4 Classifications

4.1 Logistic Regression

4.1.1 Logistic Model

Logistic function:

$$p(x_i) = \frac{e^{\beta^T x_i}}{1 + e^{\beta^T x_i}} = \frac{1}{1 + e^{-\beta^T x_i}}, \quad 0 \leq p(x_i) \leq 1, \quad -\infty < x_i < \infty \tag{4.1}$$

The logistic function will always produce an S-shaped curve.

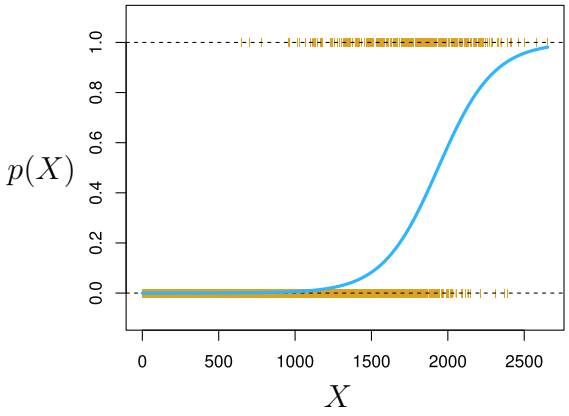

Figure 1

With a bit of manipulation, we will get a odds function:

$$\frac{p(x_i)}{1 - p(x_i)} = e^{\beta^T x_i}, \quad 0 < \frac{p(x_i)}{1 - p(x_i)} < \infty \tag{4.2}$$

The notation $p(x)$ can be interpreted as $\Pr(\mathrm{Y} = 1|\mathrm{X} = \mathrm{x})$, which means the probability of Y is True given x. The equation (**??**) imply the ratio of success compare to fail. For example, average nine of ten people go to school. That means 9 times of people go to school

compare to those who don't. So:

$$p(x) = \Pr(\text{y} = 1|\text{x}) = 9/10$$
$$\frac{p(x)}{1-p(x)} = \frac{9/10}{1-9/10} = 9$$

By taking the logarithm of the odds function, we get

$$\log\left(\frac{p(x)}{1-p(x)}\right) = \beta^T x_i \tag{4.3}$$

The left-hand side is called the log-odds or logit. We see that the logistic regression model has a logit that is linear in X.

4.1.2 Estimating the Regression Coefficients

To fit logistic model (**??**), we use Maximum Likelihood to estimate the coefficients:

Definition 1 (Likelihood Function)**.**
$X_1, X_2, \ldots, X_n$ 爲一組樣本大小爲n之隨機樣本, 記爲$X_1, X_2, \ldots, X_n \overset{i.i.d.}{\sim} f_{X_i}(x_i;\theta)$, 定義母體參數 θ 之 *likelihood function:*

$$L(\theta) = f_{X_1,X_2,\ldots,X_n}(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} f_{X_i}(x_i;\theta)$$

可解讀爲在不同的母體參數值θ之下, 抽到這組觀測的隨機樣本$(X_1, X_2, \ldots, X_n)$之可能性

4.1.2.1 Maximum Likelihood Estimator, $\hat{\theta}_{\text{MLE}}$

找一個 θ 使得拿到這組隨機樣本$(X_1, X_2, \ldots, X_n)$的可能性爲最大, 亦即, 找一個 θ 使得概似函數 $L(\theta)$ 爲最大:

$$\arg\max_{\theta} L(\theta), \text{求解 } \hat{\theta}_{\text{MLE}}$$

Unlike linear regression, we can no longer write down the MLE in closed form. Instead, we need to use an optimization algorithm to compute it. For this, we need to derive the gradient and Hessian. The likelihood function of logistic model:

$$L(\beta) = \prod_{i=1}^{n} p(x_i;\beta)^{y_i}(1-p(x_i;\beta))^{1-y_i}, \text{ where } R_{y_i} = \{0,1\} \tag{4.4}$$

The log-likelihood of logistic model:

$$\begin{aligned}
\ell(\beta) &= \sum_{i=1}^{n} y_i \log p(x_i;\beta) + (1-y_i)\log(1-p(x_i;\beta)) \\
&= \sum_{i=1}^{n} y_i \log p(x_i;\beta) + \log(1-p(x_i;\beta)) - y_i \log(1-p(x_i;\beta)) \\
&= \sum_{i=1}^{n} y_i \log \frac{p(x_i;\beta)}{1-p(x_i;\beta)} + \log(1-p(x_i;\beta)) \\
&= \sum_{i=1}^{n} y_i \beta^T x_i + \log(\frac{1}{1+e^{\beta^T x_i}}) \\
&= \sum_{i=1}^{n} \{y_i \beta^T x_i - \log(1+e^{\beta^T x_i})\}
\end{aligned}$$

First order partial differential of the log-likelihood function (gradient descent):

$$\begin{aligned}
\frac{\partial \ell(\beta)}{\partial \beta} &= \sum_{i=1}^{n} x_i (y_i - \frac{e^{\beta^T x_i}}{1+e^{\beta^T x_i}}) \\
&= \sum_{i=1}^{n} x_i (y_i - p(x_i;\beta))
\end{aligned}$$

Second order partial differential of the log-likelihood function (the s.o.c can transfer to a

Hessian matrix):

$$\begin{aligned}\frac{\partial^2 \ell(\beta)}{\partial\beta\partial\beta^T} &= \sum_{i=1}^{n} x_i(-x_i \cdot \frac{e^{-\beta^T x_i}}{1+e^{-\beta^T x_i}} \cdot \frac{1}{1+e^{-\beta^T x_i}}) \\ &= -\sum_{i=1}^{n} x_i x_i^T p(x_i;\beta)(1-p(x_i;\beta))\end{aligned}$$

