Schoonmaker-Nicolas_Project1

Nicolas Schoonmaker

1/25/2020

Using RMD from: https://github.com/wgfoote/fin-alytics/blob/master/RMD/PR01-Rstartup-HO2.Rmd

If you want to disable inline preview of the Markdown file:

In RStudio, Tools > Global Options… > R Markdown > Show equations and Image previews (Never)

```
# Install packages
#install.packages("xtable")
#install.packages("dplyr")
#install.packages("rstudioapi")
library(xtable)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(rstudioapi)
```

```
# Get the directory so we can run this from anywhere
# Get the script directory from R when running in R
if(rstudioapi::isAvailable())
{
  script.path <- rstudioapi::getActiveDocumentContext()$path
  (script.path)
  script.dir <- dirname(script.path)
}
if(!exists("script.dir"))
{
  script.dir <- getSrcDirectory(function(x) {x})
}
(script.dir)
```

```
## [1] "."
```

```
# Set my working directory
# There is a "data" folder here with the files and the script
setwd(script.dir)
# Double check the working directory
getwd()
```

```
## [1] "C:/Users/Schoon/Documents/GitHub/FIN654-Project1"
```

```
# Error check to ensure the working directory is set up and the data directory exists inside it.  Its r
if(dir.exists(paste(getwd(),"/data", sep = "")) == FALSE) {
  stop("Data directory does not exist. Make sure the working directory is set using setwd() and the data
} else {
  print("Working directory and data set up correctly")
}
```

```
## [1] "Working directory and data set up correctly"
```

```
# See some help command info
??read.csv
```

```
## starting httpd help server ... done
```

```
??head
??na.omit
```

Part 1

In this set we will build a data set using filters and `if` and `diff` statements. We will then answer some questions using plots and a pivot table report. We will then write a function to house our approach in case we would like to run the same analysis on other data sets.

Problem

Supply chain managers at our company continue to note we have a significant exposure to heating oil prices (Heating Oil No. 2, or HO2), specifically New York Harbor. The exposure hits the variable cost of producing several products. When HO2 is volatile, so is earnings. Our company has missed earnings forecasts for five straight quarters. To get a handle on HO2 we download this data set and review some basic aspects of the prices.

```
# Read in data
# package EIAdata
#
HO2 <- read.csv("data/nyhh02.csv", header = T, stringsAsFactors = F)
# stringsAsFactors sets dates as character type
head(HO2)
```

```
##       DATE DHOILNYH
## 1 6/2/1986    0.402
## 2 6/3/1986    0.393
## 3 6/4/1986    0.378
## 4 6/5/1986    0.390
## 5 6/6/1986    0.385
## 6 6/9/1986    0.373
```

```
head(HO2, n = 10)
```

```
##         DATE DHOILNYH
## 1   6/2/1986    0.402
## 2   6/3/1986    0.393
## 3   6/4/1986    0.378
## 4   6/5/1986    0.390
## 5   6/6/1986    0.385
## 6   6/9/1986    0.373
## 7  6/10/1986    0.365
## 8  6/11/1986    0.389
## 9  6/12/1986    0.394
## 10 6/13/1986    0.398
```

```
tail(HO2, n = 10)
```

```
##            DATE DHOILNYH
## 7688 12/19/2016    1.547
## 7689 12/20/2016    1.561
## 7690 12/21/2016    1.523
## 7691 12/22/2016    1.538
## 7692 12/23/2016    1.572
## 7693 12/27/2016    1.606
## 7694 12/28/2016    1.619
## 7695 12/29/2016    1.605
## 7696 12/30/2016    1.612
## 7697   1/3/2017    1.591
```

```
HO2 <- na.omit(HO2) ## to clean up any missing data
# use na.approx() as well
str(HO2) # review the structure of the data so far
```

```
## 'data.frame':    7697 obs. of  2 variables:
##  $ DATE    : chr  "6/2/1986" "6/3/1986" "6/4/1986" "6/5/1986" ...
##  $ DHOILNYH: num  0.402 0.393 0.378 0.39 0.385 0.373 0.365 0.389 0.394 0.398 ...
```

