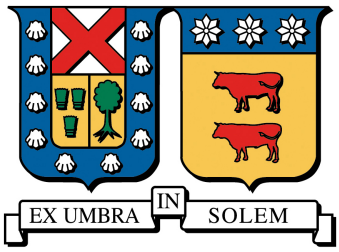

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA

DEPARTAMENTO DE MATEMÁTICA

&

DEPARTAMENTO DE INDUSTRIAS

Deep Reinforcement Learning applied to Statistical Arbitrage Investment Strategy on Cryptomarket

Tesis de Grado presentada por:

Gabriel Vergara Schifferli

como requisito parcial para optar al título de

Ingeniero Civil Matemático

&

Msc. en Ciencias de la Ingeniería Industrial

Otorgado por la Universidad Técnica Federico Santa María

Profesor Guía : Werner Kristjanpoller

Profesor Correferente : Pedro Gajardo

Agosto 2023.

TÍTULO DE LA TESIS:

Deep Reinforcement Learning applied to Statistical Arbitrage Investment Strategy on Cryptomarket

AUTOR:

Gabriel Vergara Schifferli

TRABAJO DE TESIS, presentado en cumplimiento parcial de los requisitos para el título de Ingeniero Civil Matemático & Msc. en Ciencias de la Ingeniería Industrial de la Universidad Técnica Federico Santa María.

COMISIÓN EVALUADORA:

Integrantes	Firma
Werner Kristjanpoller Universidad Técnica Federico Santa María	_____
Pedro Gajardo Universidad Técnica Federico Santa María	_____
Javier Mella Universidad de los Andes	_____

Valparaíso, Chile, Agosto 2023.

AGRADECIMIENTOS

Comenzando por agradecer a mis profesores Werner Kristjanpoller y Pedro Gajardo por haberme dado la posibilidad y confianza para realizar este tema de investigación con autonomía, flexibilidad y creatividad. Sin duda gracias al profesor Pedro por tenerme como ayudante de investigación en el departamento de Matemáticas, la cual fue fundamental para prepararme en temas de investigación, y al profesor Werner por todo el apoyo para incorporarme al programa de postgrado del departamento de Industrias. A su vez agradecerles por el todo el apoyo, disponibilidad y feedback que me dieron durante el desarrollo de la tesis.

También agradecer enormemente a todos mis compañeros quienes sin duda fueron un apoyo fundamental e incondicional para terminar el programa, gracias a ellos no solo por el apoyo académico sino también fuera del ámbito de clases que me motivó para seguir adelante en los momentos más complejos. Además la diversidad de intereses que tenemos en diferentes ramas de la matemática e ingeniería me permitieron tener una perspectiva muy amplia y entender en buena forma cuales son los diferentes focos y herramientas que se estudian y utilizan.

Finalmente, debo agradecer enormemente a mi familia por todo el apoyo y por darme todas las facilidades para poder cumplir mis objetivos, darme guía en los momentos más complejos junto con la confianza absoluta e incondicional de poder recorrer este camino.

RESUMEN

Considerando el aumento al acceso a la información de mercado, en particular el libre acceso a información detallada sobre transacciones de cryptomonedas, junto con la compleja y dinámica propiedad de los mercados financieros, donde se requieren cada vez estrategias de inversión más sofisticadas, el aprendizaje reforzado profundo (DRL) ha demostrado ser exitoso a la hora de generar estrategias de inversión en tiempo real superando a los modelos clásicos. Junto con esto, las estrategias de arbitraje estadístico buscan explotar ineficiencias temporales de mercado para generar retornos. En este aspecto se desarrolla un novedoso método de arbitraje basado en DRL. Primero, se generan portafolios de arbitraje basados en la relación de cointegración. Segundo, se extraen las series de tiempo de estos portafolios y se extraen señales utilizando diversas transformaciones. Finalmente se introduce un método de DRL para generar las decisiones de inversión, donde el agente lee la información de mercado y en función de los objetivos a un horizonte fijo determina las acciones que permitan maximizar la utilidad. Los resultados son analizados fuera de la muestra de entrenamiento, estudiando los riesgos y rendimientos que generan a nivel diario como la trayectoria agregada de decisiones y a nivel intra diario como las acciones individuales realizadas por el agente considerando la presencia y ausencia de costos de transacción. Para verificar la robustez de las decisiones de los agentes se estudia su aleatoriedad y coherencia.

Los resultados empíricos obtenidos respecto a 3 algoritmos diferentes (DQN, PPO y A2C) muestran que el agente logra determinar decisiones que generen retornos positivos, manteniendo utilidad incluso bajo costos de transacción. Además se determina que las acciones que se generan no son aleatorias y logra superar la estrategia base COIN. Junto con esto, se evidencia que existe espacio para estrategias de arbitraje en este mercado y además reducen el riesgo de forma contundente. Respecto a los agentes utilizados, todos muestran un comportamiento estable antes de costos, pero DQN y PPO mantienen rentabilidad después de costos.

Keywords: *deep learning, deep reinforcement rearning, cryptocurrencies, statistical arbitrage, pairs trading, deep reinforcement learning, risk analysis, cointegration*

CONTENIDO

INTRODUCCIÓN

Gatev et al. (2006)[20] introduce el concepto de trading de paridades, el cual refiere a un proceso sencillo el cual sigue dos lineamientros. En primer lugar, se buscan dos activos cuyo precio tenga un movimiento conjunto a lo largo de la historia en un periodo de formación. En segundo lugar, se monitorea la diferencia de precios , *spread*, entre los activos durante un periodo de trading. Si los precios divergen, el spread se abre, entonces se compra el perdedor y se vende el ganador. En el caso de que los activos o acciones seleccionadas tengan una relación de equilibrio, el spread se revertirá a su media histórica. Entonces, las posiciones se reversan y se puede obtener una ganancia a partir de esto. Luego, surge una modificación natural al trading de paridades haciéndolo extenso a un conjunto de activos y no de pares, siendo este el arbitraje estadístico. El arbitraje estadístico engloba una variedad de estrategias de inversión, las cuales identifican y explotan diferencias temporales de precios entre activos similares utilizando herramientas estadísticas, lo cual se entiende como obtener beneficio sobre ineficiencias del mercado. Si bien su forma más básica corresponde al trading de paridades, se buscan activos 'similares' los cuales tengan una relación de 'comovimiento' entre sus precios. Por lo que estas estrategias no apuestan por la dirección del mercado sino en diferencias temporales, o valorizaciones erróneas durante cortos periodos de tiempo en los activos.

Considerando que toda estrategia de arbitraje requiere resolver tres problemáticas fundamentales: Dado un gran numero de activos, ¿ Cuáles son los portafolios de activos similares que representan un movimiento conjunto?, Dado estos portafolios, ¿Cómo cuales son las señales que indican la presencia de una desviación temporal de precios? Finalmente, dadas estas señales, ¿Cómo se deben operar estos activos para optimizar la rentabilidad dadas las condiciones de mercado? En primer lugar, el concepto de obtener activos 'similares' es ambiguo y encontrar portafolios con posiciones largas y cortas no constituye una tarea fácil. En segundo lugar, cual es la información que se debe tomar en cuenta para determinar patrones y las complejas relaciones que permitan determinar estas ineficiencias del mercado, y por último la regla de operación óptima que maximice el objetivo del inversionista dadas estas complejas relaciones del mercado. Estos desafíos requieren de herramientas flexibles

que utilicen la toda la información disponible, por lo que naturalmente el uso de técnicas del aprendizaje profundo que han demostrado ser extremadamente eficientes para resolver problemas en altas dimensionalidades y encontrar complejas relaciones entre las variables. Sin embargo, el problema planteado no corresponde a un problema usual de predicción y categorización puntual, sino a un esquema completo considerando un horizonte y una regla dinámica de decisión.

Considerando estos antecedentes, este proceso está bien representado como una toma de decisiones en linea conforme cambian las condiciones de mercado. De acuerdo con los requerimientos del esquema de arbitraje, el aprendizaje reforzado profundo funciona como un sistema de control óptimo adaptivo directo. El agente en el aprendizaje reforzado profundo (DRL) es la componente que toma las decisiones sobre que acciones realizar para maximizar la recompensa establecida. Entonces, el principio fundamental del DRL es maximizar la recompensa acumulada del agente del proceso de aprendizaje, lo cual lleva a generar estrategias de toma de decisiones óptimas para diferentes tipos de problemas. El aprendizaje reforzado ha demostrado ser exitoso en diferentes tareas, desde la conducción no tripulada[63], navegación de robots [4] hasta jugar videojuegos. No solo ha tenido éxito en cumplir las tareas propuestas sino también en algunos casos sobrepasar a expertos humanos como es el caso de AlphaZero derrotando a campeones mundiales en Shogi, Ajedrez y Go [53]. Por lo tanto surge la pregunta en el ámbito del trading cuantitativo : ¿Es posible entrenar un agente DRL que pueda generar estrategias de arbitraje obteniendo beneficios y reduciendo el riesgo en un mercado de extrema volatilidad y riesgo como lo es el mercado de criptomonedas?

En este trabajo se propone un marco de trabajo unificado que toma elementos de la teoría clásica de arbitraje estadístico e innova en la aplicación del DRL para generar la estrategia de inversión. En primer lugar se tiene un módulo de generación de portafolios de arbitraje por medio de la cointegración, posteriormente se realiza una extracción de señales a partir de los precios y portafolios generados y finalmente el módulo de DRL el cual toma estas señales, las procesa y conforme a la experiencia generada en el entrenamiento toma decisiones óptimas para maximizar el retorno acumulado. Una diferencia fundamental consiste en que las acciones no se generar para maximizar cada acción de manera individual sino en un horizonte dado. Como las relaciones de cointegración de los activos no perduran durante mucho tiempo, deben ser generadas de manera dinámica y además solo se deben utilizar durante un corto periodo de tiempo.

Para construir los portafolios se utiliza una perspectiva estadística en función de la relación de cointegración de las series de precios, de este modo se construyen dos portafolios con diferentes activos que se replican entre sí, generando una neutralidad al mercado al momento de tomar posiciones contrarias en cada uno. Por lo tanto, la diferencia en el valor de estos activos sintéticos captura las desviaciones temporales de los precios de los activos

subyacentes. Posteriormente, para detectar patrones y obtener información relevante sobre el estado del mercado se utilizan diversas transformaciones e indicadores técnicos clásicos sobre la valorización de los activos, en términos de su tendencia, volatilidad y momentum. De este modo, el estado del mercado se descompone en diferentes señales las cuales generan el input al agente DRL o más bien, es la representación del mercado que el agente observa. Finalmente, para generar la regla óptima de inversión, el agente procesa esta información de mercado utilizando una red neuronal y tomando decisiones respecto a esta para maximizar el retorno total de una sesión de trading. Como cada sesión considera un portafolio de arbitraje diferente es necesario generar un simulador de escenario aleatorios sobre el cual pueda entrenarse el agente. Para esto se utiliza una amplia ventana histórica la cual contempla periodos de extrema tendencia y volatilidad junto con periodos menos volátiles y sin tendencia clara, donde los escenarios propuestos corresponden a eventos pasados y no data sintética dado que generar data sintética de mercado no es factible. En este sentido, el entrenamiento del agente se traduce en maximizar la recompensa acumulada durante un horizonte finito tal como si estuviera maximizando la puntuación en un videojuego donde se generan escenarios de manera aleatoria.

El entrenamiento se realizó utilizando una ventana de entrenamiento desde 2020-11-01 00:00:00 hasta 2022- 10-11 05:30:00 siendo un total de 34000 registros con una frecuencia de 30 minutos para 14 activos. Posteriormente el análisis fuera de muestra se realiza durante los periodos 2022-10-19 07:30:00 y termina en 2023-03-18 07:30:00 completando un total de 150 días de operación. Los resultados muestran en primer lugar, que los agentes DRL son capaces de generar estrategias con retornos positivos fuera de muestra. Los resultados son particularmente interesantes dado que se evidencia que una estrategia de arbitraje, en un mercado de extremo riesgo y muy tendencial como son las criptomonedas, es exitosa.

En segundo lugar, las estrategias empleadas logran reducir el riesgo de manera contundente, generando retornos a muy bajo riesgo en comparación con el mercado. Además, al considerar diferentes métricas de rendimiento ajustados por riesgo tanto a nivel diario como a nivel de operación intra día logran generar retornos mayores a los activos de mejor rendimiento y a un menor riesgo.

En tercer lugar, al introducir costos de transacción las estrategias mantienen retornos positivos salgo el agente A2C el cual termina siendo el con peor rendimiento, pero el agente DQN no solo vence a la estrategia de arbitraje estándar COIN sin costos de transacción sino que además la logra mejorar considerando estos costos, junto con mantener niveles de riesgo más bajos que el mercado.

Por último y cuarto lugar, se logra determinar que las acciones generadas por lo agentes no son aleatorias, sino que las políticas entrenadas generan acciones fundamentadas y no de manera errática, lo cual es consistente con los resultados obtenidos. De este modo se determinar que es posible generar una estrategia de arbitraje utilizando DRL para mejorando

los métodos estadísticos clásicos.

El documento que continúa se organiza de la siguiente manera. El capítulo 1 introduce los temas de la teoría de arbitraje y revisa los trabajos elementales en este ámbito. Capítulo 2 introduce el concepto del aprendizaje reforzado profundo, su problemática elemental de los procesos de decisión y los algunos algoritmos base sobre el cual se estructuran estos métodos. Capítulo 3 establece la conexión de como generar una estrategia de arbitraje en un marco de trabajo compatible con el aprendizaje reforzado profundo e introduce la metodología empleada para el entrenamiento del modelo y su especificación. Capítulo 4 analiza los resultados obtenidos en diferentes aspectos, se hace un análisis a nivel diario e intra diario del riesgo de las estrategias junto con el rendimiento ajustado por riesgo, y finalmente se estudia el actuar del agente. Por último, el capítulo 5 concluye el trabajo.

TRABAJOS RELACIONADOS

El trabajo es estructura sobre la teoría clásica del arbitraje estadístico, en el cual se resuelven las problemáticas de generación de portafolios, extracción de señales y asignación del capital de manera independiente. Los métodos clásicos para la generación de portafolios de arbitraje se centran principalmente en la obtención de múltiples paridades o varios portafolios de activos, utilizando métodos por distancia [20], por cointegración [59],[18], [46], por análisis de componentes principales [2] u otros métodos como el caso de cointegración parcial [9] o cópulas [65].

Por otro lado, la gran mayoría de aplicaciones de machine learning al campo del trading se resume en la predicción de retornos e.g. Qiu et al. (2016)[48] utilizó técnicas de reducción de dimensionalidad junto con redes neuronales para predecir la dirección de los retornos, Zhong et al. (2017)[68] aplicó redes neuronales para la predicción de retornos en el mercado Japonés. También otros métodos de deep learning y machine learning sea han aplicado al contexto de predicción de precios [26],[66],[45],[32].

En la gran mayoría de las aplicaciones sobre predicción de precios, las estrategias consideran en tomar posiciones largas en los activos ganadores y cortas en los perdedores en función de una predicción, un umbral o algún modelo paramétrico e.g. de reversión a la media pero siempre la estrategia de inversión se implementa como módulo anexo al mecanismo de generación de portafolios o predicción. En este aspecto las aplicaciones de DRL han innovado en este aspecto uniendo estos dos aspectos que suelen considerarse en módulos separados. Si bien el aprendizaje reforzado no es una técnica nueva, esta surge desde la teoría del control óptimo, una de las primeras aplicaciones de esta técnica aplicada a estrategias de inversiones es Moody et al. (1998) [42] donde construyen un sistema de trading de acciones a través de aprendizaje reforzado, cuya entrada es data financiera sin procesar. Neves et al.(2018)[5] genera un sistema de especulación de corto plazo en el mercado de divisas basado en aprendizaje reforzado, Deng et al.(2016)[11] proponen un sistema de DRL con redes recurrentes y procesamiento de señales para la toma de decisiones de inversión, Wu et al.(2020)[64] proponen un método de trading adapitvo basado en DRL utilizando métodos propuestos de GDQN (Gated Deep Q-learning) y GDPG (Gated Deterministic Polici Gra-

dient) con resultados positivos, Zhang et al. 2020 [67] comparan DRL en diversos mercados utilizando una función de recompensa que incorpora un nivel de volatilidad condicional objetivo, obteniendo resultados que superan a los métodos clásicos y manteniendo ganancias bajo altos costos de transacción.

Por otro lado, aplicaciones de DRL en aspectos de arbitraje son más escaza, Mulvey et al. (2020)[43] y Kim and Kim (2019)[29] utilizan métodos de machine learning en modelos paramétricos de arbitraje estadístico, por otro lado, Pelger et al. 2021[23] generaliza el modelo de arbitraje con la utilización análisis de componentes principales (PCA e IPCA) y factores alpha (Fama French) , combinando estos portafolios de arbitraje con redes convolucionales combinadas con trasnformers, análisis de fourier y redes feed forwad y el modelo paramétrico de ornstein-uhlenbeck con umbrales para la generación de decisiones es de inversión.

Capítulo 1

TEORÍA DE ARBITRAJE ESTADÍSTICO

Comenzando con conceptos básicos que servirán para el entendimiento específico de esta sección, se introduce desde el modelo CAPM.

Modelo CAPM

El modelo CAPM corresponde al acrónimo *Capital Asset Pricing Model* fué propuesto originalmente por William T. Sharpe. El impacto que el modelo ha tenido en el área de las finanzas se puede evidenciar directamente en el simple frecuente uso de la palabra *beta*. En el mundo contemporáneo de las finanzas , la palabra *beta* no hace referencia a la letra griega sino a su significado derivado del modelo CAPM.

Junto con la idea del beta, el modelo CAPM sirvió también para formalizar la idea de un portafolio de mercado, el cual en términos de este modelo actúa como un proxy del mercado.

En base a estos conceptos de beta y portafolio de mercado, CAPM intenta explicar los retornos de activos como una suma agregada de sus componentes. Por un lado un factor responde a la componente de mercado o sistémica y el segundo a una componente residual o no sistémica.

Considerando r_a el retorno del activo, r_m el retorno del portafolio de mercado se tiene el modelo sobre el retorno del activo en función del portafolio de mercado como:

$$r_a = \beta r_m + \theta_a \tag{1.0.1}$$

esta ecuación también se conoce como la recta de mercado o *Security Market Line* (SML).

De este modo, el retorno por componente de mercado o sistémico se determina como una ponderación del retorno del portafolio de mercado por el beta, mientras que la componente no sistémica por el θ. Este beta sirve como sensibilidad frente a movimientos de mercado, también se denota Δ a una sensibilidad frente a la variación directa de $1\,\%$ del subyacente , o en caso de tasas de interés 1 punto base ($0.01\,\%$), por lo tanto denotando al beta como $\Delta_{market} = \beta$ (delta de mercado), se tiene que frente a un aumento de $1\,\%$ del retorno de mercado el retorno del activo aumentará en $\beta\,\%$.

La componente no sistémica θ_a es el retorno residual del portafolio, corresponde a la porción del retorno no explicada por la componente de mercado, la cual se espera que tenga un valor de esperanza nulo.

Con esta separación de componentes se tiene hipótesis clave sobre sus factores. La afirmación del modelo es que las componentes de mercado y no sistémicas no están correlacionadas. Independiente de la validez empírica del modelo y sobre sus hipótesis, para el propósito de este documento no es relevante discutirlas puesto que solo se busca generar la noción que hay detrás del modelo. En este aspecto, si se determina la sensibilidad de un activo frente al mercado es posible cuantificar la exposición de un portafolio frente al mercado.

Estrategia Neutral al Mercado

El modelo CAPM sirve como herramienta para presentar el concepto de ser *Neutral al mercado* o más generalmente el ser neutral en términos financieros.

Las estrategias neutrales al mercado corresponden a estrategias que son neutrales a los retornos de mercado, sus retornos no son correlacionados con los del mercado. A pesar de los movimientos de mercado, ya sean alzas o bajas, periodos alcistas o bajistas, las estrategias neutrales al mercado tienden a tener un rendimiento más estable con menor volatilidad. Esto se obtiene utilizando portafolios neutrales al mercado.

Los portafolios neutrales al mercado, siguiendo el ejemplo del modelo simple pero ilustrativo CAPM, se define como un portafolio cuyo beta es nulo. Esto se traduce, en términos del modelo, que los retornos del portafolio no tienen componente sistemática, es decir, no hay componente del retorno que pueda ser explicada por variaciones de mercado y los retornos son únicamente retornos residuales no correlacionados con el mercado.

De este modo, al utilizar portafolios neutrales al mercado se centra únicamente en el comportamiento de los retornos residuales y no al estado del mercado. Al asumir que los retornos residuales tienen media nula se espera un tener un fuerte comportamiento de reversión a la media, donde los retornos oscilan en torno al valor de media. El comportamiento de reversión a la media puede ser explotado en el proceso de predicción de movimientos el cual constituye la estrategia de arbitraje.

Siguiendo el modelo CAPM, considerando portafolios solamente de posiciones largas se buscan dos portafolios que tengan sensibilidades contrarias, esto corresponde a un portafolio que tenga un beta positivo (se mueve en la dirección del mercado) y otro con beta negativo

(se mueve en dirección contraria al mercado). De esta manera, al tener dos portafolios de igual sensibilidad pero dirección contraria se obtienen la neutralidad al mercado asumiendo que ambos pesan lo mismo, por lo que si se encuentra un comportamiento de reversión a la media respecto a los retornos residuales del par, se puede generar una estrategia neutral al mercado. Por otro lado, si se tienen ambos portafolios con betas positivos, tomando una posición corta sobre uno de los portafolios se obtiene un beta negativo. Para los casos de posiciones opuestas el monto total de la posición corta se utiliza para la posición larga siendo una posición neta nula con beta nulo. Por este motivo es que también suelen llamarse portafolios dólar neutral.

A modo de ejemplo, considerando dos portafolio A y B, con betas positivos β_A y β_B y sus retornos r_A, r_B respectivamente

$$r_A = \beta_A r_A + \theta_A \tag{1.0.2}$$

$$r_B = \beta_B r_B + \theta_B \tag{1.0.3}$$

se obtiene el portafolio AB considerando una posición de c unidades sobre el A y una unidad sobre el B. Así, el retorno del portafolio se obtiene como: $r_{AB} = -cr_A + r_B$ y finalmente se obtiene el modelo:

$$r_{AB} = (-c \cdot \beta_A + \beta_B) + (-c \cdot \theta_A + \theta_B) \tag{1.0.4}$$

entonces combinando los portafolios en una proporción tal que $-c \cdot \beta_A + \beta_B = 0$ se obtiene que $c = \frac{\beta_B}{\beta_A}$ y así el portafolio AB es neutral al mercado. Sin embargo, un portafolio neutral al mercado no necesariamente es dólar neutral o vice versa, para ser dólar neutral se requiere que los betas de los portafolios sean iguales.

Trading de Paridades

Ya introducidos los conceptos anteriores, se puede entrar en la definición del trading de paridades. El nombre viene precisamente de la constricción de portafolios de pares de activos, los cuales se seleccionan para obtener la neutralidad al mercado combinando una posición larga y otra corta en una proporción determinada. Para estos portafolios se define el *spread* como la diferencia de las cotizaciones entre estos activos. Este *spread* se relaciona con el retorno residual del portafolio, por lo que estas estrategias apuestan al spread y no al valor de los activos. Para ello se espera que si el spread está muy lejos de su media este debería volver a su valor de media, por lo tanto, las posiciones se realizan en términos del spread , si está muy por sobre la media o muy por debajo se toman las posiciones respectivas considerando que el spread convergerá a la media nuevamente.

La teoría de arbitraje de paridades se basa en precios relativos de activos la cual busca explotar ineficiencias de corto plazo en el mercado producto de valorizaciones erróneas en los activos y volatilidad, la cual se lleva a cabo tomando posiciones opuestas en activos

generando retornos a partir de la diferencia de precios de los activos y no en su precio individual. Finalmente esto se extrapola a una cantidad arbitraria de activos generando portafolios que mantengan propiedades adecuadas para arbitrar.

Las preguntas fundamentales para poder realizar un esquema de arbitraje estadístico se formulan en base a las características necesarias para poder realizar un esquema de arbitraje: ¿Cómo se obtiene seleccionan los pares de activos?, ¿Cómo se calcula el spread?, ¿Cuál es la proporción se debe combinar entre los activos? y ¿Cómo se determina cuando el spread ha divergido sustancialmente de su media y cuando volverá a su media?.

1.1. Arbitraje por Distancia

El trabajo original que comenzó el concepto de trading de paridades fue introducido por Gatev et al. (2006) [20] el cual fue empleado en el mercado de acciones estadounidenses líquidas.

El método por distancia tiene como base los siguientes elementos; Selección de activos, Normalización de precios, calculo de distancia, selección de paridades y finalmente la lógica de entrada y salida. Entonces, seleccionados los activos de interés, la normalización de precios consiste en un periodo de formación de 12 meses en el cual para un conjunto de activos $\{A_i\}_{i\in\mathcal{I}}$ los precios P_t^i son normalizados mediante el máximo y mínimo del periodo de formación. Considerando $P_{t,norm}^i$ el precio normalizado del activo i:

$$M_i^T := \max_{t\leq T} P_t^i,\ m_i^T := \min_{0\leq t\leq T} P_t^i, \quad P_{t,norm}^i = \frac{P_t^i - m_i^T}{M_i^T - m_i^T} \tag{1.1.1}$$

de esta manera, $P_{t,norm}^i \in [0,1]\ \forall t \in [0,T],\ \forall i \in \mathcal{I}$,donde T corresponde al periodo de formación y $\mathcal{I}$ al conjunto de activos seleccionados.

