



R para Finanzas

Guía de Ejercicios 5

Profesor: *Víctor Macías E.*

```
library(tidyquant) # Pregunta 1, 2, 3, 6 y 7
library(dplyr)     # Preguntas 2 a 8
library(ggplot2)   # Preguntas 2, 3, 6, 7, y 8
library(ggthemes)  # Pregunta 2
library(lubridate) # Preguntas 3 y 6
library(kableExtra) # Preguntas 3 y 5
library(tidyr)
library(forcats)
```

Pregunta 1

Baja datos del precio de la acción de Shell (SHEL), BP (BP), Exxon Mobil Corporation (XOM) y Chevron Corporation (CVX) para el período 02/01/2018 hasta el 11/04/2025 desde Yahoo! Finance.

```
stock_prices <- tq_get(c("SHEL", "BP", "XOM", "CVX"),
  get = "stock.prices",
  from = "2018-01-02", to = "2025-04-12")
```

Pregunta 2

Construye un gráfico de líneas que muestre la evolución del precio de cierre.

Ejercicio 1: Representa cada acción por una línea de un color diferente

```
stock_prices |>
  ggplot(aes(x = date, y = close, col = symbol)) +
  geom_line() +
  theme_minimal()
```

Ejercicio 2: Representa cada acción por un tipo de línea diferente

```
stock_prices |>
  ggplot(aes(x = date, y = close, linetype = symbol)) +
  geom_line() +
  theme_minimal()
```

Ejercicio 3: Representa cada acción por una línea de un color diferente que es seleccionado manualmente

```
stock_prices |>
  ggplot(aes(x = date, y = close, col = symbol)) +
  geom_line(linewidth = 0.5) +
  scale_colour_manual(name=NULL,
                      values=c(BP = "blue",
                               CVX = "violet",
                               SHEL = "red",
                               XOM = "green"),
                      labels = c(
                        "BP" = "BP",
                        "CVX" = "Chevron Corporation",
                        "SHEL" = "Shell",
                        "XOM" = "Exxon Mobil Corporation")) +
  theme_minimal() +
  theme(legend.position = "top")
```

Ejercicio 4: Representa cada acción separadamente, permitiendo que el eje X sea el mismo para las cuatro acciones

```
stock_prices |>
  ggplot(aes(x = date, y = close)) +
  geom_line(col = "#790808") +
  labs(title = "Evolución diaria de los precios de cierre",
       subtitle = "2018-2025",
       caption = "Fuente: Elaboración propia en base a datos de Yahoo Finance.",
       x = "Fecha",
```

```
y = "Precio de cierre (en dólares)" +
facet_grid(rows = vars(symbol), scales = "free_y") +
theme_minimal()
```

Ejercicio 4.1: ¿Cómo cambia el gráfico si `geom_line(aes(col = symbol))` ?

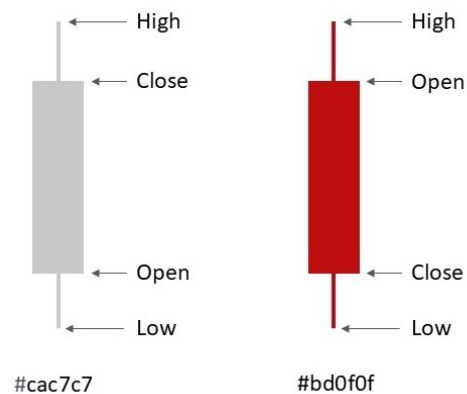
Ejercicio 4.2: Usa otros `themes`. Por ejemplo, `theme_tq` del paquete `tidyquant`.

A continuación se usa `theme_economist()` del paquete `ggthemes`.

```
stock_prices |>
  ggplot(aes(x = date, y = close)) +
  geom_line(col = "#790808") +
  labs(title = "Evolución diaria de los precios de cierre",
        subtitle = "2018-2025",
        caption = "Fuente: Elaboración propia en base a datos de Yahoo Finance.",
        x = "Fecha",
        y = "Precio de cierre (en dólares)") +
  facet_grid(rows = vars(symbol), scales = "free_y") +
  theme_economist()
```

Pregunta 3

Construye un gráficos de velas (*candlestick*), usando `geom_candlestick()`. Sólo incluye precios a contar del 3 de marzo de 2025.



