2.5 Poisson-gamma model

### 2.5.1 Summary

#### The model

$$x \sim \text{Poi}(\lambda) \quad (2.40)$$

$$\mathcal{D} = \{x_1, \dots, x_N\} \quad (2.41)$$

#### Likelihood

$$p(\mathcal{D}|\lambda) = \prod_{i=1}^N \frac{\lambda^{x_i}}{x_i!} \exp(-\lambda) \quad (2.42)$$

#### Prior

$$p(\lambda) = \text{Gamma}(\lambda; a, b) \quad (2.43)$$

#### Posterior

$$p(\lambda|\mathcal{D}) = \text{Gamma}\left(\lambda; a' = a + \sum_{i=1}^N x_i, b' = b + N\right) \quad (2.44)$$

#### Posterior predictive

$$p(\tilde{x}|\mathcal{D}) = \text{NB}\left(\tilde{x}|a', \frac{1}{1+b'}\right) \quad (2.45)$$

#### Evidence

$$p(\mathcal{D}) = \quad (2.46)$$

### 2.5.2 Derivations



## **3 Generative models**

## 4 Sampling algorithms

### 4.1 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).

### 4.2 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

$$cq^*(\mathbf{x}) > p^*(\mathbf{x}) \text{ for all } \mathbf{x} \quad (4.1)$$

The procedure that generates a sample  $\mathbf{x} \sim p$  is described in Algorithm 1 below.

---

**Algorithm 1** 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.
- 

#### 4.2.1 Why it works?

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

$$\mathcal{X} = \left\{ \boldsymbol{\alpha} \in \mathbb{R}^{d+1} : \alpha_{1:d} \in \mathbb{R}^d, \alpha_{d+1} \in [0, cq^*(\boldsymbol{\alpha})] \right\} \quad (4.2)$$

$$\mathcal{X}' = \left\{ \boldsymbol{\alpha} \in \mathbb{R}^{d+1} : \alpha_{1:d} \in \mathbb{R}^d, \alpha_{d+1} \in [0, p^*(\boldsymbol{\alpha})] \right\} \quad (4.3)$$

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

$$\Pr(\mathbf{x}, u) = \Pr(\mathbf{x}) \Pr(u) \quad (4.4)$$

$$= q(\mathbf{x}) \frac{1}{cq^*(\mathbf{x})} \quad (4.5)$$

$$= q(\mathbf{x}) \frac{1}{cZ_q q(\mathbf{x})} \quad (4.6)$$

$$= \frac{1}{cZ_q} \quad (4.7)$$

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

$$(\mathbf{x}, u) \sim \text{Unif}(\mathcal{X}) \quad (4.8)$$

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

$$(\mathbf{x}', u') = \begin{cases} (\mathbf{x}, u) & \text{if } (\mathbf{x}, u) \in \mathcal{X}' \\ \text{nothing} & \text{otherwise.} \end{cases} \quad (4.9)$$

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

$$\Pr(\mathbf{x}', u') = \begin{cases} \Pr(\mathbf{x}, u) & \text{if } (\mathbf{x}, u) \in \mathcal{X}' \\ 0 & \text{otherwise.} \end{cases} \quad (4.10)$$

which means

$$(\mathbf{x}', u') \sim \text{Unif}(\mathcal{X}') \quad (4.11)$$

Working backwards

$$\Pr(\mathbf{x}') = \frac{\Pr(\mathbf{x}', u')}{\Pr(u')} \quad (4.12)$$

$$\propto \frac{1}{1/p^*(\mathbf{x}')} \quad (4.13)$$

$$\propto p^*(\mathbf{x}') \quad (4.14)$$

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

### 4.3 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 2 below.

---

**Algorithm 2** 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  $E[f]$ .
-

### 4.3.1 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:

$$E_q[\text{numer}] = E_q \left[ \sum_r w_r f(\mathbf{x}^{(r)}) \right] \quad (4.15)$$

$$= \sum_r E_q \left[ w_r f(\mathbf{x}^{(r)}) \right] \quad (4.16)$$

$$= \sum_r E_q \left[ \frac{p^*(\mathbf{x}^{(r)})}{q^*(\mathbf{x}^{(r)})} f(\mathbf{x}^{(r)}) \right] \quad (4.17)$$

$$= \sum_r E_q \left[ \frac{Z_p p(\mathbf{x}^{(r)})}{Z_q q(\mathbf{x}^{(r)})} f(\mathbf{x}^{(r)}) \right] \quad (4.18)$$

