

driving_behavior_XGBoost_v1

August 31, 2022

```
[ ]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
[ ]: df_training = pd.read_csv("../data_mod/train_motion_data.csv")
df_test = pd.read_csv("../data_mod/test_motion_data.csv")

df_training
```

```
[ ]:
```

	AccX	AccY	Class	DiffAccX	DiffAccY	VelX	VelY
0	0.000000	0.000000	NORMAL	0.000000	0.000000	0.000000	0.000000
1	-1.624864	-1.082492	NORMAL	-1.624864	-1.082492	-0.812432	-0.541246
2	-0.594660	-0.122410	NORMAL	1.030204	0.960082	-0.297330	-0.061205
3	0.738478	-0.228456	NORMAL	1.333138	-0.106046	0.369239	-0.114228
4	0.101741	0.777568	NORMAL	-0.636737	1.006023	0.050871	0.388784
...
3639	0.915688	-2.017489	SLOW	2.374675	-1.824629	0.457844	-1.008745
3640	-1.934203	0.914925	SLOW	-2.849891	2.932414	-0.967102	0.457462
3641	-0.222845	0.747304	SLOW	1.711359	-0.167621	-0.111422	0.373652
3642	-0.349423	0.067261	SLOW	-0.126579	-0.680043	-0.174712	0.033630
3643	-0.402428	0.406218	SLOW	-0.053005	0.338957	-0.201214	0.203109

[3644 rows x 7 columns]

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

```
[ ]: AccX      0
AccY      0
Class      0
DiffAccX   0
DiffAccY   0
VelX       0
VelY       0
dtype: int64
```

0.1 Change categories to numbers

```
[ ]: df_training = df_training.replace(
      {"Class": {"NORMAL": 0, "AGGRESSIVE": 1, "SLOW": 2}})
df_test = df_test.replace(
      {"Class": {"NORMAL": 0, "AGGRESSIVE": 1, "SLOW": 2}})
df_training
```

```
[ ]:      AccX      AccY  Class  DiffAccX  DiffAccY      VelX      VelY
0      0.000000  0.000000      0  0.000000  0.000000  0.000000  0.000000
1     -1.624864 -1.082492      0 -1.624864 -1.082492 -0.812432 -0.541246
2     -0.594660 -0.122410      0  1.030204  0.960082 -0.297330 -0.061205
3      0.738478 -0.228456      0  1.333138 -0.106046  0.369239 -0.114228
4      0.101741  0.777568      0 -0.636737  1.006023  0.050871  0.388784
...
3639  0.915688 -2.017489      2  2.374675 -1.824629  0.457844 -1.008745
3640 -1.934203  0.914925      2 -2.849891  2.932414 -0.967102  0.457462
3641 -0.222845  0.747304      2  1.711359 -0.167621 -0.111422  0.373652
3642 -0.349423  0.067261      2 -0.126579 -0.680043 -0.174712  0.033630
3643 -0.402428  0.406218      2 -0.053005  0.338957 -0.201214  0.203109
```

[3644 rows x 7 columns]

0.2 Normalize the data

```
[ ]: X_training = df_training.drop(columns=["Class"])
X_training = (X_training - X_training.mean()) / X_training.std() * 100

X_training["Class"] = df_training["Class"]
X_training
```

```
[ ]:      AccX      AccY  DiffAccX  DiffAccY      VelX      VelY \
0     -4.105593   8.126800   0.010300  -0.010421  -4.105593   8.126800
1    -168.957027 -111.696347 -151.542377 -101.201825 -168.957027 -111.696347
2     -64.437130  -5.422989   96.098456   89.738101  -64.437130  -5.422989
3      70.817107 -17.161393  124.353421  -9.923577   70.817107 -17.161393
4       6.216602  94.197287  -59.378806   94.032688    6.216602  94.197287
...
3639  88.795978 -215.193071  221.498566 -170.576840  88.795978 -215.193071
3640 -200.341232  109.401604 -265.801873  274.111831 -200.341232  109.401604
3641 -26.714411  90.847295  159.630443  -15.679652 -26.714411  90.847295
3642 -39.556507  15.572024 -11.795809 -63.580862 -39.556507  15.572024
3643 -44.934120  53.091875  -4.933494  31.675331 -44.934120  53.091875

      Class
0         0
1         0
```

```

2      0
3      0
4      0
...
3639   2
3640   2
3641   2
3642   2
3643   2

```

[3644 rows x 7 columns]

```

[ ]: X_testing = df_test.drop(columns="Class")
X_testing = (X_testing - X_testing.mean()) / X_testing.std() * 100

X_testing["Class"] = df_test["Class"]
X_testing

