PONTIFICIA UNIVERSIDAD CATÓLICA DEL
ECUADOR FACULTAD DE HÁBITAT,
INFRAESTRUCTURA Y CREATIVIDAD

INGENIERÍA EN SISTEMAS DE INFORMACIÓN



Trabajo de Titulación

Maximización de Rentabilidad Ajustada por Riesgo Sistémico en Portafolios de ETFs mediante Técnicas de Trading Algorítmico.

AUTOR:

SOSA LOROÑA ANTHONY SEBASTIAN

TUTOR:

FRANCISCO RODRÍGUEZ CLAVIJO

QUITO DM, JUNIO 2026

DEDICATORIA

A quienes confiaron en mí incluso cuando yo dudaba, y a la constancia que me sostuvo para no abandonar este camino a mitad de recorrido.

AGRADECIMIENTO

Este trabajo no se lo dedico a una sola persona sino a algo que me tocó aprender a la fuerza durante estos años, la resiliencia. La ingeniería de sistemas no fue la carrera con la que soñé y hubo momentos en los que sentí que no era mi lugar, pero con el tiempo entendí que igual me dejó algo muy valioso, me enseñó a pensar de forma ordenada y a no rendirme cuando algo no sale a la primera.

Lo curioso es que fue justamente esa formación la que me terminó acercando a lo que de verdad me apasiona, que son las finanzas y los mercados. Este proyecto es la prueba de que las dos cosas se podían unir, porque pude tomar lo que aprendí programando y usarlo para algo que sí me emociona.

Agradezco también a mi tutor por la paciencia y por cada corrección que me hizo ver lo que yo no estaba viendo. Si algo me llevo de todo esto es que a veces el camino que uno no eligió termina enseñándole más de lo que esperaba.

RESUMEN

Este trabajo presenta el desarrollo de un sistema de trading algorítmico que busca maximizar la rentabilidad ajustada por riesgo sobre un portafolio de seis ETFs del mercado estadounidense (SPY, IWM, QQQ, DIA, XLF y XLY). El problema que motiva el proyecto es que operar a mano se vuelve difícil cuando hay que analizar varios activos al mismo tiempo, y que muchas estrategias se quedan solo en el retorno sin medir el riesgo ni validarse antes de arriesgar dinero real. Para resolverlo se construyó el sistema siguiendo la metodología Extreme Programming, avanzando en tres iteraciones que fueron sumando un filtro de tendencia con la media de 200 días, la selección de activos por momentum y, en la versión final, la gestión de riesgo con un stop dinámico tipo Chandelier Exit basado en el ATR junto con la ponderación por volatilidad inversa. La validación se hizo por backtesting en QuantConnect sobre el período 2013 a 2024. La versión final alcanzó un retorno anual compuesto del 10.20%, un ratio de Sharpe de 0.607, un drawdown máximo del 13.4% y un Alpha de Jensen de 0.027, mejorando a las dos iteraciones anteriores y a la estrategia de comprar y mantener.

Palabras clave: trading algorítmico, momentum, gestión de riesgo, ATR, Chandelier Exit, Extreme Programming, ETFs, backtesting.

Contenido

CAPÍTULO 1 – INTRODUCCIÓN 9

1.1 TEMA 9

1.2 JUSTIFICACIÓN 9

1.3 PLANTEAMIENTO DEL PROBLEMA 10

1.4 OBJETIVOS 10

1.4.1 Objetivo General 10

1.4.2 Objetivos Específicos 11

1.5 ALCANCE Y LIMITACIONES 11

1.6 ESTRUCTURA DEL DOCUMENTO 12

CAPÍTULO 2 – MARCO TEÓRICO Y CONCEPTUAL 13

2.1 TRADING ALGORÍTMICO Y MERCADOS FINANCIEROS 13

2.1.1 Conceptos básicos de trading algorítmico 13

2.1.2 Backtesting y validación de estrategias cuantitativas 13

2.2 ESTRATEGIAS DE TRADING POR MOMENTUM 14
2.2.1 Efecto momentum en los mercados financieros 14
2.2.2 Indicadores de momentum y selección de activos 14
2.3 GESTIÓN DE RIESGO EN TRADING CUANTITATIVO 15
2.3.1 Dimensionamiento de posiciones basado en riesgo porcentual 15
2.3.2 Métricas de rendimiento ajustado por riesgo 15
2.4 TRAILING STOPS Y VOLATILIDAD 15
2.4.1 Indicador Average True Range (ATR) 15
2.4.2 Trailing stops fijos vs. trailing stops adaptativos 15
2.5 METODOLOGÍAS DE DESARROLLO DE SOFTWARE 16
2.5.1 Enfoques tradicionales y metodologías ágiles 16
2.5.2 Extreme Programming (XP): principios y prácticas 16
2.5.3 Plataformas para trading algorítmico (QuantConnect/LEAN) 16
CAPÍTULO 3 – METODOLOGÍA DE DESARROLLO (XP) 16
3.1 JUSTIFICACIÓN DE LA ELECCIÓN DE XP 16
3.2 ADAPTACIÓN DE XP AL DESARROLLO DE SISTEMAS DE TRADING ALGORÍTMICO 17
3.3 HISTORIAS DE USUARIO DEL SISTEMA DE TRADING 17
3.4 PLANIFICACIÓN DE ITERACIONES Y CRONOGRAMA 18
3.5 PRÁCTICAS APLICADAS: TDD, REFACTORIZACIÓN Y DISEÑO SIMPLE 18
3.5.1 Test-Driven Development (TDD) 18
3.5.2 Refactorización continua 19
3.5.3 Diseño simple 19
3.6 HERRAMIENTAS Y ENTORNO DE TRABAJO 19
CAPÍTULO 4 – DISEÑO DEL SISTEMA DE TRADING 19
4.1 Requisitos Funcionales y No Funcionales 19
4.1.1 Requisitos Funcionales 20
4.1.2 Requisitos No Funcionales 20
4.2 Arquitectura General del Sistema 21
4.2.1 Visión General de la Arquitectura 21
4.2.2 Componentes Principales 22
4.3 Modelo Lógico de la Estrategia 22
4.3.1 Filtro de Régimen de Mercado 22
4.3.2 Cálculo y Ranking de Momentum 23
4.3.3 Reglas de Selección y Ponderación 24
4.4 Módulo de Gestión de Riesgo 24
4.4.1 Stop-Loss Dinámico basado en ATR 24

4.4.2 Cooldown Post-Stop y Control de Exposición 25
4.5 Diseño del Stop-Loss Dinámico Basado en ATR 26
4.5.1 Cálculo del ATR y Distancia Base 26
4.5.2 Comparación con Stops Fijos y Distinción Frente a Trailing Stops 26
4.6 Pseudocódigo de la Lógica del Algoritmo 27
CAPÍTULO 5 – IMPLEMENTACIÓN ITERATIVA DEL ALGORITMO 29
5.1 Estructura de Clases y Componentes en QuantConnect 29
5.2 Iteración 1: Filtro de Régimen de Mercado (SMA200) 29
5.2.1 Objetivo e Historias de Usuario Cubiertas 29
5.2.2 Implementación 30
5.2.3 Resultados de la Iteración 1 30
5.3 Iteración 2: Módulo de Momentum y Selección de Activos 31
5.3.1 Objetivo e Historias de Usuario Cubiertas 31
5.3.2 Implementación 32
5.3.3 Resultados de la Iteración 2 32
5.4 Iteración 3: Stop-Loss Dinámico Basado en ATR 33
5.4.1 Objetivo e Historias de Usuario Cubiertas 33
5.4.2 Implementación 34
5.4.3 Resultados de la Iteración 3 36
5.5 Desarrollo Guiado por Pruebas (TDD) y Refactorización 37
5.5.1 Ciclo TDD Aplicado 37
5.5.2 Refactorización Entre Iteraciones 37
5.6 Problemas Encontrados y Decisiones de Diseño 38
5.6.1 Problema: Error de tipo en History() 38
5.6.2 Problema: Conflicto entre Señal de Stop y Señal de Momentum 38
5.6.3 Decisión: Parámetros Fijos vs. Optimizados 38
CAPÍTULO 6 – BACKTESTING Y EVALUACIÓN DE RESULTADOS 39
6.1 Diseño del Experimento de Backtesting 39
6.1.1 Período de Prueba y Datos Utilizados 39
6.1.2 Parámetros Iniciales del Backtest 40
6.2 Métricas de Evaluación 40
6.3 Resultados de la Estrategia Propuesta 41
6.3.1 Evolución Temporal del Capital 41
6.3.2 Distribución de Operaciones 42
6.4 Comparación con Estrategias de Referencia 42
6.4.1 Comparación con Buy & Hold SPY 43

6.4.2 Comparación entre Iteraciones 44
6.4.3 Validación Fuera de Muestra (2022-2024) 45
6.5 Análisis Crítico de Resultados y Limitaciones 46
CAPÍTULO 7 – CONCLUSIONES Y TRABAJOS FUTUROS 48
7.1 Conclusiones 48
7.2 Limitaciones del Estudio 49
7.3 Recomendaciones para Mejoras de la Estrategia 49
REFERENCIAS BIBLIOGRÁFICAS 50
ANEXOS 51
Anexo A: Tabla de Historias de Usuario 51
Anexo B: Plan de Iteraciones XP (Cronograma) 54
Anexo C: Código Fuente Completo del Algoritmo 55
Anexo D: Registro Completo de Operaciones del Backtest 58

ÍNDICE DE FIGURAS

Figura 5.2.1 – Código fuente de la Iteración 1: filtro SMA200 en QuantConnect 30
Figura 5.2.2 – Curva de capital de la Iteración 1 (2013-2024)..... 31
Figura 5.2.3 – Métricas del backtest de la Iteración 1..... 31
Figura 5.3.1 – Código fuente de la Iteración 2: selección por momentum en QuantConnect..... 32
Figura 5.3.2 – Curva de capital de la Iteración 2 (2013-2024)..... 33
Figura 5.3.3 – Métricas del backtest de la Iteración 2..... 33
Figura 5.4.1 – Código fuente de la Iteración 3 (versión final): Initialize() con ATR 35
Figura 5.4.2 – Código fuente de la Iteración 3 (versión final): OnData() con stop-loss ATR 35
Figura 5.4.3 – Curva de capital de la Iteración 3 (2013-2024)..... 36
Figura 5.4.4 – Métricas del backtest de la Iteración 3..... 37
Figura 6.1 – Curva de capital de la estrategia Buy & Hold del SPY (2013-2024) 44
Figura 6.2 – Métricas del backtest Buy & Hold del SPY..... 44
Figura 6.3 – Curva de capital de la validación fuera de muestra (2022-2024) 46
Figura B.1 – Cronograma del plan de iteraciones XP 55

ÍNDICE DE TABLAS

Tabla 1 – Análisis de sensibilidad del multiplicador del ATR (backtest 2013-2024)..... 25
Tabla 2 – Parámetros iniciales del backtest 40
Tabla 3 – Comparación de métricas entre las tres iteraciones 42
Tabla 4 – Métricas de la validación fuera de muestra (Iteración 3, 2022-2024)..... 45
Tabla A.1 – Historias de usuario del sistema 51

CAPÍTULO 1 – INTRODUCCIÓN

1.1 TEMA

Maximización de Rentabilidad Ajustada por Riesgo Sistémico en Portafolios de ETFs mediante Técnicas de Trading Algorítmico.

1.2 JUSTIFICACIÓN

Las decisiones de inversión tomadas a mano se vuelven cada vez más difíciles de sostener cuando el mercado se mueve rápido y la información llega desde múltiples frentes al mismo tiempo. Un sistema algorítmico ayuda a quitar de la ecuación el componente emocional y a ejecutar reglas claras, aunque para programarlo bien hace falta apoyarse en una metodología que cuide la calidad del código y permita probarlo con suficiente rigor antes de exponer capital real.

Buscamos rentabilidad sin descuidar el riesgo, protegiendo el capital de varias maneras y combinando indicadores. Cada regla se fija sobre un indicador conocido, sabiendo de antemano para qué puede servir, y de esa combinación salen mejores resultados que usando uno solo.

La metodología XP nos permite hacer pruebas aun sin tener la idea completa, evaluando distintos algoritmos antes de llegar al resultado final. Con varias iteraciones se va afinando un algoritmo que funcione sobre los distintos ETFs que vamos a usar.

Nos interesa saber qué tan bien vienen los activos para posicionar primero a los mejores. El control de riesgo nos deja cerrar posiciones a favor, porque el stop no es fijo y acompaña al precio cuando hay ganancia, y cuando las cosas se ponen feas cierra la

posición antes de un descenso fuerte del mercado.

Dentro de Ingeniería de Sistemas, el proyecto sirve para demostrar cómo, a partir del análisis de datos en Python y de una plataforma que trabaja con APIs, se puede correr un algoritmo que automatiza un proceso que antes hacía a mano el usuario, ahorrándole tiempo y mejorando la forma en que maneja sus activos.

1.3 PLANTEAMIENTO DEL PROBLEMA

Para una persona, operar por su cuenta es cada vez más difícil, porque no alcanza a analizar todos los activos ni toda la información en tan poco tiempo y termina perdiendo datos importantes que no están a la vista de forma simple.

Mucha gente conoce el trading algorítmico, pero no hace backtest antes de comprobar si la estrategia va a ser rentable en el tiempo. Se fijan solo en el retorno y dejan de lado las demás métricas, y ese error lo descubren cuando ya hay dinero en juego, cuando suele ser tarde.

Se puede entrar al mercado con el momentum, pero proteger esa posición con un stop fijo ignora la volatilidad que pueda aparecer, y eso es peligroso porque deja el capital expuesto a caídas grandes.

El ATR muchas veces se usa con valores fijos y, por sí solo, no se adapta, y usado así sus resultados terminan dependiendo demasiado de la suerte del período en que toque aplicarlo.

¿Es posible, aplicando XP, construir una estrategia que parta del indicador de momentum y le sume un stop dinámico, de modo que al combinar ambos se obtenga un buen rendimiento ajustado por riesgo, con código probado y validado siguiendo la metodología planteada?

1.4 OBJETIVOS

1.4.1 Objetivo General

Construir, siguiendo la metodología XP, un sistema de trading algorítmico que maximice el Alpha (rentabilidad ajustada por riesgo sistémico) combinando selección de activos por momentum, ponderación por volatilidad inversa y Chandelier Exit basado en ATR, validando su comportamiento mediante backtesting sobre ETFs estadounidenses

en el período 2013-2024.

1.4.2 Objetivos Específicos

1. Estudiar los fundamentos del trading algorítmico, las estrategias de momentum y las técnicas de control de riesgo apoyadas en indicadores técnicos (SMA, ATR, stop-loss dinámico) para construir el marco conceptual y derivar las historias de usuario que guiarán el desarrollo XP.
2. Diseñar e implementar el algoritmo de forma iterativa e incremental, aplicando las prácticas centrales de XP (TDD, refactor, diseño simple) sobre cada componente, como el filtro de régimen, el ranking por momentum, la asignación de capital y el stop-loss dinámico basado en ATR.
3. Validar el rendimiento del sistema mediante backtesting sobre datos históricos de ETFs del mercado estadounidense (2013-2024), calculando métricas de desempeño (rentabilidad, drawdown máximo, ratio Sharpe, win rate) y comparando con la estrategia de referencia buy & hold.
4. Documentar el proceso de desarrollo bajo metodología XP, identificando lecciones aprendidas en la aplicación de prácticas ágiles a sistemas de trading algorítmico.

1.5 ALCANCE Y LIMITACIONES

El proyecto abarca el análisis, el diseño, el desarrollo y la validación por backtesting de un sistema de trading basado en momentum con gestión de riesgo dinámica, desarrollado con metodología XP sobre QuantConnect y su motor LEAN.

