

# FM\_Intro1

18.642

2024-09-04

## Contents

<b>1. Collect historical data</b>	<b>2</b>
1.1 Set start and end date for collection . . . . .	2
1.2 Collect time series data from Yahoo . . . . .	2
1.3 Collect time series data from FRED . . . . .	2
<b>2. Select/Filter/Merge datasets</b>	<b>3</b>
2.1 Select/Filter datasets . . . . .	3
2.2 Merge index_data1, stock_data1, and economic_data1 . . . . .	3
<b>3. Plot time series</b>	<b>4</b>
3.1 S&P 500 Index . . . . .	4
3.2 VIX Index . . . . .	5
3.3 Plot both together . . . . .	6
3.4 ARKK . . . . .	7
3.5 GME . . . . .	8
3.6 Bitcoin . . . . .	9
3.7 Nvidia . . . . .	10
3.8 Bond Yields on the same graph . . . . .	11
3.9 Crude Oil Futures . . . . .	12
<b>4. Save R workspace with all time series</b>	<b>13</b>

## 1. Collect historical data

### 1.1 Set start and end date for collection

```
date_start <- "2011-01-01"  
date_end <- "2024-08-31"
```

### 1.2 Collect time series data from Yahoo

```
#      Collect index_data for S&P 500 and VIX indexes ----  
#      Source: http://finance.yahoo.com  
#  
index_data <- tq_get(c("^GSPC", "^VIX"),  
                      get = "stock.prices",  
                      from = date_start,  
                      to = date_end)  
  
#      Collect stock_data for stocks ----  
  
#      Source: http://finance.yahoo.com  
stock_data <- tq_get(c("NVDA",  
                      "GE",  
                      "AAPL",  
                      "GOOG",  
                      "AMZN",  
                      "XOM",  
                      "GME",  
                      "ARKK",  
                      "AMC",  
                      "BTC-USD",  
                      "XLF"),  
                      get = "stock.prices",  
                      from = date_start,  
                      to = date_end)
```

### 1.3 Collect time series data from FRED

```
# Source: St. Louis Federal Reserve  
##          http://research.stlouisfed.org/fred2/  
#  
# Series name / Description  
# -----  
# SP500      / SP500 Stock market index  
# VIXCLS     / Vix volatility index  
#  
# DGS3MO     / 3-Month Treasury, constant maturity rate  
# DGS1       / 1-Year Treasury, constant maturity rate  
# DGS5       / 5-Year Treasury, constant maturity rate  
# DGS10      / 10-Year Treasury, constant maturity rate  
#  
# DAAA        / Moody's Seasoned Aaa Corporate Bond Yield  
# DBAA        / Moody's Seasoned Baa Corporate Bond Yield  
#
```

```

# DCOILWTICO / Crude Oil Prices: West Texas Intermediate (WTI) - Cushing, Oklahoma
# CBBTCUSD / Coinbase Bitcoin
economic_data <- tq_get(c(
    "SP500",
    "VIXCLS",
    "DGS3MO",
    "DGS1",
    "DGS5",
    "DGS10",
    "DAAA",
    "DBAA",
    "DCOILWTICO",
    "CBBTCUSD"),
  get = "economic.data",
  from = date_start,
  to = date_end)

```

## 2. Select/Filter/Merge datasets

### 2.1 Select/Filter datasets

The following code creates the object market\_data. For each dataset:

- 1. Select three columns (symbol, date, price)
- 2. Filter out/remove NA observations
- 3. Rename adjusted to price (for stocks/indexes)---

```

## 2.1 Select three columns (symbol, date, price)
## 2.2 Rename adjusted to price (for stocks/indexes)-----
## 2.3 Remove NA observations

index_data1 <- index_data %>%
  dplyr::select(symbol, date, adjusted) %>%
  dplyr::filter(!is.na(adjusted)) %>%
  dplyr::rename(price = adjusted)

stock_data1 <- stock_data %>%
  dplyr::select(symbol, date, adjusted) %>%
  dplyr::filter(!is.na(adjusted)) %>%
  dplyr::rename(price = adjusted)

# economic_data has same 3 columns (no select/rename needed)
economic_data1 <- economic_data %>%
  dplyr::filter(!is.na(price))

