

# SPY Data Descriptive Analysis

## Contents

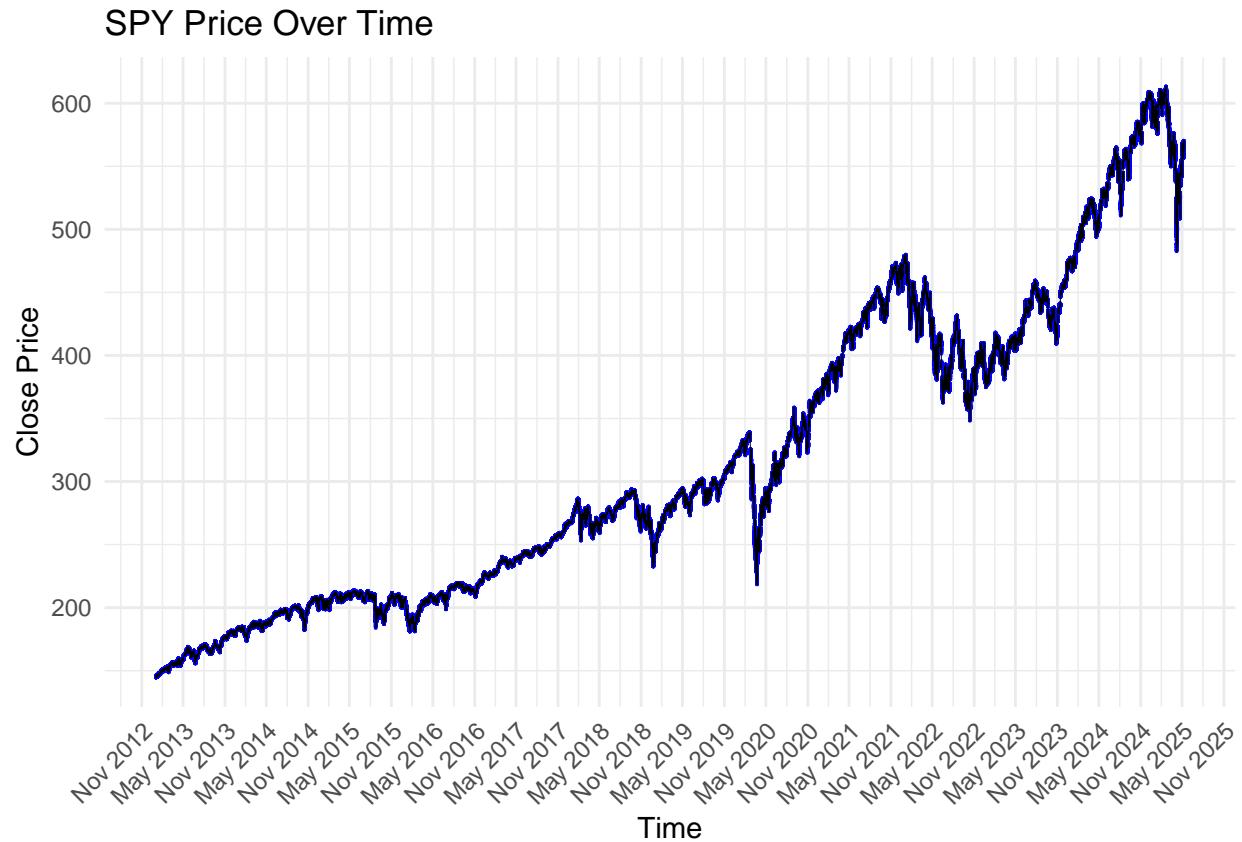
<b>Price</b>	<b>2</b>
Plots . . . . .	2
Time Series Analysis . . . . .	5
<b>Realised Volatility</b>	<b>8</b>
Computations . . . . .	8
Plots . . . . .	9

## Price

Here we provide some descriptive facts about our financial data on SPY prices.

