

code for Fars Data set Analysis 210409183

January 26, 2023

```
[ ]: !pip install pmlb
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-
wheels/public/simple/
Collecting pmlb
  Downloading pmlb-1.0.1.post3-py3-none-any.whl (19 kB)
Requirement already satisfied: requests>=2.24.0 in
/usr/local/lib/python3.8/dist-packages (from pmlb) (2.25.1)
Requirement already satisfied: pyyaml>=5.3.1 in /usr/local/lib/python3.8/dist-
packages (from pmlb) (6.0)
Requirement already satisfied: pandas>=1.0.5 in /usr/local/lib/python3.8/dist-
packages (from pmlb) (1.3.5)
Requirement already satisfied: python-dateutil>=2.7.3 in
/usr/local/lib/python3.8/dist-packages (from pandas>=1.0.5->pmlb) (2.8.2)
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.8/dist-
packages (from pandas>=1.0.5->pmlb) (2022.7)
Requirement already satisfied: numpy>=1.17.3 in /usr/local/lib/python3.8/dist-
packages (from pandas>=1.0.5->pmlb) (1.21.6)
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.8/dist-
packages (from requests>=2.24.0->pmlb) (2.10)
Requirement already satisfied: chardet<5,>=3.0.2 in
/usr/local/lib/python3.8/dist-packages (from requests>=2.24.0->pmlb) (4.0.0)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in
/usr/local/lib/python3.8/dist-packages (from requests>=2.24.0->pmlb) (1.24.3)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.8/dist-packages (from requests>=2.24.0->pmlb) (2022.12.7)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.8/dist-
packages (from python-dateutil>=2.7.3->pandas>=1.0.5->pmlb) (1.15.0)
Installing collected packages: pmlb
Successfully installed pmlb-1.0.1.post3
```

```
[ ]: from pmlb import fetch_data
import numpy as np
from matplotlib import pyplot as plt
import seaborn as sns
from sklearn.preprocessing import MinMaxScaler
import pandas as pd
from google.colab import drive
```

```
drive.mount('/content/gdrive')
```

Mounted at /content/gdrive

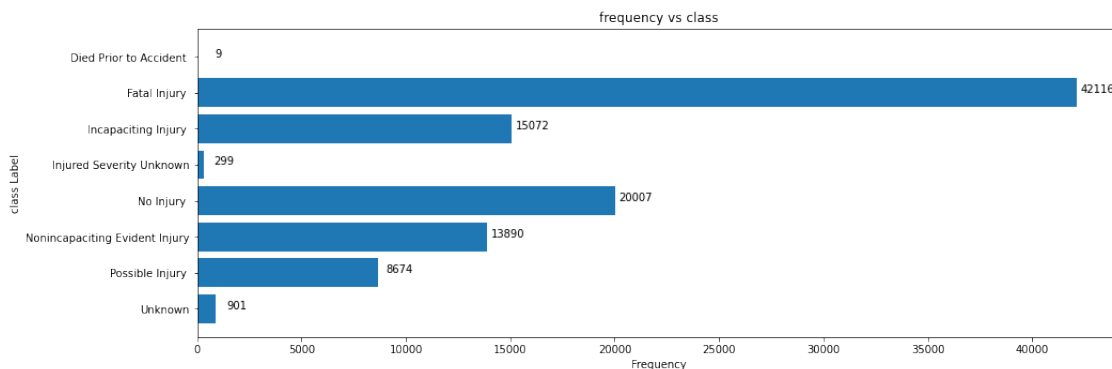
0.1 loading data

```
[ ]: from sklearn.model_selection import GridSearchCV, train_test_split
import random
random.seed(1000)
x_y=fetch_data('fars',return_X_y=0)
x=x_y.iloc[:,0:29]
y=x_y.iloc[:,29]
```

1 Data exploration

Class wise distribution

```
[ ]: import matplotlib.pyplot as plt
fig = plt.figure()
ax = fig.add_axes([0,0,2,1])
labels = ['Died Prior to Accident ', 'Fatal Injury ', 'Incapaciting Injury',
↵, 'Injured Severity Unknown', 'No Injury ', 'Nonincapaciting Evident',
↵, 'Injury', 'Possible Injury ', 'Unknown']
values = [9,42116,15072,299,20007,13890,8674,901]
ax.barh(labels,values)
plt.xlabel("Frequency")
plt.ylabel("class Label")
plt.title("frequency vs class")
ax.invert_yaxis()
for i in range(0,8):
    plt.text(values[i]+1000,i,values[i], ha="center",rotation="horizontal")
plt.show()
```



missing value chacking

```
[ ]: y.isna().sum()
```

```
[ ]: 0
```

```
[ ]: x.isna().all()
```

```
[ ]: CASE_STATE      False
      AGE            False
      SEX            False
      PERSON_TYPE    False
      SEATING_POSITION  False
      RESTRAINT_SYSTEM-USE False
      AIR_BAG_AVAILABILITY/DEPLOYMENT False
      EJECTION        False
      EJECTION_PATH    False
      EXTRICATION      False
      NON_MOTORIST_LOCATION False
      POLICE_REPORTED_ALCOHOL_INVOLVEMENT False
      METHOD_ALCOHOL_DETERMINATION False
      ALCOHOL_TEST_TYPE False
      ALCOHOL_TEST_RESULT False
      POLICE-REPORTED_DRUG_INVOLVEMENT False
      METHOD_OF_DRUG_DETERMINATION False
      DRUG_TEST_TYPE   False
      DRUG_TEST_RESULTS_(1_of_3) False
      DRUG_TEST_TYPE_(2_of_3) False
      DRUG_TEST_RESULTS_(2_of_3) False
      DRUG_TEST_TYPE_(3_of_3) False
      DRUG_TEST_RESULTS_(3_of_3) False
      HISPANIC_ORIGIN  False
      TAKEN_TO_HOSPITAL False
      RELATED_FACTOR_(1)-PERSON_LEVEL False
      RELATED_FACTOR_(2)-PERSON_LEVEL False
      RELATED_FACTOR_(3)-PERSON_LEVEL False
      RACE              False
      dtype: bool
```

No missing data

```
[ ]: x.describe()
```

```
[ ]:      CASE_STATE      AGE      SEX      PERSON_TYPE  \
count  100968.000000  100968.000000  100968.000000  100968.000000
mean      23.425977      37.106707      0.683880      3.155832
std       15.228212      22.109641      0.498972      2.551538
min         0.000000      0.000000      0.000000      0.000000
25%         9.000000      20.000000      0.000000      1.000000
```

50%	22.000000	32.000000	1.000000	1.000000
75%	38.000000	49.000000	1.000000	6.000000
max	50.000000	99.000000	2.000000	9.000000

	SEATING_POSITION	RESTRAINT_SYSTEM-USE \
count	100968.000000	100968.000000
mean	5.990700	6.400394
std	4.794034	1.900097
min	0.000000	0.000000
25%	3.000000	5.000000
50%	3.000000	7.000000
75%	6.000000	7.000000
max	25.000000	11.000000

	AIR_BAG_AVAILABILITY/DEPLOYMENT	EJECTION	EJECTION_PATH \
count	100968.000000	100968.000000	100968.000000
mean	4.979053	0.254407	1.084571
std	3.100175	0.670000	2.843937
min	0.000000	0.000000	0.000000
25%	4.000000	0.000000	0.000000
50%	4.000000	0.000000	0.000000
75%	4.000000	0.000000	0.000000
max	12.000000	3.000000	9.000000

	EXTRICATION	...	DRUG_TEST_TYPE_(2_of_3) \
count	100968.000000	...	100968.000000
mean	0.914656	...	2.253754
std	0.319389	...	0.942899
min	0.000000	...	0.000000
25%	1.000000	...	2.000000
50%	1.000000	...	2.000000
75%	1.000000	...	2.000000
max	2.000000	...	6.000000

	DRUG_TEST_RESULTS_(2_of_3)	DRUG_TEST_TYPE_(3_of_3) \
count	100968.000000	100968.000000
mean	100.089672	2.258102
std	295.089512	0.905708
min	0.000000	0.000000
25%	0.000000	2.000000
50%	0.000000	2.000000
75%	0.000000	2.000000
max	999.000000	6.000000

	DRUG_TEST_RESULTS_(3_of_3)	HISPANIC_ORIGIN	TAKEN_TO_HOSPITAL \
count	100968.000000	100968.000000	100968.000000
mean	95.441556	6.098576	1.056018

std	292.121277	1.063206	0.988896
min	0.000000	0.000000	0.000000
25%	0.000000	6.000000	0.000000
50%	0.000000	6.000000	2.000000
75%	0.000000	6.000000	2.000000
max	999.000000	8.000000	2.000000

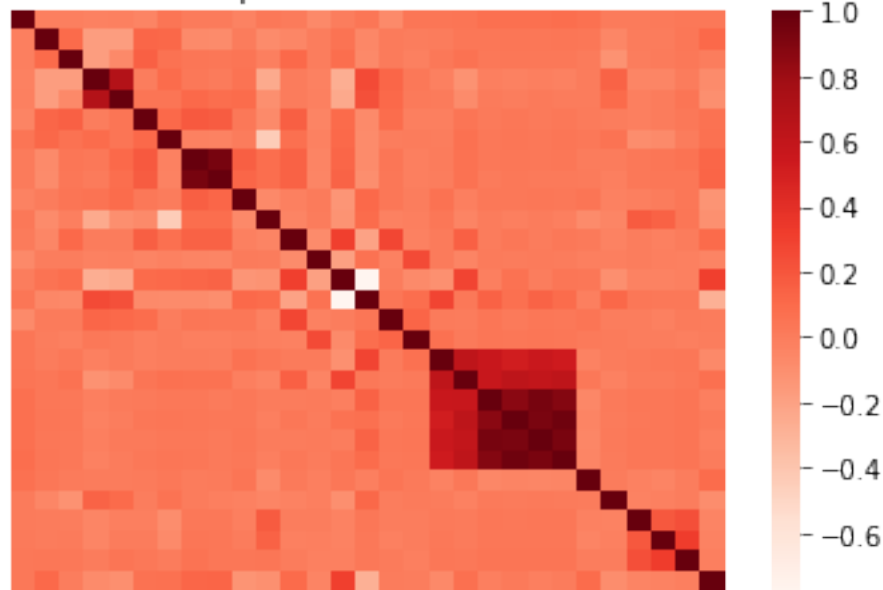
	RELATED_FACTOR_(1)-PERSON_LEVEL	RELATED_FACTOR_(2)-PERSON_LEVEL \
count	100968.000000	100968.000000
mean	26.849645	28.935366
std	3.256276	2.006201
min	0.000000	0.000000
25%	27.000000	29.000000
50%	27.000000	29.000000
75%	27.000000	29.000000
max	44.000000	47.000000

	RELATED_FACTOR_(3)-PERSON_LEVEL	RACE
count	100968.000000	100968.000000
mean	19.007507	12.587612
std	0.807820	3.110742
min	0.000000	0.000000
25%	19.000000	11.000000
50%	19.000000	11.000000
75%	19.000000	15.000000
max	32.000000	17.000000

[8 rows x 29 columns]

```
[ ]: import seaborn as sns
s=sns.heatmap(x.corr(), cmap = 'Reds')
plt.title(" heat map of the first 20 variables")
plt.axis('off')
plt.show()
```

heat map of the first 20 variables



```
#split
```

```
[ ]: train_data, test_data, train_target, test_target = train_test_split(x,y,
    ↪test_size=0.2,shuffle=0,random_state=10)
```

```
#class imbalance-correction
```

```
[ ]: train_target.value_counts()
```

```
[ ]: 1    33680
      4    16064
      2    12043
      5    11042
      6     6963
      7     726
      3     249
      0        7
      Name: target, dtype: int64
```

```
[ ]: from imblearn.over_sampling import SMOTE
      X_resampled_a, y_resampled_a = SMOTE().fit_resample(train_data, train_target)
```

```
[ ]: pd.DataFrame(y_resampled_a).value_counts()
```

```
[ ]: target
      0    33680
```

```

1      33680
2      33680
3      33680
4      33680
5      33680
6      33680
7      33680
dtype: int64