Starting with β^{old}, a single Newton update is

$$\beta^{new} = \beta^{old} - \left(\frac{\partial^2 \ell(\beta)}{\partial\beta\partial\beta^T}\right)^{-1} \frac{\partial \ell(\beta)}{\partial\beta}$$

Now we will try to simplify by vectorizing the model. Let $\mathbf{y}$ denote the vector of y_i, $\mathbf{X}$ the $N \times (p+1)$ matrix of x_i (p predictors with one intercept), $\mathbf{p}$ the vector of fitted probabilities with ith element $p(x_i;\boldsymbol{\beta}^{\text{old}})$ and $\mathbf{W}$ a $N \times N$ diagonal matrix of weights with ith diagonal element $p(x_i;\boldsymbol{\beta}^{\text{old}})(1-p(x_i;\boldsymbol{\beta}^{\text{old}}))$. The structure of the notation and the model:

$$\mathbf{y}_{N\times 1} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix}, \mathbf{X}_{N\times(p+1)} = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1p} \\ 1 & x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & x_{N2} & \cdots & x_{Np} \end{pmatrix},$$

$$\mathbf{p}_{N\times 1} = \begin{pmatrix} p(x_1;\boldsymbol{\beta}^{\text{old}}) \\ p(x_2;\boldsymbol{\beta}^{\text{old}}) \\ \vdots \\ p(x_N;\boldsymbol{\beta}^{\text{old}}) \end{pmatrix}, \boldsymbol{\beta}^{\text{old}}_{(p+1)\times 1} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix},$$

$$\mathbf{W}_{N\times N} = \begin{pmatrix} p(x_1;\boldsymbol{\beta}^{\text{old}})(1-p(x_1;\boldsymbol{\beta}^{\text{old}})) & 0 & \cdots & 0 \\ 0 & p(x_2;\boldsymbol{\beta}^{\text{old}})(1-p(x_2;\boldsymbol{\beta}^{\text{old}})) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & p(x_N;\boldsymbol{\beta}^{\text{old}})(1-p(x_N;\boldsymbol{\beta}^{\text{old}})) \end{pmatrix}$$

$$
\begin{aligned}
\frac{\partial \ell(\beta)}{\partial \beta} &= \mathbf{X}^T(\mathbf{y}-\mathbf{p}) \\
\frac{\partial^2 \ell(\beta)}{\partial \beta \partial \beta^T} &= -\mathbf{X}^T\mathbf{W}\mathbf{X} \\
\boldsymbol{\beta}^{\text{new}} &= \boldsymbol{\beta}^{\text{old}} + \left(\mathbf{X}^T\mathbf{W}\mathbf{X}\right)^{-1}\mathbf{X}^T(\mathbf{y}-\mathbf{p}) \\
&= \left(\mathbf{X}^T\mathbf{W}\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{W}\left(\mathbf{X}\boldsymbol{\beta}^{\text{old}} + \mathbf{W}^{-1}(\mathbf{y}-\mathbf{p})\right) \\
&= \left(\mathbf{X}^T\mathbf{W}\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{W}\mathbf{z}
\end{aligned}
$$

Response of weighted least squares step, sometimes known as adjusted response:

$$
\mathbf{z} = \mathbf{X}\boldsymbol{\beta}^{\text{old}} + \mathbf{W}^{-1}(\mathbf{y}-\mathbf{p}) \tag{4.5}
$$

The equations above solved repeatedly, since at each iteration $\mathbf{p}$ changes, and hence so does $\mathbf{W}$ and $\mathbf{z}$. This algorithm is referred to as iteratively reweighed least squares (IRLS).

Algorithm 1 Iteratively reweighted least squares (IRLS)

1: $\boldsymbol{\beta}^{\text{new}} = \begin{pmatrix} 1 \\ \boldsymbol{\beta}_p \end{pmatrix}$;
2: **repeat**
3: $\quad \mathbf{p} = p(\mathbf{X}; \boldsymbol{\beta}^{\text{new}})$
4: $\quad \mathbf{W}_{N\times N} = \text{diag}(\mathbf{p}(1-\mathbf{p}))$
5: $\quad \mathbf{z} = \mathbf{X}\boldsymbol{\beta}^{\text{new}} + \mathbf{w}^{-1}(\mathbf{y}-\mathbf{p})$
6: $\quad \boldsymbol{\beta}^{\text{new}} = (\mathbf{X}^T\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{W}\mathbf{z}$
7: **until** calculate loss function to check converged;

4.1.2.2 Multi-class logistic Regression

For multi-class logistic regression, the model take the form,

$$P(y = c|\vec{x}, \vec{w}) = \frac{e^{w_j^c x^j}}{\sum_c e^{w_j^c x^j}}$$

- c indicates the class, there are totally C classes
- j indicates the dimension of variables

Here, the superscript indicates column vector and subscript as row vector. The detail is described as below.

$$\vec{a} = \begin{pmatrix} \vdots \\ a^j \\ \vdots \end{pmatrix} \tag{4.6}$$

$$\vec{b} = (\cdots \quad b_i \quad \cdots) \tag{4.7}$$

$$\vec{c} = \begin{pmatrix} \\ c_i^j \\ \\ \end{pmatrix} \tag{4.8}$$

$\mathbb{I}(y = c)$ is the binary indicates the outcome class of the observation, if the observation y belongs to c then $\mathbb{I}(y = c)$ equals to 1, else $\mathbb{I}(y = c)$ equals to 0. The negative log likelihood is defined as below,

$$\begin{aligned} L(\vec{w}) = -\ell(\vec{w}) &= -\log \prod_{i=1} \prod_{c=1} P({}_iy = c|\vec{w}, {}_ix^j)^{\mathbb{I}({}_iy=c)} \\ &= -\sum_i \sum_c \mathbb{I}({}_iy = c) \log \frac{e^{w_j^c \cdot {}_ix^j}}{\sum_c e^{w_j^c \cdot {}_ix^j}} \\ &= \sum_i \sum_c \mathbb{I}({}_iy = c)(\log \sum_c e^{w_j^c \cdot {}_ix^j} - w_j^c \cdot {}_ix^j) \\ &= \sum_i (\log \sum_c e^{w_j^c \cdot {}_ix^j} - \sum_c \mathbb{I}({}_iy = c) w_j^c \cdot {}_ix^j) \end{aligned}$$

