Lab1

January 21, 2021

1 Experiment 1: Introduction To Pandas And Introduction To Numpy

1.1 Introduction To Numpy

[1]:
```
import numpy as np
```

[2]:
```
a = np.arange(15).reshape(3, 5)
```

[3]:
```
a.shape
```

[3]:
```
(3, 5)
```

[4]:
```
a.ndim
```

[4]:
```
2
```

[5]:
```
a.dtype.name
```

[5]:
```
'int64'
```

[6]:
```
a.itemsize
```

[6]:
```
8
```

[7]:
```
a.size
```

[7]:
```
15
```

[8]:
```
type(a)
```

[8]:
```
numpy.ndarray
```

[9]:
```
b = np.array([6, 7, 8])
```

[10]:
```
b
```

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

[11]:
```
type(b)
```

[11]:
```
numpy.ndarray
```

[12]:
```
b.dtype
```

[12]:
```
dtype('int64')
```

[13]:
```
c = np.array([[1, 2], [3, 4]], dtype=complex)
```

[14]:
```
c
```

[14]:
```
array([[1.+0.j, 2.+0.j],
       [3.+0.j, 4.+0.j]])
```

[15]:
```
c.dtype.name
```

[15]:
```
'complex128'
```

[16]:
```
np.zeros((3, 4))
```

[16]:
```
array([[0., 0., 0., 0.],
       [0., 0., 0., 0.],
       [0., 0., 0., 0.]])
```

[17]:
```
np.ones((2, 3, 4), dtype=np.int16)
```

[17]:
```
array([[[1, 1, 1, 1],
        [1, 1, 1, 1],
        [1, 1, 1, 1]],

       [[1, 1, 1, 1],
        [1, 1, 1, 1],
        [1, 1, 1, 1]]], dtype=int16)
```

[18]:
```
np.empty((2, 3))
```

[18]:
```
array([[4.63826776e-310, 0.00000000e+000, 0.00000000e+000],
       [0.00000000e+000, 0.00000000e+000, 0.00000000e+000]])
```

[19]:
```
np.eye(7)
```

[19]:
```
array([[1., 0., 0., 0., 0., 0., 0.],
       [0., 1., 0., 0., 0., 0., 0.],
       [0., 0., 1., 0., 0., 0., 0.],
       [0., 0., 0., 1., 0., 0., 0.],
       [0., 0., 0., 0., 1., 0., 0.],
       [0., 0., 0., 0., 0., 1., 0.],
```

```
       [0., 0., 0., 0., 0., 0., 1.]])
```

[20]:
```
np.arange(10, 30, 5)
```

[20]:
```
array([10, 15, 20, 25])
```

[21]:
```
np.arange(0, 2, 0.3)
```

[21]:
```
array([0. , 0.3, 0.6, 0.9, 1.2, 1.5, 1.8])
```

[22]:
```
np.linspace(0, 2, 9)
```

[22]:
```
array([0.  , 0.25, 0.5 , 0.75, 1.  , 1.25, 1.5 , 1.75, 2.  ])
```

[23]:
```
np.arange(0, 11, 1)**2
```

[23]:
```
array([  0,   1,   4,   9,  16,  25,  36,  49,  64,  81, 100])
```

[24]:
```
print(a)
```

```
[[ 0  1  2  3  4]
 [ 5  6  7  8  9]
 [10 11 12 13 14]]
```

[25]:
```
print(a.reshape(5, 3))
```

```
[[ 0  1  2]
 [ 3  4  5]
 [ 6  7  8]
 [ 9 10 11]
 [12 13 14]]
```

[26]:
```
print(np.arange(10000))
```

```
[   0    1    2 … 9997 9998 9999]
```

[27]:
```
A = np.array([[1, 1], [0, 1]])
B = np.array([[2, 0], [3, 4]])
```

[28]:
```
A + B
```

[28]:
```
array([[3, 1],
       [3, 5]])
```

[29]:
```
A - B
```

[29]:
```
array([[-1,  1],
       [-3, -3]])
```

[30]:
```
A * B
```

[30]:
```
array([[2, 0],
       [0, 4]])
```

[31]:
```
A @ B
```

[31]:
```
array([[5, 4],
       [3, 4]])
```

[32]:
```
A / B
```

```
<ipython-input-32-db6a01120809>:1: RuntimeWarning: divide by zero encountered in
true_divide
  A / B
```