$$= \frac{Z_p}{Z_q} \sum_r \int_{\mathbf{x}^{(r)}} p(\mathbf{x}^{(r)}) f(\mathbf{x}^{(r)}) d\mathbf{x}^{(r)} \quad (4.19)$$

$$= \frac{Z_p}{Z_q} \sum_r E_p \left[ f(\mathbf{x}^{(r)}) \right] \quad (4.20)$$

$$= \frac{Z_p}{Z_q} R E_p [f(\mathbf{x})] \quad (4.21)$$

$$E_q[\text{denom}] = E_q \left[ \sum_r w_r \right] \quad (4.22)$$

$$= \sum_r E_q \left[ \frac{p^*(\mathbf{x}^{(r)})}{q^*(\mathbf{x}^{(r)})} \right] \quad (4.23)$$

$$= \sum_r E_q \left[ \frac{Z_p p(\mathbf{x}^{(r)})}{Z_q q(\mathbf{x}^{(r)})} \right] \quad (4.24)$$

$$= \frac{Z_p}{Z_q} \sum_r \int_{\mathbf{x}^{(r)}} p(\mathbf{x}^{(r)}) d\mathbf{x}^{(r)} \quad (4.25)$$

$$= \frac{Z_p}{Z_q} R \quad (4.26)$$

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  $= E_p[f]$ ).

### 4.3.2 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

$$\text{var}_q \left[ \frac{p(\mathbf{x})}{q(\mathbf{x})} f(\mathbf{x}) \right] = E_q \left[ \frac{p^2(\mathbf{x})}{q^2(\mathbf{x})} f^2(\mathbf{x}) \right] - \left( E_q \left[ \frac{p(\mathbf{x})}{q(\mathbf{x})} f(\mathbf{x}) \right] \right)^2 \quad (4.27)$$

$$= \mathbb{E}_q \left[ \frac{p^2(\mathbf{x})}{q^2(\mathbf{x})} f^2(\mathbf{x}) \right] - (\mathbb{E}_p [f(\mathbf{x})])^2 \quad (4.28)$$

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:

$$\mathbb{E}_q \left[ \frac{p^2(\mathbf{x})}{q^2(\mathbf{x})} f^2(\mathbf{x}) \right] \geq \left( \mathbb{E}_q \left[ \frac{p(\mathbf{x})}{q(\mathbf{x})} |f(\mathbf{x})| \right] \right)^2 = (\mathbb{E}_p [|f(\mathbf{x})|])^2 \quad (4.29)$$

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

$$q_{\text{optimal}}(\mathbf{x}) = \frac{|f(\mathbf{x})|p(\mathbf{x})}{\int |f(\mathbf{x}')|p(\mathbf{x}') d\mathbf{x}'} \quad (4.30)$$

## 4.4 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 3 below.

---

### Algorithm 3 Sampling importance resampling

---

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

$$\hat{p}(d\mathbf{x}) = \sum_r \hat{w}_r \delta_{\mathbf{x}^{(r)}}(d\mathbf{x}) \quad (4.31)$$

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

---

#### 4.4.1 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

$$N(x) = \lim_{\delta x \rightarrow 0} Rq(x)\delta x \quad (4.32)$$

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

$$\lim_{\delta x \rightarrow 0} \Pr(x \leq x^{(r)} \leq x + \delta x) = N(x)\hat{w}_r \quad (4.33)$$

$$\propto \lim_{\delta x \rightarrow 0} Rq(x)\delta x \frac{p(x)}{q(x)} \quad (4.34)$$

$$\propto \lim_{\delta x \rightarrow 0} p(x) \delta x \quad (4.35)$$

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

$$\Pr(a \leq x^{(r)} \leq b) \propto \int_a^b p(x) dx \quad (4.36)$$

$$\implies x^{(r)} \sim p \quad (4.37)$$

## 4.5 Particle filtering

### 4.5.1 Sequential importance sampling (SIS)

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

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

$$q(\mathbf{x}_{1:t} | \mathbf{y}_{1:t}; \boldsymbol{\theta}) = q(\mathbf{x}_t | \mathbf{x}_{1:t-1}, \mathbf{y}_{1:t}; \boldsymbol{\theta}) q(\mathbf{x}_{1:t-1} | \mathbf{y}_{1:t}; \boldsymbol{\theta}) \quad (4.38)$$

$$= q(\mathbf{x}_t | \mathbf{x}_{1:t-1}, \mathbf{y}_{1:t}; \boldsymbol{\theta}) q(\mathbf{x}_{1:t-1} | \mathbf{y}_{1:t-1}; \boldsymbol{\theta}) \quad (4.39)$$

$$= q(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{y}_t; \boldsymbol{\theta}) \quad (4.40)$$