```

```

[ ]:
      AccX      AccY      DiffAccX      DiffAccY      VelX      VelY \
0      67.345100    -9.509000    -0.021340    -0.012385   -10.948927    14.564454
1      57.982946     10.303100    -8.494392     16.758078    58.012497    10.295271
2      270.452050   -824.010358    192.270076   -706.238535   270.496822   -824.016540
3      229.805029   -828.171460   -36.808209    -3.534656   229.846889   -828.177633
4      283.133326   -732.402479    48.242495    81.053740   283.179007   -732.408842
...
3079   -84.712435   -57.627689   -73.609489    -1.097380   -84.693107   -57.635384
3080   145.444037    51.068429    208.277716    91.996249   145.479853    51.060520
3081   121.268079   -177.287100   -21.901364   -193.309813   121.302164   -177.294558
3082    83.265000    79.069807   -34.415357   216.987532    83.296362    79.061842
3083   140.063424    35.612446    51.383072   -36.797989   140.098855    35.604567

```

```

      Class
0         1
1         1
2         1
3         1
4         1
...
3079      2
3080      2
3081      2
3082      2
3083      2

```

[3084 rows x 7 columns]

```
[ ]: X_train = X_training.drop(columns="Class")
      y_train = X_training.Class

      X_test = X_testing.drop(columns="Class")
      y_test = X_testing.Class
```

```
[ ]: from sklearn.ensemble import GradientBoostingClassifier
      from sklearn.model_selection import RandomizedSearchCV
      from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay
```

```
[ ]: xgb = GradientBoostingClassifier(n_estimators=100, learning_rate=1.0,
      ↪max_depth=1, random_state=0)

      param_grid = {'n_estimators': np.arange(50, 200), 'learning_rate': np.linspace(
      ↪5, 1.0, 15), 'max_depth': np.arange(1, 10)}

      xgb_gscv = RandomizedSearchCV(xgb, param_grid, n_iter=20, cv=5, verbose=10,
      ↪n_jobs=10, random_state=0)
      xgb_gscv.fit(X_train, y_train)
```

```
Fitting 5 folds for each of 20 candidates, totalling 100 fits
[CV 3/5; 1/20] START learning_rate=0.5714285714285714, max_depth=1,
n_estimators=82
[CV 1/5; 1/20] START learning_rate=0.5714285714285714, max_depth=1,
n_estimators=82
[CV 5/5; 1/20] START learning_rate=0.5714285714285714, max_depth=1,
n_estimators=82
[CV 1/5; 2/20] START learning_rate=0.75, max_depth=9, n_estimators=199...
[CV 3/5; 2/20] START learning_rate=0.75, max_depth=9,
n_estimators=199...[CV 4/5; 1/20] START learning_rate=0.5714285714285714,
max_depth=1, n_estimators=82

[CV 2/5; 1/20] START learning_rate=0.5714285714285714, max_depth=1,
n_estimators=82
[CV 4/5; 2/20] START learning_rate=0.75, max_depth=9, n_estimators=199...
[CV 5/5; 2/20] START learning_rate=0.75, max_depth=9, n_estimators=199...
[CV 2/5; 2/20] START learning_rate=0.75, max_depth=9, n_estimators=199...
[CV 3/5; 1/20] END learning_rate=0.5714285714285714, max_depth=1,
n_estimators=82; score=0.406 total time= 0.8s
[CV 1/5; 3/20] START learning_rate=0.75, max_depth=3, n_estimators=145...
[CV 5/5; 1/20] END learning_rate=0.5714285714285714, max_depth=1,
n_estimators=82; score=0.433 total time= 0.8s
[CV 2/5; 1/20] END learning_rate=0.5714285714285714, max_depth=1,
n_estimators=82; score=0.396 total time= 0.8s
[CV 2/5; 3/20] START learning_rate=0.75, max_depth=3, n_estimators=145...
[CV 3/5; 3/20] START learning_rate=0.75, max_depth=3,
n_estimators=145...[CV 4/5; 1/20] END learning_rate=0.5714285714285714,
```

