



MÁSTER EN INTELIGENCIA ARTIFICIAL. ONLINE

Sistema Inteligente de Optimización Dinámica de Aportaciones (DCA) sobre el NASDAQ100 mediante Modelos Compuestos Macro- Técnicos basados en Aprendizaje supervisado

TFM elaborado por: Diego Bascuas Gutiérrez

Tutor/a de TFM: Juan Manuel Moreno Lamparero

Oviedo, a 7 de julio del 2026

Resumen.....	7
Abstract.....	8
Capítulo 1. Introducción y antecedentes.....	9
1.1. Contexto de la inversión periódica y Dollar Cost Averaging.....	9
1.2. Problemática del DCA tradicional ante ciclos de mercado y volatilidad.....	10
1.3. Relevancia del NASDAQ100 como activo de referencia.....	11
1.4. Inteligencia Artificial aplicada a decisiones de inversión.....	12
1.5. Planteamiento general del sistema macro-técnico propuesto.....	13
Capítulo 2. Objetivos del proyecto.....	14
2.1. Objetivo general.....	14
2.2. Objetivos específicos.....	14
2.3. Criterios de éxito y alcance del proyecto.....	15
Capítulo 3. Marco teórico.....	18
3.1. Fundamentos de inversión cuantitativa y asset allocation.....	18
3.2. Estrategias DCA y limitaciones de la aportación constante.....	19
3.3. Series temporales financieras y aprendizaje supervisado.....	20
3.4. Detección de regímenes de mercado y filtros macroeconómicos.....	21
3.5. Indicadores técnicos y momentum aplicado a índices bursátiles.....	21
3.6. Validación temporal y riesgo de data leakage.....	22
3.7. Algoritmos utilizados.....	23
3.7.1. Regresión logística.....	23
3.7.2. Random Forest.....	24
3.7.3. Extra Trees.....	24
3.7.4. Ridge Regression.....	24
3.8. Métricas de evaluación.....	25
3.8.1. Métricas de clasificación.....	25
3.8.2. Métricas de regresión.....	25
3.8.3. Métricas financieras y de backtesting.....	26
Capítulo 4. Material y metodología.....	28

4.1. Herramientas, entorno de trabajo y librerías utilizadas.....	28
4.1.1. Entorno de ejecución y reproducibilidad.....	29
4.2. Fuentes de datos y periodo de análisis.....	31
4.2.1. Datos financieros del NASDAQ100 y referencias de mercado.....	33
4.2.2. Datos macroeconómicos y financieros externos.....	33
4.3. Justificación de la frecuencia mensual.....	34
4.4. Arquitectura general del sistema.....	35
4.4.1. Analista macroeconómico.....	35
4.4.2. Analista técnico.....	36
4.4.3. Evaluación financiera mediante backtesting.....	36
4.5. Pipeline de trabajo y artefactos generados.....	37
4.6. Estrategia de validación temporal.....	38
4.6.1. División temporal entre entrenamiento y test.....	38
4.6.2. Walk-forward validation.....	39
4.7. Consideraciones metodológicas sobre leakage, sobreajuste y tamaño muestral.....	39
Capítulo 5. Desarrollo y resultados del analista macroeconómico.....	41
5.1. Función del analista macroeconómico dentro del sistema.....	41
5.2. Construcción del dataset macro-financiero.....	42
5.2.1. Familias de variables utilizadas.....	43
5.2.2. Unificación temporal, limpieza y selección de variables.....	45
5.3. Evolución metodológica del bloque macro.....	45
5.3.1. Formulación inicial multiclase.....	46
5.3.2. Limitaciones detectadas.....	47
5.3.3. Reformulación binaria como filtro de entorno favorable.....	48
5.4. Entrenamiento, validación y selección del modelo macro final.....	49
5.5. Resultados del analista macroeconómico.....	51
5.6. Interpretación y limitaciones del bloque macro.....	53
Capítulo 6. Desarrollo y resultados del analista técnico.....	55
6.1. Función del analista técnico dentro de la estrategia DCA.....	55

6.2. Ingeniería de características técnicas del NASDAQ100.....	56
6.2.1. Momentum, tendencia y volatilidad.....	57
6.2.2. Drawdown, posición relativa y calidad retorno-riesgo.....	57
6.3. Integración del contexto macroeconómico.....	58
6.4. Construcción del target técnico de agresividad DCA.....	58
6.4.1. Retorno futuro, drawdown futuro y score técnico.....	59
6.4.2. Discretización en multiplicadores 0.5x, 1x y 2x.....	59
6.5. Experimentos de modelado técnico.....	60
6.5.1. Clasificación multiclase como baseline.....	60
6.5.2. Reformulación binaria agresiva.....	62
6.5.3. Regresión del score técnico.....	63
6.6. Resultados comparativos del analista técnico.....	64
6.7. Interpretación y limitaciones del bloque técnico.....	65
Capítulo 7. Evaluación financiera y resultados del sistema.....	67
7.1. Objetivo del backtesting y lógica de simulación DCA.....	67
7.2. Conversión de predicciones en multiplicadores de aportación.....	68
7.3. Estrategias comparadas.....	69
7.3.1. DCA base.....	70
7.3.2. Estrategias derivadas de los modelos técnicos.....	70
7.3.3. Benchmark Buy & Hold.....	71
7.4. Métricas financieras utilizadas.....	71
7.5. Resultados en tramo limpio fuera de muestra.....	73
7.6. Resultados en evaluación walk-forward histórica.....	74
7.7. Comparación frente a DCA base y Buy & Hold.....	77
7.8. Discusión de resultados financieros y eficiencia del capital extra.....	77
7.9. Limitaciones del backtest.....	78
Capítulo 8. Conclusiones y líneas futuras.....	80
8.1. Conclusiones principales.....	80
8.2. Cumplimiento del objetivo general.....	81

8.3. Cumplimiento de los objetivos específicos.....	82
8.4. Aportaciones del proyecto.....	83
8.5. Limitaciones generales del sistema propuesto.....	84
8.6. Líneas futuras.....	85
8.6.1. Ampliación a frecuencia diaria o mayor granularidad temporal.....	86
8.6.2. Validación sobre otros índices o activos financieros.....	86
8.6.3. Incorporación de nuevas fuentes de información.....	87
8.6.4. Optimización de la política de asignación de capital.....	87
Referencias bibliográficas.....	89
Anexo A. Estructura del proyecto y artefactos entregados.....	92

Resumen

Este Trabajo Fin de Máster desarrolla y evalúa un sistema de apoyo a la decisión para modular aportaciones periódicas mediante Dollar Cost Averaging sobre el NASDAQ100. El sistema combina un analista macroeconómico, un analista técnico y un módulo de evaluación financiera mediante backtesting. El bloque macroeconómico transforma variables económicas, financieras y de mercado en una señal de contexto, mientras que el bloque técnico utiliza dicha señal junto con indicadores del NASDAQ100 para estimar niveles de agresividad de aportación: reducida, base o agresiva.

La metodología se basa en aprendizaje supervisado, construcción de targets a partir de retorno y drawdown futuros, validación temporal y evaluación walk-forward. Los resultados muestran que el sistema permite construir un pipeline trazable y evaluable, aunque las métricas predictivas son moderadas y las estrategias dinámicas no demuestran una superioridad financiera clara y estable frente a un DCA base. La principal aportación del proyecto es metodológica: conectar datos macroeconómicos, señales técnicas, modelos supervisados y backtesting financiero dentro de un marco reproducible para estudiar decisiones de aportación periódica bajo incertidumbre.

Palabras clave: Dollar Cost Averaging, NASDAQ100, aprendizaje supervisado, backtesting, validación temporal, inversión cuantitativa.

Abstract

This Master's Thesis develops and evaluates a decision-support system to dynamically modulate periodic Dollar Cost Averaging contributions on the NASDAQ100. The system combines a macroeconomic analyst, a technical analyst and a financial backtesting module. The macroeconomic block transforms economic, financial and market variables into a contextual signal, while the technical block combines this signal with NASDAQ100 indicators to estimate contribution aggressiveness levels: reduced, base or aggressive.

The methodology is based on supervised learning, target construction from future returns and drawdowns, temporal validation and walk-forward evaluation. The results show that the system provides a traceable and assessable pipeline, although predictive performance remains moderate and the dynamic strategies do not demonstrate a clear and stable financial superiority over a base DCA strategy. The main contribution of the project is methodological: connecting macroeconomic data, technical signals, supervised models and financial backtesting within a reproducible framework for studying periodic contribution decisions under uncertainty.

Keywords: Dollar Cost Averaging, NASDAQ100, supervised learning, backtesting, temporal validation, quantitative investment.

Capítulo 1. Introducción y antecedentes

La inversión periódica constituye una de las formas más simples y disciplinadas de exposición progresiva a los mercados financieros. En lugar de concentrar todo el capital en un único momento, el inversor realiza aportaciones regulares a lo largo del tiempo, normalmente con una frecuencia mensual, trimestral o anual. Dentro de este enfoque, el *Dollar Cost Averaging* (DCA) se basa en invertir una cantidad fija de dinero en cada periodo, independientemente del nivel exacto del activo en ese momento. Esta regla reduce la necesidad de acertar el instante óptimo de entrada y facilita mantener una pauta de inversión sistemática.

El presente Trabajo Fin de Máster parte de esa lógica de inversión periódica, pero no se limita a reproducir una estrategia DCA fija. El proyecto estudia si es posible construir un sistema de apoyo a la decisión que module la agresividad de las aportaciones mensuales sobre el NASDAQ100 a partir de información macroeconómica, señales técnicas y aprendizaje supervisado. La finalidad no es predecir con certeza la evolución futura del mercado ni diseñar una estrategia garantizada para batir a referencias pasivas, sino desarrollar, validar y evaluar un marco macro-técnico trazable que permita transformar información histórica en decisiones de aportación reducida, base o agresiva.

Esta introducción presenta el contexto del problema, la motivación financiera y tecnológica del proyecto, y los antecedentes que justifican el diseño posterior. El desarrollo teórico detallado de los modelos, métricas y procedimientos de validación se reserva para el Capítulo 3. En este primer capítulo se ofrece una visión general: qué aporta la inversión periódica, qué limitaciones presenta el DCA tradicional, por qué el NASDAQ100 constituye un activo de referencia relevante para el estudio, cómo debe entenderse el uso de inteligencia artificial en finanzas y cuál es el planteamiento general del sistema propuesto.

1.1. Contexto de la inversión periódica y Dollar Cost Averaging

El *Dollar Cost Averaging* es una estrategia de inversión periódica en la que se aporta una cantidad monetaria constante en intervalos regulares. Al mantenerse fija la cantidad aportada, el número de participaciones adquiridas varía con el precio del activo: cuando el precio es más bajo se compran más participaciones y cuando el precio es más alto se compran menos. Esta mecánica convierte el DCA en una regla sencilla, comprensible y fácil de ejecutar, especialmente para inversores que acumulan capital de forma gradual mediante ahorro periódico.

Desde un punto de vista operativo, el atractivo del DCA reside en su disciplina. La estrategia evita que cada aportación dependa de una decisión discrecional sobre si el mercado está caro o barato, reduce el peso psicológico del momento exacto de entrada y permite implementar una política de inversión regular. En contextos reales, esta simplicidad puede ser valiosa: muchos inversores no disponen de todo el capital desde el inicio, sino que lo generan progresivamente mediante renta laboral, ahorro mensual u otras fuentes recurrentes. En esos casos, la inversión periódica no es solo una elección estratégica, sino también una restricción práctica.

No obstante, la literatura financiera ha señalado que el DCA no debe interpretarse como una regla universalmente superior a invertir todo el capital disponible al inicio. Constantinides (1979) analiza la suboptimalidad del DCA bajo determinados supuestos teóricos, especialmente cuando el inversor ya dispone del capital y los activos con prima de riesgo positiva tienden a compensar una exposición temprana. Rozeff (1994) compara la inversión inicial única con la inversión promediada y muestra que la preferencia entre ambas depende de supuestos, horizonte, riesgo asumido y comportamiento del mercado. Estas aportaciones invitan a tratar el DCA con prudencia: puede ser una estrategia disciplinada y útil, pero no necesariamente la alternativa con mayor rentabilidad esperada en todos los escenarios.

Otros trabajos han matizado la discusión al incorporar dimensiones de riesgo, comportamiento del inversor y utilidad percibida. Leggio y Lien (2003) examinan la efectividad del DCA empleando medidas de riesgo a la baja, mientras que Dichtl y Drobetz (2011) relacionan la popularidad de la estrategia con argumentos de finanzas conductuales y teoría prospectiva. Esta perspectiva es relevante para el proyecto porque el valor de una estrategia periódica no se agota en comparar rentabilidades medias. También intervienen la tolerancia al riesgo, la estabilidad de las decisiones, la capacidad de mantener una regla en periodos adversos y la forma en que se distribuye temporalmente el capital invertido.

En este TFM, el DCA se adopta como punto de partida operativo, no como una afirmación de superioridad financiera incondicional. La estrategia base representa una referencia simple: aportar una cantidad mensual constante al NASDAQ100. Sobre esa referencia se plantea una pregunta adicional: si se dispone de información macroeconómica y técnica, ¿puede construirse un sistema que ayude a decidir si en un mes concreto conviene aportar menos, mantener la aportación base o aportar más? Esta pregunta transforma el DCA fijo en un problema de modulación de aportaciones, manteniendo la lógica periódica pero introduciendo una capa de decisión dinámica.

1.2. Problemática del DCA tradicional ante ciclos de mercado y volatilidad

El DCA tradicional presenta una ventaja clara: reduce la concentración del capital en un único precio o fecha de entrada. Al distribuir las compras en el tiempo, el inversor reparte el riesgo de entrada, aunque sigue expuesto a la evolución posterior del mercado. Sin embargo, esta misma simplicidad implica una limitación: la estrategia no adapta por sí misma la agresividad de las aportaciones al contexto de mercado. Aporta la misma cantidad en periodos de fuerte crecimiento, en fases de elevada volatilidad, en correcciones profundas o en entornos macroeconómicos favorables o desfavorables.

Esta rigidez puede ser razonable para una estrategia pasiva y disciplinada, pero abre una cuestión de investigación: si los mercados atraviesan ciclos y el riesgo no permanece constante, una regla fija puede no aprovechar de forma diferenciada los distintos estados del mercado. En periodos de caída o estrés, una aportación constante puede resultar psicológicamente difícil de mantener, aunque permita comprar a precios más bajos. En periodos de fuerte tendencia alcista, la misma regla puede quedarse corta frente a estrategias con mayor exposición temprana. En periodos de valoración elevada o volatilidad intensa, aumentar aportaciones sin atender al riesgo puede no ser adecuado para todos los perfiles de inversor.

La literatura sobre DCA frente a inversión inicial única muestra precisamente que la comparación depende del recorrido de mercado, de la disponibilidad de capital, del riesgo

asumido y de las preferencias del inversor (Constantinides, 1979; Rozeff, 1994). Por ello, este trabajo evita presentar el DCA dinámico como una solución universal. La motivación no consiste en demostrar de antemano que variar aportaciones sea siempre mejor, sino en construir un marco que permita estudiar esa variación de forma ordenada y evaluable.

La problemática se vuelve especialmente relevante cuando la estrategia se aplica sobre un activo con elevada sensibilidad al ciclo financiero. Un índice de crecimiento puede ofrecer periodos prolongados de rentabilidad elevada, pero también episodios de volatilidad y drawdown significativos. En ese contexto, una aportación fija puede ocultar dos preguntas distintas: qué señal se observa en el mercado y cuánto capital conviene asignar en ese momento dentro de una política periódica. Separar ambas preguntas es importante. Una señal predictiva débil no debería convertirse automáticamente en una decisión agresiva, y una mejora en métricas de clasificación no implica necesariamente una mejora financiera.

El proyecto responde a esta problemática mediante una formulación prudente. En lugar de decidir si se compra o no se compra, el sistema conserva la lógica DCA y solo modula la cantidad aportada. Las decisiones se expresan mediante tres multiplicadores: aportación reducida, aportación base y aportación agresiva. De este modo, el sistema no sustituye la disciplina periódica por una estrategia de entrada y salida completa, sino que estudia si la intensidad de la aportación puede ajustarse con apoyo de información macroeconómica y técnica.

1.3. Relevancia del NASDAQ100 como activo de referencia

El activo de referencia del proyecto es el NASDAQ100, denominado *Nasdaq-100 Index* en la documentación oficial de Nasdaq. Según su metodología oficial, el índice está formado por valores no financieros listados en Nasdaq y utiliza una ponderación modificada por capitalización de mercado (Nasdaq, Inc., 2026). Esta definición resulta relevante porque delimita el universo de compañías sobre el que se evalúa la estrategia: no se trata de una cartera arbitraria, sino de un índice ampliamente identificado con grandes empresas cotizadas en Nasdaq y con una presencia destacada de sectores de crecimiento.

La documentación descriptiva de Nasdaq presenta el Nasdaq-100 como un índice compuesto por 100 de las mayores compañías nacionales e internacionales no financieras listadas en Nasdaq, utilizado habitualmente como indicador de exposición a empresas no financieras relevantes del mercado Nasdaq (Nasdaq, Inc., n.d.). Aunque en el lenguaje común se asocia con frecuencia al sector tecnológico, conviene evitar una simplificación excesiva. El índice no representa exclusivamente a empresas tecnológicas ni constituye una medida completa de todo el mercado estadounidense. Su composición depende de reglas concretas de elegibilidad, revisión y ponderación, y puede cambiar con el tiempo.

La elección del NASDAQ100 tiene sentido dentro de este TFM por varias razones. En primer lugar, proporciona una serie financiera reconocible y suficientemente relevante para estudiar una estrategia de inversión periódica. En segundo lugar, su perfil de crecimiento lo convierte en un activo interesante para analizar la tensión entre rentabilidad potencial y volatilidad. En tercer lugar, su comportamiento histórico permite observar periodos alcistas, correcciones y cambios de régimen que son útiles para evaluar si las señales macro-técnicas pueden tener alguna relación con la conveniencia de aportar más o menos capital.

Al mismo tiempo, utilizar el NASDAQ100 impone límites claros. Los resultados obtenidos sobre este índice no pueden generalizarse automáticamente a otros índices, ETFs, sectores o carteras multi-activo. Un sistema que se comporta de una forma determinada sobre el

NASDAQ100 puede responder de manera diferente en activos con menor volatilidad, distinta composición sectorial o sensibilidad macroeconómica alternativa. Por ello, el proyecto se plantea como una evaluación histórica sobre un activo concreto, no como una demostración general de que la modulación DCA sea válida para cualquier mercado.

La relación entre DCA y NASDAQ100 también justifica la comparación con referencias simples. Una estrategia DCA base permite medir qué ocurre si se invierte de forma periódica sin usar modelos predictivos. Buy & Hold se incorpora como benchmark pasivo de mercado, aunque su mecánica distinta exige una interpretación separada en los capítulos posteriores.

1.4. Inteligencia Artificial aplicada a decisiones de inversión

La aplicación de inteligencia artificial y aprendizaje automático a finanzas ha crecido de forma significativa por la disponibilidad de datos, herramientas computacionales y librerías de modelado. En este ámbito, los modelos supervisados permiten entrenar algoritmos a partir de variables explicativas y etiquetas históricas, con el objetivo de estimar patrones que puedan utilizarse posteriormente sobre observaciones no vistas. En finanzas, estos patrones pueden referirse a retornos, riesgos, regímenes, señales de mercado o decisiones de asignación de capital.

Sin embargo, la aplicación de aprendizaje automático en mercados financieros exige cautela metodológica. Los datos financieros son temporales, ruidosos y no estacionarios; además, las relaciones observadas en el pasado pueden debilitarse o desaparecer en el futuro. López de Prado (2018) subraya la importancia de diseñar procedimientos que reduzcan el sobreajuste y el *data leakage* en aprendizaje automático financiero. Dixon, Halperin y Bilokon (2020) también presentan el aprendizaje automático en finanzas como un conjunto de herramientas cuantitativas que requiere validación rigurosa, formulación adecuada del problema y lectura prudente de los resultados.

Esta cautela coincide con una idea más general del aprendizaje automático: los modelos no son oráculos, sino aproximaciones estadísticas dependientes de los datos, las variables elegidas, la función objetivo y el contexto de evaluación. Domingos (2012) resume varias lecciones prácticas sobre aprendizaje automático que resultan aplicables a este proyecto, entre ellas la importancia de la generalización, la representación de variables y el riesgo de confundir buen ajuste histórico con capacidad predictiva real. En el caso financiero, esta distinción es especialmente importante porque pequeñas mejoras en una métrica de clasificación pueden no traducirse en una mejora económica después de costes, restricciones, diferencias de capital invertido o cambios de régimen.

Por ello, este TFM emplea la inteligencia artificial como herramienta de apoyo a decisiones, no como mecanismo para predecir con certeza el mercado. El objetivo no es estimar el nivel futuro exacto del NASDAQ100 ni anticipar de forma infalible sus subidas y caídas. El sistema busca producir señales que puedan ayudar a modular una estrategia DCA: aportar menos, mantener una aportación base o aportar más. La utilidad de esas señales no se evalúa únicamente mediante métricas predictivas, sino también mediante backtesting financiero, comparación con referencias simples y análisis del capital invertido.

El enfoque adoptado separa tres niveles que conviene no confundir. El primer nivel es el modelado predictivo, donde se entrenan modelos supervisados con variables macroeconómicas y técnicas. El segundo nivel es la decisión de aportación, donde las

salidas de los modelos se transforman en multiplicadores DCA. El tercer nivel es la evaluación financiera, donde se mide el efecto de esas decisiones sobre valor de cartera, retorno, drawdown, TIR y eficiencia del capital adicional. Esta separación contribuye a evitar conclusiones demasiado fuertes: un modelo puede parecer razonable en clasificación y, aun así, no producir una estrategia financieramente superior.

1.5. Planteamiento general del sistema macro-técnico propuesto

El sistema propuesto se concibe como una arquitectura macro-técnica jerárquica para modular aportaciones DCA mensuales sobre el NASDAQ100. La arquitectura se compone de tres bloques principales: un analista macroeconómico, un analista técnico y un módulo de backtesting financiero. Cada bloque cumple una función diferenciada y evita concentrar toda la decisión en un único modelo.

El analista macroeconómico sintetiza información económica, financiera y de mercado en una señal de contexto. Este bloque no decide directamente cuánto aportar, sino que proporciona una lectura previa del entorno que será utilizada por la capa técnica. El analista técnico combina ese contexto con señales derivadas del comportamiento del NASDAQ100 y estima la agresividad de aportación dentro de la estrategia periódica.

El tercer bloque convierte las salidas del sistema en aportaciones mensuales simuladas y las evalúa mediante backtesting. Esta fase permite comparar reglas dinámicas frente a referencias simples y comprobar si la modulación de aportaciones tiene sentido económico, no solo predictivo. Los detalles de implementación, variables operativas, particiones temporales y métricas financieras se desarrollan en los capítulos metodológicos y de resultados.

El planteamiento general del proyecto responde a una idea central: construir un marco disciplinado, reproducible y evaluable para estudiar decisiones de aportación DCA bajo incertidumbre financiera. Por ello, la interpretación del sistema se apoya en validación temporal, separación entre entrenamiento y test, evaluación walk-forward y lectura prudente de resultados.

En consecuencia, el Capítulo 2 formula el objetivo general del Trabajo Fin de Máster: desarrollar y evaluar un sistema de apoyo a la decisión para modular dinámicamente aportaciones periódicas mediante DCA sobre el NASDAQ100, integrando contexto macroeconómico, señales técnicas, aprendizaje supervisado y evaluación financiera mediante backtesting.

Capítulo 2. Objetivos del proyecto

Este capítulo define los objetivos que guían el Trabajo Fin de Máster en su versión final. El planteamiento inicial del proyecto partía de una idea ambiciosa: utilizar técnicas de inteligencia artificial para adaptar dinámicamente las aportaciones periódicas sobre el NASDAQ100. Sin embargo, el desarrollo técnico posterior, la disponibilidad real de datos, el tamaño muestral mensual y los resultados de validación aconsejan formular los objetivos de forma prudente y alineada con lo efectivamente implementado.

Por este motivo, la finalidad académica y técnica se formula como la construcción, validación y evaluación de un sistema macro-técnico de apoyo a la decisión que permita modular la agresividad de una estrategia Dollar Cost Averaging (DCA), manteniendo una separación clara entre modelado predictivo, decisión de aportación y evaluación financiera.

La formulación de los objetivos se apoya en la lógica final del proyecto: diseñar un marco disciplinado para decidir cuánto aportar, medir esa decisión respetando el orden temporal de los datos y compararla con referencias simples mediante criterios económicos exigentes. En consecuencia, el éxito del trabajo debe valorarse por la trazabilidad, reproducibilidad e interpretabilidad del sistema y de sus limitaciones.

2.1. Objetivo general

El objetivo general del Trabajo Fin de Máster es:

Desarrollar y evaluar un sistema de apoyo a la decisión para modular dinámicamente aportaciones periódicas mediante Dollar Cost Averaging sobre el NASDAQ100, integrando contexto macroeconómico, señales técnicas, aprendizaje supervisado y evaluación financiera mediante backtesting.

Este objetivo general sitúa el trabajo en el ámbito de la inteligencia artificial aplicada a series temporales financieras. El sistema se entiende como un modulador de agresividad de aportación: a partir de información macroeconómica y técnica, estima si en un determinado mes habría sido razonable aportar menos, mantener una aportación base o aportar más capital dentro de una estrategia DCA.

La elección del NASDAQ100 como activo de referencia permite concretar el análisis sobre una serie financiera identificable y relevante para una estrategia de inversión periódica. No obstante, el alcance del objetivo se limita a evaluar el comportamiento histórico de la arquitectura sobre este índice y con los datos disponibles en el proyecto.

2.2. Objetivos específicos

Para alcanzar el objetivo general, el proyecto se descompone en los siguientes objetivos específicos:

1. Construir un dataset mensual macroeconómico, financiero y técnico que permita representar el contexto de mercado, la evolución del NASDAQ100 y las variables necesarias para entrenar modelos supervisados. Este objetivo incluye la recopilación, limpieza, transformación y alineación temporal de series heterogéneas, así como la generación de datasets diferenciados para features, etiquetas y evaluación posterior.

