

Persoonidentificatie aan de hand van accelerometer-gegevens

Omschrijving van de data:

De dataset bestaat uit accelerometergegevens van 22 unieke wandelende personen, verzameld door middel van een telefoon. Van elk persoon zijn meerdere fragmenten, met wisselende tijdsduren, opgenomen van hun wandeling. Van sommige personen zijn meer fragmenten opgenomen dan van anderen. Het doel is om personen te leren herkennen aan de hand van de accelerometergegevens in hun wandelfragmenten.

Inhoud:

`train_X_y.p` (Trainset)

1.857 wandelfragmenten van verschillende tijdsduur, bestaande uit x, y, z acceleratiewaarden per tijdseenheid (de input) met bijbehorende persoon IDs (0 t/m 21) (de target).

`test_X.p` (Testset)

626 wandelfragmenten van verschillende tijdsduur, bestaande uit x, y, z acceleratiewaarden per tijdseenheid (de input).

Inladen van de files

```
# Het inladen van de trainingsset (X_train en y_train):
```

```
In [1]: import pickle
```

```
In [2]: X_train, y_train = pickle.load(open("train_X_y.p", "rb"))
```

```
# Even kijken hoeveel data we hebben:
```

```
In [3]: len(X_train), len(y_train)
```

```
Out[3]: (1857, 1857)
```

```
# Yep, dat komt overeen met wat we hierboven hebben gezegd over 't aantal
```

```
# fragmenten die we hebben... Nu nog even kijken hoeveel klassen we hebben:
```

```
In [4]: import numpy as np
```

```
In [5]: np.unique(y_train)
```

```
Out[5]:
```

```
array([ 0.,  1.,  2.,  3.,  4.,  5.,  6.,  7.,  8.,  9., 10.,
        11., 12., 13., 14., 15., 16., 17., 18., 19., 20., 21.])
```

```

# Ok, 21 personen dus!

# Test-set inladen (daar hebben we alleen de fragmenten van dus!):
In [6]: X_test = pickle.load(open("test_X.p", "rb"))

In [7]: len(X_test)
Out[7]: 626

# Overigens is X_train een *lijst*, niet een numpy-array of matrix!
In [8]: type(X_train)
Out[8]: list

# Maar is elk fragment wel een numpy-object, en daarvan kunnen we dus de
# shape opvragen:
In [9]: X_train[0].shape
Out[9]: (47, 3)

# En zien dat het tweede fragment langer is, maar evenveel metingen per
# tijds-eenheid heeft gedaan (de x, y, z waarden)
In [10]: X_train[1].shape
Out[10]: (73, 3)

```

Opdracht:

Lever voor elk wandelfragment in de testset een voorspelling van de bijbehorende persoon ID aan. Het opgeleverde formaat is dus een list of array die 626 numerieke waarden tussen 0 en 21 bevat.

Aanleveren uitkomst van de opdracht:

Als je de uitkomst van je classifier in `y_pred` opslaat (waarbij `y_pred` dus een numpy-array of een python-lijst is), dan kan je de json-module gebruiken om het naar een tekst-file te schrijven:

```

In [11]: import json

In [12]: json.dump(list(y_pred), open("y_pred.json", "w+"))

```

Dit levert een tekst-file `y_pred.json` op. Stuur deze, **en je uitwerking** (bijvoorbeeld je ipython-notebook) naar ons op!