

CAPM como modelo de valuación.

Estimación y evidencia empírica.

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

03 de febrero de 2026, 04:07 a.m.

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

1 Introducción.

- Usaremos datos reales y tres enfoques complementarios (1) serie de tiempo; (2) sección cruzada; y (3) Fama–MacBeth, para contrastar la idea central del CAPM: el retorno esperado en exceso debe estar explicado únicamente por la exposición al factor de mercado.
- Evaluamos si el riesgo sistemático, medido por β , organiza los rendimientos esperados. Bajo el CAPM, el retorno esperado en exceso está determinado por la exposición al factor de mercado.
- Usamos 10 carteras por tamaño de EE.UU. y el factor $Mkt - RF$. Trabajamos con retornos en exceso $r_{p,t} = R_{p,t} - R_{f,t}$ y $r_{M,t} = R_{M,t} - R_{f,t}$; en la práctica, $r_{M,t}$ corresponde al factor $Mkt - RF$ de Kenneth French.
- Las series son mensuales y están expresadas en porcentajes.
- Serie de tiempo: estimamos $r_{p,t} = \alpha_p + \beta_p r_{M,t} + \varepsilon_{p,t}$ para obtener $\hat{\beta}_p$ (exposición al mercado) y $\hat{\alpha}_p$ (desvío ex post). Reportamos errores estándar OLS y HAC para inferencia robusta ante heterocedasticidad y dependencia temporal.
- Propósito: conectar la lógica económica del CAPM con evidencia empírica y mostrar cómo correcciones estándar afectan la inferencia sin alterar la interpretación del modelo.

2 Paquetes.

```
1 library(tidyverse)
2 library(lubridate)
3 library(curl)
4 library(stringr)
5 library(broom)
6 library(dplyr)
7 library(tidyr)
8 library(lmtest)
9 library(sandwich)
10 library(ggplot2)
```

3 Datos 1/2.

```
1 # Descarga y arma capm_data (retornos en exceso) para 10 carteras
2
3 read_size_deciles_vw <- function(url) {
4   tf <- tempfile(fileext = ".zip"); curl_download(url, tf, mode = "wb")
5   target <- str_subset(unzip(tf, list = TRUE)$Name, "Portfolios_Formed_on_ME")[1]
6   lines <- read_lines(unz(tf, target))
7   tag   <- str_which(lines, "Value Weight Returns -- Monthly")[1]
8   data_lines <- gsub(", ", "", lines[(tag + 1):length(lines)])
9   data_lines <- data_lines[str_detect(data_lines, "^\\s*\\d{6}")][
10  cols <- c("date_ym", "leq0", "Lo30", "Med40", "Hi30", "Lo20", "Qnt2", "Qnt3", "Qnt4", "Hi20",
11    "Lo10", "Dec2", "Dec3", "Dec4", "Dec5", "Dec6", "Dec7", "Dec8", "Dec9", "Hi10")
12  read.table(text = paste(data_lines, collapse = "\n"), header = FALSE,
13             col.names = cols, colClasses = c("character", rep("numeric", 19)),
14             sep = "", strip.white = TRUE) |>
15  mutate(date = as.Date(ymd(paste0(date_ym, "01")))) |>
16  select(date, Lo10, Dec2, Dec3, Dec4, Dec5, Dec6, Dec7, Dec8, Dec9, Hi10) |>
17  arrange(date) |>
18  distinct(date, .keep_all = TRUE)}
19
20 read_factors <- function(url) {
21   tf <- tempfile(fileext = ".zip"); curl_download(url, tf, mode = "wb")
22   target <- str_subset(unzip(tf, list = TRUE)$Name, "F-F_Research_Data_Factors")[1]
23   lines <- read_lines(unz(tf, target))
24   first <- which(str_detect(lines, "^\\s*\\d{6}")[1]
25   cn <- str_trim(str_split(lines[first - 1], ",")[[1]]); cn[1] <- "date_ym"
26   read_csv(paste(lines[(first - 1):length(lines)], collapse = "\n"),
27             col_names = cn,
28             col_types = cols(date_ym = col_character(),
29                               `Mkt-RF` = col_double(),
30                               RF = col_double(),
31                               .default = col_skip())) |>
32   filter(str_detect(date_ym, "^(\\d{6})$")) |>
33   mutate(date = as.Date(ymd(paste0(date_ym, "01")))) |>
34   select(date, `Mkt-RF`, RF) |>
35   arrange(date) |>
36   distinct(date, .keep_all = TRUE)}
```