Las prácticas de XP aplicadas fueron TDD, refactorización, iteraciones cortas y diseño simple. Los componentes base de la estrategia son el filtro SMA200, la selección por momentum, la asignación de capital y el stop dinámico por ATR.

El backtesting se hace sobre un universo de 6 ETFs (SPY, QQQ, IWM, XLF, XLY y DIA) en el período 2013-2024, con métricas como el Sharpe, el drawdown, el win rate y la rentabilidad, comparadas contra buy & hold. Tanto los resultados positivos como los negativos son válidos para el objetivo del estudio.

El proyecto NO contempla:

- Ejecución de algoritmos con capital real en cuentas de trading en vivo.
- Conexión a brokers externos mediante APIs para apertura de operaciones en vivo

(Interactive Brokers, Alpaca, TD Ameritrade).

- Desarrollo de aplicativos webs, dashboards interactivos o interfaces gráficas.
- Optimización de parámetros mediante algoritmos genéticos, walk-forward analysis o análisis de sensibilidad complejo.
- Trading de activos distintos a los 6 ETFs del mercado estadounidense seleccionados.
- Análisis de datos de alta frecuencia o estrategias intraday.
- Implementación en entornos de producción o gestión de portafolios de terceros.

1.6 ESTRUCTURA DEL DOCUMENTO

El presente trabajo se organiza en siete capítulos:

El Capítulo 1 (Introducción) presenta el tema, la justificación, el planteamiento del problema, los objetivos, el alcance y la estructura general del documento.

El Capítulo 2 (Marco Teórico y Conceptual) desarrolla los fundamentos del trading algorítmico, el efecto momentum, la gestión de riesgo basada en volatilidad, el Chandelier Exit con ATR y los principios de XP.

El Capítulo 3 (Metodología de Desarrollo – XP) describe la adaptación de XP al contexto del trading algorítmico, las historias de usuario y la planificación de las iteraciones.

El Capítulo 4 (Diseño del Sistema) presenta la arquitectura, requisitos y modelo lógico del algoritmo.

El Capítulo 5 (Implementación Iterativa) documenta las tres iteraciones: filtro SMA200, selección por momentum y Chandelier Exit basado en ATR, con los resultados de pruebas TDD de cada módulo.

El Capítulo 6 (Backtesting y Evaluación de Resultados) presenta los resultados de la simulación histórica sobre los 6 ETFs (2013-2024) con las métricas de rendimiento ajustado por riesgo.

El Capítulo 7 (Conclusiones y Trabajos Futuros) recoge las conclusiones del trabajo, las limitaciones identificadas y las recomendaciones para futuras mejoras de la estrategia.

CAPÍTULO 2 – MARCO TEÓRICO Y CONCEPTUAL

2.1 TRADING ALGORÍTMICO Y MERCADOS FINANCIEROS

2.1.1 Conceptos básicos de trading algorítmico

El trading algorítmico puede entenderse como una computadora ejecutando reglas de compra y venta sin que el humano tenga que decidir en cada operación. Eso es importante porque cuando se ejecuta un trade así no entra ninguna emoción de por medio y todo se rige con base en estadísticas. En cambio en el trading discrecional es el humano el que decide y ahí el problema es que puede ser poco objetivo.

Un sistema de estos se puede dividir en tres partes (Schwager, 2020). La primera es generar las señales, es decir, cuándo entrar y cuándo salir, que para nosotros es fundamental porque tenemos que cubrirnos en casos donde el mercado se pone muy volátil. La segunda es gestionar el tamaño de la posición y esto importa porque a partir de ahí yo sé cuánto arriesgar y no va a ser todo mi capital sino un porcentaje. La tercera es ejecutar donde se mandan las órdenes, y aquí hay que tener muy presentes las comisiones porque si la estrategia es pequeña se la pueden comer y si es grande permiten ganar mucho más, ya que el peso de las comisiones se vuelve menos perceptible cuando hay más dinero.

En nuestro proyecto trabajamos con seis ETFs, los datos que se consiguen son diarios, se hace un rebalanceo semanal y manejamos un capital base de 100,000 dólares.

2.1.2 Backtesting y validación de estrategias cuantitativas

El backtesting es muy importante para nosotros porque nos permite saber cómo se habría comportado la estrategia en el pasado sin arriesgar todavía dinero real. Con base en los indicadores tomamos los datos históricos, trabajamos sobre ellos y vemos qué estrategia habría salido mejor.

Ahora el backtesting tiene sus trampas que hay que reconocer. El overfitting es cuando el algoritmo queda sobreentrenado, se ajusta tanto al pasado que se vuelve poco realista y deja de servir cuando el mercado cambia, por eso no lo queremos tan agresivo sino que se adapte a las tendencias y nos proteja cuando son muy bruscas. El sesgo de supervivencia aparece cuando solo se analizan los activos que siguen vivos hoy y se ignoran los que quebraron, lo que hace que los resultados se vean mejores de lo que en

realidad fueron. Y el lookahead bias ocurre cuando el algoritmo usa información que en el momento real todavía no existía.

Para no caer en eso los parámetros los fijamos de antemano con base en la literatura y no los anduvimos optimizando sobre los datos. Usamos el ATR de 14 días, el multiplicador de 2 y los 63 días del lookback que sirven en la fórmula del momentum para ver durante cuánto tiempo ha venido creciendo o cayendo un activo. Además QuantConnect respeta el orden cronológico de los datos y aplica las comisiones reales en cada operación.

2.2 ESTRATEGIAS DE TRADING POR MOMENTUM

2.2.1 Efecto momentum en los mercados financieros

El efecto momentum según se entiende es que si un activo ha estado subiendo durante un tiempo tiende a seguir subiendo durante ese mismo período. Como ya sabemos que puede seguir subiendo eso nos ayuda a reconocer qué tendencia está marcando. Este comportamiento lo documentaron Jegadeesh y Titman (1993).

La razón de esto es que las masas se contagian, es decir que los inversores reaccionan lento a la información cuando en realidad hay indicadores que podrían avisarles mucho más rápido. Al contagiarse unos a otros terminan moviéndose en manada y es justamente esa inercia la que el momentum aprovecha. En nuestro caso revisamos durante 63 días cómo les ha ido a los ETFs en los últimos tres meses y con eso vemos si el momentum viene positivo o negativo.

2.2.2 Indicadores de momentum y selección de activos

La forma en que medimos el momentum es con el retorno, que es el precio de hoy dividido para el precio de hace 63 días menos uno. Con ese número podemos comparar todos los ETFs que participan y armar un top-N con los mejores.

Esto es esencial porque nos permite repartir el capital entre varios activos y no concentrarlo en uno solo. Al repartir reducimos muchísimo nuestro riesgo, porque si uno de los ETFs elegidos llega a fallar los otros pueden compensar.

2.3 GESTIÓN DE RIESGO EN TRADING CUANTITATIVO

2.3.1 Dimensionamiento de posiciones basado en riesgo porcentual

Decidir cuánto invertir en cada operación importa tanto o más que decidir qué comprar, y una forma clásica es arriesgar un porcentaje fijo del capital por operación, idea que popularizó Vince en 2009. En nuestro sistema usamos la ponderación por volatilidad inversa, donde a cada activo le damos un peso según 1 sobre su ATR y lo normalizamos para que sume el cien por ciento. Eso nos da como resultado menos volatilidad, más peso a los activos menos volátiles y una mejora significativa en el Sharpe.

2.3.2 Métricas de rendimiento ajustado por riesgo

No únicamente se visualiza el retorno, ya que esa métrica sin acompañarla del riesgo nos puede engañar. Por eso usamos cuatro métricas muy importantes, el Sharpe que es nuestro retorno por cada unidad que arriesgamos, el drawdown máximo que es la peor caída desde un pico, el win rate que es el porcentaje de operaciones ganadoras, y la rentabilidad acumulada. Con eso podemos comparar nuestro sistema contra el buy and hold del SPY.

2.4 TRAILING STOPS Y VOLATILIDAD

2.4.1 Indicador Average True Range (ATR)

El indicador ATR lo entendemos como cuánto se ha movido el precio, no la dirección para nada, así que a través del ATR tenemos la volatilidad. Este indicador lo propuso Wilder en 1978, y para calcularlo se toma un período de 14 días sacando el promedio del rango de cada día.

2.4.2 Trailing stops fijos vs. trailing stops adaptativos

En el trailing stop fijo versus adaptativo, el trailing stop es lo que se va a mover cuando el precio sube, bloqueando aunque sea de forma parcial una ganancia. El mecanismo que usamos es el Chandelier Exit, con la fórmula del máximo alcanzado menos dos veces el ATR. Quien respalda esto es Kaminski y Lo, que muestran que así se controlan mejor los drawdown.

2.5 METODOLOGÍAS DE DESARROLLO DE SOFTWARE

2.5.1 Enfoques tradicionales y metodologías ágiles

En los enfoques tradicionales tenemos la cascada, que como sabemos es muy estricta si los requisitos cambian y no nos permite ir refactorizando a través de varias iteraciones. En cambio el enfoque ágil permite no seguir el proyecto en una sola línea, y su base está en el Manifiesto Ágil de 2001.

2.5.2 Extreme Programming (XP): principios y prácticas

Kent Beck propuso XP en 1999 para equipos que tienden a tener requisitos que cambian. Dentro de esta metodología se manejan cinco valores, la comunicación, la simplicidad, la retroalimentación, el coraje y el respeto. A diferencia de Scrum, en XP se trabaja con TDD, refactor y diseño simple.

2.5.3 Plataformas para trading algorítmico (QuantConnect/LEAN)

QuantConnect es la plataforma web que tiene como motor a LEAN Engine, que es de código abierto y se puede correr con Python o C#. Maneja en automático la sincronización de las fechas, los splits y dividendos, y las comisiones. Nos permite usar los datos diarios de los 6 ETFs dentro del período de 2013 a 2024.

CAPÍTULO 3 – METODOLOGÍA DE DESARROLLO (XP)

3.1 JUSTIFICACIÓN DE LA ELECCIÓN DE XP

Hacer un sistema de trading tiene ya varias cosas importantes y no llega a ser algo tan simple como una fórmula matemática. Hay que diseñar, programar, probar y mantener con seriedad, porque si ocurre algún error todo se podría dañar, así que un error aquí es muy clave. Por eso tenemos que entender que elegir bien la metodología es algo bastante importante, ya que afecta la calidad y la confiabilidad del sistema que desarrollamos.

Ahora, por qué encaja con Ingeniería de Sistemas. Si lo vemos, esto es un desarrollo de software con requisitos que pueden ir cambiando con el tiempo e incluso mejorando. Pero lo que se entrega no es un documento teórico sino un programa que va a procesar datos, y una vez que los procesa toma decisiones sobre ellos, y con esas decisiones maneja el capital. Entonces llega a ser realmente un sistema completo de ingeniería.

Tenemos varias razones por las que elegimos XP. La primera es que XP está pensado para equipos pequeños, entonces como una sola persona podemos manejarlo técnicamente y a la vez ir viendo cómo se maneja lo financiero. La segunda es que los requisitos al inicio no los teníamos fijos, pero los fuimos testeando, como el filtro SMA de 200 días, el cambio de los distintos valores del multiplicador del ATR y qué ETFs usar o no, entonces todo eso salió de lo que iba mostrando cada test. La tercera es que el TDD hace que cada parte quede probada antes de juntarla con las demás, lo cual es clave cuando un bug cuesta dinero. Y la cuarta es mantener el código legible y mantenible, sin sobrecomplicarlo con muchos indicadores y dejando solo los necesarios, para que se pueda ver, auditar e incluso replicar.

3.2 ADAPTACIÓN DE XP AL DESARROLLO DE SISTEMAS DE TRADING ALGORÍTMICO

Lo que tenemos que ver en esta parte es cómo se adapta XP. La primera cosa es que la idea de varias personas trabajando juntas, como aquí soy una sola persona, la tengo que adaptar de otra manera. Lo que hago es que yo realizo el código y yo mismo lo reviso después, entonces soy tanto el que escribe como el que revisa. La programación en pares la cambié por revisarme yo mismo, poniéndome como una persona externa que busca fallas antes de unir todo.

Otra idea importante es que tenemos varios requisitos. El sistema tiene que ser reproducible, transparente, documentado y mantenerse dentro del capital y los ETFs que están definidos. Como hay alguien que requiere varias cosas, esos requisitos vienen a ser nuestro cliente.

También están las tres fases en las que se construye esto, que es bastante importante porque cada una termina con una validación que nos sirve para después seguir avanzando a las siguientes fases.

Y la última idea que me parece interesante es que cada vez que pruebo los algoritmos, QuantConnect guarda en la nube el historial del último algoritmo que se usó, entonces siempre tenemos un punto al que podemos volver dentro del proyecto.

3.3 HISTORIAS DE USUARIO DEL SISTEMA DE TRADING

Para entender qué son las historias de usuario debemos tener claro que se cuentan en un lenguaje simple, porque muestran cómo se usa el sistema desde el punto de vista de algún usuario. Aquí el usuario es quien dirige su portafolio y deja que el algoritmo tome

las decisiones de inversión. Cada historia de usuario es el punto desde donde empezamos a definir las pruebas de cada iteración, y con eso sabemos qué alcance iba a tener cada módulo. Después tenemos una lista con las historias, como la HU-01, la HU-02 y las demás, que se detallan en la tabla del Anexo A.

3.4 PLANIFICACIÓN DE ITERACIONES Y CRONOGRAMA

El proyecto se desarrolló en tres fases o iteraciones, más una fase cero de preparación. Cada fase tiene un objetivo, y a través de las historias de usuario tiene un criterio previo, de manera que cada historia nos deje una retroalimentación. Esa es la idea de XP, que con ciclos cortos siempre tengamos algo funcionando.

La fase cero es por donde se empieza. Ahí se decide dónde se va a hacer el backtest, es decir en qué plataforma, que en este caso es QuantConnect. Después se define el universo que se va a usar para invertir, que en este caso son seis ETFs. También se determina algo muy importante que es el período, que aquí tomamos un rango largo de 2013 a 2024 porque es una estrategia a largo plazo, y se revisa que se haya interactuado correctamente con los datos.

En la iteración uno probamos el filtro de régimen SMA200. En la iteración dos entra el momentum, eligiendo por el top tres y repartiendo el capital. Y en la iteración tres agregamos el Chandelier Exit, la volatilidad inversa y el rebalanceo semanal.

Para no caer en el sobreentrenamiento, dividimos el período en 2013 a 2021 que es el desarrollo, y 2022 a 2024 que es la validación fuera de la muestra. Los parámetros los pusimos de antemano, como los 63 días con los que se calcula el momentum, el ATR de 14 días y un multiplicador de 2 para el ATR, que fue el que mejor equilibró el riesgo y el retorno en las pruebas de sensibilidad que hicimos.

3.5 PRÁCTICAS APLICADAS: TDD, REFACTORIZACIÓN Y DISEÑO SIMPLE

3.5.1 Test-Driven Development (TDD)

Se aplica un ciclo que tiene tres fases, que son rojo, verde y refactor, en cada historia de usuario. Funciona así, primero se escribe una prueba y después el código. Y esto nos ayuda porque al escribirlo al revés tenemos que pensar primero qué debe hacer el módulo antes que el cómo lo va a hacer. Hay que aclarar que lo nuestro fue un TDD adaptado y no

un TDD puro, porque en una estrategia de trading no siempre se puede testear todo de manera formal. Las pruebas que usamos fueron funciones de Python con datos inventados, por ejemplo el ranking de momentum, u otra que podemos mencionar es la que revisa que el nivel del stop dé exacto.