[32]:
```
array([[0.5 ,  inf],
       [0.  , 0.25]])
```

[33]:
```
np.sin(np.arange(0, 2 * np.pi, np.pi / 6))
```

[33]:
```
array([ 0.00000000e+00,  5.00000000e-01,  8.66025404e-01,  1.00000000e+00,
        8.66025404e-01,  5.00000000e-01,  1.22464680e-16, -5.00000000e-01,
       -8.66025404e-01, -1.00000000e+00, -8.66025404e-01, -5.00000000e-01])
```

[34]:
```
A.dot(B)
```

[34]:
```
array([[5, 4],
       [3, 4]])
```

[35]:
```
a[:6:2]
```

[35]:
```
array([[ 0,  1,  2,  3,  4],
       [10, 11, 12, 13, 14]])
```

[36]:
```
a[::-1]
```

[36]:
```
array([[10, 11, 12, 13, 14],
       [ 5,  6,  7,  8,  9],
       [ 0,  1,  2,  3,  4]])
```

[37]:
```
a.T
```

[37]:
```
array([[ 0,  5, 10],
       [ 1,  6, 11],
       [ 2,  7, 12],
       [ 3,  8, 13],
       [ 4,  9, 14]])
```

[38]:
```
A.trace()
```

[38]:
```
2
```

[39]:
```
A[0][[False, True]]
```

[39]:
```
array([1])
```

1.2 Introduction To Pandas

[40]:
```
import pandas as pd
```

[41]:
```
s = pd.Series([1,3, 5, np.nan, 6, 8])
```

[42]:
```
s
```

[42]:
```
0    1.0
1    3.0
2    5.0
3    NaN
4    6.0
5    8.0
dtype: float64
```

[43]:
```
dates = pd.date_range('20130101', periods=6)
```

[44]:
```
dates
```

[44]:
```
DatetimeIndex(['2013-01-01', '2013-01-02', '2013-01-03', '2013-01-04',
               '2013-01-05', '2013-01-06'],
              dtype='datetime64[ns]', freq='D')
```

[45]:
```
df = pd.DataFrame(np.random.randn(6, 4), index=dates, columns=list('ABCD'))
```

[46]:
```
df
```

[46]:

	A	B	C	D
2013-01-01	0.614281	0.659087	-0.158752	0.862910
2013-01-02	0.594038	1.856501	0.435140	-0.448650
2013-01-03	-1.679148	-0.643518	-0.533022	-1.074692
2013-01-04	-0.576276	1.650922	-0.068905	-0.995353
2013-01-05	0.680622	-2.294979	-0.142549	-0.152492
2013-01-06	-1.296938	-0.858845	-0.037390	-1.258893

[47]:
```
df2 = pd.DataFrame({
    'A' : 1.0,
    'B' : pd.Timestamp('20130102'),
    'C' : pd.Series(1, index=list(range(4)), dtype='float32'),
    'D' : np.array([3] * 4, dtype='int32'),
    'E' : pd.Categorical(['test', 'train', 'test', 'train']),
```

```
    'F' : 'foo'
})
```

[48]:
```
df2
```

[48]:
```
     A          B    C  D      E    F
0  1.0 2013-01-02  1.0  3   test  foo
1  1.0 2013-01-02  1.0  3  train  foo
2  1.0 2013-01-02  1.0  3   test  foo
3  1.0 2013-01-02  1.0  3  train  foo
```

[49]:
```
df2.dtypes
```

[49]:
```
A           float64
B    datetime64[ns]
C           float32
D             int32
E          category
F            object
dtype: object
```

[50]:
```
df.head()
```

[50]:
```
                   A         B         C         D
2013-01-01  0.614281  0.659087 -0.158752  0.862910
2013-01-02  0.594038  1.856501  0.435140 -0.448650
2013-01-03 -1.679148 -0.643518 -0.533022 -1.074692
2013-01-04 -0.576276  1.650922 -0.068905 -0.995353
2013-01-05  0.680622 -2.294979 -0.142549 -0.152492
```

[51]:
```
df.tail(3)
```

[51]:
```
                   A         B         C         D
2013-01-04 -0.576276  1.650922 -0.068905 -0.995353
2013-01-05  0.680622 -2.294979 -0.142549 -0.152492
2013-01-06 -1.296938 -0.858845 -0.037390 -1.258893
```