2. Desarrollar un analista macroeconómico que actúe como filtro de contexto dentro del sistema. Este módulo debe sintetizar información macroeconómica y financiera en una señal operativa, expresada mediante variables como `macro_regime` y `macro_prob`, que pueda ser utilizada por el bloque técnico sin incorporar información futura no disponible en el momento de decisión.
3. Desarrollar un analista técnico que transforme señales del NASDAQ100 y contexto macroeconómico en niveles de agresividad DCA. Este objetivo implica construir variables relacionadas con momentum, tendencia, volatilidad, drawdown, posición relativa y calidad retorno-riesgo, y formular un target técnico orientado a decidir entre aportación reducida, aportación base y aportación agresiva.
4. Integrar el analista macroeconómico y el analista técnico en una arquitectura macro-técnica jerárquica. En esta arquitectura, el bloque macro no decide directamente la aportación, sino que proporciona contexto al bloque técnico; el bloque técnico combina dicho contexto con señales del índice y produce una decisión de agresividad; finalmente, el módulo financiero convierte esa decisión en una estrategia DCA evaluable.
5. Aplicar validación temporal y walk-forward validation para reducir el riesgo de data leakage y evaluar los modelos de forma coherente con la naturaleza temporal del problema. Este objetivo exige respetar el orden cronológico de las observaciones, separar entrenamiento y test final, y utilizar esquemas expansivos de validación que eviten entrenar con información posterior a la fecha evaluada.
6. Transformar las predicciones de los modelos en reglas de aportación mediante multiplicadores de capital 0.5x, 1x y 2x. Esta conversión conecta el resultado predictivo con una decisión financiera concreta: aportar la mitad de la cantidad base, mantener la aportación normal o duplicarla, según la agresividad estimada por el sistema.
7. Evaluar financieramente las estrategias generadas mediante backtesting, comparándolas frente a una estrategia DCA base y frente a un benchmark Buy & Hold. Esta evaluación debe considerar no solo el valor final de cartera o el PnL bruto, sino también el capital invertido, el retorno sobre capital, el drawdown, la TIR anualizada y la eficiencia del capital extra cuando existan diferencias de aportación entre estrategias.

Estos objetivos específicos reflejan la evolución metodológica real del proyecto. El trabajo no se limita a entrenar modelos, sino que construye una cadena completa desde datos brutos hasta decisiones financieras simuladas. Además, incorpora una lectura crítica de los resultados, diferenciando entre utilidad predictiva y utilidad económica. Esta distinción es esencial porque un modelo con mejores métricas de clasificación no necesariamente produce una estrategia de aportación superior frente a referencias simples.

2.3. Criterios de éxito y alcance del proyecto

Los criterios de éxito del proyecto se definen de forma realista y coherente con la naturaleza experimental del Trabajo Fin de Máster. El éxito no consiste en demostrar que el sistema bate al mercado ni en obtener una estrategia garantizada para maximizar rentabilidad, sino en construir y evaluar rigurosamente un marco de apoyo a la decisión que permita analizar si la modulación de aportaciones DCA puede estudiarse de forma trazable, temporalmente coherente y financieramente interpretable.

Un primer criterio de éxito es que el pipeline completo sea trazable y reproducible. Debe ser posible seguir la relación entre extracción de datos, construcción de datasets, generación de variables, etiquetado, entrenamiento de modelos, producción de predicciones y evaluación financiera. La existencia de artefactos intermedios, reportes y tablas finales permite auditar el proceso y evita que las conclusiones dependan de resultados aislados no conectados entre sí.

Un segundo criterio es que los datasets y artefactos estén correctamente contruidos. Esto implica que las series se alineen a frecuencia mensual, que las variables auxiliares de etiquetado se separen de las features usadas en entrenamiento y que las salidas de cada bloque sean compatibles con el siguiente. En particular, el contexto macroeconómico utilizado por el analista técnico debe proceder de las predicciones del bloque macro y no de etiquetas futuras que no estarían disponibles en una aplicación operativa.

Un tercer criterio es que los modelos se entrenen y evalúen respetando el orden temporal. En series financieras, una evaluación aleatoria o una mezcla indebida de periodos puede generar fuga de información y métricas artificialmente optimistas. Por ello, el proyecto considera satisfactorio que los modelos se validen mediante esquemas temporales, incluyendo separación entre entrenamiento y test final y validación walk-forward cuando corresponda.

Un cuarto criterio es que las predicciones puedan transformarse en decisiones de aportación. El sistema no debe quedarse únicamente en métricas predictivas abstractas; debe producir señales que puedan traducirse en multiplicadores DCA interpretables. La conversión a 0.5x, 1x y 2x permite evaluar el efecto económico de las decisiones y comprobar si una mejora predictiva tiene impacto real en una estrategia de inversión periódica.

Un quinto criterio es que el sistema pueda compararse financieramente frente a referencias claras. La estrategia DCA base representa el benchmark operativo más comparable, porque comparte la lógica de aportaciones periódicas. Buy & Hold se incorpora como referencia pasiva de mercado, con una mecánica distinta que debe interpretarse con cautela.

Un sexto criterio es que las conclusiones sean interpretables y prudentes. Si las estrategias dinámicas no superan de forma clara y estable al DCA base, el proyecto debe reconocerlo explícitamente. Del mismo modo, si Buy & Hold obtiene mayor retorno sobre capital en un periodo alcista, esta lectura debe contextualizarse sin invalidar automáticamente la utilidad metodológica del sistema DCA. La interpretación de resultados debe distinguir entre evidencia exploratoria, métricas moderadas y ausencia de prueba formal de significancia estadística.

Un séptimo criterio es que el trabajo identifique limitaciones y líneas futuras de mejora. Un sistema de apoyo a la decisión aplicado a datos financieros debe reconocer las restricciones derivadas del tamaño muestral, la frecuencia mensual, la dependencia del periodo histórico analizado, la discretización de targets, las diferencias de capital invertido y la ausencia de costes, fiscalidad o fricciones operativas. La identificación de estas limitaciones forma parte del resultado académico del proyecto.

El alcance del trabajo queda delimitado por varias decisiones metodológicas. En primer lugar, el sistema se valida sobre el NASDAQ100 como activo de referencia, por lo que los resultados no generalizan automáticamente a otros índices, ETFs, sectores o carteras multi-activo. En segundo lugar, la frecuencia principal es mensual, coherente con una estrategia

DCA de aportaciones periódicas y con la disponibilidad de muchas variables macroeconómicas, aunque esta elección limita el tamaño muestral.

En tercer lugar, el backtesting no incorpora costes de transacción, fiscalidad, restricciones de liquidez ni fricciones operativas. Por tanto, los resultados financieros deben interpretarse como una simulación histórica controlada y no como una estimación completa de rentabilidad neta para un inversor real. Bajo este alcance, el proyecto se evalúa como marco experimental y metodológico para apoyar decisiones de aportación, comparar alternativas y entender las condiciones bajo las cuales una modulación dinámica puede o no aportar valor frente a referencias simples.

En síntesis, el proyecto se considera satisfactorio si consigue construir un sistema macro-técnico completo, validar sus modelos de forma temporalmente coherente, convertir sus predicciones en reglas DCA evaluables y extraer conclusiones honestas sobre su utilidad y sus límites. Esta definición de éxito es coherente con el valor principal del trabajo: ofrecer un pipeline trazable, modular y evaluable para estudiar decisiones de aportación periódica.

Capítulo 3. Marco teórico

Este capítulo presenta los conceptos teóricos necesarios para comprender el sistema desarrollado en el Trabajo Fin de Máster. Su finalidad no es ofrecer una revisión exhaustiva de finanzas cuantitativas ni un manual general de aprendizaje automático, sino establecer el marco mínimo que justifica las decisiones metodológicas empleadas en los capítulos posteriores. El proyecto combina inversión periódica, señales macroeconómicas y técnicas, aprendizaje supervisado, validación temporal y backtesting; por tanto, el marco teórico se centra únicamente en esos elementos.

El punto de partida es una estrategia Dollar Cost Averaging (DCA) sobre el NASDAQ100. A partir de ella, el trabajo plantea una modulación dinámica de las aportaciones mediante multiplicadores de capital. La pregunta teórica no es si puede predecirse con certeza el nivel futuro del índice, sino si un conjunto de variables macro-técnicas puede organizarse en un sistema de apoyo a la decisión capaz de sugerir aportaciones reducidas, base o agresivas y de evaluar posteriormente esas decisiones con métricas financieras. Esta distinción es importante porque separa el problema predictivo del problema económico: un modelo puede ofrecer cierta señal estadística y, aun así, no generar una estrategia financieramente superior.

3.1. Fundamentos de inversión cuantitativa y asset allocation

La inversión cuantitativa se basa en el uso sistemático de datos, reglas y modelos para apoyar decisiones de inversión. A diferencia de un enfoque puramente discrecional, una aproximación cuantitativa obliga a definir variables observables, criterios de decisión, reglas de validación y métricas de evaluación. Esto no elimina la incertidumbre financiera, pero permite que el proceso sea más trazable y reproducible. En el contexto de este TFM, la dimensión cuantitativa aparece en la construcción de datasets mensuales, la definición de targets, el entrenamiento de modelos supervisados y la conversión de sus salidas en decisiones de aportación.

El concepto de *asset allocation* se refiere a la asignación de capital entre activos, estrategias o niveles de exposición. La teoría moderna de carteras ya planteaba que la decisión relevante no es solo seleccionar activos individuales, sino combinar rentabilidad esperada y riesgo dentro de una política de inversión (Markowitz, 1952). En este trabajo, el problema se formula de manera más acotada: no se decide entre múltiples activos, sino cuánto capital aportar periódicamente al NASDAQ100 dentro de una estrategia DCA. Aun así, la lógica subyacente conserva una dimensión de asignación de capital dentro de un único activo: aportar menos, mantener la aportación base o aportar más según el contexto estimado.

La gestión cuantitativa de carteras también insiste en que las señales deben evaluarse con criterios empíricos y no únicamente por su plausibilidad narrativa. Qian, Hua y Sorensen (2007) presentan la gestión cuantitativa de renta variable como un proceso que combina señales, construcción de cartera, control de riesgo y evaluación del rendimiento. Esta idea se adapta al proyecto de forma sencilla: las variables macroeconómicas y técnicas no se utilizan como reglas aisladas de compra o venta, sino como entradas de un sistema que produce una decisión concreta de aportación. El resultado final debe evaluarse mediante backtesting, no solo mediante métricas predictivas.

Por tanto, el proyecto se sitúa en un punto intermedio entre una regla pasiva y una estrategia cuantitativa completa de asignación multi-activo. Conserva la disciplina de la inversión periódica, pero introduce una capa de decisión basada en datos. La aportación principal no consiste en prometer una asignación óptima del capital, sino en construir un marco formal para estudiar si la aportación periódica puede modularse de forma trazable. Esta formulación evita confundir el uso de modelos con una capacidad automática de anticipar precios: los modelos apoyan la decisión, pero no sustituyen la incertidumbre propia de los mercados financieros.

3.2. Estrategias DCA y limitaciones de la aportación constante

Como se introdujo en el Capítulo 1, el Dollar Cost Averaging es una regla de inversión periódica basada en aportaciones constantes. En el marco teórico interesa destacar no tanto su mecánica básica, sino las implicaciones que esa regla tiene para la asignación temporal del capital, la disciplina del inversor y la comparación con alternativas de inversión inicial única.

La literatura académica, sin embargo, muestra que el DCA no debe presentarse como una estrategia universalmente superior. Constantinides (1979) analiza su suboptimalidad bajo ciertos supuestos teóricos, especialmente cuando el inversor ya dispone de todo el capital y los activos tienen rentabilidad esperada positiva. Rozeff (1994) compara la inversión inicial única con el promedio de aportaciones y plantea que la conveniencia de una u otra alternativa depende del horizonte, el riesgo y el recorrido de mercado. En consecuencia, el DCA debe entenderse como una regla de inversión periódica disciplinada, no como una garantía de maximización de rentabilidad.

También existen argumentos relacionados con riesgo a la baja y comportamiento del inversor. Leggio y Lien (2003) estudian la efectividad del DCA mediante medidas de riesgo a la baja, mientras que Dichtl y Drobetz (2011) explican parte de su popularidad desde la teoría prospectiva y las preferencias conductuales. Esta lectura es relevante para el TFM porque una estrategia de aportaciones periódicas no solo debe evaluarse por rentabilidad final, sino también por su estabilidad, su facilidad de aplicación y su capacidad de adaptarse a restricciones reales de capital.

La limitación principal del DCA tradicional es que aporta siempre la misma cantidad. Esa constancia puede ser deseable por disciplina, pero no incorpora por sí misma información sobre volatilidad, drawdown, momentum o entorno macroeconómico. El sistema desarrollado en este trabajo estudia precisamente esa limitación: mantiene la lógica periódica, pero permite que la aportación mensual se module mediante multiplicadores discretos.

En el proyecto, la aportación dinámica se expresa mediante tres niveles: $0.5x$, $1x$ y $2x$. El multiplicador $1x$ equivale al DCA base; $0.5x$ representa una aportación reducida; y $2x$ representa una aportación agresiva. Esta formulación conserva la pregunta central de una estrategia DCA, que es cuánto capital aportar periódicamente, pero introduce una decisión condicionada por el contexto. No obstante, el hecho de modular aportaciones no implica que la estrategia sea necesariamente superior. La utilidad de la modulación solo puede valorarse después de convertir las predicciones en aportaciones y comparar los resultados frente a referencias simples.

3.3. Series temporales financieras y aprendizaje supervisado

Las series temporales financieras presentan características que condicionan cualquier intento de modelado. En primer lugar, las observaciones tienen un orden cronológico que no puede ignorarse. Un dato de mercado observado en una fecha posterior no está disponible para tomar decisiones en una fecha anterior. En segundo lugar, las series financieras suelen estar afectadas por ruido, cambios de volatilidad, episodios extremos y relaciones inestables entre variables. Campbell, Lo y MacKinlay (1997) y Tsay (2010) destacan que el análisis de mercados financieros requiere prestar atención a retornos, volatilidad, dependencia temporal y comportamiento no estacionario.

En este proyecto, la frecuencia principal es mensual. Esta elección es coherente con una estrategia DCA de aportaciones periódicas y con la disponibilidad de muchas variables macroeconómicas, pero introduce una restricción importante: el número de observaciones efectivas es reducido. Aunque el periodo histórico cubra más de dos décadas, una frecuencia mensual genera solo unas pocas centenas de filas. Además, cuando los targets se construyen usando retornos o drawdowns futuros, las últimas observaciones no pueden etiquetarse sin mirar más allá del final de la serie. Por ello, la muestra disponible para entrenamiento y evaluación es limitada.

Otra particularidad relevante es que las variables financieras pueden describir dimensiones distintas del mercado. Los retornos recientes aproximan momentum; la volatilidad mide variabilidad; el drawdown resume caídas desde máximos previos; y las medias móviles pueden describir tendencia. Ninguna de estas variables predice por sí sola el futuro del índice, pero todas pueden actuar como descriptores del estado actual del mercado. En el sistema propuesto, estas variables se integran como *features* dentro del analista técnico.

El aprendizaje supervisado proporciona el marco para transformar datos históricos en modelos predictivos. En términos generales, se parte de una matriz de variables de entrada, o *features*, y de una variable objetivo, o *target*, que se desea estimar. El modelo se entrena sobre observaciones pasadas y después se evalúa sobre observaciones no utilizadas durante el entrenamiento. Hastie, Tibshirani y Friedman (2009) y Bishop (2006) presentan esta lógica como base de los problemas de clasificación y regresión.

En el TFM aparecen ambos enfoques. La clasificación se utiliza cuando el objetivo consiste en asignar una observación a una clase, por ejemplo detectar si un entorno macro-financiero es favorable o si un mes corresponde a una aportación agresiva. La regresión se utiliza cuando el objetivo es estimar una magnitud continua, como el `Score_t_tech`. Posteriormente, una predicción continua puede discretizarse en niveles de aportación. Esta coexistencia de clasificación y regresión es importante porque el proyecto no trata un único problema predictivo, sino varias formulaciones orientadas a una decisión financiera final.

La aplicación del aprendizaje supervisado a finanzas exige prudencia. Los modelos pueden aprender patrones históricos que no se repiten en el futuro, especialmente cuando el tamaño muestral es pequeño o el número de variables es elevado. López de Prado (2018) advierte sobre el sobreajuste y la importancia de diseñar validaciones adecuadas en aprendizaje automático financiero. Por ello, en este TFM el rendimiento predictivo se interpreta como una señal parcial y se complementa con evaluación financiera mediante backtesting.

3.4. Detección de regímenes de mercado y filtros macroeconómicos

Los mercados financieros no se comportan de manera uniforme en todos los periodos. Existen fases de crecimiento, corrección, elevada volatilidad, tensión financiera o recuperación. En la práctica, muchas estrategias intentan incorporar algún tipo de lectura del entorno para no tratar todos los meses como observaciones equivalentes. Esta idea puede expresarse mediante el concepto de régimen de mercado, aunque en este trabajo se utiliza de forma prudente: no se afirma que el modelo identifique regímenes económicos perfectos, sino que genera un filtro de contexto.

Un filtro macroeconómico resume información del entorno económico y financiero para apoyar una decisión posterior. Puede incorporar variables relacionadas con inflación, tipos de interés, mercado laboral, liquidez, volatilidad, divisas, materias primas o sentimiento. La utilidad de este tipo de filtro no depende de predecir con exactitud todo el ciclo económico, sino de aportar una señal sintética que pueda combinarse con otras dimensiones del mercado. En series financieras, esta función de contexto resulta especialmente relevante porque las relaciones entre señales técnicas y retornos pueden variar según el entorno macro-financiero.

En el proyecto, el analista macroeconómico cumple precisamente esta función. Su salida no decide directamente la aportación mensual, sino que produce variables de contexto como `macro_regime` y `macro_prob`. Estas variables se incorporan al analista técnico para que la decisión de agresividad DCA no dependa solo de la serie del NASDAQ100, sino también de una lectura macro-financiera previa. La arquitectura mantiene así una separación jerárquica: primero se estima el contexto, después se combina con señales técnicas y finalmente se transforma en una regla de aportación.

Desde un punto de vista teórico, esta separación ayuda a evitar una interpretación excesiva del bloque macro. El modelo macro no debe entenderse como una predicción completa del mercado ni como una clasificación perfecta del ciclo económico. Su papel es actuar como filtro de entorno favorable o desfavorable a partir de datos históricos y validación temporal. Este enfoque es coherente con la literatura de aprendizaje automático financiero, que recomienda evaluar las señales dentro de un pipeline completo y no atribuirles por separado una capacidad predictiva definitiva (López de Prado, 2018; Dixon et al., 2020).

La incorporación del filtro macro también tiene implicaciones metodológicas. Si el analista técnico utilizara directamente una etiqueta macro construida con información futura, el sistema introduciría fuga de información. Por ello, el bloque técnico debe consumir salidas generadas por el modelo macro, no etiquetas perfectas observadas retrospectivamente. Esta decisión conecta el marco teórico con la implementación del TFM: la señal macro debe estar disponible en el momento de decisión simulado.

3.5. Indicadores técnicos y momentum aplicado a índices bursátiles

Los indicadores técnicos utilizados en este proyecto no se plantean como reglas aisladas de compra o venta, sino como descriptores cuantitativos del comportamiento reciente del NASDAQ100. Su función es resumir diferentes dimensiones del estado del índice: fuerza

reciente, tendencia, volatilidad, drawdown, posición relativa y calidad retorno-riesgo. Esta interpretación es más prudente que atribuir a cada indicador una capacidad predictiva independiente.

El momentum describe la fuerza de los movimientos recientes. En el proyecto se aproxima mediante retornos a distintos horizontes y mediante indicadores como el RSI. Un retorno positivo a varios meses puede indicar que el índice viene de una fase alcista, mientras que retornos negativos pueden señalar debilidad reciente. Sin embargo, momentum no significa certeza de continuidad. En una serie financiera, una señal de fuerza puede revertirse, intensificarse o perder relevancia según el entorno de mercado.

La tendencia se resume mediante medias móviles y ratios entre precio y media. Si el precio se sitúa por encima de una media móvil, puede interpretarse como una señal descriptiva de tendencia positiva; si se sitúa por debajo, puede indicar deterioro relativo. Las pendientes de medias móviles añaden información sobre la dirección reciente de esa referencia. En el TFM, estas variables se utilizan como *features* técnicas que ayudan al modelo a describir el estado del NASDAQ100, no como reglas deterministas.

La volatilidad y el drawdown capturan dimensiones de riesgo. La volatilidad resume cuánto fluctúan los retornos, mientras que el drawdown mide la caída desde un máximo previo. En estrategias de inversión periódica, estas variables son relevantes porque la decisión de aportar más o menos no debería depender únicamente del retorno esperado, sino también del riesgo atravesado. Magdon-Ismail y Atiya (2004) formalizan el maximum drawdown como una medida de caída desde máximos que resulta útil para evaluar trayectorias de inversión.

La posición relativa dentro de un rango reciente permite identificar si el índice se encuentra cerca de máximos o en zonas más deprimidas. Esta información no equivale a decir que el activo esté barato o caro en sentido fundamental, pero puede ayudar a contextualizar el movimiento reciente. Finalmente, la calidad retorno-riesgo combina retorno futuro y drawdown futuro para construir el score técnico del proyecto. Esta variable no se usa como *feature*, sino como parte del proceso de etiquetado del target técnico.

El analista técnico integra estas señales con el contexto macroeconómico. Su objetivo no es predecir el precio exacto del NASDAQ100, sino transformar descriptores técnicos y macroeconómicos en niveles de agresividad DCA. De este modo, el análisis técnico queda subordinado a una decisión concreta de aportación. La salida del modelo solo adquiere significado financiero cuando se convierte en multiplicadores 0.5x, 1x o 2x y se evalúa mediante backtesting.

3.6. Validación temporal y riesgo de data leakage

La validación temporal es un requisito central en problemas con series financieras. En un dataset tabular convencional puede ser aceptable separar aleatoriamente observaciones de entrenamiento y test si se cumplen ciertos supuestos de independencia. En series temporales financieras, esta práctica puede ser problemática porque mezcla pasado y futuro. Un modelo que se entrena con información posterior a la fecha evaluada no reproduce una situación real de decisión y puede generar métricas artificialmente optimistas.

La regla fundamental es que cada predicción debe realizarse utilizando únicamente información que habría estado disponible antes o en la fecha de decisión. Hyndman y Athanasopoulos (2021) explican que la evaluación de modelos de series temporales debe

respetar la estructura cronológica de los datos. En el proyecto, esta idea se traduce en particiones temporales, test final y validación *walk-forward* mediante folds contruidos explícitamente por fecha. En lugar de una validación aleatoria, cada bloque de evaluación se define respetando la secuencia cronológica, de modo que el entrenamiento solo utiliza observaciones anteriores al periodo evaluado.

La validación *walk-forward* utiliza una ventana expansiva. El modelo se entrena con la historia disponible hasta una fecha determinada y se evalúa sobre el bloque temporal siguiente. Después, la ventana de entrenamiento avanza incorporando nueva información pasada y se repite el proceso. Esta metodología no garantiza que el modelo generalice perfectamente en el futuro, pero ofrece una evaluación más coherente que una partición aleatoria cuando las observaciones están ordenadas temporalmente.

El riesgo de *data leakage* aparece cuando el modelo recibe, directa o indirectamente, información que no estaría disponible en el momento de la predicción. Kaufman, Rosset, Perlich y Stitelman (2012) analizan este problema como una fuente de evaluaciones engañosas en minería de datos. En finanzas, el riesgo es especialmente importante porque muchas etiquetas se construyen con información futura: retornos posteriores, drawdowns futuros o scores calculados sobre ventanas posteriores.

En este TFM, varias columnas se utilizan para etiquetar observaciones pero no pueden introducirse como *features*. Por ejemplo, `R_t_6`, `DD_t_6`, `Score_t_tech`, `Y_tech` y otros targets derivan de información posterior a la fecha de decisión. Si se incluyeran en la matriz de entrenamiento, el modelo estaría aprendiendo con información futura. Por ello, el pipeline excluye columnas auxiliares de etiquetado, retornos futuros, drawdowns futuros, scores y targets antes de entrenar.

El *data leakage* también puede aparecer en transformaciones aparentemente neutras. Por ejemplo, imputar valores o escalar variables usando toda la muestra antes de separar entrenamiento y test puede incorporar información del periodo evaluado. Para reducir este riesgo, el proyecto utiliza pipelines de modelado en los que las transformaciones se ajustan dentro del tramo de entrenamiento. Esta decisión no elimina todos los riesgos metodológicos, pero hace que la evaluación sea más coherente con la disponibilidad temporal de los datos.

3.7. Algoritmos utilizados

Los algoritmos empleados en el proyecto pertenecen a familias conocidas de aprendizaje supervisado. Se han utilizado modelos lineales y modelos basados en árboles, tanto para clasificación como para regresión. La elección de estos algoritmos responde a una combinación de interpretabilidad relativa, disponibilidad en `scikit-learn`, adecuación a muestras tabulares y capacidad para comparar formulaciones alternativas. No debe interpretarse que un modelo más complejo garantice mejor rendimiento financiero.

3.7.1. Regresión logística

La regresión logística es un modelo supervisado utilizado para problemas de clasificación. Estima la probabilidad de pertenencia a una clase a partir de una combinación de variables de entrada. En clasificación binaria, permite modelar la probabilidad de que una observación pertenezca a la clase positiva; en formulaciones multiclase puede extenderse para asignar

observaciones entre varias categorías. Hosmer, Lemeshow y Sturdivant (2013) presentan la regresión logística como una herramienta estándar para modelar respuestas categóricas.

En el TFM, la regresión logística tiene sentido como modelo de referencia porque es relativamente simple y permite comparar si relaciones lineales entre variables macro-técnicas y target contienen señal suficiente. Su simplicidad es útil en muestras mensuales pequeñas, donde modelos demasiado flexibles pueden sobreajustar. Dentro del analista técnico, se utiliza como candidato para clasificación de niveles de agresividad DCA.

3.7.2. Random Forest

Random Forest es un método de ensamblado basado en múltiples árboles de decisión. Cada árbol se entrena sobre variaciones de los datos y de las variables consideradas, y la predicción final se obtiene agregando las predicciones individuales. Breiman (2001) introduce Random Forest como un enfoque que combina árboles para reducir la varianza frente a un único árbol de decisión.

Este tipo de modelo resulta adecuado para datasets tabulares porque puede capturar relaciones no lineales e interacciones entre variables sin exigir una especificación funcional previa. En el proyecto se utiliza como candidato en el analista macroeconómico y en el bloque técnico. Su uso debe interpretarse como una opción experimental dentro del pipeline, no como garantía de superioridad. La elección final de un modelo depende de la validación temporal y de las métricas registradas, no de la complejidad del algoritmo por sí misma.

3.7.3. Extra Trees

Extra Trees, o *Extremely Randomized Trees*, es otro método de ensamblado basado en árboles. Geurts, Ernst y Wehenkel (2006) proponen introducir aleatoriedad adicional en la selección de puntos de corte y variables, lo que puede ayudar a controlar la varianza en ciertos problemas tabulares, aunque su rendimiento depende del dataset y de la validación aplicada. Al igual que Random Forest, permite capturar relaciones no lineales y manejar interacciones entre variables.

En el TFM, Extra Trees aparece como candidato tanto en clasificación como en regresión. Su papel más relevante es explorar si un modelo basado en árboles con mayor aleatoriedad puede detectar patrones útiles en el problema técnico, especialmente en formulaciones binarias orientadas a identificar meses de aportación agresiva. No obstante, su interpretación debe seguir siendo prudente: que un modelo mejore una métrica predictiva no implica automáticamente que genere una estrategia de aportación financieramente superior.

