

# Case (1 of 2): An Overview of the S&P500

Chapter 17.

## S&P 500

The S&P 500, also called the Standard & Poor's 500, is a stock market index that tracks the performance of 500 major publicly traded companies listed on U.S. stock exchanges. It serves as a widely accepted benchmark for assessing the overall health and performance of the U.S. stock market.

S&P Dow Jones Indices, a division of S&P Global, is responsible for maintaining the index. The selection of companies included in the S&P 500 is determined by a committee, considering factors such as market capitalization, liquidity, and industry representation.

The S&P is a float-weighted index, meaning the market capitalizations of the companies in the index are adjusted by the number of shares available for public trading. [1]

The performance of the S&P 500 is frequently used to gauge the broader stock market and is commonly referenced by investors, analysts, and financial media. It provides a snapshot of how large-cap U.S. stocks are faring and is considered a reliable indicator of overall market sentiment.

Aside: Typically, the S&P 500 index consists of 500 stocks. However, in reality, there are actually 503 stocks included. This discrepancy arises because three of the listed companies have multiple share classes, and each class is considered a separate stock that needs to be included in the index. [1]

### Strengths:

1. **Diverse Representation:** The S&P 500 isn't fixated on a single industry. From technology to healthcare, it offers a panoramic view of various economic sectors, making it an inclusive representation of the U.S. corporate sector.
2. **Benchmark for Investors:** For many fund managers, outperforming the S&P 500 stands as a golden standard. It's a yardstick, establishing it as a critical touchstone for gauging investment success.

3. **Liquidity and Visibility:** Constituent companies enjoy high liquidity and are subject to rigorous screening processes, ensuring that the index represents financially viable entities.

#### **Critiques:**

1. **Market Capitalization Weighting:** The index is weighted by market capitalization, meaning companies with higher market values have a more pronounced effect on its performance. Critics argue this approach can skew perceptions, especially during market bubbles when certain sectors are overvalued.
2. **Exclusivity:** Despite its broad purview, 500 companies cannot encapsulate the entire U.S. economy. Many sectors, especially emerging industries or smaller businesses, might not be adequately represented.
3. **Potential for Complacency:** The prominence of the S&P 500 has led many investors to adopt passive investment strategies, tracking the index rather than actively managing portfolios. Detractors argue this might lead to market inefficiencies and reduced capital allocation efficacy.

While the S&P 500 remains an influential and pivotal tool for investors, its dominance prompts a double-edged sword of advantages and critiques. In a constantly evolving economic landscape, understanding both its power and limitations is essential for informed financial decision-making. [2]

The broad purpose of this Case Study is to review and analyze the different sectors and stocks within the S&P500.

## **S&P 500 Data**

**Load some useful R packages**

```
# Load the required libraries, suppressing annoying startup messages
library(dplyr, quietly = TRUE, warn.conflicts = FALSE) # For data manipulation
library(tibble, quietly = TRUE, warn.conflicts = FALSE) # For data manipulation
library(ggplot2, quietly = TRUE, warn.conflicts = FALSE) # For data visualization
library(ggpubr, quietly = TRUE, warn.conflicts = FALSE) # For data visualization

library(gsheet, quietly = TRUE, warn.conflicts = FALSE) # For Google Sheets
library(rmarkdown, quietly = TRUE, warn.conflicts = FALSE) # For writing
library(knitr, quietly = TRUE, warn.conflicts = FALSE) # For tables
library(kableExtra, quietly = TRUE, warn.conflicts = FALSE) # For tables
library(scales) # For formatting currency
```

## Read the S&P500 data from a Google Sheet into a tibble

1. We will analyze a real-world, recent dataset containing information about the S&P500 stocks, sourced from TradingView.com. [3]
2. The dataset is located in a Google Sheet and periodically updated.
3. The complete URL of the Google Sheet that has the data is  
<https://docs.google.com/spreadsheets/d/14mU1NNpeuV2RouT9MKaAWKUpvjRijzQu40DdWJgyKPQ/>
4. Its Google Sheet ID is: 14mU1NNpeuV2RouT9MKaAWKUpvjRijzQu40DdWJgyKPQ.

## Loading the data into R

1. We can use the function `gsheet2tbl` in package `gsheet` to read the Google Sheet into a tibble, as demonstrated in the following code.

```
# Read S&P500 stock data present in a Google Sheet.
library(gsheet)
prefix <- "https://docs.google.com/spreadsheets/d/"
sheetID <- "14mU1NNpeuV2RouT9MKaAWKUpvjRijzQu40DdWJgyKPQ"
url500 <- paste(prefix,sheetID) # Form the URL to connect to
sp500Data <- gsheet2tbl(url500) # Read it into a tibble called sp500Data
```

2. **Note:** This data is current, as of **Fri, Jan 5, 2024**

## S&P Global Industry Classification Standard (GICS®)

1. In this case study, we will classify and analyze the S&P 500 stocks based on the GICS standard!
2. The Global Industry Classification Standard (GICS®) was developed in 1999 by S&P Dow Jones Indices and MSCI. The GICS methodology aims to enhance the investment research and asset management process for financial professionals worldwide. The GICS methodology has been widely accepted as an industry analysis framework for investment research, portfolio management and asset allocation. [4]
3. The GICS classification consists of **11** sectors, – {Communication Services, Consumer Discretionary, Consumer Staples, Energy, Financials, Health Care, Industrials, Information Technology, Materials, Real Estate, Utilities}. The classification of each stock in the S&P 500 according to GICS is available at the following Google Sheet:

[https://docs.google.com/spreadsheets/d/1WrVA8dPYvQsc\\_mXVctgTntRLS02qd7ubzcdAsw03Lgk/](https://docs.google.com/spreadsheets/d/1WrVA8dPYvQsc_mXVctgTntRLS02qd7ubzcdAsw03Lgk/)

