STAT406- Methods of Statistical Learning Lecture 13

Matias Salibian-Barrera

UBC - Sep / Dec 2019

Classification as prediction

- In general, we have n observations (training)
- $(g_1, \mathbf{x}_1), (g_2, \mathbf{x}_2), \ldots, (g_n, \mathbf{x}_n)$
- we would like to build a classifier, a function $\hat{g}(\mathbf{x})$ to predict the true class g of a future observation $(g, \mathbf{x})$ (for which g is unknown)

Classification as prediction

- In general, there are K possible classes, c_1, c_2, ..., c_K. In other words $g \in \{c_1, c_2, \ldots, c_K\}$
- Consider the following loss function

$$L(a, b) = \begin{cases} 0 & \text{if } a = b \\ 1 & \text{if } a \neq b \end{cases}$$

Classification as prediction

- Find a classifier $\hat{g}(\mathbf{x})$ such that

$$E_{(G,\mathbf{X})}\left[L\left(G,\hat{g}(\mathbf{x})\right)\right] \le E_{(G,\mathbf{X})}\left[L\left(G,h(\mathbf{X})\right)\right]$$

for any other function h

$$E_{(G,\mathbf{X})}\left[L\left(G,\hat{g}(\mathbf{X})\right)\right] = E_{\mathbf{X}}\left\{E_{G|\mathbf{X}}\left[L\left(G,\hat{g}(\mathbf{X})\right)\right]\right\}$$
$$= E_{\mathbf{X}}\left\{\sum_{j=1}^{K} L\left(c_j,\hat{g}(\mathbf{X})\right) P\left(G=c_j \middle| \mathbf{X}\right)\right\}$$

Classification as prediction

- It is sufficient to find $\hat{g}(\mathbf{X})$ that minimizes

$$\sum_{j=1}^{K} L\left(c_j, \hat{g}(\mathbf{X})\right) P\left(G = c_j \middle| \mathbf{X}\right)$$

$$= \sum_{c_j \neq \hat{g}(\mathbf{X})} P\left(G = c_j \middle| \mathbf{X}\right)$$

$$= 1 - P\left(G = \hat{g}(\mathbf{X}) \middle| \mathbf{X}\right)$$

- Hence, the optimal classifier satisfies

$$P\left(G = \hat{g}(\mathbf{X}) \middle| \mathbf{X}\right) \geq P\left(G = c_j \middle| \mathbf{X}\right) \quad \text{for all } c_j$$

More than 2 groups

- In other words, $\hat{g}(\mathbf{X})$ should be the class with the highest probability

$$\hat{g}(\mathbf{X}) \;=\; \arg \max_{\mathbf{g} \in \{c_1, \ldots, c_K\}} P\Big(G = \mathbf{g} \,|\, \mathbf{X} \Big)$$

- "Assign $\mathbf{X}$ to the class with largest conditional (posterior) probability given $\mathbf{X}$"

Classification as prediction

- Most classifiers can be thought of as different ways to estimate or model

$$\mathbf{f}_{\mathbf{j}}(\mathbf{x}) = P\left(G = \mathbf{c}_j \,\middle|\, \mathbf{X} = \mathbf{x}\right)$$

- For example, logistic classifiers propose a model for $\mathbf{f}_j$:

$$\mathbf{f}_{\mathbf{j}}(\mathbf{x}) = \frac{\exp\left(\beta_{\mathbf{j}}\,\mathbf{x}\right)}{1 + \exp\left(\beta_{\mathbf{j}}\,\mathbf{x}\right)}$$

Classification as prediction

- Vaso example - Logistic linear model
- Data $(y_1, \mathbf{x}_1)$, $(y_2, \mathbf{x}_2)$, . . . , $(y_n, \mathbf{x}_n)$
- $y_j = 0, 1$, $\mathbf{x} = (\text{rate}, \text{volume})'$
- A possible model is

$$P\left(y_j = 1 \middle| \mathbf{x}_j\right) = \frac{\exp\left(\beta' \mathbf{x}_j\right)}{1 + \exp\left(\beta' \mathbf{x}_j\right)}$$

Classification as prediction

- We can estimate β using MLE
- Function `glm` in `R`
- Given values of `rate` and `volume` we predict a 1 if

$$\hat{P}\left(y_j = 1 \middle| \text{rate}, \text{volume}\right) \;>\; 0.5$$

Classification as prediction

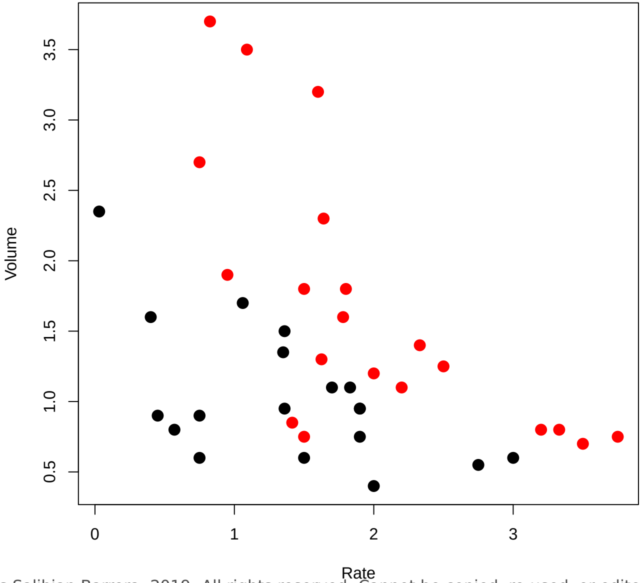
- These conditional (posterior) probabilities

