**Decision Tree**

> library(caret)

>

> eddat <- read.csv('http://math.mercyhurst.edu/~sousley/STAT\_139/data/EDDat3.csv', as.is = T);

>

> attach(eddat);

>

> sapply(eddat, function(x) sum(is.na(x))/length(x))

>

> sapply(eddat, function(x) sum(!is.na(x))/length(x))

Rec RecT Grp i1 i2 i3

>

> colSums(is.na(eddat))

>

> table(na.omit(eddat[,c(3,5,6,9,13,25,39,40,12,27,32)])$Grp)

1 2 3

79 272 101

>

> eddat <- na.omit(eddat[,c(3,5,6,9,13,25,39,40,12,27,32)])

>

> Accuracies <- c(0.00)

>

> for (i in seq(1000))

+ {

+ inTest<-createDataPartition(eddat$Grp, p = .25, list = FALSE)

+ require(rpart)

+ edata1<-rpart(Grp~ ., data=eddat,method = "class",subset = inTest,

+ parms= list(split = "gini",prior = c(1/3,1/3,1/3)),

+ control = rpart.control(usesurrogate= 0, maxsurrogate= 0))

+

+ Accuracies[i] <- confusionMatrix(eddat[inTest,"Grp"],predict(edata1,newdata= eddat[inTest,],type = "class"))$overall["Accuracy"]

+

+ }

> summary(Accuracies)

Min. 1st Qu. Median Mean 3rd Qu. Max.

0.6315789 0.7456140 0.7719298 0.7680965 0.7894737 0.8771930

**Naive Bayes**

> library(caret)

>

>

> eddat <- read.csv('http://math.mercyhurst.edu/~sousley/STAT\_139/data/EDDat3.csv', as.is = T);

>

> attach(eddat);

>

> sapply(eddat, function(x) sum(is.na(x))/length(x))

>

> sapply(eddat, function(x) sum(!is.na(x))/length(x))

>

> colSums(is.na(eddat))

>

> table(na.omit(eddat[,c(3,5,6,9,13,25,39,40,12,27,32)])$Grp)

1 2 3

79 272 101

>

> eddat <- na.omit(eddat[,c(3,5,6,9,13,25,39,40,12,27,32)])

>

> Accuracies <- c(0.00)

> for (i in seq(1000))

+ {

+ inTrain <- createDataPartition(y = as.factor(eddat$Grp), p = .70, list = FALSE)

+ train <- eddat[inTrain,]

+ test <- eddat[-inTrain,]

+ nb1 <- train(as.factor(Grp) ~ i2+i3+i6+i10+i22+i36+i37+i9+i24+i29, data = train, method = "nb",

+ trControl = trainControl(method = "cv"),

+ tuneGrid = data.frame(usekernel = TRUE, fL = 0.5, adjust = 5))

+ bps <- predict(nb1, newdata = test)

+ Accuracies[i] <- confusionMatrix(test$Grp,bps)$overall["Accuracy"]

+ }

There were 50 or more warnings (use warnings() to see the first 50)

>

> summary(Accuracies)

Min. 1st Qu. Median Mean 3rd Qu. Max.

0.5820896 0.5970149 0.6044776 0.6054627 0.6044776 0.7537313

**LDA**

> library(caret)

>

>

> eddat <- read.csv('http://math.mercyhurst.edu/~sousley/STAT\_139/data/EDDat3.csv', as.is = T);

>

> attach(eddat);

>

>

> sapply(eddat, function(x) sum(is.na(x))/length(x))

>

> sapply(eddat, function(x) sum(!is.na(x))/length(x))

>

> colSums(is.na(eddat))

>

> table(na.omit(eddat[,c(3,5,6,9,13,25,39,40,12,27,32)])$Grp)

1 2 3

79 272 101

>

> eddat <- na.omit(eddat[,c(3,5,6,9,13,25,39,40,12,27,32)])

>

>

> require(asbio)

> Accuracies <- c(0.00)

> for (i in seq(1000))

+ {

+ inTrain <- createDataPartition(y = eddat$Grp, p = .70, list = FALSE)

+ training <- eddat[inTrain,]

+ edata2 <- lda(as.matrix(eddat[,c(2:11)]), eddat[,"Grp"], data = eddat, prior = c(1/3,1/3,1/3),

+ subset = -inTrain, CV = T)

+ Accuracies[i] <- confusionMatrix(eddat[-inTrain,"Grp"], edata2$class)$overall["Accuracy"]

+ }

Error in lda.default(x, grouping, ...) :

variable 4 appears to be constant within groups

> summary(Accuracies)

Min. 1st Qu. Median Mean 3rd Qu. Max.

