

DD2434 - Machine Learning, Advanced Course
Assignment 1B

Tristan Perrot
tristanp@kth.se

November 2023



Contents

1	CAVI for Earth quakes	3
1.1	Question 1.1	3
1.2	Question 1.2	3
1.3	Question 1.3	4
A	Appendix	6
A.1	Question 1.2	6

1 CAVI for Earth quakes

1.1 Question 1.1

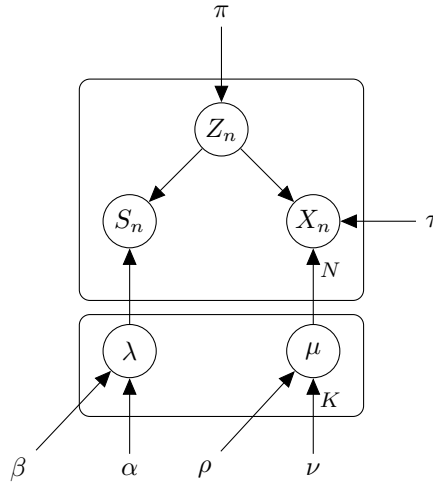


Figure 1: Directed Graphical Model for the Earthquake problem

1.2 Question 1.2

Here, we know these distributions:

- $p(Z_n|\pi) = \text{Categorical}(\pi)$
- $p(S_n|Z_n = k, \lambda_k) = \text{Poisson}(\lambda_k)$
- $p(X_n|Z_n = k, \mu_k, \tau) = \text{Normal}(\mu_k, \tau \cdot I)$
- $p(\mu_k|\nu, \rho) = \text{Normal}(\nu, \rho \cdot I)$
- $p(\lambda_k|\alpha, \beta) = \text{Gamma}(\alpha, \beta)$

Where, ρ and τ define precision and not standard variation. Then we have:

$$\begin{aligned}
 \log p(X, S, Z, \lambda, \mu|\pi, \tau, \alpha, \beta, \nu, \rho) &= \log p(X|S, Z, \lambda, \mu, \pi, \tau, \alpha, \beta, \nu, \rho) \\
 &\quad + \log p(S, Z, \lambda, \mu|\pi, \alpha, \beta, \nu, \rho) \\
 &= \log p(X|Z, \mu, \tau) + \log p(S|Z, \lambda, \mu, \pi, \alpha, \beta, \nu, \rho) \\
 &\quad + \log p(Z, \lambda, \mu|\pi, \alpha, \beta, \nu, \rho) \\
 &= \log p(X|Z, \mu, \tau) + \log p(S|Z, \lambda) + \log p(Z|\pi) \\
 &\quad + \log p(\lambda, \mu|\alpha, \beta, \nu, \rho) \\
 \log p(X, S, Z, \lambda, \mu|\pi, \tau, \alpha, \beta, \nu, \rho) &= \log p(X|Z, \mu, \tau) + \log p(S|Z, \lambda) + \log p(Z|\pi) \\
 &\quad + \log p(\mu|\nu, \rho) + \log p(\lambda|\alpha, \beta)
 \end{aligned} \tag{1}$$

Where:

$$\begin{aligned}
 \log p(X|Z, \mu, \tau) &= \sum_{n=1}^N \sum_{k=1}^K \log p(X_n|Z_n = k, \mu_k, \tau) \\
 \log p(S|Z, \lambda) &= \sum_{n=1}^N \sum_{k=1}^K \log p(S_n|Z_n = k, \lambda_k) \\
 \log p(Z|\pi) &= \sum_{n=1}^N \log p(Z_n|\pi) \\
 \log p(\mu|\nu, \rho) &= \sum_{k=1}^K \log p(\mu_k|\nu, \rho) \\
 \log p(\lambda|\alpha, \beta) &= \sum_{k=1}^K \log p(\lambda_k|\alpha, \beta)
 \end{aligned} \tag{2}$$

1.3 Question 1.3

Here, the mean field approximation is not an approximation but an equality because Z, μ, λ are independent. Therefore we have:

$$\begin{aligned}
 \log q^*(Z_n) &\stackrel{\pm}{=} \mathbb{E}_{\mu, \lambda} [\log p(X_n, S_n, Z_n, \lambda, \mu|\pi, \tau, \alpha, \beta, \nu, \rho)] \\
 &\stackrel{\pm}{=} \mathbb{E}_{\mu, \lambda} [\log p(X_n|Z_n, \mu, \tau) + \log p(S_n|Z_n, \lambda) + \log p(Z_n|\pi)] \\
 &= \mathbb{E}_{\mu} \left[\sum_{k=1}^K \mathbb{1}_{\{Z_n=k\}} \left(\log \left(\frac{\tau}{2\pi} \right) - \frac{\tau}{2} ((x_n - \mu_k)^T (x_n - \mu_k)) \right) \right] \\
 &\quad + \mathbb{E}_{\lambda} \left[\sum_{k=1}^K \mathbb{1}_{\{Z_n=k\}} (\log(\pi_k) - \lambda_k + S_n \log(\lambda_k) - \log(S_n!)) \right] \\
 &\stackrel{\pm}{=} \sum_{k=1}^K \mathbb{1}_{\{Z_n=k\}} \left(\log \left(\frac{\tau}{2\pi} \right) - \frac{\tau}{2} \mathbb{E}_{\mu} [(x_n - \mu_k)^T (x_n - \mu_k)] \right. \\
 &\quad \left. + \log(\pi_k) + \mathbb{E}_{\lambda} [-\lambda_k + S_n \log(\lambda_k)] - \log(S_n!) \right)
 \end{aligned} \tag{3}$$