$$P\left(G = \mathbf{c}_j \,\middle|\, \mathbf{X} = \mathbf{x}\right)$$

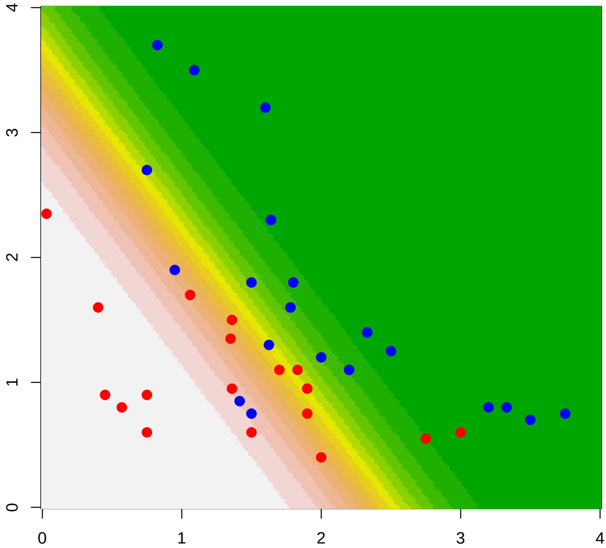
can also be used

- to quantify uncertainty in the classification for a particular value of $\mathbf{x}$
- to identify regions of the feature space where classification isn't so clear

Example - Vaso data



Logistic based probabilities



A model for $\mathbf{X}|g$

If we **model** the feature **distribution** in each **group**:

$$f\left(\mathbf{X}|\, G = c_{\mathbf{k}}\right) = f_{\mathbf{k}}\left(\mathbf{X}\right) \qquad \mathbf{k} = 1, \ldots, \mathbf{K}$$

then

$$P\left(G = c_{\mathbf{k}}|\, \mathbf{X}\right) = \frac{f\left(\mathbf{X}|\, G = c_{\mathbf{k}}\right) p_{\mathbf{k}}}{f\left(\mathbf{X}\right)} = \frac{f_{\mathbf{k}}\left(\mathbf{X}\right) p_{\mathbf{k}}}{f\left(\mathbf{X}\right)}$$

thus

$$\hat{\mathbf{g}}\left(\mathbf{X}\right) = \arg \max_{1 \leq \mathbf{k} \leq \mathbf{K}} f_{\mathbf{k}}\left(\mathbf{X}\right) p_{\mathbf{k}}$$

A model for $\mathbf{X}|g$

For example, we can assume that

$$\mathbf{X}|\, G = c_{\mathbf{k}} \sim \mathcal{N}\left(\mu_{\mathbf{k}}, \boldsymbol{\Sigma}\right)$$

then, we can estimate

$$\hat{f}_{\mathbf{k}}\left(\mathbf{X}\right) \sim \mathcal{N}\left(\hat{\mu}_{\mathbf{k}}, \widehat{\boldsymbol{\Sigma}}\right)$$

using the sample mean of each group and the pooled sample covariance matrix.

We can then find the class $\mathbf{k}$ that has the largest $\hat{f}_{\mathbf{k}}(\mathbf{X})\, p_{\mathbf{k}}$

Gaussian populations

Note that if $f_j \sim \mathcal{N}_p\left(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}\right)$, $j = 1, 2$

$$f_1(\mathbf{x})\, p_1 \ > \ f_2(\mathbf{x})\, p_2 \quad \Leftrightarrow$$

$$\log\left(\frac{f_1(\mathbf{x})\, p_1}{f_2(\mathbf{x})\, p_2}\right) > 0 \quad \Leftrightarrow$$

$$\mathbf{a}'\mathbf{x} + b \ > \ 0$$

for some $\mathbf{a} \in \mathbb{R}^p$ and $b \in \mathbb{R}$.

In other words, boundaries between classes are **linear**.

