Practice Set #2

Purpose, Process, Product

These practice sets will practice various `R` features in this chapter. Specifically we will practice reading in data, constructing data frames, pivoting information, developing metrics, writing functions, and tables. We build on this practice with the estimation of distribution parameters and plots using `ggplot2`. We will summarize our findings in debrief.

Set A

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 review a function to house our approach in case we would like to run some of 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 Brent we download this data set and review some basic aspects of the prices.

```
# Read in data
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
```

```
HO2 <- na.omit(HO2)  ## to clean up any missing data
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

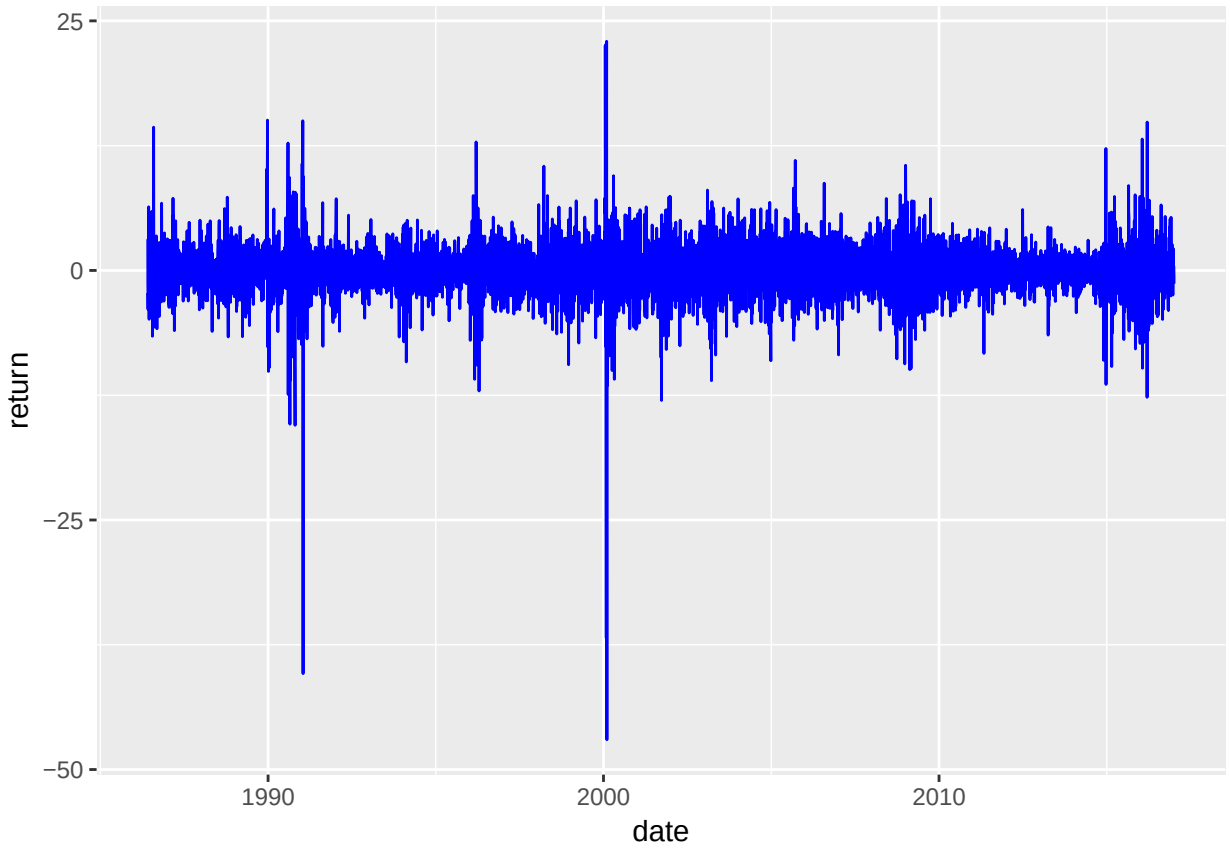
1. What is the nature of HO2 returns? We want to reflect the ups and downs of price movements, something of prime interest to management. First, we calculate percentage changes as log returns. Our interest is in the ups and downs. To look at that we use `if` and `else` statements to define a new column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

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

