Gaussian discriminant analysis

1.1 The multivariate normal distribution

The multivariate normal distribution in n-dimensions, also called the multivariate Gaussian distribution, is parameterized by a **mean vector** $\mu \in \mathbb{R}^n$ and a **covariance matrix** $\Sigma \in \mathbb{R}^{n \times n}$, where $\Sigma \geq 0$ is symmetric and positive semi-definite. Also written "$\mathcal{N} \sim (\mu, \Sigma)$", its density is given by:

$$p(x; \mu, \Sigma) = \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} \, exp \left(-\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right)$$

In the equation above, "$|\Sigma|$" denotes the determinant of the matrix Σ.

For a random variable X distributed $\mathcal{N}(\mu, \Sigma)$, the mean is (unsurprisingly) given by μ:

$$E[X] = \int_x x \, p(x; \mu, \Sigma) dx = \mu$$

The **covariance** of a vector-valued random variable Z is defined as $Cov(Z) = E[(Z - E[Z])(Z - E[Z])^T]$. This generalizes the notion of the variance of a real-valued random variable. The covariance can also be defined as $Cov(Z) = E[ZZ^T] - (E[Z])(E[Z])^T$. If $X \sim \mathcal{N}(\mu, \Sigma)$,then

$$Cov(X) = \Sigma$$

1.2 The Gaussian Discriminant Analysis model

When we have a classification problem in which the input features x are continuous-valued random variables, we can then use the Gaussian Discriminant Analysis (GDA) model, which models $p(x \mid y)$ using a multivariate normal distribution. The model is:

$$\begin{aligned} y &\sim Bernoulli(\phi) \\ x \mid y = 0 &\sim \mathcal{N}(\mu_0, \Sigma) \\ x \mid y = 1 &\sim \mathcal{N}(\mu_1, \Sigma) \end{aligned}$$

Writing out the distribution, this is:

$$\begin{aligned} p(y) &= \phi^y (1-\phi)^{1-y} \\ p(x \mid y = 0) &= \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}}\, exp\left(-\frac{1}{2}(x-\mu_0)^T \Sigma^{-1} (x-\mu_0)\right) \\ p(x \mid y = 1) &= \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}}\, exp\left(-\frac{1}{2}(x-\mu_1)^T \Sigma^{-1} (x-\mu_1)\right) \end{aligned}$$

Here, the parameters of our model are $\phi, \Sigma, \mu_0, \mu_1$. (Note that while there're two different mean vectors μ_0 and μ_1, this model is usually applied using only one covariance matrix Σ.) The log-likelihood of the data is given by

$$\begin{aligned} l(\phi, \mu_0, \mu_1, \Sigma) &= log \prod_{i=1}^{m} p(x^{(i)}, y^{(i)}; \phi, \mu_0, \mu_1, \Sigma) \\ &= log \prod_{i=1}^{m} p(x^{(i)} \mid y^{(i)}; \phi, \mu_0, \mu_1, \Sigma) p(y^{(i)}; \phi) \end{aligned}$$

By maximizing l with respect to the parameters, we find the maximum likelihood estimate of the parameters to be:

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

1.3 详细推导

1.3.1 准备工作

在推导高斯判别分析的过程中，需要用到以下四个公式：

$$\nabla_x x^T A x = 2Ax\text{，其中，}A\text{为对称矩阵} \quad (1)$$

$$\nabla_A \left|A\right| = \left|A\right|(A^{-1})^T \quad (2)$$

$$\nabla_A log\left|A\right| = A^{-1}\text{，其中，}A\text{为正定矩阵} \quad (3)$$

$$\nabla_A x^T A x = xx^T\text{，其中，}A\text{为对称矩阵} \quad (4)$$

因为，式（1）的矩阵 A 为对称矩阵，所以 x^TAx 为二次型，因此，$\nabla_x x^TAx = 2Ax$。

下证式（2）：

由

$$\left|A\right| = \sum_{i=1}^{n}(-1)^{i+j}A_{ij}\left|A_{\backslash i,\backslash j}\right| \quad (\text{对任意}\, j \in 1, \cdots, n)$$

