**Event Study Project**

FIN 6336

**Section I: Part I**

The first part of the project asks to perform the event study using an event window of days {-1, 1} surrounding the announcement dates.

I use Stata as my primary tool to conduct the study. Web library of Data and Statistical Service under [Princeton University](http://dspace.mit.edu/bitstream/handle/1721.1/2068/SWP-1523-15376412.pdf;jsessionid=08A44C77C9C78580FDEF66D7B0878C2B?sequence=1) has provided codes for event study. It is my main resource of guidance.

One disadvantage of using calendar days to establish event windows is that it cannot distinguish non-trading days. Instead, estimating windows based on trading days can overcome this hurdle. I use the following codes in Stata to set up the pre-event windows, which is from 252 trading days to two trading days prior to the event, with a total of 252 annualized trading days. Event windows are between one trading day prior to the event to one trading day after the event, totaling of three trading days.

/\*get trading days\*/

gen event\_count=permno\*e\_date /\*this is the indicator of event\*/

sort event\_count date

by event\_count: gen datenum=\_n

by event\_count: gen target=datenum if date==e\_date

egen td=min(target), by (event\_count)

drop target

gen dif=datenum - td

/\*get event windows, and pre-event windos\*/

bysort event\_count: gen event\_window=1 if dif>=-1 & dif<=1

egen count\_event\_obs=count(event\_window), by(event\_count)

bysort permno: gen estimation\_window=1 if dif<-1 & dif>=-252

egen count\_est\_obs=count(estimation\_window), by(event\_count)

replace event\_window=0 if event\_window==.

replace estimation\_window=0 if estimation\_window==.

/\*delete observations with <3 trading days of event window and <30 trading days of pre-event window\*/

tab permno if count\_event\_obs<3

tab permno if count\_est\_obs<30

drop if count\_event\_obs < 3

drop if count\_est\_obs < 30

Running OLS regression of return on value weighted market return for each firm in the pre-event window, I come up with the predicted return for each event window. They are denoted as “predicted\_return” in the following codes.

/\*The following codes are for calculating the predicted returns in the event window based on the regression results in the pre-event window\*/

set more off

gen predicted\_return=.

egen id=group(event\_count)

forvalues i=1(1)808 {

list id event\_count if id==`i' & dif==0

reg ret vwretd if id==`i' & estimation\_window==1

predict p if id==`i'

replace predicted\_return = p if id==`i' & event\_window==1

drop p

}

With predicted returns from the prior codes, it is apparent regarding the procedures about the abnormal returns and the corresponding cumulative abnormal returns (CAR) in the event window.

sort id date

gen abnormal\_return=ret-predicted\_return if event\_window==1

by id: egen cumulative\_abnormal\_return = sum(abnormal\_return)

Statistical significance is based on the test statistics J1 and J2 from Chapter 2 of CLM. I calculate J1 and J2 in both Stata and Excel. J1 and J2 equal to -0.176 and -52.47, respectively. Based on J1, the null hypothesis, which says that the market responds to the news of seasoned equity offering plan with either zero or positive returns, will not be rejected. However, the null hypothesis will be rejected if using J2 as the test statistics. CLM argues that “one would like to choose the statistics with higher power”, so, I conclude that the null hypothesis is rejected at the 1% significance level based on J2.

/\* Calculate J1 and J2\*/

sort id date

by id: egen ar\_sd = sd(abnormal\_return)

by id: egen scar=cumulative\_abnormal\_return /ar\_sd

gen j1 =(1/sqrt(3)) \* ( cumulative\_abnormal\_return /ar\_sd)

gen j2= sqrt(805\*(251-4)/(251-2))\* scar

The hypothesis, not the null hypothesis, says that when firms announce the news of seasoned equity offering plan, the market would respond to this news negatively. Based on J2, the null hypothesis is rejected and the hypothesis is therefore confirmed. The results, so far, are consistent with the views of Myers and Majluf (1984). They demonstrate that the extant asymmetric information causes underinvestment, since managers tend not to invest in all projects with positive NPV. Rather, they act in the interest of existing shareholders and therefore are willing to issue new shares when doing so can generate more benefits to old shareholders. This can deprive the welfare of the new shareholders. Therefore, the market responds to the news of seasoned equity offering plan negatively, consistent with the results in the study.

**Section II: Part II**

The second part of the project asks to use the calendar time method to compute the long run performance of the equity issues over the three year period following the issue date.

Following the calendar time method of Mitchell and Stafford (1999), I clean the post event data by creating the deadline of the post event three year window. Any observations falling beyond this three year window, ranging from the first day to three years after the event, are deleted. The following Stata codes show the details.