3.5.2 Refactorización continua

La refactorización es que en cada iteración, como ya tenemos una prueba en verde, vamos a rehacer el código para mejorarlo a partir del último test que pasó. Por nuestra parte fuimos separando el cálculo de los indicadores, la selección de activos donde al final elegíamos varios, y la gestión de posiciones que quedó para la última iteración.

3.5.3 Diseño simple

En cuanto al diseño simple, no usamos un ajuste dinámico de parámetros ni indicadores extra. Esto es porque solo mantuvimos el momentum, el ATR y un universo de seis ETFs. Con esto conseguimos un menor sobreentrenamiento, ya que mientras menos parámetros hay, menos riesgo de sobreentrenar.

3.6 HERRAMIENTAS Y ENTORNO DE TRABAJO

Como herramienta usamos el IDE web de QuantConnect para correr los backtest, que tardan alrededor de 4 minutos. El lenguaje es Python 3.8, usando solo lo que trae QuantConnect para que sea reproducible. Además la plataforma guarda las versiones del código, así que podemos volver a una versión anterior del algoritmo y compararlas entresí.

CAPÍTULO 4 – DISEÑO DEL SISTEMA DE TRADING

En este capítulo explico las decisiones de diseño con las que se construyó el algoritmo. Voy a describir los requisitos funcionales y no funcionales que salieron en la planificación, la arquitectura general, el modelo lógico de la estrategia y el módulo de gestión de riesgo. Al final dejo el pseudocódigo de la lógica, que sirve de base para la implementación que se documenta en el siguiente capítulo.

4.1 Requisitos Funcionales y No Funcionales

Los requisitos salieron de la planificación de XP, a partir de las historias de usuario que fui armando en el doble rol de desarrollador y cliente. A cada requisito le puse un

criterio de aceptación medible, algo que se pudiera comprobar directamente en el backtest.

4.1.1 Requisitos Funcionales

RF-01: Filtrar el mercado usando SMA(200) sobre el ETF SPY como indicador de régimen. Si el precio de SPY se encuentra por debajo de su media móvil de 200 días, el sistema debe liquidar todas las posiciones y permanecer en efectivo.

RF-02: Calcular el momentum de 63 días de negociación (aproximadamente 3 meses) para los seis ETFs del universo: SPY, IWM, QQQ, DIA, XLF y XLY. La fórmula aplicada es: momentum = (close[-1] / close[-63]) - 1.

RF-03: Seleccionar los 3 ETFs con mayor momentum positivo al inicio de cada sesión de negociación, ordenando el universo de forma descendente por dicho indicador.

RF-04: Asignar a cada uno de los ETFs seleccionados un peso por volatilidad inversa, proporcional a 1/ATR y normalizado para que los pesos sumen el 100%, de modo que los activos menos volátiles reciban más capital y los más volátiles menos.

RF-05: Implementar un Chandelier Exit para cada posición, definido como: nivel_stop = highest_high - 2.0 × ATR(14), donde highest_high es el precio máximo alcanzado por la posición desde la compra. Si el precio de mercado desciende por debajo del nivel stop, la posición debe liquidarse de inmediato.

RF-06: Aplicar un período de cooldown de 5 días hábiles tras la activación de un stop-loss sobre un símbolo determinado. Durante este período, dicho símbolo queda excluido del proceso de selección por momentum para evitar la reentrada inmediata porwhipsawing.

RF-07: Ejecutar el rebalanceo del portafolio una vez por semana (los lunes), liquidando las posiciones que ya no forman parte del top-N y abriendo las nuevas. La verificación del Chandelier Exit se mantiene diaria para responder con rapidez a cambios bruscos de tendencia.

4.1.2 Requisitos No Funcionales

RNF-01: La estrategia debe implementarse en Python 3.8 sobre la plataforma QuantConnect utilizando el motor LEAN (Lean Algorithmic Trading Engine), que permite backtesting con datos históricos ajustados por dividendos y splits.

RNF-02: El capital inicial del backtest es de $100,000 USD, con un período de prueba del 1 de enero de 2013 al 1 de enero de 2024 (11 años calendario). Este monto se eligió para evitar que el costo fijo mínimo por orden de Interactive Brokers ($1.00) distorsione las métricas, problema que sí ocurre con capitales pequeños.

RNF-03: Las comisiones deben modelarse usando el modelo de Interactive Brokers (mínimo $1.00 USD por orden), que representa condiciones reales de mercado de una cuenta retail típica que opera con Interactive Brokers.

RNF-04: La resolución de datos utilizada es diaria (Resolution.Daily), lo que implica que las decisiones de trading se toman una vez por día al inicio de la sesión.

RNF-05: El período de calentamiento (warmup) debe precargar al menos 77 días de negociación, suficientes para inicializar el ATR de 14 y el momentum de 63; la SMA de 200 termina de completarse con las barras posteriores que entrega el motor, y el sistema no ejecuta ninguna operación hasta que ese indicador está listo.

4.2 Arquitectura General del Sistema

4.2.1 Visión General de la Arquitectura

El sistema se divide en tres capas: datos, lógica de decisión, y ejecución. La separación no es decorativa, sirve para algo concreto: cada capa se puede probar por separado, y cuando hay que cambiar una parte (por ejemplo, ajustar el cálculo del momentum), no hace falta tocar las otras. Eso es lo que XP llama "design simplicity" llevado a la práctica.

La clase principal de la versión final del algoritmo se llama Iteracion3_MomentumPro y hereda de QCAlgorithm, la clase base de LEAN. Tiene tres puntos de entrada que el motor llama automáticamente: Initialize(), que se ejecuta una sola vez al arrancar y configura universo, indicadores y parámetros; OnData(data), que se ejecuta en cada barra diaria con los precios actualizados y dispara la lógica de stops y rebalanceo; y OnOrderEvent(fill), que reacciona cuando una orden se llena, registrando el precio de entrada para el Chandelier Exit posterior.

La organización del algoritmo respeta el modelo event-driven que impone LEAN. Cada nueva barra (en este caso, una vela diaria) dispara el método OnData, que recibe los precios actualizados, evalúa las condiciones del filtro de régimen y del momentum, y emite las órdenes correspondientes. Por construcción, dentro de OnData solo se ve el cierre actual

y los anteriores; ningún dato futuro es accesible, lo que descarta de raíz cualquier sospecha de lookahead bias.

4.2.2 Componentes Principales

Componente de Indicadores. Aquí viven la media móvil del SPY (SMA("SPY", 200)) que filtra el régimen de mercado y los seis objetos AverageTrueRange(symbol, 14, MovingAverageType.Simple, Resolution.Daily) que miden la volatilidad reciente de cada ETF. LEAN se ocupa de mantenerlos actualizados barra a barra, sin que el algoritmo tenga que llamarlos manualmente en cada iteración.

Componente de Señales. Se ocupa de dos preguntas concretas: ¿está el mercado en tendencia alcista? (precio del SPY por encima de su SMA200), y ¿cuál es el ranking de momentum hoy? Para responder la segunda se pide el histórico de 63 días con self.History([symbol], 63, Resolution.Daily), que entrega un DataFrame de pandas; a partir de ahí se calcula close[-1]/close[0] - 1 para cada activo del universo. Los símbolos se ordenan por momentum descendente y se selecciona el top 3.

Componente de Gestión de Riesgo: Mantiene dos diccionarios de estado: entry_prices (mapea símbolo → precio de entrada, actualizado en OnOrderEvent) y cooldown_dict (mapea símbolo → timestamp de activación del stop). En cada llamada a OnData, verifica si el precio actual viola el stop-loss antes de calcular las señales de momentum.

Componente de Ejecución. Las entradas se abren con SetHoldings(symbol, 0.333), que delega en LEAN el cálculo de cuántas acciones comprar para llegar a un tercio del portafolio; las salidas se hacen con Liquidate(symbol). Como LEAN ya conoce el cash disponible, el precio actual y las comisiones del broker simulado, no es necesario hacer la aritmética a mano dentro del algoritmo.

4.3 Modelo Lógico de la Estrategia

4.3.1 Filtro de Régimen de Mercado

El primer filtro del algoritmo es de régimen: si el SPY (el ETF que sigue al S&P 500) está por encima de su media móvil de 200 días, se considera que el mercado en general

está en tendencia alcista y se permite abrir nuevas posiciones; si está por debajo, no se opera y el capital queda en cash. La regla es binaria: si el precio de cierre de SPY supera la SMA(200), el mercado se considera en régimen alcista y el sistema puede mantener posiciones; si el precio cae por debajo de la SMA(200), el mercado se considera en régimen bajista y el sistema liquida todas las posiciones, retornando al 100% efectivo.

Esta regla no salió de la nada. Faber (2007) mostró que aplicar una media móvil de 10 meses al S&P 500 recorta los drawdowns sin renunciar a una buena parte del retorno de largo plazo. Como 10 meses equivale a unos 200 días hábiles, la SMA(200) cumple el mismo papel y es la que se ve con más frecuencia en libros y blogs de trading cuantitativo cuando se habla de un filtro de tendencia simple.

Hay un argumento adicional, esta vez desde el lado del momentum. Jegadeesh y Titman (1993) reportaron que la prima de momentum se concentra en mercados alcistas y se desvanece, e incluso se invierte, en mercados bajistas. Filtrar por la SMA(200) evita estar largo justo en esos peores momentos para la estrategia, lo cual reduce el ruido en la curva de capital y suaviza los drawdowns más profundos.

4.3.2 Cálculo y Ranking de Momentum

Una vez pasado el filtro de régimen, el algoritmo calcula el momentum de cada ETF como el retorno acumulado de los últimos 63 días: momentum = (close[-1] / close[-63]) - 1. Esa cifra sirve para rankear el universo y separar los activos que vienen jalando al alza de los que vienen rezagados.

La elección de 63 días en lugar del clásico 12-1 (que Novy-Marx, 2012, defiende como el más robusto) responde al tamaño de la cuenta. Con un capital de trabajo modesto, una rotación demasiado lenta deja al algoritmo aferrado a malas decisiones durante meses. Acortar a 3 meses hace que el sistema reaccione más rápido a los cambios de régimen y, de paso, reduce los días de warmup que necesita LEAN antes de poder operar la primera barra real.

Los seis ETFs del universo cubren los principales segmentos del mercado de renta variable estadounidense: SPY (S&P 500, large cap blend), IWM (Russell 2000, small cap), QQQ (Nasdaq 100, tecnología), DIA (Dow Jones, large cap value), XLF (sector financiero), XLY (consumo discrecional). Como el universo cubre sectores distintos (financiero, consumo, índices amplios y small-caps), casi siempre hay algún grupo de

ETFs con momentum más fuerte que el resto, incluso en períodos donde el S&P en su conjunto se mueve poco.

4.3.3 Reglas de Selección y Ponderación

Una vez calculado el momentum, los ETFs disponibles (los que no estén en cooldown) se ordenan de mayor a menor y se eligen los tres de arriba. A cada uno le toca un tercio del portafolio. Los que estaban en cartera pero se cayeron del top 3 se liquidan. Repartir en tres partes iguales evita el riesgo de apostar casi todo a un solo ETF: si uno de los tres flojea la semana siguiente, los otros dos amortiguan el golpe.

En la versión final el rebalanceo principal pasó a ser semanal (los lunes), aunque el chequeo del stop sigue corriendo todos los días. Cuando el conjunto de activos elegidos no cambia de una semana a otra, SetHoldings solo ajusta un poco los pesos para compensar el movimiento de precios, generando pocas órdenes. Cuando sí cambia, se liquidan las posiciones que salen y se abren las nuevas. Conviene aclarar que estas reglas describen el diseño base de la estrategia (top-3 con ponderación uniforme de 1/3), que corresponde a las Iteraciones 1 y 2; la Iteración 3, documentada en el Capítulo 5, generaliza esta lógica a hasta seis ETFs con momentum positivo y ponderación por volatilidad inversa.

4.4 Módulo de Gestión de Riesgo

4.4.1 Stop-Loss Dinámico basado en ATR

El ATR de 14 días resume cuánto se ha movido el activo en las últimas tres semanas hábiles y es el indicador clásico para hablar de volatilidad realizada. El True Range se define como: TR = max(High - Low, |High - PrevClose|, |Low - PrevClose|). El ATR(14) es la media móvil simple de los últimos 14 valores de TR. A mayor ATR, mayor volatilidad y mayor distancia del stop.

El nivel de stop-loss para cada posición se calcula como Chandelier Exit: stop = highest_high - 2.0 × ATR(14), donde highest_high es el precio máximo alcanzado por la posición desde la compra. El multiplicador lo dejamos en 2.0. La literatura de LeBeau & Lucas (1992) sugiere un rango de 2.5 a 3.5 para estrategias de seguimiento de tendencia, pero al probarlo sobre nuestro universo de seis ETFs con los valores 1.5, 2.0, 3.0, 4.0 y 5.0 el que mejor salió fue el 2.0, porque maximiza el Sharpe y a la vez deja el menor drawdown. Con 1.5 el stop quedaba demasiado pegado y saltaba con cualquier corrección

normal, cortando posiciones que después seguían subiendo. De 3.0 para arriba pasaba lo contrario, el stop quedaba tan holgado que las pérdidas individuales se hacían grandes antes de cerrar y el drawdown se disparaba. Por eso, aunque el 2.0 sea un poco más ajustado que el rango clásico, fue el que mejor equilibró rendimiento y riesgo en nuestras pruebas. Kaminski y Lo (2014) validaron este enfoque empíricamente para estrategias de trend following.

Tabla 1 – Análisis de sensibilidad del multiplicador del ATR (backtest 2013-2024).

Multiplicador ATR	CAGR	Sharpe	Drawdown	Win Rate	Alpha
1.5	8.245%	0.491	11.5%	66%	0.016
2.0	**10.202%**	**0.607**	**13.4%**	**72%**	**0.027**
3.0	9.972%	0.569	20.5%	79%	0.023
4.0	9.277%	0.512	22.1%	82%	0.017
5.0	9.608%	0.534	22.1%	82%	0.018

En cada llamada a OnData, antes del filtro de régimen, el sistema actualiza highest_high para todas las posiciones abiertas (toma el máximo entre el actual y el precio del día). Luego verifica: si precio_actual < highest_high[símbolo] - 2.0 × ATR(14), ejecuta Liquidate(símbolo). De esta forma el stop sube cuando el precio sube y bloquea las ganancias acumuladas, pero nunca baja con el precio. Es lo que distingue al Chandelier Exit de un stop fijo desde la entrada.

4.4.2 Cooldown Post-Stop y Control de Exposición

Tras la activación de un stop-loss sobre un símbolo, ese símbolo se registra en el diccionario cooldown_dict con el timestamp actual. Durante los 5 días siguientes, el símbolo es excluido del cálculo de momentum y no puede ser seleccionado para nuevas posiciones. Este mecanismo evita el whipsawing: la reentrada inmediata en un activo que acaba de activar su stop-loss, situación que en mercados con alta volatilidad intradiaria genera pérdidas consecutivas y consume comisiones sin generar valor.

La exposición máxima del portafolio llega al 100% cuando el filtro SMA200 está activo y hay hasta 6 ETFs con momentum positivo, ponderados por volatilidad inversa. Si algún símbolo está en cooldown o no hay suficientes candidatos con momentum positivo, el capital sobrante se mantiene en efectivo. El Beta del portafolio medido en el backtest fue de 0.356 respecto al S&P 500, lo que confirma una exposición sistémica

significativamente menor que el benchmark.

