


Class period 10

บทที่ 5 การเตรียมข้อมูลสำหรับการแสดงผล 2 (ต่อ)

Pandas 102 part2


1



.groupby()

- <https://www.kaggle.com/code/crawford/python-groupby-tutorial>
- คือการจับกลุ่มค่าในคอลัมน์ที่ต้องการ โดยจับค่าข้อมูลที่มีเหมือนกันในคอลัมน์ที่ต้องการเอามาไว้ด้วยกัน เช่น
- `data_covid.groupby('nationality')`
- ตัวแปรที่ใช้เก็บตารางตามด้วย `.groupby()` ("ชื่อคอลัมน์ที่ต้องการใช้จับกลุ่ม") สามารถไล่ได้หลายคอลัมน์โดยจัดคอลัมน์ที่ต้องการให้อยู่ใน list เช่น `(["ชื่อคอลัมน์1", "ชื่อคอลัมน์2"])`
- จากโค้ดต้องการจับกลุ่มค่าข้อมูลในคอลัมน์ `nationality` คำสั่ง `groupby` จะทำการจับกลุ่มข้อมูลทุกแถวทุกคอลัมน์ โดยจับกลุ่มตามค่าข้อมูลที่มีเหมือนกันในคอลัมน์ `nationality`
- `groupby` จะทำงานใน memory การดูผลลัพธ์ของการใช้ `groupby` จะต้องใช้คำสั่งเพิ่มเติม สามารถดูได้หลายแบบโดยการพิมพ์คำสั่งที่ต้องการดูต่อท้าย เช่น `data_covid.groupby('nationality').count()`

2




คำสั่งที่ใช้สำหรับดูผลลัพธ์ของ .groupby()

- ยกตัวอย่าง
- `.count()` ใช้สำหรับดูจำนวนสมาชิกในแต่ละคอลัมน์ในแต่ละกลุ่มที่แบ่งตามค่าข้อมูลที่มีเหมือนกันแต่ละค่าในคอลัมน์ที่ใช้ `groupby`
- `.mean()` ใช้ดูค่า `mean` ในแต่ละคอลัมน์ในแต่ละกลุ่ม (ดูได้แค่คอลัมน์ที่มีข้อมูลเป็นตัวเลข)
- `.max()` ใช้ดูค่าที่สูงสุดในแต่ละคอลัมน์ในแต่ละกลุ่ม (ดูได้แค่คอลัมน์ที่มีข้อมูลเป็นตัวเลข)

Summary statistics	Numpy operations	More complex operations
mean	<code>np.mean</code>	<code>agg()</code>
median	<code>np.median</code>	<code>agg(["mean", "median"])</code>
min	<code>np.min</code>	<code>agg(custom_function)</code>
max	<code>np.max</code>	
sum	<code>np.sum</code>	
describe	<code>np.product</code>	
count or size		


3



เฉลย Homework class period 9 ด้วย groupby()

- สร้างตารางใหม่ ที่ค่าใน `sex` เป็น `missing` ทั้งหมด
- `data_covid['sex'].isnull()`
- ตรวจสอบทว่าว่าง (`missing`) ในคอลัมน์ `sex` และสร้าง list logical expression `True(missing)/False(non missing)`
-
- `missing_sex = data_covid[data_covid['sex'].isnull()]`
- นำ list logical expression มาใช้เลือกข้อมูลในตารางทุก records ที่มีค่าในคอลัมน์ `sex` เป็น `missing` และเก็บตารางที่เลือกแล้วไว้ในตัวแปร `missing_sex`
- `missing_sex` ผลลัพธ์ที่ได้ตารางที่ทุก records มีค่าในคอลัมน์ `sex` เป็น `missing`


4



เฉลย Homework class period 9 ด้วย groupby()

- สรุปว่าทำไม record นั้นๆถึงเป็น `missing` ใช้ `groupby` และ `.describe()` ดูค่าทางสถิติของข้อมูลเพื่อหว่าทำไม `sex` ถึง `missing` โดยการตรวจสอบ data หลายมุม เช่น
- `missing_sex.groupby('nationality').describe()`
- `missing_sex.groupby('province_of_onset').describe()`
- `missing_sex.groupby(['province_of_onset', 'nationality']).describe()`
- `missing_sex_no_burma = missing_sex[missing_sex['nationality'] != 'Burma']`
- `missing_sex_no_burma.groupby('risk').describe()`
- `missing_sex.groupby('risk').describe()`

5



create pandas table

Creating Pandas DataFrames from Python Lists and Dictionaries

Dictionary

```

sales = [{"account": "Jones LLC", "dept": "Sales", "revenue": 150, "risk": "Low"},
         {"account": "Alpha Co", "dept": "Sales", "revenue": 200, "risk": "High"},
         {"account": "Blue Inc", "dept": "Sales", "revenue": 180, "risk": "Medium"}]
df = pd.DataFrame(sales)

```

List

```

sales = [{"account": "Jones LLC", "dept": "Sales", "revenue": 150, "risk": "Low"},
         {"account": "Alpha Co", "dept": "Sales", "revenue": 200, "risk": "High"},
         {"account": "Blue Inc", "dept": "Sales", "revenue": 180, "risk": "Medium"}]
df = pd.DataFrame(sales, columns=labels)

