

R Notebook

Ronald W Buie

```
knitr::opts_chunk$set(fig.show = "all", results = "hide")
```

Instructions: You may discuss the homework problems in small groups, but you must write up the .nal solutions and code yourself. Please turn in your code for the problems that involve coding. However, code without written answers will receive no credit. To receive credit, you must explain your answers and show your work. All plots should be appropriately labeled and legible, with axis labels, legends, etc., as needed.

1. In this problem, you will generate data with $p = 2$ features and a qualitative response with $K = 3$ classes, and $n = 50$ observations per class. You will then apply linear discriminant analysis to the data.

(a) Generate data such that the distribution of an observation in the k th class follows a $N(\mu_k, S)$ distribution, for $k = 1, \dots, K$. That is, the data follow a bivariate normal distribution with a mean vector μ_k that is specific to the k th class, and a covariance matrix S that is shared across the K classes. Choose S and μ_1, \dots, μ_K such that there is some overlap between the K classes, i.e. no linear decision boundary is able to perfectly separate the training data. Specify your choices for S and μ_1, \dots, μ_K .

```
set.seed(990)
DOTDISTANCE <- 4
STANDARDDEV <- 3
findit <- function(length,angle,coord) {
  #not sure what I'm doing wrong, but x and y work when reverse, so here goes
  #Error, at 180 degrees, a very small number is reported, instead of 0
  if(coord == "x") {
    toReturn <- length*sin(angle*(pi/180))
  }
  if(coord == "y") {
    toReturn <- length*cos(angle*(pi/180))
  }
  return(toReturn)
}
findit(5,45,"x")

DF <- data.frame(X1 = c(rnorm(50,0,STANDARDDEV),
                       rnorm(50,findit(DOTDISTANCE,120,"x"),STANDARDDEV),
```

```

      rnorm(50,findit(DOTDISTANCE,240,"x"),STANDARDDEV)) ,
X2=  c(rnorm(50,DOTDISTANCE,STANDARDDEV) ,
      rnorm(50,findit(DOTDISTANCE,120,"y"),STANDARDDEV) ,
      rnorm(50,findit(DOTDISTANCE,240,"y"),STANDARDDEV)) ,
Type = (c(rep(1,50),rep(2,50),rep(3,50) ))

```

3 sets of 50 points were created with mean locations equidistant from the origin (0,0) and at even angles (0, 120, and 240 degrees). Each set of 50 points exists 3 standard deviations from their central mean.

