

# Assignment 1

Xiang Li

2023-10-10

## Exercise 1

1

Nothing to output.

2

```
data_df = read.csv("data/Coca-Cola_stock_history.csv")
data_df$Date = as.Date(data_df$Date, format = "%Y-%m-%d")
sub_data_df = data_df[(data_df$Date >= "2000-01-01") & (data_df$Date <=
  "2019-12-31"), ]
colnames(sub_data_df)[ncol(sub_data_df)] = "StockSplits"
tail(sub_data_df, 10)
```

##	Date	Open	High	Low	Close	Volume	Dividends
## 14591	2019-12-17	49.87838	50.04342	49.77752	49.89672	9230000	0
## 14592	2019-12-18	49.99757	50.19928	49.41076	49.42910	15132100	0
## 14593	2019-12-19	49.54830	49.82336	49.43827	49.80502	13385900	0
## 14594	2019-12-20	50.37348	50.61188	49.94255	50.40099	23832800	0
## 14595	2019-12-23	50.42850	50.64855	50.28180	50.34598	9300800	0
## 14596	2019-12-24	50.24512	50.42850	50.09842	50.16261	3359300	0
## 14597	2019-12-26	50.35514	50.44683	50.23595	50.44683	6228500	0
## 14598	2019-12-27	50.43767	50.84110	50.42850	50.74941	6895500	0
## 14599	2019-12-30	50.59354	50.78608	50.48352	50.67606	6431700	0
## 14600	2019-12-31	50.61188	50.77692	50.41016	50.74941	7982600	0
##	StockSplits						
## 14591	0						
## 14592	0						
## 14593	0						
## 14594	0						
## 14595	0						
## 14596	0						
## 14597	0						
## 14598	0						
## 14599	0						
## 14600	0						

### 3

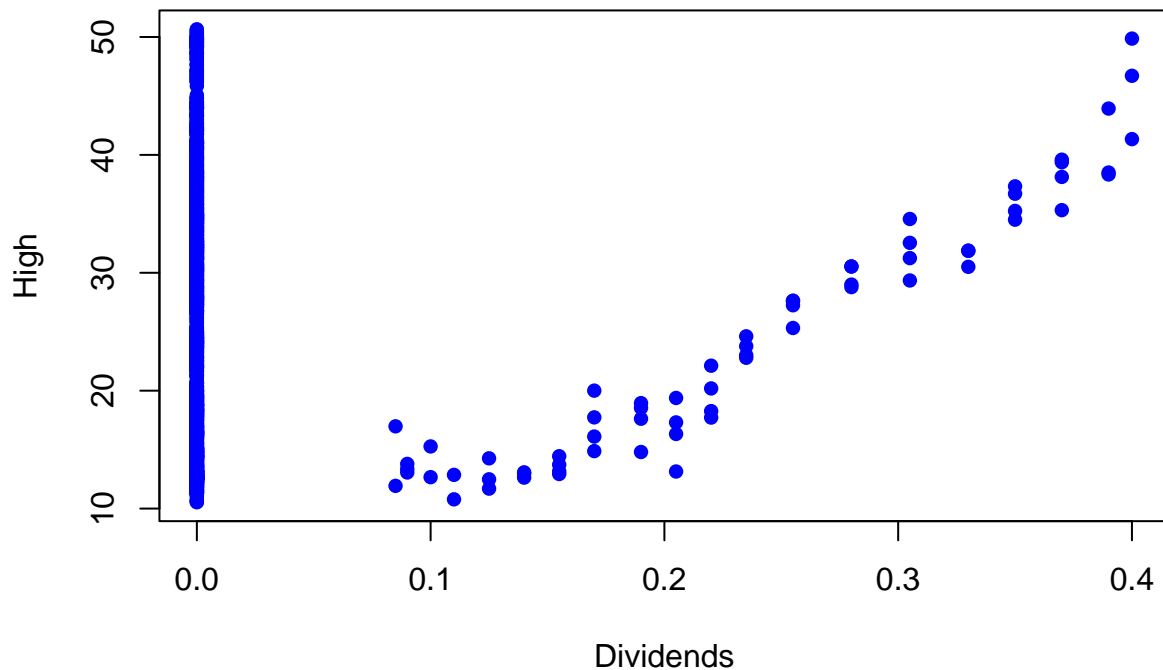
```
sub_data_df$HighVolume = sub_data_df$Volume > 9 * 10^6  
tail(sub_data_df, 10)
```