max_depth=1, n_estimators=82;; score=0.428 total time= 0.8s

[CV 1/5; 1/20] END learning_rate=0.5714285714285714, max_depth=1,
n_estimators=82;; score=0.435 total time= 0.9s
[CV 4/5; 3/20] START learning_rate=0.75, max_depth=3, n_estimators=145..
[CV 5/5; 3/20] START learning_rate=0.75, max_depth=3, n_estimators=145..
[CV 2/5; 3/20] END learning_rate=0.75, max_depth=3, n_estimators=145;;
score=0.374 total time= 2.6s
[CV 1/5; 4/20] START learning_rate=1.0, max_depth=5, n_estimators=198..
[CV 1/5; 3/20] END learning_rate=0.75, max_depth=3, n_estimators=145;;
score=0.402 total time= 2.7s
[CV 2/5; 4/20] START learning_rate=1.0, max_depth=5, n_estimators=198..
[CV 3/5; 3/20] END learning_rate=0.75, max_depth=3, n_estimators=145;;
score=0.412 total time= 2.8s
[CV 3/5; 4/20] START learning_rate=1.0, max_depth=5, n_estimators=198..
[CV 5/5; 3/20] END learning_rate=0.75, max_depth=3, n_estimators=145;;
score=0.402 total time= 2.9s
[CV 4/5; 3/20] END learning_rate=0.75, max_depth=3, n_estimators=145;;
score=0.403 total time= 3.0s
[CV 4/5; 4/20] START learning_rate=1.0, max_depth=5, n_estimators=198..
[CV 5/5; 4/20] START learning_rate=1.0, max_depth=5, n_estimators=198..
[CV 1/5; 4/20] END learning_rate=1.0, max_depth=5, n_estimators=198;;
score=0.380 total time= 5.5s
[CV 1/5; 5/20] START learning_rate=0.8214285714285714, max_depth=7,
n_estimators=123
[CV 2/5; 4/20] END learning_rate=1.0, max_depth=5, n_estimators=198;;
score=0.369 total time= 5.6s
[CV 2/5; 5/20] START learning_rate=0.8214285714285714, max_depth=7,
n_estimators=123
[CV 3/5; 4/20] END learning_rate=1.0, max_depth=5, n_estimators=198;;
score=0.398 total time= 5.6s
[CV 3/5; 5/20] START learning_rate=0.8214285714285714, max_depth=7,
n_estimators=123
[CV 5/5; 4/20] END learning_rate=1.0, max_depth=5, n_estimators=198;;
score=0.368 total time= 5.7s
[CV 4/5; 5/20] START learning_rate=0.8214285714285714, max_depth=7,
n_estimators=123
[CV 4/5; 4/20] END learning_rate=1.0, max_depth=5, n_estimators=198;;
score=0.405 total time= 5.8s
[CV 5/5; 5/20] START learning_rate=0.8214285714285714, max_depth=7,
n_estimators=123
[CV 1/5; 2/20] END learning_rate=0.75, max_depth=9, n_estimators=199;;
score=0.390 total time= 11.7s
[CV 1/5; 6/20] START learning_rate=0.7142857142857143, max_depth=8,
n_estimators=125
[CV 4/5; 2/20] END learning_rate=0.75, max_depth=9, n_estimators=199;;
score=0.398 total time= 11.5s
[CV 2/5; 6/20] START learning_rate=0.7142857142857143, max_depth=8,

```

n_estimators=125
[CV 5/5; 2/20] END learning_rate=0.75, max_depth=9, n_estimators=199;,
score=0.408 total time= 11.6s
[CV 3/5; 6/20] START learning_rate=0.7142857142857143, max_depth=8,
n_estimators=125
[CV 3/5; 2/20] END learning_rate=0.75, max_depth=9, n_estimators=199;,
score=0.416 total time= 12.0s
[CV 4/5; 6/20] START learning_rate=0.7142857142857143, max_depth=8,
n_estimators=125
[CV 2/5; 2/20] END learning_rate=0.75, max_depth=9, n_estimators=199;,
score=0.383 total time= 12.0s
[CV 5/5; 6/20] START learning_rate=0.7142857142857143, max_depth=8,
n_estimators=125
[CV 1/5; 5/20] END learning_rate=0.8214285714285714, max_depth=7,
n_estimators=123;, score=0.390 total time= 4.9s
[CV 1/5; 7/20] START learning_rate=0.8571428571428571, max_depth=5,
n_estimators=66
[CV 2/5; 5/20] END learning_rate=0.8214285714285714, max_depth=7,
n_estimators=123;, score=0.387 total time= 4.9s
[CV 2/5; 7/20] START learning_rate=0.8571428571428571, max_depth=5,
n_estimators=66
[CV 3/5; 5/20] END learning_rate=0.8214285714285714, max_depth=7,
n_estimators=123;, score=0.412 total time= 4.9s
[CV 3/5; 7/20] START learning_rate=0.8571428571428571, max_depth=5,
n_estimators=66
[CV 5/5; 5/20] END learning_rate=0.8214285714285714, max_depth=7,
n_estimators=123;, score=0.413 total time= 5.0s
[CV 4/5; 7/20] START learning_rate=0.8571428571428571, max_depth=5,
n_estimators=66
[CV 4/5; 5/20] END learning_rate=0.8214285714285714, max_depth=7,
n_estimators=123;, score=0.373 total time= 5.1s
[CV 5/5; 7/20] START learning_rate=0.8571428571428571, max_depth=5,
n_estimators=66
[CV 1/5; 7/20] END learning_rate=0.8571428571428571, max_depth=5,
n_estimators=66;, score=0.424 total time= 1.9s
[CV 1/5; 8/20] START learning_rate=0.8928571428571428, max_depth=1,
n_estimators=135
[CV 2/5; 7/20] END learning_rate=0.8571428571428571, max_depth=5,
n_estimators=66;, score=0.362 total time= 2.0s
[CV 2/5; 8/20] START learning_rate=0.8928571428571428, max_depth=1,
n_estimators=135
[CV 3/5; 7/20] END learning_rate=0.8571428571428571, max_depth=5,
n_estimators=66;, score=0.416 total time= 1.9s
[CV 3/5; 8/20] START learning_rate=0.8928571428571428, max_depth=1,
n_estimators=135
[CV 4/5; 7/20] END learning_rate=0.8571428571428571, max_depth=5,
n_estimators=66;, score=0.353 total time= 1.9s
[CV 4/5; 8/20] START learning_rate=0.8928571428571428, max_depth=1,

