STU33011 Lab 7

Make sure that you have completed the previous lab sessions (https://www.scss.tcd.ie/~arwhite/Teaching/STU33011.html) before advancing to this one.

In this session, we will

- simulate data from a multivariate normal distribution;
- look at how to implement the $k$-nearest neighbours algorithm;
- investigate how to choose $k$ using training and test sets.

Linear Discriminant Analysis

We can use the `lda` function in the `MASS` package to perform linear discriminant analysis (LDA.) Load the package and have a look at the function’s help file. Later, we will use the `qda` function, also in `MASS` to perform quadratic discriminant analysis (QDA.) Remember from lectures that LDA assumes the data within each group follows a multivariate normal distribution and that each group shares the same covariance matrix.

```r
library(MASS)
```

Exercise

Download the Salmon data set. This is available at: https://www.scss.tcd.ie/~arwhite/Teaching/STU33011/salmon.txt. Read this file under the name `salmon` using the `read.table` command. The first fifty entries are labelled as coming from Alaska, the latter fifty as coming from Canada.

To the plot the data, enter the following:

```r
plot(salmon[,-1], col = as.factor(salmon[,1]))
```
We can do something a little more fancy using the `ellipse` function in package `ellipse`. Can you understand what this code is doing?

```r
library(ellipse)
plot(salmon[,c(2,3)], col = as.factor(salmon[, 1]), xlim=c(50,190), ylim=c(290,530))
lines(ellipse(cov(salmon[c(1:50), c(2, 3)]), centre = colMeans(salmon[c(1:50), c(2, 3)]), level = c(0.5)))
lines(ellipse(cov(salmon[c(51:100), c(2, 3)]), centre = colMeans(salmon[c(51:100), c(2, 3)]), level = 0.5), col = 2)
```

Before we use LDA, we need to split the data into training, and test sets. (Why don’t we need a validation set?) For this example we will try an 80:20 split.

```r
strain <- salmon[c(1:40, 51:90), ] stest <- salmon[c(41:50, 91:100), ]
```

We can then train our classifier:

```r
lsol <- lda(strain[, c(2, 3)], grouping = strain[,1])
```

The above trains the classifier on the second and third columns of the training data `strain` using the classifier information in the first column `strain[, 1]`.

If you enter the following, you will be returned with a list of summary information concerning the computation:

```r
lsol$prior
lsol$means
```

```r
## Alaska Canada
## 0.5 0.5
```

```r
## Freshwater Marine
## Alaska 100.550 422.275
## Canada 138.625 368.650
```

`lsol$prior` provides the prior probability of group membership that is used in the analysis, which by default is taken to be the class proportions in the training data. The second section provides the estimated group means for each of the two groups. You should be able to verify both of these calculations.
Note that the pooled covariance matrix used to perform LDA is not provided. You can find this manually by calculating the following:

\[
\Sigma = \frac{(N_1 - 1)\Sigma_1 + (N_2 - 1)\Sigma_2}{N_1 + N_2 - 2}.
\]

In the above \(\Sigma\) is the estimated common covariance matrix, \(\Sigma_i\) is the estimated covariance matrix for specific group \(i\), whilst \(N_i\) is the number of data points in group \(i\). This formula extends naturally for the case of \(k\) groups:

\[
\Sigma = \frac{\sum_{i=1}^{k}(N_i - 1)\Sigma_i}{\sum_{i=1}^{k}N_i - k}.
\]

To estimate the covariance matrices for both subsets of salmon data, enter the following:

```r
alaska_salmon <- strain[strain == "Alaska", c(2,3)]
canada_salmon <- strain[strain == "Canada", c(2,3)]
n_alaska <- length(alaska_salmon)
n_canada <- length(canada_salmon)
single_cov_num <- ((n_alaska - 1) * cov(alaska_salmon) + (n_canada - 1) * cov(canada_salmon))
single_cov <- single_cov_num / (length(strain[, 1]) - 2)
single_cov
```

## Freshwater Marine
## Freshwater 8.2620891 -0.3909599
## Marine -0.3909599 27.8833251

### Classification

Remember from lectures that the classification rule for LDA is:

\[
\log \left( \frac{P(k|x)}{P(l|x)} \right) = \log \left( \frac{\pi_k}{\pi_l} \right) + \log \left( \frac{f(x|k)}{f(x|l)} \right).
\]

The multivariate normal assumption with common covariance then leads to the following:

\[
\log \left( \frac{f(x|k)}{f(x|l)} \right) = x^T \Sigma^{-1}(\mu_k - \mu_l) - 1/2(\mu_k^T \Sigma^{-1} \mu_k - \mu_l^T \Sigma^{-1} \mu_l)
\]

\[\Rightarrow \log \left( \frac{f(x|k)}{f(x|l)} \right) = \left\{ x - \frac{1}{2}(\mu_k + \mu_l) \right\}^T \Sigma^{-1}(\mu_k - \mu_l).
\]

The term \(\frac{1}{2}(\mu_k + \mu_l)\) gives the average of the group means, and so \(x - 1/2(\mu_k + \mu_l)\) gives the difference of the observation to this value. Assuming the prior probabilities are equal, \((x - \frac{1}{2}(\mu_k + \mu_l))^T \Sigma^{-1}(\mu_k - \mu_l)\) determines the classification by whether it is positive or negative. (In this case, a positive value indicates membership to Group \(k\)).