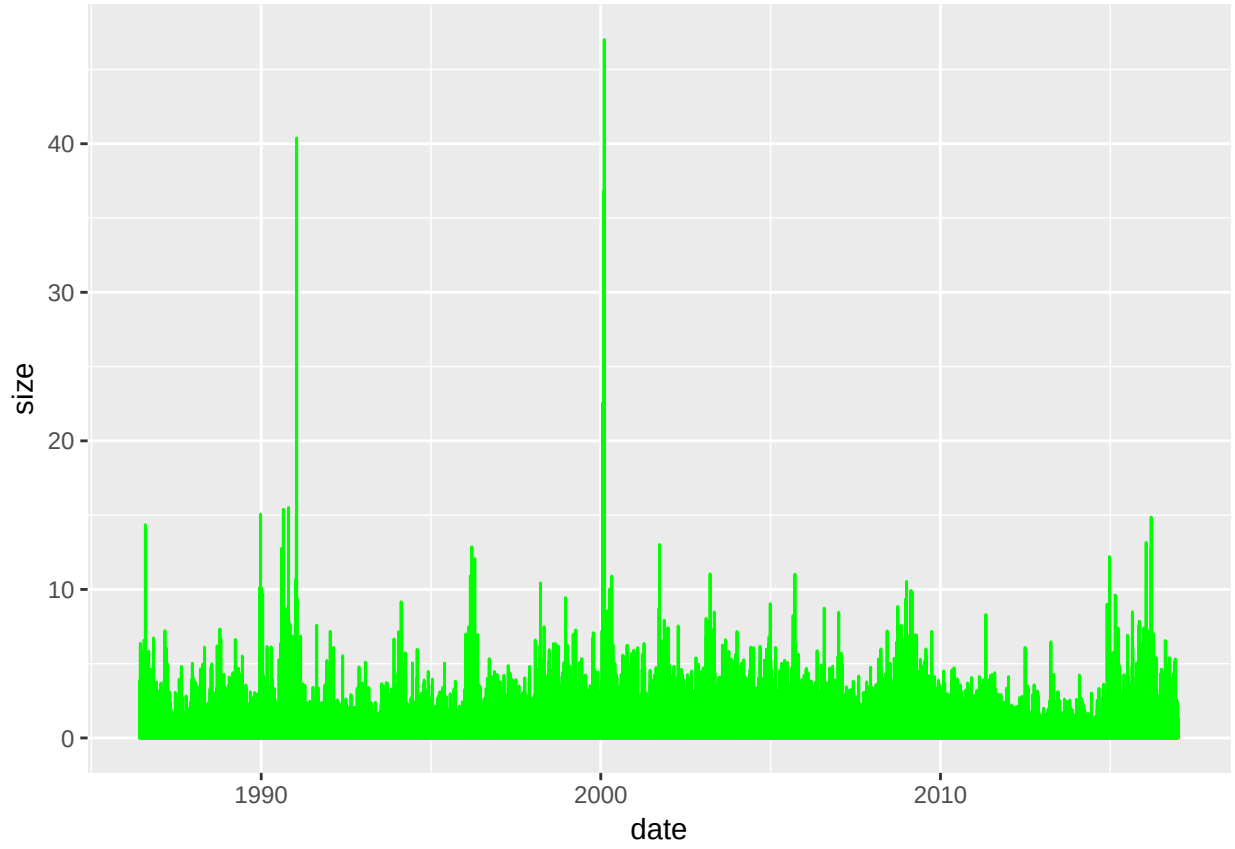
We can plot with the `ggplot2` package. In the `ggplot` statements we use `aes`, "aesthetics", to pick `x` (horizontal) and `y` (vertical) axes. Use `group =1` to ensure that all data is plotted. The added (`+`) `geom_line` is the geometrical method that builds the line plot.

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



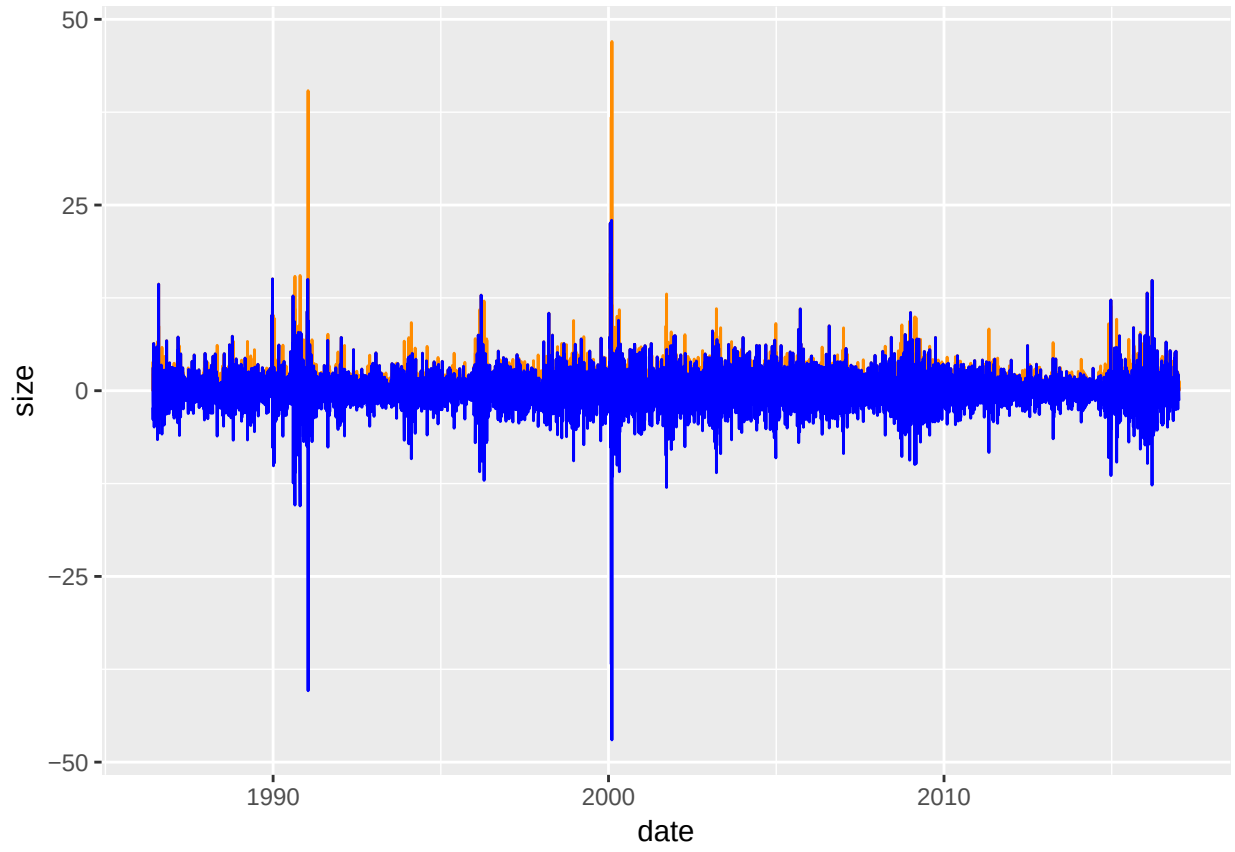
Let's try a bar graph of the absolute value of price rates. We use `geom_bar` to build this picture.

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



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