```

```

n_estimators=135
[CV 5/5; 7/20] END learning_rate=0.8571428571428571, max_depth=5,
n_estimators=66;, score=0.405 total time= 2.1s
[CV 5/5; 8/20] START learning_rate=0.8928571428571428, max_depth=1,
n_estimators=135
[CV 1/5; 8/20] END learning_rate=0.8928571428571428, max_depth=1,
n_estimators=135;, score=0.421 total time= 1.0s
[CV 1/5; 9/20] START learning_rate=0.8928571428571428, max_depth=4,
n_estimators=180
[CV 2/5; 8/20] END learning_rate=0.8928571428571428, max_depth=1,
n_estimators=135;, score=0.394 total time= 1.1s
[CV 2/5; 9/20] START learning_rate=0.8928571428571428, max_depth=4,
n_estimators=180
[CV 3/5; 8/20] END learning_rate=0.8928571428571428, max_depth=1,
n_estimators=135;, score=0.401 total time= 1.0s
[CV 3/5; 9/20] START learning_rate=0.8928571428571428, max_depth=4,
n_estimators=180
[CV 4/5; 8/20] END learning_rate=0.8928571428571428, max_depth=1,
n_estimators=135;, score=0.413 total time= 1.1s
[CV 4/5; 9/20] START learning_rate=0.8928571428571428, max_depth=4,
n_estimators=180
[CV 3/5; 6/20] END learning_rate=0.7142857142857143, max_depth=8,
n_estimators=125;, score=0.427 total time= 5.8s
[CV 5/5; 9/20] START learning_rate=0.8928571428571428, max_depth=4,
n_estimators=180
[CV 1/5; 6/20] END learning_rate=0.7142857142857143, max_depth=8,
n_estimators=125;, score=0.396 total time= 6.0s
[CV 1/5; 10/20] START learning_rate=0.8928571428571428, max_depth=7,
n_estimators=132
[CV 5/5; 8/20] END learning_rate=0.8928571428571428, max_depth=1,
n_estimators=135;, score=0.419 total time= 1.1s
[CV 2/5; 10/20] START learning_rate=0.8928571428571428, max_depth=7,
n_estimators=132
[CV 2/5; 6/20] END learning_rate=0.7142857142857143, max_depth=8,
n_estimators=125;, score=0.370 total time= 6.0s
[CV 3/5; 10/20] START learning_rate=0.8928571428571428, max_depth=7,
n_estimators=132
[CV 4/5; 6/20] END learning_rate=0.7142857142857143, max_depth=8,
n_estimators=125;, score=0.396 total time= 5.8s
[CV 4/5; 10/20] START learning_rate=0.8928571428571428, max_depth=7,
n_estimators=132
[CV 5/5; 6/20] END learning_rate=0.7142857142857143, max_depth=8,
n_estimators=125;, score=0.408 total time= 6.4s
[CV 5/5; 10/20] START learning_rate=0.8928571428571428, max_depth=7,
n_estimators=132
[CV 1/5; 9/20] END learning_rate=0.8928571428571428, max_depth=4,
n_estimators=180;, score=0.398 total time= 4.1s
[CV 1/5; 11/20] START learning_rate=0.6428571428571428, max_depth=9,

```

```

n_estimators=194
[CV 3/5; 9/20] END learning_rate=0.8928571428571428, max_depth=4,
n_estimators=180;, score=0.412 total time= 4.1s
[CV 2/5; 11/20] START learning_rate=0.6428571428571428, max_depth=9,
n_estimators=194
[CV 2/5; 9/20] END learning_rate=0.8928571428571428, max_depth=4,
n_estimators=180;, score=0.372 total time= 4.2s
[CV 3/5; 11/20] START learning_rate=0.6428571428571428, max_depth=9,
n_estimators=194
[CV 4/5; 9/20] END learning_rate=0.8928571428571428, max_depth=4,
n_estimators=180;, score=0.405 total time= 4.1s
[CV 4/5; 11/20] START learning_rate=0.6428571428571428, max_depth=9,
n_estimators=194
[CV 5/5; 9/20] END learning_rate=0.8928571428571428, max_depth=4,
n_estimators=180;, score=0.390 total time= 4.1s
[CV 5/5; 11/20] START learning_rate=0.6428571428571428, max_depth=9,
n_estimators=194
[CV 4/5; 10/20] END learning_rate=0.8928571428571428, max_depth=7,
n_estimators=132;, score=0.399 total time= 5.2s
[CV 1/5; 12/20] START learning_rate=1.0, max_depth=7, n_estimators=102...
[CV 3/5; 10/20] END learning_rate=0.8928571428571428, max_depth=7,
n_estimators=132;, score=0.405 total time= 5.3s
[CV 2/5; 12/20] START learning_rate=1.0, max_depth=7, n_estimators=102...
[CV 1/5; 10/20] END learning_rate=0.8928571428571428, max_depth=7,
n_estimators=132;, score=0.409 total time= 5.5s
[CV 3/5; 12/20] START learning_rate=1.0, max_depth=7, n_estimators=102...
[CV 2/5; 10/20] END learning_rate=0.8928571428571428, max_depth=7,
n_estimators=132;, score=0.370 total time= 5.4s
[CV 4/5; 12/20] START learning_rate=1.0, max_depth=7, n_estimators=102...
[CV 5/5; 10/20] END learning_rate=0.8928571428571428, max_depth=7,
n_estimators=132;, score=0.407 total time= 5.5s
[CV 5/5; 12/20] START learning_rate=1.0, max_depth=7, n_estimators=102...
[CV 1/5; 12/20] END learning_rate=1.0, max_depth=7, n_estimators=102;,
score=0.414 total time= 4.0s
[CV 1/5; 13/20] START learning_rate=0.8571428571428571, max_depth=8,
n_estimators=150
[CV 2/5; 12/20] END learning_rate=1.0, max_depth=7, n_estimators=102;,
score=0.377 total time= 4.1s
[CV 2/5; 13/20] START learning_rate=0.8571428571428571, max_depth=8,
n_estimators=150
[CV 4/5; 12/20] END learning_rate=1.0, max_depth=7, n_estimators=102;,
score=0.412 total time= 4.1s
[CV 3/5; 13/20] START learning_rate=0.8571428571428571, max_depth=8,
n_estimators=150
[CV 3/5; 12/20] END learning_rate=1.0, max_depth=7, n_estimators=102;,
score=0.381 total time= 4.3s
[CV 4/5; 13/20] START learning_rate=0.8571428571428571, max_depth=8,
n_estimators=150

