การจัดการข้อมูลด้วย Pandas ใน Python

การปรับแต่งและแปลงข้อมูล

สรุปบทเรียนที่แล้ว

- เรียนรู้วิธีการเลือกข้อมูลจาก DataFrame และ Series
- การเข้าถึงข้อมูลด้วยวิธีต่าง ๆ
- ความสำคัญของการเลือกข้อมูลที่ถูกต้องเพื่อการวิเคราะห์

วัตถุประสงค์

- เรียนรู้การใช้ฟังก์ชันสรุปข้อมูล (Summary Functions)
- เข้าใจการใช้ฟังก์ชันแบบไม่ระบุชื่อ (Anonymous Functions)
- เรียนรู้การแปลงข้อมูลด้วย map() และ apply()
- ฝึกปฏิบัติการแปลงข้อมูลในรูปแบบต่าง ๆ

In [1]:
```
import pandas as pd

reviews = pd.read_csv("datasets/winemag-data-130k-v2.csv", index_col=0)
```

In [2]:
```
reviews
```

Out[2]:

	country	description	designation	points	price	province	region_1	regior
0	Italy	Aromas include tropical fruit, broom, brimston...	Vulkà Bianco	87	NaN	Sicily & Sardinia	Etna	N
1	Portugal	This is ripe and fruity, a wine that is smooth...	Avidagos	87	15.0	Douro	NaN	N
2	US	Tart and snappy, the flavors of lime flesh and...	NaN	87	14.0	Oregon	Willamette Valley	Willame Va
3	US	Pineapple rind, lemon pith and orange blossom ...	Reserve Late Harvest	87	13.0	Michigan	Lake Michigan Shore	N
4	US	Much like the regular bottling from 2012, this...	Vintner's Reserve Wild Child Block	87	65.0	Oregon	Willamette Valley	Willame Va
...	...	...	...	...	...	...	...	
65494	France	Made from young vines from the Vaulorent porti...	Fourchaume Premier Cru	90	45.0	Burgundy	Chablis	N
65495	Australia	This is a big, fat, almost sweet-tasting Caber...	NaN	90	22.0	South Australia	McLaren Vale	N
65496	US	Much improved over the unripe 2005, Fritz's 20...	Estate	90	20.0	California	Dry Creek Valley	Sonc
65497	US	This wine wears its 15.8% alcohol	Block 24	90	31.0	California	Napa Valley	Na

	country	description	designation	points	price	province	region_1	regior
		better than ...						
65498	Spain	A unique take on Manzanilla Sherry, which is o...	Manzanilla	90	10.0	Andalucia	Jerez	N

65499 rows × 13 columns

ฟังก์ชันสรุปข้อมูล (Summary Functions)

- ฟังก์ชันที่ช่วยปรับโครงสร้างข้อมูลให้อยู่ในรูปแบบที่เป็นประโยชน์
- ใช้เพื่อวิเคราะห์และทำความเข้าใจข้อมูลอย่างรวดเร็ว
- For example, consider the describe() method:

In [3]:
```
reviews.describe()
```

Out[3]:

	points	price
count	65499.000000	60829.000000
mean	88.434037	35.232932
std	3.030310	39.477858
min	80.000000	4.000000
25%	86.000000	17.000000
50%	88.000000	25.000000
75%	91.000000	42.000000
max	100.000000	2500.000000

In [4]:
```
# สำหรับข้อมูลตัวเลข
reviews.points.describe()
```

Out[4]:
```
count    65499.000000
mean        88.434037
std          3.030310
min         80.000000
25%         86.000000
50%         88.000000
75%         91.000000
max        100.000000
Name: points, dtype: float64
```

In [5]:
```
# การใช้ describe() กับหลายคอลัมน์สำหรับข้อมูลตัวเลขพร้อมกัน
reviews[['points', 'price']].describe()
```

Out[5]:

	points	price
count	65499.000000	60829.000000
mean	88.434037	35.232932
std	3.030310	39.477858
min	80.000000	4.000000
25%	86.000000	17.000000
50%	88.000000	25.000000
75%	91.000000	42.000000
max	100.000000	2500.000000

In [6]:
```
# สำหรับข้อมูลประเภทข้อความ
reviews.taster_name.describe()
```

Out[6]:
```
count          51856
unique            19
top       Roger Voss
freq           13045
Name: taster_name, dtype: object
```

In [7]:
```
# การใช้ describe() กับหลายคอลัมน์สำหรับข้อมูลประเภทข้อความพร้อมกัน
reviews[['country', 'taster_name']].describe()
```

Out[7]:

	country	taster_name
count	65467	51856
unique	41	19
top	US	Roger Voss
freq	27177	13045

In [8]:
```
# การใช้ describe() กับหลายคอลัมน์สำหรับข้อมูลผสมพร้อมกัน
reviews[['country','taster_name','points']].describe()
```

Out[8]:

	points
count	65499.000000
mean	88.434037
std	3.030310
min	80.000000
25%	86.000000
50%	88.000000
75%	91.000000
max	100.000000

