

FIN3080 Assignment 4 Report

Data processing

Step 1

1. Download Daily return without dividend, it is a big file containing 9 separate CSVs. Merge 9 separate CSVs together.
2. Using the datetime part in python to convert date into the form of xx(day)-month-xxxx(year).

Stkcd	Trddt	Dretnd	Date
1	2016/1/4	-0.055046	4-Jan-16
1	2016/1/5	0.006178	5-Jan-16
1	2016/1/6	0.011404	6-Jan-16
1	2016/1/7	-0.051171	7-Jan-16
1	2016/1/8	0.016453	8-Jan-16
1	2016/1/11	-0.032374	11-Jan-16
1	2016/1/12	0.004647	12-Jan-16
1	2016/1/13	-0.009251	13-Jan-16

Step 2

1. Download Daily market return.
2. Keep records for SSE A share market by keeping the rows whose 'markettype' is 1.
3. Using the datetime part in python to convert date into the form of xx(day)-month-xxxx(year).

Markettype	Trddt	Dretmdeq	Date
1	2016/1/4	-0.087405	4-Jan-16
1	2016/1/5	-0.012811	5-Jan-16
1	2016/1/6	0.031834	6-Jan-16
1	2016/1/7	-0.090405	7-Jan-16
1	2016/1/8	0.016966	8-Jan-16
1	2016/1/11	-0.073949	11-Jan-16
1	2016/1/12	-0.003234	12-Jan-16
1	2016/1/13	-0.039239	13-Jan-16

Step 3

1. Download EPS with the short English name of stocks, industry code and code for statement type. (the data starts from 2014 so that we can calculate the SUE of 2016)
2. Exclude parent statements by dropping the rows whose 'Typrep' is 'B'.
3. Exclude ST and PT companies by dropping the rows whose 'ShortName_EN' start with 'ST ' or '*ST ' or 'PT ' or '*PT '.
4. Exclude finance companies by dropping the rows whose 'Indcd' start with 'J'. (based on industrial classification)
5. Keep interim and annual EPSs by keeping the rows whose month is '6' or '12'.
6. Use the datetime part in python to convert date into the form of '2016h1'.
7. Drop some columns.
8. Group the rows using stock codes and calculate EPS of the second half year by checking the years and subtracting them.
9. Group the rows using stock codes and calculate UE by checking the time and subtracting them.
10. Calculate the standard deviation by std function in python and divide them to get the SUE.
Exclude the outliers with the 5% and 95% quantiles. Only keep the data from 2016-1-1 to 2022-12-31.
11. Divide the stocks into 10 deciles.
12. Drop some of the lines.
13. Use pivot function in python to transfer the long dataset into a wide dataset.

Stkcd	2016h1	2016h2	2017h1	2017h2	2018h1	2018h2	2019h1	2019h2	2020h1	2020h2	2021h1	2021h2	2022h1	2022h2
2	5	7	8	9	9	9	9	4	4	7	0	0	6	4
4	4	8	2	2	6	0	3	7	6	8	4	0		
5	8	4	0	6	6	7	9	3	1					
6	2	8	6	3	7	3	0	6	6	5	9	0	2	6
7	8	9	2	1	2		5	8	4	2				
8	1	9	4	7	4	1	5	5	1	0	4	1	6	8
9	4	1	3	4	4	9	8	2	3	8	8	5	4	9
10	7	4	0		3	6						4	4	
11	3	8	9	3	1	7	4	6	8	2	9	3	1	5
12	8	4	2	3			4	5	6	9	9	4	2	8
14	0	4	8	1	4	9	4	2	7	2	0	8	8	8
16	8	7	4	9	5	2	5	4	4	7	3	7	5	0
17	5	7	1	5	8	1	2	1					0	1
18	6	1	4	3	3									
19	4	8	4	2	4	8	7	2	5	4	5	5	3	4
20	3	6	2	4	5	6	3	9	7	6	9	2	7	6
21	7	3	8	9	1	7	5	2	7	8	4	3	7	1

Step 4

1. Download the announcement date for each stock.
2. Keep the rows whose 'Repty' is 2 or 4.
3. Drop some of the columns.
4. Use pivot function in python to transfer the long dataset into a wide dataset.

