Lecture Notes on Gaussian Discriminant Analysis, Naive Bayes and EM Algorithm

Feng Li
fli@sdu.edu.cn
Shandong University, China

1 Bayes' Theorem and Inference

Bayes' theorem is stated mathematically as the following equation

$$P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)} \tag{1}$$

where $P(A \mid B)$ is the conditional probability of event A given event B happens, $P(B \mid A)$ is the conditional probability of event B given A is true, and $P(A)$ and $P(B)$ are probability of observing A and B, respectively.

We now introduce Bayesian inference by taking image recognition as an example. Our aim is to identify if there is a cat in a given image. We assume $X = [X_1, X_2, \cdots, X_n]^T$ is a random variable representing the feature vector of the given image, and $Y \in \{0, 1\}$ is a random variable representing if there is a cat in the given image. Now, given an image $x = [x_1, x_2, \cdots, x_n]^T$, out goal is to calculate

$$P(Y = y \mid X = x) = \frac{P(X = x \mid Y = y)P(Y = y)}{P(X = x)} \tag{2}$$

where $y \in \{0, 1\}$. In particular, $P(Y = y \mid X = x)$ is the probability that the image is labeled by y given that the image can be represented by feature vector x, $P(X = x \mid Y = y)$ is the probability that the image has its feature vector being x given that it is labeled by y, $P(Y = y)$ is the probability that a randomly picked image is labeled by y, and $P(X = x)$ is the probability that a randomly picked image has label y. In our case, we make decision by calculating

$$P(Y = 0 \mid X = x) = \frac{P(X = x \mid Y = 0)P(Y = 0)}{P(X = x)} \tag{3}$$

$$P(Y = 1 \mid X = x) = \frac{P(X = x \mid Y = 1)P(Y = 1)}{P(X = x)} \tag{4}$$

We argue that there is a cat in a given image, if

$$P(Y = 1 \mid X = x) \geq P(Y = 0 \mid X = x);$$

otherwise, there is not a cat. Fortunately, when comparing $P(Y = 0 \mid X = x)$ and $P(Y = 1 \mid X = x)$, we do not have to calculate $P(X = x)$, since both of them share the same denominator $P(X = x)$. Therefore, to perform Bayesian interference, the parameters we have to compute are only $P(X = x \mid Y = y)$ and $P(Y = y)$.

Recalling that, in linear regression and logistic regression, we use hypothesis function $y = h_\theta(x)$ to model the relationship between feature vector x and label y, while we now rely on Byes' theorem to characterize the relationship through parameters $\theta = \{P(X = x \mid Y = y), P(Y = y)\}_{x,y}$.

2 Gaussian Discriminant Analysis

In *Gaussian Discriminate Analysis* (GDA) model, we have the following assumptions:

- **A1**: $Y \sim \text{Bernoulli}(\psi)$: Y follows a Bernoulli distribution parameterized by ψ, and we thus have $\mathrm{P}(Y = 1) = \psi$ and $\mathrm{P}(Y = 0) = 1 - \psi$. we then define the corresponding *probability mass function* (PMF) as

$$p_Y(y; \psi) = \mathrm{P}(Y = y) = \psi^y (1 - \psi)^{1-y} \tag{5}$$

- **A2**: $X \mid Y = 0 \sim \mathcal{N}(\mu_0, \Sigma)$: The conditional probability of *continuous* random variable X given $Y = 0$ is a Gaussian distribution parameterized by μ_0 and Σ, such that the corresponding *probability density function* (PDF) is defined as

$$p_{X|Y}(x \mid 0) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu_0)^T \Sigma^{-1}(x - \mu_0)\right) \tag{6}$$

- **A3**: $X \mid Y = 1 \sim \mathcal{N}(\mu_1, \Sigma)$: The conditional probability of *continuous* random variable X given $Y = 1$ is a Gaussian distribution parameterized by μ_1 and Σ, such that the corresponding PDF is given by

$$p_{X|Y}(x \mid 1) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu_1)^T \Sigma^{-1}(x - \mu_1)\right) \tag{7}$$

Given m sample data $\{(x^{(i)}, y^{(i)})\}_{i=1,\cdots,m}$, the log-likelihood is defined as

$$\begin{aligned}
\ell(\psi, \mu_0, \mu_1, \Sigma) &= \log \prod_{i=1}^{m} p_{X,Y}(x^{(i)}, y^{(i)}; \psi, \mu_0, \mu_1, \Sigma) \\
&= \log \prod_{i=1}^{m} p_{X|Y}(x^{(i)} \mid y^{(i)}; \mu_0, \mu_1, \Sigma) p_Y(y^{(i)}; \psi) \\
&= \sum_{i=1}^{m} \log p_{X|Y}(x^{(i)} \mid y^{(i)}; \mu_0, \mu_1, \Sigma) + \sum_{i=1}^{m} \log p_Y(y^{(i)}; \psi) \quad (8)
\end{aligned}$$

where ψ, μ_0, and σ are parameters. Substituting Eq. (5)$\sim$(7) into Eq. (8) gives us a full expression of $\ell(\psi, \mu_0, \mu_1, \Sigma)$

$$\begin{aligned}
&\ell(\psi, \mu_0, \mu_1, \Sigma) \\
= \quad & \sum_{i=1}^{m} \log p_{X|Y}(x^{(i)} \mid y^{(i)}; \mu_0, \mu_1, \Sigma) + \sum_{i=1}^{m} \log p_Y(y^{(i)}; \psi) \\
= \quad & \sum_{i:y^{(i)}=0} \log \left[\frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_0)^T \Sigma^{-1} (x-\mu_0) \right) \right] \\
& + \sum_{i:y^{(i)}=1} \log \left[\frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_1)^T \Sigma^{-1} (x-\mu_1) \right) \right] \\
& + \sum_{i=1}^{m} \log \psi^{y^{(i)}} (1-\psi)^{y^{(i)}}
\end{aligned}$$

We then maximize the log-likelihood function $\ell(\psi, \mu_0, \mu_1, \Sigma)$ so as to get the optimal values for ψ, μ_0, and σ, such that the resulting GDA model can best fit the given training data. In particular, we let

$$\begin{aligned}
\nabla_{\mu_0} \ell(\psi, \mu_0, \mu_1, \Sigma) = 0 \\
\nabla_{\mu_1} \ell(\psi, \mu_0, \mu_1, \Sigma) = 0 \\
\nabla_{\Sigma} \ell(\psi, \mu_0, \mu_1, \Sigma) = 0
\end{aligned}$$

A careful derivative gives us

$$\begin{aligned}
\psi &= \frac{1}{m} \sum_{i=1}^{m} \mathbf{1}\{y^{(i)} = 1\} \\
\mu_0 &= \sum_{i=1}^{m} \mathbf{1}\{y^{(i)} = 0\} x^{(i)} / \sum_{i=1}^{m} \mathbf{1}\{y^{(i)} = 0\} \\
\mu_1 &= \sum_{i=1}^{m} \mathbf{1}\{y^{(i)} = 1\} x^{(i)} / \sum_{i=1}^{m} \mathbf{1}\{y^{(i)} = 1\} \\
\Sigma &= \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \mu_{y^{(i)}})(x^{(i)} - \mu_{y^{(i)}})^T
\end{aligned}$$

Now we can use the above results to calculate the expression of $p_Y(y)$, $p_{X|Y}(x \mid 0)$, and $p_{X|Y}(x \mid 1)$ according to our assumptions (5)∼(7), and make predictions according to Bayes' theorem (see Eq. (2)). Specifically, given a test data featured by $\tilde{x}$, we compare

$$P(Y = \tilde{y} \mid X = \tilde{x}) = p_{Y|X}(\tilde{y} \mid \tilde{x}) = \frac{p(\tilde{x} \mid \tilde{y}) p(\tilde{y})}{p(\tilde{x})}$$

where $\tilde{y} = 0, 1$.

3 Gaussian Discriminant Analysis and Logistic Regression

By far, we introduce two classification algorithms, *Logistic Regression* (LR) and GDA. We now dive into investigating the relationship between them. Given a test data sample x, we can calculate $p(y=1 \mid x)$ as follows

$$\begin{aligned} p_{Y|X}(1 \mid x) &= \frac{p_{X|Y}(x \mid 1)p_Y(1)}{p_X(x)} \\ &= \frac{p_{X|Y}(x \mid 1)p_Y(1)}{p_{X|Y}(x \mid 1)p_Y(1) + p_{X|Y}(x \mid 0)p_Y(0)} \\ &= \frac{1}{1 + \frac{p_{X|Y}(x|0)p_Y(0)}{p_{X|Y}(x|1)p_Y(1)}} \end{aligned} \tag{9}$$

According to our assumptions (5)∼(7), we have

$$\begin{aligned} & \frac{p_{X|Y}(x \mid 0)p_Y(0)}{p_{X|Y}(x \mid 1)p_Y(1)} \\ = \; & \exp\left(-\frac{1}{2}(x-\mu_0)^T\Sigma^{-1}(x-\mu_0) + \frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)\right) \cdot \frac{1-\psi}{\psi} \\ = \; & \exp\left((\mu_0-\mu_1)^T\Sigma^{-1}x + \frac{1}{2}\left(\mu_1^T\Sigma^{-1}\mu_1 - \mu_0^T\Sigma^{-1}\mu_0\right)\right) \cdot \exp\left(\log\left(\frac{1-\psi}{\psi}\right)\right) \\ = \; & \exp\left((\mu_0-\mu_1)^T\Sigma^{-1}x + \frac{1}{2}\left(\mu_1^T\Sigma^{-1}\mu_1 - \mu_0^T\Sigma^{-1}\mu_0\right) + \log\left(\frac{1-\psi}{\psi}\right)\right) \end{aligned}$$

If we assume

$$\begin{aligned} x &:= \begin{bmatrix} x \\ 1 \end{bmatrix} \\ \theta &= \begin{bmatrix} (\mu_0-\mu_1)^T\Sigma^{-1} \\ \frac{1}{2}\left(\mu_1^T\Sigma^{-1}\mu_1 - \mu_0^T\Sigma^{-1}\mu_0\right) + \log\left(\frac{1-\psi}{\psi}\right) \end{bmatrix} \end{aligned}$$

we have

$$\begin{aligned} & \frac{p_{X|Y}(x \mid 0)p_Y(0)}{p_{X|Y}(x \mid 1)p_Y(1)} \\ = \; & \exp\left((\mu_0-\mu_1)^T\Sigma^{-1}x + \frac{1}{2}\left(\mu_1^T\Sigma^{-1}\mu_1 - \mu_0^T\Sigma^{-1}\mu_0\right) + \log\left(\frac{1-\psi}{\psi}\right)\right) \\ = \; & \exp\left(\theta^T x\right) \end{aligned} \tag{10}$$

By substituting (10) into Eq. (9), we finally represent $p(y=1 \mid x)$ as

$$p_{Y|X}(1 \mid x) = \frac{1}{1+\exp(\theta^T x)} \tag{11}$$

Similarly, we have

$$\begin{aligned}
& p_{Y|X}(0 \mid x) \\
= & \frac{p_{X|Y}(x \mid 0)p_Y(0)}{p_X(x)} \\
= & \frac{p_{X|Y}(x \mid 0)p_Y(0)}{p_{X|Y}(x \mid 1)p_Y(1) + p_{X|Y}(x \mid 0)p_Y(0)} \\
= & \frac{1}{1 + \frac{p_{X|Y}(x|1)p_Y(1)}{p_{X|Y}(x|0)p_Y(0)}} \\
= & \frac{1}{1 + \exp\left((\mu_1 - \mu_0)^T \Sigma^{-1} x + \frac{\mu_0^T \Sigma^{-1}\mu_0 - \mu_1^T \Sigma^{-1}\mu_1}{2} + \log\left(\frac{\psi}{1-\psi}\right)\right)}
\end{aligned}$$

Therefore, we conclude that GDA model can be reformulated as logistic regression. But the question is, which one is better? GDA makes stronger modeling assumptions, and is more data efficient (i.e., requires less training data to learn "well") when the modeling assumptions are correct or at least approximately correct, while LR makes weaker assumptions, and is significantly more robust deviations from modeling assumptions. Hence, when the data is indeed non-Gaussian, then in the limit of large datasets, logistic regression will almost always do better than GDA. In practice, logistic regression is used more often than GDA