As well as providing information about prior probabilities and group means, calling `lsol` directly provides information regarding the coefficients of linear discriminants (use `lsol$scaling` to call this directly):

```r
lsol
```

## Call:
## `lda(strain[, c(2, 3)], grouping = strain[, 1])`

## Prior probabilities of groups:
## Alaska Canada
## 0.5 0.5
## Group means:

<table>
<thead>
<tr>
<th></th>
<th>Freshwater</th>
<th>Marine</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alaska</td>
<td>100.550</td>
<td>422.275</td>
</tr>
<tr>
<td>Canada</td>
<td>138.625</td>
<td>368.650</td>
</tr>
</tbody>
</table>

## Coefficients of linear discriminants:

<table>
<thead>
<tr>
<th></th>
<th>LD1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Freshwater</td>
<td>0.04390178</td>
</tr>
<tr>
<td>Marine</td>
<td>-0.01806237</td>
</tr>
</tbody>
</table>

These are a (scaled) version of $\Sigma^{-1}(\mu_k - \mu_l)$, and hence can be used for classifying a new observation. Note that it is the second class, here Canada, that is associated with group $k$ in the output.

### Exercise

- Interpret the output of the `lsol$scaling`. Does it make sense?
- Determine the classification for an observation with a Freshwater recording of 120 and a Marine recording of 380.

To automatically predict this observation enter:

```r
predict(lsol, c(120, 380))
```

```
$\text{class}$
[1] Canada

$\text{Levels: Alaska Canada}$

$\text{posterior}$

<table>
<thead>
<tr>
<th>Alaska</th>
<th>Canada</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3132047</td>
<td>0.6867953</td>
</tr>
</tbody>
</table>

$x$

<table>
<thead>
<tr>
<th>LD1</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2973989</td>
</tr>
</tbody>
</table>
```

To automatically predict the test data set enter:

```r
predict(lsol, stest[, c(2, 3)])
```

```
$\text{class}$

$\text{posterior}$

<table>
<thead>
<tr>
<th>Alaska</th>
<th>Canada</th>
</tr>
</thead>
<tbody>
<tr>
<td>41</td>
<td>0.999934575 6.542453e-05</td>
</tr>
<tr>
<td>42</td>
<td>0.998909821 1.090179e-03</td>
</tr>
<tr>
<td>43</td>
<td>0.999641196 3.588039e-04</td>
</tr>
<tr>
<td>44</td>
<td>0.997267179 2.732821e-03</td>
</tr>
<tr>
<td>45</td>
<td>0.991071121 8.928879e-03</td>
</tr>
</tbody>
</table>
Remember that when \( \pi_k \neq \pi_l \), the term \( \log(\frac{\pi_k}{\pi_l}) \) no longer disappears, and that this should be taken into account by default by the `predict` function.

**Exercise**

- Assess the performance of the LDA classifier on the test set. How much uncertainty is there in relation to the classification of the datapoints?

**Cross-Validation**

Rather than splitting the data into training and test sets (or training, test, and validation sets when different models are being considered), an alternative technique for measuring the performance of the model is to perform cross-validation. For the `lda` function this is achieved by incorporating the argument `CV=TRUE`:
You should now find that the output from entering \texttt{lsolcv} includes a list of how data points were classified when they were the only point left out. Additionally, a matrix of group membership probability is returned.

In order to visualise the performance of LDA under cross-validation we can produce a plot of the following form:

\begin{verbatim}
plot(salmon[, c(2, 3)], col = as.factor(salmon[, 1]), pch = as.numeric(lsol_cv$class))
\end{verbatim}

The above command plots the two numeric variables of the \texttt{salmon} data with colouring being determined by true classification and symbols being determined by the resulting classification of ‘leave-one-out’ LDA. How many misclassified points do you notice?

**Quadratic Discriminant Analysis**

The function \texttt{qda} within the package \texttt{MASS} performs quadratic discriminant analysis. Remember the difference between QDA and LDA is that the former permits each group distribution to have its own covariance matrix, whilst the latter assumes a common covariance matrix for all group distributions. The usage of \texttt{qda} the same as \texttt{lda}:

\begin{verbatim}
qsol <- qda(strain[, c(2,3)], grouping = strain[, 1])
predict(qsol, stest[, c(2, 3)])
\end{verbatim}

## $class
## Levels: Alaska Canada
##
## $posterior
## Alaska Canada
## 41 0.999999603 3.973313e-07
## 42 0.999934364 6.563641e-05

\begin{verbatim}
# $class
# Levels: Alaska Canada
#
# $posterior
# Alaska Canada
# 41 0.999999603 3.973313e-07
# 42 0.999934364 6.563641e-05
\end{verbatim}
Again you will notice in an 80:20 training:testing split we have achieved 100% correct classification. The output returned from by `qsol` provides details of the prior probability of group membership (again determined by the proportion of data points classified in that group by default), and the mean vectors for each group. To find the covariances for the two groups enter the following:

```
cov (alaska_salmon)
```

```
##         Freshwater     Marine
## Freshwater  286.6641  -170.1551
## Marine      -170.1551  1158.5635
```

```
cov (canada_salmon)
```

```
##        Freshwater     Marine
## Freshwater   357.7788   139.6603
## Marine       139.6603  1016.3359
```

**Exercise**

- Compare the covariance matrices of the Alaska and Canada groups. Does the assumption of pooled covariance (LDA) or separate covariance matrices (QDA) seem to be more appropriate?
- Assess the performance of QDA for the `salmon` data set under cross-validation and produce a plot of your results.
- Compare the performance of LDA and QDA under a 50:25:25 training:validation:testing split of the `salmon` data set. This means you should use 50% of the data to train both models, 25% of the data to assess which model appears to be the better classifier, and a further 25% of the data to more accurately assess the true classification rate of the better model.

**Exercise**

- Assess the performance of LDA and QDA for classifying the `iris` data by species.
- Assess the performance of LDA and QDA for classifying the `olive` data by either region of origin, or specific area of origin.