4 Datos 2/2.

```
1 url_size <-
  ↪ "https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/ftp/Portfolios_Formed_on_ME_CSV.zip"
2 url_factors <-
  ↪ "https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/ftp/F-F_Research_Data_Factors_CSV.zip"
3
4 size_deciles <- read_size_deciles_vw(url_size)
5 factors      <- read_factors(url_factors)
6
7 common_dates <- sort(intersect(size_deciles$date, factors$date))
8 stopifnot(length(common_dates) >= 1000)
9 last_1k_dates <- tail(common_dates, 1000)
10 if (is.numeric(last_1k_dates)) last_1k_dates <- as.Date(last_1k_dates, origin = "1970-01-01")
11
12 size_last <- filter(size_deciles, date %in% last_1k_dates)
13 fac_last   <- filter(factors,       date %in% last_1k_dates)
14
15 capm_data <- inner_join(size_last, fac_last, by = "date") |>
16   pivot_longer(cols = Lo10:Hi10, names_to = "portfolio", values_to = "ret_pct") |>
17   mutate(ret = ret_pct, rf = RF, mkt_excess = `Mkt-RF`, excess_ret = ret - rf) |>
18   select(date, portfolio, excess_ret, mkt_excess, rf)
19
20 nrow(capm_data)      # debería ser 10000 (1000 meses x 10 carteras)
```

```
## [1] 10000
```

```
1 range(capm_data$date) # rango de fechas usado
```

```
## [1] "1942-09-01" "2025-12-01"
```

```
1 tail(capm_data)
```

```
## # A tibble: 6 x 5
##   date     portfolio excess_ret mkt_excess     rf
##   <date>    <chr>        <dbl>        <dbl> <dbl>
## 1 2025-12-01 Dec5        -0.53        -0.36  0.34
## 2 2025-12-01 Dec6        -1.21        -0.36  0.34
## 3 2025-12-01 Dec7        -0.68        -0.36  0.34
## 4 2025-12-01 Dec8         0.99        -0.36  0.34
## 5 2025-12-01 Dec9        -0.92        -0.36  0.34
## 6 2025-12-01 Hi10        -0.34        -0.36  0.34
```

5 CAPM serie de tiempo (sin y con HAC).

- Estimamos $r_{p,t} = \alpha_p + \beta_p r_{M,t} + \varepsilon_{p,t}$ para 10 carteras, donde $r_{p,t}$ y $r_{M,t}$ son retornos en exceso y $r_{M,t}$ corresponde a $Mkt - RF$.
- Objetivo: obtener $\hat{\beta}_p$ y $\hat{\alpha}_p$. Comparamos errores estándar OLS con HAC Heteroskedasticity and Autocorrelation Consistent (Newey-West) para evitar inferencia optimista cuando existe autocorrelación y heterocedasticidad.

