

Rendimientos accionarios en R.

Descarga y visualización.

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

26 de octubre de 2025, 02:15 a.m.

	Fundamental	Intermedio	Especializado
Finanzas	✓	×	×
Estadística	✓	×	×
R	×	✓	×

1 Introducción.

En negocios necesitamos datos para evaluar y fundamentar decisiones.

- **Partimos de un propósito.** Un problema que resolver, una hipótesis que validar, un objetivo que alcanzar o una tarea que ejecutar.
- **Datos.** Sin información, nuestras conclusiones serían simples opiniones.
- **Analizamos y modelamos.** Usamos los datos para analizar, estimar o entrenar modelos que permitan un análisis empírico riguroso.
- **Generamos evidencia.** Los resultados nos ayudan a validar hipótesis, resolver problemas y generar nuevo conocimiento para tomar decisiones.

2 Packages.

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

3 Descargar precios y transformar a rendimientos.

```
1 symbols <- c("GOOGL", "META", "WMT", "KO", "NKE")
2
3 annual_returns <- tq_get(symbols,
4                           from = "2020-01-01",
5                           to   = "2024-12-31") |>
6   group_by(symbol) |>
7   tq_transmute(select      = adjusted,
8                 mutate_fun = periodReturn,
9                 period     = "yearly",
10                type       = "arithmetic",
11                col_rename  = "annual_return") |>
12   mutate(year = year(date),
13          annual_return = round(annual_return, 4)) |>
14   select(-date) |>
15   arrange(symbol, year)
16
17 annual_returns
```

```
## # A tibble: 25 x 3
## # Groups:   symbol [5]
##   symbol annual_return year
##   <chr>         <dbl> <dbl>
## 1 GOOGL         0.280  2020
## 2 GOOGL         0.653  2021
## 3 GOOGL        -0.391  2022
## 4 GOOGL         0.583  2023
## 5 GOOGL         0.374  2024
## 6 KO            0.0314 2020
## 7 KO            0.114  2021
## 8 KO            0.106  2022
## 9 KO           -0.0443 2023
## 10 KO           0.0848 2024
## # i 15 more rows
```

