Ngôn ngữ lập trình Python
Giới thiệu về thư viện Pandas

Vu Tien Dung - Ngo The Quyen

Nội dung

Giới thiệu về thư viện Pandas

Thư viện Pandas

- Cung cấp các cấu trúc dữ liệu và các công cụ thiết kế để làm việc với dữ liệu giống như bảng.
- Cung cấp các chức năng thao tác dữ liệu: biến đổi cấu trúc, trộn dữ liệu, sắp xếp dữ liệu, lát cắt và tổng hợp dữ liệu.
- Cho phép xử lý dữ liệu bị thiếu.
- Link: http://pandas.pydata.org/

Khai báo thư viện Pandas

```
import pandas as pd
```

Giới thiệu về thư viện Pandas

Cấu trúc dữ liệu cơ bản của thư viện Pandas

- Cấu trúc dữ liệu Series
- Cấu trúc dữ liệu DataFrame

Khai báo thư viện Pandas

```
import pandas as pd
```

Giới thiệu về thư viện Pandas

Cấu trúc dữ liệu cơ bản Series

Cấu trúc dữ liệu Series là một mảng dữ liệu một chiều được đánh chỉ mục

```
1        data=pd.Series([1,2,3,4])
2        print(data)
3        data=pd.Series([1,2,3,4],index=['a','b','c','d'])
4        print(data)
5        print(data.values)
6        print(data.index)
7        print(data*10)
8        print(data.describe())
```

Đọc dữ liệu sử dụng thư viện Pandas

Cấu trúc dữ liệu cơ bản DataFrame

DataFrame là một tập các đối tượng Series

Đọc dữ liệu từ tệp csv

```
1 df=pd.read_csv(Ten tep du lieu)
2 #Khong xem dong dau tien la dong tieu de
3 df1=pd.read_csv(tentep,header=None)
4 #Dat tieu de cho cac cot du lieu
5 df2=pd.read_csv(tentep,names=['Danh sach tieu de'])
```

Đọc dữ liệu sử dụng thư viện Pandas

Đọc dữ liệu từ một số tệp có định dạng khác

```
1
2
3
4
```

Đọc dữ liệu sử dụng thư viện Pandas

Hiển thị dữ liệu dạng khung

```
1 print(df.head())
2
```

Hiển thị 5 dòng dữ liệu đầu tiên

Các kiểu dữ liệu của DataFrame

Danh sách các kiểu dữ liệu của DataFrame

Kiểu dữ liệu Pandas	Mô tả
object	Kiểu dữ liệu tổng quát
int64	Dữ liệu số nguyên
float64	Dữ liệu số thực
datetime64, timedelta[ns]	Dữ liệu thời gian

Các kiểu dữ liệu của DataFrame

Kiểm tra kiểu dữ liệu

```
1 #Kiem tra kieu du lieu cua mot cot
2 print(df['salary'].dtype)
3 #Kiem tra kieu du lieu cua cac cot
4 print(df.dtypes)
5
```

Các thuộc tính của đối tượng DataFrame

Danh sách các thuộc tính

Thuộc tính	Mô tả
dtypes	Danh sách các kiểu dữ liệu của các cột
columns	Danh sách tên các cột dữ liệu
axes	Danh sách chỉ số dòng và tên các cột dữ liệu
ndim	Số chiều dữ liệu
size	Kích thước dữ liệu
shape	Kích thước theo từng chiều dữ liệu
values	Giá trị biểu diễn của dữ liệu

Các phương thức của đối tượng DataFrame

Phương thức	Mô tả
head(n),tail(n)	Liệt kê n dòng dữ liệu đầu tiên hoặc cuối cùng
describe()	Liệt kê các số liệu thống kê cơ bản
min(),max()	Liệt kê giá trị nhỏ nhất, lớn nhất trên mỗi cột dữ liệu số
mean(),median()	Liệt kê giá trị trung bình, trung vị trên mỗi cột dữ liệu số
std()	Liệt kê độ lệch chuẩn trên mỗi cột dữ liệu số
sample(n)	Lấy n mẫu dữ liệu ngẫu nhiên từ khung dữ liệu
dropna()	Loại bỏ tất cả các dữ liệu chưa xác định

Lựa chọn một cột của đối tượng DataFrame

- Sử dụng tên cột: df['sex']
- Sử dụng tên cột như một thuộc tính: df.sex

Gom nhóm

Sử dụng phương thức group by chúng ta có thể thực hiện các chức năng sau

- Phân chia dữ liệu thành các nhóm dựa trên bộ lọc
- Thực hiện tính toán số liệu thống kê trên mỗi nhóm dữ liệu