```

### 2.2 Merge index\_data1, stock\_data1, and economic\_data1

```

market_data <- index_data1 %>%
  full_join(stock_data1) %>%
  full_join(economic_data1)

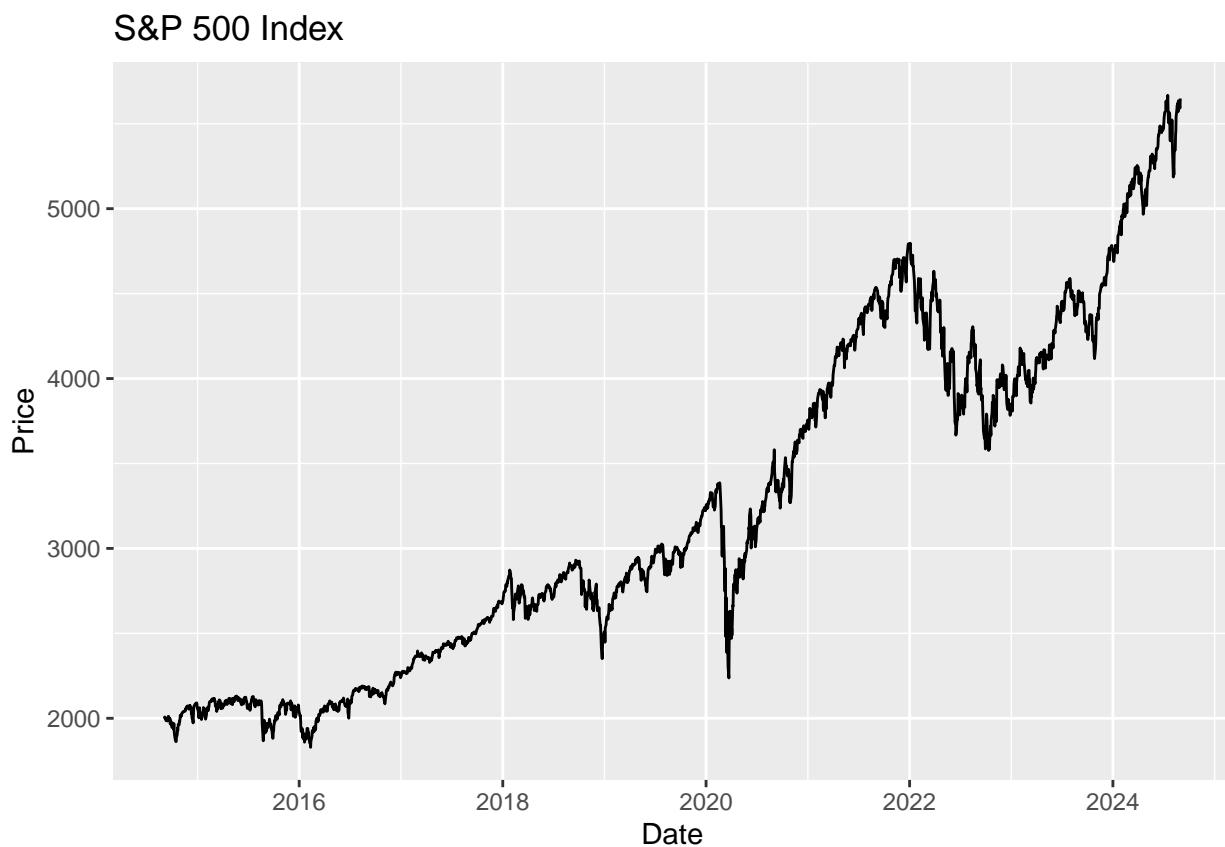
## Joining with `by = join_by(symbol, date, price)`
## Joining with `by = join_by(symbol, date, price)`

