

# Classification with DecisionTree in WDE

```
In [1]: #WDE dataset
WDE_path="C:/Users/aliba/OneDrive/Desktop/UNIVERSITA/TESI/DATASET/WalkingDistanceEstimation-master/dataset/"
classi=['armhand', 'pocket', 'calling', 'swing', 'handheld']
n_elem=500

import numpy as np
import matplotlib.pyplot as plt

#model
from sklearn.model_selection import KFold
from sklearn.model_selection import train_test_split, GridSearchCV
#regression
from sklearn import linear_model
#classification
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import classification_report

#visualization
from yellowbrick.classifier import ConfusionMatrix
#from yellowbrick.classifier import ClassificationReport

from ipynb.fs.full.functioncollection import importWDE, filtWDE, f_ext_WDE,makeeqWDE,classification_dataset,is_
```

## Import all WDE

```
In [2]: DATASET=importWDE()
```

```

PDR_Raw_2019-03-20-09-10-12 {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 288}
Outliers eliminati          {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 12}

PDR_Raw_2019-03-20-09-21-02 {'armhand': 0, 'pocket': 0, 'calling': 284, 'swing': 0, 'handheld': 0}
Outliers eliminati          {'armhand': 0, 'pocket': 0, 'calling': 15, 'swing': 0, 'handheld': 0}

PDR_Raw_2019-03-20-09-29-55 {'armhand': 0, 'pocket': 0, 'calling': 34, 'swing': 0, 'handheld': 45}
Outliers eliminati          {'armhand': 0, 'pocket': 0, 'calling': 3, 'swing': 0, 'handheld': 1}

PDR_Raw_2019-03-21-08-32-39 {'armhand': 196, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 0}
Outliers eliminati          {'armhand': 26, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 0}

PDR_Raw_2019-03-21-09-07-51 {'armhand': 527, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 0}
Outliers eliminati          {'armhand': 203, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 0}

PDR_Raw_2019-03-21-11-57-56 {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 197, 'handheld': 0}
Outliers eliminati          {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 151, 'handheld': 0}

PDR_Raw_2019-03-24-11-12-21 {'armhand': 0, 'pocket': 142, 'calling': 0, 'swing': 139, 'handheld': 0}
Outliers eliminati          {'armhand': 0, 'pocket': 8, 'calling': 0, 'swing': 10, 'handheld': 0}

PDR_Raw_2019-03-28-11-50-11 {'armhand': 0, 'pocket': 0, 'calling': 275, 'swing': 429, 'handheld': 425}
Outliers eliminati          {'armhand': 0, 'pocket': 0, 'calling': 16, 'swing': 42, 'handheld': 121}

PDR_Raw_2019-03-29-07-37-22 {'armhand': 0, 'pocket': 0, 'calling': 171, 'swing': 0, 'handheld': 0}
Outliers eliminati          {'armhand': 0, 'pocket': 0, 'calling': 64, 'swing': 0, 'handheld': 0}

PDR_Raw_2019-03-29-08-30-54 {'armhand': 0, 'pocket': 157, 'calling': 0, 'swing': 0, 'handheld': 0}
Outliers eliminati          {'armhand': 0, 'pocket': 116, 'calling': 0, 'swing': 0, 'handheld': 0}

PDR_Raw_2019-03-30-11-29-16 {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 897}
Outliers eliminati          {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 167}

PDR_Raw_2019-03-31-01-23-59 {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 1726}
Outliers eliminati          {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 202}

PDR_Raw_2019-03-31-10-04-54 {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 385, 'handheld': 0}
Outliers eliminati          {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 156, 'handheld': 0}

PDR_Raw_2019-03-31-10-33-25 {'armhand': 0, 'pocket': 386, 'calling': 0, 'swing': 3, 'handheld': 0}
Outliers eliminati          {'armhand': 0, 'pocket': 45, 'calling': 36, 'swing': 2, 'handheld': 0}

PDR_Raw_2019-03-31-12-03-05 {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 232, 'handheld': 0}
Outliers eliminati          {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 21, 'handheld': 0}

PDR_Raw_2019-03-31-12-29-51 {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 568}
Outliers eliminati          {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 205}

PDR_Raw_2019-04-01-10-45-07 {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 1214}
Outliers eliminati          {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 29}

PDR_Raw_2019-04-02-08-44-50 {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 664}
Outliers eliminati          {'armhand': 0, 'pocket': 0, 'calling': 0, 'swing': 0, 'handheld': 60}