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

$$p(\mathbf{x}_{1:t} | \mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_{1:t} | \mathbf{x}_{1:t}) p(\mathbf{x}_{1:t})}{p(\mathbf{y}_{1:t})} \quad (4.41)$$

$$= \frac{p(\mathbf{y}_t | \mathbf{x}_{1:t}, \mathbf{y}_{1:t-1}) p(\mathbf{y}_{1:t-1} | \mathbf{x}_{1:t}) p(\mathbf{x}_{1:t})}{p(\mathbf{y}_t | \mathbf{y}_{1:t-1}) p(\mathbf{y}_{1:t-1})} \quad (4.42)$$

$$= \frac{p(\mathbf{y}_t | \mathbf{x}_{1:t}, \mathbf{y}_{1:t-1}) p(\mathbf{x}_{1:t} | \mathbf{y}_{1:t-1})}{p(\mathbf{y}_t | \mathbf{y}_{1:t-1})} \quad (4.43)$$

$$= \frac{p(\mathbf{y}_t | \mathbf{x}_{1:t}, \mathbf{y}_{1:t-1}) p(\mathbf{x}_t | \mathbf{x}_{1:t-1}, \mathbf{y}_{1:t-1}) p(\mathbf{x}_{1:t-1} | \mathbf{y}_{1:t-1})}{p(\mathbf{y}_t | \mathbf{y}_{1:t-1})} \quad (4.44)$$

$$= \frac{p(\mathbf{y}_t | \mathbf{x}_t) p(\mathbf{x}_t | \mathbf{x}_{t-1}) p(\mathbf{x}_{1:t-1} | \mathbf{y}_{1:t-1})}{p(\mathbf{y}_t | \mathbf{y}_{1:t-1})} \quad (4.45)$$

$$\propto p(\mathbf{y}_t | \mathbf{x}_t) p(\mathbf{x}_t | \mathbf{x}_{t-1}) p(\mathbf{x}_{1:t-1} | \mathbf{y}_{1:t-1}) \quad (4.46)$$

$$= g(\mathbf{y}_t | \mathbf{x}_t) f(\mathbf{x}_t | \mathbf{x}_{t-1}) p(\mathbf{x}_{1:t-1} | \mathbf{y}_{1:t-1}) \quad (4.47)$$

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

$$w_t^{(r)} \propto \frac{p(\mathbf{x}_{1:t}^{(r)} | \mathbf{y}_{1:t})}{q(\mathbf{x}_{1:t}^{(r)} | \mathbf{y}_{1:t})} \quad (4.48)$$

$$\propto \frac{p(\mathbf{y}_t | \mathbf{x}_t^{(r)}) p(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}) p(\mathbf{x}_{1:t-1}^{(r)} | \mathbf{y}_{1:t-1})}{q(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t) q(\mathbf{x}_{1:t-1}^{(r)} | \mathbf{y}_{1:t-1})} \quad (4.49)$$

$$= w_{t-1}^{(r)} \frac{p(\mathbf{y}_t | \mathbf{x}_t^{(r)}) p(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)})}{q(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t)} \quad (4.50)$$

$$= w_{t-1}^{(r)} \frac{g(\mathbf{y}_t | \mathbf{x}_t^{(r)}) f(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)})}{q(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t)} \quad (4.51)$$

For  $t = 1$

$$w_1^{(r)} \propto \frac{p(\mathbf{x}_1^{(r)} | \mathbf{y}_1)}{q(\mathbf{x}_1^{(r)} | \mathbf{y}_1)} \quad (4.52)$$

$$\propto \frac{p(\mathbf{x}_1^{(r)}, \mathbf{y}_1)}{q(\mathbf{x}_1^{(r)} | \mathbf{y}_1)} \quad (4.53)$$

$$\propto \frac{p(\mathbf{y}_1 | \mathbf{x}_1^{(r)}) p(\mathbf{x}_1^{(r)})}{q(\mathbf{x}_1^{(r)} | \mathbf{y}_1)} \quad (4.54)$$

$$= \frac{g(\mathbf{y}_1 | \mathbf{x}_1^{(r)}) \mu(\mathbf{x}_1^{(r)})}{q(\mathbf{x}_1^{(r)} | \mathbf{y}_1)} \quad (4.55)$$

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 4 below.