```

```
[ ]: X_resampled_a
```

```

[ ]:
CASE_STATE  AGE  SEX  PERSON_TYPE  SEATING_POSITION  \
0           4   46   1           1           3
1          43   39   1           1           3
2           4   72   0           1           3
3           4   14   0           6           6
4           4   70   0           6           6
...
269435      2   99   2           1           3
269436     38   99   2           1          25
269437      2   99   2           1           3
269438     33   99   2           1           3
269439     38   99   2           1          25

RERAINT_SYSTEM-USE  AIR_BAG_AVAILABILITY/DEPLOYMENT  EJECTION  \
0                   5                                4         0
1                   5                                4         0
2                   5                                0         0
3                   5                                0         0
4                   5                                4         0
...
269435              11                             12         3
269436              11                             12         3
269437              11                             12         0
269438              11                             12         0
269439              11                             12         3

EJECTION_PATH  EXTRICATION  ...  DRUG_TEST_TYPE_(2_of_3)  \
0              0           1  ...                2
1              0           1  ...                2
2              0           2  ...                2
3              0           1  ...                2
4              0           1  ...                2
...
269435      9           2  ...                2
269436      9           2  ...                5
269437      0           1  ...                2

```

269438	0	1	...	2
269439	9	2	...	5

	DRUG_TEST_RESULTS_(2_of_3)	DRUG_TEST_TYPE_(3_of_3)	\
0	0	2	
1	0	2	
2	0	2	
3	0	2	
4	0	2	
...	
269435	0	2	
269436	999	5	
269437	0	2	
269438	0	2	
269439	999	5	

	DRUG_TEST_RESULTS_(3_of_3)	HISPANIC_ORIGIN	TAKEN_TO_HOSPITAL	\
0	0	6	2	
1	0	6	0	
2	0	8	2	
3	0	6	2	
4	0	6	0	
...	
269435	0	6	1	
269436	999	8	1	
269437	0	6	1	
269438	0	6	1	
269439	999	8	1	

	RELATED_FACTOR_(1)-PERSON_LEVEL	RELATED_FACTOR_(2)-PERSON_LEVEL	\
0	27	29	
1	27	29	
2	27	29	
3	27	29	
4	27	29	
...	
269435	27	29	
269436	27	29	
269437	27	29	
269438	27	29	
269439	27	29	

	RELATED_FACTOR_(3)-PERSON_LEVEL	RACE
0	19	11
1	19	11
2	19	15
3	19	11

4	19	11
...
269435	19	11
269436	19	15
269437	19	11
269438	19	11
269439	19	15

[269440 rows x 29 columns]

splitting Data into training and Test

#Normalization

```
[ ]: scaler = MinMaxScaler()
x_scaled = scaler.fit_transform(X_resampled_a)
Y=y_resampled_a
x_scaled_data= scaler.fit_transform(test_data)
```

2 FEATURE SELECTION USING DECISION TREE

```
[ ]: # explore the number of selected features for RFE
from numpy import mean
from numpy import std
from sklearn.datasets import make_classification
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import cross_validate
from sklearn.model_selection import RepeatedStratifiedKFold
from sklearn.feature_selection import RFE
from sklearn.tree import DecisionTreeClassifier
from sklearn.pipeline import Pipeline
from matplotlib import pyplot

#
# get the dataset
def get_dataset():
    X =x_scaled
    y= Y
    return X, y

# get a list of models to evaluate
def get_models():
    models = dict()
    for i in range(10, 25):
        rfe = RFE(estimator=DecisionTreeClassifier(), n_features_to_select=i)
        model = DecisionTreeClassifier()
        models[str(i)] = Pipeline(steps=[('s',rfe),('m',model)])
    return models
```

```

# evaluate a give model using cross-validation
def evaluate_model(model_a, X, y):
    cv_a = cross_validate(model_a, X,y, cv=2,verbose=1,return_estimator=True)
    return cv_a['test_score']

# define dataset
X, y = get_dataset()
# get the models to evaluate
models = get_models()
# evaluate the models and store results
results, names = list(), list()
for name, model in models.items():
    scores = evaluate_model(model, X, y)
    results.append(scores)
    names.append(name)
print('-%s %.3f (%.3f)' % (name, mean(scores), std(scores)))
# plot model performance for comparison

```

```

[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.
[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 32.0s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.

```

```
>10 0.853 (0.005)
```

```

[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 29.4s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.

```

```
>11 0.856 (0.006)
```

```

[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 29.4s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.

```

```
>12 0.857 (0.006)
```

```

[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 28.6s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.

```

```
>13 0.858 (0.006)
```

```

[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 30.3s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.

```

```
>14 0.859 (0.006)
```

```

[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 28.4s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.

```

```
>15 0.859 (0.006)
```

```

[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 25.6s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.

```

```
>16 0.861 (0.004)
```

```
[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 23.2s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.
>17 0.864 (0.007)

[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 24.5s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.
>18 0.867 (0.004)

[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 19.9s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.
>19 0.870 (0.007)

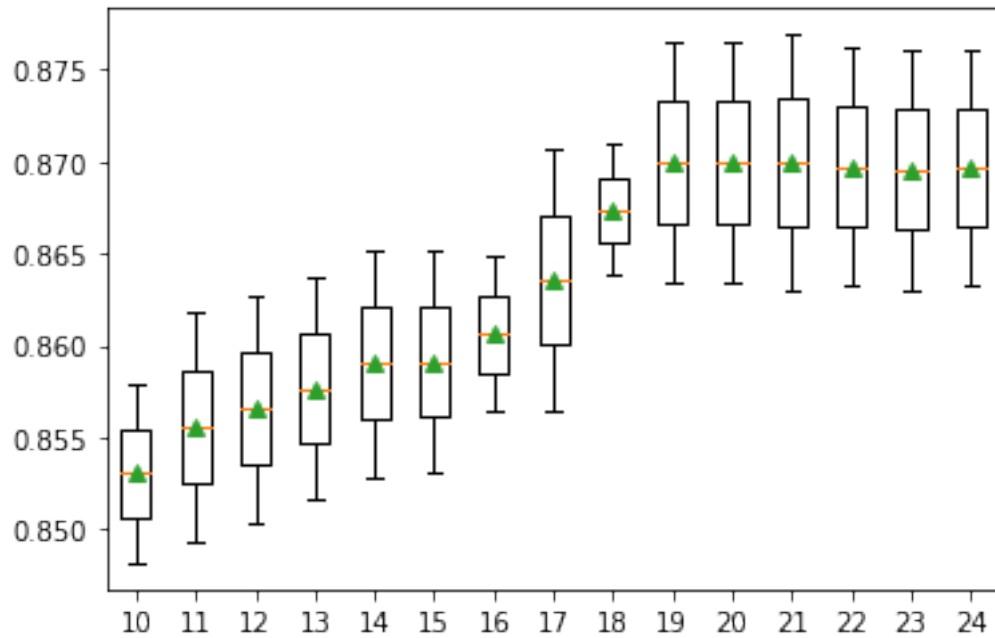
[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 18.4s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.
>20 0.870 (0.007)

[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 16.8s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.
>21 0.870 (0.007)

[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 15.5s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.
>22 0.870 (0.006)

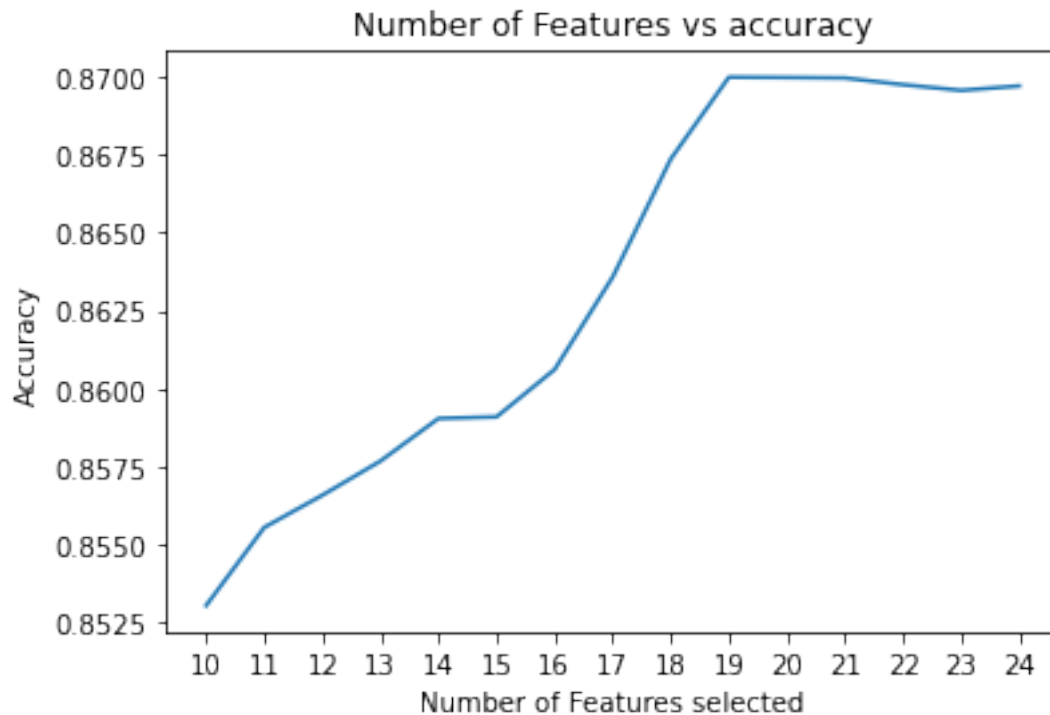
[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 14.2s finished
[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.
>23 0.870 (0.007)

[Parallel(n_jobs=1)]: Done 2 out of 2 | elapsed: 12.4s finished
>24 0.870 (0.006)
```



```
[ ]: x=pyplot.plot(names,mean(results,axis=1))
pyplot.xlabel("Number of Features selected")
pyplot.ylabel("Accuracy")
pyplot.title("Number of Features vs accuracy")
```

```
[ ]: Text(0.5, 1.0, 'Number of Features vs accuracy')
```



Result- Number of features are 18 since accuracy is higher and low standard deviation

Finding the selected variables

```
[ ]: from sklearn.feature_selection import RFE
from sklearn.tree import DecisionTreeClassifier
# load data
X = x_scaled
Y = Y
# feature extraction
model_d = DecisionTreeClassifier()
rfe_a = RFE(estimator=model_d, n_features_to_select=18)
fit = rfe_a.fit(X, Y)
print("Num Features: %d" % fit.n_features_)
```

Num Features: 18

```
[ ]: d={"variables": x.columns, "result": fit.support_, "rank": fit.ranking_}
k=pd.DataFrame(d, index=None)
k
```

```
[ ]:
      variables  result  rank
0    CASE_STATE    True    1
1         AGE    True    1
```

2	SEX	True	1
3	PERSON_TYPE	True	1
4	SEATING_POSITION	True	1
5	RESTRAINT_SYSTEM-USE	True	1
6	AIR_BAG_AVAILABILITY/DEPLOYMENT	True	1
7	EJECTION	False	4
8	EJECTION_PATH	True	1
9	EXTRICATION	False	2
10	NON_MOTORIST_LOCATION	False	7
11	POLICE_REPORTED_ALCOHOL_INVOLVEMENT	True	1
12	METHOD_ALCOHOL_DETERMINATION	True	1
13	ALCOHOL_TEST_TYPE	True	1
14	ALCOHOL_TEST_RESULT	True	1
15	POLICE-REPORTED_DRUG_INVOLVEMENT	True	1
16	METHOD_OF_DRUG_DETERMINATION	False	6
17	DRUG_TEST_TYPE	False	3
18	DRUG_TEST_RESULTS_(1_of_3)	True	1
19	DRUG_TEST_TYPE_(2_of_3)	False	9
20	DRUG_TEST_RESULTS_(2_of_3)	False	8
21	DRUG_TEST_TYPE_(3_of_3)	True	1
22	DRUG_TEST_RESULTS_(3_of_3)	False	5
23	HISPANIC_ORIGIN	False	10
24	TAKEN_TO_HOSPITAL	True	1
25	RELATED_FACTOR_(1)-PERSON_LEVEL	True	1
26	RELATED_FACTOR_(2)-PERSON_LEVEL	False	11
27	RELATED_FACTOR_(3)-PERSON_LEVEL	False	12
28	RACE	True	1