4 Naive Bayes

4.1 Assumption

Again, we assume that the m training data are denoted by $\{x^{(i)}, y^{(i)}\}_{i=1,\cdots,m}$, where $x^{(i)}$ is a n-dimensional vector with each component $x_j^{(i)} \in \{0,1\}$ ($j = 1,\cdots,n$), and $y^{(i)} \in \{1,\cdots,k\}$. For brevity, we use $[k]$ to denote set $\{1,2,\cdots k\}$. Therefore, we have $i \in [m]$, $j \in [n]$ and $y \in [k]$. In *Naive Bayes* (NB) model, the feature and label can be represented by random variables $\{X_j\}_{j\in[n]}$ and Y, respectively. Furthermore, for $\forall j \neq j'$, Naive Bayes assumes X_j and $X_{j'}$ are conditionally independent given Y. Therefore, we have

$$\begin{aligned}
& P(X_1 = x_1, X_2 = x_2, \cdots, X_n = x_n \mid Y = y) \\
= & \prod_{j=1}^{n} P(X_j = x_j \mid X_1 = x_1, X_2 = x_2, \cdots, X_{j-1} = x_{j-1}, Y = y) \\
= & \prod_{j=1}^{n} P(X_j = x_j \mid Y = y)
\end{aligned}$$

Moreover, $P(Y = y, X_1 = x_1, \cdots, X_n = x_n)$ can be calculated as

$$\begin{aligned}
& P(Y = y, X_1 = x_1, \cdots, X_n = x_n) \\
= & P(X_1 = x_1, \cdots, X_n = x_n \mid Y = y)P(Y = y) \\
= & P(Y = y) \prod_{j=1}^{n} P(X_j = x_j \mid Y = y)
\end{aligned}$$

By now, we have two set of parameters: i) $P(Y = y) = p_Y(y)$ for $\forall y \in [k]$, and ii) $P(X_j = x_j \mid Y = y) = p_{X_j|Y}(x_j \mid y)$ for $\forall x_j \in \{0,1\}$ where $j \in [n]$, $\forall y \in [k]$. For brevity, we drop the subscripts without inducing any ambiguity,

$$\begin{aligned} &p(y) := p_Y(y) \\ &p_j(x \mid y) := p_{X_j|Y}(x_j \mid y) \end{aligned}$$

In another word, $p(y)$ and $p_j(x \mid y)$ are the PMF and the conditional PMF of Y and $X_j \mid Y$, respectively. More specifically, $p(y)$ denotes the prior probability of $Y = y$, while $p_j(x_j \mid y)$ denotes the posterior probability of $X_j = x_j$ given $Y = y$.

4.2 Problem Formulation

Given a set of m training data $\{x^{(i)}, y^{(i)}\}_{i\in[m]}$, the log-likelihood function can be defined by

$$\begin{aligned} \ell(\Omega) &= \log \prod_{i=1}^{m} p(x^{(i)}, y^{(i)}) \\ &= \sum_{i=1}^{m} \log p(x^{(i)}, y^{(i)}) \\ &= \sum_{i=1}^{m} \log \left(p(y^{(i)}) \prod_{j=1}^{n} p_j(x_j^{(i)} \mid y^{(i)}) \right) \\ &= \sum_{i=1}^{m} \log p(y^{(i)}) + \sum_{i=1}^{m} \sum_{j=1}^{n} \log p_j(x_j^{(i)} \mid y^{(i)}) \qquad (12) \end{aligned}$$

where we use Ω to represent the set of parameters. Again, we would like to maximize the above objective function with respect to $\{p(y)\}_{y\in[k]}$ and $\{p_j(x \mid y)\}_{j\in[n],x\in\{0,1\},y\in[k]}$. Mathematically, our problem can be formulated as

$$\max \quad \ell(\Omega) = \sum_{i=1}^{m} \log p(y^{(i)}) + \sum_{i=1}^{m} \sum_{j=1}^{n} \log p_j(x_j^{(i)} \mid y^{(i)}) \qquad (13)$$

$$s.t. \quad \sum_{y=1}^{k} p(y) = 1 \qquad (14)$$

$$\sum_{x\in\{0,1\}} p_j(x \mid y) = 1, \ \forall y \in [k], j \in [n] \qquad (15)$$

$$p(y) \geq 0, \ \forall y \in [k] \qquad (16)$$

$$p_j(x \mid y) \geq 0, \ \forall y \in [k], j \in [n], x \in \{0,1\} \qquad (17)$$

4.3 Solutions to Naive Bayes

We calculate the optimal value of $p(y)$ for $\forall y \in [k]$, by applying Lagrange multiplier method. Let α and $\beta_j(y)$ be the Lagrange multipliers associate with constraint (14) and (15), respectively. The Lagrange function is defined as

$$\begin{aligned} L(\Omega, \alpha, \beta) &= \sum_{i=1}^{m} \log p(y^{(i)}) + \sum_{i=1}^{m} \sum_{j=1}^{n} \log p_j(x_j^{(i)} \mid y^{(i)}) \\ &\quad -\alpha \left(\sum_{y=1}^{k} p(y) - 1 \right) \\ &\quad -\sum_{y=1}^{k} \sum_{j=1}^{n} \beta_j(y) \left(\sum_{x \in \{0,1\}} p_j(x \mid y) - 1 \right) \end{aligned} \tag{18}$$

where $\beta = \{\beta_j(y)\}_{j \in [n], y \in [k]}$. According to the theory of Lagrange multiplier, if there exits $\Omega^* = \{p^*(y), p_j^*(x \mid y)\}_{j \in [n], x \in \{0,1\}, y \in [k]}$ such that $\ell(\Omega^*)$ is a maximum of $\ell(\Omega)$, there exists α^* and $\beta^* = \{\beta_j^*(y)\}_{j \in [n], y \in [k]}$ such that $(\Omega^*, \alpha^*, \beta^*)$ is a *stationary point* for the Lagrange function. To this end, we first calculate the partial derivative of $L(\Omega, \alpha, \beta)$ with respect to Ω, and let them be zeros.