Questions

1. What is the nature of HO2 returns? We want to reflect the ups and downs of price movements, something of prime interest to management. First, we calculate percentage changes as log returns. Our interest is in the ups and downs. To look at that we use `if` and `else` statements to define a new column called `direction`. We will build a data frame to house this analysis.

```
# Construct expanded data frame
return <- as.numeric(diff(log(HO2$DHOILNYH))) * 100 # Euler
size <- as.numeric(abs(return)) # size is indicator of volatility
direction <- ifelse(return > 0, "up", ifelse(return < 0, "down", "same")) # another indicator of volati
# =if(return > 0, "up", if(return < 0, "down", "same"))
date <- as.Date(HO2$DATE[-1], "%m/%d/%Y") # length of DATE is length of return +1: omit 1st observation
price <- as.numeric(HO2$DHOILNYH[-1]) # length of DHOILNYH is length of return +1: omit first observati
HO2.df <- na.omit(data.frame(date = date, price = price, return = return, size = size, direction = dire
```

```
str(HO2.df)
```

```
## 'data.frame':    7696 obs. of  5 variables:
##  $ date     : Date, format: "1986-06-03" "1986-06-04" ...
##  $ price    : num  0.393 0.378 0.39 0.385 0.373 0.365 0.389 0.394 0.398 0.379 ...
##  $ return   : num  -2.26 -3.89 3.13 -1.29 -3.17 ...
##  $ size     : num  2.26 3.89 3.13 1.29 3.17 ...
##  $ direction: Factor w/ 3 levels "down","same",..: 1 1 3 1 1 1 3 3 3 1 ...
```

```
nrow(HO2.df) # num data points
```

```
## [1] 7696
```

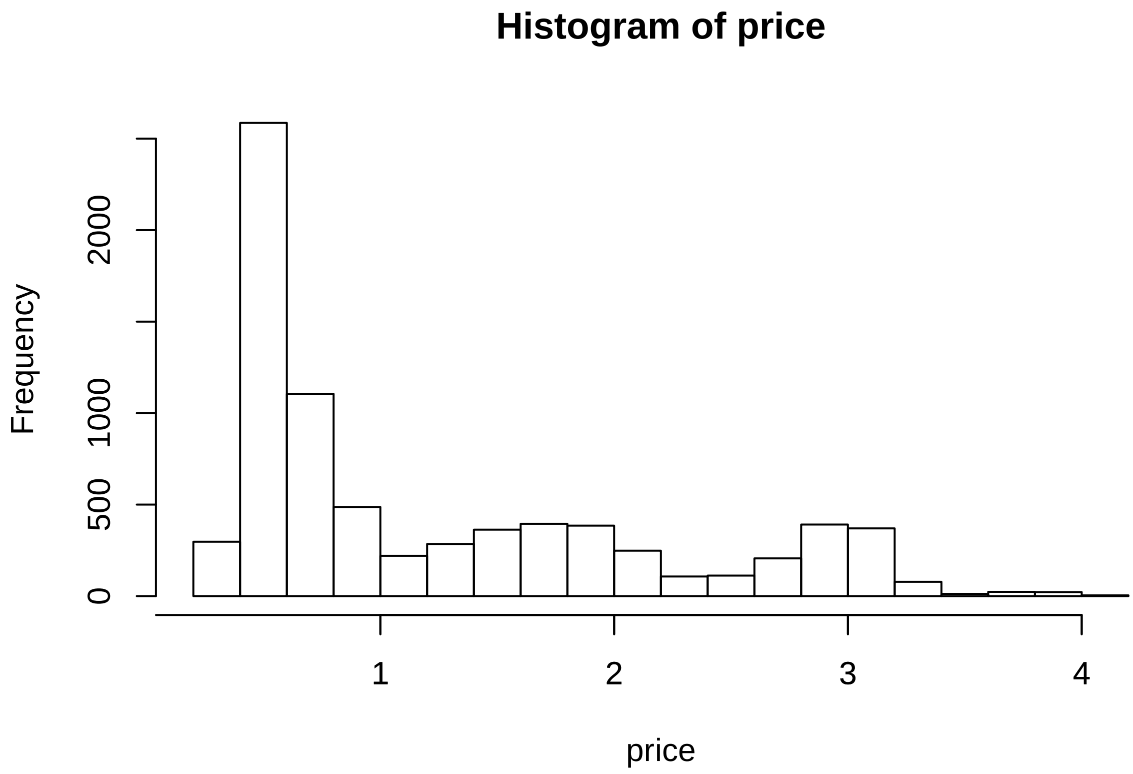
```
ncol(HO2.df) # 5, date, price, return, size, direction
```

```
## [1] 5
```

```
dim(HO2.df) # both, num rows then num cols
```