```

In totale=> armhand:723, pocket:685, calling:764, swing:1385, handheld:5827, -->9384 stride

scrivi la strudura di "DATASET"

sotto applichiamo il filtraggio

```
In [3]: filtWDE(DATASET);
```

```
Filtering:#####
Done!
```

```
In [4]: print(type(DATASET))
for c,v in DATASET.items():
    print(c,type(v),len(v),type(v[0]),v[0].keys())
```

```

<class 'dict'>
armhand <class 'list'> 723 <class 'dict'> dict_keys(['target', 'Acc_X', 'Acc_Y', 'Acc_Z', 'Gyr_X', 'Gyr_Y', 'Gyr_Z', 'SensorTimestamp'])
pocket <class 'list'> 685 <class 'dict'> dict_keys(['target', 'Acc_X', 'Acc_Y', 'Acc_Z', 'Gyr_X', 'Gyr_Y', 'Gyr_Z', 'SensorTimestamp'])
calling <class 'list'> 764 <class 'dict'> dict_keys(['target', 'Acc_X', 'Acc_Y', 'Acc_Z', 'Gyr_X', 'Gyr_Y', 'Gyr_Z', 'SensorTimestamp'])
swing <class 'list'> 1385 <class 'dict'> dict_keys(['target', 'Acc_X', 'Acc_Y', 'Acc_Z', 'Gyr_X', 'Gyr_Y', 'Gyr_Z', 'SensorTimestamp'])
handheld <class 'list'> 5827 <class 'dict'> dict_keys(['target', 'Acc_X', 'Acc_Y', 'Acc_Z', 'Gyr_X', 'Gyr_Y', 'Gyr_Z', 'SensorTimestamp'])

```

feature ext

```
Extracting armhand:#####
Extracting pocket:#####
Extracting calling:#####
Extracting swing:#####
Extracting handheld:#####
```

```
armhand <class 'dict'> dict_keys(['feature', 'target']) 723 92 723
pocket <class 'dict'> dict_keys(['feature', 'target']) 685 92 685
calling <class 'dict'> dict_keys(['feature', 'target']) 764 92 764
swing <class 'dict'> dict_keys(['feature', 'target']) 1385 92 1385
handheld <class 'dict'> dict_keys(['feature', 'target']) 5827 92 5827
```

```
armhand <class 'dict'> dict_keys(['feature', 'target']) 500 92 500
pocket <class 'dict'> dict_keys(['feature', 'target']) 500 92 500
calling <class 'dict'> dict_keys(['feature', 'target']) 500 92 500
swing <class 'dict'> dict_keys(['feature', 'target']) 500 92 500
handheld <class 'dict'> dict_keys(['feature', 'target']) 500 92 500
```

```
armhand <class 'dict'> dict_keys(['feature', 'target']) 50 92 50
pocket <class 'dict'> dict_keys(['feature', 'target']) 50 92 50
calling <class 'dict'> dict_keys(['feature', 'target']) 50 92 50
swing <class 'dict'> dict_keys(['feature', 'target']) 50 92 50
handheld <class 'dict'> dict_keys(['feature', 'target']) 50 92 50
```

```
In [11]: print(np.array(class_x_train).shape)
          print(np.array(class_y_train).shape)

(2500, 92)
(2500,)
```

```
In [12]: print(np.array(class_x_test).shape)
          print(np.array(class_y_test).shape)

          (250, 92)
          (250,)
```

Nella cella sottostante viene testato il DecisionTree per `{'min_samples_leaf':[1..20], 'min_samples_split': [1..40]}`. La computazione di 8000 fit richiede in tutto 10 minuti circa, quindi `clf.fit` è commentato per non doverlo eseguire ogni volta, nel testo sottostante il risultato dei migliori parametri e l'accuracy ottenuta con gli stessi, creiamo quindi **best\_dt** con i suddetti parametri.

