Model Classification on MFT Study

By: Aretha Kassegnin and
[redacted]

1.

	Pre:Midwest	Pre:Northeast	Pre:South	Pre: West	Total
A:Midwest	12	1	790	0	803
A:Northeast	5	0	650	0	655
A: South	9	1	1368	2	1380
A: West	9	3	578	0	590
Total	35	5	3386	2	3428

Figure ***Error! No text of specified style in document.****-1*

Figure1 ***-Error! No text of specified style in document.****-2 Matrix with all the predictor values; Pre = predicted, A =actual*

	Predicted: Other	Predicted: South	Total
Actual: Other	2028	20	2048
Actual: South	1367	13	1380
Total	3395	33	3428

Figure 1-***Error! No text of specified style in document.****-3 Matrix with predictor variable South and Other as Midwest, West and Northeast combined*

We are going to be using Figure 1-2 to perform model evaluation on classification model. We choose the neural network as our classification model.

2. The accuracy that our model will predict the region that a 12th grader is going a particular drug is about 60%. This means that about 40% of our model will be incorrect (error rate) when predicting what region, a 12th grader is a drug. The sensitivity of the model is 9% which seems correct because we know the model isn't really getting accurate results. Which we know is wrong because the data is skewed. Specificity measures the ability to classify a record negatively with our target variable which is 99% so we know that 99% of the data is wrong. Precision with our model is 39% using region as a target variable which isn't great. F1 = 4% , F2 =2% and F0.5=8% This means the model isn't doing a great job predicting the data which means are model is incorrect. Below are the results.

```
Accuracy:  0.5953909
Error rate:  0.4046091
Sensitivity:  0.00942029
Specificity:  0.9902344
Precision:  0.3939394
F1 score:  0.01840057
F2 score:  0.01170538
F0.5 score:  0.04298942
```

3. Here is also the error cost model evaluation as shown below and as you can see the accuracy and error rate are the same. The sensitivity is about 2% which is correct we know the model isn't the best at predicting region. The sensitivity is less then the specificity which means that the model is getting more predictions wrong then getting them right. Precision is 50% with the target variable region. Then F1=4%, F2 =2% and F0.5 is 8% . Again the model isn't great as the model above.

```
Accuracy:  0.5974329
Error rate:  0.4025671
Sensitivity:  0.02101449
Specificity:  0.9858398
Precision:  0.5
F1 score:  0.0403338
F2 score:  0.02599498
F0.5 score:  0.08995037
```

	Predicted: Other	Predicted: South	Total
Actual: Other	2028	20	2048
Actual: South	1367	13	1380
Total	3395	33	3428

Figure 3: No Error Cost Model

	Predicted: 0	Predicted: 1	Total
Actual: 0	2038	10	2048
Actual: 1	1375	5	1380
Total	3413	15	3428

