Carteras de inversión.

Introducción al diseño y análisis.

Dr. Martín Lozano `https://mlozanoqf.github.io/`

04 de diciembre de 2025, 04:35 a.m.

	Fundamental	Intermedio	Especializado
Finanzas	$\times$	✓	$\times$
Estadística	✓	$\times$	$\times$
R	$\times$	✓	$\times$

1 Introducción.

- Conceptos fundamentales del análisis riesgo–rendimiento.
- Explicar la frontera media–varianza, destacando cómo se obtiene mediante métodos analíticos y numéricos.
- Construir y evaluar portafolios ingenuos, de mínima varianza, de 2 y 10 activos usando datos reales.
- Análisis cuando se permiten o restringen las ventas en corto.

2 Diez empresas públicas de Estados Unidos.

Ticker	Nombre de la empresa	Industria
AMD	Advanced Micro Dev.	Computación de alto rendimiento
CNC	Centene Corp.	Servicios de salud
GIS	General Mills, Inc.	Productos de consumo envasados
LMT	Lockheed Martin Corp.	Aeroespacial y defensa
LRCX	Lam Research Corp.	Semiconductores
NEM	Newmont Corp.	Minería de metales preciosos
SNPS	Synopsys, Inc.	Software de diseño
SYF	Synchrony Financial	Servicios financieros
TRMB	Trimble Inc.	Tecnología geoespacial
TTD	The Trade Desk, Inc.	Tecnología publicitaria

3 Paquetes.

```
library(tidyquant)
library(tidyverse)
library(lubridate)
library(scales)
library(quadprog)
```

4 Inicialización, parte 1.

```r
# Empresas y rango temporal

tickers <- c("AMD","CNC","GIS","LMT","LRCX","NEM","SNPS","SYF","TRMB","TTD")
n_months <- 60L
price_start <- ymd("2020-09-01")
price_end <- price_start %m+% months(n_months + 1) - days(1)
returns_start <- price_start %m+% months(1)
returns_end <- returns_start %m+% months(n_months) - months(1)

# Descarga de precios históricos de las acciones

prices <- tq_get(tickers, from = price_start, to = price_end, get = "stock.prices")

# Cálculo de retornos mensuales por activo

monthly_returns <- prices |>
  arrange(symbol, date) |>
  group_by(symbol) |>
  tq_transmute(select = adjusted, mutate_fun = periodReturn,
               period = "monthly", type = "arithmetic",
               col_rename = "monthly_return") |>
  filter(between(date, returns_start, returns_end)) |>
  slice_head(n = n_months) |>
  ungroup()

stopifnot(n_distinct(monthly_returns$symbol) == length(tickers))

# Función: estadísticas básicas por activo

asset_stats <- function(data) {
  data |>
    group_by(symbol) |>
    summarise(ER = mean(monthly_return, na.rm = TRUE),
              SD = sd(monthly_return, na.rm = TRUE),
              SR = ER / SD, .groups = "drop")}

# Función: construcción de rendimiento acumulado por activo

cum_index <- function(data, start_date) {
  base <- tibble(date = start_date, symbol = unique(data$symbol), index_level = 1)
  data |>
    arrange(symbol, date) |>
    group_by(symbol) |>
    mutate(index_level = cumprod(1 + monthly_return)) |>
    ungroup() |>
    select(date, symbol, index_level) |>
    bind_rows(base) |>
    arrange(symbol, date)}
```

5 Inicialización, parte 2.

```
# Función: construcción de rendimiento acumulado por cartera

portfolio_series <- function(data, weights, label, start_date) {
  data |>
    filter(symbol %in% names(weights)) |>
    group_by(date) |>
    summarise(portfolio_return = sum(monthly_return * weights[symbol]), .groups = "drop") |>
    arrange(date) |>
    mutate(index_level = cumprod(1 + portfolio_return), symbol = label) |>
    select(date, symbol, index_level) |>
    bind_rows(tibble(date = start_date, symbol = label, index_level = 1))}

# Función: estadísticas de un portafolio

portfolio_stats <- function(data, weights, label) {
  pr <- data |>
    filter(symbol %in% names(weights)) |>
    group_by(date) |>
    summarise(portfolio_return = sum(monthly_return * weights[symbol]), .groups = "drop")

tibble(symbol = label, ER = mean(pr$portfolio_return, na.rm = TRUE),
       SD  = sd(pr$portfolio_return, na.rm = TRUE), SR = ER / SD)}

# Conversión de retornos a formato ancho y matriz

returns_wide <- monthly_returns |>
  select(date, symbol, monthly_return) |>
  pivot_wider(names_from = symbol, values_from = monthly_return) |>
  arrange(date)

returns_mat_all <- returns_wide |>
  select(-date) |>
  as.matrix()

# Cálculo de medias, covarianzas y estadísticas

mu_vec_all  <- colMeans(returns_mat_all, na.rm = TRUE)
cov_mat_all <- cov(returns_mat_all, use = "pairwise.complete.obs")
n_assets <- length(mu_vec_all)

stats_all <- tibble(symbol = names(mu_vec_all), ER = mu_vec_all,
                    SD = apply(returns_mat_all, 2, sd, na.rm = TRUE))

color_values_all <- setNames(scales::hue_pal()(length(mu_vec_all)),
                             names(mu_vec_all))
```

6 Matriz de correlaciones.

```r
corr_mat <- returns_wide |>
  select(-date) |>
  cor(use = "pairwise.complete.obs")

print(round(corr_mat, 2), quote = FALSE, right = TRUE)
```

```
##        AMD   CNC   GIS   LMT  LRCX   NEM  SNPS   SYF  TRMB   TTD
## AMD   1.00 -0.10 -0.16 -0.12  0.71 -0.06  0.70  0.35  0.50  0.51
## CNC  -0.10  1.00  0.37  0.37  0.14  0.20 -0.16  0.12  0.01 -0.18
## GIS  -0.16  0.37  1.00  0.46 -0.20  0.25 -0.27 -0.15 -0.20 -0.20
## LMT  -0.12  0.37  0.46  1.00  0.00  0.32 -0.22  0.20  0.01 -0.10
## LRCX  0.71  0.14 -0.20  0.00  1.00  0.04  0.56  0.59  0.61  0.38
## NEM  -0.06  0.20  0.25  0.32  0.04  1.00 -0.04 -0.08 -0.05 -0.26
## SNPS  0.70 -0.16 -0.27 -0.22  0.56 -0.04  1.00  0.33  0.48  0.31
## SYF   0.35  0.12 -0.15  0.20  0.59 -0.08  0.33  1.00  0.67  0.29
## TRMB  0.50  0.01 -0.20  0.01  0.61 -0.05  0.48  0.67  1.00  0.44
## TTD   0.51 -0.18 -0.20 -0.10  0.38 -0.26  0.31  0.29  0.44  1.00
```