3.7.4. Ridge Regression

Ridge Regression es un modelo lineal de regresión que incorpora regularización. La regularización penaliza coeficientes excesivamente grandes y puede mejorar la estabilidad cuando existen muchas variables o colinealidad entre ellas. Hoerl y Kennard (1970) introducen la regresión ridge como una técnica de estimación sesgada para problemas con variables correlacionadas.

En el proyecto, Ridge Regression se utiliza para modelar el `Score_t_tech`, una construcción específica del TFM, como variable continua. Esta formulación responde a una limitación potencial de la clasificación directa: si las fronteras entre clases 0.5x, 1x y 2x son

ruidosas, puede ser razonable estimar primero un score continuo y discretizar después la predicción. La utilidad de este enfoque no se decide solo por MAE o RMSE, sino por el comportamiento financiero de las aportaciones derivadas.

3.8. Métricas de evaluación

La evaluación del proyecto combina métricas predictivas y métricas financieras. Esta combinación es necesaria porque el objetivo final no es maximizar una métrica abstracta de clasificación o regresión, sino estudiar si las predicciones pueden convertirse en decisiones de aportación evaluables. Por ello, las métricas se organizan en tres grupos: clasificación, regresión y backtesting financiero.

3.8.1. Métricas de clasificación

La *accuracy* mide el porcentaje total de aciertos del modelo. Es intuitiva, pero puede ser engañosa cuando las clases están desbalanceadas. Si una clase aparece con mucha frecuencia, un modelo puede obtener *accuracy* alta prediciendo casi siempre esa clase. Por esta razón, en el TFM se presta especial atención a la *balanced accuracy*, que calcula el rendimiento medio por clase y reduce el sesgo hacia la clase mayoritaria.

La *precision* mide qué proporción de las predicciones positivas son realmente positivas, mientras que el *recall* mide qué proporción de los casos positivos reales detecta el modelo. El F1-score combina *precision* y *recall* en una medida única. En problemas multiclase, el F1 macro promedia el F1 de cada clase tratando todas las clases con el mismo peso. Powers (2011) revisa estas métricas y muestra que cada una captura una dimensión distinta del rendimiento de clasificación.

ROC AUC evalúa la capacidad del modelo para ordenar observaciones positivas por encima de negativas a través de distintos umbrales. Fawcett (2006) presenta el análisis ROC como una herramienta útil para estudiar clasificadores binarios. En el proyecto, ROC AUC se utiliza cuando la formulación del problema lo permite, especialmente en clasificaciones binarias. No sustituye a *balanced accuracy* ni a F1; las complementa.

La documentación oficial de *scikit-learn* recoge estas métricas como parte de su módulo de evaluación de modelos (*scikit-learn developers*, n.d.-a). En el TFM, la lectura conjunta de *accuracy*, *balanced accuracy*, *precision*, *recall*, F1, F1 macro y ROC AUC es importante porque el número de observaciones de test es reducido y algunas clases pueden aparecer con poca frecuencia. Ninguna métrica aislada debe utilizarse para afirmar que un modelo es robusto.

3.8.2. Métricas de regresión

En regresión, el error absoluto medio, o MAE, mide la distancia media entre valores reales y predichos en valor absoluto. Es fácil de interpretar porque mantiene la escala de la variable objetivo. El RMSE, o raíz del error cuadrático medio, penaliza más los errores grandes porque eleva los errores al cuadrado antes de promediarlos. Hyndman y Koehler (2006) analizan distintas medidas de precisión en predicción y advierten que cada métrica enfatiza aspectos diferentes del error.

El R^2 mide qué proporción de la variabilidad de la variable objetivo explica el modelo frente a una referencia simple. Un R^2 cercano a uno indica mayor capacidad explicativa dentro de la

muestra evaluada; un valor cercano a cero indica poca mejora frente a la referencia; y un valor negativo indica que el modelo puede comportarse peor que una predicción simple. En el proyecto, esta métrica se interpreta con cautela porque el score técnico es una variable difícil de estimar y el tamaño de test es reducido.

La documentación de `scikit-learn` incluye MAE, RMSE y R^2 dentro de sus métricas de regresión (`scikit-learn developers`, n.d.-b). En este TFM, estas métricas se aplican a la formulación de regresión del `Score_t_tech`. Sin embargo, esta es solo una de las aproximaciones evaluadas dentro del analista técnico: tras estimar el score continuo, sus predicciones pueden discretizarse en multiplicadores DCA para compararlas con las formulaciones de clasificación. Por tanto, una mejora en MAE o RMSE debe evaluarse también en términos de decisión de aportación y resultado financiero.

3.8.3. Métricas financieras y de backtesting

El backtesting transforma predicciones en decisiones financieras simuladas. En este proyecto, cada mes se aplica un multiplicador de aportación y se compran participaciones del NASDAQ100 al nivel observado del índice. A partir de esta simulación se calculan métricas de capital, rentabilidad y riesgo. La primera métrica es el capital invertido, que suma todas las aportaciones realizadas por una estrategia. Es fundamental porque las estrategias dinámicas no necesariamente invierten el mismo capital que el DCA base.

El valor final de cartera mide el valor de las participaciones acumuladas al final del periodo evaluado. El PnL, o *Profit and Loss*, se calcula como la diferencia entre valor final y capital invertido. Estas dos métricas son intuitivas, pero pueden inducir a conclusiones incompletas: una estrategia puede terminar con mayor valor final simplemente porque ha invertido más dinero. Por ello, el retorno sobre capital mide la rentabilidad acumulada respecto al capital realmente aportado.

El maximum drawdown mide la mayor caída porcentual desde un máximo acumulado de la cartera. Es una métrica de riesgo relevante porque dos estrategias con rentabilidad similar pueden experimentar trayectorias muy distintas. Magdon-Ismail y Atiya (2004) analizan el maximum drawdown como una medida útil para describir pérdidas desde máximos. En el TFM, el drawdown aparece tanto en la evaluación de cartera como en la construcción de targets, aunque estos usos deben distinguirse: el drawdown futuro sirve para etiquetar, mientras que el drawdown histórico de cartera sirve para evaluar riesgo.

La TIR anualizada, o IRR anualizada, resulta especialmente adecuada cuando existen flujos de caja periódicos. En una estrategia DCA, el capital no se invierte todo al principio, sino que entra gradualmente. Por ello, una métrica que considere la secuencia temporal de aportaciones puede ser más informativa que una tasa compuesta calculada como si todo el capital se hubiera invertido desde el inicio. El CAGR, en cambio, es más natural para estrategias tipo Buy & Hold, donde se invierte un capital inicial y se mantiene hasta el final.

La volatilidad anual mide la variabilidad anualizada de los retornos, mientras que el Sharpe ratio relaciona rentabilidad y volatilidad. Sharpe (1994) formaliza este ratio como una medida de rendimiento ajustado por riesgo. En el proyecto se utiliza una versión con tipo libre de riesgo igual a cero para el benchmark pasivo, de acuerdo con las tablas finales generadas. Esta simplificación debe interpretarse como una decisión metodológica del backtest, no como una medición completa del rendimiento ajustado por riesgo en condiciones reales.

Finalmente, el proyecto incorpora la métrica `capital_extra_efficiency`. Esta métrica compara el PnL adicional de una estrategia dinámica frente al DCA base con el capital adicional invertido. Su función es distinguir si una estrategia genera más beneficio por decidir mejor o simplemente por aportar más capital. A diferencia de métricas financieras ampliamente estandarizadas, esta medida es específica del cierre del proyecto y se justifica por la existencia de estrategias con distintos niveles de capital invertido.

En conjunto, las métricas financieras permiten cerrar el ciclo entre teoría y aplicación. Las métricas predictivas indican si los modelos capturan alguna señal respecto a los targets definidos; el backtesting indica si esa señal se transforma en una decisión económica útil. Esta separación es esencial para mantener una interpretación prudente: el sistema puede estar correctamente construido y, aun así, no demostrar una superioridad financiera clara frente a una estrategia DCA base.

Capítulo 4. Material y metodología

Este capítulo describe los materiales, fuentes de datos, herramientas y decisiones metodológicas utilizadas para construir el sistema macro-técnico desarrollado en el Trabajo Fin de Máster. La finalidad del capítulo no es presentar los resultados de rendimiento de los modelos, sino explicar cómo se ha organizado el proyecto, qué información lo alimenta, cómo se transforman los datos y qué mecanismos se han empleado para reducir riesgos habituales en el modelado de series temporales financieras.

El sistema se plantea como un marco de apoyo a la decisión para modular aportaciones periódicas mediante Dollar Cost Averaging (DCA) sobre el NASDAQ100. La arquitectura se compone de tres bloques: un analista macroeconómico mensual, un analista técnico mensual y un módulo de evaluación financiera mediante backtesting. La metodología no asume que el sistema pueda predecir el mercado con certeza ni garantizar una mejora frente a estrategias pasivas. Su objetivo es construir un pipeline trazable y temporalmente coherente que permita transformar información macroeconómica y técnica en distintos niveles de agresividad de aportación.

4.1. Herramientas, entorno de trabajo y librerías utilizadas

El proyecto se desarrolló mediante notebooks de Python, organizados en dos bloques principales: analista macroeconómico y analista técnico. Junto a los notebooks, el workspace contiene datasets intermedios en formato CSV, modelos serializados y ficheros de metadatos en JSON. Esta estructura permite seguir la trazabilidad entre extracción de datos, construcción de datasets, entrenamiento de modelos, generación de predicciones y evaluación financiera.

Las librerías utilizadas se agrupan en varias funciones metodológicas. Para la manipulación y transformación de datos se emplearon pandas y NumPy. Estas herramientas permitieron cargar series temporales, homogeneizar fechas, calcular variaciones, retornos, drawdowns y preparar matrices tabulares de entrenamiento. Para visualización e inspección se utilizó matplotlib, especialmente en gráficos de series temporales, matrices de confusión y curvas de backtesting. Para el aprendizaje supervisado se empleó scikit-learn, incluyendo Pipeline, imputación con SimpleImputer, escalado con StandardScaler, modelos de clasificación y modelos de regresión. Los modelos y metadatos se almacenaron mediante joblib y json, respectivamente.

La evaluación financiera final incorpora además `scipy.optimize.brentq` para resolver numéricamente la tasa interna de retorno anualizada cuando la estructura de flujos de caja lo requiere. Esta herramienta se utiliza en la fase de evaluación, no como parte del entrenamiento predictivo.

La Tabla 1 resume las principales herramientas utilizadas y su papel dentro del proyecto.

Tabla 1: Herramientas y librerías principales utilizadas en el proyecto.

Herramienta o librería	Uso dentro del proyecto
Python y Jupyter Notebook	Desarrollo del pipeline, ejecución de experimentos y generación de artefactos intermedios.

Herramienta o librería	Uso dentro del proyecto
pandas y NumPy	Carga, limpieza, transformación y combinación de series temporales macroeconómicas, financieras y técnicas.
matplotlib	Visualización de series, inspección de variables, matrices de confusión y curvas de evaluación financiera.
scikit-learn	Construcción de pipelines, imputación, escalado, entrenamiento y evaluación de modelos supervisados.
joblib	Persistencia de modelos entrenados para su reutilización en fases posteriores del pipeline.
json	Almacenamiento de metadatos, listas de variables y configuración de modelos finales.
SciPy	Cálculo numérico auxiliar en la evaluación financiera, especialmente para la TIR anualizada.

No se consideran tecnologías finales aquellas que aparecieron únicamente como posibilidades en planteamientos iniciales o líneas futuras, pero que no forman parte de los artefactos consolidados del workspace. Por ello, este apartado se limita a las herramientas efectivamente observables en los notebooks y artefactos finales del proyecto.

4.1.1. Entorno de ejecución y reproducibilidad

El desarrollo del proyecto se realizó en un entorno local sobre Windows 10, utilizando Anaconda como distribución de Python y Jupyter Notebook como interfaz principal de ejecución. Para documentar el entorno de trabajo, se ejecutó una comprobación del kernel activo en la fase final del proyecto. Dicha comprobación registró Python 3.12.4 empaquetado por Anaconda, arquitectura AMD64 y ejecución desde el intérprete local de python.

Adicionalmente, el desarrollo del proyecto se mantuvo bajo control de versiones mediante Git. Este control de versiones se utilizó para registrar la evolución de notebooks, scripts y artefactos generados durante las distintas iteraciones del trabajo. Git no forma parte del entorno de ejecución de los modelos, pero contribuye a la trazabilidad del desarrollo, al permitir conservar cambios, comparar versiones y mantener una organización más controlada del código fuente y de los artefactos del proyecto.

Desde el punto de vista de reproducibilidad, el proyecto se organiza mediante notebooks numerados y artefactos intermedios persistidos en disco. Los datasets se almacenan en formato CSV, los metadatos y configuraciones relevantes se guardan en ficheros JSON y los modelos entrenados se serializan mediante `joblib`. Esta estructura facilita reconstruir el flujo completo desde la extracción y transformación de datos hasta el entrenamiento de modelos, la generación de predicciones y la evaluación financiera.

Además, el uso de Pipeline de scikit-learn permite encapsular transformaciones como imputación y escalado dentro del propio proceso de entrenamiento y evaluación. Esta decisión es relevante para evitar que operaciones de preprocesamiento utilicen información del conjunto de test o de periodos futuros. Por tanto, la reproducibilidad del proyecto no depende solo de disponer de las mismas librerías, sino también de mantener la misma secuencia temporal de ejecución, las mismas particiones cronológicas y los mismos artefactos intermedios.

La Tabla 2 resume las versiones principales detectadas en el entorno utilizado para la comprobación final.

Tabla 2: Entorno de ejecución y versiones principales detectadas en la comprobación final.

Componente	Versión o configuración detectada	Función dentro del proyecto
Sistema operativo	Windows 10 10.0.19045 SP0	Entorno local de desarrollo y ejecución.
Distribución	Anaconda	Gestión del intérprete Python y entorno de ejecución.
Entorno Conda detectado	base	Entorno registrado durante la comprobación del kernel activo.
Python	3.12.4	Lenguaje principal del proyecto.
Jupyter Notebook	7.0.8	Ejecución interactiva de notebooks y experimentos.
JupyterLab	4.0.11	Entorno disponible dentro de la instalación Jupyter.
ipykernel	6.28.0	Implementación del kernel de Python

Componente	Versión o configuración detectada	Función dentro del proyecto
		utilizado por Jupyter.
pandas	2.2.2	Carga, limpieza, alineación temporal y combinación de datasets.
NumPy	1.26.4	Cálculo numérico, operaciones vectorizadas y apoyo a transformaciones.
scikit-learn	1.4.2	Pipelines, imputación, escalado, modelos supervisados y métricas.
SciPy	1.13.1	Cálculo numérico auxiliar, especialmente en la evaluación financiera.
matplotlib	3.8.4	Generación de gráficos, curvas de backtesting y visualizaciones.
joblib	1.4.2	Serialización y reutilización de modelos entrenados.

Esta información documenta las versiones principales observadas en el kernel activo durante la comprobación final.

4.2. Fuentes de datos y periodo de análisis

El proyecto utiliza datos financieros, técnicos y macroeconómicos organizados a frecuencia mensual. La construcción de los datasets se realiza de forma progresiva. En primer lugar, el bloque macro transforma series económicas y de mercado en variables mensuales comparables. En segundo lugar, se construye un dataset maestro macroeconómico depurado. En tercer lugar, el bloque técnico genera señales mensuales del NASDAQ100 y las integra con el contexto macroeconómico producido por el modelo macro final.

Finalmente, el módulo financiero utiliza las salidas de los modelos técnicos para simular estrategias DCA.

La Tabla 3 resume los principales datasets generados. Se incluyen tanto datasets de features como datasets etiquetados, ya que cumplen funciones distintas dentro del pipeline.

Tabla 3: Datasets principales generados durante el proyecto.

Artefacto	Filas	Periodo	Función metodológica
MASTER_features_clean_v2.csv	300	2001-02 a 2026-01	Dataset macro-financiero limpio con variables seleccionadas antes del etiquetado.
MASTER_labeled_v2.csv	294	2001-02 a 2025-07	Dataset macro etiquetado con variables auxiliares de retorno, drawdown, score y etiqueta macro binaria.
NASDAQ_tecnico_features_v2.csv	300	2001-02 a 2026-01	Dataset de señales técnicas mensuales del NASDAQ100.
MASTER_tecnico_labeled_v1.csv	283	2002-01 a 2025-07	Dataset técnico final con señales técnicas, contexto macro histórico y target de agresividad DCA.

La diferencia entre los rangos de los datasets de features y los datasets etiquetados se debe al uso de horizontes futuros para construir las etiquetas. Cuando se calcula un retorno futuro o un drawdown futuro a seis meses, las últimas observaciones disponibles no pueden etiquetarse sin utilizar información posterior al final de la serie. Por ese motivo, algunos

ficheros de features llegan hasta enero de 2026, mientras que los datasets etiquetados finalizan en julio de 2025.

4.2.1. Datos financieros del NASDAQ100 y referencias de mercado

El NASDAQ100 constituye el activo de referencia sobre el que se evalúa la estrategia de aportaciones periódicas. En los datasets aparece representado mediante la variable `NASDAQ_level`, que actúa como serie central para calcular retornos, drawdowns, señales técnicas y valor de cartera en el backtesting.

Además del NASDAQ100, el bloque macro incorpora referencias de mercado como el S&P 500 y el Russell 2000. Estas series se utilizan principalmente para construir medidas relativas de amplitud o comportamiento comparado, como `NDX_SPX_ratio_yoy` y `RUT_SPX_ratio`. Estas variables no sustituyen al activo principal, sino que aportan información contextual sobre fortaleza o debilidad relativa frente a referencias de mercado más amplias.

En el bloque técnico se emplea un histórico extendido del NASDAQ100 para calcular indicadores que requieren ventanas temporales, como medias móviles, percentiles, RSI o drawdowns. Después de calcular estas variables, el dataset se recorta al rango temporal útil del sistema. Esta decisión evita perder un número excesivo de observaciones iniciales por el arranque de indicadores con ventanas de varios meses.

Las variables técnicas generadas se agrupan en cinco dimensiones: momentum, tendencia, riesgo, posición relativa y calidad retorno-riesgo. Entre ellas se encuentran retornos a uno, tres y seis meses, RSI de 14 periodos, ratios precio/media móvil, pendientes de medias móviles, drawdown a 12 meses, volatilidad a 3 meses, percentil a 24 meses, distancia al máximo de 24 meses y ratio retorno-volatilidad a 3 meses.

4.2.2. Datos macroeconómicos y financieros externos

El analista macroeconómico parte de series heterogéneas de carácter económico, financiero y de mercado. Los notebooks documentan fuentes procedentes de FRED para varias variables macroeconómicas y financieras estadounidenses, de la OCDE para el indicador líder compuesto del G20, de Investing.com para algunas referencias de mercado y de cálculos propios para ratios relativos entre índices.

La Tabla 4 resume las principales familias de información consideradas durante la construcción del bloque macroeconómico. La tabla no pretende listar las variables finales del modelo, sino describir las fuentes y dimensiones económicas exploradas durante la fase de extracción y construcción del dataset. La selección final de variables empleadas por el modelo macroeconómico se detalla posteriormente en el capítulo dedicado al analista macro.

Tabla 4: Familias de variables macroeconómicas y financieras consideradas en la construcción del bloque macro.

Familia	Ejemplos de series o variables	Información que aportan al sistema
Inflación	CPI y Core CPI	Presión de precios agregada y subyacente.

Familia	Ejemplos de series o variables	Información que aportan al sistema
Tipos y curva	DGS10, T10Y2Y, T10Y3M	Nivel de tipos y pendiente de la curva.
Crédito y condiciones financieras	High Yield OAS, NFCI	Estrés financiero, coste del crédito y condiciones agregadas.
Mercado laboral	Initial Claims, empleo no agrícola	Deterioro o fortaleza del empleo.
Actividad y confianza	Indicador líder G20, confianza del consumidor	Ciclo económico y expectativas.
Liquidez, volatilidad y divisas	M2 real, VXN, DXY	Liquidez real, volatilidad implícita y comportamiento del dólar.
Materias primas y amplitud	Petróleo, oro, ratios NDX/SPX y Russell/SPX	Señales de mercado complementarias y fortaleza relativa.

La extracción no se limita a almacenar niveles brutos. Los notebooks transforman las series en señales más informativas mediante variaciones interanuales, cambios mensuales, medias móviles, percentiles, dummies de estrés o medidas relativas. En la versión final del dataset macro se reduce el número de variables para mejorar cobertura histórica, disminuir redundancia y limitar el riesgo de sobreajuste asociado a una muestra mensual de tamaño reducido.

4.3. Justificación de la frecuencia mensual

La frecuencia mensual se adopta por coherencia con el objetivo del sistema. La estrategia DCA evaluada se basa en aportaciones periódicas, y el proyecto decide la agresividad de dichas aportaciones en términos mensuales. Por ello, el sistema no se plantea como una estrategia de trading intradía ni como un modelo de señales diarias.

La elección mensual también responde a la naturaleza de las fuentes macroeconómicas utilizadas. Muchas de las series relevantes se publican con frecuencia mensual o tienen una periodicidad que no justifica una actualización diaria de la decisión de aportación. Trabajar en frecuencia mensual permite integrar datos diarios, semanales y mensuales dentro de una misma estructura temporal, evitando comparar directamente observaciones con granularidades incompatibles.

Esta decisión metodológica tiene ventajas y limitaciones. Por un lado, reduce la complejidad operativa y mantiene la coherencia con una estrategia de inversión periódica. Por otro lado, limita el tamaño muestral: incluso con más de dos décadas de historia, el número de observaciones mensuales sigue siendo bajo para entrenar y validar modelos supervisados. Por este motivo, el proyecto combina frecuencia mensual con selección razonada de variables, validación temporal e interpretación prudente.

4.4. Arquitectura general del sistema

La arquitectura del sistema se estructura como un flujo jerárquico de decisión compuesto por tres módulos principales: análisis macroeconómico, análisis técnico y evaluación financiera. La Figura 1 sintetiza cómo las variables de entrada se transforman progresivamente en una decisión dinámica de aportación y, finalmente, en una simulación comparativa de resultados.

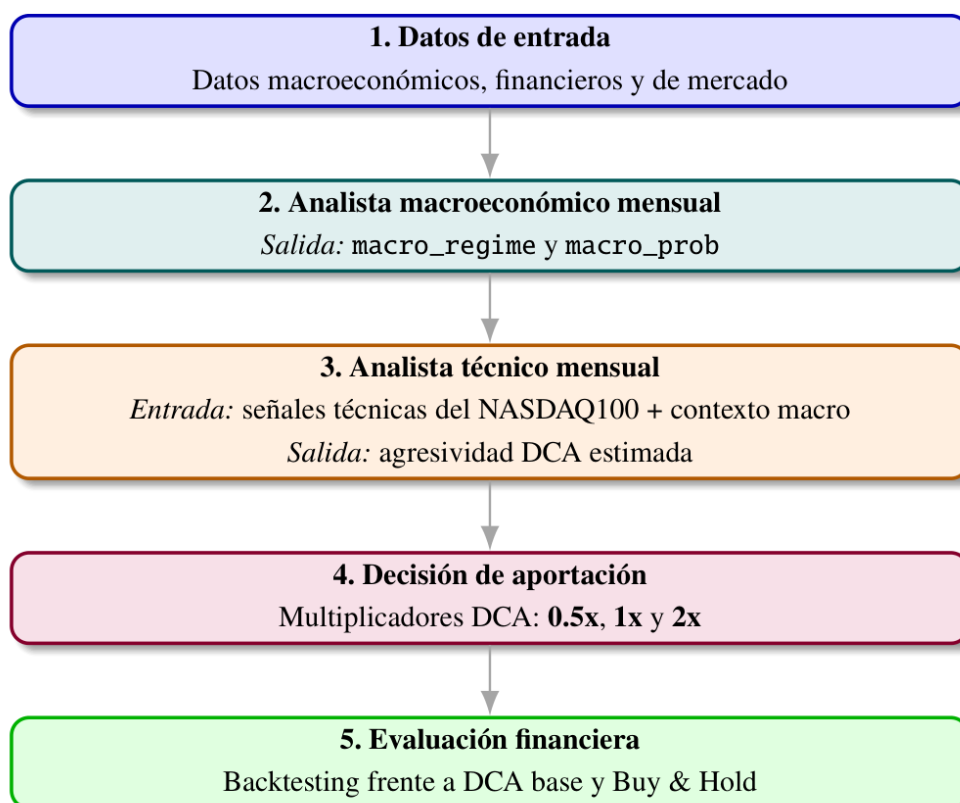


Figura 1: Arquitectura general del sistema de optimización dinámica de aportaciones DCA.

La arquitectura tiene carácter jerárquico. El analista macroeconómico no decide directamente cuánto aportar, sino que genera una señal de contexto. El analista técnico combina dicho contexto con señales del NASDAQ100 y estima una agresividad de aportación. Finalmente, el módulo financiero transforma esas decisiones en simulaciones de cartera y permite compararlas frente a estrategias de referencia.

4.4.1. Analista macroeconómico

El analista macroeconómico tiene como objetivo aportar una lectura del entorno macro-financiero. La versión final del proyecto reformula el problema macro como un filtro binario de entorno favorable, en lugar de mantener una clasificación multiclasa de regímenes. Esta

reformulación responde a la evolución metodológica del proyecto y a las limitaciones asociadas al tamaño de la muestra mensual.

El dataset macro final se construye a partir de variables seleccionadas y se etiqueta mediante información futura de retorno y drawdown a seis meses. En el entrenamiento final, el target seleccionado es `Y_favorable_bin_6m`. El modelo consolidado en los metadatos es un `random_forest`, entrenado con 19 features, incluyendo variables macro seleccionadas y codificación temporal del mes mediante seno y coseno.

La salida operativa más importante para el resto del sistema es `macro_model_final_history_predictions.csv`. Este fichero contiene `macro_regime` y `macro_prob`, que se incorporan al dataset técnico. De este modo, el analista técnico no utiliza la etiqueta macro real, sino el contexto generado por el bloque macro.

4.4.2. Analista técnico

El analista técnico transforma la evolución mensual del NASDAQ100 en variables relacionadas con momentum, tendencia, riesgo, posición relativa y calidad retorno-riesgo. Estas variables se combinan con `macro_regime` y `macro_prob`, generadas por el analista macroeconómico.

El target técnico se construye a partir de dos magnitudes futuras: retorno acumulado a seis meses y peor drawdown futuro dentro de esa ventana. A partir de ambas se calcula un score técnico definido como retorno futuro menos el valor absoluto del drawdown futuro. Posteriormente, el score se discretiza en tres niveles: aportación reducida 0.5x, aportación base 1x y aportación agresiva 2x. La etiqueta conserva así un significado económico vinculado con cuánto aportar, no únicamente con la dirección futura del mercado.

El bloque técnico compara tres formulaciones: clasificación multiclase como baseline, reformulación binaria centrada en detectar la clase agresiva y regresión del score técnico con discretización posterior. Esta comparación permite evaluar no solo algoritmos, sino también distintas definiciones del problema.