可得

$$\frac{\partial}{\partial A_{kl}}\left|A\right| = \frac{\partial}{\partial A_{kl}}\sum_{i=1}^{n}(-1)^{i+j}A_{ij}\left|A_{\backslash i,\backslash j}\right| = (-1)^{k+l}\left|A_{\backslash k,\backslash l}\right| = (adj(A))_{lk}$$

其中，adj(A) 表示矩阵 A 的伴随矩阵。因此

$$\nabla_A\left|A\right| = (adj(A))^T = \left|A\right|(A^{-1})^T$$

下证式（3）：

因为，矩阵 A 为正定矩阵，所以，$\left|A\right| > 0$，即 $log\left|A\right|$ 存在，由

$$\frac{\partial\, log\left|A\right|}{\partial A_{ij}} = \frac{\partial\, log\left|A\right|}{\partial\left|A\right|}\frac{\partial\left|A\right|}{\partial A_{ij}} = \frac{1}{\left|A\right|}\frac{\partial\left|A\right|}{\partial A_{ij}}$$

以及式（2）可得

$$\nabla_A log\left|A\right| = \frac{1}{\left|A\right|}\nabla_A \left|A\right| = A^{-1}$$

因为，矩阵 A 为对称矩阵，所以，上式最后的结果没有转置符号。

下证式（4）：

由

$$\frac{\partial\,(x^TAx)}{\partial\,A_{lk}} = \frac{\partial}{\partial\,A_{lk}}\sum_i\sum_j A_{ij}x_ix_j = x_lx_k$$

可得

$$\nabla_A x^TAx = xx^T$$

1.3.2 推导GDA最大似然估计最佳参数

对数似然函数：

$$\begin{aligned}
l(\phi,\mu_0,\mu_1,\Sigma) &= log\prod_{i=1}^{m}p(x^{(i)},y^{(i)}) \\
&= log\prod_{i=1}^{m}p(x^{(i)}\mid y^{(i)})p(y^{(i)}) \\
&= \sum_{i=1}^{m}log\,p(x^{(i)}\mid y^{(i)})+\sum_{i=1}^{m}log\,p(y^{(i)}) \\
&= \sum_{i=1}^{m}log\left(p(x^{(i)}\mid y^{(i)}=0)^{1-y^{(i)}}\cdot p(x^{(i)}\mid y^{(i)}=1)^{y^{(i)}}\right)+\sum_{i=1}^{m}log\,p(y^{(i)}) \\
&= \sum_{i=1}^{m}(1-y^{(i)})log\left(p(x^{(i)}\mid y^{(i)}=0)\right)+\sum_{i=1}^{m}y^{(i)}log\left(p(x^{(i)}\mid y^{(i)}=1)\right)+\sum_{i=1}^{m}log\,p(y^{(i)})
\end{aligned}$$

注意，此函数分为三个部分，μ_0 只与第一部分有关，μ_1 只与第二部分有关，ϕ 只与第三部分有关，Σ 与第一和第二部分有关。

首先，求 ϕ，即

$$\begin{aligned}
\nabla_\phi\, l(\phi, \mu_0, \mu_1, \Sigma) &= \nabla_\phi \ \sum_{i=1}^{m} log\, p(y^{(i)}) \\
&= \nabla_\phi \ \sum_{i=1}^{m} log\, \phi^{y^{(i)}}(1-\phi)^{(1-y^{(i)})} \\
&= \nabla_\phi \ \sum_{i=1}^{m} \left(y^{(i)} log\, \phi + (1-y^{(i)}) log\, (1-\phi)\right) \\
&= \sum_{i=1}^{m} \left(y^{(i)}\frac{1}{\phi} - (1-y^{(i)})\frac{1}{1-\phi}\right) \\
&= \sum_{i=1}^{m} \left(1\{y^{(i)} = 1\}\frac{1}{\phi} - 1\{y^{(i)} = 0\}\frac{1}{1-\phi}\right)
\end{aligned}$$