---

**Algorithm 4** Sequential importance sampling

---

1: Sample from proposal ▷ Initialisation

$$\mathbf{x}_1^{(r)} \sim q(\cdot | \mathbf{y}_1^{(r)}, \boldsymbol{\theta}), r = 1, \dots, R \quad (4.56)$$

2: Compute weights

$$w_1^{(r)} \propto \frac{g(\mathbf{y}_1 | \mathbf{x}_1^{(r)}) \mu(\mathbf{x}_1^{(r)})}{q(\mathbf{x}_1^{(r)} | \mathbf{y}_1)}, r = 1, \dots, R \quad (4.57)$$

3: Normalise weights

$$\hat{w}_1^{(r)} = \frac{w_1^{(r)}}{\sum_{r'} w_1^{(r')}}, r = 1, \dots, R \quad (4.58)$$

4: We can resample from

$$\hat{p}(\mathrm{d}\mathbf{x}_1 \mid \mathbf{y}_1, \boldsymbol{\theta}) = \sum_r \hat{w}_1^{(r)} \delta_{\mathbf{x}_1^{(r)}}(\mathrm{d}\mathbf{x}_1) \quad (4.59)$$

to estimate

$$p(\mathbf{x}_1 \mid \mathbf{y}_1, \boldsymbol{\theta}) \quad (4.60)$$

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

▷ Main loop

6:     Compute weights

$$w_t^{(r)} \propto w_{t-1}^{(r)} \frac{g(\mathbf{y}_t \mid \mathbf{x}_t^{(r)}, \boldsymbol{\theta}) f(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{A_{t-1}^{(r)}}, \boldsymbol{\theta})}{q(\mathbf{x}_t^{(r)} \mid \mathbf{x}_{t-1}^{A_{t-1}^{(r)}}, \mathbf{y}_t, \boldsymbol{\theta})}, r = 1, \dots, R \quad (4.61)$$

7:     Normalise weights

$$\hat{w}_t^{(r)} = \frac{w_t^{(r)}}{\sum_{r'} w_t^{(r')}}, r = 1, \dots, R \quad (4.62)$$

8:     We can resample from

$$\hat{p}(\mathrm{d}\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}, \boldsymbol{\theta}) = \sum_r \hat{w}_t^{(r)} \delta_{\mathbf{x}_{1:t}^{(r)}}(\mathrm{d}\mathbf{x}_{1:t}) \quad (4.63)$$

to estimate

$$p(\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}, \boldsymbol{\theta}) \quad (4.64)$$

The reason why it works is the same as in the case of Sampling importance resampling described in section 4.4.

### 4.5.2 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:

$$S_{\text{eff}} \triangleq \frac{S}{1 + \text{var} \left[ w_t^{(r)*} \right]} \quad (4.65)$$

$$\hat{S}_{\text{eff}} \approx \frac{1}{\sum_r \left( w_t^{(r)} \right)^2} \quad (4.66)$$



where  $w_t^{(r)*} = p(\mathbf{x}_t^{(r)} | \mathbf{y}_{1:t}) / q(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t)$  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.

### 4.5.3 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 5 below in which we resample during every tie step.

---

#### Algorithm 5 Generic particle filter

---

- 1: Sample from proposal ▷ Initialisation

$$\mathbf{x}_1^{(r)} \sim q(\cdot | \mathbf{y}_1, \boldsymbol{\theta}), r = 1, \dots, R \quad (4.67)$$

- 2: Compute weights

$$w_1^{(r)} \propto \frac{p(\mathbf{x}_1^{(r)} | \mathbf{y}_1, \boldsymbol{\theta})}{q(\mathbf{x}_1^{(r)} | \mathbf{y}_1, \boldsymbol{\theta})}, r = 1, \dots, R \quad (4.68)$$

- 3: Normalise weights

$$\hat{w}_1^{(r)} = \frac{w_1^{(r)}}{\sum_{r'} w_1^{(r')}}, r = 1, \dots, R \quad (4.69)$$

- 4: We can resample from

$$\hat{p}(d\mathbf{x}_1 | \mathbf{y}_1, \boldsymbol{\theta}) = \sum_r \hat{w}_1^{(r)} \delta_{\mathbf{x}_1^{(r)}}(d\mathbf{x}_1) \quad (4.70)$$

to estimate

$$p(\mathbf{x}_1 | \mathbf{y}_1, \boldsymbol{\theta}) \quad (4.71)$$

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

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

$$A_{t-1}^{(r)} \sim \text{Cat}(\hat{w}_{t-1}), r = 1, \dots, R \quad (4.72)$$

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

$$\mathbf{x}_t^{(r)} \sim q(\cdot | \mathbf{x}_{t-1}^{A_{t-1}^{(r)}}, \mathbf{y}_t, \boldsymbol{\theta}), r = 1, \dots, R \quad (4.73)$$