Since

$$\frac{\partial}{\partial p(y)} L(\Omega, \alpha, \beta) = \sum_{i: y^{(i)} = y} \frac{\partial}{\partial p(y)} \log p(y) - \alpha = \frac{count(y)}{p(y)} - \alpha = 0$$

where

$$count(y) = \sum_{i=1}^{m} \mathbf{1}(y^{(i)} = y), \ \forall y \in [k]$$

denotes the number of training data whose label is y, we have

$$p(y) = \frac{count(y)}{\alpha} \tag{19}$$

Substituting the above equation into (14), we get

$$\sum_{y=1}^{k} p(y) = \sum_{y=1}^{k} \frac{count(y)}{\alpha} = \frac{m}{\alpha} = 1$$

hence, $\alpha = m$. According to (19),

$$p(y) = \frac{count(y)}{\alpha} = \frac{\sum_{i=1}^{m} \mathbf{1}(y^{(i)} = y)}{m} \tag{20}$$

Similarly, by letting

$$\frac{\partial}{\partial p_j(x \mid y)} L(\Omega, \alpha, \beta) = 0$$

we get

$$\frac{count_j(x \mid y)}{p_j(x \mid y)} - \beta_j(y) = 0 \tag{21}$$

where

$$count_j(x \mid y) = \sum_{i=1}^{m} \mathbf{1}(y^{(i)} = y \wedge x_j^{(i)} = x), \ \forall y \in [k], \ \forall x \in \{0,1\}$$

denotes the number of training data with its j-th feature being x and label being y, and hence, $p_j(x \mid y)$ can be written as

$$p_j(x \mid y) = \frac{count_j(x \mid y)}{\beta_j(y)} \tag{22}$$

Substituting the above equation into (15), we get

$$\beta_j(y) = count(y)$$

for $\forall j \in [n], y \in [k]$. Therefore, according to Eq. (22)

$$p_j(x \mid y) = \frac{count_j(x \mid y)}{\beta_j(y)} = \frac{\sum_{i=1}^m \mathbf{1}(y^{(i)} = y \wedge x_j^{(i)} = x)}{\sum_{i=1}^m \mathbf{1}(y^{(i)} = y)} \tag{23}$$

Remark: *We assume binary features ($X_j \in \{0,1\}$ for $\forall j \in [n]$) in the above discussion. What if $X_j \in \{1, 2, \cdots, v\}$? Can we get similar results? Check it by yourselves!*

4.4 Laplace Smoothing

Consider the following special case, in the give finite training data, $\bar{x}$ never happens for some j, such that you cannot find any training data without its j-th feature being $\bar{x}$. In this case, when calculating $p_j(\bar{x} \mid y)$, one trivial choice is to let it being zero. It follows that, given a test data $x = (x_1, \cdots, x_{\bar{j}} = \bar{x}, \cdots, x_n)$ where the $\bar{j}$-th feature is $\bar{x}$, we have

$$\begin{aligned} p(y \mid x) &= p(y) \prod_{j=1}^n p_j(x_j \mid y) \\ &= p(y) p_1(x_1 \mid y) p_1(x_2 \mid y) \cdots p_{\bar{j}}(\bar{x} \mid y) \cdots p_n(x_n \mid y) \\ &= 0 \end{aligned}$$

for $\forall y$. It is shown that, even the remaining features all have very "strong" conditional probabilities, $p(y \mid x)$ is forcibly set to be zero due to only one feature value that does not appear in the finite training data. Apparently, this is quite unreasonable! Similarly, when some of the label values (e.g., $\bar{y}$) doe not appear in the given training data, we have

$$p(\bar{y}) = \frac{\sum_{i=1}^m \mathbf{1}(y^{(i)} = \bar{y})}{m} = 0$$

, such that for $\forall x$, we have $p(y \mid x) = 0$.

One method to address the above problem is Laplace smoothing. In particular, we set

$$p(y) = \frac{\sum_{i=1}^m \mathbf{1}(y^{(i)} = y) + 1}{m + k}$$

$$p_j(x \mid y) = \frac{\sum_{i=1}^m \mathbf{1}(y^{(i)} = y \wedge x_j^{(i)} = x) + 1}{\sum_{i=1}^m \mathbf{1}(y^{(i)} = y) + v_j}$$

where v_j is the number of possible values of the j-th feature. In our case where $x_j \in \{0, 1\}$ for $\forall j \in [n]$, we have $v_j = 2$ for $\forall j$. Note that, $p(y)$ satisfies the following two conditions

$$\begin{aligned}
&p(y) \geq 0,\ \forall y \in [k] \\
&\sum_{y=1}^{k} p(y) = \sum_{y=1}^{k} \frac{\sum_{i=1}^{m} \mathbf{1}(y^{(i)} = y) + 1}{m + k} = \frac{\sum_{y=1}^{k} \sum_{i=1}^{m} \mathbf{1}(y^{(i)} = y) + k}{m + k} = 1
\end{aligned}$$

Similarly,

$$\begin{aligned}
&p_j(x \mid y) \geq 0,\ \forall j \in [n],\ x \in \{0, 1\},\ y \in [k] \\
&\sum_{x \in \{0,1\}} p_j(x \mid y) = 1,\ \forall j \in [n],\ y \in [k]
\end{aligned}$$

5 Naive Bayes for Multinomial Distribution

In this model, a training sample may involves a different number of features. We assume that the i-th training sample $x^{(i)}$ has n_i features. For $\forall i \in [m]$, $x^{(i)}$ has each of its features drawn from a sample space $[v] = \{1, 2, \cdots, v\}$ identically and independently. Let X_j and Y be the random variables representing the j-th feature and the label. We define

$$p(t \mid y) = P(X_j = t \mid Y = y)$$

for some j. In another word, $p(t \mid y)$ is the conditional probability that $t \in [v]$ occurs once (at some position) in the feature vector given that the data sample is labeled by y. Also, $p(t \mid y)$ should respect the following conditions: i) $p(t \mid y) \geq 0$, and ii) $\sum_{t=1}^{v} p(t \mid y) = 1$. We also define

$$p(y) = P(Y = y)$$

for $\forall y \in [k]$. We denote by Ω the set of parameters, i.e., $\Omega = \{p(y), p(t \mid y)\}_{t \in [v], y \in [k]}$

Given a set of m training data $\{(x^{(i)}, y^{(i)})\}_{i \in [m]}$, the log-likelihood function can be defined by

$$
\begin{aligned}
\ell(\Omega) &= \log \prod_{i=1}^{m} p(x^{(i)}, y^{(i)}) \\
&= \log \prod_{i=1}^{m} p(x^{(i)} \mid y^{(i)}) p(y^{(i)}) \\
&= \log \prod_{i=1}^{m} \sum_{y=1}^{k} \mathbf{1}(y^{(i)} = y) p(x^{(i)} \mid y) p(y) \\
&= \sum_{i=1}^{m} \log \left(\sum_{y=1}^{k} \mathbf{1}(y^{(i)} = y) \left(p(x^{(i)} \mid y) p(y) \right) \right) \\
&= \sum_{i=1}^{m} \sum_{y=1}^{k} \mathbf{1}(y^{(i)} = y) \log \left(p(x^{(i)} \mid y) p(y) \right) \\
&= \sum_{i=1}^{m} \sum_{y=1}^{k} \mathbf{1}(y^{(i)} = y) \log \left(p(y) \prod_{j=1}^{n_i} p(x_j^{(i)} \mid y) \right) \\
&= \sum_{i=1}^{m} \sum_{y=1}^{k} \mathbf{1}(y^{(i)} = y) \log \left(p(y) \prod_{t=1}^{v} p(t \mid y)^{count^{(i)}(t)} \right) \\
&= \sum_{i=1}^{m} \sum_{y=1}^{k} \mathbf{1}(y^{(i)} = y) \left(\log p(y) + \sum_{t=1}^{v} count^{(i)}(t) \log p(t \mid y) \right)
\end{aligned}
$$

where $count^{(i)}(t) = \sum_{j=1}^{n_i} \mathbf{1}(x_j^{(i)} = t)$ is the number of features in $x^{(i)}$ whose values are t (i.e., how many time t occurs in $x^{(i)}$).

