FIN511 Week 2

First code chunk used to load the necessary libraries in R if you don't have these libraries, please uncomment the code and install them or install them in R-Studio.

```
# load libraries

# install readxl and tidyverse (or just ggplot2 and dyplr)
# install.packages("readxl","ggplot2","dplyr")

library(ggplot2)
library(readxl)
library(reshape2)
```

The following code chunk shows a very "dumb" way to load excel spreadsheets in R directly, skipping lines if necessary, like in this case and use the function "head" to see if the spreadsheet is loaded properly

```
# Load all the datasets and separate sheets, clean up the data and leave a single row for column names.
# You will still need the original excel sheet to read the instructions at the top or additional inform
# modify the files paths to match your needs
# I didn't set working directories or anything else to keep the customization to a minimum

# This excel spreadsheet is not very clean, data starts on row 4, there is an empty column and on colum
# Let see how to clean it up.


df.small_value <- read_xlsx("/Users/ataru074/Desktop/Education/MBA/03 2020 Fall/FIN511 Investments/Smal
```

```
## New names:
## * `` -> ...6
## * `` -> ...9
## * `` -> ...10
```

```
# we observe that column 6 is empty and 9 and 10 are not useful
head(df.small_value)
```

```
## # A tibble: 6 x 10
##     date Mkt_RF   SMB   HML    RF ...6  SmallValue SmallValue_RF ...9  ...10
##    <dbl>  <dbl> <dbl> <dbl> <dbl> <lgl>      <dbl>         <dbl> <lgl> <chr>
## 1 192701  -0.06 -0.56  4.83  0.25 NA          3.93          3.68 NA    SmallVal~
## 2 192702   4.18 -0.1   3.17  0.26 NA          6.62          6.36 NA    1.486704~
## 3 192703   0.13 -1.6  -2.67  0.3  NA         -3.17         -3.47 NA    <NA>
## 4 192704   0.46  0.39  0.67  0.25 NA          3.48          3.23 NA    SmallVal~
## 5 192705   5.44  1.41  4.92  0.3  NA         13.3          13.0  NA    <NA>
## 6 192706  -2.34  0.48 -1.53  0.26 NA         -3.94         -4.2  NA    <NA>
```

```
# let's clean up: nothing easier, we just say that our dataframe is the same dataframe minus column 6,9
# just remember that the indexes in the square brakets are [rows, columns] and notice the "-c" that ind
# minus this list, otherwise we would have kept only column 6, 9, 10

df.small_value <- df.small_value[,-c(6,9,10)]
```

```
# double check
head(df.small_value)
```

```
## # A tibble: 6 x 7
##     date Mkt_RF   SMB   HML    RF SmallValue SmallValue_RF
##    <dbl>  <dbl> <dbl> <dbl> <dbl>      <dbl>         <dbl>
## 1 192701  -0.06 -0.56  4.83  0.25       3.93          3.68
## 2 192702   4.18 -0.1   3.17  0.26       6.62          6.36
## 3 192703   0.13 -1.6  -2.67  0.3       -3.17         -3.47
## 4 192704   0.46  0.39  0.67  0.25       3.48          3.23
## 5 192705   5.44  1.41  4.92  0.3       13.3          13.0
## 6 192706  -2.34  0.48 -1.53  0.26      -3.94         -4.2
```

Evaluation of Small-Value Stock investment strategy

here we are going to compute the required values and present them.

```
# computation of the values, rounded to the third digit
av.return.in.pct = round(mean(df.small_value$SmallValue),digits = 3)
av.exc.ret.in.pct = round(mean(df.small_value$SmallValue_RF),digits = 3)
us.stock.mkt.exc.ret = round(mean(df.small_value$Mkt_RF), digits = 3)
small.cap.minus.large.cap = round(mean(df.small_value$SMB), digits = 3)
value.min.growth = round(mean(df.small_value$HML), digits = 3)
```

Now we can report:

Small-Value Stocks Return(SmallValue): 1.487%

Small-Value Stocks Excess Return (SmallValue_RF): 1.203%

US Stock Market Excess Return (Mkt_RF): 0.65%

Small-cap minus Large-cap Stocks (SMB): 0.226%

Value minus Growth Stocks (HML): 0.397%

Small value stocks CAPM

```
# step 1 let build the model (lm is the function for linear models, and the regression is a linear mode
model <- lm(SmallValue_RF ~ Mkt_RF ,data=df.small_value)

# step 2 let see the model and extract the coefficients
model
```

```
##
## Call:
## lm(formula = SmallValue_RF ~ Mkt_RF, data = df.small_value)
##
## Coefficients:
## (Intercept)       Mkt_RF
##       0.341        1.325
```

```
alpha <- model$coefficients[[1]]
beta <- model$coefficients[[2]]

# step 3 let verify r-squared
r_squared <- summary(model)$r.squared
```

```r
# let show the anova table to verify if the model is valid
anova(model)
```

```
## Analysis of Variance Table
##
## Response: SmallValue_RF
##             Df Sum Sq Mean Sq F value    Pr(>F)
## Mkt_RF       1  54377   54377  3264.8 < 2.2e-16 ***
## Residuals 1054  17555      17
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# and the summary of the model
summary(model)
```

```
##
## Call:
## lm(formula = SmallValue_RF ~ Mkt_RF, data = df.small_value)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -17.242  -2.183  -0.257   1.687  44.557
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.34101    0.12649   2.696  0.00713 **
## Mkt_RF       1.32532    0.02319  57.139  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.081 on 1054 degrees of freedom
## Multiple R-squared:  0.756,  Adjusted R-squared:  0.7557
## F-statistic:  3265 on 1 and 1054 DF,  p-value: < 2.2e-16
```