The gradient,

$$\begin{aligned}\nabla_{w^c} L(\vec{w}) &= \sum_i \big(\frac{e^{w_j^c \cdot {}_i x^j}}{\sum_c e^{w_j^c \cdot {}_i x^j}} {}_i x^j - \mathbb{I}({}_i y = c) {}_i x^j\big) \\ &= \sum_i \big(P({}_i y = c | {}_i \vec{x}, \vec{w}) - \mathbb{I}({}_i y = c)\big) {}_i x^j \\ &= \big(P(y = c|\vec{x}, \vec{w}) - \mathbb{I}(y = c)\big)\vec{x}^T\end{aligned}$$

4.1.2.3 Multi-class logistic Regression : Example

3 Class, 3 Observation, 2 Dimension

$$\vec{y} = \begin{pmatrix} {}_i y \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \tag{4.9}$$

$$\vec{x} = \begin{pmatrix} {}_i x^j \end{pmatrix} = \begin{pmatrix} {}_1 x^1 & {}_2 x^1 & {}_3 x^1 \\ {}_1 x^2 & {}_2 x^2 & {}_3 x^2 \end{pmatrix} \tag{4.10}$$

$$\vec{w} = \begin{pmatrix} w_j^c \end{pmatrix} = \begin{pmatrix} w_1^1 & w_2^1 \\ w_1^2 & w_2^2 \\ w_1^3 & w_2^3 \end{pmatrix} \tag{4.11}$$

$$\vec{s} = \vec{w}\vec{x} = \begin{pmatrix} {}_i s^c \end{pmatrix} = \begin{pmatrix} {}_1 s^1 & {}_2 s^1 & {}_3 s^1 \\ {}_1 s^2 & {}_2 s^2 & {}_3 s^2 \\ {}_1 s^3 & {}_2 s^3 & {}_3 s^3 \end{pmatrix} \tag{4.12}$$

$$P(y=c|\vec{x},\vec{w}) = e^{\vec{s}} / \sum_c e^{s^c} = \begin{pmatrix} \frac{e^{is^c}}{\sum_c e^{is^c}} \end{pmatrix} = \tag{4.13}$$

$$\begin{pmatrix} e^{1s^1}/\sum_c e^{is^c} & e^{2s^1}/\sum_c e^{is^c} & e^{3s^1}/\sum_c e^{is^c} \\ e^{1s^2}/\sum_c e^{is^c} & e^{2s^2}/\sum_c e^{is^c} & e^{3s^2}/\sum_c e^{is^c} \\ e^{1s^3}/\sum_c e^{is^c} & e^{2s^3}/\sum_c e^{is^c} & e^{3s^3}/\sum_c e^{is^c} \end{pmatrix} \tag{4.14}$$

$$\mathbb{I}(\vec{y}=c) = \begin{pmatrix} \mathbb{I}({}_1y=1) & \mathbb{I}({}_2y=1) & \mathbb{I}({}_3y=1) \\ \mathbb{I}({}_1y=2) & \mathbb{I}({}_2y=2) & \mathbb{I}({}_3y=2) \\ \mathbb{I}({}_1y=3) & \mathbb{I}({}_2y=3) & \mathbb{I}({}_3y=3) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{4.15}$$

Loss function,

$$\vec{L} = \mathbb{I}(\vec{y}=c)P(y=c|\vec{x},\vec{w}) = -\sum \log \begin{pmatrix} e^{1s^1}/\sum_c e^{is^c} & 0 & 0 \\ 0 & e^{2s^2}/\sum_c e^{is^c} & 0 \\ 0 & 0 & e^{3s^3}/\sum_c e^{is^c} \end{pmatrix} \tag{4.16}$$

Now compute the gradient,

$$P(y=c|\vec{x},\vec{w}) - \mathbb{I}(\vec{y}=c) = \begin{pmatrix} e^{1s^1}/\sum_c e^{is^c} - 1 & e^{2s^1}/\sum_c e^{is^c} & e^{3s^1}/\sum_c e^{is^c} \\ e^{1s^2}/\sum_c e^{is^c} & e^{2s^2}/\sum_c e^{is^c} - 1 & e^{3s^2}/\sum_c e^{is^c} \\ e^{1s^3}/\sum_c e^{is^c} & e^{2s^3}/\sum_c e^{is^c} & e^{3s^3}/\sum_c e^{is^c} - 1 \end{pmatrix} \tag{4.17}$$

$$\nabla_{w^c} L(\vec{w}) = \big(P(y=c|\vec{x},\vec{w}) - \mathbb{I}(y=c)\big)\vec{x}^T = \tag{4.18}$$

$$\begin{pmatrix} e^{1s^1}/\sum_c e^{is^c} - 1 & e^{2s^1}/\sum_c e^{is^c} & e^{3s^1}/\sum_c e^{is^c} \\ e^{1s^2}/\sum_c e^{is^c} & e^{2s^2}/\sum_c e^{is^c} - 1 & e^{3s^2}/\sum_c e^{is^c} \\ e^{1s^3}/\sum_c e^{is^c} & e^{2s^3}/\sum_c e^{is^c} & e^{3s^3}/\sum_c e^{is^c} - 1 \end{pmatrix} \begin{pmatrix} {}_1x^1 & {}_1x^2 \\ {}_2x^1 & {}_2x^2 \\ {}_3x^1 & {}_3x^2 \end{pmatrix} \tag{4.19}$$

4.1.3 Discussion

1. How to calculate the fitness of logistic model?
2. How to check the estimated coefficients is significant or not?
3. How to perform the hypothesis testing for logistic regression?
4. What is the assumptions of logistic regression model?