7 Pares de correlaciones mínimas.

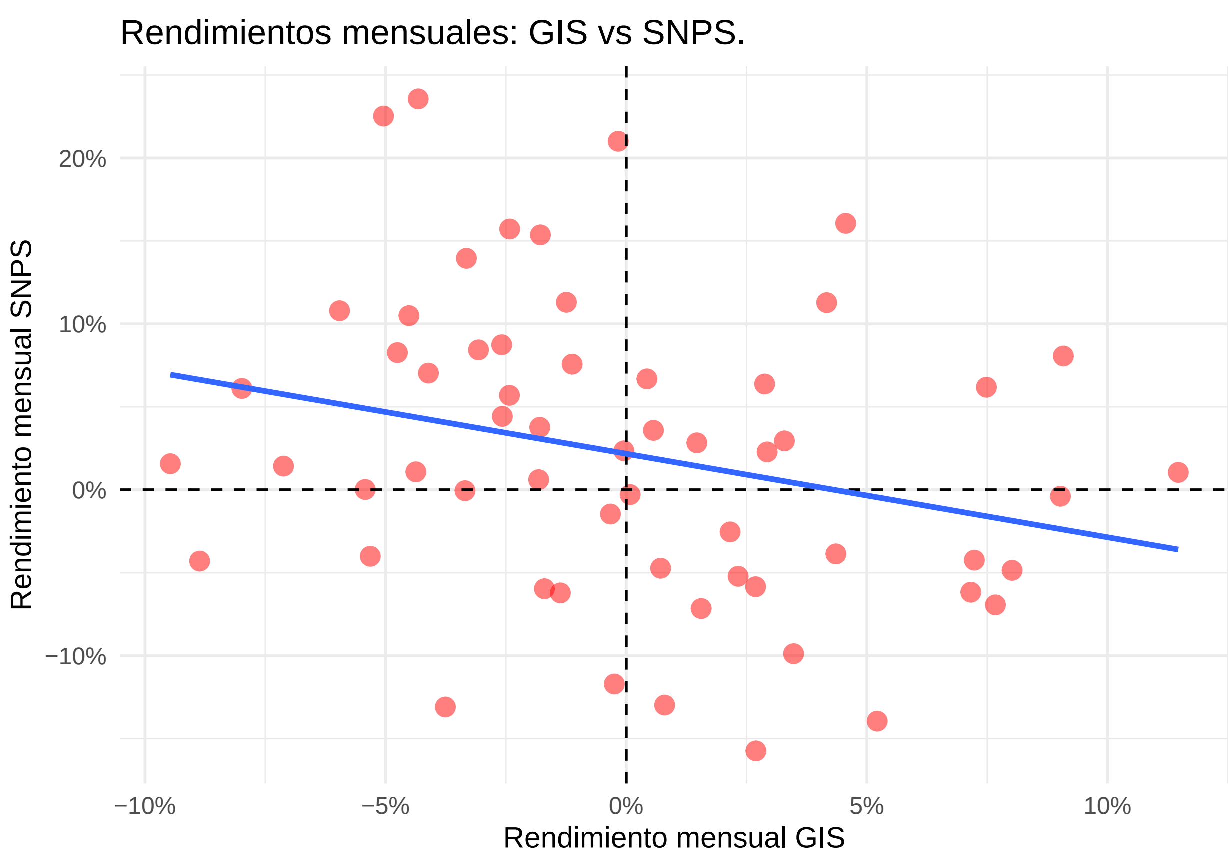
$\sigma^2_{port} = w^2_A\sigma^2_A + w^2_B\sigma^2_B + 2w_Aw_B\sigma_A\sigma_B\rho_{A,B}.$

```r
lowest_pairs <- corr_mat |>
  as_tibble(rownames = "symbol_i") |>
  pivot_longer(-symbol_i, names_to = "symbol_j", values_to = "corr") |>
  filter(symbol_i < symbol_j) |>
  arrange(corr) |>
  slice_head(n = 10)

print(lowest_pairs)
```

```
## # A tibble: 10 x 3
##    symbol_i symbol_j   corr
##    <chr>    <chr>     <dbl>
##  1 GIS      SNPS     -0.266
##  2 NEM      TTD      -0.263
##  3 LMT      SNPS     -0.217
##  4 GIS      TTD      -0.201
##  5 GIS      LRCX     -0.201
##  6 GIS      TRMB     -0.198
##  7 CNC      TTD      -0.180
##  8 AMD      GIS      -0.162
##  9 CNC      SNPS     -0.156
## 10 GIS      SYF      -0.148
```

8 Ilustración de la correlación.

```r
# Pasar a formato ancho con una columna para GIS y otra para SNPS
gis_snps_wide <- monthly_returns |>
  filter(symbol %in% c("GIS", "SNPS")) |>
  select(date, symbol, monthly_return) |>
  pivot_wider(names_from = symbol, values_from = monthly_return) |>
  drop_na()

# Scatter GIS (eje X) vs SNPS (eje Y)
ggplot(gis_snps_wide, aes(x = GIS, y = SNPS)) +
  geom_point(size = 3, alpha = 0.5, col = "red") +
  geom_hline(yintercept = 0, linetype = "dashed") +
  geom_vline(xintercept = 0, linetype = "dashed") +
  scale_x_continuous(labels = percent_format(accuracy = 1)) +
  scale_y_continuous(labels = percent_format(accuracy = 1)) +
  labs(title = "Rendimientos mensuales: GIS vs SNPS.",
    x = "Rendimiento mensual GIS", y = "Rendimiento mensual SNPS") +
  geom_smooth(method = "lm", se = FALSE) +
  theme_minimal()
```



9 Riesgo-rendimiento: 10 activos individuales.

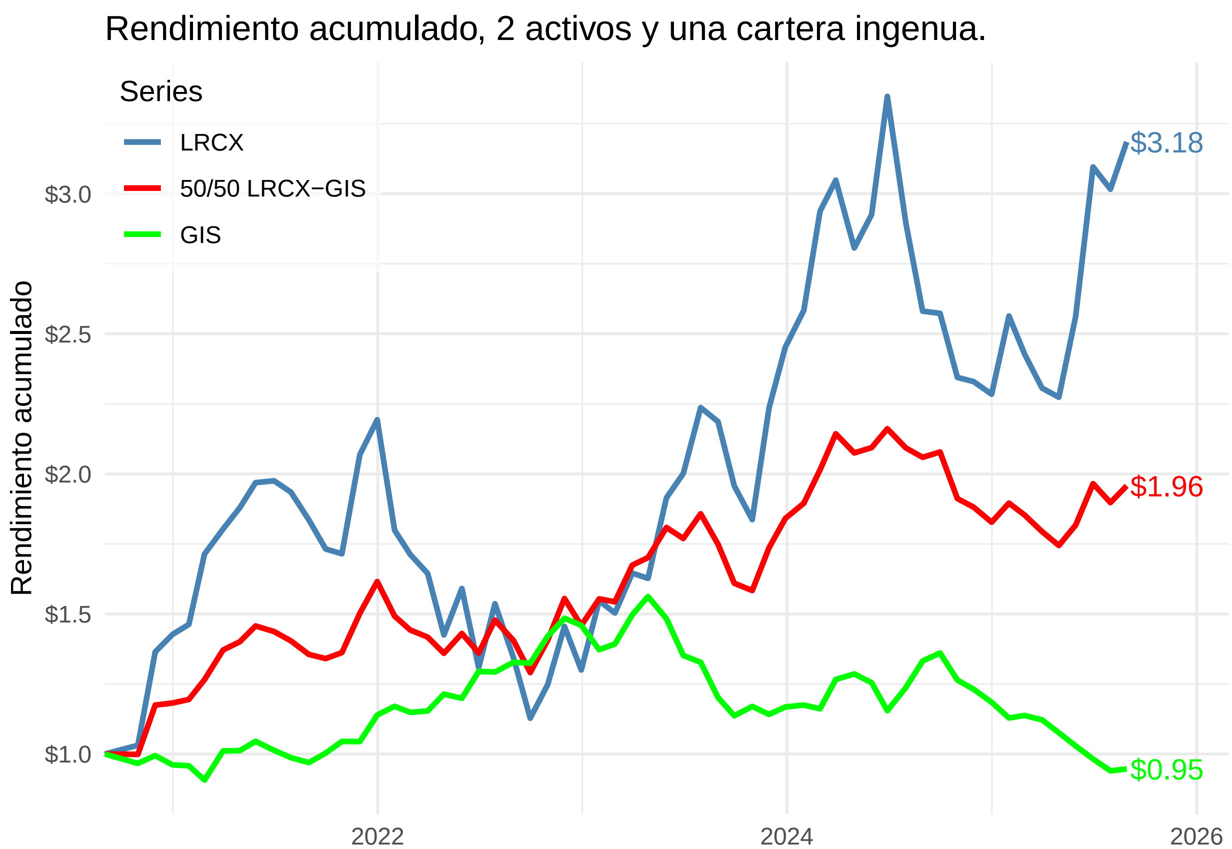
```
stats <- asset_stats(monthly_returns) |>
  arrange(-SR) |>
  print()
```

```
## # A tibble: 10 x 4
##    symbol        ER     SD       SR
##    <chr>      <dbl>  <dbl>    <dbl>
##  1 SYF     0.0259   0.106   0.244
##  2 SNPS    0.0216   0.0902  0.240
##  3 LRCX    0.0257   0.111   0.231
##  4 AMD     0.0223   0.150   0.149
##  5 TRMB    0.0132   0.0970  0.136
##  6 LMT     0.00709  0.0647  0.110
##  7 NEM     0.0107   0.103   0.103
##  8 TTD     0.0170   0.184   0.0926
##  9 GIS     0.000182 0.0477  0.00382
## 10 CNC    -0.00542  0.102  -0.0532
```

