R Code from
Mahler's Guide to

Advanced Statistical Learning

CAS Exam MAS-2

prepared by
Howard C. Mahler, FCAS

Howard Mahler
hmahler@mac.com
www.howardmahler.com/Teaching

Section 4, Classification Trees

```
install.packages("tree")
library(tree)
library(datasets)
data(iris)

iris.tree=tree(Species~.,data=iris)
summary(iris.tree)
plot(iris.tree)
text(iris.tree , pretty=0)

cv.iris=cv.tree(iris.tree)
plot(cv.iris$size, cv.iris$dev , type="b")

prune.iristree=prune.misclass(iris.tree , best=4)
summary(prune.iristree)
plot(prune.iristree)
text(prune.iristree , pretty=0)
```

From Solutions to Problems:

```
install.packages("mlbench")
library(mlbench)
data(PimaIndiansDiabetes2)
install.packages("tree")
library(tree)
diabetes.tree=tree(diabetes~., data=PimaIndiansDiabetes2)
diabetes.tree=prune.misclass(diabetes.tree , best=9)
plot(diabetes.tree)
text(diabetes.tree , pretty=0, cex=0.9)
```

Section 5, Bagging

```
install.packages("tree")
library(tree)
install.packages("randomForest")
library(randomForest)

dos=c(1, 4, 7, 10, 13, 16, 19, 22, 25, 28, 31, 34, 37, 40, 43, 46, 49, 52, 55, 58, 61, 64, 67, 70)
eff=c(3 , 5 ,10 , 9, 10, 11, 16, 18, 15, 44, 62, 85, 95, 90, 82, 59, 50, 28, 15, 7, 11, 4, 12, 3)

drug.tree=tree(eff~dos)
plot(drug.tree)
text(drug.tree,pretty=0)

s1=sample(1:24,replace=TRUE)
s1
        9 8 12 21 8 14 17 10 20 22 11 7 18 21 9 12 21 10 18 17 21 1 18 12

drug.tree1=tree(eff[s1]~dos[s1])
plot(drug.tree1)
text(drug.tree1,pretty=0)

bag.drug=randomForest(eff~dos,mtry=1,importance=TRUE)
bag.drug
        Call:
        randomForest(formula = eff ~ dos, mtry = 1, importance = TRUE)
        Type of random forest: regression
        Number of trees: 500
        No. of variables tried at each split: 1
        Mean of squared residuals: 76.12459
        % Var explained: 91.83

yhat.drug=predict(bag.drug,dos)
plot(dos,yhat.drug)
```

Section 7, Boosting

```
install.packages("gbm")
library("gbm")
dos=c(1, 4, 7, 10, 13, 16, 19, 22, 25, 28, 31, 34, 37, 40, 43, 46, 49, 52, 55, 58, 61, 64, 67, 70)
eff=c(3 , 5 ,10 , 9, 10, 11, 16, 18, 15, 44, 62, 85, 95, 90, 82, 59, 50, 28, 15, 7, 11, 4, 12, 3)

boost.drug =gbm(eff~dos, distribution="gaussian", n.trees =500 , interaction.depth =1,
        n.minobsinnode =1, shrinkage=0.01)
yhat.boostdrug=predict(boost.drug)
plot(dos,yhat.boostdrug)
```

Section 8, BART

```
install.packages("BART") # only need to install a package once
library(BART)

install.packages("kmed") # only need to install a package once
library(kmed)

attach(heart)
heart$sex <- as.integer(as.logical(heart$sex)) # convert True/False to 1/0
heart$fbs <- as.integer(as.logical(heart$fbs))
heart$exang <- as.integer(as.logical(heart$exang))
heart$class<-+ sapply(heart$class, as.logical) # convert values greater than 1 to 1

set.seed (1)
train <- sample (1: nrow(heart), nrow(heart) / 2)

xheart<-heart[,1:13]
yheart<-heart[,14]
xhearttrain=xheart[train,]
yhearttrain=yheart[train]
xhearttest=xheart[-train,]
yhearttest=yheart[-train]

bartheartfit<-pbart(xhearttrain,yhearttrain,x.test=xhearttest)

mean((yhearttest - bartheartfit$prob.test.mean)^2)
```

Section 9, Principal Components Analysis

```
x=matrix(c(-2, 1, 3, 6, 11, 15, 17, 20, 1, -1, 4,-4, 0, 8, -8, -6, -4, 0, 4, 8, 12, 16, 20, 24) , ncol=3)
agents=prcomp(x,scale=TRUE)

agents$rotation
agents$x

agents$sdev^2

agents$sdev^2/sum(agents$sdev^2)

biplot(agents,scale=0)
```

```
library(datasets)
data(iris)
iris.meas=iris[,c(1,2,3,4)]
names(iris.meas)
pca.out=prcomp(iris.meas,scale=TRUE)
pca.out$rotation
pca.out$sdev^2
pca.out$sdev^2/sum(pca.out$sdev^2)
```

From Solutions to Problems:

```
library(ISLR)
data(Auto)
mtcars.pca<-prcomp(mtcars[,c(1:7,10,11)],center=TRUE,scale=TRUE)
biplot(mtcars.pca,scale=0,xlabs=1:32)
mtcars.pca$rotation
mtcars.pca$x
summary(mtcars.pca)
```