Para entender mejor este tipo de gráficos, a continuación se presentan los precios de las acciones correspondientes al 07 y 08 de abril de 2025:

```
stock_prices |>
  filter(date == ymd("2025-04-07") | date == ymd("2025-04-08")) |>
  select(symbol, date, open, high, low, close) |>
  rename_with(~ stringr::str_to_title(.x)) |>
  kbl(booktabs = T, digits = 2, linesep = "") |>
  kable_styling(font_size = 10,
                latex_options = c("hold_position")) |>
  row_spec(seq(2, 8, 2), bold = T, color="#FFFFFF", background="#bd0f0f") |>
  row_spec(seq(1, 7, 2), bold = T, color="#5D5D5D", background="#CAC7C7")
```

Symbol	Date	Open	High	Low	Close
SHEL	2025-04-07	60.43	64.12	60.33	62.32
SHEL	2025-04-08	63.14	63.56	59.01	59.75
BP	2025-04-07	26.80	28.33	26.43	27.17
BP	2025-04-08	27.90	27.91	25.75	26.11
XOM	2025-04-07	100.39	105.80	98.46	102.94
XOM	2025-04-08	105.46	105.60	99.01	100.77
CVX	2025-04-07	138.73	146.19	137.29	140.15
CVX	2025-04-08	144.15	144.49	134.87	136.93

```
stock_prices |>
  filter(date >= ymd("2025-03-03")) |>
  ggplot(aes(x = date, y = close)) +
  geom_candlestick(aes(open = open, high = high, low = low, close = close),
                  colour_up = "#CAC7C7", fill_up = "#CAC7C7",
                  colour_down = "#bd0f0f", fill_down = "#bd0f0f") +
  labs(title = "Evolución de los precios",
        subtitle = "03 de Marzo al 11 de Abril de 2025",
        caption = "Fuente: Elaboración propia en base a datos de Yahoo Finance.",
        x = NULL,
        y = "Precio de la acción (en dólares)") +
  facet_grid(rows = vars(symbol), scales = "free_y") +
  theme_tq()
```

Pregunta 4

Calcule el retorno diario de cada acción, usando el retorno aritmético.

```
# Cálculo de retornos diarios
retornos_diarios <- stock_prices |>
```

```
group_by(symbol) |>
mutate(daily_retorno = ((close - lag(close))/lag(close))*100) |>
ungroup()
```

Pregunta 5

Construye una tabla donde se muestre el número de observaciones, media aritmética, desviación estándar, valor mínimo, percentil 5, percentil 25, percentil 50 (mediana), percentil 75 y valor máximo de los retornos diarios para cada una de las acciones. Interpreta el valor de los percentiles 5 y 95.

```
retornos_diarios |>
group_by(symbol) |>
summarise(
  n = n(), # Número de observaciones
  mean = mean(daily_retorno, na.rm = TRUE), # Media aritmética
  sd = sd(daily_retorno, na.rm = TRUE), # Desviación estándar
  min = min(daily_retorno, na.rm = TRUE), # Mínimo
  p05 = quantile(daily_retorno, 0.05, na.rm = TRUE), # Percentil 5
  p25 = quantile(daily_retorno, 0.25, na.rm = TRUE), # Percentil 25 o primer cuartil
  p50 = median(daily_retorno, na.rm = TRUE), # Percentil 50 o segundo cuartil
  p75 = quantile(daily_retorno, 0.75, na.rm = TRUE), # Percentil 75 o tercer cuartil
  p95 = quantile(daily_retorno, 0.95, na.rm = TRUE),
  max = max(daily_retorno, na.rm = TRUE)) |> # Máximo
kbl(booktabs = T, digits = 3, linesep = "", escape = FALSE) |>
kable_styling(font_size = 10, latex_options = c("striped", "hold_position"))
```

symbol	n	mean	sd	min	p05	p25	p50	p75	p95	max
BP	1830	-0.002	2.154	-19.104	-3.177	-0.998	0.000	0.943	2.999	21.605
CVX	1830	0.024	2.036	-22.125	-2.797	-0.845	0.084	0.929	2.843	22.741
SHEL	1830	0.017	2.089	-17.172	-2.902	-0.944	0.046	0.949	3.006	19.679
XOM	1830	0.030	1.961	-12.225	-2.998	-0.996	0.009	1.068	3.031	12.687

Pregunta 6

Construye un gráfico de líneas que muestre la evolución de los retornos diarios de las cuatro acciones en el período 02/01/2020 al 08/04/2025.

```
retornos_diarios |>
  filter(date >= ymd("2020-01-02")) |>
  ggplot(aes(x=date, y=daily_retorno)) +
  geom_line(col = "#646464") +
  labs(title = "Evolución de los retornos diarios",
        subtitle = "2020-2025",
        caption = "Fuente: Elaboración propia en base a datos de Yahoo Finance.",
        x = NULL,
        y = "Retorno diario (%)") +
  facet_grid(rows=vars(symbol), scales="free_y") +
  theme_tq()
```

Pregunta 7

Construye un histograma de los retornos diarios.

```
retornos_diarios |>
  ggplot(aes(x=daily_retorno)) +
  geom_histogram(aes(y=after_stat(density),fill=symbol), bins = 40,
                 col="#FFFFFF") +
  labs(title = "Distribución de los retornos diarios",
        subtitle = "2018-2025",
        caption = "Fuente: Elaboración propia en base a datos de Yahoo Finance.",
        x = "Retorno diario (%)",
        y = "Densidad") +
  facet_wrap(~symbol, ncol = 2, scales = "free") +
  theme_tq() +
  scale_fill_tq() +
  theme(legend.position="none")
```