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

















[illegible]





[illegible]























[illegible]

[illegible]

[illegible]





[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]



[illegible]

[illegible]



[illegible]



[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]



[illegible]

[illegible]



[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

```
[CV] END .....min_samples_leaf=20, min_samples_split=38; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=38; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=38; total time= 0.1s
[CV] END .....min_samples_leaf=20, min_samples_split=38; total time= 0.1s
[CV] END .....min_samples_leaf=20, min_samples_split=38; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=39; total time= 0.1s
[CV] END .....min_samples_leaf=20, min_samples_split=39; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=39; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=39; total time= 0.1s
[CV] END .....min_samples_leaf=20, min_samples_split=39; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=39; total time= 0.1s
[CV] END .....min_samples_leaf=20, min_samples_split=39; total time= 0.1s
[CV] END .....min_samples_leaf=20, min_samples_split=39; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=39; total time= 0.1s
[CV] END .....min_samples_leaf=20, min_samples_split=39; total time= 0.1s
[CV] END .....min_samples_leaf=20, min_samples_split=39; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=40; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=40; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=40; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=40; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=40; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=40; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=40; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=40; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=40; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=40; total time= 0.0s
[CV] END .....min_samples_leaf=20, min_samples_split=40; total time= 0.0s
```

C:\Users\aliba\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.10\_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\sklearn\model\_selection\\_validation.py:378: FitFailedWarning:  
200 fits failed out of a total of 8000.  
The score on these train-test partitions for these parameters will be set to nan.  
If these failures are not expected, you can try to debug them by setting error\_score='raise'.

Below are more details about the failures:

-----  
200 fits failed with the following error:

Traceback (most recent call last):

File "C:\Users\aliba\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.10\_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\sklearn\model\_selection\\_validation.py", line 686, in \_fit\_and\_score  
estimator.fit(X\_train, y\_train, \*\*fit\_params)

File "C:\Users\aliba\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.10\_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\sklearn\tree\\_classes.py", line 969, in fit  
super().fit()

File "C:\Users\aliba\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.10\_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\sklearn\tree\\_classes.py", line 265, in fit  
check\_scalar()

File "C:\Users\aliba\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.10\_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\sklearn\utils\\_validation.py", line 1480, in check\_scalar  
raise ValueError()

ValueError: min\_samples\_split == 1, must be >= 2.

warnings.warn(some\_fits\_failed\_message, FitFailedWarning)