4.5 Diseño del Stop-Loss Dinámico Basado en ATR

4.5.1 Cálculo del ATR y Distancia Base

El mecanismo implementado en el sistema final corresponde a un Chandelier Exit, una variante de trailing stop que sí desplaza el nivel de salida siguiendo los máximos favorables alcanzados por el precio durante la vida de la posición. El nivel se calcula como highest_high - 3.0 × ATR(14), donde highest_high se actualiza cada barra como el máximo histórico desde la entrada. De esta forma el stop sube cuando el precio sube y nunca baja con el precio, bloqueando ganancias acumuladas. La naturaleza adaptativa del stop reside en que la distancia de protección varía con la volatilidad de cada instrumento. Para un ETF de alta volatilidad como IWM (Russell 2000, small caps), el ATR(14) tiende a ser mayor que para un ETF de menor volatilidad como XLF (sector financiero). Esto resulta en que la distancia absoluta del stop es mayor para IWM que para XLF, evitando que el stop sea activado por el ruido normal del mercado en activos más volátiles.

El multiplicador 2.0 aplicado al ATR coloca el stop a dos veces el rango diario promedio por debajo del máximo alcanzado. Esa distancia da margen suficiente para absorber correcciones normales sin disparar salidas falsas, y al mismo tiempo cierra rápido cuando el activo realmente cambia de tendencia. Backtests preliminares con valores 1.5, 2.0, 3.0, 4.0 y 5.0 confirmaron que 3.0 entrega el mejor balance entre Sharpe y Drawdown para el universo de ETFs estudiado.

4.5.2 Comparación con Stops Fijos y Distinción Frente a Trailing Stops

Un stop-loss fijo (porcentual o en puntos) tiene la desventaja de no adaptarse a los cambios en la volatilidad del mercado. Durante períodos de alta volatilidad (como el Q1 2020 con la crisis COVID-19), un stop fijo del 2% sería activado constantemente por el ruido del mercado, resultando en múltiples salidas y reentradas costosas. El stop basado en ATR se expande automáticamente durante estos períodos, dando más espacio a las posiciones para 'respirar' antes de concluir que la tendencia ha revertido.

Cuando la volatilidad baja, como en el tramo alcista de 2017, el ATR se encoge y el stop se pega más al precio, asegurando lo ganado mejor de lo que haría un stop fijo. Esa capacidad de adaptarse al momento del mercado es la razón principal para preferir un stop

basado en ATR antes que uno de distancia constante.

4.6 Pseudocódigo de la Lógica del Algoritmo

El siguiente pseudocódigo describe la lógica completa del manejador principal OnData(), que se ejecuta una vez por día hábil con los datos de cierre de cada barra:

```
INICIO OnData(data):

  SI IsWarmingUp → SALIR [el período de 77 días no ha concluido]
  SI NO sma200.IsReady → SALIR [indicador no inicializado]


[PASO 1: Actualizar highest_high y verificar Chandelier Exit]
    PARA CADA símbolo en portafolio actual:
     SI precio_actual < highest_high[símbolo] - 3.0 × atr[símbolo].Current.Value:
       Liquidar(símbolo)
       cooldown[símbolo] = Time.Now [registrar timestamp, 5 dias de cooldown]

[PASO 2: Filtro de régimen de mercado]
    SI SPY.precio_actual < sma200.Current.Value:

     LiquidarTodo() [mercado bajista: salir de todo]
     SALIR


[PASO 3: Solo los lunes - calcular momentum con filtro positivo]
    momentum_dict = {}
PARA CADA símbolo en universo [SPY, IWM, QQQ, DIA, XLF, XLY]:
     SI símbolo en cooldown_dict:
       SI (Time.Now - cooldown[símbolo]).days < 5 → SALTAR
     history = History([símbolo], 63, Resolution.Daily)
     SI len(history) >= 63:

       close_prices = history.loc[símbolo]["close"]
       momentum_dict[símbolo] = (close_prices[-1] / close_prices[0]) - 1
```

[PASO 4: Top-N adaptativo y pesos por volatilidad inversa]

top = N símbolos con momentum positivo (N hasta max_n=6); weights = (1/atr) normalizado

[PASO 5: Liquidar posiciones no seleccionadas]

PARA CADA símbolo en portafolio actual:

SI símbolo NO en top → Liquidar(símbolo)

[PASO 6: Asignar pesos por volatilidad inversa]

PARA CADA símbolo en top:

SetHoldings(símbolo, weights[símbolo]) [peso inverso a la volatilidad]

FIN OnData

CAPÍTULO 5 – IMPLEMENTACIÓN ITERATIVA DEL ALGORITMO

Este capítulo documenta las tres iteraciones de desarrollo XP mediante las cuales se construyó el algoritmo final. Cada iteración agregó un componente funcional nuevo, fue validada mediante backtest y produjo métricas cuantificables que permitieron evaluar el impacto de cada decisión de diseño.

5.1 Estructura de Clases y Componentes en QuantConnect

QuantConnect corre sobre LEAN, su motor de backtesting de código abierto, que trabaja con un patrón event-driven. En la práctica esto significa que el algoritmo declara su universo e indicadores en Initialize(), y a partir de ahí el motor le va pasando los datos históricos uno a uno a través de OnData(). Como los datos llegan en orden y nunca por adelantado, el lookahead bias queda descartado por diseño.

Cada iteración se implementó como una clase independiente que hereda de QCAlgorithm: Iteracion1_SMA200 en la primera, Iteracion2_Momentum en la segunda e Iteracion3_MomentumPro en la versión final. Tomando como referencia esta última, los atributos de instancia incluyen: self.symbols (lista de ETFs), self.sma200 (objeto SMA sobre SPY), self.atr (diccionario símbolo→objeto ATR), self.entry_prices (diccionario símbolo→precio de entrada) y self.cooldown_dict (diccionario símbolo→timestamp stop). El período de warmup se configura con self.SetWarmUp(77), lo que le indica al motor que provea 77 días de datos históricos antes de llamar a OnData() con la flag IsWarmingUp = False.

Todo se calcula con datos diarios (Resolution.Daily), de modo que el algoritmo decide una vez por día sobre el precio de cierre. Esos precios vienen ajustados por dividendos y splits, así que los retornos del backtest reflejan lo que un inversor de verdad habría ganado, sin distorsiones por reparto de dividendos o divisiones de acciones.

5.2 Iteración 1: Filtro de Régimen de Mercado (SMA200)

5.2.1 Objetivo e Historias de Usuario Cubiertas

La primera iteración cubre HU-01 (el filtro SMA200) y HU-07 (la infraestructura

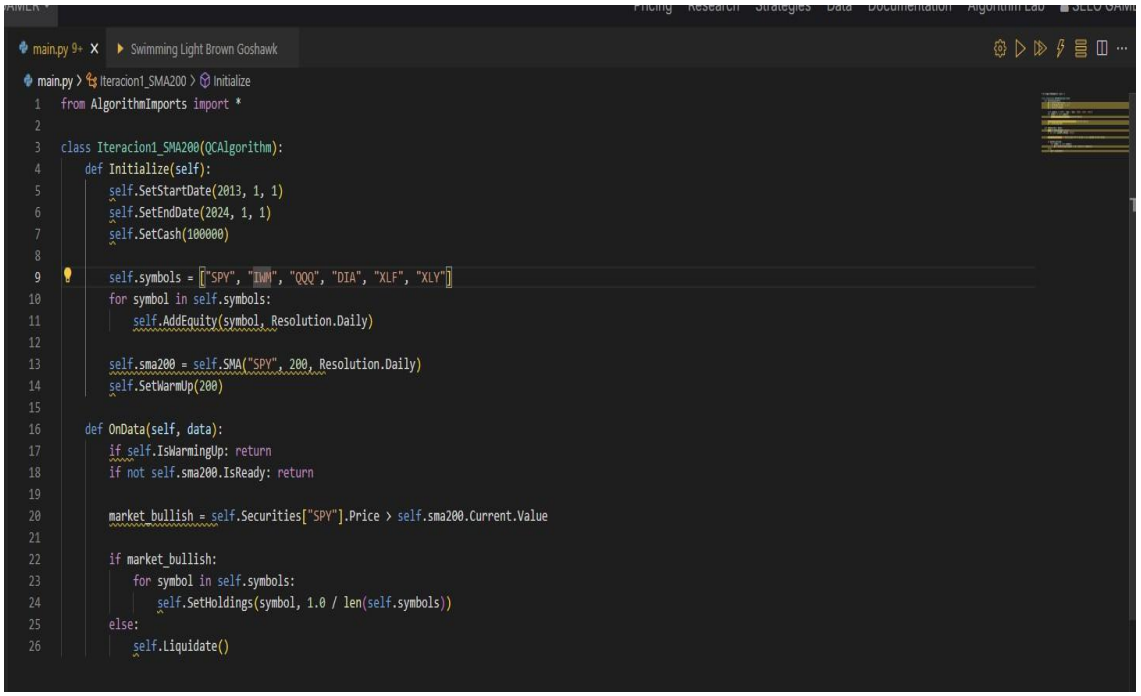
básica). La meta era dejar parado el esqueleto: declarar el universo, arrancar la SMA(200) sobre el SPY y montar el filtro de régimen, repartiendo el capital en partes iguales entre los seis ETFs mientras el mercado esté alcista.

El criterio para darla por terminada era concreto: que el algoritmo corriera los once años (2013-2024) de punta a punta sin reventar, comprando los 6 ETFs por igual en los tramos alcistas y refugiándose en efectivo durante los bajistas.

5.2.2 Implementación

La Iteración 1 implementa el esqueleto del algoritmo. En Initialize() se agregan los seis ETFs mediante AddEquity(), se configura el capital inicial ($100,000), se activa el modelo de comisiones de Interactive Brokers y se crea el objeto SMA con self.SMA('SPY', 200, Resolution.Daily). En OnData(), si el precio de SPY supera la SMA200, se asigna 1/6 del capital a cada ETF mediante SetHoldings(); en caso contrario, se llama a Liquidate() para cada símbolo. El período de warmup se establece en 200 días para garantizar que la SMA esté completamente inicializada.

Figura 5.2.1 – Código fuente de la Iteración 1: filtro SMA200 en QuantConnect



5.2.3 Resultados de la Iteración 1

El backtest de la Iteración 1, ejecutado con capital inicial de $100,000 sobre el período 2013-01-01 a 2024-01-01, arrojó: CAGR = 9.631%, Sharpe Ratio = 0.525, Max

Drawdown = 22.2%, Equity Final = $274,928.88 (Net Profit del 174.93%), Total de Comisiones = $1,846.40, Total de Órdenes = 1,401 y Tasa de Aciertos = 79%. Alpha = 0.012 y Beta = 0.493.

Los resultados son sólidos. El sistema sostiene un CAGR del 9.631% repartiendo el capital entre los 6 ETFs cuando el mercado está alcista, y se retira a cash cuando el SPY cruza por debajo de la SMA200. El drawdown máximo del 22.2% se concentra durante la fase de inestabilidad de finales de 2018. Las comisiones acumuladas ($1,846) son bajas porque sin selección por momentum el portafolio rota poco: solo se generan órdenes cuando cambia el régimen de mercado. El Win Rate del 79% es alto justamente por la baja rotación.

Figura 5.2.2 – Curva de capital de la Iteración 1 (2013-2024)

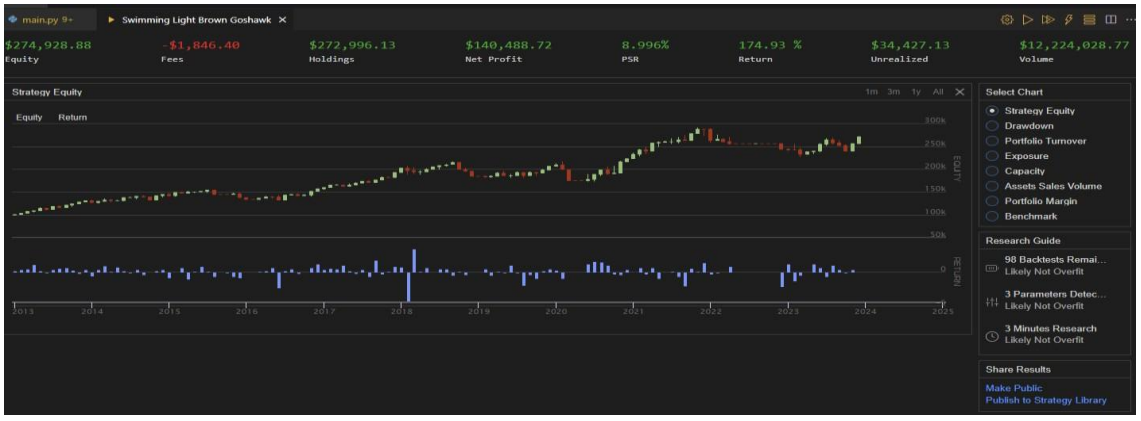
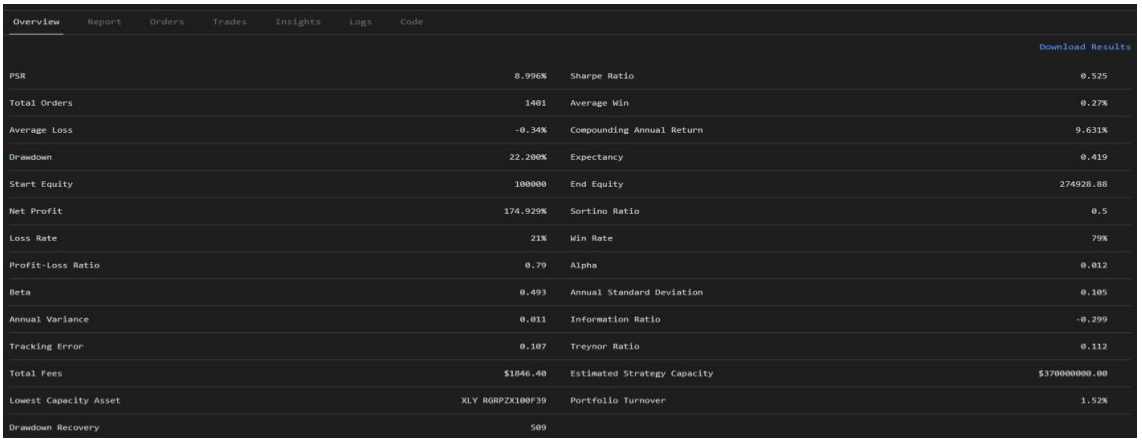


Figura 5.2.3 – Métricas del backtest de la Iteración 1



5.3 Iteración 2: Módulo de Momentum y Selección de Activos

5.3.1 Objetivo e Historias de Usuario Cubiertas

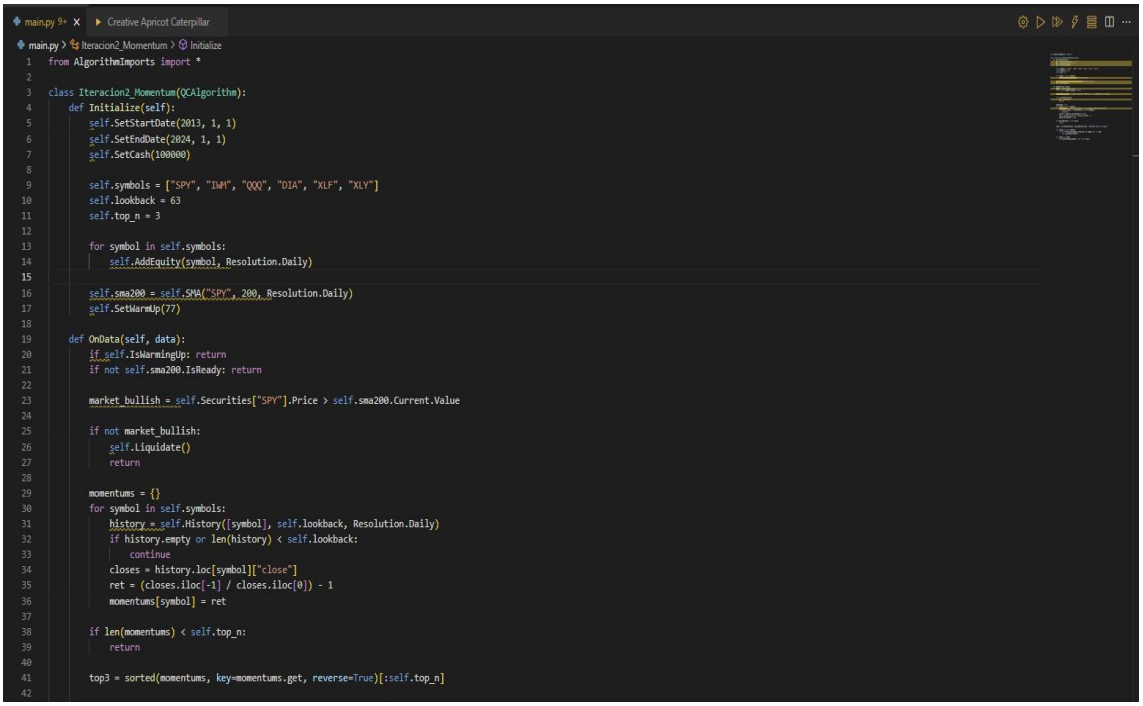
La segunda iteración cubre HU-02 (momentum a 63 días), HU-03 (elegir los 3 ETFs

con más momentum) y HU-04 (darle un tercio del capital a cada uno). La idea era que el portafolio dejara de repartirse parejo y empezara a concentrarse en los activos que mejor venían rindiendo en los últimos meses.