[52]:
```
df.index
```

[52]:
```
DatetimeIndex(['2013-01-01', '2013-01-02', '2013-01-03', '2013-01-04',
               '2013-01-05', '2013-01-06'],
              dtype='datetime64[ns]', freq='D')
```

[53]:
```
df.columns
```

[53]:
```
Index(['A', 'B', 'C', 'D'], dtype='object')
```

[54]:
```
df.to_numpy()
```

[54]:
```
array([[ 0.6142808 ,  0.6590874 , -0.15875225,  0.86291032],
       [ 0.59403769,  1.85650126,  0.43513976, -0.44865025],
       [-1.67914806, -0.64351822, -0.53302172, -1.07469183],
       [-0.57627615,  1.65092243, -0.06890529, -0.99535347],
       [ 0.68062168, -2.29497859, -0.14254923, -0.15249244],
       [-1.29693779, -0.85884469, -0.03739002, -1.25889255]])
```

[55]:
```
df2.to_numpy()
```

[55]:
```
array([[1.0, Timestamp('2013-01-02 00:00:00'), 1.0, 3, 'test', 'foo'],
       [1.0, Timestamp('2013-01-02 00:00:00'), 1.0, 3, 'train', 'foo'],
       [1.0, Timestamp('2013-01-02 00:00:00'), 1.0, 3, 'test', 'foo'],
       [1.0, Timestamp('2013-01-02 00:00:00'), 1.0, 3, 'train', 'foo']],
      dtype=object)
```

[56]:
```
df.describe()
```

[56]:

	A	B	C	D
count	6.000000	6.000000	6.000000	6.000000
mean	-0.277237	0.061528	-0.084246	-0.511195
std	1.055082	1.612605	0.310658	0.791421
min	-1.679148	-2.294979	-0.533022	-1.258893
25%	-1.116772	-0.805013	-0.154701	-1.054857
50%	0.008881	0.007785	-0.105727	-0.722002
75%	0.609220	1.402964	-0.045269	-0.226532
max	0.680622	1.856501	0.435140	0.862910

[57]:
```
df.T
```

[57]:

	2013-01-01	2013-01-02	2013-01-03	2013-01-04	2013-01-05	2013-01-06
A	0.614281	0.594038	-1.679148	-0.576276	0.680622	-1.296938
B	0.659087	1.856501	-0.643518	1.650922	-2.294979	-0.858845
C	-0.158752	0.435140	-0.533022	-0.068905	-0.142549	-0.037390
D	0.862910	-0.448650	-1.074692	-0.995353	-0.152492	-1.258893

[58]:
```
df.sort_index(axis=1, ascending=False)
```

[58]:

	D	C	B	A
2013-01-01	0.862910	-0.158752	0.659087	0.614281
2013-01-02	-0.448650	0.435140	1.856501	0.594038
2013-01-03	-1.074692	-0.533022	-0.643518	-1.679148
2013-01-04	-0.995353	-0.068905	1.650922	-0.576276
2013-01-05	-0.152492	-0.142549	-2.294979	0.680622
2013-01-06	-1.258893	-0.037390	-0.858845	-1.296938

[59]:
```
df.sort_values(by='B')
```

[59]:
```
                   A         B         C         D
2013-01-05  0.680622 -2.294979 -0.142549 -0.152492
2013-01-06 -1.296938 -0.858845 -0.037390 -1.258893
2013-01-03 -1.679148 -0.643518 -0.533022 -1.074692
2013-01-01  0.614281  0.659087 -0.158752  0.862910
2013-01-04 -0.576276  1.650922 -0.068905 -0.995353
2013-01-02  0.594038  1.856501  0.435140 -0.448650
```

[60]:
```
df['A']
```

[60]:
```
2013-01-01    0.614281
2013-01-02    0.594038
2013-01-03   -1.679148
2013-01-04   -0.576276
2013-01-05    0.680622
2013-01-06   -1.296938
Freq: D, Name: A, dtype: float64
```

[61]:
```
df[0:3]
```

[61]:
```
                   A         B         C         D
2013-01-01  0.614281  0.659087 -0.158752  0.862910
2013-01-02  0.594038  1.856501  0.435140 -0.448650
2013-01-03 -1.679148 -0.643518 -0.533022 -1.074692
```

[62]:
```
df['20130102':'20130104']
```

[62]:
```
                   A         B         C         D
2013-01-02  0.594038  1.856501  0.435140 -0.448650
2013-01-03 -1.679148 -0.643518 -0.533022 -1.074692
2013-01-04 -0.576276  1.650922 -0.068905 -0.995353
```