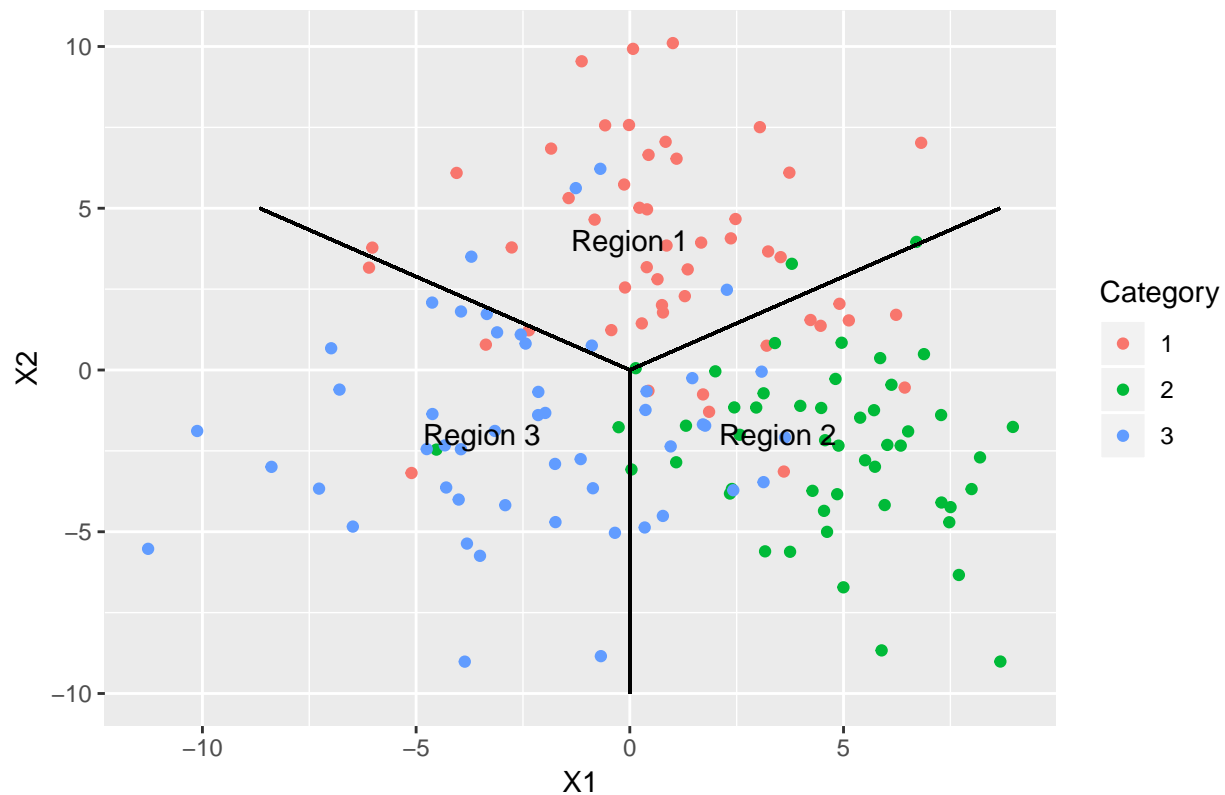
(b) Plot the data, with the observations in each class displayed in a different color. Compute and display the Bayes decision boundary (or Bayes decision boundaries) on this plot. This plot should look something like the right-hand panel of Figure 4.6 in the textbook (although no need to worry about shading the background, and also you don't need to display the LDA decision boundary for this sub-problem ? you will do that in the next sub-problem). Be sure to label which region(s) of the plot correspond to each class.

```

library(ggplot2)
linelength <- 10
ggplot(DF, aes(x = X1,y=X2, color=as.factor(Type))) +
  geom_point() +
  ggtitle("Question 1 Data with Bayes Decision Boundary") +
  labs(colour= "Category")+
  geom_segment(aes(x=0, y=0, xend = 0, yend = findit(linelength,180,"y")), color = "black") +
  geom_segment(aes(x=0, y=0, xend = findit(linelength,300,"x"), yend=findit(linelength,300,"y")), color = "black") +
  geom_segment(aes(x=0, y=0, xend = findit(linelength,60,"x"), yend=findit(linelength,60,"y")), color = "black") +
  annotate(geom = "text", x= 0, y = DOTDISTANCE, label = "Region 1") +
  annotate(geom = "text",x = findit(DOTDISTANCE,120,"x"), y=findit(DOTDISTANCE,120,"y"), label= "Region 2") +
  annotate(geom = "text",x = findit(DOTDISTANCE,240,"x"), y=findit(DOTDISTANCE,240,"y"), label= "Region 3")

```

Question 1 Data with Bayes Decision Boundary



(c) Fit a linear discriminant analysis model to the data, and make a plot that displays the observations as well as the decision boundary (or boundaries) corresponding to this fitted model. How does the LDA decision boundary (which can be viewed as an estimate of the Bayes decision boundary) compare to the Bayes decision boundary that you computed and plotted in (b)?

```
library(MASS)

#this function adopted from Michael Hahsler
#http://michael.hahsler.net/SMU/EMIS7332/R/viz_classifier.html
decisionplot <- function(model, data, class = NULL, predict_type = "class",
  resolution = 100, showgrid = TRUE, ...) {

  if(!is.null(class)) cl <- data[,class] else cl <- 1
  data <- data[,1:2]
  k <- length(unique(cl))

  plot(data, col = as.integer(cl)+1L, pch = as.integer(cl)+1L, ...)

  # make grid
  r <- sapply(data, range, na.rm = TRUE)
  xs <- seq(r[1,1], r[2,1], length.out = resolution)
  ys <- seq(r[1,2], r[2,2], length.out = resolution)
  g <- cbind(rep(xs, each=resolution), rep(ys, time = resolution))
  colnames(g) <- colnames(r)
```

```

g <- as.data.frame(g)

### guess how to get class labels from predict
### (unfortunately not very consistent between models)
p <- predict(model, g, type = predict_type)
if(is.list(p)) p <- p$class
p <- as.factor(p)

if(showgrid) points(g, col = as.integer(p)+1L, pch = ".")

z <- matrix(as.integer(p), nrow = resolution, byrow = TRUE)
contour(xs, ys, z, add = TRUE, drawlabels = FALSE,
        lwd = 2, levels = (1:(k-1))+.5)

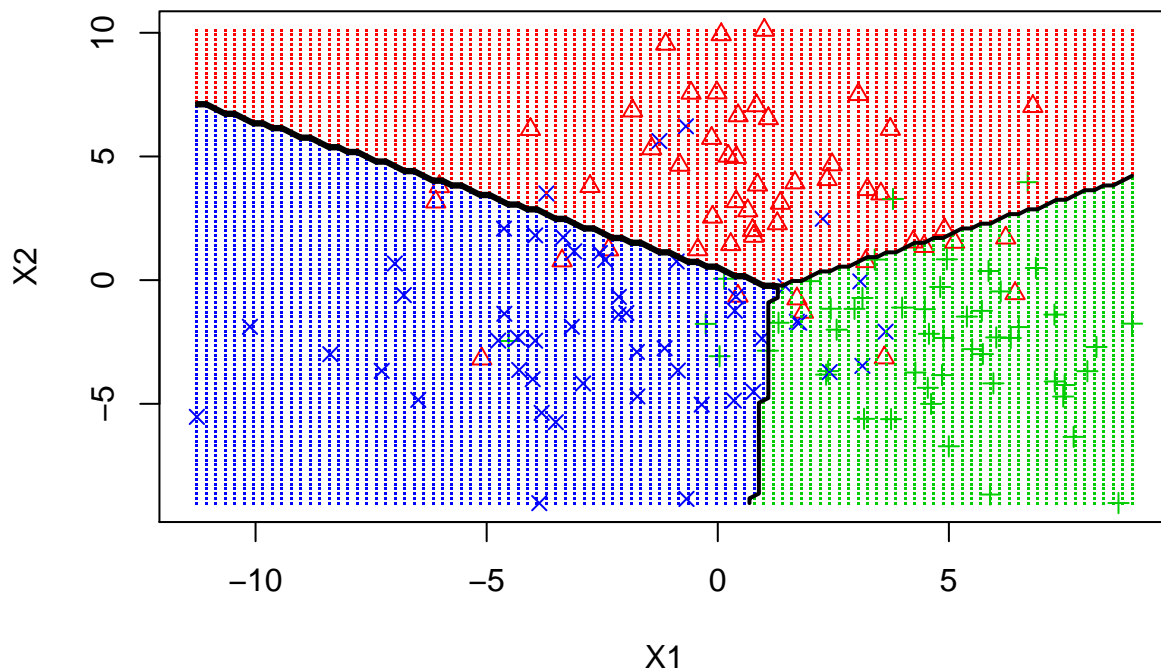
invisible(z)
}

test <- lda(Type~., data = DF)

decisionplot(test, DF, class = "Type", main = "Plotting LDA")