```


[CV 5/5; 12/20] END learning_rate=1.0, max_depth=7, n_estimators=102;,
 score=0.409 total time= 4.2s
 [CV 5/5; 13/20] START learning_rate=0.8571428571428571, max_depth=8,
 n_estimators=150
 [CV 1/5; 11/20] END learning_rate=0.6428571428571428, max_depth=9,
 n_estimators=194; , score=0.395 total time= 10.3s
 [CV 1/5; 14/20] START learning_rate=0.9285714285714286, max_depth=6,
 n_estimators=189
 [CV 2/5; 11/20] END learning_rate=0.6428571428571428, max_depth=9,
 n_estimators=194; , score=0.369 total time= 10.3s
 [CV 2/5; 14/20] START learning_rate=0.9285714285714286, max_depth=6,
 n_estimators=189
 [CV 3/5; 11/20] END learning_rate=0.6428571428571428, max_depth=9,
 n_estimators=194; , score=0.422 total time= 10.5s
 [CV 3/5; 14/20] START learning_rate=0.9285714285714286, max_depth=6,
 n_estimators=189
 [CV 4/5; 11/20] END learning_rate=0.6428571428571428, max_depth=9,
 n_estimators=194; , score=0.406 total time= 10.3s
 [CV 4/5; 14/20] START learning_rate=0.9285714285714286, max_depth=6,
 n_estimators=189
 [CV 5/5; 11/20] END learning_rate=0.6428571428571428, max_depth=9,
 n_estimators=194; , score=0.393 total time= 10.2s
 [CV 5/5; 14/20] START learning_rate=0.9285714285714286, max_depth=6,
 n_estimators=189
 [CV 1/5; 13/20] END learning_rate=0.8571428571428571, max_depth=8,
 n_estimators=150; , score=0.399 total time= 6.9s
 [CV 1/5; 15/20] START learning_rate=1.0, max_depth=1, n_estimators=133...
 [CV 2/5; 13/20] END learning_rate=0.8571428571428571, max_depth=8,
 n_estimators=150; , score=0.407 total time= 6.9s
 [CV 2/5; 15/20] START learning_rate=1.0, max_depth=1, n_estimators=133...
 [CV 4/5; 13/20] END learning_rate=0.8571428571428571, max_depth=8,
 n_estimators=150; , score=0.383 total time= 7.0s
 [CV 3/5; 15/20] START learning_rate=1.0, max_depth=1, n_estimators=133...
 [CV 3/5; 13/20] END learning_rate=0.8571428571428571, max_depth=8,
 n_estimators=150; , score=0.429 total time= 7.4s
 [CV 4/5; 15/20] START learning_rate=1.0, max_depth=1, n_estimators=133...
 [CV 1/5; 15/20] END learning_rate=1.0, max_depth=1, n_estimators=133; ,
 score=0.428 total time= 1.0s
 [CV 5/5; 15/20] START learning_rate=1.0, max_depth=1, n_estimators=133...
 [CV 2/5; 15/20] END learning_rate=1.0, max_depth=1, n_estimators=133; ,
 score=0.392 total time= 1.0s
 [CV 1/5; 16/20] START learning_rate=0.75, max_depth=6, n_estimators=177...
 [CV 3/5; 15/20] END learning_rate=1.0, max_depth=1, n_estimators=133; ,
 score=0.407 total time= 1.1s
 [CV 2/5; 16/20] START learning_rate=0.75, max_depth=6, n_estimators=177...
 [CV 4/5; 15/20] END learning_rate=1.0, max_depth=1, n_estimators=133; ,
 score=0.410 total time= 1.1s
 [CV 3/5; 16/20] START learning_rate=0.75, max_depth=6, n_estimators=177...