4. For this file, the Google Sheet ID is 1WrVA8dPYvQsc\_mXVctgTntRLS02qd7ubzcdAsw03Lgk and we read this classification data into a tibble, we name `gics`, using similar code.

```
# Read GICS classificaiton of S&P 500 stocks from a Google Sheet.
library(gsheet)
prefix2 <- "https://docs.google.com/spreadsheets/d/"
sheetID2 <- "1WrVA8dPYvQsc_mXVctgTntRLS02qd7ubzcdAsw03Lgk"
urlgics <- paste(prefix2, sheetID2) # Form the URL to connect to
gics <- gsheets2tbl(urlgics) # Read it into a tibble called gics
```

5. Next, we join the two tibbles, using “Stock” as the key and name our joint tibble `sp500`, as follows.

```
# Merging dataframes
sp500 <- merge(sp500Data,
              gics ,
              id = "Stock")
```

## Review the S&P 500 data

1. The data corresponds to **503** companies that are part of the S&P500 and includes 39 data columns, as of **Fri, Jan 5, 2024**

```
dim(sp500)
```

```
[1] 503 39
```

2. The first ten stocks in the S&P500 data, their GICS Sector and their recent prices are as follows:

```
sp500 %>%  
  select(Stock, Description, GICSSector) %>%  
  head(10) %>%  
  kable("html", caption = "The first 10 companies in the S&P500 dataset") %>%  
  kable_styling()
```

Table 0.1: The first 10 companies in the S&P500 dataset

Stock	Description	GICSSector
A	Agilent Technologies, Inc.	Health Care
AAL	American Airlines Group, Inc.	Industrials
AAPL	Apple Inc.	Information Technology
ABBV	AbbVie Inc.	Health Care
ABNB	Airbnb, Inc.	Consumer Discretionary
ABT	Abbott Laboratories	Health Care
ACGL	Arch Capital Group Ltd.	Financials
ACN	Accenture plc	Information Technology
ADBE	Adobe Inc.	Information Technology
ADI	Analog Devices, Inc.	Information Technology

### 3. Data Columns

- The data comprises of the following 39 columns:

```
colnames(sp500)
```

```
[1] "Stock"  
[2] "Date"  
[3] "Description"  
[4] "Sector"  
[5] "Industry"  
[6] "Market Capitalization"
```

[7] "Price"  
 [8] "52 Week Low"  
 [9] "52 Week High"  
 [10] "Return on Equity (TTM)"  
 [11] "Return on Assets (TTM)"  
 [12] "Return on Invested Capital (TTM)"  
 [13] "Gross Margin (TTM)"  
 [14] "Operating Margin (TTM)"  
 [15] "Net Margin (TTM)"  
 [16] "Price to Earnings Ratio (TTM)"  
 [17] "Price to Book (FY)"  
 [18] "Enterprise Value/EBITDA (TTM)"  
 [19] "EBITDA (TTM)"  
 [20] "EPS Diluted (TTM)"  
 [21] "EBITDA (TTM YoY Growth)"  
 [22] "EBITDA (Quarterly YoY Growth)"  
 [23] "EPS Diluted (TTM YoY Growth)"  
 [24] "EPS Diluted (Quarterly YoY Growth)"  
 [25] "Price to Free Cash Flow (TTM)"  
 [26] "Free Cash Flow (TTM YoY Growth)"  
 [27] "Free Cash Flow (Quarterly YoY Growth)"  
 [28] "Debt to Equity Ratio (MRQ)"  
 [29] "Current Ratio (MRQ)"  
 [30] "Quick Ratio (MRQ)"  
 [31] "Dividend Yield Forward"  
 [32] "Dividends per share (Annual YoY Growth)"  
 [33] "Price to Sales (FY)"  
 [34] "Revenue (TTM YoY Growth)"  
 [35] "Revenue (Quarterly YoY Growth)"  
 [36] "Technical Rating"  
 [37] "Security"  
 [38] "GICSSector"  
 [39] "GICSSubIndustry"

- The names of the data columns are self-explanatory. The Financial terms are explained in depth on multiple external websites such as [www.Investopedia.com](http://www.Investopedia.com)

## Rename Data Columns

4. The names of the data columns are lengthy and confusing. We will rename the data columns to make it easier to work with the data.

```

# Define a mapping of new column names
new_names <- c(
  "Stock", "Date", "StockName", "Sector", "Industry",
  "MarketCap", "Price", "Low52Wk", "High52Wk",
  "ROE", "ROA", "ROIC", "GrossMargin",
  "OperatingMargin", "NetMargin", "PE",
  "PB", "EVEBITDA", "EBITDA", "EPS",
  "EBITDA_YOY", "EBITDA_QYOY", "EPS_YOY",
  "EPS_QYOY", "PFCF", "FCF",
  "FCF_QYOY", "DebtToEquity", "CurrentRatio",
  "QuickRatio", "DividendYield",
  "DividendsPerShare_YOY", "PS",
  "Revenue_YOY", "Revenue_QYOY", "Rating",
  "Security", "GICSSector", "GICSSubIndustry"
)
# Rename the columns using the new_names vector
colnames(sp500)<-new_names

```

5. We review the column names again after renaming them, using the `colnames()` function.

```
colnames(sp500)
```

```

[1] "Stock"           "Date"           "StockName"
[4] "Sector"         "Industry"       "MarketCap"
[7] "Price"          "Low52Wk"       "High52Wk"
[10] "ROE"           "ROA"           "ROIC"
[13] "GrossMargin"   "OperatingMargin" "NetMargin"
[16] "PE"            "PB"            "EVEBITDA"
[19] "EBITDA"        "EPS"           "EBITDA_YOY"
[22] "EBITDA_QYOY"  "EPS_YOY"       "EPS_QYOY"
[25] "PFCF"          "FCF"           "FCF_QYOY"
[28] "DebtToEquity" "CurrentRatio"  "QuickRatio"
[31] "DividendYield" "DividendsPerShare_YOY" "PS"
[34] "Revenue_YOY"  "Revenue_QYOY"  "Rating"
[37] "Security"      "GICSSector"    "GICSSubIndustry"

```

## Understand the Data Columns

6. Our next goal is to gain a deeper understanding of what the data columns mean. We reorganize the column names into eight tables, labeled Table 1a, 1b.. 1h.

