Pandas

Module pour l'analyse de données, pouvant se substituer à l'utilisation d'un tableur. Une différence fondamentale de la librairie pandas avec NumPy, c'est que les tableaux NumPy (NumPy arrays) ont le même type (dtype) pour le tableau entier, tandis que les tableaux pandas (pandas DataFrames) sont caractérisés par un type unique (dtype) par colonne.

Installation

- Instructions sur le site officiel
- Installé avec Anaconda
- Ubuntu : pip3 install pandas

Documentation

- Documentation officielle
- 10 minutes to pandas
- cookbook
- Visualisation
- Pandas Cheat Sheet for Data Science in Python

Applications, exemples

Préambule : variable aléatoire et distributions

De nombreuses grandeurs mesurées suivent une loi de distribution normale pour leur probabilité : *cf.* Loi normale

Voir aussi les documents de statistique élémentaire (niveau licence, France) sur le site wikistat.fr

- **Variable aléatoire** : une variable aléatoire X est définie sur l'espace des observables (espace des événements possibles). À chaque valeur possible x correspond une probabilité $P(x)$ que X soit égale à x
  - Variable aléatoire discrète : si $x_1, x_2, x_3, ...$ constitue l'ensemble discret des valeurs possibles de X, les $P(x_i)$ forment la **distribution de probabilité** de la variable aléatoire X
  - Variable aléatoire continue : si x peut varier continûment, $P(x)$ est la densité de probabilité que la variable prenne une valeur comprise entre x et $x+dx$. L'unité de $P(x)$ est donc en inverse de celle de l'espace des x et seul $P(x) dx$ a la dimension d'une probabilité (nombre) : $P(x) dx = P(x \le X < x+dx)$
  - Positivité :
    - $P(x_i) \ge 0$ pour tout x_i (variable aléatoire discrète)
    - $P(x) \ge 0$ pour tout x (variable aléatoire continue)
  - Normalisation :

        - $\sum_{x_i} P(x_i) =1$ (variable aléatoire discrète)
        - $\int_{\Omega} P(x) dx = 1$ (variable aléatoire continue)
- Toute l'information sur une expérience est contenue dans la distribution $P(x)$}
- Une description **équivalente** est donnée par l'ensemble de toutes les grandeurs caractéristiques appelées {\bf moments de la distribution} :
    - $<X^n> = \sum_i x_i^n P(x_i)$ (variable aléatoire discrète, avec n fini)
    - $<X^n> = \int_{\Omega} x^n P(x) dx$ (variable aléatoire continue, avec n infini)
- Une description **simplifiée** est obtenue en ne tenant compte que de quelques plus petites valeurs de n :
    - Premier moment: moyenne $<X>$ (ou espérance mathématique)
    - Second moment: largeur de la distribution (variance σ^2)
    - Troisième moment : asymétrie (skewness)
    - Quatrième moment : aplatissement (kurtosis)
    - ...
- Les deux premiers moments
    - **Valeur moyenne ou espérance**
        - $<X> = \sum_i x_i \ P(x_i)$ ou $<X> = \int_{{\Omega}} x \ P(x) dx$ avec ${\Omega}$ le volume de l'espace des phases/observables
    - **Variance**
        - La variance $Var(X)$ ou σ^2 caractérise la largeur de la distribution (ou l'écart à la moyenne) : $\sigma^2 = <(X - <X>)^2> = <X^2> - <X>^2$. La racine carrée est l'écart type, σ.

Statistiques sur les dimensions des humains (body dimensions)

Programme basé sur Exploring Relationships in Body Dimensions.

Extensions :

- Tester et utiliser en mode “Jupyter”
- créer des régressions
- autres représentations
- différentiation suivant le genre, l'âge
- utiliser d'autres fonctions, comme nsmallest() et nlargest(), value_counts() (se baser sur la documentation officielle)
- ...

jse-dataset-body-dimensions-read-10.py

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Created on Tue Mar  5 04:13:51 2019
Statistics on Body dimensions :
http://jse.amstat.org/v11n2/datasets.heinz.html

without requests lib, using pandas.read_csv

@author: Didier Villers
"""
```

```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

# using this VARIABLE DESCRIPTIONS with PEP 8 Python style :
names = [
    'Biacromial diameter',
    'Biiliac diameter',
    'Bitrochanteric diameter',
    'Chest depth',
    'Chest diameter',
    'Elbow diameter',
    'Wrist diameter',
    'Knee diameter',
    'Ankle diameter',
    'Shoulder girth',
    'Chest girth',
    'Waist girth',
    'Navel girth',
    'Hip girth',
    'Thigh girth',
    'Bicep girth',
    'Forearm girth',
    'Knee girth',
    'Calf maximum girth',
    'Ankle minimum girth',
    'Wrist minimum girth',
    'Age',
    'Weight',
    'Height',
    'Gender',
    ]
# using Pandas column names without white spaces
names = [name.replace(' ', '_') for name in names]
print(names)

namesfr = [
    'Largeur des épaules',
    'Largueur des hanches',
    'Largueur entre têtes de fémur',
    'Epaisseur du thorax',
    'Largueur du thorax',
    'Largueur du coude',
    'Largueur du poignet',
    'Largueur du genou',
    'Largueur de la cheville',
    'Tour d’épaules',
    'Tour de poitrine',
    'Tour de taille',
    'Tour au niveau du nombril',
    'Tour de hanches',
```