```

Plotting LDA



The LDA decision boundaries are similar, but not geometric. The boundaries on the positive Y axis are less acute relative to each other, and the boundary on the negative y axis is no longer perfectly vertical.

(d) Report the $K \times K$ confusion matrix for the LDA model on the training data. The rows of this confusion matrix represent the predicted class labels, and the columns represent the true class labels. (See Table 4.4 in the textbook for an example of a confusion matrix.) Also, report the training error (i.e. the proportion of observations that are mis-classified.)

```
library(xtable)
DFConfusionMatrix <- table(predict(test, type="class")$class, DF$Type)

xtable::xtable( DFConfusionMatrix)
```

% latex table generated in R 3.5.2 by xtable 1.8-3 package % Mon Jan 28 23:17:48 2019

	1	2	3
1	36	2	4
2	8	43	7
3	6	5	39

- (e) Generate $n = 50$ test observations in each of the K classes, using the bivariate normal distributions from (a). Report the $K \times K$ confusion matrix, as well as the test error, that results from applying the model `.t` to the training data in (c) to your test data.
 - (f) Compare your results from (d) and (e), and comment on your findings.
2. In this problem, you will apply quadratic discriminant analysis to the data from Q1.
 - (a) Fit a quadratic discriminant analysis model to the training data from Q1, and make a plot that displays the observations as well as the QDA decision boundary (or boundaries) corresponding to this fitted model. Be sure to label which region(s) of the plot correspond to each class. How does the QDA decision boundary compare to the Bayes decision boundary that you computed in Q1(b)?
 - (b) Report the $K \times K$ confusion matrix for the QDA model on the training data, as well as the training error.
 - (c) Repeat (b), but this time using the test data generated in Q1. (That is, apply the model `.t` to the training data in (a) in order to calculate the test error.)
 - (d) Compare your results in (b) and (c), and comment on your findings.
 - (e) Which method had smaller training error in this example: LDA or QDA? Comment on your findings.
 - (f) Which method had smaller test error in this example: LDA or QDA? Comment on your findings.
 3. In this example, you will generate data in such a way that QDA will have a smaller test error than LDA. Once again, take $K = 3$, $p = 2$, and $n = 50$ observations per class.
 - (a) Generate data such that the observations in the k th class follow a $N(\mu_k, S_k)$ distribution, for $k = 1, \dots, K$. This is similar to the data generated in Q1, except that now each class has its own covariance matrix. Choose μ_1, \dots, μ_K and S_1, \dots, S_K so that the Bayes decision boundary (or boundaries) is highly non-linear, and also so that there is some overlap between the K classes. Specify your choices for S_1, \dots, S_K and μ_1, \dots, μ_K .
 - (b) Plot the data, with the observations in each class displayed in a different color. Compute and display the Bayes decision boundary (or Bayes decision boundaries) on this plot. Be sure to label which region(s) of the plot correspond to each class.
 - (c) Fit an LDA model to the data, and also fit a QDA model to the data. Make a plot that displays the observations in each class in a different color, as well as the LDA decision boundary and the QDA decision boundary. Be sure to label which region(s) of the plot correspond to each class.

