



Universidad de San Andrés  
Escuela de Administración y Negocios  
Magister en Finanzas

**Funciones de adquisición de optimización bayesiana en el  
ámbito de optimización global para negociación algorítmica**

Autor: Francisco Cucullu

DNI: 36716663

Director externo: Manuel Maurette

Codirector: Gabriel Basaluzzo

Málaga, Marzo 2023

## Tabla de contenido

### Contenido

1	Introducción.....	4
2	Revisión literaria .....	5
3	Descripción del problema.....	6
3.1	Algoritmo de optimización.....	6
3.2	Estrategias de negociación.....	7
3.2.1	Estrategias de medias móviles .....	7
3.2.2	Estrategias con indicadores.....	8
3.3	Generación de simulaciones .....	8
4	Herramientas básicas .....	9
4.1	Definiciones Financieras .....	9
4.2	Definiciones estadísticas .....	9
5	Optimización bayesiana.....	12
5.1	Introducción al modelo.....	12
5.2	Optimización con procesos Gaussianos.....	13
5.2.1	Funciones de media y covarianza.....	14
5.3	Función de adquisición .....	15
5.3.1	Probabilidad de mejora.....	15
5.3.2	Mejora esperada.....	16
5.3.3	Límite de confianza superior .....	17
6	Difusión con saltos y simulación .....	17
6.1	Modelo de Black-Scholes .....	18
6.1.1	Estimación de los parámetros .....	19
6.2	Modelo de difusión con saltos .....	19
6.2.1	Proceso de Poisson simple.....	20
6.2.2	Proceso de Poisson compuesto .....	21
6.2.3	Introducción al modelo de Merton .....	22
6.2.4	Estimación de los parámetros .....	23
6.2.5	Estimación de $\lambda$ .....	23
6.2.6	Estimación de los parámetros de los saltos.....	23
6.2.7	Estimación de los parámetros de la difusión normal.....	24
7	Datos .....	25
8	Experimento.....	26
8.1	Búsqueda de datos .....	26

8.2	Procesamiento de datos .....	27
8.3	Separación de muestras.....	27
8.4	Análisis de datos y generación de simulaciones .....	27
8.5	Ejecución del optimizador.....	28
9	Resultados .....	28
9.1	Demora computacional .....	28
9.2	Rendimiento obtenido.....	30
10	Conclusiones .....	32
11	Referencias.....	34

## Resumen

El objetivo de este trabajo es mostrar la alta sensibilidad de los resultados de un algoritmo de operatoria algorítmica según su parametrización. Para ello, se plantea ejecutar variaciones en un método de optimización global y analizar los resultados. Se busca encontrar diferencias estadísticamente significativas en su rendimiento y en la demora computacional para hallar el conjunto óptimo de parámetros en diferentes estrategias de negociación algorítmica. En particular, se estudiarán variaciones en las funciones de adquisición de la optimización bayesiana. El trabajo utilizará información del mercado de criptomonedas por contar con varios atributos convenientes, como su operatoria continua, su liquidez y la disponibilidad de información.

## 1 Introducción

La motivación de este trabajo se basa en la tendencia a pensar que los problemas de negociación algorítmica se centran únicamente en el desarrollo de estrategias de negociación capaces de actuar mejor que el mercado de forma recurrente. Existe una extensa literatura que estudia diferentes métodos o técnicas que permitan esto, sobre todo en la relacionada a técnicas de operatoria de alta frecuencia. Pero los modelos reales implican muchos otros factores aparte de la estrategia en sí, y omitir su correcta parametrización termina afectando el rendimiento del modelo y el consumo de recursos computacionales.

Cuando se opera en los mercados bursátiles con algoritmos que responden a condiciones cuantitativas, particularmente con técnicas de aprendizaje automático (o *Machine Learning*, en inglés), es habitual que se tengan que ajustar numerosas variables para alcanzar la configuración que optimiza los retornos, la cual incluso cambia con el paso del tiempo. También es habitual que se desconozca la expresión cerrada de la función objetivo y su comportamiento gráfico, transformándose así en problemas de “caja negra”. Al no tener pistas sobre la ubicación del óptimo global, se recae generalmente en métodos de optimización eficaces pero matemática y computacionalmente ineficientes. Si el modelo usado necesita calibrarse recurrentemente, esta ineficiencia se vuelve inaceptable en términos operativos. Por esto mismo, se utilizará la optimización bayesiana como método adecuado según la literatura recabada y se analizará variaciones de su parametrización

