# Financial Data Analysis Tutorial

## Contents

# Data

## Raw Data

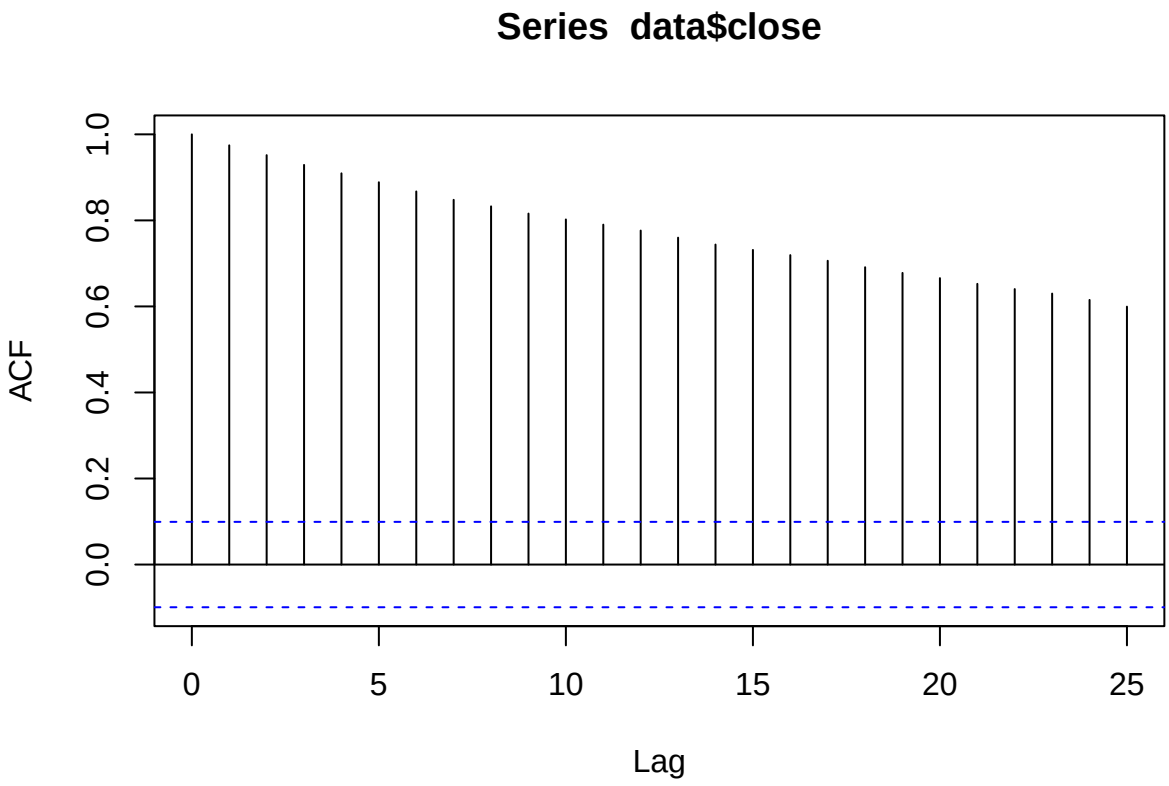
```
#market prices (loads and names them automatically)

#S&P500
data_loader(symbol="SPY")

#STOXX50
data_loader(symbol="VGK")




#CSI 300 (China)
data_loader(symbol="ASHR")
```