- 8:     Compute weights

$$w_t^{(r)} \propto w_{t-1}^{(r)} \frac{g(\mathbf{y}_t | \mathbf{x}_t^{(r)}, \boldsymbol{\theta}) f(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{A_{t-1}^{(r)}}, \boldsymbol{\theta})}{q(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{A_{t-1}^{(r)}}, \mathbf{y}_t, \boldsymbol{\theta})}, r = 1, \dots, R \quad (4.74)$$

9:     Normalise weights

$$\hat{w}_t^{(r)} = \frac{w_t^{(r)}}{\sum_{r'} w_t^{(r')}}, r = 1, \dots, R \quad (4.75)$$

10:    We can resample from

$$\hat{p}(\mathrm{d}\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}, \boldsymbol{\theta}) = \sum_r \hat{w}_t^{(r)} \delta_{\mathbf{x}_{1:t}^{(r)}}(\mathrm{d}\mathbf{x}_{1:t}) \quad (4.76)$$

to estimate

$$p(\mathbf{x}_{1:t} \mid \mathbf{y}_{1:t}, \boldsymbol{\theta}) \quad (4.77)$$

---

#### 4.5.4 Particle filter animation

#### 4.5.5 The proposal distribution

It is common to use the following proposal distribution

$$q(\mathbf{x}_{1:t}^{(r)} | \mathbf{y}_{1:t}) = q(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t) \quad (4.78)$$

$$= p(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}) \quad (4.79)$$

$$= f(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}) \quad (4.80)$$

Hence the weight equation in (4.51) becomes

$$w_t^{(r)} \propto w_{t-1}^{(r)} \frac{g(\mathbf{y}_t | \mathbf{x}_t^{(r)}) f(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)})}{q(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t)} \quad (4.81)$$

$$= w_{t-1}^{(r)} g(\mathbf{y}_t | \mathbf{x}_t^{(r)}) \quad (4.82)$$

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

$$q(\mathbf{x}_{1:t}^{(r)} | \mathbf{y}_{1:t}) = q(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t) \quad (4.83)$$

$$= p(\mathbf{x}_t^{(r)} | \mathbf{x}_{t-1}^{(r)}, \mathbf{y}_t) \quad (4.84)$$

$$= \frac{p(\mathbf{y}_t | \mathbf{x}_t, \mathbf{x}_{t-1}^{(r)}) p(\mathbf{x}_t, \mathbf{x}_{t-1}^{(r)})}{p(\mathbf{x}_{t-1}^{(r)} | \mathbf{y}_t)} \quad (4.85)$$

$$= \frac{p(\mathbf{y}_t | \mathbf{x}_t) p(\mathbf{x}_t | \mathbf{x}_{t-1}^{(r)})}{p(\mathbf{y}_t | \mathbf{x}_{t-1}^{(r)})} \quad (4.86)$$

$$= \frac{g(\mathbf{y}_t | \mathbf{x}_t) f(\mathbf{x}_t | \mathbf{x}_{t-1}^{(r)})}{p(\mathbf{y}_t | \mathbf{x}_{t-1}^{(r)})} \quad (4.87)$$

The weight equation in (4.51) becomes

$$w_t^{(r)} \propto w_{t-1}^{(r)} p(\mathbf{y}_t | \mathbf{x}_{t-1}^{(r)}) \quad (4.88)$$

$$= w_{t-1}^{(r)} \int p(\mathbf{y}_t, \mathbf{x}'_t | \mathbf{x}_{t-1}^{(r)}) d\mathbf{x}' \quad (4.89)$$

$$= w_{t-1}^{(r)} \int p(\mathbf{y}_t | \mathbf{x}'_t, \mathbf{x}_{t-1}^{(r)}) p(\mathbf{x}'_t | \mathbf{x}_{t-1}^{(r)}) d\mathbf{x}' \quad (4.90)$$

$$= w_{t-1}^{(r)} \int p(\mathbf{y}_t | \mathbf{x}'_t) p(\mathbf{x}'_t | \mathbf{x}_{t-1}^{(r)}) d\mathbf{x}' \quad (4.91)$$

$$= w_{t-1}^{(r)} \int g(\mathbf{y}_t | \mathbf{x}'_t) f(\mathbf{x}'_t | \mathbf{x}_{t-1}^{(r)}) d\mathbf{x}'_t \quad (4.92)$$

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.

## 4.6 Sequential Monte Carlo

(to do: improve to be more rigorous)

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

$$p_t(x_t) = p_{t-1}(x_{t-1})M_t(x_{t-1}, x_t) \quad (4.93)$$

Also assume that

$$\tilde{p}_t(x_{0:t}) = p_t(x_t) \sum_{k=1}^t L_k(x_k, x_{k-1}) \quad (4.94)$$

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

The generic algorithm for Sequential Monte Carlo (SMC) can be found in Algorithm 6.