Stkcd	2016h1	2016h2	2017h1	2017h2	2018h1	2018h2	2019h1	2019h2	2020h1	2020h2	2021h1	2021h2	2022h1	2022h2
1	2016/8/12	2017/3/17	2017/8/11	2018/3/15	2018/8/16	2019/3/7	2019/8/8	2020/2/14	2020/8/28	2021/2/2	2021/8/20	2022/3/10	2022/8/18	2023/3/9
2	2016/8/22	2017/3/27	2017/8/25	2018/3/27	2018/8/21	2019/3/26	2019/8/21	2020/3/18	2020/8/28	2021/3/31	2021/8/30	2022/3/31	2022/8/31	2023/3/31
4	2016/8/31	2017/4/11	2017/8/31	2018/4/28	2018/8/25	2019/4/26	2019/8/31	2020/4/28	2020/8/31	2021/4/28	2021/8/27	2022/4/30	2022/8/25	2023/4/29
5	2016/8/25	2017/4/27	2017/8/26	2018/4/21	2018/8/30	2019/4/25	2019/8/29	2020/5/23	2020/8/29	2021/4/30	2021/8/27	2022/4/27	2022/8/27	2023/4/29
6	2016/8/19	2017/3/30	2017/8/18	2018/3/30	2018/8/17	2019/3/30	2019/8/16	2020/4/9	2020/8/28	2021/3/31	2021/8/27	2022/4/16	2022/8/25	2023/4/28
7	2016/8/30	2017/4/19	2017/8/31	2018/4/24	2018/8/31	2019/4/30	2019/8/31	2020/4/29	2020/8/28	2021/4/29	2021/8/27	2022/4/30	2022/8/30	2023/4/29
8	2016/8/25	2017/4/1	2017/8/30	2018/4/26	2018/8/25	2019/4/25	2019/8/29	2020/4/24	2020/8/29	2021/4/20	2021/8/28	2022/4/29	2022/8/31	2023/4/22
9	2016/8/30	2017/4/27	2017/8/30	2018/4/28	2018/8/30	2019/4/30	2019/8/31	2020/4/29	2020/8/28	2021/3/27	2021/8/25	2022/3/31	2022/8/23	2023/4/15
10	2016/8/24	2017/4/22	2017/8/29	2018/4/27	2018/8/31	2019/4/25	2019/8/2	2020/4/30	2020/8/25	2021/4/21	2021/8/31	2022/4/28	2022/8/29	2023/4/28

Step 5

Merge data from 1~4 together, some of the data are shown below.

