# tricks to include my path into sys.path to activate my package

import os

import sys

nb\_dir = os.path.split(os.getcwd())[0]

if nb\_dir not in sys.path:

    sys.path.append(nb\_dir)

sys.path

### import packages

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from IPython.display import clear\_output

from datetime import datetime

from sas7bdat import SAS7BDAT

from PACKAGE.Random\_Forest\_Package\_v1 import RFM\_Shawn as RFM\_CLF

from PACKAGE.Gradient\_Boosting\_Package\_v2 import GBM\_Shawn as GBM\_CLF

import pytz

import sas7bdat

#pd.set\_option('display.height', 1000)

#pd.set\_option('display.width', 1000)

pd.set\_option('display.max\_rows', 100)

pd.set\_option('display.max\_columns', 500)

### data preparation

print('Data load strs at '+ datetime.now(pytz.timezone('Asia/Hong\_Kong')).strftime('%y-%m-%d %H:%M:%S %z'))

csvfile1='/spare/share/nbroot/users/43715625/FCC/MODEL/amh\_fcc\_hp\_train\_impute.txt'

csvfile2='/spare/share/nbroot/users/43715625/FCC/MODEL/amh\_fcc\_hp\_valid\_impute.txt'

#csvfile3='/spare/share/nbroot/users/43425632/03.Machine\_Learning/MODEL\_BASE\_VALID\_VAR\_1612.csv'

# sas\_train='/spare/share/nbroot/users/43715625/MYH\_BTCIP/DATA/bt\_cip\_cards\_train\_cleaned.sas7bdat'

# sas\_valid='/spare/share/nbroot/users/43715625/MYH\_BTCIP/DATA/bt\_cip\_cards\_valid\_cleaned.sas7bdat'

# sas\_ot='/spare/share/nbroot/users/43715625/MYH\_BTCIP/DATA/bt\_cip\_cards\_ootvalid\_cleaned.sas7bdat'

input\_train=pd.DataFrame(pd.read\_csv(csvfile1))

input\_valid=pd.DataFrame(pd.read\_csv(csvfile2))

# df\_ot=pd.DataFrame(pd.read\_sas(sas\_ot))

#outvalid1=pd.DataFrame(pd.read\_csv(csvfile3))

# train=df\_train.drop(['CUST\_ID'], axis=1)

# valid=df\_valid.drop(['CUST\_ID'], axis=1)

#outvalid1\_for\_score=outvalid1.drop(['Cust\_ID','response'], axis=1)

print('Data load ends at '+ datetime.now(pytz.timezone('Asia/Hong\_Kong')).strftime('%y-%m-%d %H:%M:%S %z'))

print('Data as of '+ datetime.now(pytz.timezone('Asia/Hong\_Kong')).strftime('%y-%m-%d %H:%M:%S %z'))

print("#------------------- Separate Line 1 -------------------# ")

print("Train Data : ")

input\_train.info()

print("#------------------- Separate Line 2 -------------------# ")

print("Valid Data : ")

input\_valid.info()

print("#------------------- Separate Line 3 -------------------# ")

# print("Test Data : ")

# it\_to\_score.info()

print("#------------------- Separate Line 4 -------------------# ")

# development

print('Model train strs at '+ datetime.now(pytz.timezone('Asia/Hong\_Kong')).strftime('%y-%m-%d %H:%M:%S %z'))

gb\_clf=GBM\_CLF(mode='medium', df\_train=input\_train, df\_valid=input\_valid, str\_resp='response',str\_id='CUST\_ID')

gb\_clf.\_training()

print('Model train ends at '+ datetime.now(pytz.timezone('Asia/Hong\_Kong')).strftime('%y-%m-%d %H:%M:%S %z'))

gb\_clf.best\_param

gb\_clf.best\_model\_kpi

### data preparation

print('Data load strs at '+ datetime.now(pytz.timezone('Asia/Hong\_Kong')).strftime('%y-%m-%d %H:%M:%S %z'))

csvfile3='/spare/share/nbroot/users/43715625/FCC/MODEL/amh\_fcc\_hp\_ot\_1607\_impute.txt'

csvfile4='/spare/share/nbroot/users/43715625/FCC/MODEL/amh\_fcc\_hp\_ot\_1609\_impute.txt'

#csvfile3='/spare/share/nbroot/users/43425632/03.Machine\_Learning/MODEL\_BASE\_VALID\_VAR\_1612.csv'

# sas\_train='/spare/share/nbroot/users/43715625/MYH\_BTCIP/DATA/bt\_cip\_cards\_train\_cleaned.sas7bdat'

# sas\_valid='/spare/share/nbroot/users/43715625/MYH\_BTCIP/DATA/bt\_cip\_cards\_valid\_cleaned.sas7bdat'

# sas\_ot='/spare/share/nbroot/users/43715625/MYH\_BTCIP/DATA/bt\_cip\_cards\_ootvalid\_cleaned.sas7bdat'

# input\_train=pd.DataFrame(pd.read\_csv(csvfile1))

# input\_valid=pd.DataFrame(pd.read\_csv(csvfile2))

input\_ot\_1607=pd.DataFrame(pd.read\_csv(csvfile3))

input\_ot\_1609=pd.DataFrame(pd.read\_csv(csvfile4))

# train=df\_train.drop(['CUST\_ID'], axis=1)

# valid=df\_valid.drop(['CUST\_ID'], axis=1)

#outvalid1\_for\_score=outvalid1.drop(['Cust\_ID','response'], axis=1)

print('Data load ends at '+ datetime.now(pytz.timezone('Asia/Hong\_Kong')).strftime('%y-%m-%d %H:%M:%S %z'))

gb\_clf.\_modeltest(input\_ot\_1607)

gb\_clf.\_modeltest(input\_ot\_1609)

to\_score1=input\_ot\_1607.drop(['response'], axis=1)

scored=clf.\_score(df\_to\_score=to\_score1)

scored.to\_csv('/spare/share/nbroot/users/43715625/FCC/OUTPUT/scored\_1607\_re.csv')

to\_score2=input\_ot\_1609.drop(['response'], axis=1)

scored2=clf.\_score(df\_to\_score=to\_score2)

scored2.to\_csv('/spare/share/nbroot/users/43715625/FCC/OUTPUT/scored\_1609\_re.csv')

driver\_importance=pd.DataFrame(clf.importance[clf.importance.model\_cnt==1])

driver\_impartance\_t=pd.DataFrame(driver\_importance.T)

driver\_impartance\_t.to\_csv('/spare/share/nbroot/users/43715625/FCC/OUTPUT/DIT\_e100.csv')

clf.best\_model