

Estimación muestral de la volatilidad no condicional.

Ejemplo aplicado al S&P500.

Dr. Martín Lozano <https://mlozanoqf.github.io/>

12 de noviembre de 2025, 03:36 p.m.

	Fundamental	Intermedio	Especializado
Finanzas	✗	✓	✗
Estadística	✓	✗	✗
R	✓	✗	✗

1 Introducción.

- Estimar la volatilidad no condicional de los rendimientos del índice S&P 500, utilizando datos históricos de precios diarios.
- Rendimientos como cambios porcentuales y logaritmos naturales.
- Base para futuros modelos financieros: Black–Scholes, Markowitz, VaR, Monte Carlo, Random Walk, Geometric Brownian Motion, Vasicek, riesgo de crédito de Merton, CAPM, APT.

2 Paquetes.

```
1 library(tidyquant)
2 library(dplyr)
3 library(knitr)
4 library(tidyr)
5 library(ggplot2)
```

3 Descarga de datos.

```
1 S <- tq_get("GSPC", from = "2015-07-10", to = "2020-07-10") |>
2   dplyr::select(date, S = close)
3
4 kable(rbind(head(S, 6), tail(S, 2)), digits = 10, format.args = list(scientific = FALSE))
```

date	S
2015-07-10	2076.62
2015-07-13	2099.60
2015-07-14	2108.95
2015-07-15	2107.40
2015-07-16	2124.29
2015-07-17	2126.64
2020-07-08	3169.94
2020-07-09	3152.05

4 Visualización del índice.

```
1 ggplot(S, aes(x = date, y = S)) +
2   geom_line(color = "blue") +
3   labs(title = "Índice S&P 500.", x = "Fecha", y = "Valor") +
4   scale_x_date(date_breaks = "1 year", date_labels = "%Y") +
5   theme_minimal(base_size = 13) +
6   theme(plot.title = element_text(face = "bold", hjust = 0.5),
7         axis.text.x = element_text(angle = 0, hjust = 0.5))
```



5 Rendimientos logarítmicos (l) y en cambios porcentuales (c).

$$u_{l,i} = \ln\left(\frac{S_i}{S_{i-1}}\right) \rightarrow u_{l,2} = \ln\left(\frac{2099.60}{2076.62}\right)$$

$$\rightarrow \ln(1.011066) \rightarrow u_{l,2} = 0.0110052685.$$

$$u_{c,i} = \frac{S_i - S_{i-1}}{S_{i-1}} \rightarrow u_{c,2} = \frac{S_i}{S_{i-1}} - 1$$

$$\rightarrow u_{c,2} = \frac{2099.60}{2076.62} - 1 \rightarrow u_{c,2} = 0.0110660492.$$

```

1 S <- S |>
2   mutate(i = dplyr::row_number(),
3         u_l = log(S / lag(S)),
4         u_c = (S / lag(S)) - 1)
5
6 kable(rbind(head(S, 6), tail(S, 2)), digits = 10, format.args = list(scientific = FALSE))

```

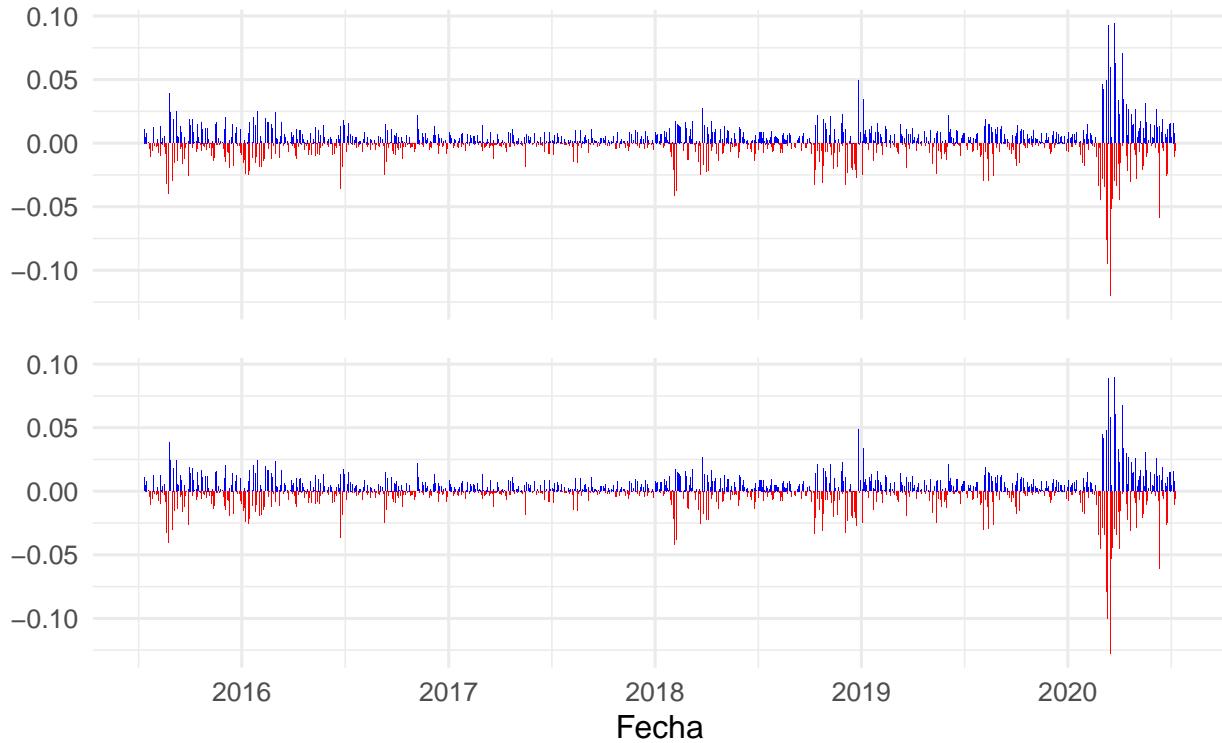
	date	S	i	u_l	u_c
	2015-07-10	2076.62	1	NA	NA
	2015-07-13	2099.60	2	0.0110052685	0.0110660492
	2015-07-14	2108.95	3	0.0044432732	0.0044531592
	2015-07-15	2107.40	4	-0.0007352563	-0.0007349861
	2015-07-16	2124.29	5	0.0079827334	0.0080146804
	2015-07-17	2126.64	6	0.0011055716	0.0011061830
	2020-07-08	3169.94	1258	0.0077969863	0.0078274619
	2020-07-09	3152.05	1259	-0.0056595915	-0.0056436062

