**INDIVIDUAL ASSIGNMENT-5**

Q1) AVERAGE TRUE CTR FOR BEHAVIOURAL:

This gives us the avg CTR of the behavioral policy as 0.01145116

Base revenue = 0.88/100 \*100000 = 880

New revenue = 1.145/100 \* 100000 = 1145

Percentage improvement in revenue = 30.11 %

The revenue also improves by the same percentage.

AVERAGE TRUE CTR FOR CONTEXTUAL:

This gives us the avg CTR of the behavioral policy as 0.01115444

Base revenue = 0.88/100 \*100000 = 880

New revenue = 1.115/100 \* 100000 = 1115

Percentage improvement in revenue = 26.70 %

The revenue also improves by the same percentage.

Thus, we can see that like all the other models, the CTR for contextual is lower

than behavioral

|  |  |  |  |
| --- | --- | --- | --- |
|  | CONTEXTUAL | BEHAVIOURAL | FULL/OPTIMAL |
| CTR | 1.115% | 1.145% | 1.38% |
| IMPROVEMENT | 26.70 % | 30.11 % | 56.8% |

Q2) Code:

# Define the 'Contextual' model

contextual.model2 = click ~ ctrpat

# Building the classification tree for the 'Contextual' model

library(rpart)

contextual.tree2 = rpart(formula = contextual.model2, data = ctrtrain,

control = rpart.control(cp = 0.00015))

# Visualizing the 'Contextual' classification tree

library(rpart.plot)

rpart.plot(contextual.tree2)



The tree structure is shown in the graph. This tree has three *leaves* or endpoints. Each leaf is the endpoint of a series of splits and as such represents a partition of the X-space. All leaves are mutually exclusive, i.e, an observation in our data can only belong to one leaf.

Each leaf in the plot contains two pieces of information –

1) the average prediction for observations in that leaf, and

2) the percentage of observations that fall under this leaf.

For example, the leftmost leaf in the tree plot tells us that datapoints/observations with 𝑐trpat < 0.0085 are similar. On average, the probability of click for these datapoints is 0.0027. These observations account for 45% of our data. Similarly, the rightmost leaf consists of observations with 𝑐trpat ≥ 0.03 and have a click probability of 0.036(lesser than we had included impat and clpat in our model). This however consists of a very small fraction of the data almost 2% which is much smaller compared to the 54% that have a ctrpat value <0.03

We see that the prediction values remain almost the same. Thus, when we use only ctrpat as an explanatory variable, we can see that the tree has lesser branches as the number of input variables decrease. But what we can also observe, that is different from the original contextual model is that the tree is more balanced in terms of the percentage values.

Q3) Code:

This is the code to create the new tree, called full.tree2 with a complexity parameter of 0.00001.

TESTING DATA :

#FULL CART MODEL USING CP = 0.00001

full.model = click ~ impua+clua+ctrua+imppat+clpat+ctrpat+ impup+clup+ctrup+imput+clut+ctrut

full.tree2 = rpart(formula = full.model, data = ctrtest\_real, control = rpart.control(cp =0.00001))

rpart.plot(full.tree2)

full.CART.prediction3 = predict(full.tree2, ctrtest\_real)

ctrtest\_real$full.CART.pred3 = full.CART.prediction3

#AUC FOR FULL CART USING THE FULL.TREE2 MODEL

install.packages("pROC")

library(pROC)

auc.cart.full3 = roc(ctrtest\_real$click, ctrtest\_real$full.CART.pred3)

auc\_curve.cart.full3 = plot.roc(ctrtest\_real$click, ctrtest\_real$full.CART.pred3, main = "ROC curve Full CART 2",

percent=TRUE, cex.main=0.75, cex.lab = 0.75)

auc(auc.cart.full3)

library(ROCR)

pred.cart.full3 = prediction(ctrtest\_real$full.CART.pred3, ctrtest\_real$click)

ROC.cart.full3 = performance(pred.cart.full3, "tpr", "fpr")

plot(ROC.cart.full3, colorize = T, lwd = 2) #from left to right, the cutoff is being reduced

abline(a=0, b=1, lty = 2)

AUC.tmp3 = performance(pred.cart.full3, "auc")

as.numeric([AUC.tmp3@y.values](mailto:AUC.tmp3@y.values))

 

AUC VALUE IS 0.9881837 which is much more than 0.7 and extremely close to one which makes this a very reasonable model.

#RIG VALUES FOR MODEL

RIG(ctrtest\_real$full.CART.pred3, ctrtest\_real$click)

RIG VALUE FOR TESTING DATA 59.0167

TRAINING DATA:

#FOR TRAINING DATA

full.model = click ~ impua+clua+ctrua+imppat+clpat+ctrpat+ impup+clup+ctrup+imput+clut+ctrut

full.tree2 = rpart(formula = full.model, data = ctrtrain, control = rpart.control(cp =0.00001))

rpart.plot(full.tree2)

full.CART.prediction3b = predict(full.tree2, ctrtrain)

ctrtrain$full.CART.pred3b = full.CART.prediction3b

#AUC FOR FULL CART USING THE FULL.TREE2 MODEL

install.packages("pROC")

library(pROC)

auc.cart.full3b = roc(ctrtrain$click, ctrtrain$full.CART.pred3b)

auc\_curve.cart.full3b = plot.roc(ctrtrain$click, ctrtrain$full.CART.pred3b, main = "ROC curve Full CART 2b",

percent=TRUE, cex.main=0.75, cex.lab = 0.75)

auc(auc.cart.full3b)

library(ROCR)

pred.cart.full3b = prediction(ctrtrain$full.CART.pred3b, ctrtrain$click)

ROC.cart.full3b = performance(pred.cart.full3b, "tpr", "fpr")

plot(ROC.cart.full3b, colorize = T, lwd = 2) #from left to right, the cutoff is being reduced

abline(a=0, b=1, lty = 2)

AUC.tmp3b = performance(pred.cart.full3b, "auc")

as.numeric(AUC.tmp3b@y.values)

#RIG VALUES FOR MODEL

RIG(ctrtrain$full.CART.pred3b, ctrtrain$click)

 

AUC VALUE WHEN WE USE THE TRAINING DATASET IS 0.9885091 WHICH IS MORE THAN WHEN WE USED THE TESTING DATA.

RIG FOR TRAINING DATA IS:

58.7368, This is lower than when we use the testing data.

We can observe that the results for the testing and training data are very close and both provide extremely good results with values of area under the curve almost equal to one, which tells us the accuracy of the model. The values of the relative information gain are almost 60% which tells that these models are better predictive models, the one with the testing data being marginally superior to the other.

Note : I am not sure if this was the right way to do it. But in my earlier attempt I did not change the data in the model and got a negative value for information gain. When I further read about this, I realized that even if the RIG value is small it can never be negative. Thus, I changed the input in the model. I am a little concerned about how high the values of the AUC and RIG are.