##	Date	Open	High	Low	Close	Volume	Dividends
## 14591	2019-12-17	49.87838	50.04342	49.77752	49.89672	9230000	0
## 14592	2019-12-18	49.99757	50.19928	49.41076	49.42910	15132100	0
## 14593	2019-12-19	49.54830	49.82336	49.43827	49.80502	13385900	0
## 14594	2019-12-20	50.37348	50.61188	49.94255	50.40099	23832800	0
## 14595	2019-12-23	50.42850	50.64855	50.28180	50.34598	9300800	0
## 14596	2019-12-24	50.24512	50.42850	50.09842	50.16261	3359300	0
## 14597	2019-12-26	50.35514	50.44683	50.23595	50.44683	6228500	0
## 14598	2019-12-27	50.43767	50.84110	50.42850	50.74941	6895500	0
## 14599	2019-12-30	50.59354	50.78608	50.48352	50.67606	6431700	0
## 14600	2019-12-31	50.61188	50.77692	50.41016	50.74941	7982600	0
##	StockSplits	HighVolume					
## 14591	0	TRUE					
## 14592	0	TRUE					
## 14593	0	TRUE					
## 14594	0	TRUE					
## 14595	0	TRUE					
## 14596	0	FALSE					
## 14597	0	FALSE					
## 14598	0	FALSE					
## 14599	0	FALSE					
## 14600	0	FALSE					

### 4

```
temp = sub_data_df[sub_data_df$HighVolume, ]  
plot(temp$Dividends, temp$High, main = "Scatterplot High-Dividends",  
      xlab = "Dividends", ylab = "High", col = "blue", pch = 16)
```

## Scatterplot High–Dividends



If we ignore the points which dividends are equal to 0, the highest price on dividends paid day increases with the increasing of dividends.

Explanation: A company which is able to pay dividends to shareholders generally has a healthy finance. Thus, more investors would like to buy its stock. To improve the opportunity that their buy orders can deal, they will give higher bid price which may improve the highest price of that day.

5

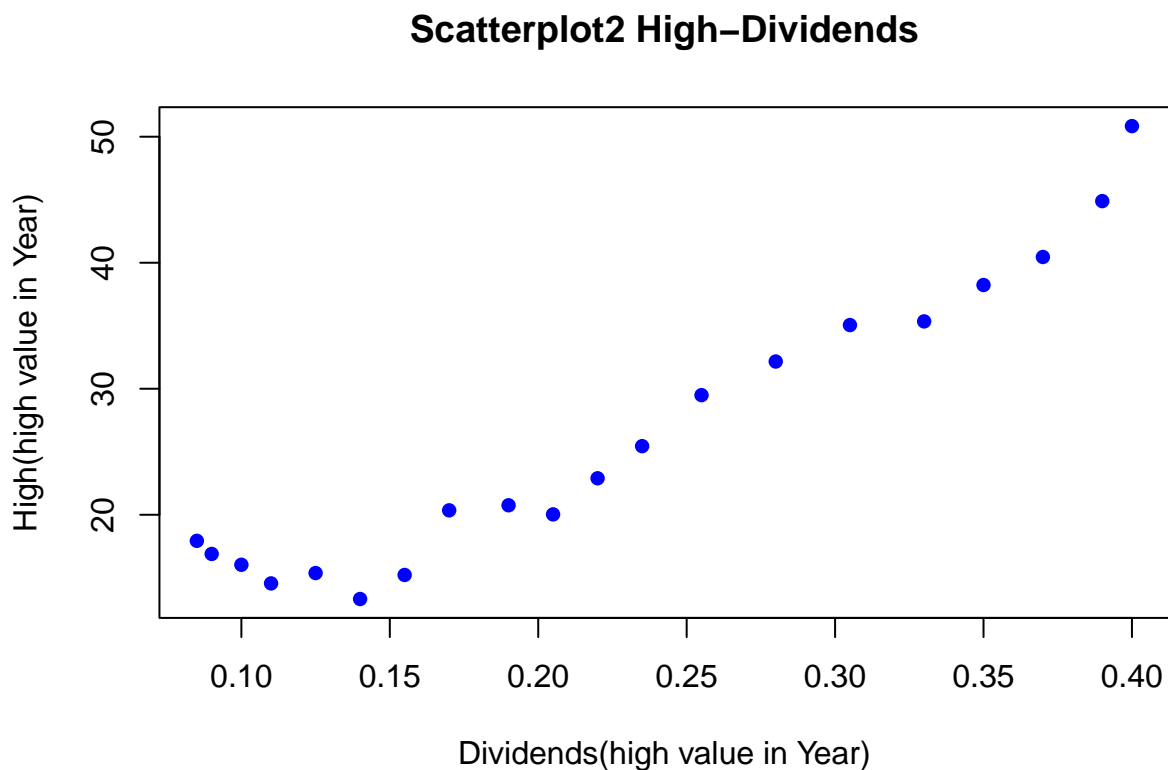
```
sub_data_df$Year = strftime(sub_data_df$Date, "%Y")
num_col = c("Open", "High", "Low", "Close", "Volume", "Dividends",
            "StockSplits")
high_value_df = aggregate(sub_data_df[, num_col], by = list(Year = sub_data_df$Year),
                           max)
high_value_df
```