6 Visualización de los rendimientos.

```
1 rtns_long <- S |>
2   dplyr::select(date, u_l, u_c) |>
3   tidyverse::pivot_longer(c(u_l, u_c), names_to = "serie", values_to = "retorno") |>
4   dplyr::mutate(signo = retorno >= 0)
5
6   ylim <- range(rtns_long$retorno, na.rm = TRUE)
7
8   ggplot(rtns_long, aes(x = date, y = retorno, fill = signo)) +
9     geom_col(show.legend = FALSE) +
10    scale_fill_manual(values = c(`TRUE` = "blue", `FALSE` = "red")) +
11    scale_x_date(date_breaks = "1 year", date_labels = "%Y") +
12    scale_y_continuous(limits = ylim) +
13    facet_grid(rows = vars(serie), switch = "y") +
14    labs(title = "Rendimientos diarios del S&P 500.",
15         subtitle = "Arriba: logarítmicos; abajo: cambios porcentuales.",
16         x = "Fecha", y = NULL) +
17    theme_minimal(base_size = 12) +
18    theme(strip.text.y.left = element_blank(),
19          panel.spacing = unit(1, "lines"),
20          axis.ticks.x = element_blank())
```

Rendimientos diarios del S&P 500.

Arriba: logarítmicos; abajo: cambios porcentuales.



7 Volatilidad constante.

$s = \frac{1}{n-1} \sum_{i=1}^n (u_i - \bar{u})^2$. Diferencias al cuadrado: $(u_i - \bar{u})^2$.

El primer valor válido de i es 2 porque al calcular los rendimientos perdemos una observación.

```
1 ul_mean <- mean(S$u_l, na.rm = TRUE)
2 uc_mean <- mean(S$u_c, na.rm = TRUE)
3 S <- S |>
4   mutate(ul_dev2 = (u_l - ul_mean)^2,
5         uc_dev2 = (u_c - uc_mean)^2) |>
6   select(date, i, ul_dev2, uc_dev2)
7
8 kable(rbind(head(S, 6), tail(S, 2)), digits = 10, format.args = list(scientific = FALSE))
```

date	i	ul_dev2	uc_dev2
2015-07-10	1	NA	NA
2015-07-13	2	0.0001139245	0.0001136209
2015-07-14	3	0.0000169048	0.0000163735
2015-07-15	4	0.0000011385	0.0000013035
2015-07-16	5	0.0000585379	0.0000578808
2015-07-17	6	0.0000005988	0.0000004892
2020-07-08	1258	0.0000557301	0.0000550671
2020-07-09	1259	0.0000358959	0.0000366067

8 Estimación muestral de la volatilidad no condicional.

$$s = \frac{1}{n-1} \sum_{i=1}^n (u_i - \bar{u})^2.$$

```
1 n_valid <- sum(!is.na(S$ul_dev2))
2
3 metrics <- tibble::tibble(series = c("log returns (l)", "pct change (c)" ),
4                               daily_var = c(sum(S$ul_dev2, na.rm = TRUE) / (n_valid - 1),
5                                             sum(S$uc_dev2, na.rm = TRUE) / (n_valid - 1))) |>
6   mutate(daily_vol = sqrt(daily_var), annual_vol = daily_vol * sqrt(252))
7
8 knitr::kable(metrics, digits = c(0, 10, 6, 6), format.args = list(scientific = FALSE),
9               col.names = c("Series", "Daily variance",
10                           "Daily volatility", "Annual volatility"))
10
```

Series	Daily variance	Daily volatility	Annual volatility
log returns (l)	0.0001505991	0.012272	0.194810
pct change (c)	0.0001490365	0.012208	0.193797

9 Conclusión.

- Los rendimientos logarítmicos y porcentuales generan resultados prácticamente equivalentes.
- El enfoque no condicional, asume una volatilidad constante en el tiempo y, por tanto, no captura la naturaleza dinámica y heterocedástica observada en los mercados financieros.