- a. The column names described in Table 1a. concern basic **Company Information** of each stock.

---

Table 1a: Data Columns giving basic Company Information

ColumnName	Description
Stock	Stock Ticker (e.g. AAL)
Date	Date (e.g. "7/15/2023")
StockName	Name of the company (e.g. "American Airlines Group, Inc.")
GICSSector	Sector, as per GICS Classification
GICSSubIndustry	Sub-Industry, as per GICS Classification
MarketCap	Market capitalization of the company
Price	Recent Stock Price

---

- b. The column names described in Table 1b. are related to **Technical Analysis**, including the 52-Week High and Low prices.

---

Table 1b: Data Columns related to Pricing and Technical Analysis

ColumnName	Description
Low52Wk	52-Week Low Price
High52Wk	52-Week High Price
Rating	Technical Rating

---

- c. The column names described in Table 1c. are related to the **Profitability** of each stock.

---

Table 1c: Data Columns related to Profitability

ColumnName	Description
ROE	Return on Equity
ROA	Return on Assets
ROIC	Return on Invested Capital
GrossMargin	Gross Profit Margin
OperatingMargin	Operating Profit Margin
NetMargin	Net Profit Margin

---

- d. The column names described in Table 1d are related to the **Earnings** of each stock.

---

Table 1d: Data Columns related to Earnings

ColumnName	Description
PE	Price-to-Earnings Ratio

---



---

Table 1d: Data Columns related to Earnings

ColumnName	Description
PB	Price-to-Book Ratio
EVEBITDA	Enterprise Value to EBITDA Ratio
EBITDA	EBITDA
EPS	Earnings per Share
EBITDA_YOY	EBITDA Year-over-Year Growth
EBITDA_QYOY	EBITDA Quarterly Year-over-Year Growth
EPS_YOY	EPS Year-over-Year Growth
EPS_QYOY	EPS Quarterly Year-over-Year Growth

---

e. The column names described in Table 1e are related to the **Free Cash Flow** of each stock.

---

Table 1e: Data Columns related to Free Cash Flow

ColumnName	Description
PFCF	Price-to-Free Cash Flow
FCF	Free Cash Flow
FCF_QYOY	Free Cash Flow Quarterly Year-over-Year Growth

---

f. The column names described in Table 1f concern the **Liquidity** of each stock.

---

Table 1f: Data Columns related to Liquidity

ColumnName	Description
DebtToEquity	Debt-to-Equity Ratio
CurrentRatio	Current Ratio
QuickRatio	Quick Ratio

---

g. The column names described in Table 1g are related to the **Revenue** of each stock.

---

Table 1g: Data Columns related to Revenue

ColumnName	Description
PS	Price-to-Sales Ratio
Revenue_YOY	Revenue Year-over-Year Growth
Revenue_QYOY	Revenue Quarterly Year-over-Year Growth

---

h. The column names described in Table 1h are related to the **Dividends** of each stock.

Table 1h: Data Columns related to Dividends

ColumnName	Description
DividendYield	Dividend Yield
DividendsPerShare__YOY	Annual Dividends per Share Year-over-Year Growth

## Stock Prices, 52-Week Low, High; Market Cap in Billions

We want to analyze stock prices relative to their 52 Week Low and 52 Week High respectively, to understand their relative price attractiveness.

Hence, a new column named `Low52WkPerc` is being added. The column contains the percentage change between the current price (`Price`) and its 52-week low (`Low52Wk`). The formula used is:

$$Low52WkPerc = \frac{(CurrentPrice - 52WeekLow) * 100}{52WeekLow}$$

Another column named `High52WkPerc` represents the percentage change between the 52-week high (`High52Wk`) and the current price (`Price`). We round off the data to two decimal places for clarity.

```
library(dplyr)
sp500 <- sp500 %>%
  mutate(Low52WkPerc = round((Price - Low52Wk) * 100 / Low52Wk, 2),
         High52WkPerc = round((High52Wk - Price) * 100 / High52Wk, 2),
         MarketCapBillions = round(MarketCap / 1e9, 3) # Convert MarketCap to billions
  )
```

For convenience, we format the Prices.

```
library(dplyr)
library(scales) # For formatting currency

sp500 <- sp500 %>%
  mutate(
    Price = scales::dollar(round(Price, 2)), # format the Price as a dollar amount
    High52Wk = scales::dollar(round(High52Wk, 2)), # format the 52 Week High
    Low52Wk = scales::dollar(round(Low52Wk, 2)) # format the 52 Week Low
  )
```

## Analysis of Stock Ratings

1. In the data, the S&P500 shares have Technical Ratings such as {Strong Buy, Buy, Neutral, Sell, Strong Sell}. Since each Stock has a unique Technical Rating, it makes sense to model the data column Rating as a `factor()` variable.

```
sp500$Rating <- as.factor(sp500$Rating)
```

2. We confirm that Rating is now modelled as a factor variable, using `str()` and use `levels()` to review the different levels it can take.

```
str(sp500$Rating)
```

```
Factor w/ 5 levels "Buy","Neutral",...: 1 1 3 4 1 3 3 4 2 3 ...
```

```
levels(sp500$Rating)
```

```
[1] "Buy"          "Neutral"      "Sell"         "Strong Buy"   "Strong Sell"
```