##selected variables

```
[ ]: k_selected=k[k["result"]==True]
      k_selected
```

	variables	result	rank
0	CASE_STATE	True	1
1	AGE	True	1
2	SEX	True	1
3	PERSON_TYPE	True	1
4	SEATING_POSITION	True	1
5	RESTRAINT_SYSTEM-USE	True	1
6	AIR_BAG_AVAILABILITY/DEPLOYMENT	True	1
8	EJECTION_PATH	True	1
11	POLICE_REPORTED_ALCOHOL_INVOLVEMENT	True	1
12	METHOD_ALCOHOL_DETERMINATION	True	1
13	ALCOHOL_TEST_TYPE	True	1
14	ALCOHOL_TEST_RESULT	True	1
15	POLICE-REPORTED_DRUG_INVOLVEMENT	True	1

18	DRUG_TEST_RESULTS_(1_of_3)	True	1
21	DRUG_TEST_TYPE_(3_of_3)	True	1
24	TAKEN_TO_HOSPITAL	True	1
25	RELATED_FACTOR_(1)-PERSON_LEVEL	True	1
28	RACE	True	1

##Ignored variables

```
[ ]: k_not_selected=k[k["result"]==False]
      k_not_selected
```

```
[ ]:
      variables  result  rank
7      EJECTION   False    4
9      EXTRICATION  False    2
10     NON_MOTORIST_LOCATION  False    7
16     METHOD_OF_DRUG_DETERMINATION  False    6
17      DRUG_TEST_TYPE  False    3
19      DRUG_TEST_TYPE_(2_of_3)  False    9
20      DRUG_TEST_RESULTS_(2_of_3)  False    8
22      DRUG_TEST_RESULTS_(3_of_3)  False    5
23      HISPANIC_ORIGIN  False   10
26  RELATED_FACTOR_(2)-PERSON_LEVEL  False   11
27  RELATED_FACTOR_(3)-PERSON_LEVEL  False   12
```

2.1 dropping traindata

```
[ ]: x_new=X_resampled_a.drop(k_not_selected["variables"],axis=1)
```

Feature selection completed

2.2 Normalisation of selected data

```
[ ]: scaler = MinMaxScaler()
      x_scaled_new = scaler.fit_transform(x_new)
```

2.3 Test_data preparation

2.3.1 feature selection

```
[ ]: test_data_new=test_data.drop(k_not_selected["variables"],axis=1)
```

2.3.2 Normalisation

```
[ ]: scaler = MinMaxScaler()
      x_test_scaled = scaler.fit_transform(test_data_new)
```

3 Decision Tree

3.1 Hyper parameter tuning (GridSearchCV)

```
[ ]: from sklearn.tree import DecisionTreeClassifier
      from sklearn.model_selection import GridSearchCV
      from sklearn.metrics import classification_report, confusion_matrix
      parameters = {'criterion':('gini', 'entropy'), 'max_depth':np.
        ↳arange(8,12),'splitter': ['best'],'min_impurity_decrease':np.arange(0,0.2,0.
        ↳05),'min_samples_leaf':(1,2)}
      model=DecisionTreeClassifier()
      clf_a = GridSearchCV(model, parameters,verbose=3,scoring='accuracy',cv=10)
      clf_a.fit(x_scaled_new,Y)
      print(clf_a.best_params_)
```

Fitting 10 folds for each of 32 candidates, totalling 320 fits

```
[CV 1/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0,
splitter=best;; score=0.698 total time= 1.0s
[CV 2/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0,
splitter=best;; score=0.700 total time= 0.9s
[CV 3/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0,
splitter=best;; score=0.711 total time= 0.9s
[CV 4/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0,
splitter=best;; score=0.712 total time= 0.9s
[CV 5/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0,
splitter=best;; score=0.712 total time= 0.9s
[CV 6/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0,
splitter=best;; score=0.712 total time= 0.9s
[CV 7/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0,
splitter=best;; score=0.714 total time= 0.9s
[CV 8/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0,
splitter=best;; score=0.713 total time= 0.9s
[CV 9/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0,
splitter=best;; score=0.711 total time= 0.9s
[CV 10/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0,
splitter=best;; score=0.713 total time= 0.9s
[CV 1/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05,
splitter=best;; score=0.546 total time= 0.5s
[CV 2/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05,
splitter=best;; score=0.550 total time= 0.5s
[CV 3/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05,
splitter=best;; score=0.551 total time= 0.5s
[CV 4/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05,
splitter=best;; score=0.549 total time= 0.5s
[CV 5/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05,
splitter=best;; score=0.550 total time= 0.5s
[CV 6/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05,
splitter=best;; score=0.548 total time= 0.5s
```


[CV 7/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05,
 splitter=best;; score=0.550 total time= 0.6s
 [CV 8/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05,
 splitter=best;; score=0.551 total time= 0.6s
 [CV 9/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05,
 splitter=best;; score=0.550 total time= 0.6s
 [CV 10/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05,
 splitter=best;; score=0.552 total time= 0.6s
 [CV 1/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 2/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 3/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 4/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 5/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 6/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 7/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 8/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 9/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 10/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 1/10] END criterion=gini, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 2/10] END criterion=gini, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 3/10] END criterion=gini, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 4/10] END criterion=gini, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 5/10] END criterion=gini, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 6/10] END criterion=gini, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 7/10] END criterion=gini, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total

```

time= 0.2s
[CV 8/10] END criterion=gini, max_depth=8,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.125 total
time= 0.2s
[CV 9/10] END criterion=gini, max_depth=8,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.125 total
time= 0.2s
[CV 10/10] END criterion=gini, max_depth=8,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.125 total
time= 0.2s
[CV 1/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0,
splitter=best;, score=0.712 total time= 1.0s
[CV 2/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0,
splitter=best;, score=0.712 total time= 0.9s
[CV 3/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0,
splitter=best;, score=0.723 total time= 0.9s
[CV 4/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0,
splitter=best;, score=0.724 total time= 1.0s
[CV 5/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0,
splitter=best;, score=0.724 total time= 1.0s
[CV 6/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0,
splitter=best;, score=0.724 total time= 1.0s
[CV 7/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0,
splitter=best;, score=0.723 total time= 1.0s
[CV 8/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0,
splitter=best;, score=0.725 total time= 0.9s
[CV 9/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0,
splitter=best;, score=0.723 total time= 1.0s
[CV 10/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0,
splitter=best;, score=0.725 total time= 1.0s
[CV 1/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05,
splitter=best;, score=0.546 total time= 0.6s
[CV 2/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05,
splitter=best;, score=0.550 total time= 0.6s
[CV 3/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05,
splitter=best;, score=0.551 total time= 0.6s
[CV 4/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05,
splitter=best;, score=0.549 total time= 0.6s
[CV 5/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05,
splitter=best;, score=0.550 total time= 0.5s
[CV 6/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05,
splitter=best;, score=0.548 total time= 0.5s
[CV 7/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05,
splitter=best;, score=0.550 total time= 0.5s
[CV 8/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05,
splitter=best;, score=0.551 total time= 0.6s
[CV 9/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05,
splitter=best;, score=0.550 total time= 0.5s

```

[CV 10/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05,
 splitter=best;; score=0.552 total time= 0.6s
 [CV 1/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 2/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 3/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 4/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 5/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 6/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 7/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 8/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 9/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 10/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 1/10] END criterion=gini, max_depth=9,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 2/10] END criterion=gini, max_depth=9,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 3/10] END criterion=gini, max_depth=9,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 4/10] END criterion=gini, max_depth=9,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 5/10] END criterion=gini, max_depth=9,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 6/10] END criterion=gini, max_depth=9,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 7/10] END criterion=gini, max_depth=9,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 8/10] END criterion=gini, max_depth=9,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 9/10] END criterion=gini, max_depth=9,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total

```

time= 0.2s
[CV 10/10] END criterion=gini, max_depth=9,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.125 total
time= 0.2s
[CV 1/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0,
splitter=best;, score=0.720 total time= 1.0s
[CV 2/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0,
splitter=best;, score=0.724 total time= 1.0s
[CV 3/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0,
splitter=best;, score=0.735 total time= 1.0s
[CV 4/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0,
splitter=best;, score=0.735 total time= 1.0s
[CV 5/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0,
splitter=best;, score=0.732 total time= 1.0s
[CV 6/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0,
splitter=best;, score=0.735 total time= 1.0s
[CV 7/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0,
splitter=best;, score=0.735 total time= 1.0s
[CV 8/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0,
splitter=best;, score=0.735 total time= 1.0s
[CV 9/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0,
splitter=best;, score=0.735 total time= 1.0s
[CV 10/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0,
splitter=best;, score=0.735 total time= 1.0s
[CV 1/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05,
splitter=best;, score=0.546 total time= 0.5s
[CV 2/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05,
splitter=best;, score=0.550 total time= 0.6s
[CV 3/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05,
splitter=best;, score=0.551 total time= 0.5s
[CV 4/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05,
splitter=best;, score=0.549 total time= 0.6s
[CV 5/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05,
splitter=best;, score=0.550 total time= 0.6s
[CV 6/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05,
splitter=best;, score=0.548 total time= 0.6s
[CV 7/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05,
splitter=best;, score=0.550 total time= 0.6s
[CV 8/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05,
splitter=best;, score=0.551 total time= 0.6s
[CV 9/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05,
splitter=best;, score=0.550 total time= 0.6s
[CV 10/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05,
splitter=best;, score=0.552 total time= 0.6s
[CV 1/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1,
splitter=best;, score=0.125 total time= 0.2s
[CV 2/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1,
splitter=best;, score=0.125 total time= 0.2s

```

[CV 3/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 4/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 5/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 6/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 7/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 8/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 9/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 10/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 1/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 2/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 3/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 4/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 5/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 6/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 7/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 8/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 9/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 10/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 1/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0,
 splitter=best;; score=0.727 total time= 1.2s

[CV 2/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0,
 splitter=best;; score=0.731 total time= 1.3s
 [CV 3/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0,
 splitter=best;; score=0.741 total time= 1.1s
 [CV 4/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0,
 splitter=best;; score=0.745 total time= 1.0s
 [CV 5/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0,
 splitter=best;; score=0.742 total time= 1.1s
 [CV 6/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0,
 splitter=best;; score=0.744 total time= 1.0s
 [CV 7/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0,
 splitter=best;; score=0.744 total time= 1.0s
 [CV 8/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0,
 splitter=best;; score=0.743 total time= 1.0s
 [CV 9/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0,
 splitter=best;; score=0.742 total time= 1.0s
 [CV 10/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0,
 splitter=best;; score=0.743 total time= 1.0s
 [CV 1/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05,
 splitter=best;; score=0.546 total time= 0.5s
 [CV 2/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05,
 splitter=best;; score=0.550 total time= 0.5s
 [CV 3/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05,
 splitter=best;; score=0.551 total time= 0.5s
 [CV 4/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05,
 splitter=best;; score=0.549 total time= 0.6s
 [CV 5/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05,
 splitter=best;; score=0.550 total time= 0.5s
 [CV 6/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05,
 splitter=best;; score=0.548 total time= 0.6s
 [CV 7/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05,
 splitter=best;; score=0.550 total time= 0.5s
 [CV 8/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05,
 splitter=best;; score=0.551 total time= 0.6s
 [CV 9/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05,
 splitter=best;; score=0.550 total time= 0.5s
 [CV 10/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05,
 splitter=best;; score=0.552 total time= 0.6s
 [CV 1/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 2/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 3/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 4/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 5/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s