[CV 5/5; 13/20] END learning_rate=0.8571428571428571, max_depth=8,
 n_estimators=150;, score=0.401 total time= 7.5s
 [CV 4/5; 16/20] START learning_rate=0.75, max_depth=6, n_estimators=177...
 [CV 5/5; 15/20] END learning_rate=1.0, max_depth=1, n_estimators=133;,
 score=0.427 total time= 1.0s
 [CV 5/5; 16/20] START learning_rate=0.75, max_depth=6, n_estimators=177...
 [CV 1/5; 14/20] END learning_rate=0.9285714285714286, max_depth=6,
 n_estimators=189;, score=0.416 total time= 6.3s
 [CV 1/5; 17/20] START learning_rate=0.9642857142857142, max_depth=8,
 n_estimators=56
 [CV 2/5; 14/20] END learning_rate=0.9285714285714286, max_depth=6,
 n_estimators=189;, score=0.395 total time= 6.4s
 [CV 2/5; 17/20] START learning_rate=0.9642857142857142, max_depth=8,
 n_estimators=56
 [CV 5/5; 14/20] END learning_rate=0.9285714285714286, max_depth=6,
 n_estimators=189;, score=0.398 total time= 6.3s
 [CV 3/5; 17/20] START learning_rate=0.9642857142857142, max_depth=8,
 n_estimators=56
 [CV 4/5; 14/20] END learning_rate=0.9285714285714286, max_depth=6,
 n_estimators=189;, score=0.410 total time= 6.4s
 [CV 4/5; 17/20] START learning_rate=0.9642857142857142, max_depth=8,
 n_estimators=56
 [CV 3/5; 14/20] END learning_rate=0.9285714285714286, max_depth=6,
 n_estimators=189;, score=0.405 total time= 6.7s
 [CV 5/5; 17/20] START learning_rate=0.9642857142857142, max_depth=8,
 n_estimators=56
 [CV 1/5; 17/20] END learning_rate=0.9642857142857142, max_depth=8,
 n_estimators=56;, score=0.409 total time= 2.6s
 [CV 1/5; 18/20] START learning_rate=0.5714285714285714, max_depth=2,
 n_estimators=97
 [CV 2/5; 17/20] END learning_rate=0.9642857142857142, max_depth=8,
 n_estimators=56;, score=0.383 total time= 2.6s
 [CV 2/5; 18/20] START learning_rate=0.5714285714285714, max_depth=2,
 n_estimators=97
 [CV 3/5; 17/20] END learning_rate=0.9642857142857142, max_depth=8,
 n_estimators=56;, score=0.417 total time= 2.6s
 [CV 3/5; 18/20] START learning_rate=0.5714285714285714, max_depth=2,
 n_estimators=97
 [CV 4/5; 17/20] END learning_rate=0.9642857142857142, max_depth=8,
 n_estimators=56;, score=0.380 total time= 2.7s
 [CV 4/5; 18/20] START learning_rate=0.5714285714285714, max_depth=2,
 n_estimators=97
 [CV 1/5; 16/20] END learning_rate=0.75, max_depth=6, n_estimators=177;,
 score=0.412 total time= 6.0s
 [CV 5/5; 18/20] START learning_rate=0.5714285714285714, max_depth=2,
 n_estimators=97
 [CV 1/5; 18/20] END learning_rate=0.5714285714285714, max_depth=2,
 n_estimators=97;, score=0.420 total time= 1.2s