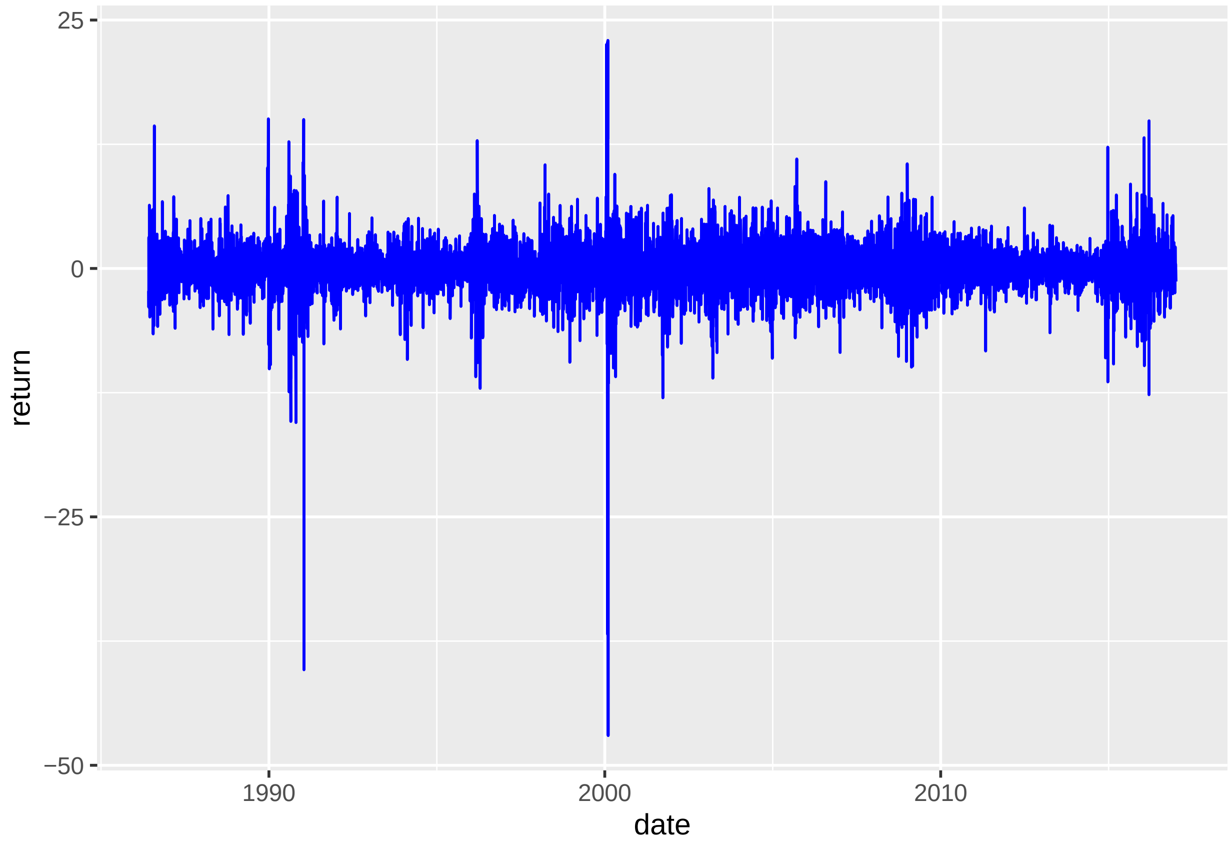
```
## [1] 7696    5
```

```
hist(price) # histogram the price
axis(side=1, at=seq(0,4,1), labels=seq(0,4,1))
```