Average return (in %): 1.487%

Average excess return (in %): 1.203%

CAPM alpha (in %): 0.3410119

CAPM market beta: 1.3253152

R-squared: 0.7559529

3-Factor model

```r
# step 1 let build the model
model.3 <- lm(SmallValue_RF ~ Mkt_RF + SMB + HML ,data=df.small_value)

# step 2 let see the model and extract the coefficients
model.3
```

```
##
## Call:
## lm(formula = SmallValue_RF ~ Mkt_RF + SMB + HML, data = df.small_value)
##
## Coefficients:
```

```
## (Intercept)       Mkt_RF          SMB          HML
##     0.01224      1.02618      0.92929      0.78847
```

```
# notice the index values reflect the order in which the variables are put in the equation
# except for alpha (intercept) which will be 1
alpha.3 <- model.3$coefficients[[1]]
beta.3.market <- model.3$coefficients[[2]]
beta.3.size <- model.3$coefficients[[3]]
beta.3.value <- model.3$coefficients[[4]]
# step 3 let verify r-squared
r_squared.3 <- summary(model.3)$r.squared

# let show the anova table to verify if the model is valid
anova(model.3)
```

```
## Analysis of Variance Table
##
## Response: SmallValue_RF
##             Df  Sum Sq Mean Sq F value    Pr(>F)
## Mkt_RF       1   54377   54377  103192 < 2.2e-16 ***
## SMB          1    9210    9210   17478 < 2.2e-16 ***
## HML          1    7790    7790   14784 < 2.2e-16 ***
## Residuals 1052    554       1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# and the summary of the model
summary(model.3)
```

```
##
## Call:
## lm(formula = SmallValue_RF ~ Mkt_RF + SMB + HML, data = df.small_value)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5297 -0.3798 -0.0035  0.3638  6.1848
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.012245   0.022592   0.542    0.588
## Mkt_RF      1.026177   0.004459 230.137   <2e-16 ***
## SMB         0.929286   0.007324 126.889   <2e-16 ***
## HML         0.788475   0.006485 121.590   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7259 on 1052 degrees of freedom
## Multiple R-squared:  0.9923, Adjusted R-squared:  0.9923
## F-statistic: 4.515e+04 on 3 and 1052 DF,  p-value: < 2.2e-16
```

3-Factor alpha (in %): 0.0122446

3-Factor market beta: 1.0261767

3-Factor size beta: 0.9292861

3-Factor value: 0.7884745

R-squared: 0.9922934

Load the Three mystery securities

```
df.3.myst <- read_xlsx("/Users/ataru074/Desktop/Education/MBA/03 2020 Fall/FIN511 Investments/ThreeMyst
```

```
## New names:
## * `` -> ...6
```

```
df.3.myst <- df.3.myst[,-6]

head(df.3.myst)
```

```
## # A tibble: 6 x 8
##     date Mkt_RF   SMB   HML    RF `Security 1` `Security 2` `Security 3`
##    <dbl>  <dbl> <dbl> <dbl> <dbl>        <dbl>        <dbl>        <dbl>
## 1 199501   1.8  -2.95  1.64  0.42         1.67         4.59        -1.71
## 2 199502   3.63 -0.33  0.38  0.4          3.14        -1.24         7.74
## 3 199503   2.18 -0.38 -2.05  0.46         0.38         5.91         6.28
## 4 199504   2.11 -0.42  1.7   0.44         2.6         -4.35         6.01
## 5 199505   2.9  -2.25  1.92  0.54         2.11         3.92         2.26
## 6 199506   2.72  3.06 -2.99  0.47         7.05        -0.92        10.4
```

generate basic stats and CAPM regression for mystery security 1

```
# computation of the values, rounded to the third digit
mys.1.av.ret = round(mean(df.3.myst$`Security 1`),digits = 3)
mys.1.av.ex.ret = round(mean(df.3.myst$`Security 1` - df.3.myst$RF),digits = 3)

# step 1 let build the model (lm is the function for linear models, and the regression is a linear mode
model.m.1 <- lm(`Security 1`~ Mkt_RF ,data=df.3.myst)

# step 2 let see the model and extract the coefficients
model.m.1
```

```
##
## Call:
## lm(formula = `Security 1` ~ Mkt_RF, data = df.3.myst)
##
## Coefficients:
## (Intercept)       Mkt_RF
##     -0.6779       1.7326
```

```
mys.1.CAPM.alpha <- model.m.1$coefficients[[1]]
mys.1.CAPM.beta <- model.m.1$coefficients[[2]]

# step 3 let verify r-squared
mys.1.r.squared <- summary(model.m.1)$r.squared

# let show the anova table to verify if the model is valid
anova(model.m.1)
```

```
## Analysis of Variance Table
##
## Response: Security 1
##            Df  Sum Sq Mean Sq F value    Pr(>F)
## Mkt_RF      1 14566.7 14566.7  403.05 < 2.2e-16 ***
## Residuals 238  8601.5    36.1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# and the summary of the model
summary(model.m.1)
```

```
##
## Call:
## lm(formula = `Security 1` ~ Mkt_RF, data = df.3.myst)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.8128  -3.0524  -0.7313   2.1674  30.8249
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.6779     0.3925  -1.727   0.0854 .
## Mkt_RF        1.7326     0.0863  20.076   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.012 on 238 degrees of freedom
## Multiple R-squared:  0.6287, Adjusted R-squared:  0.6272
## F-statistic: 403.1 on 1 and 238 DF,  p-value: < 2.2e-16
```