4.4.3. Evaluación financiera mediante backtesting

La evaluación financiera conecta las predicciones de los modelos con una simulación de aportaciones periódicas. Las salidas del analista técnico se convierten en multiplicadores de capital: 0.5x, 1x o 2x. Con estos multiplicadores se simula la compra mensual de participaciones del NASDAQ100.

El backtesting calcula capital aportado, participaciones compradas, valor de cartera, beneficio o pérdida, retorno sobre capital y drawdown. En las tablas finales del proyecto también se preparan campos para TIR anualizada, CAGR, volatilidad anual, Sharpe y eficiencia del capital extra, siempre condicionados por la información disponible en cada curva o estrategia. La interpretación detallada de estas métricas corresponde al capítulo de resultados.

Las estrategias se comparan frente a dos referencias. La primera es `dca_base`, que representa la comparación operativa más directa al compartir la lógica de aportaciones mensuales. La segunda es Buy & Hold, que sirve como benchmark de mercado, aunque no replica la misma mecánica de aportaciones periódicas.

4.5. Pipeline de trabajo y artefactos generados

El pipeline del proyecto está organizado en notebooks numerados. Esta estructura permite reconstruir el flujo completo desde la extracción de datos hasta la evaluación financiera final. La Tabla 5 resume los principales notebooks y artefactos generados; en una versión final maquetada, esta tabla también podría trasladarse a anexos si se desea aligerar el cuerpo principal de la memoria.

Tabla 5: Resumen del pipeline de notebooks y artefactos principales.

Notebook o bloque	Función metodológica	Artefactos principales
01_extract_features	Extracción y transformación de series macro-financieras a frecuencia mensual.	CSV de features por indicador.
02_build_master_dataset_v1/v2/v3	Construcción y evolución del dataset maestro macro.	MASTER_features_clean_v2.csv; MASTER_labeled_v2.csv.
03_train_macro_model_v1/v2	Entrenamiento del analista macro y generación del contexto histórico final.	Modelo macro final, metadatos, reports y predicciones históricas.
04_extract_features_tecnico	Cálculo de señales técnicas mensuales del NASDAQ100.	NASDAQ_tecnico_features_v2.csv.
05_build_dataset_tecnico	Integración de señales técnicas, contexto macro y target DCA.	MASTER_tecnico_labeled_v1.csv.
06_train_model_tecnico	Entrenamiento del baseline técnico multiclase.	Modelo técnico base, metadatos y reportes de evaluación.
07_experimentos_clasificacion_tecnico	Diagnóstico del problema técnico y evaluación de variantes binarias.	Reports de impacto macro, clasificación binaria y separabilidad.

Notebook o bloque	Función metodológica	Artefactos principales
08_regresion_score_tecnico	Modelado alternativo del score técnico como variable continua.	Reports de regresión y predicciones de test.
09_backtest_tecnico	Conversión de predicciones en estrategias DCA y simulación financiera.	Summaries y curvas de backtest limpio y walk-forward.
10_evaluacion_final_sistema	Consolidación final de modelos, estrategias, benchmarks y tablas limpias.	Tablas finales para la memoria y la defensa.

Los artefactos finales más relevantes para la memoria incluyen datasets maestros, modelos serializados, ficheros JSON de metadatos y tablas CSV consolidadas. En el cuerpo de la memoria se describen los elementos necesarios para comprender la metodología, mientras que la estructura simplificada del repositorio y los artefactos entregados se resumen en el Anexo A.

4.6. Estrategia de validación temporal

La validación del proyecto se diseña respetando el orden temporal de las observaciones. En series financieras, mezclar aleatoriamente datos pasados y futuros puede producir fugas de información y estimaciones optimistas. Por ello, el pipeline utiliza particiones cronológicas y validación walk-forward.

La regla general es que cada modelo debe entrenarse únicamente con información que habría estado disponible antes del periodo evaluado. Esta regla se aplica al bloque macro, al bloque técnico y al backtesting. La validación temporal no solo sirve para evaluar modelos, sino también para generar predicciones macro históricas que el analista técnico pueda consumir sin utilizar etiquetas futuras.

4.6.1. División temporal entre entrenamiento y test

La Tabla 6 resume las particiones temporales consolidadas en los artefactos finales.

Tabla 6: Particiones temporales utilizadas en los bloques principales.

Bloque	Entrenamiento	Test o evaluación final
Analista macroeconómico	2001-02 a 2023-12	2024-01 a 2025-01

Bloque	Entrenamiento	Test o evaluación final
Analista técnico multiclase	2002-01 a 2023-12	2024-01 a 2025-07
Backtest limpio	Modelos cerrados con historia previa a 2024	Simulación 2024-01 a 2025-07
Evaluación walk-forward	Ventanas expansivas anuales	Simulación histórica acumulada hasta 2025- 07

La diferencia entre el test macro y el test técnico se debe a la disponibilidad de etiquetas futuras y al horizonte de seis meses utilizado para construir algunos targets. Por ello, el periodo modelable no coincide exactamente en todos los bloques. Esta diferencia debe explicarse para evitar interpretar los datasets como si todos compartieran idéntico rango de entrenamiento y test.

4.6.2. Walk-forward validation

La validación walk-forward utiliza una lógica expansiva. Para cada bloque temporal, el modelo se entrena con la historia disponible hasta ese momento y se evalúa sobre el periodo siguiente. A continuación, la ventana de entrenamiento avanza incorporando nueva información pasada y el proceso se repite. Esta metodología respeta el orden cronológico de los datos y resulta más adecuada para series temporales que una partición aleatoria.

En el analista macro, el contexto histórico utilizado por el bloque técnico se genera de forma pseudo out-of-sample. Hasta 2023 se emplea entrenamiento expansivo anual, mientras que desde 2024 se utiliza el modelo final entrenado sobre el periodo de entrenamiento disponible. El objetivo es disponer de `macro_regime` y `macro_prob` históricos sin reconstruir el contexto macro dentro del notebook técnico ni utilizar etiquetas futuras como entrada.

En la evaluación financiera, el walk-forward anual permite comparar estrategias DCA entrenando con la información histórica previa, prediciendo el año correspondiente y aplicando la estrategia sobre ese periodo. Esta validación no garantiza que el sistema generalice en el futuro, pero ofrece una lectura más exigente que una única partición puntual.

4.7. Consideraciones metodológicas sobre leakage, sobreajuste y tamaño muestral

El primer riesgo metodológico es el *data leakage*. Este riesgo aparece cuando el modelo recibe, directa o indirectamente, información que no habría estado disponible en el momento de la predicción. En este proyecto es especialmente relevante porque los targets se construyen mediante retornos y drawdowns futuros. Para mitigarlo, las columnas auxiliares de etiquetado, como retornos futuros, drawdowns futuros, scores y targets, se excluyen de las matrices de entrenamiento. Además, las operaciones de imputación y escalado se integran en pipelines de `scikit-learn`, de forma que se ajustan sobre el conjunto de entrenamiento dentro de cada experimento.

Un segundo riesgo es introducir en el analista técnico información macroeconómica conocida solo retrospectivamente. Para evitarlo, el dataset técnico no utiliza la etiqueta macro real como predictor. En su lugar, consume `macro_regime` y `macro_prob`, que proceden del analista macro. Esta decisión mantiene la lógica operativa de la arquitectura: el bloque técnico utiliza una predicción o señal generada por el bloque macro, no una etiqueta futura perfecta.

El tercer riesgo es el sobreajuste. La muestra mensual contiene pocas centenas de observaciones, por lo que un número excesivo de variables o una formulación demasiado ambiciosa pueden producir modelos inestables. El proyecto responde a este problema mediante selección razonada de variables macro, comparación de formulaciones alternativas, validación temporal y lectura prudente.

También debe considerarse la naturaleza del target técnico. Muchas observaciones pueden situarse cerca de los umbrales que separan los multiplicadores 0.5x, 1x y 2x. Esto implica que pequeñas variaciones del score pueden cambiar la clase asignada, por lo que el problema multiclase no debe interpretarse como una separación limpia y perfectamente estable.

En conjunto, estas decisiones metodológicas justifican presentar el sistema como un modulador experimental y disciplinado de aportaciones DCA. El valor del proyecto no reside en garantizar que una estrategia bata siempre al mercado, sino en construir un pipeline trazable, temporalmente coherente y evaluable con criterios predictivos y financieros.

Capítulo 5. Desarrollo y resultados del analista macroeconómico

Este capítulo desarrolla el bloque macroeconómico del sistema macro-técnico propuesto en el Trabajo Fin de Máster. A diferencia del capítulo metodológico anterior, aquí se describe con mayor detalle cómo se construye el analista macro, cómo evoluciona su formulación durante el proyecto, qué modelo final se selecciona y qué resultados produce. El objetivo no es presentar el analista macroeconómico como un predictor perfecto de mercado, sino como una capa de contexto que transforma información macroeconómica y financiera mensual en una señal utilizable por el analista técnico.

La lectura correcta del capítulo debe partir de una idea central: el bloque macro no decide por sí solo cuánto aportar en la estrategia DCA. Su función es estimar si el entorno macro-financiero puede considerarse favorable o no favorable según la definición operativa aprendida a partir de los datos. Esa estimación se materializa posteriormente en dos variables, `macro_regime` y `macro_prob`, que el bloque técnico incorpora junto con señales propias del NASDAQ100. La decisión final de agresividad de aportación y su evaluación económica se estudian en los capítulos posteriores, al integrar el contexto macro con el modelo técnico y con el backtesting.

5.1. Función del analista macroeconómico dentro del sistema

El sistema desarrollado se organiza como una arquitectura jerárquica. En primer lugar, el analista macroeconómico procesa información económica, financiera y de mercado para producir una lectura mensual del entorno. En segundo lugar, el analista técnico utiliza esa lectura, junto con indicadores propios del NASDAQ100, para modelar la agresividad de aportación en una estrategia de Dollar Cost Averaging. Finalmente, el módulo de evaluación financiera convierte las salidas predictivas en reglas de aportación y compara las estrategias resultantes frente a referencias pasivas.

Dentro de esa arquitectura, el analista macro cumple una función de filtro de contexto. No pretende resolver directamente la decisión de inversión mensual ni sustituir al analista técnico. Su salida resume si, con la información disponible hasta cada fecha, el entorno macro-financiero se parece más a un escenario favorable o al resto de escenarios. Esta salida se expresa mediante `macro_regime`, que representa la clase predicha, y `macro_prob`, que recoge la probabilidad estimada asociada al entorno favorable.

Esta separación de responsabilidades es importante por dos motivos. Primero, evita que el bloque macro tenga que resolver un problema excesivamente amplio, como predecir de forma completa el ciclo económico o anticipar todos los movimientos del mercado. Segundo, permite que el bloque técnico consuma una señal macro ya generada de forma coherente, sin utilizar la etiqueta macro real como si fuera información disponible en el momento de la decisión. De este modo, el sistema mantiene una lógica operativa más realista: el analista técnico no recibe una verdad futura, sino una predicción o señal generada por el analista macro.

La función del bloque macro también condiciona la interpretación de sus resultados. Una métrica aceptable en el modelo macro no implica, por sí sola, que el sistema completo vaya a mejorar la rentabilidad o el perfil de riesgo frente a una estrategia pasiva. La utilidad financiera de la señal macro solo puede valorarse después, cuando se integra con el analista técnico y se somete a simulaciones de backtesting. Por tanto, en este capítulo los resultados se interpretan como evidencia sobre la coherencia predictiva del filtro macro, no como prueba final de superioridad económica.

5.2. Construcción del dataset macro-financiero

El analista macroeconómico se construye a partir de un dataset mensual que integra series macroeconómicas, financieras y de mercado. La construcción del dataset se realiza en dos fases principales. Primero, las series originales se transforman en variables mensuales comparables. Después, esas variables se fusionan por fecha, se depuran, se seleccionan y se etiquetan para formular un problema supervisado.

La fase de extracción de variables no entrena ningún modelo. Su función es traducir fuentes heterogéneas a un lenguaje tabular común. Las series originales tienen frecuencias, escalas y significados distintos: algunas son mensuales, otras proceden de mercados financieros, otras reflejan indicadores de empleo, inflación, crédito o confianza. Para que puedan alimentar un clasificador, se normalizan a frecuencia mensual y se transforman en señales con interpretación económica.

La versión madura del dataset macro se materializa en dos artefactos principales. El primero es `MASTER_features_clean_v2.csv`, que contiene las variables limpias y seleccionadas antes del etiquetado. Este fichero tiene 300 observaciones y cubre el periodo comprendido entre febrero de 2001 y enero de 2026. El segundo es `MASTER_labeled_v2.csv`, que añade las columnas auxiliares necesarias para construir la etiqueta macro y contiene 294 observaciones entre febrero de 2001 y julio de 2025. La diferencia temporal se debe al uso de un horizonte futuro de seis meses para calcular retornos y drawdowns usados en el etiquetado.

El dataset etiquetado incluye, además de las variables predictoras, las columnas auxiliares `R_t_6`, `DD_t_6`, `Score_t` y `Y_macro_bin`. La variable `R_t_6` representa el retorno acumulado del NASDAQ100 durante los seis meses posteriores a cada observación t , mientras que `DD_t_6` recoge el peor drawdown observado dentro de ese mismo horizonte temporal. A partir de ambas magnitudes se construye `Score_t`, una métrica sintética diseñada para combinar rentabilidad futura y riesgo de caída en una única puntuación. Finalmente, `Y_macro_bin` corresponde a la etiqueta macro binaria incluida en este artefacto, donde los periodos considerados favorables reciben la etiqueta 1 y el resto la etiqueta 0. Esta formulación se interpreta como una etiqueta macro inicial o comparable dentro del bloque macro; la selección posterior del modelo final se discute en la comparación de targets. Estas columnas se generan como variables auxiliares de etiquetado o como targets del aprendizaje supervisado, ya que incorporan información futura que no estaría disponible en el momento de realizar una predicción real. Por este motivo, no deben confundirse con variables de entrada del modelo.

Tabla 7: Artefactos principales del dataset macro-financiero.

Artefacto	Filas	Periodo	Función dentro del bloque macro
MASTER_features_clean_v2.csv	300	2001-02 a 2026-01	Dataset macro-financiero limpio con variables seleccionadas antes de construir etiquetas.
MASTER_labeled_v2.csv	294	2001-02 a 2025-07	Dataset etiquetado con columnas auxiliares de retorno, drawdown, score y etiqueta macro.
macro_model_final_history_predictions.csv	300	2001-02 a 2026-01	Salida histórica del analista macro, con macro_regime, macro_prob, fuente de predicción y metadatos.

5.2.1. Familias de variables utilizadas

Las variables utilizadas por el bloque macro cubren varias dimensiones del entorno económico y financiero. El objetivo no es incluir toda la información disponible, sino representar familias que puedan aportar señales complementarias sobre inflación, tipos de interés, condiciones financieras, empleo, actividad, liquidez, volatilidad y comportamiento relativo de mercado.

Una decisión relevante del proceso de extracción es que las series no se incorporan únicamente en nivel bruto. En muchos casos se calculan variaciones interanuales, cambios mensuales, medias móviles, percentiles, dummies de estrés o ratios relativos. Esta transformación es importante porque una serie económica en nivel puede ser menos informativa que su variación, su tendencia o su posición relativa frente a la historia reciente.

Por ejemplo, una inflación interanual es más interpretable para el modelo que el índice de precios en bruto, y la pendiente de la curva o la persistencia de una inversión de curva aportan información distinta al simple nivel de un tipo de interés.

La Tabla 8 resume las familias representadas en la versión final del modelo macro. La tabla se centra en las variables finalmente utilizadas por el modelo seleccionado, no en todas las columnas generadas durante las fases intermedias de extracción.

Tabla 8: Familias de variables usadas en el modelo macro final.

Familia	Variables finales asociadas	Información representada
Inflación	CPI_yoy_pct, CoreCPI_yoy_pct	Presión de precios agregada y subyacente.
Tipos y curva	DGS10_mom_change, T10Y2Y_level	Movimiento del bono a 10 años y pendiente de curva.
Crédito y condiciones financieras	HY_OAS_level, NFCI_level	Estrés de crédito y condiciones financieras agregadas.
Mercado laboral	ICSA_yoy_pct, NFP_yoy_pct	Deterioro o fortaleza del empleo.
Actividad y confianza	G20_CLI_mom_change, US_Conf_yoy	Ciclo económico, indicadores adelantados y sentimiento.
Liquidez, volatilidad y divisas	M2_real_yoy_pct, VXN_level, DXY_yoy	Liquidez real, volatilidad implícita y comportamiento del dólar.
Materias primas y amplitud	Oil_yoy, Gold_return_3m, NDX_SPX_ratio_yoy, RUT_SPX_ratio	Presiones externas, refugio relativo y fortaleza comparada de índices.
Estacionalidad mensual	Mes_sin, Mes_cos	Codificación cíclica del mes del año.

La presencia de NASDAQ_level en el dataset maestro cumple una función importante para calcular retornos futuros, drawdowns y variables auxiliares de etiquetado. Sin embargo, la lista final de 19 features guardada para el modelo macro no incluye NASDAQ_level como

predictor final. Esta distinción evita confundir la serie de referencia utilizada para construir etiquetas con las variables efectivamente suministradas al clasificador final.

5.2.2. Unificación temporal, limpieza y selección de variables

La integración del dataset macro se realiza mediante la fusión mensual de los bloques de features por fecha. Esta unificación temporal es necesaria porque las series originales no comparten necesariamente la misma frecuencia ni el mismo inicio histórico. Al cruzar indicadores con coberturas diferentes, el rango final utilizable puede reducirse de forma considerable. Por ello, la versión madura del dataset prioriza una selección más contenida de variables con mejor cobertura histórica, en lugar de conservar todas las columnas generadas inicialmente.

La limpieza se aplica después de decidir qué variables merecen mantenerse. Esta decisión metodológica es relevante: si primero se eliminaran observaciones por nulos considerando todas las columnas disponibles, variables marginales con poco histórico podrían obligar a perder años enteros de muestra. En cambio, filtrar primero las variables permite conservar más historia útil y reducir el ruido antes del entrenamiento.

La selección de variables no debe interpretarse como una tarea meramente cosmética. En una muestra mensual con pocas centenas de observaciones, el número de variables condiciona de forma directa la estabilidad del aprendizaje. Demasiadas features pueden introducir redundancia, facilitar el sobreajuste y empeorar la interpretación. Por ese motivo, la versión final del dataset macro reduce la complejidad y conserva una representación razonada de las principales familias macro-financieras.

El resultado de esta fase es un dataset suficientemente compacto para ser defendible en el contexto del TFM. La formulación final no elimina los riesgos propios de una muestra temporal pequeña, pero sí mejora la coherencia metodológica respecto a las primeras versiones, que intentaban resolver un problema más ambicioso con más variables y menos control sobre la cobertura histórica.

5.3. Evolución metodológica del bloque macro

El bloque macro no nace directamente con su formulación final. A lo largo del proyecto se documentan varias versiones que muestran una evolución metodológica clara: primero se formula un problema multiclase más ambicioso; después se detectan problemas de desbalance, tamaño muestral y métricas engañosas; finalmente se reformula el objetivo como un filtro binario de entorno favorable.

Esta evolución es importante para entender el resultado final. La versión definitiva no es simplemente una reducción arbitraria del problema, sino una respuesta a las limitaciones observadas durante el desarrollo. El proyecto aprende que, con datos mensuales y una muestra limitada, clasificar muchos regímenes distintos puede ser menos defendible que identificar una señal binaria más alineada con el papel real del analista macro dentro del sistema.

Tabla 9: Evolución metodológica del bloque macroeconómico.

Etapas y notebook principal	Planteamiento	Aprendizaje metodológico
Construcción inicial del dataset	Construcción de un dataset maestro amplio y formulación inicial de variables y etiquetas macro.	Sirve como arranque del pipeline, pero mantiene una estructura todavía demasiado amplia y poco alineada con la arquitectura final.
Entrenamiento multiclase inicial	Clasificación multiclase mediante <code>Y_macro</code> , con 56 features y partición temporal clásica.	La accuracy aislada resulta poco informativa; las métricas equilibradas muestran debilidad por desbalance y dificultad para separar regímenes.
Transición a etiqueta binaria	Reformulación del dataset hacia una etiqueta binaria de entorno favorable frente al resto y ampliación de historia útil.	El objetivo empieza a ajustarse mejor a la función del analista macro como filtro de contexto, no como clasificador completo de ciclos.
Dataset macro maduro	Selección manual razonada de variables, corrección de nomenclatura y horizonte coherente de seis meses.	La reducción de dimensionalidad mejora la cobertura histórica, reduce redundancias y hace el experimento más defendible con una muestra mensual limitada.
Cierre del modelo macro	Comparación de targets binarios y modelos candidatos mediante validación temporal y test externo.	Se selecciona <code>Y_favorable_bin_6m</code> con <code>random_forest</code> y se exporta el contexto macro histórico usado por el bloque técnico.

5.3.1. Formulación inicial multiclase

La primera formulación del analista macro planteaba un problema multiclase mediante el target `Y_macro`. Esta versión intentaba distinguir varios regímenes de mercado combinando

retorno futuro y drawdown futuro. La idea económica de fondo era razonable: no basta con saber si el mercado termina subiendo o bajando, también importa el riesgo asumido durante el camino.

En esta etapa inicial, el modelo trabajaba con 56 features y utilizaba un split temporal clásico dividido en entrenamiento, validación y test. El conjunto de entrenamiento cubría de enero de 2004 a diciembre de 2018, la validación de enero de 2019 a diciembre de 2022 y el test de enero de 2023 a abril de 2025. Los modelos comparados eran una regresión logística multinomial y un random forest, y la evaluación no se limitaba a accuracy, sino que también incluía balanced accuracy y F1 macro.

El mejor modelo de esta versión fue un `random_forest`. En el test final de esa etapa obtuvo una accuracy de 0.6786, una balanced accuracy de 0.3016 y un F1 macro de 0.2695. La diferencia entre estas métricas es la clave para interpretar correctamente el experimento. La accuracy parecía aceptable en términos superficiales, pero las métricas equilibradas eran bajas. Esto indicaba que el modelo probablemente estaba acertando con mayor frecuencia la clase dominante, pero no distinguía adecuadamente todas las clases del problema multiclase.

Por tanto, la formulación inicial no debe presentarse como la versión final del sistema, sino como una fase exploratoria que permitió diagnosticar el problema. Su valor metodológico reside precisamente en mostrar que el pipeline podía entrenar y evaluar modelos respetando el orden temporal, pero que la formulación del objetivo era demasiado exigente para el tamaño y la estructura de la muestra disponible.

5.3.2. Limitaciones detectadas

La principal limitación detectada fue el tamaño muestral. Al trabajar con frecuencia mensual, incluso un histórico de más de dos décadas produce solo unas pocas centenas de observaciones. Esta restricción es especialmente importante cuando se pretende entrenar un clasificador multiclase con decenas de variables. La relación entre número de observaciones, número de features y número de clases puede volverse desfavorable y aumentar el riesgo de modelos inestables.

La segunda limitación fue el desbalance de clases. En un problema multiclase de regímenes macro-financieros, algunas situaciones aparecen con mucha más frecuencia que otras. Si se evalúa únicamente con accuracy, el modelo puede parecer razonable aunque esté aprendiendo sobre todo a reconocer la clase mayoritaria. Por este motivo, las métricas equilibradas fueron decisivas: la balanced accuracy y el F1 macro mostraron que el rendimiento real era más débil de lo que sugería la accuracy simple.

La tercera limitación fue conceptual. Pedir al analista macro que identifique cuatro regímenes distintos exige una granularidad que quizá no es necesaria para la arquitectura final. El sistema no necesita que el bloque macro describa con precisión todo el ciclo económico, sino que aporte una señal útil de contexto para el analista técnico. Desde esta perspectiva, una formulación binaria puede ser más coherente con la función operativa del bloque.

La cuarta limitación se relaciona con la validación temporal. La primera versión ya respetaba un split cronológico, pero el desarrollo posterior incorporó una validación walk-forward anual más adecuada para series temporales. En un contexto con pocas observaciones, una única fotografía de test puede depender mucho del periodo elegido. El walk-forward permite

observar el comportamiento del modelo a través de varios bloques temporales y reduce el riesgo de interpretar de forma excesiva un único resultado puntual.

Estas limitaciones no invalidan la fase inicial. Al contrario, justifican la transición metodológica posterior. El abandono de la clasificación multiclase no se debe a una preferencia arbitraria por modelos más simples, sino a la evidencia de que el problema inicial era demasiado ambicioso para la muestra disponible y para el papel que el analista macro debía cumplir dentro del sistema completo.

5.3.3. Reformulación binaria como filtro de entorno favorable

La reformulación del bloque macro convierte el problema en una clasificación binaria: entorno favorable frente al resto. Esta decisión alinea el objetivo del modelo con su función real dentro del sistema. En lugar de clasificar todos los matices posibles del ciclo macro-financiero, el analista macro intenta identificar meses en los que el entorno posterior a seis meses fue favorable y presentó un drawdown contenido, de acuerdo con la definición operativa construida en el notebook final.

El entrenamiento final no se limita a aceptar automáticamente la etiqueta binaria inicial `Y_macro_bin`. El notebook de cierre compara tres alternativas: `Y_macro_bin`, `Y_favorable_bin_6m` y `Y_risk_bin_6m`. Las tres representan formas distintas de convertir el comportamiento futuro del mercado en una etiqueta binaria. La comparación de targets es una decisión metodológica relevante porque, en aprendizaje supervisado, elegir qué se quiere predecir puede ser tan importante como elegir el algoritmo.

El target final seleccionado es `Y_favorable_bin_6m`. Su definición registrada en los metadatos es: 1 = entorno favorable robusto a 6m ; 0 = resto. Esta elección es coherente con el papel del bloque macro como filtro de contexto favorable para el sistema técnico. Además, según el artefacto de comparación de targets, esta formulación obtiene la mejor *balanced accuracy* de test entre las alternativas comparadas, aunque no domina todas las métricas posibles.

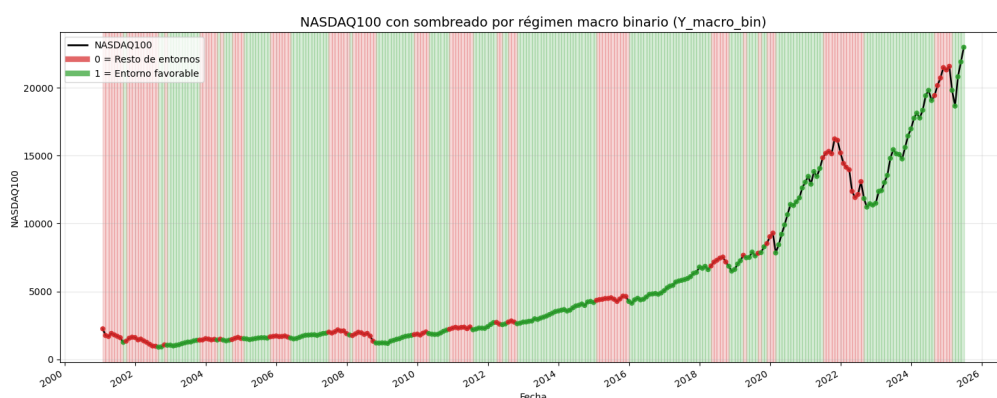


Figura 2: Representación del NASDAQ100 junto con el sombreado binario de entorno macro utilizado durante la fase de comparación de etiquetas. La figura permite comprobar visualmente cómo una señal macro binaria se distribuye sobre la evolución histórica del índice, sin sustituir a la evaluación cuantitativa de targets presentada en la tabla posterior.

