```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split


from google.colab import drive
drive.mount('/gdrive')
%cd /gdrive
```

```
Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=9473189

Enter your authorization code:
··········
Mounted at /gdrive
/gdrive
```

```
%cd /gdrive/My\ Drive/CSE512Data
!ls
```

```
/gdrive/.shortcut-targets-by-id/1_NaeHeaL1Atv72DZfAUz97ht_NGuXeI3/CSE512Data
'2020-05-16 00:00:22.1814430.csv'     '2020-05-16 00:00:58.29761911.csv'
'2020-05-16 00:00:24.9546041.csv'     '2020-05-16 00:01:00.10923012.csv'
'2020-05-16 00:00:31.6322502.csv'      all_accelerometer_data_pids_13.csv
'2020-05-16 00:00:36.5933463.csv'      clean_tac
'2020-05-16 00:00:42.6664554.csv'      good_now.csv
'2020-05-16 00:00:46.0082825.csv'      may_be_cleaned.csv
'2020-05-16 00:00:47.8559756.csv'      mega2.csv
'2020-05-16 00:00:47.8559756.gsheet'   mega.csv
'2020-05-16 00:00:48.7453017.csv'      phone_types.csv
'2020-05-16 00:00:50.5036288.csv'      pids.txt
'2020-05-16 00:00:51.7498629.csv'      raw_tac
'2020-05-16 00:00:52.77848910.csv'     README.txt
```

```
frame = pd.read_csv('mega2.csv')


def threshold(value):
  def resp(x):
    if x >= value:
      return 1
    else:
      return 0
  return resp

frame['TAC_reading'] = frame['TAC_reading'].apply(threshold(0.08))


frame = frame[[x for x in frame.columns if x != 'Unnamed: 0']]
frame = frame[[x for x in frame.columns if x != 'pid']]
frame = frame[[x for x in frame.columns if x != 'window10']]
frame = frame[[x for x in frame.columns if x != 'win_10_x_FFT_spectral_centroid_spread']]
frame = frame[[x for x in frame.columns if x != 'win_10_y_FFT_spectral_centroid_spread']]
```

```
frame = frame[[x for x in frame.columns if x != 'win_10_z_FFT_spectral_centroid_spread']]
frame = frame[[x for x in frame.columns if x != 'x_FFT_spectral_centroid_spread']]
frame = frame[[x for x in frame.columns if x != 'y_FFT_spectral_centroid_spread']]
frame = frame[[x for x in frame.columns if x != 'z_FFT_spectral_centroid_spread']]
frame = frame[[x for x in frame.columns if x != 'key']]


frame = frame.dropna()
frame
```

	x_mean	x_variance	x_median	x_min	x_max	x_rms	x_energy	
0	0.000000	0.000000e+00	0.000000	0.000000	0.000000	0.000000	0.000000e+00	0.
1	-1.093507	1.071096e+00	-0.511803	-2.556815	-0.017652	1.505607	2.266853e+00	-0.
2	-0.028878	9.801984e-04	-0.032244	-0.136399	0.065830	0.042593	1.814152e-03	-0.
3	-0.011776	1.371459e-04	-0.015459	-0.039830	0.022391	0.016608	2.758143e-04	0.
4	-0.017732	1.035722e-04	-0.019142	-0.040878	0.013389	0.020445	4.180111e-04	0.
...	...	...	...	...	...	...	...	
369795	-0.000212	2.845938e-07	-0.000300	-0.001400	0.000900	0.000574	3.297500e-07	-0.
369796	-0.000215	2.847750e-07	-0.000200	-0.001400	0.000800	0.000575	3.310000e-07	-0.
369797	-0.000212	3.240937e-07	-0.000200	-0.001500	0.000800	0.000608	3.692500e-07	-0.
369798	-0.000256	1.932281e-07	-0.000200	-0.001200	0.000600	0.000509	2.589744e-07	-0.
369799	-0.000200	4.313333e-07	-0.000150	-0.001500	0.001100	0.000687	4.713333e-07	0.

369800 rows × 57 columns

```
toxicated = 0
intoxicated = 0
for i in frame['TAC_reading']:
  if i == 1:
    intoxicated+=1
  else:
    toxicated+=1
print(toxicated)
print(intoxicated)
```

```
283207
86593
```

```
x_values, y_values = frame[[x for x in frame.columns if x != 'TAC_reading']].to_numpy(), fram

x_train, x_test, y_train, y_test = train_test_split(x_values, y_values, test_size = 0.25, ran
```

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
from sklearn import tree


random_forest_instance = RandomForestClassifier(max_depth=8, random_state=0,n_estimators=100)
random_forest_instance.fit(x_train, y_train)
random_forest_pred = random_forest_instance.predict(x_test)
accuracy_score(random_forest_pred, y_test)
```

```
0.8343861546782044
```