---

### Algorithm 6 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.
-

## 4.7 Markov chain Monte Carlo methods

### 4.7.1 Definitions

**Definition 4.7.1.** 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 4.7.1** (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

$$\frac{1}{T} \sum_{t=1}^T f(\mathbf{X}_t) \xrightarrow[n \rightarrow \infty]{a.s.} \mathbb{E}[f(\mathbf{X})] \quad \text{where } \mathbf{X} \sim \pi \quad (4.95)$$

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

If further, it is aperiodic, then

$$\Pr(\mathbf{X}_T = \mathbf{x} \mid \mathbf{X}_0 = \mathbf{x}_0) \xrightarrow[n \rightarrow \infty]{} \pi(\mathbf{x}) \quad \forall \mathbf{x}, \mathbf{x}_0 \in \mathcal{X}. \quad (4.96)$$

A MC following these conditions is ergodic

**Definition 4.7.2.** 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 4.7.3.** A pmf  $\pi$  on  $\mathcal{X}$  is a stationary (invariant) distribution (w.r.t.  $\mathcal{T}$ ) if

$$\pi(\mathbf{X} = \mathbf{x}') = \sum_{\mathbf{x} \in \mathcal{X}} \pi(\mathbf{X} = \mathbf{x}) \mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') \quad \forall \mathbf{x}' \quad (4.97)$$

**Definition 4.7.4.** 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 4.7.5.** An irreducible MC  $(\mathbf{X}_t)$  is aperiodic if  $\forall a \in \mathcal{X}$ ,

$$\gcd\{t : \Pr(\mathbf{X}_t = a \mid \mathbf{X}_0 = a) > 0\} = 1. \quad (4.98)$$

**Definition 4.7.6.** 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 4.7.2.** 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 4.7.7.** A finite state MC described by  $\mathcal{T}$  is reversible if there exists a unique distribution  $\pi$  such that, for all  $\mathbf{x}, \mathbf{x}' \in \mathcal{X}$

$$\pi(\mathbf{x})\mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') = \pi(\mathbf{x}')\mathcal{T}(\mathbf{x}' \rightarrow \mathbf{x}). \quad (4.99)$$

This equation is called the detailed balance (DB).

**Proposition 4.7.1.** 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 (4.99), we want to prove the stationarity equation (4.97) to ensure  $\pi$  is a stationary distribution of  $\mathcal{T}$ . We have

$$\sum_{\mathbf{x} \in \mathcal{X}} \pi(\mathbf{x})\mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') = \sum_{\mathbf{x} \in \mathcal{X}} \pi(\mathbf{x}')\mathcal{T}(\mathbf{x}' \rightarrow \mathbf{x}) \quad (4.100)$$

$$= \sum_{\mathbf{x} \in \mathcal{X}} \pi(\mathbf{x}') \Pr(\mathbf{x} \mid \mathbf{x}') \quad (4.101)$$

$$= \pi(\mathbf{x}') \sum_{\mathbf{x} \in \mathcal{X}} \Pr(\mathbf{x} \mid \mathbf{x}') \quad (4.102)$$

$$= \pi(\mathbf{x}') \quad (4.103)$$

which proves the equation (4.97).  $\pi$  is the unique stationary distribution of  $\mathcal{T}$  because of Theorem 4.7.2.  $\square$

**Proposition 4.7.2.** 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

$$\mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') = \Pr(\mathbf{x}' \mid \mathbf{x}) \quad (4.104)$$

$$= \sum_k \Pr(\mathbf{x}', k \mid \mathbf{x}) \quad (4.105)$$

$$= \sum_k \Pr(\mathbf{x}' \mid k, \mathbf{x}) \Pr(k \mid \mathbf{x}) \quad (4.106)$$

$$= \sum_k \mathcal{T}_k(\mathbf{x} \rightarrow \mathbf{x}') p_k \quad (4.107)$$

Using this, we can prove the detailed balance as follows

$$\pi(\mathbf{x})\mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') = \pi(\mathbf{x}) \sum_k \mathcal{T}_k(\mathbf{x} \rightarrow \mathbf{x}') p_k \quad (4.108)$$

$$= \sum_k \pi(\mathbf{x}) \mathcal{T}_k(\mathbf{x} \rightarrow \mathbf{x}') p_k \quad (4.109)$$

$$= \sum_k \pi(\mathbf{x}') \mathcal{T}_k(\mathbf{x}' \rightarrow \mathbf{x}) p_k \quad (4.110)$$

$$= \pi(\mathbf{x}') \sum_k \mathcal{T}_k(\mathbf{x}' \rightarrow \mathbf{x}) p_k \quad (4.111)$$

$$= \pi(\mathbf{x}') \mathcal{T}(\mathbf{x}' \rightarrow \mathbf{x}) \quad (4.112)$$

