CS509 Pattern Recognition Assignment 2

Wentao Lu (002276355), Yi Ren (002269013)

1 Pareto distribution

The log-likelihood function is given by:

$$\begin{aligned}\ln L(\alpha|D) &= \sum_{i=1}^{n} \ln f(x_i|\alpha) = \sum_{i=1}^{n} \ln \frac{\alpha - 1}{x_i^{\alpha}} \\ &= n \ln(\alpha - 1) - \alpha \sum_{i=1}^{n} (\ln x_i) \\ \Rightarrow \frac{d \ln L(\alpha|D)}{d\alpha} &= \frac{n}{\alpha - 1} - \sum_{i=1}^{n} (\ln x_i)\end{aligned}$$

Set the derivative equal to 0, we have:

$$\hat{\alpha} = \frac{n}{\sum_{i=1}^{n} (\ln x_i)} + 1, \text{ where } x_i \geq 1$$

2 Uniform distribution

The log-likelihood function is given by:

$$\begin{aligned}\ln L(\alpha|D) &= \sum_{i=1}^{n} \ln f(x_i|\alpha) = \sum_{i=1}^{n} \ln \frac{1}{a} \\ &= -\sum_{i=1}^{n} \ln a = -n \ln a\end{aligned}$$

Since the log-likelihood function monotonically decreases as a increases, to maximize it, we want a to be as small as possible. On the other hand, we also have the constraint that $a \geq x$, which must hold true for every x_i, so $a \geq max(x_i)$. Combine the two, we have the MLE estimate:

$$\hat{a} = max(x_i), \text{ where } 0 \leq x_i \leq a$$

3 Problem 3

1. Since $P(Z_{ik} = 1) = P(\omega_i)$, and $P(Z_{ik} = 0) = 1 - P(\omega_i)$, so Z_{ik} is a Bernoulli random variable with probability $P(\omega_i)$, then $P(Z_{ik}|P(\omega_i)) = P(\omega_i)^{Z_{ik}}(1 - P(\omega_i))^{1-Z_{ik}}$. Given that samples are drawn by successive, **independent** selections, we have:

$$\begin{aligned}P(Z_{i1}, \ldots, Z_{in}|P(\omega_i)) &= \prod_{k=1}^{n} P(Z_{ik}|P(\omega_i)) \\ &= \prod_{k=1}^{n} P(\omega_i)^{Z_{ik}}(1 - P(\omega_i)^{1-Z_{ik}}\end{aligned}$$

2. The log-likelihood function is given by:

$$\begin{aligned}
\ln L(P(\omega_i)|Z_{i1},\ldots,Z_{in}) &= \sum_{k=1}^{n} \ln P(Z_{ik}|P(\omega_i)) \\
&= \sum_{k=1}^{n} \ln[P(\omega_i)^{Z_{ik}}(1-P(\omega_i))^{1-Z_{ik}}] \\
&= \sum_{k=1}^{n} [Z_{ik} \ln P(\omega_i) + (1-Z_{ik}) \ln(1-P(\omega_i))] \\
&= \ln P(\omega_i) \sum_{k=1}^{n} Z_{ik} + \ln(1-P(\omega_i)) \sum_{k=1}^{n} (1-Z_{ik}) \\
&= \ln P(\omega_i) \sum_{k=1}^{n} Z_{ik} + \ln(1-P(\omega_i))(n - \sum_{k=1}^{n} Z_{ik})
\end{aligned}$$

Set the derivative equal to 0, we have:

$$\begin{aligned}
\frac{d \ln L(P(\omega_i)|Z_{i1},\ldots,Z_{in})}{dP(\omega_i)} &= \frac{\sum_{k=1}^{n} Z_{ik}}{P(\omega_i)} - \frac{n - \sum_{k=1}^{n} Z_{ik}}{1-P(\omega_i)} = 0 \\
\Rightarrow (1-P(\omega_i)) \sum_{k=1}^{n} Z_{ik} &= P(\omega_i)(n - \sum_{k=1}^{n} Z_{ik}) \\
\Rightarrow \widehat{P(\omega_i)} &= \frac{1}{n} \sum_{k=1}^{n} Z_{ik}
\end{aligned}$$

To verbally interpret this result, the MLE estimate of the prior is simply the sample mean of Z_{ik}, or the frequency of samples drawn from the state of nature ω_i. This is intuitive since the more samples we see from class ω_i, the higher the prior should be, this is equivalent to tossing a coin several times and use the frequency of heads or tails as the prior of heads or tails. In fact, it's the same as how we extend the Bernoulli to Binomial distribution.