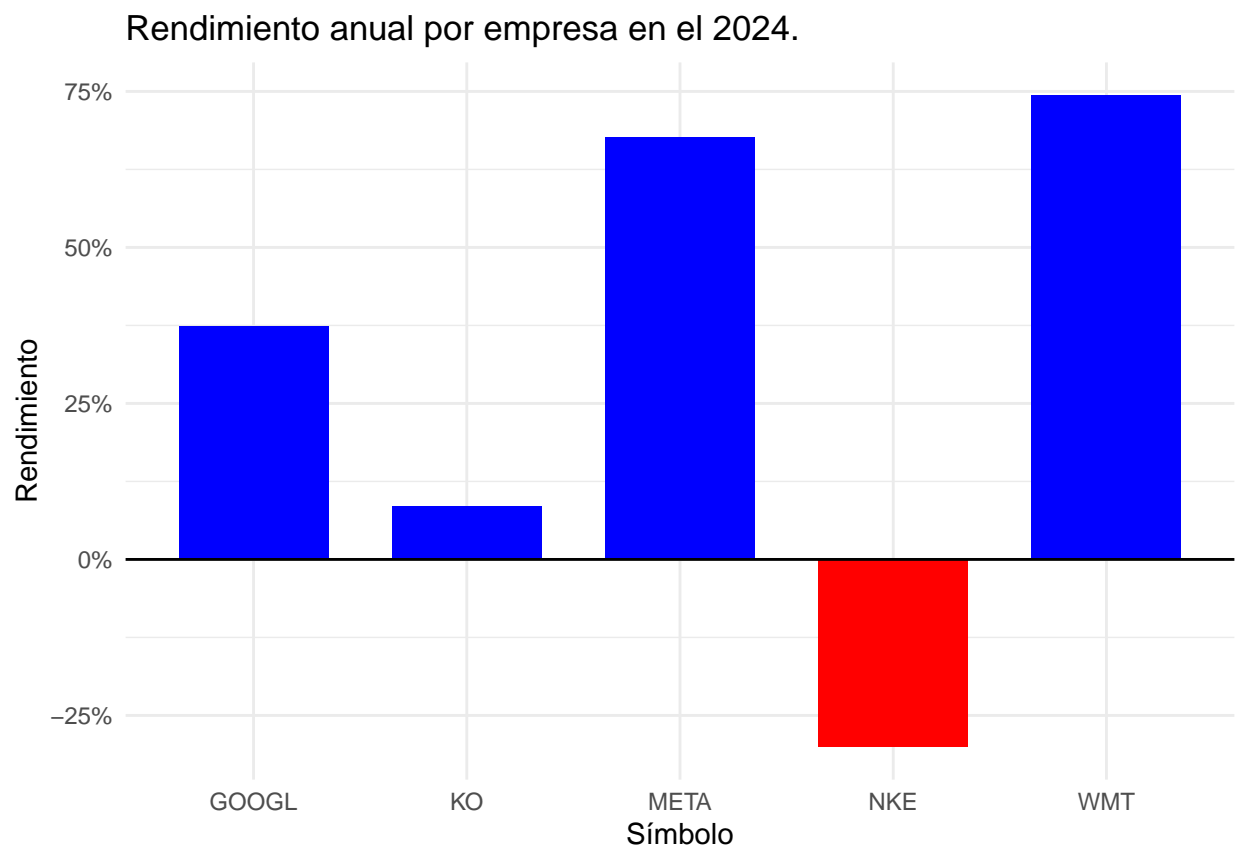
4 Visualizar en tabla.

```
1 annual_returns_wide <- annual_returns |>
2   mutate(year = as.character(year)) |>
3   pivot_wider(names_from = year, values_from = annual_return)
4
5 annual_returns_wide
```

```
## # A tibble: 5 x 6
## # Groups:   symbol [5]
##   symbol `2020` `2021` `2022` `2023` `2024`
##   <chr>   <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 GOOGL  0.280  0.653 -0.391  0.583  0.374
## 2 KO     0.0314 0.114  0.106 -0.0443 0.0848
## 3 META   0.302  0.231 -0.642  1.94   0.677
## 4 NKE    0.397  0.187 -0.290 -0.0601 -0.301
## 5 WMT    0.232  0.0197 -0.0046 0.129  0.744
```