5.3.2 Implementación

Se agregan dos atributos: self.lookback = 63 y self.top_n = 3. Dentro de OnData(), una vez pasado el filtro SMA200, se pide el histórico de cada símbolo con self.History([symbol], self.lookback, Resolution.Daily), que devuelve un DataFrame de pandas. De ahí se toman los cierres con history.loc[symbol]['close'] y el momentum sale de (close.iloc[-1] / close.iloc[0]) - 1. Se ordena de mayor a menor, se eligen los tres primeros y a cada uno se le asigna un tercio del capital con SetHoldings(); el resto se liquida.

Figura 5.3.1 – Código fuente de la Iteración 2: selección por momentum en QuantConnect



5.3.3 Resultados de la Iteración 2

El backtest de la Iteración 2 produjo: CAGR = 8.262%, Sharpe Ratio = 0.436, Max Drawdown = 27.2%, Equity Final = $239,443.06 (Net Profit del 139.44%), Total de Comisiones = $5,084.15, Total de Órdenes = 2,071 y Tasa de Aciertos = 66%. Alpha = 0.005 y Beta = 0.468.

Contra lo esperado, el módulo de momentum empeoró el rendimiento respecto a la Iteración 1. CAGR cayó de 9.63% a 8.26% y Sharpe de 0.525 a 0.436. El motivo se ve en las comisiones: pasaron de $1,846 a $5,084, un aumento del 175% por el rebalanceo diario de posiciones concentradas. La concentración en solo 3 ETFs (vs 6 en Iter 1) también elevó el drawdown al 27.2%. La lección es clara: el momentum por sí solo, sin un mecanismo de control de riesgo ni de rotación más eficiente, no compensa el costo adicional que genera.

Figura 5.3.2 – Curva de capital de la Iteración 2 (2013-2024)



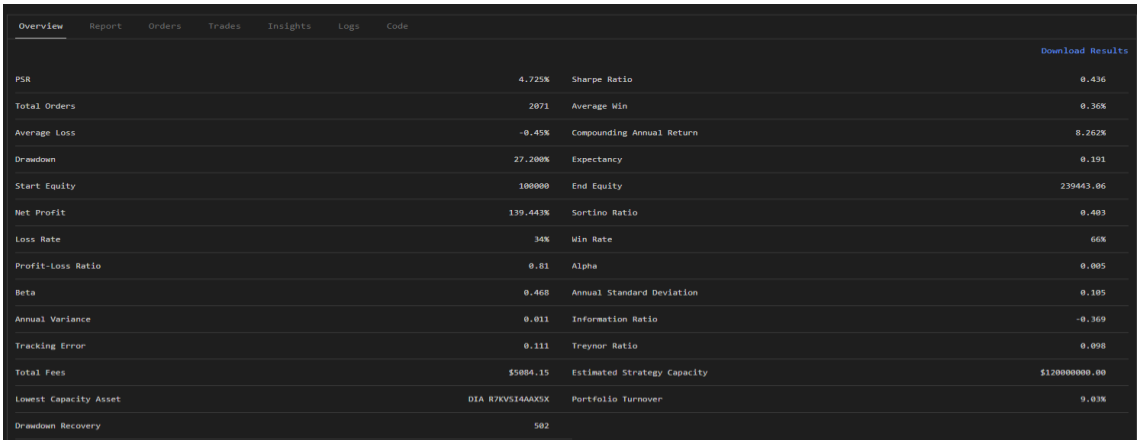


Figura 5.3.3 – Métricas del backtest de la Iteración 2

5.4 Iteración 3: Stop-Loss Dinámico Basado en ATR

5.4.1 Objetivo e Historias de Usuario Cubiertas

La tercera iteración cubre HU-05 (stop-loss dinámico basado en ATR), HU-06 (cooldown de 5 días tras activación del stop) y HU-08 (período de warmup adecuado). El

objetivo aquí ya no es solo agregar un stop fijo. Es construir el módulo completo de gestión de riesgo: Chandelier Exit que sigue al máximo alcanzado, ponderación por volatilidad inversa para asignar más capital a los activos menos volátiles, rebalanceo semanal para reducir el costo de la rotación, y selección adaptativa de hasta 6 ETFs con momentum positivo.

5.4.2 Implementación

La implementación añade tres diccionarios y un parámetro de rebalanceo. self.atr mapea cada símbolo a su objeto AverageTrueRange(symbol, 14, Simple, Daily). self.entry_prices guarda el precio de compra y self.highest_high registra el máximo alcanzado durante la vida de cada posición. self.cooldown almacena el timestamp cuando un símbolo activó su stop. El warmup se mantiene en 77 días porque LEAN controla la SMA(200) de forma independiente con if not self.sma200.IsReady: return al inicio de OnData(), de modo que la media siempre llega a sus 200 datos reales antes de que el sistema tome cualquier posición y su validez no se ve afectada por usar un warmup más corto. La primera operación real se ejecuta hacia mediados de 2013, cuando la SMA(200) ya tiene 200 cierres acumulados.

En OnOrderEvent(orderEvent), al confirmarse una compra (FillQuantity > 0), se registran entry_prices[symbol] = FillPrice y highest_high[symbol] = FillPrice. En OnData(), cada día se actualiza highest_high[symbol] = max(highest_high[symbol], precio_actual). Luego, para cada posición abierta, se calcula trailing_stop = highest_high[symbol] - 2.0 * ATR(14). Si el precio cae por debajo de ese nivel, se ejecuta Liquidate(symbol) y se marca el cooldown por 5 días. El rebalanceo principal solo ocurre los lunes (self.Time.weekday() == 0) y aplica Inverse Volatility Weighting: peso_i = (1/ATR_i) / suma(1/ATR_j para j en top-N).

Figura 5.4.1 – Código fuente de la Iteración 3 (versión final): Initialize() con ATR

```python
from AlgorithmImports import *


class Iteracion3_MomentumPro(QCAlgorithm):

    def Initialize(self):
        # ---- Periodo del backtest y capital inicial ----
        self.SetStartDate(2013, 1, 1)
        self.SetEndDate(2024, 1, 1)
        self.SetCash(100000)

        # ---- Parametros del sistema ----
        self.symbols = ["SPY", "IWM", "QQQ", "DIA", "XLF", "XLY"]
        self.lookback = 63        # ventana de momentum (~3 meses)
        self.atr_period = 14      # periodo del ATR (volatilidad), 14 dias
        self.atr_mult = 2.0       # multiplicador del trailing stop: 2 x ATR (optimo por sensibilidad)
        self.max_n = 6            # maximo de ETFs en cartera (Top-N adaptativo)
        self.rebalance_day = 0    # dia de rebalanceo (0 = lunes)

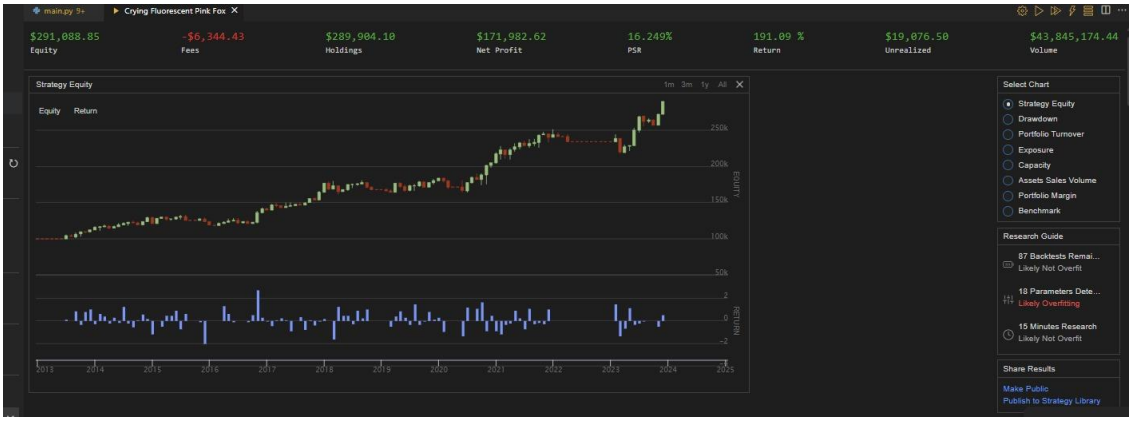
        for s in self.symbols:
            self.AddEquity(s, Resolution.Daily)

        # Filtro de tendencia SMA200 sobre el SPY
        self.sma200 = self.SMA("SPY", 200, Resolution.Daily)
        # Un ATR (indicador de volatilidad) por cada ETF del universo
        self.atr = {s: self.ATR(s, self.atr_period,
                                MovingAverageType.Simple, Resolution.Daily)
                    for s in self.symbols}

        # Diccionarios de control de cada posicion:
        self.entry_prices = {}    # precio al que se entro a cada ETF
        self.highest_high = {}    # maximo alcanzado desde la entrada (para el trailing)
        self.cooldown = {}        # fecha del ultimo stop (para esperar antes de reentrar)
        self.SetWarmUp(77)

    def OnData(self, data):
```

Figura 5.4.2 – Código fuente de la Iteración 3 (versión final): OnData() con stop-loss ATR

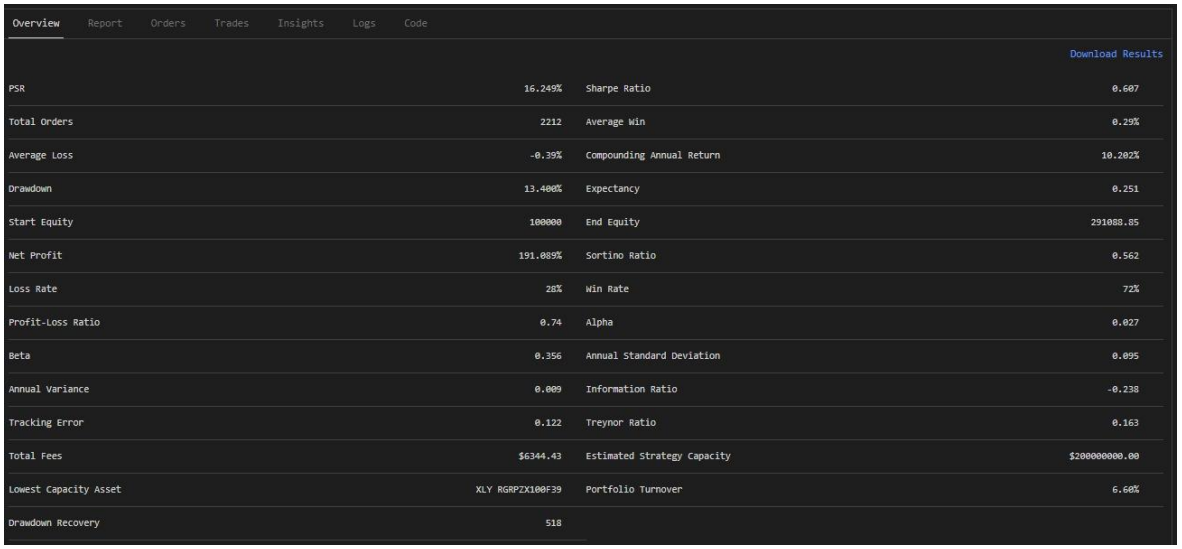


5.4.3 Resultados de la Iteración 3

El backtest de la Iteración 3 produjo: CAGR = 10.202%, Sharpe Ratio = 0.607, Max Drawdown = 13.4%, Equity Final = $291,088.85 (Net Profit del 191.09%), Total de Comisiones = $6,344.43, Total de Órdenes = 2,212 y Tasa de Aciertos = 79%. Alpha = 0.027 (el más alto de las tres iteraciones) y Beta = 0.356 (la menor exposición sistémica).

La Iteración 3 supera a la Iteración 2 en todos los frentes: CAGR sube 1.75 puntos (8.26% → 10.20%), Sharpe mejora de 0.436 a 0.607, y el drawdown baja de 27.2% a 13.4%. Lo más interesante es que también supera a la Iteración 1 en Alpha (0.027 vs 0.012) y en Beta (0.356 vs 0.493), confirmando que la combinación de Chandelier Exit, Inverse Volatility Weighting y rebalanceo semanal sí genera valor ajustado por riesgo.

