

# **Personal notes – Bayesian machine learning**

Tuan Anh Le

August 24, 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>Sampling algorithms</b>                          | <b>9</b> |
| 3.1      | Introduction . . . . .                              | 9        |
| 3.2      | Importance sampling . . . . .                       | 9        |
| 3.2.1    | Convergence of estimator as $R$ increases . . . . . | 9        |
| 3.2.2    | Optimal proposal distribution . . . . .             | 10       |
| 3.3      | Sampling importance resampling . . . . .            | 11       |
| 3.3.1    | Why it works? . . . . .                             | 11       |

|       |                              |    |
|-------|------------------------------|----|
| 3.4   | Rejection sampling . . . . . | 12 |
| 3.4.1 | Why it works? . . . . .      | 12 |
| 3.5   | Particle filtering . . . . . | 13 |

# **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

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

#### Evidence

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

### 2.5.2 Derivations



## 3 Sampling algorithms

### 3.1 Introduction

Let  $p$  be a probability distribution with pdf  $p(\mathbf{x})$ , 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  $Z_p = \int p^*(\mathbf{x}) d\mathbf{x}$ ). We want to achieve the following:

**Problem 1** Generate samples  $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(R)}\}$  (shorthand notation  $\{\mathbf{x}^{(r)}\}$ ) from the probability distribution  $p$ .

**Problem 2** Estimate the expectation of an arbitrary function  $f$  given that  $\mathbf{x} \sim p$ ,  $E[f]$ .

### 3.2 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

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  $E[f]$ .

#### 3.2.1 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:

$$E_q[\text{numer}] = E_q \left[ \sum_r w_r f(\mathbf{x}^{(r)}) \right] \quad (3.1)$$

$$= \sum_r E_q \left[ w_r f(\mathbf{x}^{(r)}) \right] \quad (3.2)$$

$$= \sum_r E_q \left[ \frac{p^*(\mathbf{x}^{(r)})}{q^*(\mathbf{x}^{(r)})} f(\mathbf{x}^{(r)}) \right] \quad (3.3)$$

$$= \sum_r E_q \left[ \frac{Z_p p(\mathbf{x}^{(r)})}{Z_q q(\mathbf{x}^{(r)})} f(\mathbf{x}^{(r)}) \right] \quad (3.4)$$

$$= \frac{Z_p}{Z_q} \sum_r \int_{\mathbf{x}^{(r)}} p(\mathbf{x}^{(r)}) f(\mathbf{x}^{(r)}) d\mathbf{x}^{(r)} \quad (3.5)$$

$$= \frac{Z_p}{Z_q} \sum_r \mathbb{E}_p [f(\mathbf{x}^{(r)})] \quad (3.6)$$

$$= \frac{Z_p}{Z_q} R \mathbb{E}_p [f(\mathbf{x})] \quad (3.7)$$

$$\mathbb{E}_q[\text{denom}] = \mathbb{E}_q \left[ \sum_r w_r \right] \quad (3.8)$$

$$= \sum_r \mathbb{E}_q \left[ \frac{p^*(\mathbf{x}^{(r)})}{q^*(\mathbf{x}^{(r)})} \right] \quad (3.9)$$

$$= \sum_r \mathbb{E}_q \left[ \frac{Z_p p(\mathbf{x}^{(r)})}{Z_q q(\mathbf{x}^{(r)})} \right] \quad (3.10)$$

$$= \frac{Z_p}{Z_q} \sum_r \int_{\mathbf{x}^{(r)}} p(\mathbf{x}^{(r)}) d\mathbf{x}^{(r)} \quad (3.11)$$

$$= \frac{Z_p}{Z_q} R \quad (3.12)$$

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

### 3.2.2 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

$$\text{var}_q \left[ \frac{p(\mathbf{x})}{q(\mathbf{x})} f(\mathbf{x}) \right] = \mathbb{E}_q \left[ \frac{p^2(\mathbf{x})}{q^2(\mathbf{x})} f^2(\mathbf{x}) \right] - \left( \mathbb{E}_q \left[ \frac{p(\mathbf{x})}{q(\mathbf{x})} f(\mathbf{x}) \right] \right)^2 \quad (3.13)$$

$$= \mathbb{E}_q \left[ \frac{p^2(\mathbf{x})}{q^2(\mathbf{x})} f^2(\mathbf{x}) \right] - (\mathbb{E}_p [f(\mathbf{x})])^2 \quad (3.14)$$

