Figures for Chapter 7

John H Maindonald

October 28, 2012

```
fig7.1 <- function(plotit=TRUE){
    library(MASS)
    fgl.lda <- lda(type ~ ., data=fgl)
    scores <- predict(fgl.lda)$x
    library(lattice)
    gph <- xyplot(scores[,2] ~ scores[,1], groups=fgl$type,
                  xlab="Discriminant 1", ylab="Discriminant 2",
                  aspect=1, scales=list(tck=0.4),
                  auto.key=list(columns=3),
                  par.settings=simpleTheme(alpha=0.6, pch=1:6))
    gph
}

fig7.2 <- function(){
    gph <- xyplot(length ~ breadth, groups=species, data=cuckoos,
                  type=c("p"), auto.key=list(space="right"), aspect=1,
                  scales=list(tck=0.5), par.settings=simpleTheme(pch=16))
    library(latticeExtra)  # This package has the function layer()
    LDmat <- cuckoos.lda$scaling
    ld1 <- LDmat[,1]
    ld2 <- LDmat[,2]
    gm <- sapply(cuckoos[, c("length", "breadth")], mean)
    av1 <- gm[1] + ld1[2]/ld1[1]*gm[2]
    av2 <- gm[1] + ld2[2]/ld2[1]*gm[2]
    addlayer <- layer(panel.abline(av1, -ld1[2]/ld1[1], lty=1),
                      panel.abline(av2, -ld2[2]/ld2[1], lty=2))
    gph + addlayer
}

fig7.3 <- function(){
    ## This will show decision boundaries
    gph <- xyplot(length ~ breadth, groups=species, data=cuckoos,
                  type=c("p"), auto.key=list(space="right"), aspect=1,
                  scales=list(tck=0.5), par.settings=simpleTheme(pch=16))
    x <- pretty(cuckoos$breadth, 20)
    y <- pretty(cuckoos$length, 20)
    Xcon <- expand.grid(breadth=x, length=y)
    cucklda.pr <- predict(cuckoos.lda, Xcon)$posterior
    cuckqda.pr <- predict(cuckoos.qda, Xcon)$posterior
    m <- match("wren", colnames(cucklda.pr))
    ldadiff <- apply(cucklda.pr, 1, function(x)x[m]-max(x[-m]))
    qdadiff <- apply(cuckqda.pr, 1, function(x)x[m]-max(x[-m]))
    addlayer1 <- as.layer(contourplot(ldadiff ~ breadth*length,
                                      at=c(-1,0,1), labels=c("", "lda",""),
                                      label.style="flat",
                                      data=Xcon), axes=FALSE)
    addlayer2 <-as.layer(contourplot(qdadiff ~ breadth*length,
                                     at=c(-1,0,1), labels=c("", "qda",""),
                                     label.style="flat",
                                     data=Xcon), axes=FALSE)
    gph + addlayer1 + addlayer2
}

fig7.4 <- function(seed=47){
    b.rpart <- rpart(rfac ~ cig+poll, data=bronchit)
    plot(b.rpart, uniform=TRUE)
    text(b.rpart, xpd=TRUE)
      ## xpd=TRUE allows labels to extend outside of figure region
}

fig7.5 <- function(){
    b001.rpart <- rpart(rfac ~ cig+poll, cp=0.001, minsplit=15,
                    data=bronchit)
    plotcp(b001.rpart)
}

fig7.6 <-
function () {
plot.root <- function(text='Reduction in "error" (Gini) = 20.55',
                      cutoff="cig<4.375", left="138/11", rt="28/35",
                      xlef=0.15, xrt=0.85,
                      treetop=0.85, treebot=0.1){
    par(mar=rep(0,4))
    plot(0:1, 0:1, axes=F, xlab="",ylab="", type="n")
    lines(c(xlef,xlef, xrt,xrt), c(.1,treetop,treetop,.1))
    lines(c(.5,.5),c(-0.01,0.01)+treetop)
    chh <- strheight("O")
    text(.5, treetop+chh, cutoff)
    text(c(xlef,xrt), rep(.1-chh,2), c(left,rt))
    legend(x=0.5, y=1, xjust=0.5, yjust=1, xpd=TRUE,
       legend=text, bg='gray')
}
```

```
    par(fig=c(0,0.5,0,1))
    plot.root(text='Decrease in "error" = 20.55',
              cutoff="cig<4.375", left="138/11", rt="28/35",
              treetop=0.6, treebot=0.1)
    par(fig=c(0.5,1,0,1), new=TRUE)
    plot.root(text='Decrease in "error" = 2.90',
              cutoff="poll<58.55", left="98/16", rt="68/30",
              treetop=0.6, treebot=0.1)
}

fig7.7 <-
function ()
{
    set.seed(31)   # Reproduce the trees shown
    oldpar <- par(mfrow=c(3,3))
    num <- 1:nrow(bronchit)
    for(i in 1:9){
        useobs <- sample(num, replace=TRUE)
        dset <- bronchit[useobs, ]
        b.rpart <- rpart(rfac ~ cig+poll, data=dset,
                         control=rpart.control(maxdepth=2))
        plot(b.rpart, uniform=TRUE)
        text(b.rpart, xpd=TRUE, cex=1.2)
    }
}

fig7.8 <- function(){
    parset <- simpleTheme(pch=1:2)
    bronchit.rf <- randomForest(rfac ~ cig+poll, proximity=TRUE,
                                data=bronchit)
    points <- cmdscale(1-bronchit.rf$proximity)
    gph <- xyplot(points[,2] ~ points[,1], groups=bronchit$rfac,
                  xlab="Axis 1", ylab="Axis 2",
                  par.settings=parset, aspect=1,
                  auto.key=list(columns=2))