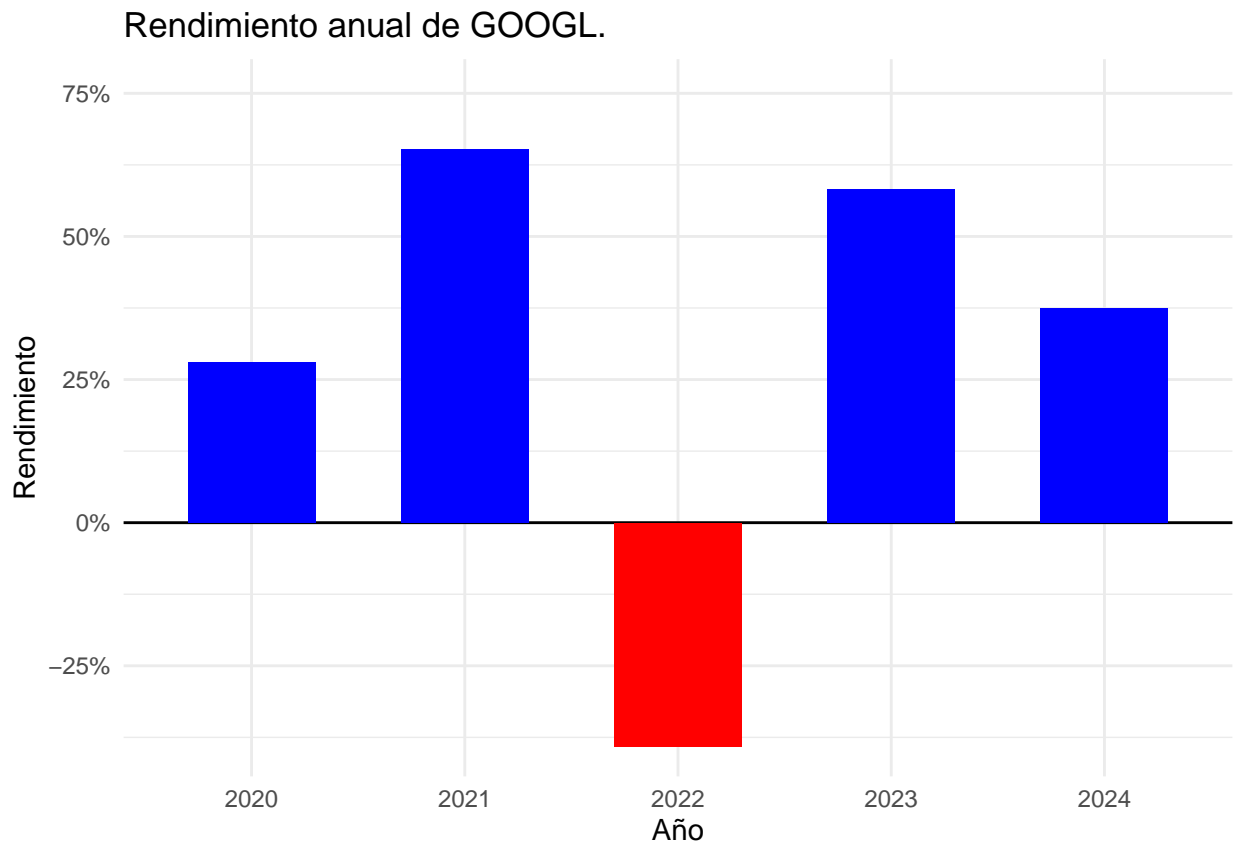
5 Datos en sección cruzada.

```
1 annual_returns |>
2   filter(year == 2024) |>
3   mutate(direction = if_else(annual_return >= 0,
4                               "positive", "negative")) |>
5   ggplot(aes(x = symbol,
6             y = annual_return, fill = direction)) +
7   geom_col(width = 0.7) +
8   geom_hline(yintercept = 0) +
9   scale_fill_manual(values = c(positive = "blue",
10                                negative = "red")) +
11   scale_y_continuous(labels =
12     scales::percent_format(accuracy = 1)) +
13   labs(title = "Rendimiento anual por empresa en el 2024.",
14        x = "Símbolo",
15        y = "Rendimiento") +
16   theme_minimal() +
17   theme(legend.position = "none")
```



