Financial Decision Making–Computer Assignment 2: Event Study

YiTao Hu

Executive Summary

To study stock excess returns around dividend declaration dates, I computed the mean ex-dividend daily stock excess returns for a time window of ten days before the dividend declaration date to ten days afterward across all dividend declaration events for 118 stocks between 2014/01/01 to 2018/12/31. The results show that the mean stock excess returns before or after the declaration date approximately are white noise, while stocks in average witness relatively significant negative returns on the dividend declaration date. From the result, I conclude that the US stock market is relatively close to efficient market because there are no signals of information leakage or post-annoucement momentum, and stock excess returns adjust for dividends at a very fast pace.

Assumptions and Methodologies

To initiate the study, I collected the ex-dividend daily returns and dividend declaration dates for all stocks in CRSP database. For the market return, I selected value-weighted return (ex-dividend) variable from CRSP database to approximate a value-weighted market portfolio. The formula for excess return is computed as follows:

$$ExcessReturn = StockReturn_i - VWMarketReturn$$

Because of limited computational power, I selected the return data of 118 stocks or first 100,000 observations out of all stocks as a representative sample to calculate the arithmetic mean excess returns around dividend declaration dates. After computing the time series of excess returns across all stocks and dates, I used dividend declaration date as a filter to filter out the excess returns for the 21 days range. Then, I computed the arithmetic mean excess returns for each day in the window acorss all stocks and dividend events. Mathematically, the formula is as followed:

$$\bar{R}_t = \frac{1}{N \times K} \sum_{i=1}^{N} \sum_{j=1}^{K} R_{i,j,t}$$

for t= -10, -9,...9,10

Because this formula only calculates the arithmetic means instead of value-weighted means, one key assumption I am making here is that all stocks are equally weighted to compute the mean returns around the dividend declaration dates.

Results and Interpretations

From excess returns table and plot around dividend declaration days, we can observe that all returns, except the one on dividend declaration day, are approximately randomly distributed or are white noise around zeros. (This is because we substract the market returns, which deducts a upward drift terms for all stocks in average sense.) However, on exact dividend declaration date, we can observe a relatively large negative excess returns of 60 basis points. This explains the stock price adjustment for the dividend distribution.

From these observations, we conclude that the US stock market is close to semi-strong efficiency market, where investors cannot earn excess return by analyzing all public information, because there are no abnormal excess returns prior to or after the declaration date, which implies no inside trading and investors' under-reaction. Actually, the fact that stock experienced a significant drop on the declaration date indicates that the investors

react quickly to adjust stock price for dividend distributions.The results also confirms MM theory that dividend policy has no impacts on firm value in an efficiency market.

Overall, from this event study, we conclude that there are no arbitrage opportunities through trading around the dividend declaration dates.

Code, Table and Plot

```
#import the data and packages
library(readr)
library(reshape2)
library(reshape)
```

```
##
## Attaching package: 'reshape'
```

```
## The following objects are masked from 'package:reshape2':
##
##     colsplit, melt, recast
```

```
Data_raw <- read_csv("HW2.csv",
                     col_types = cols(DCLRDT = col_date(format = "%Y/%m/%d"),
                                                 date = col_date(format = "%Y/%m/%d")))

Sample_data=Data_raw[1:1000000,]
Sample_data$ExRtn=Sample_data$RETX-Sample_data$vwretx
Sample_data=Sample_data[,c(2,3,4,7)]# grab relavent cols
declare_date=na.omit(Sample_data$DCLRDT)
declalre_date_idx=which(! is.na(Sample_data$DCLRDT))

#initialize the dataframe
event_performance=data.frame(matrix(data = 0,
                                    nrow = length(declare_date)-1,ncol = 21))
colnames(event_performance)=seq(-10,10,by = 1)
idx=1
#filter out the returns around dividend declaration dates
for (t in declalre_date_idx[-1])
  { if (Sample_data$TICKER[(t-10)]==Sample_data$TICKER[(t+10)] &&
        !is.na(Sample_data$TICKER[(t+10)]))
  { event_performance[idx,]=Sample_data$ExRtn[(t-10):(t+10)]
  idx=idx+1
}

}
output=data.frame(t(colMeans(event_performance)))
colnames(output)=seq(-10,10,by = 1)
row.names(output)='MeanExcessReturn'
```

The time series of mean excess returns around declaration Date

```
output
```

```
##                             -10           -9            -8           -7
## MeanExcessReturn -0.0004715816 -0.000124779 -0.0006485191 0.0004453683
##                              -6           -5           -4           -3
## MeanExcessReturn 5.671689e-05 0.0002662069 0.0002377882 0.0001457845
```

```
##                              -2           -1            0             1
## MeanExcessReturn -0.0001627371 0.0001240813 -0.007535827 -0.0003915657
##                               2            3             4             5
## MeanExcessReturn -0.0001532211 -0.001092613 -0.0002138211 -7.638578e-05
##                               6            7            8             9
## MeanExcessReturn 0.0001256661 0.0005091274 0.0005277748 -3.573457e-06
##                              10
## MeanExcessReturn -7.26772e-05
```

The plot of time series of mean excess returns around declaration Date

```
#plot the returns around dividend payment dates
plot(y = colMeans(event_performance), x = seq(-10,10,by = 1),type='l',
     xlab = 'Relative to Dividend Declaration Date',
     ylab = 'Mean Daily Return',
     main='Event Study of Returns around Dividend Declaration Date')
abline(v=0, col="blue")
text(x=0,y=-0.007,"Div Declare Date")
```