

PRÁCTICA

Módulo: Diseño de Algoritmos de Inversión y Backtesting Avanzado

1. Objetivo de la práctica

El estudiante deberá **diseñar, programar y realizar un backtesting completo** de un algoritmo de inversión. Se busca validar la eficacia de una estrategia de Momentum en condiciones operativas reales, evaluando su capacidad para generar Alpha tras considerar los costes transaccionales y comparar su robustez con la de modelos aleatorios.

2. Especificaciones de la Simulación

- **Capital Inicial:** 250.000 \$.
- **Periodo de Backtest:** Desde el 1 de enero de 2015 hasta la actualidad.
- **Universo de Inversión:**
 - Activos que formen parte del S&P durante los últimos 13 meses.

Histórico de precios:

https://drive.google.com/file/d/1nvubXdAu0EONlrP_yrURZbnPhBQ-uDaB/view?usp=sharing

- **Benchmark:** SPY (S&P 500 ETF Trust).

3. El Algoritmo: Lógica de Inversión

Para cada mes del periodo de backtesting, el alumno debe programar un algoritmo que replica la esencia de la metodología del ETF MSCI Momentum, pero utilizando un universo de inversión diferente y un rebalanceo mensual que realizaremos el último día hábil bursátil de cada mes.

- **Paso A: Retorno Acumulado con "Lag" de 1 Mes**

Se deben calcular dos variables de retorno para cada uno de los 10 activos (9 sectores + GLD)-utilizaremos retornos logarítmicos:-

1. **Momentum 12 Meses (R_12):** Rentabilidad desde el mes t-13 al mes t-1.
2. **Momentum 6 Meses (R_6):** Rentabilidad desde el mes t-7 al mes t-1.

(Nota: El mes t-1 es el mes anterior al rebalanceo; se excluye el mes actual para evitar el ruido de la reversión a la media).

- **Paso B: Normalización por Factor (Z-Score)**

Debido a que las volatilidades de los períodos de 6 y 12 meses son distintas, no se pueden sumar directamente. El alumno debe estandarizar los retornos **dentro del**

universo de ese mes:

$$z = \frac{(r_i - \mu_r)}{\sigma_r}$$

Donde μ es la media de retornos de los activos y σ su desviación típica en ese periodo concreto. Se obtendrá un Z_{-6} y un Z_{-12} para cada activo.

- **Paso C: Puntuación Compuesta y Selección**

1. **Score Final:** Se calcula como la media simple de los 2 Z-scores calculados
2. **Selección:** Se ordenan los activos de mayor a menor score y se seleccionan los **20 primeros**.
3. **Pesos:** El capital disponible se reparte al **5 %** en cada uno de los 20 activos.

4. Reglas de Ejecución y Costes

La simulación de las órdenes deberá seguir este protocolo estricto:

- **Venta:** Todas las posiciones que salen de la cartera se cierran al precio **OPEN** del día de rebalanceo.
- **Compra:** Las nuevas posiciones se adquieren al precio **CLOSE** del mismo día.
- **Costes de Transacción:** Se aplicará una comisión del **0,23%** sobre el valor efectivo de cada operación, con un **mínimo de 23\$** por orden.
- Si tenemos algún activo en cartera y deja de cotizar, el día que salga de mercado o del índice lo vendemos a precio close, el dinero obtenido se queda en liquidez hasta el siguiente rebalanceo.