C:\Users\aliba\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.10\_qbz5n2kfra8p0\LocalCache\local-packages\Python310\site-packages\sklearn\model\_selection\\_search.py:953: UserWarning: One or more of the test scores are non-finite: [ nan 0.9944 0.996 0.9944 0.996 0.9944 0.996 0.9948 0.9956 0.9924 0.9952

```
0.9936 0.9944 0.9956 0.9952 0.9948 0.9956 0.9948 0.994 0.9952 0.9952
0.994 0.9944 0.9948 0.9932 0.994 0.9944 0.9936 0.9944 0.996 0.994
0.9944 0.9952 0.9948 0.9948 0.9952 0.9952 0.9964 0.9944 0.9944 0.9956
nan 0.9944 0.9928 0.9928 0.9944 0.996 0.994 0.994 0.9956 0.9948
0.9944 0.994 0.9936 0.9944 0.9956 0.9948 0.9944 0.9952 0.9948 0.994
0.9952 0.9952 0.9952 0.994 0.996 0.9948 0.9952 0.9932 0.9948 0.9932
0.9948 0.994 0.9956 0.994 0.9936 0.994 0.994 0.994 0.9952 0.9936
nan 0.9944 0.9944 0.994 0.994 0.9952 0.9956 0.9948 0.994 0.9956
0.9948 0.996 0.9952 0.9944 0.9952 0.9956 0.9952 0.9952 0.9952 0.9944
0.9952 0.9944 0.9956 0.9956 0.996 0.9948 0.9952 0.9956 0.9956 0.996
0.9952 0.9956 0.9956 0.9952 0.996 0.9948 0.9948 0.9952 0.9948 0.9948
nan 0.9956 0.9964 0.996 0.9952 0.9956 0.9964 0.996 0.9956 0.996
0.996 0.9964 0.9956 0.996 0.9964 0.996 0.9964 0.9964 0.9964 0.996
0.996 0.996 0.996 0.9956 0.9956 0.9964 0.996 0.996 0.996 0.996
0.996 0.9956 0.996 0.9964 0.996 0.9964 0.9956 0.996 0.996 0.9956
nan 0.9956 0.996 0.996 0.996 0.996 0.9964 0.9956 0.9956 0.996 0.9964
0.9952 0.996 0.996 0.9956 0.9956 0.996 0.9956 0.996 0.9964 0.9952
0.9964 0.9964 0.9956 0.996 0.9964 0.9956 0.996 0.996 0.9956 0.996
0.996 0.996 0.9956 0.996 0.996 0.996 0.996 0.996 0.996 0.9956
nan 0.996 0.9964 0.9956 0.9964 0.996 0.996 0.9956 0.9952 0.9956
0.9952 0.9956 0.9964 0.9956 0.9956 0.996 0.9956 0.996 0.9964 0.9956
0.996 0.996 0.996 0.9964 0.996 0.9956 0.9964 0.996 0.9956 0.996
0.9964 0.9956 0.9956 0.996 0.9956 0.9956 0.9956 0.996 0.9956 0.996
nan 0.996 0.9964 0.996 0.9964 0.9956 0.9964 0.9964 0.9956 0.996
0.9964 0.9956 0.9964 0.996 0.996 0.9956 0.996 0.9956 0.9964 0.996
0.9964 0.9964 0.996 0.9964 0.996 0.996 0.996 0.996 0.996 0.9964
0.9964 0.9956 0.9964 0.9964 0.9956 0.996 0.996 0.996 0.996 0.996
nan 0.996 0.9952 0.9956 0.9956 0.996 0.9956 0.996 0.9956 0.9956
0.9952 0.9948 0.9956 0.9956 0.9956 0.996 0.9952 0.9952 0.9952 0.9956
0.9952 0.9956 0.996 0.996 0.9956 0.9952 0.9956 0.9952 0.9952 0.9952
```

```
0.9956 0.996 0.996 0.9956 0.9952 0.9956 0.9956 0.996 0.9952 0.9956
nan 0.9944 0.9952 0.9956 0.996 0.9952 0.9952 0.9944 0.996 0.9956
0.9952 0.9956 0.996 0.9956 0.9952 0.9948 0.9948 0.9952 0.9948 0.9952
0.996 0.9952 0.9956 0.9956 0.9948 0.9952 0.9952 0.9952 0.9956 0.9952
0.9948 0.996 0.9948 0.9952 0.996 0.9948 0.9952 0.996 0.9948 0.9956