```

Row Oriented

	account	dept	revenue	risk
0	Jones LLC	Sales	150	Low
1	Alpha Co	Sales	200	High
2	Blue Inc	Sales	180	Medium

from_records

Column Oriented

```

sales = [{"account": "Jones LLC", "dept": "Sales", "revenue": 150, "risk": "Low"},
         {"account": "Alpha Co", "dept": "Sales", "revenue": 200, "risk": "High"},
         {"account": "Blue Inc", "dept": "Sales", "revenue": 180, "risk": "Medium"}]
df = pd.DataFrame.from_records(sales)

sales = [{"account": "Jones LLC", "dept": "Sales", "revenue": 150, "risk": "Low"},
         {"account": "Alpha Co", "dept": "Sales", "revenue": 200, "risk": "High"},
         {"account": "Blue Inc", "dept": "Sales", "revenue": 180, "risk": "Medium"}]
df = pd.DataFrame.from_items(sales)

from_dict

```

When using a dictionary, column order is not preserved. Explicit order them.

```
df = df[["account", "dept", "revenue", "risk"]]
```

Practical Business Python - pypython.com

6

1

ตัวอย่างการสร้างตาราง pandas

- แบบ Dictionary ใช้ `pd.DataFrame()`
- ขั้นตอนการสร้าง สร้าง list ขึ้นมาและเขียนค่าแต่ละ record ที่ต้องการในรูปแบบ dictionary โดย index จะเป็นชื่อคอลัมน์และ value จะเป็นค่าของ record นั้นๆ เช่น

```
records = [
    {'account': 'Jones LLC', 'Jan': 150, 'Feb': 200, 'Mar': 140},
    {'account': 'Alpha Co', 'Jan': 200, 'Feb': 210, 'Mar': 215},
    {'account': 'Blue Inc', 'Jan': 50, 'Feb': 90, 'Mar': 95}
]
records_df = pd.DataFrame(records)
```

Dictionary

	account	Jan	Feb	Mar
0	Jones LLC	150	200	140
1	Alpha Co	200	210	215
2	Blue Inc	50	90	95

6

7

ตัวอย่างการสร้างตาราง pandas

- แบบ List ใช้ `pd.DataFrame.from_records()`
- ขั้นตอนการสร้าง กำหนดตัวแปร 2 ตัว
- ตัวแปรที่ 1 ใช้เก็บ value เป็นค่าของ record นั้นๆ โดยสร้าง list ขึ้นมาและเขียนค่าแต่ละ record ที่ต้องการ
- ตัวแปรที่ 2 ใช้เก็บชื่อคอลัมน์ สร้าง list ขึ้นมาและเขียนชื่อคอลัมน์ที่ต้องการ
- การใช้งาน
- `df = pd.DataFrame.from_records(ตัวแปรที่1, columns=ตัวแปรที่2)`
- `df`

```
df = pd.DataFrame.from_records(sales, columns=labels)
```

List

	account	Jan	Feb	Mar
0	Jones LLC	150	200	140
1	Alpha Co	200	210	215
2	Blue Inc	50	90	95

7

8

Simple Visualization

- โหม่งศึกษาควมไม่สอดคล้องกันของ Iris จากลิงค์นี้
- <https://raw.githubusercontent.com/pandas-dev/pandas/master/pandas/tests/io/data/csv/iris.csv>
- ดาวน์โหลดข้อมูลจาก link และเก็บข้อมูลไว้ในตัวแปร
- `df = pd.read_csv("https://raw.githubusercontent.com/pandas-dev/pandas/master/pandas/tests/io/data/csv/iris.csv")`
- `df`
- ลอง `df.groupby("Name").count()` ดูพื้นที่ของดอกไม้

8

9

parallel_coordinates

- `pd.plotting.parallel_coordinates(df, 'Name')`
- การทำงานจะใช้ชื่อคอลัมน์เป็นแกน x
- และใช้ค่าในแต่ละ record เป็นแกน y
- โดยมองค่าข้อมูลแต่ละช่องในตารางเป็น 1 จุด
- 1 record คือ 1 เส้น จากตามค่าข้อมูลแต่ละจุดของ record นั้นๆในแต่ละคอลัมน์
- จัดกลุ่มโดยการแบ่งตามคอลัมน์ Name

9

10

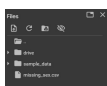
scatter_matrix

- `pd.plotting.scatter_matrix(df)`
- การนำค่าที่เป็นตัวเลขของแต่ละ record ในแต่ละคอลัมน์มาเปรียบเทียบกัน
- ดังนั้น แกน x และ แกน y จะเหมือนกัน คือชื่อคอลัมน์และค่าในแต่ละคอลัมน์
- จะสังเกตว่าถ้าเป็นข้อมูลคอลัมน์เดียวกันเปรียบเทียบกัน จะเห็นเป็นกราฟ histogram
- แต่ถ้าเป็นข้อมูลคนละคอลัมน์มาเปรียบเทียบกัน จะสามารถดูความสัมพันธ์ของข้อมูลที่อยู่คนละคอลัมน์ได้

10

11

save table

- ใช้คำสั่ง `.to_csv()` ในการบันทึกเป็นไฟล์ csv เช่น
- `missing_sex = data_covid[data_covid['sex'].isnull()]`
- `missing_sex`
- ต้องการบันทึกตารางในตัวแปร `missing_sex` สามารถบันทึกได้โดย
- ชื่อตัวแปรตารางที่ต้องการ `.to_csv('ชื่อไฟล์ที่ต้องการบันทึก.csv')` เช่น
- `missing_sex.to_csv('missing_sex.csv')`
- 
- ผลลัพธ์จะอยู่รูปไฟล์คอมพิวเตอร์ด้านซ้ายมือของหน้า google colab สามารถกดดาวน์โหลดได้

11

12