Section 10, Missing Values and Matrix Completion

```
qqna=matrix(c(8,11,18,16,21,25,27,18,11,9,14,11,10,18,11,4,24,24,14,18, 22,26,30, 34),ncol=3)
qindex.na=matrix(c(3,8,4,7,1,2,1,1,2,2,3,3),ncol=2)
qqna[qindex.na]<-NA

getmissing<-function(X){
return(createindices(X)[which(is.na(X)),])}

getnonmissing<-function(X){
return(createindices(X)[which(!is.na(X)),])}

createindices<-function(X){
nrows<-dim(X)[1]
ncols<-dim(X)[2]
cols<-rep(NA,nrows*ncols)
for(i in 1:ncols){cols[(1:nrows)+(i-1)*nrows]<-i}
rows<-rep(1:nrows,ncols)
return(cbind(rows,cols))}

pcapprox<-function(pc,means){
c<-createindices(pc$x)
return(means[c[,2]]+pc$x[c[,1],1]*pc$rotation[c[,2],1])} # Using only 1st principal component

matrixcomplete<-function(X){
d<-RMSold<-1e12
iter<-1
miss<-getmissing(X)
nonmiss<-getnonmissing(X)
Xp<-X
Xbar<-colMeans(X,na.rm=TRUE)
Xp[miss]<-Xbar[miss[,2]]
while(d>tol){
Y<-pcapprox(prcomp(Xp),Xbar)
Y<-matrix(Y, ncol=dim(X)[2])
Xp[miss]<-Y[miss]
RMS<-sum((Y[nonmiss]-Xp[nonmiss])^2)
d<-RMSold-RMS
print(c(iter, RMS))
RMSold<-RMS
iter<-iter+1
Xbar<-colMeans(Xp)}
return(Xp)}

tol<-0.001
matrixcomplete(qqna)
```

Section 11, K-Means Clustering

```
kmeans(matrix(c(11,19,28,34,40,45,56,62,30,27,42,15,53,38),7,2),2)
```

From Solutions to Problems:

```
kmeans(matrix(c(48,34,79,61,25,47,82,9,33,6,7,1,84,47,50,71,45,96,67,26),10,2),3)
```

```
kmeans(matrix(
c(3,7,15,22,34,40,56,61,78,92,70,83,35,6,14,80,17,26,81,2,87,8,83,96,49,72,25,47,18,67),10,3),
3, nstart=20)
```

```
kmeans(matrix(c(24,21,35,12,18,11,27,14,4,10),5,2),2,nstart=20)
```

Section 12, Hierarchical Clustering

```
plot(hclust (dist(matrix(c(11,19,28,34,40,45,56,62,30,27,42,15,53,38),7,2)),
        method ="complete"))


plot(hclust (dist(matrix(c(11,19,28,34,40,45,56,62,30,27,42,15,53,38),7,2)), method ="single"))


plot(hclust (dist(matrix(c(11,19,28,34,40,45,56,62,30,27,42,15,53,38),7,2)), method ="average"))


plot(hclust (dist(matrix(c(11,19,28,34,40,45,56,62,30,27,42,15,53,38),7,2)), method ="centroid"))


plot(hclust (dist(matrix(c(11,19,28,34,40,45,56,62,30,27,42,15,53,38),7,2),
        method="manhattan"), method ="single"))
```

```
ex3=matrix(c(3,7,15,22,34,40,56,61,78,92,70,83,35,6,14,80,17,26,81,2,87,8,83,96,49,72,25,47,
                18,67), 10, 3)
plot(hclust (as.dist(1-cor(t(ex3))), method ="complete"))
```

From Solutions to Problems:

```
plot(hclust(dist(c(14, 9, 43, 39, 27))^2, method="centroid"))
```

```
dat=matrix(c(5,4,3,5,5,3,2,5,6,0,3,1,6,5),7,2)
plot(hclust(dist(dat), method ="complete"))
plot(hclust(dist(dat), method ="average"))
```

```
genes=matrix(c(0.597, 0.357, 0.191, 0.693, 0.318, 0.014, 0.165, 0.975, 0.334, 0.115, 0.753,
        0.217,0.944, 0.964, 0.99, 0.894, 0.794, 0.977,0.935, 0.043, 0.451, 0.402, 0.62, 0.2), 6, 4)
plot(hclust(as.dist(1-cor(t(genes)))))
```

```
plot(hclust(as.dist(
  matrix(c(0,9,3,6,11,9,0,7,5,10,3,7,0,12,2,6,5,12,0,8,11,10,2,8,0),5,5)), method="complete"))
plot(hclust(as.dist(
  matrix(c(0,9,3,6,11,9,0,7,5,10,3,7,0,12,2,6,5,12,0,8,11,10,2,8,0),5,5)), method="single"))
```

```
data = matrix(c(0.725, 0.581, 0.846, 0.217, -0.027, -0.381, 2.442, -0.315, 1.162,   -0.252, -1.130,
        1.446, 2.033, -0.334, 1.151, -0.248, 1.938, -0.276, 0.795, 2.021),10,2)
plot( hclust(dist(data), method="complete") )

kmeans(data, 2)
```