Otra manera de realizar una estandarización de los precios es construyendo un índice de retorno acumulado para cada activo, considerando r_t^i el log-retorno del activo i, entonces el retorno acumulado $r^i[t] = \sum_{k=0}^t r_t^i$ donde $P_t^i = P_0^i e^{r^i[t]}$.

Una vez normalizados los precios se calcula la distancia entre los diferentes activos :

$$\mathrm{SSD}_{i,j} = \sum_{k=0}^{T} (P_{t,norm}^i - P_{t,norm}^j)^2 \tag{1.1.2}$$

de esta manera, se obtienen $C_{|\mathcal{I}|}^2$ paridades diferentes y se escoge usualmente el 10 % con menor distancia entre si. Luego, una posición sobre la paridad i,j se abre cuando el spread $s_t^{i,j} = p_t^i - p_t^j$ (con p_t^k los precios normalizados) supera el umbral de 2 desviaciones estándar

y se cierra cuando se revierte a la media. En términos muestrales empíricos,

$$s_{i,j}^2 = \frac{1}{T}\sum_{t=1}^{T}(p_t^i - p_t^j)^2 - \left(\frac{1}{T}\sum_{t=1}^{T}(p_t^i - p_t^j)\right)^2 \tag{1.1.3}$$

$$\bar{\text{SSD}}_{i,j} = \sum_{k=0}^{T}(p_t^i - p_t^j)^2 = s_{i,j}^2 + \left(\frac{1}{T}\sum_{t=1}^{T}(p_t^i - p_t^j)\right)^2 \tag{1.1.4}$$

y la media,

$$\bar{s}_{i,j} = \frac{1}{T}\sum_{t=1}^{T} s_t^{i,j} \tag{1.1.5}$$

Entonces, los parámetros de la estrategia consiste en un periodo de formación e.g. 12 meses para posteriormente durante el periodo de trading e.g. 6 meses utilizar estos parámetros (valores de normalización , media y desviación estándar) para ejecutar la lógica de operación. De esta manera, luego del periodo de formación, las posiciones se toman cuando $|s_t^{i,j} - \bar{s}_{i,j}| \geq 2s_{i,j}^2$ y se cierran cuando $s_t^{i,j}$ revierte a su media $\bar{s}_{i,j}$.

Esta estrategia mostró tener retornos anormales durante el periodo de estudio utilizado 1962 - 2002, sin embargo presenta varios problemas tal como esta planteada, uno de ellos el hecho de que varias de las paridades encontradas bajo esta distancia no convergen (Do y Faff 2010)[13], la rentabilidad del método decae de forma importante posterior al año 2009 (Do y Faff 2012) [12] y la métrica de distancia no es analíticamente óptima (Krauss et. al 2017) [31] puesto que para maximizar los retornos de esta estrategia se deben buscar paridades que presenten un movimiento conjunto pero que exhiban alta volatilidad y una fuerte reversión a la media, entonces la paridad 'ideal' bajo la métrica propuesta sería una paridad cuya SSD sea cero, lo cual no entregaría retorno alguno y la selección de paridades propuesta por GGR es propensa a optar por pares que presentan baja volatilidad limitando los rendimientos.

Modificaciones a esta estrategia se pueden realizar directamente cambiando la métrica de distancia o comovimiento, el periodo de formación, la cantidad de desviaciones estándar, utilizar una ventana móvil, entre otras alternativas. Chen et al. (2019) [6] optan por identificar las paridades mediante correlación de Pearson en sus retornos, luego con las correlaciones en los retornos para cada par se construye una métrica de divergencia $d^{i,j}$ entre el activo i y su par j:

$$d_t^{i,j} = \beta(r_t^i - r_f) - (r_t^j - r_f) \tag{1.1.6}$$

donde β corresponde al coeficiente de regresión de los retornos del activo i respecto a j con r_f la tasa libre de riesgo. Considerando un caso univariado se selecciona el par con mayor correlación mientras que para el caso cuasi-multivariado corresponde al 50 % de los activos con mayor correlación realizando un portafolio equiponderado de comovimiento. Luego se

ordenan todas las acciones en orden descendente respecto a la divergencia de retorno del mes anterior, donde se toma una posición larga en el decil 10 y corta en el decil 1 con un periodo de mantención de 1 mes.

A diferencia de la métrica de selección SSD en el nivel de precios, la correlación en el nivel de retornos :

$$s^2_{r_i - r_j} = s^2_{r_i} + s^2_{r_j} - 2\hat{\rho}_{r_i, r_j} s_{r_i} s_{r_j} \tag{1.1.7}$$

donde r_i corresponde al retorno del activo i. Considerando para correlaciones altas entre activos, la volatilidad del par es decreciente en la correlación sin embargo, los niveles de volatilidad para cada activo pueden variar significativamente. Entonces, considerando el caso 'ideal' de correlación 1, pero con el retorno de un activo siempre siendo 2 veces el otro, la métrica de divergencia de retornos captura este efecto de comovimiento mientras que la métrica de distancia SSD no lo hace. Adicionalmente, al realizar la estrategia con un portafolio de activos en vez de una paridad aporta una mayor diversificación con un nivel de información mayor. Sin embargo, una correlación en el nivel de retornos no necesariamente comparte una relación de equilibrio y no se tiene fundamento para que la medida de divergencia exhiba un efecto de reversión a la media.

1.2. Arbitraje por PCA

El modelo de arbitraje mediante análisis de componentes principales fue introducido por Avellaneda y Lee (2010)[2] donde se aplicó un método de PCA utilizando EFTs sectoriales. Para ello se consideraron los residuos de los retornos de acciones utilizando PCA y modelando la señal de trading como un proceso de reversión a la media.

Considerando $\{r_i\}_{i \in \mathcal{I}}$ los retornos de activos y F el retorno de un portafolio de mercado. Entonces para cada activo se tiene el modelo de retorno:

$$r_i = \beta F + \tilde{r}_i \tag{1.2.1}$$

descomponiendo el retorno de un activo en una componente sistémica $\beta_i F$ y otra componente idiosincrática o no correlacionada $\tilde{r}_i$, alternativamente se considera un modelo multifactorial:

$$r_i = \sum_{j=1}^{m} \beta_{i,j} F_j + \tilde{r}_i \tag{1.2.2}$$

En este caso, se consideran m factores los cuales pueden ser interpretados como retornos de portafolios de referencia representando la componente sistémica. Un portafolio se considera

neutral al mercado cuando la posición tomada en cada activo $\{Q_i\}_{i\in\mathcal{I}}$ es tal que

$$\bar{\beta}_j = \sum_{i\in\mathcal{I}} \beta_{i,j} Q_i = 0, \quad \forall j \tag{1.2.3}$$

de este modo, los coeficientes $\bar{\beta}_j$ corresponden a los betas del portafolio o las proyecciones del los retornos del portafolio en los diferentes factores. Entonces, un portafolio neutral al mercado consiste en betas nulos, por ello no está correlacionado con el portafolio de mercado o factores sistémicos de los retornos de mercado. Entonces los retornos del portafolio cumplen

$$\sum_{i\in\mathcal{I}} Q_i r_i = \sum_{i\in\mathcal{I}} Q_i \left[\sum_{j=1}^{m} \beta_{i,j} F_j\right] + \sum_{i\in\mathcal{I}} Q_i \tilde{r}_i \tag{1.2.4}$$

$$= \sum_{j=1}^{m} \left[\sum_{i\in\mathcal{I}} \beta_{i,j} Q_i\right] F_j + \sum_{i\in\mathcal{I}} Q_i \tilde{r}_i \tag{1.2.5}$$

$$= \sum_{i\in\mathcal{I}} Q_i \tilde{r}_i \tag{1.2.6}$$

por lo tanto, un portafolio neutral al mercado está únicamente afectado por retornos en la componente idiosinctrática.

Para generar los factores se estandarizan los retornos considerando una ventana de M intervalos de tiempo, entonces se tiene el retorno estandarizado

$$y_t^i = \frac{r_t^i - \bar{r}_i}{s_i} \tag{1.2.7}$$

con s_i la desviación estándar y $\bar{r}_i$ el retorno medio en los últimos M tiempos. Entonces, se obtiene la correlación empírica

$$\hat{\rho}_{ij} = \frac{1}{M-1} \sum_{t=1}^{M} y_t^i y_t^j \tag{1.2.8}$$

Luego para extraer información de la data se utiliza un análisis de componentes principales. Para ello se consideran vectores propios y valores propios de la correlación empírica y se ordenan los valores propios de forma decreciente:

$$\mathcal{I} \geq \lambda_1 \geq \lambda_2 \cdots \geq \lambda_{\mathcal{I}} \geq 0 \tag{1.2.9}$$

y los vectores propios correspondientes como:

$$v^{(j)} = (v_1^{(j)}, \ldots, v_{\mathcal{I}}^{(j)}), \quad j \in \mathcal{I} \tag{1.2.10}$$

de esta manera, la varianza porcentual explicada por el $j-$ésimo valor propio corresponde al cociente $\lambda_j / \sum_{i\in\mathcal{I}} \lambda_i$. Entonces cada valor propio λ_j está relacionado con un portafolio propio E_j compuesto por los diferentes activos, cuyos pesos están relacionados por su vector propio $v^{(j)}$ y volatilidad s_j:

$$w_{i,j} = \frac{v_i^{(j)}}{s_i}, \quad \bar{w}_{i,j} = \frac{w_{i,j}}{\sum_{i\in\mathcal{I}} w_{i,j}} \tag{1.2.11}$$

entonces el $j-$portafolio propio está definido a través del vector de pesos:

$$\bar{w}_j = (\bar{w}_{1,j}, \ldots, \bar{w}_{\mathcal{I},j}) \tag{1.2.12}$$

y su retorno

$$F_{j,t} = \bar{w}_j \mathbf{r}_t^T, \quad \mathbf{r}_t = (r_t^1, \ldots, r_t^{\mathcal{I}}) \tag{1.2.13}$$

finalmente se debe considerar que los retornos entre dos portafolios propios no están correlacionados por construcción. Los valores propios corresponden a portafolios propios compuestos por los diferentes activos, cuyos pesos están dados por su vector propio y la volatilidad. Con estos portafolios propios, que por construcción son no correlacionados, son utilizados como factores del modelo multifactorial :

$$r_t^i = \sum_{j=1}^{m} \beta_{i,j} F_{j,t} + \tilde{r}_t^i \tag{1.2.14}$$

de esta manera, $\tilde{r}_t^i$ corresponde a la señal de trading para cada activo en donde las posiciones se toman largas (cortas) en el activo i mientras que se toman posiciones cortas (largas) en los portafolios propios con pesos $\beta_{i,j}$.

1.2.1. Generación de Señal

Se propone un método de valorización de activos basado en el rendimiento relativo a sectores industriales o los factores de PCA. Asumiendo dos acciones sobre un mismo sector industrial o de características similares, P y Q tales que presenten un comovimiento, se espera que los retornos de ambos se rastreen entre si luego de controlar por beta. Considerando P_t y Q_t las series de sus precios respectivamente, se puede modelar el sistema como:

$$\ln(P_t/P_{t_0}) = \alpha(t - t_0) + \beta \ln(Q_t/Q_{t_0}) + X_t \tag{1.2.15}$$

que en su forma diferencial se expresa como:

$$\frac{dP_t}{dt} = \alpha dt + \beta \frac{dQ_t}{dt} + dX_t \tag{1.2.16}$$

donde X_t es un proceso estacionario o de reversión a la media, el cual será referido como el residual y el parámetro α se denota *drift*.

Generalizando a N acciones a tiempo continuo $S_1(t), \ldots, S_N(t)$, donde t es el tiempo, se formula en términos de un modelo multifactorial:

$$\frac{dS_i(t)}{dt} = \alpha_i dt + \sum_{j=1}^{N} \beta_i \frac{dI_j(t)}{dt} + dX_i(t) \tag{1.2.17}$$

donde el término

$$\sum_{j=1}^{N} \beta_{i,j} \frac{dI_j(t)}{dt}$$

corresponde a la componente sistemática del retorno (desde los retornos de los portafolios propios) . Aplicando a portafolios compuestos por ETFs, $I_j(t)$ representa el precio de mercado del $j-$ésimo ETF utilizado para generar el mercado mientras que los factores $\beta_{i,j}$ son los pesos respectivos. El componente idiosincrático o no sistémico viene dado por los términos :

$$\alpha_i dt + dX_i(t)$$

donde α_i es el drift de la parte idiosincrático, e.g. $\alpha_i dt$ es el exceso de retorno en relación al mercado. Luego, el término $dX_i(t)$ se asume como el incremento de un proceso estacionario que modela las fluctuaciones de precios correspondiente a las sobre reacciones de mercado o fluctuaciones idiosincráticas. Asumiendo que el drift mide las desviaciones sistemáticas del sector y que las fluctuaciones son procesos de estocásticos de reversión a la media sobre el mercado, mediante test estadístico se acepta o rechaza el modelo para cada activo y se construye la estrategia de arbitraje.

Para modelar el spread se introduce el siguiente modelo paramétrico:

Definition 1.2.1: ***Proceso de Ornstein Uhlenbeck:***

Sea $X(t)$ un proceso estocástico, se define un proceso de acuerdo a la dinámica:

$$dX(t) = \kappa(\mu - X(t))dX(t) + \sigma dW(t), \quad \kappa, \sigma > 0 \tag{1.2.18}$$

Este proceso es estacionario y autoregresivo de orden 1 AR(1). En particular $dX(t)$ tiene esperanza incondicional nula pero su esperanza condicional es

$$\mathbb{E}\left[dX(t)|X(s), s \leq t\right] = \kappa(\mu - X(t))dt$$

la cual cambia de signo según $\mu - X(t)$.

Los parámetros α_i, κ_i y σ_i son específicos para cada activo. Se puede reescribir en términos de incrementos temporales uniformes Δt:

$$X_i(t_0 + \Delta t) = e^{-\kappa_i \Delta t} X_i(t_0) + (1 - e^{-\kappa_i \Delta t})\mu_i + \sigma_i \int_{t_0}^{t_0+\Delta t} e^{-\kappa(t_0+\Delta t-s)} dW_i(s) \quad (1.2.19)$$

luego, tomando $\Delta t \to \infty$ se tiene que la distribución de equilibrio para $X_i(t)$ es normal con momentos:

$$\mathbb{E}[X_i] = \mu_i, \quad \mathbb{V}[X_i] = \frac{\sigma_i^2}{2\kappa_i}$$

entonces se tiene la esperanza condicional del residual para cada activo a 1 día

$$\alpha_i dt + \kappa_i(\mu_i + X_i(t))dt$$

finalmente, para determinar las entradas y salidas se define el score considerando $\alpha = 0$ y la volatilidad de equilibrio:

$$\sigma_{eq,i} = \frac{\sigma_i}{\sqrt{2\kappa_i}}, \quad s_i = \frac{X_i(t) - \mu_i}{\sigma_{eq,i}}$$

y para el caso de $\alpha \neq 0$:

$$s_{mod,i} = s_i - \frac{\alpha_i}{\kappa_i \sigma_{eq,i}}$$

donde

$$\text{Comprar apertura si } s_i < -\bar{s}_{bo} \quad (1.2.20)$$

$$\text{Vender apertura si } s_i > +\bar{s}_{so} \quad (1.2.21)$$

$$\text{Cerrar posición corta si } s_i < +\bar{s}_{bc} \quad (1.2.22)$$

$$\text{Cerrar posición larga si } s_i > -\bar{s}_{sc} \quad (1.2.23)$$

con la apertura de una posición se hace comprando 1 unidad del activo correspondiente y vendiendo $\beta_{j,i}$ unidades del set de ETFs. Mientras que los umbrales de apertura y cierre se determinan durante un periodo de entrenamiento. Cabe notar que la incorporación del drift suele ser despreciable.

1.3. Arbitraje por Cointegración

El método por cointegración viene desde la base teórica de los modelos de series de tiempo coitnegrados de Engel y Granger (1987) [15], donde Galenko (2012) [18] generan una estrategia de trading en base a esta propiedad que se detalla a continuación:

Definition 1.3.1: *Un proceso estocástico Y_t es estacionario si su primer y segundo momento son invariantes:*

$$\mathbb{E}[Y_t] = \mu \ \forall t$$
$$\mathbb{E}\left[(Y_t - \mu)(Y_{t-h} - \mu)^T\right] = \Gamma_Y(h) = \Gamma_Y(-h)^T \ \forall t, h = 0, 1, 2, \ldots$$

donde μ es finito en todos sus términos, y $\Gamma_Y(h)$ es una matriz finita de covarianza.

Luego, un proceso univariado Y_t es integrado de orden d , $Y_t \sim I(d)$ si el proceso original Y_t no es estacionario pero el proceso $d-$diferenciado es estacionario, un proceso integrado de orden 0 es estacionario.

Definition 1.3.2: *Considerando $X_t = (x_t^1, x_t^2, \ldots, x_t^n)^T$ un proceso multivariado tal que todas sus componentes son estacionarias, $x_t^i \sim I(0)$ y existe $b \in \mathbb{R}^n$ tal que $b^T X_t \sim I(0)$, entonces X_t es cointegrado y b son sus coeficientes de cointegración.*

Considerando un proceso de precios $P_t = (P_t^1, \ldots, P_t^N)$ y p_t la serie de log-precios donde $p_t \sim I(1)$ y $r_t \sim I(0)$ el proceso de logretornos, entonces si b es su constante de cointegración el proceso cointegrado resultante $Y_t = b^T p_t \sim I(0)$. Definiendo

$$Z_t = \Delta Y_t = Y_t - Y_{t-1} = \sum_{i=1}^{N} b_i r_t^i \tag{1.3.1}$$

Entonces, Z_t es el retorno del portafolio cuyos pesos son las constantes de cointegración b.

Proposición 1.3.1: *Considerando la serie de logprecios p_t y el proceso $Y_t = b^T p_t$ con b los coeficientes de cointegración. Si $\lim_{p\to\infty} \mathbb{C}ov[Y_t, Y_{t-p}] = 0$. Entonces la serie de logprecios Y_t es coointegrada si y sólo si el proceso $Z_t = b^T r_t$ tiene las siguientes propiedades :*

$$\mathbb{E}[Z_t] = 0$$
$$\mathbb{V}[Z_t] = -2\sum_{p\geq 1} \mathbb{C}ov[Z_t, Z_{t-p}]$$
$$\sum_{p\geq 1} p\mathbb{C}ov[Z_t, Z_{t-p}] < \infty$$

de este modo se puede expresar la relación entre el proceso cointegrado Y_t y su diferenciación Z_t como:

$$Y_t = Y_t - Y_{t-1} + Y_{t-1} - Y_{t-2} + \cdots = \sum_{p\geq 0} Z_{t-p} = \sum_{p\geq 0} b^T r_{t-p} \tag{1.3.2}$$

con esta relación el resultado de la proposición 1.3.1 tiene diferentes interpretaciones. Para hacer estacionaria una suma ponderada de variables aleatorias estacionarias la autocovarianza requiere de una estructura particular, de este modo, se presenta un balance entre la incertidumbre de los retornos y la combinación que añade certidumbre. No se conoce donde terminaran los precios cointegrados pero sí que siguen una tendencia estocástica. La certidumbre que surge en la adición de las variables es suficiente para determinar que si la serie está divergiendo, entonces están ligadas para converger eventualmente.

Con esta característica, el proceso Z_t presenta una reversión a la media y la suma de sus autocovarianzas debe ser negativa por la proposición 1.3.1 pero también su varianza depende de la historia hasta el tiempo t.

Una estrategia inicial de inversión utilizando la característica de reversión a la media se explota comprando (vendiendo) el portafolio cuando Z_t está bajo (sobre) un valor crítico. Si bien se tiene la característica de reversión a la media no se tiene una noción sobre la velocidad a la cual se revierte. Para esto , las series pueden separase durante un periodo de tiempo pero eventualmente deben re-converger. El término agregado $\sum_{p\geq 1} Z_{t-p}$ puede cuantificar esta divergencia.

Proposición 1.3.2: *Considerando la estrategia de trading en la cual en cada periodo se compra* $-b_i \sum_{p\geq 1} Z_{t-p}$ *de los activos* $i, i = 1, 2, \ldots, N$ *y se vende en el periodo siguiente. Denotando* π_t *el beneficio descrito por la estrategia. Entonces:*

$$\pi_t = -\sum_{p\geq 1} Z_{t-p} Z_t \tag{1.3.3}$$

y

$$\mathbb{E}\left[\pi_t\right] = \frac{1}{2}\mathbb{V}\left[Z_t\right] > 0 \tag{1.3.4}$$

Esta proposición muestra que la estrategia tiene un retorno esperado estrictamente positivo, dado que la varianza es positiva. Para ejemplificar esta estrategia considerando 2 activos, entonces los logretornos r_t^1 y r_t^2:

$$r_t^1 = r_t^2 + c + (\gamma - 1)Y_{t-1} + \xi_t^1 \tag{1.3.5}$$

$$r_t^2 = \mu + \xi_t^2 \tag{1.3.6}$$

$$Y_t = c + \gamma Y_{t-1} + \xi_t^1 \tag{1.3.7}$$

donde $\mu \in \mathbb{R}$ y $(\xi_t^1, \xi_t^2) \sim N(0, \Sigma)$ con varianza σ_1^2 , σ_2^2 y covarianza ρ, γ el coeficiente del proceso autorregresivo de orden 1 Y_t y c la constante del proceso. El proceso $r_t^1 \sim$ AR(1) mientras que $r_t^2 \sim N(\mu, \sigma_2^2)$ entonces por proposición 1.3.2 la estrategia tiene retorno esperado positivo.

La estrategia propuesta por Galenko et al. (2012) [18] consiste en generar una estrategia dollar-neutral. En primer lugar identificando activos cointegrados y su vector de cointegración b, se divide en dos grupos de activos que se dirigen en una misma dirección:

$$i \in L \iff b_i \geq 0 \tag{1.3.8}$$

$$i \in S \iff b_i < 0 \tag{1.3.9}$$

entonces, dependiendo de los conjuntos L y S, se toman posiciones cortas y largas:

$$-\frac{b_i C \text{sign}\left(\sum_{p\geq} 1 Z_{t-p+1}\right)}{\sum_{j\in L} b_j},\ i \in L \tag{1.3.10}$$

$$\frac{b_i C \text{sign}\left(\sum_{p\geq 1} Z_{t-p+1}\right)}{\sum_{j\in L} b_j},\ i \in S \tag{1.3.11}$$

La construcción de este portafolio consiste en el hecho de que el término $\sum_{p\geq 1} Z_{t-p}$ mide cuanto ha divergido la serie de la tendencia estocástica y sign $\left(\sum_{p\geq 1} Z_{t-p+1}\right)$ provee la dirección de la posición del activo i. Un signo positivo es una posición larga mientras el negativo es corta, el denominador normaliza los pesos de los activos ta que la cantidad neta de la posición larga coincide con la corta. De este modo, la estrategia se estructura de la siguiente manera:

- **Paso 1:** Seleccionar un intervalo de tiempo de tamaño W de historia para estimar los coeficientes de cointegración b por $\tilde{b}$.
- **Paso 2:** Utilizar la estimación de los coeficientes $\tilde{b}$ para obtener las estimaciones de la realización histórica del proceso $\tilde{Z}_t$.
- **Paso 3:** Calcular la suma $\sum_{p=1}^{P} \tilde{Z}_{t-p+1}$ donde P es el parámetro de rezago máximo. (En teoría P es infinito pero en la práctica se usa una cantidad finita. Realizando Backtest se puede seleccionar un valor adecuado).
- **Paso 4:** Particionar los activos en términos de sus direcciones L y S.
- **Paso 5:** Tomar las posiciones adecuadas según la metodología expuesta.
- **Paso 6:** Revertir las operaciones en el periodo de tiempo siguiente. Actualizar la data histórica y volver al paso 1.

Finalmente, se debe utilizar una ventana de tiempo para estimar el modelo de cointegración y el nivel de rezagos a utilizar, además se debe determinar el tiempo sobre el cual la estrategia seguirá utilizando el modelo y cuando debe reestimarse. Utilizando data histórica es posible optimizar estos valores para luego determinar la ventana óptima a utilziar y el periodo óptimo de aplicación.