```
ggplot(HO2.df, aes(date, size)) + geom_bar(stat = "identity",
    colour = "darkorange") + geom_line(data = HO2.df,
    aes(date, return), colour = "blue")
```



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 and write a `knitr::kable()` report.

```
# Load the data_moments() function
# data_moments function INPUTS: r
# vector OUTPUTS: list of scalars
# (mean, sd, median, skewness,
# kurtosis)
data_moments <- function(data) {
    require(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, kurtosis = kurtosis.r)
    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	
0.0179	2.5236	0	-1.4353	
3. Let's	pivot 'siz	eandr	eturnon	direction'. What is the average and range of returns by direction? How often mi

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

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

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

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

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

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

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

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

```
# Pivoting
require(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 = mean(return), return.sd = sd(return),
    quantile.5 = quantile(return, 0.05),
    quantile.95 = quantile(return, 0.95),
    percent = (length(return)/HO2.count) *
        100)
# 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

Set B

We will use the data from the previous lab to investigate the distribution of returns we generated. This will entail fitting the data to some parametric distributions as well as writing a function to house results from the previous set.

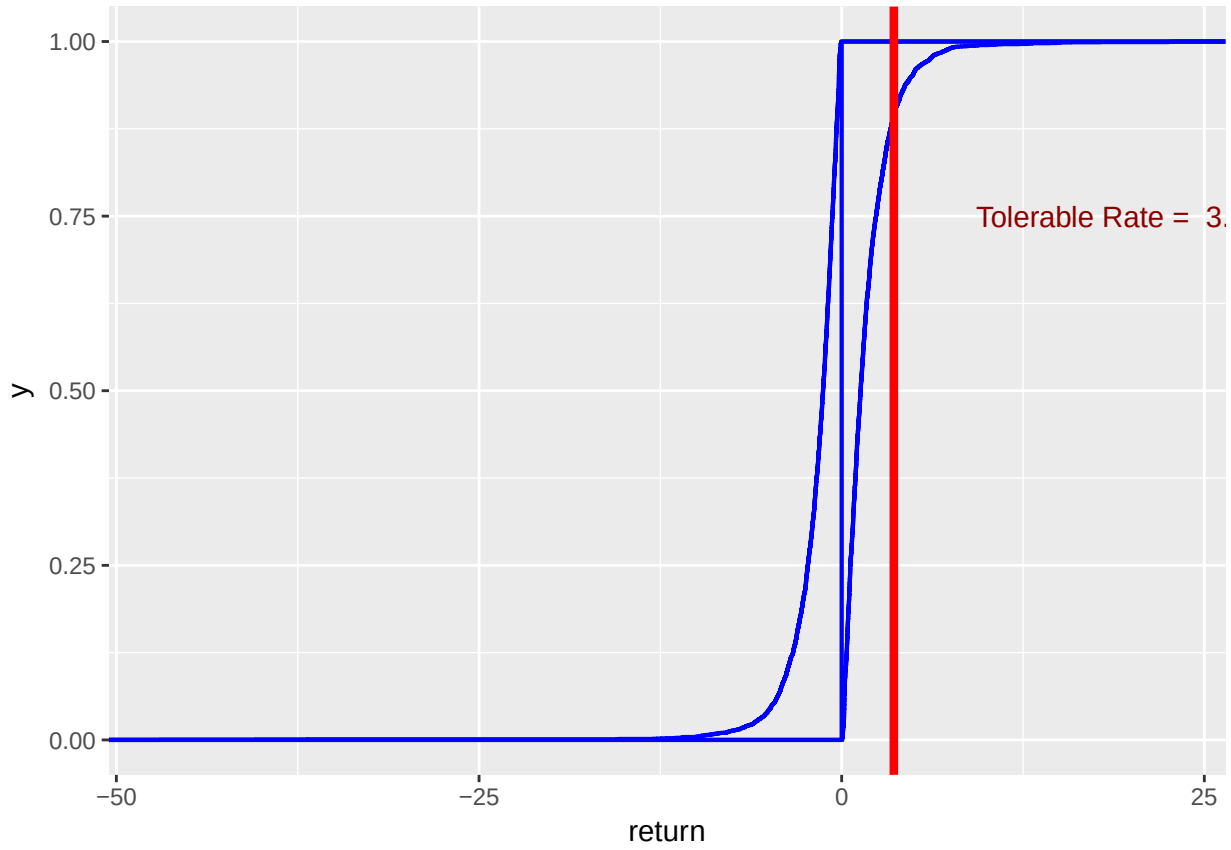
Problem

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? Let's use the `HO2.df` data frame with `ggplot2` and the cumulative relative frequency function `stat_ecdf`.

```
HO2.tol.pct <- 0.95
HO2.tol <- quantile(HO2.df$return, HO2.tol.pct)
HO2.tol.label <- paste("Tolerable Rate = ",
    round(HO2.tol, 2))
ggplot(HO2.df, aes(return, fill = direction)) +
    stat_ecdf(colour = "blue", size = 0.75) +
    geom_vline(xintercept = HO2.tol,
        colour = "red", size = 1.5) +
    annotate("text", x = HO2.tol + 15,
        y = 0.75, label = HO2.tol.label,
        colour = "darkred")
```



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

```
## 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 volatility
    date <- as.Date(HO2$DATE[-1], "%m/%d/%Y")  # length of DATE is length of return +1: omit 1st observ
```

```
    price <- as.numeric(HO2$DHOILNYH[-1])  # length of DHOILNYH is length of return +1: omit first obse
    HO2.df <- na.omit(data.frame(date = date,
        price = price, return = return,
        size = size, direction = direction))  # clean up data frame by omitting NAs
    require(dplyr)
    ## 1: filter if necessary pivot.table
    ## <- filter(HO2.df, size >
    ## 0.5*max(size)) 2: set up data frame
    ## for by-group processing
    pivot.table <- group_by(HO2.df, direction)
    ## 3: calculate the summary metrics
    options(dplyr.width = Inf)  ## to display all columns
    HO2.count <- length(HO2.df$return)
    pivot.table <- summarise(pivot.table,
        return.avg = mean(return), return.sd = sd(return),
        quantile.5 = quantile(return,
            0.05), quantile.95 = quantile(return,
            0.95), percent = (length(return)/HO2.count) *
            100)
    output.list <- list(table = pivot.table,
        df = HO2.df)
    return(output.list)
}
```

Let's test `HO2_movement()`.

```
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 fu	nction) means	less work tom	orrow (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.

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

```
##         m             s             df
##    1.33270760    0.95179926    2.71456370
##   (0.02213093) (0.02087303) (0.13999289)
```

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

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

Practice Set Debrief

1. List the R skills needed to complete these practice sets.
2. What are the packages used to compute and graph results. Explain each of them.
3. How well did the results begin to answer the business questions posed at the beginning of each practice set?