4 Problem 4

In a nutshell, the problem description tells us that $\hat{x}_i \sim N(\theta, \sigma^2)$ where σ is known, but θ is unknown. In addition, we know from old experiments that the prior $p(\theta) \sim N(0, \Sigma^2)$.

1. By definition, the likelihood function can be written as:

$$\begin{aligned}
L(\theta) = P(\hat{x}|\theta) &= \prod_{i=1}^{n} P(\hat{x}_i|\theta) \text{ ($\hat{x}_i$ are independently drawn samples)} \\
&= \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \cdot exp[-\frac{(\hat{x}_i - \theta)^2}{2\sigma^2}] \\
&= c \cdot exp[-\frac{1}{2\sigma^2}(\sum_{i=1}^{n} \hat{x}_i{}^2 - 2\theta n\bar{x} + n\theta^2)] \\
&= c \cdot exp[-\frac{n}{2\sigma^2}(\frac{1}{n}\sum_{i=1}^{n} \hat{x}_i{}^2 - 2\theta\bar{x} + \theta^2 + \bar{x}^2 - \bar{x}^2)] \\
&= c \cdot exp[-\frac{n}{2\sigma^2}(\bar{x} - \theta)^2 - \frac{n}{2\sigma^2}(\frac{1}{n}\sum_{i=1}^{n} \hat{x}_i{}^2 - \bar{x}^2)] \\
&= c \cdot exp[-\frac{n}{2\sigma^2}(\bar{x} - \theta)^2] \cdot exp[-\frac{n}{2\sigma^2}(\frac{1}{n}\sum_{i=1}^{n} \hat{x}_i{}^2 - \bar{x}^2)] \\
&= c' \cdot exp[-\frac{n}{2\sigma^2}(\bar{x} - \theta)^2] \text{ (the last term on the line above is constant)} \\
&= L_0 \cdot exp[-\frac{(\bar{x} - \theta)^2}{2\sigma^2/n}]
\end{aligned}$$

2. The posterior probability can be computed as:

$$\begin{aligned}
p(\theta|\hat{x}) &= \frac{p(\hat{x}|\theta)p(\theta)}{p(\hat{x})} \\
&= c \cdot L(\theta)p(\theta) \\
&= c' \cdot exp[-\frac{n}{2\sigma^2}(\theta - \bar{x})^2] \cdot exp(-\frac{\theta^2}{2\Sigma^2}) \\
&= c' \cdot exp\{-\frac{1}{2}[\frac{n}{\sigma^2}(\theta - \bar{x})^2 + \frac{\theta^2}{\Sigma^2}]\}
\end{aligned}$$

   Notice that the term inside the square brackets on the line above is a quadratic function of θ, so after some heavy calculations, we can rearrange it into its complete quadratic form:

$$\frac{n}{\sigma^2}(\theta - \bar{x})^2 + \frac{\theta^2}{\Sigma^2} = (\frac{1}{\Sigma^2} + \frac{n}{\sigma^2})(\theta - \bar{x}\frac{\Sigma^2}{\Sigma^2 + \frac{\sigma^2}{n}})^2$$

   Hence, the posterior probability is proportional to a Gaussian distribution:

$$p(\theta|\hat{x}) \sim k \cdot N(\bar{x}\frac{\Sigma^2}{\Sigma^2 + \frac{\sigma^2}{n}}, (\frac{1}{\Sigma^2} + \frac{n}{\sigma^2})^{-1})$$

   where k is a constant independent of θ

3. Based on the previous result, as $n \to \infty$, the posterior distribution mean $\to \bar{x}$, and the variance $\to \frac{\sigma^2}{n}$ (because $\frac{n}{\sigma^2}$ dominates $\frac{1}{\Sigma^2}$). Therefore, Σ plays no role anymore, so the posterior distribution becomes independent of the prior. In particular, the limit of variance goes to 0 and uncertainty is completely eliminated, so with infinite data available, we can be 100% confident about our result.

4. As $n \to \infty$, now that the posterior distribution is the normal distribution:

$$p(\theta|\hat{x}) \to k \cdot N(\bar{x}, \frac{\sigma^2}{n})$$

As we know from the lecture, the MLE estimate of the mean of θ is just the sample mean $\bar{x}$ because this is Gaussian. On the other hand, the mean of the posterior distribution can be computed as:

$$\int \theta p(\theta|\hat{x}) d\theta = E(\theta|\hat{x})$$

which also happens to be the sample mean $\bar{x}$ because this is Gaussian.

