

파이썬 머신러닝 판다스 데이터분석

Lecture (7)



Dr. Heesuk Kim

Part 0. 개발환경 준비

Part 1. 판다스 입문

Part 2. 데이터 입출력

Part 3. 데이터 살펴보기

Part 4. 시각화 도구

Part 5. 데이터 사전처리

Part 6. 데이터프레임의 다양한 응용

Part 7. 머신러닝 데이터 분석



Part 1. 판다스 입문

1. 데이터과학자가 판다스를 배우는 이유
2. 판다스 자료구조
 - 2-1. 시리즈
 - 2-2. 데이터프레임
3. 인덱스 활용
4. 산술 연산
 - 4-1. 시리즈 연산
 - 4-2. 데이터프레임 연산



Part 1. 판다스 입문

3. 인덱스 활용

• 특정 열을 행 인덱스로 설정

: set_index() 메소드를 사용하여 데이터프레임의 특정 열을 행 인덱스로 설정한다. 새로운 객체를 반환한다.

특정 열을 행 인덱스로 설정: DataFrame 객체.set_index(['열 이름'] 또는 '열 이름')



예제 1-16 특정 열을 행 인덱스로 설정

(File: example/part1/1.16_set_index.py)

```
1 # -*- coding: utf-8 -*-
2
3 import pandas as pd
4
5 # DataFrame() 함수로 데이터프레임 변환. 변수 df에 저장
6 exam_data = {'이름' : [ '서준', '우현', '인아'],
7              '수학' : [ 90, 80, 70],
8              '영어' : [ 98, 89, 95],
9              '음악' : [ 85, 95, 100],
10             '체육' : [ 100, 90, 90]}
11 df = pd.DataFrame(exam_data)
12 print(df)
```

```
13 print('\n')
14
15 # 특정 열(column)을 데이터프레임의 행 인덱스(index)로 설정
16 ndf = df.set_index(['이름'])
17 print(ndf)
18 print('\n')
19 ndf2 = ndf.set_index('음악')
20 print(ndf2)
21 print('\n')
22 ndf3 = ndf.set_index(['수학', '음악'])
23 print(ndf3)
```

<실행 결과> 코드 전부 실행

	이름	수학	영어	음악	체육
0	서준	90	98	85	100
1	우현	80	89	95	90
2	인아	70	95	100	90

	수학	영어	음악	체육
이름				
서준	90	98	85	100
우현	80	89	95	90
인아	70	95	100	90

	수학	영어	체육
음악			
85	90	98	100
95	80	89	90
100	70	95	90

	수학	음악	영어	체육
90	85	98	100	
80	95	89	90	
70	100	95	90	

※ 이때, 기존 행 인덱스는 삭제된다.



Spyder (Python 3.7)

File Edit Search Source Run Debug Consoles Projects Tools View Help

Editor - D:\W\>>>2020년_1학기_LECTURE\의료정보처리(1)\source\part1\1,16_set_index.py

```

1
2
3 import pandas as pd
4
5 # DataFrame() 함수로 데이터프레임 변환. 변수 df에 저장
6 exam_data = {'이름' : [ '서준', '우현', '인아'],
7              '수학' : [ 90, 80, 70],
8              '영어' : [ 98, 89, 95],
9              '음악' : [ 85, 95, 100],
10             '체육' : [ 100, 90, 90]}
11 df = pd.DataFrame(exam_data)
12 print(df)
13 print('\n')
14
15 # 특정 열(column)을 데이터프레임의 행 인덱스(index)로 설정
16 ndf = df.set_index(['이름'])
17 print(ndf)
18 print(df)
19 print('\n')
20 ndf2 = ndf.set_index('음악')
21 print(ndf2)
22 print('\n')
23 ndf3 = ndf.set_index(['수학', '음악'])
24 print(ndf3)

```

Variable explorer

Name	Type	Size	Value
------	------	------	-------

IPython console

Console 6/A

Python 3.7.3 (default, Apr 24 2019, 15:29:51) [MSC v.1915 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.6.1 -- An enhanced Interactive Python.

In [1]:

IPython console History log

Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 1 Column: 1 Memory: 81 %

Part 1. 판다스 입문

3. 인덱스 활용

• 행 인덱스 재배열

reindex() 메소드를 사용하면, 데이터프레임의 행 인덱스를 새로운 배열로 재지정할 수 있다. 기존 객체를 변경하지 않고, 새로운 데이터프레임 객체를 반환한다.

새로운 배열로 행 인덱스를 재지정: `DataFrame` 객체.`reindex(새로운 인덱스 배열)`

기존 데이터프레임에 존재하지 않는 행 인덱스가 새롭게 추가되는 경우, 그 행의 데이터 값은 NaN 값이 입력된다.

