Homework1

February 17, 2020

1 Assignment 1

Finance 5330: Financial Econometrics Tyler J. Brough Due Date: January 29, 2019

Consider the daily stock returns of American Express (AXP), Caterpillar (CAT), and Starbucks (SBUX) from January 1999 to December 2008. The data are daily prices in the file *stock-data-hwk1.txt*.

- (a) Calculate simple returns for the three series.
- (b) Express the simple returns in percentages. Compute the sample mean, standard deviation, skewness, excess kurtosis, minimum, and maximum of the percentage simple returns.
- (c) Transform the simple returns to log returns.
- (d) Express the log returns in percentages. Compute the sample mean, standard deviation, skewness, excess kurtosis, minimum, and maximum of the percentage log returns.
- (e) Test the null hypothesis that the mean of the log returns of each stock is zero. That is, perform three separate tests. Use 5% significance level to draw your conclusion.
- (f) Plot histograms for each of the three series (both simple and log returns - so six graphs total).
- (g) Test the null hypothesis that the lag-2 autocorrelation is zero for log returns.

Here is some code to extract the price time series from the raw data in Python:

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

[2]:
```
rawData = pd.read_csv("stock-data-hwk1.csv")
```

[3]:
```
rawData.head()
```

[3]:

	PERMNO	date	TICKER	PRC	vwretd	ewretd
0	18542	19990104	CAT	47.3750	-0.000822	0.011409
1	18542	19990105	CAT	47.2500	0.011879	0.009512
2	18542	19990106	CAT	48.5000	0.021143	0.014866
3	18542	19990107	CAT	48.9375	-0.000798	0.003560
4	18542	19990108	CAT	51.0000	0.004602	0.009410

[4]:
```python
tickers = ['AXP', 'CAT', 'SBUX']

ind = rawData.TICKER == tickers[0]
axpData = rawData[ind]

ind = rawData.TICKER == tickers[1]
catData = rawData[ind]

ind = rawData.TICKER == tickers[2]
sbuxData = rawData[ind]
```

[5]:
```python
axpData.head()
```

[5]:

	PERMNO	date	TICKER	PRC	vwretd	ewretd
2515	59176	19990104	AXP	101.5000	-0.000822	0.011409
2516	59176	19990105	AXP	99.5625	0.011879	0.009512
2517	59176	19990106	AXP	103.6250	0.021143	0.014866
2518	59176	19990107	AXP	104.8750	-0.000798	0.003560
2519	59176	19990108	AXP	108.0625	0.004602	0.009410

[7]:
```python
axpData.tail()
```

[7]:

	PERMNO	date	TICKER	PRC	vwretd	ewretd
5025	59176	20081224	AXP	17.97	0.004514	0.005042
5026	59176	20081226	AXP	17.91	0.007191	0.011107
5027	59176	20081229	AXP	17.70	-0.004365	-0.015163
5028	59176	20081230	AXP	18.00	0.024764	0.021418
5029	59176	20081231	AXP	18.55	0.017404	0.034456

[6]:
```python
catData.head()
```

[6]:

	PERMNO	date	TICKER	PRC	vwretd	ewretd
0	18542	19990104	CAT	47.3750	-0.000822	0.011409
1	18542	19990105	CAT	47.2500	0.011879	0.009512
2	18542	19990106	CAT	48.5000	0.021143	0.014866
3	18542	19990107	CAT	48.9375	-0.000798	0.003560
4	18542	19990108	CAT	51.0000	0.004602	0.009410

[8]:
```python
catData.tail()
```

[8]:

	PERMNO	date	TICKER	PRC	vwretd	ewretd
2510	18542	20081224	CAT	41.91	0.004514	0.005042
2511	18542	20081226	CAT	42.72	0.007191	0.011107
2512	18542	20081229	CAT	42.34	-0.004365	-0.015163
2513	18542	20081230	CAT	43.66	0.024764	0.021418
2514	18542	20081231	CAT	44.67	0.017404	0.034456

[9]:
```
sbuxData.head()
```

[9]:

	PERMNO	date	TICKER	PRC	vwretd	ewretd
5030	77702	19990104	SBUX	53.8750	-0.000822	0.011409
5031	77702	19990105	SBUX	52.0000	0.011879	0.009512
5032	77702	19990106	SBUX	51.5625	0.021143	0.014866
5033	77702	19990107	SBUX	51.7500	-0.000798	0.003560
5034	77702	19990108	SBUX	52.8750	0.004602	0.009410

[10]:
```
sbuxData.tail()
```

[10]:

	PERMNO	date	TICKER	PRC	vwretd	ewretd
7540	77702	20081224	SBUX	9.34	0.004514	0.005042
7541	77702	20081226	SBUX	9.35	0.007191	0.011107
7542	77702	20081229	SBUX	9.03	-0.004365	-0.015163
7543	77702	20081230	SBUX	9.36	0.024764	0.021418
7544	77702	20081231	SBUX	9.46	0.017404	0.034456

You can access the price time series as (for example for AXP):

[11]:
```
axpData.PRC
```

[11]:
```
2515    101.5000
2516     99.5625
2517    103.6250
2518    104.8750
2519    108.0625
2520    106.2500
2521    102.3750
2522     98.6875
2523     96.0000
2524    104.3750
2525    100.5000
2526    102.0000
2527    102.5000
2528     98.5000
2529    101.5000
2530    101.7500
2531     99.4375
2532    100.4375
2533    102.8750
2534    100.7500
2535     99.6875
2536    100.5000
2537    100.2500
2538     98.1250
2539     97.0625
```

```
2540     95.9375
2541     96.7500
2542    101.0000
2543     99.9375
2544    102.1875
          …
5000     19.3800
5001     18.7400
5002     17.2300
5003     18.6900
5004     21.1800
5005     21.3700
5006     22.3000
5007     23.3100
5008     19.6400
5009     20.7600
5010     21.8700
5011     20.8400
5012     21.7800
5013     24.4400
5014     23.2900
5015     21.5600
5016     20.1300
5017     20.3400
5018     19.3400
5019     20.0600
5020     19.8100
5021     18.9000
5022     19.4300
5023     18.4200
5024     17.9600
5025     17.9700
5026     17.9100
5027     17.7000
5028     18.0000
5029     18.5500
Name: PRC, Length: 2515, dtype: float64
```

[ ]: