

# Financial Data Analysis

## Contents

<b>Data</b>	<b>2</b>
Raw Data . . . . .	2
<b>Quick Analysis</b>	<b>2</b>
SPY April 2nd 2025 . . . . .	2
Realised Volatility . . . . .	6

# Data

## Raw Data

```
#political shocks
raw_truths <- read.csv(here("data/political_data", "trump_all_truths.csv"))
raw_tweets <- read.csv(here("data/political_data", "tweets.csv"))

#market prices
raw_ONEQ <- read.csv(here("data/market_data", "ONEQ.csv")) #USA
raw_SMI <- read.csv(here("data/market_data", "SMI.csv")) #CH
raw_VTHR <- read.csv(here("data/market_data", "VTHR.csv")) #USA
raw_VTI <- read.csv(here("data/market_data", "VTI.csv")) #USA
raw_DAX <- read.csv(here("data/market_data", "DAX.csv")) #DE
raw_ASHR <- read.csv(here("data/market_data", "ASHR.csv")) #CHINA
data_loader(year=2021,months=1:6,symbol="SPY")
data_loader(year=2024,months=1:12,symbol="SPY")
data_loader(year=2025,months=1:4,symbol="SPY")
data_loader(year=2024,months=1:12,symbol="VGK") #EU
data_loader(year=2025,months=1:4,symbol="VGK")
data_loader(year=2025,months=1:2,symbol="ASHR")
```

## Quick Analysis

### SPY April 2nd 2025

```
#extract a particular day
day_SPY_0402 = day_selector(raw_SPY_2025_04,2025,04,02) #april 2nd 2025

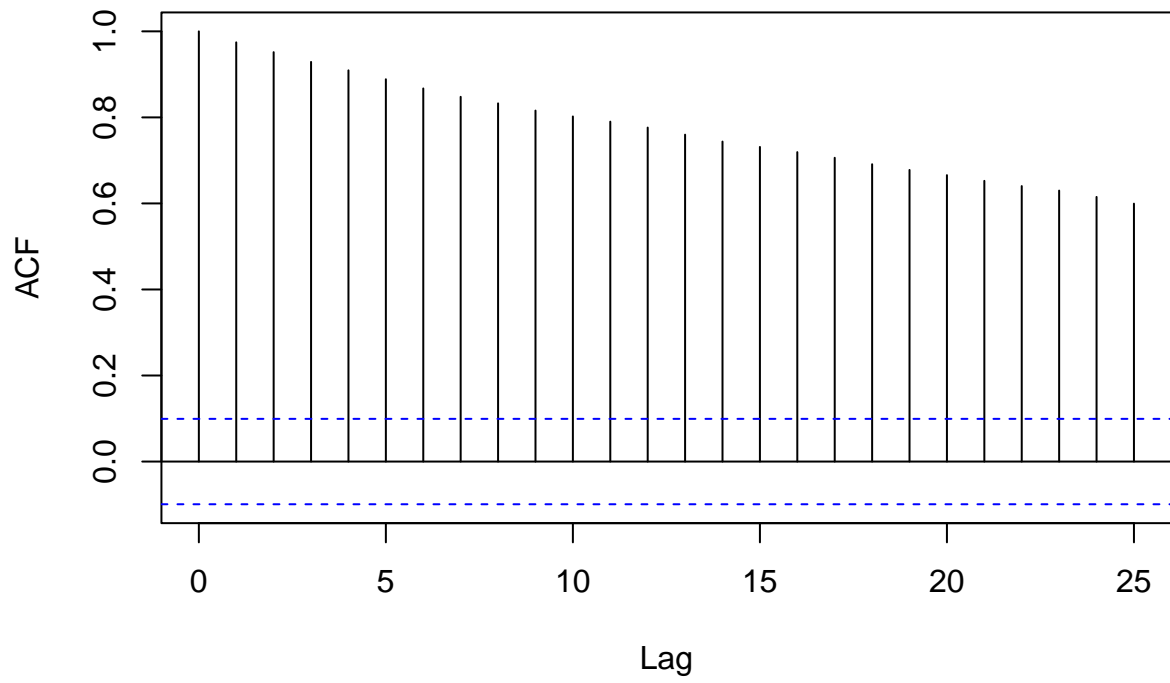
#let's plot it
day_plotter(day_SPY_0402,"SPY Price on April 2nd 2025")
```