Tabla 10: Comparación de targets macro binarios.

Target	Mejor modelo	CV BA	CV ROC	Test BA	F1	Test ROC
Y_favorable_bin_6m	random_forest	0.5674	0.6241	0.5714	0.6667	0.8571
Y_risk_bin_6m	extra_trees	0.7231	0.7622	0.5000	0.0000	1.0000
Y_macro_bin	random_forest	0.6302	0.5895	0.5000	0.7619	0.8250

La Tabla 10 debe interpretarse con cautela. El target de riesgo presenta valores altos en validación cruzada y un ROC AUC de test igual a 1.0000, pero el test contiene una proporción positiva muy baja para esa definición y su F1 de test es 0.0000. Por tanto, ese resultado no debe leerse como una superioridad robusta. Del mismo modo, Y_macro_bin obtiene un F1 más alto en test, pero su balanced accuracy de test es 0.5000. La selección de Y_favorable_bin_6m se entiende mejor combinando dos criterios: su coherencia funcional como filtro de entorno favorable y la regla de selección implementada en los metadatos, basada en balanced accuracy de test, ROC AUC de test y balanced accuracy media en validación.

5.4. Entrenamiento, validación y selección del modelo macro final

El entrenamiento final del bloque macro parte de MASTER_labeled_v2.csv y utiliza el target Y_favorable_bin_6m. La matriz de entrada se construye con 19 features finales: 17 variables macro-financieras seleccionadas y dos variables de estacionalidad mensual, Mes_sin y Mes_cos. Quedan excluidas las columnas de fecha, las variables auxiliares de etiquetado, los retornos futuros, los drawdowns futuros, los scores y cualquier target. Esta exclusión es imprescindible para evitar fuga de información, ya que esas columnas solo se conocen después de observar la ventana futura usada para etiquetar.

La partición externa registrada en los metadatos utiliza entrenamiento desde febrero de 2001 hasta diciembre de 2023 y test desde enero de 2024 hasta enero de 2025. Dentro del tramo de entrenamiento se aplica validación walk-forward anual, siguiendo la lógica temporal descrita en el Capítulo 4: cada evaluación usa únicamente historia anterior al periodo evaluado.

Los modelos candidatos comparados en el cierre final son logreg, random_forest y extra_trees. La evaluación considera accuracy, balanced accuracy, F1, precision, recall y ROC AUC cuando resulta aplicable. La versión final seleccionada es random_forest, guardada como macro_model_final.joblib. La lista de variables se conserva en macro_model_final_features.json y los metadatos completos en macro_model_final_metadata.json.

Tabla 11: Configuración del entrenamiento macro final.

Elemento	Configuración registrada
Dataset de entrada	MASTER_labeled_v2.csv.
Target final	Y_favorable_bin_6m.
Definición del target	1 = entorno favorable robusto a 6m ; 0 = resto.
Número de features	19.
Modelos candidatos	logreg, random_forest, extra_trees.
Modelo seleccionado	random_forest.
Entrenamiento externo	2001-02 a 2023-12.
Test externo	2024-01 a 2025-01.
Validación interna	Walk-forward anual dentro del tramo de entrenamiento.
Criterio de ordenación experimental registrado	test_bal_acc_then_test_roc_auc_then_cv_bal_acc_mean.
Artefactos finales	macro_model_final.joblib, macro_model_final_features.json, macro_model_final_metadata.json.

La elección de random_forest debe entenderse como la selección final dentro del diseño experimental implementado: modelos candidatos evaluados, target definido, criterio de ordenación registrado y restricciones del dataset disponible. Los metadatos no documentan una búsqueda exhaustiva de hiperparámetros ni una comparación con todos los clasificadores posibles, por lo que la conclusión correcta es que random_forest fue el modelo seleccionado por el pipeline del proyecto para Y_favorable_bin_6m, no una demostración de superioridad universal.

Además de entrenar el modelo final, el bloque macro genera macro_model_final_history_predictions.csv. Este fichero contiene macro_regime, macro_prob, la fuente de predicción histórica y el número de filas usadas para entrenar en cada caso. Hasta 2023, el contexto histórico se genera mediante una lógica expansiva anual, de forma que cada predicción utiliza únicamente información anterior al periodo evaluado. Desde 2024, se utiliza el modelo final entrenado sobre el tramo de entrenamiento disponible. Esta salida actúa como la conexión operativa entre el analista macro y el analista técnico, ya que transforma el análisis macroeconómico en dos variables reutilizables por el módulo posterior: una etiqueta de régimen y una probabilidad asociada.

5.5. Resultados del analista macroeconómico

Los resultados finales del analista macro se resumen en tres niveles: validación walk-forward, test externo y predicciones históricas exportadas para el bloque técnico. La validación walk-forward constituye la referencia principal para valorar la estabilidad temporal del modelo, ya que evalúa su comportamiento en múltiples ventanas respetando el orden cronológico de la serie. En esta validación temporal, el modelo final random_forest para Y_favorable_bin_6m registra 11 folds, de los cuales 10 tienen ROC AUC válido. La accuracy media es 0.5606, la balanced accuracy media 0.5674, el F1 medio 0.4851, la precision media 0.6826, el recall medio 0.4464 y el ROC AUC medio 0.6241.

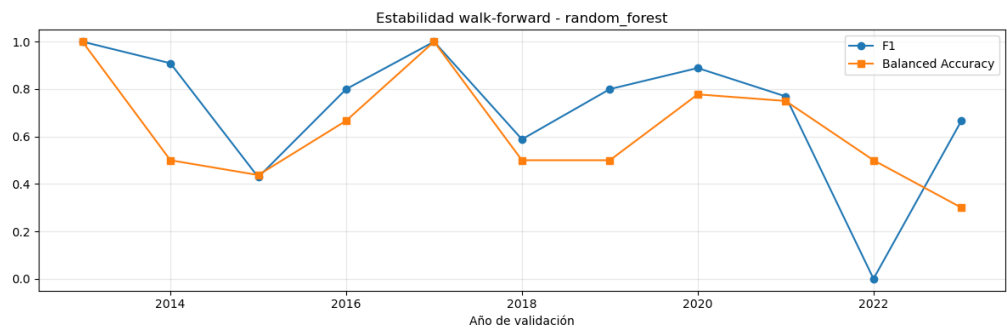


Figura 3: Evolución de F1 y balanced accuracy del modelo macroeconómico en validación walk-forward. La figura complementa las métricas agregadas mostrando la variabilidad temporal del rendimiento entre folds.

Estas métricas walk-forward apuntan a una señal moderada: el modelo muestra cierta capacidad de ordenación y una balanced accuracy media superior a la referencia aleatoria, pero no alcanza un rendimiento suficientemente alto como para considerarse un predictor macroeconómico robusto de forma aislada. Por tanto, su utilidad debe interpretarse como una capa de contexto probabilística para el sistema completo, no como una regla autónoma de inversión.

El test externo se conserva como comprobación final fuera de muestra. Este tramo contiene 13 observaciones entre enero de 2024 y enero de 2025, por lo que sus métricas deben interpretarse con especial cautela. En dicho periodo, el modelo obtiene una accuracy de 0.5385, balanced accuracy de 0.5714, F1 de 0.6667, precision de 0.5000, recall de 1.0000 y ROC AUC de 0.8571. El resultado indica que el modelo captura todos los meses etiquetados como favorables en el test, aunque también clasifica como favorables varios meses que realmente pertenecen a la clase no favorable.

Tabla 12: Resultados principales del modelo macro final.

Métrica o elemento	Valor	Interpretación prudente
Target final	Y_favorable_bin_6m	Entorno favorable robusto a 6 meses frente al resto.
Modelo final	random_forest	Modelo seleccionado

Métrica o elemento	Valor	Interpretación prudente
		por la regla implementada.
Features finales	19	Variables macro-financieras seleccionadas y estacionalidad mensual.
CV balanced accuracy media	0.5674	Señal moderada, no concluyente.
CV ROC AUC media	0.6241	Capacidad de ordenación limitada pero superior a aleatoriedad en media.
Test balanced accuracy	0.5714	Resultado ligeramente superior a 0.5, condicionado por 13 observaciones.
Test F1	0.6667	Debe leerse junto con precision y recall.
Test precision	0.5000	La mitad de las predicciones favorables del test fueron favorables reales.
Test recall	1.0000	Detecta todos los favorables del test, pero no implica robustez general.
Test ROC AUC	0.8571	Valor alto, pero muy sensible al tamaño reducido del test.

La matriz de confusión del test ayuda a interpretar mejor estas cifras. El test contiene 6 observaciones positivas y 7 negativas. El modelo predice 12 meses como favorables y solo 1 como no favorable. En consecuencia, obtiene 6 verdaderos positivos, 1 verdadero negativo, 6 falsos positivos y 0 falsos negativos.

Tabla 13: Matriz de confusión del test macro final.

	Predicho favorable	Predicho no favorable
Real favorable	6	0
Real no favorable	6	1

Esta matriz confirma que el buen recall procede de no dejar escapar meses favorables en el pequeño test disponible. Sin embargo, también muestra una capacidad limitada para descartar meses no favorables bajo el umbral de clasificación utilizado. Por tanto, el resultado no debe resumirse diciendo que el modelo “acierta” el régimen macro, sino que produce una señal favorable con cierta capacidad de ordenación, pero con muchos falsos positivos en el tramo de test.

El fichero de predicciones históricas cumple una función distinta a la evaluación de métricas. Su objetivo es proporcionar al bloque técnico una serie mensual de contexto macro. De las 300 filas de `macro_model_final_history_predictions.csv`, 11 observaciones iniciales no tienen predicción por falta de historia previa suficiente, 110 aparecen con `macro_regime = 1` y 179 con `macro_regime = 0`. Esta salida no debe interpretarse como una etiqueta macro verdadera, sino como el contexto estimado por el modelo para ser consumido posteriormente por el analista técnico.

5.6. Interpretación y limitaciones del bloque macro

La interpretación global del analista macro debe ser prudente. El bloque final es metodológicamente más coherente que la formulación inicial porque reduce el número de variables, abandona una clasificación multiclase demasiado ambiciosa, compara targets alternativos, aplica validación temporal y exporta una señal histórica utilizable por el bloque técnico. Su función, no obstante, sigue siendo la de filtro de contexto dentro del sistema completo.

La primera limitación es el tamaño muestral. Aunque el periodo histórico cubre más de dos décadas, la frecuencia mensual genera pocas observaciones efectivas. Esto afecta tanto al entrenamiento como a la evaluación. Para mitigar la dependencia de una única partición, la valoración de estabilidad del modelo se apoya en validación walk-forward anual dentro del tramo de entrenamiento, lo que permite observar su comportamiento en múltiples ventanas temporales. Aun así, el test final fuera de muestra contiene solo 13 meses, por lo que sus métricas deben interpretarse como una comprobación adicional y no como una evidencia estadística concluyente por sí sola.

La segunda limitación es la sensibilidad a la definición de la etiqueta. El target `Y_favorable_bin_6m` está construido a partir de retorno futuro y drawdown futuro a seis meses. Esta formulación es defendible porque conecta la señal macro con un entorno favorable para el mercado, pero no representa una verdad económica universal. Otros umbrales, horizontes o combinaciones de retorno y riesgo podrían producir etiquetas distintas y, por tanto, modelos distintos.

La tercera limitación es el desbalance y la lectura de métricas. En el test final, el recall de 1.0000 puede parecer muy positivo, pero la matriz de confusión muestra que el modelo predice la clase favorable en 12 de 13 observaciones. Esto genera falsos positivos y reduce la capacidad de discriminar meses no favorables. Por ese motivo, las métricas deben

analizarse conjuntamente: ROC AUC, balanced accuracy, F1, precision, recall y matriz de confusión aportan lecturas complementarias y ninguna de ellas debe usarse de forma aislada.

La cuarta limitación es que la selección del modelo está condicionada por el diseño experimental implementado. El `random_forest` final es el modelo seleccionado entre los candidatos evaluados y bajo la regla registrada en los metadatos. No se documenta en los artefactos finales una búsqueda exhaustiva de arquitecturas o hiperparámetros. Por tanto, la conclusión debe formularse en términos de coherencia del pipeline y no como superioridad estadística definitiva.

La quinta limitación se relaciona con la función económica del bloque. El analista macro no decide la aportación mensual y no demuestra por sí solo una mejora de rentabilidad. Su utilidad práctica depende de si `macro_regime` y `macro_prob` ayudan al analista técnico a modular mejor la estrategia DCA y de si esa modulación supera las referencias en el backtesting. Esa evaluación corresponde a los capítulos posteriores.

En conclusión, el analista macroeconómico final representa una capa de contexto razonable y trazable dentro del sistema. Su desarrollo muestra una evolución metodológica madura: partir de una formulación multiclase ambiciosa, diagnosticar sus limitaciones, simplificar el objetivo hacia un filtro binario de entorno favorable y cerrar el bloque con una señal macro-financiera temporalmente validada y preparada para integrarse con el resto del sistema.

Capítulo 6. Desarrollo y resultados del analista técnico

Este capítulo desarrolla el segundo bloque predictivo del sistema: el analista técnico. Su función es transformar señales mensuales del NASDAQ100, junto con el contexto macroeconómico generado en el capítulo anterior, en una estimación de agresividad de aportación para una estrategia DCA. La lógica del bloque técnico no consiste en predecir de forma exacta el nivel futuro del índice ni en anticipar todos sus movimientos, sino en aproximar qué multiplicador de aportación habría sido razonable en cada mes dado el equilibrio posterior entre retorno y riesgo.

El bloque técnico se apoya en tres elementos. En primer lugar, utiliza variables técnicas calculadas sobre el NASDAQ100, agrupadas en familias de momentum, tendencia, volatilidad, drawdown, posición relativa y calidad retorno-riesgo. En segundo lugar, incorpora el contexto macroeconómico procedente del analista macro mediante `macro_regime` y `macro_prob`. En tercer lugar, construye un target de agresividad DCA a partir del retorno futuro, el drawdown futuro y un score técnico ajustado por riesgo. A partir de esta información se prueban varias formulaciones: una clasificación multiclase directa, una reformulación binaria orientada a detectar aportaciones agresivas y una regresión del score técnico como variable continua.

6.1. Función del analista técnico dentro de la estrategia

DCA

El analista técnico ocupa la segunda capa del sistema macro-técnico. Mientras que el analista macroeconómico proporciona una lectura mensual del entorno, el analista técnico se centra en el comportamiento del NASDAQ100 y en cómo ese comportamiento puede traducirse en distintos niveles de aportación DCA. Su objetivo operativo es generar una decisión de agresividad, no una predicción puntual del precio futuro del índice.

La decisión se formula en términos de multiplicadores de aportación. El sistema distingue entre una aportación reducida, una aportación base y una aportación agresiva. En el dataset técnico estas decisiones se codifican como $Y_{tech} = 0.5$, $Y_{tech} = 1$ y $Y_{tech} = 2$. Para facilitar el entrenamiento de modelos estándar, también se crea una versión ordinal llamada `Y_tech_ord`, donde las clases 0, 1 y 2 representan respectivamente los multiplicadores $0.5x$, $1x$ y $2x$.

Esta formulación permite conectar el problema predictivo con la estrategia de inversión. En lugar de preguntar si el NASDAQ100 subirá o bajará, el bloque técnico pregunta qué nivel de aportación habría sido razonable dado el equilibrio entre retorno futuro y riesgo futuro. Esta diferencia es clave: el analista técnico no pretende ser un predictor exacto del mercado, sino un componente que ayuda a modular la cantidad aportada dentro de una estrategia DCA disciplinada.

La utilidad financiera de esta señal no se decide en este capítulo. Aquí se evalúa si las formulaciones técnicas tienen coherencia predictiva y qué limitaciones presentan. La conversión de esas salidas en estrategias de inversión y su comparación económica con referencias pasivas se reserva para el capítulo de backtesting.

6.2. Ingeniería de características técnicas del NASDAQ100

La ingeniería de características técnicas se realiza sobre el histórico del NASDAQ100 a frecuencia mensual. El notebook de extracción técnica no entrena ningún modelo; su función es transformar la serie del índice en un conjunto de señales mensuales que describen diferentes dimensiones del estado del mercado. Para evitar perder demasiadas observaciones al inicio por el cálculo de ventanas móviles, los indicadores se calculan sobre un histórico extendido y posteriormente se recortan al rango temporal útil del sistema.

El artefacto resultante es `NASDAQ_tecnico_features_v2.csv`. Este fichero contiene 300 filas, cubre el periodo comprendido entre febrero de 2001 y enero de 2026 e incluye 19 columnas. Entre ellas se encuentran columnas de fecha, el nivel del NASDAQ100, placeholders de contexto macro y las variables técnicas calculadas. En este punto del pipeline, `macro_regime` y `macro_prob` aparecen todavía vacías, porque la integración real del contexto macro se realiza en el notebook posterior.

Tabla 14: Artefactos principales del bloque técnico.

Artefacto	Filas	Periodo	Función dentro del bloque técnico
NASDAQ_tecnico_features_v2.csv	300	2001-02 a 2026-01	Dataset de señales técnicas mensuales del NASDAQ100, antes de integrar contexto macro real y target técnico.
MASTER_tecnico_labeled_v1.csv	283	2002-01 a 2025-07	Dataset técnico final con señales técnicas, contexto macro y columnas de etiquetado DCA.
tecnico_model_v1_metadata.json	—	—	Metadatos del baseline multiclase: target, modelo seleccionado, periodos, features y métricas de test.

Las variables técnicas no se introducen como reglas fijas de compra o venta. Cada una actúa como descriptor de una dimensión del mercado. Por ejemplo, los retornos recientes aproximan momentum; los ratios respecto a medias móviles describen tendencia; la volatilidad y el drawdown capturan riesgo; y la posición relativa frente al histórico reciente ayuda a identificar si el índice se encuentra cerca de máximos o en zonas deprimidas.

6.2.1. Momentum, tendencia y volatilidad

El primer grupo de variables técnicas recoge la fuerza reciente del índice. Dentro de momentum se incluyen `return_1m`, `return_3m`, `return_6m` y `rsi_14`. Los retornos a uno, tres y seis meses permiten observar la evolución reciente a distintos horizontes, mientras que el RSI resume la intensidad relativa de los movimientos recientes.

El segundo grupo describe tendencia. El proceso de extracción calcula medias móviles de seis y doce meses y, a partir de ellas, genera ratios y pendientes. Las columnas finales relevantes son `price_ma_6m_ratio`, `price_ma_12m_ratio`, `ma_6m_slope` y `ma_12m_slope`. Los ratios precio/media móvil indican si el índice cotiza por encima o por debajo de su referencia reciente, mientras que las pendientes capturan si la tendencia media se está fortaleciendo o debilitando.

La volatilidad se representa mediante `volatility_3m`. Esta variable resume la variabilidad reciente del índice y resulta relevante porque la decisión de aportación no depende únicamente del retorno esperado, sino también de la incertidumbre asociada a ese retorno. En un sistema DCA, una señal técnica con buen momentum pero elevada volatilidad puede no tener la misma lectura que otra con avance más estable.

6.2.2. Drawdown, posición relativa y calidad retorno-riesgo

El segundo conjunto de variables se centra en el riesgo acumulado, la posición relativa del índice y la calidad del movimiento. El drawdown se recoge mediante `drawdown_12m`, que mide el deterioro frente a máximos recientes dentro de una ventana anual. Esta variable permite distinguir entre mercados que simplemente han corregido de forma moderada y mercados que han sufrido caídas más profundas.

La posición relativa se captura mediante `percentile_24m` y `distance_to_high_24m`. El percentil a 24 meses indica la posición del índice frente a su distribución reciente, mientras que la distancia al máximo de 24 meses ayuda a identificar si el mercado se encuentra cerca de máximos o alejado de ellos. Estas variables son útiles porque dos meses con el mismo retorno reciente pueden encontrarse en contextos técnicos muy distintos.

Finalmente, la calidad retorno-riesgo se aproxima con `return_vol_ratio_3m`. Esta señal relaciona retorno reciente y volatilidad, de forma que intenta distinguir movimientos fuertes pero inestables de avances más proporcionados al riesgo asumido. Su utilidad debe interpretarse como una variable descriptiva más dentro del conjunto técnico, no como una regla autónoma de decisión.

Tabla 15: Familias de variables técnicas del NASDAQ100.

Familia	Variables principales	Información representada
Momentum	<code>return_1m</code> , <code>return_3m</code> , <code>return_6m</code> , <code>rsi_14</code>	Fuerza reciente del índice a distintos horizontes.
Tendencia	<code>price_ma_6m_ratio</code> , <code>price_ma_12m_ratio</code> ,	Relación con medias móviles y dirección de la

Familia	Variables principales	Información representada
	ma_6m_slope, ma_12m_slope	tendencia.
Volatilidad y drawdown	volatility_3m, drawdown_12m	Variabilidad reciente y deterioro frente a máximos.
Posición relativa	percentile_24m, distance_to_high_24m	Ubicación del índice frente a su historia reciente.
Calidad retorno-riesgo	return_vol_ratio_3m	Relación entre retorno reciente y volatilidad.

6.3. Integración del contexto macroeconómico

La integración macro-técnica se realiza en el dataset final del analista técnico. El fichero técnico inicial contiene placeholders para `macro_regime` y `macro_prob`, pero esos valores no se inventan en la fase de extracción técnica. En el notebook de construcción del dataset técnico se carga el contexto macro histórico final producido por el analista macroeconómico y se fusiona con las señales técnicas por fecha. Así, el bloque técnico opera con una señal macro generada por el modelo, no con una etiqueta retrospectiva perfecta.

Durante la integración se comprueba la existencia de las columnas necesarias y se eliminan las filas iniciales sin contexto macro usable. En la versión actual del pipeline se pierden 11 filas iniciales del año 2001 por falta de historia previa suficiente para generar una predicción macro comparable. Tras integrar las señales técnicas, el contexto macro y las columnas de etiquetado, el dataset final `MASTER_tecnico_labeled_v1.csv` contiene 283 filas, cubre el periodo entre enero de 2002 y julio de 2025 y dispone de 24 columnas.

En este dataset final, `macro_regime` toma el valor 0 en 179 observaciones y el valor 1 en 104 observaciones. Esta distribución no debe leerse como una descripción económica definitiva de todo el periodo, sino como el contexto generado por el analista macro y consumido por el bloque técnico.

6.4. Construcción del target técnico de agresividad DCA

El target técnico se construye para traducir el comportamiento futuro del NASDAQ100 en una decisión de agresividad DCA. La pregunta que se intenta responder no es si el mercado subirá o bajará en términos absolutos, sino qué multiplicador de aportación habría sido razonable dado el equilibrio entre retorno futuro y riesgo futuro.

Para ello, el dataset técnico final añade columnas auxiliares calculadas con información posterior a cada fecha: `R_t_6`, `DD_t_6`, `Score_t_tech`, `Y_tech` y `Y_tech_ord`. Estas columnas son necesarias para construir y evaluar el target, pero no deben introducirse como

variables predictoras del modelo. Usarlas como features produciría fuga de información, porque contienen o derivan de información futura.

6.4.1. Retorno futuro, drawdown futuro y score técnico

El primer componente del target es R_{t_6} , que representa el retorno futuro a seis meses. El segundo es DD_{t_6} , que mide el peor drawdown futuro dentro de esa misma ventana. Estos dos elementos permiten distinguir entre meses que terminan ofreciendo un buen retorno con riesgo contenido y meses cuyo resultado final puede estar acompañado de caídas relevantes durante el camino.

El score técnico se define como:

$$\text{Score}_{t_tech} = R_{t_6} - |DD_{t_6}|$$

Esta fórmula penaliza el retorno futuro por el drawdown sufrido en la ventana. Así, un mes obtiene un score mayor si combina retorno posterior elevado y drawdown contenido. Por el contrario, un retorno positivo acompañado de una caída intensa durante el periodo recibe una valoración menos favorable. Esta construcción es coherente con una estrategia DCA, porque la decisión de aportar más o menos capital no depende solo del retorno final, sino también del riesgo atravesado hasta alcanzarlo.

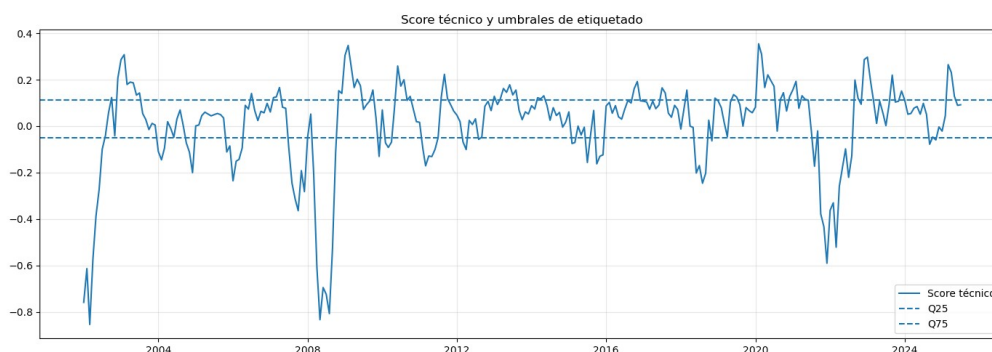


Figura 4: Evolución temporal del score técnico y de los umbrales de discretización utilizados para construir los niveles de agresividad DCA. Los cortes Q25 y Q75 separan las observaciones en aportación reducida, base y agresiva.

6.4.2. Discretización en multiplicadores 0.5x, 1x y 2x

Una vez calculado Score_{t_tech} , el valor continuo se discretiza mediante percentiles globales. Las observaciones situadas por debajo de Q25 se etiquetan como $Y_{tech} = 0.5$; las situadas entre Q25 y Q75 se etiquetan como $Y_{tech} = 1$; y las situadas por encima de Q75 se etiquetan como $Y_{tech} = 2$. Esta discretización convierte el score en una decisión de aportación reducida, base o agresiva.

Además, el pipeline crea Y_{tech_ord} , una codificación ordinal compatible con modelos de clasificación estándar. En esta codificación, 0 representa el multiplicador 0.5x, 1 representa el multiplicador 1x y 2 representa el multiplicador 2x. La distribución global del dataset final es simétrica en los extremos: 71 observaciones pertenecen a la clase 0.5x, 141 a la clase 1x y 71 a la clase 2x.