3. The `table()` function allows us to count how many stocks have each Rating. see how many stocks have ratings ranging from “Strong Sell” to “Strong Buy”. This completes our review of Rating.

```
table(sp500$Rating)
```

Buy	Neutral	Sell	Strong Buy	Strong Sell
192	51	178	58	24

## Analysis of GICS Sectors in the S&P500

- The S&P 500 comprises a wide array of sectors, reflecting the diverse American corporate landscape.
- The data showcases the S&P500 divided across 11 Sectors. Each stock belongs to a unique sector and it makes sense to model `GICSSector` as a `factor`.

```
sp500$GICSSector <- as.factor(sp500$GICSSector)
```

2. We confirm that `GICSSector` is now modelled as a factor variable and review the different levels it can take.

```
str(sp500$GICSSector)
```

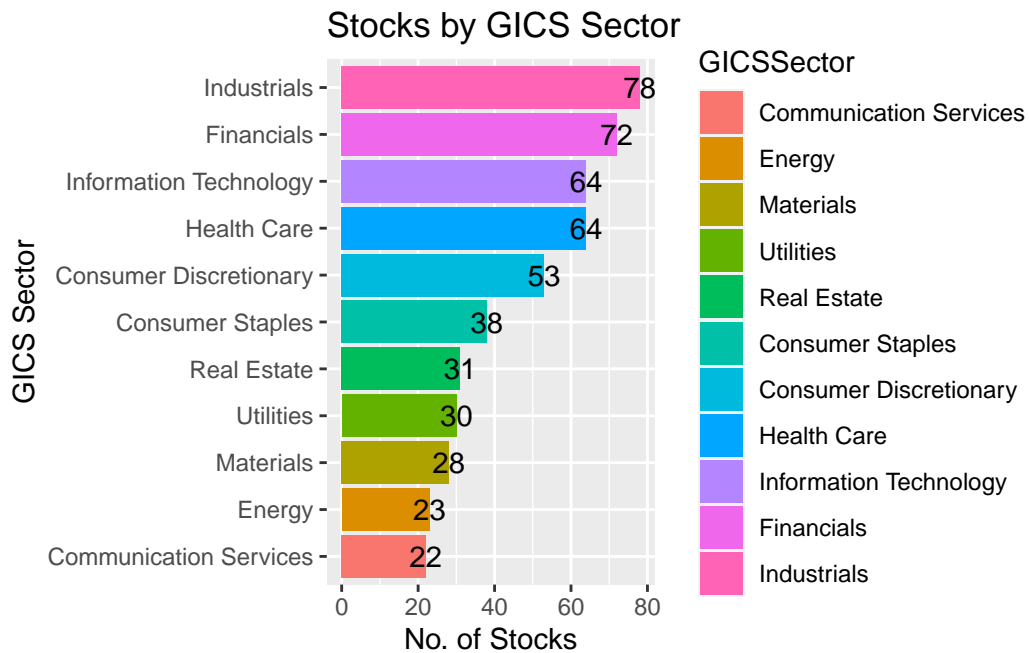
```
Factor w/ 11 levels "Communication Services",...: 6 7 8 6 2 6 5 8 8 8 ...
```

```
levels(sp500$GICSSector)
```

```
[1] "Communication Services" "Consumer Discretionary" "Consumer Staples"  
[4] "Energy"                "Financials"            "Health Care"  
[7] "Industrials"           "Information Technology" "Materials"  
[10] "Real Estate"           "Utilities"
```

- We note that the S&P500 consists of 503 stocks, divided across 11 sectors.

```
library(ggplot2) # For creating plots  
library(dplyr)   # For data manipulation  
  
sp500 %>%  
  mutate(  
    # Reorder the 'GICSSector' factor levels based on the count of each sector  
    # 'table(GICSSector)[GICSSector]' calculates the frequency of each sector  
    # 'reorder' reorders the levels of 'GICSSector' based on these frequencies  
    GICSSector = reorder(GICSSector,  
                        table(GICSSector)[GICSSector])  
  ) %>%  
  # Start a ggplot with 'GICSSector' on the y-axis  
  ggplot(aes(y = GICSSector)) +  
    # Create a bar plot; 'geom_bar' counts the frequency for each sector  
    # 'fill = GICSSector' colors the bars based on the sector  
    geom_bar(aes(fill = GICSSector)) +  
    # Add text labels on the bars showing the count of stocks in each sector  
    # 'stat = "count"' calculates the count for each sector  
    # 'label = after_stat(count)' adds these counts as labels on the bars  
    geom_text(stat = 'count',  
             aes(label = after_stat(count))) +  
  labs(title = "Stocks by GICS Sector", # Title of the plot  
       x = "No. of Stocks",           # Label for the x-axis  
       y = "GICS Sector")             # Label for the y-axis
```



- Thus, we can see how many stocks are part of each sector. We can sum them to confirm that they add up to 503 stocks.