/\*get the sample from one day after the event and three years after the event, any observations falls beyond this range are deleted\*/

gen event\_date=date(idate,"MDY") /\*event\_date is the issue date\*/

gen end\_e\_date=date(e\_date,"MDY") /\*e\_date is the date three years later\*/

gen diff=end\_e\_date-date

drop if diff<0

gen difff=event\_date - date

drop if difff>0

With the cleaned date, I count the numbers of observations each day in order to form the event portfolio and obtain the equally weighted average of the returns. Running the OLS regression of the generated daily average excess returns on market risk premium, SMB, HML, or on market risk premium alone, closes the calendar time method. The following codes show how the above can be done in Stata.

/\*count number of firms each day to get the equally weighted average of the returns\*/

bysort date: egen no\_perm = count(event\_count)

by date: egen total\_ret=sum(ret)

gen avg\_ret=total\_ret/no\_perm

sort avg\_ret smallminusbigreturn highminuslowreturn

duplicates drop avg\_ret smallminusbigreturn highminuslowreturn , force

gen excess\_r = avg\_ret - ff

reg excess\_r excessreturnonthemarket smallminusbigreturn highminuslowreturn

reg excess\_r excessreturnonthemarket

The results are reported in Table I and Table II.

Table I: Results based on Fama - French Three Factor Model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Excess Return** | **Coefficient** | **SE** | **t Value** | **P Value** |
| Market Risk Premium | 1.494 | 0.085 | 17.56 | 0 |
| SMB | 1.113 | 0.123 | 9.08 | 0 |
| HML | 0.002 | 0.175 | 0.01 | 0.992 |
| Intercept | -0.085 | 0.0005 | -160.9 | 0 |

The R2 of this model is 9.51%.

Table II: Results based on CAPM model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Excess Return** | **Coefficient** | **SE** | **t Value** | **P Value** |
| Market Risk Premium | 1.189 | 0.0616 | 19.3 | 0 |
| Intercept | -0.085 | 0.0005 | -159.87 | 0 |

The R2 of this model is 7.76%.

The results based on Fama-French Three Factor Model as presented in Table I show that small and risky firms are compensated with higher returns. The results based on CAPM Model as presented in Table II show that firms with higher β (risky firms) gain higher returns.

The intercepts in both models are negatively significant with similar values. This suggests that SEO firms tend to underperform in the long run. This could serve as new evidence supporting the market to under evaluate firms with future SEO plans, the findings in Part I.

From the perspective of market efficiency, investors would respond to the SEO news in a short period, such as three days in this study, based on which, this SEO study is possible. Nevertheless, if the market is inefficient, the market responsive period to the SEO news could be much longer. For instance, the momentum phenomenon of the stock returns is strong evidence against market efficient hypothesis. Combing the results of Part I and II, it is possible to derive a conclusion of market inefficiency due to the impact of momentum starting from SEO news and lasting for a long period, such as three years. In specific, the negative returns in part I could have been carried over for three years, as shown in part II. However, it is fallacy or a joint test problem to use asset pricing models to evaluate or discuss market efficiency hypothesis, or vice versa (Fama and French (1993)). Thus, more robustness tests needs to be conducted to strengthen the conclusions.

Regarding the statistical reliability of the results based on the calendar time method, Mitchell and Stafford do not discuss any further, besides the regular approach based on t value and p values. However, they evaluate the significance of the results of the BHARs in Section II, using bootstrap. For each sample firm, they generate “a pseudo-sample that has the same size-BE/ME distribution, the same number of observations, and the same calendar time frequency as the original event sample”, to calculate the BHAR. They repeat the procedure 1000 times to generate an empirical distribution of the BHAR and calculate the p-value to draw conclusions about statistical significance.

**Reference:**

Campbell, John, Andrew Lo, & Craig MacKinlay, The Econometrics of Financial Markets.

Fama, Eugene F.; French, Kenneth R. (1993). "Common Risk Factors in the Returns on Stocks and Bonds". Journal of Financial Economics 33 (1): 3–56

Mitchell, Mark, Erik Stafford, 2000, The Journal of Business.

Myers, Stewart and Nicholas Majluf, 1984, Corporate finance and investment decision when firms have information that investors do not have, Journal of Financial Economics, Vol. 13, No. 2, 1984, pp. 187-221.

**Appendix I: Codes for Section I**

clear

set mem 300m

/\* merge: The following merges 2 sets of data: crsp for stock returns, event raw date\*/

use event\_study\_full\_sample.dta, clear

sort permno

save newfile1.dta, replace

use event.dta, clear

sort permno

save newfile2.dta, replace

use newfile1.dta, clear

merge permno using newfile2.dta

tab \_merge

drop \_merge

save even\_full\_adate.dta

clear

/\* get the estimation window and the event window\*/

gen e\_date=date(adate,"MDY")

gen event\_count=permno\*e\_date

by event\_count, sort: gen nvals = \_n == 1

count if nvals

drop nvals

sort event\_count date

by event\_count: gen datenum=\_n

by event\_count: gen target=datenum if date==e\_date

egen td=min(target), by (event\_count)

drop target

gen dif=datenum - td

bysort event\_count: gen event\_window=1 if dif>=-1 & dif<=1

egen count\_event\_obs=count(event\_window), by(event\_count)

bysort permno: gen estimation\_window=1 if dif<-1 & dif>=-252

egen count\_est\_obs=count(estimation\_window), by(event\_count)

replace event\_window=0 if event\_window==.