Tabla 16: Discretización del target técnico de agresividad DCA.

Regla sobre el score	Y_tech	Y_tech_ord	Interpretación DCA
Por debajo de Q25	0.5	0	Aportación reducida.
Entre Q25 y Q75	1	1	Aportación base.
Por encima de Q75	2	2	Aportación agresiva.

Aunque esta distribución global evita un desbalance extremo en el dataset completo, el tramo de test final es mucho más pequeño y está concentrado en la clase intermedia. En el periodo de test de enero de 2024 a julio de 2025 hay 19 observaciones: 2 pertenecen a la clase 0, 13 a la clase 1 y 4 a la clase 2. Por ello, las métricas de test deben interpretarse con cautela.

6.5. Experimentos de modelado técnico

El bloque técnico no se limita a un único modelo. El proyecto compara tres formulaciones para estudiar si el problema se resuelve mejor como clasificación multiclase, como detección binaria de un extremo de agresividad o como regresión de un score continuo. Esta comparación es relevante porque las tres aproximaciones no contestan exactamente la misma pregunta.

La clasificación multiclase intenta predecir directamente los tres niveles de agresividad DCA. La reformulación binaria agresiva pregunta si un mes pertenece o no al caso de aportación 2x. La regresión del score técnico intenta predecir primero Score_t_tech como variable continua y después discretizar la predicción. Por tanto, sus métricas no deben compararse como si todas midieran el mismo problema con idéntico objetivo.

6.5.1. Clasificación multiclase como baseline

El baseline técnico se formula como un problema de clasificación multiclase sobre Y_tech_ord. El objetivo es predecir directamente si la aportación mensual debería ser reducida, base o agresiva. Este baseline utiliza 17 features: macro_regime, macro_prob, las señales técnicas del NASDAQ100 y las variables estacionales Mes_sin y Mes_cos. Se excluyen las columnas de fecha, el nivel bruto del índice, las columnas auxiliares de etiquetado y los targets.

El split temporal principal entrena con observaciones desde enero de 2002 hasta diciembre de 2023 y evalúa en test desde enero de 2024 hasta julio de 2025. El conjunto de entrenamiento contiene 264 observaciones y el test 19. Dentro del entrenamiento se aplica validación walk-forward anual. Los modelos candidatos son logistic_regression, random_forest y extra_trees; la selección se basa en balanced_accuracy_mean y, después, en f1_macro_mean. El modelo seleccionado es logistic_regression.

Tabla 17: Resultados del baseline técnico multiclase.

Métrica o elemento	Valor	Lectura prudente
Modelo seleccionado	logistic_regression	Mejor modelo según la regla de selección del baseline.
Número de features	17	Incluye contexto macro, señales técnicas y estacionalidad.
CV accuracy media	0.3561	Rendimiento limitado en clasificación directa de tres clases.
CV balanced accuracy media	0.3706	Métrica principal de selección; mejora poco sobre una referencia aleatoria multiclase.
CV F1 macro medio	0.2503	Indica dificultad para equilibrar rendimiento entre clases.
Test accuracy	0.3684	Test pequeño de 19 observaciones.
Test balanced accuracy	0.3526	Resultado débil para separación de tres niveles.
Test F1 macro	0.2654	Confirma que el baseline no resuelve de forma robusta el problema multiclase.

La conclusión del baseline no es que el pipeline esté mal construido, sino que la formulación directa en tres clases resulta exigente para la muestra disponible. Con 283 observaciones totales y un test de 19 meses, separar de forma estable aportación reducida, base y agresiva es difícil, especialmente si muchas observaciones se encuentran cerca de los umbrales de discretización.

6.5.2. Reformulación binaria agresiva

El notebook de experimentos de clasificación estudia si el problema técnico puede formularse de manera más simple. En lugar de intentar distinguir simultáneamente 0.5x, 1x y 2x, se prueban casos binarios centrados en detectar extremos. La variante más relevante para el sistema es el caso agresivo, definido como x2 vs resto. Esta formulación pregunta si un mes debe recibir una aportación agresiva o no.

Para el caso agresivo, el mejor modelo por validación cruzada es `extra_trees`. Sus métricas en validación son: accuracy media 0.7121, balanced accuracy media 0.5859, F1 medio 0.1576, precision media 0.1818, recall medio 0.1515 y ROC AUC medio 0.6273. En el test final obtiene accuracy 0.5789, balanced accuracy 0.5500, F1 0.3333, precision 0.2500, recall 0.5000 y ROC AUC 0.6500.

Tabla 18: Resultados de la reformulación binaria agresiva.

Métrica o elemento	Valor	Lectura prudente
Objetivo	x2 vs resto	Detectar meses de aportación agresiva.
Modelo seleccionado por CV	<code>extra_trees</code>	Seleccionado por su comportamiento en validación cruzada.
CV balanced accuracy media	0.5859	Mejora frente al baseline multiclase, pero mide otro problema.
CV F1 medio	0.1576	Valor bajo; la clase positiva sigue siendo difícil.
CV ROC AUC medio	0.6273	Señal moderada de ordenación.
Test balanced accuracy	0.5500	Resultado positivo pero no concluyente.
Test F1	0.3333	Debe interpretarse junto con precision y recall.
Test ROC AUC	0.6500	Señal moderada, no superioridad definitiva.

Esta línea es prometedora porque simplifica el problema y se centra en detectar un extremo de interés operativo: los meses en los que podría tener sentido aumentar la aportación. Sin embargo, no debe presentarse como ganadora absoluta. En test aislado, otros modelos del mismo experimento obtienen mejores valores en algunas métricas, pero la selección

documentada se basa en validación cruzada. Además, sus resultados predictivos todavía deben traducirse a comportamiento financiero mediante backtesting antes de extraer conclusiones económicas.

6.5.3. Regresión del score técnico

La tercera formulación explora si el problema puede abordarse mejor como regresión. En lugar de clasificar directamente el multiplicador DCA, se intenta predecir `Score_t_tech` como variable continua y después convertir la predicción en clases mediante los mismos criterios de discretización. Esta idea responde a una posible limitación del target multiclase: si las fronteras entre clases son ruidosas, modelar primero la magnitud continua podría capturar información que se pierde al discretizar demasiado pronto.

Los regresores comparados son `ridge`, `random_forest_reg` y `extra_trees_reg`. La selección principal se realiza por MAE en validación walk-forward, y el modelo elegido es `ridge`. Para este modelo, el MAE medio de validación es 0.1262, el RMSE medio 0.1566 y el R^2 medio -3.6179. Tras discretizar las predicciones, obtiene accuracy media 0.5379, balanced accuracy media 0.4687 y F1 macro medio 0.3403 en validación.

En test, `ridge` obtiene MAE 0.0754, RMSE 0.1086, R^2 -0.6196, accuracy discretizada 0.6842, balanced accuracy discretizada 0.3333 y F1 macro discretizado 0.2708. El resultado muestra que la regresión puede aportar una lectura continua del score, pero no desplaza con claridad al baseline cuando la predicción se convierte de nuevo en una decisión discreta.

Tabla 19: Resultados de la regresión del score técnico con `ridge`.

Métrica o elemento	Valor	Lectura prudente
Objetivo continuo	<code>Score_t_tech</code>	Predicción continua antes de discretizar.
Modelo seleccionado	<code>ridge</code>	Elegido por MAE en validación walk-forward.
MAE CV	0.1262	Error medio absoluto sobre el score.
RMSE CV	0.1566	Penaliza errores mayores en el score.
R^2 CV	-3.6179	No indica buen ajuste explicativo continuo.
CV balanced accuracy discretizada	0.4687	Métrica tras convertir predicción continua a clases.
CV F1 macro discretizado	0.3403	Mejora parcial frente a algunas métricas del

Métrica o elemento	Valor	Lectura prudente
		baseline, pero con otro enfoque.
Test MAE	0.0754	Error continuo en test.
Test balanced accuracy discretizada	0.3333	No mejora claramente la decisión discreta final.
Test F1 macro discretizado	0.2708	Resultado cercano al baseline multiclase.

Conviene señalar un matiz importante: en test aislado, `extra_trees_reg` obtiene mejores métricas discretizadas que `ridge` en algunas columnas. Sin embargo, el notebook selecciona el modelo por estabilidad y error continuo en validación, no por elegir a posteriori el mejor resultado puntual de test. Esta decisión evita una lectura oportunista del tramo final.

6.6. Resultados comparativos del analista técnico

La comparación de resultados debe hacerse con cuidado porque las tres variantes no resuelven exactamente el mismo problema. El baseline multiclase intenta predecir directamente tres clases. La variante binaria agresiva solo busca detectar la clase 2x frente al resto. La regresión predice un score continuo y después lo discretiza. Por ello, las métricas de la Tabla 20 deben interpretarse como una comparación orientativa de formulaciones, no como una clasificación absoluta de modelos equivalentes.

Tabla 20: Comparativa consolidada de variantes técnicas.

Variante	Objetivo	Modelo	Métricas principales
Multiclase base	0.5 / 1 / 2	logistic_regression	CV BA: 0.3706; Test BA: 0.3526; Test F1: 0.2654.
Binario agresivo	2x vs resto	extra_trees	CV BA: 0.5859; Test BA: 0.5500; Test F1: 0.3333.
Regresión discretizada	Score_t_tech a 0.5 / 1 / 2	ridge	CV BA: 0.4687; Test BA: 0.3333; Test F1: 0.2708.

La línea binaria agresiva aparece como la variante más interesante desde el punto de vista predictivo, especialmente en balanced accuracy. Sin embargo, esa lectura debe matizarse por dos razones. Primero, la tarea binaria es distinta y más simple que la tarea multiclase. Segundo, una mejora predictiva no implica automáticamente una mejora financiera. Para

saber si detectar meses 2x aporta valor económico, la señal debe transformarse en aportaciones y evaluarse mediante backtesting.

El análisis del impacto macro también es relevante. El experimento con todas las features, incluyendo `macro_regime` y `macro_prob`, obtiene una balanced accuracy media de validación de 0.3559 y una balanced accuracy de test de 0.4615. Al eliminar el contexto macro, esas cifras pasan a 0.3518 en validación y 0.3782 en test. Esta comparación sugiere que el contexto macro aporta señal práctica, aunque no resuelve por sí solo la dificultad del problema técnico.

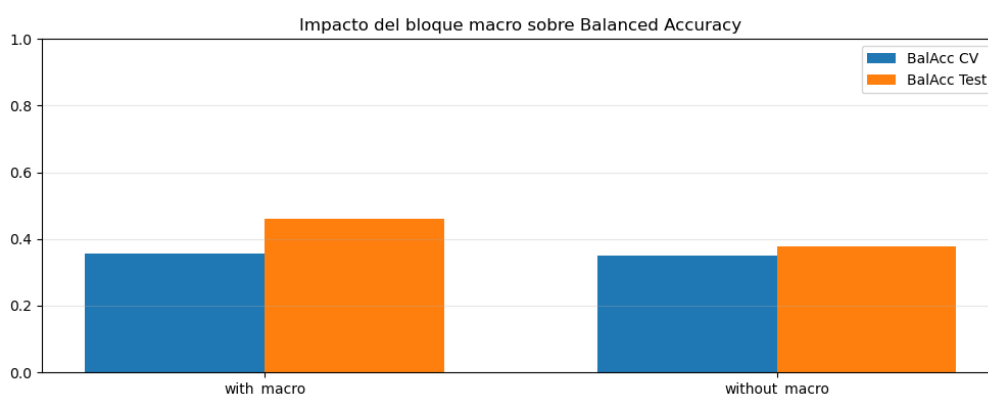


Figura 5: Impacto del contexto macroeconómico sobre la balanced accuracy del bloque técnico. La comparación permite valorar si las variables macro generadas por el analista macro aportan señal adicional al modelo técnico.

6.7. Interpretación y limitaciones del bloque técnico

El desarrollo del analista técnico muestra que el problema de modular aportaciones DCA es más complejo que una simple clasificación de mercado. El baseline multiclase está metodológicamente bien construido, pero sus resultados son débiles. Esto indica que separar directamente tres niveles de agresividad no es trivial con la muestra disponible y con un target construido por percentiles.

La primera limitación es el tamaño muestral. El dataset final contiene 283 observaciones mensuales, de las cuales solo 19 pertenecen al test final. Aunque la distribución global de `Y_tech` es equilibrada por diseño, el test está concentrado en la clase intermedia: 13 de las 19 observaciones corresponden a 1x. Esta concentración hace que las métricas de test puedan ser inestables y obliga a evitar conclusiones fuertes basadas únicamente en ese tramo.

La segunda limitación es la compresión del target cerca de los umbrales. El análisis de bandas muestra que aproximadamente el 49.8233% de las observaciones cae en una banda de ± 0.05 alrededor de los cortes de discretización. Esto significa que muchas observaciones están cerca de cambiar de clase. En consecuencia, pequeñas variaciones del score pueden convertir una observación 0.5x en 1x, o una observación 1x en 2x, sin que necesariamente exista una separación económica muy clara.

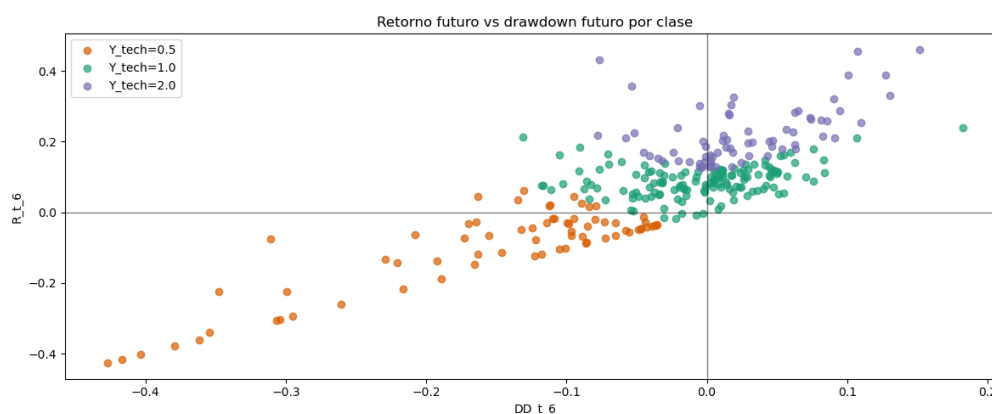


Figura 6: Relación entre retorno futuro y drawdown futuro por clase de agresividad DCA. La figura muestra la lógica económica del target técnico y evidencia que la separación entre clases no es trivial.

La tercera limitación es la sensibilidad a la formulación del problema. La clasificación multiclase, la clasificación binaria agresiva y la regresión del score ofrecen lecturas distintas. La binaria agresiva parece más prometedora, pero no debe convertirse automáticamente en ganadora absoluta. La regresión del score aporta una perspectiva continua, pero sus resultados discretizados no muestran una mejora clara de la decisión final. Por tanto, cada formulación debe entenderse como una forma distinta de aproximar el problema, no como una sustitución inequívoca de las demás.

La cuarta limitación es la diferencia entre rendimiento predictivo y utilidad financiera. Una métrica como balanced accuracy o F1 macro informa sobre la capacidad del modelo para clasificar observaciones, pero no mide directamente rentabilidad, drawdown de cartera, eficiencia del capital extra ni comportamiento frente a una estrategia DCA base. Esas dimensiones económicas corresponden al capítulo de backtesting.

En conjunto, el bloque técnico aporta una capa esencial al sistema porque transforma señales del NASDAQ100 y contexto macro en decisiones de agresividad DCA. Sus resultados muestran aprendizaje parcial y justifican explorar formulaciones alternativas, pero también obligan a una interpretación prudente. El analista técnico no debe presentarse como predictor exacto del índice ni como prueba autónoma de mejora financiera. Su utilidad real se evalúa al convertir sus salidas en estrategias de aportación y compararlas económicamente en el capítulo siguiente.

Capítulo 7. Evaluación financiera y resultados del sistema

Este capítulo evalúa la utilidad financiera del sistema macro-técnico desarrollado en los capítulos anteriores. Después de construir un analista macroeconómico y un analista técnico, la pregunta relevante deja de ser únicamente si los modelos mejoran alguna métrica predictiva. La cuestión central pasa a ser si las señales generadas por esos modelos pueden transformarse en decisiones de aportación dentro de una estrategia Dollar Cost Averaging (DCA) y si esas decisiones producen un comportamiento económico razonable frente a referencias simples.

La evaluación financiera se realiza mediante backtesting. El backtest no pretende demostrar que el sistema pueda anticipar el mercado con certeza ni garantizar una mejora frente a estrategias pasivas. Su función es más concreta: convertir las predicciones de los modelos técnicos en multiplicadores de aportación, simular compras mensuales del NASDAQ100 y comparar el resultado con una estrategia DCA base y con un benchmark pasivo Buy & Hold. De esta forma, el capítulo actúa como puente entre el bloque de machine learning y la interpretación económica final del Trabajo Fin de Máster.

La lectura de los resultados debe hacerse con cautela. Una estrategia puede mostrar mayor valor final de cartera simplemente porque ha invertido más capital. Del mismo modo, una variante con mejores métricas predictivas en balanced accuracy o F1 no tiene por qué convertirse automáticamente en una mejor estrategia financiera. Por este motivo, el análisis combina métricas de rentabilidad, riesgo, capital invertido y eficiencia del capital extra.

7.1. Objetivo del backtesting y lógica de simulación DCA

El objetivo del backtesting es evaluar si las salidas del sistema tienen utilidad práctica como reglas de aportación. En los capítulos predictivos se medía la capacidad de los modelos para clasificar observaciones o estimar un score técnico. En este capítulo, esas salidas se traducen a una decisión económica mensual: cuánto capital aportar al NASDAQ100 en cada fecha.

La lógica de simulación es deliberadamente sencilla y trazable. Cada mes se parte de una aportación base y se multiplica por un factor de agresividad. La regla operativa puede expresarse como:

$$contribution_t = base_amount \times multiplier_t$$

En el backtest, la aportación base mensual es 100 unidades monetarias normalizadas (u.m.n.). No se fija una divisa concreta: la simulación trabaja con una escala arbitraria y comparable entre estrategias. Si el multiplicador es 1.0, se aportan 100 u.m.n.; si es 0.5, se aportan 50 u.m.n.; y si es 2.0, se aportan 200 u.m.n. Con el capital aportado en cada mes se compran participaciones teóricas del NASDAQ100 al nivel observado del índice en esa fecha. A partir de ahí se calculan las participaciones acumuladas, el capital invertido acumulado, el valor de cartera, el beneficio o pérdida acumulado, la rentabilidad sobre capital y el drawdown.

La simulación no incorpora costes de transacción, fiscalidad ni fricciones operativas adicionales. Su propósito es aislar el efecto de la regla de aportación generada por el

sistema y compararla de forma homogénea con referencias sencillas. Por ello, los resultados deben interpretarse como una evaluación histórica controlada, no como una garantía de comportamiento futuro.

El backtesting se realiza en dos horizontes. El primero es un tramo limpio fuera de muestra, desde enero de 2024 hasta julio de 2025. Este tramo reproduce la pregunta: qué habría ocurrido si el sistema se hubiera cerrado a finales de 2023 y se hubiera aplicado posteriormente. El segundo horizonte es una evaluación walk-forward histórica desde enero de 2013 hasta julio de 2025, en la que cada año se entrena con la información disponible hasta diciembre del año anterior y se simulan las decisiones del año siguiente. Esta segunda evaluación permite observar el comportamiento del sistema en un recorrido temporal más amplio, aunque tampoco constituye una prueba formal de significancia estadística.

7.2. Conversión de predicciones en multiplicadores de aportación

El sistema técnico genera salidas de naturaleza distinta según la formulación utilizada. Para poder evaluarlas financieramente, todas se convierten a multiplicadores de aportación DCA. Esta conversión es necesaria porque el objetivo final no es solo clasificar meses, sino decidir cuánto aportar en cada mes.

En la estrategia multiclase, el modelo predice directamente una clase ordinal. La clase 0 se transforma en multiplicador 0.5, la clase 1 en multiplicador 1.0 y la clase 2 en multiplicador 2.0. Esta regla respeta el significado económico del target técnico definido en el capítulo anterior: aportación reducida, aportación base y aportación agresiva.

En la estrategia binaria agresiva, el modelo no intenta distinguir simultáneamente los tres niveles. Su objetivo es detectar si un mes pertenece al caso de aportación agresiva. Cuando el modelo predice clase positiva, el multiplicador aplicado es 2.0. Cuando no predice clase positiva, el multiplicador aplicado es 1.0. Por tanto, esta estrategia no reduce aportaciones a 0.5; únicamente decide entre aportación base y aportación doble.

En la estrategia de regresión discretizada, el modelo predice primero `Score_t_tech` como variable continua. Posteriormente, esa predicción se discretiza con los cuantiles Q25 y Q75 calculados en el tramo de entrenamiento. Las predicciones situadas por debajo o en Q25 se convierten en multiplicador 0.5, las situadas entre Q25 y Q75 en multiplicador 1.0, y las situadas por encima o en Q75 en multiplicador 2.0. Esta regla mantiene la lógica del target original, pero aplicada a una predicción continua.

La Tabla 21 resume la conversión aplicada a cada variante.

Tabla 21: Conversión de predicciones técnicas en multiplicadores DCA.

Estrategia	Salida predictiva utilizada	Conversión a multiplicador
modelo_multiclase_v1	Clase ordinal 0, 1 o 2	0 -> 0.5; 1 -> 1.0; 2 -> 2.0.
binario_agresivo	Clase binaria de aportación agresiva	Positivo -> 2.0; negativo -> 1.0.

Estrategia	Salida predictiva utilizada	Conversión a multiplicador
		Predicción $\leq$ Q25 $\rightarrow$
regresion_discretizada	Predicción continua de Score_t_tech	0.5; entre Q25 y Q75 $\rightarrow$ 1.0; predicción $\geq$ Q75 $\rightarrow$ 2.0.
dca_base	No utiliza modelo predictivo	Multiplicador constante 1.0.

Esta homogeneización permite comparar estrategias con objetivos predictivos distintos dentro de una misma métrica económica: cuánto capital se aporta y qué valor de cartera se obtiene a lo largo del tiempo.

7.3. Estrategias comparadas

La evaluación compara cinco líneas principales. Cuatro de ellas se basan en la lógica DCA mensual, mientras que una corresponde a un benchmark pasivo Buy & Hold. Las estrategias DCA son dca_base, modelo_multiclase_v1, binario_agresivo y regresion_discretizada. El benchmark pasivo se denomina buy_hold_capital_base.

La Tabla 22 resume la función de cada estrategia dentro del capítulo.

Tabla 22: Estrategias evaluadas en el backtesting financiero.

Estrategia	Tipo de referencia	Descripción operativa
dca_base	Benchmark operativo DCA	Aporta 100 u.m.n. cada mes con multiplicador constante 1.0.
modelo_multiclase_v1	Estrategia dinámica	Usa el baseline técnico multiclase para elegir entre 0.5, 1.0 y 2.0.
binario_agresivo	Estrategia dinámica	Usa el clasificador binario agresivo para elegir entre 1.0 y 2.0.
regresion_discretizada	Estrategia dinámica	Predice el score técnico continuo y lo convierte en 0.5, 1.0 o 2.0.
buy_hold_capital_base	Benchmark pasivo de mercado	Invierte al inicio un capital equivalente al

Estrategia	Tipo de referencia	Descripción operativa
		capital total aportado por dca_base.

7.3.1. DCA base

dca_base es el benchmark operativo principal del capítulo. Representa una estrategia DCA simple, sin predicción y sin ajuste dinámico. Cada mes aporta exactamente la cantidad base, por lo que su multiplicador es siempre 1.0.

Esta referencia es fundamental porque comparte la misma mecánica que las estrategias dinámicas: aportaciones periódicas mensuales y compra progresiva de participaciones del NASDAQ100. Por tanto, cuando se evalúa si el sistema macro-técnico aporta valor dentro de una lógica DCA, la comparación más directa no es Buy & Hold, sino dca_base.

En el tramo limpio de enero de 2024 a julio de 2025, dca_base invierte 1900 u.m.n., obtiene un valor final de cartera de 2229.6365 u.m.n. y genera un retorno sobre capital de 0.1735. En la evaluación walk-forward de enero de 2013 a julio de 2025, invierte 15100 u.m.n., alcanza un valor final de 52230.0775 u.m.n. y obtiene un retorno sobre capital de 2.4589. Estos valores sirven como referencia para analizar si las estrategias dinámicas compensan el capital adicional que puedan invertir.

7.3.2. Estrategias derivadas de los modelos técnicos

Las estrategias dinámicas proceden de las tres formulaciones técnicas seleccionadas en el capítulo anterior. El backtest no utiliza modelos distintos a los ganadores documentados en los experimentos predictivos. Esta alineación es importante porque garantiza que la evaluación financiera representa el sistema final y no una aproximación intermedia.

La primera estrategia, modelo_multiclase_v1, utiliza el baseline multiclase basado en logistic_regression. Su objetivo predictivo era clasificar directamente los tres niveles de aportación: 0.5, 1.0 y 2.0. En el backtest, cada clase predicha se transforma en su multiplicador correspondiente.

La segunda estrategia, binario_agresivo, utiliza extra_trees para detectar el caso x2 vs resto. Esta variante se había mostrado como la línea más interesante desde el punto de vista predictivo, especialmente en balanced accuracy, pero su utilidad económica debe comprobarse de forma separada. En el backtest solo distingue entre aportación base y aportación doble.

La tercera estrategia, regresion_discretizada, utiliza ridge para predecir el score técnico continuo. Después discretiza la predicción en multiplicadores mediante los cuantiles del score en el entrenamiento. Esta formulación permite evaluar si modelar el score como variable continua se traduce en una mejor decisión de aportación.

La Tabla 23 muestra los modelos que alimentan las tres estrategias dinámicas.

Tabla 23: Modelos técnicos utilizados como entrada de las estrategias dinámicas.

Variante técnica	Objetivo predictivo	Modelo	Test BA
Multiclase base	0.5 / 1 / 2	logistic_regression	0.3526
Binario agresivo	2x vs resto	extra_trees	0.5500
Regresión discretizada	Score_t_tech a 0.5 / 1 / 2	ridge	0.3333

Esta tabla se incluye solo para mantener la trazabilidad entre los capítulos predictivos y la evaluación financiera. No debe interpretarse como una clasificación económica de estrategias. La utilidad financiera se analiza con las tablas de backtesting.

7.3.3. Benchmark Buy & Hold

buy_hold_capital_base se utiliza como benchmark pasivo de mercado. Su construcción es distinta a la de las estrategias DCA: en lugar de aportar cada mes, invierte al inicio del periodo un capital igual al capital total que dca_base acabaría aportando en ese mismo horizonte. Después mantiene la posición hasta el final.

Esta referencia permite observar qué habría ocurrido con una exposición pasiva completa al NASDAQ100 durante el periodo analizado. Sin embargo, no es equivalente a DCA. La diferencia es metodológica y económica: Buy & Hold adelanta todo el capital al primer mes, mientras que DCA reparte las aportaciones a lo largo del tiempo. Por tanto, Buy & Hold sirve como referencia de mercado, pero no sustituye al benchmark operativo dca_base.