[63]:
```
df.loc[dates[0]]
```

[63]:
```
A    0.614281
B    0.659087
C   -0.158752
D    0.862910
Name: 2013-01-01 00:00:00, dtype: float64
```

[64]:
```
df.loc[:, ["A", "B"]]
```

[64]:
```
                   A         B
2013-01-01  0.614281  0.659087
2013-01-02  0.594038  1.856501
2013-01-03 -1.679148 -0.643518
2013-01-04 -0.576276  1.650922
2013-01-05  0.680622 -2.294979
```

```
2013-01-06 -1.296938 -0.858845
```

[65]:
```
df.at[dates[0], 'A']
```

[65]:
```
0.6142807961660794
```

[66]:
```
df.iloc[3]
```

[66]:
```
A   -0.576276
B    1.650922
C   -0.068905
D   -0.995353
Name: 2013-01-04 00:00:00, dtype: float64
```

[67]:
```
df[df['A'] > 0]
```

[67]:
```
                   A         B         C         D
2013-01-01  0.614281  0.659087 -0.158752  0.862910
2013-01-02  0.594038  1.856501  0.435140 -0.448650
2013-01-05  0.680622 -2.294979 -0.142549 -0.152492
```

[68]:
```
s1 = pd.Series([1, 2, 3, 4, 5, 6], index=pd.date_range("20130102", periods=6))
```

[69]:
```
s1
```

[69]:
```
2013-01-02    1
2013-01-03    2
2013-01-04    3
2013-01-05    4
2013-01-06    5
2013-01-07    6
Freq: D, dtype: int64
```

[70]:
```
df['F'] = s1
```

[71]:
```
df
```

[71]:
```
                   A         B         C         D    F
2013-01-01  0.614281  0.659087 -0.158752  0.862910  NaN
2013-01-02  0.594038  1.856501  0.435140 -0.448650  1.0
2013-01-03 -1.679148 -0.643518 -0.533022 -1.074692  2.0
2013-01-04 -0.576276  1.650922 -0.068905 -0.995353  3.0
2013-01-05  0.680622 -2.294979 -0.142549 -0.152492  4.0
2013-01-06 -1.296938 -0.858845 -0.037390 -1.258893  5.0
```

[72]:
```
df.fillna(value=5)
```

[72]:
```
                   A         B         C         D    F
2013-01-01  0.614281  0.659087 -0.158752  0.862910  5.0
2013-01-02  0.594038  1.856501  0.435140 -0.448650  1.0
2013-01-03 -1.679148 -0.643518 -0.533022 -1.074692  2.0
2013-01-04 -0.576276  1.650922 -0.068905 -0.995353  3.0
2013-01-05  0.680622 -2.294979 -0.142549 -0.152492  4.0
2013-01-06 -1.296938 -0.858845 -0.037390 -1.258893  5.0
```

[73]:
```
pd.isna(df)
```

[73]:
```
                A      B      C      D      F
2013-01-01  False  False  False  False   True
2013-01-02  False  False  False  False  False
2013-01-03  False  False  False  False  False
2013-01-04  False  False  False  False  False
2013-01-05  False  False  False  False  False
2013-01-06  False  False  False  False  False
```

[74]:
```
df.mean()
```

[74]:
```
A   -0.277237
B    0.061528
C   -0.084246
D   -0.511195
F    3.000000
dtype: float64
```

[75]:
```
df.mean(axis=1)
```

[75]:
```
2013-01-01    0.494382
2013-01-02    0.687406
2013-01-03   -0.386076
2013-01-04    0.602078
2013-01-05    0.418120
2013-01-06    0.309587
Freq: D, dtype: float64
```

[76]:
```
df.apply(np.cumsum)
```

[76]:
```
                   A         B         C         D     F
2013-01-01  0.614281  0.659087 -0.158752  0.862910   NaN
2013-01-02  1.208318  2.515589  0.276388  0.414260   1.0
2013-01-03 -0.470830  1.872070 -0.256634 -0.660432   3.0
2013-01-04 -1.047106  3.522993 -0.325540 -1.655785   6.0
2013-01-05 -0.366484  1.228014 -0.468089 -1.808278  10.0
2013-01-06 -1.663422  0.369170 -0.505479 -3.067170  15.0
```

[77]:
```
df.apply(lambda x: x.max() - x.min())
```

[77]:
```
A    2.359770
B    4.151480
C    0.968161
D    2.121803
F    4.000000
dtype: float64
```

