

Group Assignment 1

Simon Yu

2025-10-01

Table of contents

Question 1	3
Question 2	3
Question 3	3
Question 4	3
Question 5	4
Question 6	4
Question 7	5
Question 8	5
Question 9	5
Question 10	6

List of Figures

List of Tables

```
library(dplyr)
```

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

```
# filter data for Potbelly (PBPB)
```

```
pbbp_data <- filter(data, tic == "PBPB")
```

```
# convert datadate to Date type
```

```
pbbp_data$datadate <- as.Date(pbbp_data$datadate, format = "%d/%m/%Y")
```

Question 1

```
#1. Add a new column named daily_return
```

```
pbbp_data <- mutate(pbbp_data, daily_return = (prccd - lag(prccd)) / lag(prccd))
```

```
pbbp_data <- filter(pbbp_data, !is.na(daily_return))
```

Question 2

```
#2. Add a new column named 10-day momentum indicator
```

```
pbbp_data <- mutate(pbbp_data, momentum_10 = prccd - lag(prccd, 10))
```

```
pbbp_data <- filter(pbbp_data, !is.na(momentum_10))
```

Question 3

```
#3. Add a new column named daily_range
```

```
pbbp_data <- mutate(pbbp_data, daily_range = prchd - prcld)
```

Question 4

```
#4. Add a new column named MFV
```

```
pbbp_data <- mutate(pbbp_data,
```

```
mfv = ((prccd - prcld) - (prchd - prccd)) * cshtd / (prchd - prcld))
```

```
# show the table containing only the new 4 metrics columns
pbbp_data_metrics <- filter(pbbp_data[ ,
                             c("datadate", "daily_return", "momentum_10", "daily_range", "mfv")])

head(pbbp_data_metrics, 10)
```

A data.frame: 10 × 5

	datadate <date>	daily_return <dbl>	momen- tum_10 <dbl>	daily_range <dbl>	mfv <dbl>
1	2020-09-17	-0.037470726	-0.30	0.2200	-26952.18
2	2020-09-18	0.041362530	0.11	0.3147	255757.31
3	2020-09-21	-0.072429907	-0.09	0.2100	-84781.00
4	2020-09-22	0.015113350	0.12	0.1200	217511.67
5	2020-09-23	-0.024813896	-0.02	0.1750	-82726.46
6	2020-09-24	-0.035623410	-0.34	0.2400	-54846.67
7	2020-09-25	-0.042216359	-0.30	0.2100	-140455.62
8	2020-09-28	0.049586777	-0.16	0.2650	25997.55
9	2020-09-29	-0.005249344	-0.47	0.2250	75771.11
10	2020-09-30	0.000000000	-0.48	0.2000	-29916.60

Question 5

```
library(lubridate)
```

```
#5. Add a new column named month
pbbp_data <- mutate(pbbp_data, month = month(datadate))
```

Question 6

```
#6. Add a new column named year
pbbp_data <- mutate(pbbp_data, year = year(datadate))
```

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

A data.frame: 10 × 3

	date <date>	month <dbl>	year <dbl>
1	2020-09-17	9	2020
2	2020-09-18	9	2020
3	2020-09-21	9	2020
4	2020-09-22	9	2020
5	2020-09-23	9	2020
6	2020-09-24	9	2020
7	2020-09-25	9	2020
8	2020-09-28	9	2020
9	2020-09-29	9	2020
10	2020-09-30	9	2020

Question 7

```
#7. Calculate the total trade volume (cshtd) for June 2023
trade_volume_2023_06 <- filter(pbp_data, year == 2023 & month == 6)

sum(trade_volume_2023_06$cshtd)
```

6780601

Question 8

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

[1] 0.001311608

Question 9

```
#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

```
#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"
```