ST4061 – Computer Intensive Statistical Analytics II
ST6041 – Machine Learning and Statistical Analytics II
2023-24
Continuous Assessment Component 1

INTRUCTIONS

- Provide your answers in the template document provided.
- Paste the R code you used for each question item as indicated.
- Submit your answer document via Canvas. Your file will be renamed automatically by Canvas, so you do not need to worry about file naming.

Question 1 - No R coding is required for this question.

Figure 1 below shows the output of a simple random forest consisting of 4 trees fitted to a sample of training data points. This dataset comprises of 5 variables: Length, Width, Leaf, Curve and Age.

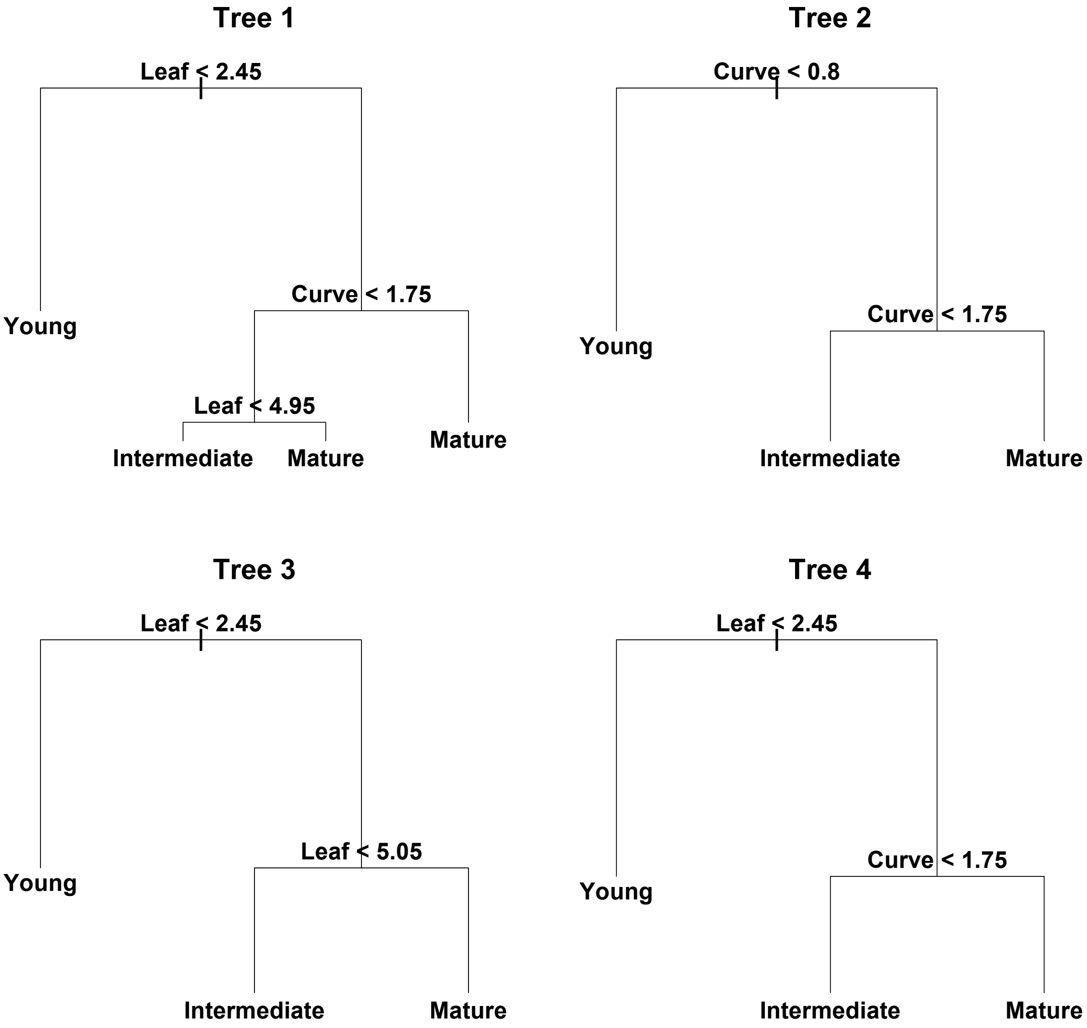

Figure 1 – Diagram for Question 1(a).

