Applied Data Analytics

Pandas basics

Merging Series / DataFrames

Hans-Martin von Gaudecker and Aapo Stenhammar

Same units, different sources

- Country data on life expectancy and GDP per capita
- Annual data on life expectancy and countries' distance from the equator
- Individuals' occupational classifications and labels thereof

→ Combine data from different sources into various colums of a single DataFrame

Syntax

- `life_exp` holds a Series or DataFrame with data on life expectancy
- `gdp_pc` holds a Series or DataFrame with data on GDP per capita
- Both have the same index, `country` and `year`

```
df = pd.merge(left=life_exp, right=gdp_pc, left_index=True, right_index=True)
```

Syntax

- `life_exp` holds a DataFrame with `country`, `year`, `lifeExp`
- `gdp_pc` holds a DataFrame with `country`, `year`, `gdpPercap`
- Both indices do not matter and will be discarded

```
df = pd.merge(
    left=life_exp,
    right=gdp_pc,
    left_on=["country", "year"],
    right_on=["country", "year"],
)
```

```
df = pd.merge(
    left=life_exp,
    right=gdp_pc,
    on=["country", "year"],
)
```

What if not all data are present?

country	year	lifeExp
Cuba	2002	77.158
Cuba	2007	78.273
Spain	2002	79.78

country	year	gdpPercap
Cuba	2007	8948.1
Spain	2002	24835.5
Spain	2007	28821.1

```
df = pd.merge(
    left=life_exp,
    right=gdp_pc,
    on=["country", "year"],
    how="outer",
)
```

country	year	lifeExp	gdpPercap
Cuba	2002	77.158	
Cuba	2007	78.273	8948.1
Spain	2002	79.78	24835.5
Spain	2007		28821.1

The how keyword to merging

- `how="inner"` : only rows with matching keys in both DataFrames
- `how="left"` : all rows from the left DataFrame
- `how="right"` : all rows from the right DataFrame
- `how="outer"` : all rows from both DataFrames

```
df = pd.merge(
    left=life_exp,
    right=cap,
    left_index=True,
    right_index=True,
)
```

Types of merges

"Index" refers to what the index should be

- 1:1 — index levels of both DataFrames are the same
- m:1 — The left DataFrame’s index levels are a strict superset of the right DataFrame’s index levels
- 1:m — The right DataFrame’s index levels are a strict superset of the left DataFrame’s index levels
- m:m — Both DataFrames overlap in a strict subset of both index levels