[CV 6/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 7/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 8/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 9/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 10/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1,
 splitter=best;; score=0.125 total time= 0.2s
 [CV 1/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 2/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 3/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 4/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 5/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 6/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 7/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 8/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 9/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 10/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.125 total
 time= 0.2s
 [CV 1/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0,
 splitter=best;; score=0.706 total time= 0.9s
 [CV 2/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0,
 splitter=best;; score=0.706 total time= 0.9s
 [CV 3/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0,
 splitter=best;; score=0.718 total time= 1.2s
 [CV 4/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0,
 splitter=best;; score=0.719 total time= 1.4s

[CV 5/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0,
 splitter=best;; score=0.721 total time= 1.6s
 [CV 6/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0,
 splitter=best;; score=0.723 total time= 1.6s
 [CV 7/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0,
 splitter=best;; score=0.724 total time= 1.9s
 [CV 8/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0,
 splitter=best;; score=0.721 total time= 1.6s
 [CV 9/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0,
 splitter=best;; score=0.723 total time= 1.8s
 [CV 10/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0,
 splitter=best;; score=0.722 total time= 1.8s
 [CV 1/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05,
 splitter=best;; score=0.614 total time= 1.5s
 [CV 2/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05,
 splitter=best;; score=0.615 total time= 1.1s
 [CV 3/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05,
 splitter=best;; score=0.628 total time= 1.1s
 [CV 4/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05,
 splitter=best;; score=0.627 total time= 1.1s
 [CV 5/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05,
 splitter=best;; score=0.628 total time= 1.1s
 [CV 6/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05,
 splitter=best;; score=0.630 total time= 1.0s
 [CV 7/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05,
 splitter=best;; score=0.630 total time= 1.1s
 [CV 8/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05,
 splitter=best;; score=0.632 total time= 1.0s
 [CV 9/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05,
 splitter=best;; score=0.630 total time= 0.7s
 [CV 10/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05,
 splitter=best;; score=0.633 total time= 0.6s
 [CV 1/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.559 total time= 0.5s
 [CV 2/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 3/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 4/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.559 total time= 0.5s
 [CV 5/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 6/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 7/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.560 total time= 0.5s
 [CV 8/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.563 total time= 0.5s

[CV 9/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 10/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1,
 splitter=best;; score=0.565 total time= 0.5s
 [CV 1/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.559 total
 time= 0.5s
 [CV 2/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total
 time= 0.5s
 [CV 3/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total
 time= 0.5s
 [CV 4/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.559 total
 time= 0.5s
 [CV 5/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total
 time= 0.5s
 [CV 6/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total
 time= 0.5s
 [CV 7/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.560 total
 time= 0.5s
 [CV 8/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.563 total
 time= 0.5s
 [CV 9/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total
 time= 0.5s
 [CV 10/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.565 total
 time= 0.5s
 [CV 1/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0,
 splitter=best;; score=0.718 total time= 0.9s
 [CV 2/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0,
 splitter=best;; score=0.716 total time= 0.9s
 [CV 3/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0,
 splitter=best;; score=0.726 total time= 0.9s
 [CV 4/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0,
 splitter=best;; score=0.726 total time= 0.9s
 [CV 5/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0,
 splitter=best;; score=0.729 total time= 0.9s
 [CV 6/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0,
 splitter=best;; score=0.737 total time= 0.9s
 [CV 7/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0,
 splitter=best;; score=0.737 total time= 0.9s

[CV 8/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0,
 splitter=best;; score=0.730 total time= 0.9s
 [CV 9/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0,
 splitter=best;; score=0.733 total time= 0.9s
 [CV 10/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0,
 splitter=best;; score=0.729 total time= 0.9s
 [CV 1/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05,
 splitter=best;; score=0.614 total time= 0.6s
 [CV 2/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05,
 splitter=best;; score=0.615 total time= 0.6s
 [CV 3/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05,
 splitter=best;; score=0.628 total time= 0.6s
 [CV 4/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05,
 splitter=best;; score=0.627 total time= 0.6s
 [CV 5/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05,
 splitter=best;; score=0.628 total time= 0.6s
 [CV 6/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05,
 splitter=best;; score=0.630 total time= 0.6s
 [CV 7/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05,
 splitter=best;; score=0.630 total time= 0.6s
 [CV 8/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05,
 splitter=best;; score=0.632 total time= 0.6s
 [CV 9/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05,
 splitter=best;; score=0.630 total time= 0.6s
 [CV 10/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05,
 splitter=best;; score=0.633 total time= 0.6s
 [CV 1/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.559 total time= 0.5s
 [CV 2/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 3/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 4/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.559 total time= 0.5s
 [CV 5/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 6/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 7/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.560 total time= 0.5s
 [CV 8/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.563 total time= 0.5s
 [CV 9/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 10/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1,
 splitter=best;; score=0.565 total time= 0.5s
 [CV 1/10] END criterion=entropy, max_depth=9,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.559 total

```

time= 0.5s
[CV 2/10] END criterion=entropy, max_depth=9,
min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total
time= 0.5s
[CV 3/10] END criterion=entropy, max_depth=9,
min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total
time= 0.5s
[CV 4/10] END criterion=entropy, max_depth=9,
min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.559 total
time= 0.5s
[CV 5/10] END criterion=entropy, max_depth=9,
min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total
time= 0.5s
[CV 6/10] END criterion=entropy, max_depth=9,
min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total
time= 0.5s
[CV 7/10] END criterion=entropy, max_depth=9,
min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.560 total
time= 0.5s
[CV 8/10] END criterion=entropy, max_depth=9,
min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.563 total
time= 0.5s
[CV 9/10] END criterion=entropy, max_depth=9,
min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total
time= 0.5s
[CV 10/10] END criterion=entropy, max_depth=9,
min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.565 total
time= 0.5s
[CV 1/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0,
splitter=best;; score=0.722 total time= 0.9s
[CV 2/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0,
splitter=best;; score=0.724 total time= 0.9s
[CV 3/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0,
splitter=best;; score=0.733 total time= 1.0s
[CV 4/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0,
splitter=best;; score=0.731 total time= 0.9s
[CV 5/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0,
splitter=best;; score=0.733 total time= 0.9s
[CV 6/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0,
splitter=best;; score=0.745 total time= 1.0s
[CV 7/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0,
splitter=best;; score=0.742 total time= 0.9s
[CV 8/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0,
splitter=best;; score=0.736 total time= 0.9s
[CV 9/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0,
splitter=best;; score=0.738 total time= 0.9s
[CV 10/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0,
splitter=best;; score=0.738 total time= 1.0s

```

[CV 1/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05,
 splitter=best;; score=0.614 total time= 0.6s
 [CV 2/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05,
 splitter=best;; score=0.615 total time= 0.6s
 [CV 3/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05,
 splitter=best;; score=0.628 total time= 0.6s
 [CV 4/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05,
 splitter=best;; score=0.627 total time= 0.6s
 [CV 5/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05,
 splitter=best;; score=0.628 total time= 0.6s
 [CV 6/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05,
 splitter=best;; score=0.630 total time= 0.6s
 [CV 7/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05,
 splitter=best;; score=0.630 total time= 0.6s
 [CV 8/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05,
 splitter=best;; score=0.632 total time= 0.6s
 [CV 9/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05,
 splitter=best;; score=0.630 total time= 0.6s
 [CV 10/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05,
 splitter=best;; score=0.633 total time= 0.6s
 [CV 1/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.559 total time= 0.5s
 [CV 2/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 3/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 4/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.559 total time= 0.5s
 [CV 5/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 6/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 7/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.560 total time= 0.5s
 [CV 8/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.563 total time= 0.5s
 [CV 9/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.562 total time= 0.5s
 [CV 10/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1,
 splitter=best;; score=0.565 total time= 0.5s
 [CV 1/10] END criterion=entropy, max_depth=10,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.559 total
 time= 0.5s
 [CV 2/10] END criterion=entropy, max_depth=10,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total
 time= 0.5s
 [CV 3/10] END criterion=entropy, max_depth=10,
 min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total

```

time= 0.5s
[CV 4/10] END criterion=entropy, max_depth=10,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.559 total
time= 0.5s
[CV 5/10] END criterion=entropy, max_depth=10,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.562 total
time= 0.5s
[CV 6/10] END criterion=entropy, max_depth=10,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.562 total
time= 0.5s
[CV 7/10] END criterion=entropy, max_depth=10,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.560 total
time= 0.5s
[CV 8/10] END criterion=entropy, max_depth=10,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.563 total
time= 0.5s
[CV 9/10] END criterion=entropy, max_depth=10,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.562 total
time= 0.6s
[CV 10/10] END criterion=entropy, max_depth=10,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.565 total
time= 0.9s
[CV 1/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0,
splitter=best;, score=0.729 total time= 2.1s
[CV 2/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0,
splitter=best;, score=0.730 total time= 2.0s
[CV 3/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0,
splitter=best;, score=0.741 total time= 2.9s
[CV 4/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0,
splitter=best;, score=0.740 total time= 1.8s
[CV 5/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0,
splitter=best;, score=0.741 total time= 1.0s
[CV 6/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0,
splitter=best;, score=0.751 total time= 1.0s
[CV 7/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0,
splitter=best;, score=0.748 total time= 1.0s
[CV 8/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0,
splitter=best;, score=0.747 total time= 1.0s
[CV 9/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0,
splitter=best;, score=0.745 total time= 1.0s
[CV 10/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0,
splitter=best;, score=0.748 total time= 1.0s
[CV 1/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05,
splitter=best;, score=0.614 total time= 0.6s
[CV 2/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05,
splitter=best;, score=0.615 total time= 0.6s
[CV 3/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05,
splitter=best;, score=0.628 total time= 0.6s

```

[CV 4/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05, splitter=best;; score=0.627 total time= 0.6s

[CV 5/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05, splitter=best;; score=0.628 total time= 0.6s

[CV 6/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05, splitter=best;; score=0.630 total time= 0.6s

[CV 7/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05, splitter=best;; score=0.630 total time= 0.6s

[CV 8/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05, splitter=best;; score=0.632 total time= 0.6s

[CV 9/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05, splitter=best;; score=0.630 total time= 0.6s

[CV 10/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05, splitter=best;; score=0.633 total time= 0.6s

[CV 1/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1, splitter=best;; score=0.559 total time= 0.5s

[CV 2/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1, splitter=best;; score=0.562 total time= 0.5s

[CV 3/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1, splitter=best;; score=0.562 total time= 0.5s

[CV 4/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1, splitter=best;; score=0.559 total time= 0.5s

[CV 5/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1, splitter=best;; score=0.562 total time= 0.5s

[CV 6/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1, splitter=best;; score=0.562 total time= 0.5s

[CV 7/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1, splitter=best;; score=0.560 total time= 0.5s

[CV 8/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1, splitter=best;; score=0.563 total time= 0.5s

[CV 9/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1, splitter=best;; score=0.562 total time= 0.5s

[CV 10/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1, splitter=best;; score=0.565 total time= 0.5s

[CV 1/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.559 total time= 0.6s

[CV 2/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total time= 0.5s

[CV 3/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total time= 0.5s

[CV 4/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.559 total time= 0.5s

[CV 5/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.15000000000000002, splitter=best;; score=0.562 total

```

time= 0.9s
[CV 6/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.562 total
time= 0.9s
[CV 7/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.560 total
time= 1.2s
[CV 8/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.563 total
time= 1.0s
[CV 9/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.562 total
time= 0.9s
[CV 10/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002, splitter=best;, score=0.565 total
time= 1.0s
{'criterion': 'entropy', 'max_depth': 11, 'min_impurity_decrease': 0.0,
'splitter': 'best'}