By now, we formulate our NB model for multinomial distribution as follows

$$
\begin{aligned}
\max \quad & \ell(\Omega) = \sum_{i=1}^{m} \sum_{y=1}^{k} \mathbf{1}(y^{(i)} = y) \left(\log p(y) + \sum_{t=1}^{v} count^{(i)}(t) \log p(t \mid y) \right) \\
s.t. \quad & p(y) \geq 0, \ \forall y \in [k] \\
& p(t \mid y) \geq 0, \ \forall t \in [v] \ \forall y \in [k] \\
& \sum_{y=1}^{k} p(y) = 1, \\
& \sum_{t=1}^{v} p(t \mid y) = 1, \ \forall y \in [k]
\end{aligned}
$$

Applying Lagrange multiplier, we get the following optimal solution to the above optimization problem

$$
\begin{aligned}
p(t \mid y) &= \frac{\sum_{i=1}^{m} \mathbf{1}(y^{(i)} = y) count^{(i)}(t)}{\sum_{i=1}^{m} \mathbf{1}(y^{(i)} = y) \sum_{t=1}^{v} count^{(i)}(t)} \\
p(y) &= \frac{\sum_{i=1}^{m} \mathbf{1}(y^{(i)} = y)}{m}
\end{aligned}
$$

6 Expectation-Maximization Algorithm

We hereby look at *Expectation-Maximization* (EM) algorithm.

6.1 Convex Sets and Convex Functions

A set C is *convex* if the line segment between any two points in C lies in C, i.e., for $\forall x_1, x_2 \in C$ and $\forall \theta$ with $0 \leq \theta \leq 1$, we have

$$\theta x_1 + (1-\theta)x_2 \in C$$

A function $f : \mathbb{R}^n \to R$ is *convex*, if $\mathbf{dom} f$ is a convex set and if for all $x, y \in \mathbf{dom} f$ and λ with $0 \leq \lambda \leq 1$, we have

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda)f(y)$$

If

$$f(\lambda x + (1-\lambda)y) \geq \lambda f(x) + (1-\lambda)f(y)$$

f is said to be *concave.* One typical example of concave function is log.

6.2 Jensen's Inequality

Theorem 1. *Jensen's Inequality Let $f(x)$ be a convex function defined on an interval $\mathcal{I}$. If $x_1, x_2, \cdots, x_N \in \mathcal{I}$ and $\lambda_1, \lambda_2, \cdots, \lambda_N \geq 0$ with $\sum_{i=1}^{N} \lambda_i = 1$*

$$f(\sum_{i=1}^{N} \lambda_i x_i) \leq \sum_{i=1}^{N} \lambda_i f(x_i) \tag{24}$$

Proof. When $N = 1$, the result is trivial. When $N = 2$, we have

$$f(\lambda_1 x_1 + \lambda_2 x_2) \leq \lambda_1 f(x_1) + \lambda_2 f(x_2)$$

due to convexity of $f(x)$. When $N \geq 3$, the proof is by induction. We assume that, the Jensen's inequality holds when $N = k-1$, i.e.

$$f(\sum_{i=1}^{k-1} \lambda_i x_i) \leq \sum_{i=1}^{k-1} \lambda_i f(x_i)$$

We then prove that, the Jensen's inequality still holds for $N = k$. In particular,

$$
\begin{aligned}
f(\sum_{i=1}^{k}\lambda_i x_i) &= f(\sum_{i=1}^{k-1}\lambda_i x_i + \lambda_k x_k) \\
&= f((1-\lambda_k)\sum_{i=1}^{k-1}\frac{\lambda_i}{1-\lambda_k}x_i + \lambda_k x_k) \\
&\leq (1-\lambda_k)f(\sum_{i=1}^{k-1}\frac{\lambda_i}{1-\lambda_k}x_i) + \lambda_k f(x_k) \\
&\leq (1-\lambda_k)\sum_{i=1}^{k-1}\frac{\lambda_i}{1-\lambda_k}f(x_i) + \lambda_k f(x_k) \\
&= \sum_{i=1}^{k-1}\lambda_i f(x_i) + \lambda_k f(x_k) \\
&= \sum_{i=1}^{k}\lambda_i f(x_i)
\end{aligned}
$$

□

The inequality can be generalized to infinite sums, integrals, and expected values. For example, assuming X is a random variable, we have

$$f[E(X)] \leq E[f(X)] \tag{25}$$

The equality holds if X is a constant.

When f is a concave function, the Jensen's inequality can be re-written as

$$f(\sum_{i=1}^{N}\lambda_i x_i) \geq \sum_{i=1}^{N}\lambda_i f(x_i) \tag{26}$$

while its probabilistic form becomes

$$f(E[X]) \geq E[f(X)] \tag{27}$$

For example, when $f(x) = \log x$, we have

$$\log(\sum_{i=1}^{N}\lambda_i x_i) \geq \sum_{i=1}^{N}\lambda_i \log(x_i) \tag{28}$$

$$\log(E[X]) \geq E[\log(X)] \tag{29}$$

6.3 EM Algorithm

Let $\{x^{(1)}, x^{(2)}, \cdots, x^{(m)}\}$ be a set of training data without labels. The log-likelihood function can be defined by

$$
\begin{aligned}
\ell(\theta) &= \log\prod_{i=1}^{m}p(x^{(i)};\theta) \\
&= \sum_{i=1}^{m}\log\sum_{z^{(i)}\in\Omega}p(x^{(i)}, z^{(i)};\theta) \qquad (30)
\end{aligned}
$$

where θ denotes the full set of unknown parameters, while $z^{(i)} \in \Omega$ is so-called "*latent variable*" with Ω being the set of its all possible values. In fact, $z^{(i)}$ is an analogue of label, which we "guess" for each training data. Specifically, supposing $X^{(i)}$ and $Z^{(i)}$ are the random variables representing the features and the label of the i-th data sample, $p(x^{(i)};\theta) = P(X^{(i)} = x^{(i)})$ is the marginal PMF of $X^{(i)}$, while $p(x^{(i)}, z^{(i)};\theta) = P(X^{(i)} = x^{(i)}, Z^{(i)} = z^{(i)})$ is the joint PMF of $(X^{(i)}, Z^{(i)})$.