5. Análisis de Resultados y Robustez

El backtesting debe incluir las siguientes métricas y pruebas de estrés:

- **Cuadro de Métricas:** Generar una tabla comparativa de CAGR, Volatilidad, Ratio Sharpe, Ratio Sortino, Máximo Drawdown, Beta y Alpha frente al SPY.
- **Visualización:**
 - Evolución de la rentabilidad acumulada (%) del algoritmo frente al SPY.
 - Histograma de retornos mensuales de nuestro algoritmo vs benchmark.
 - **Análisis de recurrencia de retornos.** Gráfico de dispersión (Scatter Plot) de rentabilidades anuales y trimestrales y del algoritmo vs. SPY.
- **Test de Monte Carlo (>25.000.000 de Monos):** Comparar el algoritmo con al menos 25 millones de agentes aleatorios (con un coste de rebalanceo del 0.23% x2 -compra/venta supondremos que vendemos y compramos el 100% de los activos y no tendremos en cuenta el mínimo de 23\$ por orden-). Demostrar en el notebook que el tiempo de ejecución es inferior a 24 horas.

6. Entregable

Se entregará un único fichero zip que incluirá 5 Notebooks:

- Notebook 1: Carga de Datos
- Notebook 2: EDA (Exploratory Data Analysis) y preparación de los datos
- Notebook 3: Implementación de la estrategia (utilizaremos retornos logarítmicos para calcular las señales): debe generarse un fichero .CSV con los 20 activos seleccionados en cada fecha de rebalanceo.
- Notebook 4: Ejecución y costes
- Notebook 5: Comparativa con benchmarks y métricas; resumen de resultados y un apartado final de análisis crítico en el que se discuta el impacto de la comisión mínima y la robustez del sistema: ¿Cómo nos está afectando el sesgo de supervivencia? ¿Cómo hemos garantizado que no tengamos un problema de look-ahead? ¿Crees que existe un problema de overfitting? ¿Hemos realizado un rebalanceo irrealista? ¿Cuánto dinero hemos pagado en comisiones de compraventa?

Todos los ficheros se comprimirán en un único fichero zip.

Nota importante: Los Notebooks deberán entregarse con todas las celdas ejecutadas. Las celdas que no estén ejecutadas no se corregirán y se puntuarán con un 0 en ese apartado.

7. Restricciones Técnicas (Software)

Para garantizar que el estudiante desarrolla su propia lógica de cálculo y no depende de librerías especializadas:

- **Librerías Permitidas:** Solo se pueden utilizar [Numpy](#), [pandas](#), [yfinance](#), [Matplotlib](#) y [Seaborn](#), [Scipy](#), [pyarrow](#).
- **Penalización:** La utilización de cualquier otra librería significará que la **puntuación máxima de la práctica será de un 5,0**.

8. Criterios de Evaluación

La calificación se basará en el rigor técnico y la capacidad de análisis (0-10 puntos):

- **Implementación Técnica (4 puntos):** Correcta aplicación de la lógica de momentum de MSCI, cálculo de z-scores y gestión del rebalanceo mensual.
- **Motor de Backtesting y Costes (2 puntos):** Rigor en la ejecución de Open/Close y aplicación exacta de la estructura de costes (0,23 % con un mínimo de 23 \$).
- **Análisis de Robustez (2 puntos):** Ejecución correcta del test de Monte Carlo y del filtro de confianza, con una interpretación coherente de los resultados, así como un tiempo óptimo para el cálculo de 25.000.000 de monos aleatorios (en el Notebook deberá imprimirse el tiempo de cálculo en la celda de ejecución de los monos).
- **Calidad de Visualización y Reflexión (2 puntos):** Claridad de los gráficos presentados y profundidad en el análisis de las conclusiones finales.

Bonificaciones Especiales (Extra) siempre manteniendo que la máxima nota será un 10:

- **+1 Punto:** Por la arquitectura del software. Uso correcto de **clases y funciones** que faciliten la modularidad y la reutilización del código.
- **+1 Punto:** Por la calidad del código. Cumplimiento de la normativa **PEP 8**, limpieza visual y **comentarios detallados** que expliquen la lógica de cada bloque.

Penalización:

Solo se pueden utilizar **Numpy, pandas, yfinance, Matplotlib y Seaborn, Scipy, pyarrow**. significará que la **puntuación máxima de la práctica será de 5,0**.