Pandas

Cours M2 CCN

Modélisation des problèmes scientifiques

2019/2020

Pourquoi utiliser pandas ?

- Pour charger, traiter et analyser des bases de données

- Pourquoi pandas plutôt qu'excel, ou libreoffice calc ? Parce qu'excel gère mal les grosses bases de données !

  Comparaison excel-pandas

Variable

Individus

Prénom	Age	Taille
Adele	45	168
Alex	30	182
Alice	28	153
Bob	52	161
...	...	...

Dataframe avec pandas

Import

Entrée [1]:
```
import numpy as np
import pandas as pd
```

Création d'une base de données

Entrée [2]:
```
coucou = pd.DataFrame({"id" : np.arange(1,7,1),
                       "bonjour" : np.array(("Salut","Hello","Buongiorno","Guten Tag", "Buenos dias", "Demat")),
                       "langue" : np.array(("Français","Anglais","Italien", "Allemand", "Espagnol", "Breton"))})

coucou.head() # affiche les 5 premières lignes
```

Out[2]:

	id	bonjour	langue
0	1	Salut	Français
1	2	Hello	Anglais
2	3	Buongiorno	Italien
3	4	Guten Tag	Allemand
4	5	Buenos dias	Espagnol

Import d'une base de données

Entrée [3]:
```
pd.read_table("ronfle.csv", delimiter =',', index_col = 0).head()
```

Out[3]:

	AGE	POIDS	TAILLE	ALCOOL	SEXE	RONFLE	TABA
P0001	47	71	158	0	0	0	1
P0002	56	58	164	7	0	1	0
P0003	46	116	208	3	0	0	1
P0005	70	96	186	3	0	0	1
P0006	51	91	195	2	0	1	1

Marche en local ou avec une page web

Attributs des dataframes

Index

Entrée [4]:
```
coucou.index # appel à l'index
# l'index est la colonne grise qui identifie chacun des individus devant les données.
# On peut fixer la variable id comme index par la commande suivante:
coucou.set_index("id")
```

Out[4]:

id	bonjour	langue
1	Salut	Français
2	Hello	Anglais
3	Buongiorno	Italien
4	Guten Tag	Allemand
5	Buenos dias	Espagnol
6	Demat	Breton

L'index permet d'identifier les individus

Types des variables

Entrée [3]:
```
coucou.dtypes
```

Out[3]:
```
id          int32
bonjour    object
langue     object
dtype: object
```

Types de variables

Nombre total de données

Entrée [5]:
```
coucou.size
```

Out[5]:
```
18
```

Nombre de colonnes

Entrée [6]:
```
coucou.columns # libellés des colonnes
```

Out[6]:
```
Index(['id', 'bonjour', 'langue'], dtype='object')
```

Dimensions du DataFrame

Entrée [7]:
```
coucou.shape
```

Out[7]:
```
(6, 3)
```

Accès et filtrage des données

Appel aux données

Entrée [8]:
```
# Variable(s)

coucou["bonjour"] # pour appeler bonjour seulement

coucou[["bonjour","langue"]] # pour appeler deux variables ou plus

# Individus

coucou.loc[2] # le troisième individu
```

Out[8]:
```
id                  3
bonjour     Buongiorno
langue         Italien
Name: 2, dtype: object
```

Requête

Entrée [9]:
```
coucou.query("id < 4")
```

Out[9]:

	id	bonjour	langue
0	1	Salut	Français
1	2	Hello	Anglais
2	3	Buongiorno	Italien

Entrée [10]:
```
coucou.query("langue == 'Français'")
```

Out[10]:

	id	bonjour	langue
0	1	Salut	Français

coucou[coucou.columns[i]]
donne la i+1-ème colonne

Requêtes directes sur la base de données :

9 → renvoie le dataframe avec les individus ayant un id inférieur à 4

10 → renvoie le dataframe avec les individus pour lesquels la variable langue vaut ‘Français’

Aggrégation de données

Agrégation sur les dataframes

Entrée [23]:
```
glace = pd.DataFrame({"glace" : np.array(('pistache', 'menthe', 'citron', 'ananas', 'pistache', '
                      "couleur" : np.array(('vert', 'vert', 'jaune', 'jaune', 'vert', 'jaune', 'jaune', '
                      "quantite" : np.array([1,2,3,1,2,3,2,2,3,2])})
glace
```

Out[23]:

	glace	couleur	quantite
0	pistache	vert	1
1	menthe	vert	2
2	citron	jaune	3
3	ananas	jaune	1
4	pistache	vert	2
5	ananas	jaune	3
6	citron	jaune	2
7	citron	jaune	2
8	ananas	jaune	3
9	menthe	vert	2

Entrée [24]:
```
glace.groupby(["glace"]).sum()
```

Out[24]:

glace	quantite
ananas	7
citron	7
menthe	4
pistache	3

Entrée [25]:
```
glace.groupby(["couleur"]).mean()
```

Out[25]:

couleur	quantite
jaune	2.333333
vert	1.750000

Variables quantitatives : tout ce qui est dénombrable, une mesure numérique
Ici la quantité de glace

Variables qualitatives : une catégorie, un renseignement, une qualification
Ici un parfum ou une couleur de glace

Group by, comme en SQL

Pour aller plus loin

- Jointures de tables
- Ajouter ou supprimer des lignes/colonnes
- Comprendre les series
- Gestion des manquants (fillna)
- Export de données au format csv (to_csv)
- S’intéresser au type de pandas pkl pour un import plus rapide

Quelques liens sur pandas

- La doc : https://pandas.pydata.org/
- Encore lui ! http://eric.univ-lyon2.fr/~ricco/tanagra/fichiers/fr_Tanagra_Data_Manipulation_Pandas.pdf
- Un script sur le github

- Pareil, il faut tester les fonctions et se les approprier. N'hésitez pas à googler 'pandas python' et à lire un ou deux article(s) pour vous faire une idée sur la librairie, ce qui est utile, les avantages, les faiblesses, etc.

  Ça évite d'apprendre « parce que c'est au programme » ;)