```

3.2 predictions

```

[ ]: grid_predictions = clf_a.predict(x_test_scaled)

# print classification report
print(classification_report(test_target, grid_predictions))

```

	precision	recall	f1-score	support
0	0.00	0.00	0.00	2
1	1.00	1.00	1.00	8436
2	0.59	0.37	0.46	3029
3	0.13	0.58	0.21	50
4	0.84	0.90	0.87	3943
5	0.43	0.68	0.53	2848
6	0.36	0.07	0.12	1711
7	0.28	0.79	0.41	175
accuracy			0.76	20194
macro avg	0.45	0.55	0.45	20194
weighted avg	0.76	0.76	0.74	20194

F1 score (weighted)=0.75

3.3 conclusion

```
[ ]: from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.model_selection import cross_validate
model=DecisionTreeClassifier(criterion='entropy',max_depth=12,min_impurity_decrease=0.
    ↪0,splitter='best',min_samples_leaf=2)
cv_results_a = cross_validate(model, x_scaled_new,Y,
    ↪cv=10,verbose=1,return_estimator=True)
accuracies_a = cv_results_a['test_score']
accuracies_a=list(accuracies_a)
models=cv_results_a['estimator']
model = models[accuracies_a.index(max(accuracies_a))]
predictions = model.predict(x_test_scaled)

# print classification report
print(classification_report(test_target, predictions))
```

[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.

	precision	recall	f1-score	support
0	0.00	0.00	0.00	2
1	1.00	1.00	1.00	8436
2	0.49	0.68	0.57	3029
3	0.13	0.48	0.21	50
4	0.85	0.91	0.88	3943
5	0.44	0.36	0.40	2848
6	0.35	0.11	0.17	1711
7	0.40	0.77	0.52	175
accuracy			0.76	20194
macro avg	0.46	0.54	0.47	20194
weighted avg	0.75	0.76	0.75	20194

[Parallel(n_jobs=1)]: Done 10 out of 10 | elapsed: 12.7s finished

```
[ ]: # saving the model
import pickle
filename = '/content/gdrive/MyDrive/Ml_project/fars_models/decision_tree.sav'
pickle.dump(model, open(filename, 'wb'))
```


3.4 Tree image

```
[ ]: import matplotlib.pyplot as plt
      from sklearn import tree
      %matplotlib inline
      plt.figure(figsize=(30,30))
      tree.plot_tree(model, filled=0, fontsize=8,max_depth=5)
```

3.5 Accuracy =0.75

#Random Forest

3.6 Hyper parameter tuning (GridSearch CV)

```
[ ]: from sklearn.ensemble import RandomForestClassifier
      from sklearn.model_selection import GridSearchCV
      from sklearn.metrics import classification_report, confusion_matrix
      parameters = {'criterion':('gini', 'entropy'), 'max_depth':np.
        ↳arange(8,12),'min_impurity_decrease':np.arange(0,0.2,0.05)}
      model=RandomForestClassifier()
      clf_a = GridSearchCV(model, parameters,verbose=3,scoring='accuracy',cv=10)
      clf_a.fit(x_scaled_new,Y)
      print(clf_a.best_params_)
```

Fitting 10 folds for each of 32 candidates, totalling 320 fits

```
[CV 1/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0;,
score=0.711 total time= 20.8s
[CV 2/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0;,
score=0.709 total time= 20.2s
[CV 3/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0;,
score=0.724 total time= 20.1s
[CV 4/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0;,
score=0.728 total time= 17.8s
[CV 5/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0;,
score=0.728 total time= 16.9s
[CV 6/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0;,
score=0.724 total time= 17.5s
[CV 7/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0;,
score=0.726 total time= 18.6s
[CV 8/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0;,
score=0.728 total time= 23.2s
[CV 9/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0;,
score=0.729 total time= 20.9s
[CV 10/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.0;,
score=0.728 total time= 20.1s
[CV 1/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05;,
score=0.602 total time= 9.9s
[CV 2/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05;,
```

```

score=0.576 total time= 12.9s
[CV 3/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05;,
score=0.576 total time= 10.7s
[CV 4/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05;,
score=0.594 total time= 12.7s
[CV 5/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05;,
score=0.573 total time= 10.6s
[CV 6/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05;,
score=0.604 total time= 8.5s
[CV 7/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05;,
score=0.601 total time= 8.6s
[CV 8/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05;,
score=0.578 total time= 8.5s
[CV 9/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05;,
score=0.601 total time= 8.4s
[CV 10/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.05;,
score=0.603 total time= 9.9s
[CV 1/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1;,
score=0.125 total time= 6.1s
[CV 2/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1;,
score=0.125 total time= 5.5s
[CV 3/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1;,
score=0.125 total time= 5.6s
[CV 4/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1;,
score=0.125 total time= 5.5s
[CV 5/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1;,
score=0.125 total time= 5.6s
[CV 6/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1;,
score=0.125 total time= 5.8s
[CV 7/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1;,
score=0.125 total time= 6.3s
[CV 8/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1;,
score=0.125 total time= 5.9s
[CV 9/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1;,
score=0.125 total time= 6.7s
[CV 10/10] END criterion=gini, max_depth=8, min_impurity_decrease=0.1;,
score=0.125 total time= 7.0s
[CV 1/10] END criterion=gini, max_depth=8,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 6.4s
[CV 2/10] END criterion=gini, max_depth=8,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 5.9s
[CV 3/10] END criterion=gini, max_depth=8,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 6.4s
[CV 4/10] END criterion=gini, max_depth=8,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 5.9s
[CV 5/10] END criterion=gini, max_depth=8,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 6.0s
[CV 6/10] END criterion=gini, max_depth=8,

```

```

min_impurity_decrease=0.15000000000000002;; score=0.125 total time= 5.4s
[CV 7/10] END criterion=gini, max_depth=8,
min_impurity_decrease=0.15000000000000002;; score=0.125 total time= 5.3s
[CV 8/10] END criterion=gini, max_depth=8,
min_impurity_decrease=0.15000000000000002;; score=0.125 total time= 6.7s
[CV 9/10] END criterion=gini, max_depth=8,
min_impurity_decrease=0.15000000000000002;; score=0.125 total time= 6.7s
[CV 10/10] END criterion=gini, max_depth=8,
min_impurity_decrease=0.15000000000000002;; score=0.125 total time= 6.1s
[CV 1/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0;;
score=0.719 total time= 21.0s
[CV 2/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0;;
score=0.719 total time= 20.4s
[CV 3/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0;;
score=0.733 total time= 19.6s
[CV 4/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0;;
score=0.740 total time= 18.5s
[CV 5/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0;;
score=0.734 total time= 19.7s
[CV 6/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0;;
score=0.732 total time= 19.2s
[CV 7/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0;;
score=0.738 total time= 19.8s
[CV 8/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0;;
score=0.736 total time= 20.7s
[CV 9/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0;;
score=0.734 total time= 20.7s
[CV 10/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.0;;
score=0.738 total time= 19.0s
[CV 1/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05;;
score=0.593 total time= 8.5s
[CV 2/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05;;
score=0.593 total time= 8.7s
[CV 3/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05;;
score=0.593 total time= 9.5s
[CV 4/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05;;
score=0.598 total time= 9.7s
[CV 5/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05;;
score=0.573 total time= 8.9s
[CV 6/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05;;
score=0.605 total time= 11.1s
[CV 7/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05;;
score=0.574 total time= 10.4s
[CV 8/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05;;
score=0.602 total time= 12.0s
[CV 9/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05;;
score=0.602 total time= 10.1s
[CV 10/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.05;;

```

```

score=0.579 total time= 9.7s
[CV 1/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1;,
score=0.125 total time= 6.2s
[CV 2/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1;,
score=0.125 total time= 6.3s
[CV 3/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1;,
score=0.125 total time= 5.9s
[CV 4/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1;,
score=0.125 total time= 6.1s
[CV 5/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1;,
score=0.125 total time= 6.1s
[CV 6/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1;,
score=0.125 total time= 5.8s
[CV 7/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1;,
score=0.125 total time= 6.4s
[CV 8/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1;,
score=0.125 total time= 6.0s
[CV 9/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1;,
score=0.125 total time= 5.7s
[CV 10/10] END criterion=gini, max_depth=9, min_impurity_decrease=0.1;,
score=0.125 total time= 5.7s
[CV 1/10] END criterion=gini, max_depth=9,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 5.6s
[CV 2/10] END criterion=gini, max_depth=9,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 6.3s
[CV 3/10] END criterion=gini, max_depth=9,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 5.8s
[CV 4/10] END criterion=gini, max_depth=9,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 5.6s
[CV 5/10] END criterion=gini, max_depth=9,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 5.6s
[CV 6/10] END criterion=gini, max_depth=9,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 6.1s
[CV 7/10] END criterion=gini, max_depth=9,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 5.9s
[CV 8/10] END criterion=gini, max_depth=9,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 5.6s
[CV 9/10] END criterion=gini, max_depth=9,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 5.7s
[CV 10/10] END criterion=gini, max_depth=9,
min_impurity_decrease=0.15000000000000002;, score=0.125 total time= 7.1s
[CV 1/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0;,
score=0.728 total time= 23.3s
[CV 2/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0;,
score=0.729 total time= 22.1s
[CV 3/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0;,
score=0.739 total time= 21.6s
[CV 4/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0;,

```

score=0.747 total time= 20.8s
 [CV 5/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0;;
 score=0.745 total time= 24.0s
 [CV 6/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0;;
 score=0.745 total time= 23.6s
 [CV 7/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0;;
 score=0.745 total time= 22.1s
 [CV 8/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0;;
 score=0.746 total time= 22.1s
 [CV 9/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0;;
 score=0.745 total time= 21.4s
 [CV 10/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.0;;
 score=0.747 total time= 21.1s
 [CV 1/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05;;
 score=0.573 total time= 12.7s
 [CV 2/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05;;
 score=0.598 total time= 10.4s
 [CV 3/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05;;
 score=0.600 total time= 13.3s
 [CV 4/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05;;
 score=0.570 total time= 10.0s
 [CV 5/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05;;
 score=0.596 total time= 9.9s
 [CV 6/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05;;
 score=0.602 total time= 9.9s
 [CV 7/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05;;
 score=0.605 total time= 11.6s
 [CV 8/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05;;
 score=0.603 total time= 9.2s
 [CV 9/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05;;
 score=0.605 total time= 9.8s
 [CV 10/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.05;;
 score=0.607 total time= 8.9s
 [CV 1/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1;;
 score=0.125 total time= 5.3s
 [CV 2/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1;;
 score=0.125 total time= 5.5s
 [CV 3/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1;;
 score=0.125 total time= 5.4s
 [CV 4/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1;;
 score=0.125 total time= 5.4s
 [CV 5/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1;;
 score=0.125 total time= 5.5s
 [CV 6/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1;;
 score=0.125 total time= 5.5s
 [CV 7/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1;;
 score=0.125 total time= 5.5s
 [CV 8/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1;;

score=0.125 total time= 5.7s
 [CV 9/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1;,
 score=0.125 total time= 5.5s
 [CV 10/10] END criterion=gini, max_depth=10, min_impurity_decrease=0.1;,
 score=0.125 total time= 5.7s
 [CV 1/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 5.7s
 [CV 2/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 5.5s
 [CV 3/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 5.6s
 [CV 4/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 5.6s
 [CV 5/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 5.9s
 [CV 6/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 6.1s
 [CV 7/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 5.7s
 [CV 8/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 5.9s
 [CV 9/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 6.3s
 [CV 10/10] END criterion=gini, max_depth=10,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 5.6s
 [CV 1/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0; ,
 score=0.735 total time= 22.5s
 [CV 2/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0; ,
 score=0.735 total time= 22.8s
 [CV 3/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0; ,
 score=0.750 total time= 22.8s
 [CV 4/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0; ,
 score=0.756 total time= 20.6s
 [CV 5/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0; ,
 score=0.753 total time= 21.4s
 [CV 6/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0; ,
 score=0.752 total time= 22.1s
 [CV 7/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0; ,
 score=0.755 total time= 21.4s
 [CV 8/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0; ,
 score=0.756 total time= 20.2s
 [CV 9/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0; ,
 score=0.756 total time= 21.3s
 [CV 10/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.0; ,
 score=0.756 total time= 23.9s
 [CV 1/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05; ,
 score=0.600 total time= 8.6s
 [CV 2/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05; ,

