# THE BANK PERSONAL LOAN PROJECT:

THE CODE:

import pandas as pd

import numpy as np

from sklearn import tree

from sklearn import preprocessing

df = pd.read\_excel(r"C:\letsupgrage assignment\Bank\_Personal\_Loan\_Modelling.xlsx", sheet\_name=1)

df.info()

from sklearn.ensemble import RandomForestClassifier

rf\_model = RandomForestClassifier(n\_estimators=1000, max\_features=2, oob\_score=True)

features = ["Age", "Experience", "Income", "Family",

"CCAvg", "Education", "Mortgage", "Securities Account",

"CD Account", "Online", "CreditCard"]

rf\_model.fit(X=df[features], y=df["Personal Loan"])

print("\n oob accuracy:")

print(rf\_model.oob\_score\_);

print("\n")

for features, imp in zip(features, rf\_model.feature\_importances\_):

print(features, imp);

predictors = pd.DataFrame([df["Income"], df["Family"], df["CCAvg"],

df["Education"]]).T

tree\_model = tree.DecisionTreeClassifier(max\_depth=12)

tree\_model.fit(X = predictors, y = df["Personal Loan"])

with open("Dtree3.dot", 'w')as f:

f = tree.export\_graphviz(tree\_model, feature\_names=["Income", "Family",

"CCAvg", "Education"], out\_file=f);

print("\n the accuracy of the model")

print(tree\_model.score(X = predictors, y = df["Personal Loan"]))

THE OUTPUT:

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 5000 entries, 0 to 4999

Data columns (total 14 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 ID 5000 non-null int64

1 Age 5000 non-null int64

2 Experience 5000 non-null int64

3 Income 5000 non-null int64

4 ZIP Code 5000 non-null int64

5 Family 5000 non-null int64

6 CCAvg 5000 non-null float64

7 Education 5000 non-null int64

8 Mortgage 5000 non-null int64

9 Personal Loan 5000 non-null int64

10 Securities Account 5000 non-null int64

11 CD Account 5000 non-null int64

12 Online 5000 non-null int64

13 CreditCard 5000 non-null int64

dtypes: float64(1), int64(13)

memory usage: 547.0 KB

oob accuracy:

0.9876

Age 0.045909723801300914

Experience 0.04519535861399452

Income 0.34241137924223636

Family 0.09636496220521862

CCAvg 0.18465611016697384

Education 0.16392025139181307

Mortgage 0.04348125811211044

Securities Account 0.005541729282717805

CD Account 0.05426496521248538

Online 0.008497020991864194

CreditCard 0.009757240979284851

the accuracy of the model

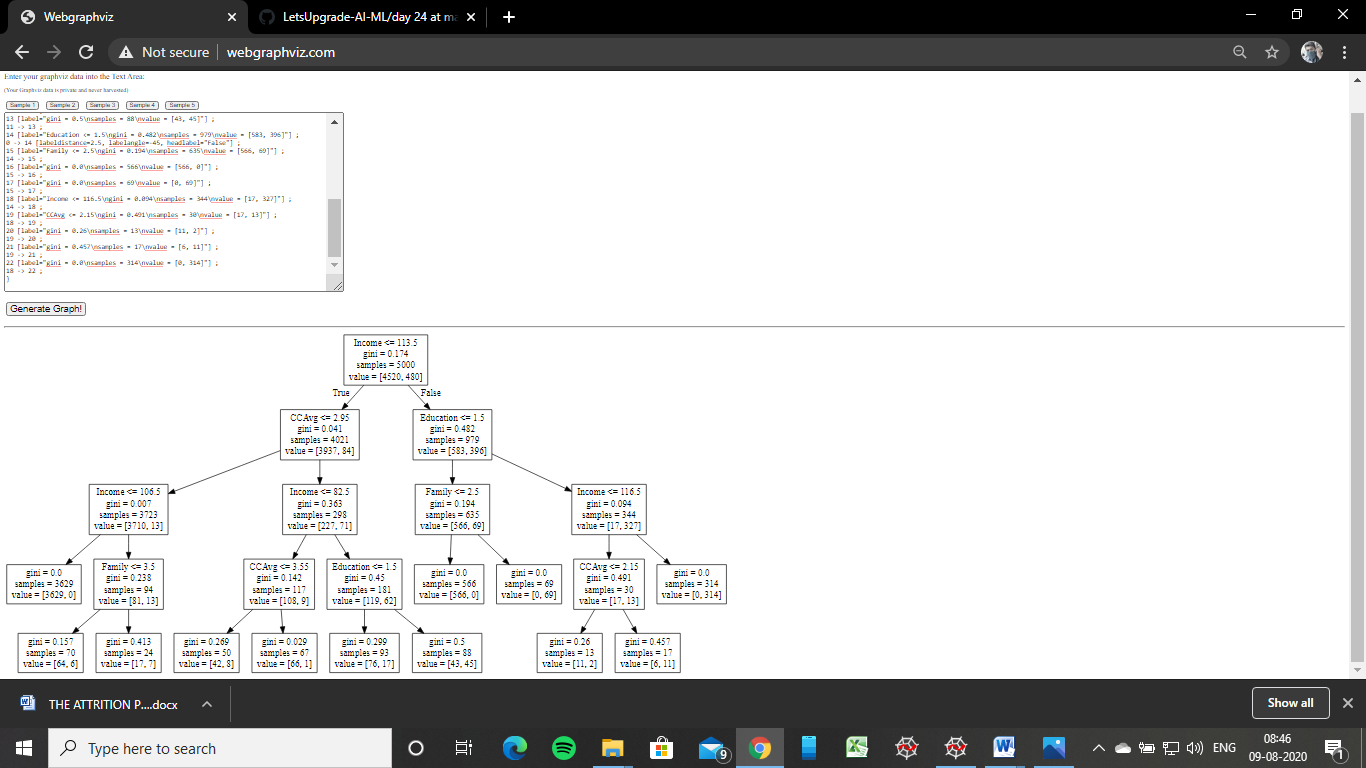
0.9994

In this project the important independent variable are the "Age", "Experience", "Income", "Family", "CCAvg", "Education", "Mortgage", "Securities Account", "CD Account", "Online", "CreditCard"These are the most important independent variable of the project.

” for features, imp in zip(features, rf\_model.feature\_importances\_):

print(features, imp);”

By using this features we can calculate the accuracy of the project. By calculating the accuracy of this project is “0.999”. This accuracy is the highest form, this tells the project has a high accuracy rate.



The rules:

* If income is less than 106.5 and the family is less than 3, then the loan is not given.
* If income is less than 82.5 and the ccavg is less than 3.5, then the loan is not given.
* If income is less than 82.5 and the education is less than 1.5, then the loan is given.
* If income is less than 116.5 and the ccavg is less, then 2.15, then the loan is not given.
* If income is greater than 116.5, then the loan is given.