lec05

August 9, 2021

[1]:
```
from datascience import *
import numpy as np

%matplotlib inline
import matplotlib.pyplot as plots
plots.style.use('fivethirtyeight')
```

0.1 Arrays

[2]:
```
my_array = make_array(1, 2, 3, 4)
```

[3]:
```
my_array
```

[3]:
```
array([1, 2, 3, 4], dtype=int64)
```

[4]:
```
my_array * 2
```

[4]:
```
array([2, 4, 6, 8], dtype=int64)
```

[5]:
```
my_array ** 2
```

[5]:
```
array([ 1,  4,  9, 16], dtype=int64)
```

[6]:
```
my_array + 1
```

[6]:
```
array([2, 3, 4, 5], dtype=int64)
```

[7]:
```
my_array # array is unchanged
```

[7]:
```
array([1, 2, 3, 4], dtype=int64)
```

[8]:
```
len(my_array)
```

[8]:
```
4
```

[9]:
```
sum(my_array)
```

[9]: 10

[10]:
```
sum(my_array) / len(my_array)
```

[10]: 2.5

[11]:
```
np.average(my_array)
```

[11]: 2.5

[12]:
```
another = make_array(70, 60, 90, 80)
```

[13]:
```
my_array + another
```

[13]: array([71, 62, 93, 84], dtype=int64)

[14]:
```
yet_another = make_array(5, 6, 7)
```

[15]:
```
my_array + yet_another
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-15-a4a5e45ad569> in <module>
----> 1 my_array + yet_another

ValueError: operands could not be broadcast together with shapes (4,) (3,)
```

[16]:
```
tunas = make_array('bluefin', 'albacore', 'jim')
tunas
```

[16]: array(['bluefin', 'albacore', 'jim'], dtype='<U8')

[17]:
```
tunas * 4
```

```
---------------------------------------------------------------------------
UFuncTypeError                            Traceback (most recent call last)
<ipython-input-17-c34521c521fa> in <module>
----> 1 tunas * 4

UFuncTypeError: ufunc 'multiply' did not contain a loop with signature matching
 ↪types (dtype('<U8'), dtype('<U8')) -> dtype('<U8')
```

[18]:
```
tunas.item(0) # NOTE: indexing starts at 0!
```

[18]: 'bluefin'

[19]:
```
tunas.item(2)
```

[19]:
```
'jim'
```

[20]:
```
tunas.item(3)
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-20-5031eceaa018> in <module>
----> 1 tunas.item(3)

IndexError: index 3 is out of bounds for axis 0 with size 3
```

0.2 Columns of Tables are Arrays

[21]:
```
nba = Table.read_table('nba_salaries.csv').relabeled(3, 'SALARY')
warriors = nba.where('TEAM', 'Golden State Warriors')
```

[22]:
```
warriors
```

[22]:

PLAYER	POSITION	TEAM	SALARY
Klay Thompson	SG	Golden State Warriors	15.501
Draymond Green	PF	Golden State Warriors	14.2609
Andrew Bogut	C	Golden State Warriors	13.8
Andre Iguodala	SF	Golden State Warriors	11.7105
Stephen Curry	PG	Golden State Warriors	11.3708
Jason Thompson	PF	Golden State Warriors	7.00847
Shaun Livingston	PG	Golden State Warriors	5.54373
Harrison Barnes	SF	Golden State Warriors	3.8734
Marreese Speights	C	Golden State Warriors	3.815
Leandro Barbosa	SG	Golden State Warriors	2.5

… (4 rows omitted)

[23]:
```
warriors.select('SALARY')
```

[23]:

SALARY
15.501
14.2609
13.8
11.7105
11.3708
7.00847
5.54373
3.8734
3.815
2.5

```
… (4 rows omitted)
```

[24]:
```
warriors.column('SALARY')
```

[24]:
```
array([15.501    , 14.26087 , 13.8      , 11.710456, 11.370786,  7.008475,
        5.543725,  3.873398,  3.815    ,  2.5      ,  2.008748,  1.270964,
        1.13196 ,  0.289755])
```

[25]:
```
np.average(warriors.column('SALARY'))
```

[25]:
```
6.72036692857143
```

[26]:
```
raptors = nba.where('TEAM', 'Toronto Raptors')
```

[27]:
```
np.average(warriors.column('SALARY')) - np.average(raptors.column('SALARY'))
```