## Plots

```
#All Time  
price_plotter(raw_SPY, "SPY Price Over Time")
```



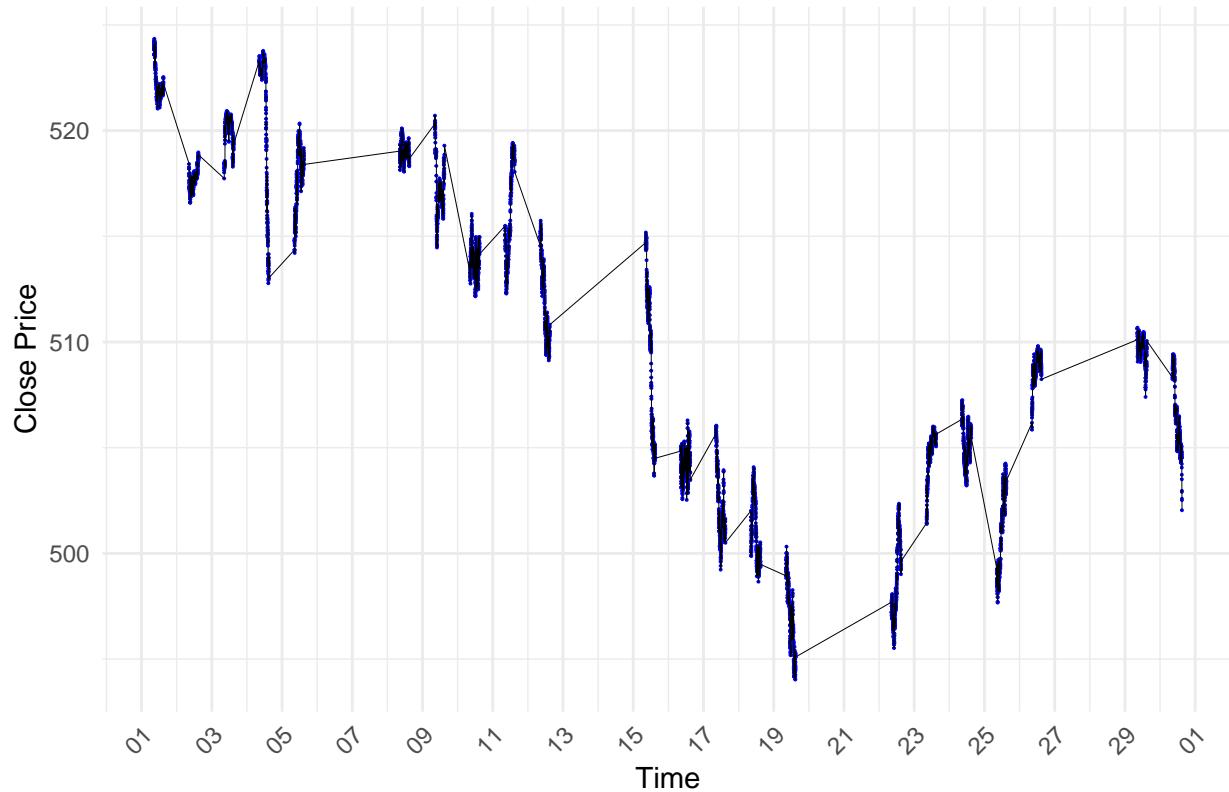
```
#2024  
SPY_2024 = year_selector(raw_SPY, 2024)  
price_plotter_year(SPY_2024, "SPY Price - 2024")
```

## SPY Price – 2024



```
#April 2025
SPY_2025_04 = month_selector(raw_SPY, 2024, 04)
price_plotter_month(SPY_2025_04, "SPY Price - April 2025")
```

## SPY Price – April 2025



```
#9th of April 2025
SPY_2025_04_09 = day_selector(raw_SPY,2024,04,09)
price_plotter_day(SPY_2025_04_09,"SPY Price - 9th of April 2025")
```