令其为零，即

$$\sum_{i=1}^{m}\left(1\{y^{(i)}=1\}\frac{1}{\phi}-1\{y^{(i)}=0\}\frac{1}{1-\phi}\right)=0$$

$$\frac{\sum_{i=1}^{m}1\{y^{(i)}=1\}}{\phi}-\frac{\sum_{i=1}^{m}1\{y^{(i)}=0\}}{1-\phi}=0$$

$$\frac{\sum_{i=1}^{m}1\{y^{(i)}=1\}}{\phi}=\frac{\sum_{i=1}^{m}1\{y^{(i)}=0\}}{1-\phi}$$

$$\sum_{i=1}^{m}1\{y^{(i)}=1\}-\phi\sum_{i=1}^{m}1\{y^{(i)}=1\}=\phi\sum_{i=1}^{m}1\{y^{(i)}=0\}$$

$$\phi\left(\sum_{i=1}^{m}1\{y^{(i)}=0\}+\sum_{i=1}^{m}1\{y^{(i)}=1\}\right)=\sum_{i=1}^{m}1\{y^{(i)}=1\}$$

$$\phi=\frac{\sum_{i=1}^{m}1\{y^{(i)}=1\}}{\sum_{i=1}^{m}1\{y^{(i)}=0\}+\sum_{i=1}^{m}1\{y^{(i)}=1\}}$$

注意到，$\sum_{i=1}^{m}1\{y^{(i)}=0\}+\sum_{i=1}^{m}1\{y^{(i)}=1\}=m$，因此，

$$\phi=\frac{\sum_{i=1}^{m}1\{y^{(i)}=1\}}{m}$$

其次，求 μ_0，即

$$\begin{aligned}\nabla_{\mu_0} l(\phi,\mu_0,\mu_1,\Sigma) &= \nabla_{\mu_0} \sum_{i=1}^{m}(1-y^{(i)}) log\, p(x^{(i)} \mid y^{(i)}=0) \\ &= \nabla_{\mu_0} \sum_{i=1}^{m}(1-y^{(i)})(log \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}} - \frac{1}{2}(x^{(i)}-\mu_0)^T\Sigma^{-1}(x^{(i)}-\mu_0)) \\ &= \sum_{i=1}^{m}(1-y^{(i)})\Sigma^{-1}(x^{(i)}-\mu_0) \\ &= \sum_{i=1}^{m} 1\{y^{(i)}=0\}\Sigma^{-1}(x^{(i)}-\mu_0)\end{aligned}$$

令其为零，可得

$$\mu_0 = \frac{\sum_{i=1}^{m} 1\{y^{(i)}=0\}x^{(i)}}{\sum_{i=1}^{m} 1\{y^{(i)}=0\}}$$

同理可得

$$\mu_1 = \frac{\sum_{i=1}^{m} 1\{y^{(i)}=1\}x^{(i)}}{\sum_{i=1}^{m} 1\{y^{(i)}=1\}}$$

最后，求 Σ，在此之前，先证明

$$\nabla_\Sigma \Sigma^{-1} = -\Sigma^{-1}\Sigma^{-1}$$

由

$$\begin{aligned}\frac{\partial I}{\partial x} &= \frac{\partial (A^{-1}A)}{\partial x} \\ &= A^{-1}\frac{\partial A}{\partial x} + \frac{\partial A^{-1}}{\partial x}A \\ &= 0\end{aligned}$$

可得

$$\frac{\partial A^{-1}}{\partial x}A = -A^{-1}\frac{\partial A}{\partial x}$$

两边右乘 A^{-1}，可得

$$\frac{\partial A^{-1}}{\partial x} = -A^{-1}\frac{\partial A}{\partial x}A^{-1}$$