4.2 Linear Discriminant Analysis (LDA)

Difference between Linear Discriminant Analysis and Logistic Analysis:

- when the classes are well-separated, the parameter estimates for the logistic regression model are surprisingly unstable. LDA does not suffer from this problem
- If n is small and the distribution of the predictors X is approximately normal in each of the classes, LDA is more stable than logistic regression
- LDA is preferable when we have more than two response classes.
- Logistic regression can take both qualitative and quantitative as predictor variables. However LDA can only take quantitative predictor variables, due to the assumption of multivariate Gaussian Distribution.
- LDA assumes that the classes have a common covariance matrix (Continuous random variable X)

4.2.1 Bayes Theorem for Classification

Classify an observation into one of K classes, where $K \geq 2$.

Notations:

- Y: The qualitative response variable with K different categories, $R_Y = 1, \ldots, K$
- π_k: The prior probability that a randomly chosen observation comes from the kth class
- $f_k(X)$: The density function of X for an observation comes form the kth class
- $h^*(x)$: Indicate the class which the training data x belongs to

Note:

$$\hat{\pi}_k = n_k/n = \hat{Pr}(Y = k)$$
$$f_k(X) = Pr(X = x|Y = k)$$

From the law of total probability, the above notations can be stated as below:

$$\begin{aligned} Pr(Y = k|X = x) &= \frac{P(X = x|Y = k)P(Y = k)}{\sum_{\ell=1}^{K} P(X = x|Y = l)P(Y = l)} \\ &= \frac{\pi_k f_k(x)}{\sum_{\ell=1}^{K} \pi_\ell f_\ell(x)} \end{aligned}$$

The abbreviation for $Pr(Y = k|X = x)$ will be denoted as $p_k(X)$ which is the posterior probability that an observation $X = x$ belongs to the kth class. In the following section, we have to make assumptions and approximates $f_k(X)$ to build a classifier that approximates the Bayes classifier.

Definition 2 (The law of total probability)**.**
設 $A_1, A_2, \ldots, A_n$ 爲樣本空間S中之一組分割, 則對於樣本空間S之任意事件B而言,

$$P(B) = \sum_{i=1}^{n} P(B|A_i)P(A_i)$$

Definition 3 (Bayes' Theorem)**.**

設 $A_1, A_2, \ldots, A_n$ 爲樣本空間S中之一組分割, **B**是樣本空間**S**上之任意事件, 則

$$P(A_k|B) = \frac{P(B|A_k)P(A_k)}{\sum_{i=1}^{n} P(B|A_i)P(A_i)}$$

In below we define the Bayes Classifier for two class $y = \{0, 1\}$. Let

$$r(x) = P(Y = 1|X = x) \tag{4.20}$$

denote the regression function. From Bayes' Theorem we have

$$\begin{aligned} r(x) &= \frac{P(X = x|Y = 1)P(Y = 1)}{P(X = x|Y = 0)P(Y = 0) + P(X = x|Y = 1)P(Y = 1)} \\ &= \frac{f_1(x)\pi}{f_1(x)\pi + f_0(x)(1 - \pi)} \end{aligned}$$

Definition 4 (The Bayes Classification rule $h^*(x)$)**.**

$$h^*(x) = \begin{cases} 1 & \text{if } r(x) > \frac{1}{2} \\ 0 & \text{otherwise.} \end{cases}$$

The set $\{x : P(Y = 1|X = x) = P(Y = 0|X = x)\}$ *is called the decision boundary.*

4.2.2 Linear Discriminant Analysis for p=1

In the assumption of the Linear Discriminant Analysis, we assume the independent variable X follow Gaussian distribution. So the $f_k(x)$ takes the form

$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{1}{2\sigma_k^2}(x-\mu_k)^2} \tag{4.21}$$

LDA also assume homoscedasticity for $\sigma_1^2 = \sigma_2^2 = \cdots = \sigma_K^2$, so the simplify version of $f_k(x)$

$$f_k(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(x-\mu_k)^2} \tag{4.22}$$

By plugging (**??**) into (**??**), we get

$$p_k(x) = \frac{\pi_k \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(x-\mu_k)^2}}{\sum_{\ell=1}^{K} \pi_\ell \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(x-\mu_\ell)^2}} \tag{4.23}$$

In order to estimate the parameters from (**??**), we apply MAP,

$$\begin{aligned} \hat{p}_k &= \arg\max_k P(Y = k|X = x) \\ &= \arg\max_k \pi_k f_k(x) \end{aligned} \tag{4.24}$$

The objective of LDA is to find a k that maximizes posterior probability($\hat{p}_k$) among all Kth posterior probabilities. From the above model, in order to meet this objective, we will try to find a k which will maximize the conditional probability ($\pi_k f_k(x)$). Since, the maximum of $\pi_k f_k(x)$, imply the largest probability among all Kth posterior probabilities. In order to simplify (**??**), we take the log of (**??**)

$$\begin{aligned} \arg\max_k \delta_k(x) &= \arg\max_k \left(\log\left(\pi_k f_k(x)\right) \right) \\ &= \arg\max_k \left(\log \pi_k + x\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \underline{\log \frac{1}{\sqrt{2\pi}\sigma} - \frac{x^2}{2\sigma^2}} \right) \\ &= \arg\max_k \left(\log \pi_k + x\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \mathrm{C} \right) \end{aligned} \tag{4.25}$$