[CV 1/5; 19/20] START learning_rate=0.9285714285714286, max_depth=5,
 n_estimators=171
 [CV 5/5; 17/20] END learning_rate=0.9642857142857142, max_depth=8,
 n_estimators=56;; score=0.405 total time= 2.8s
 [CV 2/5; 19/20] START learning_rate=0.9285714285714286, max_depth=5,
 n_estimators=171
 [CV 2/5; 16/20] END learning_rate=0.75, max_depth=6, n_estimators=177;;
 score=0.377 total time= 6.0s
 [CV 3/5; 19/20] START learning_rate=0.9285714285714286, max_depth=5,
 n_estimators=171
 [CV 2/5; 18/20] END learning_rate=0.5714285714285714, max_depth=2,
 n_estimators=97;; score=0.374 total time= 1.3s
 [CV 4/5; 19/20] START learning_rate=0.9285714285714286, max_depth=5,
 n_estimators=171
 [CV 5/5; 16/20] END learning_rate=0.75, max_depth=6, n_estimators=177;;
 score=0.400 total time= 6.0s
 [CV 5/5; 19/20] START learning_rate=0.9285714285714286, max_depth=5,
 n_estimators=171
 [CV 3/5; 16/20] END learning_rate=0.75, max_depth=6, n_estimators=177;;
 score=0.420 total time= 6.3s
 [CV 1/5; 20/20] START learning_rate=0.6428571428571428, max_depth=6,
 n_estimators=116
 [CV 3/5; 18/20] END learning_rate=0.5714285714285714, max_depth=2,
 n_estimators=97;; score=0.401 total time= 1.2s
 [CV 2/5; 20/20] START learning_rate=0.6428571428571428, max_depth=6,
 n_estimators=116
 [CV 4/5; 16/20] END learning_rate=0.75, max_depth=6, n_estimators=177;;
 score=0.394 total time= 6.3s
 [CV 3/5; 20/20] START learning_rate=0.6428571428571428, max_depth=6,
 n_estimators=116
 [CV 4/5; 18/20] END learning_rate=0.5714285714285714, max_depth=2,
 n_estimators=97;; score=0.407 total time= 1.3s
 [CV 4/5; 20/20] START learning_rate=0.6428571428571428, max_depth=6,
 n_estimators=116
 [CV 5/5; 18/20] END learning_rate=0.5714285714285714, max_depth=2,
 n_estimators=97;; score=0.393 total time= 1.2s
 [CV 5/5; 20/20] START learning_rate=0.6428571428571428, max_depth=6,
 n_estimators=116
 [CV 2/5; 20/20] END learning_rate=0.6428571428571428, max_depth=6,
 n_estimators=116;; score=0.353 total time= 3.9s
 [CV 1/5; 20/20] END learning_rate=0.6428571428571428, max_depth=6,
 n_estimators=116;; score=0.413 total time= 3.9s
 [CV 1/5; 19/20] END learning_rate=0.9285714285714286, max_depth=5,
 n_estimators=171;; score=0.401 total time= 4.8s
 [CV 5/5; 20/20] END learning_rate=0.6428571428571428, max_depth=6,
 n_estimators=116;; score=0.401 total time= 3.9s
 [CV 4/5; 20/20] END learning_rate=0.6428571428571428, max_depth=6,
 n_estimators=116;; score=0.387 total time= 4.0s

```
[CV 3/5; 20/20] END learning_rate=0.6428571428571428, max_depth=6,
n_estimators=116;, score=0.401 total time= 4.2s
[CV 2/5; 19/20] END learning_rate=0.9285714285714286, max_depth=5,
n_estimators=171;, score=0.383 total time= 5.1s
[CV 3/5; 19/20] END learning_rate=0.9285714285714286, max_depth=5,
n_estimators=171;, score=0.392 total time= 5.0s
[CV 4/5; 19/20] END learning_rate=0.9285714285714286, max_depth=5,
n_estimators=171;, score=0.395 total time= 4.8s
[CV 5/5; 19/20] END learning_rate=0.9285714285714286, max_depth=5,
n_estimators=171;, score=0.389 total time= 4.7s
```

```
[ ]: RandomizedSearchCV(cv=5,
                        estimator=GradientBoostingClassifier(learning_rate=1.0,
                                                                max_depth=1,
                                                                random_state=0),
                        n_iter=20, n_jobs=10,
                        param_distributions={'learning_rate': array([0.5
0.53571429, 0.57142857, 0.60714286, 0.64285714,
0.67857143, 0.71428571, 0.75, 0.78571429, 0.82142857,
0.85714286, 0.89285714, 0.92857143, 0.96428571, 1. ]),
                        'max_depth': array([1, 2, 3, 4, 5, 6,
7,...
115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127,
128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140,
141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153,
154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166,
167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179,
180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192,
193, 194, 195, 196, 197, 198, 199])},
                        random_state=0, verbose=10)
```

```
[ ]: best_params = xgb_gscv.best_params_
best_params
```

```
[ ]: {'n_estimators': 82, 'max_depth': 1, 'learning_rate': 0.5714285714285714}
```

```
[ ]: xgb_gscv.best_score_
```

```
[ ]: 0.4195974464492983
```

0.2.1 Check for overfitting

```
[ ]: xgb_gscv.score(X_train, y_train)
```

```
[ ]: 0.47667398463227223
```

```
[ ]: xgb_gscv.score(X_test, y_test)
```

