

## Risk Management HW3

### Jinzhou Yao

```
In [6]: from yahoofinancials import YahooFinancials
import pandas as pd
from pandas_datareader import data
import numpy as np
from math import sqrt, pi, log, e, erf
from scipy.stats import norm
from scipy.stats import multivariate_normal
from scipy.optimize import fsolve
```

### Collect one year daily stock prices for 10 companies.

```
In [30]: tickers = ['MSFT', 'AAPL', 'AMZN', 'VZ', 'XOM', 'TSLA', 'FB', 'IBM', 'GOOG', 'BABA']
start_date = '2018-04-01'
end_date = '2019-04-01'
panel_data = data.DataReader(tickers, 'yahoo', start_date, end_date)
adjclose = panel_data['Adj Close']
adjclose = adjclose.fillna(method='ffill')
print(adjclose.head())
```

Symbols	AAPL	AMZN	BABA	FB	GOOG \
Date					
2018-04-02	164.180008	1371.989990	177.610001	155.389999	1006.469971
2018-04-03	165.864349	1392.050049	174.669998	156.110001	1013.409973
2018-04-04	169.036072	1410.569946	172.070007	155.100006	1025.140015
2018-04-05	170.208206	1451.750000	172.570007	159.339996	1027.810059
2018-04-06	165.854523	1405.229980	167.520004	157.199997	1007.039978

Symbols	IBM	MSFT	TSLA	VZ	XOM
Date					
2018-04-02	143.319534	87.050652	252.479996	44.606892	70.245407
2018-04-03	143.109436	88.220894	267.529999	44.928486	71.972282
2018-04-04	147.187363	90.797409	286.940002	45.335209	71.828377
2018-04-05	147.101395	90.846581	305.720001	45.628426	72.931656
2018-04-06	143.797058	88.732277	299.299988	44.909573	71.828377

### Compute daily returns, and then mean and standard deviation.

```
In [31]: ret =adjclose.diff()/adjclose.shift(1)
ret.drop(ret.index[0], inplace=True)
mu = ret.apply(np.average, axis=0)
sigma = ret.apply(np.std, axis=0)
print("Return:\n",ret.head())
print("Mean:\n",mu)
print("Standard deviation:\n",sigma)
```

Return:

Symbols	AAPL	AMZN	BABA	FB	GOOG	IBM \
Date						
2018-04-03	0.010259	0.014621	-0.016553	0.004634	0.006895	-0.001466
2018-04-04	0.019122	0.013304	-0.014885	-0.006470	0.011575	0.028495
2018-04-05	0.006934	0.029194	0.002906	0.027337	0.002605	-0.000584
2018-04-06	-0.025579	-0.032044	-0.029264	-0.013430	-0.020208	-0.022463
2018-04-09	0.009918	0.000605	0.014028	0.004644	0.008351	0.014080

Symbols	MSFT	TSLA	VZ	XOM
Date				
2018-04-03	0.013443	0.059609	0.007210	0.024583
2018-04-04	0.029205	0.072553	0.009053	-0.001999
2018-04-05	0.000542	0.065449	0.006468	0.015360
2018-04-06	-0.023273	-0.021000	-0.015754	-0.015128
2018-04-09	0.005985	-0.032208	0.000640	0.000000

Mean:

Symbols	
AAPL	0.000844
AMZN	0.001359
BABA	0.000324
FB	0.000747
GOOG	0.000835
IBM	0.000107
MSFT	0.001379
TSLA	0.001184
VZ	0.001115
XOM	0.000665

dtype: float64

Standard deviation:

Symbols	
AAPL	0.018991
AMZN	0.022373
BABA	0.021597
FB	0.023881
GOOG	0.016429
IBM	0.015169
MSFT	0.016223
TSLA	0.037408
VZ	0.012380
XOM	0.012708

dtype: float64

## Collect long-term liabilities of each firm3

```
In [32]: yahoo_financials = YahooFinancials(tickers)
balance_sheet_data = yahoo_financials.get_financial_stmts('annual', 'balance')
balance_sheet_all = balance_sheet_data['balanceSheetHistory']
total_liab = {}
short_liab = {}
equity = {}
for key, value in balance_sheet_all.items():
    a = value[0]
    for k1, v1 in a.items():
        total_liab[key] = v1['totalLiab']
        short_liab[key] = v1['totalCurrentLiabilities']
        equity[key] = v1['totalStockholderEquity']
total = pd.Series(total_liab)
short = pd.Series(short_liab)
long = total - short
equity = pd.Series(equity)
print("Long-term liabilities:\n", long/1000000000)
```

Long-term liabilities:

MSFT	117.642000
AAPL	141.712000
AMZN	50.708000
VZ	172.189000
XOM	90.530000
TSLA	13.433874
FB	6.190000
IBM	68.225000
GOOG	20.544000
BABA	144.876000

dtype: float64

## Sharp Ratio

```
In [33]: sharp = (mu*252 - 0.02325) / (sigma*sqrt(252))
sharp
```

```
Out[33]: Symbols
AAPL      0.628771
AMZN      0.898894
BABA      0.170301
FB         0.435298
GOOG      0.717363
IBM        0.015158
MSFT      1.259302
TSLA      0.463179
VZ         1.310906
XOM        0.715182
dtype: float64
```