En el tramo limpio, Buy & Hold invierte 1900 u.m.n. al inicio y alcanza un valor final de 2574.1851 u.m.n., con un retorno sobre capital de 0.3548. En la evaluación walk-forward, invierte 15100 u.m.n. al inicio y alcanza 128350.6357 u.m.n., con un retorno sobre capital de 7.5000. Estos resultados reflejan la fuerte tendencia alcista del NASDAQ100 en el periodo observado, pero no implican que la mecánica DCA quede invalidada, porque la estructura temporal de aportaciones es diferente.

7.4. Métricas financieras utilizadas

La evaluación utiliza métricas de rentabilidad, riesgo, capital invertido y eficiencia. La Tabla 24 resume su interpretación.

Tabla 24: Métricas financieras empleadas en la evaluación.

Métrica	Interpretación
Capital invertido	Suma de todas las aportaciones realizadas por la estrategia durante el periodo.
Valor final de cartera	Valor de las participaciones

Métrica	Interpretación
	acumuladas al final del horizonte evaluado.
PnL	Diferencia entre valor final de cartera y capital invertido.
Retorno sobre capital	Rentabilidad acumulada calculada como valor de cartera dividido entre capital invertido menos uno.
Maximum drawdown	Mayor caída porcentual desde un máximo acumulado del valor de cartera.
Multiplicador medio	Media de los multiplicadores mensuales aplicados por la estrategia DCA.
TIR anualizada	Rentabilidad anual implícita considerando los flujos de caja mensuales de aportación y el valor final de cartera.
CAGR	Tasa anual compuesta calculada sobre una inversión inicial y un valor final, utilizada especialmente para el benchmark Buy & Hold.
Volatilidad anual	Volatilidad anualizada de los retornos mensuales del activo en el benchmark Buy & Hold.
Sharpe rf0	Ratio Sharpe calculado con tipo libre de riesgo igual a cero para el benchmark pasivo.
_efficiency	PnL adicional generado por cada unidad adicional de capital invertido frente a dca_base.

La distinción entre TIR anualizada y CAGR es especialmente importante. En una estrategia con aportaciones periódicas, el capital no se invierte todo desde el primer día. Por ello, la

TIR anualizada resulta más informativa que un CAGR simple, porque incorpora la secuencia temporal de flujos de caja. El CAGR es más natural para Buy & Hold, donde el capital se invierte al inicio y se mantiene durante todo el periodo.

La métrica `capital_extra_efficiency` se calcula comparando cada estrategia con `dca_base`. Conceptualmente mide:

$$\frac{PnL_{estrategia} - PnL_{DCA}}{Capital_{estrategia} - Capital_{DCA}}$$

Si una estrategia invierte más capital que `dca_base`, esta métrica indica cuánto PnL adicional obtiene por cada unidad extra invertida. Si no invierte capital adicional, la métrica no se calcula. Esta medida es necesaria porque una cartera final más alta puede deberse simplemente a haber aportado más dinero.

7.5. Resultados en tramo limpio fuera de muestra

El tramo limpio fuera de muestra cubre desde enero de 2024 hasta julio de 2025. Son 19 observaciones mensuales. Esta muestra es útil porque representa un periodo posterior al entrenamiento principal, pero su reducido tamaño obliga a evitar conclusiones fuertes. Los resultados deben leerse como evidencia histórica limitada, no como una prueba definitiva.

La Tabla 25 resume los resultados principales del tramo limpio.

Tabla 25: Resultados financieros en el tramo limpio fuera de muestra, 2024-01 a 2025-07.

Estrategia	Capital (u.m.n.)	Valor final (u.m.n.)	Retorno	MDD	TIR	Eficiencia capital extra
buy_hold						—
_capital_base	1900	2574.19	0.3548	-0.1024	0.2244	
binario_agresivo	2700	3195.29	0.1834	-0.0312	0.2226	0.2071
modelo_multiclase_v1	3050	3605.45	0.1821	-0.0562	0.2265	0.1964
dca_base	1900	2229.64	0.1735	-0.0107	0.2304	—
regresion_discretizada	1900	2229.64	0.1735	-0.0107	0.2304	—

En el tramo limpio, Buy & Hold presenta el mayor retorno sobre capital, 0.3548. Esta lectura debe entenderse como benchmark pasivo de mercado; durante este tramo concreto, estar invertido desde el principio en el NASDAQ100 fue muy favorable.

Entre las estrategias DCA, `binario_agresivo` obtiene el mayor retorno sobre capital, 0.1834, ligeramente por encima de `modelo_multiclase_v1`, que obtiene 0.1821, y de `dca_base`, que obtiene 0.1735. Sin embargo, estas diferencias son reducidas y están condicionadas por el capital invertido. `binario_agresivo` invierte 2700 u.m.n., 800 más que `dca_base`; `modelo_multiclase_v1` invierte 3050 u.m.n., 1150 más que `dca_base`. Por tanto, no basta con observar el valor final o el PnL bruto.

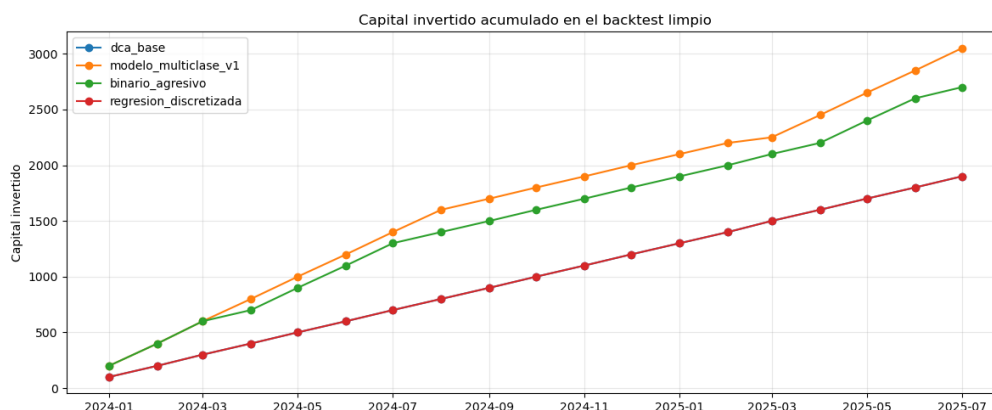


Figura 7: Capital invertido acumulado por estrategia en el tramo limpio fuera de muestra. La figura ayuda a interpretar las diferencias de valor final y PnL teniendo en cuenta que algunas estrategias dinámicas aportan más capital que el DCA base.

`regresion_discretizada` se comporta exactamente igual que `dca_base` en este tramo. Ambas estrategias invierten 1900 u.m.n., alcanzan un valor final de 2229.6365 u.m.n. y obtienen un retorno de 0.1735. Esto indica que, al aplicar el modelo seleccionado de regresión y discretizar sus predicciones, las decisiones resultantes en el tramo limpio no se diferencian de una aportación mensual constante.

En términos de drawdown máximo, `dca_base` y `regresion_discretizada` muestran el menor deterioro, con -0.0107. `binario_agresivo` registra -0.0312 y `modelo_multiclase_v1` -0.0562. Buy & Hold tiene un drawdown de -0.1024, mayor que el de las estrategias DCA, coherente con su exposición completa desde el inicio.

La lectura prudente es que, en estos 19 meses, las estrategias dinámicas muestran cierta mejora frente a `dca_base` en retorno sobre capital, pero no demuestran una superioridad clara y estable. El periodo es demasiado corto para extraer conclusiones fuertes, y parte de la mejora bruta procede de invertir más capital.

7.6. Resultados en evaluación walk-forward histórica

La evaluación walk-forward histórica cubre desde enero de 2013 hasta julio de 2025, con 151 meses simulados. En cada año se entrena con la información disponible hasta diciembre del año anterior y se aplican las decisiones durante el año siguiente. Esta metodología es más amplia que el tramo limpio final y reduce la dependencia de un periodo muy corto, aunque no constituye una prueba formal de superioridad estadística.

La Tabla 26 resume los resultados de esta evaluación.

Tabla 26: Resultados financieros en la evaluación walk-forward, 2013-01 a 2025-07.

Estrategia	Capital (u.m.n.)	Valor final (u.m.n.)	Retorno	MDD	TIR	Eficiencia capital extra
buy_hold _capital_ base	15100	128350.64	7.5000	-0.3297	0.1867	–
dca_base	15100	52230.08	2.4589	-0.3040	0.1845	–
regresion_discretizada	15250	52404.49	2.4364	-0.3052	0.1844	0.1627
binario_agresivo	17100	55741.94	2.2598	-0.3044	0.1851	0.7559
modelo_multiclase_v1	17500	56607.12	2.2347	-0.2991	0.1873	0.8238

Antes de interpretar los resultados, conviene precisar que cada métrica responde a una pregunta distinta. *Capital* indica cuánto dinero total ha aportado cada estrategia; *Valor final* muestra cuánto vale la cartera al cierre del periodo; *Retorno* expresa la rentabilidad acumulada sobre el capital invertido; *MDD* recoge la peor caída desde máximos durante la simulación; *TIR* estima la rentabilidad anualizada considerando tanto los flujos mensuales de aportaciones como el valor final de cartera; y *Eficiencia capital extra* aproxima la utilidad del capital adicional invertido frente a dca_base. Por ello, un mayor valor final no debe leerse de forma aislada, ya que puede deberse simplemente a un mayor volumen de aportaciones.

En la evaluación walk-forward, Buy & Hold vuelve a obtener el mayor retorno sobre capital, 7.5000, coherente con su papel de benchmark de mercado y con la tendencia alcista del NASDAQ100 durante el periodo.

Entre las estrategias DCA, dca_base obtiene el mayor retorno sobre capital, 2.4589. regresion_discretizada queda muy cerca, con 2.4364, pero invierte 150 u.m.n. adicionales. binario_agresivo y modelo_multiclase_v1 terminan con mayor valor final de cartera que dca_base, pero también invierten bastante más capital: 17100 y 17500 u.m.n., respectivamente, frente a 15100 de dca_base. Por tanto, su mayor valor final no implica por sí solo una mejor eficiencia financiera.

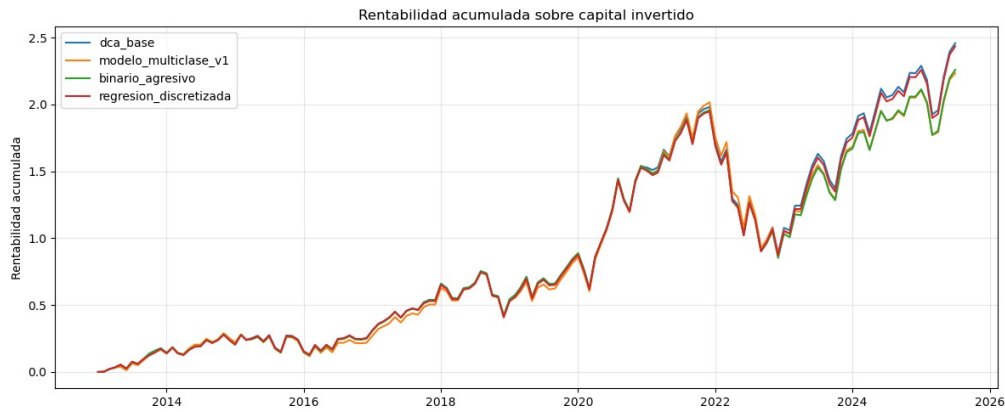


Figura 8: Rentabilidad acumulada sobre capital invertido en la evaluación walk-forward histórica para las estrategias DCA evaluadas. La figura compara la eficiencia económica de las variantes dinámicas frente al DCA base.

La TIR anualizada ofrece una lectura adicional especialmente relevante en estrategias DCA con aportaciones variables, porque no todas las estrategias invierten la misma cantidad ni lo hacen en los mismos meses. Desde esta perspectiva, `modelo_multiclase_v1` obtiene una TIR de 0.1873, ligeramente por encima de `binario_agresivo` con 0.1851, `dca_base` con 0.1845 y `regresion_discretizada` con 0.1844. Las diferencias son pequeñas y no deben interpretarse como prueba estadística de superioridad. Más bien indican que, bajo flujos de caja mensuales, las estrategias dinámicas no desplazan de forma clara a la referencia DCA base, aunque `modelo_multiclase_v1` presenta una TIR ligeramente superior.

En drawdown máximo, las diferencias también son moderadas. `modelo_multiclase_v1` registra -0.2991, `dca_base` -0.3040, `binario_agresivo` -0.3044 y `regresion_discretizada` -0.3052. Aunque `modelo_multiclase_v1` presenta un drawdown máximo algo menor, no aparece una reducción clara y general del drawdown por parte de las estrategias dinámicas. El propio resultado sugiere que la modulación de aportaciones afecta más al capital desplegado que a una mejora contundente del perfil de riesgo.

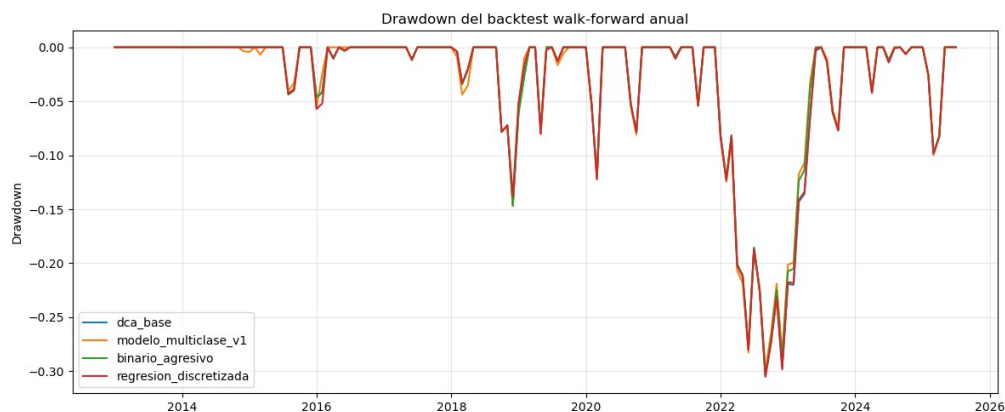


Figura 9: Drawdown acumulado de las estrategias durante la evaluación walk-forward histórica. La figura complementa la comparación de rentabilidad mostrando la evolución del riesgo de caída desde máximos.

7.7. Comparación frente a DCA base y Buy & Hold

La comparación frente a `dca_base` y Buy & Hold debe realizarse en planos distintos. `dca_base` es el benchmark operativo principal porque comparte la mecánica de aportaciones periódicas. Buy & Hold es una referencia pasiva de mercado, útil para contextualizar el comportamiento del NASDAQ100, pero no tiene la misma estructura temporal de inversión.

Frente a `dca_base`, el tramo limpio muestra que `binario_agresivo` y `modelo_multiclase_v1` obtienen retornos sobre capital algo superiores. Sin embargo, ambas estrategias invierten más capital. `regresion_discretizada` no se diferencia de `dca_base` en ese periodo. Por tanto, el tramo limpio no permite afirmar que las estrategias dinámicas batan de forma robusta al DCA base.

En el walk-forward histórico, `dca_base` es competitivo e incluso obtiene el mayor retorno sobre capital entre las estrategias DCA. Las estrategias dinámicas pueden acabar con más valor final de cartera, pero esto se debe en parte a que aportan más capital. Esta diferencia es especialmente visible en `binario_agresivo` y `modelo_multiclase_v1`, cuyos capitales invertidos son 2000 y 2400 u.m.n. superiores al DCA base en el periodo walk-forward.

Frente a Buy & Hold, ninguna estrategia DCA alcanza el mismo retorno sobre capital en los periodos analizados. Esta comparación contextualiza el coste de oportunidad de invertir gradualmente en un mercado alcista, pero no invalida la lógica DCA porque se trata de mecánicas de inversión distintas.

El resultado más honesto es que el sistema macro-técnico no ofrece una superioridad financiera clara y estable frente a `dca_base`. Sí permite construir una evaluación trazable de reglas de aportación dinámicas y estudiar cómo cambian el capital invertido, el valor final, la rentabilidad y el riesgo. En ese sentido, el valor del capítulo no está en vender una estrategia ganadora, sino en medir con rigor si la modulación de aportaciones aporta algo frente a referencias simples.

7.8. Discusión de resultados financieros y eficiencia del capital extra

La eficiencia del capital extra es una métrica clave para interpretar las estrategias dinámicas. Si una estrategia invierte más que `dca_base`, puede generar más PnL aunque su rentabilidad sobre capital sea inferior. Por eso conviene preguntar cuánto beneficio adicional obtiene por cada unidad adicional invertida.

En el tramo limpio, `binario_agresivo` invierte 800 u.m.n. más que `dca_base` y genera 165.6519 u.m.n. adicionales de PnL. Su `capital_extra_efficiency` es 0.2071. `modelo_multiclase_v1` invierte 1150 u.m.n. adicionales y genera 225.8088 u.m.n. adicionales de PnL, con una eficiencia de 0.1964. Estas cifras son positivas, pero no suficientemente fuertes como para afirmar superioridad robusta, especialmente dado que el tramo limpio tiene solo 19 meses.

En el walk-forward histórico, `regresion_discretizada` invierte 150 u.m.n. adicionales frente a `dca_base` y genera 24.4102 u.m.n. adicionales de PnL, con eficiencia 0.1627. `binario_agresivo` invierte 2000 u.m.n. adicionales y genera 1511.8616 u.m.n. adicionales

de PnL, con eficiencia 0.7559. `modelo_multiclase_v1` invierte 2400 u.m.n. adicionales y genera 1977.0411 u.m.n. adicionales de PnL, con eficiencia 0.8238.

Estas cifras matizan la lectura basada en retorno sobre capital. En walk-forward, `dca_base` tiene el mayor retorno sobre capital entre estrategias DCA, pero algunas variantes dinámicas convierten el capital adicional en PnL adicional positivo. Aun así, esto no significa necesariamente que sean mejores estrategias en términos globales. Una estrategia que exige más capital puede no ser adecuada para todos los perfiles de inversor, y la mejora en PnL debe evaluarse junto con drawdown, TIR, simplicidad y estabilidad.

El caso de `binario_agresivo` es especialmente ilustrativo. En el capítulo técnico era la variante más prometedora desde el punto de vista predictivo. En el tramo limpio también obtiene el mayor retorno sobre capital entre las estrategias DCA dinámicas. Sin embargo, en el walk-forward queda por debajo de `dca_base` en retorno sobre capital. Esto confirma que una mejora predictiva no se traduce automáticamente en una mejora financiera clara.

`regresion_discretizada` ofrece otra lectura. En el tramo limpio se comporta igual que `dca_base`; en el walk-forward se mantiene muy cerca, pero ligeramente por debajo en retorno sobre capital. Su comportamiento sugiere que modelar el score técnico de forma continua no genera, en los artefactos actuales, una ventaja económica evidente al convertir las predicciones de nuevo en decisiones discretas de aportación.

En conjunto, la discusión financiera apunta a una conclusión prudente: el sistema consigue transformar señales macro-técnicas en reglas de aportación evaluables, pero los resultados no demuestran una mejora clara, estable y robusta frente a un DCA base simple. La modulación de agresividad es metodológicamente trazable, pero su ventaja económica no queda probada de forma concluyente.

7.9. Limitaciones del backtest

El backtest presenta varias limitaciones que condicionan la interpretación de los resultados. La primera es el tamaño del tramo limpio fuera de muestra. El periodo 2024-01 a 2025-07 contiene solo 19 meses. Aunque es útil como evaluación final posterior al entrenamiento, es demasiado corto para sostener conclusiones fuertes sobre superioridad financiera.

La segunda limitación es que la evaluación walk-forward histórica, aunque más amplia, sigue siendo una simulación sobre un único recorrido histórico. El periodo 2013-01 a 2025-07 está dominado por una evolución muy favorable del NASDAQ100. No se ha realizado una prueba formal de significancia estadística ni una evaluación exhaustiva sobre múltiples activos, universos o escenarios de mercado. Por tanto, no debe afirmarse que las diferencias observadas sean estadísticamente significativas.

La tercera limitación es la diferencia de capital invertido entre estrategias. Las variantes dinámicas pueden aportar más capital que `dca_base`. Por ello, el valor final de cartera y el PnL bruto no son suficientes para decidir qué estrategia es mejor. Es necesario mirar retorno sobre capital, TIR, drawdown y eficiencia del capital extra.

La cuarta limitación es la comparación con Buy & Hold. Este benchmark es útil para contextualizar el mercado, pero su mecánica de inversión inicial única no es equivalente a una estrategia de aportaciones periódicas.

La quinta limitación es que el backtest no incorpora costes de transacción, fiscalidad, restricciones de liquidez ni posibles desviaciones operativas. Tampoco evalúa preferencias

individuales de riesgo, capacidad de aportar capital adicional o tolerancia a variaciones del multiplicador mensual. Estas dimensiones podrían modificar la utilidad práctica de una estrategia dinámica.

Finalmente, el valor principal del backtest es metodológico: comprueba de extremo a extremo que las señales macro-técnicas pueden convertirse en reglas de aportación medibles y comparables. Los resultados actuales, no obstante, no prueban una superioridad robusta frente a `dca_base`; muestran una forma estructurada de modular aportaciones y evaluarlas económicamente bajo las restricciones descritas.

Capítulo 8. Conclusiones y líneas futuras

Este capítulo recoge las conclusiones finales del Trabajo Fin de Máster a partir del desarrollo metodológico y de los resultados obtenidos en los capítulos anteriores. La lectura global debe realizarse con prudencia: el proyecto no plantea un sistema capaz de predecir el mercado con certeza ni una estrategia garantizada para batir a referencias pasivas. Su finalidad ha sido construir, implementar y evaluar una arquitectura macro-técnica que permita modular aportaciones periódicas mediante Dollar Cost Averaging (DCA) sobre el NASDAQ100, utilizando datos macroeconómicos, señales técnicas, aprendizaje supervisado y backtesting.

Las conclusiones se formulan diferenciando tres planos. En primer lugar, el plano metodológico, donde se valora si el sistema completo ha sido construido de forma trazable y temporalmente coherente. En segundo lugar, el plano predictivo, donde se interpretan las métricas de los modelos macroeconómico y técnico sin sobrestimar su robustez. En tercer lugar, el plano financiero, donde se evalúa si las predicciones transformadas en aportaciones generan una mejora clara frente a estrategias de referencia. Esta separación es importante porque una señal predictiva moderadamente útil no se traduce necesariamente en una ventaja económica robusta.

8.1. Conclusiones principales

La principal conclusión del proyecto es que se ha desarrollado un sistema completo, modular y evaluable para estudiar la modulación de aportaciones DCA a partir de información macroeconómica y técnica. El sistema integra tres bloques conectados: un analista macroeconómico mensual, un analista técnico mensual y un módulo de evaluación financiera mediante backtesting. Esta arquitectura permite pasar desde datos económicos y de mercado hasta decisiones concretas de aportación, y posteriormente medir su efecto sobre capital invertido, valor de cartera, retorno, drawdown, TIR y eficiencia del capital adicional.

El analista macroeconómico aporta contexto al resto del sistema mediante dos salidas operativas: `macro_regime` y `macro_prob`. Su función no es decidir directamente cuánto invertir, sino resumir el entorno macro-financiero en una señal operativa que pueda ser incorporada por el bloque técnico. En la versión final, el bloque macro se formula como un filtro binario de entorno favorable a seis meses, entrenado con validación temporal y con una selección reducida de variables para limitar el ruido asociado a una muestra mensual relativamente pequeña. Esta reformulación resulta metodológicamente más defendible que una clasificación multiclase ambiciosa, aunque sus métricas deben interpretarse con cautela, especialmente porque el tramo de test final es limitado.

El analista técnico transforma señales del NASDAQ100 y contexto macroeconómico en decisiones de agresividad DCA. Para ello, utiliza variables de momentum, tendencia, riesgo, posición relativa y calidad retorno-riesgo, combinadas con la salida del analista macro. El target técnico se construye a partir de retorno futuro y drawdown futuro a seis meses, sintetizados en un score y discretizados en tres niveles de aportación: $0.5x$, $1x$ y $2x$. Esta formulación conecta el problema predictivo con una decisión financiera concreta, pero también introduce sensibilidad a la forma de construir el target, a los percentiles utilizados y al solapamiento entre clases.

Los experimentos del bloque técnico muestran que la formulación multiclase directa es exigente para el tamaño y la naturaleza de la muestra. El modelo multiclase final alcanza una *balanced accuracy* de test de 0.3526 y un F1 macro de 0.2654. La reformulación binaria agresiva, orientada a detectar meses de aportación 2x frente al resto, obtiene mejores resultados predictivos relativos, con *balanced accuracy* de test de 0.5500 y ROC AUC de 0.6500. No obstante, estas cifras siguen siendo moderadas y no permiten presentar el modelo técnico como una herramienta predictiva robusta en términos financieros. La regresión del score técnico ofrece una alternativa conceptualmente razonable, pero al discretizar sus predicciones no genera una mejora clara en test frente al resto de aproximaciones.

El backtesting permite convertir las predicciones de los modelos en aportaciones reales y evaluar su impacto financiero. Esta capa es esencial porque evita limitar la evaluación a métricas de clasificación o regresión. En el tramo limpio fuera de muestra, correspondiente al periodo 2024-01 a 2025-07, las estrategias dinámicas muestran diferencias pequeñas frente a *dca_base*. *binario_agresivo* obtiene un retorno sobre capital de 0.1834, *modelo_multiclase_v1* alcanza 0.1821 y *dca_base* obtiene 0.1735. Sin embargo, las dos primeras estrategias invierten más capital que *dca_base*, y el periodo contiene solo 19 meses. Por tanto, estos resultados deben interpretarse como indicios exploratorios, no como evidencia robusta de superioridad financiera.

En la evaluación walk-forward histórica, que cubre el periodo 2013-01 a 2025-07, *dca_base* sigue siendo una referencia muy competitiva. Entre las estrategias DCA, obtiene el mayor retorno sobre capital, con 2.4589, frente a 2.4364 de *regresion_discretizada*, 2.2598 de *binario_agresivo* y 2.2347 de *modelo_multiclase_v1*. Algunas estrategias dinámicas terminan con mayor valor final de cartera, pero también requieren más capital invertido. Esta diferencia confirma la necesidad de evaluar las estrategias no solo por PnL bruto, sino también por retorno sobre capital, TIR, drawdown y eficiencia del capital extra.

Buy & Hold obtiene el mayor retorno sobre capital en los periodos analizados: 0.3548 en el tramo limpio y 7.5000 en el walk-forward histórico. Esta comparación debe tratarse como benchmark de mercado, no como una estrategia equivalente a DCA. Su resultado contextualiza el coste de oportunidad de invertir gradualmente en un periodo muy favorable para el NASDAQ100, sin invalidar por ello la lógica de aportaciones periódicas.