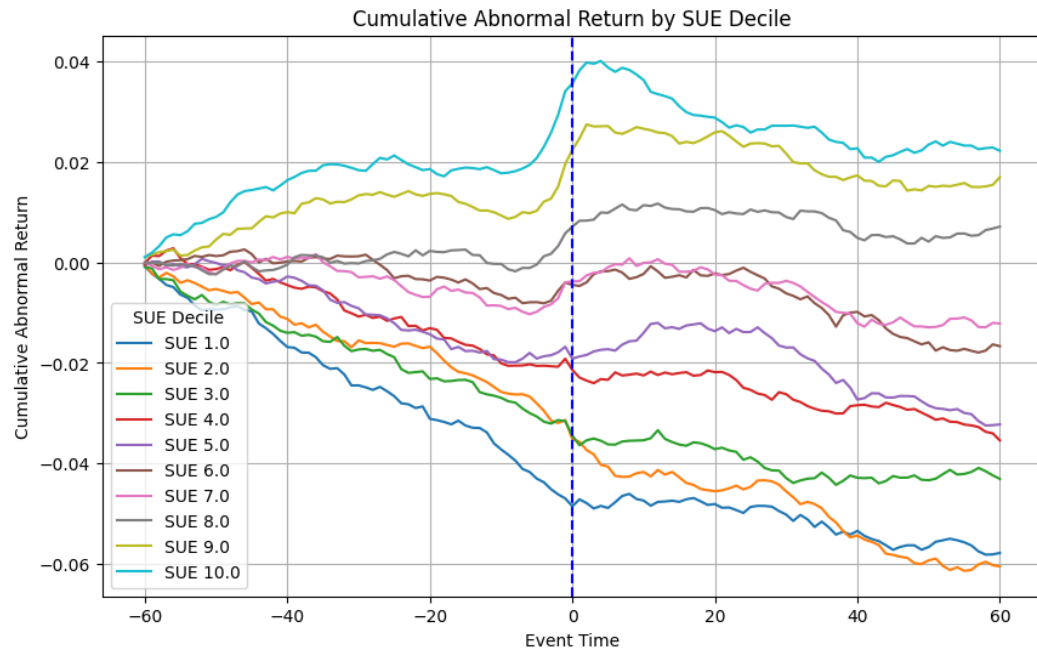
stkrw	code	name	market	ref	2016h1	2016h2	2017h1	2017h2	2018h1	2018h2	2019h1	2019h2	2020h1	2020h2	2021h1	2021h2	2022h1	2022h2
1	000001	平安银行	SH	000001	2016/8/12	2017/3/17	2017/8/11	2018/3/15	2018/8/16	2019/3/7	2019/8/8	2020/2/14	2020/8/28	2021/2/2	2021/8/20	2022/3/10	2022/8/18	2023/3/9
2	000002	万科A	SH	000002	2016/8/22	2017/3/27	2017/8/25	2018/3/27	2018/8/21	2019/3/26	2019/8/21	2020/3/18	2020/8/28	2021/3/31	2021/8/30	2022/3/31	2022/8/31	2023/3/31
3	000003	招商银行	SH	000003	2016/8/31	2017/4/11	2017/8/31	2018/4/28	2018/8/25	2019/4/26	2019/8/31	2020/4/28	2020/8/31	2021/4/28	2021/8/27	2022/4/30	2022/8/25	2023/4/29
4	000004	浦发银行	SH	000004	2016/8/25	2017/4/27	2017/8/26	2018/4/21	2018/8/30	2019/4/25	2019/8/29	2020/5/23	2020/8/29	2021/4/30	2021/8/27	2022/4/27	2022/8/27	2023/4/29
5	000005	民生银行	SH	000005	2016/8/19	2017/3/30	2017/8/18	2018/3/30	2018/8/17	2019/3/30	2019/8/16	2020/4/9	2020/8/28	2021/3/31	2021/8/27	2022/4/16	2022/8/25	2023/4/28
6	000006	中信银行	SH	000006	2016/8/30	2017/4/19	2017/8/31	2018/4/24	2018/8/31	2019/4/30	2019/8/31	2020/4/29	2020/8/28	2021/4/29	2021/8/27	2022/4/30	2022/8/30	2023/4/29
7	000007	工商银行	SH	000007	2016/8/25	2017/4/1	2017/8/30	2018/4/26	2018/8/25	2019/4/25	2019/8/29	2020/4/24	2020/8/29	2021/4/20	2021/8/28	2022/4/29	2022/8/31	2023/4/22
8	000008	农业银行	SH	000008	2016/8/30	2017/4/27	2017/8/30	2018/4/28	2018/8/30	2019/4/30	2019/8/31	2020/4/29	2020/8/28	2021/3/27	2021/8/25	2022/3/31	2022/8/23	2023/4/15
9	000009	建设银行	SH	000009	2016/8/24	2017/4/22	2017/8/29	2018/4/27	2018/8/31	2019/4/25	2019/8/2	2020/4/30	2020/8/25	2021/4/21	2021/8/31	2022/4/28	2022/8/29	2023/4/28

Step 6

1. Input the dataset.
2. Derive the ARs by subtracting the return of the stocks and the return of the market.
3. For each announcement date, find out the [-60,60] days of it. Group all the data by date index and SUE deciles and use cumsum function in python to calculate the CARs.
4. Derive the mean CAR of each day.

event_date	deciles	CAR
-60	0	-0.00073
-60	1	-0.00046
-60	2	-0.00091
-60	3	-0.0001
-60	4	-0.00067
-60	5	-0.00043
-60	6	6.72E-05
-60	7	5.89E-05
-60	8	0.001062
-60	9	0.001065
-59	0	-0.00223

5. Plot the figure.



Analysis

It is easy to find that at the announcement date, the CARs of each portfolio have different slope with before. What's more, after the announcement date, the slope of each portfolio is still different from before. For example, for decile 9, 10, after the announcement date the curve keeps decreasing, it is likely that the performance of these portfolios doesn't reach people's expectation. The drift lasts for almost 40 days, **which is a proof for the phenomena of PEAD in the A share market.**