(a) Quote the predicted values from the model for the following test points:

	Length	**Width**	**Leaf**	**Curve**	**Prediction**
Obs1	4.5	2.3	1.3	0.3	
Obs2	5.0	3.5	4.3	0.3	
Obs3	6.1	3.0	4.9	1.8	
Obs4	7.2	3.0	5.8	1.9	
Obs5	5.1	3.8	2.5	0.4	

(b) Calculate the overall misclassification rate from the model, assuming the true test values were as follows:

Test point	**Obs1**	**Obs2**	**Obs3**	**Obs4**	**Obs5**
True value	Young	Young	Intermediate	Mature	Young

(c) Based on the confusion matrix below, obtained after predicting a number of new test points from the model, calculate:

i. The number of test points in the test dataset.

ii. The overall correct classification rate.

iii. The correct classification rate for class "Young".

iv. The misclassification rate for class "Mature".

		Predicted		
		Young	Intermediate	Mature
Actual	Young	8	3	1
	Intermediate	2	12	4
	Mature	0	3	9

Question 2

Consider the R code below:

```
require(ISLR)
require(glmnet)

dat = na.omit(Hitters)
dat$Salary = log(dat$Salary)
x = model.matrix(Salary~.+0, data=dat)
y = dat$Salary
n = nrow(x)
K = 10
crit1 = crit2 = crit3 = crit4 = numeric(K)
folds = cut(1:n,K,labels=FALSE)
set.seed(1)
for(k in 1:K){
 i.train = which(folds!=k)
 x.train = x[i.train,]
 y.train = y[i.train]
 x.test = x[-i.train,]
         y.test = y[-i.train]
 mod1 = glmnet(x.train,y.train,alpha=0.5)
 out = cv.glmnet(x.train,y.train,alpha=0.5)
 mod2 = glmnet(x.train,y.train,alpha=0.5, lambda=out$lambda.min)
 f1 = predict(mod1,newx=x.train)[,1]
 f2 = predict(mod2,newx=x.train)[,1]
 p1 = predict(mod1,newx=x.test)[,1]
 p2 = predict(mod2,newx=x.test)[,1]
 crit1[k] = mean((f1-y.train)^2)
 crit2[k] = mean((f2-y.train)^2)
 crit3[k] = mean((p1-y.test)^2)
 crit4[k] = mean((p2-y.test)^2)
}
par(font=2, font.axis=2, font.lab=2, pch=20)
boxplot(cbind(crit1,crit2,crit3,crit4))
```

(a) Is this a regression or a classification problem? Justify your answer.

(b) Name the model used to generate mod1. Justify your answer.

(c) What type of cross-validation is applied to model 2? Justify your answer.

Question 3

For this question you are required to use the following packages:

```
require(ISLR)
require(class)
require(pROC)
```

Consider the dataset Smarket from library ISLR. Here the response variable is Smarket$Direction:

```
x = Smarket[,-9]
y = Smarket$Direction
set.seed(4061)
train = sample(1:nrow(Smarket),1000)
```

(a) Fit a random forest classifier (using all default values) to the training set. Quote the **training** misclassification rate obtained from it.

(b) Generate a prediction of the 250 test observations from this random forest. Compute and plot the corresponding ROC. Quote the associated AUC.

(c) Generate a classification using the kth-nearest neighbour (kNN) classifier with k=2. Compute and plot the corresponding ROC (adding to the plot of (b)). Quote the associated AUC.
Hint: function attributes() *may be useful here.*

(d) Split the sample into training and test sets using the R instruction:

```
set.seed(4061)
M = 1000
train = sample(1:nrow(Smarket), M)
```

Compute test-set misclassification errors obtained from the kNN classifier for each value of k between 1 and 10. Plot this curve.