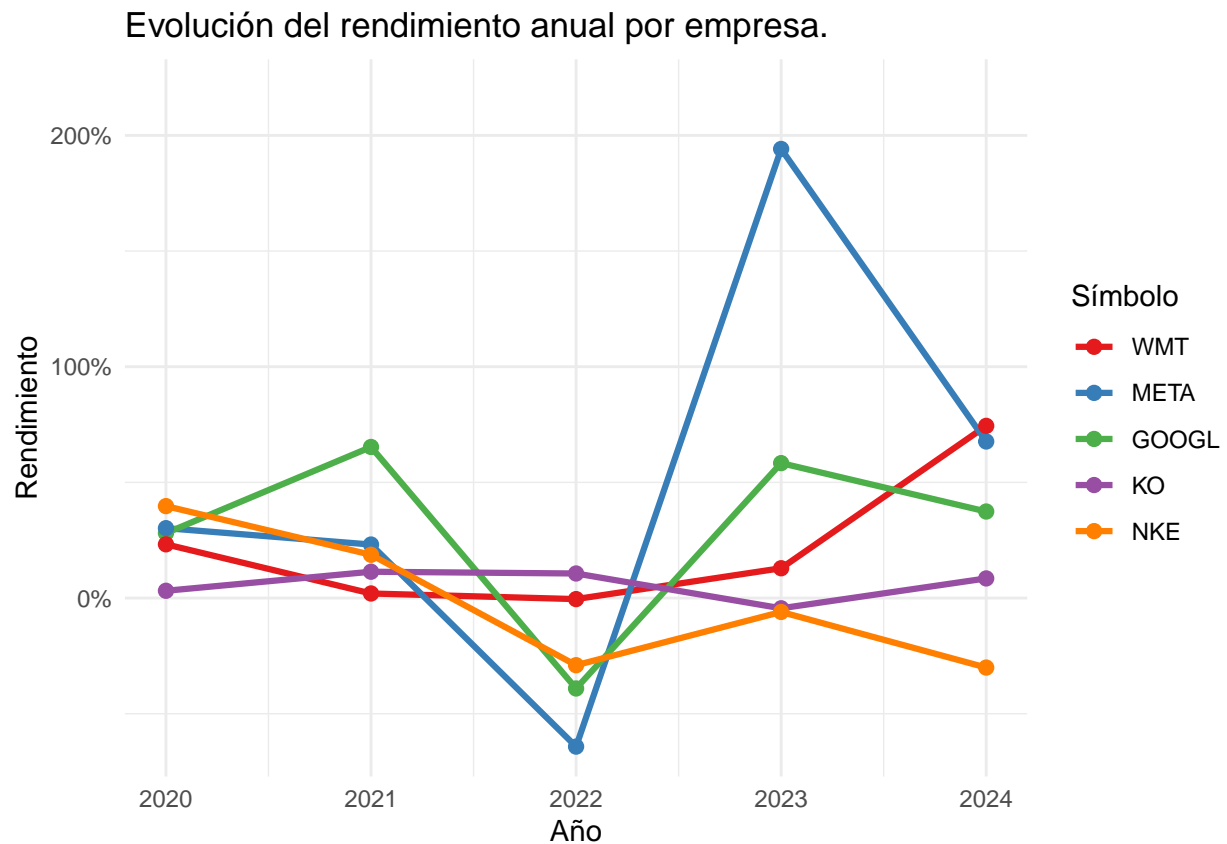
6 Datos en serie de tiempo.

```
1 annual_returns |>
2   ungroup() |>
3   filter(symbol == "GOOGL") |>
4   mutate(direction = if_else(annual_return >= 0,
5                               "positive", "negative")) |>
6   ggplot(aes(x = factor(year),
7               y = annual_return, fill = direction)) +
8   geom_hline(yintercept = 0) +
9   geom_col(width = 0.6) +
10  scale_fill_manual(values = c(positive = "blue",
11                                negative = "red")) +
12  scale_y_continuous(labels =
13                      scales::percent_format(accuracy = 1),
14                      expand = expansion(mult = c(0.05, 0.15))) +
15  labs(title = "Rendimiento anual de GOOGL.",
16       x = "Año",
17       y = "Rendimiento") +
18  theme_minimal() +
19  theme(legend.position = "none")
```