In conclusion, the posterior distribution converges to the MLE solution. **Q.E.D.**

5 Problem 5

In a Gaussian distribution, we can combine the two partial derivatives of the log-likelihood function w.r.t. the mean and variance, let them both equal to 0, then solving the combined equations will give us the MLE estimate. For the univariate case we have:

$$\hat{\mu} = \frac{1}{n} \sum_{k=1}^{n} x_k$$
$$\hat{\sigma}^2 = \frac{1}{n} \sum_{k=1}^{n} (x_k - \hat{\mu})^2$$

For the multivariate case we have:

$$\hat{\boldsymbol{\mu}} = \frac{1}{n} \sum_{k=1}^{n} \boldsymbol{x}_k$$
$$\hat{\boldsymbol{\Sigma}} = \frac{1}{n} \sum_{k=1}^{n} (\boldsymbol{x}_k - \hat{\boldsymbol{\mu}})(\boldsymbol{x}_k - \hat{\boldsymbol{\mu}})^T$$

1. 1-dimensional MLE

```python
import os
import numpy as np

def mle_univariate(array):
    mu = np.mean(array)
    sigma = np.mean((array - mu) ** 2)
    return mu, sigma

def mle_mulvariate(array):
    mu = np.mean(array, axis=0)
    sigma = (1 / array.shape[0]) * (array - mu).T @ (array - mu)
    return mu, sigma

if __name__ == '__main__':
    datafile = os.path.join(os.getcwd(), 'Assignment2_dataset.txt')
    w1 = np.loadtxt(datafile, skiprows=1, max_rows=10)

    for i in range(3):
        mu, sigma = mle_univariate(w1[:, i])
        with np.printoptions(precision=2, suppress=True):
            print(f'x{i + 1} mu:', np.round(mu, 4))
```

```
            print(f'x{i + 1} sigma:', np.round(sigma, 4))

    '''result'''
    # x1 mu: -0.0709
    # x2 mu: -0.6047
    # x3 mu: -0.9110
    # x1 sigma: 0.9062
    # x2 sigma: 4.2007
    # x3 sigma: 4.5419
```

2. 2-dimensional MLE

```
    ...  # same code as before

    for i in range(3):
        combo = [1, 2, 3]
        combo.pop(i)
        array = np.delete(w1, i, axis=1)
        mu, sigma = mle_mulvariate(array)

        with np.printoptions(precision=4, suppress=True):
            print(f'x{combo} mu:', mu)
            print(f'x{combo} sigma:', sigma)

    '''result'''
    # x[2, 3] mu:    [-0.6047 -0.911 ]
    # x[2, 3] sigma: [[4.2007 0.7337]
    #                 [0.7337 4.5419]]
    # x[1, 3] mu:    [-0.0709 -0.911 ]
    # x[1, 3] sigma: [[0.9062 0.3941]
    #                 [0.3941 4.5419]]
    # x[1, 2] mu:    [-0.0709 -0.6047]
    # x[1, 2] sigma: [[0.9062 0.5678]
    #                 [0.5678 4.2007]]
```

3. 3-dimensional MLE

```
    ...  # same code as before

    mu, sigma = mle_mulvariate(w1)
    with np.printoptions(precision=4, suppress=True):
        print('mu:', mu)
        print('sigma:', sigma)

    '''result'''
    # mu:    [-0.0709 -0.6047 -0.911 ]
    # sigma: [[0.9062 0.5678 0.3941]
    #         [0.5678 4.2007 0.7337]
    #         [0.3941 0.7337 4.5419]]
```

4. If the 3-dimensional model is separable, the covariance matrix reduces to a diagonal matrix. In other words, the 3 features are mutually independent since any pair of them has a correlation of 0. In this case, we can simply treat it as an 1-dimensional model and apply the univariate MLE on each feature, therefore the code is the same as 1, but applied on ω_2.

```
    w2 = np.loadtxt(datafile, skiprows=12, max_rows=10)

    for i in range(3):
        mu, sigma = mle_univariate(w2[:, i])
```

```
        with np.printoptions(precision=4, suppress=True):
            print(mu, sigma)

    '''result'''
    # mu:    [-0.1126 0.4299 0.0037 ]
    # sigma: [[0.0539  0.      0.]
    #         [0.      0.0460  0.]
    #         [0.      0.      0.0073]]
```

5. It's easy to see that the mean in every case turns out to be the same, which is self-evident because the mean of any single feature has nothing to do with other features at all. Besides, the MLE estimate of mean is unbiased since its expectation equals the true mean.