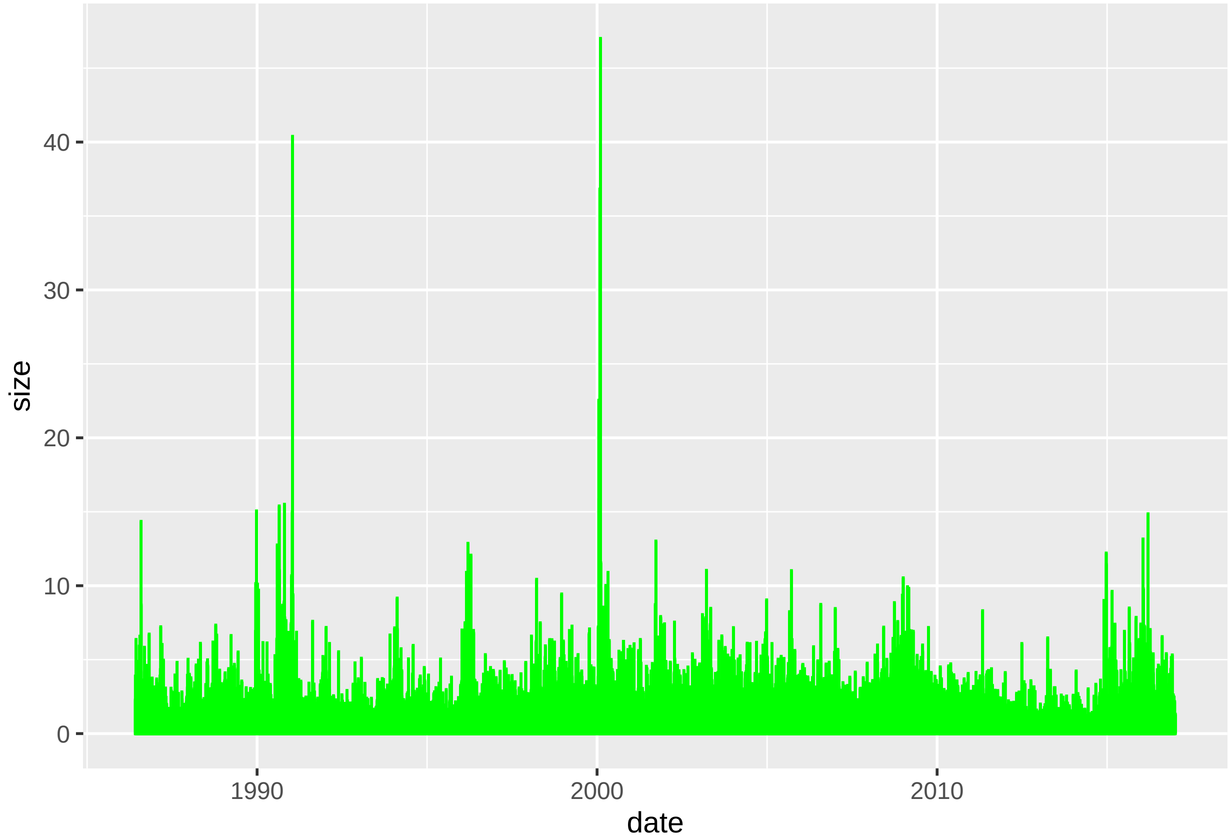
We can plot with the ggplot2 package. In the ggplot statements we use aes, "aesthetics", to pick x (horizontal) and y (vertical) axes. Use group =1 to ensure that all data is plotted. The added (+) geom_line is the geometrical method that builds the line plot.

```
library(ggplot2)
p <- ggplot(HO2.df, aes(x = date, y = return, group = 1)) + geom_line(colour = "blue")
p
```