6 Estimación MCO.

```
1 order_port <-
  ↪ c("Lo10", "Dec2", "Dec3", "Dec4", "Dec5", "Dec6", "Dec7", "Dec8", "Dec9", "Hi10", "Mkt-RF")
2
3 # OLS base
4 capm_ts <- capm_data |>
  filter(!is.na(excess_ret), !is.na(mkt_excess)) |>
  group_by(portfolio) |>
  do(tidy(lm(excess_ret ~ mkt_excess, data = .))) |>
  ungroup() |>
  mutate(term = recode(term, `Intercept` = "alpha", mkt_excess = "beta")) |>
10 select(portfolio, term, estimate, p.value) |>
11 pivot_wider(names_from = term, values_from = c(estimate, p.value)) |>
12 mutate(portfolio = factor(portfolio, levels = order_port)) |>
13 arrange(portfolio)
14
15 # OLS con SE Newey-West (puntuales iguales)
16 capm_ts_hac <- capm_data |>
  filter(!is.na(excess_ret), !is.na(mkt_excess)) |>
  group_by(portfolio) |>
  group_modify(~ {
    fit <- lm(excess_ret ~ mkt_excess, data = .x)
    nwvc <- NeweyWest(fit, lag = 3, prewhite = FALSE) # ajusta lag (3-6 típico)
    tidy(coeftest(fit, vcov. = nwvc))}) |>
  ungroup() |>
  mutate(term = recode(term, `Intercept` = "alpha", mkt_excess = "beta")) |>
25 select(portfolio, term, estimate, p.value) |>
26 pivot_wider(names_from = term, values_from = c(estimate, p.value)) |>
27 mutate(portfolio = factor(portfolio, levels = order_port)) |>
28 arrange(portfolio)
29
30 capm_ts      # sin corrección

## # A tibble: 10 x 5
##   portfolio estimate_alpha estimate_beta p.value_alpha p.value_beta
##   <fct>        <dbl>       <dbl>        <dbl>        <dbl>
## 1 Lo10         0.154       1.15        0.247      1.23e-192
## 2 Dec2         0.0424      1.21        0.694      3.09e-266
## 3 Dec3         0.0628      1.20        0.480        0
## 4 Dec4         0.0508      1.17        0.534        0
## 5 Dec5         0.0373      1.16        0.593        0
## 6 Dec6         0.0296      1.12        0.616        0
## 7 Dec7         0.0484      1.12        0.341        0
## 8 Dec8         0.0342      1.08        0.425        0
## 9 Dec9         0.0381      1.02        0.261        0
## 10 Hi10        0.00250     0.939       0.925        0
```

```
1 capm_ts_hac # con SE HAC

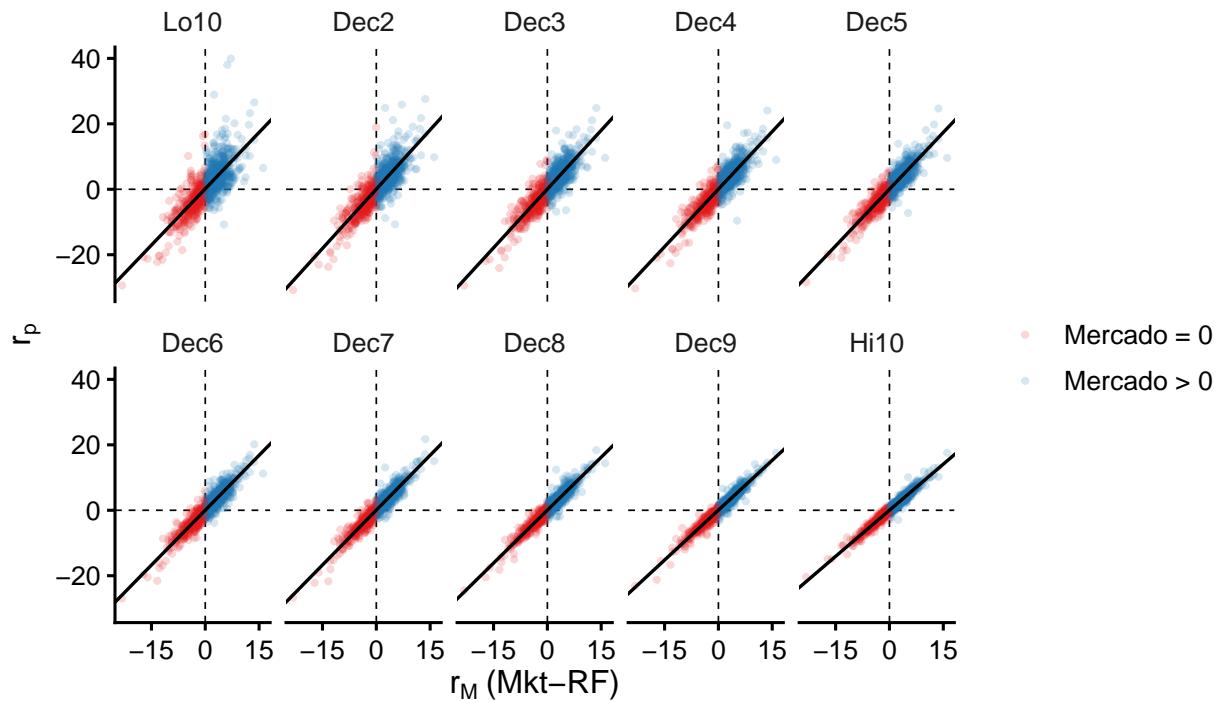
## # A tibble: 10 x 5
##   portfolio estimate_alpha estimate_beta p.value_alpha p.value_beta
##   <fct>          <dbl>        <dbl>       <dbl>        <dbl>
## 1 Lo10           0.154        1.15      0.266    1.06e-134
## 2 Dec2           0.0424       1.21      0.697    2.02e-213
## 3 Dec3           0.0628       1.20      0.486    2.92e-245
## 4 Dec4           0.0508       1.17      0.541    1.98e-252
## 5 Dec5           0.0373       1.16      0.586    3.62e-295
## 6 Dec6           0.0296       1.12      0.632     0
## 7 Dec7           0.0484       1.12      0.372     0
## 8 Dec8           0.0342       1.08      0.453     0
## 9 Dec9           0.0381       1.02      0.303     0
## 10 Hi10          0.00250     0.939     0.929     0
```

7 Visualización MCO.

```
1 order_port <- c("Lo10", "Dec2", "Dec3", "Dec4", "Dec5", "Dec6", "Dec7", "Dec8", "Dec9", "Hi10")
2
3 line_df <- capm_ts |>
4   transmute(portfolio, alpha = estimate_alpha, beta = estimate_beta) |>
5   mutate(portfolio = factor(portfolio, levels = order_port))
6
7 capm_plot_df <- capm_data |>
8   mutate(portfolio = factor(portfolio, levels = order_port))
9
10 ggplot(capm_plot_df, aes(x = mkt_excess, y = excess_ret)) +
11   geom_hline(yintercept = 0, linetype = "dashed", linewidth = 0.3) +
12   geom_vline(xintercept = 0, linetype = "dashed", linewidth = 0.3) +
13   geom_point(aes(color = mkt_excess > 0), alpha = 0.18, size = 0.7) +
14   geom_abline(data = line_df, aes(intercept = alpha, slope = beta),
15     show.legend = FALSE, linewidth = 0.6) +
16   facet_wrap(~ portfolio, ncol = 5) +
17   scale_color_manual(values = c("TRUE" = "#1f78b4", "FALSE" = "#e31a1c"),
18     labels = c("FALSE" = "Mercado 0", "TRUE" = "Mercado > 0"),
19     name = "") +
20   scale_x_continuous(breaks = c(-15, 0, 15)) +
21   scale_y_continuous(breaks = scales::pretty_breaks(n = 4)) +
22   labs(title = expression(atop("CAPM en exceso: serie de tiempo por cartera",
23     r[p,t] == alpha[p] + beta[p] %.% r[M,t])),
24     x = expression(r[M,t]~"(Mkt-RF)"),
25     y = expression(r[p,t])) +
26   theme_classic(base_size = 12) + # quita cuadrícula
27   theme(axis.text.x = element_text(angle = 0, vjust = 0.5),
28     strip.background = element_blank())
```