score=0.576 total time= 8.9s
 [CV 3/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05;,
 score=0.593 total time= 10.1s
 [CV 4/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05;,
 score=0.594 total time= 9.7s
 [CV 5/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05;,
 score=0.596 total time= 9.3s
 [CV 6/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05;,
 score=0.603 total time= 8.8s
 [CV 7/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05;,
 score=0.599 total time= 9.0s
 [CV 8/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05;,
 score=0.578 total time= 9.5s
 [CV 9/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05;,
 score=0.580 total time= 8.6s
 [CV 10/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.05;,
 score=0.604 total time= 9.8s
 [CV 1/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1;,
 score=0.125 total time= 6.9s
 [CV 2/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1;,
 score=0.125 total time= 5.4s
 [CV 3/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1;,
 score=0.125 total time= 5.7s
 [CV 4/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1;,
 score=0.125 total time= 5.7s
 [CV 5/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1;,
 score=0.125 total time= 5.8s
 [CV 6/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1;,
 score=0.125 total time= 6.0s
 [CV 7/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1;,
 score=0.125 total time= 7.6s
 [CV 8/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1;,
 score=0.125 total time= 6.8s
 [CV 9/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1;,
 score=0.125 total time= 6.1s
 [CV 10/10] END criterion=gini, max_depth=11, min_impurity_decrease=0.1;,
 score=0.125 total time= 6.0s
 [CV 1/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 5.8s
 [CV 2/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 6.1s
 [CV 3/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 7.7s
 [CV 4/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 5.8s
 [CV 5/10] END criterion=gini, max_depth=11,
 min_impurity_decrease=0.15000000000000002; , score=0.125 total time= 6.2s
 [CV 6/10] END criterion=gini, max_depth=11,

```

min_impurity_decrease=0.15000000000000002;; score=0.125 total time= 7.4s
[CV 7/10] END criterion=gini, max_depth=11,
min_impurity_decrease=0.15000000000000002;; score=0.125 total time= 6.6s
[CV 8/10] END criterion=gini, max_depth=11,
min_impurity_decrease=0.15000000000000002;; score=0.125 total time= 6.8s
[CV 9/10] END criterion=gini, max_depth=11,
min_impurity_decrease=0.15000000000000002;; score=0.125 total time= 6.1s
[CV 10/10] END criterion=gini, max_depth=11,
min_impurity_decrease=0.15000000000000002;; score=0.125 total time= 6.7s
[CV 1/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0;;
score=0.709 total time= 19.7s
[CV 2/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0;;
score=0.713 total time= 18.0s
[CV 3/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0;;
score=0.725 total time= 16.6s
[CV 4/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0;;
score=0.727 total time= 17.7s
[CV 5/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0;;
score=0.724 total time= 21.9s
[CV 6/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0;;
score=0.728 total time= 19.3s
[CV 7/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0;;
score=0.727 total time= 18.3s
[CV 8/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0;;
score=0.726 total time= 18.4s
[CV 9/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0;;
score=0.728 total time= 22.5s
[CV 10/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.0;;
score=0.726 total time= 22.7s
[CV 1/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05;;
score=0.605 total time= 13.5s
[CV 2/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05;;
score=0.618 total time= 12.7s
[CV 3/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05;;
score=0.607 total time= 12.6s
[CV 4/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05;;
score=0.612 total time= 13.3s
[CV 5/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05;;
score=0.616 total time= 13.6s
[CV 6/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05;;
score=0.601 total time= 12.2s
[CV 7/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05;;
score=0.612 total time= 12.4s
[CV 8/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05;;
score=0.619 total time= 12.1s
[CV 9/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05;;
score=0.621 total time= 12.6s
[CV 10/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.05;;

```


score=0.620 total time= 12.8s
 [CV 1/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1;;
 score=0.589 total time= 11.4s
 [CV 2/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1;;
 score=0.600 total time= 10.8s
 [CV 3/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1;;
 score=0.598 total time= 10.8s
 [CV 4/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1;;
 score=0.593 total time= 11.1s
 [CV 5/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1;;
 score=0.599 total time= 11.7s
 [CV 6/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1;;
 score=0.599 total time= 13.2s
 [CV 7/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1;;
 score=0.599 total time= 11.7s
 [CV 8/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1;;
 score=0.600 total time= 12.8s
 [CV 9/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1;;
 score=0.584 total time= 12.4s
 [CV 10/10] END criterion=entropy, max_depth=8, min_impurity_decrease=0.1;;
 score=0.601 total time= 10.5s
 [CV 1/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002;; score=0.586 total time= 9.3s
 [CV 2/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002;; score=0.599 total time= 9.8s
 [CV 3/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002;; score=0.591 total time= 10.4s
 [CV 4/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002;; score=0.570 total time= 12.4s
 [CV 5/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002;; score=0.599 total time= 10.3s
 [CV 6/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002;; score=0.597 total time= 10.3s
 [CV 7/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002;; score=0.595 total time= 10.1s
 [CV 8/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002;; score=0.598 total time= 9.2s
 [CV 9/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002;; score=0.577 total time= 10.4s
 [CV 10/10] END criterion=entropy, max_depth=8,
 min_impurity_decrease=0.15000000000000002;; score=0.593 total time= 10.2s
 [CV 1/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0;;
 score=0.721 total time= 20.3s
 [CV 2/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0;;
 score=0.720 total time= 22.3s
 [CV 3/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0;;
 score=0.734 total time= 22.3s
 [CV 4/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0;;

score=0.739 total time= 21.2s
 [CV 5/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0;;
 score=0.737 total time= 29.3s
 [CV 6/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0;;
 score=0.736 total time= 21.8s
 [CV 7/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0;;
 score=0.740 total time= 21.1s
 [CV 8/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0;;
 score=0.737 total time= 18.9s
 [CV 9/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0;;
 score=0.738 total time= 23.2s
 [CV 10/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.0;;
 score=0.741 total time= 31.4s
 [CV 1/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05;;
 score=0.616 total time= 16.7s
 [CV 2/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05;;
 score=0.619 total time= 17.4s
 [CV 3/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05;;
 score=0.605 total time= 16.0s
 [CV 4/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05;;
 score=0.599 total time= 12.5s
 [CV 5/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05;;
 score=0.610 total time= 13.5s
 [CV 6/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05;;
 score=0.616 total time= 13.7s
 [CV 7/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05;;
 score=0.612 total time= 11.8s
 [CV 8/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05;;
 score=0.621 total time= 12.2s
 [CV 9/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05;;
 score=0.611 total time= 12.4s
 [CV 10/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.05;;
 score=0.612 total time= 12.5s
 [CV 1/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1;;
 score=0.594 total time= 10.4s
 [CV 2/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1;;
 score=0.588 total time= 10.7s
 [CV 3/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1;;
 score=0.598 total time= 11.4s
 [CV 4/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1;;
 score=0.583 total time= 12.4s
 [CV 5/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1;;
 score=0.598 total time= 13.1s
 [CV 6/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1;;
 score=0.596 total time= 14.0s
 [CV 7/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1;;
 score=0.591 total time= 16.0s
 [CV 8/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1;;

score=0.593 total time= 15.7s
 [CV 9/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1;;
 score=0.600 total time= 14.3s
 [CV 10/10] END criterion=entropy, max_depth=9, min_impurity_decrease=0.1;;
 score=0.602 total time= 11.7s
 [CV 1/10] END criterion=entropy, max_depth=9,
 min_impurity_decrease=0.15000000000000002;; score=0.584 total time= 9.8s
 [CV 2/10] END criterion=entropy, max_depth=9,
 min_impurity_decrease=0.15000000000000002;; score=0.595 total time= 9.3s
 [CV 3/10] END criterion=entropy, max_depth=9,
 min_impurity_decrease=0.15000000000000002;; score=0.591 total time= 9.1s
 [CV 4/10] END criterion=entropy, max_depth=9,
 min_impurity_decrease=0.15000000000000002;; score=0.558 total time= 9.6s
 [CV 5/10] END criterion=entropy, max_depth=9,
 min_impurity_decrease=0.15000000000000002;; score=0.597 total time= 9.8s
 [CV 6/10] END criterion=entropy, max_depth=9,
 min_impurity_decrease=0.15000000000000002;; score=0.590 total time= 11.2s
 [CV 7/10] END criterion=entropy, max_depth=9,
 min_impurity_decrease=0.15000000000000002;; score=0.599 total time= 12.5s
 [CV 8/10] END criterion=entropy, max_depth=9,
 min_impurity_decrease=0.15000000000000002;; score=0.600 total time= 10.7s
 [CV 9/10] END criterion=entropy, max_depth=9,
 min_impurity_decrease=0.15000000000000002;; score=0.601 total time= 10.8s
 [CV 10/10] END criterion=entropy, max_depth=9,
 min_impurity_decrease=0.15000000000000002;; score=0.601 total time= 9.9s
 [CV 1/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0;;
 score=0.731 total time= 18.7s
 [CV 2/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0;;
 score=0.733 total time= 22.9s
 [CV 3/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0;;
 score=0.747 total time= 24.2s
 [CV 4/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0;;
 score=0.750 total time= 24.5s
 [CV 5/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0;;
 score=0.748 total time= 20.4s
 [CV 6/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0;;
 score=0.749 total time= 21.6s
 [CV 7/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0;;
 score=0.749 total time= 21.3s
 [CV 8/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0;;
 score=0.749 total time= 21.0s
 [CV 9/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0;;
 score=0.753 total time= 19.6s
 [CV 10/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.0;;
 score=0.751 total time= 19.5s
 [CV 1/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05;;
 score=0.614 total time= 12.0s
 [CV 2/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05;;

