8. Unsupervised learning

Hair Albeiro Parra Barrera

Libraries

```
## here() starts at C:/Users/jairp/OneDrive/Desktop_remote/HEC Montreal/2. Fall 2023/Machine Learning Applied

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

## Loading required package: xts

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

##
## ############################ Warning from 'xts' package ###########################
## #                                                                                 #
## # The dplyr lag() function breaks how base R's lag() function is supposed to  #
## # work, which breaks lag(my_xts). Calls to lag(my_xts) that you type or       #
## # source() into this session won't work correctly.                            #
## #                                                                                 #
## # Use stats::lag() to make sure you're not using dplyr::lag(), or you can add #
## # conflictRules('dplyr', exclude = 'lag') to your .Rprofile to stop           #
## # dplyr from breaking base R's lag() function.                                #
## #                                                                                 #
## # Code in packages is not affected. It's protected by R's namespace mechanism #
## # Set `options(xts.warn_dplyr_breaks_lag = FALSE)` to suppress this warning.  #
## #                                                                                 #
## ###################################################################################

##
## Attaching package: 'xts'

## The following objects are masked from 'package:dplyr':
##
##     first, last
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v readr     2.1.4
## v ggplot2   3.4.3     v stringr   1.5.0
## v lubridate 1.9.2     v tibble    3.2.1
## v purrr     1.0.1     v tidyr     1.3.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter()         masks stats::filter()
## x xts::first()            masks dplyr::first()
## x readr::guess_encoding() masks rvest::guess_encoding()
## x dplyr::lag()            masks stats::lag()
## x xts::last()             masks dplyr::last()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
## Loading required package: PerformanceAnalytics
##
##
## Attaching package: 'PerformanceAnalytics'
##
##
## The following object is masked from 'package:graphics':
##
##     legend
```

0. Scraping the SP500

In order to test the logic within the strategy, I have fetched functions that retrieve a number of sample stocks by sector from the SP500.

```
# to obtain relative paths
library(here)

# Load code into environment
source(here("functions", "fetch_sp500_sectors.R"))
```

0.0.1 SP500 Economic Sectors

The following function fetches and extract the economic sectors from the SP500, taken from Wikipedia.

```
# fetch the sectors as a dataframe
sp500_sectors <- f_get_sp500_sectors()
head(sp500_sectors)
```

```
##   tickers                sectors
## 1     MMM             Industrials
## 2     AOS             Industrials
## 3     ABT             Health Care
## 4    ABBV             Health Care
## 5     ACN Information Technology
## 6    ATVI Communication Services
```

0.0.2 SP500 Sector Weight

```
# Load the required packages
library(tidyverse)
library(tidyquant)

# Get the SP500 index data from Yahoo Finance
sp500 <- tq_index("SP500")
```

Getting holdings for SP500

```
# wrap into a single argument funciton
fetch_sp500_sector_data <- function(x){f_fetch_sector_data(x, sp500, sp500_sectors)}

# call the function
fetch_sp500_sector_data("Information Technology")
```

```
##    ticker                 sector       weight shares_held
## 1    AAPL Information Technology 0.0702540411   162587331
## 2     ACN Information Technology 0.0054092441     6978653
## 3    ADBE Information Technology 0.0064376791     5042633
## 4     ADI Information Technology 0.0023957535     5547304
## 5    ADSK Information Technology 0.0011986744     2364476
## 6    AKAM Information Technology 0.0004527619     1688579
## 7    AMAT Information Technology 0.0031202204     9290330
## 8     AMD Information Technology 0.0042684461    17876854
## 9    ANET Information Technology 0.0012258229     2774214
## 10   ANSS Information Technology 0.0007174977      960263
## 11    APH Information Technology 0.0013639386     6587672
## 12   AVGO Information Technology 0.0091687667     4565580
## 13   CDAY Information Technology 0.0002769393     1689245
## 14   CDNS Information Technology 0.0017075778     3006994
## 15    CDW Information Technology 0.0007554639     1483072
## 16    CRM Information Technology 0.0055845083    10775563
## 17   CSCO Information Technology 0.0059741894    45083323
## 18   CTSH Information Technology 0.0009606982     5587355
## 19    DXC Information Technology 0.0001149851     2269880
## 20   ENPH Information Technology 0.0004673880     1506871
## 21   EPAM Information Technology 0.0004091228      632680
## 22   FFIV Information Technology 0.0002566800      655960
## 23   FICO Information Technology 0.0006079364      275000
## 24   FSLR Information Technology 0.0004881657     1182003
## 25   FTNT Information Technology 0.0010480163     7176766
## 26    GEN Information Technology 0.0002935494     6225452
## 27    GLW Information Technology 0.0006537789     8439997
## 28    HPE Information Technology 0.0006021637    14274684
## 29    HPQ Information Technology 0.0006380162     9582733
## 30    IBM Information Technology 0.0036908088    10080291
## 31   INTC Information Technology 0.0039858438    46302774
## 32   INTU Information Technology 0.0038822246     3098282
## 33     IT Information Technology 0.0007570690      872668
## 34   JNPR Information Technology 0.0002447159     3552262
## 35   KEYS Information Technology 0.0006460521     1985130
## 36   KLAC Information Technology 0.0016793525     1512524
## 37   LRCX Information Technology 0.0022362320     1474686
## 38   MCHP Information Technology 0.0011383852     6022129
## 39   MPWR Information Technology 0.0005781856      528479
## 40   MSFT Information Technology 0.0652498121    82197285
## 41    MSI Information Technology 0.0012857532     1853861
## 42     MU Information Technology 0.0020407883    12098141
```

```
## 43    NOW Information Technology 0.0030810379     2256813
## 44   NTAP Information Technology 0.0004397708     2332269
## 45   NVDA Information Technology 0.0278453531    27326163
## 46   NXPI Information Technology 0.0013745668     2852032
## 47     ON Information Technology 0.0010793822     4778149
## 48   ORCL Information Technology 0.0047348769    17416570
## 49   PANW Information Technology 0.0019119448     3383747
## 50   PAYC Information Technology 0.0003498207      538084
## 51    PTC Information Technology 0.0004631308     1314763
## 52   QCOM Information Technology 0.0033150958    12360118
## 53   QRVO Information Technology 0.0002544268     1083162
## 54    ROP Information Technology 0.0014453544     1173832
## 55   SEDG Information Technology 0.0002098106      619661
## 56   SNPS Information Technology 0.0018579069     1683323
## 57    STX Information Technology 0.0003499503     2152206
## 58   SWKS Information Technology 0.0004229214     1768127
## 59    TDY Information Technology 0.0005351846      519669
## 60    TEL Information Technology 0.0010632406     3473282
## 61    TER Information Technology 0.0004047260     1703925
## 62   TRMB Information Technology 0.0003472683     2748428
## 63    TXN Information Technology 0.0040029226    10045284
## 64    TYL Information Technology 0.0004400400      460939
## 65   VRSN Information Technology 0.0004939882      992567
## 66    WDC Information Technology 0.0003908576     3540219
## 67   ZBRA Information Technology 0.0003244866      571159
```