SPY Price on April 2nd 2025

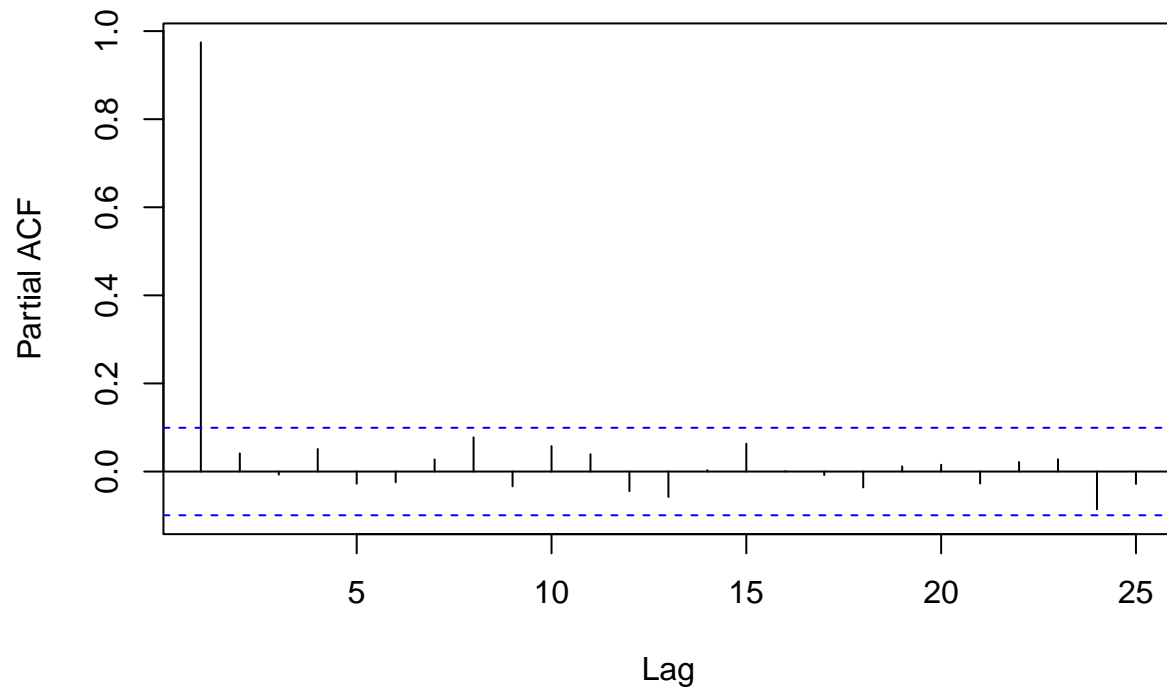


```
#quickly test some ARMA specifications  
quick_arma(day_SPY_0402,1,0,0) #checking AR1,AR2,AR3
```

**Series data\$close**



## Series data\$close



```
##
##               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(day_SPY_0402,2,0,0) #checking AR2,AR3,AR4
```

## Realised Volatility

```
#for a particular day (outputs scalar)
r.vol(day_SPY_0402)
```

```
## [1] 0.08152862
```

```
#for a month (outputs vector of each day's realised volatility)
r.vol_month(raw_SPY_2025_04)
```

```
## [1] 0.07851997 0.08152862 0.13381914 0.44172333 2.05732267 0.43592888
## [7] 1.67344102 0.73881997 0.36128756 0.17859522 0.11181995
```

```
#for each hour in a day (outputs a vector of each hour's realised volatility)
r.vol_hour(day_SPY_0402)
```