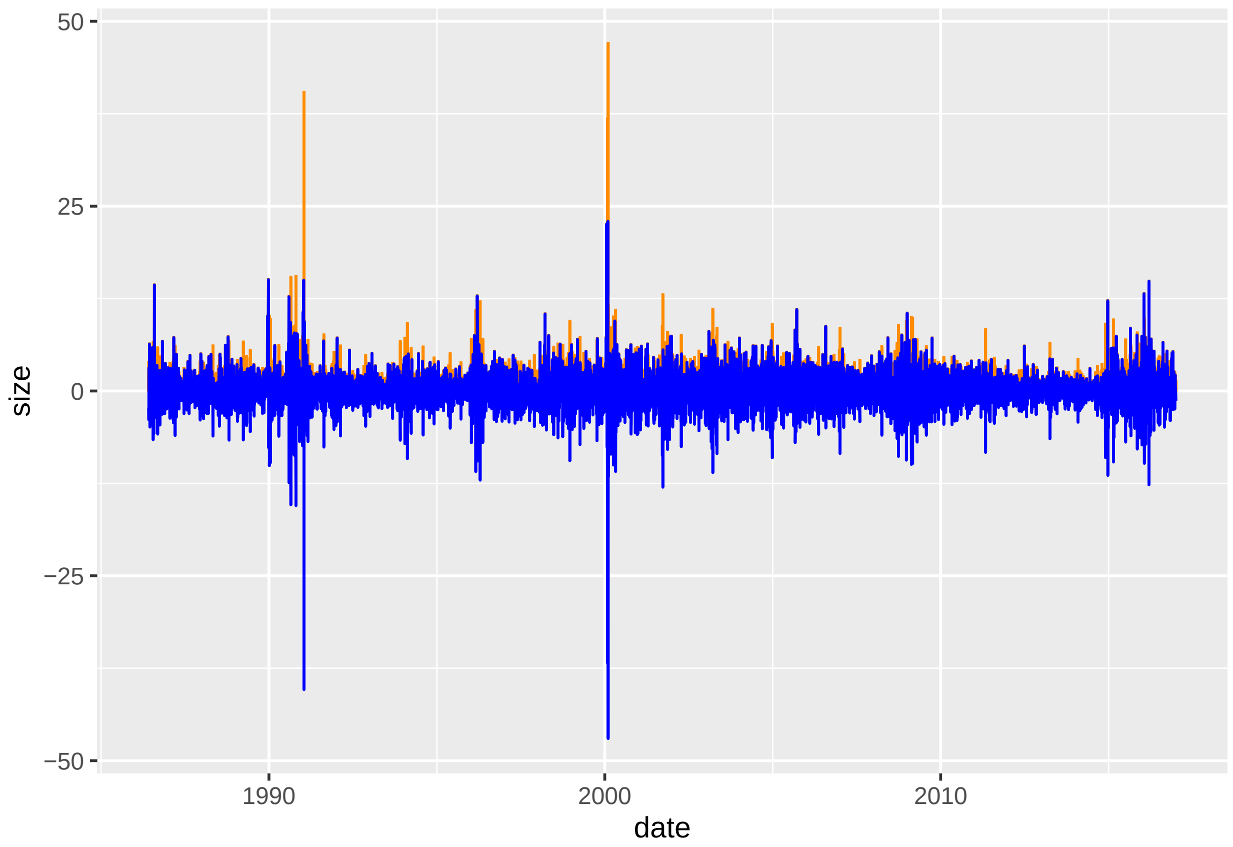
Let's try a bar graph of the absolute value of price rates. We use `geom_bar` to build this picture.

```
# library(ggplot2)
p <- ggplot(HO2.df, aes(x = date, y = size, group = 1)) + geom_bar(stat = "identity", colour = "green")
p
```



Now let's build an overlay of `return` on `size`.

```
p <- ggplot(HO2.df, aes(date, size)) + geom_bar(stat = "identity", colour = "darkorange") + geom_line(da
p
```



2. Let's dig deeper and compute mean, standard deviation, etc. Load the `data_moments()` function. Run the function using the `HO2.df$return` subset of the data and write a `knitr::kable()` report.

```r
# Load the data_moments() function
## data_moments function
## INPUTS: vector
## OUTPUTS: list of scalars (mean, sd, median, skewness, kurtosis)
data_moments <- function(data){
  library(moments)
  mean.r <- mean(data)
  sd.r <- sd(data)
  median.r <- median(data)
  skewness.r <- skewness(data)
  kurtosis.r <- kurtosis(data)
  result <- data.frame(mean = mean.r, std_dev = sd.r, median = median.r, skewness = skewness.r, kurtosi
  return(result)
}
# Run data_moments()
answer <- data_moments(HO2.df$return)
# Build pretty table
answer <- round(answer, 4)
knitr::kable(answer)
```

mean	std_dev	median	skewness	kurtosis
0.0179	2.5236	0	-1.4353	38.2595

3. Let's pivot `size` and `return` on `direction`. What is the average and range of returns by direction? How often might we view positive or negative movements in HO2?

```r
# Counting
table(HO2.df$return < 0) # one way
```

```
##
## FALSE  TRUE
##  4039  3657
```

```r
table(HO2.df$return > 0)
```

```
##
## FALSE  TRUE
##  3936  3760
```

```r
table(HO2.df$direction) # this counts 0 returns as negative
```

```
##
## down same   up
## 3657  279 3760
```

```r
table(HO2.df$return == 0)
```

```
##
## FALSE  TRUE
##  7417   279
```

```r
# Pivoting
library(dplyr)
## 1: filter to those houses with fairly high prices
# pivot.table <-  filter(HO2.df, size > 0.5*max(size))
## 2: set up data frame for by-group processing
pivot.table <-  group_by(HO2.df, direction)
## 3: calculate the summary metrics
options(dplyr.width = Inf) ## to display all columns
HO2.count <- length(HO2.df$return)
pivot.table <-  summarise(pivot.table, return.avg = round(mean(return), 4), return.sd = round(sd(return
# Build visual
knitr::kable(pivot.table, digits = 2)
```

