February 23, 2017

FINISHED

```
%pyspark
from pandas import Series, DataFrame
import numpy as np, pandas as pd
df = DataFrame([[1.4,np.nan],[7.1,-4.5],
                [np.nan,np.nan],[0.75,-1.3]],
                index=['a','b','c','d'],
                columns=['one','two'])

df
```

```
    one  two
a  1.40  NaN
b  7.10 -4.5
c   NaN  NaN
d  0.75 -1.3
```

Took 0 sec. Last updated by anonymous at February 23 2017, 7:22:13 PM. (outdated)

FINISHED

```
%pyspark

df.sum()
```

```
one    9.25
two   -5.80
dtype: float64
```

Took 0 sec. Last updated by anonymous at February 23 2017, 7:24:24 PM.

FINISHED

```
%pyspark

df.sum(axis=1)
```

```
a    1.40
b    2.60
c    0.00
d   -0.55
dtype: float64
```

Took 0 sec. Last updated by anonymous at February 23 2017, 7:24:41 PM.

FINISHED

```
%pyspark

df.mean(axis=1,skipna=False)
```

```
a      NaN
b    1.300
c      NaN
d   -0.275
dtype: float64
```

Took 0 sec. Last updated by anonymous at February 23 2017, 7:24:54 PM.

```
%pyspark

df.idxmax()
```

FINISHED

```
one    b
two    d
dtype: object
```

Took 0 sec. Last updated by anonymous at February 23 2017, 7:25:13 PM.

```
%pyspark

df.describe()
```

FINISHED

```
/Users/Shared/anaconda/lib/python3.5/site-packages/numpy/lib/function_base.py:3834: RuntimeWar
ning: Invalid value encountered in percentile
  RuntimeWarning)
            one       two
count  3.000000  2.000000
mean   3.083333 -2.900000
std    3.493685  2.262742
min    0.750000 -4.500000
25%         NaN       NaN
50%         NaN       NaN
75%         NaN       NaN
max    7.100000 -1.300000
```

Took 0 sec. Last updated by anonymous at February 23 2017, 7:25:26 PM.

```
%pyspark

obj = Series(['a','a','b','c'] * 4)
```

FINISHED

Took 0 sec. Last updated by anonymous at February 23 2017, 7:25:51 PM.

```
%pyspark

obj
obj.describe()
```

FINISHED

```
count     16
unique     3
top        a
freq       8
dtype: object
```

Took 0 sec. Last updated by anonymous at February 23 2017, 7:26:10 PM.

```
%pyspark

from pandas_datareader import data as web
all_data = {}
for ticker in ['AAPL','IBM','MSFT','GOOG']:
  all_data[ticker] = web.get_data_yahoo(ticker)
price = DataFrame({tic: data['Adj Close']
    for tic, data in all_data.items()})
volume = DataFrame({tic: data['Volume']
    for tic, data in all_data.items()})
```

FINISHED

Took 1 sec. Last updated by anonymous at February 23 2017, 7:26:45 PM.

```
%pyspark

returns = price.pct_change()
returns.tail()
```

FINISHED

```
                AAPL      GOOG       IBM      MSFT
Date
2017-02-15  0.003629 -0.001792  0.008605 -0.000619
2017-02-16 -0.001181  0.006325 -0.001376 -0.000155
2017-02-17  0.002734  0.004744 -0.004189  0.001550
2017-02-21  0.007221  0.004335 -0.002269 -0.002012
2017-02-22  0.002999 -0.001082  0.004937 -0.002016
```

Took 0 sec. Last updated by anonymous at February 23 2017, 7:27:12 PM.

```
%pyspark

returns.MSFT.corr(returns.IBM)
```

FINISHED

```
0.49515377802280919
```

Took 0 sec. Last updated by anonymous at February 23 2017, 7:27:26 PM.

```
%pyspark

returns.MSFT.cov(returns.IBM)
```

FINISHED

```
8.5977652563835427e-05
```

Took 0 sec. Last updated by anonymous at February 23 2017, 7:27:39 PM.

```
%pyspark

returns.corr()
```

FINISHED

```
          AAPL      GOOG       IBM      MSFT
AAPL  1.000000  0.409541  0.381549  0.388972
GOOG  0.409541  1.000000  0.402872  0.470820
IBM   0.381549  0.402872  1.000000  0.495154
MSFT  0.388972  0.470820  0.495154  1.000000
```

Took 0 sec. Last updated by anonymous at February 23 2017, 7:28:06 PM.

```
%pyspark

returns.corrwith(returns.IBM)
```

FINISHED

```
AAPL    0.381549
GOOG    0.402872
IBM     1.000000
MSFT    0.495154
dtype: float64
```

Took 0 sec. Last updated by anonymous at February 23 2017, 7:28:36 PM.

```
%pyspark

returns.corrwith(volume)
```

FINISHED

```
AAPL   -0.074323
GOOG   -0.009670
IBM    -0.194432
MSFT   -0.091017
dtype: float64
```

Took 0 sec. Last updated by anonymous at February 23 2017, 7:28:49 PM.

READY