10 Rendimiento acumulado, cartera ingenua.

$R_{port} = 0.5\mu_{LRCX} + 0.5\mu_{GIS}$.

```r
focus <- c("LRCX", "GIS")
color_values <- c("LRCX"="steelblue","GIS"="green","50/50 LRCX-GIS"="red")

cum_focus <- monthly_returns |>
  filter(symbol %in% focus) |>
  cum_index(price_start)

portfolio_50_50 <- portfolio_series(monthly_returns, c(LRCX = 0.5, GIS = 0.5),
                                    "50/50 LRCX-GIS", price_start)

plot_data <- bind_rows(cum_focus, portfolio_50_50)

last_vals <- plot_data |>
  group_by(symbol) |>
  summarise(final_value = last(index_level), .groups = "drop") |>
  arrange(desc(final_value))

label_data <- plot_data |>
  group_by(symbol) |>
  filter(date == max(date)) |>
  mutate(label = paste0("$", formatC(index_level, format="f", digits=2))) |>
  ungroup()

ggplot(plot_data, aes(date, index_level,
                      color = factor(symbol, levels = last_vals$symbol))) +
  geom_line(linewidth = 1) +
  geom_text(data = label_data, aes(label = label),
            hjust = -0.05, vjust = 0.5, show.legend = FALSE) +
  scale_color_manual(values = color_values, limits = last_vals$symbol) +
  scale_x_date(expand = expansion(mult = c(0, 0.1))) +
  scale_y_continuous(labels = dollar_format(prefix = "$", accuracy = 0.1)) +
  labs(title = "Rendimiento acumulado, 2 activos y una cartera ingenua.",
       y = "Rendimiento acumulado",
       x = NULL, color = "Series") +
  theme_minimal() +
  theme(legend.position = c(0,1), legend.justification = c(0,1),
        legend.background = element_rect(fill = alpha("white", 0.6), color = NA))
```



Rendimiento acumulado, 2 activos y una cartera ingenua.
Series
LRCX
50/50 LRCX−GIS
GIS
Rendimiento acumulado
$3.0
$2.5
$2.0
$1.5
$1.0
2022
2024
2026
$3.18
$1.96
$0.95

11 Riesgo-rendimiento: cartera ingenua y componentes.

$R_{port} = 0.5\mu_{LRCX} + 0.5\mu_{GIS}$.

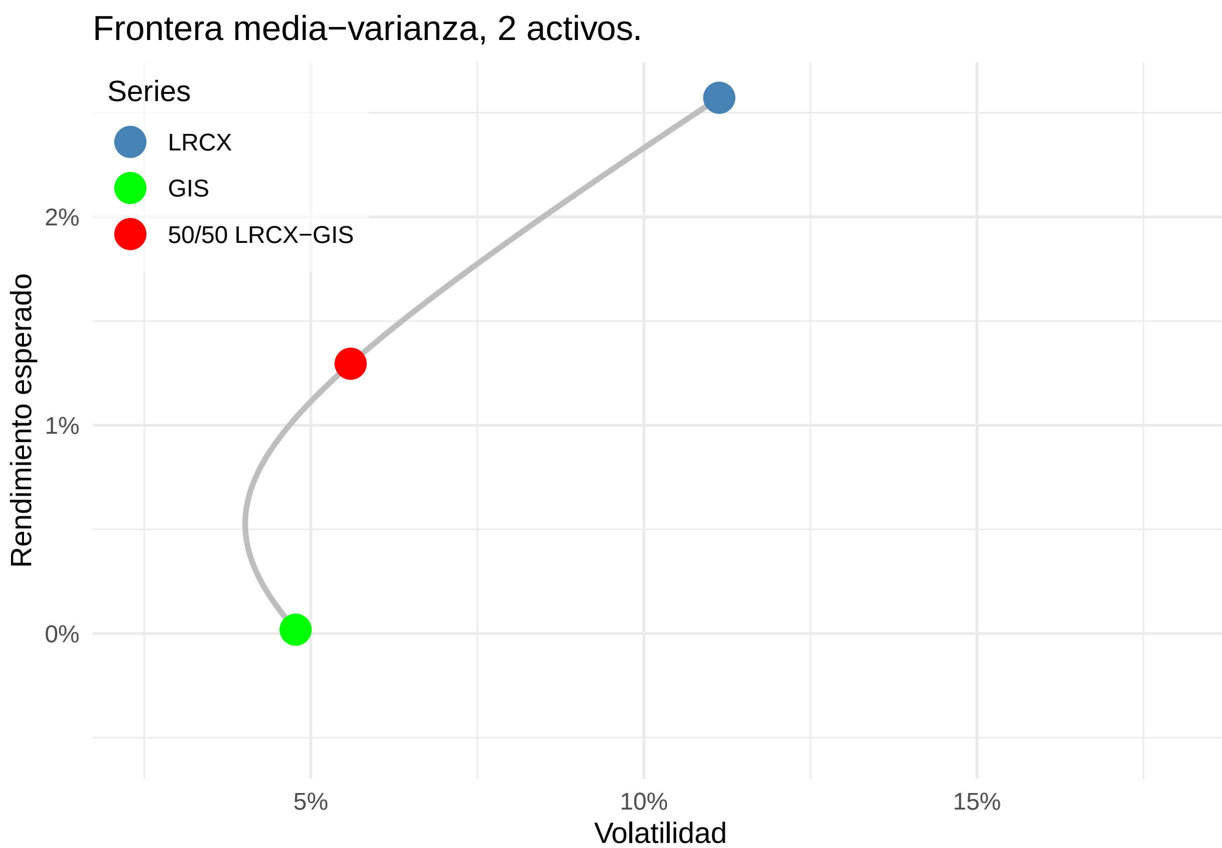
```r
weights_50_50 <- c(LRCX = 0.5, GIS = 0.5)

stats_lrcx_gis <- stats |>
  filter(symbol %in% names(weights_50_50))

portfolio_stats_50_50 <- portfolio_stats(monthly_returns, weights_50_50, "50/50 LRCX-GIS")

final_table <- bind_rows(stats_lrcx_gis, portfolio_stats_50_50) |>
  arrange(-SR)

print(final_table)
```

```
## # A tibble: 3 x 4
##   symbol                ER     SD      SR
##   <chr>              <dbl>  <dbl>   <dbl>
## 1 50/50 LRCX-GIS 0.0130   0.0560 0.231
## 2 LRCX           0.0257   0.111  0.231
## 3 GIS            0.000182 0.0477 0.00382
```