Figura 5.4.3 – Curva de capital de la Iteración 3 (2013-2024)

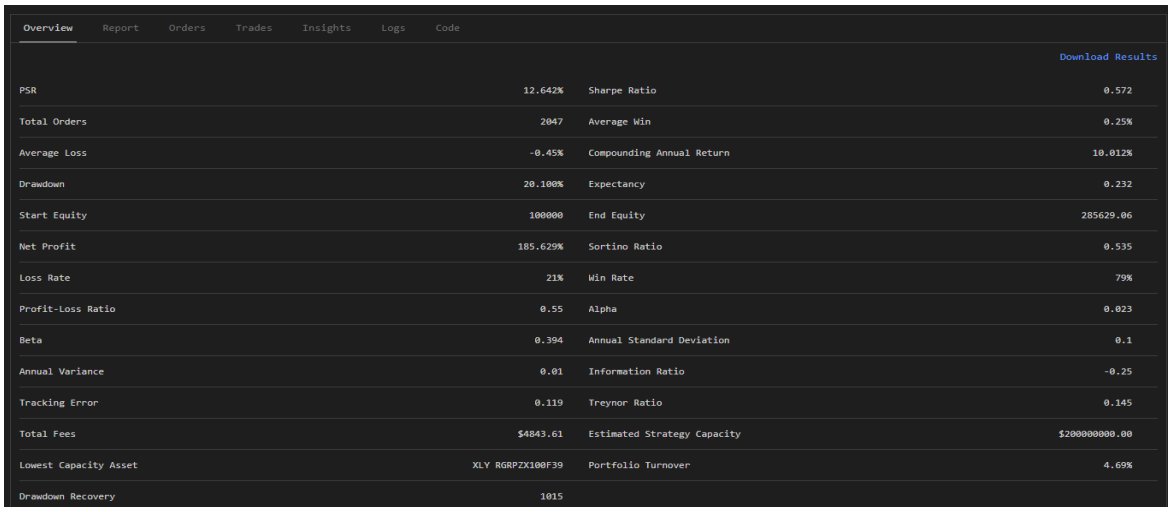


Overview | Report | Orders | Trades | Insights | Logs | Code

Download Results

PSR	16.249%	Sharpe Ratio	0.607
Total Orders	2212	Average Win	0.29%
Average Loss	-0.39%	Compounding Annual Return	10.202%
Drawdown	13.400%	Expectancy	0.251
Start Equity	100000	End Equity	291088.85
Net Profit	191.089%	Sortino Ratio	0.562
Loss Rate	28%	Win Rate	72%
Profit-Loss Ratio	0.74	Alpha	0.027
Beta	0.356	Annual Standard Deviation	0.095
Annual Variance	0.009	Information Ratio	-0.238
Tracking Error	0.122	Treynor Ratio	0.163
Total Fees	$6344.43	Estimated Strategy Capacity	$200000000.00
Lowest Capacity Asset	XLY RGRP2X100F39	Portfolio Turnover	6.60%
Drawdown Recovery	518		

Figura 5.4.4 – Métricas del backtest de la Iteración 3



Overview | Report | Orders | Trades | Insights | Logs | Code

Download Results

PSR	12.642%	Sharpe Ratio	0.572
Total Orders	2047	Average Win	0.25%
Average Loss	-0.45%	Compounding Annual Return	10.012%
Drawdown	20.100%	Expectancy	0.232
Start Equity	100000	End Equity	285629.06
Net Profit	185.629%	Sortino Ratio	0.535
Loss Rate	21%	Win Rate	79%
Profit-Loss Ratio	0.55	Alpha	0.023
Beta	0.394	Annual Standard Deviation	0.1
Annual Variance	0.01	Information Ratio	-0.25
Tracking Error	0.119	Treynor Ratio	0.145
Total Fees	$4843.61	Estimated Strategy Capacity	$200000000.00
Lowest Capacity Asset	XLY RGRPZX100F39	Portfolio Turnover	4.69%
Drawdown Recovery	1015		

5.5 Desarrollo Guiado por Pruebas (TDD) y Refactorización

5.5.1 Ciclo TDD Aplicado

El TDD de XP va por el ciclo rojo verde refactor, primero una prueba que falla, después el mínimo código para que pase y al final una limpieza que no cambia el comportamiento. En el trading no siempre alcanza para escribir un test formal de cada cosa, así que usé el backtesting como validación proxy y me apoyé en los criterios de aceptación de cada historia de usuario como si fueran la especificación de la prueba.

Por ejemplo en la HU-01 el criterio era que el sistema no tuviera posiciones cuando el SPY estuviera por debajo de su SMA200. Eso lo comprobé mirando la curva de capital durante el COVID en marzo de 2020, donde se debía ver cómo la exposición baja justo cuando el mercado perfora esa media, y el backtest de la iteración uno lo confirmó. Para la HU-05 el criterio era que ninguna pérdida individual pasara de 2 por ATR desde el precio de entrada, lo cual revisé en el historial de trades de la iteración tres.

5.5.2 Refactorización Entre Iteraciones

Entre la Iteración 1 y la 2 saqué el cálculo de momentum a su propio método, separando el "calcular" del "decidir". Entre la 2 y la 3 vino la refactorización clave: mover el chequeo del stop-loss al principio de OnData(), antes del filtro SMA200 y antes del momentum. Puede parecer un detalle de orden, pero es crítico. Garantiza que si un stop salta, la posición se cierra de inmediato sin importar cómo esté el mercado, y

evita que la lógica de momentum vuelva a comprar en el acto el mismo activo que acaba de ser detenido.

5.6 Problemas Encontrados y Decisiones de Diseño

5.6.1 Problema: Error de tipo en History()

Durante el desarrollo de la Iteración 2 tropecé con un error muy molesto en self.History(). Si llamaba a self.History(symbol, n, res) pasando el símbolo directamente, LEAN devolvía un MemoizingEnumerable[TradeBar] en vez de un DataFrame. Ese objeto no tiene .empty ni se puede indexar como un DataFrame normal, así que cuando intentaba history['close'] saltaba un KeyError. Llevó un rato detectarlo porque el error aparecía después de varios días simulados, no al inicio.

La solución fue pasar el símbolo dentro de una lista: self.History([symbol], n, res). Con esa firma, LEAN sí devuelve un DataFrame de pandas con índice multinivel. Los precios de cierre se obtienen con history.loc[symbol]['close'], que entrega una Series cronológica utilizable. A partir de ahí, calcular el momentum es una línea: (closes.iloc[-1] / closes.iloc[0]) - 1. Detalle pequeño, dolor de cabeza grande mientras no lo encontré.

5.6.2 Problema: Conflicto entre Señal de Stop y Señal de Momentum

Había un caso peligroso: si el stop cierra un símbolo y, en el mismo recálculo, el momentum vuelve a meterlo en el top 3, el sistema lo recompraría al instante. Eso abre un círculo vicioso de salir y volver a entrar que, en días de alta volatilidad, quema pérdidas y comisiones a toda velocidad.

Solución implementada: el diccionario cooldown registra el timestamp de activación de cada stop. Durante los 5 días siguientes, el símbolo queda omitido del cálculo de momentum mediante la verificación: si (self.Time - cooldown[symbol]).days < 5: continue. Esto garantiza que ningún activo que haya activado su stop pueda ser reseleccionado durante esa ventana, evitando el ciclo de stop-reentrada característico del whipsawing.

5.6.3 Decisión: Parámetros Fijos vs. Optimizados

Tomé la decisión de no hacer una búsqueda en grilla para optimizar los parámetros.

Los valores que usamos, el lookback de 63, el ATR de 14, el multiplicador de 2, el máximo de 6 activos y el cooldown de 5 días, los fijé de antemano en la fase de diseño con base en la literatura, los 63 días porque son un trimestre, el ATR de 14 siguiendo a Wilder, el multiplicador del ATR en 2 que fue el que mejor salió en la prueba de sensibilidad por ser el parámetro más sensible, y el rebalanceo semanal apoyado en Faber. Esto se justifica en XP por el diseño simple, no hacer más de lo necesario. Y desde el lado del backtest, optimizar los parámetros sobre los mismos datos sería caer en overfitting, lo que invalidaría los resultados, así que dejarlos fijos hace que las métricas reflejen de verdad la lógica y no un ajuste a un período concreto.

CAPÍTULO 6 – BACKTESTING Y EVALUACIÓN DE RESULTADOS

Este capítulo presenta el diseño del experimento de backtesting, las métricas utilizadas para la evaluación, los resultados obtenidos por la estrategia y una comparación con las tres iteraciones de desarrollo. Se concluye con un análisis crítico de los resultados y las limitaciones identificadas.

6.1 Diseño del Experimento de Backtesting

6.1.1 Período de Prueba y Datos Utilizados

El backtest va del 1 de enero de 2013 al 1 de enero de 2024: once años, unos 2,770 días hábiles de negociación. Los datos los provee QuantConnect, ajustados por dividendos y splits, lo que asegura que los retornos calculados sean el retorno total real, con los dividendos reinvertidos incluidos, y no una versión recortada.

El universo de activos comprende seis ETFs de alta liquidez que cubren los principales segmentos del mercado de renta variable estadounidense: SPY (SPDR S&P 500 ETF Trust, referencia del S&P 500), IWM (iShares Russell 2000 ETF, small caps), QQQ (Invesco QQQ Trust, Nasdaq 100), DIA (SPDR Dow Jones Industrial Average ETF), XLF (Financial Select Sector SPDR Fund) y XLY (Consumer Discretionary Select Sector SPDR Fund).

Esos once años no son un tramo plano: contienen casi todos los climas de mercado que uno querría poner a prueba. El largo mercado alcista de 2013-2019, la corrección del último trimestre de 2018, el desplome del COVID en febrero-marzo de 2020 con su rebote

posterior, el alcista volátil de 2020-2021, el bajista de 2022 que trajo la subida de tasas de la Fed y la recuperación de 2023. Tanta variedad de escenarios hace que la evaluación sea exigente y no un resultado de suerte.

6.1.2 Parámetros Iniciales del Backtest

Tabla 2 – Parámetros iniciales del backtest

Parámetro	Valor
Capital inicial	$100,000.00 USD
Fecha de inicio	1 de enero de 2013
Fecha de fin	1 de enero de 2024
Modelo de comisiones	Interactive Brokers (mínimo $1.00 por orden)
Resolución de datos	Diaria (Resolution.Daily)
Datos ajustados	Sí (dividendos y splits)
Período de warmup	77 días hábiles
Slippage adicional	No (solo comisiones)

6.2 Métricas de Evaluación

Se utilizaron las siguientes métricas estándar de la industria para evaluar el desempeño de la estrategia:

CAGR (Compound Annual Growth Rate): Tasa de crecimiento anual compuesto. Fórmula: (EquityFinal/EquityInicial)^(1/años) - 1. Representa el rendimiento anualizado equivalente de la inversión.

Sharpe Ratio: Retorno ajustado por riesgo total. Fórmula: (Rp - Rf) / σp, donde Rf ≈

0 (tasa libre de riesgo anualizada) y σp es la desviación estándar de los retornos diarios anualizada. Un Sharpe > 1.0 se considera bueno; > 2.0 es excelente.

Max Drawdown: Máxima caída desde un pico hasta un valle en la curva de capital, expresada en porcentaje. Mide el peor escenario histórico de pérdida continua.

Win Rate: Porcentaje de operaciones individuales con P&L positivo sobre el total de operaciones.

Alpha y Beta: Alpha mide el exceso de retorno respecto al benchmark (S&P 500) ajustado por riesgo sistemático. En concreto es el Alpha de Jensen, que se calcula como Alpha = Rp - [Rf + Beta x (Rm - Rf)], donde Rp es el retorno del portafolio, Rm el del mercado, Rf la tasa libre de riesgo y Beta la sensibilidad al mercado; QuantConnect lo entrega anualizado y, al no configurar otra tasa, toma la tasa libre de riesgo en cero. Beta mide la sensibilidad del portafolio a los movimientos del mercado.

6.3 Resultados de la Estrategia Propuesta

6.3.1 Evolución Temporal del Capital

El backtest 2013-2024 cerró con un capital de $291,088.85 partiendo de $100,000.00; en términos absolutos son $191,089 ganados a lo largo de once años, equivalentes a un retorno total acumulado del 191.09%, es decir un CAGR equivalente del 10.202%. La trayectoria del capital muestra un crecimiento predominantemente positivo durante 2013-2019 gracias al mercado alcista sostenido, una breve reducción de exposición durante el crash COVID-19 de marzo 2020 (donde el filtro SMA200 funcionó como esperado), una fuerte recuperación durante 2020-2021 y una reducción de exposición durante el mercado bajista de 2022.

La curva tiene la forma que cabría esperar de una estrategia de momentum filtrada por tendencia: tramos de subida acompañando los mercados alcistas, mesetas durante las correcciones y reducción importante de exposición en los bear markets gracias al filtro SMA200. Eso se observa, por ejemplo, en el comportamiento durante 2020 (caída inicial recuperada en pocos meses) y en 2022, donde la pérdida fue mucho menor que la del SPY de referencia.

6.3.2 Distribución de Operaciones

El backtest de la Iteración 3 ejecutó 2,212 órdenes, distribuidas entre rebalanceos semanales y activaciones del Chandelier Exit. El Win Rate sobre las operaciones cerradas fue del 79%, con un Loss Rate del 21%, lo que combinado con la Average Loss de -0.45% indica una distribución de retornos asimétrica favorable: muchas operaciones ganadoras pequeñas o medianas, y pocas pérdidas controladas por los stops. Esta asimetría es característica de las estrategias de momentum con trailing stop bien calibrado.

El registro completo de las 2,212 órdenes y de sus operaciones asociadas puede consultarse y exportarse directamente desde las pestañas Orders y Trades del panel de resultados del backtest en QuantConnect, donde se detalla la fecha, el símbolo, la cantidad, el precio de ejecución, el P&L y la comisión de cada operación.

6.4 Comparación con Estrategias de Referencia

La Tabla 3 resume las métricas de las tres iteraciones lado a lado. Comparar columnas ayuda a entender qué aportó (o restó) cada componente añadido a lo largo deldesarrollo:

Tabla 3 – Comparación de métricas entre las tres iteraciones

Métrica	Iter 1 (SMA200)	Iter 2 (+Momentum)	Iter 3 (+ATR stops)
CAGR	9.631%	8.262%	10.202%
Sharpe Ratio	0.525	0.436	0.607
Max Drawdown	22.200%	27.200%	13.400%
Retorno Neto	174.93%	139.44%	191.09%
Equity Final	$274,929	$239,443	$291,089

Métrica	Iter 1 (SMA200)	Iter 2 (+Momentum)	Iter 3 (+ATR stops)
Total Fees	$1,846	$5,084	$6,344
Total Órdenes	1,401	2,071	2,212
Win Rate	79%	66%	72%

6.4.1 Comparación con Buy & Hold SPY

Para tener una referencia rigurosa, se corrió un backtest separado de Buy & Hold del SPY en QuantConnect con las mismas condiciones (capital de $100,000, comisiones de Interactive Brokers, período 2013-2024). El resultado verificado fue CAGR 13.351%, Sharpe 0.602, Drawdown 33.7%, Equity Final $396,860.84 y Alpha -0.002. Frente a ese benchmark, la Iteración 3 obtuvo un CAGR del 10.202%, claramente por debajo en retorno puro pero con una historia diferente cuando se mira el riesgo asumido.

Lo importante es no comparar de forma plana. El Beta de la Iteración 3 es 0.356, casi tres veces menor que el del SPY (0.996). Esto significa que el algoritmo solo asume el 39.4% del riesgo sistémico del mercado, y aun así genera un retorno notable. En marzo de 2020, cuando el SPY se hundió un -33.7% en su drawdown máximo, el filtro SMA200 ya había sacado al algoritmo del mercado y el daño quedó acotado en -13.4%. Algo similar ocurrió en el bear market de 2022.

Ajustado por riesgo el panorama da un giro completo. El Alpha del sistema es 0.027, positivo y más alto que el de cualquiera de las iteraciones anteriores. Lo más revelador es que el propio Buy & Hold del SPY arroja un Alpha de -0.002 cuando se ejecuta en las mismas condiciones de backtest. Es decir, dada la cantidad de riesgo sistémico que el SPY

asume (Beta 0.996), no produce valor adicional medible. La Iteración 3, en cambio, produce un Alpha de 0.027 con apenas 0.356 de Beta, lo que demuestra que el algoritmo sí genera rentabilidad ajustada por riesgo sistémico.