    gph
    }

fig7.9 <- function(){
    form <- paste("~", paste(paste("V", 2:10, sep= ""),
                             collapse="+"))
    gph <- bwplot(formula(paste("Class", form)),
                  scales=list(x="free"),
                  data=Vowel, outer=TRUE, layout=c(3,3))
    gph
}
```

```
compareTargets <-
function(rfobj, prior1, prior2){
    nam1 <- deparse(substitute(prior1))
    nam2 <- deparse(substitute(prior2))
    print(c(nam1,nam2))
    err <- rfobj$confusion[,3]
    err1 <- sum(err*prior1)/sum(prior1)
    err2 <- sum(err*prior2)/sum(prior2)
    errvec <- c(err, err1,err2)
    names(errvec) <- c("error-good", "error-bad", nam1, nam2)
    errvec
  }

bestsize <- function(n0=696, mtry=9, nselect=800,
                     form=CARAVAN ~ ., data=tic0[,-1])
{
    ticm.rf <- randomForest(form, sampsize=c(n0,348),
                            mtry=mtry, data=data)
    nr <- (1:nrow(tic0))[order(ticm.rf$votes[,2],
                           decreasing=T)[1:nselect]]
    sum(tic0[nr, 86]=="insurance")
}

ldaErr <- function(train.lda=spam01.lda, train=spam01, test=spam2,
                   traingp=spam01[,'type'], testgp=spam2[,'type']){
    trainCV.lda <- update(train.lda, CV=TRUE)
    prior01 <- train.lda$prior
    ldaRates <- c(loo=1-confusion(traingp,
                            trainCV.lda$class,
                            printit=NULL)$overall,
              trainerr=1-confusion(traingp,
                                 predict(train.lda)$class,
                                 printit=NULL)$overall,
              testerr=1-confusion(testgp,
                                predict(train.lda,
                                        newdata=test)$class,
                                prior=prior01, printit=NULL)$overall)
     ldaRates
}

rpartErr <- function(train.rp=spam01.rp, train=spam01, test=spam2,
                     outcome='type'){
    cptab <- train.rp$cptable
    nbest <- which.min(cptab[,"xerror"])
    rnprop <- prop.table(table(train.rp$y))
    xcv <- cptab[nbest,"xerror"] * min(rnprop)
    trainerr <- cptab[nbest,"rel error"] * min(rnprop)
    class2 <- predict(train.rp, newdata=test, type="class")
    testerr <- 1-confusion(test[, outcome], class2, printit=FALSE,
                        prior=rnprop)$overall
    c(cverror=xcv, trainerror=trainerr, testerror=testerr)
}

rfErr <- function(train.rf=spam01.rf, train=spam01, test=spam2,
                  outcome='type'){
    trainClass <- predict(train.rf, newdata=spam01, type="class")
    testClass <- predict(train.rf, newdata=test, type="class")
    rnprop <- prop.table(table(train[, outcome]))
    rfRates <- c(OOBerr=train.rf$err.rate[train.rf$ntree, "OOB"],
             trainerr=1-confusion(train$type, trainClass,
                                 printit=FALSE)$overall,
             testerr=1-confusion(spam2$type, testClass, printit=FALSE,
                                prior=rnprop)$overall)
    rfRates
}

fig7.10 <- function(){
library(MASS)
library(lattice)
library(mlbench)
data(Vowel)
form <- paste("~", paste(paste("V", 2:10, sep= ""),
                        collapse="+"))
gph <- bwplot(formula(paste("Class", form)),
              scales=list(x="free"),
              data=Vowel, outer=TRUE, layout=c(3,3))
gph
}

library(MASS)
library(DAAG)
library(latticeExtra)
fig7.1()
cuckoos.lda <- lda(species ~ length + breadth, data=cuckoos)
cuckoos.qda <- qda(species ~ length + breadth, data=cuckoos)
fig7.2()
fig7.3()
library(rpart)
if(require(SMIR)){
library(SMIR); data(bronchit)
}
if(exists('bronchit')){
bronchit <- within(bronchit,
                  rfac <- factor(r, labels=c("abs","pres")))
```

```
fig7.4()
fig7.5()
}
fig7.6()
if(exists('bronchit')){
fig7.7()
fig7.8()
}
if(require(kernlab)){
data(ticdata)
}
## Use first 5822 observstions for training
if(exists('ticdata')){
tic0 <- ticdata[1:5822, ]
tic1 <- ticdata[-(1:5822), ]
bestsize(n0=150)
}
```

```
[1] 133
```

```
if(require(kernlab)){
data(spam)
}
if(exists('spam')){
nr <- sample(1:nrow(spam))
spam0 <- spam[nr[1:2601],]      ## Training
spam1 <- spam[nr[2602:3601],]   ## Holdout
spam01 <- spam[nr[1:3601],]     ## Use for training,
                                ## if holdout not needed
spam2 <- spam[nr[3602:4601],]   ## Test
spam01.lda <- lda(type~., data=spam01)
ldaError <- ldaErr()
set.seed(29)     ## Make results precisely reproducible
spam01.rp <- rpart(type~., data=spam01, cp=0.0001)
rpartError <- rpartErr()
set.seed(29)
spam01.rf <- randomForest(type ~ ., data=spam01)
rfError <- rfErr()
}
if(require(mlbench)){
data(Vowel)
}
fig7.9()
```