## MarketCap by GICS Sector

1. We review the Market Cap of S&P500 stocks across GICS Sectors. We summarize the total Market Cap for each GICS Sector, using the following code.

```
# Calculate Market Cap by Sector

MarketCapbySector <- sp500 %>%
  mutate(Market_Cap_Billions = round(MarketCap / 1000000000, 2)) %>%
  group_by(GICSSector) %>%
  summarise(MarketCapBillions = sum(Market_Cap_Billions, na.rm = TRUE)) %>%
  arrange(-MarketCapBillions)
```

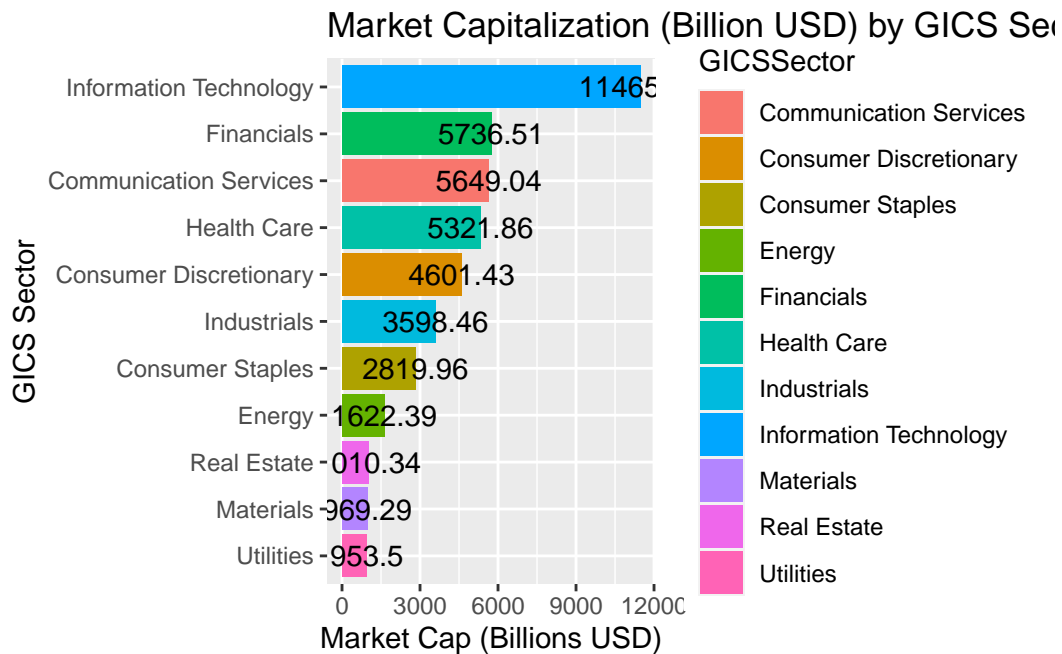
2. We create a bar plot of Market Cap by GICS Sector

```
# Create a bar plot of Market Cap by GICS Sector,
ggplot(MarketCapbySector,
  aes(y = reorder(GICSSector,
    MarketCapBillions), # Y-axis: GICSSector reordered
```

```

x = MarketCapBillions, # X-axis: Market Capitalization in billions
fill = GICSSector)) + # Fill color of the bars based on GICSSector
geom_bar(stat = "identity") + # 'stat = "identity"' to use MarketCapBillions for bars
labs(title = "Market Capitalization (Billion USD) by GICS Sector",
y = "GICS Sector", # Label for the y-axis
x = "Market Cap (Billions USD)") + # Label for the x-axis
geom_text(aes(label = MarketCapBillions)) # Add text labels to the bars

```



3. The S&P500 has a combined Market Cap of 43748.27 Billion USD.

### Highest Market Cap Stocks in each GICS Sector

1. Suppose we wanted to find the top two stocks with the highest market capitalization in each GICS Sector.
2. We could group the data by **GICSSector**; arrange the data in descending order of **MarketCap** within each sector; slice the top 2 entries for each group. Here's the R code to accomplish this:

```

# Find the top two stocks by MarketCap in each GICS Sector
top_stocks_by_sector <- sp500 %>%
  group_by(GICSSector) %>%

```

```

arrange(desc(MarketCap)) %>%
slice_head(n = 2) %>%
ungroup() %>%
arrange(GICSSector) # Arrange the final data by GICSSector

# Select only the specified columns and create a table using kable
top_stocks_by_sector %>%
  select(GICSSector, Stock, StockName, MarketCapBillions) %>%
  kable("html", caption = "Top Two Stocks by Market Capitalization, for each GICS Sector")
  kable_styling()

```

Table 0.10: Top Two Stocks by Market Capitalization, for each GICS Sector

GICSSector	Stock	StockName	MarketCapBillions
Communication Services	GOOG	Alphabet Inc.	1723.840
Communication Services	GOOGL	Alphabet Inc.	1723.430
Consumer Discretionary	AMZN	Amazon.com, Inc.	1504.950
Consumer Discretionary	TSLA	Tesla, Inc.	751.020
Consumer Staples	WMT	Walmart Inc.	421.765
Consumer Staples	PG	Procter & Gamble Company (The)	349.243
Energy	XOM	Exxon Mobil Corporation	412.872
Energy	CVX	Chevron Corporation	284.572
Financials	BRK.B	Berkshire Hathaway Inc. New	796.444
Financials	V	Visa Inc.	523.588
Health Care	LLY	Eli Lilly and Company	583.748
Health Care	UNH	UnitedHealth Group Incorporated	499.797
Industrials	BA	Boeing Company (The)	149.284
Industrials	CAT	Caterpillar, Inc.	147.054
Information Technology	AAPL	Apple Inc.	2831.530
Information Technology	MSFT	Microsoft Corporation	2750.200
Materials	LIN	Linde plc	198.572
Materials	SHW	Sherwin-Williams Company (The)	76.078
Real Estate	PLD	Prologis, Inc.	123.424
Real Estate	AMT	American Tower Corporation (REIT)	100.440
Utilities	NEE	NextEra Energy, Inc.	126.878
Utilities	SO	Southern Company (The)	78.045

## Prices relative to 52-Week-Low and 52-Week-High of each GICS Sector

We analyze the distribution of stock prices relative to their 52 Week Low, measured as  $\text{Low52WkPerc} = (\text{Price} - \text{52-Week-Low}) / (\text{52-Week-Low})$ . In fact, we review this distribution for each GICS Sector, to understand which sector has stocks priced relatively closest to their 52-Week Low prices.