□

**Proposition 4.7.3.** *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

$$\mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') = \Pr(\mathbf{x}' | \mathbf{x}) \quad (4.113)$$

$$= \sum_{\mathbf{x}_{1:K-1}} \Pr(\mathbf{x}', \mathbf{x}_{K-1}, \dots, \mathbf{x}_1 | \mathbf{x}) \quad (4.114)$$

$$= \sum_{\mathbf{x}_{1:K-1}} \Pr(\mathbf{x}_K, \dots, \mathbf{x}_1 | \mathbf{x}_0) \quad (4.115)$$

$$= \sum_{\mathbf{x}_{1:K-1}} \Pr(\mathbf{x}_1 | \mathbf{x}_0) \cdots \Pr(\mathbf{x}_K | \mathbf{x}_{K-1}) \quad (4.116)$$

$$= \sum_{\mathbf{x}_{1:K-1}} \mathcal{T}_1(\mathbf{x}_0 \rightarrow \mathbf{x}_1) \cdots \mathcal{T}_K(\mathbf{x}_{K-1} \rightarrow \mathbf{x}_K) \quad (4.117)$$

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

$$\sum_{\mathbf{x} \in \mathcal{X}} \pi(\mathbf{x}) \mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') = \sum_{\mathbf{x}_0} \pi(\mathbf{x}_0) \sum_{\mathbf{x}_{1:K-1}} \mathcal{T}_1(\mathbf{x}_0 \rightarrow \mathbf{x}_1) \cdots \mathcal{T}_K(\mathbf{x}_{K-1} \rightarrow \mathbf{x}_K) \quad (4.118)$$

$$= \sum_{\mathbf{x}_{0:K-1}} \pi(\mathbf{x}_0) \mathcal{T}_1(\mathbf{x}_0 \rightarrow \mathbf{x}_1) \cdots \mathcal{T}_K(\mathbf{x}_{K-1} \rightarrow \mathbf{x}_K) \quad (4.119)$$

$$= \sum_{\mathbf{x}_{0:K-1}} \mathcal{T}_1(\mathbf{x}_1 \rightarrow \mathbf{x}_0) \pi(\mathbf{x}_1) \cdots \mathcal{T}_K(\mathbf{x}_{K-1} \rightarrow \mathbf{x}_K) \quad (4.120)$$

...

$$= \sum_{\mathbf{x}_{0:K-1}} \mathcal{T}_1(\mathbf{x}_1 \rightarrow \mathbf{x}_0) \cdots \mathcal{T}_K(\mathbf{x}_K \rightarrow \mathbf{x}_{K-1}) \pi(\mathbf{x}_K) \quad (4.121)$$

$$= \pi(\mathbf{x}_K) \sum_{\mathbf{x}_{0:K-1}} \mathcal{T}_K(\mathbf{x}_K \rightarrow \mathbf{x}_{K-1}) \cdots \mathcal{T}_1(\mathbf{x}_1 \rightarrow \mathbf{x}_0) \quad (4.122)$$

$$= \pi(\mathbf{x}_K) \sum_{\mathbf{x}_{0:K-1}} \Pr(\mathbf{x}_{0:K-1} | \mathbf{x}_K) \quad (4.123)$$

$$= \pi(\mathbf{x}_K). \quad (4.124)$$

□



### 4.7.2 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  $q(\cdot | \mathbf{x}) \equiv q(\mathbf{x} \rightarrow \cdot)$ . 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 7.

---

#### Algorithm 7 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

$$\mathcal{A}(\mathbf{x}^{(t-1)} \rightarrow \mathbf{x}^{(t)}) = \min \left( 1, \frac{\pi^*(\mathbf{x}^{(t)}) q^*(\mathbf{x}^{(t)} \rightarrow \mathbf{x}^{(t-1)})}{\pi^*(\mathbf{x}^{(t-1)}) q^*(\mathbf{x}^{(t-1)} \rightarrow \mathbf{x}^{(t)})} \right) \quad (4.125)$$