score=0.611 total time= 12.7s
 [CV 3/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05;;
 score=0.611 total time= 15.0s
 [CV 4/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05;;
 score=0.613 total time= 14.4s
 [CV 5/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05;;
 score=0.615 total time= 16.0s
 [CV 6/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05;;
 score=0.612 total time= 20.1s
 [CV 7/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05;;
 score=0.611 total time= 13.0s
 [CV 8/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05;;
 score=0.610 total time= 14.3s
 [CV 9/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05;;
 score=0.613 total time= 12.9s
 [CV 10/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.05;;
 score=0.615 total time= 13.3s
 [CV 1/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1;;
 score=0.595 total time= 10.9s
 [CV 2/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1;;
 score=0.596 total time= 11.0s
 [CV 3/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1;;
 score=0.597 total time= 10.5s
 [CV 4/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1;;
 score=0.595 total time= 11.4s
 [CV 5/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1;;
 score=0.599 total time= 10.5s
 [CV 6/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1;;
 score=0.596 total time= 10.8s
 [CV 7/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1;;
 score=0.598 total time= 11.2s
 [CV 8/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1;;
 score=0.601 total time= 11.0s
 [CV 9/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1;;
 score=0.601 total time= 12.0s
 [CV 10/10] END criterion=entropy, max_depth=10, min_impurity_decrease=0.1;;
 score=0.601 total time= 11.3s
 [CV 1/10] END criterion=entropy, max_depth=10,
 min_impurity_decrease=0.15000000000000002;; score=0.592 total time= 11.1s
 [CV 2/10] END criterion=entropy, max_depth=10,
 min_impurity_decrease=0.15000000000000002;; score=0.599 total time= 10.9s
 [CV 3/10] END criterion=entropy, max_depth=10,
 min_impurity_decrease=0.15000000000000002;; score=0.602 total time= 12.8s
 [CV 4/10] END criterion=entropy, max_depth=10,
 min_impurity_decrease=0.15000000000000002;; score=0.595 total time= 9.6s
 [CV 5/10] END criterion=entropy, max_depth=10,
 min_impurity_decrease=0.15000000000000002;; score=0.598 total time= 9.1s
 [CV 6/10] END criterion=entropy, max_depth=10,

```

min_impurity_decrease=0.15000000000000002;; score=0.591 total time= 10.1s
[CV 7/10] END criterion=entropy, max_depth=10,
min_impurity_decrease=0.15000000000000002;; score=0.597 total time= 9.8s
[CV 8/10] END criterion=entropy, max_depth=10,
min_impurity_decrease=0.15000000000000002;; score=0.599 total time= 10.1s
[CV 9/10] END criterion=entropy, max_depth=10,
min_impurity_decrease=0.15000000000000002;; score=0.600 total time= 8.9s
[CV 10/10] END criterion=entropy, max_depth=10,
min_impurity_decrease=0.15000000000000002;; score=0.578 total time= 9.0s
[CV 1/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0;;
score=0.739 total time= 22.7s
[CV 2/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0;;
score=0.743 total time= 21.4s
[CV 3/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0;;
score=0.755 total time= 22.0s
[CV 4/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0;;
score=0.760 total time= 21.8s
[CV 5/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0;;
score=0.758 total time= 22.2s
[CV 6/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0;;
score=0.758 total time= 21.9s
[CV 7/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0;;
score=0.760 total time= 20.6s
[CV 8/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0;;
score=0.761 total time= 19.6s
[CV 9/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0;;
score=0.762 total time= 20.0s
[CV 10/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.0;;
score=0.763 total time= 20.2s
[CV 1/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05;;
score=0.605 total time= 11.7s
[CV 2/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05;;
score=0.613 total time= 11.9s
[CV 3/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05;;
score=0.615 total time= 12.9s
[CV 4/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05;;
score=0.609 total time= 12.8s
[CV 5/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05;;
score=0.609 total time= 13.4s
[CV 6/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05;;
score=0.613 total time= 13.5s
[CV 7/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05;;
score=0.614 total time= 13.9s
[CV 8/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05;;
score=0.611 total time= 15.3s
[CV 9/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05;;
score=0.616 total time= 14.3s
[CV 10/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.05;;

```

```

score=0.613 total time= 12.3s
[CV 1/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1;,
score=0.587 total time= 11.1s
[CV 2/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1;,
score=0.598 total time= 11.7s
[CV 3/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1;,
score=0.587 total time= 10.7s
[CV 4/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1;,
score=0.594 total time= 11.5s
[CV 5/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1;,
score=0.584 total time= 9.7s
[CV 6/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1;,
score=0.597 total time= 10.7s
[CV 7/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1;,
score=0.598 total time= 11.6s
[CV 8/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1;,
score=0.603 total time= 10.5s
[CV 9/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1;,
score=0.600 total time= 10.8s
[CV 10/10] END criterion=entropy, max_depth=11, min_impurity_decrease=0.1;,
score=0.604 total time= 10.5s
[CV 1/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002;, score=0.595 total time= 8.5s
[CV 2/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002;, score=0.584 total time= 9.1s
[CV 3/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002;, score=0.598 total time= 9.0s
[CV 4/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002;, score=0.591 total time= 9.3s
[CV 5/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002;, score=0.574 total time= 9.2s
[CV 6/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002;, score=0.594 total time= 9.2s
[CV 7/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002;, score=0.600 total time= 9.2s
[CV 8/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002;, score=0.602 total time= 9.3s
[CV 9/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002;, score=0.577 total time= 9.6s
[CV 10/10] END criterion=entropy, max_depth=11,
min_impurity_decrease=0.15000000000000002;, score=0.601 total time= 9.4s
{'criterion': 'entropy', 'max_depth': 11, 'min_impurity_decrease': 0.0}

```

3.7 conclusion

```
[ ]: from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.model_selection import cross_validate
model=RandomForestClassifier(criterion='entropy',max_depth=11,min_impurity_decrease=0.
    ↪0)
cv_results = cross_validate(model, x_scaled_new,Y,
    ↪cv=10,verbose=1,return_estimator=True)
accuracies = cv_results['test_score']
accuracies=list(accuracies)
models=cv_results['estimator']
model = models[accuracies.index(max(accuracies))]
predictions = model.predict(x_test_scaled)

# print classification report
print(classification_report(test_target, predictions))
```

[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.

[Parallel(n_jobs=1)]: Done 10 out of 10 | elapsed: 3.1min finished

	precision	recall	f1-score	support
0	0.00	0.00	0.00	2
1	1.00	1.00	1.00	8436
2	0.53	0.71	0.60	3029
3	0.10	0.56	0.17	50
4	0.84	0.95	0.89	3943
5	0.47	0.40	0.43	2848
6	0.52	0.05	0.08	1711
7	0.52	0.83	0.64	175
accuracy			0.78	20194
macro avg	0.50	0.56	0.48	20194
weighted avg	0.78	0.78	0.76	20194

```
[ ]: # saving the model
import pickle
filename = '/content/gdrive/MyDrive/Ml_project/fars_models/random_forest_tree.
    ↪sav'
pickle.dump(model, open(filename, 'wb'))
```

3.8 Accuracy=0.76

#SVM (SVC)

3.9 Hyper parameter tuning (Bayesian optimization)

```
[ ]: !pip install scikit-optimize
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-
wheels/public/simple/
Requirement already satisfied: scikit-optimize in /usr/local/lib/python3.8/dist-
packages (0.9.0)
Requirement already satisfied: numpy>=1.13.3 in /usr/local/lib/python3.8/dist-
packages (from scikit-optimize) (1.21.6)
Requirement already satisfied: scipy>=0.19.1 in /usr/local/lib/python3.8/dist-
packages (from scikit-optimize) (1.7.3)
Requirement already satisfied: joblib>=0.11 in /usr/local/lib/python3.8/dist-
packages (from scikit-optimize) (1.2.0)
Requirement already satisfied: pyaml>=16.9 in /usr/local/lib/python3.8/dist-
packages (from scikit-optimize) (21.10.1)
Requirement already satisfied: scikit-learn>=0.20.0 in
/usr/local/lib/python3.8/dist-packages (from scikit-optimize) (1.0.2)
Requirement already satisfied: PyYAML in /usr/local/lib/python3.8/dist-packages
(from pyaml>=16.9->scikit-optimize) (6.0)
Requirement already satisfied: threadpoolctl>=2.0.0 in
/usr/local/lib/python3.8/dist-packages (from scikit-learn>=0.20.0->scikit-
optimize) (3.1.0)
```

```
[ ]: # Assumption using only 5000 data since tuning takes more than 10 hrs and
↳ colabs crashes
from skopt.space import Real,Categorical,Integer
from sklearn.metrics import classification_report, confusion_matrix
from skopt import BayesSearchCV
from sklearn.svm import SVC
parameters = {'C':Real(0.1,2), 'kernel':['poly'], 'gamma':Real(0.5,2), 'degree':
↳ (3,4,5)}
model=SVC()
clf_a = BayesSearchCV(model,
↳ search_spaces=parameters,verbose=3,scoring='accuracy',cv=2,random_state=0,n_iter=20,optimize
↳ 'GP'})
clf_a.fit(x_scaled_new[:5000],Y[:5000])
print(clf_a.best_params_)
```

```
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=1.10883191340908, degree=4, gamma=1.4340646289441554,
kernel=poly;; score=0.725 total time= 2.3s
[CV 2/2] END C=1.10883191340908, degree=4, gamma=1.4340646289441554,
kernel=poly;; score=0.726 total time= 1.7s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=0.5909061577482214, degree=5, gamma=1.8626575605049618,
kernel=poly;; score=0.716 total time= 18.8s
[CV 2/2] END C=0.5909061577482214, degree=5, gamma=1.8626575605049618,
```



```

kernel=poly;; score=0.713 total time= 16.4s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=1.1045143099433568, degree=3, gamma=1.0134251737470477,
kernel=poly;; score=0.750 total time= 0.7s
[CV 2/2] END C=1.1045143099433568, degree=3, gamma=1.0134251737470477,
kernel=poly;; score=0.759 total time= 0.7s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=0.38009764270722635, degree=5, gamma=1.2049160605443756,
kernel=poly;; score=0.723 total time= 2.3s
[CV 2/2] END C=0.38009764270722635, degree=5, gamma=1.2049160605443756,
kernel=poly;; score=0.719 total time= 2.3s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=0.5417766536375034, degree=3, gamma=1.592271783771687,
kernel=poly;; score=0.748 total time= 0.8s
[CV 2/2] END C=0.5417766536375034, degree=3, gamma=1.592271783771687,
kernel=poly;; score=0.749 total time= 0.8s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=1.6925161268510815, degree=5, gamma=1.6001100550593257,
kernel=poly;; score=0.715 total time= 16.5s
[CV 2/2] END C=1.6925161268510815, degree=5, gamma=1.6001100550593257,
kernel=poly;; score=0.713 total time= 19.9s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=1.9375493272801696, degree=5, gamma=1.140323729980283,
kernel=poly;; score=0.718 total time= 5.9s
[CV 2/2] END C=1.9375493272801696, degree=5, gamma=1.140323729980283,
kernel=poly;; score=0.716 total time= 4.8s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=1.0699291881951203, degree=4, gamma=1.582666036979052,
kernel=poly;; score=0.727 total time= 2.8s
[CV 2/2] END C=1.0699291881951203, degree=4, gamma=1.582666036979052,
kernel=poly;; score=0.730 total time= 3.1s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=0.27490354411991014, degree=5, gamma=1.6902230690996811,
kernel=poly;; score=0.718 total time= 6.1s
[CV 2/2] END C=0.27490354411991014, degree=5, gamma=1.6902230690996811,
kernel=poly;; score=0.716 total time= 5.0s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=0.8021202570609925, degree=4, gamma=0.8926081622191394,
kernel=poly;; score=0.741 total time= 0.8s
[CV 2/2] END C=0.8021202570609925, degree=4, gamma=0.8926081622191394,
kernel=poly;; score=0.735 total time= 0.7s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=0.1, degree=3, gamma=0.5, kernel=poly;; score=0.705 total time=
0.4s
[CV 2/2] END C=0.1, degree=3, gamma=0.5, kernel=poly;; score=0.712 total time=
0.4s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=0.1, degree=3, gamma=2.0, kernel=poly;; score=0.752 total time=

```

```

0.7s
[CV 2/2] END C=0.1, degree=3, gamma=2.0, kernel=poly;; score=0.759 total time=
0.6s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=2.0, degree=3, gamma=1.226387661599312, kernel=poly;; score=0.747
total time= 1.0s
[CV 2/2] END C=2.0, degree=3, gamma=1.226387661599312, kernel=poly;; score=0.746
total time= 0.9s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=2.0, degree=3, gamma=1.8547118109652554, kernel=poly;;
score=0.735 total time= 2.0s
[CV 2/2] END C=2.0, degree=3, gamma=1.8547118109652554, kernel=poly;;
score=0.733 total time= 1.6s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=1.7003680144079296, degree=3, gamma=0.5, kernel=poly;;
score=0.743 total time= 0.5s
[CV 2/2] END C=1.7003680144079296, degree=3, gamma=0.5, kernel=poly;;
score=0.756 total time= 0.5s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=0.38581590214464956, degree=3, gamma=2.0, kernel=poly;;
score=0.748 total time= 0.9s
[CV 2/2] END C=0.38581590214464956, degree=3, gamma=2.0, kernel=poly;;
score=0.748 total time= 0.8s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=0.1, degree=4, gamma=0.5, kernel=poly;; score=0.726 total time=
0.4s
[CV 2/2] END C=0.1, degree=4, gamma=0.5, kernel=poly;; score=0.735 total time=
0.4s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=1.3858411298446123, degree=3, gamma=1.2054325016396876,
kernel=poly;; score=0.749 total time= 0.8s
[CV 2/2] END C=1.3858411298446123, degree=3, gamma=1.2054325016396876,
kernel=poly;; score=0.750 total time= 0.8s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=2.0, degree=4, gamma=0.5, kernel=poly;; score=0.747 total time=
0.6s
[CV 2/2] END C=2.0, degree=4, gamma=0.5, kernel=poly;; score=0.754 total time=
0.5s
Fitting 2 folds for each of 1 candidates, totalling 2 fits
[CV 1/2] END C=2.0, degree=3, gamma=0.7514158242775547, kernel=poly;;
score=0.752 total time= 0.7s
[CV 2/2] END C=2.0, degree=3, gamma=0.7514158242775547, kernel=poly;;
score=0.760 total time= 0.7s
OrderedDict([('C', 2.0), ('degree', 3), ('gamma', 0.7514158242775547),
('kernel', 'poly')])