To maximize the above log-likelihood function $\ell(\theta)$, the basic idea of the EM algorithm is to repeatedly construct a lower-bound on ℓ (E-step), and then optimize the lower-bound (M-step). We assume that the i-th training sample has its label following a probability distribution Q_i. In another word, $Q_i(z^{(i)})$ represents the probability that the i-th training sample has its label being $z^{(i)} \in \Omega$ (i.e., $Q_i(z^{(i)}) = P(Z^{(i)} = z^{(i)})$). $Q_i(z^{(i)})$ should satisfy the following conditions:

$$\sum_{z^{(i)}\in\Omega} Q_i(z^{(i)}) = 1,$$
$$Q_i(z^{(i)}) \geq 0, \ \forall z^{(i)} \in \Omega$$

Also, suppose $\phi(Z^{(i)})$ is a function of random variable $Z^{(i)}$. we then have

$$E(\phi(Z^{(i)})) = \sum_{z^{(i)}\in\Omega} Q(z^{(i)})\phi(z^{(i)}) \tag{31}$$

We re-write the log-likelihood function as follows

$$\begin{aligned}
\ell(\theta) &= \sum_{i=1}^{m} \log \sum_{z^{(i)}\in\Omega} p(x^{(i)}, z^{(i)};\theta) \\
&= \sum_{i=1}^{m} \log \sum_{z^{(i)}\in\Omega} Q_i(z^{(i)}) \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})} \\
&= \sum_{i=1}^{m} \log \sum_{z^{(i)}\in\Omega} Q_i(z^{(i)})\phi(z^{(i)}) \\
&= \sum_{i=1}^{m} \log E_{Z^{(i)}\sim Q_i}\left[\phi(Z^{(i)})\right] \\
&\geq \sum_{i=1}^{m} E_{Z^{(i)}\sim Q_i}\left[\log \phi(Z^{(i)})\right] \\
&= \sum_{i=1}^{m} \sum_{z^{(i)}\in\Omega} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})}
\end{aligned} \tag{32}$$

In particular, the first two equality is very trivial, and we have the third one by assuming $\phi : \Omega \rightarrow \mathbb{R}$ is a function of $Z^{(i)}$ such that

$$\phi(z^{(i)}) = \frac{p(x^{(i)}, z^{(i)};\theta)}{Q_i(z^{(i)})}.$$

The forth equality holds according to Eq. (31). By applying Jensen's inequality to (concave) log function, we have the inequality in the fifth line. The sixth equality also comes from Eq. (31)

To tighten the lower bound, we should let the equality (in the forth line) hold. According to Jensen's inequality, the equality holds if $p(x^{(i)}, z^{(i)}; \theta)/Q_i(z^{(i)})$ is a constant. Assume

$$\frac{p(x^{(i)}, z^{(i)}; \theta)}{Q_i(z^{(i)})} = c \tag{33}$$

where c is a constant. Since $\sum_{z^{(i)} \in \Omega} Q_i(z^{(i)}) = 1$, we have

$$\sum_{z^{(i)} \in \Omega} p(x^{(i)}, z^{(i)}; \theta) = c \sum_{z^{(i)} \in \Omega} Q_i(z) = c \tag{34}$$

Then, $Q_i(z^{(i)})$ can be re-written as

$$\begin{aligned} Q_i(z^{(i)}) &= \frac{p(x^{(i)}, z^{(i)}; \theta)}{c} \\ &= \frac{p(x^{(i)}, z^{(i)}; \theta)}{\sum_{z^{(i)} \in \Omega} p(x^{(i)}, z^{(i)}; \theta)} \\ &= \frac{p(x^{(i)}, z^{(i)}; \theta)}{p(x^{(i)}; \theta)} \\ &= p(z^{(i)} \mid x^{(i)}; \theta) \end{aligned}$$

In another world, $Q_i(z^{(i)})$ is the conditional probability that the i-th data sample is labeled by $z^{(i)}$ given it is featured by $x^{(i)}$.

In the EM algorithm, we repeat the following step until convergence

- (E-step) For each i, set

$$Q_i(z^{(i)}) = p(z^{(i)} \mid x^{(i)}; \theta)$$

- (M-step) set

$$\theta := \arg\max_{\theta} \sum_i \sum_{z^{(i)} \in \Omega} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta)}{Q_i(z^{(i)})}$$

Theorem 2. *The EM algorithm is converged.*

Proof. Suppose $\theta^{[t]}$ and $\theta^{[t+1]}$ be the (input) parameters for the t-th and the $(t+1)$-th iterations, respectively. The convergence of the EM algorithm can be proved by showing the EM algorithm monotonically increases the log-likelihood function, i.e., $\ell(\theta^{[t+1]}) \geq \ell(\theta^{[t]})$.

In the t-th iteration, we start with calculating $Q_i^{[t]}(z^{(i)})$ according to $\theta^{[t]}$

$$Q_i^{[t]}(z^{(i)}) = p(z^{(i)} \mid x^{(i)}; \theta^{[t]})$$

such that Jensen's inequality holds with equality, and hence

$$\ell(\theta^{[t]}) = \sum_{i=1}^{m} \sum_{z^{(i)} \in \Omega} Q_i^{[t]}(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta^{[t]})}{Q_i^{[t]}(z^{(i)})}$$

We then calculate $\theta^{[t+1]}$by maximizing the right hand side of the above equation over θ; therefore, we have

$$\begin{aligned}
\ell(\theta^{[t+1]}) &\geq \sum_{i=1}^{m} \sum_{z^{(i)}\in\Omega} Q_i^{[t]}(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta^{[t+1]})}{Q_i^{[t]}(z^{(i)})} \\
&\geq \sum_{i=1}^{m} \sum_{z^{(i)}\in\Omega} Q_i^{[t]}(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta^{[t]})}{Q_i^{[t]}(z^{(i)})} \\
&= \ell(\theta^{[t]})
\end{aligned}$$