1.4. Otros métodos

Además de los métodos anteriormente presentados existen una gran cantidad de variantes de las mismas, por ejemplo, Yu y Renjie (2017) [46] genera una variante del modelo de coitnegración utilizan una estrategia neutral al mercado :

Definition 1.4.1: *Sea $\boldsymbol{x}_t$ una serie de tiempo p-dimensional cuyos componentes son integrados de orden 1, es decir, $\Delta \boldsymbol{x}_t = \boldsymbol{x}_t - \boldsymbol{x}_{t-1}$ es estacionario. Considerando un modelo VAR(k) de $\boldsymbol{x}_t$ el cual puede ser reescrito como:*

$$\Delta \boldsymbol{x}_t = \mu + \sum_{i=1}^{k-1} \Gamma_i \Delta \boldsymbol{x}_{t-i} + \Pi \boldsymbol{x}_{t-1} + \varepsilon_t \tag{1.4.1}$$

donde $\Gamma_1, \ldots, \Gamma_{k-1}$ y Π son matrices $p \times p$ y $\varepsilon_t \overset{iid}{\sim} N(0, \Lambda)$. Si rank($\Pi$) = p, entonces $\boldsymbol{x}_t$ es estacionario. Si rank(Π) = 0 entonces $\Pi = 0$ por lo que $\Delta \boldsymbol{x}_t \sim$ VAR(k-1) y no hay cointegración. Si $1 \leq rank(\Pi) = r \leq p-1$, entonces existen matrices $p \times r$ de rango r, A y B , tales que $\Pi = AB'$ y $b_1' \boldsymbol{x}_t, \ldots, b_r' \boldsymbol{x}_t$ son estacionarios, donde $B = (b_1 \ldots, b_r)$. Cuando Π cumple la condición anterior, el modelo se denomina modelo con corrección de error (ECM) y $b_1, \ldots, b_r$ son los vectores de cointegración.

Siguiendo a Johansen y Juselius (1992) [27], se puede descomponer la constante $\mu = Ab_0 + A_\perp \gamma$, donde b_0 son los interceptos en la relación de cointegración, $A_\perp$ es una matriz $p \times (p-r)$ ortogonal a las columnas de A y γ vector de pendientes.

Tomando $\boldsymbol{x}_t$ los retornos acumulados de p acti os, se pueden utilizar las r columnas de B para formar r portafolios de cointegración (COIN).

Para construir portafolios COIN neutrales al mercado, se deben agregar condiciones adicionales. Suponiendo que se busca un portafolio COIN neutral respecto a L índices ($L < p$), considerar una matriz $p \times L$ de coeficientes beta relativos a L índices como:

$$C = \begin{pmatrix} \mathbb{C}\text{ov}\left[R_1, L_1\right] / \mathbb{V}\left[L_1\right] & \mathbb{C}\text{ov}\left[R_1, L_2\right] / \mathbb{V}\left[L_2\right] & \dots & \mathbb{C}\text{ov}\left[R_1, L_L\right] / \mathbb{V}\left[L_L\right] \\ \mathbb{C}\text{ov}\left[R_2, L_1\right] / \mathbb{V}\left[L_1\right] & \mathbb{C}\text{ov}\left[R_2, L_2\right] / \mathbb{V}\left[L_2\right] & \dots & \mathbb{C}\text{ov}\left[R_2, L_L\right] / \mathbb{V}\left[L_L\right] \\ \vdots & \vdots & \dots & \vdots \\ \mathbb{C}\text{ov}\left[R_p, L_1\right] / \mathbb{V}\left[L_1\right] & \mathbb{C}\text{ov}\left[R_p, L_2\right] / \mathbb{V}\left[L_2\right] & \dots & \mathbb{C}\text{ov}\left[R_p, L_L\right] / \mathbb{V}\left[L_L\right] \end{pmatrix} \tag{1.4.2}$$

donde R_i son los retornos del $i-$ésimo activo y L_i los retornos del $i-$ésimo índice. Considerando H el complemento ortogonal a C, $H'C = 0$. Se tiene que

1. Todas las r columnas de B forman portafolios COINMAN ssi B puede expresarse como $B = H\varphi$ donde φ es una matriz $(p - L) \times r$ y,

2. Exactamente $r_1 (\leq r)$ columnas de B forman portafolios COINMAN ssi $B = \langle H\varphi, \psi \rangle$, donde $\varphi = (\varphi_1, \dots, \varphi_{r_1})$ y ψ son $(p - L) \times r_1$ y $p \times (r - r_1)$ matrices.

Tomando $b_i = H\varphi_i$, como vector de cointegración, se tiene que el portafolio formado por b_i tiene vetas nulos para todos los L índices:

$$(\beta_{i1}, \dots, \beta_{iL}) = \left(\frac{\mathbb{C}\text{ov}\left[b_i' R, L_1\right]}{\mathbb{V}\left[L_1\right]}, \dots, \frac{\mathbb{C}\text{ov}\left[b_i' R, L_L\right]}{\mathbb{V}\left[L_L\right]} \right) = \varphi_i' H' C = 0 \tag{1.4.3}$$

con $R = (R_1, \dots, R_p)'$.

Inferencia bajo las hipótesis $B = H\varphi$ y $B = \langle H\varphi, \psi \rangle$ han sido propuestas por Johansen (1991) [28] y Johansen y Juselius (1992) [27].

Si bien la estrategia sigue la misma base sobre un grupo de activos cointegrados, esta deja de ser dólar neutral y es neutral al mercado, siendo neutral a un set de índices de mercado a elección.

Otro método empleado por Clegg y Krauss (2018) [9] consiste en una variante de la cointegración, siendo esta cointegración parcial , otro método estadístico basado en cópulas por Xie et al. (2016)[65], un método utilizando un modelo de cointegración con mezcal logística autorregresiva Cheng et al. (2011) [7] entre muchos otros métodos.

Capítulo 2

INTRODUCCIÓN AL APRENDIZAJE REFORZADO PROFUNDO

El aprendizaje reforzado profundo busca estudiar como aprender a resolver problemas complejos, los cuales requieren de encontrar una solución a una secuencia de decisiones en estados de alta dimensión. Para hacer café se requiere tostar los granos, a una temperatura y tiempo adecuado, molerlos a cierta granulometría, calentar agua y realizar el filtrado o infusión dependiendo del método. No es posible utilizar granos enteros, calentarlos junto con el agua y posteriormente molerlos.

El aprendizaje reforzado profundo es la combinación entre el aprendizaje profundo y el aprendizaje reforzado. El objetivo es aprender las acciones óptimas que maximizan el premio para todos los estados que el ambiente puede tomar. El área a del deep learning trata sobre aproximar funciones en problemas en altas dimensiones, problemas complejos tales que los métodos usuales fallan en encontrar soluciones exactas. El aprendizaje profundo utilizar redes neuronales profundas para encontrar estas aproximaciones, en ambientes de dimensiones grandes, tales como reconocimiento de voz o imágenes. Mientras que el campo del aprendizaje reforzado trata sobre aprender desde una retroalimentación, por ensayo y error. En este aspecto, no se requiere un set de datos de prueba para entrenar, escoge las acciones y aprende desde la retroalimentación que el ambiente provee. Entonces el agente, mediante un proceso de ensayo y error, comete tanto errores como aciertos los cuales son el material por el cual aprende a interactuar con el ambiente.

Deep Learning

Los Algoritmos clásicos de aprendizaje de máquinas aprenden un modelo predictivo sobre los datos, utilizando métodos como regresiones lineales, árboles de decisiones, máquinas de soporte vectorial, redes neuronales entre otros. Los modelos buscan generalizar las relaciones de los datos para realizar predicciones. Esto se traduce a aproximar una función de los datos, se propone un modelo que relaciona las variables y se busca estimar los parámetros que mejoran la predicción o representan mejor las relaciones.

En la actualidad, con la alta capacidad de computacional, se ha podido explotar modelos más complejos y las redes neuronales profundas han tomado una gran importancia en las tareas de aprendizaje de máquinas a un nuevo nivel y han podido llevar sus capacidades a aplicaciones en altas dimensiones extremadamente complejas como es el reconocimiento de imágenes entre otros.

De este modo, el aprendizaje profundo ha permitido resolver problemas en altas dimensiones en tiempo real, permitido que las máquinas de aprendizaje sean aplicadas a tareas diarias como sería el reconocimiento de voz en smartphones por ejemplo.

Reinforcement Learning

El campo del aprendizaje reforzado consiste en que un agente (máquina) aprende por interacción con el ambiente. En el aprendizaje supervisado se requiere de datos preexistentes catalogados para poder estimar una función de los datos; el aprendizaje reforzado solo requiere de un ambiente que provee una retroalimentación por las acciones que realiza el agente. Este requerimiento es fácil de tener, pues no se necesita datos de ejemplo.

Los agentes generan, mediante sus acciones, sus propios datos *on-the-fly*, de manera online, a través del premio que obtiene desde el ambiente. Estos agentes pueden escoger que acción tomar para aprender por el refuerzo del ambiente a través de un premio. En este sentido, el agente aprende a través de la experiencia, mediante ensayo y error. Para esto se requiere construir una política de acciones que deben ser tomadas de acuerdo al estado de la naturaleza en el cual se encuentra.

Deep Reinforcement Learning

El aprendizaje reforzado profundo combina los métodos de resolución de problemas altodimensionales y complejos con el aprendizaje reforzado, permitiendo un aprendizaje interactivo en altas dimensiones. Una de las principales razones de interés en este campo es que funciona muy bien bajo la capacidad computacional actual y en diferentes aplicaciones. Por ejemplo aprender a jugar videojuegos. En particular, es utilizado para problemas de decisiones secuenciales.

A diferencia entre el aprendizaje supervisado y no supervisado, el aprendizaje reforzado conforma un tercer paradigma en el aprendizaje de máquinas. En primer lugar esta forma de aprendizaje aprende por interacción, en contraste con el supervisado y no supervisado,

el set de datos se produce de manera dinámica. Este marco de aprendizaje busca aprender una política del ambiente interactuando con él. En este sentido, se reconoce un *agente* que realiza la acciones y aprende la política mientras que el *ambiente* provee la retroalimentación necesaria a las acciones del agente. Este agente se entiende como el 'humano' o robot, y el ambiente como el mundo. El objetivo final es encontrar las acciones para cada estado que maximicen el retorno acumulado esperado de largo plazo, no en la acción que maximice el retorno al tiempo inmediato sino en un horizonte. La función óptima que transforma los estados en acciones se denomina la *política óptima*. Es de esta manera, que bajo este paradigma no se tiene un supervisor y no hay un set de datos estáticos, en cambio, está el ambiente que determina que tan bueno es el estado en el cual el agente se encuentra.

En segundo lugar, una diferencia fundamental respecto al ML clásico es el premio que recibe, o *reward value*. La calidad de la acción que llevó al estado actual se cuantifica, pero de manera parcial, en un marco de aprendizaje supervisado se tiene información total, una etiqueta o valor que determina la respuesta correcta. Esto deja al aprendizaje reforzado (RL) entremedio del supervisado y no supervisado. Finalmente, una diferencia fundamental es la aplicación en al resolución de problemas de decisiones secuenciales. El aprendizaje supervisado y no supervisado aprenden relaciones a un paso, mientras que el RL aprende una política, que es la solución a un problema de múltiples pasos.

En prácticamente la mayoría de los usos de ML en finanzas, específicamente, en el trading de activos financieros, esta dedicado a técnicas supervisadas. En la gran mayoría de estas aplicaciones se separan en dos componentes separadas: Un modelo predictivo, e.g. redes neuronales , utilizando diferentes variables explicativas o presuntamente explicativas , y en segundo lugar un módulo de trading el cual se alimenta de las proyecciones realizadas para emitir señales de trading, e.g. si sobrepasa un umbral se realiza cierta acción.

A pesar de la gran popularidad de este tipo de acercamiento en dos fases, se tienen varias limitaciones que podrían llevar a un rendimiento sub óptimo. En primer lugar, el objetivo del modelo de predicción, e.g. minimizar el error de predicción, no necesariamente coincide con la meta final de un inversionista, e.g. maximizar retorno ajustado por riesgo. En segundo lugar, usualmente se utiliza solo la predicción como input al módulo de trading, entonces se tiene información adicional que es desechada. En tercer lugar, restricciones impuestas por el ambiente e.g. costos de transacción o falta de liquidez solamente son incorporadas en la optimización del módulo de trading.

En el marco del aprendizaje reforzado (RL) se busca combinar la tarea de predicción junto con la construcción de portafolios en un formato integrado considerando al mismo tiempo diferentes restricciones como serían los costos de transacción entre otros en alineación con los objetivos de inversionista. De ahora en adelante, el *agente de trading* busca aprender mediante la interacción con el ambiente, permitiéndose incorporar restricciones en su proceso de toma de decisiones. Los agentes de RL se pueden clasificar en 3 categorías:

- **Critic-only**: Estos tipos de agentes son los más frecuentes en cuanto a aplicaciones de RL en mercados financieros. Como idea general estos agentes buscan aprender la función valor de la cual buscan comparar , criticar, el resultado esperado de diferentes acciones, la cuales podrían ser tomar una posición larga, o bajar la posición por ejemplo. Durante este proceso de decisión se busca realizar la acción que otorgue el mejor resultado en términos de la función valor considerando el estado actual del ambiente.

- **Actor-only**: Este tipo de agentes, a diferencia de los de crítica, dadas las condiciones actuales, estados de ambiente actual, realizan una acción sin calcular y comparar el resultado esperado de diferentes acciones. Por lo tanto, el agente aprende directamente una política, una forma de actuar, para ejecutar dado el estado actual. Las ventajas que se encuentran en este marco de acción consisten esencialmente en que se puede realizar en un espacio de acción continuo y típicamente tienen una convergencia más rápida de aprendizaje.

- **Actor-critic**: En este caso, el agente busca combinar las ventajas de cada uno de los anteriores. La idea principal es utilizar simultáneamente un actor, el cual determina la acción del agente dado el estado actual, y la crítica, que juzga la acción seleccionada. Entonces, busca aprender una política la cual es considerada la mejor según la crítica. A pesar de sus potenciales ventajas, este método de RL es el menos utilizado en mercados financieros.

2.1. Procesos de decisión de Markov

El aprendizaje reforzado surge como heurística para resolver problemas complejos de procesos de decisiones de Markov en los cuales se modelo un ambiente estocástico y un marco de acción sobre el mismo.

2.1.1. MDP Discreto

Un proceso de decisión de Markov discreto es una manera de definir tareas de toma de decisiones secuenciales con ciertas propiedades. El término discreto refiere a que los estados y acciones son atómicos, es decir discretos. Los MDPs incluyen ambientes estocásticos y pueden manejar múltiples objetivos a través del concepto de utilidad. Las formulaciones de MDP incluyen todos los problemas de camino más corto por ejemplo.

Definition 2.1.1: *Proceso de Decisión de Markov Discreto (DMDP)*
Un DMDP es una 5-tupla $(\mathcal{S}, \mathcal{A}, T, R, \gamma)$ *donde:*

1. ***Espacio de Estados:*** $\mathcal{S}$

   El conjunto de estados corresponde a los estados posibles de la naturaleza. En el caso discreto este espacio es finito.

2. ***Espacio de Acciones:*** $\mathcal{A}$

   Es el espacio de acciones posibles a realizar, en el caso discreto de tamaño finito.

3. ***Dinámica de Transición:*** $T(s'|s, a)$

   La dinámica de transición determina como la naturaleza reacciona a las acciones realizadas en conjunto con el estado actual. Corresponde a una distribución de probabilidad condicional

   $$T : \mathcal{S} \times \mathcal{A} \mapsto p(\mathcal{S})$$

   representado como un tablero de tamaño $|\mathcal{S}| \times |\mathcal{A}| \times |\mathcal{S}|$ *donde cada entrada corresponden a una acción y un estado mientras la salida corresponde a una probabilidad.*

4. ***Función de Recompensa:*** $R(s, a, s')$

   Esta función cuantifica la calidad de cada transición, la cual mapea el tablero de transiciones a un valor real:

   $$R : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$$

   el cual tiene la interpretación de otorgar una recompensa $R(s, a, s')$ *al tomar la acción* a *en el estado* s *y terminar en el estado* s'*. También puede interpretarse como función de costo en el caso de problemas de camino más corto por ejemplo, siendo una recompensa negativa.*

5. ***Factor de Descuento:*** γ

   Corresponde al término de ponderación o peso de las recompensas futuras, en donde $\gamma \in [0, 1]$ *determina que tanto se pesan las recompensas en el futuro respecto a la recompensa inmediata.*

Además, se considera la distribución del estado inicial $p_0(s)$ *sobre los estados* $\mathcal{S}$*.*

Notación: *En cada intervalo de tiempo se observa el estado* s_t *y se toma la acción* a_t *donde posterior a la acción se observa* $s_t \sim T(\cdot|s, a)$*.*

Se denomina al retorno $r_t = R(s_t, a_t, s_{t+1})$*, donde para cada transición se escribe también la 4-tupla* (s, a, r, s') *donde el estado futuro o siguiente es* s'*.*

La **Política** π gobierna las acciones a realizar dado el estado de la naturaleza. Es una función $\pi : \mathcal{S} \mapsto p(\mathcal{A})$ que mapea el espacio de estados a una distribución de probabilidad sobre el espacio de acciones. En el caso discreto se tiene el conjunto de pares estado-acción de tamaño $|\mathcal{S}| \times |\mathcal{A}|$. Además, la política es estacionaria, es decir, es invariante en el tiempo, esto es $\pi(a|s_t) \overset{d}{=} \pi(a|s_{t+h})$ si $s_t = s_{t+h} \forall h$. Para el caso de una política determinista, se denota $\pi(s)$, donde solo se selecciona una única acción para cada estado.

2.1.2. Retorno acumulado y valor

Para un MDP se busca resolver un problema de toma de decisiones secuencial cumpliendo con un criterio de optimalidad. Para definir un MDP se requiere de una función de recompensa, donde la secuencia de decisiones a tomar debe obtener el mayor retorno posible. La suma de todas las recompensas obtenidas de denomina recompensa acumulada o retorno acumulado (retorno sin pérdida de generalidad). Dado que los estados de la naturaleza son estocásticos, la política también puede ser estocástica, se busca obtener un retorno en *promedio*. La esperanza de retorno se denomina *valor* (también *utilidad*). A continuación se definen estos términos formalmente:

Definition 2.1.2: ***Retorno (MDP):*** *En un MDP, la secuencia de acciones define una secuencia de estados y retornos por lo que determina una* ***trayectoria*** *de estados, acciones, retornos y estados siguientes. Para una trayectoria comenzando en tiempo* t *de tamaño* n *como:*

$$h_t = \{s_t, a_t, r_t, s_{t+1}, a_{t+1}, r_{t+1}, \ldots, a_{t+n}, r_{t+n}, s_{t+n+1}\}$$

Luego, dada una trayectoria h_t *y un factor de descuento* $\gamma \in [0, 1)$ *se define el retorno* G_t *de la trayectoria:*

$$G_t = \sum_{h=0}^{n} \gamma^h r_{t+h} \tag{2.1.1}$$

Como el retorno de una trayectoria corresponde a una realización y la naturaleza junto con la política son estocásticos, para una misma política y estado inicial la trayectoria viene siendo

la realización de un proceso estocástico, se busca cuantificar el retorno esperado denominado valor. Para ello se define el valor de estado, y valor de estado-acción:

Definition 2.1.3: ***Valor de Estado:*** *Se define el valor de estado $V(s)$ como el retorno esperado cuando el agente comienza en el estado s y actúa bajo la política π:*

$$V^{\pi}(s) = \mathbb{E}_{\pi,T}\left[\sum_{h=0}^{\infty} \gamma^h r_{t+h} | s_t = s\right] \tag{2.1.2}$$

Para el par (s,a) se define el valor $Q^{\pi}(s,a)$, a diferencia de la función valor $V^{\pi}(s)$, esta depende de la acción a:

$$Q^{\pi}(s,a) = \mathbb{E}_{\pi,T}\left[\sum_{h=0}^{\infty} \gamma^h r_{t+h} | s_t = s, a_t = a\right] \tag{2.1.3}$$

2.1.3. Función Valor óptima y política óptima

Dado un MDP se está interesado en encontrar la mejor política, donde la mejor política está definida en términos del valor. Entonces se busca encontrar la política que maximiza el valor.

Para determinar que una política es superior a otra, se requiere determinar cuando una función valor es superior a otra:

Definition 2.1.4: ***Orden parcial:*** *Para dos políticas π y π' asociadas a sus funciones valor $V^{\pi}(s)$ y $V^{\pi'}(s)$ se define el orden:*

$$\pi' \geq \pi \iff V^{\pi'}(s) \geq V^{\pi}(s) \quad \forall s \in \mathcal{S} \tag{2.1.4}$$

es decir, la política π' es mayor a la política π si y sólo si su retorno esperado es al menos igual o mayor para todos los estados de la naturaleza.

Para cada política en el MDP se asocia una función valor, pero se tiene que para todas las funciones valor posibles en el MDP se tiene una óptima.

Teorema 2.1.5: ***Valor óptimo:*** *Para un MDP dado, existe una función valor óptima $V^*(s)$:*

$$V^*(s) = \max_{\pi} V^{\pi}(s) \tag{2.1.5}$$

donde

$$V^*(s) \geq V^{\pi}(s) \quad \forall \pi \in \Pi, s \in \mathcal{S}$$

donde la política que alcanza el valor óptima corresponde a la política óptima π^* :

$$\pi^*(s) = \arg\max_{\pi} V^{\pi}(s) = \arg\max_{\pi} Q^{\pi}(s,a) \tag{2.1.6}$$

pueden existir múltiples políticas que alcanzan el valor óptimo, sin pérdida de generalidad denotadas π^*.

En un MDP la política óptima es siempre *greedy*, es decir, para cada estado hay una acción que es la mejor (o múltiples que entregan el mismo valor).

De esta manera, el objetivo de la optimización en un MDP es encotrar la política óptima $\pi^*(s)$. Se puede también buscar la mejor función valor $V^*(s)$ o $Q^*(s,a)$ dado que los valores se relacionan directamente con la política. Para esto se debe introducir primero la ecuación de Bellman y el principio de programación dinámica.

2.1.4. Ecuación de Bellman

Una de las características de la función valor es que esta puede ser reescrita de forma recursiva.

Definition 2.1.6: ***Ecuación de Bellman para valor de estados:*** *La ecuación de Bellman para los valores de estado* $V(s)$ *está dada por:*

$$V(s) = \mathbb{E}_{a\sim\pi(\cdot|s)}\mathbb{E}_{s'\sim T(\cdot|a,s)}\left[r + \gamma V(s')\right] \tag{2.1.7}$$

$$= \sum_{a\in\mathcal{A}} \pi(a|s)\left[\sum_{s'\in\mathcal{S}} T(s'|s,a)[r + \gamma V(s')]\right]$$

la ecuación también se puede escribir para la función Q*:*

$$Q(s,a) = \mathbb{E}_{s'\sim T(\cdot|a,s)}\left[r + \gamma\mathbb{E}_{a'\sim\pi(|s')}[Q(s',a')]\right] \tag{2.1.8}$$

$$= \sum_{s'\in\mathcal{S}} T(s'|s,a)\left[r + \gamma\sum_{a'\in\mathcal{A}}[\pi(a|s)Q(s',a')]\right]$$

En esta formulación recursiva se toma esperanza sobre todas las posibles acciones y posteriormente las transiciones o el caso inverso para la función Q, entonces no se requiere tomar esperanza sobre y todas las trayectorias posibles.

Las funciones valor V y Q están estrechamente relacionadas, donde para una política la función valor de estado y estado-acción son maneras diferentes de representar la función valor, donde para la función Q de estado-acción se puede obtener el valor del estado para una política:

$$V(s) = \mathbb{E}_{a\sim\pi}[Q(s,a)] = \sum_{a\in\mathcal{A}} \pi(a|s)Q(s,a) \tag{2.1.9}$$

mientras que para la función valor de estado V se puede obtener la función Q como:

$$Q(s,a) = \mathbb{E}_{s'\sim T}[r + \gamma V(s')] = \sum_{s'\in\mathcal{S}} T(s'|s,a)[r + \gamma V(s')] \tag{2.1.10}$$

luego, para estas relaciones, sustituyendo cada una en la otra se obtiene la ecuación de Bellman respectiva.

2.1.5. Programación Dinámica

El método de solución de un MDP sobre el cual se construyen la mayoría de los algoritmos de búsqueda consiste en el principio de programación dinámica (DP). Este principio consiste reducir un problema grande en pequeños subproblemas más fáciles de resolver. En particular, se revisará el principio respecto a los MDP, pues es más general.

- **Fortalezas:**
  - El DP es un enfoque clásico para resolver un MDP.
  - No requiere heurística
  - Garantiza la obtención de una solución óptima
- **Desventajas:**
  - Requiere el almacenamiento de todos los estados
  - Barrer todos los estados es ineficientes

El DP es la base de la mayoría de los enfoques para resolver el MDP,como es el RL. Para esto se realizan dos procedimientos

- **Evaluación de política:** $(\pi \rightarrow V/Q)$: para una política π dada, como se calcula $V^{\pi}(s)$ y $Q^{\pi}(s,a)$.
- **Mejora de política:** $(V/Q \rightarrow \pi)$: dado los valores $V^{\pi}(s)$ y $Q^{\pi}(s,a)$ como obtener una mejor política π.