ถ้าเราต้องการข้อมูลสถิติสรุปแบบง่ายๆ เกี่ยวกับคอลัมน์ใน DataFrame หรือ Series โดยทั่วไปจะมีฟังก์ชัน(method) pandas ที่เป็นประโยชน์ซึ่งจะช่วยให้ทำได้ ตัวอย่างเช่น

In [9]:
```
# หาค่าเฉลี่ยใช้ mean()
reviews.points.mean()
```

Out[9]:
```
88.43403716087269
```

In [10]:
```
# ดูค่าที่ไม่ซ้ำใช้ unique()
reviews.taster_name.unique()
```

Out[10]:
```
array(['Kerin O’Keefe', 'Roger Voss', 'Paul Gregutt',
       'Alexander Peartree', 'Michael Schachner', 'Anna Lee C. Iijima',
       'Virginie Boone', 'Matt Kettmann', nan, 'Sean P. Sullivan',
       'Jim Gordon', 'Joe Czerwinski', 'Anne Krebiehl\xa0MW',
       'Lauren Buzzeo', 'Mike DeSimone', 'Jeff Jenssen',
       'Susan Kostrzewa', 'Carrie Dykes', 'Fiona Adams',
       'Christina Pickard'], dtype=object)
```

In [11]:
```
# นับจำนวนค่าที่ซ้ำกันใช้ value_counts()
reviews.taster_name.value_counts()
```

Out[11]:
```
taster_name
Roger Voss            13045
Michael Schachner      7752
Kerin O'Keefe          5313
Paul Gregutt           4851
Virginie Boone         4696
Matt Kettmann          3035
Joe Czerwinski         2605
Sean P. Sullivan       2358
Anna Lee C. Iijima     2134
Jim Gordon             2032
Anne Krebiehl MW       1769
Lauren Buzzeo           938
Susan Kostrzewa         593
Jeff Jenssen            234
Mike DeSimone           231
Alexander Peartree      210
Carrie Dykes             45
Fiona Adams              11
Christina Pickard         4
Name: count, dtype: int64
```

In [12]:
```
reviews.points.value_counts()
```

Out[12]:
```
points
87     8872
88     8423
90     7697
86     6179
91     6016
89     5724
85     5082
92     4917
84     3490
93     3268
94     1905
83     1442
82      923
95      678
81      305
96      262
80      155
97       99
98       39
99       15
100       8
Name: count, dtype: int64
```

ฟังก์ชันแบบไม่ระบุชื่อ (Anonymous Functions)

- เรียกอีกอย่างว่า Lambda Function
- เป็นฟังก์ชันที่ไม่จำเป็นต้องมีชื่อ
- [illegible] ในบรรทัดเดียว

- เหมาะสำหรับฟังก์ชันง่าย ๆ ที่ใช้เพียงครั้งเดียว

In [13]:
```
# ฟังก์ชันแบบปกติ
def f(x):
    return x**2 + x - 1
```

In [14]:
```
# ฟังก์ชันแบบ lambda
g = lambda x: x**2 + x - 1
```

In [15]:
```
# ทั้งสองฟังก์ชันให้ผลลัพธ์เหมือนกัน
x = 10
print('f(x) =',f(x))
print('g(x) =',g(x))
```

```
f(x) = 109
g(x) = 109
```

In [16]:
```
# ฟังก์ชัน lambda ที่รับหลายตัวแปร
h = lambda x, y, z: x**2 + y**2 + z**2
```

In [17]:
```
# ทดสอบการใช้งาน
x, y, z = 0, 1, 1
value = h(x, y, z)
print(value)
```

```
2
```

Map คืออะไร?

- Map คือการแปลงค่าจากชุดข้อมูลหนึ่งไปเป็นอีกชุดข้อมูลหนึ่ง
- ใช้ในการสร้างข้อมูลรูปแบบใหม่จากข้อมูลที่มีอยู่
- ใช้ในการแปลงข้อมูลจากรูปแบบหนึ่งไปเป็นอีกรูปแบบหนึ่ง
- Pandas มีวิธีการ Map สองแบบหลัก ๆ: map() และ apply()

การใช้ map()

- ใช้กับ Series (คอลัมน์เดียว)
- ส่งผ่านค่าแต่ละค่าในคอลัมน์ไปยังฟังก์ชัน
- คืนค่าเป็น Series ใหม่

In [18]:
```
review_points_mean = reviews.points.mean()
reviews.points.map(lambda p: p - review_points_mean)
```

Out[18]:
```
0       -1.434037
1       -1.434037
2       -1.434037
3       -1.434037
4       -1.434037
           ...
65494    1.565963
65495    1.565963
65496    1.565963
65497    1.565963
65498    1.565963
Name: points, Length: 65499, dtype: float64
```

การใช้ apply()

- ใช้กับ DataFrame ทั้งหมด
- สามารถแปลงข้อมูลทั้งแถวหรือทั้งคอลัมน์
- สามารถทำงานที่ซับซ้อนกว่า map()

In [19]:
```
def remean_points(row):
    row['points'] = row['points'] - review_points_mean
    return row
```