In practice we don't know the parameters of the Gaussian distributions(unknown μ_k, σ^2), and will need to estimate them using our training data:

- $\hat{\pi}_k = n_k/n$
- $\hat{\mu}_k = \sum_{i:y_i=k} x_i/n_k$
- $\hat{\sigma}^2 = \sum_{k=1}^{K}(n_k - 1)S_k/(n-k)$ (pooled variance, homoscedasticity property)
- $S_k = \sum_{i:y_i=k} (x_i - \hat{\mu}_k)^2/(n_k - 1)$

The discriminant function,

$$\arg\max_k \hat{\delta}_k(x) = \arg\max_k \left(\left(\log \hat{\pi}_k - \frac{\hat{\mu_k}^2}{2\hat{\sigma^2}} \right) + x\frac{\hat{\mu}_k}{\hat{\sigma}^2} \right) \tag{4.26}$$

The reason linear in the name of LDA, is because the discriminant functions are linear functions of x

4.2.3 Linear Discriminant Analysis for $p > 1$

Assume that $X = \begin{pmatrix} X_1 & X_1 & \dots & X_p \end{pmatrix}$ drown from a multivariate Gaussian distribution, with a class-specific mean vector and a common covariance matrix. To indicate that a p-dimensional random variable X has a multivariate Gaussian Distribution, we write

$$X \sim N(\mu, \mathbf{\Sigma})$$
$$E(X) = \sum_{i=1}^{p} X_i/p = \mu$$
$$Cov(X) = \mathbf{\Sigma}$$

$\mathbf{\Sigma}$ is the $p \times p$ covariance matrix of X. The multivariate Gaussian Distribution density is defined as

$$f(x) = \frac{1}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T \mathbf{\Sigma}^{-1}(x-\mu)}$$

Using the same method as describe for $p = 1$, we get the optimization function for $p > 1$ version

$$\begin{aligned}
\arg\max_k \delta_k(x) &= \arg\max_k \left(\log\left(\pi_k f_k(x)\right) \right) \\
&= \arg\max_k \left(\log \pi_k + \log\left(\frac{1}{(2\pi)^{p/2}|\mathbf{\Sigma}|^{1/2}}\right) - \frac{1}{2}(x-\mu_k)^T \mathbf{\Sigma}^{-1}(x-\mu_k) \right) \\
&= \arg\max_k \left(\log \pi_k + x^T \mathbf{\Sigma}^{-1}\mu_k - \frac{1}{2}\mu_k^T \mathbf{\Sigma}^{-1}\mu_k + \mathrm{C} \right) \\
&= \arg\max_k \left(\log \pi_k + x^T \mathbf{\Sigma}^{-1}\mu_k - \frac{1}{2}\mu_k^T \mathbf{\Sigma}^{-1}\mu_k \right)
\end{aligned}$$

The parameters $(\hat{\pi}_k, \hat{\mu}_k)$ are estimated as before except $\mathbf{\Sigma}$. Let S_k denote the $p \times p$ covariance matrix for class k. Then the sample covariance matrix $\mathbf{\Sigma}$ can be estimated by substituting in the pooled covariance matrix S :

$$S = \frac{\sum_{k=1}^{K}(n_k - 1)S_k}{n - K}$$

The discriminant function for $p > 1$

$$\arg\max_k \hat{\delta}_k(x) = \arg\max_k \left(\log \hat{\pi}_k + x^T S^{-1}\hat{\mu}_k - \frac{1}{2}\hat{\mu}_k^T S^{-1}\hat{\mu}_k \right)$$

Example:

Two predictors with 11 observations and $k = 2$:

$$[x^T, y] = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \end{array} \begin{array}{c} \begin{array}{ccc} X_1 & X_2 & y \end{array} \\ \left(\begin{array}{cc|c} 1 & 2 & 1 \\ 2 & 3 & 1 \\ 3 & 3 & 1 \\ 4 & 5 & 1 \\ 5 & 5 & 1 \\ 1 & 0 & 2 \\ 2 & 1 & 2 \\ 3 & 1 & 2 \\ 3 & 2 & 2 \\ 5 & 3 & 2 \\ 6 & 5 & 2 \end{array}\right) \end{array}$$

$$x^{k=1} = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{array} \begin{array}{c} \begin{array}{ccc} X_1 & X_2 & y \end{array} \\ \left(\begin{array}{cc|c} 1 & 2 & 1 \\ 2 & 3 & 1 \\ 3 & 3 & 1 \\ 4 & 5 & 1 \\ 5 & 5 & 1 \end{array}\right) \end{array}$$

$$x^{k=2} = \begin{array}{c} \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \end{array} \begin{array}{c} \begin{array}{ccc} X_1 & X_2 & y \end{array} \\ \left(\begin{array}{cc|c} 1 & 0 & 2 \\ 2 & 1 & 2 \\ 3 & 1 & 2 \\ 3 & 2 & 2 \\ 5 & 3 & 2 \\ 6 & 5 & 2 \end{array}\right) \end{array}$$

$$\hat{\mu}_1 = \begin{bmatrix} 3 & 3.6 \end{bmatrix} \qquad \hat{\mu}_2 = \begin{bmatrix} 3.3 & 2 \end{bmatrix}$$

$$S_1 = \begin{bmatrix} \sigma_{11} = 2.5 & \sigma_{12} = 2 \\ \sigma_{21} = 2 & \sigma_{22} = 1.8 \end{bmatrix} \qquad S_2 = \begin{bmatrix} \sigma_{11} = 3.47 & \sigma_{12} = 3.2 \\ \sigma_{21} = 3.2 & \sigma_{22} = 3.2 \end{bmatrix}$$