SPY Price – 9th of April 2025

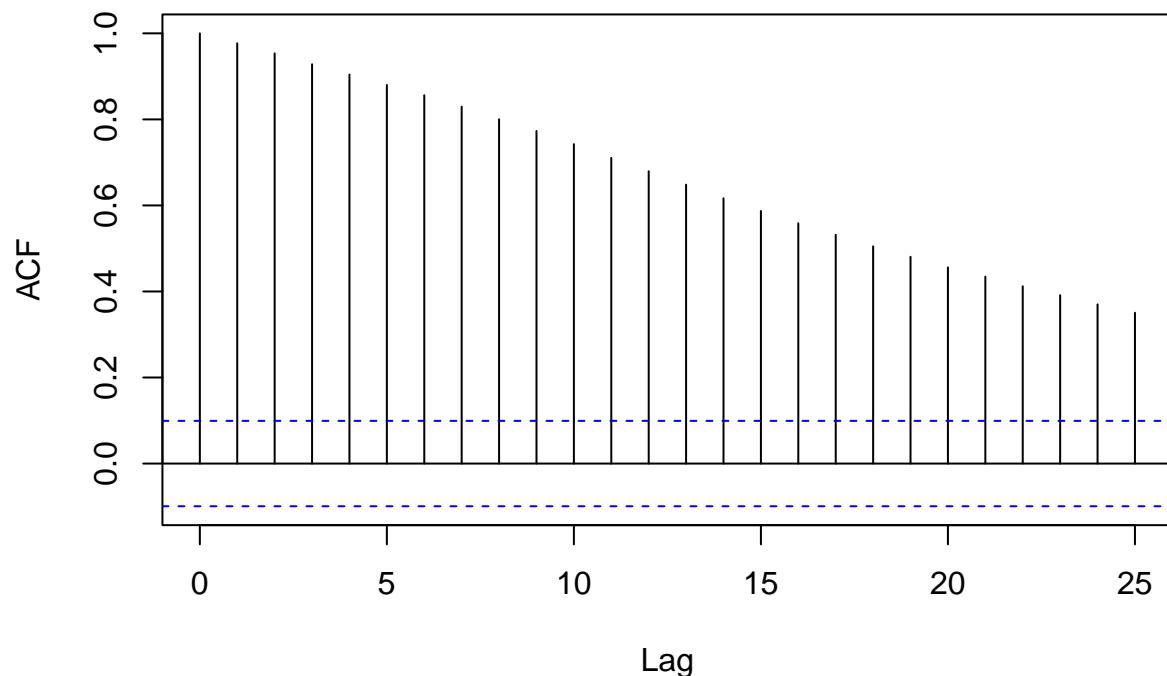


## Time Series Analysis

This is not at all related to our final analysis. It is just in order to see the financial data on its own and how it behaves as a time series.