##	Year	Open	High	Low	Close	Volume	Dividends	StockSplits
## 1	2000	17.89677	17.93028	17.62865	17.93028	27540400	0.085	0
## 2	2001	16.55177	16.89128	16.36503	16.51985	23672000	0.090	0
## 3	2002	15.89098	16.03218	15.80516	15.95742	37382000	0.100	0
## 4	2003	14.45425	14.55433	14.41137	14.51145	31069400	0.110	0
## 5	2004	15.24548	15.37480	15.02995	15.23111	61406200	0.125	0
## 6	2005	13.29848	13.31614	13.20434	13.31320	39747600	0.140	0
## 7	2006	15.15908	15.22386	15.06036	15.11589	53627400	0.155	0
## 8	2007	20.24963	20.35087	20.16736	20.27810	50341600	0.170	0

## 9	2008	20.68626	20.75270	20.47427	20.74321	79214400	0.190	0
## 10	2009	19.92781	20.02889	19.85033	19.91433	124169000	0.205	0
## 11	2010	22.77954	22.89421	22.75173	22.85598	63648600	0.220	0
## 12	2011	25.33793	25.44443	25.21014	25.28469	50096800	0.235	0
## 13	2012	29.37658	29.49261	29.23515	29.41647	98967500	0.255	2
## 14	2013	31.87937	32.16078	31.68684	31.90900	33575600	0.280	0
## 15	2014	34.60277	35.05459	34.45476	34.92216	55953500	0.305	0
## 16	2015	35.26362	35.34412	35.01409	35.28778	44065300	0.330	0
## 17	2016	38.04524	38.23181	37.72076	38.03712	40360300	0.350	0
## 18	2017	40.31787	40.45420	39.85777	40.41160	37526500	0.370	0
## 19	2018	44.81049	44.88933	44.30789	44.79157	32477700	0.390	0
## 20	2019	50.61188	50.84110	50.48352	50.74941	58905400	0.400	0

6

```
plot(high_value_df$Dividends, high_value_df$High, main = "Scatterplot2 High-Dividends",
     xlab = "Dividends(high value in Year)", ylab = "High(high value in Year)",
     col = "blue", pch = 16)
```



The positive-related relationship between dividends and high price is more clear.

Missing information: The dividends values of most samples are equal to 0, but they didn't show in this plot.

```

num_col = c("Open", "High", "Low", "Close", "Volume", "Dividends",
            "StockSplits")
result_ls = list()
for (c in num_col) {
  c_min = min(sub_data_df[, c])
  c_min_date = as.character(sub_data_df$Date[which.min(sub_data_df[,
    c])])
  c_max = max(sub_data_df[, c])
  result_ls = c(result_ls, list(c(c, c_min, c_min_date, c_max -
    c_min)))
}
names(result_ls) = num_col
result_ls

```

```

## $Open
## [1] "Open"          "10.39588929" "2003-03-11"  "40.21598925"
##
## $High
## [1] "High"          "10.53599404" "2003-03-10"  "40.30510497"
##
## $Low
## [1] "Low"           "10.37066808" "2003-03-05"  "40.1128473"
##
## $Close
## [1] "Close"         "10.38748169" "2003-03-10"  "40.36192703"
##
## $Volume
## [1] "Volume"        "2147400"      "2003-12-26"  "122021600"
##
## $Dividends
## [1] "Dividends"     "0"            "2000-01-03"  "0.4"
##
## $StockSplits
## [1] "StockSplits"   "0"            "2000-01-03"  "2"

```

## Exercise 2

```

mybeerbottlefunction = function(num) {
  if ((num <= 0) | (length(num) > 1) | num%%1 != 0) {
    stop("The input is not available!!!!")
  } else {
    lyric = paste0(num, " Bottles of Beers Song\n\n\n")
    loop_v = rev(1:num)
    for (n in loop_v) {
      if (n > 2) {
        lyric = paste0(lyric, n, " bottles of beer on the wall, ",
          n, " bottles of beer. Take one down, pass it around, ",
          n - 1, " bottles of beer on the wall...\n\n\n")
      }
    }
  }
}

```

```

    } else if (n == 1) {
        lyric = paste0(lyric, "1 bottle of beer on the wall, 1 bottle of beer. ",
            "Take one down, pass it around, no more bottles of beer on the ",
            "wall.")
    } else if (n == 2) {
        lyric = paste0(lyric, "2 bottles of beer on the wall, 2 bottles of beer. ",
            "Take one down, pass it around, 1 bottle of beer on the wall...\n\n\n")
    }
    }
    return(lyric)
}
}
# cat(mybeerbottlefunction(5))
writeLines(strwrap(mybeerbottlefunction(5), width = 100))

```

```

## 5 Bottles of Beers Song
##
## 5 bottles of beer on the wall, 5 bottles of beer. Take one down, pass it around, 4 bottles of beer
## on the wall...
##
## 4 bottles of beer on the wall, 4 bottles of beer. Take one down, pass it around, 3 bottles of beer
## on the wall...
##
## 3 bottles of beer on the wall, 3 bottles of beer. Take one down, pass it around, 2 bottles of beer
## on the wall...
##
## 2 bottles of beer on the wall, 2 bottles of beer. Take one down, pass it around, 1 bottle of beer
## on the wall...
##
## 1 bottle of beer on the wall, 1 bottle of beer. Take one down, pass it around, no more bottles of
## beer on the wall.

```