```

### 3. Plot time series

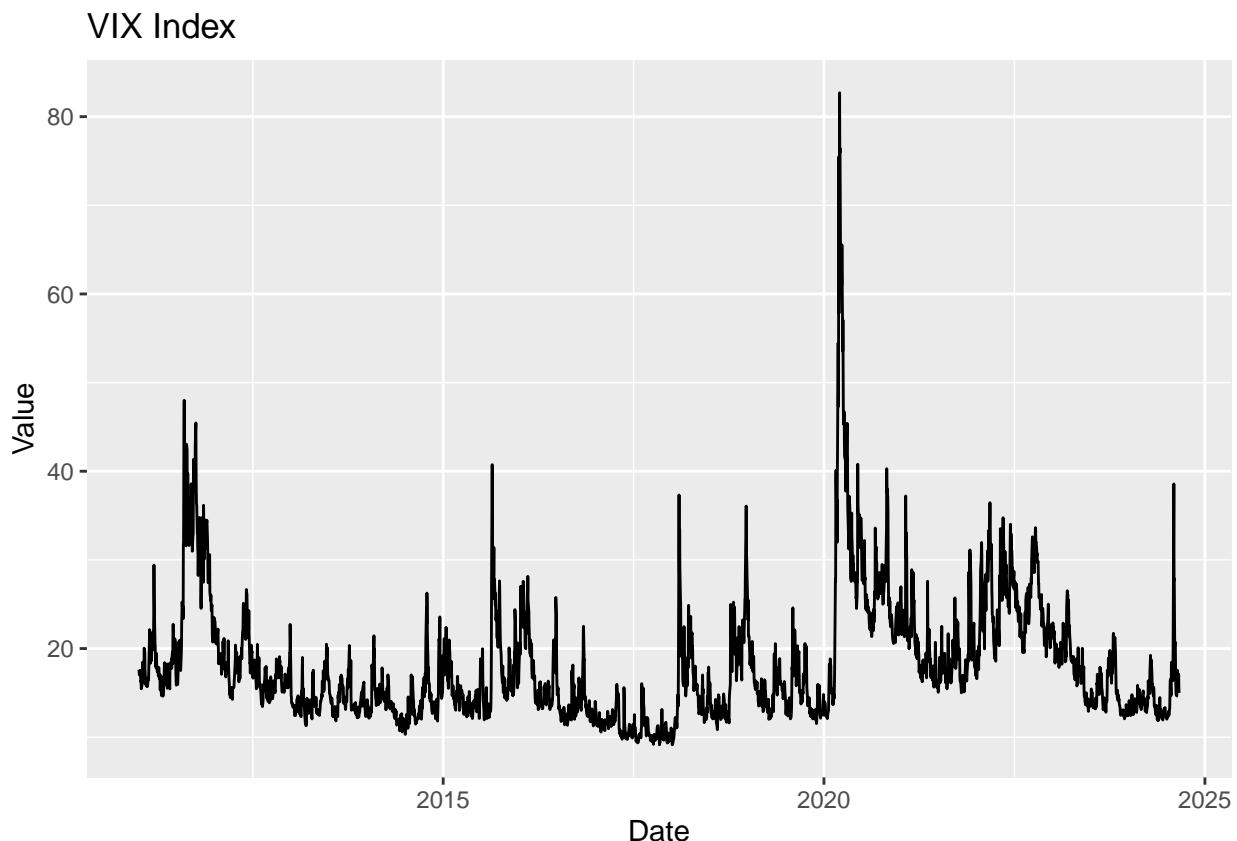
#### 3.1 S&P 500 Index

```
market_data %>%
  filter(symbol == "SP500") %>%
  ggplot(market_data, mapping = aes(date, price)) +
  geom_line() +
  labs(title="S&P 500 Index", x="Date", y="Price")
```