El objetivo de la evaluación de la política es encontrar el valor de $V^{\pi}(s)$ para una política π. El enfoque es el siguiente:

1. Inicializar una tabla de valor para $V(s)$ o $Q(s,a)$. Todas las entradas pueden ser aleatorias salvo los estados terminales, donde se tiene que para los estados terminales $V = Q = 0$.
2. Iterar por todos los estados s (o par estado-acción (s,a)), donde en cada iteración de evalúa la ecuación de Bellman para actualizar el estado específico.
3. Repetir el procedimiento hasta que todos los valores converjan. Los valores de convergencia son las estimaciones de $V^{\pi}(s)$ y $Q^{\pi}(s,a)$, el valor de la política π.

este algoritmo en base a una política particular y un umbral de convergencia calcula el valor de la política. Se barren todos los estados posibles y para cada estado se calcula la ecuación de Bellman.

Luego, para un MDP se tiene que la política óptima es siempre globalmente codiciosa, es decir, siempre hay una (o múltiples) acción que es la mejor, donde se busca obtener una trayectoria óptima. De este modo, la nueva política siempre es *greedy* respecto a la estimación de valor actual, entonces se deja toda la masa de probabilidad en la acción que actualmente tiene la mayor estimación de valor en su estado. La acción se puede identificar mediante:

$$\pi(s) \leftarrow \arg\max_a \mathbb{E}_{s' \sim T(\cdot|a,s)} \left[r + \gamma V(s') \right] \tag{2.1.11}$$

o para el caso del par estado-acción:

$$\pi(s) \leftarrow \arg\max_a Q(s, a) \tag{2.1.12}$$

Cuando se almacenan las estimaciones de Q, la mejora de política es sencilla, se observan los valores en la tabla de valores. Cuando se almacenan solo los valores de V, la mejora de política requiere de la evaluación de la ecuación de Bellman una vez más.

Las actualizaciones en la iteración pueden cambiar la política en ciertos estados, llevando a una nueva política π'. Luego, esto se realiza de manera iterativa.

Entonces, se tienen dos esquemas para aplicar DP.

Policy Iteration

- Se repite la evaluación de política y la mejora de política hasta converger.
- Primero se realiza la evaluación de política.
- Luego, se realiza la mejora de política, donde se busca la mejor decisión en cada estado.
- Cuando la política deje de cambiar, se convergió a la política óptima.

La otra alternativa es realizar una iteración de valor (VI). La política puede ser almacenada de manera implícita de las acciones, en este caso solo se requiere almacenar el valor y la política se deriva directamente del valor. Por ejemplo, para una política *greedy*:

$$\pi(s) = \arg\max_a \mathbb{E}_{s' \sim T(\cdot|a,s)} \left[r + \gamma V(s') \right] \tag{2.1.13}$$

o

$$\pi(s) = \arg\max_a Q(s, a|\pi) \tag{2.1.14}$$

Value Iteration

En este caso solo se requiere un barrido de evaluación de política antes de realizar una mejor. Es más, la política se representa en forma implícita desde la tabla de valor. De este modo, se formula la evaluación de política y mejora en una sola ecuación. Para ello se implementa la ecuación de Bellman, pero se maximiza sobre las acciones, esto se conoce como el operador de optimalidad de Bellman, y la ecuación subyacente de optimalidad de Bellman:

- V-iteration V(s):

$$V(s) \leftarrow \max_a \mathbb{E}_{s' \sim T(\cdot|a,s)} \left[r + \gamma V(s') \right] \tag{2.1.15}$$

$$V(s) \leftarrow \max_a \sum_{s' \in \mathcal{S}} T(s'|s,a) \left[r + \gamma V(s') \right] \tag{2.1.16}$$

- Q-iteration Q(s,a):

$$Q(s,a) \leftarrow \mathbb{E}_{s' \sim T} \left[r_t + \gamma \max_a Q(s', a') \right] \tag{2.1.17}$$

$$Q(s,a) \leftarrow \sum_{s' \in \mathcal{S}} T(s'|s,a) \left[r_t + \gamma \max_a Q(s', a') \right] \tag{2.1.18}$$

de esta manera, las iteraciones por valor o Q-valor son sumamente sencillas y conllevan una modificación simple respecto a la evaluación de la política.

2.2. Q-Learning

El Q-learning se basa en ver la tabla de Q-Value $Q(s,a)$. Para encontrar la función Q óptima se utuliza la ecuación de Bellman (Bellman y Dreyfus 1962 [3]) la cual tiene una solución única $Q^*(s,a)$:

$$Q^*(s,a) = (\mathcal{B}Q^*)(s,a) \tag{2.2.1}$$

donde $\mathcal{B}$ es el operador de Bellman que mapea una función $K : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ a otra función $\mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ definida como:

$$(\mathcal{B}K)(s,a) = \mathbb{E}_{s' \sim T(\cdot|s,a)} \left[R(s,a,s') + \gamma \max_{a' \in \mathcal{A}} K(s', a') \right] \tag{2.2.2}$$

Este operador $\mathcal{B}$ es contractivo, y por teorema de contracción de Banach existe un punto fijo.

Una prueba de convergencia a la función valor óptima (Warkins y Dayan 1992 [62]) se tiene bajo las condiciones de que:

- El par estado-acción sean discretos
- Todas las acciones son repetidamente muestreadas en todos los estados, para asegurar exploración suficiente sin necesidad de un modelo de transición.

pues, para el caso discreto se puede para cualquier par de funciones acotadas $K, K' : \mathcal{S}\times\mathcal{A} \to \mathbb{R}$ se tiene la condición contractiva:

$$\left\|TK - TK'\right\|_\infty \leq \gamma \left\|K - K'\right\|_\infty$$

En un marco de alta dimensionalidad (caso continuo) para los estados y acciones, es necesario una parametrización de la función $Q(s, a; \theta)$, donde θ refiere a los parámetros que determinan el Q-Value.

2.2.1. Fitted Q-Learning

Dado un set de datos D, se obtienen diferentes experiencias generadas como tupla (s, a, r, s') donde el siguiente estado s' tiene una distribución condicional $s' \sim T(s, a)$ y la recompensa r de $R(s, a, s')$. En este marco de desarrollo,(Gordon 1996 [21]), el algoritmo comienza en un estado inicial aleatorio de los valores de $Q(s, a; \theta_0)$ donde θ_0 son los parámetros iniciales. Luego, la aproximación en una iteración k $Q(s, a; \theta_k)$ se actualiza mediante el valor objetivo

$$Y_k^Q = r + \gamma \max_{a'\in\mathcal{A}} Q(s', a'; \theta_k) \tag{2.2.3}$$

Riedmiller (2005) [50] planteó utilizar una red neuronal para parametrizar Q (NFQ), en la cual el estado se puede utilizar como input a la Q-red y un output diferente se tiene para cada acción posible. De esta manera se tiene una estructura eficiente que tiene la ventaja de obtener el cálculo de $\max_{a'} Q(s', a'; \theta_k)$ en un solo paso dado un s'.

AL parametrizar Q como una red neuronal, se utilizó una pérdida cuadrático con un algoritmo de gradiente descendiente :

$$L_{DQN} = \left(Q(s, a; \theta_k) - Y_k^Q\right)^2 \tag{2.2.4}$$

Por lo tanto, se tiene la actualización de parámetros:

$$\theta_{k+1} = \theta_k + \alpha \left(Y_k^Q - Q(s, a; \theta_k)\right) \nabla_{\theta_k} Q(s, a; \theta_k) \tag{2.2.5}$$

donde α corresponde a la tasa de aprendizaje.

La utilización de la pérdida cuadrática asegura que la estimación sea asintóticamente insesgada: $\mathbb{E}\left[\left(Q(s,a;\theta_k) - Y_k^Q\right)^2\right] \to 0$

2.2.2. DQN

Desde la idea de NFQ, se tiene el algoritmo de Q-red profundo (deep Q-network o DQN) introducido por Mnih et al. 2015 [40] el cual es capaz de obtener muy buen rendimiento en un marco online para una variedad de juegos de ATARI, aprendiendo directamente de los píxeles.

Para esto se utilizan dos heurísticas para limitar inestabilidades de la red:

- El objetivo de la Q-red se cambia respeto a NFQ 2.2.3 por $Q(s',a';\theta_k^-)$ donde los parámetros θ_k^- se actualizan cada C iteraciones con la siguiente asignación: $\theta_k^- = \theta_k$. Esto previene que inestabilidades se propaguen rápidamente y reduce el riesgo de divergencia a medida que las respuestas Y_k^Q se mantengan fijas para C iteraciones.

- En un marco online, la memoria de replay (Lin 1992 [37]) mantiene toda la información por N_{replay} intervalos de tiempo, donde la experiencia es recolectada por una política $\epsilon-$greedy. Entonces las actualizaciones se realizan en un set de tuplas (s,a,r,s') llamados *mini-batches* seleccionados aleatoriamente dentro de la memoria de replay. Esta técnica permite actualizaciones que cubren una gran cantidad del espacio de estado-acción. Adicionalmente una actualización de mini-batch tiene menos varianza respecto a una única tupla. En consecuencia, provee la posibilidad de realizar actualizaciones de parámetros mas grandes, manteniendo una paralelización eficiente del algoritmo.

Adicional a la Q-red objetivo y la memoria de replay, DQN utiliza otras dos heurísticas importantes. Para mantener los valores en una escala razonable y asegurar una aprendizaje apropiado en la práctica, las recompensas se truncan ente -1 y +1. Este truncado limita la escala de error y hace más fácil utilizar la misma tasa de aprendizaje durante múltiples juegos (aún así, introduce un sesgo). Además, se utilizan técnicas generales de preprocesamiento para los inputs, se utiliza como primera capa de la red una capa convolucional y la optimización se realiza con un algoritmo de gradiente descendiente estocástico RMSprop (Tieleman 2012 [57]).

2.2.3. Double DQN

En el DQN se utilizan los mismos valores para seleccionar y evaluar una acción. Esto provoca una tendencia a seleccionar valores sobre estimados en caso de ruido o poca precisión, generando estimaciones de valor muy optimistas y un sesgo en esta dirección. El método de estimación doble utiliza dos estimaciones para separar cada variable desacoplando el estimador de valor de la selección y el valor (Hasselt 2010 [24]). Entonces, independiente si

los errores en las estimaciones de Q-valores son debido a incertidumbre, aproximación de función, no estacionariedad u otras fuentes, este método busca eliminar el sesgo positivo del DQN en la estimación de los valores de la acción. En el DDQN (Van Hasselt et al. 2016 [58]) el valor objetivo Y_k^Q se cambia por:

$$Y_k^{DDQN} = r + \gamma Q(s', \arg\max_{a \in \mathcal{A}} Q(s', a; \theta_k); \theta_k^-) \tag{2.2.6}$$

lo que reduce la sobre estimación de los Q-valores aprendidas así como también mejor la estabilidad de el algoritmo, produciendo un mejor rendimiento. Comparando con el DQN los parámetros de la red objetivo θ_k^- son utilizados para la evaluación de la acción greedy actual. Notar que la política de igual manera se sigue escogiendo de acuerdo a los valores obtenidos por los pesos actuales θ.

2.2.4. Dueling network architecture

Wang et al. 2015 [60] utiliza una arquitectura de red desacoplada en el valor y la función de ventaja $A^\pi(s,a) = Q^\pi(s,a) - V^\pi(s)$, obteniendo una mejora en el rendimiento. La función de Q-valor está dada por:

$$Q(s,a;\theta^{(1)},\theta^{(2)},\theta^{(3)}) = V(s;\theta^{(1)},\theta^{(3)}) + \left(A(s,a;\theta^{(1)},\theta^{(2)}) - \max_{a' \in \mathcal{A}} A(s,a';\theta^{(1)},\theta^{(2)}) \right) \tag{2.2.7}$$

luego, para $a^* \in \arg\max_{a' \in \mathcal{A}} Q(s,a';\theta^{(1)},\theta^{(2)},\theta^{(3)})$ se obtiene $Q(s,a^*;\theta^{(1)},\theta^{(2)},\theta^{(3)}) = V(s;\theta^{(1)},\theta^{(3)})$.

De este modo, $V(s;\theta^{(1)},\theta^{(3)})$ provee una estimación de la función valor, mientras que la otra componente una estimación de la función de ventaja. La actualización es realizada como en el DQN y solo se cambia la estructura de la red. Este acercamiento pierde en cierto sentido la noción de la función valor V y A, pero incrementa la estabilidad en la optimización de parámetros de la red.

2.3. Policy-learning

A diferencia de los algoritmos basados en la función Q, los métodos de gradiente de política tienen como objetivo maximizar el retorno acumulado encontrando una política que lo haga. Estos métodos utilizan una señal de aprendizaje derivada desde el muestreo de parámetros de instancias de políticas y el conjunto de políticas se desarrolla hacia políticas que generen mejores retornos (e.g. Salimans et al. 2017 [51]).

2.3.1. Deterministic policy Gradient DPO

El algoritmo de Deep Deterministic Policy Gradient DDPG (Silver et al. 2014 [54], Lillicrap et al. 2015 [36]) introducen una representación directa de la política en un formato que puede extender algoritmos como el DQN o NFQ para extenderlo a un espacio de acciones continuas.

Considerando la política determinista $\pi(s) : \mathcal{S} \rightarrow \mathcal{A}$. En un espacio de acciones discreto, un acercamiento directo es construir un esquema iterativo del tipo:

$$\pi_{k+1}(s) = \arg\max_{a \in \mathcal{A}} Q^{\pi_k}(s, a) \tag{2.3.1}$$

donde π_k es la política de la k-ésima iteración, denotando $\pi_w(s)$ una política determinista diferenciable. En este caso una alternativa para este problema resulta ser un esquema de grandiente descendiente en la dirección del gradiente de Q, lo que da origen al algoritmo DDPG:

$$\nabla_w V^{\pi_w}(s_0) = \mathbb{E}_{s \sim \rho^{\pi_w}} \left[\nabla_w(\pi_w) \nabla(Q^{\pi_w}(s, a)) | a = \pi_w(s) \right] \tag{2.3.2}$$

Lo cual implica depender de $\nabla(Q^{\pi_w}(s, a))$ junto con $\nabla_w(\pi_w)$ lo cual requiere de métodos de actor-crítica.

2.3.2. Métodos de Actor-Critic

Se tiene que las políticas representadas por una red neuronal pueden ser actualizadas por métodos de gradiente para los casos de políticas determinista o estocásticas. Para este efecto, se requiere una estimación de la función valor de la política actual, un acercamiento es utilizar una arquitectura de actor-crítica que consiste en dos componentes; un actor y una crítica. (Konda y Tsitsiklis 2000 [30]) El actor refiere a la política o determinación de la acción mientras que la crítica refiere a la estimación de la función valor. En DRL ambos pueden ser representados a través de una red neuronal (Minh et al. 2016[39]), donde el actor utiliza gradientes de la política para aproixmar los parámetros w mientras que la crítica se parametrizada por θ aproxima la función valor de la política π_w:

$$Q(s, a; \theta) \approx Q^{\pi_w}(s, a)$$

Crítica

Considerando las tuplas (s, a, r, s') , el acercamiento *off-policy* mas sencillo para estimar la crítica es realizar un algoritmo de bootstrapping en cada iteración, donde el valor actual

$Q(s, a; \theta)$ se actualiza hacia el objetivo:

$$Y_k^Q = r + \gamma Q(s', a = \pi(s'); \theta) \tag{2.3.3}$$

Si bien es simple, resulta ser poco eficiente junto con ser susceptible a inestabilidades y una propagación de recompensa lenta hacia atrás. (Sutton, 1996 [56]). Un esquema ideal para este efecto se basa en la eficiencia del muestreo y en la eficiencia computacional, realizándolo de forma tal que pueda utilizar las trayectorias de *off* y *on-policy*, e.g. utilizando memoria sobre los eventos anteriores y debe ser estable con una rápida propagación de la recompensa de los métodos *on-policy* para muestras obtenidas desde el comportamiento en la política. Se tienen bastantes métodos que combinan data de *on-* y *off-policy* para la evaluación de la política (Precup, 2000 [47], Munos et al. 2016 [44], Wang et al. 2016b [61] y Gruslys et al. 2017 [22]).

Actor

Desde la ecaución 2.3.2, el gradiente de la política *off-policy* en la fase de mejora para el caso estocástico se obtiene como:

$$\nabla_w V^{\pi_w}(s_0) = \mathbb{E}_{s\sim\rho^{\pi_w}, a\sim\pi_\beta}\left[\nabla_w(\pi_w)\nabla(Q^{\pi_w}(s, a))|a = \pi_w(s)\right] \tag{2.3.4}$$

donde β es un política de comportamiento generalmente diferente de π, que produce un gradiente generalmente sesgado. Este acercamiento se comporta de forma adecuada en la practica pero el uso de un estimador sesgado del gradiente de política dificulta el estudio de la convergencia sin la hipótesis GLIE (Munos et al. 2016 [44], Gruslys et al. 2017 [22]) (Greedy in the Limit with Infinite Exploration, donde las políticas debe ser *greedy* en el límite de una configuración de aprendizaje online donde el agente ha acumulado una cantidad infinita de experiencia. Singh et al., 2000[55]).

Los métodos de actor-crítica, se han investigado algoritmos asincrónicos para realizar la estimación del gradiente de política *on-policy* sin necesidad de realizar un *replay* de escenarios anteriores, donde múltiples agentes se ejecutan en paralelo y los actores en entrenamiento son entrenados de manera asincrónica (Mnih et al. 2016 [39]). La paralelización de los agentes asegura que cada agente experimente eventos diferentes a un paso de tiempo dado. Por lo tanto, los retornos de n pasos pueden ser utilizados sin la necesidad de introducir sesgo. Este acercamiento se puede aplicar para cualquier algoritmo de aprendizaje que requiera datos de *on-policy* sin la necesidad de mantener rejugabilidad de eventos anteriores. Sin embargo, esta técnica no es eficiente en cuanto al muestreo.

Mas detalles sobre métodos y referencias se pueden ver en Vincent François-Lavet, Peter Henderson, Riashat Islam, Marc G. Bellemare and Joelle Pineau (2018) [17] .

Capítulo 3

ESTRATEGIA DE ARBITRAJE MEDIANTE DRL

Uno de los desafíos para establecer una estrategia de arbitraje mediante DRL es principalmente que el portafolio de arbitraje cambia constantemente. Las relaciones de arbitraje, o más bien los parámetros utilizados para generar estos portafolios, son válidas durante un corto periodo de tiempo respecto a la ventana de estimación. Es por este motivo que se debe realizar un marco de trabajo en el que un agente pueda tomar decisiones de forma adecuada manteniendo siempre un portafolio que sea coherente y dinámico.

Uno de los desafíos del DRL corresponde a la toma de decisiones cuando existe una gran cantidad de acciones a tomar, si se opta por generar una acción individual para la posición de cada activo, considerando que el agente busca crear un portafolio de arbitraje la dimensionalidad presenta un gran problema a la hora de tener una política estable y coherente. Para esto se presenta un formato compacto en el cual existe un modelo de cointegración para generar los portafolios de arbitraje, en particular se generan dos portafolios de los cuales se realizan posiciones opuestas obteniendo así solo dos acciones a tomar, larga-corta o cota-larga para los portafolios respectivo. Adicionalmente, es necesario generar una caracterización de los estados del mercado para que el agente tome decisiones en base a relaciones no lineales complejas de estos estados previos.

Con esto definido, el siguiente paso es la generación de una trayectoria de decisiones. Para esto se debe realizar un procedimiento de *'gamification'* de las decisiones de arbitraje. Inspirado en como se estructura un agente para jugar un videojuego, se establece un *'juego'* de trading. Esto consiste en comenzar con los dos portafolios de arbitraje definidos y por consistencia siempre en la misma posición, de este modo, el agente debe observar el estado

del mercado y tomar una decisión sobre la posición actual, si la mantiene o la invierte. De esta manera, el agente debe tomar esta decisión de manera dinámica en cada intervalo de tiempo durante una ventana de tiempo definida, la cual está predefinida en la generación del activo a arbitrar.

En este aspecto, el agente debe tomar la secuencia de decisiones que maximice su puntuación, en este caso, alguna métrica de rendimiento. Ahora, en cada sesión de juego las condiciones iniciales pueden cambiar.

Para ejemplificar esto, utilizando el clásico juego de google del dinosaurio que debe esquivar los obstáculos del camino, el dinosaurio comienza en una misma posición y comienza a moverse. Se deben tomar 3 decisiones; Saltar, agacharse o estar parado. Luego, el escenario va cambiando aleatoriamente, existen cactuses en el camino con diferente tamaño y pájaros. Es de esta forma, que el agente debe observar el estado , que en este caso podría definirse como la imagen actual (el ordenamiento de colores en la grilla de píxeles) la distancia a los obstáculos entre otras formas. Luego, el agente debe ir tomando decisiones a medida que avanza el tiempo para maximizar la distancia recorrida donde se puede definir la recompensa de cada acción de manera unitaria dado que si la acción esquivó el obstáculo entonces la distancia recorrida aumentó.

Aterrizando estos conceptos al juego de arbitraje, se tienen 2 acciones, largo-corto y corto-largo. Estas decisiones se toman observando el estado del mercado donde la definición natural de recompensa sería el retorno de 1 paso para la posición tomada. Posteriormente, al momento de terminar la sesión, e.g. 100 intervalos de tiempo se obtiene la recompensa total de la trayectoria, e.g. el retorno acumulado, el ratio de sharpe de la trayectoria entre otras.

Finalmente, es necesario crear un motor de generación de escenarios del mercado. Dado que las interacciones del mercado son extremadamente complejas, un motor de simulación sintético utilizando modelos estocásticos para la generación de precios e índices de mercado, tiene sus limitaciones, por este motivo se utiliza un acercamiento empírico considerando una ventana temporal amplia para poder generar la mayor cantidad de escenarios posibles de forma que el agente pueda 'jugar' estos escenarios y aprender las complejas relaciones que existen en el mercado para poder maximizar la recompensa total.

3.1. Agente de arbitraje en DRL

3.1.1. Inicialización del juego de arbitraje

- Se tiene el conjunto de activos $X^i \in \mathcal{X}\ \forall i \in \mathcal{I}$

- Se determina el horizonte de juego, considerando periodo de confección del portafolio

T_i y el periodo de término T_f.

- Considerando los precios normalizados de los activos $X_t^i \ \forall t \in [0, T_f]$ donde $X_0^i = 1 \ \forall i \in \mathcal{I}$ (se toma la serie X_t^i y se divide por X_0^i)

- Se confecciona el portafolio de arbitraje por Cointegración:

$$P_t = \beta^T X_t = P_t^L - P_t^S$$

donde β es el vector de cointegración, $i \in L \iff \beta_i \geq 0$, $i \in S \iff \beta_i < 0$, luego considerando un nivel de capital a invertir C, se obtienen los portafolios cointegrados:

$$P_t^L = C\frac{\sum_{j\in L}\beta_j X_t^j}{\sum_{j\in L}\beta_j}, \quad P_t^S = C\frac{\sum_{j\in S}\beta_j X_t^j}{\sum_{j\in S}\beta_j}$$

L_t es el portafolio compuesto por los activos con coeficientes positivos mientras que S_t el portafolio con coeficientes negativos.

- Los coeficientes de cointegración se determinan considerando el intervalo de tiempo $[0, T_i]$.

- De este modo, sin pérdida de generalidad, considerando la ventana de juego en $[0, T] = [T_i, T_f]$, se determina el activo $P_t = P_t^L - P_t^S$ a arbitrar.

3.1.2. Definición del Juego

Se determina el espacio de acciones posibles $A = \{1, -1\}$ donde :

- $a_t = 1$: se está en una posición larga en P_t^L y corta en P_t^S.

- $a_t = -1$: se está en una posición corta en P_t^L y larga en P_t^S.

El espacio de estados $\mathcal{S}$ se determina como:

- Considerando los OHLCV de cada activo $\mathcal{X}$ se construyen los OHLCV aproximados de los portafolios L y S

- Para los portafolios L y S se obtiene un set de indicadores técnicos TI e.g. RSI, OBV, Volatilidad EWMA entre otros.

- De este modo se obtiene una serie de estados $s_t^{S,k}$ y $s_t^{L,k}$ para todo $k \in TI$, además se tiene el estado de la dirección de la posición actual $s_t^a = a_{t-1}$.

3.1.3. Recompensa del Juego

- Considerando los retornos $r_t^L = \ln(P_t^L) - \ln(P_{t-1}^L)$ y $r_t^S = \ln(P_t^S) - \ln(P_{t-1}^S)$, entonces el retorno de la posición de arbitraje en $t-1$ es $r_t = r_t^L - r_t^S$.

- Considerando el nivel de capital C y una posición larga en L corta en S, si P_t^L retorna r_t^L se tiene un capital neto $Ce^{r_t^L}$, si P_t^S retorna r_t^S se obtiene $-Ce^{r_t^s}$, así la posición neta $Ce^{r_t^L} - Ce^{r_t^s} = Ce^{r_t}$.

- Al término del periodo de arbitraje se obtiene una posición neta

$$C\sum_{t=1}^{T} a_{t-1}r_t$$

  asumiendo que se debe mantener un capital de garantía para poder tomar las posiciones cortas necesarias.