We also do this for stock prices relative to their 52 Week High, measured as  $\text{High52WkPerc} = (\text{52-Week-High} - \text{Price}) / (\text{52-Week-High})$ .

Table 2

```
library(dplyr)
library(kableExtra)

# Calculate Q1, Median, and Q3 for Low52WkPerc and High52WkPerc within each GICS sector
sector_summary_stats <- sp500 %>%
  group_by(GICSSector) %>%
  summarise(
    Low_Q1 = round(quantile(Low52WkPerc, 0.25, na.rm = TRUE), 1),
    Low_Median = round(median(Low52WkPerc, na.rm = TRUE), 1),
    Low_Q3 = round(quantile(Low52WkPerc, 0.75, na.rm = TRUE), 1),
    High_Q1 = round(quantile(High52WkPerc, 0.25, na.rm = TRUE), 1),
    High_Median = round(median(High52WkPerc, na.rm = TRUE), 1),
    High_Q3 = round(quantile(High52WkPerc, 0.75, na.rm = TRUE), 1)
  ) %>%
  ungroup()

# Sort by Low Q1
sector_summary_stats <- sector_summary_stats %>%
  arrange(Low_Q1)

# Reapply formatting with percentage symbol
sector_summary_stats <- sector_summary_stats %>%
  mutate(
    across(ends_with("Q1"), ~paste0(., "%")),
    across(ends_with("Median"), ~paste0(., "%")),
    across(ends_with("Q3"), ~paste0(., "%"))
  )

# Create a formatted table using kable
sector_summary_stats %>%
  kable("html",
```



```
caption = "Q1, Median, and Q3 of Low52WkPerc and High52WkPerc by GICS Sector, Sorted by Low Q1"
kable_styling(bootstrap_options = c("striped", "hover", "condensed", "responsive"))
```

Table 0.11: Q1, Median, and Q3 of Low52WkPerc and High52WkPerc by GICS Sector, Sorted by Low Q1

GICSSector	Low_Q1	Low_Median	Low_Q3	High_Q1	High_Median	High_Q3
Consumer Staples	9.5%	18%	22.1%	9.2%	15.5%	24%
Energy	13.9%	27.4%	30.2%	7.1%	12.7%	16.1%
Utilities	14.9%	18.2%	22.1%	7.3%	9.5%	17%
Consumer Discretionary	17.4%	30.4%	47.9%	4.3%	12.7%	21.6%
Real Estate	19.6%	27.8%	36.1%	5.5%	8.1%	14.6%
Communication Services	19.9%	30.1%	56.5%	4.1%	10.8%	18.9%
Materials	19.9%	27.5%	35.1%	4.6%	7.7%	17.3%
Industrials	20.4%	29.5%	42%	3.9%	6.7%	10.8%
Health Care	21.1%	29.7%	39.7%	3.9%	11.2%	19.3%
Financials	25.1%	34%	41.5%	1.8%	7.4%	14.7%
Information Technology	25.6%	37.6%	56.5%	5.8%	8.4%	13.4%

Notice that Low\_Q1 represents the 25th percentile of the distribution of Low52WkPerc, while High\_Q3 represents the 75th percentile of the distribution of High52WkPerc.

We want to identify the GICS Sector(s) whose stocks are closest to their 52-Week Low and relatively furthest from their 52-Week High prices. One way of doing this is to create a scatterplot, as follows.

```
library(ggplot2)
library(dplyr)

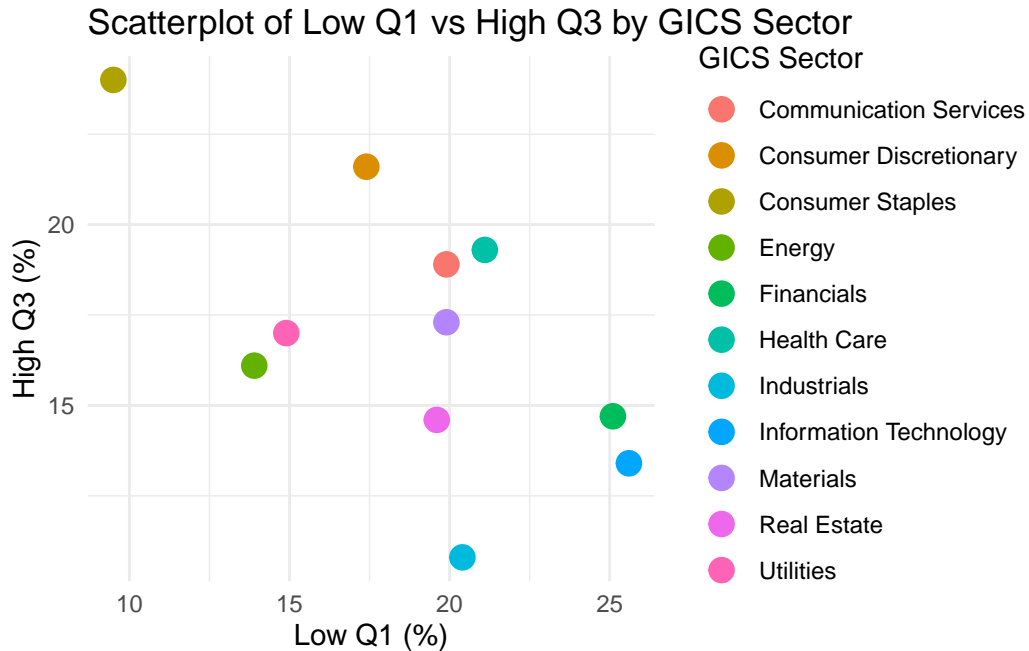
# Convert percentage strings back to numeric for plotting
plot_data <- sector_summary_stats %>%
  mutate(
    Low_Q1_numeric = as.numeric(gsub("%", "", Low_Q1)),
    High_Q3_numeric = as.numeric(gsub("%", "", High_Q3))
  )

# Create a scatterplot with larger circle sizes
ggplot(plot_data, aes(x = Low_Q1_numeric, y = High_Q3_numeric, color = GICSSector)) +
  geom_point(size = 4) + # Increase the size of the points
  labs(
```

```

title = "Scatterplot of Low Q1 vs High Q3 by GICS Sector",
x = "Low Q1 (%)",
y = "High Q3 (%)",
color = "GICS Sector"
) +
theme_minimal() # Use a minimal theme for better aesthetics