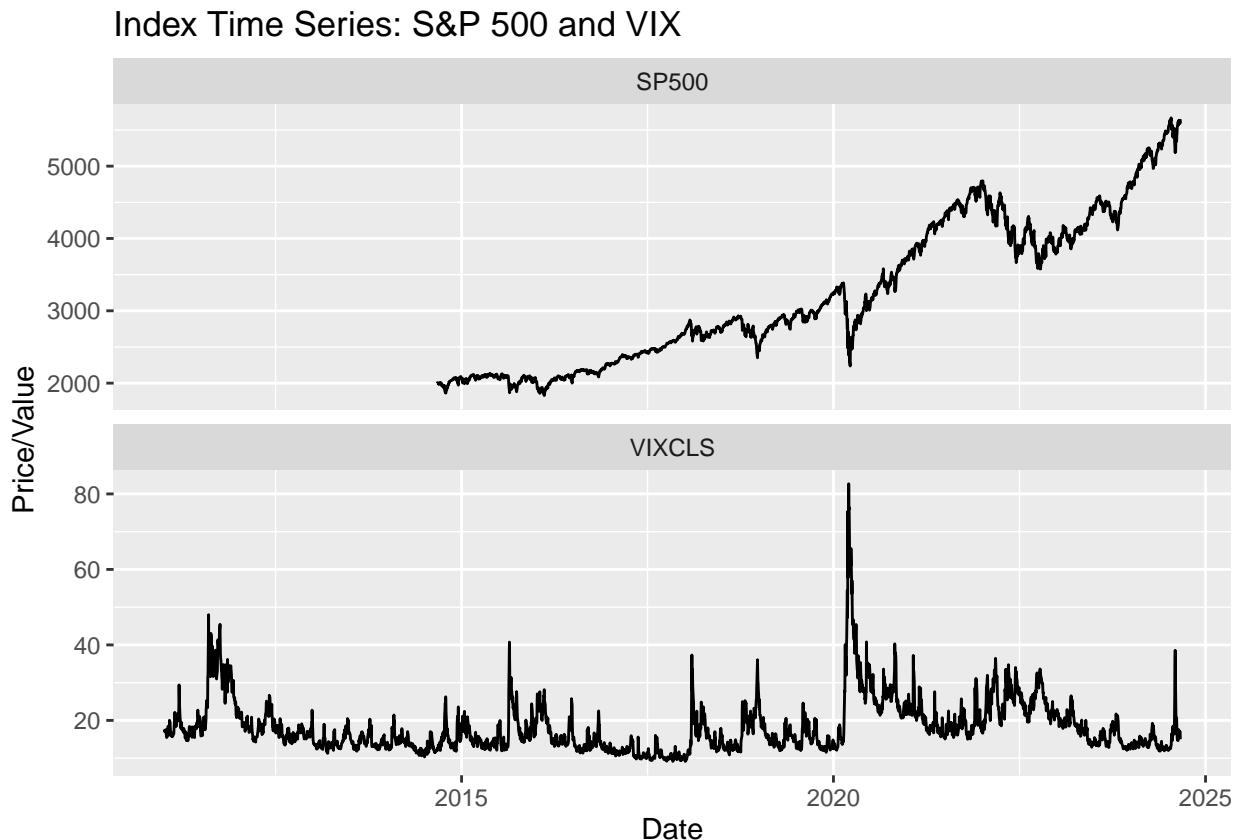
### 3.2 VIX Index

```
market_data %>%
  filter(symbol == "VIXCLS") %>%
  ggplot(market_data, mapping = aes(date, price)) +
  geom_line() +
  labs(title="VIX Index", x="Date", y="Value")
```



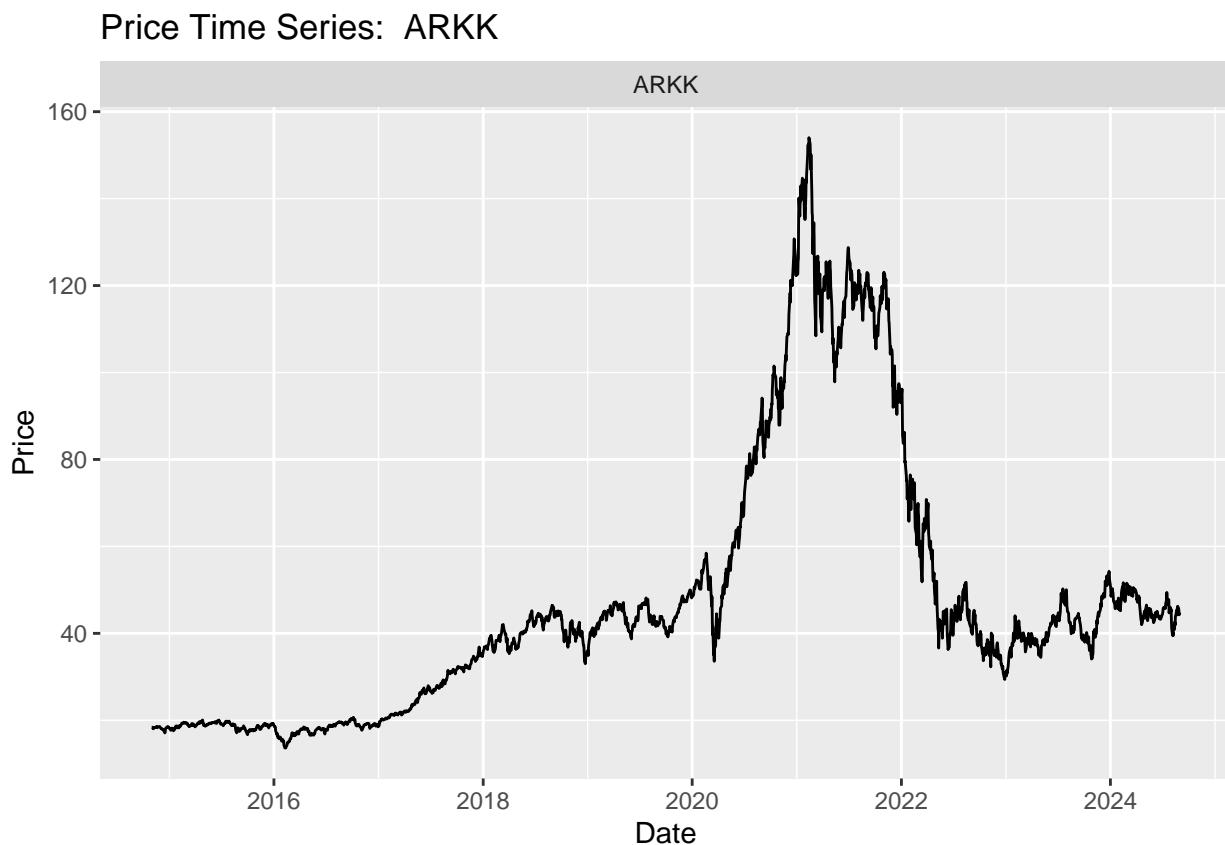
### 3.3 Plot both together

```
market_data %>%
  filter(symbol %in% c("SP500", "VIXCLS")) %>%
  ggplot( mapping = aes(date, price), col=symbol) +
  geom_line() +
  facet_wrap(~ symbol, ncol = 1, scale = "free_y") +
  labs(x="Date", y="Price/Value", title="Index Time Series: S&P 500 and VIX")
```