direction	return.avg	return.sd	quantile.5	quantile.95	percent
down	-1.77	1.99	-4.78	-0.19	47.52
same	0.00	0.00	0.00	0.00	3.63
up	1.76	1.75	0.18	4.82	48.86

```r
# Here is how we can produce a LaTeX formatted and rendered table
#
library(xtable)
HO2.caption <- "Heating Oil No. 2: 1986-2016"
```

```
print(xtable(t(pivot.table), digits = 2, caption = HO2.caption, align=rep("r", 4), table.placement="V")
```

```
## % latex table generated in R 3.6.2 by xtable 1.8-4 package
## % Sun Jan 26 15:07:26 2020
## \begin{table}[ht]
## \centering
## \begin{tabular}{rrrr}
##   \hline
##  & 1 & 2 & 3 \\
##   \hline
## direction & down & same & up \\
##   return.avg & -1.7718 &  0.0000 &  1.7598 \\
##   return.sd & 1.9862 & 0.0000 & 1.7460 \\
##   quantile.5 & -4.7761 &  0.0000 &  0.1817 \\
##   quantile.95 & -0.1894 &  0.0000 &  4.8203 \\
##   percent & 47.52 &  3.63 & 48.86 \\
##    \hline
## \end{tabular}
## \caption{Heating Oil No. 2: 1986-2016}
## \end{table}
```

```
print(xtable(answer), digits = 2)
```

```
## % latex table generated in R 3.6.2 by xtable 1.8-4 package
## % Sun Jan 26 15:07:26 2020
## \begin{table}[ht]
## \centering
## \begin{tabular}{rrrrrr}
##   \hline
##  & mean & std\_dev & median & skewness & kurtosis \\
##   \hline
## 1 & 0.02 & 2.52 & 0.00 & -1.44 & 38.26 \\
##    \hline
## \end{tabular}
## \end{table}
```

Part 2

We will use the data from Part 1 to investigate the distribution of returns we generated. This will entail fitting the data to some parametric distributions as well as

Problem

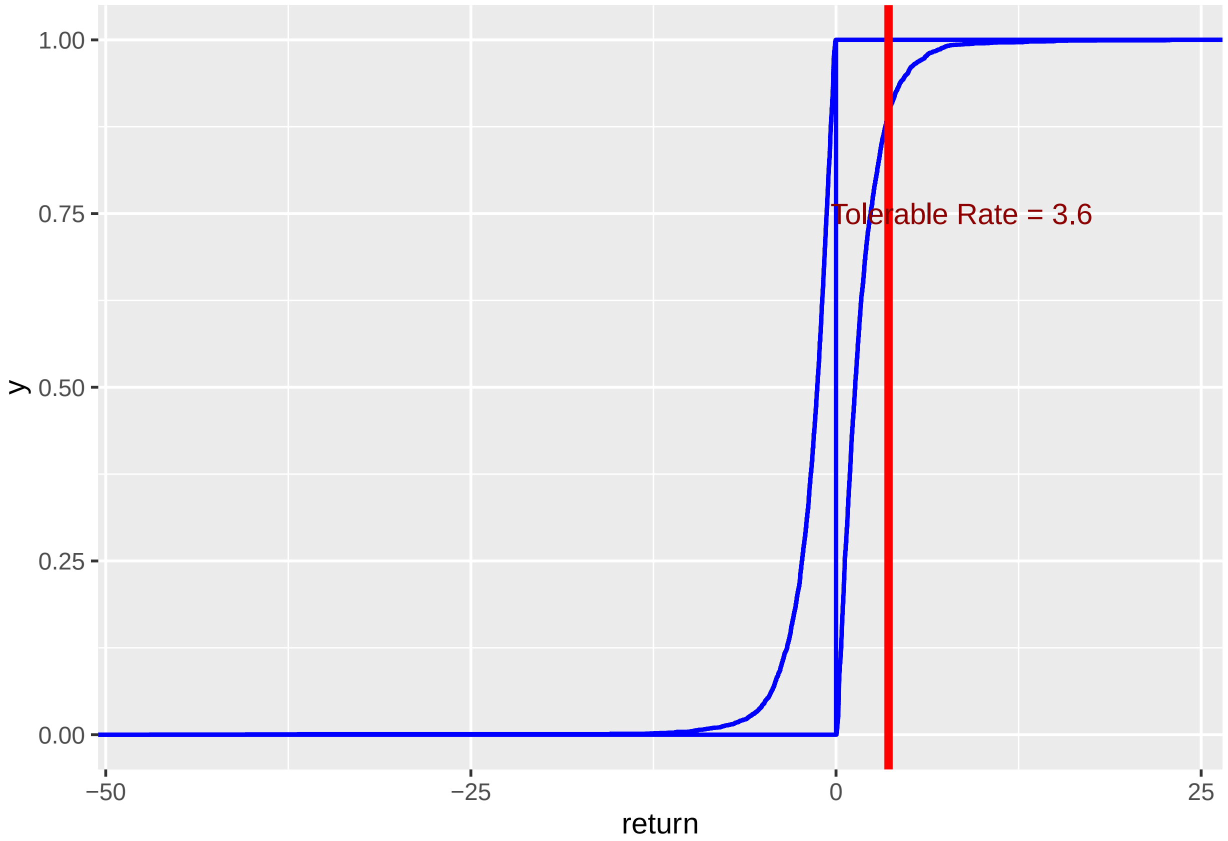
We want to further characterize the distribution of up and down movements visually. Also we would like to repeat the analysis periodically for inclusion in management reports.

