

# Informe\_ARIMAS

Azahara Martinez, Iyán Álvarez, Maria de los Ángeles Diaz, Flor Pellegrini

2025-12-11

## Introducción

El objetivo del proyecto es construir y evaluar distintas estrategias de inversión dinámicas aplicadas a una cartera de cinco activos financieros. Para ello, seguimos dos pasos principales:

- Pronosticar los rendimientos con una función one-step-ahead (Sección 2).
- Optimizar la composición de la cartera en cada periodo de test utilizando tres criterios de utilidad diferentes (Secciones 3.1, 3.2 y 4).

Partimos del archivo `stock_returns_train_2.csv`, que transformamos a rendimientos en tanto por uno y dividimos en dos bloques: 96 periodos para entrenamiento y 24 para test. Todas las decisiones de evaluación y comparación de métodos se realizan sobre este periodo de test, con el fin de analizar el comportamiento fuera de muestra de las distintas estrategias.

## Predicción one-step-ahead

### Técnica utilizada y justificación

En la Sección~2 utilizamos modelos ARIMA univariantes para pronosticar los rendimientos de cada activo un paso adelante. La idea es la siguiente:

- Para cada activo y fecha del periodo de test, usamos únicamente la información disponible hasta ese momento.
- Ajustamos automáticamente un modelo ARIMA con la función `auto.arima` del paquete `forecast`, ya que permite seleccionar el orden del modelo de forma sistemática, evitando un ajuste manual activo por activo y es razonablemente flexible que captura tanto tendencia como autocorrelación.
- Actualizamos el modelo con los datos más recientes y obtenemos el pronóstico one-step-ahead con `forecast`

El flujo se implementa en una función `oneStepAhead`, que es llamada repetidamente por una función auxiliar `getPred_ts` (proporcionada en `eval_funcs.R`) para recorrer la serie completa. De este modo, obtenemos dos matrices para el periodo de test:

$$\hat{\mu}_t \quad (\text{rendimientos esperados}) \quad \text{y} \quad \hat{\sigma}_t \quad (\text{desviaciones estándar}), \quad t \in \{T + 1, \dots, T + r\}.$$

## Optimización media-varianza

En las Secciones~3.1 y 3.2 utilizamos el marco clásico de media-varianza para decidir los pesos de la cartera en cada fecha del periodo de test.

### Construcción de la matriz de covarianza

A partir de las desviaciones estándar pronosticadas  $\hat{\sigma}_t$  (haciendo uso de la función anterior) y de los rendimientos históricos, construimos una matriz de covarianzas dinámica, llamada `sigma_t`, que permite combinar información de volatilidades pronosticadas con correlaciones históricas relativamente estables. Además, esta proviene del producto matricial de una matriz diagonal llamada D, con las desviaciones estándar de cada activo y de la matriz de correlación R a partir de los rendimientos pasados. De esta forma, hallamos:

$$\Sigma_t = D_t R D_t$$

mediante la función `construcion_sigma_t`.

### Función de utilidad media-varianza

En cada fecha  $t$ , la cartera  $\alpha_t$  se elige maximizando una utilidad de tipo:

$$U(\alpha_t) = \alpha_t^\top \mu_t - \frac{\gamma}{2} \alpha_t^\top \Sigma_t \alpha_t,$$

En ambos casos se impone la restricción  $\sum_i \alpha_{i,t} = 1$ . La clave de este apartado reside en si se permite o no que algunos pesos sean negativos. A continuación veremos el proceso elegido para la selección de los gammas.

### Selección de $\gamma$ (criterio de Sharpe)

Para fijar el parámetro  $\gamma$  utilizamos un criterio operativo basado en desempeño fuera de muestra. Como  $\gamma$  regula el balance entre el retorno esperado y el riesgo en los modelos de utilidad, vamos a elegirlo según la eficiencia rentabilidad–volatilidad del portafolio generado. Para hacerlo práctico, probamos distintos valores de  $\gamma$  y, para cada uno, generamos las carteras dinámicas  $\alpha_t(\gamma)$  en el periodo de test y obtenemos sus retornos que tendrán la siguiente forma:

```
\[
  r_{\{p,t\}}(\gamma) \;=\; \sum_i \alpha_{\{i,t\}}(\gamma), \; x_{\{i,t\}}.
\]
```

A continuación evaluamos un indicador de desempeño como el Sharpe Ratio:

$$\text{Sharpe}(\gamma) = \frac{\overline{r_p(\gamma)}}{\text{sd}(r_p(\gamma))},$$

donde  $\overline{r_p(\gamma)}$  es la media de los retornos del portafolio y  $\text{sd}(\cdot)$  su desviación estándar.