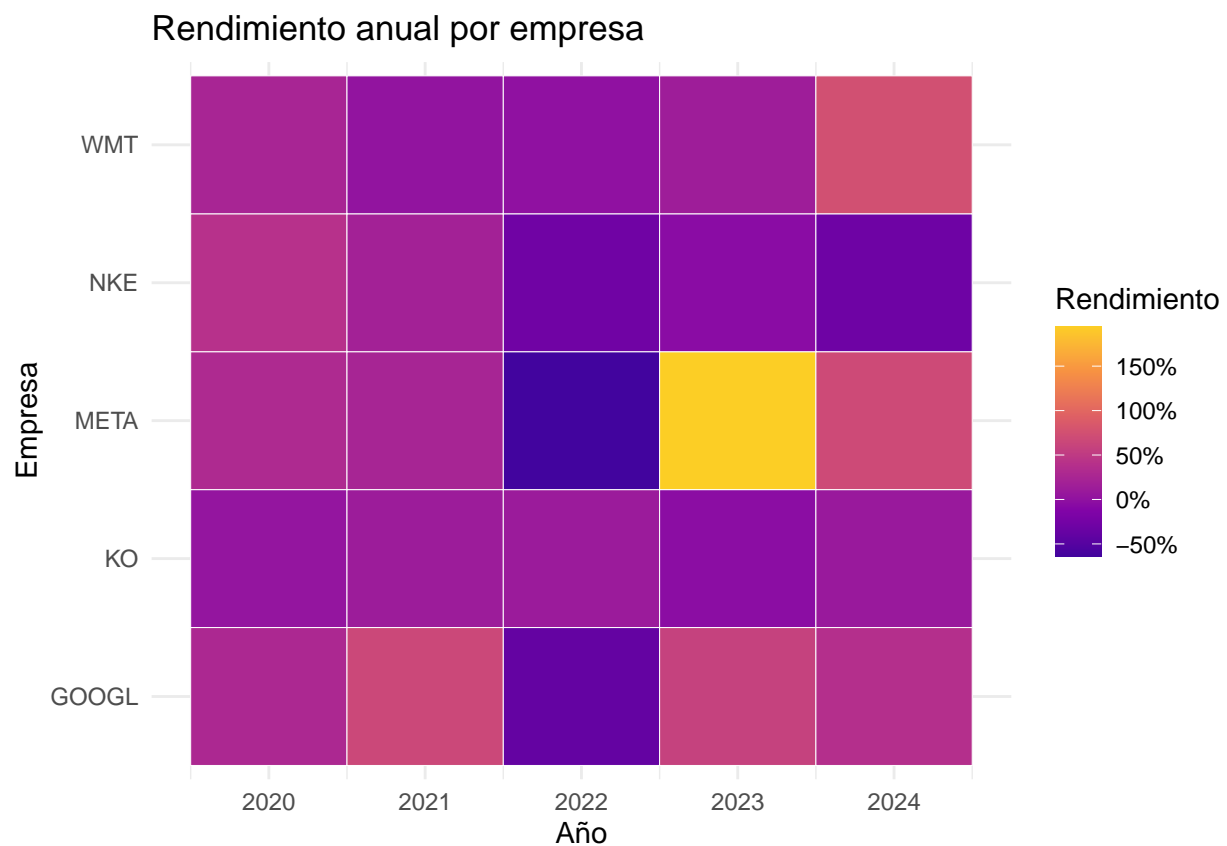
7 Datos de panel.

```
1 legend_order <- annual_returns |>
2   filter(year == 2024) |>
3   arrange(desc(annual_return)) |>
4   pull(symbol)
5
6 annual_returns |>
7   ungroup() |>
8   mutate(symbol = factor(symbol, levels = legend_order)) |>
9   ggplot(aes(x = year, y = annual_return, color = symbol)) +
10  geom_line(linewidth = 1.1) +
11  geom_point(size = 2.2) +
12  scale_color_brewer(palette = "Set1") +
13  scale_x_continuous(breaks = seq(2020, 2024, by = 1)) +
14  scale_y_continuous(labels =
15    scales::percent_format(accuracy = 1),
16    expand = expansion(mult = c(0.05, 0.15))) +
17  labs(title = "Evolución del rendimiento anual por empresa.",
18    x = "Año",
19    y = "Rendimiento",
20    color = "Símbolo") +
21  theme_minimal()
```