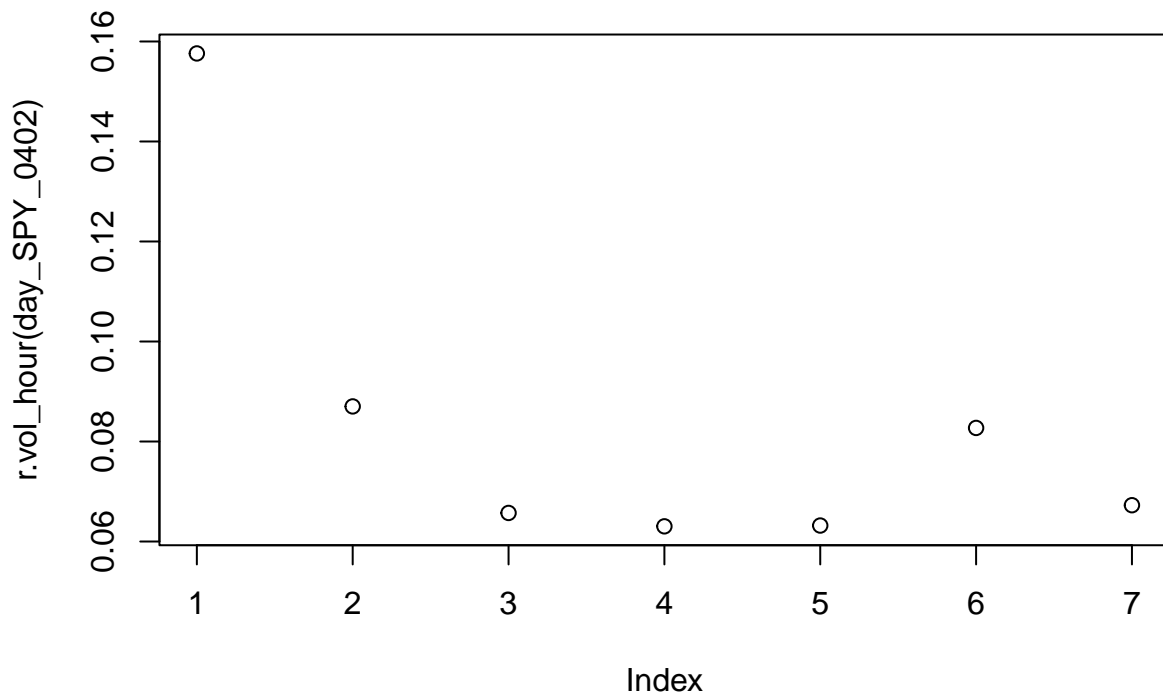
```
## [1] 0.15760939 0.08701794 0.06571201 0.06303564 0.06319524 0.08271313 0.06726031
```

```
#for each hour in a day for each day in a month (outputs a matrix)
month_hour = r.vol_month_hour(raw_SPY_2025_04)
huxtable(data.frame(month_hour))
```

```
#plots
```

```
#hours in a day
plot(r.vol_hour(day_SPY_0402))
```

X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11
0.141	0.158	0.278	0.812	1.35	0.578	1.27	0.825	0.812	0.353	0.178
0.13	0.087	0.148	0.445	7.3	0.278	0.526	0.492	0.625	0.205	0.222
0.0766	0.0657	0.137	0.815	2.38	0.29	0.401	0.288	0.314	0.165	0.131
0.0657	0.063	0.16	0.319	0.531	0.452	0.189	0.958	0.256	0.222	0.0745
0.0492	0.0632	0.0927	0.284	0.918	0.439	7.33	1.12	0.355	0.182	0.0878
0.0596	0.0827	0.0745	0.216	0.582	0.496	1.22	0.72	0.153	0.0414	0.0669
0.0613	0.0673	0.112	0.419	0.6	0.61	0.618	0.796	0.17	0.137	0.0517



```
#days in a month
plot(r.vol_month(raw_SPY_2025_04))
```