12 Frontera media-varianza: 2 activos.

$$ER_{port} = w_A \mu_A + w_B \mu_B.$$
$$\sigma^2_{port} = w^2_A \sigma^2_A + w^2_B \sigma^2_B + 2 w_A w_B \sigma_{A,B}.$$
$$\sigma^2_{port} = w^2_A \sigma^2_A + w^2_B \sigma^2_B + 2 w_A w_B \sigma_A \sigma_B \rho_{A,B}.$$

```r
returns_matrix_2 <- returns_wide |>
  select(all_of(focus))

cov_mat_2 <- cov(returns_matrix_2, use = "pairwise.complete.obs")
mu_vec_2  <- colMeans(returns_matrix_2, na.rm = TRUE)

frontier_data <- tibble(weight_lrcx = seq(0, 1, length.out = 200)) |>
  mutate(weight_gis = 1 - weight_lrcx,
         ER = weight_lrcx * mu_vec_2["LRCX"] + weight_gis * mu_vec_2["GIS"],
         variance = weight_lrcx^2 * var(returns_matrix_2$LRCX, na.rm = TRUE) +
           weight_gis^2 * var(returns_matrix_2$GIS, na.rm = TRUE) +
           2 * weight_lrcx * weight_gis * cov_mat_2["LRCX", "GIS"],
         SD = sqrt(variance))

mean_var_points <- final_table |>
  mutate(symbol = factor(symbol, levels = c("LRCX", "GIS", "50/50 LRCX-GIS"))) |>
  arrange(symbol)

ggplot() +
  geom_path(data = frontier_data, aes(SD, ER), color = "grey", linewidth = 1) +
  geom_point(data = mean_var_points, aes(SD, ER, color = symbol), size = 5) +
  scale_color_manual(values = color_values, limits = levels(mean_var_points$symbol)) +
    scale_x_continuous(limits = c(0.025, 0.18), labels = percent_format(accuracy = 1)) +
  scale_y_continuous(limits = c(min(mu_vec_all), max(mu_vec_all)),
                     labels = percent_format(accuracy = 1)) +
  labs(title = "Frontera media-varianza, 2 activos.",
       x = "Volatilidad", y = "Rendimiento esperado", color = "Series") +
  theme_minimal() +
  theme(legend.position = c(0,1), legend.justification = c(0,1),
        legend.background = element_rect(fill = alpha("white", 0.6), color = NA))
```


Frontera media−varianza, 2 activos.
Series
LRCX
GIS
50/50 LRCX−GIS
Rendimiento esperado
2%
1%
0%
5%
10%
15%
Volatilidad

13 Cartera de mínima varianza: método numérico de optimización.

$\min_{\mathbf{w}} \ \mathbf{w}^\top \Sigma \mathbf{w} \quad \text{sujeto a:} \quad \mathbf{1}^\top \mathbf{w} = 1.$

```r
minvar2_label <- "Min-Var 2-Asset (LRCX-GIS)"

assets_2 <- focus
qp_2 <- solve.QP(Dmat = 2 * cov_mat_2,
                 dvec = rep(0, length(assets_2)),
                 Amat = matrix(1, nrow = length(assets_2), ncol = 1),
                 bvec = 1, meq = 1)

weights_minvar2_qp <- setNames(qp_2$solution, assets_2)

print(weights_minvar2_qp)
```

```
##      LRCX       GIS
## 0.1990058 0.8009942
```

14 Resumen de componentes y carteras.

```r
weights_50_50 <- c(LRCX = 0.5, GIS = 0.5)

stats_lrcx_gis <- stats |>
  filter(symbol %in% names(weights_50_50))
stats_minvar2_qp <- stats |>
  filter(symbol %in% names(weights_minvar2_qp))

portfolio_stats_50_50 <- portfolio_stats(monthly_returns, weights_50_50, "50/50 LRCX-GIS")
portfolio_stats_minvar2_qp <-
  portfolio_stats(monthly_returns, weights_minvar2_qp, "MV LRCX-GIS")

final_table <- bind_rows(stats_lrcx_gis, portfolio_stats_50_50,
                         portfolio_stats_minvar2_qp) |>
  arrange(-SR)

print(final_table)
```

```
## # A tibble: 4 x 4
##   symbol               ER     SD      SR
##   <chr>             <dbl>  <dbl>   <dbl>
## 1 50/50 LRCX-GIS 0.0130   0.0560 0.231
## 2 LRCX           0.0257   0.111  0.231
## 3 MV LRCX-GIS    0.00526  0.0401 0.131
## 4 GIS            0.000182 0.0477 0.00382
```

15 Cartera de mínima varianza: método analítico algebráico para 2 activos.

$w_1^* = \frac{\sigma_2^2 - \sigma_{12}}{\sigma_1^2 + \sigma_2^2 - 2\sigma_{12}}$, $w_2^* = 1 - w_1^*$.

```r
Sigma_2 <- cov_mat_2[assets_2, assets_2]
var1  <- Sigma_2[1, 1]; var2 <- Sigma_2[2, 2]; cov12 <- Sigma_2[1, 2]
w1 <- (var2 - cov12) / (var1 + var2 - 2 * cov12)
weights_minvar2_closed <- setNames(c(w1, 1 - w1), assets_2)
print(weights_minvar2_closed)
```

```
##      LRCX       GIS
## 0.1990058 0.8009942
```

16 Cartera de mínima varianza: método analítico matricial para 2 activos.

$w^* = \frac{\Sigma^{-1}\mathbf{1}}{\mathbf{1}^\top\Sigma^{-1}\mathbf{1}}$.

```r
Sigma_2 <- cov_mat_2[assets_2, assets_2, drop = FALSE]
ones <- rep(1, length(assets_2))
Sigma_inv <- solve(Sigma_2)
num <- Sigma_inv %*% ones
den <- as.numeric(t(ones) %*% Sigma_inv %*% ones)
w_star <- as.numeric(num / den)
names(w_star) <- assets_2
print(w_star)
```