- (d) Report the confusion matrix for the training data, as well as the training error, for each of the models in (c).
- (e) Now generate $n = 50$ test observations per class, using the bivariate normal distributions from (a). Report the $K \times K$ confusion matrices, as well as the test errors, that result from applying the models trained in (c) on the training data to your test data.
- (f) Which method had smaller training error in this example: LDA or QDA? Comment on your findings.
- (g) Which method had smaller test error in this example: LDA or QDA? Comment on your findings.
4. In this problem, you will perform binary classification on a real (not simulated) data set of your choice (from the book website, from the internet, or from another source). In order to perform binary classification, you will of course need a qualitative response variable with values in $K = 2$ classes. Let n_k denote the number of observations in the k th class, for $k = 1, 2$. You will want to have $n_k \geq p$ for $k = 1, 2$ in this data set, so if you have chosen a data set with $p = n_k$ or $p < n_k$ for some k , then you should just work with a subset of the features. Also, please choose a data set with $p = 4$.
- (a) Describe the data. Where did you get it from? What is the value of p ? What is the sample size in each class? What are the response and features measuring? Are the features qualitative or quantitative?
- (b) Make some scatterplots displaying Y on the y-axis and X_j on the x-axis, for $j = 1, \dots, p$. Based on these scatterplots, do you think it will be possible to accurately predict Y using X_1, \dots, X_p ? Explain your answer.
- (c) Fit a logistic regression model to predict Y using X_1, \dots, X_p , and make a table like Table 4.3 in the textbook displaying the coefficient estimates, standard errors, and p-values. If any of your predictors are qualitative, then be sure to clearly indicate how to interpret the corresponding coefficients in the table.
- (d) For which predictors (if any) can you reject the null hypothesis $H_0 : \beta_j = 0$, where β_j is the coefficient in the logistic regression model from (c)? Comment on your results, and relate them to your answer in (b).
- (e) Write a sentence providing an interpretation for the coefficient associated with X_1 . For instance, you could say something along the lines of "A one-unit increase in X_1 is associated with" (complete the sentence!).
5. This problem has to do with logistic regression.
- (a) Suppose you fit a logistic regression to some data and find that for a given observation $x = (x_1, \dots, x_p)^T$, the estimated log-odds equals 0.7. What is $P(Y = 1 | X = x)$?
- (b) In the same setting as (a), suppose you are now interested in the observation $x^* = (x_1 + 1, x_2 - 1, 2x_3, x_4, \dots, x_p)^T$. In other words, $x_1^* = x_1 + 1$,

$x_2^* = x_2 - 1$, $x_3^* = 2x_3$, and $x_j^* = x_j$ for $j = 4, \dots, p$. Write out a simple expression for $P(Y = 1 | X = x^*)$. Your answer will involve the estimated coefficients in the logistic regression model, as well as the number 0.7.

6. Suppose we wish to perform classification of a binary response in a setting with $p = 1$: that is, $X \in \mathbb{R}$, and $Y \in \{1, 2\}$. We assume that the observations in Class 1 are drawn from a $N(\mu, s^2)$ distribution, and that the observations in Class 2 are drawn from an $\text{Uniform}[-2, 2]$ distribution. (a) Derive an expression for the Bayes decision boundary: that is, for the set of x such that $P(Y = 1 | X = x) = P(Y = 2 | X = x)$. Write it out as simply as you can.

- (b) Suppose (for this sub-problem only) that $\mu = 0$, $s = 1$, $p_1 = 0.45$ (here, p_1 is the prior probability that an observation belongs to Class 1). Describe the Bayes classifier in this case: what range of x values will get assigned to Class 1, and what range of x values will get assigned to Class 2? Write out your answer as simply as you can. Draw a picture showing the set of x values assigned to Class 1 and the set of x values assigned to Class 2.

(c) Now suppose we observe n training observations $(x_1, y_1), \dots, (x_n, y_n)$. Explain how you could use these observations to estimate μ , s , and p_1 (instead of using the values that were given in part (b)).

(d) Given a test observation $X = x_0$, provide an estimate of

$P(Y = 1 \mid X = x_0)$. Your answer should involve only the training observations $(x_1, y_1), \dots, (x_n, y_n)$ and the test observation x_0 , and no unknown parameters.