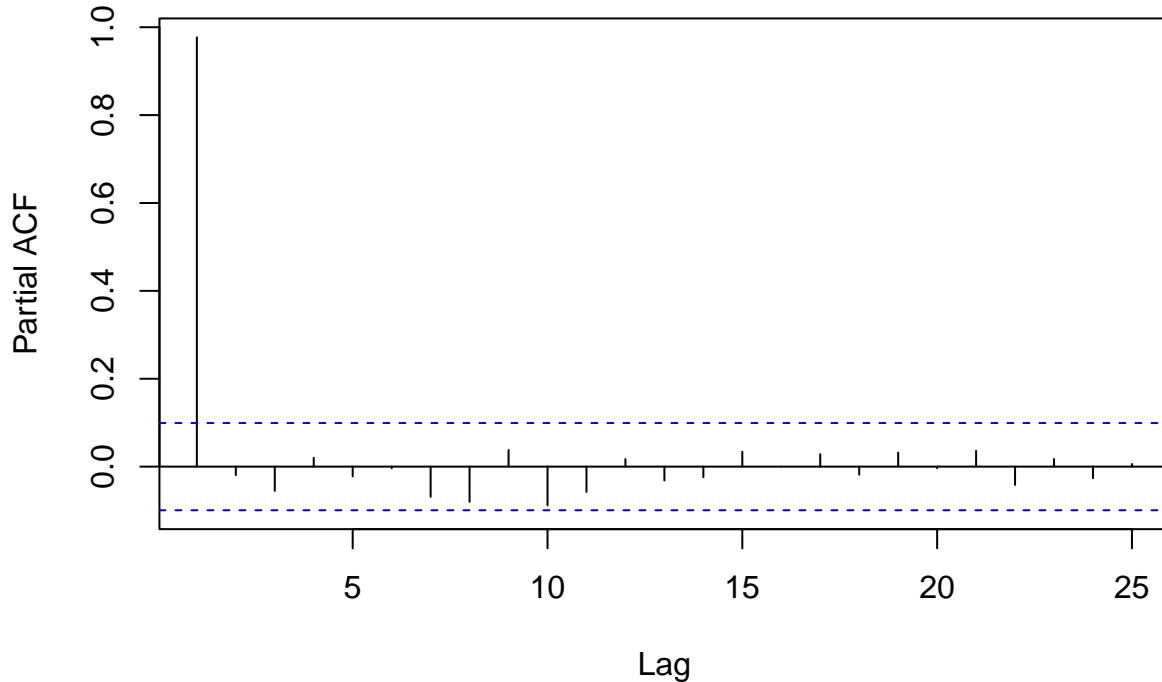
```
acf(SPY_2025_04_09$close)
```

### Series SPY\_2025\_04\_09\$close



```
pacf(SPY_2025_04_09$close)
```

## Series SPY\_2025\_04\_09\$close



```
AR1 = arima(SPY_2025_04_09$close, c(1,0,0), method="ML")
AR2 = arima(SPY_2025_04_09$close, c(2,0,0), method="ML")
AR3 = arima(SPY_2025_04_09$close, c(3,0,0), method="CSS")
table1 = export_summs(AR1,AR2,AR3, model.names = c("AR1","AR2","AR3"), digits = 4)
huxtable::caption(table1) <- "AR Estimations"
huxtable::set_width(table1, 0.8)
```

```
AR1res = as.numeric(AR1$residuals)
AR1res_lagged <- lag(AR1res, 1)
iidcheck1 = lm(AR1res ~ AR1res_lagged)
AR2res = as.numeric(AR2$residuals)
AR2res_lagged <- lag(AR2res, 1)
iidcheck2 = lm(AR2res ~ AR2res_lagged)
AR3res = as.numeric(AR3$residuals)
AR3res_lagged <- lag(AR3res, 1)
iidcheck3 = lm(AR3res ~ AR3res_lagged)
table2 = export_summs(iidcheck1,iidcheck2,iidcheck3,
                      model.names = c("AR1 Residuals","AR2 Residuals","AR3 Residuals"),
                      digits = 4)
huxtable::caption(table2) <- "Checking Residuals"
huxtable::set_width(table2, 0.8)
```

Table 1: AR Estimations

	AR1	AR2	AR3
ar1	0.9944 (0.0047)	1.0598 (0.0509)	1.0399 (0.0507)
intercept	518.3317 (1.1978)	518.2751 (1.1607)	517.0057 (0.4637)
ar2		-0.0659 (0.0511)	0.0210 (0.0731)
ar3			-0.0796 (0.0505)
nobs	390	390	390
sigma	0.1723	0.1719	0.1701
logLik	130.2386	131.0665	
AIC	-254.4771	-254.1331	
BIC	-242.5787	-238.2685	
nobs.1	390.0000	390.0000	390.0000

\*\*\* p < 0.001; \*\* p < 0.01; \* p < 0.05.