[78]:
```
s = pd.Series(["A", "B", "C", "Aaba", "Baca", np.nan, "CABA", "dog", "cat"])
```

[79]:
```
s.str.lower()
```

[79]:
```
0       a
1       b
2       c
3    aaba
4    baca
5     NaN
6    caba
7     dog
8     cat
dtype: object
```

[80]:
```
df = pd.DataFrame(np.random.randn(10, 4))
```

[81]:
```
df
```

[81]:

	0	1	2	3
0	-1.285396	-0.799128	0.133516	-0.378753
1	0.255040	-1.097208	-0.008231	1.413290
2	-0.207827	-0.364421	-0.081548	1.193606
3	0.994157	1.710302	0.605819	0.562993
4	0.240294	0.653899	0.218709	1.077028
5	1.070931	0.970839	0.874105	0.330688
6	0.036431	0.814720	0.932403	-2.913828
7	0.072013	0.755732	1.262736	-0.515840
8	0.119049	0.308180	-0.505441	-1.294223
9	1.891651	1.277985	0.210752	-1.625409

[82]:
```
pieces = [df[:3], df[3:7], df[7:]]
```

[83]:
```
pieces
```

[83]:

	0	1	2	3
0	-1.285396	-0.799128	0.133516	-0.378753
1	0.255040	-1.097208	-0.008231	1.413290
2	-0.207827	-0.364421	-0.081548	1.193606,

	0	1	2	3
3	0.994157	1.710302	0.605819	0.562993

```
4  0.240294  0.653899  0.218709  1.077028
5  1.070931  0.970839  0.874105  0.330688
6  0.036431  0.814720  0.932403 -2.913828,
          0         1         2         3
7  0.072013  0.755732  1.262736 -0.515840
8  0.119049  0.308180 -0.505441 -1.294223
9  1.891651  1.277985  0.210752 -1.625409]
```

[84]:
```
pd.concat(pieces)
```

[84]:
```
          0         1         2         3
0 -1.285396 -0.799128  0.133516 -0.378753
1  0.255040 -1.097208 -0.008231  1.413290
2 -0.207827 -0.364421 -0.081548  1.193606
3  0.994157  1.710302  0.605819  0.562993
4  0.240294  0.653899  0.218709  1.077028
5  1.070931  0.970839  0.874105  0.330688
6  0.036431  0.814720  0.932403 -2.913828
7  0.072013  0.755732  1.262736 -0.515840
8  0.119049  0.308180 -0.505441 -1.294223
9  1.891651  1.277985  0.210752 -1.625409
```

[85]:
```
left = pd.DataFrame({"key": ["foo", "foo"], "lval": [1, 2]})
```

[86]:
```
right = pd.DataFrame({"key": ["foo", "foo"], "rval": [4, 5]})
```

[87]:
```
pd.merge(left, right, on="key")
```

[87]:
```
   key  lval  rval
0  foo     1     4
1  foo     1     5
2  foo     2     4
3  foo     2     5
```

[88]:
```
df.groupby(1).sum()
```

[88]:
```
                  0         2         3
1
-1.097208  0.255040 -0.008231  1.413290
-0.799128 -1.285396  0.133516 -0.378753
-0.364421 -0.207827 -0.081548  1.193606
 0.308180  0.119049 -0.505441 -1.294223
 0.653899  0.240294  0.218709  1.077028
 0.755732  0.072013  1.262736 -0.515840
 0.814720  0.036431  0.932403 -2.913828
 0.970839  1.070931  0.874105  0.330688
 1.277985  1.891651  0.210752 -1.625409
 1.710302  0.994157  0.605819  0.562993
```

[89]:
```
df.sort_values(by=1)
```

[89]:

	0	1	2	3
1	0.255040	-1.097208	-0.008231	1.413290
0	-1.285396	-0.799128	0.133516	-0.378753
2	-0.207827	-0.364421	-0.081548	1.193606
8	0.119049	0.308180	-0.505441	-1.294223
4	0.240294	0.653899	0.218709	1.077028
7	0.072013	0.755732	1.262736	-0.515840
6	0.036431	0.814720	0.932403	-2.913828
5	1.070931	0.970839	0.874105	0.330688
9	1.891651	1.277985	0.210752	-1.625409
3	0.994157	1.710302	0.605819	0.562993

[ ]:
```

```