Pregunta 8

Construye un diagrama de caja que muestre la distribución de los retornos diarios.

```
retornos_diarios |>
  ggplot(aes(x=symbol, y = daily_retorno)) +
  geom_boxplot(fill = "#cf902f",
               outlier.colour = "#840718",
               outlier.size = 0.7) +
  labs(title = "Distribución de los retornos diarios",
```

```

    subtitle = "2018-2025",
    caption = "Fuente: Elaboración propia en base a datos de Yahoo Finance.",
    x = NULL,
    y = "Retorno diario (%)" +
coord_flip() +
theme_minimal()

```

Pregunta 9

Calcula la matriz de correlación de los retornos diarios de las 4 empresas.

```

retornos_diarios |>
  select(date, symbol, daily_retorno) |>
  pivot_wider(names_from = symbol, values_from = daily_retorno) |>
  select(-date) |>
  cor(use = "complete.obs")

```

Pregunta 10

Construye un gráfico de dispersión (*scatter plot*) que muestre la relación entre los retornos diarios de *BP* y *CVX*.

```

retornos_diarios |>
  select(date, symbol, daily_retorno) |>
  pivot_wider(names_from = symbol, values_from = daily_retorno) |>
  ggplot(aes(x = BP, y = CVX)) +
    geom_hline(yintercept = 0, col = "#a5a5a5") +
    geom_vline(xintercept = 0, col = "#a5a5a5") +
    geom_point(col = "#98002e", alpha = 0.4, size = 0.2) +
    geom_smooth(method = "lm", se = FALSE, linetype = "dashed",
                linewidth = 0.5, col = "#434351") +
  labs(title = "Relación entre retornos diarios de BP y CVX",
       caption = "Fuente: Elaboración propia en base a datos de Yahoo Finance.",
       x = "Retorno diario de BP (%)",
       y = "Retorno diario de CVX (%)") +
  theme_minimal()

```

Pregunta 11

Construye un gráfico de barras que muestre el promedio de los retornos diarios por mes y compañía.

```
retornos_diarios |>
  mutate(
    month = month(date, label = TRUE, abbr = FALSE)) |>
  group_by(symbol, month) |>
  summarise(nobs = n(),
            retorno_promedio = mean(daily_retorno, na.rm = TRUE)) |>
  ungroup() |>
  ggplot(aes(x = fct_rev(month), y = retorno_promedio, fill = symbol)) +
  geom_bar(stat = "identity",
           position = position_dodge2(preserve = "single", padding = 0.1)) +
  labs(title = "Promedios mensuales de los retornos diarios",
       subtitle = "2019-2025",
       caption = "Fuente: Elaboración propia en base a datos de Yahoo Finance.",
       x = NULL,
       y = "Promedio de retorno diario (%)") +
  scale_fill_brewer(name = NULL, palette = "Set2") +
  coord_flip() +
  theme_minimal() +
  theme(legend.position = "top")
```

Pregunta 12

Construye un gráfico de barras donde se muestre el día de la semana con un mayor porcentaje de días con retornos negativos.

```
ret_negativo <-
  retornos_diarios |>
  filter(!is.na(daily_retorno)) |>
  mutate(dia_semana = wday(date, label = TRUE, abbr = FALSE)) |>
  group_by(dia_semana) |>
  summarise(nobs = n(),
            days_negativo = sum(daily_retorno < 0, na.rm = TRUE),
            retorno_negativo = 100*mean(daily_retorno < 0, na.rm = TRUE)) |>
  ungroup() |>
  mutate(porcentaje = 100*days_negativo/sum(days_negativo))
```



```
ret_negativo |>
  ggplot(aes(x = dia_semana, y = retorno_negativo)) +
  geom_bar(stat = "identity", fill = "#8c3218") +
  labs(title = "Porcentaje de días con retornos negativos",
        subtitle = "2019-2025",
        caption = "Fuente: Elaboración propia en base a datos de Yahoo Finance.",
        x = NULL,
        y = "Porcentaje de días") +
  theme_minimal()
```

```
ret_negativo |>
  ggplot(aes(x = dia_semana, y = retorno_negativo)) +
  geom_bar(stat = "identity", fill = "#8c3218") +
  labs(title = "Porcentaje de días con retornos negativos",
        subtitle = "2019-2025",
        caption = "Fuente: Elaboración propia en base a datos de Yahoo Finance.",
        x = NULL,
        y = NULL) +
  geom_text(aes(label= paste0(round(retorno_negativo,1),"%")),
            vjust=-0.5,
            size= 3) +
  scale_y_continuous(expand = c(0,0), limits = c(0, 55)) +
  theme(rect=element_blank(),
        axis.text.y=element_blank(),
        axis.ticks=element_blank())
```