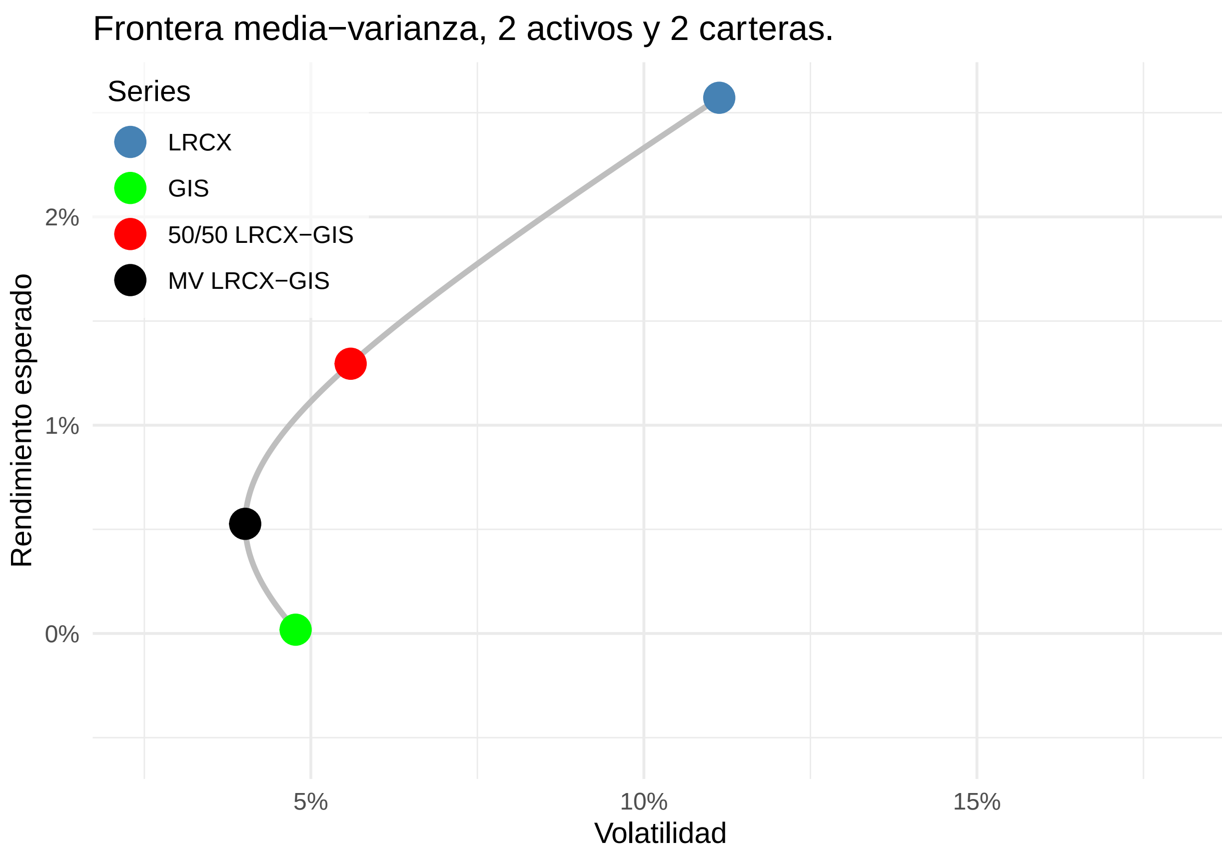
```
##      LRCX       GIS
## 0.1990058 0.8009942
```

17 Frontera media-varianza: 2 activos, cartera MV.

$ER_{port} = \sum_{i=1}^{n} w_i \mu_i$.

$\sigma_{port} = \sqrt{\mathbf{w}^\top \Sigma \mathbf{w}}$.

```r
minvar2_label <- "MV LRCX-GIS"
minvar_point_2 <- tibble(symbol = minvar2_label,
         ER = sum(weights_minvar2_qp * mu_vec_2),
         SD = sqrt(as.numeric(t(weights_minvar2_qp) %*% cov_mat_2 %*% weights_minvar2_qp)))

mean_var_points_ext <- bind_rows(stats_lrcx_gis, portfolio_stats_50_50, minvar_point_2) |>
  mutate(symbol = factor(symbol,
                         levels = c("LRCX", "GIS", "50/50 LRCX-GIS", minvar2_label))) |>
  arrange(symbol)

color_values_ext <- c(color_values, setNames("black", minvar2_label))

ggplot() +
  geom_path(data = frontier_data, aes(SD, ER), color = "grey", linewidth = 1) +
  geom_point(data = mean_var_points_ext, aes(SD, ER, color = symbol), size = 5) +
  scale_color_manual(values = color_values_ext, limits = levels(mean_var_points_ext$symbol)) +
  scale_x_continuous(limits = c(0.025, 0.18), labels = percent_format(accuracy = 1)) +
  scale_y_continuous(limits = c(min(mu_vec_all), max(mu_vec_all)),
                     labels = percent_format(accuracy = 1)) +
  labs(title = "Frontera media-varianza, 2 activos y 2 carteras.",
       x = "Volatilidad", y = "Rendimiento esperado", color = "Series") +
  theme_minimal() +
  theme(legend.position = c(0, 1), legend.justification = c(0, 1),
        legend.background = element_rect(fill = alpha("white", 0.6), color = NA))
```


Frontera media−varianza, 2 activos y 2 carteras.
Series
LRCX
GIS
50/50 LRCX−GIS
MV LRCX−GIS
2%
1%
0%
Rendimiento esperado
5%
10%
15%
Volatilidad

18 Cartera de mínima varianza: método numérico para 10 activos.

$\min_{\mathbf{w}} \ \mathbf{w}^\top \Sigma \mathbf{w} \quad \text{sujeto a:} \quad \mathbf{1}^\top \mathbf{w} = 1, \ \ \mathbf{w} \geq \mathbf{0}.$

```r
minvar10_ns_label <- "MV 10-Asset (No Shorts)"

weights_minvar10_ns <- solve.QP(2 * cov_mat_all, rep(0, length(mu_vec_all)),
           cbind(rep(1, length(mu_vec_all)), diag(length(mu_vec_all))),
           c(1, rep(0, length(mu_vec_all))), meq = 1)$solution

names(weights_minvar10_ns) <- names(mu_vec_all)

minvar_point_10_ns <- tibble(symbol = minvar10_ns_label,
         ER = sum(weights_minvar10_ns * mu_vec_all),
         SD = sqrt(as.numeric(t(weights_minvar10_ns) %*% cov_mat_all%*% weights_minvar10_ns)),
         SR = ER/SD)

minvar_point_10_ns
```

```
## # A tibble: 1 x 4
##   symbol                       ER     SD    SR
##   <chr>                     <dbl>  <dbl> <dbl>
## 1 MV 10-Asset (No Shorts) 0.00717 0.0337 0.213
```

19 Cartera de mínima varianza: método numérico para 10 activos, pesos.

$\min_{\mathbf{w}} \; \mathbf{w}^\top \Sigma \mathbf{w} \quad \text{sujeto a:} \quad \mathbf{1}^\top \mathbf{w} = 1, \quad \mathbf{w} \geq \mathbf{0}.$

```
minvar_weights_table_ns <- tibble(symbol   = names(mu_vec_all),
         weight = weights_minvar10_ns,
         weight_pct = scales::percent(weights_minvar10_ns, accuracy = 0.01)) |>
  arrange(desc(weight)) |>
  bind_rows(tibble(symbol = "Total",
                   weight = sum(weights_minvar10_ns),
                   weight_pct = scales::percent(sum(weights_minvar10_ns),
                                                accuracy = 0.01)))

print(minvar_weights_table_ns)
```

```
## # A tibble: 11 x 3
##    symbol   weight weight_pct
##    <chr>     <dbl> <chr>
##  1 GIS    5.40e- 1 54.01%
##  2 SNPS   1.83e- 1 18.34%
##  3 LMT    1.12e- 1 11.17%
##  4 TRMB   4.62e- 2 4.62%
##  5 NEM    4.52e- 2 4.52%
##  6 SYF    3.05e- 2 3.05%
##  7 TTD    3.00e- 2 3.00%
##  8 CNC    1.30e- 2 1.30%
##  9 AMD    1.31e-17 0.00%
## 10 LRCX   0        0.00%
## 11 Total  1    e+ 0 100.00%
```

