da410\_assignment5

Jason Grahn

2/3/2019

Chapter 9 Page 336-337:#9.7 (a)(b)(c) 9.7 c, don’t need to generate classification table, just use nearest neighbor method to assign 3rd vector of both two matrix into group1 or group2. For the detail, look at the example.

You can use the data1.txt and data2.txt to read into data.

Calculate step by step (you can use R to do matrix calculation, DO NOT use knn functions), make sure to provide intermediate steps.

# 9.7 Do a classification analysis on the beetle data in Table 5.5 as follows:

beetles <- read.table(here::here("assignment05/T5\_5\_FBEETLES.DAT")) %>%   
 rename(experiment\_number = V1,  
 species = V2,  
 y1 = V3,  
 y2 = V4,  
 y3 = V5,  
 y4 = V6) %>%   
 mutate(species = factor(species,   
 levels = c(1, 2)))  
  
head(beetles,5)

## experiment\_number species y1 y2 y3 y4  
## 1 1 1 189 245 137 163  
## 2 2 1 192 260 132 217  
## 3 3 1 217 276 141 192  
## 4 4 1 221 299 142 213  
## 5 5 1 171 239 128 158

## (a) Find the cutoff point THEN the classification function.

Cutoff point = Classification function =

# group by species  
beet1 <- subset(x = beetles, subset = (species == 1), y1:y4)  
beet2 <- subset(x = beetles, subset = (species == 2), y1:y4)  
  
# find group means for all variables for each species  
mean1 <- apply(X = beet1, FUN = mean, MARGIN = 2)  
mean1

## y1 y2 y3 y4   
## 194.4737 267.0526 137.3684 185.9474

mean2 <- apply(X = beet2, FUN = mean, MARGIN = 2)  
mean2

## y1 y2 y3 y4   
## 179.55 290.80 157.20 209.25

# find covariance matrix for each species  
var1 <- var(beet1)  
var1

## y1 y2 y3 y4  
## y1 187.59649 176.86257 48.37135 113.58187  
## y2 176.86257 345.38596 75.97953 118.78070  
## y3 48.37135 75.97953 66.35673 16.24269  
## y4 113.58187 118.78070 16.24269 239.94152

var2 <- var(beet2)  
var2

## y1 y2 y3 y4  
## y1 101.83947 128.06316 36.98947 32.59211  
## y2 128.06316 389.01053 165.35789 94.36842  
## y3 36.98947 165.35789 167.53684 66.52632  
## y4 32.59211 94.36842 66.52632 177.88158

# take a pause here. Check for equal covariances between groups (species)  
 # If ==, then proceed; if != then we need a different procedure  
# heplots::boxM(cbind(y1, y2, y3, y4) ~ species, data = beetles)  
# can't use heplots because samplesize is too small. using hotelling   
beetles.manova <- stats::manova(cbind(y1, y2, y3, y4) ~ species,   
 data = beetles)  
  
tidy.beetles.manova <- broom::tidy(beetles.manova,   
 test = "Hotelling-Lawley",  
 intercept = FALSE)  
tidy.beetles.manova

## # A tibble: 2 x 7  
## term df hl statistic num.df den.df p.value  
## <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 species 1 3.61 30.7 4 34 7.52e-11  
## 2 Residuals 37 NA NA NA NA NA

#### because hotelling t-test has p-value <0.001, we conclude equal covariance

# find \_pooled\_ sample variance for each species  
#length of each set  
length1 <- nrow(beet1)  
length2 <- nrow(beet2)  
  
#variance of each set  
pvar1 <- (length1 - 1) \* var1  
pvar2 <- (length2 - 1) \* var2   
  
#use length and variance to get pooled variance  
sp1 <- 1 / (length1 + length2 - 2) \* (pvar1 + pvar2)  
sp1

## y1 y2 y3 y4  
## y1 143.55910 151.8034 42.52660 71.99253  
## y2 151.80341 367.7878 121.87653 106.24467  
## y3 42.52660 121.8765 118.31408 42.06401  
## y4 71.99253 106.2447 42.06401 208.07290

# find the cutoff point (this is half the difference between group means)   
cutoff <- .5\*(mean1 - mean2) %\*% solve(sp1) %\*% (mean1 + mean2)  
cutoff <- round(cutoff,2)  
cutoff

## [,1]  
## [1,] -15.81

# find the classification function  
species\_prediction <- apply(beetles[,3:6], 1, function(y) {  
 z <- (mean1 - mean2) %\*% solve(sp1) %\*% y })  
  
species\_prediction

## [1] -4.640975 -12.769954 -3.599007 -8.333783 -8.398459 -8.319691  
## [7] -5.998789 -7.104974 -14.269659 -8.795039 -5.377660 -6.685556  
## [13] -7.531402 -11.533464 -10.922504 -11.168223 -7.646623 -12.531815  
## [19] -14.524616 -26.822874 -17.458688 -26.373153 -24.334824 -25.694962  
## [25] -26.666573 -27.486750 -15.047912 -21.363037 -23.600898 -28.771377  
## [31] -22.022042 -20.074530 -26.994104 -21.101538 -23.623558 -16.050876  
## [37] -23.304407 -17.341090 -18.974629

## (b) Find the classification table using the linear classification function in part (a).

beetles <- beetles %>%   
 #bring the species prediction data into the data frame  
 mutate(species\_prediction = species\_prediction,  
 #bring the cutoff data into the data frame  
 cutoff = cutoff,  
 #use the cutoff to determine the classification  
 classification = if\_else(species\_prediction >= cutoff, 1, 2))  
  
table(beetles$species, beetles$classification)

##   
## 1 2  
## 1 19 0  
## 2 1 19

## (c) Describe how you use KNN thought to solve it instead of writing out R code or doing any real calculation.

Start by selecting a data point in the first group. Calculating a “normalized” distance from data point 1 in Group1 to all other points in both groups. This is *typically* Euclidian distance. Select the *k*-value. If the number of groups is even, this has to be an odd number greater than the amount of groups. If the number of groups is odd, you can use this number. The algorithm will determine who are the **nearest neightbors** to a new data point. Whichever group has the highest number of neighbors to the new data point is considered the majority group. This is the group the new data point is assigned to.