0.6888889 0.6944444 0.7000000 0.7000000 0.7055556 0.7111111

**KNN**

> library(caret)

>

>

> eddat <- read.csv('http://math.mercyhurst.edu/~sousley/STAT\_139/data/EDDat3.csv', as.is = T);

>

> attach(eddat);

>

>

> sapply(eddat, function(x) sum(is.na(x))/length(x))

>

> sapply(eddat, function(x) sum(!is.na(x))/length(x))

>

> colSums(is.na(eddat))

>

> table(na.omit(eddat[,c(3,5,6,9,13,25,39,40,12,27,32)])$Grp)

1 2 3

79 272 101

>

> eddat <- na.omit(eddat[,c(3,5,6,9,13,25,39,40,12,27,32)])

>

> Accuracies <- c(0.00)

>

> for (i in seq(1000))

+ {

+ inTrain <- createDataPartition(y = as.factor(eddat$Grp), p = .70, list = FALSE)

+ training <- eddat[inTrain,]

+ testing <- eddat[-inTrain,]

+ knn4 <- train(as.factor(Grp) ~ i2+i3+i6+i10+i22+i36+i37+i9+i24+i29, data = training, method = "knn",

+ preProcess = c("center", "scale"), tuneLength = 10,

+ trControl = trainControl(method = "cv"))

+ update(knn4, list(.k = 3))

+ knn4\_pred <- predict(knn4,newdata = testing)

+ Accuracies[i] <- confusionMatrix(knn4\_pred,as.factor(testing$Grp))$overall["Accuracy"]

+ }

> summary(Accuracies)

Min. 1st Qu. Median Mean 3rd Qu. Max.

0.6492537 0.7238806 0.7462687 0.7452463 0.7611940 0.8283582

**FDA**

> library(mda)

> library(caret)

> eddat <- read.csv('http://math.mercyhurst.edu/~sousley/STAT\_139/data/EDDat3.csv', as.is = T);

> attach(eddat);

> sapply(eddat, function(x) sum(is.na(x))/length(x))

> sapply(eddat, function(x) sum(!is.na(x))/length(x))

> colSums(is.na(eddat))

> table(na.omit(eddat[,c(3,5,6,9,13,25,39,40,12,27,32)])$Grp)

1 2 3

79 272 101

> eddat <- na.omit(eddat[,c(3,5,6,9,13,25,39,40,12,27,32)])

>

> Accuracies <- c(0.00)

> for (i in seq(1000))

+ {

+

+ edat1 <- fda(Grp ~ i2+i3+i6+i10+i22+i36+i37+i9+i24+i29, data = eddat, prior = c(1/3,1/3,1/3), CV = TRUE, method =

+ polyreg, degree = 3)

+

+ Accuracies[i] <- confusionMatrix(eddat$Grp, predict(edat1))$overall["Accuracy"]

+ }

There were 50 or more warnings (use warnings() to see the first 50)

> summary(Accuracies)

Min. 1st Qu. Median Mean 3rd Qu. Max.

0.8606195 0.8606195 0.8606195 0.8606195 0.8606195 0.8606195

**Multinomial**

> library(nnet)

>

> eddat <- read.csv('http://math.mercyhurst.edu/~sousley/STAT\_139/data/EDDat3.csv', as.is = T);

> attach(eddat);

> sapply(eddat, function(x) sum(is.na(x))/length(x))

> sapply(eddat, function(x) sum(!is.na(x))/length(x))

> colSums(is.na(eddat))

> table(na.omit(eddat[,c(3,5,6,9,13,25,39,40,12,27,32)])$Grp)

1 2 3

79 272 101

> eddat <- na.omit(eddat[,c(3,5,6,9,13,25,39,40,12,27,32)])

>

> Accuracies <- c(0.00)

> for (i in seq(1000))

+ {

+

+ mnmfit <- multinom(Grp ~ i2+i3+i6+i10+i22+i36+i37+i9+i24+i29,data=eddat, trace=FALSE)

+ #summary(mnmfit)

+

+ # get predictions on a new data set

+ round(predict(mnmfit, newdata=eddat, "probs"),4)

+

+ predict(mnmfit, newdata=eddat, "class")

+

+ #Multinominal

+

+ confusionMatrix(predict(mnmfit, newdata=eddat, "class"), eddat$Grp)

+

+ Accuracies[i] <- confusionMatrix(predict(mnmfit, newdata=eddat, "class"), eddat$Grp)$overall["Accuracy"]

+ }

> summary(Accuracies)

Min. 1st Qu. Median Mean 3rd Qu. Max.

0.7986726 0.7986726 0.7986726 0.7986726 0.7986726 0.7986726