8 Datos de panel visualización alternativa.

```
1 annual_returns |>
2   ggplot(aes(x = year, y = symbol, fill = annual_return)) +
3   geom_tile(color = "white") +
4   scale_fill_viridis_c(option = "C",
5     begin = 0.1, end = 0.9,
6     labels =
7       scales::percent_format(accuracy = 1)) +
8   labs(title = "Rendimiento anual por empresa",
9     x = "Año",
10    y = "Empresa", fill = "Rendimiento" ) +
11   theme_minimal()
```



9 Rendimientos acumulados.

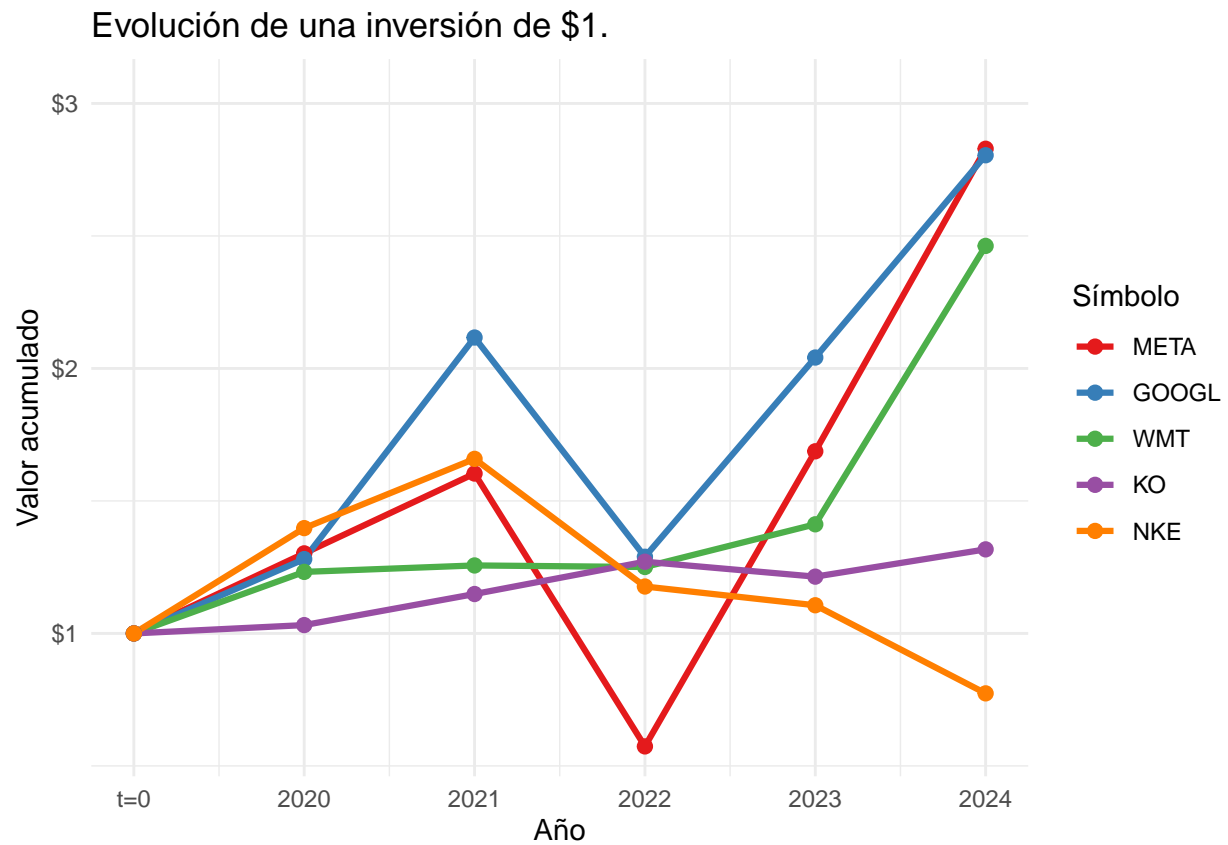
```
1 cumulative_returns <- annual_returns |>
2   ungroup() |>
3   arrange(symbol, year) |>
4   group_by(symbol) |>
5   mutate(cum_value = cumprod(1 + annual_return)) |>
6   ungroup() |>
7   mutate(symbol = factor(symbol, levels = legend_order))
8
9 legend_order_cum <- cumulative_returns |>
10  filter(year == max(year)) |>
11  arrange(desc(cum_value)) |>
12  pull(symbol)
13
14 cumulative_with_start <- cumulative_returns |>
15  mutate(symbol = factor(symbol, levels = legend_order_cum)) |>
16  select(symbol, year, cum_value) |>
17  bind_rows(
18    annual_returns |>
19      distinct(symbol) |>
20      mutate(year = 2019, cum_value = 1,