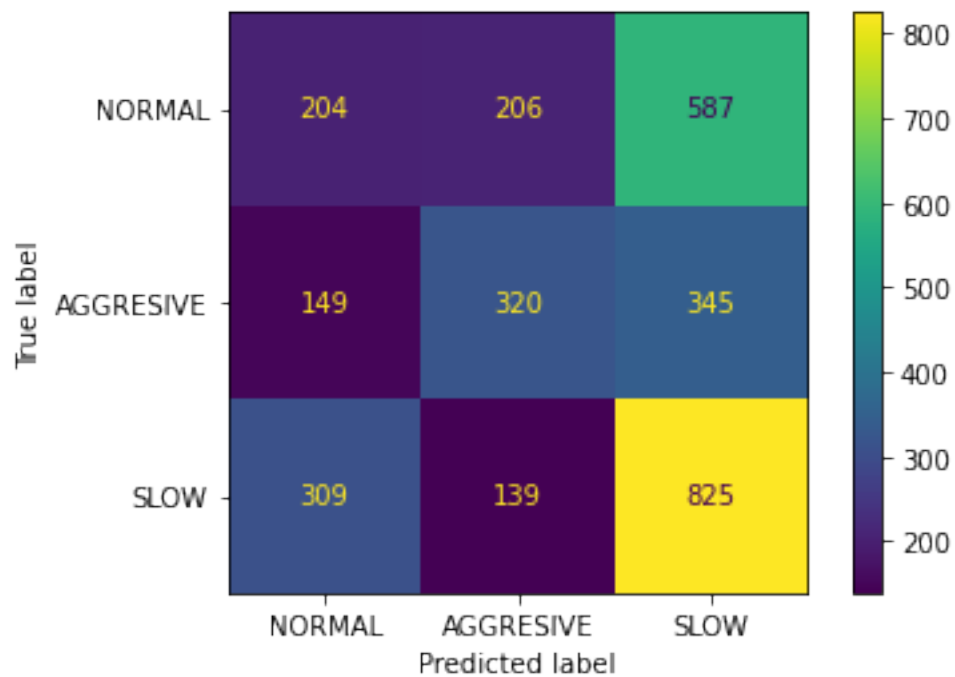
```
[ ]: 0.4374189364461738
```

```
[ ]: classes = ["NORMAL", "AGGRESIVE", "SLOW"]
```

```
[ ]: y_pred = xgb_gscv.predict(X_test)

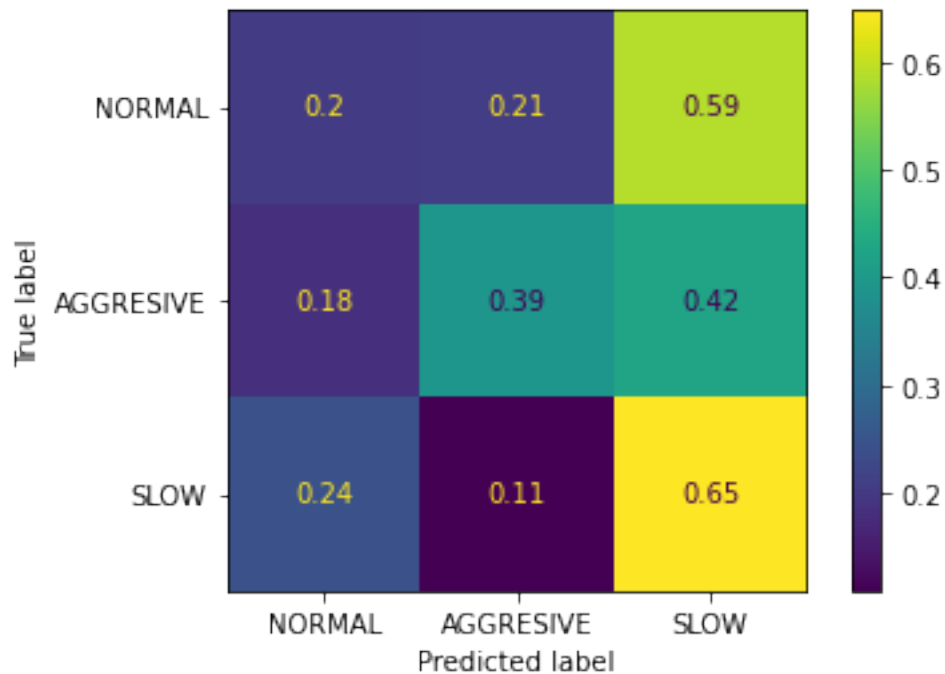
CM = confusion_matrix(y_test, y_pred)
display = ConfusionMatrixDisplay(confusion_matrix=CM,
                                  display_labels=classes)
display.plot()
```

```
[ ]: <sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at
0x7f720ebab070>
```



```
[ ]: CM_norm = confusion_matrix(y_test, y_pred, normalize="true")
display = ConfusionMatrixDisplay(confusion_matrix=CM_norm,
                                  display_labels=classes)
display.plot()
```

```
[ ]: <sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at
0x7f725d125100>
```



```
[ ]: def evaluate(model, test_features, test_labels):
    accuracy = model.score(test_features, test_labels)
    print('Model Performance')
    print('Accuracy = {:.3f}%'.format(accuracy))

    return accuracy

base_model = GradientBoostingClassifier(n_estimators=100, learning_rate=1.0,
    ↪max_depth=1, random_state=0)
base_model.fit(X_train, y_train)
base_accuracy = evaluate(base_model, X_test, y_test)

best_random = xgb_gscv.best_estimator_
random_accuracy = evaluate(best_random, X_test, y_test)

print(f'Improvement of {100 * (random_accuracy - base_accuracy) / base_accuracy:
    ↪.3f}%')
```

```
Model Performance
Accuracy = 0.432%.
Model Performance
Accuracy = 0.437%.
Improvement of 1.200%.
```

```
[ ]:
```