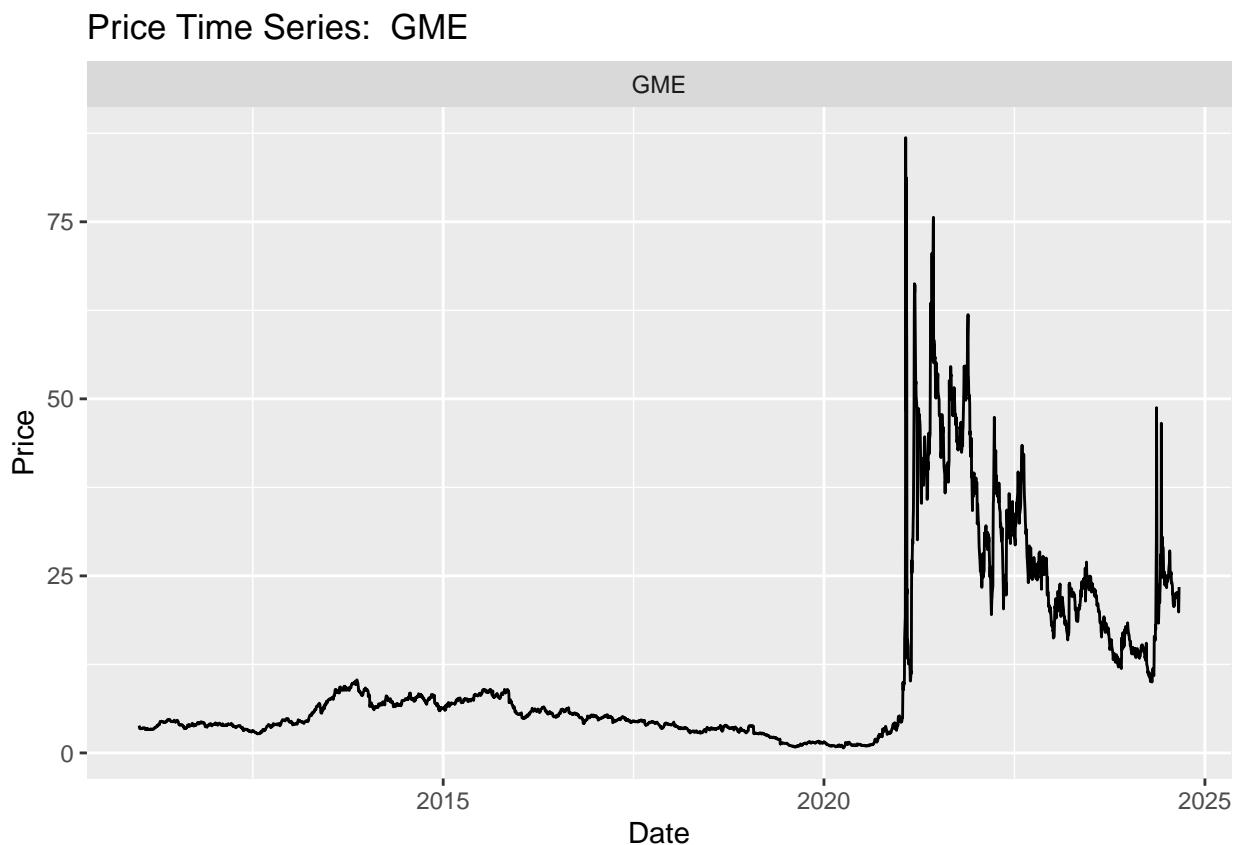
### 3.4 ARKK

```
market_data %>%
  filter(symbol %in% c("ARKK")) %>%
  ggplot( mapping = aes(date,price)) +
  geom_line() +
  facet_wrap(~ symbol, ncol = 1, scale = "free_y") +
  labs(x="Date", y="Price",
       title="Price Time Series: ARKK")
```