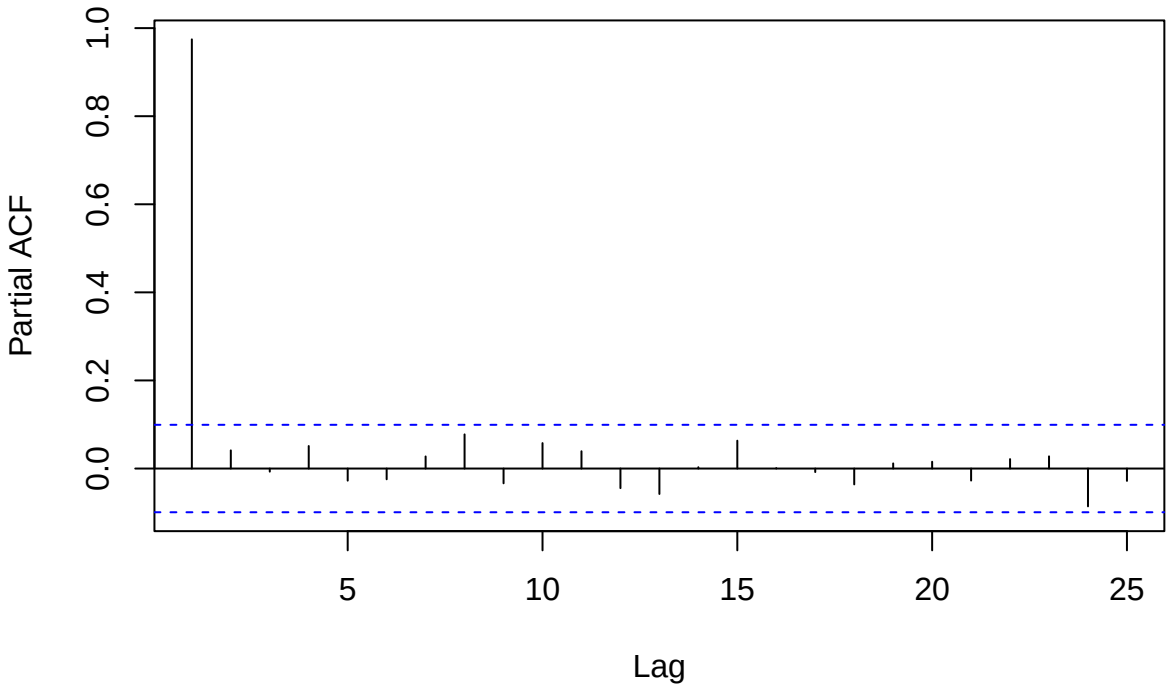
# Quick Analysis

## SPY April 2nd 2025

```
#extract a particular day
SPY_25_04_02 = day_selector(raw_SPY,2025,04,02) #april 2nd 2025

#let's plot it
price_plotter_day(SPY_25_04_02,"SPY Price on April 2nd 2025")
```


```
#quickly test some ARMA specifications
quick_arma(SPY_25_04_02,1,0,0) #checking AR1,AR2,AR3
```


Series data$close
ACF
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
Lag




```
##                                AR Estimations
##
##                                  AR-1        AR-2         AR-3
##
##                  ar1            0.9975      0.9728       1.4609
##                                 (0.0030)    (0.0514)   (NaN)
##                  intercept    561.0971    561.3655      562.5635
##                                 (3.2897)    (3.4352)    (22.1897)
##                  ar2                         0.0249       0.0770
##                                              (0.0515)     (0.0013)
##                  ar3                                      -0.5386
##                                                           (0.0007)
##
##                  nobs         390         390           390
##                  sigma          0.2854      0.2853        0.3414
##                  logLik       -67.0847    -66.9808     -135.4359
##                  AIC          140.1693    141.9615      280.8718
##                  BIC          152.0678    157.8261      300.7025
##                  nobs.1       390.0000    390.0000      390.0000
##
##                  *** p < 0.001; ** p < 0.01; * p <
##                  0.05.
##
## Column names: names, AR-1, AR-2, AR-3
##                                Checking Residuals
##
```

```
##                          AR-1 Residuals   AR-2 Residuals   AR-3 Residuals
##
##        (Intercept)             0.0302 *         0.0291 *       -0.0051
##                               (0.0145)         (0.0145)        (0.0171)
##        REG1res_lagged         -0.0476
##                               (0.0510)
##        REG2res_lagged                          -0.0217
##                                                (0.0511)
##        REG3res_lagged                                          -0.1733 ***
##                                                                (0.0503)
##
##        N                     389              389             389
##        R2                      0.0022           0.0005          0.0297
##
##        *** p < 0.001; ** p < 0.01; * p < 0.05.
##
## Column names: names, AR-1 Residuals, AR-2 Residuals, AR-3 Residuals
```

```
#quick_arma(SPY_25_04_02,2,0,0) #checking AR2,AR3,AR4


#extract a particular month
SPY_24_09 = month_selector(raw_SPY,2024,09) #november 2024


#extract a particular year
SPY_24 = year_selector(raw_SPY,2024) #2024
```

# Realised Volatility

## Computations

```
#avg per day for each month of any dataset
vol_SPY_daily = r.vol_daily(raw_SPY,merge=F)
head(vol_SPY_daily)
```

| timestamp | r__vol__d |
|---|---|
| 2019-01-02 | 0.0295 |
| 2019-01-03 | 0.0365 |
| 2019-01-04 | 0.0241 |
| 2019-01-07 | 0.0165 |
| 2019-01-08 | 0.0136 |
| 2019-01-09 | 0.0144 |