```



- **This scatterplot suggests that the Consumer Staples sector is closest to 52-Week Low and furthest from 52-Week High.**

Next, we can do the reverse to determine which Sector is furthest away from 52-Week Low and closest to 52-Week High prices.

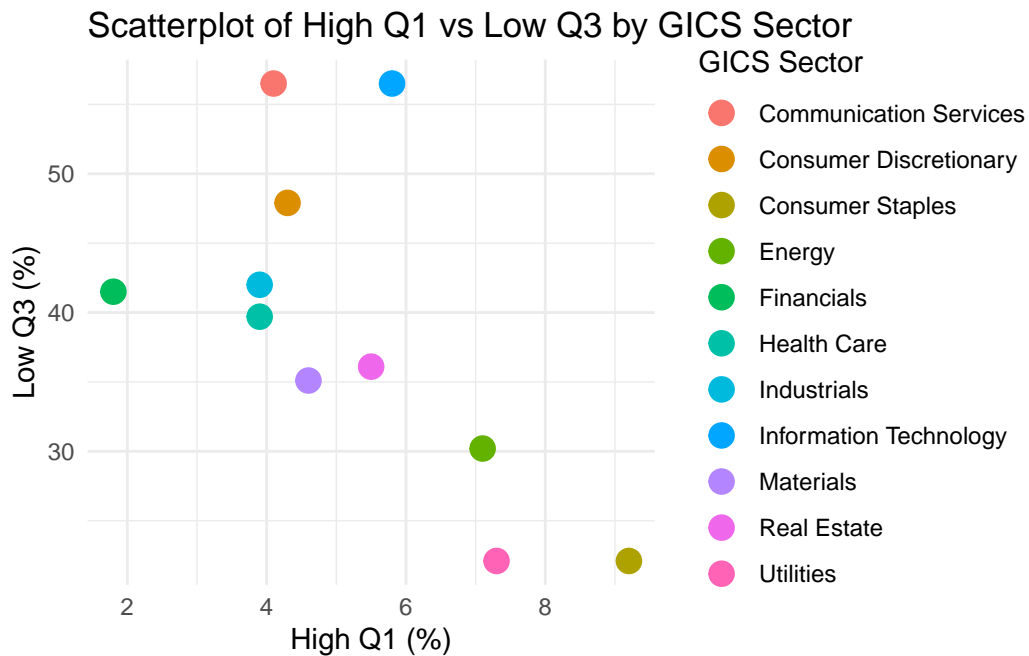
```

library(ggplot2)
library(dplyr)

# Convert percentage strings back to numeric for plotting
plot_data <- sector_summary_stats %>%
mutate(
  High_Q1_numeric = as.numeric(gsub("%", "", High_Q1)),
  Low_Q3_numeric = as.numeric(gsub("%", "", Low_Q3))
)

```

```
# Create a scatterplot with larger circle sizes
ggplot(plot_data, aes(x = High_Q1_numeric, y = Low_Q3_numeric, color = GICSSector)) +
  geom_point(size = 4) + # Increase the size of the points
  labs(
    title = "Scatterplot of High Q1 vs Low Q3 by GICS Sector",
    x = "High Q1 (%)",
    y = "Low Q3 (%)",
    color = "GICS Sector"
  ) +
  theme_minimal() # Use a minimal theme for better aesthetics
```



- This scatterplot suggests that the Communication Services sector is closest to 52-Week Low and furthest from 52-Week High.

### Profitability of each GICS Sector

The following table gives us the Median of different profitability metrics {ROE, ROA, ROIC, OPM, NPM}, for each GICS sector.

```
library(dplyr)
library(kableExtra)
```

```

# Calculate the median values for Profitability metrics
median_financials <- sp500 %>%
  group_by(GICSSector) %>%
  summarise(
    Med_ROE = round(median(ROE, na.rm = TRUE), 1),
    Med_ROA = round(median(ROA, na.rm = TRUE), 1),
    Med_ROIC = round(median(ROIC, na.rm = TRUE), 1),
    Med_OPM = round(median(OperatingMargin, na.rm = TRUE), 1),
    Med_NPM = round(median(NetMargin, na.rm = TRUE), 1)
  ) %>%
  arrange(desc(Med_ROE)) # Sort the table based on Median ROE in descending order

# Create a formatted table using kable
median_financials %>%
  kable("html",
        caption = "Median Financial Metrics by GICS Sector") %>%
  kable_styling()

```

Table 0.12: Median Financial Metrics by GICS Sector

GICSSector	Med_ROE	Med_ROA	Med_ROIC	Med_OPM	Med_NPM
Information Technology	25.9	10.8	16.0	21.6	17.5
Consumer Discretionary	24.5	12.5	18.1	13.8	9.6
Industrials	23.4	8.3	13.0	16.5	11.2
Energy	22.7	11.1	15.2	25.4	16.2
Consumer Staples	21.3	6.5	9.9	14.6	8.2
Materials	16.5	5.7	9.1	13.9	8.5
Financials	14.5	2.2	8.4	23.1	17.0
Communication Services	12.9	4.7	5.7	17.7	8.6
Health Care	12.6	5.7	8.6	15.1	10.0
Utilities	9.4	2.4	3.7	19.9	11.6
Real Estate	7.2	3.4	3.9	31.2	19.7

- We notice that the Information Technology GICS Sector has the relatively largest profitability numbers.
- We also note that the Consumer Staples has fairly good profitability numbers as well, relative to all the other sectors.