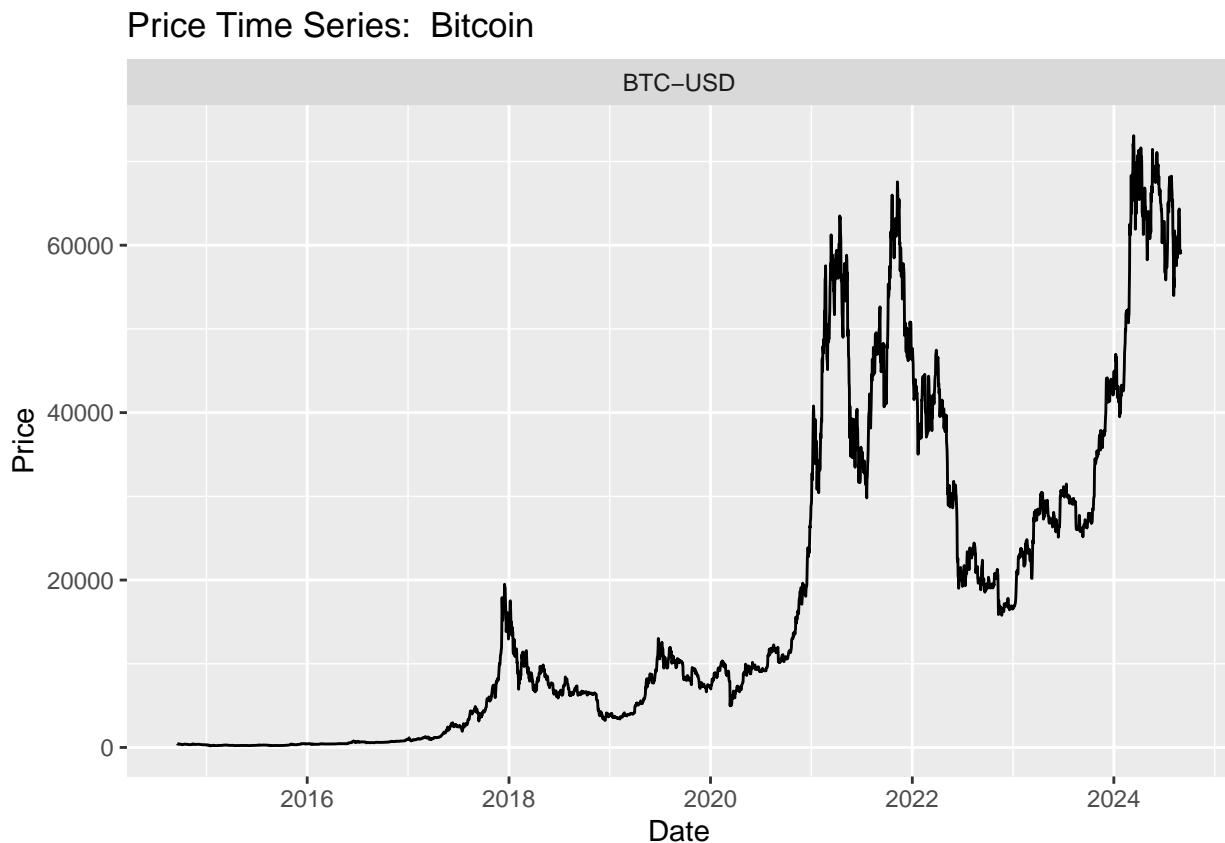
### 3.5 GME

```
market_data %>%
  filter(symbol %in% c("GME")) %>%
  ggplot( mapping = aes(date,price)) +
  geom_line() +
  facet_wrap(~ symbol, ncol = 1, scale = "free_y") +
  labs(x="Date", y="Price",
       title="Price Time Series: GME")
```