CAPM en exceso: serie de tiempo por cartera

$$r_p = \alpha_p + \beta_p \cdot r_M$$



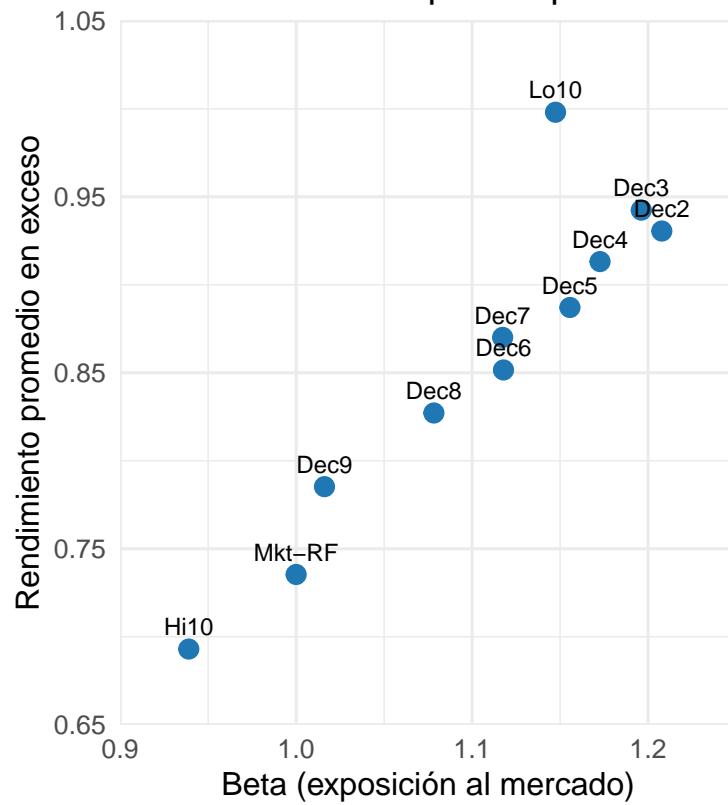
8 Resultados. CAPM serie de tiempo (sin y con HAC).

- Las $\hat{\beta}_p$ se ubican entre ≈ 0.94 y ≈ 1.21 y son altamente significativas. En general, las carteras de mayor tamaño tienden a mostrar menor exposición al mercado.
- Las $\hat{\alpha}_p$ son pequeñas en magnitud, del orden de décimas de punto porcentual mensual, y no son estadísticamente distintas de cero. HAC (Newey-West) cambia poco la inferencia, pero protege contra autocorrelación y heterocedasticidad.

9 Visualización de Betas y rendimientos en exceso.

```
1 order_port <-
2   ↳ c("Lo10", "Dec2", "Dec3", "Dec4", "Dec5", "Dec6", "Dec7", "Dec8", "Dec9", "Hi10", "Mkt-RF")
3
4 # Totales promedio por cartera
5 beta_ret <- capm_ts |>
6   mutate(portfolio = factor(portfolio, levels = order_port)) |>
7   select(portfolio, beta = estimate_beta) |>
8   inner_join(capm_data |>
9     group_by(portfolio) |>
10    summarise(mean_total = mean(excess_ret, na.rm = TRUE),
11              .groups = "drop"), by = "portfolio")
12
13 # Punto de mercado: beta=1, mkt_excess
14 mkt_point <- tibble(portfolio = factor("Mkt-RF", levels = order_port),
15                      beta = 1,
16                      mean_total = mean(capm_data$mkt_excess, na.rm = TRUE))
17
18 beta_ret <- bind_rows(beta_ret, mkt_point)
19
20 ggplot(beta_ret, aes(x = beta, y = mean_total, label = portfolio)) +
21   geom_point(size = 3, color = "#1f78b4") +
22   geom_text(vjust = -0.8, size = 3) +
23   labs(title = "CAPM en exceso: puntos por cartera",
24         x = "Beta (exposición al mercado)",
25         y = "Rendimiento promedio en exceso") +
26   theme_minimal(base_size = 12) +
27   scale_x_continuous(limits = c(0.9, 1.25), expand = c(0, 0)) +
28   scale_y_continuous(limits = c(0.65, 1.05), expand = c(0, 0)) +
29   coord_equal()
```