La conclusión financiera final es prudente. El sistema demuestra que es posible conectar datos macroeconómicos, señales técnicas, aprendizaje supervisado, validación temporal y evaluación financiera en reglas de aportación medibles. Sin embargo, los resultados actuales no demuestran una mejora clara, estable y robusta frente a *dca_base*. El proyecto se defiende mejor como una contribución metodológica y experimental que como una estrategia de inversión superior.

8.2. Cumplimiento del objetivo general

El objetivo general definido en el Capítulo 2 consiste en desarrollar y evaluar un sistema de apoyo a la decisión para modular dinámicamente aportaciones periódicas mediante Dollar Cost Averaging sobre el NASDAQ100, integrando contexto macroeconómico, señales técnicas, aprendizaje supervisado y evaluación financiera mediante backtesting.

Este objetivo general se cumple en términos metodológicos y experimentales. El proyecto no se queda en una formulación teórica, sino que construye datasets, entrena modelos, genera predicciones, traduce esas predicciones en decisiones de aportación y evalúa las

estrategias resultantes frente a referencias. El Capítulo 4 define la arquitectura y los artefactos principales; el Capítulo 5 desarrolla el analista macroeconómico; el Capítulo 6 construye y evalúa el analista técnico; y el Capítulo 7 mide el comportamiento financiero de las estrategias generadas.

El cumplimiento del objetivo general no debe confundirse con la demostración de una mejora económica robusta. El sistema cumple su objetivo como marco de apoyo a la decisión porque permite observar cómo cambiaría una estrategia DCA al incorporar contexto macro-técnico y porque ofrece métricas para comparar dichas decisiones frente a referencias simples. Por tanto, el objetivo se considera cumplido como construcción y evaluación de un pipeline completo, no como validación definitiva de una estrategia ganadora.

8.3. Cumplimiento de los objetivos específicos

En primer lugar, se cumple el objetivo de construir una base de datos macro-financiera y técnica con frecuencia mensual. El proyecto genera datasets diferenciados para features macroeconómicas, etiquetas macro, señales técnicas y dataset técnico final. Esta separación facilita la trazabilidad entre extracción, limpieza, etiquetado y modelado, y queda respaldada por los artefactos descritos en el Capítulo 4.

En segundo lugar, se cumple el objetivo de desarrollar un analista macroeconómico. El bloque macro evoluciona desde formulaciones iniciales más ambiciosas hacia una versión binaria centrada en detectar entornos favorables. La salida final del modelo se resume en `macro_regime` y `macro_prob`, variables que posteriormente se integran en el dataset técnico. La principal fortaleza de este bloque no es ofrecer una predicción macro perfecta, sino proporcionar una señal de contexto coherente con la arquitectura jerárquica del sistema.

En tercer lugar, se cumple el objetivo de desarrollar un analista técnico capaz de transformar información del NASDAQ100 y contexto macroeconómico en niveles de agresividad DCA. El proyecto construye variables de momentum, tendencia, volatilidad, drawdown, posición relativa y calidad retorno-riesgo. Posteriormente formula un target técnico basado en retorno futuro y drawdown futuro, discretizado en tres multiplicadores de aportación. El resultado es un dataset técnico final de 283 observaciones, desde 2002-01 hasta 2025-07, con 24 columnas y una distribución equilibrada del target en los tres niveles de agresividad.

En cuarto lugar, se cumple el objetivo de integrar ambos módulos en una arquitectura macro-técnica jerárquica. El bloque macroeconómico no decide directamente la aportación, sino que proporciona contexto al bloque técnico; el bloque técnico combina dicho contexto con señales del NASDAQ100 y genera una estimación de agresividad; y el módulo financiero convierte esa estimación en una estrategia DCA evaluable. Esta separación mantiene la trazabilidad entre contexto, señal técnica, decisión de aportación y resultado financiero.

En quinto lugar, se cumple el objetivo de aplicar validación temporal y walk-forward validation. El proyecto respeta el orden cronológico de las observaciones, diferencia entrenamiento y test final, y utiliza esquemas de validación expansiva para reducir el riesgo de data leakage. Esta decisión es coherente con la naturaleza temporal de los datos financieros y evita interpretar como robustos resultados que podrían depender de una partición inadecuada.

En sexto lugar, se cumple el objetivo de transformar las predicciones en reglas de aportación. Las salidas de los modelos se traducen en multiplicadores DCA interpretables: aportación reducida, aportación base y aportación agresiva, representadas mediante $0.5x$, $1x$ y $2x$. Esta conversión conecta las métricas predictivas con una decisión financiera concreta.

En séptimo lugar, se cumple el objetivo de evaluar financieramente las estrategias mediante backtesting. El módulo financiero compara las estrategias dinámicas frente a `dca_base` y Buy & Hold, distinguiendo entre calidad predictiva y utilidad económica. Los resultados muestran que una mejora en métricas como *balanced accuracy* o ROC AUC no garantiza una mejora clara de retorno sobre capital ni de eficiencia financiera.

En conjunto, los objetivos específicos se consideran alcanzados como hitos de construcción, integración, validación temporal y evaluación financiera. No obstante, este cumplimiento no implica que las estrategias dinámicas hayan demostrado una superioridad financiera robusta frente a referencias simples. La lectura final debe mantenerse en términos metodológicos: el sistema permite estudiar y evaluar la modulación de aportaciones DCA, aunque sus resultados financieros actuales no conduzcan a una conclusión plenamente favorable para las estrategias dinámicas.

8.4. Aportaciones del proyecto

La aportación principal del proyecto es metodológica. El trabajo conecta datos macroeconómicos, señales técnicas, aprendizaje supervisado, validación temporal y evaluación financiera dentro de un único pipeline. Esta integración permite estudiar la inversión periódica no solo como una regla fija de aportación, sino como un proceso susceptible de ser modulado en función del contexto, siempre bajo una evaluación empírica y prudente.

Una primera aportación es la arquitectura jerárquica macro-técnica. El analista macro no toma la decisión final de inversión, sino que proporciona contexto al analista técnico. Esta separación resulta útil porque evita atribuir a una única capa del sistema toda la responsabilidad predictiva. Además, permite que el bloque técnico utilice una señal macro generada de forma operativa, en lugar de incorporar directamente etiquetas macro futuras que no estarían disponibles en una situación real.

Una segunda aportación es la construcción de un target técnico vinculado a la decisión de aportación. El proyecto no intenta únicamente predecir si el NASDAQ100 subirá o bajará, sino que traduce retorno futuro y drawdown futuro en un score que representa la conveniencia relativa de aportar menos, aportar una cantidad base o aportar más. Este enfoque aproxima el problema de aprendizaje supervisado a la pregunta financiera que se desea responder: cuánto capital aportar en cada periodo dentro de una estrategia DCA.

Una tercera aportación es la comparación de formulaciones alternativas del bloque técnico. La memoria muestra que el baseline multiclase no resuelve de forma sólida la separación entre tres niveles de agresividad y que una reformulación binaria puede ser más prometedora para detectar meses de aportación agresiva. También analiza la regresión del score técnico como vía intermedia. Esta evolución evidencia una actitud experimental adecuada: no se fuerza una única formulación si los resultados sugieren limitaciones estructurales.

Una cuarta aportación es la conexión explícita entre métricas predictivas y métricas financieras. El backtesting evita concluir únicamente a partir de métricas de clasificación. Al transformar las predicciones en aportaciones reales, el proyecto comprueba si las señales tienen utilidad económica. Esta capa permite observar que una variante puede parecer mejor desde el punto de vista predictivo, pero no necesariamente superar a `dca_base` en retorno sobre capital o en robustez financiera.

Una quinta aportación es la incorporación de la eficiencia del capital extra como criterio interpretativo. Dado que algunas estrategias dinámicas invierten más capital que el DCA base, el PnL bruto o el valor final de cartera pueden inducir a conclusiones equivocadas. La métrica de eficiencia del capital extra ayuda a distinguir entre ganar más por decidir mejor y ganar más por haber invertido más capital. Esta distinción resulta especialmente relevante para evaluar estrategias de aportaciones variables.

En conjunto, el proyecto aporta una metodología reproducible y defendible para analizar estrategias DCA dinámicas. Su aportación no consiste en demostrar una estrategia superior, sino en construir un marco que permite formular, entrenar, comparar y auditar decisiones de aportación bajo criterios temporales y financieros explícitos.

8.5. Limitaciones generales del sistema propuesto

La primera limitación general es el tamaño muestral reducido, derivado de la frecuencia mensual. Aunque el periodo histórico cubre más de dos décadas, el número de observaciones efectivas sigue siendo bajo para entrenar y validar modelos supervisados. Esta restricción afecta tanto al bloque macroeconómico como al bloque técnico. Además, los horizontes futuros utilizados para construir los targets reducen el número de observaciones etiquetables, ya que las últimas filas no pueden etiquetarse sin usar información posterior al final de la serie.

La segunda limitación es la longitud del test limpio. El tramo final fuera de muestra cubre el periodo 2024-01 a 2025-07, con 19 meses. Aunque este test es útil para simular una evaluación posterior al entrenamiento, resulta demasiado corto para sostener conclusiones fuertes sobre superioridad financiera. Las pequeñas diferencias observadas entre estrategias dinámicas y `dca_base` pueden depender de características específicas de ese periodo.

La tercera limitación es que la evaluación se centra únicamente en el NASDAQ100. Este índice tiene una composición sectorial y un comportamiento histórico particulares, especialmente en periodos dominados por crecimiento tecnológico y fases alcistas prolongadas. Por ello, los resultados no pueden generalizarse automáticamente a otros índices, ETFs, sectores o carteras multi-activo. La evaluación walk-forward histórica también depende de un único recorrido temporal, lo que limita la capacidad de inferir robustez general.

La cuarta limitación es la dependencia del periodo histórico analizado. En el walk-forward, el periodo 2013-01 a 2025-07 fue favorable para el NASDAQ100. Este contexto beneficia especialmente a estrategias con exposición elevada o inversión temprana, como Buy & Hold. Por tanto, parte de los resultados puede estar condicionada por el régimen de mercado observado y no por una ventaja estructural de las señales.

La quinta limitación se encuentra en la construcción de los targets. Tanto el bloque macro como el bloque técnico emplean retorno y drawdown futuros para etiquetar observaciones.

Esta decisión es metodológicamente razonable para aprendizaje supervisado, pero implica que la calidad del modelo depende de cómo se defina la ventana futura, cómo se ponderen retorno y riesgo, y cómo se transformen esos valores en clases. En el bloque técnico, la discretización por percentiles genera fronteras que pueden ser sensibles a pequeños cambios en el score. La existencia de muchas observaciones cerca de los umbrales dificulta separar de forma robusta las clases $0.5x$, $1x$ y $2x$.

La sexta limitación son las métricas predictivas moderadas. El modelo macro final ofrece una formulación coherente como filtro de contexto, pero no debe sobreinterpretarse por una métrica puntual elevada en un test pequeño. El modelo técnico, por su parte, muestra dificultades para separar de forma estable los niveles de agresividad. La variante binaria agresiva mejora la lectura predictiva respecto al baseline multiclase, pero sus métricas no son suficientemente fuertes para sostener por sí solas una conclusión de utilidad financiera robusta.

La séptima limitación afecta al backtest. La simulación no incorpora costes de transacción, fiscalidad, restricciones de liquidez, retrasos operativos ni fricciones prácticas asociadas a modificar aportaciones periódicas. Tampoco incorpora preferencias individuales de riesgo, disponibilidad real de capital adicional o restricciones presupuestarias del inversor. Estas dimensiones podrían cambiar la conveniencia práctica de una estrategia que aumenta o reduce aportaciones en determinados meses.

La octava limitación es la diferencia de capital invertido entre estrategias. Las estrategias dinámicas pueden aportar más capital que `dca_base`. En consecuencia, comparar valor final de cartera o PnL bruto no es suficiente. Una estrategia puede terminar con más valor simplemente porque ha invertido más. Por este motivo, el análisis debe considerar retorno sobre capital, TIR, drawdown y eficiencia del capital extra.

La novena limitación es la ausencia de una prueba formal de significancia estadística. El proyecto compara resultados de backtesting y validación temporal, pero no demuestra que las diferencias observadas sean estadísticamente significativas. Por tanto, las conclusiones deben formularse como evidencia exploratoria y no como confirmación definitiva de superioridad.

Finalmente, Buy & Hold no es una estrategia equivalente a DCA. Su inclusión es útil como referencia de mercado, pero la comparación debe recordar su distinta estructura temporal de inversión. En consecuencia, el benchmark operativo más comparable para las estrategias dinámicas sigue siendo `dca_base`.

8.6. Líneas futuras

Las líneas futuras se derivan directamente de las limitaciones observadas. El objetivo de estas extensiones no debería ser aumentar la complejidad del sistema de forma indiscriminada, sino mejorar su robustez, evaluar su generalización y estudiar si la modulación de aportaciones puede aportar valor en condiciones más diversas. Cualquier mejora futura debería mantener la misma disciplina metodológica: evitar fuga de información, respetar la disponibilidad temporal de los datos, comparar frente a referencias simples y evaluar tanto métricas predictivas como métricas financieras.

8.6.1. Ampliación a frecuencia diaria o mayor granularidad temporal

Una primera línea futura consiste en ampliar el sistema a una frecuencia diaria, semanal o a una granularidad temporal superior a la mensual. La frecuencia mensual utilizada en este trabajo es coherente con una estrategia DCA de aportaciones periódicas y con la naturaleza de muchas variables macroeconómicas, pero limita el tamaño muestral disponible. Una frecuencia más alta podría aumentar el número de observaciones, mejorar la estimación de variables técnicas y permitir estudiar señales de corto plazo relacionadas con momentum, volatilidad, drawdown o recuperación tras caídas.

Esta ampliación debería diseñarse con cautela. Muchas series macroeconómicas no están disponibles a frecuencia diaria, por lo que habría que definir mecanismos de alineación temporal que respeten la fecha real de publicación de cada dato. Además, una mayor granularidad puede introducir más ruido, aumentar el riesgo de sobreajuste y hacer más relevantes los costes operativos. Si el sistema pasara a generar cambios frecuentes en las aportaciones, sería necesario evaluar costes de transacción, restricciones prácticas y estabilidad de las señales.

Una vía intermedia podría ser trabajar a frecuencia semanal para el bloque técnico, manteniendo el contexto macro a frecuencia mensual o actualizándolo solo cuando haya nueva información disponible. Esta solución permitiría aumentar la muestra sin convertir el sistema en una estrategia de trading diario. En cualquier caso, la ampliación temporal debería evaluarse mediante validación walk-forward y pruebas fuera de muestra adicionales.

8.6.2. Validación sobre otros índices o activos financieros

Una segunda línea futura consiste en validar el sistema sobre otros índices, ETFs, sectores o carteras multi-activo. La evaluación actual se centra en el NASDAQ100, por lo que no permite saber si la arquitectura macro-técnica generaliza a activos con distinta composición, volatilidad, sensibilidad macroeconómica o comportamiento cíclico. Probar el sistema sobre el S&P 500, Russell 2000, índices internacionales, ETFs sectoriales o combinaciones multi-activo permitiría analizar si las señales aprendidas son específicas del NASDAQ100 o si capturan patrones más generales.

Esta validación debería mantener una comparación clara frente a referencias simples. En cada activo o universo, sería necesario comparar las estrategias dinámicas frente a un DCA base equivalente y frente a un benchmark de mercado interpretado correctamente. También sería conveniente estudiar periodos con características distintas, incluyendo fases alcistas, bajistas, laterales y episodios de elevada volatilidad. De este modo se podría evaluar si el sistema solo funciona en un recorrido histórico favorable o si mantiene un comportamiento razonable en entornos de mercado más variados.

La extensión a carteras multi-activo también permitiría estudiar si la modulación de aportaciones puede combinarse con decisiones de diversificación. No obstante, esta línea aumentaría la complejidad del problema, ya que habría que separar la decisión de cuánto aportar de la decisión de dónde asignar el capital. Por ello, debería abordarse de forma gradual y con métricas de riesgo y eficiencia claramente definidas.

8.6.3. Incorporación de nuevas fuentes de información

Una tercera línea futura consiste en incorporar nuevas fuentes de información que complementen las variables macroeconómicas, financieras y técnicas utilizadas. Entre las fuentes potenciales se encuentran indicadores de sentimiento financiero, métricas de valoración, beneficios empresariales, volatilidad implícita, liquidez de mercado o datos alternativos. Estas variables podrían aportar información sobre expectativas, posicionamiento, tensión financiera o valoración relativa que no quede plenamente recogida en las señales actuales.

La incorporación de nuevas fuentes debe realizarse con criterios estrictos. Cada variable debería tener cobertura histórica suficiente, disponibilidad temporal verificable y una justificación económica clara. En series financieras, no basta con que una variable sea informativa *ex post*; es necesario que estuviera disponible en el momento en que el sistema habría tomado la decisión. De lo contrario, se introduciría riesgo de data leakage.

También debe evitarse que el aumento de variables deteriore la capacidad de generalización. Dado que el tamaño muestral mensual es reducido, añadir muchas columnas puede incrementar el sobreajuste. Por ello, cualquier ampliación de fuentes debería acompañarse de selección razonada de variables, validación temporal y comparación frente a versiones más simples del sistema. La mejora futura no debería medirse solo por métricas predictivas, sino por su efecto final sobre las estrategias DCA evaluadas mediante backtesting.

8.6.4. Optimización de la política de asignación de capital

Una cuarta línea futura consiste en optimizar la política de asignación de capital. En la versión actual, las predicciones se traducen en multiplicadores discretos 0.5x, 1x y 2x. Esta regla es sencilla, interpretable y adecuada para una primera evaluación, pero no necesariamente óptima desde el punto de vista financiero. Futuras versiones podrían estudiar otros multiplicadores, límites de capital extra, reglas de suavizado de señales, umbrales adaptativos o restricciones explícitas de drawdown.

Esta optimización debería incorporar el capital invertido como variable central. Una política que aporta más en muchos meses puede aumentar el valor final de cartera, pero no necesariamente mejorar el retorno sobre capital ni la eficiencia del capital extra. Por ello, los criterios de optimización deberían incluir retorno sobre capital, TIR, drawdown, estabilidad de las decisiones, simplicidad operativa y `capital_extra_efficiency`. También sería recomendable imponer límites al capital adicional para que la estrategia sea comparable y realista para distintos perfiles de inversor.

Otra posible mejora sería suavizar las decisiones mensuales para evitar cambios bruscos entre 0.5x, 1x y 2x. La salida de los modelos podría transformarse en una probabilidad o score continuo que modulase la aportación de manera gradual, siempre dentro de límites predefinidos. Esta aproximación podría reducir la sensibilidad a errores de clasificación cerca de los umbrales, aunque debería evaluarse cuidadosamente para no introducir complejidad innecesaria.

Finalmente, la optimización de capital debe validarse fuera de muestra y sobre más de un periodo histórico. Optimizar multiplicadores sobre el mismo backtest usado para evaluar la estrategia generaría riesgo de sobreajuste. Por tanto, cualquier política optimizada debería probarse con validación walk-forward, activos adicionales y periodos no utilizados durante el

diseño. Solo así podría analizarse si la mejora es robusta o si depende de ajustes específicos al pasado.

Como línea futura adicional, podría estudiarse una ampliación del sistema desde un DCA dinámico hacia una gestión dinámica de exposición. En la versión actual, el pipeline macro-técnico modula únicamente las nuevas aportaciones mensuales, pero no actúa sobre el capital ya invertido. Una extensión más ambiciosa podría incorporar reglas externas, acotadas y no aprendidas directamente por el modelo, orientadas a reducir parcialmente la exposición cuando se acumulen señales macro-técnicas negativas durante varios meses y a recomprar de forma gradual cuando se acumulen señales de recuperación.

Esta posibilidad debe interpretarse con prudencia, ya que se aleja del DCA tradicional y aumenta de forma significativa la complejidad operativa. Su evaluación exigiría una validación walk-forward específica, control estricto del sobreajuste, consideración de costes de transacción y fiscalidad, límites mínimos de exposición y métricas adicionales como rotación de cartera, tiempo fuera de mercado, TIR, drawdown y eficiencia del capital. No constituiría una mejora demostrada por los resultados actuales, sino una línea futura con potencial para analizar no solo la modulación de aportaciones, sino también la gestión parcial del capital ya desplegado.

Referencias bibliográficas

- Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.
<https://doi.org/10.1007/978-0-387-45528-0>
- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32.
<https://doi.org/10.1023/A:1010933404324>
- Campbell, J. Y., Lo, A. W., & MacKinlay, A. C. (1997). *The econometrics of financial markets*. Princeton University Press. <https://doi.org/10.1515/9781400830213>
- Constantinides, G. M. (1979). A note on the suboptimality of dollar-cost averaging as an investment policy. *The Journal of Financial and Quantitative Analysis*, 14(2), 443–450. <https://doi.org/10.2307/2330513>
- Dichtl, H., & Drobetz, W. (2011). Dollar-cost averaging and prospect theory investors: An explanation for a popular investment strategy. *Journal of Behavioral Finance*, 12(1), 41–52. <https://doi.org/10.1080/15427560.2011.555029>
- Dixon, M. F., Halperin, I., & Bilokon, P. (2020). *Machine learning in finance: From theory to practice*. Springer Cham. <https://doi.org/10.1007/978-3-030-41068-1>
- Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM*, 55(10), 78–87. <https://doi.org/10.1145/2347736.2347755>
- Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8), 861–874. <https://doi.org/10.1016/j.patrec.2005.10.010>
- Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning*, 63(1), 3–42. <https://doi.org/10.1007/s10994-006-6226-1>
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed.). Springer. <https://doi.org/10.1007/978-0-387-84858-7>
- Hoerl, A. E., & Kennard, R. W. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1), 55–67.
<https://doi.org/10.1080/00401706.1970.10488634>
- Hosmer, D. W., Jr., Lemeshow, S., & Sturdivant, R. X. (2013). *Applied logistic regression* (3rd ed.). John Wiley & Sons. <https://doi.org/10.1002/9781118548387>
- Hyndman, R. J., & Athanasopoulos, G. (2021). *Forecasting: Principles and practice* (3rd ed.). OTexts. <https://otexts.com/fpp3/>

- Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4), 679–688.
<https://doi.org/10.1016/j.ijforecast.2006.03.001>
- Kaufman, S., Rosset, S., Perlich, C., & Stitelman, O. (2012). Leakage in data mining: Formulation, detection, and avoidance. *ACM Transactions on Knowledge Discovery from Data*, 6(4), Article 15. <https://doi.org/10.1145/2382577.2382579>
- Leggio, K. B., & Lien, D. (2003). An empirical examination of the effectiveness of dollar-cost averaging using downside risk performance measures. *Journal of Economics and Finance*, 27(2), 211–223. <https://doi.org/10.1007/BF02827219>
- López de Prado, M. (2018). *Advances in financial machine learning*. John Wiley & Sons.
- Magdon-Ismail, M., & Atiya, A. F. (2004). Maximum drawdown. *Risk*, 17(10), 99–102.
<https://ssrn.com/abstract=874069>
- Markowitz, H. (1952). Portfolio selection. *The Journal of Finance*, 7(1), 77–91.
<https://doi.org/10.1111/j.1540-6261.1952.tb01525.x>
- Nasdaq, Inc. (n.d.). *Nasdaq-100 Index overview*. Recuperado el 8 de junio de 2026, de <https://indexes.nasdaq.com/Index/Overview/NDX>
- Nasdaq, Inc. (2026). *Nasdaq-100 Index Methodology*.
https://indexes.nasdaq.com/docs/Methodology_NDX.pdf
- Powers, D. M. W. (2011). Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, 2(1), 37–63.
- Qian, E. E., Hua, R. H., & Sorensen, E. H. (2007). *Quantitative equity portfolio management: Modern techniques and applications*. Chapman & Hall/CRC.
- Rozeff, M. S. (1994). Lump-sum investing versus dollar-averaging. *The Journal of Portfolio Management*, 20(2), 45–50. <https://doi.org/10.3905/jpm.1994.409474>
- scikit-learn developers. (n.d.-a). *Model evaluation: Quantifying the quality of predictions*. Recuperado el 8 de junio de 2026, de https://scikit-learn.org/stable/modules/model_evaluation.html
- scikit-learn developers. (n.d.-b). *Regression metrics*. Recuperado el 8 de junio de 2026, de https://scikit-learn.org/stable/modules/model_evaluation.html#regression-metrics
- Sharpe, W. F. (1994). The Sharpe ratio. *The Journal of Portfolio Management*, 21(1), 49–58.
<https://doi.org/10.3905/jpm.1994.409501>

Tsay, R. S. (2010). *Analysis of financial time series* (3rd ed.). John Wiley & Sons.
<https://doi.org/10.1002/9780470644560>

Anexo A. Estructura del proyecto y artefactos entregados

Este anexo resume la estructura principal del repositorio entregado junto con la memoria. Se ha simplificado el árbol de carpetas para mostrar únicamente los elementos relevantes para la trazabilidad y reproducibilidad del pipeline macro-técnico. Se omiten carpetas auxiliares de exportaciones automáticas de notebooks y ficheros intermedios no necesarios para comprender el flujo final.

```
IA TFM - Sistema Inteligente de optimizacion DCA/
|-- Analista Macro/
|   |-- *.csv                      Datos macro y de mercado originales
|   |-- OCDE (G20)/                Datos auxiliares del CLI G20 de la OCDE
|   |-- SCRIPTS/
|       |-- 01_extract_features.ipynb  Extraccion de features macro
|       |-- 02_build_master_dataset_*  Construccion y evolucion del dataset macro
|       |-- 03_train_macro_model_*    Entrenamiento macro y cierre final
|       |-- 99_apoyo.ipynb            Utilidad auxiliar de inspeccion
|       |-- features/                 Features macro por indicador
|       |-- master_data/              Datasets maestros macro
|       |-- models/                   Modelos macro entrenados y metadatos
|       |-- reports/                  Resultados, comparativas y predicciones macro
|-- Analista Tecnico/
|   |-- SCRIPTS/
|       |-- 04_extract_features_tecnico.ipynb
|       |-- 05_build_dataset_tecnico.ipynb
|       |-- 06_train_model_tecnico.ipynb
|       |-- 07_experimentos_clasificacion_tecnico.ipynb
|       |-- 08_regresion_score_tecnico.ipynb
|       |-- 09_backtest_tecnico.ipynb
|       |-- 10_evaluacion_final_sistema.ipynb
|       |-- features_tecnico/          Features tecnicas del NASDAQ
|       |-- dataset_tecnico/           Dataset tecnico final etiquetado
|       |-- models/                   Modelo tecnico base y metadatos
|       |-- reports/                  Reportes tecnicos, backtests y tablas finales
```

La carpeta del analista macroeconómico agrupa la extracción de variables, la construcción del dataset macro, el entrenamiento del modelo macro final y los reportes asociados. La carpeta del analista técnico agrupa la generación de señales técnicas, la construcción del dataset técnico, los experimentos de clasificación y regresión, el backtesting y la evaluación final del sistema. Los artefactos principales entregados incluyen notebooks, datasets CSV, modelos serializados, metadatos JSON y reportes de resultados.