

# Quant Asset Management PS1

Nitish Ramkumar - 704873047

*Collaborators: Ian Laker, Forrest Henslee, Stephan Du Toit, Yvonne Yi Tong*

## Question 1

### Sample Data

Data for this question was retrieved from CRSP (MSF,DSENNAMES,DSEDELIST). The range is from February 1926 to December 2015.

Note: Steps on generation of test data is explained after the answers for the questions.

### Clean up

For calculating the value weighted returns, we need to initially clean up the data provided to the function. This involves i) Check if SHRCOD is 10 or 11 and EXCHCD is either 1,2 or 3. ii) Check if returns are not -99,-88,-77,-66,-55,-44 (invalid values)

### Convert to Monthly (if data is in other format)

The code also doesn't make any assumption of monthly returns (as the input format is date). So the returns are converted to monthly returns by summing the log returns.

```
##Annualize returns and handle delist
#convert to log returns
CRSP_Stocks[,c('month','year','RET','DLRET','RETX','DLRETX')]:=
    .(month(DATE),year(DATE),log(1+RET),log(1+DLRET),
      log(1+RETX),log(1+DLRETX))

#annualize (if data is daily) by adding within groups of year and month
ret <- CRSP_Stocks[,.(ret=sum(RET,na.rm = TRUE),delistret=sum(DLRET,na.rm = TRUE)),
  by=.(PERMNO,year,month)]
```

### Handle Delisting

We add the the log returns and log delist returns across every month and year record. The NA entries of delist returns are converted to 0 in the previous step.

This code doesn't handle the case of change in exchange code during a delist. No specific information regarding this is provided in the Fama French website.

## Marketcap calculation

- 1) Market cap should be calculated using the absolute value of price and shares outstanding. This is because if a valid closing price is not available, the price is stored as negative of average of bid/ask spread.
- 2) Code ignores NA while calculating metrics, as this means that a valid price or shares outstanding was not available at the beginning of the month.

## Metrics

The output metrics are calculated by using 1 month lag marketcap value for each PERMNO.

The results for the sample data are as below.

```
##      Year Month Stock_lag_MV Stock_Vw_Ret Stock_Ew_Ret
##    1: 1926     3    26014.10  -0.06448837 -0.097205954
##    2: 1926     4    24057.52   0.03676637  0.031609375
##    3: 1926     5    25353.38   0.01274423  0.003200069
##    4: 1926     6    25244.15   0.05465176  0.049888822
##    5: 1926     7    26518.04   0.03166632  0.012721520
##    ---
## 1074: 2015     8   22572653.30 -0.06001967 -0.045829474
## 1075: 2015     9   21310098.81 -0.03071571 -0.058052532
## 1076: 2015    10   20584704.89  0.07741655  0.049700033
## 1077: 2015    11   22072303.06  0.00562297  0.024499684
## 1078: 2015    12   22070992.64 -0.02160539 -0.045472408
```

## Question 2

### Sample Data

- 1) The output from Question 1 is used, but truncated to July 1926 - December 2015.
- 2) Fama French data, which contains the market portfolio, SMB, HML and risk free returns.

## Statistics calculation

- 1) Mean and variance of the returns are calculated and then annualized on a monthly basis.
- 2) Kurtosis and skewness is calculated using moments library functions.
- 3) Risk free rate is removed from the calculated value weighted return before comparing with the Fama French Market-Rf field.

The results for the sample data are as below.

```
##              Estimated      Actual
## Mean              0.08049536 0.0805235
## Standard Deviation 0.18642178 0.1866794
## Sharpe Ratio       0.43179161 0.4313464
## Skewness           0.20159265 0.1951098
## Excess Kurtosis     7.78834420 7.7411061
```

## Question 3

### Sample data

Same sample data as Question 2.

### Calculation

- 1) The correlation and max difference using the calculated (average weight - risk free) returns and actual market portfolio returns.

```
##      Correlation Max difference
##      0.999978390      0.004635904
```

The absolute maximum difference is almost 0. The small difference might be because the sample data set doesn't specifically manage the change in exchanges after a delist happens.

### Test Data

The value weighted returns are calculated based on data from WRDS. The code for pulling data from WRDS is as below. Some important pointers are:

- 1) Returns, price and stocks outstanding are found in **CRSPQ.MSF**
- 2) EXCHCD and SHRCN are found in **CRSPQ.DSENNAMES**
- 3) DLRET and DLRETX are found in **CRSPQ.DSEDELIST**
- 4) Remove Delist entries which have DLSTCD = 100, as it means the stock is active.

```
readFromFile = TRUE
if(!readFromFile){
  ##WRDS CONNECTION#####
  library(rJava)
  options(java.parameters = '-Xmx4g')
  library(RJDBC)

  sasPath <- "C:\\Users\\nitis\\OneDrive\\Documents\\WRDS_Drivers"
  sasCore <- paste0(sasPath, "\\sas.core.jar")
  sasDriver <- paste0(sasPath, "\\sas.intrnet.javatools.jar")
  .jinit()
  .jaddClassPath(c(sasCore, sasDriver))
  drv <- JDBC("com.sas.net.sharenet.ShareNetDriver", sasDriver, identifier.quote="")
  wrds <- dbConnect(drv, "jdbc:sharenet://wrds-cloud.wharton.upenn.edu:8551/", user, pass)

  # Function to get data #
  getData <- function(sql, n = -1){
    #setup connection
    res <- dbSendQuery(wrds, sql)
    dbHasCompleted(res)

    #perform fetch
    returnData <- fetch(res, n)
  }
}
```

```

    #clear memory
    dbClearResult(res)
    return(returnData)
}
#####

setwd("C:/_UCLA/Quarter3/237H_QuantAsset/ProblemSets/PS1")
#LIBRARIES
require(data.table)
require(dplyr)
require(moments)
require(lubridate)

#GET DATA - Q1
sql1 <- "SELECT DATE,PERMNO,RET,RETX, PRC, SHROUT FROM CRSPQ.MSF WHERE DATE between
'01feb1926'd and '31dec2015'd"
sql2 <- "SELECT PERMNO, NAMEENDT, NAMEDT, SHRCD, EXCHCD FROM CRSPQ.DSENNAMES
WHERE NAMEENDT >= mdy(01,02,1926) AND NAMEDT <= mdy(12,31,2015)"
sql3 <- "SELECT PERMNO,DLSTDT,DLSTCD,NWPERM,DLRET,DLRETX FROM CRSPQ.DSEDELIST
WHERE DLSTDT between '01feb1926'd and '31dec2015'd"
stocks <- getData(sql1)
stocknames <- getData(sql2)
delist <- getData(sql3)

#FORMAT DATA
stocks$DATE <- as.Date(stocks$DATE)
stocknames$NAMEDT <- as.Date(stocknames$NAMEDT)
stocknames$NAMEENDT <- as.Date(stocknames$NAMEENDT)
delist$DLSTDT <- as.Date(delist$DLSTDT)

#Stock returns info in data table
stocks <- data.table(stocks)
stocks[,c("month","year"):=.(month(DATE),year(DATE))]
setkey(stocks,PERMNO)
setorder(stocks,PERMNO,DATE)

#Put EXCHCD and SHRCD information into data table
stocknames <- data.table(stocknames)
setorder(stocknames,PERMNO)
stocknames[,c("nameMnth","nameYr","namEndMnth","namEndYr"):=.(month(NAMEDT),
                                                                year(NAMEDT),month(NAMEENDT),year(NAMEENDT))]
setkey(stocknames,PERMNO)

tret <- stocknames[stocks,nomatch=0,allow.cartesian=TRUE]
tret <- tret[((nameYr < year) | (nameYr == year & nameMnth <= month))
            & ((namEndYr > year) | (namEndYr == year & namEndMnth >= month))]
setorder(tret,PERMNO,DATE)
tret <- tret[,c("NAMEDT","NAMEENDT","nameMnth","nameYr","namEndMnth","namEndYr"):=NULL]

#Delist info in data table
delist <- data.table(delist)
delist <- delist[,.(PERMNO,DLSTDT,DLSTCD,NWPERM,DLRET,DLRETX)]
delist[,c("month","year"):=.(month(DLSTDT),year(DLSTDT))]

```

```

delist[,c("DLSTDT", "DLSTCD", "NWPERM"):=NULL]
setkey(delist, PERMNO, month, year)

#merge stock and delist info
setkey(tret, PERMNO, year, month)
stockInfo <- merge(delist, tret, by = c("PERMNO", "year", "month"), all = TRUE)
setorder(stockInfo, PERMNO, DATE)
stockInfo <- stockInfo[,.(PERMNO, DATE, SHRCDD, EXCHCD, RET, RETX, DLRET, DLRETX, PRC, SHROUT)]

stockInfo$PERMNO <- as.integer(stockInfo$PERMNO)
stockInfo$SHRCDD <- as.integer(stockInfo$SHRCDD)
stockInfo$EXCHCD <- as.integer(stockInfo$EXCHCD)
stockInfo$SHROUT <- as.integer(stockInfo$SHROUT)
stockInfo$RET <- as.factor(stockInfo$RET)
stockInfo$RETX <- as.factor(stockInfo$RETX)
stockInfo$DLRET <- as.factor(stockInfo$DLRET)
stockInfo$DLRETX <- as.factor(stockInfo$DLRETX)

```

For the Fama French factor information, we get a .csv fom the Fama French website. The factor returns need to be converted to decimal by dividing by 100.

```

#GETDATA - Q2
datafactors <- read.csv("F-F_Research_Data_Factors.csv", header=TRUE, sep = ",",
                        stringsAsFactors = FALSE, skip = 3, nrow = 1088)
datafactors$X <- ymd(paste0(datafactors$X, "01"))
datafactors <- datafactors[datafactors$X >= '1926-07-01' &
                           datafactors$X <= '2015-12-31',]
datafactors <- datafactors%>%mutate(Year=year(datafactors$X),
                                   Month=month(datafactors$X))%>%
  mutate(Market_Minus_Rf = Mkt.RF, Rf=Rf)%>%
  select(Year, Month, Market_Minus_Rf, SMB, HML, Rf)%>%
  group_by(Year, Month)%>%
  summarize(Market_Minus_Rf = prod(1+Market_Minus_Rf/100)-1,
            SMB = prod(1+SMB/100)-1,
            HML = prod(1+HML/100)-1,
            Rf = prod(1+Rf/100)-1)

datafactors <- data.table(datafactors)

save(stockInfo, datafactors, file="Hwk1.RData")
}

```