## 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)
```

```
#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)
```

Table 2: Checking Residuals

	AR1 Residuals	AR2 Residuals	AR3 Residuals
(Intercept)	-0.0085 (0.0087)	-0.0091 (0.0087)	0.0000 (0.0086)
AR1res_lagged	0.0634 (0.0511)		
AR2res_lagged		-0.0074 (0.0511)	
AR3res_lagged			0.0099 (0.0513)
N	389	389	389
R2	0.0040	0.0001	0.0001

\*\*\* p < 0.001; \*\* p < 0.01; \* p < 0.05.

timestamp	r_vol_d
2013-01-02	0.00197
2013-01-03	0.00178
2013-01-04	0.0012
2013-01-07	0.000854
2013-01-08	0.00115
2013-01-09	0.000924

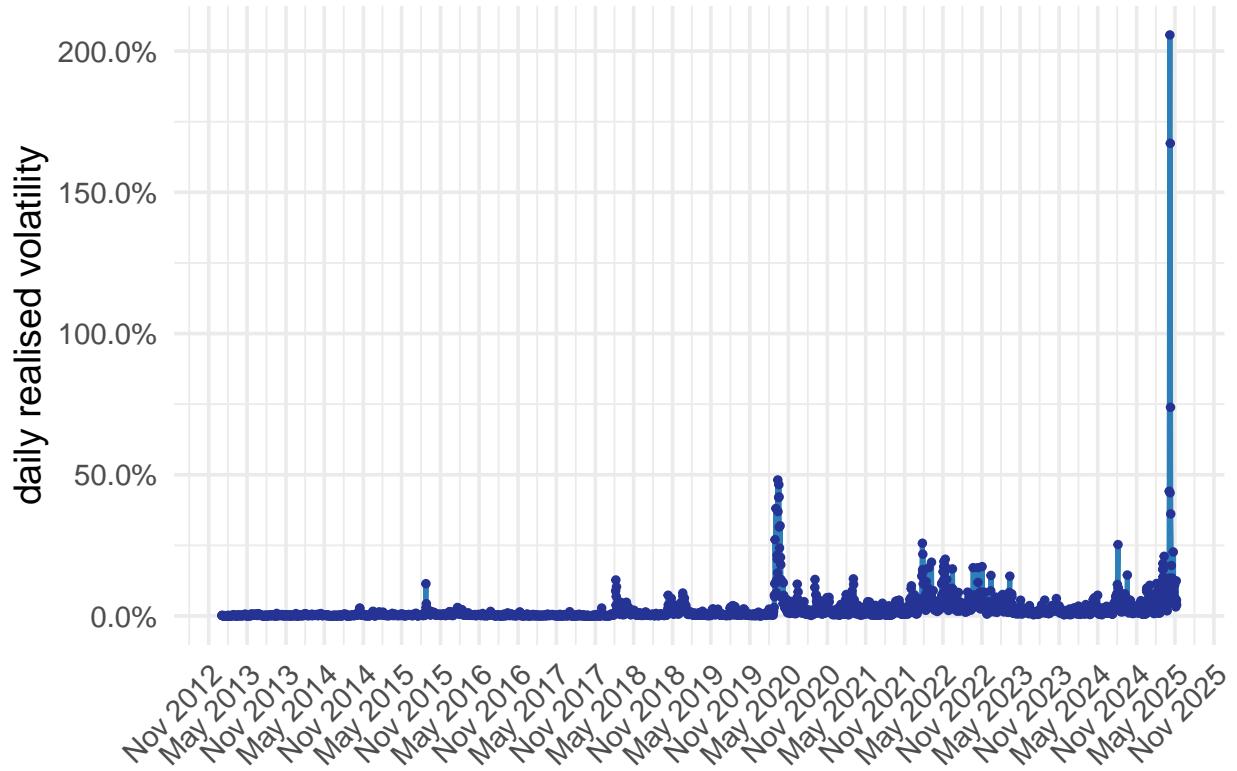
