Pandas dropna

SHARP SIGHT

What you'll learn

- How to use the Pandas `dropna()` method

- How to drop rows with missing values

Pandas dropna overview

Dropna deletes rows of a DataFrame that have missing values

- Data cleaning is one of the most important tasks of data science
- Handling missing values is one of the most common data cleaning issues
- The `dropna()` method enables you delete rows with missing values
- Several ways to use `dropna()`
  - delete rows with *any* missing values
  - delete rows where all of the data is missing
  - delete rows where data is missing for specific columns

Example: You have a pandas dataframe

name	region	sales	expenses
William	East	50000	42000
Emma	NaN	52000	43000
Sofia	East	90000	NaN
Markus	South	NaN	44000
Edward	West	42000	38000
Thomas	West	72000	39000
Ethan	South	49000	42000
Olivia	West	NaN	NaN
Arun	West	67000	39000
Anika	East	65000	44000
Paulo	South	67000	45000

The DataFrame has missing values in several different columns

name	region	sales	expenses
William	East	50000	42000
Emma	NaN	52000	43000
Sofia	East	90000	NaN
Markus	South	NaN	44000
Edward	West	42000	38000
Thomas	West	72000	39000
Ethan	South	49000	42000
Olivia	West	NaN	NaN
Arun	West	67000	39000
Anika	East	65000	44000
Paulo	South	67000	45000

You can use dropna() to delete rows with missing data

```
salespeople.dropna()
```

name	region	sales	expenses
William	East	50000	42000
Emma	NaN	52000	43000
Sofia	East	90000	NaN
Markus	South	NaN	44000
Edward	West	42000	38000
Thomas	West	72000	39000
Ethan	South	49000	42000
Olivia	West	NaN	NaN
Arun	West	67000	39000
Anika	East	65000	44000
Paulo	South	67000	45000

You can use dropna() to delete rows with missing data

```
salespeople.dropna()
```

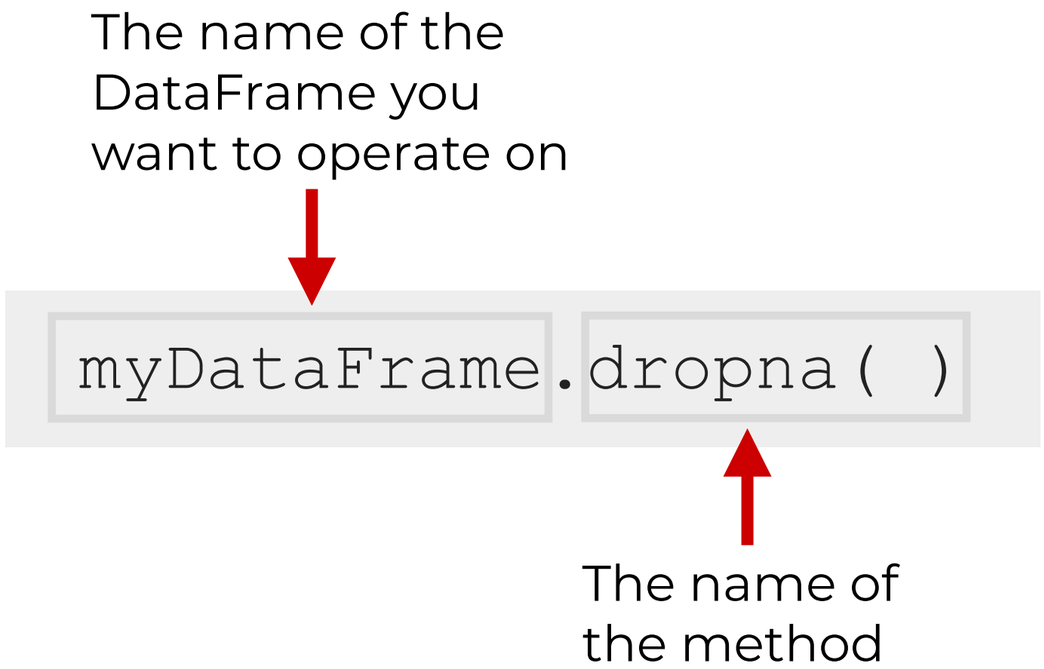
name	region	sales	expenses
William	East	50000	42000
~~Emma~~	~~NaN~~	~~52000~~	~~43000~~
~~Sofia~~	~~East~~	~~90000~~	~~NaN~~
~~Markus~~	~~South~~	~~NaN~~	~~44000~~
Edward	West	42000	38000
Thomas	West	72000	39000
Ethan	South	49000	42000
~~Olivia~~	~~West~~	~~NaN~~	~~NaN~~
Arun	West	67000	39000
Anika	East	65000	44000
Paulo	South	67000	45000

Out:

name	region	sales	expenses
William	East	50000	42000
Edward	West	42000	38000
Thomas	West	72000	39000
Ethan	South	49000	42000
Arun	West	67000	39000
Anika	East	65000	44000
Paulo	South	67000	45000

Pandas dropna syntax

Syntax: pandas dropna



This is a simple version of the syntax ... there are more complex versions that use other parameters

Parameters of Pandas dropna

Parameters of dropna

"How" you want to drop rows ... if any values are missing or if all values are missing (optional)

Whether you want to operate on directly on the DataFrame (optional, default is `False`)

```
myDataFrame.dropna(how =, subset =, inplace = )
```

Which columns you want to look for missing values (optional)

The parameters of Pandas dropna

Parameter	What it does	Format
`how=`	Specifies “how” the method will decide to drop a row from the DataFrame.	Allowed values are `'any'` and `'all'`
`inplace=`	Specify whether you want to directly modify the DataFrame that you’re operating on.	`True` or `False`
`subset=`	Specify the subset of columns where dropna will look for missing values.	A list of column names

The how parameter

- The how parameter enables you to specify “how” the method will decide to drop a row from the DataFrame.

- There are two arguments to the `how` parameter
  - If `how = 'any'`, dropna will drop the row if any of the values in that row are missing.
  - If `how = 'all'`, dropna will drop the row only if all of the values in that row are missing.

The subset parameter

- The `subset` parameter enables you to specify the subset of columns where dropna will look for missing values.

- Example: If you have a DataFrame with 10 variables, you can choose to look for missing values in only 2 of those variables
  - If dropna finds missing values in one of those variables, it will drop the row
  - If dropna finds missing values in other variables, it will keep the row

The inplace parameter

- The `inplace` parameter enables you to specify whether you want to directly modify the DataFrame that you're operating on.

- By default, the dropna method does not directly modify the DataFrame you're working with
  - Default setting is `inplace = False`

- If you set `inplace = True`, the dropna method will modify your DataFrame directly.
  - If you set `inplace = True`, dropna will drop all missing values from your original dataset
  - Be careful!

The output of Pandas dropna

- By default, `dropna` creates a *new* DataFrame
  - That's because by default, `inplace` is set to `inplace = False`

- If you set `inplace = True`, `dropna` will directly modify the original DataFrame

Recap

Recap of what we learned

- You can drop rows of missing data with `dropna`

- The parameters of `dropna` help control the exact behavior
  - `how`
  - `subset`
  - `inplace`

- **Next Steps**: Watch the code walkthrough video for clear examples of Pandas `dropna`