Elegimos como  $\gamma$  óptimo aquel que maximiza el Sharpe dentro de un rango razonable, de modo que se favorezcan portafolios eficientes en términos de rentabilidad–riesgo y se eviten soluciones excesivamente apalancadas.

## Media-varianza con posiciones cortas (Sección 3.1)

Para este caso, utilizamos una versión clásica del modelo de media-varianza en la que la única restricción impuesta es que la suma de los pesos sea igual a uno, es decir,  $\sum_{i=1}^N \alpha_i = 1$  permitiendo que algunos pesos puedan tomar valores negativos. Bajo estas condiciones, el problema de optimización en cada fecha  $t$  consiste en maximizar la utilidad cuadrática

$$U(\alpha_t) = \alpha_t^\top \mu_t - \frac{\gamma}{2} \alpha_t^\top \Sigma_t \alpha_t,$$

donde  $\mu_t$  es el vector de rendimientos esperados y  $\Sigma_t$  la matriz de covarianza dinámica construida previamente.

Dado que existe una única restricción lineal, se puede resolver mediante el método de Lagrange. Introduciendo el multiplicador  $\lambda^*$  asociado a la restricción, las condiciones de primer orden (gradiente de Lagrange igual a cero) se escriben como

$$\Sigma \alpha - \frac{1}{\gamma} \mu + \frac{\lambda^*}{\gamma} \mathbf{1} = 0.$$

Resolviendo para  $\alpha$  se obtiene:

$$\alpha = \frac{1}{\gamma} \Sigma^{-1} \mu - \frac{\lambda^*}{\gamma} \Sigma^{-1} \mathbf{1}, \quad \lambda^* = \frac{\mathbf{1}^\top \Sigma^{-1} \mu - \gamma}{\mathbf{1}^\top \Sigma^{-1} \mathbf{1}}.$$

La función `posiciones_cortas` implementa exactamente esta solución, calculando primero el valor de  $\lambda^*$  y luego los pesos  $\alpha$ . Esta formulación permite obtener de manera eficiente los pesos óptimos sin recurrir a métodos iterativos, y hace explícita la dependencia de los pesos respecto a los retornos esperados, la aversión al riesgo y la estructura de covarianza. Al permitir posiciones negativas, la función puede generar apalancamiento y mayor volatilidad, por lo que la selección de  $\gamma$  es clave.

## Media-varianza sin posiciones cortas (Sección 3.2)

En este caso imponemos que todos los pesos sean no negativos,  $\alpha_i \geq 0$ , además de la condición de suma uno. Esto elimina el apalancamiento asociado a posiciones cortas y produce carteras más fáciles de interpretar para un inversor típico.

El problema de optimización de media-varianza con estas restricciones se puede plantear como un problema de programación cuadrática. De esta forma, se propone implementar la función `solve.QP` del paquete `quadprog`. La función `sin_posiciones_cortas` construye la matriz de penalización cuadrática como  $D = \gamma \cdot \Sigma$ , el vector lineal como  $d = \mu$ , y define las restricciones que garantizan la suma de pesos igual a uno y la no negatividad de cada  $\alpha_i$ . Luego, ajusta la solución eliminando valores negativos muy pequeños por errores numéricos y redondeando los resultados para mayor claridad. Cabe destacar que el valor de  $\gamma$  también es calculado con Sharpe Ratio.

## Utilidad logarítmica

En esta sección reemplazamos la función de utilidad de media-varianza por una utilidad logarítmica aproximada, que tiene en cuenta tanto el crecimiento de la riqueza como la penalización del riesgo y la restricción de que la suma de los pesos tiene que sumar 1. Para garantizar esta condición, los pesos se parametrizan mediante una transformación softmax que sigue la siguiente expresión:

$$\alpha_i = \frac{\exp(z_i)}{\sum_{j=1}^N \exp(z_j)}$$

La optimización se realiza utilizando la función optim de R (método BFGS, visto en clase), aplicada a la función objetivo negativa de U. La implementación concreta se encuentra en la función `getAlphaLog`. De forma análoga a los casos de media-varianza, el parámetro de aversión al riesgo  $\gamma_{LOG}$  se elige mediante un análisis de rejilla, buscando maximizar el Sharpe Ratio en el periodo de prueba.

## División del trabajo

Aunque el equipo está formado por cuatro integrantes, no dividimos el trabajo en partes aisladas. En su lugar, optamos por trabajar de manera completamente colaborativa, reuniéndonos en varias sesiones presenciales para avanzar todas las secciones del proyecto juntos. Esta metodología nos permitió combinar perfiles y puntos fuertes diferentes, contrastar decisiones de modelización en tiempo real, detectar errores más rápidamente y aprovechar distintas perspectivas en cada etapa. Consideramos que “cuatro cabezas piensan más que una”, por lo que preferimos avanzar de forma conjunta en lugar de repartir tareas individuales.