Gaussian populations

We can also write this in term of class probabilities

$$\frac{P\left(G = c_1 \middle| \mathbf{X}\right)}{P\left(G = c_2 \middle| \mathbf{X}\right)} > 1 \quad \Leftrightarrow \quad f_1(\mathbf{x})\, p_1 \; > \; f_2(\mathbf{x})\, p_2$$

$$\Leftrightarrow \quad \log\left(\frac{f_1(\mathbf{x})\, p_1}{f_2(\mathbf{x})\, p_2}\right) > 0 \quad \Leftrightarrow \quad \mathbf{a}'\mathbf{x} + b \; > \; 0$$

Gaussian populations

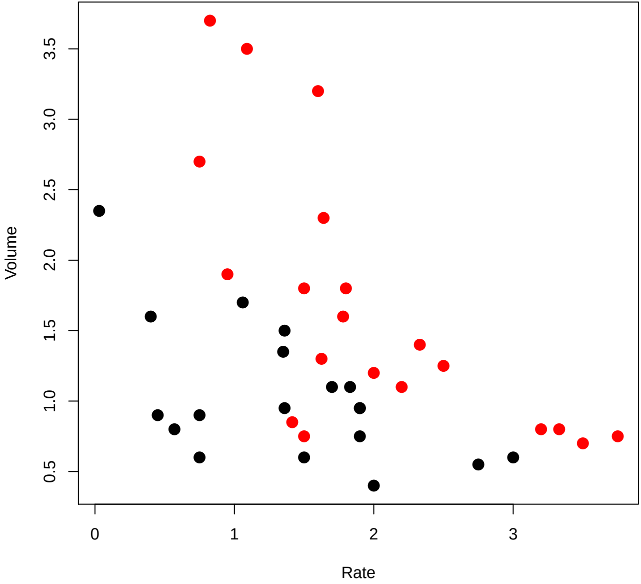
In fact, for normally distributed features we have

$$\log\left(\frac{P\left(G = c_1 \middle| \mathbf{X}\right)}{P\left(G = c_2 \middle| \mathbf{X}\right)}\right) =$$

$$\log\left(\frac{P\left(G = c_1 \middle| \mathbf{X}\right)}{1 - P\left(G = c_1 \middle| \mathbf{X}\right)}\right) = \mathbf{a}'\mathbf{x} + b$$

With two classes, we have also estimated **a** and *b* using logistic regression.

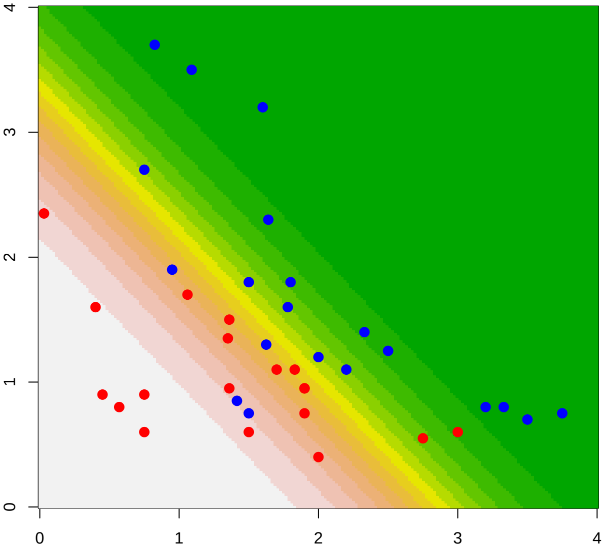
Example - Vaso data



Example - Vaso data

```
library(MASS)

data(vaso, package='robustbase')

plot(Volume ~ Rate, pch=19, col=c('red', 'blue')[Y+1],
      data=vaso, cex=1.3)

a.lda <- lda(Y ~ Volume + Rate, prior = c(.5, .5),
     data=vaso)

aa <- seq(0, 4, length=200)
bb <- seq(0, 4, length=200)
dd <- expand.grid(aa, bb)
names(dd) <- c('Volume', 'Rate')

pr.lda <- predict(a.lda, newdata=dd)$posterior[,1]
image(aa, bb, matrix(pr.lda, 200, 200),
     col=terrain.colors(15), xlab='', ylab='')
points(Volume ~ Rate, pch=19, col=c('red', 'blue')[Y+1],
     data=vaso, cex=1.3)
```

Gaussian-based probabilities



Example - Vaso data

- Note that if we do not assume Gaussian features but insist that

$$\log\left(\frac{P\left(G=1|\,\mathbf{X}\right)}{P\left(G=2|\,\mathbf{X}\right)}\right) =$$

$$\log\left(\frac{P\left(G=1|\,\mathbf{X}\right)}{1-P\left(G=1|\,\mathbf{X}\right)}\right) =$$

$$\mathbf{a}'\mathbf{x}+b$$

we can use `glm` to estimate $\hat{\mathbf{a}}$ and $\hat{b}$:

$\hat{\mathbf{a}} = (-3.88, -2.65)'$ and $\hat{b} = 9.53$

Logistic-based probabilities

```
data(vaso, package='robustbase')

a <- glm(Y ~ Volume + Rate, data=vaso, family=binomi

aa <- seq(0, 4, length=200)
bb <- seq(0, 4, length=200)
dd <- expand.grid(aa, bb)
names(dd) <- c('Volume', 'Rate')

yy <- predict(a, newdata=dd, type='response')

image(aa, bb, matrix(1-yy, 200, 200),
     col=terrain.colors(15), xlab='', ylab='')
points(Volume ~ Rate, pch=19, col=c('red', 'blue')[Y
     data=vaso, cex=1.3)
```

Logistic-based probabilities