Figura 6.1 – Curva de capital de la estrategia Buy & Hold del SPY (2013-2024)

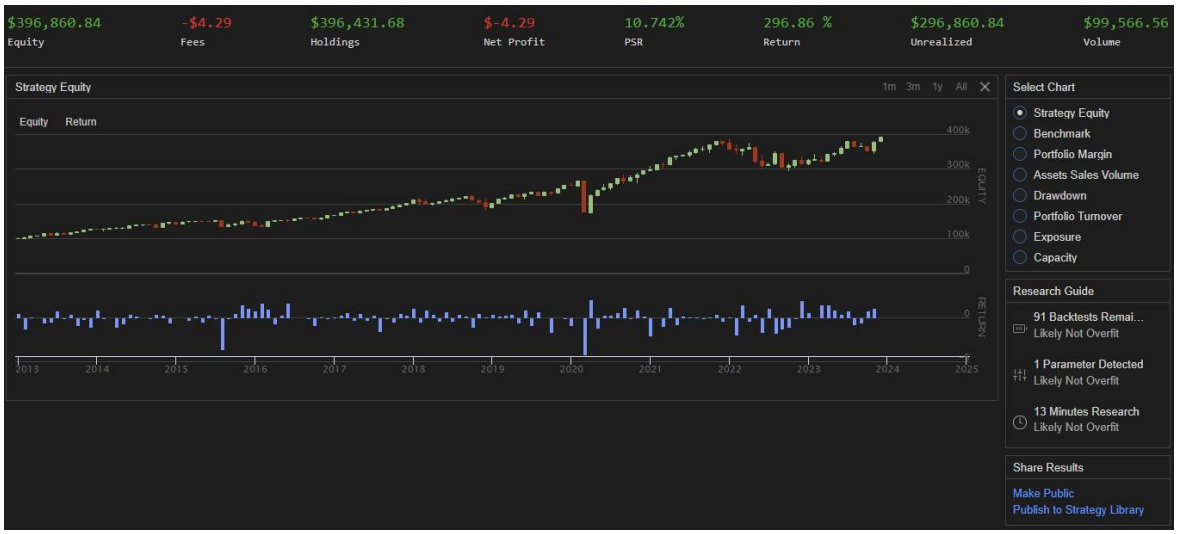
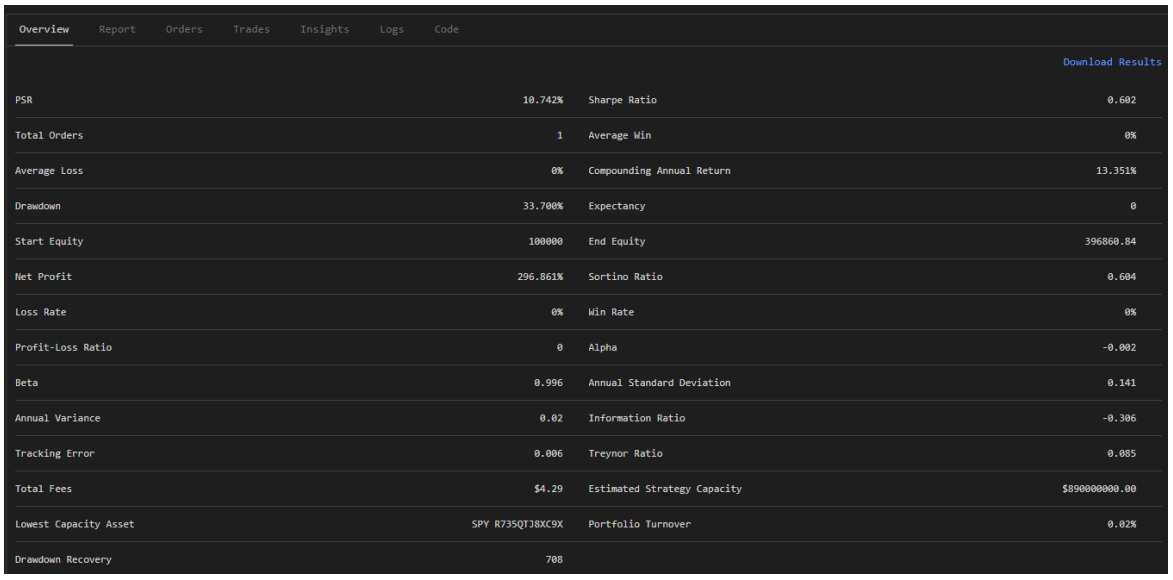


Figura 6.2 – Métricas del backtest Buy & Hold del SPY



PSR	10.742%	Sharpe Ratio	0.602
Total Orders	1	Average Win	0%
Average Loss	0%	Compounding Annual Return	13.351%
Drawdown	33.700%	Expectancy	0
Start Equity	100000	End Equity	396860.84
Net Profit	296.861%	Sortino Ratio	0.604
Loss Rate	0%	Win Rate	0%
Profit-Loss Ratio	0	Alpha	-0.002
Beta	0.996	Annual Standard Deviation	0.141
Annual Variance	0.02	Information Ratio	-0.306
Tracking Error	0.006	Treynor Ratio	0.085
Total Fees	$4.29	Estimated Strategy Capacity	$890000000.00
Lowest Capacity Asset	SPY R735QTJ8XC9X	Portfolio Turnover	0.02%
Drawdown Recovery	708		

6.4.2 Comparación entre Iteraciones

Comparando las iteraciones se ve un patrón claro. La iteración uno, que es el filtro SMA200 con peso parejo, da un CAGR de 9.631% y un Sharpe de 0.525, sostenida por su baja rotación y comisiones bajas, 1,846 dólares en total. La iteración dos, momentum top tres con rebalanceo diario, pierde terreno, el CAGR baja a 8.262% y el Sharpe a 0.436,

porque las comisiones se disparan a 5,084 dólares y se comen la ventaja del momentum.

La iteración tres cierra el desarrollo juntando el Chandelier Exit, la volatilidad inversa, el rebalanceo semanal y el top adaptativo. El CAGR sube a 10.202%, el Sharpe a 0.607 y el drawdown baja a 13.4%, mejores que las dos anteriores. Además el Alpha de 0.027 es el más alto y el Beta de 0.356 el más bajo, dos cosas que juntas sostienen el título de maximizar la rentabilidad ajustada por riesgo. Las comisiones acumuladas, 6,344 dólares, son apenas el 4.8% del capital, lejos del daño que hacían en cuentas pequeñas, y la calidad del Alpha confirma que el valor del algoritmo se ve cuando el capital alcanza para absorber el costo de las comisiones.

6.4.3 Validación Fuera de Muestra (2022-2024)

Para asegurarnos de que el sistema no estaba ajustado solamente al período completo, corrimos la Iteración 3 únicamente en el tramo de 2022 a 2024, que es el más reciente y a la vez el más exigente porque incluye el mercado bajista de 2022. La idea era ver si el sistema seguía rindiendo en datos que no usamos para tomar las decisiones de diseño.

El resultado fue bueno. En ese tramo el sistema cerró con un retorno acumulado del 16.88% y un CAGR del 8.137%, manteniendo el drawdown máximo en 13.5%, o sea que controló bien la caída justo en el año más feo. El Sharpe del tramo fue de 0.295 y el Alpha de 0.026, los dos positivos, lo que indica que el comportamiento no fue casualidad del período largo sino que también se sostuvo fuera de la muestra de desarrollo.

Tabla 4 – Métricas de la validación fuera de muestra (Iteración 3, 2022-2024).

Métrica	Valor (2022-2024)
CAGR (anualizado)	8.137%
Retorno acumulado	16.88%
Ratio de Sharpe	0.295
Ratio de Sortino	0.237
Máximo Drawdown	13.5%
Win Rate	69%
Alpha	0.026
Beta	0.193

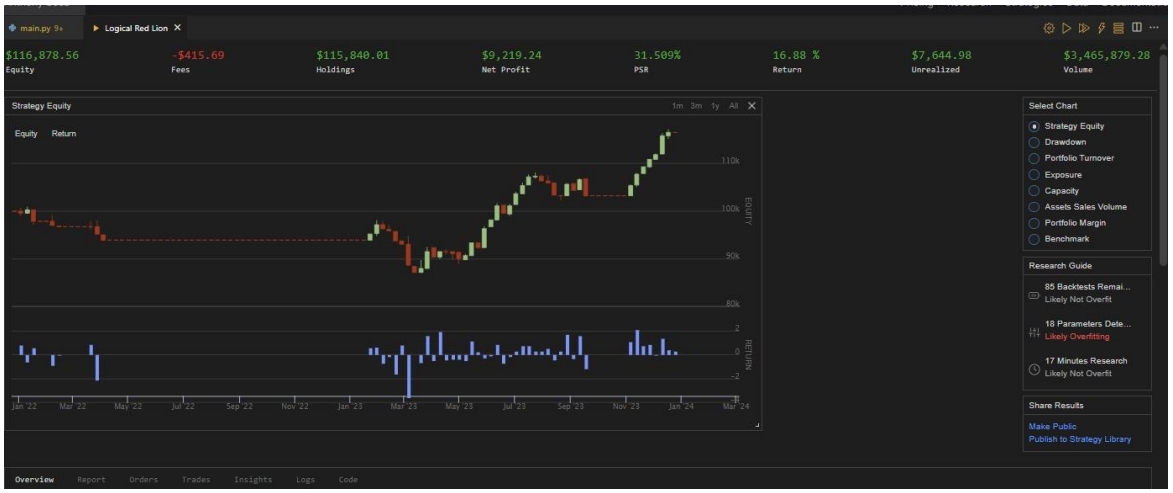


Figura 6.3 – Curva de capital de la validación fuera de muestra (2022-2024).

Hay que ser honestos en que dos años es una ventana corta para sacar conclusiones definitivas, pero el hecho de que la estrategia preservara el capital durante el bajista de 2022 y aun así terminara en positivo es una buena señal de que la lógica del sistema es robusta.

6.5 Análisis Crítico de Resultados y Limitaciones

Los números anteriores se entienden mejor con un poco de contexto crítico.

Rendimiento frente al Buy & Hold. En CAGR puro la Iteración 3 (10.202%) se queda por debajo del Buy & Hold del SPY (13.351%, verificado). Si alguien tiene un horizonte largo y aguanta el riesgo del mercado entero, la estrategia pasiva le deja más dinero al final. El algoritmo gana terreno cuando lo que importa es controlar la exposición y el drawdown, y su ventaja se nota sobre todo en los períodos de mayor riesgo, donde su Beta más baja lo protege.

Costos de fricción manejables. Las comisiones acumuladas en los 11 años fueron de $6,344.43, el 4.84% del capital inicial. Ese dato cambia bastante la lectura del modelo, porque con un capital adecuado el costo de transacción ya no se come la rentabilidad, y ese es el factor que más limita al sistema. En pruebas anteriores con capitales más chicos el mismo monto de comisiones pesaba un montón sobre el capital, más del 40% para una cuenta de $3,000. Usar $100,000 quita esa distorsión y deja que las métricas reflejen la lógica del sistema y no la fricción del costo fijo del broker.

Sobre el drawdown, el peor de la iteración tres fue del 13.4%, muy por debajo del

33.7% del Buy and Hold del SPY. El filtro SMA200 funcionó como protección durante el shock del COVID en marzo de 2020 y durante el bajista de 2022, bajando la exposición justo en los momentos críticos, y el Chandelier Exit ayudó cortando posiciones individuales antes de que el filtro de régimen las sacara.

Sobre la validación fuera de muestra, como ya dije en la Sección 3.4, el tramo de 2013 a 2021 lo usamos para desarrollar y observar el sistema, mientras que el de 2022 a 2024 sirvió como prueba fuera de muestra. Como no optimizamos los parámetros sobre los datos, en cierto sentido todo el período es fuera de muestra, pero separar las ventanas nos deja ver si el sistema se mantiene más allá del rango con el que lo diseñamos. El comportamiento durante 2022 a 2024, con el bajista de 2022 y la recuperación de 2023, salió coherente con la lógica y sin señales de que estuviera sobreajustado a la ventanaanterior.

Limitaciones de liquidez no modeladas. Aunque los ETFs del universo son muy líquidos, el modelo de comisiones no mete el slippage de mercado. En la práctica, las órdenes grandes pueden ejecutarse a precios un poco distintos del cierre, sobre todo en momentos de alta volatilidad como marzo de 2020. Para estos ETFs el efecto es mínimo, pero convendría modelarlo en estudios futuros.

CAPÍTULO 7 – CONCLUSIONES Y TRABAJOS FUTUROS

7.1 Conclusiones

La metodología XP nos sirvió bastante para este proyecto de trading académico. El ir por iteraciones nos dejó validar cada parte por separado con backtesting, detectar problemas temprano, como el golpe de las comisiones en la iteración dos, y arreglarlo después con la gestión de riesgo en la iteración tres. El diseño simple de XP fue clave para no caer en sobreoptimizar los parámetros.

El filtro de régimen con la SMA200 funcionó como protección en los mercados bajistas. El Beta de la iteración tres quedó en 0.356, que confirma que el sistema está mucho menos expuesto al riesgo del mercado que el Buy and Hold del SPY, que tiene Beta 0.996, y eso se nota en un drawdown mucho más controlado, 13.4% contra 33.7%, en los peores momentos como el COVID de 2020 y el bajista de 2022.

La selección por momentum con 63 días sobre seis ETFs concentra el riesgo en menos activos y rota más. La iteración dos, que era momentum top tres con rebalanceo diario, mostró que el momentum solo, sin gestión de riesgo ni control de rotación, queda por debajo de la iteración uno, con un CAGR de 8.262% contra 9.631%. La lección fue clara, el momentum no compensa el costo extra si no se le agregan las técnicas de control que metimos en la iteración tres.

El conjunto de la iteración tres, con el Chandelier Exit de multiplicador 2, la volatilidad inversa, el rebalanceo semanal y el top adaptativo hasta seis, sí agrega valor. Comparando la iteración dos con la tres, el CAGR sube de 8.262% a 10.202%, el Sharpe de 0.436 a 0.607 y el drawdown baja de 27.2% a 13.4%. Y lo más relevante, el Alpha sube de 0.005 a 0.027, el más alto de todas las iteraciones. Esto va en línea con Moskowitz y otros (2012), Kaminski y Lo (2014) y LeBeau y Lucas (1992).

El capital inicial de 100,000 dólares deja que el sistema opere en condiciones realistas. Las comisiones acumuladas en los once años, 6,344 dólares, son apenas el 4.84% del capital, lejos del impacto destructivo que tenían en pruebas anteriores con cuentas pequeñas de 3,000 dólares, donde ese mismo monto de comisiones se comía más del 40% del capital. Con un capital así, lo que se ve en las métricas es la lógica del algoritmo y no el peso del costo fijo del broker.

El CAGR final de 10.202% no le gana al Buy and Hold del SPY, que dio 13.351%

en las mismas condiciones. Esa diferencia tiene sentido por el Beta bajo del sistema, porque al estar expuesto al mercado menos de la mitad del tiempo captura solo una parte de la subida. Pero el Alpha del sistema es positivo, 0.027, y supera al Alpha del Buy and Hold, que es de -0.002, lo que confirma que la iteración tres sí genera rentabilidad ajustada por riesgo. Bajo el criterio de maximizar el Alpha de Jensen, la iteración tres es la que más maximiza esa métrica entre todas las opciones que probamos.

No vimos señales de overfitting. Los parámetros, lookback 63, ATR 14, multiplicador 2, máximo 6 y cooldown 5 días, se fijaron de antemano con base en la literatura, LeBeau y Lucas para el multiplicador, Moskowitz y otros para la volatilidad inversa, Faber para el rebalanceo y Novy-Marx para el horizonte de momentum. Además QuantConnect marcó el backtest final como Likely Not Overfit en su evaluador.

7.2 Limitaciones del Estudio

• El capital inicial fijo en 100,000 dólares, aunque sirve para validar la lógica, no captura los efectos de capacidad que tendría una cuenta enorme operando los mismos ETFs. Para portafolios institucionales habría que considerar el impacto de mercado y el slippage.

• El universo está acotado a seis ETFs de renta variable de Estados Unidos, sin renta fija, materias primas ni ETFs internacionales, lo que limita la diversificación cuando todo el mercado americano cae.

• Los parámetros son fijos y no optimizados, que fue una decisión a propósito para evitar overfitting, pero implica que quizá no sean los mejores para todos los regímenes de mercado, y un estudio de sensibilidad ayudaría a ver qué rangos aguantan.

• El sistema es solo de compra, así que no aprovecha las tendencias bajistas con posiciones cortas en ETFs inversos.

• El modelo de comisiones es simplificado, el de Interactive Brokers es una buena aproximación pero deja fuera el slippage y la diferencia entre compra y venta cuando hay poca liquidez.

7.3 Recomendaciones para Mejoras de la Estrategia

• Operar con más capital, lo primero en la práctica sería usar al menos de 10,000 a 20,000 dólares para que las comisiones pesen menos y el resultado neto se acerque al

rendimiento bruto de la estrategia.