[27]:
```
2.3278598697479005
```

0.3 Ranges

[28]:
```
make_array(0, 1, 2, 3, 4, 5, 6)
```

[28]:
```
array([0, 1, 2, 3, 4, 5, 6], dtype=int64)
```

[29]:
```
np.arange(7)
```

[29]:
```
array([0, 1, 2, 3, 4, 5, 6])
```

[30]:
```
np.arange(5, 11)
```

[30]:
```
array([ 5,  6,  7,  8,  9, 10])
```

[31]:
```
np.arange(0, 20, 2)
```

[31]:
```
array([ 0,  2,  4,  6,  8, 10, 12, 14, 16, 18])
```

[32]:
```
np.arange(0, 21, 2)
```

[32]:
```
array([ 0,  2,  4,  6,  8, 10, 12, 14, 16, 18, 20])
```

[33]:
```
np.arange(0, 1, 0.1)
```

[33]:
```
array([0. , 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9])
```

1 Ways to Create a Table

1.1 Creating a Table from Scratch

[34]:
```
streets = make_array('Bancroft', 'Durant', 'Channing', 'Haste')
streets
```

[34]:
```
array(['Bancroft', 'Durant', 'Channing', 'Haste'], dtype='<U8')
```

[35]:
```
Table()
```

[35]:

[36]:
```
southside = Table().with_column('Streets', streets)
southside
```

[36]:
```
Streets
Bancroft
Durant
Channing
Haste
```

[37]:
```
southside.with_column('Blocks from campus', np.arange(4))
```

[37]:
```
Streets  | Blocks from campus
Bancroft | 0
Durant   | 1
Channing | 2
Haste    | 3
```

[38]:
```
southside
```

[38]:
```
Streets
Bancroft
Durant
Channing
Haste
```

[39]:
```
southside = southside.with_column('Blocks from campus', np.arange(4))
southside
```

[39]:
```
Streets  | Blocks from campus
Bancroft | 0
Durant   | 1
Channing | 2
Haste    | 3
```

[40]:
```
Table().with_columns(
    'Streets', streets,
    'Blocks from campus', np.arange(4)
)
```

[40]:
```
Streets  | Blocks from campus
Bancroft | 0
Durant   | 1
Channing | 2
Haste    | 3
```

1.2 Reading a Table from a File

[41]:
```
du_bois = Table.read_table('du_bois.csv')
du_bois
```

[41]:
```
CLASS         | ACTUAL AVERAGE | RENT | FOOD | CLOTHES | TAXES | OTHER | STATUS
100-200       | 139.1          | 0.19 | 0.43 | 0.28    | 0.001 | 0.099 | POOR
200-300       | 249.45         | 0.22 | 0.47 | 0.23    | 0.04  | 0.04  | POOR
300-400       | 335.66         | 0.23 | 0.43 | 0.18    | 0.045 | 0.115 | FAIR
400-500       | 433.82         | 0.18 | 0.37 | 0.15    | 0.055 | 0.245 | FAIR
500-750       | 547            | 0.13 | 0.31 | 0.17    | 0.05  | 0.34  |
COMFORTABLE
750-1000      | 880            | 0    | 0.37 | 0.19    | 0.08  | 0.36  |
COMFORTABLE
1000 and over | 1125           | 0    | 0.29 | 0.16    | 0.045 | 0.505 | WELL-
TO-DO
```

[42]:
```
du_bois.column('ACTUAL AVERAGE')
```

[42]:
```
array([ 139.1 ,  249.45,  335.66,  433.82,  547.  ,  880.  , 1125.  ])
```

[43]:
```
du_bois.column('FOOD')
```

[43]:
```
array([0.43, 0.47, 0.43, 0.37, 0.31, 0.37, 0.29])
```

[44]:
```
du_bois.column('ACTUAL AVERAGE') * du_bois.column('FOOD')
```

[44]:
```
array([ 59.813 , 117.2415, 144.3338, 160.5134, 169.57  , 325.6   ,
       326.25  ])
```

[45]:
```
food_dollars = du_bois.column('ACTUAL AVERAGE') * du_bois.column('FOOD')
du_bois.with_columns('Food $', food_dollars)
```