```

3.10 conclusion

```
[ ]: from sklearn.svm import SVC
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.model_selection import cross_validate
model=SVC(C=2,gamma=0.75,kernel='poly',degree=3)
cv_results = cross_validate(model, x_scaled_new[:50000],Y[:50000],
    ↪cv=5,verbose=1,return_estimator=True)
accuracies = cv_results['test_score']
accuracies=list(accuracies)
models=cv_results['estimator']
model = models[accuracies.index(max(accuracies))]
predictions = model.predict(x_test_scaled)

# print classification report
print(classification_report(test_target, predictions))
```

[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.
[Parallel(n_jobs=1)]: Done 5 out of 5 | elapsed: 17.2min finished

	precision	recall	f1-score	support
0	0.00	0.00	0.00	2
1	1.00	0.99	1.00	8436
2	0.52	0.58	0.55	3029
3	0.42	0.22	0.29	50
4	0.84	0.99	0.90	3943
5	0.44	0.54	0.49	2848
6	0.53	0.04	0.08	1711
7	0.70	0.67	0.69	175
accuracy			0.78	20194
macro avg	0.56	0.50	0.50	20194
weighted avg	0.77	0.78	0.76	20194

/usr/local/lib/python3.8/dist-packages/sklearn/metrics/_classification.py:1318:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.

_warn_prf(average, modifier, msg_start, len(result))

/usr/local/lib/python3.8/dist-packages/sklearn/metrics/_classification.py:1318:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.

_warn_prf(average, modifier, msg_start, len(result))

/usr/local/lib/python3.8/dist-packages/sklearn/metrics/_classification.py:1318:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to

control this behavior.

```
_warn_prf(average, modifier, msg_start, len(result))
```

3.11 Accuracy - 0.78

```
[ ]: # save the model to disk since retraining is expensive
import pickle
filename = '/content/gdrive/MyDrive/Ml_project/fars_models/SVC.sav'
pickle.dump(model, open(filename, 'wb'))
```

4 K- nearest neighbour

4.1 Hyper parameter tuning (GridSearch CV)

```
[ ]: from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import GridSearchCV
from sklearn.metrics import classification_report, confusion_matrix
parameters = {'n_neighbors':np.arange(3,5), 'metric':['manhattan']}
model=KNeighborsClassifier()
clf_a = GridSearchCV(model, parameters,verbose=3,scoring='accuracy',cv=5)
clf_a.fit(x_scaled_new,Y)
print(clf_a.best_params_)
```

Fitting 5 folds for each of 2 candidates, totalling 10 fits

```
[CV 1/5] END ..metric=manhattan, n_neighbors=3;, score=0.792 total time= 5.9min
[CV 2/5] END ..metric=manhattan, n_neighbors=3;, score=0.839 total time= 5.8min
[CV 3/5] END ..metric=manhattan, n_neighbors=3;, score=0.865 total time= 5.7min
[CV 4/5] END ..metric=manhattan, n_neighbors=3;, score=0.870 total time= 5.7min
[CV 5/5] END ..metric=manhattan, n_neighbors=3;, score=0.869 total time= 5.7min
[CV 1/5] END ..metric=manhattan, n_neighbors=4;, score=0.824 total time= 6.7min
[CV 2/5] END ..metric=manhattan, n_neighbors=4;, score=0.849 total time= 7.2min
[CV 3/5] END ..metric=manhattan, n_neighbors=4;, score=0.871 total time= 6.8min
[CV 4/5] END ..metric=manhattan, n_neighbors=4;, score=0.877 total time= 7.0min
[CV 5/5] END ..metric=manhattan, n_neighbors=4;, score=0.873 total time= 6.9min
{'metric': 'manhattan', 'n_neighbors': 4}
```

4.2 conclusion

```
[ ]: from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.model_selection import cross_validate
model=KNeighborsClassifier(n_neighbors=4,metric='manhattan')
cv_results = cross_validate(model, x_scaled_new,Y,
    ↪cv=5,verbose=1,return_estimator=True)
accuracies = cv_results['test_score']
accuracies=list(accuracies)
models=cv_results['estimator']
```

```

model = models[accuracies.index(max(accuracies))]
predictions = model.predict(x_test_scaled)

# print classification report
print(classification_report(test_target, model.predict(x_test_scaled)))

```

[Parallel(n_jobs=1)]: Using backend SequentialBackend with 1 concurrent workers.
[Parallel(n_jobs=1)]: Done 5 out of 5 | elapsed: 28.7min finished

	precision	recall	f1-score	support
0	0.00	0.00	0.00	2
1	0.98	0.95	0.97	8436
2	0.49	0.58	0.53	3029
3	0.17	0.48	0.25	50
4	0.84	0.87	0.85	3943
5	0.43	0.38	0.40	2848
6	0.27	0.23	0.25	1711
7	0.46	0.71	0.56	175
accuracy			0.73	20194
macro avg	0.46	0.52	0.48	20194
weighted avg	0.74	0.73	0.73	20194

```

[ ]: # save the model to disk since retraining is expensive
import pickle
filename = '/content/gdrive/MyDrive/ML_project/fars_models/knn.sav'
pickle.dump(model, open(filename, 'wb'))

```

5 Ensembled method stacking

```

[ ]: import pickle
from sklearn.ensemble import StackingClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.model_selection import cross_validate
knn= pickle.load(open('/content/gdrive/MyDrive/ML_project/fars_models/knn.sav', 'rb'))
svc= pickle.load(open('/content/gdrive/MyDrive/ML_project/fars_models/SVC.sav', 'rb'))
random_forest= pickle.load(open('/content/gdrive/MyDrive/ML_project/fars_models/
    random_forest_tree.sav', 'rb'))
decision_trr= pickle.load(open('/content/gdrive/MyDrive/ML_project/fars_models/
    decision_tree.sav', 'rb'))

```

```

models = ␣
↪[('knn',knn),('svc',svc),('random_forest',random_forest),('decision_trr',decision_trr)]

stacking_model = LogisticRegression(solver='sag',random_state=0,max_iter=200)
stacked_model = StackingClassifier( estimators = models,final_estimator = ␣
↪stacking_model)
results_cv = cross_validate(stacked_model, x_scaled_new[:20000],Y[:20000],␣
↪cv=5,verbose=1,return_estimator=True)

```

6 Cv plot for the ensemble model

```
[ ]: results_cv['test_score'][:3]
```

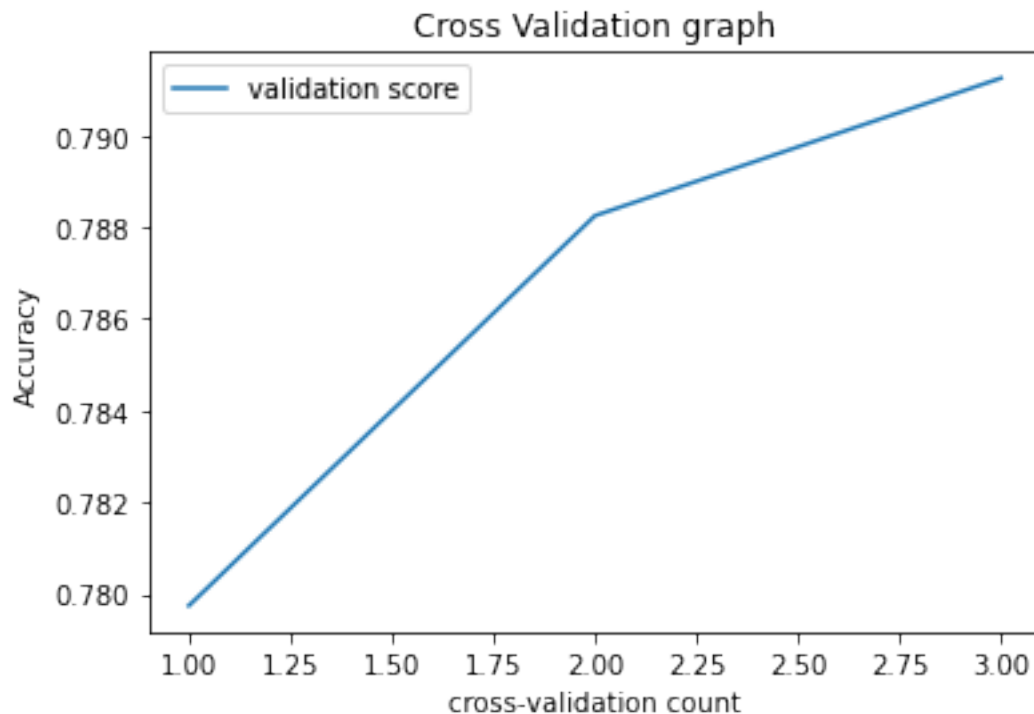
```
[ ]: array([0.77975, 0.78825, 0.79125])
```

```

[ ]: d={'t':[1,2,3],'validation score':results_cv['test_score'][:3]}
pd.DataFrame(d).plot.line('t','validation score')
plt.title("Cross Validation graph")
plt.xlabel("cross-validation count")
plt.ylabel("Accuracy")

```

```
[ ]: Text(0, 0.5, 'Accuracy')
```



```
[ ]: accuracies = results_cv['test_score']
      accuracies=list(accuracies)
      models=results_cv['estimator']
      model = models[accuracies.index(max(accuracies))]
      predictions = model.predict(x_test_scaled)

      # print classification report
      print(classification_report(test_target, predictions))
```

	precision	recall	f1-score	support
0	0.00	0.00	0.00	2
1	1.00	1.00	1.00	8436
2	0.59	0.61	0.60	3029
3	0.92	0.22	0.35	50
4	0.84	0.99	0.91	3943
5	0.46	0.59	0.51	2848
6	0.52	0.06	0.10	1711
7	0.71	0.67	0.69	175
accuracy			0.79	20194
macro avg	0.63	0.52	0.52	20194
weighted avg	0.79	0.79	0.77	20194

```
/usr/local/lib/python3.8/dist-packages/sklearn/metrics/_classification.py:1318:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
```

```
_warn_prf(average, modifier, msg_start, len(result))
```

```
/usr/local/lib/python3.8/dist-packages/sklearn/metrics/_classification.py:1318:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
```

```
_warn_prf(average, modifier, msg_start, len(result))
```

```
/usr/local/lib/python3.8/dist-packages/sklearn/metrics/_classification.py:1318:
UndefinedMetricWarning: Precision and F-score are ill-defined and being set to
0.0 in labels with no predicted samples. Use `zero_division` parameter to
control this behavior.
```

```
_warn_prf(average, modifier, msg_start, len(result))
```

```
#appendix
```

```
[ ]: !pip install lazypredict
      from lazypredict.Supervised import LazyClassifier
      clf=LazyClassifier(verbose=1,ignore_warnings=True,custom_metric=None)
      models,prediction=clf.fit(x_scaled_new[:10000],x_test_scaled[:10000],Y[:
      ↪10000],test_target[:10000])
```

```
print(models)
```