The first inequality comes from the fact that

$$\ell(\theta) \geq \sum_{i=1}^{m} \sum_{z^{(i)}\in\Omega} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta)}{Q_i(z^{(i)})}$$

holds for $\forall\theta$, and in particular holds for $Q_i = Q_i^{[t]}$, according to Eq. (32). We have the inequality in the second line, because $\theta^{[t+1]}$ is calculated by

$$\theta^{[t+1]} = \arg\max_{\theta} \sum_{i} \sum_{z^{(i)}\in\Omega} Q_i^{[t]}(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \theta)}{Q_i^{[t]}(z^{(i)})}$$

□

6.4 Applying EM Algorithm to Gaussian Discriminant Analysis

Again, assume $\{x^{(i)}\}_{i\in[m]}$ is the set of training data without labels and the latent variable is $z^{(i)}$. For each training data, the probability that it is labeled by $j \in [k]$ is denoted by ϕ_j. Obviously, we have $\sum_{j=1}^{k} \phi_j = 1$ and $\phi_j \geq 0$ for $\forall j \in [k]$. We also suppose that $x \mid z = j \sim \mathcal{N}(\mu_j, \Sigma_j)$, i.e., the conditional probability of observing feature x in a training sample given that its label is j follows a Gaussian distribution parametrized by μ_j and Σ_j

According to the rules of the EM algorithm, we have repeat the following steps until convergence

- (E-step) For each i, j, set

$$\omega_j^{(i)} = \frac{p(x^{(i)} \mid z^{(i)} = j; \mu, \Sigma)p(z^{(i)} = j; \phi)}{\sum_{l=1}^{k} p(x^{(i)} \mid z^{(i)} = l; \mu, \Sigma)p(z^{(i)} = l; \phi)}$$

- (M-step) Update the parameters

$$\begin{aligned}
\phi_j &= \frac{1}{m}\sum_{i=1}^{m} \omega_j^{(i)} \\
\mu_j &= \frac{\sum_{i=1}^{m} \omega_j^{(i)} x^{(i)}}{\sum_{i=1}^{m} \omega_j^{(i)}} \\
\Sigma_j &= \frac{\sum_{i=1}^{m} \omega_j^{(i)} (x^{(i)} - \mu_j)(x^{(i)} - \mu_j)^T}{\sum_{i=1}^{m} \omega_j^{(i)}}
\end{aligned}$$

More details are given in the following.

6.4.1 E-Step in Applying EM to GDA

In each iteration, we first calculating $Q_i(z^{(i)} = j)$ for $\forall i, j$. In particular,

$$\begin{aligned} w_j^{(i)} &= Q_i(z^{(i)} = j) \\ &= p(z^{(i)} = j \mid x^{(i)}; \phi, \mu, \Sigma) \\ &= \frac{p(x^{(i)} \mid z^{(i)} = j; \mu, \Sigma) p(z^{(i)} = j; \phi)}{\sum_{l=1}^{k} p(x^{(i)} \mid z^{(i)} = l; \mu, \Sigma) p(z^{(i)} = l; \phi)} \end{aligned}$$

where

$$p(x^{(i)} \mid z^{(i)} = j; \mu_j, \Sigma_j) = \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x^{(i)} - \mu_j)^T \Sigma_j^{-1} (x^{(i)} - \mu_j)\right)$$

and

$$p(z^{(i)} = j; \phi) = \phi_j$$

6.4.2 M-Step in Applying EM to GDA

We then maximize

$$\begin{aligned} & \sum_{i=1}^{m} \sum_{z^{(i)}} Q_i(z^{(i)}) \log \frac{p(x^{(i)}, z^{(i)}; \phi, \mu, \Sigma)}{Q_i(z^{(i)})} \\ = \; & \sum_{i=1}^{m} \sum_{j=1}^{k} Q_i(z^{(i)} = j) \log \frac{p(x^{(i)} \mid z^{(i)} = j; \mu, \Sigma) p(z^{(i)} = j; \phi)}{Q_i(z^{(i)} = j)} \\ = \; & \sum_{i=1}^{m} \sum_{j=1}^{k} \omega_j^{(i)} \log \frac{\frac{1}{(2\pi)^{n/2}|\Sigma_j|^{1/2}} \exp\left(-\frac{1}{2}(x^{(i)} - \mu_j)^T \Sigma_j^{-1} (x^{(i)} - \mu_j)\right) \phi_j}{\omega_j^{(i)}} \\ = \; & -\sum_{i=1}^{m} \sum_{j=1}^{k} \omega_j^{(i)} \left[\log\left((2\pi)^{\frac{n}{2}} |\Sigma_j|^{\frac{1}{2}}\right) + \frac{1}{2}(x^{(i)} - \mu_j)^T \Sigma_j^{-1} (x^{(i)} - \mu_j)\right] \\ & + \sum_{i=1}^{m} \sum_{j=1}^{k} \omega_j^{(i)} \log \phi_j - \sum_{i=1}^{m} \sum_{j=1}^{k} \omega_j^{(i)} \log \omega_j^{(i)} \end{aligned}$$

over μ_j and Σ_j

For $\forall \mu_j$ $(j \in [k])$, we first calculate the corresponding partial derivative

$$\begin{aligned} & \nabla_{\mu_l} \sum_{i=1}^{m} \sum_{j=1}^{k} \omega_j^{(i)} \log \frac{\frac{1}{(2\pi)^{n/2}|\Sigma_j|^{1/2}} \exp\left(-\frac{1}{2}(x^{(j)} - \mu_j)^T \Sigma_j^{-1} (x^{(j)} - \mu_j)\right) \phi_j}{\omega_j^{(i)}} \\ = \; & \nabla_{\mu_l} \sum_{i=1}^{m} \sum_{j=1}^{k} \omega_j^{(i)} \frac{1}{2}(x^{(i)} - \mu_j)^T \Sigma_j^{-1} (x^{(i)} - \mu_j) \\ = \; & \frac{1}{2} \sum_{i=1}^{m} \omega_l^{(i)} \, \nabla_{\mu_l} \left(2\mu_l^T \Sigma_l^{-1} x^{(i)} - \mu_l^T \Sigma_l^{-1} \mu_l\right) \\ = \; & \sum_{i=1}^{m} \omega_l^{(i)} \left(\Sigma_l^{-1} x^{(i)} - \Sigma_l^{-1} \mu_l\right) \end{aligned}$$

and then let it be zero. Hence, we have

$$\mu_l = \frac{\sum_{i=1}^m \omega_l^{(i)} x^{(i)}}{\sum_{i=1}^m \omega_l^{(i)}}$$