$$S = \frac{(n_1 - 1)S_1 + (n_2 - 1)S_2}{n - K} = \frac{\begin{bmatrix} 10 & 8 \\ 8 & 7.2 \end{bmatrix} + \begin{bmatrix} 17.3 & 16 \\ 16 & 16 \end{bmatrix}}{11 - 2} = \frac{\begin{bmatrix} 27.3 & 24 \\ 24 & 23.2 \end{bmatrix}}{11 - 2}$$

$$\hat{\pi}_1 = \frac{5}{11} \qquad \hat{\pi}_2 = \frac{6}{11}$$

Definition 5 (The Bayes Classification rule for LDA)**.**

If $X|Y = 0 \sim N(\mu_0, \mathbf{\Sigma})$ *and* $X|Y = 1 \sim N(\mu_1, \mathbf{\Sigma})$*, then the Bayes rule is*

$$h^*(x) = \begin{cases} 1 & \text{if } x^T\mathbf{\Sigma}^{-1}(\mu_1 - \mu_0) > \frac{1}{2}(\mu_1 + \mu_0)^T\mathbf{\Sigma}^{-1}(\mu_1 - \mu_0) - \log\frac{\pi_1}{\pi_0} \\ 0 & \text{otherwise.} \end{cases}$$

The decision boundary takes the form

$$\frac{\delta_1(x)}{\delta_0(x)} = \frac{\log \pi_1 + x^T \mathbf{\Sigma}^{-1} \mu_1 - \frac{1}{2} \mu_1^T \mathbf{\Sigma}^{-1} \mu_1}{\log \pi_0 + x^T \mathbf{\Sigma}^{-1} \mu_0 - \frac{1}{2} \mu_0^T \mathbf{\Sigma}^{-1} \mu_0} > 1$$

$$x^T \mathbf{\Sigma}^{-1} (\mu_1 - \mu_0) > \frac{1}{2} (\mu_1 + \mu_0)^T \mathbf{\Sigma}^{-1} (\mu_1 - \mu_0) - \log \frac{\pi_1}{\pi_0}$$

4.2.4 Quadratic Discriminant Analysis

Linear Discriminant Analysis assumes that the observations within each class are drawn from a multivariate Gaussian Distribution. Each class has a specific mean vector (μ_k) and a common covariance matrix ($\mathbf{\Sigma}$). Unlike LDA, Quadratic Discriminant Analysis assumes that each class has its own covariance matrix ($\mathbf{\Sigma}_k$). It assumes that an observation from kth class is of the form $X \sim N(\mu_k, \mathbf{\Sigma}_k)$. Under this assumption, the optimization function for Bayes classifier takes the form

$$\arg\max_k \delta_k(x) = \arg\max_k \left(-\frac{1}{2} \log |\mathbf{\Sigma}_k| - \frac{1}{2} (x - \mu_k)^T \mathbf{\Sigma}_k^{-1} (x - \mu_k) + \log \pi_k \right)$$

Definition 6 (The Bayes Classification rule for QDA)**.**
If $X|Y = 0 \sim N(\mu_0, \mathbf{\Sigma}_0)$ *and* $X|Y = 1 \sim N(\mu_1, \mathbf{\Sigma}_1)$*, then the Bayes rule is*

$$h^*(x) = \begin{cases} 1 & \text{if } r_1^2 < r_0^2 + 2\log(\frac{\pi_1}{\pi_0}) + \log\left(\frac{|\mathbf{\Sigma_0}|}{|\mathbf{\Sigma_1}|}\right) \\ 0 & \text{otherwise.} \end{cases}$$

where

$$r_i^2 = (x - \mu_i)^T \Sigma_i^{-1} (x - \mu_i), i = 0, 1$$

The upper Bayes rule is generated from the decision boundary between any of the two

classes. The decision boundary takes the form

$$\frac{\delta_k(x)}{\delta_l(x)} = \frac{-\frac{1}{2}\log|\mathbf{\Sigma}_k| - \frac{1}{2}(x-\mu_k)^T\mathbf{\Sigma}_k^{-1}(x-\mu_k) + \log\pi_k}{-\frac{1}{2}\log|\mathbf{\Sigma}_l| - \frac{1}{2}(x-\mu_l)^T\mathbf{\Sigma}_l^{-1}(x-\mu_l) + \log\pi_l} > 1$$

$$\log|\mathbf{\Sigma}_k| - (x-\mu_k)^T\mathbf{\Sigma}_k^{-1}(x-\mu_k) + 2\log\pi_k > \log|\mathbf{\Sigma}_l| - (x-\mu_l)^T\mathbf{\Sigma}_l^{-1}(x-\mu_l) + 2\log\pi_l$$

$$\log|\mathbf{\Sigma}_k| - r_k^2 + 2\log\pi_k > \log|\mathbf{\Sigma}_l| - r_l^2 + 2\log\pi_l$$

$$r_l^2 + 2\log\frac{\pi_k}{\pi_l} + \log\frac{|\mathbf{\Sigma}_k|}{|\mathbf{\Sigma}_l|} > r_k^2$$

4.3 A Comparison of Classification Methods

	Logistic	Linear Discriminant	Quadratic Discriminant	KNN
Model Assumption	No special assumption	$X\|Y=k \sim N(\mu_k, \mathbf{\Sigma})$	$X\|Y=k \sim N(\mu_k, \mathbf{\Sigma}_k)$	No special assumption
Predicator Variable	Continuous and Discrete	Continuous	Continuous	Continuous or Discrete
Response Variable	Discrete			
Linear data prediction	Good	Good	Bad	Bad
Non-linear data prediction	Bad	Bad	Good	Good

4.4 Exercises

1. Using a little bit of algebra, prove that (4.2) is equivalent to (4.3). In other words, the logistic function representation and logit representation for the logistic regression

model are equivalent.