Figure 4: Error Cost Model

Appendix

```
library(caret)
set.seed(25)
data$SouthBinary <- as.factor(ifelse(data$region == "South", "South", "Other"))
inTrain <- createDataPartition(y = data$region,
                               p = .75,
                               list = FALSE)

data.train <- data[ inTrain ,  ]
dim(data.train)[1]
dim(data)[1]
dim(data.train)[1]/dim(data)[1]
#testing data
data.test <- data[ -inTrain ,  ]
dim(data.test)[1]/dim(data)[1]

head(data.train)
lapply(X = data.train, FUN = summary)
is.factor(data.train$region)

test.q <- subset(x = data.test, select = c("alcohol_12","weed_12","LSD_12",
                                           "MDMA_12","coke_12","amp_12","tranq_12",
                                           "meth_12"))

library(nnet)
library(NeuralNetTools)
test.q <- subset(x = data.test, select = c("alcohol_12","weed_12","LSD_12",
                                           "MDMA_12","coke_12","amp_12","tranq_12",
                                           "meth_12"))
nnet01 <-  nnet(region ~ weed_12+ alcohol_12+ LSD_12+ MDMA_12 + meth_12
                + amp_12+ tranq_12 + coke_12, data = data.train, size = 3)
plotnet(nnet01)
pred1 <- predict(nnet01, newdata = data.test, type = "class")
pred2 <- predict(nnet01, newdata = test.q, type ="class")
t1 <- table(data.test$region, pred1)
t1
t2 <- table(data.test$region, pred2)
t2
```

```
row.names(t2) <- c("A:Midwest", "A:Northeast","A: South", "A: West")
colnames(t2) <- c("Pre:Midwest", "Pre:Northeast","Pre:South","Pre: West")
t2 <- addmargins(A=t2, FUN = list(Total = sum), quiet = TRUE)
t2

####South and Other
nnet02 <-  nnet(SouthBinary ~ weed_12+ alcohol_12+ LSD_12+ MDMA_12 + meth_12
        + amp_12+ tranq_12 + coke_12, data = data.train, size = 3)
plotnet(nnet02)
pred3 <- predict(nnet02, newdata = data.test, type = "class")
pred4 <- predict(nnet02, newdata = test.q, type ="class")
t3 <- table(data.test$SouthBinary, pred3)
t4 <- table(data.test$SouthBinary, pred4)
t3
t4
row.names(t4) <- c("Actual: Other", "Actual: South")
colnames(t4) <- c("Predicted: Other", "Predicted: South")
t4 <- addmargins(A=t4, FUN = list(Total = sum), quiet = TRUE)
t4
#####error cost
error.cost <- matrix(c(0,4,1,0), byrow = TRUE, ncol=2)
dimnames(error.cost) <- list(c("0","1"), c("0","1"))
error.cost

nnet03 <-  nnet(SouthBinary ~ weed_12+ alcohol_12+ LSD_12+ MDMA_12 + meth_12
        + amp_12+ tranq_12 + coke_12, data = data.train,costs = error.cost,
        size =3)

plotnet(nnet03)
ypred.costs <- predict(object = nnet03,newdata = test.q, type ="class")
t6 <- table(data.test$SouthBinary, ypred.costs)
row.names(t6) <- c("Actual: 0", "Actual: 1")
colnames(t6) <- c("Predicted: 0", "Predicted: 1")
t6 <- addmargins(A=t6, FUN = list(Total = sum), quiet = TRUE)
t6


library(rpart)
cart01 <- rpart(SouthBinary ~ weed_12+ alcohol_12+ LSD_12+ MDMA_12 + meth_12
        + amp_12+ tranq_12 + coke_12,
        data = data.train,
        method = "class")
library(rpart.plot)
rpart.plot(cart01, type = 4, extra = 102)




eval_metrics <- function(table) {

  TN <- table[1,1]
```

```
  TP <- table[2,2]
  FN <- table[2,1]
  FP <- table[1,2]

  # calculate measures
  Accuracy <- (TN + TP) / (TP + TN + FP + FN)
  ErrorRate <- 1 - Accuracy
  Sensitivity <- TP / (TP + FN)  # Sensitivity
  Specificity <- TN / (FP + TN)  # Specificity
  Precision <- TP / (TP + FP)  # Precision
  F1 <- 2 * Precision * Sensitivity / (Precision + Sensitivity)  # F1 score
  F2 <- 5 * Precision * Sensitivity / (4 * Precision + Sensitivity)  # F2 score
  F05 <- 1.25 * Precision * Sensitivity / (0.25 * Precision + Sensitivity)  # F0.5 score

  cat("Accuracy: ", Accuracy, "\n")
  cat("Error rate: ", ErrorRate, "\n")
  cat("Sensitivity: ", Sensitivity, "\n")
  cat("Specificity: ", Specificity, "\n")
  cat("Precision: ", Precision, "\n")
  cat("F1 score: ", F1, "\n")
  cat("F2 score: ", F2, "\n")
  cat("F0.5 score: ", F05, "\n")


}

eval_metrics(t4)
eval_metrics(t6)
```