### 3.6 Bitcoin

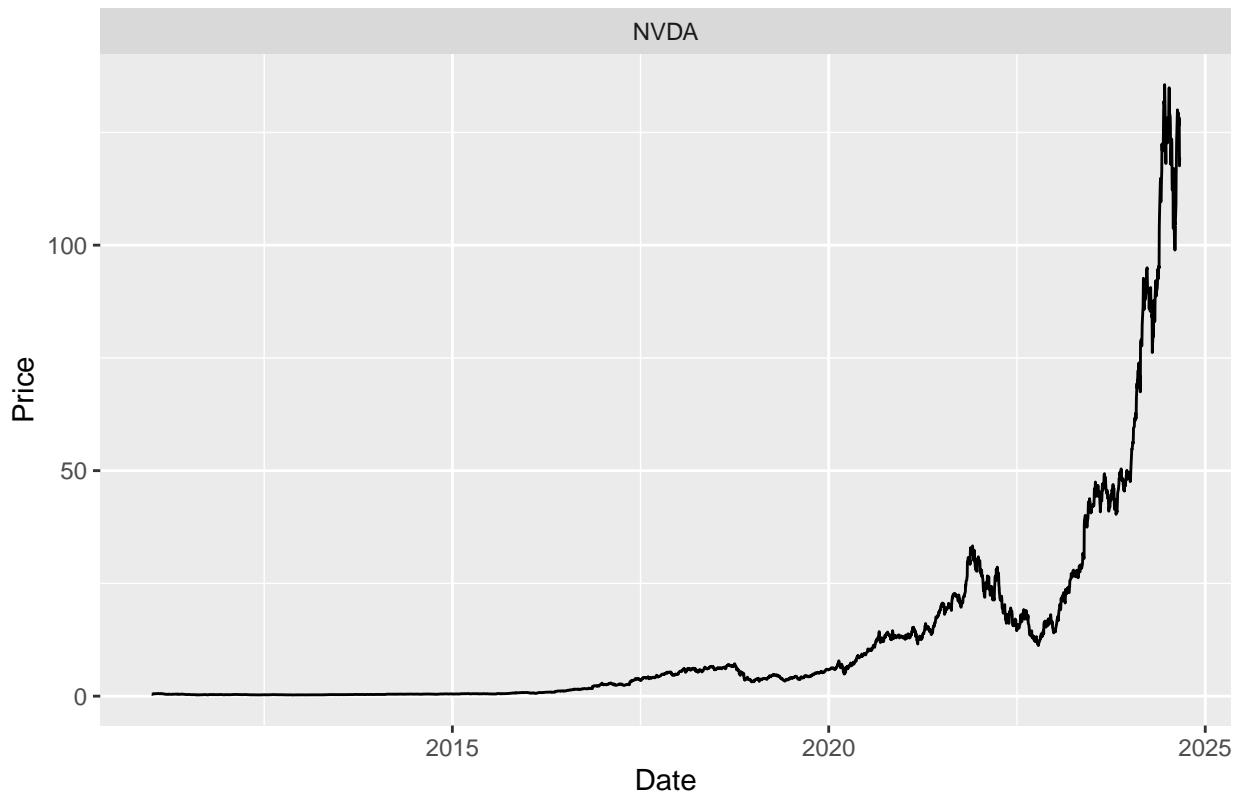
```
market_data %>%
  filter(symbol %in% c("BTC-USD")) %>%
  ggplot( mapping = aes(date,price)) +
  geom_line() +
  facet_wrap(~ symbol, ncol = 1, scale = "free_y") +
  labs(x="Date", y="Price",
       title="Price Time Series: Bitcoin")
```



