

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import os
#-----LOADING DATA-----
os.getcwd()
os.chdir('C:\\Users\\Serving Minds\\Desktop\\Exa-mobility')
df=pd.read_csv('steady position with single stand.csv')
```

In [2]:

```
df['State'] = pd.Series(['steady']*1818)
```

In [3]:

```
df1=pd.read_csv('steady pos with double stand.csv')
```

In [4]:

```
df1['State'] = pd.Series(['steady']*1785)
```

In [5]:

```
pieces = (df,df1)
df_final = pd.concat(pieces, ignore_index = True)
```

In [6]:

```
for col in df_final.columns:
    print(col)
```

```
raw_ax
raw_ay
raw_az
cal_ax
cal_ay
cal_az
raw_gx
raw_gy
raw_gz
cal_gx
cal_gy
cal_gz
raw_mx
raw_my
raw_mz
cal_mx
cal_my
cal_mz
filtered_ax
filtered_ay
filtered_az
filtered_gx
filtered_gy
filtered_gz
filtered_mx
filtered_my
filtered_mz
time_sec
yaw
pitch
roll
linearAccX
linearAccY
linearAccZ
time_d
```

timea
State

In [7]:

```
my_data=df_final.iloc[:,18:21]
```

In [8]:

```
my_data.min(axis = 0)
```

Out[8]:

```
filtered_ax    -0.329236
filtered_ay    -0.041991
filtered_az     0.907742
dtype: float64
```

In [9]:

```
my_data.max(axis = 0)
```

Out[9]:

```
filtered_ax     0.174783
filtered_ay     0.260015
filtered_az     0.999851
dtype: float64
```

In [10]:

```
def algo(x,y,z):
    if (x>=-0.33 and x<=0.175) and (y>=-0.04 and y<=0.27) and (z>=0.91 and z<=1):
        return 'steady'
    else:
        return 'moving'
```

In [11]:

```
df2=pd.read_csv('fast tilt.csv')
df2['State'] = pd.Series(['moving']*856)
```

In [12]:

```
df3=pd.read_csv('slow tilt.csv')
df3['State'] = pd.Series(['moving']*816)
```

In [13]:

```
df5=pd.read_csv('movement with bump.csv')
df5['State'] = pd.Series(['moving']*553)
```

In [14]:

```
pieces1 = (df_final,df2,df3,df5)
df_final_1 = pd.concat(pieces1, ignore_index = True)
```

In [15]:

```
df_final_1 = df_final_1.sample(frac=1).reset_index(drop=True)
```

In [16]:

```
test_data=df_final_1.iloc[:,18:21]
test_data.head(5)
```

Out[16]:

	filtered_ax	filtered_ay	filtered_az
0	0.029970	0.058396	0.997844
1	-0.261127	0.181294	0.948127
2	-0.260894	0.180746	0.948296
3	-0.260720	0.183048	0.947902
4	0.029087	0.057878	0.997900

In [17]:

```
my_list = []
for i in range(len(test_data)) :
    my_list.append(str(algo(test_data.loc[i, "filtered_ax"], test_data.loc[i, "filtered_ay"], test_data.loc[i, "filtered_az"])))
```

In [18]:

```
print(my_list)
```

```
['steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady',
'steady', 'moving', 'steady', 'steady', 'moving', 'moving', 'steady', 'moving', 'steady',
'steady', 'moving', 'moving', 'moving', 'steady', 'steady', 'steady', 'steady', 'steady',
'steady', 'steady', 'steady', 'steady', 'moving', 'moving', 'steady', 'steady', 'steady',
'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady',
'moving', 'steady', 'steady', 'moving', 'steady', 'moving', 'steady', 'steady', 'moving',
'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'moving', 'steady', 'steady',
'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'moving', 'steady', 'moving',
'moving', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'moving', 'steady',
'steady', 'steady', 'steady', 'moving', 'steady', 'moving', 'steady', 'steady', 'steady',
'moving', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'moving', 'steady',
'steady', 'steady', 'steady', 'steady', 'moving', 'steady', 'steady', 'steady', 'moving',
'steady', 'moving', 'moving', 'steady', 'steady', 'steady', 'steady', 'steady', 'moving',
'steady', 'moving', 'steady', 'steady', 'steady', 'steady', 'moving', 'steady', 'steady',
'steady', 'steady', 'steady', 'moving', 'steady', 'moving', 'steady', 'moving', 'steady',
'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'moving', 'steady', 'steady',
'moving', 'steady', 'steady', 'steady', 'steady', 'steady', 'moving', 'steady', 'moving',
'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'moving',
'steady', 'steady', 'moving', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady',
'moving', 'steady', 'moving', 'moving', 'steady', 'steady', 'steady', 'moving', 'steady',
'steady', 'steady', 'steady', 'moving', 'steady', 'moving', 'moving', 'steady', 'steady',
'steady', 'steady', 'steady', 'moving', 'steady', 'steady', 'steady', 'moving', 'steady',
'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'moving',
'moving', 'steady', 'moving', 'moving', 'steady', 'steady', 'steady', 'moving', 'steady',
'steady', 'steady', 'steady', 'moving', 'steady', 'moving', 'moving', 'steady', 'steady',
'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'moving',
'moving', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady',
'steady', 'steady', 'steady', 'steady', 'moving', 'steady', 'steady', 'steady', 'steady',
'moving', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady', 'steady',
```



```

1      steady      steady
2      steady      steady
3      steady      steady
4      steady      steady
...      ...      ...
5823    steady      steady
5824    steady      steady
5825    moving      moving
5826    steady      steady
5827    steady      steady

```

[5828 rows x 2 columns]

In [21]:

```

from sklearn.metrics import confusion_matrix
cf_matrix=confusion_matrix(actual_data, my_list)

```

In [22]:

```
print(cf_matrix)
```

```

[[ 906 1319]
 [   2 3601]]

```

In [23]:

```

from sklearn.metrics import accuracy_score
print(accuracy_score(actual_data,my_list))

```

0.7733356211393274

In [24]:

```

import seaborn as sns
ax = plt.axes()
sns.heatmap(cf_matrix, annot=True)
ax.set_title('CONFUSION MATRIX')

```

Out[24]:

Text(0.5, 1.0, 'CONFUSION MATRIX')