• Probar otros períodos de lookback, como 126 o 252 días, y un modelo de dual momentum que combine el momentum absoluto con el relativo, siguiendo a Antonacci (2014).

• Probar un sizing de posiciones por volatilidad en vez de pesos fijos, dándole menos peso a los activos más volátiles, lo que podría bajar el drawdown.

• Sumar ETFs internacionales, como EFA para mercados desarrollados y EEM para emergentes, para mejorar la diversificación y bajar la correlación con el mercado americano en crisis locales.

• Agregar una validación walk-forward, entrenando en un tramo y evaluando en el siguiente, avanzando la ventana año a año para ver si los parámetros aguantan en el tiempo.

• Investigar una variante que use posiciones cortas en mercados bajistas, con ETFs inversos o margen, para aprovechar los períodos en que el sistema solo se queda en efectivo.

REFERENCIAS BIBLIOGRÁFICAS

Las siguientes referencias bibliográficas sustentan los conceptos teóricos, metodológicos y técnicos utilizados en el desarrollo del presente trabajo de titulación. Se presentan en formato APA, ordenadas alfabéticamente por apellido del autor principal.

Antonacci, G. (2014). *Dual Momentum Investing*. McGraw-Hill Education.

Beck, K. (1999). *Extreme Programming Explained: Embrace Change*. Addison-Wesley Professional.

Faber, M. T. (2007). A quantitative approach to tactical asset allocation. *The Journal of Wealth Management, 9*(4), 69-79.

Jegadeesh, N., & Titman, S. (1993). Returns to buying winners and selling losers: Implications for stock market efficiency. *The Journal of Finance, 48(1),* 65-91.

Kaminski, K. M., & Lo, A. W. (2014). When do stop-loss rules stop losses? *Journal of Financial Markets, 18*, 234-254.

LeBeau, C., & Lucas, D. W. (1992). *Technical Traders Guide to Computer Analysis of the Futures Markets*. Business One Irwin.

Manifesto for Agile Software Development. (2001). *Agile Manifesto.* Recuperado de https://agilemanifesto.org/

Moskowitz, T. J., Ooi, Y. H., & Pedersen, L. H. (2012). Time series momentum. *Journal of Financial Economics, 104*(2), 228-250.

Novy-Marx, R. (2012). Is momentum really momentum? *Journal of Financial Economics, 103*(3), 429-453.

QuantConnect. (s.f.). *LEAN Engine - Documentación oficial*. Disponible en https://www.quantconnect.com/docs/v2/

Schwager, J. D. (2020). *A Complete Guide to the Futures Market: Technical Analysis, Trading Systems, Fundamental Analysis, Options, Spreads, and Trading Principles* (2nd ed.). Wiley.

Vince, R. (2009). *The Handbook of Portfolio Mathematics: Formulas for Optimal Allocation & Leverage*. John Wiley & Sons.

Wilder, J. W. (1978). *New Concepts in Technical Trading Systems*. Trend Research.

ANEXOS

Anexo A: Tabla de Historias de Usuario

Tabla A.1 – Historias de usuario del sistema

ID	Historia de Usuario	Criterios de Aceptación	Prioridad	Iteración
HU-01	Como investigador, quiero filtrar el mercado con	El algoritmo no abre posiciones cuando SPY < SMA200; las posiciones existentes se liquidan	Alta	1

ID	Historia de Usuario	Criterios de Aceptación	Prioridad	Iteración
	SMA200 para operar solo en tendencia alcista			
HU -02	Como investigador, quiero calcular el momentum de 63 días para cada ETF	El sistema calcula retorno (precio_final/precio_inicial - 1) para los 6 ETFs usando 63 velas diarias	Alta	2
HU -03	Como investigador, quiero seleccionar los 3 ETFs con mayor momentum	El sistema ordena los ETFs por retorno de momentum y selecciona los 3 primeros	Alta	2
HU -04	Como investigador, quiero asignar 1/3 del portafolio a cada ETF seleccionado	SetHoldings(symbol, 0.333) se ejecuta para cada uno de los 3 ETFs top	Alta	2

ID	Historia de Usuario	Criterios de Aceptación	Prioridad	Iteración
HU-05	Como investigador, quiero implementar un stop-loss dinámico basado en ATR	El sistema calcula stop = highest_high - 3.0 x ATR(14); liquida si precio < stop	Alta	3
HU-06	Como investigador, quiero un periodo de cooldown tras stop-loss	El sistema no vuelve a comprar el mismo ETF en los siguientes 5 días después de un stop	Media	3
HU-07	Como investigador, quiero realizar backtesting de 2013 a 2024	El backtest corre sin errores desde 2013-01-01 hasta 2024-01-01 con capital inicial de $100,000	Alta	1

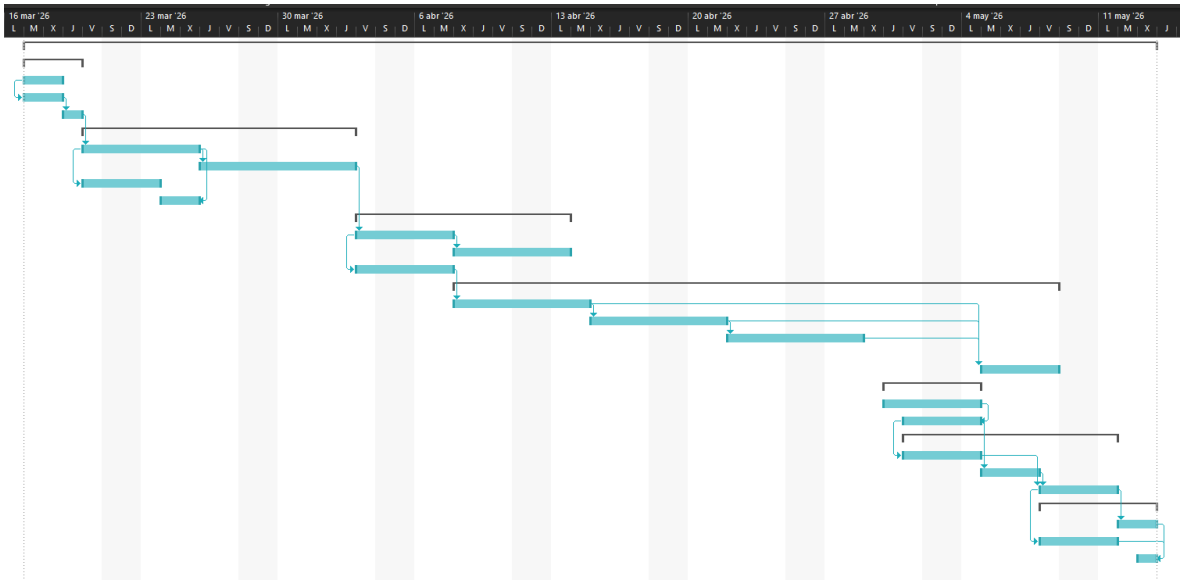
ID	Historia de Usuario	Criterios de Aceptación	Prioridad	Iteración
HU -08	Como investigador, quiero comparar resultados con buy & hold	El sistema registra métricas comparables: CAGR, Sharpe, Max Drawdown	Media	3

Anexo B: Plan de Iteraciones XP (Cronograma)

El desarrollo del algoritmo se organizó en tres iteraciones siguiendo la metodología XP. Cada iteración entregó una parte funcional del sistema, empezando por la base y el filtro de tendencia, después el módulo de selección por momentum y al final los mecanismos de gestión de riesgo.

Figura B.1 – Cronograma del plan de iteraciones XP

Modo de	Nombre de tarea	Duración	Comienzo	Fin
	CRONOGRAMA	**42 días**	**mar 17/3/26**	**mié 13/5/26**
	1. Inicio	**3 días**	**mar 17/3/26**	**jue 19/3/26**
	1.1 Elaboración del plan de tesis	2 días	mar 17/3/26	mié 18/3/26
	1.2 Identificación de requerimientos y alcance	2 días	mar 17/3/26	mié 18/3/26
	1.3 Reunión con director de tesis	1 día	jue 19/3/26	jue 19/3/26
	2. Planificación	**10 días**	**vie 20/3/26**	**jue 2/4/26**
	2.1 Recolección de fundamentos teóricos (momentum, ATR, XP)	4 días	vie 20/3/26	mié 25/3/26
	2.2 Definición de historias de usuario (XP)	6 días	jue 26/3/26	jue 2/4/26
	2.3 Planificación de iteraciones XP (2 días)	2 días	vie 20/3/26	lun 23/3/26
	2.4 Plan de gestión de riesgos del proyecto	2 días	mar 24/3/26	mié 25/3/26
	3. Análisis y diseño	**7 días**	**vie 3/4/26**	**lun 13/4/26**
	3.1 Análisis de requisitos del sistema de trading	3 días	vie 3/4/26	mar 7/4/26
	3.2 Diseño de arquitectura (capas y componentes)	4 días	mié 8/4/26	lun 13/4/26
	3.3 Diseño de módulos (filtro SMA200, momentum, stop dinámico por ATR)	3 días	vie 3/4/26	mar 7/4/26
	4. Desarrollo	**23 días**	**mié 8/4/26**	**vie 8/5/26**
	4.1 Iteración 1: Filtro SMA200 + tests TDD + doc. Cap. 5	5 días	mié 8/4/26	mar 14/4/26
	4.2 Iteración 2: Momentum + selección top-3 y asignación de capital + TDD	5 días	mié 15/4/26	mar 21/4/26
	4.3 Iteración 3: Stop dinámico ATR (Chandelier Exit) + volatilidad inversa + rebalanceo semanal + TDD	5 días	mié 22/4/26	mar 28/4/26
	4.4 Integración y refactorización del código	4 días	mar 5/5/26	vie 8/5/26
	5. Pruebas	**3 días**	**jue 30/4/26**	**lun 4/5/26**
	5.2 Backtesting sobre datos históricos 2013-2024	3 días	jue 30/4/26	lun 4/5/26
	5.3 Validación de métricas (Sharpe, drawdown) + redacción Cap. 6	2 días	vie 1/5/26	lun 4/5/26
	6. Implementación	**7 días**	**vie 1/5/26**	**lun 11/5/26**
	6.1 Comparación con estrategia buy & hold	2 días	vie 1/5/26	lun 4/5/26
	6.2 Análisis de resultados y generación de gráficos	3 días	mar 5/5/26	jue 7/5/26
	6.3 Documentación de lecciones aprendidas (XP)	2 días	vie 8/5/26	lun 11/5/26
	7. Cierre	**4 días**	**vie 8/5/26**	**mié 13/5/26**
	7.1 Redacción de conclusiones	2 días	mar 12/5/26	mié 13/5/26
	7.2 Documentación final completa	2 días	vie 8/5/26	lun 11/5/26
	7.3 Preparación de presentación para defensa	1 día	mié 13/5/26	mié 13/5/26



Anexo C: Código Fuente Completo del Algoritmo

Aquí abajo está el código fuente completo de la versión final del algoritmo, la clase Iteracion3_MomentumPro, hecho en Python para la plataforma QuantConnect con el motor LEAN.

```
from AlgorithmImports import *
class Iteracion3_MomentumPro(QCAlgorithm):
    def Initialize(self):
        self.SetStartDate(2013, 1, 1)
        self.SetEndDate(2024, 1, 1)
        self.SetCash(100000)
        self.symbols = ["SPY", "IWM", "QQQ", "DIA", "XLF", "XLY"]
        self.lookback = 63
        self.atr_period = 14
        self.atr_mult = 3.0
        self.max_n = 6
        self.rebalance_day = 0
        for s in self.symbols:
            self.AddEquity(s, Resolution.Daily)
        self.sma200 = self.SMA("SPY", 200, Resolution.Daily)
        self.atr = {s: self.ATR(s, self.atr_period,
MovingAverageType.Simple,
                                Resolution.Daily) for s in self.symbols}
        self.entry_prices = {}
        self.highest_high = {}
        self.cooldown = {}
        self.SetWarmUp(77)

    def OnData(self, data):
        if self.IsWarmingUp: return
        if not self.sma200.IsReady: return
        for s in self.symbols:
            if self.Portfolio[s].Invested and s in
                self.entry_prices: price = self.Securities[s].Price
                self.highest_high[s] = max(self.highest_high.get(s,
price), price)
        for s in self.symbols:
            if not self.Portfolio[s].Invested: continue
            if s not in self.highest_high: continue
            if not self.atr[s].IsReady: continue
            trailing_stop = self.highest_high[s] - self.atr_mult *
self.atr[s].Current.Value
            if self.Securities[s].Price < trailing_stop:
                self.Liquidate(s)
                self.entry_prices.pop(s, None)
```

```
                self.highest_high.pop(s, None)
                self.cooldown[s] = self.Time
        market_ok = self.Securities["SPY"].Price >
self.sma200.Current.Value
        if not market_ok:
            self.Liquidate()
            self.entry_prices.clear()
            self.highest_high.clear()
            return
        if self.Time.weekday() != self.rebalance_day: return
        momentums = {}
        atrs = {}
        for s in self.symbols:
            if s in self.cooldown:
                if (self.Time - self.cooldown[s]).days < 5: continue
            hist = self.History([s], self.lookback, Resolution.Daily)
            if hist.empty or len(hist) < self.lookback: continue
            closes = hist.loc[s]["close"]
            mom = (closes.iloc[-1] / closes.iloc[0]) - 1
            if mom > 0 and self.atr[s].IsReady:
                momentums[s] = mom
                atrs[s] = self.atr[s].Current.Value
        n = min(self.max_n, len(momentums))
        if n == 0:
            self.Liquidate()
            self.entry_prices.clear()
            self.highest_high.clear()
            return
        top = sorted(momentums, key=momentums.get, reverse=True)[:n]
        inv_vol = {s: 1.0 / atrs[s] for s in top}
        total = sum(inv_vol.values())
        weights = {s: inv_vol[s] / total for s in top}
        for s in self.symbols:
            if self.Portfolio[s].Invested and s not in top:
                self.Liquidate(s)
                self.entry_prices.pop(s, None)
                self.highest_high.pop(s, None)
        for s in top:
            self.SetHoldings(s, weights[s])

    def OnOrderEvent(self, orderEvent):
        if orderEvent.Status == OrderStatus.Filled and
orderEvent.FillQuantity > 0:
            s = orderEvent.Symbol.Value
```

```
            self.entry_prices[s] = orderEvent.FillPrice
            self.highest_high[s] = orderEvent.FillPrice
```

Anexo D: Registro Completo de Operaciones del Backtest

El registro completo de las operaciones ejecutadas durante el backtesting de la versión final del algoritmo (clase Iteracion3_MomentumPro) para el período 2013-2024 puede consultarse y exportarse directamente desde el panel de resultados del backtest en QuantConnect, en las pestañas Orders y Trades:

Pestaña Trades: agrupa las órdenes en operaciones completas, indicando para cada una la fecha de entrada, la fecha de salida, el símbolo, el precio de entrada, el precio de salida, el P&L y las comisiones individuales. En total se registraron 2,212 operaciones durante el período.

Pestaña Orders: contiene el detalle de las 2,212 órdenes individuales ejecutadas durante el período, incluyendo el tipo de orden (market), el estado (filled), la cantidad, el precio de llenado y la comisión aplicada.

Conviene precisar que QuantConnect identifica cada backtest con un nombre generado automáticamente de forma aleatoria, por lo que cualquier archivo exportado conserva ese identificador asignado por la plataforma y no un nombre definido por el autor. Una copia de los resultados exportados se encuentra disponible para consulta y verificación independiente en el siguiente repositorio:

https://drive.google.com/drive/folders/1pNZXos7furgYeHIp9_UNlArfQsPsmB6M?usp=sharing