```
#can then filter out years, months, or days
vol_24h = year_selector(vol_SPY_hourly,2024)
vol_24_08h = month_selector(vol_SPY_hourly,2024,08)
vol_24_11_04h = day_selector(vol_SPY_hourly,2024,11,04) #vector
```

## Plots

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

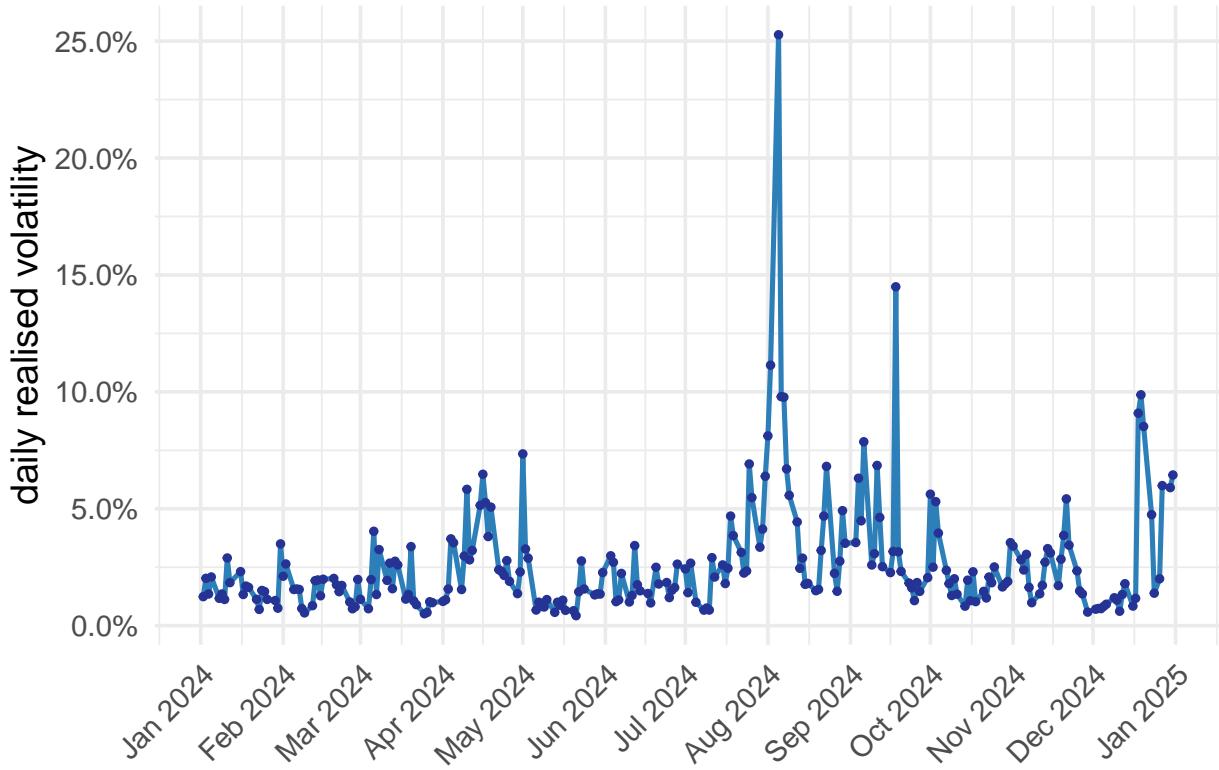
timestamp	r_vol_h
2013-01-02 09:00:00	0.00403
2013-01-02 10:00:00	0.00251
2013-01-02 11:00:00	0.00183
2013-01-02 12:00:00	0.00109
2013-01-02 13:00:00	0.001
2013-01-02 14:00:00	0.00149

## SPY Volatility Over Time



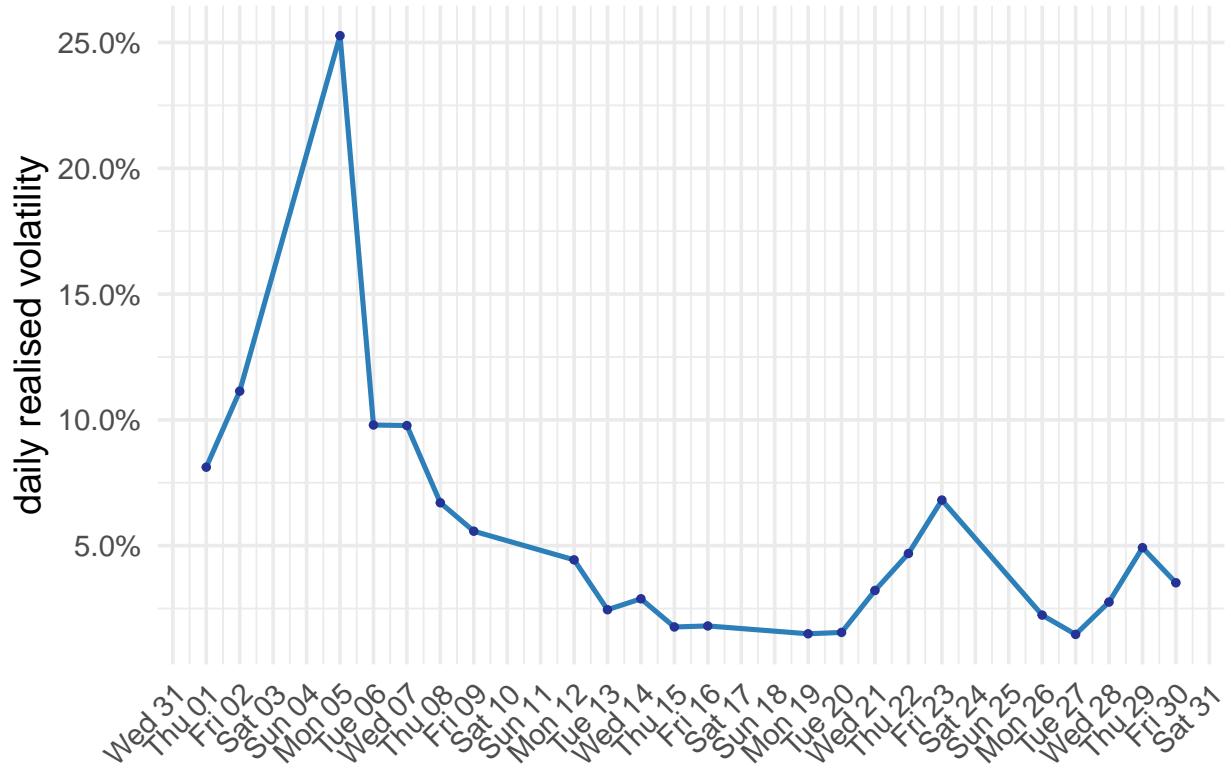
```