Gom nhóm

Ví dụ 1

```
1       #Nhom du lieu su dung cot rank
2        df_rank=df.groupby(['rank'])
3       #Tinh toan gia tri trung binh tren moi cot so cho cac nhom
        du lieu
4       print(df_rank.mean())
5
```

Gom nhóm

Mỗi nhóm đối tượng được tạo, chúng ta có thể tính toán số liệu thống kê trên mỗi nhóm này

Ví dụ 1

```
1  #Tinh toan luong trung binh cho cac nhom phan loai
2   df_rank=df.groupby('rank')['salary'].mean()
3  print(df_rank)
4
5  df_rank=df.groupby('rank')[['salary']].mean()
6  print(df_rank)
7
```

Gom nhóm

Chú ý khi thực hiên phép toán gom nhóm

- Không xảy ra quá trình gom nhóm hoặc tách nhóm khi chưa thực sự cần thiết. Tạo đối tượng gom nhóm chỉ xác minh rằng bạn đã truyền một ánh xạ hợp lệ
- Mặc định các khóa trong mỗi nhóm phân loại được sắp xếp. Để tăng tốc độ xử lý bạn có thể đặt sort = False.

Filtering

Giới thiệu

Để tập hợp dữ liệu, chúng ta có thể áp dụng phương pháp lập chỉ mục Boolean. Chỉ mục này thường được gọi là bộ lọc. Ví dụ: nếu chúng tôi muốn tập hợp các hàng trong đó giá trị tiền lương lớn hơn 120K:

```
1
2
3
```

```
df_sub=df[df['salary']>120000]
print(df_sub)
```

Filtering

- Bất kỳ phép toán so sánh (>,>=,<,<=,==,!=) nào cũng có thể sử dụng để tập hợp dữ liệu

```
1
2
3
```

```
df_f=df[df['sex']=='Famale']
print(df_f)
```

Slicing

Slicing

Một số cách để tập hợp khung dữ liệu

- Sử dụng một hoặc nhiều cột
- Sử dụng một hoặc nhiều dòng
- Một tập hợp các dòng và cột

Slicing

Slicing

Khi chọn một cột, có thể sử dụng một bộ dấu ngoặc đơn, đối tượng kết quả sẽ là một Series (không phải là Khung dữ liệu):

```
1 print(df['salary'])
```

Khi chọn nhiều cột, sử dụng một bộ dấu ngoặc kép, đối tượng kết quả sẽ là một DataFrame

```
1 print(df[['rank','salary']])
```

Slicing

Lựa chọn các dòng

Chỉ định phạm vi các hàng được lựa chọn bằng cách sử dụng ":"

```
1    print(df[10:20])
```

Chú ý dòng đầu tiên được lựa chọn có vị trí là 0, và giá trị cuối trong danh sách phạm vi được bỏ qua.

Data Frames

Phương thức loc

Phương thức loc: xác định phạm vi các hàng được lựa chọn cùng với các nhãn

```
1    print(df.loc[10:20,['rank','salary']])
```

Phương thức iloc

Phương thức iloc: xác định phạm vi các hàng và cột lựa chọn dựa trên vị trí

```
1    print(df_sub.iloc[10:20,[0,3,4,5]])
```

Data Frames

Phương thức iloc

```
1  print(df.iloc[0])
2  print(df.iloc[3])
3  print(df.iloc[-1])
4  print(df.iloc[:,0])
5  print(df.iloc[:,-1])
6  print(df.iloc[0:7])
7  print(df.iloc[:,0:2])
8  print(df.iloc[1:3,0:2])
9  print(df.iloc[[0,5],[1,3]])
```

Data Frames

Phương thức Sorting

- Sắp xếp dữ liệu theo các giá trị trong một cột. Theo mặc định, việc sắp xếp sẽ diễn ra theo thứ tự tăng dần và khung dữ liệu mới được trả về.

```
1 df_sorted=df.sort_values(by='service')
2 print(df_sorted.head())
```

- Sắp xếp dữ liệu sử dụng hai hoặc nhiều cột.

```
1 df_sorted=df.sort_values(by=['service','salary'],ascending=[True,Fa
2 print(df_sorted.head())
```

Các hàm tổng hợp

Giới thiệu

Tập hợp - tính toán số liệu thống kê trên mỗi nhóm như

- Tính toán các giá trị tổng hoặc các giá trị trung bình trê mỗi nhóm dữ liệu
- Tính toán kích thước các nhóm dữ liệu

Một số hàm thống kê

- min,max
- count, sum, prod
- mean, median, mode, mad
- std, var

Các hàm tổng hợp

Phương thức Agg

Phương thức Agg: Khi nhiều số liệu thống kê được tính toán trên mỗi cột

```
1    print(df[['phd','salary']].agg(['min','mean','max']))
```