Questions

1. How can we show the differences in the shape of ups and downs in HO2, especially given our tolerance for risk? We can use the `HO2.df` data frame with ggplot2 and the cumulative relative frequency function `stat_ecdf` to begin to understand this data.

```
HO2.tol.pct <- 0.95
HO2.tol <- quantile(HO2.df$return, HO2.tol.pct)
HO2.tol.label <- paste("Tolerable Rate = ", round(HO2.tol, 2), sep = "")
```

```r
ggplot(HO2.df, aes(return, fill = direction)) + stat_ecdf(colour = "blue", size = 0.75) + geom_vline(xi
```



2. How can we regularly, and reliably, analyze HO2 price movements? For this requirement, let's write a function similar to `data_moments`. Name this new function `HO2_movement()`.

```r
## HO2_movement(file, caption)
## input: HO2 csv file from /data directory
## output: result for input to kable in $table and xtable in $xtable;
##         data frame for plotting and further analysis in $df.
## Example: HO2.data <- HO2_movement(file = "data/nyhh02.csv", caption = "HO2 NYH")
HO2_movement <- function(file = "data/nyhh02.csv", caption = "Heating Oil No. 2: 1986-2016"){
  # Read file and deposit into variable
  HO2 <- read.csv(file, header = T, stringsAsFactors = F)
  # stringsAsFactors sets dates as character type
  HO2 <- na.omit(HO2) ## to clean up any missing data
  # Construct expanded data frame
  return <- as.numeric(diff(log(HO2$DHOILNYH))) * 100
  size <- as.numeric(abs(return)) # size is indicator of volatility
  direction <- ifelse(return > 0, "up", ifelse(return < 0, "down", "same")) # another indicator of vola
  date <- as.Date(HO2$DATE[-1], "%m/%d/%Y") # length of DATE is length of return +1: omit 1st observati
  price <- as.numeric(HO2$DHOILNYH[-1]) # length of DHOILNYH is length of return +1: omit first observa
  HO2.df <- na.omit(data.frame(date = date, price = price, return = return, size = size, direction = di
  require(dplyr)
  ## 1: filter if necessary
  # pivot.table <-  filter(HO2.df, size > 0.5*max(size))
  ## 2: set up data frame for by-group processing
  pivot.table <-  group_by(HO2.df, direction)
  ## 3: calculate the summary metrics
  options(dplyr.width = Inf) ## to display all columns
  HO2.count <- length(HO2.df$return)
  pivot.table <-  summarise(pivot.table, return.avg = mean(return), return.sd = sd(return), quantile.5 =
  # Construct transpose of pivot table with xtable()
  require(xtable)
  pivot.xtable <- xtable(t(pivot.table), digits = 2, caption = caption, align=rep("r", 4), table.placeme
  HO2.caption <- "Heating Oil No. 2: 1986-2016"
  output.list <- list(table = pivot.table, xtable = pivot.xtable, df = HO2.df)
return(output.list)
}
```

Test `HO2_movement()` with data and display results in a table with 2 decimal places.

```
knitr::kable(HO2_movement(file = "data/nyhh02.csv")$table, digits = 2)
```

direction	return.avg	return.sd	quantile.5	quantile.95	percent
down	-1.77	1.99	-4.78	-0.19	47.52
same	0.00	0.00	0.00	0.00	3.63
up	1.76	1.75	0.18	4.82	48.86

Morale: more work today (build the function) means less work tomorrow (write yet another report).