21            symbol = factor(symbol, levels =
22                          legend_order_cum))) |>
23  arrange(symbol, year)
24
25 cumulative_with_start
```

```
## # A tibble: 30 x 3
##   symbol  year cum_value
##   <fct> <dbl>   <dbl>
## 1 META   2019     1
## 2 META   2020   1.30
## 3 META   2021   1.60
## 4 META   2022   0.574
## 5 META   2023   1.69
## 6 META   2024   2.83
## 7 GOOGL  2019     1
## 8 GOOGL  2020   1.28
## 9 GOOGL  2021   2.12
## 10 GOOGL 2022   1.29
## # i 20 more rows
```

```

1 cumulative_with_start |>
2   ggplot(aes(x = year, y = cum_value, color = symbol)) +
3   geom_line(linewidth = 1.1) +
4   geom_point(size = 2.2) +
5   scale_color_brewer(palette = "Set1") +
6   scale_x_continuous(breaks = c(2019, seq(2020, 2024)),
7                     labels = c("t=0", 2020:2024)) +
8   scale_y_continuous(labels =
9                       scales::dollar_format(accuracy = 1),
10                      expand =
11                        expansion(mult = c(0.05, 0.15))) +
12   labs(title = "Evolución de una inversión de $1.",
13        x = "Año",
14        y = "Valor acumulado",
15        color = "Símbolo") +
16   theme_minimal()