[45]:
```
CLASS         | ACTUAL AVERAGE | RENT | FOOD | CLOTHES | TAXES | OTHER | STATUS
| Food $
100-200       | 139.1          | 0.19 | 0.43 | 0.28    | 0.001 | 0.099 | POOR
```

```
| 59.813
200-300       | 249.45          | 0.22 | 0.47 | 0.23    | 0.04  | 0.04  | POOR
| 117.241
300-400       | 335.66          | 0.23 | 0.43 | 0.18    | 0.045 | 0.115 | FAIR
| 144.334
400-500       | 433.82          | 0.18 | 0.37 | 0.15    | 0.055 | 0.245 | FAIR
| 160.513
500-750       | 547             | 0.13 | 0.31 | 0.17    | 0.05  | 0.34  |
COMFORTABLE | 169.57
750-1000      | 880             | 0    | 0.37 | 0.19    | 0.08  | 0.36  |
COMFORTABLE | 325.6
1000 and over | 1125            | 0    | 0.29 | 0.16    | 0.045 | 0.505 | WELL-
TO-DO  | 326.25
```

[46]:
```
du_bois.select('CLASS', 'ACTUAL AVERAGE', 'FOOD', 'Food $')
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-46-7b1cd563197a> in <module>
----> 1 du_bois.select('CLASS', 'ACTUAL AVERAGE', 'FOOD', 'Food $')

~\Anaconda3\lib\site-packages\datascience\tables.py in select(self,␣
↪*column_or_columns)
   1182         table = type(self)()
   1183         for label in labels:
-> 1184             self._add_column_and_format(table, label, np.
↪copy(self[label]))
   1185         return table
   1186

~\Anaconda3\lib\site-packages\datascience\tables.py in __getitem__(self,␣
↪index_or_label)
    196
    197     def __getitem__(self, index_or_label):
--> 198         return self.column(index_or_label)
    199
    200     def __setitem__(self, index_or_label, values):

~\Anaconda3\lib\site-packages\datascience\tables.py in column(self,␣
↪index_or_label)
    363         if (isinstance(index_or_label, str)
    364                 and index_or_label not in self.labels):
--> 365             raise ValueError(
    366                 'The column "{}" is not in the table. The table contain␣
↪'
    367                 'these columns: {}'
```

```
ValueError: The column "Food $" is not in the table. The table contains these␣
↪columns: CLASS, ACTUAL AVERAGE, RENT, FOOD, CLOTHES, TAXES, OTHER, STATUS
```

[47]:
```
food_dollars = du_bois.column('ACTUAL AVERAGE') * du_bois.column('FOOD')

du_bois = du_bois.with_columns('Food $', food_dollars)

du_bois
```

[47]:
```
CLASS         | ACTUAL AVERAGE | RENT | FOOD | CLOTHES | TAXES | OTHER | STATUS
| Food $
100-200       | 139.1          | 0.19 | 0.43 | 0.28    | 0.001 | 0.099 | POOR
| 59.813
200-300       | 249.45         | 0.22 | 0.47 | 0.23    | 0.04  | 0.04  | POOR
| 117.241
300-400       | 335.66         | 0.23 | 0.43 | 0.18    | 0.045 | 0.115 | FAIR
| 144.334
400-500       | 433.82         | 0.18 | 0.37 | 0.15    | 0.055 | 0.245 | FAIR
| 160.513
500-750       | 547            | 0.13 | 0.31 | 0.17    | 0.05  | 0.34  |
COMFORTABLE | 169.57
750-1000      | 880            | 0    | 0.37 | 0.19    | 0.08  | 0.36  |
COMFORTABLE | 325.6
1000 and over | 1125           | 0    | 0.29 | 0.16    | 0.045 | 0.505 | WELL-
TO-DO  | 326.25
```

[48]:
```
du_bois.select('CLASS', 'ACTUAL AVERAGE', 'FOOD', 'Food $')
```

[48]:
```
CLASS         | ACTUAL AVERAGE | FOOD | Food $
100-200       | 139.1          | 0.43 | 59.813
200-300       | 249.45         | 0.47 | 117.241
300-400       | 335.66         | 0.43 | 144.334
400-500       | 433.82         | 0.37 | 160.513
500-750       | 547            | 0.31 | 169.57
750-1000      | 880            | 0.37 | 325.6
1000 and over | 1125           | 0.29 | 326.25
```

[49]:
```
du_bois.labels
```

[49]:
```
('CLASS',
 'ACTUAL AVERAGE',
 'RENT',
 'FOOD',
 'CLOTHES',
 'TAXES',
 'OTHER',
 'STATUS',
```

```
       'Food $')
```

[50]:
```
du_bois.num_rows
```

[50]:
```
7
```

[51]:
```
du_bois.num_columns
```

[51]:
```
9
```

[ ]:
```

```