nan 0.996 0.996 0.9944 0.9948 0.9948 0.9956 0.9948 0.996 0.9944
0.9952 0.996 0.996 0.996 0.996 0.9944 0.994 0.9952 0.996 0.9952
0.996 0.9956 0.9956 0.9956 0.9956 0.9948 0.9944 0.9948 0.996 0.9948 0.996
0.9956 0.9956 0.9944 0.996 0.9952 0.996 0.9948 0.9948 0.9956 0.9944
nan 0.9952 0.9948 0.9948 0.9952 0.9952 0.9952 0.9952 0.9952 0.9948
0.9948 0.9948 0.9952 0.9948 0.9952 0.9952 0.9952 0.9948 0.9948 0.9948
0.9948 0.9952 0.9948 0.9948 0.9948 0.9952 0.9952 0.9952 0.9952 0.9952
0.9948 0.9948 0.9952 0.9948 0.9952 0.9948 0.9948 0.9952 0.9952 0.9948
nan 0.9952 0.9948 0.9948 0.9952 0.9952 0.9952 0.9948 0.9948 0.9952
0.9952 0.9952 0.9948 0.9952 0.9948 0.9948 0.9952 0.9948 0.9948 0.9952
0.9952 0.9952 0.9952 0.9948 0.9948 0.9952 0.9952 0.9948 0.9948 0.9952
0.9948 0.9948 0.9952 0.9948 0.9948 0.9952 0.9948 0.9952 0.9948 0.9948
nan 0.9948 0.9948 0.9952 0.9948 0.9952 0.9952 0.9948 0.9948 0.9952 0.9948
0.9948 0.9948 0.9948 0.9952 0.9952 0.9948 0.9948 0.9952 0.9948 0.9952
0.9948 0.9952 0.9952 0.9952 0.9948 0.9948 0.9952 0.9948 0.9948
0.9952 0.9952 0.9948 0.9948 0.9952 0.9948 0.9948 0.9948 0.9948 0.9952
nan 0.9956 0.9956 0.9956 0.9956 0.9952 0.9956 0.9952 0.9952 0.9956
0.9952 0.9956 0.9956 0.9952 0.9956 0.9952 0.9952 0.9952 0.9952 0.9952
0.9952 0.9952 0.9952 0.9952 0.9952 0.9952 0.9956 0.9956 0.9956 0.9956
0.9956 0.9956 0.9956 0.9952 0.9952 0.9956 0.9956 0.9952 0.9952 0.9952
nan 0.9956 0.9952 0.9956 0.9956 0.9952 0.9952 0.9952 0.9956 0.9952 0.9952
0.9952 0.9952 0.9952 0.9952 0.9952 0.9952 0.9952 0.9952 0.9952 0.9952
0.9956 0.9952 0.9952 0.9952 0.9956 0.9956 0.9952 0.9952 0.9952 0.9956
0.9952 0.9952 0.9952 0.9956 0.9952 0.9952 0.9956 0.9952 0.9952 0.9956
nan 0.9956 0.9952 0.9952 0.9956 0.9956 0.9952 0.9952 0.9952 0.9952 0.9956
0.9956 0.9956 0.9952 0.9952 0.9952 0.9952 0.9956 0.9956 0.9956 0.9956
0.9952 0.9956 0.9952 0.9952 0.9952 0.9952 0.9956 0.9956 0.9956 0.9952
0.9956 0.9952 0.9956 0.9956 0.9952 0.9952 0.9952 0.9956 0.9952 0.9956
nan 0.9952 0.9952 0.9952 0.9952 0.9952 0.9952 0.9956 0.9956 0.9952 0.9956
0.9956 0.9956 0.9952 0.9952 0.9952 0.9956 0.9956 0.9952 0.9956 0.9952
0.9952 0.9956 0.9952 0.9952 0.9952 0.9952 0.9956 0.9956 0.9956 0.9952
0.9956 0.9952 0.9952 0.9952 0.9952 0.9952 0.9956 0.9956 0.9956 0.9952
nan 0.9956 0.9952 0.9952 0.9952 0.9952 0.9956 0.9952 0.9956 0.9956 0.9952
0.9956 0.9956 0.9956 0.9956 0.9956 0.9956 0.9952 0.9952 0.9956 0.9952
0.9952 0.9952 0.9956 0.9952 0.9956 0.9952 0.9956 0.9956 0.9956 0.9952
0.9956 0.9952 0.9956 0.9956 0.9952 0.9952 0.9952 0.9952 0.9952 0.9952
nan 0.9956 0.9956 0.9956 0.9956 0.9956 0.9956 0.9956 0.9956 0.9956 0.9956
0.9956 0.9952 0.9952 0.9956 0.9952 0.9956 0.9956 0.9956 0.9956 0.9956