6. Likewise, we can observe that the variance stays the same across all three cases for the same reason, which is exclusive to a specific feature. However, our multivariate MLE has also introduced non-zero covariance terms off the diagonal, which serve to model the level of dependence between two different features, this piece of information in some scenarios can be very useful if the problem of interest does **not** explicitly assumes **i.i.d**.

   In practice, we do not always have samples that are drawn independently, a typical example is the ARIMA-GARCH model where time series data features such as stock prices are strongly autocorrelated. In such a case, **it's important to note that** the multivariate MLE is more likely to give us reliable estimates since we also need to incoroporate the covariance or "dependence" between features in the likelihood function.

6 EM algorithm for a Gaussian mixture

From the exercise, it's easy to observe that in the general arbitrary case, the optimal choice for $\boldsymbol{\mu_k}$ is independent of $\boldsymbol{\Sigma_k}$. As a result, now in our special case where all components share a common covariance matrix, the constraint still has no impact on the choice of $\boldsymbol{\mu_k}$, which must be the same as we computed before in the exercise:

$$\boldsymbol{\mu_k} = \frac{\sum_{i=1}^{n} \gamma(z_{ik}) x_i}{\sum_{i=1}^{n} \gamma(z_{ik})}$$

Now that's turn to solve that common covariance matrix, start from the log-likelihood function:

$$\begin{aligned}
\ln L(\Theta|X) &= \sum_{i=1}^{n}\sum_{k=1}^{K} \gamma(z_{ik})[\ln \alpha_k + \ln N(\boldsymbol{x_i}|\boldsymbol{\mu_k}, \boldsymbol{\Sigma})] \\
&= \sum_{i=1}^{n}\sum_{k=1}^{K} \gamma(z_{ik})\{\ln \alpha_k + \ln[\frac{1}{(2\pi)^{\frac{d}{2}}|\boldsymbol{\Sigma}|^{\frac{1}{2}}} exp[-\frac{1}{2}(\boldsymbol{x_i} - \boldsymbol{\mu_k})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x_i} - \boldsymbol{\mu_k})]]\} \\
&= \sum_{i=1}^{n}\sum_{k=1}^{K} \gamma(z_{ik})[\ln \alpha_k - \frac{d}{2}\ln 2\pi - \frac{1}{2}\ln|\boldsymbol{\Sigma}| - \frac{1}{2}(\boldsymbol{x_i} - \boldsymbol{\mu_k})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x_i} - \boldsymbol{\mu_k})]
\end{aligned}$$

$$\begin{aligned}
\frac{\partial L(\Theta)}{\partial \boldsymbol{\Sigma}} &= \sum_{i=1}^{n}\sum_{k=1}^{K} \gamma(z_{ik})[0 - 0 - \frac{1}{2}\boldsymbol{\Sigma}^{-1} + \frac{1}{2}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x_i} - \boldsymbol{\mu_k})(\boldsymbol{x_i} - \boldsymbol{\mu_k})^T \boldsymbol{\Sigma}^{-1}] \\
&= -\frac{1}{2}\sum_{i=1}^{n}\sum_{k=1}^{K} \gamma(z_{ik})[\boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1}(\boldsymbol{x_i} - \boldsymbol{\mu_k})(\boldsymbol{x_i} - \boldsymbol{\mu_k})^T \boldsymbol{\Sigma}^{-1}] = 0
\end{aligned}$$

$$\implies [\sum_{i=1}^{n}\sum_{k=1}^{K}\gamma(z_{ik})]\boldsymbol{\Sigma}^{-1} = \boldsymbol{\Sigma}^{-1}[\sum_{i=1}^{n}\sum_{k=1}^{K}\gamma(z_{ik})(\boldsymbol{x_i}-\boldsymbol{\mu_k})(\boldsymbol{x_i}-\boldsymbol{\mu_k})^T]\boldsymbol{\Sigma}^{-1}$$

$$\implies \sum_{i=1}^{n}\sum_{k=1}^{K}\gamma(z_{ik}) = \boldsymbol{\Sigma}^{-1}[\sum_{i=1}^{n}\sum_{k=1}^{K}\gamma(z_{ik})(\boldsymbol{x_i}-\boldsymbol{\mu_k})(\boldsymbol{x_i}-\boldsymbol{\mu_k})^T]$$

$$\implies \boldsymbol{\Sigma} = \frac{\sum_{i=1}^{n}\sum_{k=1}^{K}\gamma(z_{ik})(\boldsymbol{x_i}-\boldsymbol{\mu_k})(\boldsymbol{x_i}-\boldsymbol{\mu_k})^T}{\sum_{i=1}^{n}\sum_{k=1}^{K}\gamma(z_{ik})}$$