To calculate ϕ, we have to resolve the following optimization problem

$$\begin{aligned} \max \quad & \sum_{i=1}^m \sum_{j=1}^k \omega_j^{(i)} \log \phi_j \\ \text{s.t.} \quad & \sum_{j=1}^k \phi_j = 1 \end{aligned}$$

Using Lagrange multiplier method, we have

$$\phi_j = \frac{1}{m} \sum_{i=1}^m \omega_j^{(i)}$$

To calculating Σ_j, we have to minimize

$$\begin{aligned} & \sum_{i=1}^m \sum_{j=1}^k \omega_j^{(i)} \left[\log \left((2\pi)^{\frac{n}{2}} |\Sigma_j|^{\frac{1}{2}} \right) + \frac{1}{2}(x^{(i)} - \mu_j)^T \Sigma_j^{-1} (x^{(i)} - \mu_j) \right] \\ = \quad & \frac{n \log 2\pi}{2} \sum_{i=1}^m \sum_{j=1}^k \omega_j^{(i)} + \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^k \omega_j^{(i)} \log |\Sigma_j| \\ & + \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^k \omega_j^{(i)} (x^{(i)} - \mu_j)^T \Sigma_j^{-1} (x^{(i)} - \mu_j) \end{aligned}$$

Letting its gradient with respect to Σ_j be zeros, we have

$$\Sigma_j = \frac{\sum_{i=1}^m \omega_j^{(i)} (x^{(i)} - \mu_j)(x^{(i)} - \mu_j)^T}{\sum_{i=1}^m \omega_j^{(i)}}$$

by applying

$$\begin{aligned} & \nabla_X \mathrm{tr}(AX^{-1}B) = -(X^{-1}BAX^{-1})^T \\ & \nabla_A |A| = |A|(A^{-1})^T \end{aligned}$$

6.5 Applying EM Algorithm to Naive Bayes

Let $y^{(i)}$ be the latent variable. For the i-th training sample, we assume $\delta(y \mid i)$ denotes conditional the probability of being labeled by y given feature vector $x^{(i)}$. Then, in EM algorithm, we repeat the following steps until convergence

- (E-step) For each $i = 1, \cdots, m$ and $y = 1, \cdots, k$ set

$$\delta(y \mid i) = p(y \mid x^{(i)}) = \frac{p(y) \prod_{j=1}^n p_j(x_j^{(i)} \mid y)}{\sum_{y'=1}^k p(y') \prod_{j=1}^n p_j(x_j^{(i)} \mid y')}$$

- (M-step) Update the parameters

$$p(y) = \frac{1}{m}\sum_{i=1}^{m}\delta(y \mid i) \tag{35}$$

$$p_j(x \mid y) = \frac{\delta(y \mid i)}{\sum_{i=1}^{m}\delta(y \mid i)} \tag{36}$$

We introduce the details as follows. We first calculate $\delta(y \mid i) = Q_i(y)$ for $\forall i \in [m], y \in [k]$ as

$$\begin{aligned}
\delta(y \mid i) &= p(y \mid x^{(i)}) \\
&= \frac{p(x^{(i)} \mid y)p(y)}{p(x^{(i)})} \\
&= \frac{p(y)\prod_{j=1}^{n} p_j(x_j^{(i)} \mid y)}{\sum_{y'=1}^{k} p(x^{(i)}, y')} \\
&= \frac{p(y)\prod_{j=1}^{n} p_j(x_j^{(i)} \mid y)}{\sum_{y'=1}^{k} p(x^{(i)} \mid y')p(y')} \\
&= \frac{p(y)\prod_{j=1}^{n} p_j(x_j^{(i)} \mid y)}{\sum_{y'=1}^{k} p(y')\prod_{j=1}^{n} p_j(x_j^{(i)} \mid y')}
\end{aligned}$$

Then, we maximize

$$\begin{aligned}
& \sum_{i=1}^{m}\sum_{y=1}^{k}\delta(y \mid i)\log\frac{p(x^{(i)}, z^{(i)} = y)}{\delta(y \mid i)} \\
= & \sum_{i=1}^{m}\sum_{y=1}^{k}\delta(y \mid i)\log\frac{p(x^{(i)} \mid z^{(i)} = y)p(z^{(i)} = y)}{\delta(y \mid i)} \\
= & \sum_{i=1}^{m}\sum_{y=1}^{k}\delta(y \mid i)\log\frac{p(y)\prod_{j=1}^{n} p_j(x_j^{(i)} \mid y)}{\delta(y \mid i)} \\
= & \sum_{i=1}^{m}\sum_{y=1}^{k}\delta(y \mid i)\left[\log p(y) + \sum_{j=1}^{n}\log p_j(x_j^{(i)} \mid y) - \log\delta(y \mid i)\right] \\
= & \sum_{i=1}^{m}\sum_{y=1}^{k}\delta(y \mid i)\log p(y) + \sum_{i=1}^{m}\sum_{y=1}^{k}\sum_{j=1}^{n}\delta(y \mid i)\log p_j(x_j^{(i)} \mid y) \\
& -\sum_{i=1}^{m}\sum_{y=1}^{k}\delta(y \mid i)\log\delta(y \mid i)
\end{aligned}$$

over $p(y)$ and $p_j(x \mid y)$. The maximization problem can be formulated as follows:

$$\begin{aligned}
\max \quad & \sum_{i=1}^{m}\sum_{y=1}^{k}\delta(y \mid i)\log p(y) + \sum_{y=1}^{k}\sum_{j=1}^{n}\sum_{x\in\{0,1\}}\left(\sum_{i:x_j^{(i)}=x}\delta(y \mid i)\right)\log p_j(x \mid y) \\
s.t. \quad & \sum_{y=1}^{k} p(y) = 1 \\
& \sum_{x\in\{0,1\}} p_j(x \mid y) = 1,\ \forall y = 1,\cdots,k,\ \forall j = 1,\cdots,n \\
& p(y) \geq 0,\ \forall y = 1,\cdots,k \\
& p_j(x \mid y) \geq 0,\ \forall j = 1,\cdots,n,\quad \forall x \in \{0,1\},\quad \forall y = 1,\cdots,k
\end{aligned}$$

By applying Lagrange multiplier method, we have the optimal solution shown in (35) and (36).