Coding2:Term Project

Shah Ali Gardezi

12-21-2021

The goal of this term project is to scrape all of information presented on CSIMarkets website about stock market of 500 companies arranged alphabetically. The information is then to be added in a dataframe which will then be stored into RDS.

Please refer to *webscraper.r* R-script for the code.

Web Scraping Methodology

1. First of all given web link was modified such that we can get a link for each table in alphabetical order. Each table had 20 entries so first 26 tables were shortlisted.

```
# url = paste0("https://csimarket.com/markets/Stocks.php?days=yy&pageA=",page,"#tablecomp2")
```

2. All the hrefs (web links) from the given website link were scrapped using the below function. The list of the links scrapped was filtered to keep only the ones which were required (companies in alphabatical order).

```
# page <- read_html(url) #just read the html once
#     web <- page %>%
#       html_nodes("table") %>% html_nodes("tr") %>% html_nodes("a") %>%
#       html_attr("href")
```

3. The list of links was used to scrape each companies profile.
4. Scraped company's name using

```
# df = read_html(url1)
# CompanyName = df %>%
#     html_nodes(xpath = '//*[contains(concat( " ", @class, " " ), concat( " ", "Naziv", " "
#   Company = CompanyName
```

5. For tables the entire web page was scraped. Later those tables were analysed and the ones which met our interest were filtered.

```
# tables_list = df %>%
#     html_nodes("table")%>%
#     html_table(fill= TRUE)
```

6. Got industry and sector using

```
#   t3 = tables_list[10]
#   t3 = as.data.frame(t3)
#   t3$X1 = str_replace_all(t3$X1, "[^[:alnum:]]", " ")
#   Industry = unlist(strsplit(t3$X1[1] , "                    "))[2]
#   Sector = unlist(strsplit(t3$X1[2] , "          "))[2]
#   ColNames = c("Company","Industry","Sector")
#   val = c(Company,Industry,Sector)
```

7. Scrapped first table required using

```
#   t1 = tables_list[13]
#   t1 = as.data.frame(t1)
#   ColNames1 = gsub(":", "", t1$X1)
#   ColNames2 = gsub(":", "", t1$X3)
#   val1 =  t1$X2
#   val2 =  t1$X4
```

8. Scrapped second table required using

```
  # t2 = tables_list[15]
  # t2 = as.data.frame(t2)
  # t2$X1 = gsub("[\r\n]", "", t2$X1)
  # t2$X1 = gsub("    ", "", t2$X1)
  # ColNames3 = t2$X1
  # val3 =  t2$X2
```

9. Combined all the scrapped data and made a data frame

```
  # Names = c(ColNames,ColNames1,ColNames2,ColNames3)
  # Number = c(val,val1,val2,val3)
  #
  # datadf = data.frame(Names,Number)
  #
  # datadf = dcast(datadf, Number ~ Names)
  # datadf = datadf[1,c(1:20)]
  # names(datadf) = Names
  # datadf[1,] = Number
  # datadf
```

10. The above steps from 4 to 9 were repeated for each company
11. All the data frames were combined using rbind().
12. the data frame was saved as a rds file using

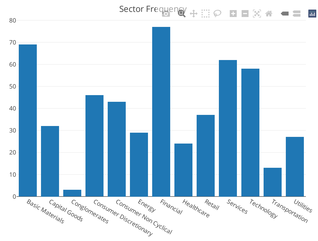
```
# saveRDS(dataFrame, file="DataFrame.rds")
```

Reading RDS file and processing the data for visualization

```
df2 <- readRDS("DataFrame.rds")
df2$Sector = as.factor(df2$Sector)
df2$Industry = as.factor(df2$Industry)
df2$Employees = as.numeric(gsub(",","",df2$Employees))
df2$`Net Income (TTM) (Millions $)` = as.numeric(gsub(",","",df2$`Net Income (TTM) (Millio
```

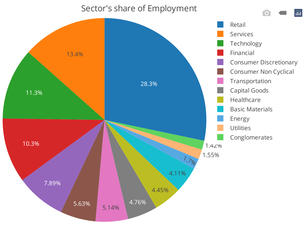
Plot 1 : Sector Frequency

```
fig <- plot_ly(x = df2$Sector, type = "histogram")%>%
  layout(title = 'Sector Frequency')
fig
```



Plot 2: Sector's share of Employment

```
fig <- plot_ly(df2, labels = ~Sector, values = ~Employees, type = 'pie')%>%
  layout(title = "Sector's share of Employment")
fig
```



Plot 3 : "Sectorwise Net Income spread"

```
fig <- plot_ly(df2, y = ~`Net Income (TTM) (Millions $)`, color = ~Sector, type = "box")%>%
  layout(title = "Sectorwise Net Income spread")

fig
```