Answer:

$$p(x) = \frac{e^{\beta^T x}}{1 + e^{\beta^T x}}$$
$$1 - p(x) = \frac{1}{1 + e^{\beta^T x}}$$
$$\frac{p(x)}{1 - p(x)} = e^{\beta^T x}$$

2. It was stated in the text that classifying an observation to the class for which (4.12) is largest is equivalent to classifying an observation to the class for which (4.13) is largest. Prove that this is the case. In other words, under the assumption that the observations in the kth class are drawn from a $N(\mu_k, \sigma^2)$ distribution, the Bayes' classifier assigns an observation to the class for which the discriminant function is maximized.

Answer:

From the Bayes' Theorem we know (4.12), for any class k, the total probability $\sum_{l=1}^{K} \pi_l f_l(x)$ for each class is the same. However, the prior probability π_k and the probability $f_k(x)$ will differ depending on it's k. So, the objective is to find the largest $\pi_k f_k(x)$ among the range of $(\pi_1 f_1(x), \ldots, \pi_k f_k(x), \ldots, \pi_K f_K(x))$. With the logarithm transformation we get

$$\delta_k(x) = \log\left(\pi_k f_k(x)\right)$$

3. This problem relates to the QDA model, in which the observations within each class are drawn from a normal distribution with a class specific mean vector and a class specific covariance matrix We consider the simple case where $p = 1$; i.e. there is only one feature. Suppose that we have K classes, and that if an observation belongs to the

kth class then X comes from a one-dimensional normal distribution, $X \sim N(\mu_k, \sigma_k^2)$. Recall that the density function for the one-dimensional normal distribution is given in (4.11). Prove that in this case, the Bayes' classifier is not linear. Argue that it is in fact quadratic.

Answer:

Since we have

$$f_k(x) = \frac{1}{(2\pi)^{1/2}\sigma_k} e^{-\frac{1}{2}(x-\mu_k)^T \sigma_k^{-1}(x-\mu_k)}$$

For $\delta_k(x)$,

$$\begin{aligned}
\delta_k(x) &= \log\left(f_k(x)\pi_k\right) \\
&= -\log(2\pi)^{1/2} - \log\sigma_k - \frac{1}{2}(x-\mu_k)^T\sigma_k^{-1}(x-\mu_k) + \log\pi_k \\
&= -\log\sigma_k - \frac{1}{2}(x-\mu_k)^T\sigma_k^{-1}(x-\mu_k) + \log\pi_k + c \\
&= -\log\sigma_k + \log\pi_k - \frac{1}{2}\sigma_k^{-1}(x-\mu_k)^2
\end{aligned}$$

c can be cancelled due to it is merely a constant for all K functions. The above formula will generate an quadratic term x^2.

4. When the number of features p is large, there tends to be a deterioration in the performance of KNN and other local approaches that perform prediction using only observations that are near the test observation for which a prediction must be made. This phenomenon is known as the curse of dimensionality, and it ties into the fact that non-parametric approaches often perform poorly when p is large. We will now investigate this curse.

   (a) Suppose that we have a set of observations, each with measurements on $p = 1$ feature, X. We assume that X is uniformly (evenly) distributed on [0, 1].

Associated with each observation is a response value. Suppose that we wish to predict a test observations response using only observations that are within 10% of the range of X closest to that test observation. For instance, in order to predict the response for a test observation with $X = 0.6$, we will use observations in the range [0.55, 0.65]. On average, what fraction of the available observations will we use to make the prediction?

(b) Now suppose that we have a set of observations, each with measurements on $p = 2$ features, X_1 and X_2 . We assume that (X_1, X_2) are uniformly distributed on $[0, 1] \times [0, 1]$. We wish to predict a test observations response using only observations that are within 10% of the range of X_1 and within 10% of the range of X_2 closest to that test observation. For instance, in order to predict the response for a test observation with $X_1 = 0.6$ and $X_2 = 0.35$, we will use observations in the range $[0.55, 0.65]$ for X_1 and in the range $[0.3, 0.4]$ for X_2 . On average, what fraction of the available observations will we use to make the prediction?

(c) Now suppose that we have a set of observations on p = 100 features. Again the observations are uniformly distributed on each feature, and again each feature ranges in value from 0 to 1. We wish to predict a test observation's response using observations within the 10 % of each feature's range that is closest to that test observation. What fraction of the available observations will we use to make the prediction?

(d) Using your answers to parts (a)-(c), argue that a drawback of KNN when p is large is that there are very few training observations "near" any given test observation.

(e) Now suppose that we wish to make a prediction for a test observation by creating a p-dimensional hypercube centred around the test observation that contains, on average, 10% of the training observations. For $p = 1$, 2, and 100, what is the length of each side of the hypercube? Comment on your answer.

Answer:

(a) $X \sim U(0,1), \int_{x-0.05}^{x+0.05} 1dx = 0.1$

(b)

$$X_1, X_2 \overset{iid}{\sim} U(0,1)$$

$$P(x_1 - 0.05 < X_1 < x_1 + 0.05) \times P(x_2 - 0.05 < X_2 < x_2 + 0.05) = 0.1^2$$

(c) $X_1, \ldots, X_{100} \overset{iid}{\sim} U(0,1) = 0.1^{100}$

(d)

$$\lim_{p \to \infty} 0.1^p = 0$$

From the proof above, we know when p is large, the training data will hardly cover any of the observation we want to train on. That means, the observed data is not in the raw data, so if we plug this observation in the prediction model, the prediction results are certainly not what we expected.