因此

$$\nabla_{\Sigma}\Sigma^{-1} = -\Sigma^{-1}\Sigma^{-1}$$

于是

$$
\begin{aligned}
\nabla_{\Sigma}\, l(\phi, \mu_0, \mu_1, \Sigma) &= \nabla_{\Sigma} \left(\sum_{i=1}^{m} (1 - y^{(i)})\, log\, p(x^{(i)} \mid y^{(i)} = 0\, ; \mu_0, \Sigma) + \sum_{i=1}^{m} y^{(i)}\, log\, p(x^{(i)} \mid y^{(i)} = 1\, ; \mu_1, \Sigma) \right) \\
&= \nabla_{\Sigma} \left(\sum_{i=1}^{m} (1 - y^{(i)})\, log\, \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} e^{-\frac{1}{2}(x^{(i)} - \mu_0)^T \Sigma^{-1} (x^{(i)} - \mu_0)} + \sum_{i=1}^{m} y^{(i)}\, log\, \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} e^{-\frac{1}{2}(x^{(i)} - \mu_1)^T \Sigma^{-1} (x^{(i)} - \mu_1)} \right) \\
&= \nabla_{\Sigma} \left(\sum_{i=1}^{m} log\, \frac{1}{(2\pi)^{n/2} |\Sigma|^{1/2}} - \frac{1}{2} \sum_{i=1}^{m} (x^{(i)} - \mu_{y^{(i)}})^T \Sigma^{-1} (x^{(i)} - \mu_{y^{(i)}}) \right) \\
&= \nabla_{\Sigma} \left(\sum_{i=1}^{m} (-\frac{n}{2} log\, 2\pi - \frac{1}{2} log\, |\Sigma|) - \frac{1}{2} \sum_{i=1}^{m} (x^{(i)} - \mu_{y^{(i)}})^T \Sigma^{-1} (x^{(i)} - \mu_{y^{(i)}}) \right) \\
&= -\frac{m}{2} \Sigma^{-1} + \frac{1}{2} \sum_{i=1}^{m} (x^{(i)} - \mu_{y^{(i)}})(x^{(i)} - \mu_{y^{(i)}})^T \Sigma^{-1} \Sigma^{-1}
\end{aligned}
$$

令其为零，可得

$$
\Sigma = \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \mu_{y^{(i)}})(x^{(i)} - \mu_{y^{(i)}})^T
$$

1.3.3 小结

通过最大化似然函数，得到四个参数的估计值为：

$$\phi = \frac{1}{m}\sum_{i=1}^{m} 1\{y^{(i)} = 1\}$$

$$\mu_0 = \frac{\sum_{i=1}^{m} 1\{y^{(i)} = 0\}x^{(i)}}{\sum_{i=1}^{m} 1\{y^{(i)} = 0\}}$$

$$\mu_1 = \frac{\sum_{i=1}^{m} 1\{y^{(i)} = 1\}x^{(i)}}{\sum_{i=1}^{m} 1\{y^{(i)} = 1\}}$$

$$\Sigma = \frac{1}{m}\sum_{i=1}^{m}(x^{(i)} - \mu_{y^{(i)}})(x^{(i)} - \mu_{y^{(i)}})^T$$

1.4 Discussion: GDA and logistic regression

The GDA model has a an interesting relationship to logistic regression. If we view the quantity $p(y = 1 | x; \phi, \mu_0, \mu_1, \Sigma)$ as a function of x, we'll find that it can be expressed in the form

$$p(y = 1 | x; \phi, \mu_0, \mu_1, \Sigma) = \frac{1}{1 + exp(-\theta^T x)}$$

where θ is some appropriate function of $\phi, \Sigma, \mu_0, \mu_1$. This is exactly the form that logistic regression——a discriminative algorithm——used to model $p(y = 1 | x)$.

推导：

设

$$
\begin{aligned}
A &= -\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1) \\
B &= -\frac{1}{2}(x-\mu_0)^T\Sigma^{-1}(x-\mu_0) \\
C &= \frac{1}{(2\pi)^{n/2}|\Sigma|^{1/2}}
\end{aligned}
$$

因此

$$
\begin{aligned}
p(y=1\mid x) &= Ce^A \\
p(y=0\mid x) &= Ce^B
\end{aligned}
$$

根据上一节的结果以及贝叶斯公式，可得

$$
\begin{aligned}
p(y=1\,|\,x) &= \frac{p(x\,|\,y=1)\,p(y=1)}{p(x)} \\
&= \frac{p(x\,|\,y=1)\,p(y=1)}{p(x\,|\,y=1)\,p(y=1)+p(x\,|\,y=0)\,p(y=0)} \\
&= \frac{Ce^{A}\phi}{Ce^{A}\phi + Ce^{B}(1-\phi)} \\
&= \frac{1}{1+\frac{1-\phi}{\phi}e^{B-A}} \\
&= \frac{1}{1+e^{(B-A+ln\frac{1-\phi}{\phi})}} \\
&= \frac{1}{1+e^{-(A-B+ln\frac{\phi}{1-\phi})}}
\end{aligned}
$$