20 Cartera de mínima varianza: método analítico matricial para 10 activos, pesos.

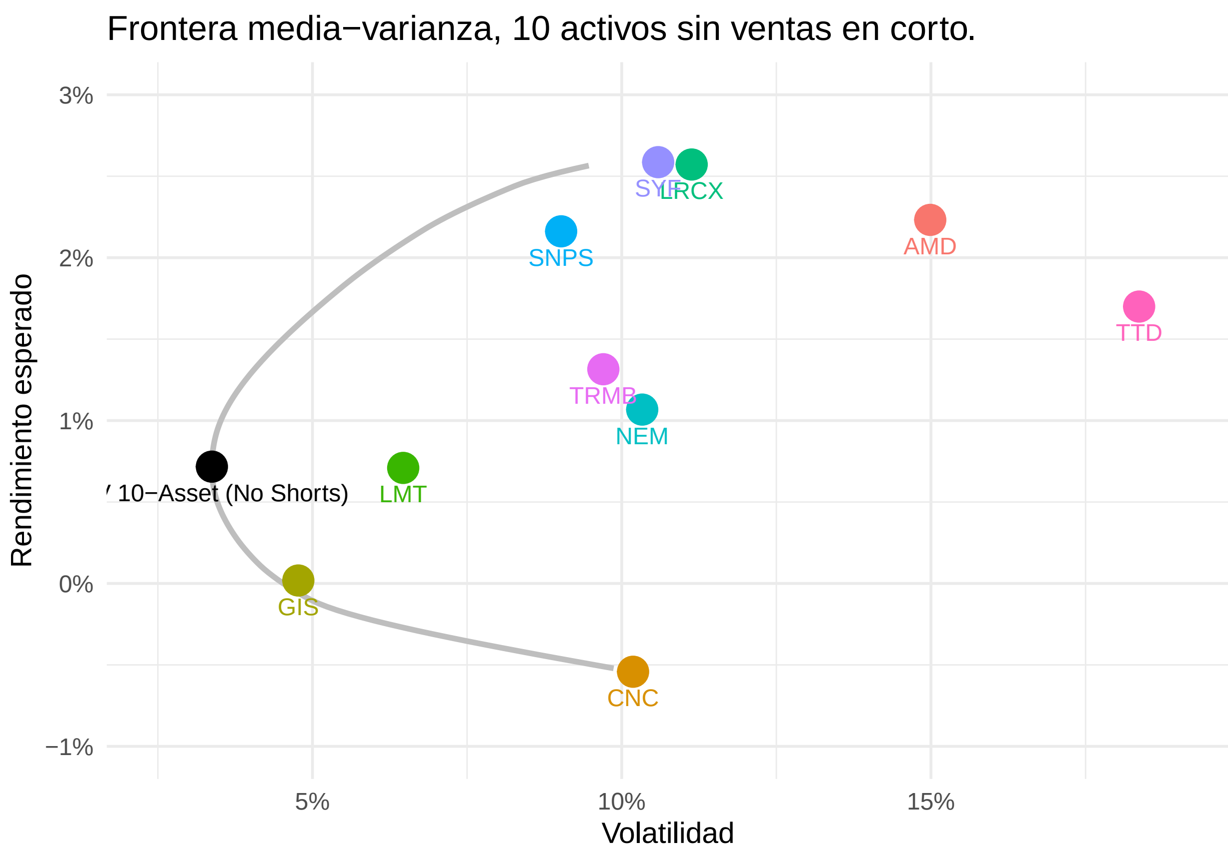
$w^* = \dfrac{\Sigma^{-1}\mathbf{1}}{\mathbf{1}^\top\Sigma^{-1}\mathbf{1}}$. Markowitz, ventas en corto permitidas.

```r
minvar10_short_label <- "MV (shorts)"

Sigma_all  <- cov_mat_all
ones_all   <- rep(1, length(mu_vec_all))
inv_Sigma_all <- solve(Sigma_all)
A_all <- as.numeric(t(ones_all) %*% inv_Sigma_all %*% ones_all)

weights_minvar10_short <- as.numeric(inv_Sigma_all %*% ones_all / A_all)
names(weights_minvar10_short) <- names(mu_vec_all)

minvar_point_10_short <-
  tibble(symbol = minvar10_short_label,
    ER = sum(weights_minvar10_short * mu_vec_all),
    SD = sqrt(as.numeric(t(weights_minvar10_short) %*% Sigma_all %*% weights_minvar10_short)))

weights_gmv_table_short <- tibble(symbol = names(mu_vec_all),
         weight = weights_minvar10_short,
         weight_pct = scales::percent(weights_minvar10_short, accuracy = 0.01)) |>
  arrange(desc(weight)) |>
  bind_rows(tibble(symbol = "Total",
                   weight = sum(weights_minvar10_short),
                   weight_pct = scales::percent(sum(weights_minvar10_short),
                                                accuracy = 0.01)))

print(weights_gmv_table_short)
```

```
## # A tibble: 11 x 3
##    symbol   weight weight_pct
##    <chr>     <dbl> <chr>
##  1 GIS     0.539   53.93%
##  2 SNPS    0.240   24.05%
##  3 LMT     0.106   10.62%
##  4 TRMB    0.0490  4.90%
##  5 TTD     0.0469  4.69%
##  6 NEM     0.0419  4.19%
##  7 LRCX    0.0416  4.16%
##  8 SYF     0.0187  1.87%
##  9 CNC     0.00510 0.51%
## 10 AMD    -0.0893  -8.93%
## 11 Total   1       100.00%
```