#avg per day volatility in a year
dvol_plotter(vol_24d, breaks="monthly",
             title="Realised Volatility - SPY 2024")
```

## Realised Volatility – SPY 2024



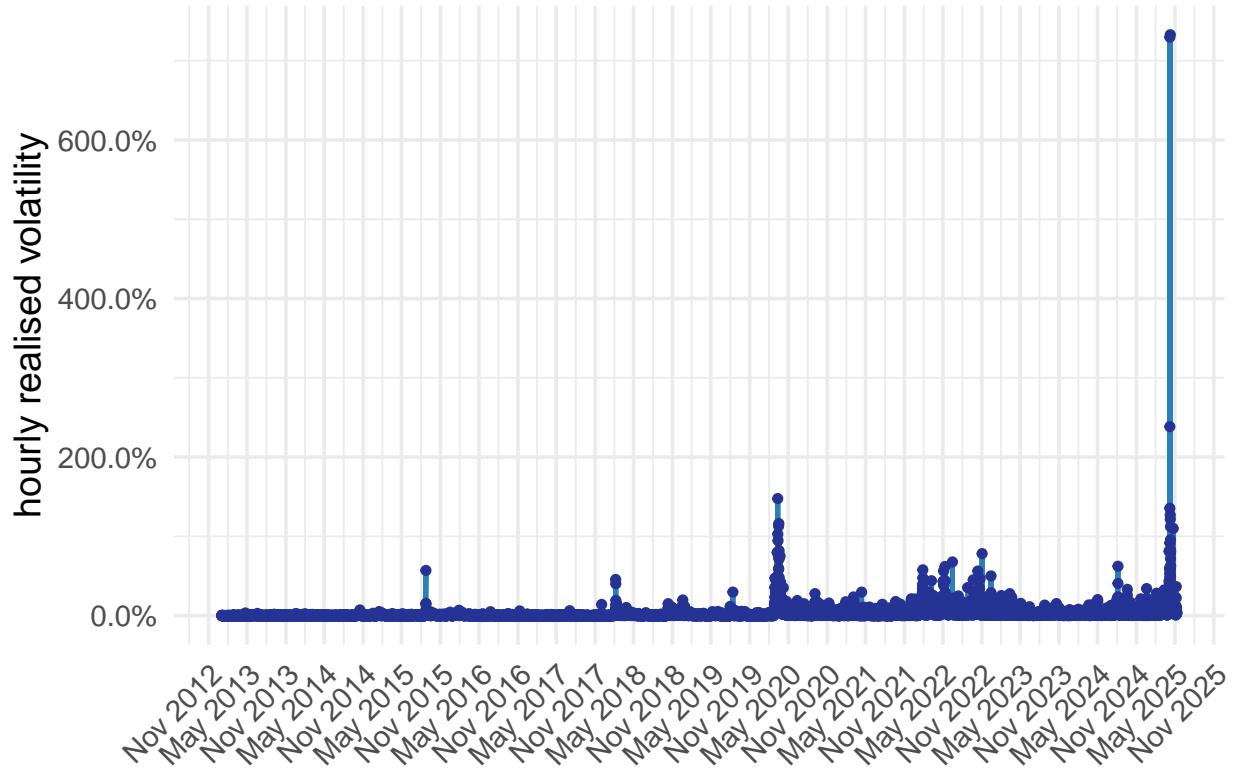
```
#avg per day volatility in a month
dvol_plotter(vol_24_08d,breaks="daily",
             title="Realised Volatility - SPY August 2024")
```

## Realised Volatility – SPY August 2024



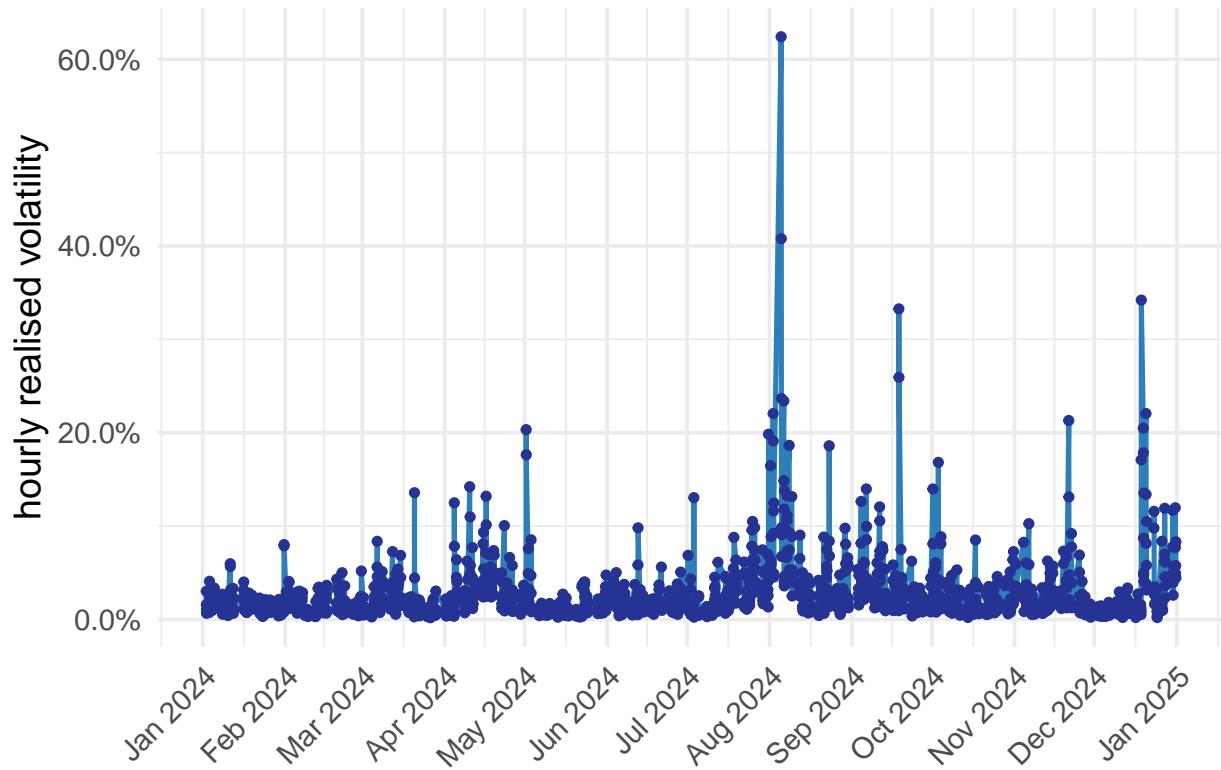
```
#hourly volatility all time
hvol_plotter(vol_SPY_hourly, breaks="yearly",
             title="SPY Volatility Over Time")
```