## Earnings of each GICS Sector

We review the distribution of Price to Earnings (PE) and Price to Book (PB) ratios, across different sectors. Obviously, the smaller the PE and PB ratio, the better it is.

We also review how the Earnings Per Share have increased or decreased across the different GICS sectors.

```
library(dplyr)
library(kableExtra)

# Assuming 'sp500' is your dataset and it includes the columns PE, PB, EPS_QYOY, and EPS_YOY

# Calculate the median values for Earnings related metrics
median_earnings <- sp500 %>%
  group_by(GICSSector) %>%
  summarise(
    Med_PE = round(median(PE, na.rm = TRUE), 1),
    Med_PB = round(median(PB, na.rm = TRUE), 1),
    Med_EPS_QYOY = round(median(EPS_QYOY, na.rm = TRUE), 1),
    Med_EPS_YOY = round(median(EPS_YOY, na.rm = TRUE), 1)
  ) %>%
  ungroup() # Removed the sorting to keep the table unsorted

# Create a formatted table using kable
median_earnings %>%
  kable("html",
        caption = "Median Earnings Metrics (PE, PB, EPS_QYOY, EPS_YOY) by GICS Sector") %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed", "responsive"))
```

Table 0.13: Median Earnings Metrics (PE, PB, EPS\_QYOY, EPS\_YOY) by GICS Sector

GICSSector	Med_PE	Med_PB	Med_EPS_QYOY	Med_EPS_YOY
Communication Services	25.7	2.0	20.1	2.1
Consumer Discretionary	21.9	3.7	14.0	11.3
Consumer Staples	23.4	4.4	7.6	-1.4
Energy	11.1	2.5	-26.5	-16.0
Financials	15.3	2.0	9.8	4.5
Health Care	29.5	3.7	0.1	0.5
Industrials	26.0	5.3	11.2	12.5
Information Technology	30.6	6.4	4.4	8.6
Materials	21.7	2.7	-27.4	-29.4
Real Estate	41.6	2.3	-17.1	-20.4

GICSSector	Med_PE	Med_PB	Med_EPS_QYOY	Med_EPS_YOY
Utilities	19.2	1.9	5.5	-0.9

- **We conclude that from a Price to Earnings ratio perspective, the PE ratio is relatively lowest for the Energy sector.**
- **From an Earnings per Share Growth perspective, the Consumer Discretionary GICS sector has performed the best.**

Thus, in this survey, we have analyzed the distribution of prices and also the distribution of profitability and earnings growth metrics for the different GICS sector. With this, we conclude our brief descriptive survey of the S&P500 stocks.

## Summary of Chapter 17 – Case (1 of 2): An Overview of the S&P500

Chapter Summary: S&P 500 Case Study

This chapter presents a comprehensive case study of the S&P 500, a crucial index in the U.S. stock market, comprising 500 major publicly traded companies. The chapter begins by introducing the S&P 500, including its management by S&P Dow Jones Indices and its significance as a benchmark for U.S. stock market health. Key strengths of the S&P 500 are highlighted, such as its diverse representation across various economic sectors and its role as a benchmark for investors. However, the chapter also discusses critiques, including the potential for market misrepresentation due to its market capitalization weighting and the issue of exclusivity, as it doesn't fully represent the entire U.S. economy.

The chapter then delves into a detailed data analysis using R programming, starting with loading necessary R packages and reading S&P 500 data from a Google Sheet. The data, sourced from TradingView.com, is organized into a tibble and classified according to the Global Industry Classification Standard (GICS). Further, the chapter categorizes data columns into various aspects like company information, profitability, earnings, and others. It also includes renaming and understanding these data columns for more accessible analysis.

Significant analyses in the chapter include: 1. Stock Prices Analysis: Comparison of stock prices relative to their 52-Week Low and High. 2. GICS Sectors Analysis: Examination of the distribution of stocks and market capitalization across different GICS sectors. 3. Profitability and Earnings Analysis: Evaluation of the profitability and earnings of companies in each GICS sector.

Each analysis is supported with R code snippets and explanations, providing a clear understanding of the methodologies and results. The chapter concludes by summarizing insights

from the analyses, such as sector-wise profitability and earnings growth, offering a nuanced understanding of the S&P 500 index's composition and performance.

This chapter serves as an in-depth review and analysis of the S&P 500, blending theoretical knowledge with practical data analysis skills, making it a valuable resource for those interested in financial markets and data analytics.

## References

### S&P 500

[1] <https://www.investopedia.com/terms/s/sp500.asp>

[2] S&P Global: S&P Global. (n.d.). S&P 500. Retrieved September 14, 2023, from <https://www.spglobal.com/spdji/en/indices/equity/sp-500/>

MarketWatch: MarketWatch. (n.d.). S&P 500 Index. Retrieved September 14, 2023, from <https://www.marketwatch.com/investing/index/spx>

Bloomberg: Bloomberg. (n.d.). S&P 500 Index (SPX:IND). Retrieved September 14, 2023, from <https://www.bloomberg.com/quote/SPX:IND>

[3] TradingView.com <https://www.tradingview.com/screener/>

[4] GICS: Global Industry Classification Standard: <https://www.spglobal.com/spdji/en/landing/topic/gics/>