Now, if we take the entire expression that is multiplied by $\mathbb{1}_{\{Z_n=k\}}$ and we call it $u_{n,k}$, we have:

$$q^*(Z_n) \propto \prod_{k=1}^K u_{n,k}^{\mathbb{1}_{\{Z_n=k\}}} \tag{4}$$

And if we normalize by taking $r_{n,k} = \frac{u_{n,k}}{\sum_{i=1}^K u_{n,i}}$ we get:

$$q^*(Z_n) = \prod_{k=1}^K r_{n,k}^{\mathbb{1}_{\{Z_n=k\}}} \tag{5}$$

Wich means that $q^*(Z_n)$ is a categorical distribution with parameters $r_{n,k}$. There for we have the expectation of Z_n easily because $\mathbb{E}[z_{n,k}] = r_{n,k}$ where $z_{n,k} = \mathbb{1}_{\{S_n=k\}}$. Note that $r_{n,k}$ depends of

the expected value of μ_k , μ_k^2 , λ_k and $\log \lambda_k$. We will be able to compute these expected values by finding $q^*(\mu_k)$ and $q^*(\lambda_k)$.

Let us compute $q^*(\mu_k)$:

$$\begin{aligned}
 \log q^*(\mu_k) &\stackrel{\pm}{=} \mathbb{E}_{Z,\lambda}[\log p(X, S, Z = k, \lambda_k, \mu_k | \pi, \tau, \alpha, \beta, \nu, \rho)] \\
 &\stackrel{\pm}{=} \mathbb{E}_{Z,\lambda}[\log p(X|Z = k, \mu_k, \tau) + \log p(\mu_k | \nu, \rho)] \\
 &= \mathbb{E}_{Z,\lambda} \left[\sum_{n=1}^N \mathbb{1}_{\{Z_n=k\}} \left(\log \left(\frac{\tau}{2\pi} \right) - \frac{\tau}{2} ((x_n - \mu_k)^T (x_n - \mu_k)) \right) \right] \\
 &\quad + \log \left(\frac{\rho}{2\pi} \right) - \frac{\rho}{2} ((\mu_k - \nu)^T (\mu_k - \nu)) \\
 &\stackrel{\pm}{=} \sum_{n=1}^N r_{n,k} \left(\log \left(\frac{\tau}{2\pi} \right) - \frac{\tau}{2} ((x_n - \mu_k)^T (x_n - \mu_k)) \right) - \frac{\rho}{2} ((\mu_k - \nu)^T (\mu_k - \nu))
 \end{aligned} \tag{6}$$

$$\begin{aligned}
 \log q^*(\lambda_k) &\stackrel{\pm}{=} \mathbb{E}_{Z,\mu}[\log p(X, S, Z = k, \lambda_k, \mu_k | \pi, \tau, \alpha, \beta, \nu, \rho)] \\
 &\stackrel{\pm}{=} \mathbb{E}_{Z,\mu}[\log p(S|Z = k, \lambda_k) + \log p(\lambda_k | \alpha, \beta)] \\
 &= \mathbb{E}_Z \left[\sum_{n=1}^N \mathbb{1}_{\{Z_n=k\}} (-\lambda_k + S_n \log(\lambda_k) - \log(S_n!)) \right] \\
 &\quad + \log \left(\frac{\beta^\alpha}{\Gamma(\alpha)} \right) + (\alpha - 1) \log(\lambda_k) - \beta \lambda_k \\
 &\stackrel{\pm}{=} \sum_{n=1}^N r_{n,k} (-\lambda_k + S_n \log(\lambda_k)) + (\alpha - 1) \log(\lambda_k) - \beta \lambda_k \\
 &= \left(\alpha + \sum_{n=1}^N S_n r_{n,k} - 1 \right) \log(\lambda_k) - \left(\beta + \sum_{n=1}^N r_{n,k} \right) \lambda_k
 \end{aligned} \tag{7}$$

Therefore, we have $q^*(\lambda_k) = \text{Gamma} \left(\alpha + \sum_{n=1}^N S_n r_{n,k}, \beta + \sum_{n=1}^N r_{n,k} \right)$. And we can compute the expected value of λ_k and $\log \lambda_k$ easily.

$$\begin{aligned}
 \mathbb{E}[\lambda_k] &= \frac{\alpha + \sum_{n=1}^N S_n r_{n,k}}{\beta + \sum_{n=1}^N r_{n,k}} \\
 \mathbb{E}[\log \lambda_k] &= \psi \left(\alpha + \sum_{n=1}^N S_n r_{n,k} \right) - \log \left(\beta + \sum_{n=1}^N r_{n,k} \right)
 \end{aligned} \tag{8}$$

A Appendix

A.1 Question 1.2

```
import matplotlib.pyplot as plt
import numpy as np
from scipy.stats import gamma, norm
from scipy.special import psi
np.random.seed(14)

def generate_data(mu, tau, N):
    # Insert your code here
    D = np.random.normal(mu, np.sqrt(1/tau), N)

    return D

MU = 1
TAU = 0.5

dataset_1 = generate_data(MU, TAU, 10)
dataset_2 = generate_data(MU, TAU, 100)
dataset_3 = generate_data(MU, TAU, 1000)

# Visualize the datasets via histograms
# Insert your code here
fig, axs = plt.subplots(1, 3, figsize=(12, 4))
axs[0].hist(dataset_1, bins=20)
axs[1].hist(dataset_2, bins=20)
axs[2].hist(dataset_3, bins=20)
plt.tight_layout()
plt.savefig('12_data.png')
plt.show()
```