## SPY Volatility Over Time



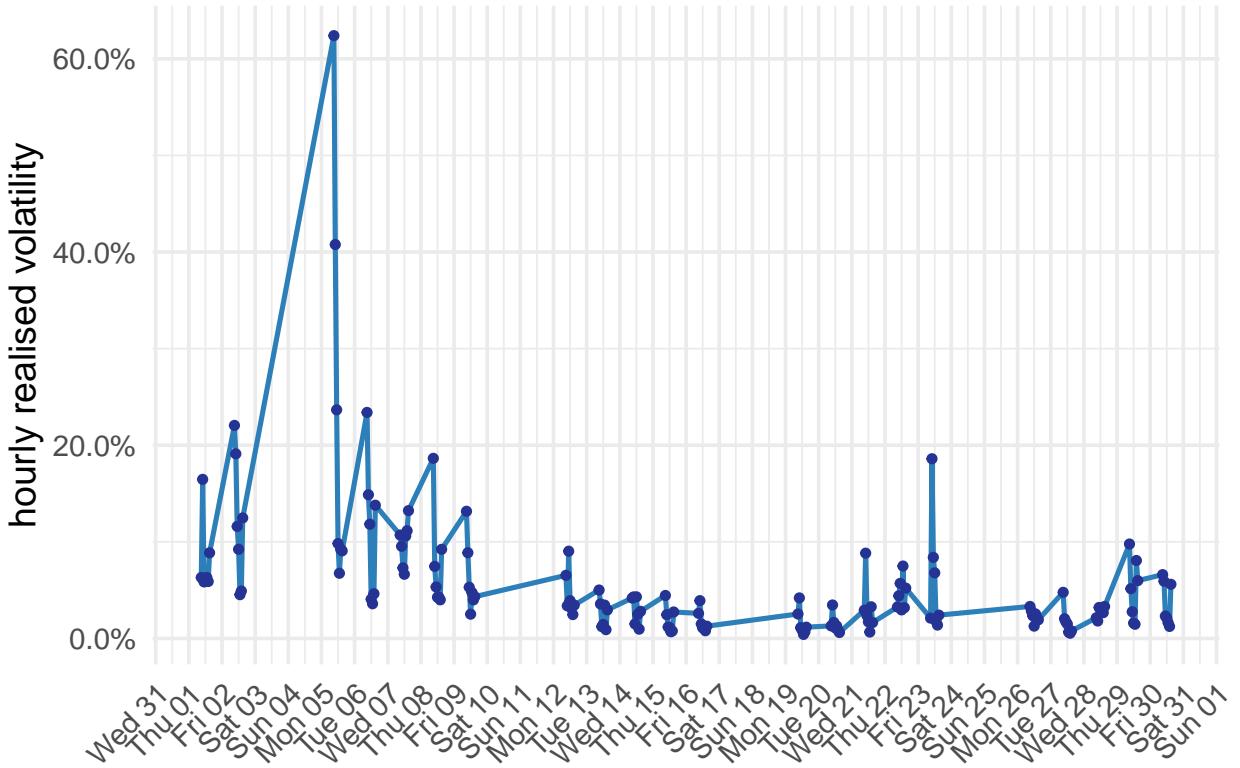
```
#hourly volatility in a year  
hvol_plotter(vol_24h, breaks="monthly",  
             title="Realised Volatility - SPY 2024")
```

## Realised Volatility – SPY 2024



```
#hourly volatility in a month  
hvol_plotter(vol_24_08h,breaks="daily",  
             title="Realised Volatility - SPY August 2024")
```

# Realised Volatility – SPY August 2024



```
#hourly volatility in a day
hvol_plotter(vol_24_11_04h, breaks="hourly",
             title="Realised Volatility - SPY 4th of November 2024")
```

## Realised Volatility – SPY 4th of November 2024