CAPM en exceso: puntos por cartera



10 Regresión en sección cruzada (sin y con Shanken).

- Antes estimamos $\hat{\beta}_p$ para cada cartera a partir de una regresión de serie de tiempo. Ahora usamos esas $\hat{\beta}_p$ como insumo para evaluar la predicción central del CAPM en la sección cruzada.
- Específicamente, relacionamos el retorno promedio en exceso de cada cartera con su exposición al mercado: $\bar{r}_p = \gamma_0 + \gamma_1 \hat{\beta}_p + u_p$, donde \bar{r}_p es el promedio temporal de $r_{p,t}$.
- Interpretación: γ_1 es el precio del riesgo de mercado, la compensación promedio por una unidad adicional de β , y γ_0 es el intercepto de la relación riesgo–retorno.
- Preguntas: ¿existe una relación positiva riesgo–retorno $\gamma_1 > 0$? y ¿es el intercepto compatible con cero en retornos en exceso $\gamma_0 = 0$?
- Reportamos OLS y un ajuste estilizado Shanken para reflejar que $\hat{\beta}_p$ es un regresor estimado, lo que tiende a inflar los errores estándar respecto al caso con β observada.

11 Estimación regresión en sección cruzada.

```
1 cross_df <- capm_ts |>
2   select(portfolio, beta = estimate_beta) |>
3   inner_join(capm_data |>
4     group_by(portfolio) |>
5     summarise(mean_excess = mean(excess_ret, na.rm = TRUE),
6               .groups = "drop"), by = "portfolio")
7
8 cs_reg <- lm(mean_excess ~ beta, data = cross_df)
9 cs_tidy <- tidy(cs_reg) # OLS sin corrección
10
11 # Shanken (corrige que beta es estimado)
12 sigma_f2 <- var(capm_data$mkt_excess, na.rm = TRUE)
13 mu_f      <- mean(capm_data$mkt_excess, na.rm = TRUE)
14 shanken_fac <- 1 + (mu_f^2 / sigma_f2)
15
16 vcov_shanken <- vcov(cs_reg) * shanken_fac
17 cs_shanken <- coeftest(cs_reg, vcov. = vcov_shanken) |>
18 tidy() # estimate, std.error, statistic, p.value
19
20 cs_tidy # sin corrección
```

```
## # A tibble: 2 x 5
##   term      estimate std.error statistic p.value
##   <chr>     <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept) -0.184     0.169    -1.09  0.307
## 2 beta         0.945     0.151     6.27  0.000241
```

```
1 cs_shanken # con SE Shanken
```

```
## # A tibble: 2 x 5
##   term      estimate std.error statistic p.value
##   <chr>     <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept) -0.184     0.171    -1.08  0.313
## 2 beta         0.945     0.153     6.18  0.000266
```

12 Resultados. Sección cruzada de promedios (sin y con Shanken).

- El precio del riesgo γ_1 es positivo y estadísticamente relevante en la muestra ($\hat{\gamma}_1 \approx 0.95\%$ mensual). El ajuste estilo Shanken incrementa levemente los errores estándar sin alterar el signo ni el mensaje económico.
- El intercepto γ_0 no difiere significativamente de cero, consistente con rendimientos en exceso correctamente centrados.
- Este resultado anticipa lo que veremos con Fama–MacBeth: en lugar de usar solo promedios, estimaremos el precio del riesgo mes a mes para evaluar su variación temporal y realizar inferencia con errores estándar adecuados.

13 Visualización de Betas, rendimientos en exceso y linea de regresión.

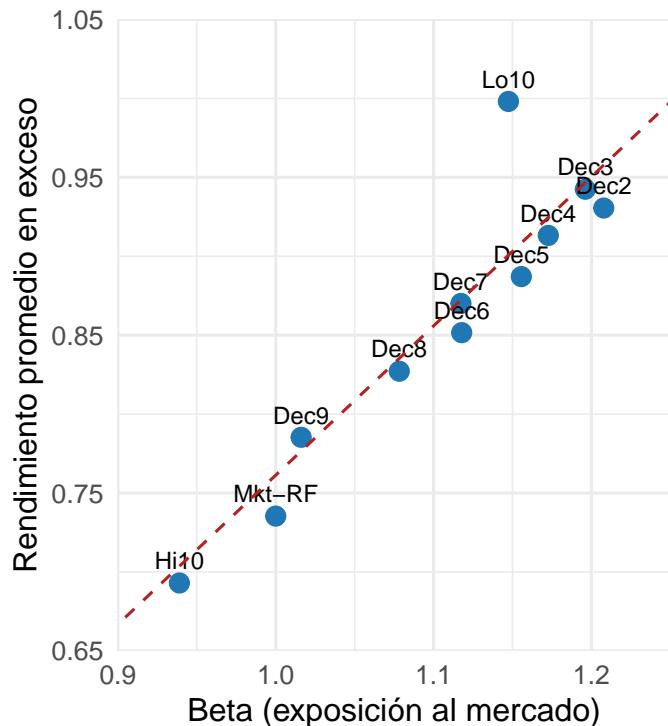
```

1 alpha_cs <- coef(cs_reg)[["(Intercept)"]]
2 slope_rm <- coef(cs_reg)[["beta"]]
3
4 ggplot(beta_ret, aes(x = beta, y = mean_total, label = portfolio)) +
5   geom_point(size = 3, color = "#1f78b4") +
6   geom_text(vjust = -0.8, size = 3) +
7   geom_abline(intercept = alpha_cs, slope = slope_rm,
8             linetype = "dashed", color = "firebrick") +
9   labs(title = expression(atop("CAPM en exceso: puntos por cartera y línea",
10                           E(r[p]) == alpha
11                           + beta %.% E(r[M]))),
12       x = "Beta (exposición al mercado)",
13       y = "Rendimiento promedio en exceso") +
14   theme_minimal(base_size = 12) +
15   scale_x_continuous(limits = c(0.9, 1.25), expand = c(0, 0)) +
16   scale_y_continuous(limits = c(0.65, 1.05), expand = c(0, 0)) +
17   coord_equal()