21 Frontera media-varianza: 10 activos, sin ventas en corto.

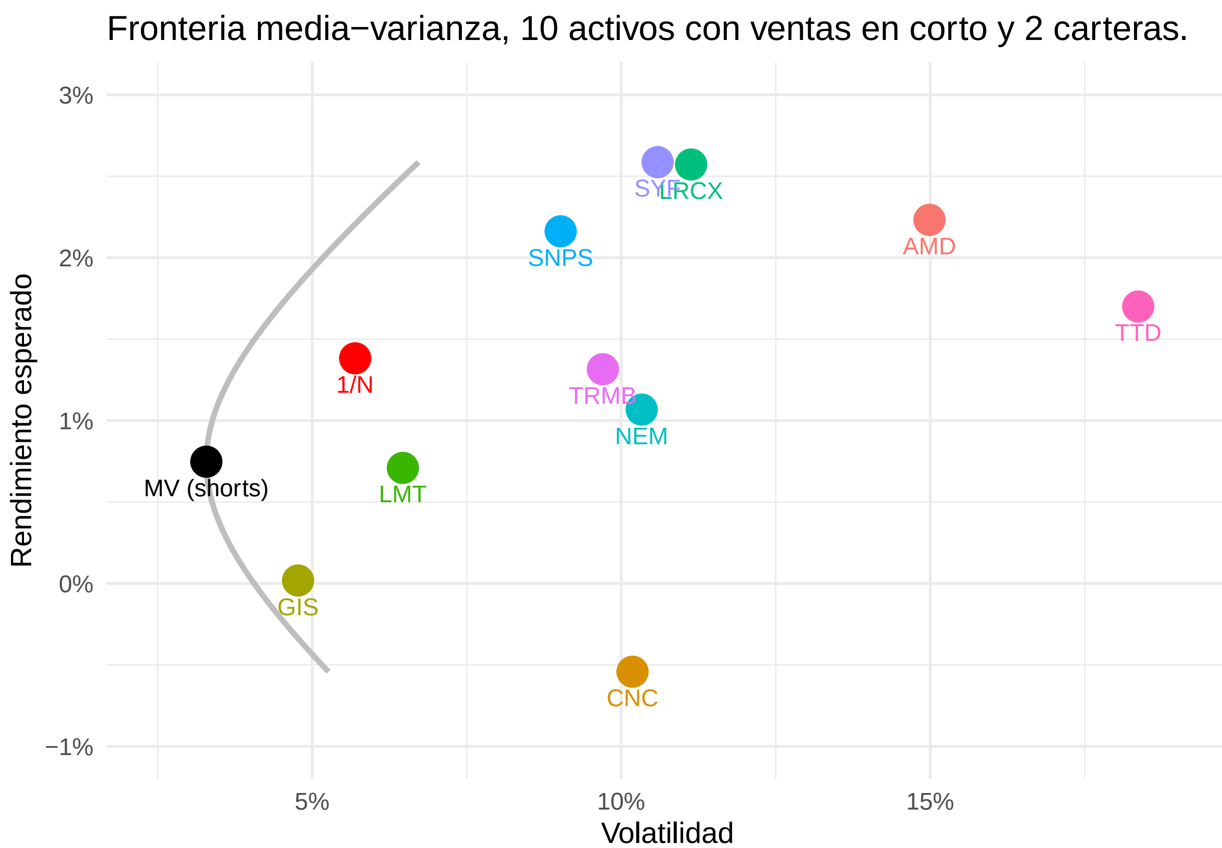
```r
target_returns <- seq(min(mu_vec_all), max(mu_vec_all), length.out = 150)
Amat_ns <- rbind(rep(1, n_assets), mu_vec_all, diag(n_assets))

frontier_points_ns <- map_dfr(target_returns, function(tr) {
  bvec <- c(1, tr, rep(0, n_assets))
  sol <- tryCatch(solve.QP(2 * cov_mat_all, rep(0, n_assets),
                           t(Amat_ns), bvec, meq = 2), error = function(e) NULL)
  if (is.null(sol)) return(NULL)
  tibble(ER = tr,
         SD = sqrt(as.numeric(t(sol$solution) %*% cov_mat_all %*% sol$solution)))}) |>
  drop_na()

stats_all_ext_ns  <- bind_rows(stats_all, minvar_point_10_ns)
color_values_ext_ns <- c(color_values_all, setNames("black", minvar10_ns_label))

ggplot() +
  geom_path(data = frontier_points_ns, aes(SD, ER), color = "grey", linewidth = 1) +
  geom_point(data = stats_all_ext_ns, aes(SD, ER, color = symbol), size = 5) +
  geom_text(data = stats_all_ext_ns, aes(SD, ER, label = symbol, color = symbol),
            vjust = 2, size = 3, show.legend = FALSE) +
  scale_color_manual(values = color_values_ext_ns, limits = stats_all_ext_ns$symbol) +
  scale_x_continuous(limits = c(0.025, 0.19), labels = percent_format(accuracy = 1)) +
  scale_y_continuous(limits = c(-0.01, 0.03), labels = percent_format(accuracy = 1)) +
  labs(title = "Frontera media-varianza, 10 activos sin ventas en corto.",
       x = "Volatilidad", y = "Rendimiento esperado") +
  guides(color = "none") +
  theme_minimal()
```



Frontera media−varianza, 10 activos sin ventas en corto.
3%
2%
1%
0%
−1%
Rendimiento esperado
SYF
LRCX
SNPS
AMD
TTD
TRMB
NEM
' 10−Asset (No Shorts)
LMT
GIS
CNC
5%
10%
15%
Volatilidad

22 Frontera media-varianza: 10 activos, con ventas en corto.

```
Amat_short <- rbind(rep(1, n_assets), mu_vec_all)

  frontier_points_short <- map_dfr(target_returns, function(tr) {
    bvec <- c(1, tr)
    sol <- tryCatch(solve.QP(2 * cov_mat_all, rep(0, n_assets),
                             t(Amat_short), bvec, meq = 2), error = function(e) NULL)
    if (is.null(sol)) return(NULL)
    tibble(ER = tr,
           SD = sqrt(as.numeric(t(sol$solution) %*% cov_mat_all %*% sol$solution)))}) |>
    drop_na()

  # Punto 1/N (igual ponderación)
  weights_1N <- setNames(rep(1 / n_assets, n_assets), names(mu_vec_all))
  point_1N <- tibble(symbol = "1/N", ER = sum(weights_1N * mu_vec_all),
    SD = sqrt(as.numeric(t(weights_1N) %*% cov_mat_all %*% weights_1N)))

  stats_all_ext_short <- bind_rows(stats_all, minvar_point_10_short, point_1N)
  color_values_ext_short <- c(color_values_all,
                              setNames("black", minvar10_short_label), "1/N" = "red")

  ggplot() +
    geom_path(data = frontier_points_short, aes(SD, ER), color = "grey", linewidth = 1) +
    geom_point(data = stats_all_ext_short, aes(SD, ER, color = symbol), size = 5) +
    geom_text(data = stats_all_ext_short, aes(SD, ER, label = symbol, color = symbol),
              vjust = 2, size = 3, show.legend = FALSE) +
    scale_color_manual(values = color_values_ext_short, limits = stats_all_ext_short$symbol) +
    scale_x_continuous(limits = c(0.025, 0.19), labels = percent_format(accuracy = 1)) +
    scale_y_continuous(limits = c(-0.01, 0.03), labels = percent_format(accuracy = 1)) +
    labs(title = "Fronteria media-varianza, 10 activos con ventas en corto y 2 carteras.",
         x = "Volatilidad", y = "Rendimiento esperado") +
    guides(color = "none") +
    theme_minimal()
```


Fronteria media−varianza, 10 activos con ventas en corto y 2 carteras.
Rendimiento esperado
3%
2%
1%
0%
−1%
SYF
LRCX
SNPS
AMD
TTD
1/N
TRMB
NEM
MV (shorts)
LMT
GIS
CNC
5%
10%
15%
Volatilidad