3. Suppose we wanted to simulate future movements in HO2 returns. What distribution might we use to run those scenarios? Here, let's use the `MASS` package's `fitdistr()` function to find the optimal fit of the HO2 data to a parametric distribution. We will use the `gamma` distribution to simulate future heating oil #2 price scenarios.

```
library(MASS)
```

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

```
HO2.data <- HO2_movement(file = "data/nyhh02.csv", caption = "HO2 NYH")$df
str(HO2.data)
```

```
## 'data.frame':    7696 obs. of  5 variables:
##  $ date     : Date, format: "1986-06-03" "1986-06-04" ...
##  $ price    : num  0.393 0.378 0.39 0.385 0.373 0.365 0.389 0.394 0.398 0.379 ...
##  $ return   : num  -2.26 -3.89 3.13 -1.29 -3.17 ...
##  $ size     : num  2.26 3.89 3.13 1.29 3.17 ...
##  $ direction: Factor w/ 3 levels "down","same",..: 1 1 3 1 1 1 3 3 3 1 ...
```

```
fit.gamma.up <- fitdistr(HO2.data[HO2.data$direction == "up", "return"], "gamma", hessian = TRUE)
```

```
## Warning in densfun(x, parm[1], parm[2], ...): NaNs produced
```

```
fit.gamma.up
```

```
##      shape         rate    
##   1.30753665   0.74299635 
##  (0.02716171) (0.01872184)
```

```
# fit.t.same <- fitdistr(HO2.data[HO2.data$direction == "same", "return"], "gamma", hessian = TRUE) # a
fit.t.down <- fitdistr(HO2.data[HO2.data$direction == "down", "return"], "t", hessian = TRUE)
```

```
## Warning in dt((x - m)/s, df, log = TRUE): NaNs produced

## Warning in dt((x - m)/s, df, log = TRUE): NaNs produced

## Warning in dt((x - m)/s, df, log = TRUE): NaNs produced

## Warning in log(s): NaNs produced

## Warning in log(s): NaNs produced

## Warning in dt((x - m)/s, df, log = TRUE): NaNs produced

## Warning in dt((x - m)/s, df, log = TRUE): NaNs produced

## Warning in dt((x - m)/s, df, log = TRUE): NaNs produced

## Warning in dt((x - m)/s, df, log = TRUE): NaNs produced

## Warning in dt((x - m)/s, df, log = TRUE): NaNs produced

## Warning in log(s): NaNs produced

## Warning in log(s): NaNs produced

## Warning in log(s): NaNs produced

## Warning in dt((x - m)/s, df, log = TRUE): NaNs produced

## Warning in log(s): NaNs produced
```

```
fit.t.down
```

```
##         m              s             df     
##   -1.30565487    0.91307703    2.50894659 
##  ( 0.02170850) ( 0.02061868) ( 0.12442996)
```

```
fit.gamma.down <- fitdistr(-HO2.data[HO2.data$direction == "down", "return"], "gamma", hessian = TRUE) #
fit.gamma.down
```

```
##      shape         rate    
##   1.31056202   0.73969342 
##  (0.02761041) (0.01889467)
```

Conclusion

Skills and Tools

The following methods & packages in R were used to explore the data; 1. We used "if" and "else" statements to define a new column called "direction" and then created a data frame to house this analysis. 2. We used the ggplot package to plot our data so that we could visualize the size and returns 3. We used the data moments function to find the standard deviation, mean, skewness and kurtosis 4. We used the knitr package to format the data moments function into a table 5. We used the table function to pivot size and return on direction 6. We used the dplyr to filter to show results for how often we might observe fluctuations in the movements of HO2 7. We used the MASS package to find the optimal fit of the HO2 data 8. We also used our statistical analysis methods to summarize and draw conclusions from the data

Data Insights

We can see from the summary statistics that our data is highly skewed with most of the extreme data below the median. The kurtosis helps us determine that our data is heavily tailed in the negative returns. If we look at all the data that we observed there are only 279 instances where there was no change in direction which indicates that heating oil is a volatile commodity. We can see from the graphs that there is extreme volatility in both the early 1990's and 2000. The extreme fluctuation in the early 1990's was due to the invasion of Kuwait followed by the Gulf War which created a major supply shock that led to a sharp increase in price of oil. In 2000 the high fluctuation may have been due to a decision by OPEC members to curb production in 1999 to reverse the preceding years decline in prices.

Business Remarks

We used a model that fit the gamma distribution which allows for the skew and the heavy tail that is in our data set. This model is especially useful for time-sensitive material. To help combat our exposure of variable costs we would need to put a procedure in place that requires upper managements approval when there are costs rise above our tolerable rate of 3.6.