```
    'Tour de cuisse',
    'Tour du biceps',
    'Tour de l'avant-bras',
    'Tour de genou',
    'Plus grande circonférence du mollet',
    'Plus petite circonférence de la cheville',
    'Plus petite circonférence du poignet',
    'Âge',
    'Poids',
    'Taille',
    'Genre',
    ]

dict_names_fr = dict(zip(names, namesfr))
print(dict_names_fr)

file_url = "http://linus.umons.ac.be/body.dat.txt" # file copy
#file_url = "http://jse.amstat.org/datasets/body.dat.txt"
# using read_csv
#
https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.read_
csv.html
# https://www.datacamp.com/community/tutorials/pandas-read-csv

df = pd.read_csv(file_url, header=None, names=names, delimiter='  | ',
engine='python', index_col=False)
#
print(df)

# pandas misc
#
https://stackoverflow.com/questions/15315452/selecting-with-complex-cri
teria-from-pandas-dataframe
#
print(df.columns)
print(df.Age)
print(df.dtypes)
print(df.describe())
print(df[df.Gender == 1].describe())
print(df[df.Age == 20].describe())
print(df.sort_values(by = 'Height'))

print(df.query('Age > 25 and Age < 30'))
print(df.query('25 < Age < 30'))

plt.figure()
ax = df[df.Gender == 1].plot.scatter(x='Height', y='Weight',
color='Red', label='Male');
df[df.Gender == 0].plot.scatter(x='Height', y='Weight', color='Green',
label='Female', ax=ax);
```

```
plt.figure()
df.Height.plot.hist()

plt.figure()
df.Weight.plot.hist()
```

Interface utilisateur graphique

- PandasGUI
  - PandasGUI: Analyzing Pandas dataframes with a Graphical User Interface - Accessing Pandas Dataframes with a simple click of the mouse Parul Pandey, Medium, 24/10/2020
- bamboolib (closed source - non libre)

Références

- https://www.tutorialspoint.com/python_pandas/index.htm
- https://medium.com/@devopslearning/introduction-to-pandas-for-data-analysis-c14bb9b1c21b (limité)
- First Python Notebook. A step-by-step guide to analyzing data with Python and the Jupyter Notebook (The course will teach you how to use pandas to read, filter, join, group, aggregate and rank structured data. You will also learn how to record, remix and republish your analysis using the Jupyter Notebook) → commencer au chapitre 3 “Import pandas into a Jupyter Notebook”
- Python Data Analysis with pandas
- Python Pandas Tutorial : Learn Pandas for Data Analysis
- Minimally Sufficient Pandas
  - *cf.* aussi Master Data Analysis with Python et les données disponibles comme exemples
- Python for Data Science: 8 Concepts You May Have Forgotten
- 23 great Pandas codes for Data Scientists
- fonction merge : https://towardsdatascience.com/why-and-how-to-use-merge-with-pandas-in-python-548600f7e738
- Helpful Python Code Snippets for Data Exploration in Pandas
- Selecting Subsets of Data in Pandas: Part 1
- Statistical Data Analysis in Python
- How to Make Boxplots in Python with Pandas and Seaborn? (and Gapminder dataset)
- Box plot visualization with Pandas and Seaborn
- Complete Guide to Data Visualization with Python (avec différentes librairies : matplotlib, Seaborn, bokeh, Altair, Folium avec des cartes,…)
- Introducing Bamboolib — a GUI for Pandas (utilisation gratuite ou payante via une activation nécessaire dans jupyter)
  - https://bamboolib.8080labs.com/
  - https://github.com/tkrabel/bamboolib
- Violin Plot — It’s Time to Ditch the Box Plots
- Reshape pandas dataframe with melt in Python — tutorial and visualization
  - pandas.melt (documentation)

- groupby :
  - Learn how to master groupby function in Python now | by WY Fok | Towards Data Science
  - Less known Pandas groupby applications in Python | by WY Fok | Aug, 2020 | Towards Data Science
- remplacer Excel (ou calc de libreoffice)
  - Ditch Excel! — A Primer to Python - Pandas one-liners for popular excel stuff, Medium 01/09/2020
- How NOT to write pandas code

Datasets

- Adult Data Set Predict whether income exceeds $50K/yr based on census data. Also known as “Census Income” dataset
- https://archive.ics.uci.edu/ml/datasets.php

Exemples divers

- http://pbpython.com/pandas_transform.html
- http://blog.yhat.com/posts/visualize-nba-pipelines.html
- https://tomaugspurger.github.io/
- régressions linéaires :
  - http://www.xavierdupre.fr/app/ensae_teaching_cs/helpsphinx/notebooks/td2a_eco_regressions_lineaires.html
- Effectively visualize data across time to tell better stories (Pandas & Plotly)

From:
https://dvillers.umons.ac.be/wiki/ - **Didier Villers, UMONS - wiki**

Permanent link:
https://dvillers.umons.ac.be/wiki/teaching:progappchim:pandas?rev=1603796231

Last update: **2020/10/27 11:57**