```



10 Rendimientos acumulados visualización alternativa.

```
1 initial_point <- annual_returns |>
2   distinct(symbol) |>
3   mutate(year = 2019,
4           cum_value = 1,
5           symbol = factor(symbol, levels = legend_order_cum))
6
7 cumulative_facets <- cumulative_returns |>
8   mutate(symbol = factor(symbol, levels = legend_order_cum)) |>
9   select(symbol, year, cum_value) |>
10  bind_rows(initial_point) |>
11  arrange(symbol, year) |>
12  mutate(year = factor(year, levels = c(2019, 2020:2024),
13                                     labels = c("t=0", 2020:2024)))
14
15 cumulative_facets
```

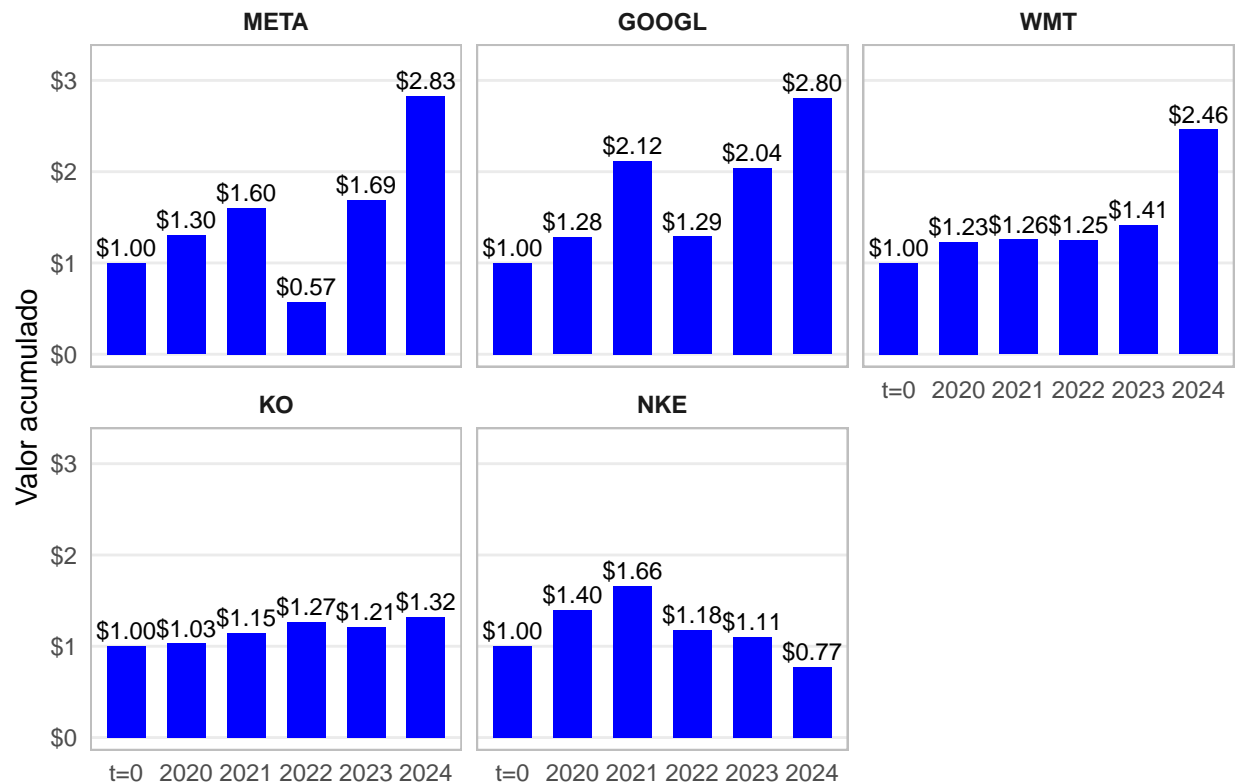
```
## # A tibble: 30 x 3
##   symbol year  cum_value
##   <fct> <fct>    <dbl>
## 1 META  t=0        1
## 2 META  2020      1.30
## 3 META  2021      1.60
## 4 META  2022      0.574
## 5 META  2023      1.69
## 6 META  2024      2.83
## 7 GOOGL t=0        1
## 8 GOOGL 2020      1.28
## 9 GOOGL 2021      2.12
## 10 GOOGL 2022      1.29
## # i 20 more rows
```

```

1 cumulative_facets |>
2   ggplot(aes(x = year, y = cum_value)) +
3     geom_col(width = 0.65, fill = "blue", color = NA) +
4     geom_text(aes(label =
5       scales::dollar(cum_value, accuracy = 0.01)),
6       vjust = -0.4, size = 3) +
7     scale_y_continuous(labels =
8       scales::dollar_format(accuracy = 1),
9       expand =
10        expansion(mult = c(0.05, 0.20))) +
11     facet_wrap(~ symbol, ncol = 3) +
12     labs(title = "Valor acumulado de una inversión de $1.",
13        x = NULL,
14        y = "Valor acumulado") +
15     theme_minimal(base_size = 11) +
16     theme(panel.grid.minor = element_blank(),
17        panel.grid.major.x = element_blank(),
18        panel.border =
19          element_rect(color = "gray", fill = NA,
20            linewidth = 0.6),
21        strip.text = element_text(face = "bold"),
22        legend.position = "none")

```

Valor acumulado de una inversión de \$1.



11 Conclusión.

En negocios necesitamos datos para evaluar y fundamentar decisiones.

- R facilita el análisis financiero reproducible al integrar en un solo flujo la descarga, transformación y visualización de datos con paquetes como `tidyquant`, `dplyr` y `ggplot2`.
- Este enfoque promueve decisiones financieras mejor fundamentadas.