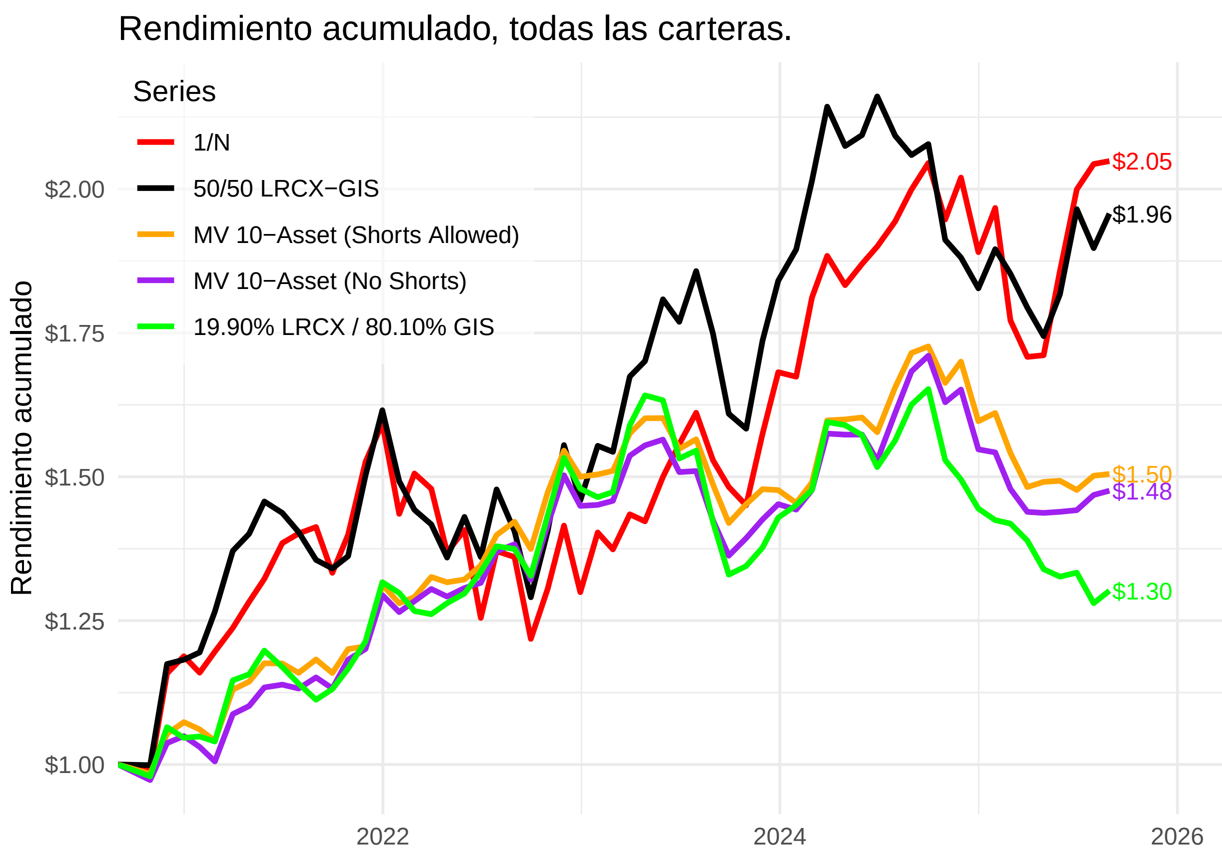
23 Riesgo-rendimiento: 10 activos individuales y carteras.

```r
weights_50_50 <- c(LRCX = 0.5, GIS = 0.5)
  custom_weights  <- c(LRCX = 0.1990055, GIS = 0.8009945)
  portfolio_label <- "19.90% LRCX / 80.10% GIS"

weights_1N <- setNames(rep(1 / n_assets, n_assets), names(mu_vec_all))

weights_specs <- list("50/50 LRCX-GIS" = weights_50_50,
                      "19.90% LRCX / 80.10% GIS" = custom_weights,
                      "MV 10-Asset (No Shorts)" = weights_minvar10_ns,
                      "MV 10-Asset (Shorts Allowed)" = weights_minvar10_short,
                      "1/N" = weights_1N)

  portfolio_stats_ext <- imap_dfr(weights_specs, ~ portfolio_stats(monthly_returns, .x, .y))

  final_table <- bind_rows(asset_stats(monthly_returns), portfolio_stats_ext) |>
    mutate(symbol = factor(symbol, levels = c(tickers, names(weights_specs)))) |>
    arrange(symbol)

  print(final_table)
```

```
## # A tibble: 15 x 4
##    symbol                             ER     SD       SR
##    <fct>                           <dbl>  <dbl>    <dbl>
##  1 AMD                          0.0223   0.150    0.149
##  2 CNC                         -0.00542  0.102   -0.0532
##  3 GIS                          0.000182 0.0477   0.00382
##  4 LMT                          0.00709  0.0647   0.110
##  5 LRCX                         0.0257   0.111    0.231
##  6 NEM                          0.0107   0.103    0.103
##  7 SNPS                         0.0216   0.0902   0.240
##  8 SYF                          0.0259   0.106    0.244
##  9 TRMB                         0.0132   0.0970   0.136
## 10 TTD                          0.0170   0.184    0.0926
## 11 50/50 LRCX-GIS               0.0130   0.0560   0.231
## 12 19.90% LRCX / 80.10% GIS     0.00526  0.0401   0.131
## 13 MV 10-Asset (No Shorts)      0.00717  0.0337   0.213
## 14 MV 10-Asset (Shorts Allowed) 0.00747  0.0329   0.227
## 15 1/N                          0.0138   0.0569   0.243
```

24 Rendimiento acumulado, todas las carteras.

```r
portfolio_cum <- imap_dfr(weights_specs,
                            ~ portfolio_series(monthly_returns, .x, .y, price_start))

  # Último valor real de cada serie (usa la fecha máxima, no la última fila)
  last_values_port <- portfolio_cum |>
    group_by(symbol) |>
    slice_max(order_by = date, n = 1, with_ties = FALSE) |>
    select(symbol, final_value = index_level) |>
    arrange(desc(final_value))

  # Orden de series según rendimiento final (mayor a menor)
  series_levels <- last_values_port$symbol

  palette_base <- c("50/50 LRCX-GIS" = "black",
                    "19.90% LRCX / 80.10% GIS" = "green",
                    "MV 10-Asset (No Shorts)" = "purple",
                    "MV 10-Asset (Shorts Allowed)" = "orange",
                    "1/N" = "red")

  portfolio_colors <- palette_base[series_levels]

  portfolio_cum <- portfolio_cum |>
    mutate(symbol = factor(symbol, levels = series_levels))

  label_data_port <- portfolio_cum |>
    group_by(symbol) |>
    slice_max(order_by = date, n = 1, with_ties = FALSE) |>
    mutate(label = paste0("$", formatC(index_level, format = "f", digits = 2))) |>
    ungroup()

  ggplot(portfolio_cum, aes(date, index_level, color = symbol)) +
    geom_line(linewidth = 1) +
    geom_text(data = label_data_port, aes(label = label),
              hjust = -0.05, vjust = 0.5, size = 3.2, show.legend = FALSE) +
    scale_color_manual(values = portfolio_colors, breaks = series_levels,
                       limits = series_levels, drop = FALSE) +
    scale_x_date(expand = expansion(mult = c(0, 0.12))) +
    scale_y_continuous(labels = dollar_format(prefix = "$")) +
    labs(title = "Rendimiento acumulado, todas las carteras.",
         y = "Rendimiento acumulado", x = NULL, color = "Series") +
    theme_minimal() +
    theme(legend.position = c(0, 1), legend.justification = c(0, 1),
          legend.background = element_rect(fill = alpha("white", 0.6), color = NA))
```


Rendimiento acumulado, todas las carteras.
Series
1/N
50/50 LRCX−GIS
MV 10−Asset (Shorts Allowed)
MV 10−Asset (No Shorts)
19.90% LRCX / 80.10% GIS
Rendimiento acumulado
$2.00
$1.75
$1.50
$1.25
$1.00
2022
2024
2026
$2.05
$1.96
$1.50
$1.48
$1.30

25 Conclusión.

- La diversificación reduce riesgo y permite construir portafolios con mejor relación riesgo–rendimiento que los activos individuales.
- La frontera media–varianza es una herramienta poderosa para visualizar las decisiones de inversión y comprender el intercambio entre riesgo y retorno.