5. We now examine the differences between LDA and QDA.

   (a) If the Bayes decision boundary is linear, do we expect LDA or QDA to perform better on the training set? On the test set?

   (b) If the Bayes decision boundary is non-linear, do we expect LDA or QDA to perform better on the training set? On the test set?

   (c) In general, as the sample size n increases, do we expect the test prediction accuracy of QDA relative to LDA to improve, decline, or be unchanged? Why?

   (d) True or False: Even if the Bayes decision boundary for a given problem is linear, we will probably achieve a superior test error rate using QDA rather than LDA

because QDA is flexible enough to model a linear decision boundary. Justify your answer.

Answer:

(a) When data is linear, QDA performs better than LDA in training set due to QDA has lower bias. However, LDA performs better than QDA in test set due to QDA has higher variance.

(b) When data is non-linear, QDA performs better than LDA both on the training set and test set.

(c) When sample size n increases, the test prediction accuracy will improve since over-fitting issue of QDA will become minor.

(d) False. When the true data is linear, the test error rate will rise due to QDA is more likely to course over-fitting.

6. Suppose we collect data for a group of students in a statistics class with variables $X_1 =$ hours studied, $X_2 =$ undergrad GPA, and $Y =$ receive an A. We fit a logistic regression and produce estimated coefficient, $\hat{\beta}_0 = -6, \hat{\beta}_1 = 0.05, \hat{\beta}_2 = 1$

   (a) Estimate the probability that a student who studies for 40h and has an undergrad GPA of 3.5 gets an A in the class.

   (b) How many hours would the student in part (a) need to study to have a 50 % chance of getting an A in the class?

Answer:

(a)

$$f(x) = \frac{1}{1+e^{-\beta^T x}} = \frac{1}{1+e^{-(-6+0.05x_1+x_2)}} = \frac{1}{1+e^{-(-6+0.05\times 40+3.5)}} = 0.3775$$

(b)

$$\frac{1}{1+e^{2.5-0.05x_1}} = 0.5, x_1 = 50$$

7. Suppose that we wish to predict whether a given stock will issue a dividend this year ("Yes" or "No") based on X, last year's percent profit. We examine a large number of companies and discover that the mean value of X for companies that issued a dividend was $\bar{X} = 10$, while the mean for those that didn't was $\bar{X} = 0$. In addition, the variance of X for these two sets of companies was $\bar{\sigma}^2 = 36$. Finally, 80 % of companies issued dividends. Assuming that X follows a normal distribution, predict the probability that a company will issue a dividend this year given that its percentage profit was $X = 4$ last year.

Answer:

$$\hat{\sigma}^2 = 36$$

$$\hat{\mu}_1 = 10 \qquad \hat{\mu}_0 = 0$$

$$\hat{\pi}_1 = 0.8 \qquad \hat{\pi}_0 = 0.2$$

$$\hat{f}_1(x=4) = \frac{1}{\sqrt{2\pi}\hat{\sigma}} e^{-\frac{1}{2\hat{\sigma}^2}(x-\hat{\mu}_1)^2} = \frac{1}{\sqrt{2\pi}\times 6} e^{-\frac{1}{2\times 36}(4-10)^2} = 0.0403$$

$$\hat{f}_2(x=4) = \frac{1}{\sqrt{2\pi}\hat{\sigma}} e^{-\frac{1}{2\hat{\sigma}^2}(x-\hat{\mu}_2)^2} = \frac{1}{\sqrt{2\pi}\times 6} e^{-\frac{1}{2\times 36}(4-0)^2} = 0.0532$$

$$\hat{P}(Y=1|X=4) = \frac{\hat{f}_1(x)\hat{\pi}_1}{\hat{f}_0(x)\hat{\pi}_0 + \hat{f}_1(x)\hat{\pi}_1} = \frac{0.0403\times 0.8}{0.0403\times 0.8 + 0.0532\times 0.2} = 0.752$$

8. Suppose that we take a data set, divide it into equally-sized training and test sets, and then try out two different classification procedures. First we use logistic regression and get an error rate of 20 % on the training data and 30 % on the test data. Next we use 1-nearest neighbours (i.e. K = 1) and get an average error rate (averaged over both test and training data sets) of 18 %. Based on these results, which method should we prefer to use for classification of new observations? Why?

Answer:

For KNN if $K = 1$, the model does not exist training error rate. Therefore the error rate for KNN(k=1) test sets equals to 36% (18% × 2). That means logistic regression with test error rate 30% performs better than KNN.

9. This problem has to do with odds.

(a) On average, what fraction of people with an odds of 0.37 of defaulting on their credit card payment will in fact default?

(b) Suppose that an individual has a 16% chance of defaulting on her credit card payment. What are the odds that she will default?

Answer:

(a)

$$\frac{p(x)}{1-p(x)} = 0.37$$
$$p(x) = 0.27$$

We have on average 27% of people defaulting on their credit card payment.

(b)

$$\frac{p(x)}{1-p(x)} = \frac{0.16}{1-0.16} = 0.19$$

The odds that she will default is then 19%.

4.5 Discussion

1. What is the purpose of partial differential on Newton's update?
2. What is the definition of Maximum a posteriori estimation(MAP)?
3. What is Fisher linear discriminant function ?

Reference

張翔 (2012),《提綱契領學統計》, 四版, 鼎茂圖書

Friedman, J., Hastie, T., & Tibshirani, R. (2001). *The elements of statistical learning* (Vol. 1). Springer, Berlin: Springer series in statistics.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning* (Vol. 6). New York: springer.

Wasserman, L. (2013). *All of statistics: a concise course in statistical inference*. Springer Science & Business Media.

J Li, Linear Discriminant Analysis, `http://sites.stat.psu.edu/~jiali/course/stat597e/notes2/lda.pdf`