- 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

$$\mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') = \begin{cases} q(\mathbf{x} \rightarrow \mathbf{x}') \mathcal{A}(\mathbf{x} \rightarrow \mathbf{x}') & \text{if } \mathbf{x} \neq \mathbf{x}' \\ q(\mathbf{x} \rightarrow \mathbf{x}) + \sum_{\mathbf{x}', \mathbf{x}' \neq \mathbf{x}} q(\mathbf{x} \rightarrow \mathbf{x}') (1 - \mathcal{A}(\mathbf{x} \rightarrow \mathbf{x}')) & \text{if } \mathbf{x} = \mathbf{x}' \end{cases} \quad (4.126)$$

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

$$\pi(\mathbf{x}) \mathcal{T}(\mathbf{x} \rightarrow \mathbf{x}') = \pi(\mathbf{x}) q(\mathbf{x} \rightarrow \mathbf{x}') \min \left( 1, \frac{\pi(\mathbf{x}') q(\mathbf{x}' \rightarrow \mathbf{x})}{\pi(\mathbf{x}) q(\mathbf{x} \rightarrow \mathbf{x}')} \right) \quad (4.127)$$

$$= \min (\pi(\mathbf{x}) q(\mathbf{x} \rightarrow \mathbf{x}'), \pi(\mathbf{x}') q(\mathbf{x}' \rightarrow \mathbf{x})) \quad (4.128)$$

$$= \pi(\mathbf{x}') q(\mathbf{x}' \rightarrow \mathbf{x}) \min \left( 1, \frac{\pi(\mathbf{x}) q(\mathbf{x} \rightarrow \mathbf{x}')}{\pi(\mathbf{x}') q(\mathbf{x}' \rightarrow \mathbf{x})} \right) \quad (4.129)$$

$$= \pi(\mathbf{x}') \mathcal{T}(\mathbf{x}' \rightarrow \mathbf{x}) \quad (4.130)$$

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.

### 4.7.3 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 (8) is given below.

---

**Algorithm 8** 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  $p$ :

$$p(\mathbf{x})\mathcal{T}_i(\mathbf{x} \rightarrow \mathbf{x}') = p(\mathbf{x})p(\mathbf{x}_{-i}, x'_i \mid \mathbf{x}) \quad (4.131)$$

$$= p(\mathbf{x}_{-i}, x'_i, \mathbf{x}) \quad (4.132)$$

$$= p(\mathbf{x}, x'_i, \mathbf{x}_{-i}) \quad (4.133)$$

$$= p(\mathbf{x}')p(\mathbf{x} \mid x'_i, \mathbf{x}_{-i}) \quad (4.134)$$

$$= p(\mathbf{x}')\mathcal{T}_i(\mathbf{x}' \rightarrow \mathbf{x}) \quad (4.135)$$

This is the premise of Proposition 4.7.3, 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  $p(\mathbf{x}' \mid \mathbf{x}) = p(x'_i \mid \mathbf{x})$  the acceptance probability is one (shown below) and so it is equivalent to one sampling step in Gibbs sampling.

$$\mathcal{A}(\mathbf{x} \rightarrow \mathbf{x}') = \min \left( 1, \frac{p(\mathbf{x}')p(\mathbf{x} \mid \mathbf{x}')}{p(\mathbf{x})p(\mathbf{x}' \mid \mathbf{x})} \right) \quad (4.136)$$

$$= \min \left( 1, \frac{p(\mathbf{x}', \mathbf{x})}{p(\mathbf{x}', \mathbf{x})} \right) \quad (4.137)$$

$$= 1 \quad (4.138)$$

## 4.8 Particle Markov Chain Monte Carlo

### 4.8.1 Particle independent Metropolis Hastings (PIMH) sampler

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

---

**Algorithm 9** Particle independent Metropolis Hastings sampler

---

1: Run SMC targetting

▷ Initial sweep  $s = 0$

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

2: Sample

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

3: Let

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

denote the corresponding marginal likelihood estimate.

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

▷ Main loop

5:   Run SMC targeting

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

6:   Sample

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

7:   Let

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

denote the corresponding marginal likelihood estimate

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

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

9:   **if** success **then**

10:     Set

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

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

11:   **else**

12:     Set

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

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

---

**4.8.2 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 10** 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 targetting

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

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

$$\boldsymbol{\theta}(s) = \boldsymbol{\theta}(s-1)$$

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

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

### 4.8.3 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 11 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

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

11: Normalise weights

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

12: We can resample from

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

to estimate

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

### Particle Gibbs sampler

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

#### Algorithm 12 Particle Gibbs sampler

- |  |                           |
|--|---------------------------|
| 1: Set $\theta(0)$ , $\mathbf{x}_{1:T}(0)$ , $B_{1:T}(0)$ 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 11) 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

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

## **5 Nonparametric Bayesian models**

# Bibliography

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