因为

$$
\begin{aligned}
A-B+ln\frac{\phi}{1-\phi} &= -\frac{1}{2}(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)+\frac{1}{2}(x-\mu_0)^T\Sigma^{-1}(x-\mu_0)+ln\frac{\phi}{1-\phi} \\
&= \frac{1}{2}\big((x-\mu_0)^T\Sigma^{-1}(x-\mu_0)-(x-\mu_1)^T\Sigma^{-1}(x-\mu_1)\big)+ln\frac{\phi}{1-\phi} \\
&= \frac{1}{2}\big(x^T\Sigma^{-1}x-x^T\Sigma^{-1}\mu_0-\mu_0^T\Sigma^{-1}x+\mu_0^T\Sigma^{-1}\mu_0-x^T\Sigma^{-1}x+x^T\Sigma^{-1}\mu_1+\mu_1^T\Sigma^{-1}x-\mu_1^T\Sigma^{-1}\mu_1\big)+ln\frac{\phi}{1-\phi} \\
&= \frac{1}{2}\big(x^T\Sigma^{-1}(\mu_1-\mu_0)+(\mu_1-\mu_0)^T\Sigma^{-1}x+\mu_0^T\Sigma^{-1}\mu_0-\mu_1^T\Sigma^{-1}\mu_1\big)+ln\frac{\phi}{1-\phi} \\
&= (\mu_1-\mu_0)^T\Sigma^{-1}x+\frac{1}{2}(\mu_0^T\Sigma^{-1}\mu_0-\mu_1^T\Sigma^{-1}\mu_1)+ln\frac{\phi}{1-\phi} \\
&= (\mu_1-\mu_0)^T\Sigma^{-1}x+D
\end{aligned}
$$

其中，D 是一个常数。注意，由于 Σ 是一个对称矩阵，所以

$$x^T\Sigma^{-1}(\mu_1 - \mu_0) = (\mu_1 - \mu_0)^T\Sigma^{-1}x$$

因此

$$x^T\Sigma^{-1}(\mu_1 - \mu_0) + (\mu_1 - \mu_0)^T\Sigma^{-1}x = 2(\mu_1 - \mu_0)^T\Sigma^{-1}x$$

于是

$$\begin{aligned}(\theta_1, \cdots, \theta_n)^T &= (\mu_1 - \mu_0)^T\Sigma^{-1} \\ \theta_0 &= D\end{aligned}$$

When would we prefer one model over another? GDA and logistic regression will, in general, give different decision boundaries when trained on the same dataset. Which is better?

We just argued that if $p(x \mid y)$ is multivariate gaussian (with shared Σ), then $p(y \mid x)$ necessarily follows a logistic function. The converse, however, is not true; i.e., $p(y \mid x)$ being a logistic regression does not imply $p(x \mid y)$ is multivariate gaussian. This shows that GDA makes *stronger* modeling assumptions about the data than does logistic regression. It turns out that when these modeling assumptions are correct, then GDA will find better fits to the data, and is a better model. Specifically, when $p(y)$ is indeed gaussian (with shared Σ), then GDA is **asymptotically efficient**. Informally, this means that in the limit of very large training sets (large m), there is no algorithm that is strictly better than GDA (in terms of, say, how accurately they estimate $p(y \mid x)$). In particular, it can be shown that in this setting, GDA will be a better algorithm than logistic regression; and more generally, even for small training set sizes, we would generally expect GDA to better.

In contrast, by making significantly weaker assumptions, logistic regression is also more *robust* and less sensitive to incorrect modeling assumptions. There are many different sets of assumptions that would lead to $p(y \mid x)$ taking the form of a logistic function. For example, if $x \mid y = 0 \sim Poisson(\lambda_0)$, and $x \mid y = 1 \sim Poisson(\lambda_1)$, then $p(y \mid x)$ will be logistic. Logistic regression will also work well on Poisson data like this. But if we were to use GDA on such data——and fit Gaussian distributions to such non-Gaussian data——then the results will be less predictable, and GDA may (or may not) do well.

To summarize: 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. Logistic regression makes weaker assumptions, and is significantly more robust to deviations from modeling assumptions. Specifically, when the data is indeed non-Gaussian, then in the limit of large datasets, logistic regression will almost always do better than GDA. For this reason, in practice logistic regression is used more often than GDA.

```
In [ ]:
```