Projet Final Python for Data Analysis : MEU-Mobile KSD Data Set

Sabri LACHIHAB - IBO 3

Scrapping de la page web

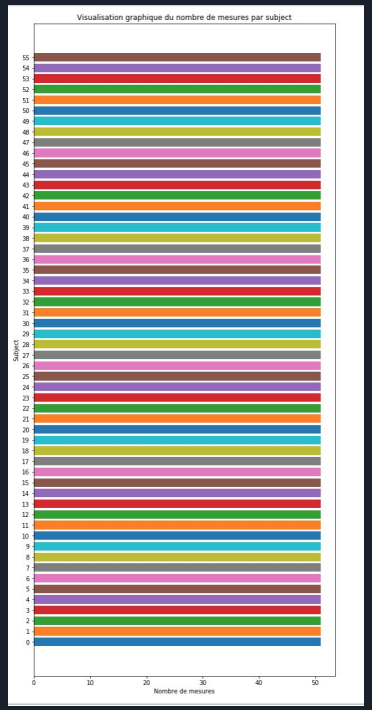
- Scrapper la page web pour récupérer le lien du data folder (url contenant toute la data du dataset
- récupérer les liens à télécharger depuis l'url du data folder
- Afficher les urls à télécharger
- Téléchargement du dataset et sauvegarde de celui-ci

Présentation du data set

Le MEU-Mobile KSD (Keystroke Dynamics) dataset de UCI Machine Learning Repository contient 51 enregistrements pour chacun des 56 sujets - 2856 enregistrements total - des haptiques, de quantité de mouvement et de caractéristiques temporelles mesurées d'une séquence commune (.tie5Roanl) tapé sur un appareil mobile Nexus 7.

Il y a 71 fonctionnalités surveillées,caractérisé par les attributs Hold, Up-Down, Down-Down, Pressure, Finger-Area, Average Hold, Average Pressure, Average Area.

Visualisation des données (½)



On a 2856 samples réparties sur 56 subjets de 51 samples.

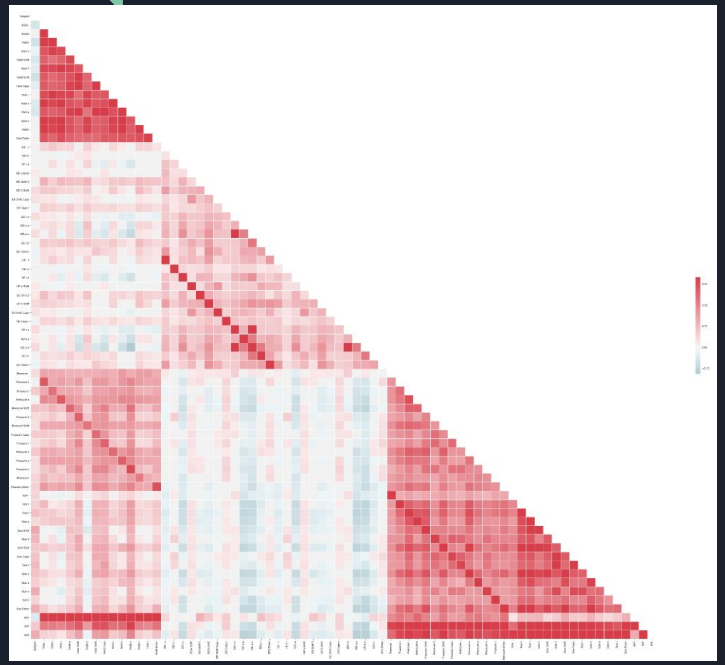
On a 71 features dans ce dataset.

Graph présent sur le notebook

71 features

	Subject	Hold .	...	AvA
56 subjects 51 records/ subject	1	89	…	0.2880184
	…	…	…	…
	56	80	…	0.260369

Visualisation des données (2/2)



On a la matrice de corrélation de ce dataset

On a bien une corrélation entre les differents mesures de Hold, Pressure et Finger-Area avec leurs moyennes

On a bien un corrélations aussi avec les différentes mesures des Hold

Le Modèle

- Choix de faire une classification à multi classe pour savoir à qu'elle classe (Subject) appartient le téléphone

- Choix de faire une RandomForest Classifier car on beaucoup de données et qu'on a des corrélations sur le modèle (facilite la classification multi classe)

Grid Search

```
La GridSearchCV met 929.75 secondes pour afficher les 324 candidats pour l'optimisation des hyperparametres.
Model with rank: 1
Mean validation score: 0.882 (std: 0.047)
Parameters: {'bootstrap': False, 'criterion': 'entropy', 'max_depth': 9, 'max_features': 'auto', 'min_samples_split':
3, 'n_estimators': 71}

Model with rank: 2
Mean validation score: 0.880 (std: 0.036)
Parameters: {'bootstrap': False, 'criterion': 'entropy', 'max_depth': 9, 'max_features': 'sqrt', 'min_samples_split':
3, 'n_estimators': 71}

Model with rank: 3
Mean validation score: 0.878 (std: 0.044)
Parameters: {'bootstrap': False, 'criterion': 'entropy', 'max_depth': 9, 'max_features': 'sqrt', 'min_samples_split':
7, 'n_estimators': 71}
```

Voici les 3 meilleures résultats de notre grid search

Optimisation du modèle

- Accuracy de 91%
- Gain de 10%

ROC & API

- Roc non faite car on a un problème de classification binaire ce qui rend impossible faire un ROC sur python

- Api fait avec Flask Api
  - Modèle sauvegardé
  - Sauvegarde de X_test extrait du dataframe
  - http://127.0.0.1:5001/api pour voir les prédictions