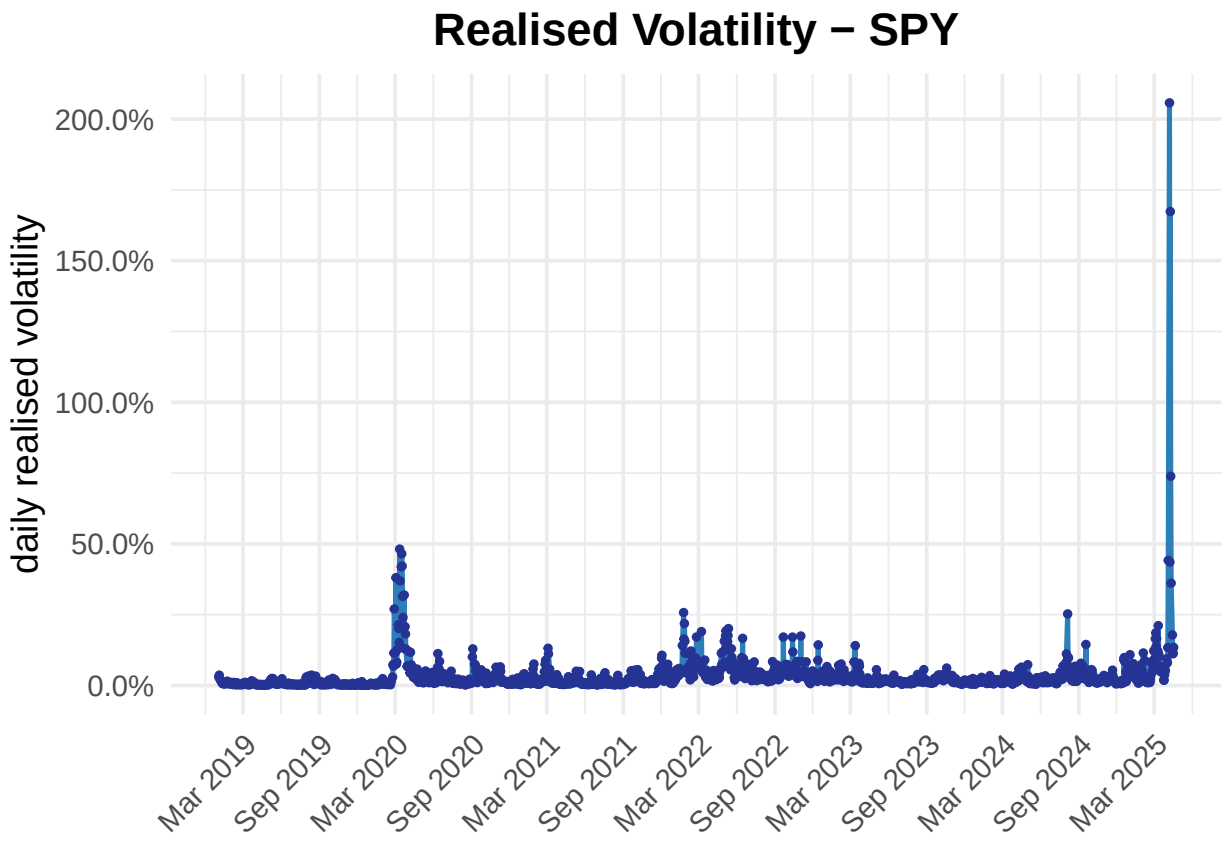
```
#can then filter out years, months, or days
vol_24d = year_selector(vol_SPY_daily,2024)
vol_24_08d = month_selector(vol_SPY_daily,2024,08)
vol_24_11_04d = day_selector(vol_SPY_daily,2024,11,04) #scalar


#avg per hour for each day of each month of any dataset
vol_SPY_hourly = r.vol_hourly(raw_SPY,merge=F)
head(vol_SPY_hourly)
```

| timestamp | r__vol__h |
|---|---|
| 2019-01-02 09:00:00 | 0.034 |
| 2019-01-02 10:00:00 | 0.0401 |
| 2019-01-02 11:00:00 | 0.0363 |
| 2019-01-02 12:00:00 | 0.0185 |
| 2019-01-02 13:00:00 | 0.0185 |
| 2019-01-02 14:00:00 | 0.0199 |