replace estimation\_window=0 if estimation\_window==.

tab permno if count\_event\_obs<3

tab permno if count\_est\_obs<30

drop if count\_event\_obs < 3

drop if count\_est\_obs < 30

\*count unique observation of permno \* edate for unique event

by event\_count, sort: gen nvals = \_n == 1

count if nvals

drop nvals

set more off

gen predicted\_return=.

drop if event\_count==18121

egen id=group(event\_count)

forvalues i=1(1)808 {

list id event\_count if id==`i' & dif==0

reg ret vwretd if id==`i' & estimation\_window==1

predict p if id==`i'

replace predicted\_return = p if id==`i' & event\_window==1

drop p

}

\*Here, we created a variable "id" that numbers the companies from 1 to however many there are. The N is the number of company-event combinations that have complete data. This process iterates over the companies, runs a regression in the estimation window for each, and then uses that regression to predict a 'normal' return in the event window.

\*Abnormal and Cumulative Abnormal Returns

\*We can now calculate the abnormal and cumulative abnormal returns for our data. The daily abnormal return is computed by subtracting the predicted normal return from the actual return for each day in the event window. The sum of the abnormal returns over the event window is the cumulative abnormal return.

sort id date

gen abnormal\_return=ret-predicted\_return if event\_window==1

by id: egen cumulative\_abnormal\_return = sum(abnormal\_return)

\*Here we simply calculate the abnormal return for each observation in the event window. Then we set the cumulative abnormal return equal to the sum of the abnormal returns for each company.

\*Testing for Significance

\*We are going to compute a test statistic, test, to check whether the average abnormal return for each stock is statistically different from zero.

\* test = (1/n :AR)/(AR\_SD)

\*where AR is the abnormal return and AR\_SD is the abnormal return standard deviation.

\*Get J1 and J2

sort id date

by id: egen ar\_sd = sd(abnormal\_return)

by id: egen scar=cumulative\_abnormal\_return /ar\_sd

gen j1 =(1/sqrt(3)) \* ( cumulative\_abnormal\_return /ar\_sd)

gen j2= sqrt(808\*(251-4)/(3-2))\* scar

\*regression to test the overall event effect

reg cumulative\_abnormal\_return if dif==0, robust

**Appendix II: Codes for Section II**

clear

set mem 300m

/\* merge code merge three datesets: crsp return, ff factors and the seo data\*/

use event\_study\_full\_sample.dta, clear

sort date

save newfile1.dta, replace

use ff\_crsp\_month.dta, clear

sort date

save newfile2.dta, replace

use newfile1.dta, clear

merge date using newfile2.dta

tab \_merge

drop \_merge

save dd.dta

clear

use dd.dta, clear

sort permno

save newfile1.dta, replace

use seo\_raw\_e\_date.dta, clear

sort permno

save newfile2.dta, replace

use newfile1.dta, clear

merge permno using newfile2.dta

tab \_merge

drop \_merge

save final\_sq\_al.dta

clear

use ff\_rate.dta

gen f\_date=date(date, "MDY")

gen ff\_date=date

drop date ff\_o

gen date=f\_date

drop f\_date

save ff\_rate1.dta

use final\_sq\_al.dta, clear

sort date

save newfile1.dta, replace

use ff\_rate1.dta, clear

sort date

save newfile2.dta, replace

use newfile1.dta, clear

merge date using newfile2.dta

tab \_merge

drop \_merge

save final\_sq\_final.dta

drop if ff==.

/\*get the reducde sample from one day after the event and three years after the event, any observations falls beyond this range are deleted\*/

gen event\_date=date(idate,"MDY")

gen end\_e\_date=date(e\_date,"MDY")

gen diff=end\_e\_date-date

drop if diff<0

gen difff=event\_date - date

drop if difff>0

/\*this is the id for each event, there might have multievent in the same firm\*/

gen event\_count=permno\*event\_date

/\*count no. of firms each day to get the equally weighted average of the returns\*/

bysort date: egen no\_perm = count(event\_count)

by date: egen total\_ret=sum(ret)

gen avg\_ret=total\_ret/no\_perm

sort avg\_ret smallminusbigreturn highminuslowreturn

duplicates drop avg\_ret smallminusbigreturn highminuslowreturn , force

gen excess\_r = avg\_ret - ff

reg excess\_r excessreturnonthemarket smallminusbigreturn highminuslowreturn

reg excess\_r excessreturnonthemarket