0.0.3 Retrieving top sectors and stocks

Pack everything into one function to retrieve all the data

```
# Retrieve top 10 stocks by weight for each sector in the top 5 sectors from the SP500 (by weight)
sector_list <- f_retrieve_top_sp500(top_n_sectors = 6, top_n_stocks = 10, only_tickers=TRUE)
sector_list
```

```
## $Industrials
##  [1] "ADP" "BA"  "CAT" "DE"  "GE"  "HON" "LMT" "RTX" "UNP" "UPS"
##
## $`Health Care`
##  [1] "ABBV" "ABT"  "AMGN" "DHR"  "JNJ"  "LLY"  "MRK"  "PFE"  "TMO"  "UNH"
##
## $`Information Technology`
##  [1] "AAPL" "ACN"  "ADBE" "AMD"  "AVGO" "CRM"  "CSCO" "MSFT" "NVDA" "ORCL"
##
## $`Communication Services`
##  [1] "ATVI"  "CMCSA" "DIS"   "GOOG"  "GOOGL" "META"  "NFLX"  "T"     "TMUS"
## [10] "VZ"
##
## $Financials
##  [1] "BAC"   "BRK-B" "GS"    "JPM"   "MA"    "MMC"   "MS"    "SPGI"  "V"
## [10] "WFC"
##
## $`Consumer Discretionary`
##  [1] "ABNB" "AMZN" "BKNG" "HD"   "LOW"  "MCD"  "NKE"  "SBUX" "TJX"  "TSLA"
```

This logic is implemented under **functions/fetch_sp500_sectors.R**

0.0.4 Retrieving top sectors and stocks

```
# function to fetch all the information for one ticker into a nice xts dataframe
sp500_stocks <- lapply(sector_list, f_fetch_all_tickers, start_date="2018-01-01", end_date="2022-12-01")

# update format so that it becomes a named list of lists
sp500_stocks <- lapply(sp500_stocks, function(sector_l){
  setNames(sector_l$stock_data, sector_l$tickers)
  }
)
```

```
# Show the available sectors
names(sp500_stocks)
```

```
## [1] "Industrials"            "Health Care"            "Information Technology"
## [4] "Communication Services" "Financials"             "Consumer Discretionary"
```

```
# Show available stocks for Industrials
names(sp500_stocks$Industrials)
```

```
##  [1] "ADP" "BA"  "CAT" "DE"  "GE"  "HON" "LMT" "RTX" "UNP" "UPS"
```

```
# access the xts of the stocks in industrials
sp500_stocks$Industrials$ADP
```

```
##            ADP.Open ADP.High ADP.Low ADP.Close ADP.Volume ADP.Adjusted
## 2018-01-02   116.03   116.45  115.25    115.99    2453000      102.9319
## 2018-01-03   116.18   117.70  115.60    117.25    1521500      104.0500
## 2018-01-04   117.63   118.90  117.47    118.37    1236900      105.0440
## 2018-01-05   118.55   118.77  117.26    118.30    1451100      104.9818
## 2018-01-08   118.37   118.58  117.39    117.94    2683000      104.6624
## 2018-01-09   117.74   119.03  117.50    118.76    2317200      105.3901
## 2018-01-10   118.28   118.40  117.05    117.65    2037000      104.4050
## 2018-01-11   117.76   117.78  116.28    117.18    1157000      103.9879
## 2018-01-12   117.81   118.93  117.34    118.47    1644300      105.1327
## 2018-01-16   118.52   119.79  118.16    119.39    2099200      105.9492
##        ...
## 2023-09-11   249.28   249.89  246.89    248.20    1297200      248.2000
## 2023-09-12   247.05   248.68  246.90    248.02    1303300      248.0200
## 2023-09-13   247.44   249.13  246.95    247.81    1364400      247.8100
## 2023-09-14   248.43   248.74  246.35    248.29    1350200      248.2900
## 2023-09-15   248.44   248.91  244.69    245.31    2898800      245.3100
## 2023-09-18   246.16   248.14  245.70    247.28    1218500      247.2800
## 2023-09-19   246.44   247.05  243.97    245.84    1030700      245.8400
## 2023-09-20   247.21   247.21  243.80    243.87    1364700      243.8700
## 2023-09-21   242.63   243.09  238.62    238.72    1260100      238.7200
## 2023-09-22   237.62   240.93  237.62    239.35    1026175      239.3500
```