Se define el retorno del arbitraje como $R_t = a_{t-1}r_t$, luego en base a este retorno, entonces se la medida de rendimiento natural:

$$U_T = \sum_{t=1}^{T} R_t$$

Por lo tanto, la formulación de problema general está dada por:

Considerando que las decisiones a_t son determinadas por la política π la cual está determinada por una red neuronal con pesos Θ, se tiene que el problema asociado a la maximización de la utilidad de arbitraje:

$$\max_{\Theta} U_T(R|\Theta) \tag{3.1.1}$$

$$\text{s.t. } a_t = \mathbf{F}(\mathcal{S}_{t-1}, \mathcal{S}_{t-2} \dots) \tag{3.1.2}$$

donde $\mathbf{F}$ es la red neuronal.

3.2. Especificación del Modelo

El modelo de arbitraje mediante DRL se define a base de dos módulos; Construcción de activos de arbitraje y la definición de estados.

Para generar la estrategia se consideraron intervalos de tiempo de 30minutos, es decir, los precios y actualización de la información ocurre en intervalos de 30 minutos. Para esto se consideró una ventana de 6 días, lo cual corresponde a 288 intervalos de 30 minutos.

Posteriormente se utiliza una ventana de 1 día, 48 intervalos de 30 minutos para realizar la estrategia de arbitraje.

Los activos utilizados corresponden a 14 crypto activos. Estos son considerados de acuerdo a las características de sus algoritmos y contienen un sesgo de supervivencia del mas fuerte.

1. **BTC**: Bitcoin
2. **ETH**: Ethereum
3. **BNB**: BNB
4. **XRP**: Ripple
5. **ADA**: Cardano
6. **SOL**: Solana
7. **DOT**: Polkadot
8. **BCH**: Bitcoincash
9. **LTC**: Litecoin
10. **AVAX**: Avalanche
11. **ALGO**: Algorand
12. **AAVE**: Aave
13. **UNI**: UniSwap
14. **CAKE**: PancakeSwap

3.2.1. Construcción de portafolios

Considerando la ventana de 6 días, 288 intervalos de tiempo, se construyen dos portafolios A y B desde las relaciones de cointegración. El primer paso es considerar el set de activos $\mathcal{A}$ el cual contiene los 14 activos definidos anteriormente. Luego, se tiene la serie de precios para cada activo $P_t = (P_t^1, P_t^2, \ldots, P_t^{14})$,

- Set de Activos : $\mathcal{A} = \{a_i\}_{i=1}^{14}$, contiene los 14 activos utilizados.
- Serie de precios $\{P_t\}_{t=t_0}^{t_0 T}$ con $P_t = (P_t^1, \ldots P^N, t)$ los precios de cierre en tiempo t se los activos, donde t_0 es el tiempo de inicio de la serie y T es el término. Se utiliza $T - t_0 = 288$.
- Serie de precios normalizada: $\{x_t\}$, donde los precios se normalizan todos comenzando desde el valor 100 y transformados a precios logarítmicos: $x_t^1 = \ln(P_t^1 / P_{t_0}^1)$

Considerando el modelo general de cointegración 1.4.1 :

$$\Delta \mathbf{x}_t = \mu + \sum_{i=1}^{k-1} \Gamma_i \Delta \mathbf{x}_{t-i} + \Pi \mathbf{x}_{t-1} + \varepsilon_t$$

donde $\Delta x_t = x_t - x_{t-1}$. Se considera el test de Johansen donde se selecciona el rango a un nivel de significancia del 10 %. Posteriormente se tiene que $1 \leq \text{Rank}(\Pi) = r \leq p - 1$ con p la cantidad de activos, se obtienen las matrices A y B de dimensión $p \times r$ de rango r tales que $\Pi = AB'$. Finalmente se obtienen r portafolios de cointegración definidos a través

de las constantes de cointegración $\{b_i\}_{i=1}^r$. De este modo, se obtienen el $i-$ésimo par de cointegración definido por $b_i = (b_i^1, \ldots, b_i^N)$ donde se generan dos activos dependiendo de su signo en el vector de cointegración:

$$p \in L_i \iff b_i^p \geq 0 \tag{3.2.1}$$

$$p \in S_i \iff b_i^p < 0 \tag{3.2.2}$$

después los pesos son normalizados para cada vector de cointegración:

$$\frac{b_i^p}{\sum_{j \in L_i} b_i^p}, \ p \in L_i \tag{3.2.3}$$

$$\frac{b_i^p}{\sum_{j \in S_i} b_j^p}, \ p \in S_i \tag{3.2.4}$$

de este modo, se obtienen los portafolios $\{L_i\}_{i=1}^p$ y $\{S_i\}_{i=1}^p$, posteriormente se generan dos portafolios equiponderando los p portafolios obtenidos, definiendo los portafolios A y B, donde los pesos para cada activo $w_A = (w_A^1, \ldots, w_A^N)$, $w_B = (w_B^1, \ldots, w_B^N)$ son tales que $\sum w_K^i = 1, w_K^i \geq 0 \ \forall i, K = A, B$. Además se tiene que si $W_A^i > 0 \Rightarrow W_B^i = 0$ y $W_B^i > 0 \Rightarrow W_A^i = 0$.

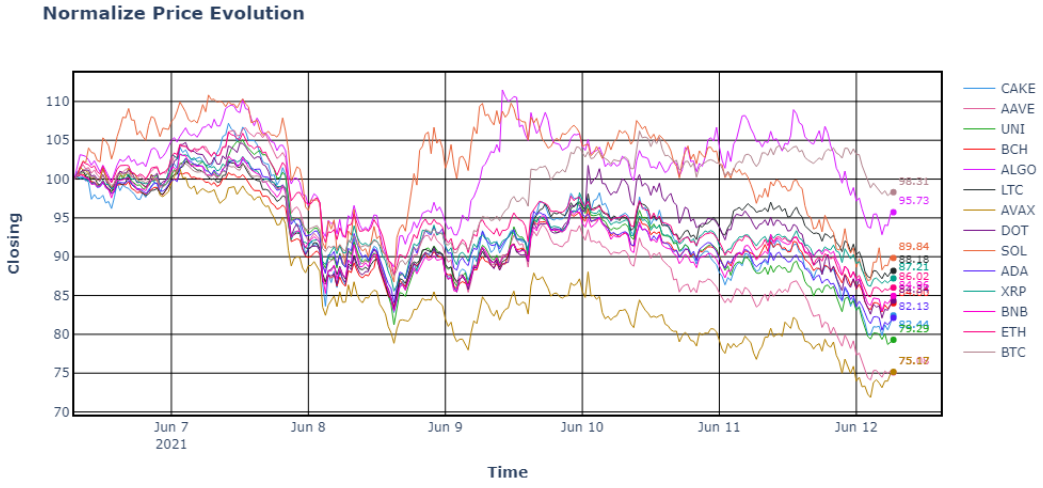

Figura 3.1. Evolución de precios normalizados para los activos seleccionados. Corresponde a 14 series normalizadas comenzando del valor 100.

Seleccionando (un ejemplo) los pesos para los activos de arbitraje:

Activo	A[%]	B[%]
CAKE	5.313	0.
AAVE	0.867	0.
UNI	14.699	0.
BCH	35.971	0.
ALGO	4.231	0.
LTC	7.71	0.
AVAX	0.	1.081
DOT	0.	8.854
SOL	0.	2.089
ADA	0.	41.312
XRP	23.769	0.
BNB	0.	37.559
ETH	7.44	0.
BTC	0.	9.104

Cuadro 3.1. Pesos calculados para la generación de los activos de arbitraje A y B para el ejemplo.

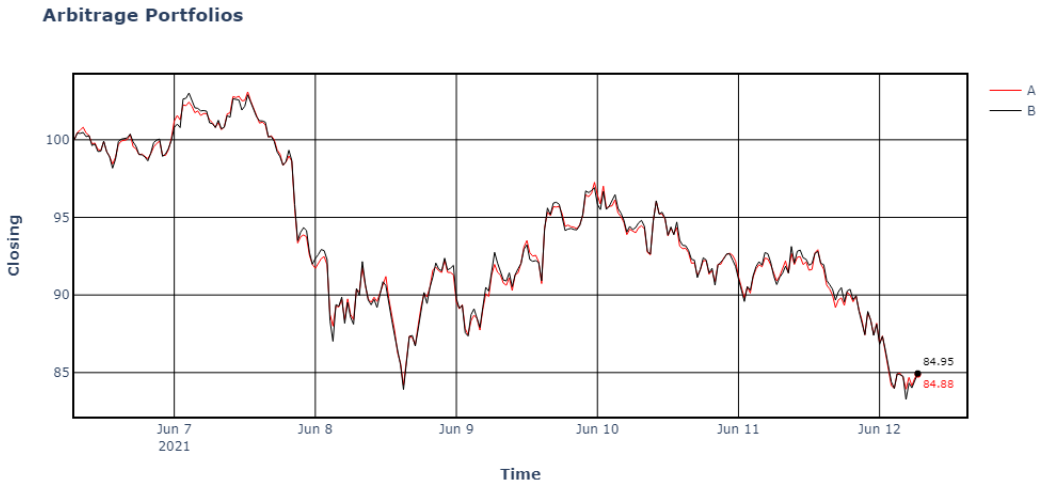

Figura 3.2. Evolución de precio de los activos A y B generados desde la relación de cointegración equiponderando. Se tiene un rango de coitnegración de orden 2.

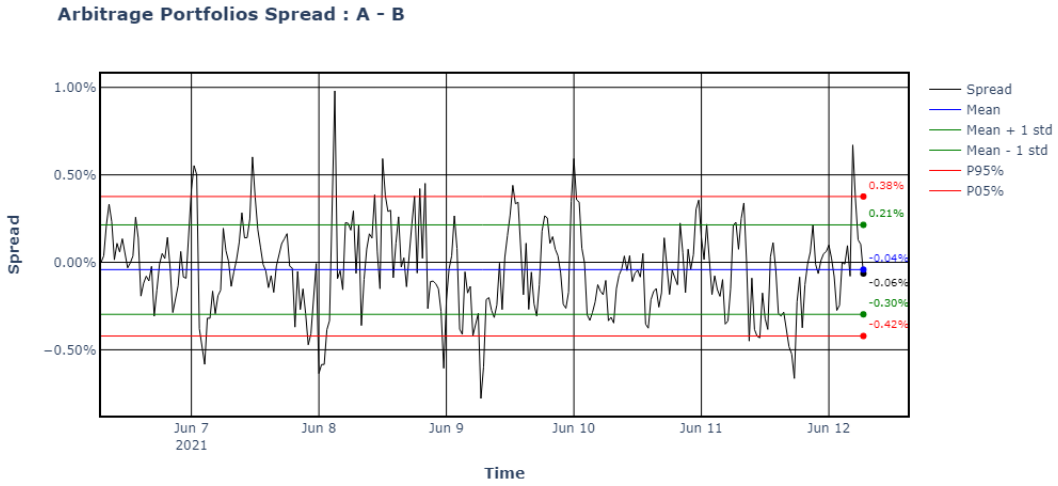

Figura 3.3. Evolución del spread entre el portafolio A y B, Spread = A - B. Se grafican estadísticas adicionales, su media, media más y menos una desviación estándar y sus percentiles 0.95 y 0.05.

3.2.2. Indicadores técnicos y definición de estado

Definidos los activos sintéticos a arbitrar, se debe definir el estado del mercado para que el agente pueda tomar decisiones. Para ello, se construye una serie de indicadores técnicos los cuales se espera que tengan información importante que permita generar señales que el agente pueda aprender y utilizar para maximizar la recompensa.

1. **OLHC**: 2 Estados

   Considerando los pesos de los portafolios, se tienen las velas de 30 minutos para cada activo en el intervalo. De este modo se construyen los valores *Open*, *High, Low, Close* y *Volume* para los activos A y B. Los valores son normalizados en función del precio de cierre de la primera vela, por lo que el valor *Open, High y Low* de la primera vela no corresponden al valor 100.

   Considerando estos valores, se definen inicialmente 8 estados, considerando los valores normalizados para $OHLC$, $\mathbf{O}_t = (O_t^1, \ldots, O_t^N)'$ y de modo similar para los demás:

$$\mathrm{O}_t^i = \mathrm{Open}_t^i / \mathrm{Close}_0^i, \forall i$$

$$\mathrm{H}_t^i = \mathrm{High}_t^i / \mathrm{Close}_0^i, \forall i$$

$$\mathrm{L}_t^i = \mathrm{Low}_t^i / \mathrm{Close}_0^i, \forall i$$

$$\mathrm{C}_t^i = \mathrm{Close}_t^i / \mathrm{Close}_0^i, \forall i$$

Considerando los pesos $W_A = (w_A^1, \ldots, w_A^N)'$ y $W_B = (w_B^1, \ldots, w_B^N)'$, se obtienen los $OHLC$ para los portafolios A y B:

$$O_t^A = \mathbf{O}_t W_A', \quad O_t^B = \mathbf{O}_t W_B'$$

$$H_t^A = \mathbf{H}_t W_A', \quad H_t^B = \mathbf{H}_t W_B'$$

$$L_t^A = \mathbf{O}_t W_A', \quad L_t^B = \mathbf{L}_t W_B'$$

$$C_t^A = \mathbf{C}_t W_A', \quad C_t^B = \mathbf{C}_t W_B'$$

Estas series son utilizadas para la construcción de otros indicadores, sin embargo solo se consideran las series de cierre en los estados,¿.

2. **MACD**: 2 Estados

   El MACD (Media Móvil Convergencia Divergencia) es un indicador técnico que se utiliza comúnmente en el trading de acciones para identificar cambios en el impulso, tendencia y fuerza del precio de una acción. El MACD se calcula restando la media móvil exponencial (EMA) de 26 períodos de la EMA de 12 períodos, y luego trazando una EMA de 9 períodos del MACD como una línea de señal.

   $$\text{MACD(l,s)}_t = \text{EMA}(l)_t - \text{EMA}(s)_t$$

   donde

   $$\text{EMA}(k)_t = \text{Precio}_t k + \text{EMA}_{t-1}(1-k)$$

   Este indicador es utilizado con una media de largo plazo de 14 periodos y una media de corto plazo de 5 periodos. Se calculara para los precios de cierre de los portafolios A y B.

3. **RSI**: 2 Estados

   RSI(N)

   $RSI = 100 - 100/(1 + RS)$

   $RS = AvgU/AvgD$

   $AvgU =$ Promedio de todas las subidas en N periodos

   $$AvgU = \frac{1}{N} \sum [\mathbf{Precio}_{t-i} - \mathbf{Precio}_{t-i-1}]^+$$

   $AvgD =$ Promedio de todas las bajadas en N periodos

   $$AvgD = \frac{1}{N} \sum [\mathbf{Precio}_{t-i} - \mathbf{Precio}_{t-i-1}]^-$$

este indicador es utilizado con $N = 20$ respecto al precio de cierre. Se calcula para los portafolios A y B, por lo tanto se generan 2 series adicionales.

4. **KFMEAN**: 2 Estados

   Se utiliza un filtro de Kalman para generar una señal de media móvil del precio de los activos sintéticos A y B. Este filtro consiste en modelar la serie de precios P_t considerando una media móvil no observable μ_t de la siguiente manera:

$$\text{Ecuación de Medición: } P_t = \mu_t + \varepsilon_t, \quad \varepsilon_t \sim N(0, \sigma^2) \tag{3.2.5}$$

$$\text{Ecuación de Estado: } \mu_t = \mu_{t-1} + \nu_t, \quad \epsilon_t \sim N(0, \sigma_\mu^2) \tag{3.2.6}$$

   adicionalmente, se obtiene la ganancia de Kalman:

$$K_t = \frac{\mathbb{C}\text{ov}\left[\mu_t, \nu_t | \mathcal{F}_{t-1}\right]}{\mathbb{V}\left[\nu_t | \mathcal{F}_{t-1}\right]} = \frac{\mathbb{C}\text{ov}\left[\mu_t, \nu_t | \mathcal{F}_{t-1}\right]}{\mathbb{V}\left[\mu_t | \mathcal{F}_{t-1}\right] + \mathbb{V}\left[\varepsilon_t | \mathcal{F}_{t-1}\right]}, \quad \nu_t = P_t - \mathbb{E}\left[P_t | \mathcal{F}_{t-1}\right] \tag{3.2.7}$$

   entonces se tiene el MLE de μ_t dado el precio P_t como:

$$\hat{\mu}_t = \mathbb{E}\left[\mu_t | P_t\right] = (1 - K_t)\mathbb{E}\left[\mu_t | \mathcal{F}_{t-1}\right] + K_t P_t \tag{3.2.8}$$

   donde las volatilidades son estimadas en la muestra de entrenamiento. Más detalle sobre filtros pueden ser consultados en Durbin & Koopman 2012 [14].

5. **ST**: 2 Estados

   Este indicador *Stock Strength* propuesto en Wu, Chen, Wang, Troiano, Loia y Fujita 2020 [64], pretende discriminar si el mercado está en un régimen tendencial o volátil:

$$ST = \frac{P_n - P_1}{\sum_{i=1}^{n} |P_{i+1} - P_1|} \tag{3.2.9}$$

   el cual es aplicado a una ventana de 10 intervalos a los activos sintéticos A y B. Este mide el retorno acumulado a un intervalo de n periodos dividido por la suma de los retornos absolutos del periodo, por lo tanto si se está en un momento volátil este indicador es cercano a cero, mientras que en un régimen tendencial es muy positivo o muy negativo.

6. **EWMA**: 3 Estados

   Para introducir dependencia de la volatilidad condicional, se calcula la volatilidad condicional por EWMA (Exponentially Weighted Moving Average) para la serie de

retornos de los activos sintéticos A y B, junto con la serie del spread A-B.

$$\sigma_t^2 = \lambda\sigma_t^2 + (1-\lambda)r_t^2 \tag{3.2.10}$$

se utiliza un parámetro de memoria $\lambda = 0.94$ por lo cual un shock de volatilidad tiene harta persistencia en el tiempo.

7. **BIAS**: 2 Estados

   El Bias mide la desviación relativa del precio respecto a un valor de media móvil referencial. En este caso se aplica a la media móvil obtenida a través del filtro de kalman.

$$\text{Bias}_t = \frac{P_t - \mu_t}{\mu_t} \tag{3.2.11}$$

   Esta medida es aplicada a los activos sintéticos A y B.

8. **Signs**: 2 Estados

   Otra medida de fuerza relativa, corresponde al signo de los retornos obtenidos. A diferencia de las medidas anteriores, no se mide el efecto neto o comparación de fuerza respecto a los retornos en términos cuantitativos sino a la cantidad de retornos positivos versus los negativos.

$$\text{Signs}_t = \frac{\sum_{i=0}^{N}[\text{sign}(r_{t-i})]^+}{N} \tag{3.2.12}$$

   esta medida es aplicada a ambos activos sintéticos A y B, con una ventana de 10 periodos. A diferencia de otras medias, esta toma una cantidad discreta de valores.

9. **SRSI**: 2 Estados

   Adicionalmente, se aplica el RSI estocástico, el cual a diferencia del RSI normal, este es una versión escalada con un escalador de mínimo y máximo a un periodo determinado.

$$\text{SRSI}_t = \frac{\text{RSI}_t - \text{mín}_{k=t-N:t}\,\text{RSI}_k}{\text{máx}_{k=t-N:t}\,\text{RSI}_k - \text{mín}_{k=t-N:t}\,\text{RSI}_k} \tag{3.2.13}$$

   Se aplica para los dos sintéticos A y B con una ventana de 20 periodos.

10. **Candle Shadows**: 4 Estados

    Finalmente, una métrica de volatilidad de los precios respecto a la amplitud de la velas del periodo es aplicada.

$$\text{Upper}_t = \frac{\text{High}_t - \text{máx}\{\text{Open}_t, \text{Close}_t\}}{\text{High}_t - \text{Low}_t} \tag{3.2.14}$$

$$\text{Lower}_t = \frac{\text{mín}\{\text{Open}_t, \text{Close}_t\} - \text{High}_t}{\text{High}_t - \text{Low}_t} \tag{3.2.15}$$

estas medidas son aplicadas a los sintéticos A y B.

Adicionalmente, se introducen el spread A - B como un estado, la media móvil calculada a través del filtro de kalman para el spread y finalmente un índice de turbulencia de mercado el cual corresponde a la distancia de Mahalanobis de los retornos de mercado:

$$\textbf{Turbulence: } T_t = \sqrt{(\mathbf{r}_t - \mu)'\Sigma^{-1}(\mathbf{r}_t - \mu)} \tag{3.2.16}$$

donde μ corresponde al vector de medias de los retornos del periodo de entrenamiento, r_t a los retornos de los activos y Σ a la matriz de covarianza de los retornos durante el periodo de entrenamiento, generando así, un total de 26 estados que se espera que el agente pueda utilizar para aprender sobre las interacciones del mercado.

3.2.3. Configuración de Entrenamiento

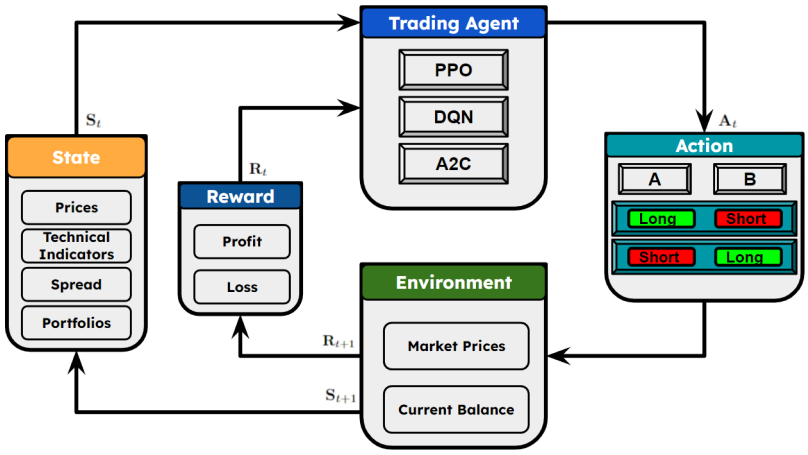

Figura 3.4. Esquema de interacción del sistema de aprendizaje reforzado profundo para la aplicación de trading. Los estados corresponden a los definidos a través de indicadores técnicos, volatilidades y otros generados desde los activos sintéticos y spread. Las acciones posibles corresponden a una posición larga en A y corta en B o una posición corta en A y larga en B.

Generación de Escenarios

Para la generación de escenarios se utilizó una ventana histórica de velas con una frecuencia de 30 minutos. El periodo de entrenamiento comienza desde 2020-11-01 00:00:00 hasta 2022-10-11 05:30:00 siendo un total de 34000 velas de precios para los 14 activos de

estudio. De este modo, se utiliza una ventana de 336 intervalos de tiempo para generar un escenario de trading. Este intervalo se divide en 2 componentes: los primeros 288 valores son utilizados para la confección de portafolios y estados, posteriormente comenzando desde el último se tienen 48 intervalos para tomar 47 decisiones generando mas de 30.000 escenarios distintos.

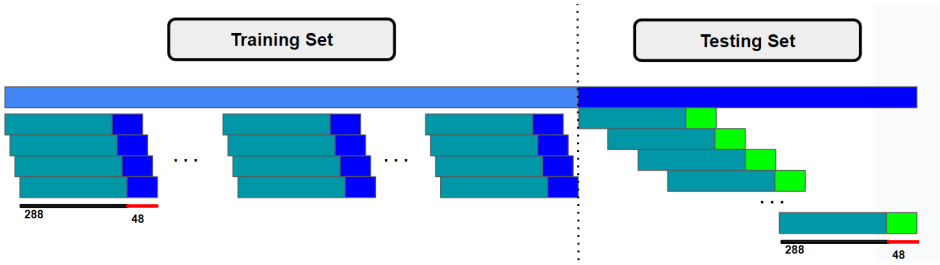

Figura 3.5. Esquema de segmentación de escenarios de entrenamiento y prueba. En el conjunto de entrenamiento existe solapamiento de escenarios mientras que en el conjunto de prueba no ocurre y tampoco existe información utilizada para entrenamiento.

Posteriormente, el periodo de prueba contempla su inicio a la fecha 2022-10-19 07:30:00 y termina en 2023-03-18 07:30:00 completando un total de 150 días de transacciones.

Parámetros y Entrenamiento

Para entrenar los modelos de DRL, se utiliza el lenguaje de programación Python 3.8, donde sobre la librería Stable Baselines3 se entrenan los agentes Advantage Actor Critic (A2C) (Mnih et al. 2016[39]), Deep Q Network (DQN) (Mnih et al. 2013[41],Mnih et al. 2015[40]), Proximal Policy Optimization algorithm (PPO) (Schulman et al. 2017[52]).

Se utilizó un factor gamma de 0.995, utilizando la política MlpPolicy la cual corresponde a una red neuronal feedforward con múltiples capas ocultas densas la cual retorna las probabilidades de las acciones. Más detalles sobre la estructura de la red se pueden ver en la documentación de la librería, para el alcance de este trabajo no se profundizó en la estructura de la red. Respecto a la cantidad se entrenó con un total de pasos de 1e6, donde en cada escenario ocurren 47 pasos, es decir, se entrenó en un total de aproximadamente 21.000 escenarios. Además, los escenarios fueron seleccionado aleatoriamente de la muestra de manera equiprobable.

La función de recompensa utilizada corresponde al retorno logarítmico y la recompensa total al retorno acumulado.