```

CAPM en exceso: puntos por cartera y línea

$$E(r_p) = \alpha + \beta \cdot E(r_M)$$



14 Fama–MacBeth (sin y con Newey–West).

- Fama–MacBeth: para cada mes estimamos $r_{p,t} = \lambda_{0,t} + \lambda_{1,t}\hat{\beta}_p + \eta_{p,t}$, usando $\hat{\beta}_p$ fijas la regresión de serie de tiempo.
- Interesa el precio del riesgo $\lambda_{1,t}$ y su promedio $\bar{\lambda}_1$. Para la inferencia sobre $\bar{\lambda}_1$, comparamos errores estándar básicos con HAC (Newey–West) sobre la serie temporal de $\hat{\lambda}_{1,t}$ (y análogamente para $\hat{\lambda}_{0,t}$).
- Valor agregado frente a la sección cruzada: en lugar de estimar un solo γ_1 usando medias, Fama–MacBeth estima $\lambda_{1,t}$ mes a mes. Esto permite estudiar la variación temporal del precio del riesgo y construir inferencia a partir de la dispersión temporal de $\hat{\lambda}_{1,t}$, con correcciones HAC cuando hay dependencia en el tiempo.
- Como el punto $Mkt - RF$ es solo una referencia visual y no entra en la regresión cruzada, la recta no está obligada a pasar exactamente por él.

15 Estimación Fama Mac-Beth 1/2.

```
1 # Betas fijos serie de tiempo
2 betas <- capm_ts |>
3   select(portfolio, beta = estimate_beta)
4
5 # Regresiones mensuales:  $r_{\{p,t\}} \sim \beta_{\text{portfolio}_p}$ 
6 fm_monthly <- capm_data |>
7   select(date, portfolio, ret = excess_ret) |>
8   left_join(betas, by = "portfolio") |>
9   group_by(date) |>
10  group_modify(~ tidy(lm(ret ~ beta, data = .x))) |>
11  ungroup() |>
12  mutate(term = recode(term, `Intercept` = "lambda0", beta = "lambda1"))
13
14 # Series 1_t y 0_t
15 lambda1_series <- fm_monthly |>
16   filter(term == "lambda1") |>
17   select(date, lambda1 = estimate, se = std.error, t = statistic, p = p.value)
18
19 lambda0_series <- fm_monthly |>
20   filter(term == "lambda0") |>
21   select(date, lambda0 = estimate, se = std.error, t = statistic, p = p.value)
22
23 # Resumen FM básico ( $sd/sqrt(T)$ )
24 lambda1_mean_basic <- lambda1_series |>
25   summarise(estimate = mean(lambda1, na.rm = TRUE),
26             std.error = sd(lambda1, na.rm = TRUE) / sqrt(n()),
27             statistic = estimate / std.error,
28             p.value = 2 * pt(-abs(statistic), df = n() - 1)) |>
29   mutate(method = "FM básico", factor = "lambda1", nw_lag = NA_integer_) |>
30   select(method, factor, estimate, std.error, statistic, p.value, nw_lag)
31
32 lambda0_mean_basic <- lambda0_series |>
33   summarise(estimate = mean(lambda0, na.rm = TRUE),
34             std.error = sd(lambda0, na.rm = TRUE) / sqrt(n()),
35             statistic = estimate / std.error,
36             p.value = 2 * pt(-abs(statistic), df = n() - 1)) |>
37   mutate(method = "FM básico", factor = "lambda0", nw_lag = NA_integer_) |>
38   select(method, factor, estimate, std.error, statistic, p.value, nw_lag)
```