In [20]:
```
reviews.apply(remean_points, axis='columns')
```

Out[20]:

	country	description	designation	points	price	province	region_1	re
0	Italy	Aromas include tropical fruit, broom, brimston...	Vulkà Bianco	-1.434037	NaN	Sicily & Sardinia	Etna	
1	Portugal	This is ripe and fruity, a wine that is smooth...	Avidagos	-1.434037	15.0	Douro	NaN	
2	US	Tart and snappy, the flavors of lime flesh and...	NaN	-1.434037	14.0	Oregon	Willamette Valley	Will
3	US	Pineapple rind, lemon pith and orange blossom ...	Reserve Late Harvest	-1.434037	13.0	Michigan	Lake Michigan Shore	
4	US	Much like the regular bottling from 2012, this...	Vintner's Reserve Wild Child Block	-1.434037	65.0	Oregon	Willamette Valley	Will
...	...	...	...	...	...	...	...	
65494	France	Made from young vines from the Vaulorent porti...	Fourchaume Premier Cru	1.565963	45.0	Burgundy	Chablis	
65495	Australia	This is a big, fat, almost sweet-tasting Caber...	NaN	1.565963	22.0	South Australia	McLaren Vale	
65496	US	Much improved over the unripe 2005, Fritz's 20...	Estate	1.565963	20.0	California	Dry Creek Valley	S
65497	US	This wine wears its 15.8% alcohol	Block 24	1.565963	31.0	California	Napa Valley	

	country	description	designation	points	price	province	region_1	re
		better than ...						
65498	Spain	A unique take on Manzanilla Sherry, which is o...	Manzanilla	1.565963	10.0	Andalucia	Jerez	

65499 rows × 13 columns

ความแตกต่างของ axis

- `axis='columns'` หรือ `axis=1` : ทำงานกับแถว
- `axis='index'` หรือ `axis=0` : ทำงานกับคอลัมน์

ข้อสังเกต

map() และ apply() จะ return ค่า Series และ DataFrames ที่ถูกแปลงใหม่ตามลำดับ โดยจะไม่แก้ไขข้อมูลเดิมที่เรียกใช้ หากเราดูที่แถวแรกของreviews เราจะเห็นว่ามันยังคงมีค่าคะแนนเดิมอยู่

In [21]:
```
reviews.head(1)
```

Out[21]:

	country	description	designation	points	price	province	region_1	region_2	tast
0	Italy	Aromas include tropical fruit, broom, brimston...	Vulkà Bianco	87	NaN	Sicily & Sardinia	Etna	NaN	

การใช้ Operator เพื่อความเร็วในการคำนวณ

- Pandas มีการดำเนินการแมป (mapping) ที่ใช้บ่อยหลายอย่างเป็นฟังก์ชัน built-ins ในตัว
- Pandas เข้าใจการทำงานระหว่าง Series กับ single value
- มีความเร็วสูงกว่าการใช้ map() หรือ apply()

In [22]:
```
review_points_mean = reviews.points.mean()
reviews.points - review_points_mean
```

Out[22]:
```
0       -1.434037
1       -1.434037
2       -1.434037
3       -1.434037
4       -1.434037
           ...
65494    1.565963
65495    1.565963
65496    1.565963
65497    1.565963
65498    1.565963
Name: points, Length: 65499, dtype: float64
```

จากคำสั่ง Code ด้านบนนี้ เราจะดำเนินการระหว่างค่าต่างๆ มากมายทางด้านซ้ายมือ (ทุกค่าในซีรีส์) และค่าเดียวทางด้านขวามือ (ค่าเฉลี่ย) Pandas จะพิจารณานิพจน์นี้และคำนวณว่าเราต้องลบค่าเฉลี่ยออกจากค่าทุกค่าในชุดข้อมูล

นอกจากนี้ Pandas ยังเข้าใจด้วยว่าต้องทำอย่างไรหากเราดำเนินการเหล่านี้ระหว่างซีรีส์ที่มีความยาวเท่ากัน ตัวอย่างเช่น วิธีง่ายๆ ในการรวมข้อมูลประเทศและภูมิภาคในชุดข้อมูลคือทำดังต่อไปนี้:

In [23]:
```
# การผสมข้อมูลด้วย Operator : สามารถใช้กับ Series ที่มีความยาวเท่ากัน
# การรวมข้อมูลประเทศกับภูมิภาค
reviews.country + " - " + reviews.region_1
```

Out[23]:
```
0                   Italy - Etna
1                            NaN
2          US - Willamette Valley
3        US - Lake Michigan Shore
4          US - Willamette Valley
                   ...
65494             France - Chablis
65495    Australia - McLaren Vale
65496       US - Dry Creek Valley
65497             US - Napa Valley
65498                Spain - Jerez
Length: 65499, dtype: object
```

ความแตกต่างระหว่าง Operator และ map()/apply()

- Operator: เร็วกว่า แต่ใช้ได้กับการคำนวณพื้นฐาน
- map()/apply(): ยืดหยุ่นกว่า สามารถใช้กับโลจิกที่ซับซ้อน

Your turn

If you haven't started the exercise, you can now.

In [ ]:
```
```