Capítulo 4

ANÁLISIS DE RESULTADOS

El ratio de Sharpe es una de las métricas de rendimiento más populares en la cual se ajusta el retorno esperado frente a otra alternativa, generalmente una tasa libre de riesgo o más general el retorno esperado de una inversión de muy bajo riesgo alternativa, sobre la desviación estándar del retorno. Esta métrica supone que la distribución de los retornos se puede categorizar en la familia de distribuciones de posición y escala. Entonces, los elementos fundamentales de esta métrica corresponden a un valor de posición (la media) y un valor de escala (desviación estándar) que juega el rol de medida de riesgo. Sin embargo, hay más características de una distribución que su posición y escala, estos corresponden al primer y segundo momento de la distribución pero también son de igual importancia momentos de orden mayor como la asimetría y la curtosis (parámetro de forma), donde es conocido que los retornos exhiben características en estos dos momentos superiores; son negativamente asimétricos y presentan curtosis elevada, mayor a una distribución normal. Por lo tanto, como el ratio de Sharpe solo contiene los dos primeros momentos se ignoran las características sobre los momentos superiores. Por este motivo se plantea una métrica que considere estos elementos.

La métrica propuesta, en contraste con el ratio de Sharpe, busca una monotonía estricta respecto a la dominancia estocástica por lo que no puede ser establecida únicamente con la posición y escala de la distribución. La métrica *medida de rendimiento económico* (EPM) propuesta se obtiene dividiendo la media de una oportunidad de inversión por su *índice económico de riesgo* de Aumann & Serrano (2008)[1], Ulrich & Pigorsch (2012)[25], denotada como índice de AS. Si los retornos se distribuyen normal entonces la métrica coincide con el índice de Sharpe en términos de orden (si $r \sim F \xrightarrow{d} N(\mu, \sigma^2)$ entonces EPM $\rightarrow 2\text{Sharpe}^2$), además esta métrica es apropiada tanto para frecuencias altas como bajas mientras que

el índice de Sharpe es más apropiada para bajas frecuencias. Para esto se propone un estimador de momentos para el EPM tanto paramétrico como no paramétrico. Para su estimación paramétrica se propone que los retornos siguen una distribución hiperbólica generalizada (GHYP), que puede modelar asimetría como colas pesadas. De este modo, se consideran momentos de orden superior para la métrica EPM en una forma sencilla de obtener e interpretar. Uno de los resultados de la métrica coincide que para retornos con asimetría relativamente alta y curtosis baja la métrica establece un orden mayor respecto al ratio de Sharpe.

4.1. Métricas de Riesgo y Rendimiento

Para contrastar las diferentes estrategias implementadas, se utilizarán diverso criterios de rendimiento. En primer lugar se registra la rentabilidad final obtenida de la estrategia (PnL), también se añaden las métricas de *risk-reward* como el índice de Sharpe junto con el índice de Calmar y el índice de rendimiento económico (EPM) utilizando una distribución hiperbólica generalizada.

Economic Performance Measure

Considerando $\tilde{r} \sim F$ y r^f la tasa libre de riesgo, entonces $r = \tilde{r} - r^f \sim F$ corresponde al exceso de retorno. Se define la métrica de rendimiento económico (EPM) relativa al índice AS de riesgo:

$$\mathrm{EPM}(r) = \frac{E(r)}{AS(r)} = \frac{E(\tilde{r}) - r^f}{AS(\tilde{r} - r^f)} \tag{4.1.1}$$

considerando $M(t)$ la MGF de los retornos, se tiene que el índice AS es el valor de $s > 0$ tal que $M(-1/s) = 1$, $AS(r) = \{s > 0 : M(-1/s) = 1\}$.

Asumiendo que r es generado por una distribución hiperbólica generalizada :

$$r \sim \mathrm{GH}(\lambda, \chi, \psi, \mu, \sigma^2, \gamma), \quad r \overset{d}{=} \mu + W\gamma + \sqrt{W}A\mathbf{Z} \tag{4.1.2}$$

donde $W \sim \mathrm{GIG}(\lambda, \chi, \psi)$ (inversa gaussiana generalizada) , $\mathbf{Z} \sim N(0, 1)$. Los parámetros λ, χ, ψ determinan la forma de la distribución, esto es, cuanto peso se asigna a las colas y al centro. En general, para valores grandes de estos parámetros la distribución se acerca a una distribución normal. Mientras μ corresponde al parámetro de posición y σ^2 a la escala. Por último el parámetro γ corresponde al parámetro de asimetría, $\gamma = 0$ corresponde a una distribución simétrica. De esta manera se tiene la formulación estocástica:

$$r|_W \sim N(\mu + W\gamma, W\sigma^2) \tag{4.1.3}$$

entonces se tiene que $E(r) = \mu + E(W)\gamma$ y $Var(r) = \gamma^2 Var(W) + E(W)\sigma^2$. Luego la función de densidad:

$$f_r(x) = \int_0^\infty f_{r|W}(x|w) f_W(w) dw = \int_0^\infty \frac{e^{\frac{(x-\mu)\gamma}{\sigma^2}}}{\sqrt{2\pi\sigma^2 w}} \exp\left\{ -\frac{Q(x)}{2w} - \frac{w\gamma^2}{2\sigma^2} \right\} f_W(w) dw \quad (4.1.4)$$

$$f_r(x) = \frac{(\sqrt{\psi/\chi})^\lambda (\psi + \gamma^2/\sigma^2)^{1/2-\lambda}}{\sqrt{2\pi\sigma^2}\mathbf{K}_\lambda(\sqrt{\chi\psi})} \times \frac{\mathbf{K}_{\lambda-1/2}\left(\sqrt{(\chi + Q(x))(\psi + \gamma^2/\sigma^2)}\right) e^{\frac{\gamma(x-\mu)}{\sigma^2}}}{\left(\sqrt{(\chi + Q(x))(\psi + \gamma^2/\sigma^2)}\right)^{1/2-\lambda}} \quad (4.1.5)$$

donde $Q(x) = (x-\mu)^2/\sigma^2$ y $\mathbf{K}_\lambda(\cdot)$ una función de Bessel modificada de tercer tipo.

La MGF se puede obtener fácilmente dado que es una mezcla de normales, entonces:

$$\mathbf{M}_{GH}(t) = E\left[E\left[e^{tr}|W\right]\right] = e^{t\mu} E(\exp(W(t\gamma + 1/2t^2\sigma^2))) \quad (4.1.6)$$

$$= e^{t\mu}\left(\frac{\psi}{\psi - 2t\gamma - t^2\sigma^2}\right)^{\lambda/2} \frac{\mathbf{K}_\lambda(\sqrt{\psi(\chi - 2t\gamma t^2\sigma^2)})}{\mathbf{K}_\lambda(\sqrt{\chi\psi})}, \quad \chi \geq 2t\gamma + t^2\sigma^2 \quad (4.1.7)$$

Siguiendo el EPM utilizando una NIG, la GH tiene el caso particular de que coincide con la NIG cuando $\lambda = -1/2$.

Entonces, siguiendo a Tiantian Li, Young Shin Kim,Qi Fan y Fumin Zhu (2021)[35] definiendo el coeficiente de Aumann-Serrano $R_{AS}(X)$ como la solución de $E(\exp(-X/R)) = 1$ se tiene que $R_{AS}(r) = \sigma^2/(\gamma + \sqrt{\gamma^2 + \sigma^2\psi})$ por lo tanto:

$$\text{EPM}_{GH}(r) = \frac{E(r)}{R_{AS}(r)} = \frac{E(r)(\gamma + \sqrt{\gamma^2 + \sigma^2\psi})}{\sigma^2} \quad (4.1.8)$$

Maximal Drawdown

Considerando $W(t)$ el capital total en el tiempo:

$$W^*(t) := \max_{0\leq\tau\leq t} W(\tau), \quad t^* := \arg\max_{0\leq\tau\leq t} W(\tau) \quad (4.1.9)$$

$$\textbf{Drawdown:} \quad D(t) := \frac{W^*(t) - W(t)}{W^*(t)} = 1 - \frac{W(t)}{W^*(t)} \quad (4.1.10)$$

La pérdida acumulada desde el máximo hasta el tiempo actual , t.

$$D(t) = \begin{cases} 0 & \text{si } t^* = t, \text{no hay pérdida acumulada} \\ 1 - W(t)/W^*(t) & \text{si } t^* < t, \text{hay una pérdida } D(t) \text{ acumulada} \end{cases} \quad (4.1.11)$$

Por lo tanto, se define el Drawdown máximo como

$$\textbf{Maximal Drawdown: } D^*(t) = \max_{\tau \leq t} D(\tau) \tag{4.1.12}$$

finalmente se tiene el índice de Calmar como:

$$\textbf{Calmar : } \frac{E(r)}{D^*} \tag{4.1.13}$$

Value at Risk

Corresponde a la pérdida máxima esperada con una probabilidad α sobre un periodo:

$$VaR_\alpha(X) = \text{ínf}\{x : F_X(x) \geq \alpha\} = F_X^{-1}(\alpha) \tag{4.1.14}$$

Expected Shortfall

El Expected Shortfall corresponde a la pérdida esperada dado que la pérdida excede el $\alpha-$cuantil del VaR:

$$ES_\gamma(X) = \frac{1}{\gamma}\int_0^\gamma VaR_\alpha(X)d\alpha \tag{4.1.15}$$

donde $ES_\gamma(X)$ es el *Expected Shortfall* de la v.a. X a un nivel γ, esto es, dada una pérdida r se tiene que $ES = E[r|r < VaR]$.

4.2. Resultados

Los resultados se presentan considerando el rendimiento diario de las estrategias como consecuencia de la trayectoria de decisiones realizadas a nivel intra diario. Además se realizan las comparaciones y mediciones de riesgo y retorno ajustado por riesgo medido a través de diferentes métricas.

Las medidas de riesgo de VaR y ES contemplan los riesgos de cola, o las pérdidas extremas incurridas a un nivel del 5 %, de la distribución incondicional de los retornos. Estas medidas no se ven influenciadas por retornos en la cola superior de la distribución, es decir, miden los valores extremos a un percentil establecido, generalmente al 5 % o 1 % los cuales sirven de umbrales para determinar los peores escenarios que pueden ocurrir, sin considerar los escenarios extremos pero positivos. Estas se estiman de acuerdo a la distribución empírica.

La desviación estándar es una medida de riesgo popular y de fácil entendimiento, esta corresponden al nivel de variabilidad o la escala en la cual se obtiene los retornos. Si bien mide correctamente la variabilidad, se ve fácilmente influenciada por retornos positivos y en particular por retornos positivos extremos, lo cual no necesariamente se asocian a un mayor riesgo.

Otra medición relevante corresponde al Drawdown, el cual mide las pérdidas *peak to peak*, es decir, las pérdidas acumuladas desde el mayor valor de la cartera. Esta medición es fundamental al evaluar estrategias en entornos de riesgos extremos, pues mide la amplitud porcentual de las pérdidas generadas.

Finalmente, considerado que los tienen características en sus momentos superiores que pueden determinar si una estrategia es mejor a otra en términos de la capacidad de generación de rentabilidad respecto al riesgo que traen, se miden a través del coeficiente de Aumann-Serrano, AS, y posteriormente comparando su rendimiento ajustado por el riesgo considerando los cuatro momentos de la distribución a través de la medida de rendimiento económico EPM.

4.2.1. Rendimiento diario

Respecto a los activos utilizados, estudiando su rendimiento desde 2022-01-01 00:00:00 a la fecha 2023-03-17 07:30:00, siendo un total de 440 días de trading considerando una granularidad diaria:

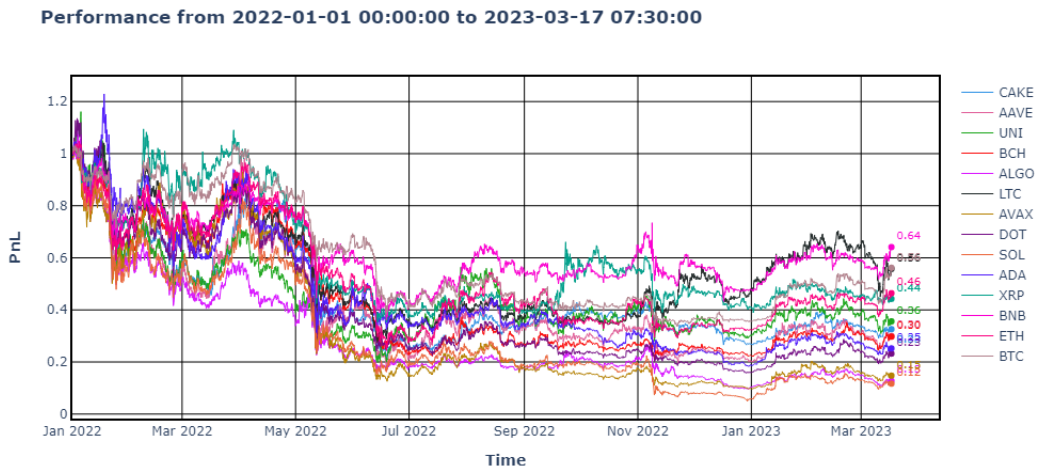

Figura 4.1. Rendimiento de los 14 activos considerando su valor normalizado al precio de cierre de 2022-01-01 a las 00:00 horas. Corresponde a una muestra de un total de 440 días de transacción.

En términos de sus retornos diarios:

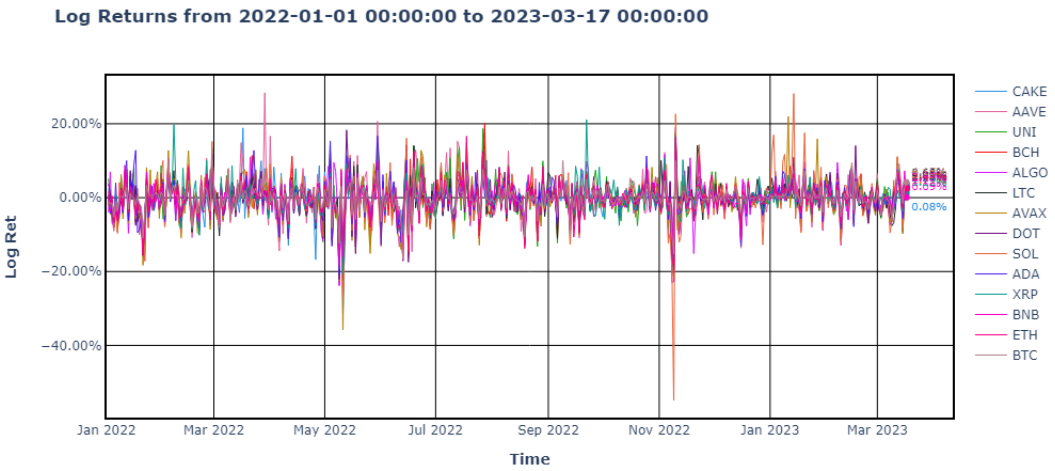

Figura 4.2. Retornos logarítmicos diarios de los 14 activos considerando su valor normalizado al precio de cierre de 2022-01-01 a las 00:00 horas. Corresponde a una muestra de un total de 440 días de transacción.

En términos de sus estadísticas:

	SOL	AVAX	ALGO	CAKE	AAVE	DOT	UNI	ETH	ADA	BNB	XRP	LTC	BCH	BTC
mean	-0.005	-0.0044	-0.0048	-0.0026	-0.0028	-0.0035	-0.0024	-0.0018	-0.0032	-0.0011	-0.0019	-0.0014	-0.0028	-0.0014
std	0.0649	0.058	0.0505	0.0464	0.0606	0.0475	0.0534	0.0434	0.0468	0.0369	0.042	0.0447	0.0429	0.033
min	-0.549	-0.3578	-0.2618	-0.315	-0.2431	-0.2229	-0.2141	-0.191	-0.207	-0.2028	-0.2067	-0.2086	-0.1832	-0.167
1 %	-0.1607	-0.1509	-0.1442	-0.1694	-0.1617	-0.1519	-0.1399	-0.1432	-0.1314	-0.1253	-0.1147	-0.124	-0.1365	-0.1061
5 %	-0.1011	-0.0999	-0.086	-0.0694	-0.1056	-0.0804	-0.0914	-0.0723	-0.0786	-0.0601	-0.0645	-0.0763	-0.0749	-0.0536
50 %	-0.0054	-0.003	0.0028	0.0001	-0.0013	-0.0016	0.0	-0.0007	-0.0024	-0.0007	-0.0003	-0.0012	-0.0003	-0.0012
95 %	0.0914	0.0765	0.0662	0.0602	0.0869	0.0687	0.078	0.067	0.0701	0.0512	0.0612	0.0671	0.0606	0.0489
99 %	0.1575	0.128	0.1015	0.1009	0.1542	0.1075	0.1393	0.1082	0.1235	0.0867	0.1017	0.1	0.1074	0.0917
max	0.2816	0.2198	0.1891	0.1886	0.2836	0.1831	0.1871	0.1664	0.1673	0.1287	0.2113	0.1771	0.2012	0.1353
skew	-1.2265	-0.6114	-0.9191	-1.2515	-0.0664	-0.4903	-0.1946	-0.3799	-0.2319	-0.8471	0.0507	-0.3563	-0.3495	-0.4649
kurt	12.3616	4.1204	3.8486	7.3705	2.524	2.8801	1.4932	2.9112	2.3647	4.6532	5.173	2.509	3.1772	4.3016
ES99 %	0.203	0.1668	0.1253	0.1299	0.1938	0.1402	0.1565	0.1375	0.1452	0.1024	0.1574	0.1345	0.1359	0.1072
ES95 %	0.1311	0.1171	0.0913	0.0891	0.1341	0.095	0.1132	0.093	0.1032	0.0759	0.096	0.0938	0.0886	0.0739
ES5 %	-0.1571	-0.1434	-0.1314	-0.1281	-0.145	-0.1201	-0.1254	-0.1105	-0.1167	-0.0937	-0.0997	-0.1103	-0.1126	-0.087
ES1 %	-0.2763	-0.2266	-0.2216	-0.2119	-0.2101	-0.1856	-0.1778	-0.1651	-0.1646	-0.1637	-0.1625	-0.162	-0.1596	-0.1321

Cuadro 4.1. Resumen de estadísticas de la muestra de 440 retornos logarítmicos diarios. Los valores 1 %, 5 %, 95 % y 99 % corresponden a los respectivos percentiles mientras que los valores de $ES1\,\%$ y similares correspodnen al Expected Shortfall al nivel de confianza dado. Estos se obtienen desde la distribución empírica de los datos obtenidos.

El activo con mayor pérdida diaria en el periodo corresponde a Solana (SOL) con retorno

mínimo del -0.549 y en términos porcentuales corresponde a una pérdida a un día del 42.25 %, mientras el con menor pérdida máxima es Bitcoin (BTC) con una pérdida máxima en un día del 15.38 %. Respecto a las ganancia máximas se tiene el con mayor ganancia máxima a un día Aave (AAVE) con una ganancia de 32.79 % y el con menor ganancia máxima es BnB con 13.73 %.

Se observa que durante el periodo todos obtienen un retorno medio negativo, también todos son negativamente asimétricos salvo Ripple (XRP) el cual tiene un coeficiente de asimetría positivo de 0.05. Respecto a los valores de exceso de curtosis, todos presentan niveles de exceso pero en particular Solana obtiene un exceso de 14.

En cuanto a la correlación de los activos en su primera diferencia logarítmica:

	CAKE	AAVE	UNI	BCH	ALGO	LTC	AVAX	DOT	SOL	ADA	XRP	BNB	ETH	BTC
CAKE	1.0	0.7425	0.7155	0.7036	0.7417	0.7161	0.7476	0.7693	0.7042	0.7358	0.6906	0.8503	0.7511	0.7226
AAVE	0.7425	1.0	0.8204	0.749	0.7629	0.7582	0.7912	0.8035	0.7566	0.7507	0.7103	0.7684	0.8224	0.7761
UNI	0.7155	0.8204	1.0	0.7746	0.7632	0.7851	0.7823	0.8075	0.7585	0.7758	0.7041	0.7627	0.8045	0.7562
BCH	0.7036	0.749	0.7746	1.0	0.7511	0.826	0.7758	0.8013	0.7333	0.7671	0.7417	0.7785	0.8215	0.8287
ALGO	0.7417	0.7629	0.7632	0.7511	1.0	0.7761	0.8078	0.8113	0.7721	0.7952	0.7622	0.7794	0.7801	0.7495
LTC	0.7161	0.7582	0.7851	0.826	0.7761	1.0	0.7638	0.8102	0.7442	0.797	0.7532	0.7897	0.8202	0.8014
AVAX	0.7476	0.7912	0.7823	0.7758	0.8078	0.7638	1.0	0.8358	0.8087	0.7932	0.7528	0.8155	0.83	0.7856
DOT	0.7693	0.8035	0.8075	0.8013	0.8113	0.8102	0.8358	1.0	0.7974	0.8279	0.7518	0.8231	0.8348	0.7919
SOL	0.7042	0.7566	0.7585	0.7333	0.7721	0.7442	0.8087	0.7974	1.0	0.7611	0.7183	0.773	0.7991	0.7654
ADA	0.7358	0.7507	0.7758	0.7671	0.7952	0.797	0.7932	0.8279	0.7611	1.0	0.7458	0.777	0.7911	0.7799
XRP	0.6906	0.7103	0.7041	0.7417	0.7622	0.7532	0.7528	0.7518	0.7183	0.7458	1.0	0.7448	0.7563	0.7389
BNB	0.8503	0.7684	0.7627	0.7785	0.7794	0.7897	0.8155	0.8231	0.773	0.777	0.7448	1.0	0.8313	0.8202
ETH	0.7511	0.8224	0.8045	0.8215	0.7801	0.8202	0.83	0.8348	0.7991	0.7911	0.7563	0.8313	1.0	0.895
BTC	0.7226	0.7761	0.7562	0.8287	0.7495	0.8014	0.7856	0.7919	0.7654	0.7799	0.7389	0.8202	0.895	1.0

Cuadro 4.2. Matriz de correlación de pearson de los retornos logarítmicos diarios.

Se tiene que el activo con menor correlación es Ripple (XRP) con PancakeSwap (CAKE) con una correlación del 69 %. En general los activos están muy correlacionados, se observa directo de los gráficos anteriores. Esto dificulta la posibilidad de diversificación y reducción del riesgo sistémico al generar portafolios de solamente posiciones largas de estos activos, de este modo, junto con el alto grado de volatilidad de los activos generan un mercado apropiado para actividades de arbitraje en cortos intervalos de tiempo maximizando la utilidad en base a ineficiencias de corto plazo.

Estudiando los retornos y rendimientos diarios de los activos para el periodo de prueba:

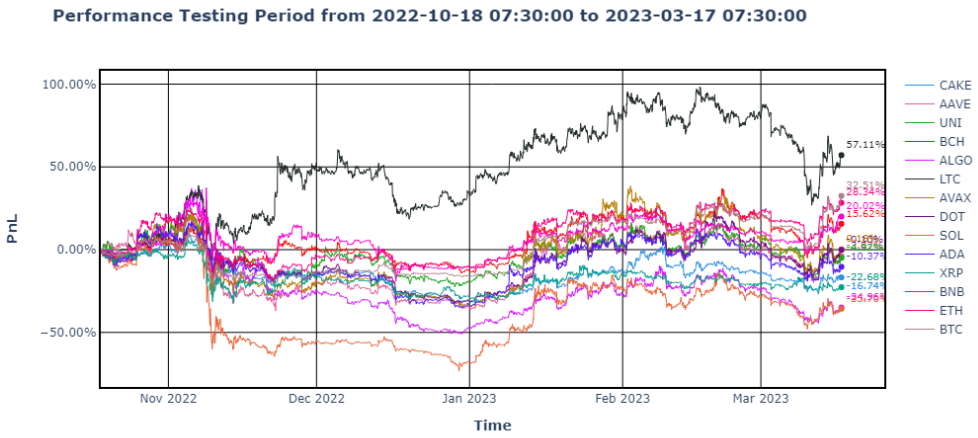

Figura 4.3. Rendimiento de 150 días de los activos de estudio.

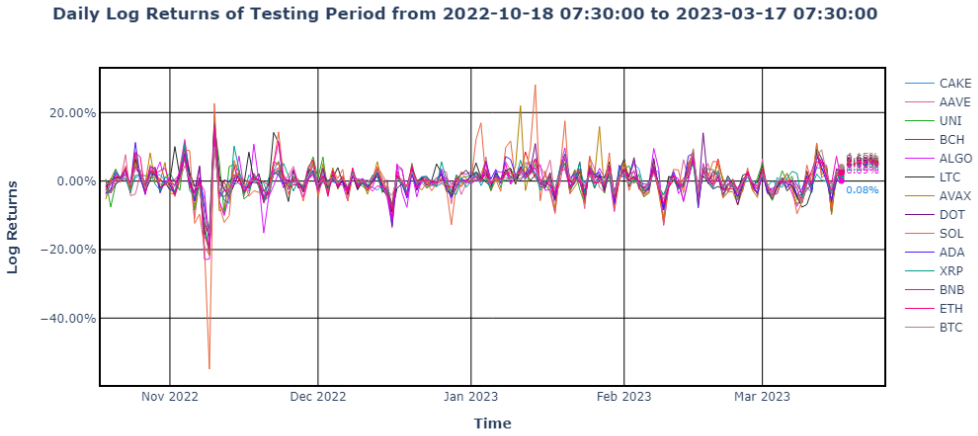

Figura 4.4. Retornos logarítmicos del periodo de prueba.