16 Estimación Fama Mac-Beth 2/2.

```
1 # Resumen FM con Newey-West sobre las series
2 nw_L <- 3 # rezago HAC (3-6 típico en mensual)
3
4 lambda1_mean_nw <- {
5   fit <- lm(lambda1 ~ 1, data = lambda1_series)
6   vc <- NeweyWest(fit, lag = nw_L, prewhite = FALSE)
7   coeftest(fit, vcov. = vc) |>
8     tidy() |>
9     transmute(method = "FM + NW", factor = "lambda1",
10            estimate = estimate, std.error = std.error,
11            statistic = statistic, p.value = p.value,
12            nw_lag = nw_L)}
13
14 lambda0_mean_nw <- {
15   fit <- lm(lambda0 ~ 1, data = lambda0_series)
16   vc <- NeweyWest(fit, lag = nw_L, prewhite = FALSE)
17   coeftest(fit, vcov. = vc) |>
18     tidy() |>
19     transmute(method = "FM + NW", factor = "lambda0",
20            estimate = estimate, std.error = std.error,
21            statistic = statistic, p.value = p.value,
22            nw_lag = nw_L)}
23
24 # Comparación lado a lado
25 lambda_mean <- bind_rows(lambda1_mean_basic, lambda0_mean_basic,
26                           lambda1_mean_nw, lambda0_mean_nw)
27
28 lambda1_series # serie mensual de  $\lambda_t$ 
```

```
## # A tibble: 1,000 x 5
##   date      lambda1     se      t      p
##   <date>     <dbl>  <dbl>  <dbl>  <dbl>
## 1 1942-09-01    12.9   13.4   0.965  0.363
## 2 1942-10-01    22.7   5.22   4.36   0.00242
## 3 1942-11-01   -14.4   5.17  -2.79   0.0235
## 4 1942-12-01   -16.5   8.12  -2.03   0.0765
## 5 1943-01-01    70.7   35.8   1.97   0.0840
## 6 1943-02-01    50.7   36.2   1.40   0.198
## 7 1943-03-01    36.0   10.5   3.44   0.00878
## 8 1943-04-01    26.3   10.9   2.41   0.0426
## 9 1943-05-01    34.6   17.9   1.94   0.0883
## 10 1943-06-01   -8.74   7.74  -1.13   0.292
## # i 990 more rows
```

```

1 lambda0_series # serie mensual de O_t

## # A tibble: 1,000 x 5
##   date      lambda0     se      t      p
##   <date>     <dbl>  <dbl>  <dbl>  <dbl>
## 1 1942-09-01 -10.0  14.9 -0.673 0.520
## 2 1942-10-01 -15.6  5.83 -2.67  0.0284
## 3 1942-11-01  14.3  5.78  2.47  0.0385
## 4 1942-12-01  22.1  9.07  2.43  0.0409
## 5 1943-01-01 -63.2 40.1 -1.58  0.153
## 6 1943-02-01 -43.7 40.4 -1.08  0.311
## 7 1943-03-01 -29.5 11.7 -2.52  0.0356
## 8 1943-04-01 -25.2 12.2 -2.07  0.0727
## 9 1943-05-01 -29.1 20.0 -1.46  0.182
## 10 1943-06-01 10.3  8.65  1.20  0.266
## # i 990 more rows

