

Day 12: *Stack and Unstack in Pandas*

Maybe Dot wouldn't end up living in Dubai – the luxurious lifestyle really appealed to them, but they just weren't sure their pockets were deep enough for such a high-class existence. Either way, it was time for them to depart from the Persian Gulf and move to a wholly different continent. They were headed southeast – to Australia. Dot was in for a very long flight, so they made sure to load their tablet up with books before they departed. They lay back in their seat and started reading away, and the time flew by. Before they knew it, they had landed in Perth.

Entering the city, Dot headed over to Kings Park, where they wandered around looking at the gardens amid the beautiful sunset. A fellow park visitor struck up a conversation with them. "Are you a tourist? What a great park this is, don't you think?" the friendly stranger opined. "You should visit during the summer, though, when everything's properly in bloom and the weather's hot." Dot blinked at them, stunned. "What do you mean? This is the summer," they said. The stranger burst into a chuckle, shaking his head. "A funny one, are you?" he said as he trotted away. Dot stood thinking for a few minutes before their blunder dawned on them. They were now in the southern hemisphere – maybe that meant that the seasons were reversed? They pulled up some data about historical weather patterns in Australia to validate this. Let's look at the differences in temperatures between seasons to help Dot make sense of things.

Tutorial

In today's challenge, we will take a look at two more pandas functions that are often used for data wrangling: **stack** and **unstack**.

```
import pandas as pd

# we define the dataframe
df = pd.DataFrame([[25.69, 7692000], [5.084, 268021]],
                  index=['Australia', 'New Zealand'],
                  columns=['population', 'area'])

# we apply the function stack()
stacked = df.stack()
```

Observe what happened with our DataFrame. The `stack()` function stacks both columns into one, and creates something we call a **MultilIndex**.

```
print(stacked.index)
```

If you want to learn more, you can find information about the MultilIndex in [this article](#).

The **MultilIndex** is also the output of the `group by` method when we use it on more than one column.

```
df = pd.DataFrame(["Southern", "Southern", "Southern", "Southern"],
                  ["Austria", "Australia", "New Zealand", "New Zealand"],
                  ["Sydney", "Melbourne", "Auckland", "Wellington"],
                  [5.312, 5.078, 1.463, 0.215])
```

```
index=['hemisphere','country','city','population']
).transpose()
```

```
grouped = df.groupby(["hemisphere","country"])[["population"]].mean()
```

Observe what happened with our DataFrame.

```
print(grouped.index)
```

Now, we can use the **unstack()** function to expand the MultiIndex into separate columns.

```
grouped.unstack(level=1)
```

Play around with the parameter level. We can set it as either 0 or 1. What difference does it make which you choose?

We can see more examples of stack and unstack [here](#) and [here](#) respectively.

In [1]:

```
import pandas as pd
df = pd.read_csv('aus_weather.csv')
df.head()
```

Out[1]:

	Date	Location	MinTemp	MaxTemp	Rainfall	Sunshine	WindSpeed9am	WindSpeed3pm	Humidity
0	2008-12-01	Albury	13.4	22.9	0.6	NaN	20.0	24.0	
1	2008-12-02	Albury	7.4	25.1	0.0	NaN	4.0	22.0	
2	2008-12-03	Albury	12.9	25.7	0.0	NaN	19.0	26.0	
3	2008-12-04	Albury	9.2	28.0	0.0	NaN	11.0	9.0	
4	2008-12-05	Albury	17.5	32.3	1.0	NaN	7.0	20.0	

Challenge

Step I: Using the season column, filter the DataFrame so it contains only rows from summer and winter.

Step II: Using groupby() and unstack(), compute the difference between the average temperature in summer and winter(at 9am) for all locations.

Question: What is the difference between the average temperatures in summer and winter in Adelaide, Albany and Albury?

In [2]:

```
# SOLUTION

df = df[df.season.isin(["winter","summer"])]
```

```
unstack = df.groupby(["Location","season"])[['Temp9am']].mean().unstack()  
(unstack.iloc[:,0] - unstack.iloc[:,1]).tail(6)
```

Out[2]:

Location	
Walpole	7.580851
Watsonia	9.739332
Williamstown	10.725402
Witchcliffe	8.162081
Wollongong	7.459804
Woomera	13.454238

dtype: float64

In [5]:

```
print((unstack.iloc[:,0] - unstack.iloc[:,1])['Adelaide'])  
print((unstack.iloc[:,0] - unstack.iloc[:,1])['Albany'])  
print((unstack.iloc[:,0] - unstack.iloc[:,1])['Albury'])
```

10.71286498520401
7.395200302343159
14.257000981580923