Se observa en ambos gráficos, tanto en el periodo de testeo como en la visión mas general del mercado, si bien es altamente correlacionado, al muy corto plazo las series de precios normalizadas de todos los activos tienen un movimiento conjunto que posteriormente comienza a diverger, sin embargo mantienen la misma dirección de movimiento considerando que en momentos puntuales un activo se aleja de la tendencia de mercado y luego se mantiene pero a un nivel más alto o más bajo. Esto reafirma el potencial que tienen estrategias de arbitraje estadístico para generar retornos fuera de la tendencia de mercado en un ambiente muy riesgoso.

	SOL	ALGO	AAVE	UNI	ETH	AVAX	XRP	BNB	ADA	LTC	CAKE	BCH	DOT	BTC
mean	-0.0027	-0.0027	-0.0005	-0.0005	0.0018	0.0002	-0.0016	0.0013	-0.0006	0.003	-0.001	0.0011	0.0001	0.002
std	0.0778	0.0523	0.0485	0.0445	0.0379	0.0515	0.0367	0.0344	0.0387	0.0429	0.032	0.0338	0.0394	0.0294
min	-0.549	-0.2292	-0.2173	-0.2141	-0.191	-0.2158	-0.1977	-0.2028	-0.159	-0.1549	-0.1705	-0.148	-0.1523	-0.1526
1 %	-0.1661	-0.1901	-0.1516	-0.1188	-0.1223	-0.1187	-0.1053	-0.0915	-0.1087	-0.1138	-0.1139	-0.1015	-0.1198	-0.0846
5 %	-0.0989	-0.0877	-0.0723	-0.0672	-0.0467	-0.0749	-0.0471	-0.0426	-0.0567	-0.0563	-0.0498	-0.0456	-0.0573	-0.033
50 %	-0.0035	0.0028	0.0026	0.0017	0.0001	0.0	-0.0001	0.0006	-0.0006	0.0009	-0.0023	0.0012	-0.0021	-0.0003
95 %	0.1054	0.0666	0.0679	0.0548	0.0641	0.0676	0.0495	0.05	0.0679	0.0702	0.0399	0.052	0.0677	0.0475
99 %	0.2016	0.1121	0.1039	0.0852	0.0785	0.1686	0.1051	0.0939	0.0969	0.1274	0.0667	0.0799	0.1041	0.0916
max	0.2816	0.1891	0.1573	0.1634	0.164	0.2198	0.1684	0.1287	0.1495	0.1771	0.1223	0.1432	0.1409	0.1002
skew	-1.7878	-0.9414	-0.8094	-0.7112	-0.7481	0.2979	-0.4053	-0.9359	-0.0879	0.2875	-1.0798	-0.3568	-0.0945	-0.5225
kurt	16.9434	5.0628	3.8278	4.0287	7.5853	4.42	8.7499	9.6284	3.4917	3.3669	7.1512	5.015	2.8169	6.9032
ES99 %	0.2541	0.1526	0.1399	0.126	0.1235	0.1988	0.1379	0.1212	0.1312	0.1595	0.0964	0.1143	0.1252	0.096
ES95 %	0.1685	0.1028	0.0914	0.0846	0.0833	0.1239	0.0817	0.0782	0.0903	0.1087	0.0618	0.0743	0.0932	0.073
ES5 %	-0.1848	-0.1431	-0.1233	-0.1054	-0.0907	-0.1126	-0.0859	-0.0781	-0.0896	-0.0937	-0.0833	-0.078	-0.0904	-0.0672
ES1 %	-0.3718	-0.2283	-0.2036	-0.1791	-0.176	-0.1684	-0.1666	-0.1578	-0.1454	-0.144	-0.1429	-0.1406	-0.1373	-0.1286

Cuadro 4.3. Resumen de estadísticas de la muestra de 150 retornos logarítmicos diarios correspondiente al periodo de prueba. Los valores 1 %, 5 %, 95 % y 99 % corresponden a los respectivos percentiles mientras que los valores de $ES1\,\%$ y similares corresponden al Expected Shortfall al nivel de confianza dado. Estos se obtienen desde la distribución empírica de los datos obtenidos.

Generando los resultados para las diferentes estrategias de arbitraje fuera de la muestra de entrenamiento, se consideran 150 días de trading:

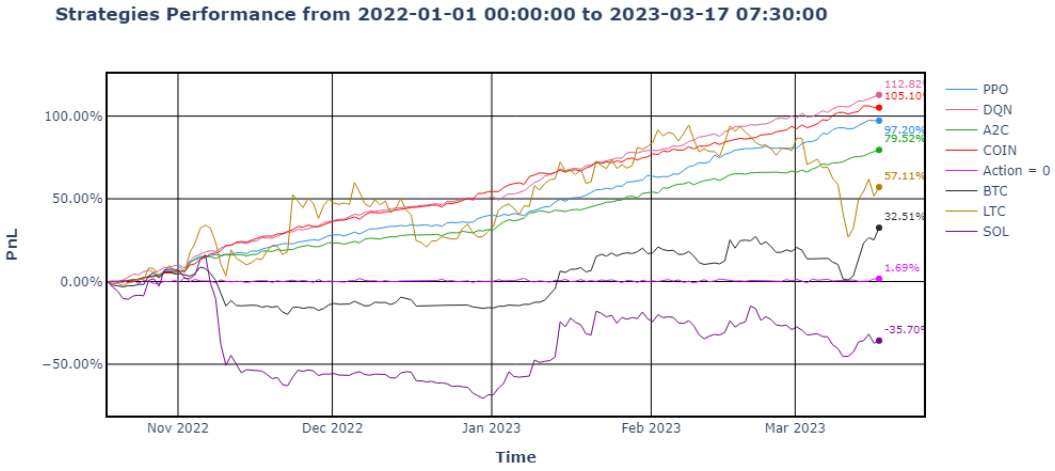

Figura 4.5. Gráfico de *Profit & Loss* para las diferentes estrategias utilizadas. Se tienne 3 agentes de trading diferente, utilizando métodos de entrenamiento: PPO, DQN y A2C. La línea magenta de Action = 0, corresponde a una estrategia generando el protafolio de cointegración y no realizar ninguna acción, únicamente tomar posición larga y otra corta en los dos protafolios sin operar intradía. Si bien una estrategia de *Buy & Hold* de BTC no es comparable con las estrategias nuetrales, se presenta a modo de comparación y como indicador de rendimiento del mercado. Adicionalmente se muestra Litecoin siendo el con mejor rendimiento del periodo y Solana con el peor rendimiento del periodo.

Las estrategias generadas a partir de los agentes PPO, A2C y DQN corresponden a direcciones de posiciones del los portafolios siendo cambiadas o mantenidas a intervalos de 30 minutos durante 1 día de operación. La estrategia COIN corresponde a la estrategia base de arbitraje desde la cual se espera mejorar con el aprendizaje reforzado profundo. La estrategia denominada 'Action = 0' corresponde a generar un portafolio de arbitraje y mantener una posición larga y otra corta sin cambios durante un día, esto corresponde a una referencia considerando que no se generan acciones para maximizar retorno, se aprecia que en comparación con las demás estrategias, al ser una posición neutral mantiene su propósito sin ser perturbada, o en una medida muy baja, por movimientos de mercado. Esto muestra que los portafolios generados pueden mantener una posición neutral en el tiempo, siempre y cuando estos sean rebalanceados cada día.

En términos de las métricas de riesgo y rendimiento:

	PPO	DQN	A2C	COIN	Action = 0	BTC	LTC	SOL
mean	0.0045	0.005	0.0039	0.0048	0.0001	0.0019	0.003	-0.0029
std	0.0081	0.0076	0.0076	0.0074	0.0083	0.0277	0.0459	0.0719
min	-0.0238	-0.0156	-0.0101	-0.0106	-0.0243	-0.0851	-0.1192	-0.3712
1 %	-0.0096	-0.0132	-0.0094	-0.0083	-0.0182	-0.0826	-0.1163	-0.1958
5 %	-0.0055	-0.0069	-0.0063	-0.0066	-0.0127	-0.0395	-0.065	-0.0994
50 %	0.0036	0.0047	0.0033	0.0043	0.0002	0.0008	0.0012	-0.0045
95 %	0.0165	0.0173	0.0169	0.0164	0.0131	0.0455	0.0724	0.1096
99 %	0.0233	0.0212	0.0218	0.0276	0.0199	0.1017	0.1339	0.1984
max	0.0531	0.0404	0.05	0.0365	0.027	0.1072	0.2554	0.3327
skew	1.3872	0.5246	1.7443	0.9152	0.1308	0.6646	1.0851	-0.031
kurt	8.6045	2.6451	8.224	2.2967	0.4272	4.1571	6.4271	7.7479
ES99 %	0.0387	0.0317	0.036	0.033	0.0247	0.1059	0.1985	0.2684
ES95 %	0.0238	0.0217	0.0236	0.0237	0.0178	0.0781	0.1186	0.1736
ES5 %	-0.0096	-0.0104	-0.0079	-0.0079	-0.0167	-0.0579	-0.0931	-0.1632
ES1 %	-0.017	-0.0148	-0.01	-0.0096	-0.0217	-0.0843	-0.1192	-0.3002

Cuadro 4.4. Resumen de estadísticas de retornos logarítmicos diarios de las diferentes estrategias considerando además 3 activos de particular interés.

Considerando las métricas de rendimiento y riesgo:

	Max D.	ES 5 %	VaR 5 %	σ	AS
PPO	2.35 %	-0.964 %	-0.555 %	0.806 %	0.00444
DQN	1.55 %	-1.039 %	-0.695 %	0.758 %	0.00388
A2C	1.08 %	-0.788 %	-0.635 %	0.760 %	0.00375
COIN	1.06 %	-0.787 %	-0.656 %	0.739 %	0.00386
A = 0	3.22 %	-1.674 %	-1.267 %	0.834 %	0.00475
BTC	26.30 %	-5.792 %	-3.948 %	2.775 %	0.01558
LTC	34.82 %	-9.312 %	-6.504 %	4.591 %	0.02902
SOL	74.78 %	-16.316 %	-9.940 %	7.189 %	0.03629

Cuadro 4.5. Resumen de métricas de riesgo para los diferentes agentes. Adicionalmente se muestran a modo de referencia de mercado los tres activos BTC, LTC y SOL.

	Calmar	Sharpe	EPM	R[%]
PPO	0.192	0.561	0.816	97.20
DQN	0.325	0.664	1.220	112.82
A2C	0.363	0.513	0.870	79.52
COIN	0.454	0.648	1.102	105.10
A = 0	0.003	0.013	0.051	1.69
BTC	0.007	0.068	0.048	32.51
LTC	0.009	0.066	0.040	57.11
SOL	-0.003	-0.040	-0.124	-35.70

Cuadro 4.6. Resumen de métricas de rendimiento para retornos diarios de las diferentes estrategias y activos. La estrategia con mejor rendimiento en la métrica se colorea verde mientras la peor con rojo.

En términos de las medidas de riesgo, durante el periodo de prueba la estrategia base COIN consigue tener el menor riesgo medido por Drawdown Máximo, Expecter Shortfall al 5 % y en desviación estándar. En cuanto al agente PPO es el más riesgoso en 3 métricas, mientras que el agente DQN es el segundo más riesgoso en términos de pérdidas extremas medidas por Expected Shortfall y Value at Risk. Sin embargo, en comparación al mercado, considerando el activo de mejor rendimiento en el periodo (LTC), el de pero rendimiento (SOL) y un activo más consolidado siendo el benchmark de mercado (BTC), todas las estrategias presentan una mejora sustancial en términos de reducir el riesgo. Si bien las estrategias consideran la mitad del capital en posición larga y la otra mitad en posición corta, mientras que las estrategias de *Buy & Hold* de mercado tienen una posición larga con todo el capital, no son directamente comparables, sin embargo, se obtienen estrategias con un riesgo muy inferior al mercado. Considerando la estrategia neutral (A = 0), se obtienen riesgos similares a las estrategias activas, lo cual muestra la efectividad de la estrategia de confección de portafolios.

En términos de medidas de rendimiento, al considerar el retorno obtenido ajustado por riesgo en base a diferentes métricas, la estrategia generada por el agente DQN resulta ser la mejor durante el periodo en términos del riesgo y retorno generado. Al tener todos riesgos similares, el retorno acumulado resulta ser el determinante.

4.2.2. Rendimiento intra día

Si bien se tienen los resultados a nivel diario, estos surgen de una trayectoria de 47 decisiones en base a diferentes criterios. Adicionalmente, se estudia también los riesgos intra día y el comportamiento de las estrategias a lo largo de su secuencia de decisiones para determinar la eficiencia de las posiciones dólar neutral y si los agentes logran generar una diferencia importante en términos del riesgo intra día en comparación a la estrategia benchmark COIN.

Los resultados para movimientos intra día de las estrategias mantienen la misma tendencia que los resultados agregados a 1 día. En primer lugar, todas las medias son positivas e incluso sus medianas (percentil 50 %) son positivos a diferencia de la estrategia neutra y los activos de comparación. Como es de esperar los retornos mínimos y máximos son más acotados respecto a los activos de referencia. Por otro lado, al ver las correlaciones de las estrategias, se tiene que todas las estrategias incluida la estrategia neutra no presentan correlación importante con los activos de referencia. Esto refuerza el hecho de que la construcción de los portafolios es correcta y presentan un beta muy cercano a 1, por lo que al tomar posiciones opuestas se obtiene un beta nulo.

	PPO[%]	DQN[%]	A2C[%]	COIN[%]	A=0[%]	BTC[%]	LTC[%]	SOL[%]
mean	0.0093	0.0104	0.0081	0.0099	-0.0002	0.004	0.0064	-0.0059
std	0.125	0.1249	0.1251	0.125	0.1254	0.3829	0.6473	1.0169
min	-1.152	-1.152	-1.152	-1.152	-1.3438	-4.99	-9.8114	-13.9475
1 %	-0.253	-0.2453	-0.2533	-0.2453	-0.3011	-1.0746	-1.753	-2.8196
5 %	-0.133	-0.1293	-0.1342	-0.132	-0.1466	-0.45	-0.8404	-1.2111
50 %	0.0015	0.0019	0.0003	0.0023	0.0	-0.0017	0.0	0.0
95 %	0.1527	0.1575	0.1515	0.1553	0.1378	0.4582	0.8344	1.1842
99 %	0.4094	0.4072	0.4094	0.4152	0.3273	1.1913	1.7591	2.9252
max	2.5711	2.5711	2.5711	2.5711	2.5711	5.9289	10.6266	14.5611
ES99 %	0.7872	0.7875	0.7977	0.7991	0.6512	1.9508	3.0154	5.0321
ES95 %	0.3351	0.3402	0.3358	0.3415	0.2874	0.9309	1.5155	2.3795
ES5 %	-0.2173	-0.2108	-0.2171	-0.2104	-0.2661	-0.8848	-1.4466	-2.299
ES1 %	-0.3798	-0.3724	-0.3714	-0.3599	-0.5477	-1.7507	-2.6744	-4.6847
skew	3.9495	4.0539	4.1584	4.1946	1.7761	0.7441	0.7746	0.2941
kurt	53.6798	53.6576	53.6325	53.6066	54.2627	35.6178	39.1638	39.5361

Cuadro 4.7. Estadísticas de resumen para retornos logarítmicos intra día, a frecuencia de 30 minutos. Resultado se presentan anivel procentual

	PPO	DQN	A2C	COIN	A=0	BTC	LTC	SOL
PPO	1.0	0.65	0.6493	0.7208	0.2874	0.0112	0.0465	0.0065
DQN	0.65	1.0	0.7121	0.7519	0.1248	0.0035	0.0355	0.0036
A2C	0.6493	0.7121	1.0	0.7354	-0.0385	0.0127	0.0503	0.0159
COIN	0.7208	0.7519	0.7354	1.0	0.1861	-0.0009	0.0431	-0.0085
A=0	0.2874	0.1248	-0.0385	0.1861	1.0	-0.0009	0.0361	-0.0105
BTC	0.0112	0.0035	0.0127	-0.0009	-0.0009	1.0	0.6737	0.699
LTC	0.0465	0.0355	0.0503	0.0431	0.0361	0.6737	1.0	0.5902
SOL	0.0065	0.0036	0.0159	-0.0085	-0.0105	0.699	0.5902	1.0

Cuadro 4.8. Matriz de correlación de retornos logarítmicos intra día.

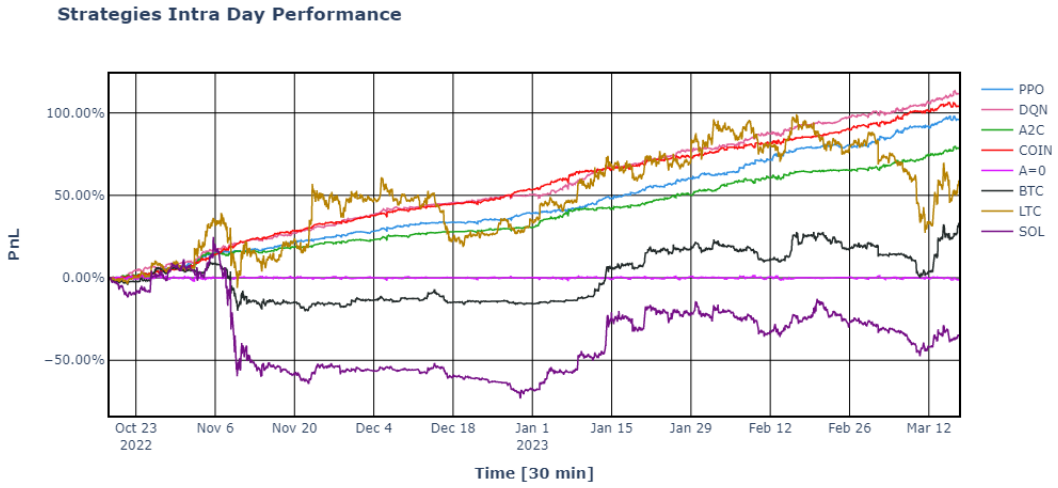

Figura 4.6. Gráfico de *Profit & Loss* de las estrategias a nivel intra día, con una frecuencia de 30 minutos. Se presenta la estrategia neutra A=0, las estrategias realizadas por agentes DRL; PPO, DQN y A2C, junto con la estrategia de comparación COIN.

	Max D.	ES 5 %	VaR 5 %	σ	AS
PPO	2.526 %	-0.2173 %	-0.1330 %	0.1250 %	0.00092
DQN	2.235 %	-0.2108 %	-0.1293 %	0.1249 %	0.00092
A2C	2.421 %	-0.2171 %	-0.1342 %	0.1251 %	0.00091
COIN	1.983 %	-0.2104 %	-0.1320 %	0.1250 %	0.00091
A = 0	4.035 %	-0.2661 %	-0.1466 %	0.1254 %	0.00091
BTC	27.023 %	-0.8848 %	-0.4500 %	0.3829 %	0.00215
LTC	36.117 %	-1.4466 %	-0.8404 %	0.6473 %	0.02139
SOL	78.471 %	-2.2990 %	-1.2111 %	1.0169 %	0.08455

Cuadro 4.9. Resumen de métricas de riesgo para los diferentes agentes para retornos intra día. Las estrategias con menor riesgo en la métrica de comparación se colorean verde mientras los con mayor riesgo en rojo. Adicionalmente se muestran a modo de referencia de mercado los tres activos BTC, LTC y SOL.

	Calmar	Sharpe	EPM
PPO	0.0037	0.0747	0.01646
DQN	0.0047	0.0833	0.02049
A2C	0.0033	0.0644	0.00306
COIN	0.0050	0.0792	0.02547
A = 0	0.0000	-0.0013	0.00000
BTC	0.0001	0.0104	-0.00778
LTC	0.0002	0.0098	0.00303
SOL	-0.0001	-0.0058	-0.00065

Cuadro 4.10. Resumen de métricas de rendimiento para retornos intra día de las diferentes estrategias y activos. La estrategia con mejor rendimiento en la métrica se colorea verde mientras la peor con rojo.

En comparación a los resultados diarios, respecto a las métricas de riesgo seleccionadas, los agentes menos riesgosos corresponden a DQN y la estrategia bnechmark COIN, mientras que los agentes PPO y A2C resultan ser mas riesgosos. Sin embargo, los resultados muestran que no solo a nivel diario agregado, sino que también a nivel intra día el riesgo se reduce sustancialmente, siendo alrededor de un tercio del riesgo de Bitcoin y también la eficiencia del portafolio de arbitraje se mantiene al controlar en la estrategia A=0.

En términos de sus métricas de rendimiento, las estrategias DQN y COIN siguen siendo las con mejor rendimiento, DQN bajo el índice de Sharpe mientras que COIN en Calmar y EPM. En este caso, el agente A2C resulta ser el de peor rendimiento en todas las métricas.

4.2.3. Rendimiento bajo costos de transacción

Midiendo el rendimiento de las estrategias bajo costos de transacción, considerando que la estrategia aplica durante un día posiblemente reestructurando la posición varias veces durante el día por un máximo de 47 veces en un día. Si bien los agentes no fueron entrenados considerando costos de transacción se estudia el impacto de estos en las estrategias de alta frecuencia.

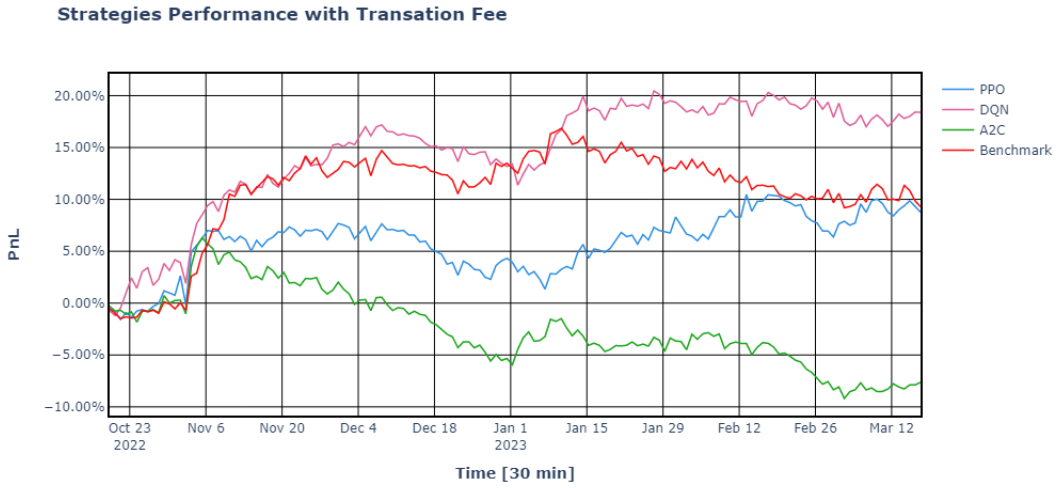

Figura 4.7. Resultados de estrategias al aplicar un costo de transacción del 0.02 % por operación.

	PPO[%]	DQN[%]	A2C[%]	COIN[%]
mean	0.0607	0.1182	-0.0514	0.063
std	0.8217	0.7715	0.7662	0.7561
min	-2.789	-1.9423	-1.4623	-1.4953
1 %	-1.3883	-1.7357	-1.3862	-1.2577
5 %	-0.9963	-1.0925	-1.083	-1.1079
50 %	-0.0567	0.1079	-0.0891	0.016
95 %	1.3074	1.3573	1.2455	1.2842
99 %	1.91	1.7353	1.7663	2.402
max	4.998	3.6171	4.5487	3.2512
ES99 %	3.4853	2.7592	3.1797	2.897
ES95 %	2.025	1.7898	1.9301	1.9846
ES5 %	-1.3952	-1.4484	-1.2346	-1.2236
ES1 %	-2.1312	-1.8741	-1.4553	-1.3783
skew	1.3834	0.4656	1.7125	0.8922
kurt	8.4942	2.2687	7.9248	2.1252

Cuadro 4.11. Estadísticas de resumen para estategia diaria considerando costos de transacción.

	Max D.	ES 5 %	VaR 5 %	σ	AS
PPO	5.87[%]	-1.39[%]	-0.99[%]	0.82[%]	0.0036
DQN	4.96[%]	-1.44[%]	-1.09[%]	0.77[%]	0.0030
A2C	14.57[%]	-1.23[%]	-1.08[%]	0.77[%]	0.0032
COIN	6.56[%]	-1.22[%]	-1.10[%]	0.76[%]	0.0037

Cuadro 4.12. Resumen de métricas de riesgo para las diferentes estrategias considerando costos de transacción.

	Calmar	Sharpe	EPM	R[%]
PPO	0.0103	0.0738	0.1558	8.77
DQN	0.02385	0.1533	0.4004	18.39
A2C	-0.0035	-0.0670	-0.1430	-7.63
COIN	0.0096	0.0834	0.1597	9.29

Cuadro 4.13. Resumen de métricas de rendimiento para estrategias considerando costos de transacción.