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:

$$\mathbb{E}_q \left[ \frac{p^2(\mathbf{x})}{q^2(\mathbf{x})} f^2(\mathbf{x}) \right] \geq \left( \mathbb{E}_q \left[ \frac{p(\mathbf{x})}{q(\mathbf{x})} |f(\mathbf{x})| \right] \right)^2 = (\mathbb{E}_p [|f(\mathbf{x})|])^2 \quad (3.15)$$

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

$$q_{\text{optimal}}(\mathbf{x}) = \frac{|f(\mathbf{x})|p(\mathbf{x})}{\int |f(\mathbf{x}')|p(\mathbf{x}') d\mathbf{x}'} \quad (3.16)$$

### 3.3 Sampling importance resampling

In Sampling importance resampling (SIR), we approximate the pdf of  $p$  as point masses and resample from them to get samples  $\{\mathbf{x}^{(r)}\}$  which are approximately  $\sim p$ . The process is as follows:

1. Generate samples  $\{\mathbf{z}^{(r)}\}$  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. Resample from a probability distribution with the pmf

$$f(\mathbf{x}) = \sum_r \hat{w}_r \delta_{\mathbf{z}^{(r)}}(\mathbf{x}) \quad (3.17)$$

5. The resulting samples  $\{\mathbf{x}^{(r)}\}$  are approximately  $\sim p$ .

#### 3.3.1 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

$$N(x) = \lim_{\delta x \rightarrow 0} Rq(x)\delta x \quad (3.18)$$

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

$$\lim_{\delta x \rightarrow 0} \Pr(x \leq x^{(r)} \leq x + \delta x) = N(x)\hat{w}_r \quad (3.19)$$

$$\propto \lim_{\delta x \rightarrow 0} Rq(x)\delta x \frac{p(x)}{q(x)} \quad (3.20)$$

$$\propto \lim_{\delta x \rightarrow 0} p(x)\delta x \quad (3.21)$$

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

$$\Pr(a \leq x^{(r)} \leq b) \propto \int_a^b p(x) dx \quad (3.22)$$

$$\implies x^{(r)} \sim p \quad (3.23)$$

### 3.4 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

$$cq^*(\mathbf{x}) > p^*(\mathbf{x}) \text{ for all } \mathbf{x} \quad (3.24)$$

The procedure that generates a sample  $\mathbf{x} \sim p$  is

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.

#### 3.4.1 Why it works?

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

$$\mathcal{X} = \{\boldsymbol{\alpha} \in \mathbb{R}^{d+1} : \alpha_{1:d} \in \mathbb{R}^d, \alpha_{d+1} \in [0, cq^*(\boldsymbol{\alpha})]\} \quad (3.25)$$

$$\mathcal{X}' = \{\boldsymbol{\alpha} \in \mathbb{R}^{d+1} : \alpha_{1:d} \in \mathbb{R}^d, \alpha_{d+1} \in [0, p^*(\boldsymbol{\alpha})]\} \quad (3.26)$$

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

$$\Pr(\mathbf{x}, u) = \Pr(\mathbf{x}) \Pr(u) \quad (3.27)$$

$$= q(\mathbf{x}) \frac{1}{cq^*(\mathbf{x})} \quad (3.28)$$

$$= q(\mathbf{x}) \frac{1}{cZ_q q(\mathbf{x})} \quad (3.29)$$

$$= \frac{1}{cZ_q} \quad (3.30)$$

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

$$(\mathbf{x}, u) \sim \text{Unif}(\mathcal{X}) \quad (3.31)$$

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

$$(\mathbf{x}', u') = \begin{cases} (\mathbf{x}, u) & \text{if } (\mathbf{x}, u) \in \mathcal{X}' \\ \text{nothing} & \text{otherwise.} \end{cases} \quad (3.32)$$

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

$$\Pr(\mathbf{x}', u') = \begin{cases} \Pr(\mathbf{x}, u) & \text{if } (\mathbf{x}, u) \in \mathcal{X}' \\ 0 & \text{otherwise.} \end{cases} \quad (3.33)$$

which means

$$(\mathbf{x}', u') \sim \text{Unif}(\mathcal{X}') \quad (3.34)$$

Working backwards

$$\Pr(\mathbf{x}') = \frac{\Pr(\mathbf{x}', u')}{\Pr(u')} \quad (3.35)$$

$$\propto \frac{1}{1/p^*(\mathbf{x}')} \quad (3.36)$$

$$\propto p^*(\mathbf{x}') \quad (3.37)$$

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

### 3.5 Particle filtering