```

```

1 lambda_mean # tabla comparativa básica vs NW para O y 1

## # A tibble: 4 x 7
##   method    factor estimate std.error statistic p.value nw_lag
##   <chr>     <chr>    <dbl>     <dbl>     <dbl>    <dbl>    <dbl>
## 1 FM básico lambda1  0.945     0.484     1.95   0.0512    NA
## 2 FM básico lambda0 -0.184     0.466    -0.395   0.693    NA
## 3 FM + NW   lambda1  0.945     0.521     1.81   0.0701    3
## 4 FM + NW   lambda0 -0.184     0.480    -0.383   0.702    3

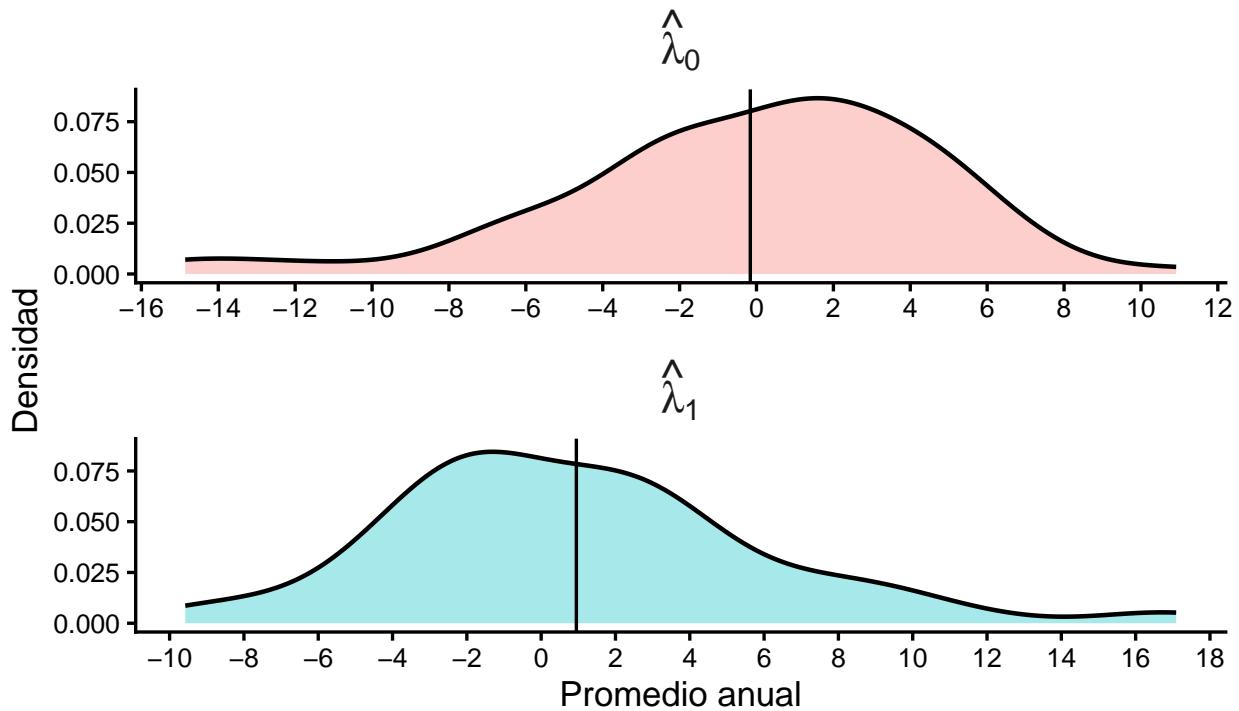
```

17 Visualización de las estimaciones Fama Mac-Beth.

```
1 lambda_year <- bind_rows(
2   lambda1_series |> transmute(date, value = lambda1, factor = "hat(lambda)[1]"),
3   lambda0_series |> transmute(date, value = lambda0, factor = "hat(lambda)[0]")
4 ) |>
5   mutate(year = year(date)) |>
6   group_by(factor, year) |>
7   summarise(value = mean(value, na.rm = TRUE), .groups = "drop")
8
9 # Medias por panel (para la linea vertical)
10 lambda_year_mean <- lambda_year |>
11   group_by(factor) |>
12   summarise(mu = mean(value, na.rm = TRUE), .groups = "drop")
13
14 ggplot(lambda_year, aes(x = value)) +
15   geom_density(aes(fill = factor), alpha = 0.35, adjust = 1.1, linewidth = 0.8) +
16   geom_vline(data = lambda_year_mean, aes(xintercept = mu),
17             linetype = "solid", linewidth = 0.6) +
18   facet_wrap(~ factor, scales = "free_x", ncol = 1, labeller = label_parsed) +
19   scale_x_continuous(breaks = scales::breaks_width(2)) +
20   labs(title = "Fama-MacBeth: densidad de promedios anuales",
21       subtitle = "Línea sólida: media por panel",
22       x = "Promedio anual", y = "Densidad") +
23   theme_classic(base_size = 13) +
24   theme(legend.position = "none",
25         strip.text = element_text(size = 16),
26         strip.background = element_blank())
```

Fama–MacBeth: densidad de promedios anuales

Línea sólida: media por panel



18 Resultados. Fama–MacBeth (sin y con Newey–West).

- El precio del riesgo de mercado es positivo en promedio: $\bar{\lambda}_1 \approx 0.95\%$ mensual, coherente con la sección cruzada. La evidencia es moderada, con $p \approx 0.05$ (SE básicos) y $p \approx 0.07$ (HAC Newey–West), reflejando la variación temporal de $\hat{\lambda}_{1,t}$.
- El intercepto promedio es $\bar{\lambda}_0 \approx -0.18\%$ y no es estadísticamente distinto de cero; la corrección Newey–West aumenta modestamente los errores estándar sin cambiar el mensaje central.

19 Conclusión.

- Los datos respaldan la lógica central del CAPM: el rendimiento esperado en exceso aumenta con β y el intercepto es compatible con cero.
- El precio del riesgo estimado es $\bar{\lambda}_1 \approx 0.95\%$ mensual. Como aproximación lineal, esto equivale a $\approx 11\%$ anualizado; la evidencia estadística es moderada, pero el signo y el orden de magnitud se mantienen bajo correcciones HAC.
- Las α_p no muestran desvíos sistemáticos. Una vez controlada la compensación por riesgo de mercado vía β , no queda un excedente sistemático sin explicar.
- El CAPM luce coherente pero no definitivo: en estas 10 carteras el ajuste es relativamente favorable, en parte porque trabajamos con carteras diversificadas y una muestra larga; aun así, modelos multifactor (tamaño, value, momentum) pueden capturar variación adicional y reducir alphas residuales.