En comparación a las estrategias sin considerar costos de transacción, esta ves el agente entrenado por DQN logra superar en todas las métricas de rendimiento a la estrategia benchmark COIN. Respecto a las medidas de riesgo, el agente DQN logra ser el mejor en 3 métricas de riesgo y el peor en una. Respecto a los riesgos de cola o pérdidas máximas generadas las métricas de VaR y ES muestran que DQN se expone a mayores pérdidas que las demás estrategias, sin embargo, en cuanto a VaR las diferencias de todas las estrategias son mínimas. En cuanto a volatilidad incondicional también todos tienen resultados prácticamente idénticos salvo PPO con un 0.05 % más volátil. En cuanto al máximo Drawdown, la estrategia por DQN resulta tener la pérdida máxima acumulada más baja de la muestra, mientras que A2C resulta ser el con mayor pérdida. Esta medida muestra que si bien todas tienen niveles similares de volatilidad y riesgos de cola, existen diferencias respecto a su gestión y capacidad de generar rentabilidad controlando pérdidas más allá de una operación en particular.

En términos de la medida de riesgo que contempla diferentes momentos de la distribución incondicional de los retornos de Aumann Serrano, se obtiene que DQN permanece siendo la estrategia menos riesgosa, pues considerando que todas presentan volatilidad similar y pérdidas extremas similares, esta medida confirma lo que se observa en el Drawdown máximo.

4.2.4. Actividad del Agente

Si bien se midió la capacidad de generación de retornos de las diferentes estrategias y el nivel de riesgo que exponen junto con su retorno ajustado por riesgo, no se mide la capacidad de predicción de los agente respecto a movimientos de mercado más allá de si el retorno es positivo o negativo.

Para el nivel de actividad total, se realiza el test de Cobertura Incondicional (UC) Kupiec (1995)[33] , donde se testea si la acción es aleatoria bajo la hipótesis nula de que la acción se realiza de forma aleatoria equiprobable:

Bajo la hipótesis de una correcta especificación del modelo, la sucesión de acciones debe

ser independiente y seguir una distribución Bernoulli.

$$H_0 : A_t \sim Bernoulli(\alpha), \tag{4.2.1}$$

$$f(A_t, p) = (1-p)^{1-A_t} p^{A_t} \tag{4.2.2}$$

Para $\{A_t\}_t$ anterior,

$$L(\pi) = \prod (1-\pi)^{1-I_{t+1}} \pi^{i_{i+1}} = (1-\pi)^{T_0} \pi^{T_1} \tag{4.2.3}$$

se tiene que

$$\hat{\pi}_{MLE} = \frac{T_1}{T_0 + T_1} \tag{4.2.4}$$

Bajo $H0 : \pi = \alpha$

$$LR_{uc} = -2 \ln \left[\frac{L(\alpha)}{L(\hat{\pi})} \right] \sim \chi^2_1, \tag{4.2.5}$$

por lo tanto, se busca rechazar la hipótesis nula y determinar que el agente no realiza acciones de manera aleatoria.

Además, considerando la distribución condicional de la acciones se tiene el test de cobertura condicional (CC) Christoffsen (1998) [8].

Considerando que $\{A_t\}_t$ presenta una dependencia temporal y sigue una sucesión de Markov con matriz de probabilidad de transición:

$$\Pi = \begin{pmatrix} 1-\pi_{01} & \pi_{01} \\ 1-\pi_{11} & \pi_{11} \end{pmatrix}, \tag{4.2.6}$$

donde π_{01} es la probabilidad de transición de 0 a 1, con verosimiltud:

$$L(\Pi) = (1-\pi_{01})^{T_{00}} \pi_{01}^{T_{01}} (1-\pi_{11})^{T_{10}} \pi_{11}^{T_{11}}, \tag{4.2.7}$$

Si la $\{A_t\}_t$ es independiente del tiempo, se tiene que $\pi_{01} = \pi_{11} = \pi$, entonces:

$$\hat{\Pi}_{MLE} = \begin{pmatrix} 1-\hat{\pi} & \hat{\pi} \\ 1-\hat{\pi} & \hat{\pi} \end{pmatrix}, \quad LR_{cc} = -2 \ln \left[\frac{L(\alpha)}{L(\hat{\Pi})} \right] \sim \chi^2_1 \tag{4.2.8}$$

por lo tanto se busca rechazar la hipótesis nula de que las acciones son aleatorias e independientes entre sí.

Agent	State 1 ($\hat{\pi}$)	State 0	UC	CC
PPO	42.44[%]	57.56[%]	<1e6	<1e6
DQN	47.79[%]	52.21[%]	<1e6	<1e6
A2C	70.27[%]	29.73[%]	<1e6	<1e6
COIN	48.31[%]	51.69[%]	<1e6	<1e6

Cuadro 4.14. Cantidad relativa de acciones totales en cada estado, sobre un total de 47 acciones realizadas por día por 150 días generando en total 7050 acciones realizadas.

Agent	$\hat{\pi}_{0,0}$	$\hat{\pi}_{0,1}$	$\hat{\pi}_{1,0}$	$\hat{\pi}_{1,1}$
PPO	55.07[%]	44.93[%]	60.92[%]	39.07[%]
DQN	50.57[%]	49.43[%]	54.02[%]	45.98[%]
A2C	31.74[%]	68.26[%]	28.89[%]	71.11[%]
COIN	48.09[%]	51.91[%]	55.52[%]	44.48[%]

Cuadro 4.15. MLE de Probabilidades de transición para las acciones realizadas por cada agente.

de este modo, se descarta el hecho de que las acciones realizadas por los diferentes agentes son aleatorias equiprobables.

En términos de la capacidad predictiva sobre la dirección de movimiento del precio del activo sintético, se estudia si las acciones predicen dicho movimiento.

Considerando el retorno dado la acción a_t en tiempo t , se determina que el agente predijo correctamente la dirección de movimiento de precios del activo sintético si :

- **Verdadero Positivo :**

  Se determina que el retorno siguiente será positivo por lo tanto se mantiene la posición actual ($a_{t-1} = a_t$) y el retorno realizado dado que no hubo cambio de dirección fue positivo (La acción genera un retorno positivo).

- **Falso Negativo :**

  Se determina que el retorno siguiente será negativo, por lo tanto se cambia la dirección de la posición ($a_{t-1} \neq a_t$) y el retorno realizado dado el cambio de dirección resulta negativo (la acción genera un retorno negativo).

- **Falso Positivo :**

  Se determina que el retorno siguiente será positivo y se mantiene la dirección ($a_{t-1} = a_t$), resultando en un retorno negativo (la acción genera un retorno negativo).

- **Verdadero Negativo :**

  Se determina que el retorno siguiente será negativo por lo tanto se invierte la dirección de la posición ($a_{t-1} \neq a_t$) y en consecuencia el retorno es positivo (la acción genera un retorno positivo).

$$\text{Verdadero Positivo (TP): } a_{t-1} = a_t \ \& \ r_t > 0 \tag{4.2.9}$$

$$\text{Falso Negativo (FN): } a_{t-1} \neq a_t \ \& \ r_t < 0 \tag{4.2.10}$$

$$\text{Falso Positivo (FP): } a_{t-1} = a_t \ \& \ r_t < 0 \tag{4.2.11}$$

$$\text{Verdadero Negativo (TN): } a_{t-1} \neq a_t \ \& \ r_t > 0 \tag{4.2.12}$$

entonces las métricas usuales de rendimiento respecto a estos valores:

- **Accuracy :**

  Corresponde al total de aciertos respecto al total de casos.

$$\text{Accuracy } = \frac{TP + TN}{TP + FP + FN + TN} \tag{4.2.13}$$

- **Precision :**

  Corresponde a la proporción de las predicciones positivas que resultan ser positivas.

$$\text{Precision } = \frac{TP}{TP + FP} \tag{4.2.14}$$

- **Recall :**

  Corresponde a los verdaderos positivos sobre el total de aciertos.

$$\text{Recall } = \frac{TP}{TP + FN} \tag{4.2.15}$$

Se debe considerar el significado de cada métrica para el efecto del problema estudiado. Los falsos negativos (FN) y verdaderos negativos (TN) tienen un particular elemento dado que estos se generan a partir del cambio de dirección de la posición, por lo tanto, los FN conllevan a una pérdida por movimiento de mercado y además por costo de transacción, mientras que los TN generan un retorno positivo por movimiento de mercado y una pérdida por costo de transacción al cerrar la posición actual y abrirla en la otra dirección mientras que los TP y FP no generan impacto en resultado por costos de transacción.

De este modo, la métrica de Precision genera resultados por movimientos de mercado sin costos de transacción. La métrica de Accuracy contempla el total de aciertos generando resultados a partir de movimientos de mercado y costos de transacción mientras que Recall corresponde a la proporción de resultados positivos por movimientos de mercado sin costos de transacción respecto a estos mismos mas retornos positivos generados por movimiento de mercado en la dirección contraria junto con el costo de transacción asociado.

Agent	TP	FN	FP	TN	Recall	Accuracy	Presicion
PPO	3036	468	2897	499	86.64 %	51.23 %	51.17 %
DQN	2964	473	2881	582	86.24 %	51.39 %	50.71 %
A2C	3355	132	3281	132	96.21 %	50.54 %	50.56 %
COIN	3212	293	3069	326	91.64 %	51.28 %	51.14 %

Cuadro 4.16. Resultados de predicción de movimiento considerando diferentes métrica de TP,FP,TN y FN junto con Recall, Precision y Accuracy. Para el total de 47 acciones diarias se consideran 46 descartando la primera dado que el estado inicial siempre es igual.

En términos de estas métricas los resultados son mixtos. Respecto al Recall, el agente A2C resulta ser el mejor, siendo el con mejor sin embargo también es el peor en las métricas de Accuracy y Presicion, por lo que al ver sus probabilidades de transición se observa que es el menos propenso a cambiar de estado 1-1, y el más propenso a cambiar de 0-1, permaneciendo muy estático en términos de su decisión sin generar predicciones agregadas de mercado consistentes lo cual impacta en su resultado siendo el con peor rendimiento en prácticamente todos los ámbitos. DQN resulta ser el con mejor Accuracy, esto es, el que mejor predice los movimientos de mercado siendo la métrica más importante al momento de generar retornos positivos. Por otro lado es el peor en Recall.

Finalmente, la estrategia COIN resulta ser la más sólida en el global de estas métricas, nunca es le mejor pero tampoco el peor. En general, se aprecia que las estrategias PPO, DQN y COIN generan acciones con una estructura similar lo que se puede observar en las probabilidades estacionarias de los estados y sus probabilidades de transición, junto con sus métricas de predictibilidad direccional similares. Sin embargo, obteniendo la correlación entre las trayectorias de las acciones:

	PPO	DQN	A2C	COIN
PPO	1	0.45	0.50	0.49
DQN	-	1	0.47	0.56
A2C	-	-	1	0.56
COIN	-	-	-	1

Cuadro 4.17. matriz de correlación de las trayectorias de las acciones de las diferentes estrategias.

pese a tener un comportamiento similar, su correlación es positiva y en general en torno al 50 % por lo que no se podría determinar que generan las mismas trayectorias.

Capítulo 5

CONCLUSIONES Y TRABAJOS FUTUROS

En este documento se presenta un método de trading para generar utilidades en el mercado de criptomonedas basado en el aprendizaje reforzado profundo utilizando un esquema unificado de generación de portafolios de arbitraje, descomposición y extracción de señales de mercado y de toma de decisiones para seleccionar posiciones sobre los activos sintéticos generados. El método propuesto se implementa en un esquema de aprendizaje reforzado profundo utilizando 3 tipos de agentes (algoritmos) diferentes para la toma de decisiones; PPO, DQN y A2C.

Los portafolios son generados a partir de la relación de cointegración de los diferentes activos, posteriormente se extraen señales desde el mercado utilizando diferentes indicadores técnicos y transformaciones los cuales definen los estados del mercado para el esquema de DRL. Finalmente se genera un simulador de escenarios de mercado utilizando una amplia ventana histórica para el entrenamiento de la red en donde se fija una ventana de 7 días para la generación de portafolios y de 1 día (47 decisiones) en la cual el agente debe tomar las acciones respectivas para generar utilidad.

Se realiza un exhaustivo análisis fuera de muestra, tanto a nivel diario sobre el resultado agregado de la trayectoria de decisiones del agente a lo largo de una sesión de un día como a nivel intra día considerando cada decisión tomada, sobre el riesgo de las estrategias generadas utilizando varias medidas de riesgo. Además se estudia el rendimiento ajustado por riesgo en diferentes medidas. El análisis se realiza sin y con costos de transacción, junto con un estudio de las acciones realizadas por los agentes para ver si efectivamente hay un comportamiento coherente y no aleatorio.

Según las pruebas realizadas, el método propuesto es capaz de generar utilidad y reducir el riesgo en un ambiente de extrema volatilidad como lo es el mercado de criptomonedas y se mantiene rentable bajo costos de transacción. Adicionalmente, los resultados de la actividad del agente muestran que las acciones generadas no son aleatorias y estas presentan una mejora sustancial (DQN) a la estrategia de arbitraje base COIN.

Si bien los resultados de la metodología propuesta muestran ser rentables bajo costos de transacción, no se profundiza en la estructuración de la red utilizada y tampoco considera los costos de transacción que su entrenamiento como elemento fundamenta de una estrategia de trading de alta frecuencia.

Bibliografía

[1] Robert J Aumann y Roberto Serrano. "An economic index of riskiness". En: *Journal of Political Economy* 116.5 (2008), págs. 810-836.

[2] Marco Avellaneda y Jeong-Hyun Lee. "Statistical arbitrage in the US equities market". En: *Quantitative Finance* 10.7 (2010), págs. 761-782.

[3] RE Bellman y S Dreyfus. "Applied Dynamic Programming, Princeton Unlv". En: *Press, Princeton, NJ* (1962).

[4] Hee Rak Beom y Hyung Suck Cho. "A sensor-based navigation for a mobile robot using fuzzy logic and reinforcement learning". En: *IEEE transactions on Systems, Man, and Cybernetics* 25.3 (1995), págs. 464-477.

[5] João Carapuço, Rui Neves y Nuno Horta. "Reinforcement learning applied to Forex trading". En: *Applied Soft Computing* 73 (2018), págs. 783-794.

[6] Huafeng Chen et al. "Empirical investigation of an equity pairs trading strategy". En: *Management Science* 65.1 (2019), págs. 370-389.

[7] Xixin Cheng, Philip LH Yu y Wai Keung Li. "Basket trading under co-integration with the logistic mixture autoregressive model". En: *Quantitative Finance* 11.9 (2011), págs. 1407-1419.

[8] Peter F Christoffersen. "Evaluating interval forecasts". En: *International economic review* (1998), págs. 841-862.

[9] Matthew Clegg y Christopher Krauss. "Pairs trading with partial cointegration". En: *Quantitative Finance* 18.1 (2018), págs. 121-138.

[10] Quang-Vinh Dang. "Reinforcement learning in stock trading". En: *International conference on computer science, applied mathematics and applications*. Springer. 2019, págs. 311-322.

[11] Yue Deng et al. "Deep direct reinforcement learning for financial signal representation and trading". En: *IEEE transactions on neural networks and learning systems* 28.3 (2016), págs. 653-664.

[12] Binh Do y Robert Faff. "Are pairs trading profits robust to trading costs?" En: *Journal of Financial Research* 35.2 (2012), págs. 261-287.

[13] Binh Do y Robert Faff. "Does simple pairs trading still work?" En: *Financial Analysts Journal* 66.4 (2010), págs. 83-95.

[14] James Durbin y Siem Jan Koopman. *Time series analysis by state space methods*. Vol. 38. OUP Oxford, 2012.

[15] Robert F Engle y Clive WJ Granger. "Co-integration and error correction: representation, estimation, and testing". En: *Econometrica: journal of the Econometric Society* (1987), págs. 251-276.

[16] Thomas G Fischer. *Reinforcement learning in financial markets-a survey*. Inf. téc. FAU Discussion Papers in Economics, 2018.

[17] Vincent François-Lavet et al. "An introduction to deep reinforcement learning". En: *Foundations and Trends® in Machine Learning* 11.3-4 (2018), págs. 219-354.

[18] Alexander Galenko, Elmira Popova e Ivilina Popova. "Trading in the presence of cointegration". En: *The Journal of Alternative Investments* 15.1 (2012), págs. 85-97.

[19] Prakhar Ganesh y Puneet Rakheja. "Deep Reinforcement Learning in High Frequency Trading". En: *CoRR* abs/1809.01506 (2018). arXiv: 1809.01506. URL: http://arxiv.org/abs/1809.01506.

[20] Evan Gatev, William N Goetzmann y K Geert Rouwenhorst. "Pairs trading: Performance of a relative-value arbitrage rule". En: *The Review of Financial Studies* 19.3 (2006), págs. 797-827.

[21] Geoffrey J Gordon. "Stable fitted reinforcement learning". En: *Advances in neural information processing systems* 8 (1995).

[22] Audrunas Gruslys et al. "The reactor: A fast and sample-efficient actor-critic agent for reinforcement learning". En: *arXiv preprint arXiv:1704.04651* (2017).

[23] Jorge Guijarro-Ordonez, Markus Pelger y Greg Zanotti. "Deep learning statistical arbitrage". En: *arXiv preprint arXiv:2106.04028* (2021).

[24] H. V. Hasselt. "Double Q-learning". En: *Advances in Neural Information Processing Systems* (2010), págs. 2613-2621.

[25] Ulrich Homm y Christian Pigorsch. "Beyond the Sharpe ratio: An application of the Aumann–Serrano index to performance measurement". En: *Journal of Banking & Finance* 36 (ago. de 2012), págs. 2274-2284. DOI: 10.1016/j.jbankfin.2012.04.005.

[26] Huy D Huynh, L Minh Dang y Duc Duong. "A new model for stock price movements prediction using deep neural network". En: *Proceedings of the 8th International Symposium on Information and Communication Technology*. 2017, págs. 57-62.

[27] Søren Johansen. "Cointegration in partial systems and the efficiency of single-equation analysis". En: *Journal of econometrics* 52.3 (1992), págs. 389-402.

[28] Søren Johansen. "Estimation and hypothesis testing of cointegration vectors in Gaussian vector autoregressive models". En: *Econometrica: journal of the Econometric Society* (1991), págs. 1551-1580.

[29] Taewook Kim y Ha Young Kim. "Optimizing the pairs-trading strategy using deep reinforcement learning with trading and stop-loss boundaries". En: *Complexity* 2019 (2019), págs. 1-20.

[30] Vijay Konda y John Tsitsiklis. "Actor-critic algorithms". En: *Advances in neural information processing systems* 12 (1999).

[31] Christopher Krauss. "Statistical arbitrage pairs trading strategies: Review and outlook". En: *Journal of Economic Surveys* 31.2 (2017), págs. 513-545.

[32] Mahinda Mailagaha Kumbure et al. "Machine learning techniques and data for stock market forecasting: A literature review". En: *Expert Systems with Applications* (2022), pág. 116659.

[33] Paul H Kupiec et al. *Techniques for verifying the accuracy of risk measurement models*. Vol. 95. 24. Division of Research y Statistics, Division of Monetary Affairs, Federal . . ., 1995.

[34] Mao Liang Li, Chin Man Chui y Chang Qing Li. "Is pairs trading profitable on China AH-share markets?" En: *Applied Economics Letters* 21.16 (2014), págs. 1116-1121.

[35] Tiantian Li et al. "Aumann–Serrano index of risk in portfolio optimization". En: *Mathematical Methods of Operations Research* 94.2 (2021), págs. 197-217.

[36] Timothy P Lillicrap et al. "Continuous control with deep reinforcement learning". En: *arXiv preprint arXiv:1509.02971* (2015).

[37] Long-Ji Lin. "Self-improving reactive agents based on reinforcement learning, planning and teaching". En: *Machine learning* 8 (1992), págs. 293-321.

[38] Yan-Xia Lin, Michael McCrae y Chandra Gulati. "Loss protection in pairs trading through minimum profit bounds: A cointegration approach". En: *Advances in Decision Sciences* 2006 (2006).

[39] Volodymyr Mnih et al. "Asynchronous methods for deep reinforcement learning". En: *International conference on machine learning*. PMLR. 2016, págs. 1928-1937.

[40] Volodymyr Mnih et al. "Human-level control through deep reinforcement learning". En: *nature* 518.7540 (2015), págs. 529-533.

[41] Volodymyr Mnih et al. "Playing atari with deep reinforcement learning". En: *arXiv preprint arXiv:1312.5602* (2013).

[42] John E Moody et al. "Reinforcement Learning for Trading Systems and Portfolios." En: *KDD*. 1998, págs. 279-283.

[43] John M Mulvey et al. "Optimizing a portfolio of mean-reverting assets with transaction costs via a feedforward neural network". En: *Quantitative Finance* 20.8 (2020), págs. 1239-1261.

[44] Rémi Munos et al. "Safe and efficient off-policy reinforcement learning". En: *Advances in neural information processing systems* 29 (2016).

[45] Larry Olanrewaju Orimoloye et al. "Comparing the effectiveness of deep feedforward neural networks and shallow architectures for predicting stock price indices". En: *Expert Systems with Applications* 139 (2020), pág. 112828.

[46] LH Philip y Renjie Lu. "Cointegrated market-neutral strategy for basket trading". En: *International Review of Economics & Finance* 49 (2017), págs. 112-124.

[47] Doina Precup. "Eligibility traces for off-policy policy evaluation". En: *Computer Science Department Faculty Publication Series* (2000), pág. 80.

[48] Mingyue Qiu, Yu Song y Fumio Akagi. "Application of artificial neural network for the prediction of stock market returns: The case of the Japanese stock market". En: *Chaos, Solitons & Fractals* 85 (2016), págs. 1-7.

[49] Hossein Rad, Rand Kwong Yew Low y Robert Faff. "The profitability of pairs trading strategies: distance, cointegration and copula methods". En: *Quantitative Finance* 16.10 (2016), págs. 1541-1558.

[50] Martin Riedmiller. "Neural fitted Q iteration–first experiences with a data efficient neural reinforcement learning method". En: *Machine Learning: ECML 2005: 16th European Conference on Machine Learning, Porto, Portugal, October 3-7, 2005. Proceedings 16*. Springer. 2005, págs. 317-328.

[51] Tim Salimans et al. "Evolution strategies as a scalable alternative to reinforcement learning". En: *arXiv preprint arXiv:1703.03864* (2017).

[52] John Schulman et al. "Proximal policy optimization algorithms". En: *arXiv preprint arXiv:1707.06347* (2017).

[53] David Silver et al. "A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play". En: *Science* 362.6419 (2018), págs. 1140-1144.

[54] David Silver et al. "Deterministic policy gradient algorithms". En: *International conference on machine learning*. Pmlr. 2014, págs. 387-395.

[55] Satinder Singh et al. "Convergence results for single-step on-policy reinforcement-learning algorithms". En: *Machine learning* 38 (2000), págs. 287-308.

[56] Richard S Sutton. "Generalization in reinforcement learning: Successful examples using sparse coarse coding". En: *Advances in neural information processing systems* 8 (1995).

[57] Tijmen Tieleman, Geoffrey Hinton et al. "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude". En: *COURSERA: Neural networks for machine learning* 4.2 (2012), págs. 26-31.

[58] Hado Van Hasselt, Arthur Guez y David Silver. "Deep reinforcement learning with double q-learning". En: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 30. 1. 2016.

[59] Ganapathy Vidyamurthy. *Pairs Trading: quantitative methods and analysis*. Vol. 217. John Wiley & Sons, 2004.

[60] Ziyu Wang et al. "Dueling network architectures for deep reinforcement learning". En: *International conference on machine learning*. PMLR. 2016, págs. 1995-2003.

[61] Ziyu Wang et al. "Sample efficient actor-critic with experience replay". En: *arXiv preprint arXiv:1611.01224* (2016).

[62] Christopher JCH Watkins y Peter Dayan. "Q-learning". En: *Machine learning* 8 (1992), págs. 279-292.

[63] Peter Wolf et al. "Learning how to drive in a real world simulation with deep q-networks". En: *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE. 2017, págs. 244-250.

[64] Xing Wu et al. "Adaptive stock trading strategies with deep reinforcement learning methods". En: *Information Sciences* 538 (2020), págs. 142-158.

[65] Wenjun Xie et al. "Pairs trading with copulas". En: *The Journal of Trading* 11.3 (2016), págs. 41-52.

[66] ShuiLing Yu y Zhe Li. "Forecasting stock price index volatility with LSTM deep neural network". En: *Recent Developments in Data Science and Business Analytics: Proceedings of the International Conference on Data Science and Business Analytics (ICDSBA-2017)*. Springer. 2018, págs. 265-272.

[67] Zihao Zhang, Stefan Zohren y Stephen Roberts. "Deep reinforcement learning for trading". En: *The Journal of Financial Data Science* 2.2 (2020), págs. 25-40.

[68] Heliang Zheng et al. "Learning multi-attention convolutional neural network for fine-grained image recognition". En: *Proceedings of the IEEE international conference on computer vision*. 2017, págs. 5209-5217.