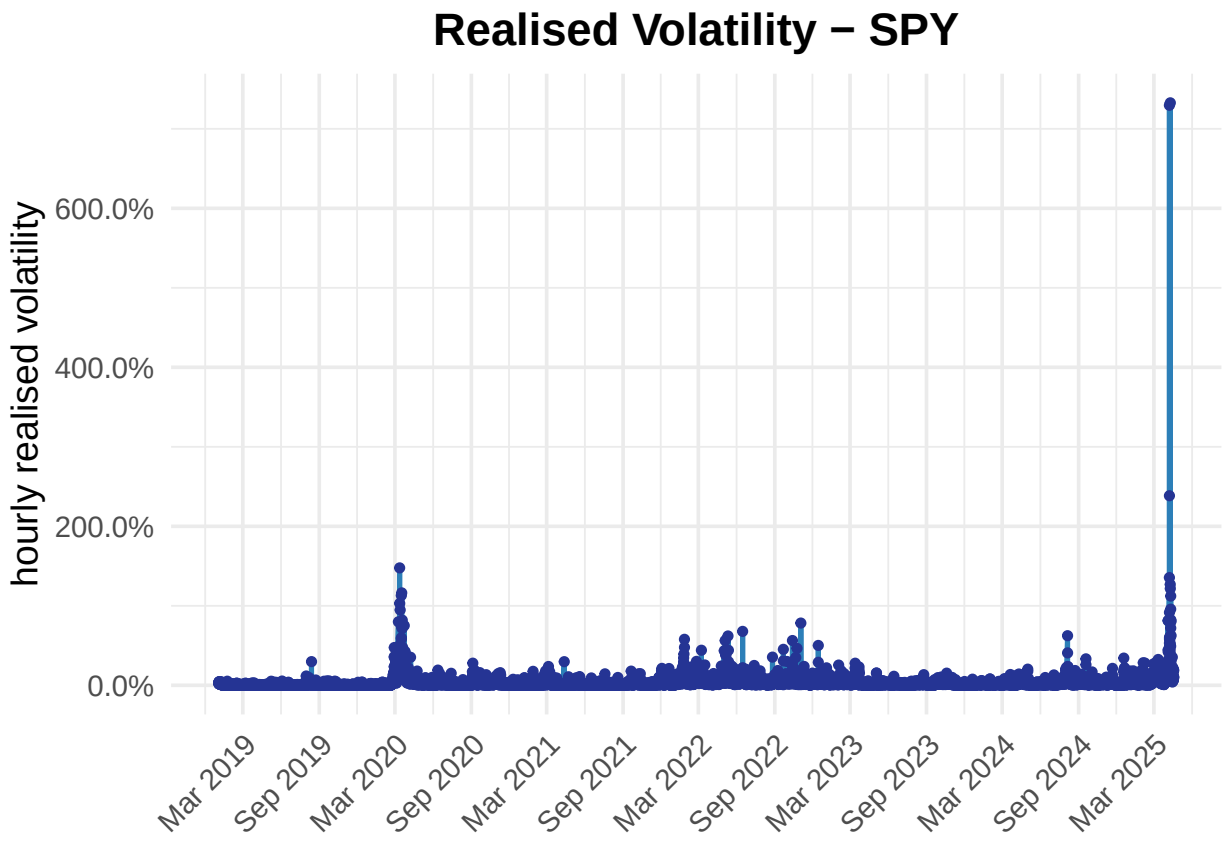
```
#can then filter out years, months, or days
vol_24h = year_selector(vol_SPY_hourly,2024)
vol_24_08h = month_selector(vol_SPY_hourly,2024,08)
vol_24_11_04h = day_selector(vol_SPY_hourly,2024,11,04) #vector
```

## Plots

```
#avg per day volatility all time
dvol_plotter(vol_SPY_daily,breaks="yearly",
             title="Realised Volatility - SPY")
```




```
#hourly volatility all time
hvol_plotter(vol_SPY_hourly,breaks="yearly",
             title="Realised Volatility - SPY")
```


Realised Volatility − SPY
hourly realised volatility
600.0%
400.0%
200.0%
0.0%
Mar 2019
Sep 2019
Mar 2020
Sep 2020
Mar 2021
Sep 2021
Mar 2022
Sep 2022
Mar 2023
Sep 2023
Mar 2024
Sep 2024
Mar 2025