예제 1-17

새로운 배열로 행 인덱스 재지정

(File: example/part1/1.17_reindex.py)

```
1 # -*- coding: utf-8 -*-
2
3 import pandas as pd
4
5 # 딕셔너리 정의
6 dict_data = {'c0':[1,2,3], 'c1':[4,5,6], 'c2':[7,8,9], 'c3':[10,11,12], 'c4':[13,14,15]}
7
8 # 딕셔너리를 데이터프레임으로 변환. 인덱스를 [r0, r1, r2]로 지정
9 df = pd.DataFrame(dict_data, index=['r0', 'r1', 'r2'])
10 print(df)
11 print('\n')
12
13 # 인덱스를 [r0, r1, r2, r3, r4]로 재지정
14 new_index = ['r0', 'r1', 'r2', 'r3', 'r4']
15 ndf = df.reindex(new_index)
16 print(ndf)
17 print('\n')
18
```

```
19 # reindex로 발생한 NaN값을 숫자 0으로 채우기
20 new_index = ['r0', 'r1', 'r2', 'r3', 'r4']
21 ndf2 = df.reindex(new_index, fill_value=0)
22 print(ndf2)
```

〈실행 결과〉 코드 전부 실행

	c0	c1	c2	c3	c4
r0	1	4	7	10	13
r1	2	5	8	11	14
r2	3	6	9	12	15

	c0	c1	c2	c3	c4
r0	1.0	4.0	7.0	10.0	13.0
r1	2.0	5.0	8.0	11.0	14.0
r2	3.0	6.0	9.0	12.0	15.0
r3	NaN	NaN	NaN	NaN	NaN
r4	NaN	NaN	NaN	NaN	NaN

	c0	c1	c2	c3	c4
r0	1	4	7	10	13
r1	2	5	8	11	14
r2	3	6	9	12	15
r3	0	0	0	0	0
r4	0	0	0	0	0

예제의 12행에서 새롭게 추가된 'r3', 'r4' 인덱스에 해당하는 모든 열에 대해 NaN 값이 입력된다. 이럴 경우, 데이터가 존재하지 않는다는 뜻의 NaN 대신 유효한 값으로 채우려면 예제의 21행과 같이 fill_value 옵션에 원하는 값(0)을 입력한다.

NaN은 “Not a Number” 라는 뜻이다. 유효한 값이 존재하지 않는 누락 데이터를 말한다.

Spyder (Python 3.7)

File Edit Search Source Run Debug Consoles Projects Tools View Help

Editor - D:\W\>>>2020년_1학기_LECTURE\의료정보처리(1)\source\Wpart1\1.17_reindex.py

1,16_set_index.py* 1,17_reindex.py*

```

1
2
3 import pandas as pd
4
5 # 딕셔너리를 정의
6 dict_data = {'c0':[1,2,3], 'c1':[4,5,6], 'c2':[7,8,9], 'c3':[10,11,12], 'c4':[13,14,15]}
7
8 # 딕셔너리를 데이터프레임으로 변환. 인덱스를 [r0, r1, r2]로 지정
9 df = pd.DataFrame(dict_data, index=['r0', 'r1', 'r2'])
10 print(df)
11 print('\n')
12
13 # 인덱스를 [r0, r1, r2, r3, r4]로 재지정
14 new_index = ['r0', 'r1', 'r2', 'r3', 'r4']
15 ndf = df.reindex(new_index)
16 print(ndf)
17 print('\n')
18
19 # reindex로 발생한 NaN값을 숫자 0으로 채우기
20 new_index = ['r0', 'r1', 'r2', 'r3', 'r4']
21 ndf2 = df.reindex(new_index, fill_value=0)
22 print(ndf2)
23

```

Variable explorer

Name	Type	Size	Value
dict_data	dict	5	{'c0':[1, 2, 3], 'c1':[4, 5, 6], 'c2':[7, 8, 9], 'c3':[10, 11, 12], 'c ...

IPython console

Console 8/A

```

NameError: name 'pd' is not defined

In [3]:
In [3]: df = pd.DataFrame(dict_data, index=['r0', 'r1', 'r2'])
Traceback (most recent call last):

  File "<ipython-input-3-880746f3a9e3>", line 1, in <module>
    df = pd.DataFrame(dict_data, index=['r0', 'r1', 'r2'])

NameError: name 'pd' is not defined

In [4]:
In [4]: import pandas as pd

In [5]:

```

IPython console History log

Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 12 Column: 1 Memory: 84 %

Any Question?

Thank you.

