

STAT406- Methods of Statistical Learning Lecture 13

Matias Salibian-Barrera

UBC - Sep / Dec 2019

Classification as prediction

- In general, we have n observations (training)
- $(g_1, \mathbf{x}_1), (g_2, \mathbf{x}_2), \dots, (g_n, \mathbf{x}_n)$
- 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, \dots, c_K . In other words $g \in \{c_1, c_2, \dots, c_K\}$
- 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

$$E_{(G,\mathbf{x})} [L (G, \hat{g}(\mathbf{x}))] \leq E_{(G,\mathbf{x})} [L (G, h(\mathbf{x}))]$$

for any other function h

$$\begin{aligned} E_{(G,\mathbf{x})} [L (G, \hat{g}(\mathbf{x}))] &= E_{\mathbf{x}} \{ E_{G|\mathbf{x}} [L (G, \hat{g}(\mathbf{x}))] \} \\ &= E_{\mathbf{x}} \left\{ \sum_{j=1}^K L (c_j, \hat{g}(\mathbf{x})) P (G = c_j | \mathbf{x}) \right\} \end{aligned}$$

Classification as prediction

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

$$\begin{aligned} \sum_{j=1}^K L(c_j, \hat{g}(\mathbf{X})) P(G = c_j | \mathbf{X}) \\ = \sum_{c_j \neq \hat{g}(\mathbf{X})} P(G = c_j | \mathbf{X}) \\ = 1 - P(G = \hat{g}(\mathbf{X}) | \mathbf{X}) \end{aligned}$$

- Hence, the optimal classifier satisfies

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

More than 2 groups

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

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

- “Assign \mathbf{X} to the class with largest posterior probability given \mathbf{X} ”

Classification as prediction

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

$$\mathbf{f}_{\mathbf{j}}(\mathbf{x}) = P\left(G = \mathbf{c}_{\mathbf{j}} \mid \mathbf{X} = \mathbf{x}\right)$$

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

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

Classification as prediction

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

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

Classification as prediction

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

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

Classification as prediction

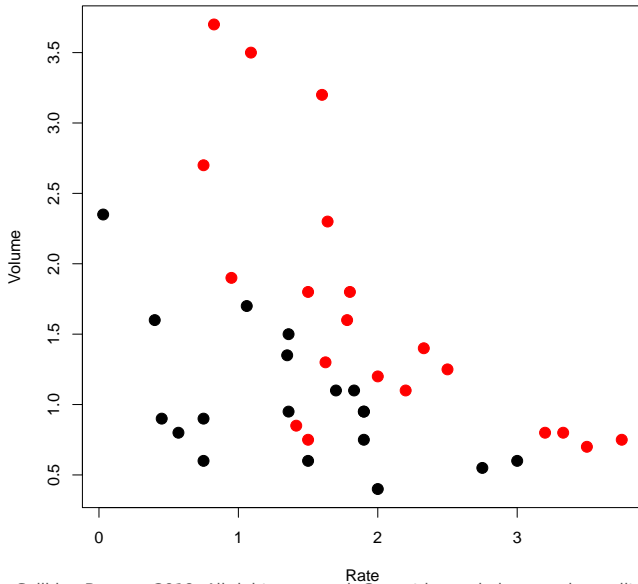
- These posterior probabilities

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

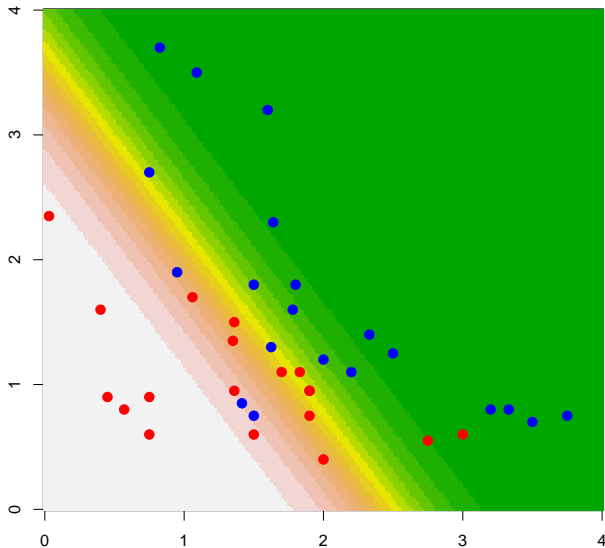
can also be used

- 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**:

$$f(\mathbf{X} | G = c_{\mathbf{k}}) = f_{\mathbf{k}}(\mathbf{X}) \quad \mathbf{k} = 1, \dots, \mathbf{K}$$

then

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

thus

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

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

For example, we can assume that

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

then, we can estimate

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

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

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

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

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

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

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

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

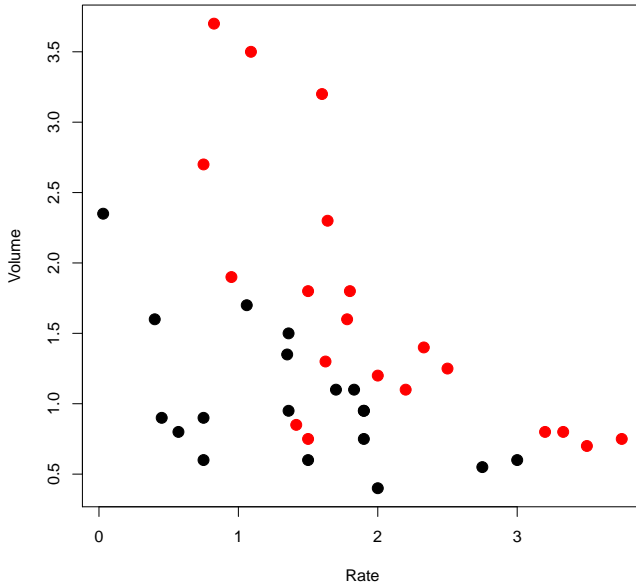
Gaussian populations

In fact, for normally distributed features we have

$$\log \left(\frac{P(G = c_1 | \mathbf{X})}{P(G = c_2 | \mathbf{X})} \right) =$$
$$\log \left(\frac{P(G = c_1 | \mathbf{X})}{1 - P(G = c_1 | \mathbf{X})} \right) = \mathbf{a}'\mathbf{x} + b$$

With two classes, we have also estimated \mathbf{a} and b using logistic regression.

Example - Vaso data



Example - Vaso data

```
library(MASS)

data(vaso, package='robustbase')

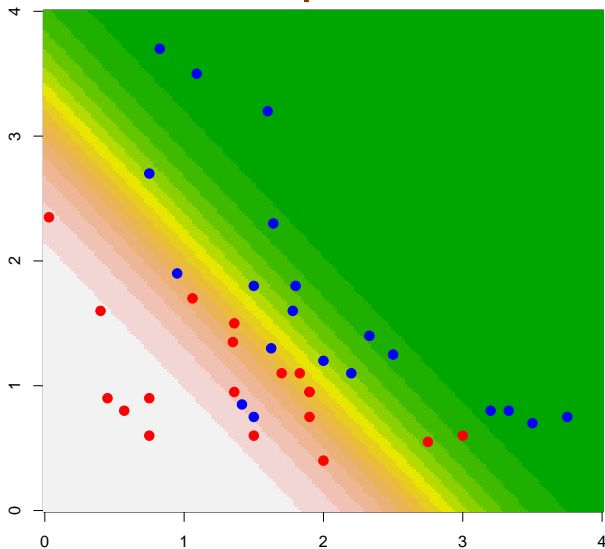
plot(Volume ~ Rate, pch=19, col=c('red', 'blue')[Y+1],
      data=vaso, cex=1.3)

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

$$\log \left(\frac{P(G = 1 | \mathbf{X})}{P(G = 2 | \mathbf{X})} \right) =$$
$$\log \left(\frac{P(G = 1 | \mathbf{X})}{1 - P(G = 1 | \mathbf{X})} \right) =$$
$$\mathbf{a}'\mathbf{x} + b$$

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

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

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