### 3.7 Nvidia

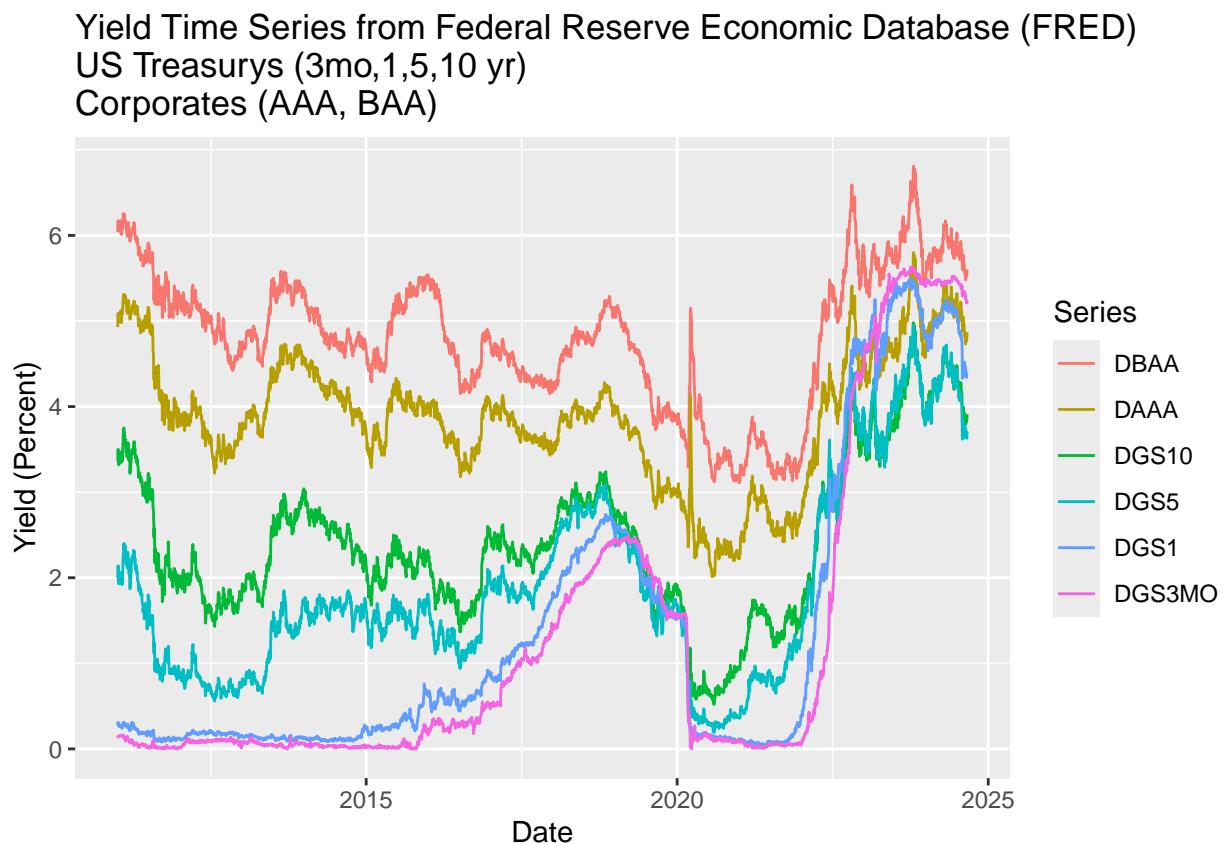
```
market_data %>%
  filter(symbol %in% c("NVDA")) %>%
  ggplot( mapping = aes(date,price)) +
  geom_line() +
  facet_wrap(~ symbol, ncol = 1, scale = "free_y") +
  labs(x="Date", y="Price",
       title="Price Time Series: Nvidia")
```

Price Time Series: Nvidia



### 3.8 Bond Yields on the same graph

```
list_yields<-c("DGS3MO", "DGS1", "DGS5", "DGS10", "DAAA", "DBAA")  
  
market_data %>%  
  filter(symbol %in% list_yields) %>%  
  mutate(Series = factor(symbol, levels=rev(list_yields))) %>%  
  ggplot(market_data,  
         mapping = aes(date, price, color = Series)) +  
  geom_line() +  
  labs(x="Date", y="Yield (Percent)",  
       title="Yield Time Series from Federal Reserve Economic Database (FRED) \nUS Treasurys (3mo,1,5,10 yr)  
Corporates (AAA, BAA)"
```



### 3.9 Crude Oil Futures

```
market_data %>%
  filter(symbol == "DCOILWTICO") %>%
  ggplot(market_data, mapping = aes(date, price)) +
  geom_line() +
  geom_abline(intercept=0,slope=0,col='red',lwd=2) +
  labs(x="Date", y="Price", title="Time Series of Crude Oil Future (DCOILWTICO)")
```

Time Series of Crude Oil Future (DCOILWTICO)



#### 4. Save R workspace with all time series

```
save(file="data_fm_intro1.Rdata",list=ls())
```

MIT OpenCourseWare  
<https://ocw.mit.edu>

## 18.642 Topics in Mathematics with Applications in Finance

Fall 2024

For information about citing these materials or our Terms of Use, visit: <https://ocw.mit.edu/terms>.