Group Assignment 1

Simon Yu

2025-10-01

Table of contents

List of Figures

List of Tables

```r
library(dplyr)
```

```r
data <- read.csv("compustat_food_bev.csv")

# filter data for Potbelly (PBPB)
pbpb_data <- filter(data, tic == "PBPB")

# convert datadate to Date type
pbpb_data$datadate <- as.Date(pbpb_data$datadate, format = "%d/%m/%Y")
```

Question 1

```r
#1. Add a new column named daily_return
#   to store daily return value

#2. Drop the rows whose daily return values are NA
pbpb_data <- mutate(pbpb_data, daily_return = (prccd - lag(prccd)) / lag(prccd))
pbpb_data <- filter(pbpb_data, !is.na(daily_return))
```

Question 2

```r
#1. Add a new column named overnight_return
#   to store overnight return value

#2. Drop the rows whose overnight_return values are NA
pbpb_data <- mutate(pbpb_data, overnight_return = (prcod - lag(prccd)) / lag(prccd))
pbpb_data <- filter(pbpb_data, !is.na(overnight_return))
```

Question 3

```r
# Add a new column named close_open_change
# to store daily close-open change value

pbpb_data <- mutate(pbpb_data, close_open_change = prccd - prcod)
```

Question 4

```r
# Add a new column named MFV
pbpb_data <- mutate(pbpb_data,
            mfv = ((prccd - prcld) - (prchd - prccd)) * cshtrd / (prchd - prcld))
```

```r
# show the table containing only the new 4 metrics columns
pbpb_data_metrics <- filter(pbpb_data[ ,
            c("datadate", "daily_return", "overnight_return", "close_open_change", "mfv")])

head(pbpb_data_metrics)
```

A data.frame: 6 × 5

	datadate <date>	daily_return <dbl>	overnight_return <dbl>	close_open_change <dbl>	mfv <dbl>
1	2020-09-03	-0.05442177	0.004535147	-0.26	-264658.17
2	2020-09-04	-0.02637890	0.014388489	-0.17	-163452.05
3	2020-09-08	-0.03694581	-0.002463054	-0.14	-284337.00
4	2020-09-09	0.01023018	0.002557545	0.03	11618.32
5	2020-09-10	0.04556962	-0.002531646	0.19	101495.33
6	2020-09-11	-0.04842615	0.007263923	-0.23	-116163.57

Question 5

```r
library(lubridate)
```

```r
# Add a new column named month
pbpb_data <- mutate(pbpb_data, month = month(datadate))
```

Question 6

```r
# Add a new column named year
pbpb_data <- mutate(pbpb_data, year = year(datadate))
```

```
# show the table containing only the new 2 date columns
pbpb_data_dates <- filter(pbpb_data[ , c("datadate", "month", "year")])
head(pbpb_data_dates, 10)
```

A data.frame: 10 × 3

	datadate <date>	month <dbl>	year <dbl>
1	2020-09-03	9	2020
2	2020-09-04	9	2020
3	2020-09-08	9	2020
4	2020-09-09	9	2020
5	2020-09-10	9	2020
6	2020-09-11	9	2020
7	2020-09-14	9	2020
8	2020-09-15	9	2020
9	2020-09-16	9	2020
10	2020-09-17	9	2020

Question 7

```
# Calculate the total trade volume (cshtrd) for June 2023
trade_volume_2023_06 <- filter(pbpb_data, year == 2023 & month == 6)

sum(trade_volume_2023_06$cshtrd)
```

6780601

Question 8

```
# Calculate the mean of daily return over the period
mean_daily_return <- mean(pbpb_data$daily_return)
print(mean_daily_return)
```

```
[1] 0.001263381
```

Question 9

```
# Find the date for maximum high price over the period
max_high_price <- max(pbpb_data$prchd)

date_max_high_price <- filter(pbpb_data, prchd == max_high_price)

print(date_max_high_price$datadate)
```

```
[1] "2023-04-26"
```

Question 10

```
# Find the date for largest daily return over the period
max_daily_return <- max(pbpb_data$daily_return)
date_max_daily_return <- filter(pbpb_data, daily_return == max_daily_return)
print(date_max_daily_return$datadate)
```

```
[1] "2021-03-15"
```