0.9952 0.9952 0.9956 0.9952 0.9956 0.9952 0.9956 0.9952 0.9952 0.9952
0.9952 0.9952 0.9956 0.9952 0.9956 0.9952 0.9956 0.9952 0.9952 0.9952
0.9952 0.9952 0.9956 0.9956 0.9956 0.9952 0.9952 0.9952 0.9952 0.9956]
warnings.warn(
```

```
C:\Users\aliba\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.10_qbz5n2kfra8p0\LocalCache\local-packa
ges\Python310\site-packages\sklearn\model_selection\_search.py:953: UserWarning: One or more of the train score
s are non-finite: [ nan 1. 0.99995556 0.99995556 0.99982222 0.99982222
```

```
0.99977778 0.99973333 0.99973333 0.99973333 0.99973333 0.99973333
0.99973333 0.99973333 0.99973333 0.99973333 0.99973333 0.99973333
0.99973333 0.99973333 0.99973333 0.99973333 0.99973333 0.99973333
0.99973333 0.99973333 0.99973333 0.99973333 0.99973333 0.99973333
0.99973333 0.99973333 0.99973333 0.99973333 0.99973333 0.99973333
0.99973333 0.99973333 0.99973333 0.99973333 nan 0.99866667
0.99866667 0.99866667 0.99857778 0.99853333 0.99808889 0.99773333
0.99773333 0.99773333 0.99773333 0.99773333 0.99773333 0.99773333
0.99773333 0.99773333 0.99773333 0.99773333 0.99773333 0.99773333
0.99773333 0.99773333 0.99773333 0.99773333 0.99773333 0.99773333
0.99773333 0.99773333 0.99773333 0.99773333 0.99773333 0.99773333
0.99773333 0.99773333 0.99773333 0.99773333 0.99773333 0.99773333
0.99773333 0.99773333 nan 0.99782222 0.99782222 0.99782222
0.99782222 0.99782222 0.9976 0.99742222 0.99742222 0.99742222
0.99742222 0.99742222 0.99742222 0.99742222 0.99742222 0.99742222
0.99742222 0.99742222 0.99742222 0.99742222 0.99742222 0.99742222
0.99742222 0.99742222 0.99742222 0.99742222 0.99742222 0.99742222
0.99742222 0.99742222 0.99742222 0.99742222 0.99742222 0.99742222
nan 0.99715556 0.99715556 0.99715556 0.99715556 0.99715556
0.99715556 0.99715556 0.99715556 0.99715556 0.99715556 0.99715556
0.99715556 0.99715556 0.99715556 0.99715556 0.99715556 0.99715556
0.99715556 0.99715556 0.99715556 0.99715556 0.99715556 0.99715556
0.99715556 0.99715556 0.99715556 0.99715556 0.99715556 0.99715556
0.99715556 0.99715556 0.99715556 0.99715556 nan 0.99706667
0.99706667 0.99706667 0.99706667 0.99706667 0.99706667 0.99706667
0.99706667 0.99706667 0.99706667 0.99706667 0.99706667 0.99706667
0.99706667 0.99706667 0.99706667 0.99706667 0.99706667 0.99706667
0.99706667 0.99706667 0.99706667 0.99706667 0.99706667 0.99706667
```





	precision	recall	f1-score	support
armhand	0.98	1.00	0.99	50
calling	1.00	0.98	0.99	50
handheld	1.00	0.92	0.96	50
pocket	0.98	0.98	0.98	50
swing	0.93	1.00	0.96	50
accuracy			0.98	250
macro avg	0.98	0.98	0.98	250
weighted avg	0.98	0.98	0.98	250
None				