La hipótesis de este trabajo es que los rendimientos y la demora computacional de un modelo de negociación algorítmica basado en optimización bayesiana varían significativamente según la función de adquisición utilizada. Las investigaciones se suelen concentrar únicamente en la estrategia de negociación omitiendo el hecho que en la ejecución real, la tardanza del modelo y la diferencia de resultados de la optimización podría implicar una pérdida de retornos potenciales. Es así que, para implementar algoritmos de operatoria cuantitativa profesional no es suficiente contar con modelos de optimización matemática que incorporen eficientemente el reducido conocimiento extraído de una función desconocida y costosa de reproducir, como lo hace la optimización bayesiana. Sino que también es necesario estudiar y adaptar correctamente el optimizador a cada problema en particular, porque omitir el análisis de qué función de adquisición usar podría cambiar los retornos obtenidos. Asimismo, se observará si esta elección podría reducir el problema computacional generado por el gran tamaño del experimento. Para llevar esto a cabo, se utilizará también un modelo de difusión con saltos para generar un conjunto de simulaciones y así evitar un problema de sobre-ajuste que ocurriría al usar únicamente los datos históricos.

Este trabajo presenta la siguiente estructura: la sección 2 contiene la revisión literaria, abarcando tanto a publicaciones en las que se estudia métodos de optimización global, como también publicaciones en donde se hace énfasis en la optimización bayesiana como método superador en diferentes campos del *Machine Learning*; en la sección 3 se detalla el problema y la propuesta para solucionarlo, abarcando detalles tanto del algoritmo de optimización como la generación del experimento; en las secciones 4, 5 y 6 se expone la teoría matemática necesaria para fundamentar el trabajo separada en herramientas básicas, teoría sobre la optimización bayesiana y teoría relacionada al modelo generador de los precios de las simulaciones; en las secciones 7 y 8 se detallan los datos que se utilizarán y la metodología del experimento; para concluir, en las secciones 9 y 10 se comentarán los resultados del trabajo y las conclusiones.

## 2 Revisión literaria

La optimización bayesiana ha tomado una significativa relevancia en los avances recientes de las ciencias matemáticas. Existen diversas aplicaciones en la literatura de los últimos años que van desde técnicas de robótica hasta optimización en el reconocimiento de imágenes. En su mayoría, se la ha aplicado en problemas diametralmente diferentes al de este trabajo pero que comparten al menos tres factores importantes: se desconoce la expresión funcional del problema, su reproducción es costosa y existen múltiples máximos locales.

Entre los primeros trabajos en los que se utiliza esta técnica en problemas relacionados a la optimización global de una función con múltiples máximos locales, se encuentran los aportes de Mockus, Tiesis y Zilinskas (1978). Los autores comentan que la mayoría de métodos de búsqueda de extremos han sido desarrollados en base a la aproximación cuadrática y que en algunos problemas de optimización global la función a minimizar puede ser considerada como la realización de alguna función de carácter estocástico. Llegan a la conclusión que la optimización bayesiana requiere significativamente menos observaciones (es decir, menos evaluaciones de la función objetivo) que otros métodos como los algoritmos de Montecarlo que proponen también en su trabajo. Esta ventaja se llega pagando el costo de una mayor cantidad de cálculos derivados de la búsqueda del próximo punto a evaluar. La diferencia entre la cantidad de cálculos tiene una relación directa con la dificultad de la función objetivo y su costo de reproducción.

Otro aporte relacionado a la eficiencia en la optimización es el trabajo de Bergstra et. al. (2011). En el campo de estudio particular de los autores, se recae habitualmente en la optimización manual dada la incapacidad de probar distintas configuraciones por el costo que significaban los ajustes. Su trabajo introduce dos algoritmos de optimización secuencial y demuestran que superan ampliamente el rendimiento generado por el ajuste manual por parte de expertos. Estos algoritmos de optimización secuencial, que son variaciones de la optimización bayesiana tradicional, superan incluso a algoritmos de fuerza bruta como el de búsqueda aleatoria y también a otros de tipo de redes neuronales. Gracias a esto, demuestran que los trabajos de investigación no tendrían que limitarse a representaciones reducidas del objeto de estudio con pocos parámetros ajustables manualmente, sino que podría ampliarse los experimentos para optimizar de manera más eficiente algoritmos que tengan una flexibilidad más cercana a la observada en la práctica.

Posteriormente, se ha incorporado esta técnica a problemas de *Machine Learning*, particularmente a algoritmos de aprendizaje reforzado. En el trabajo de Lizotte et. al. (2007) se comenta que a pesar de que existen técnicas para automatizar el aprendizaje de los algoritmos, la

mayoría sufre de las tres principales desventajas que se han comentado previamente. Estas desventajas incrementan la necesidad de contar con gran cantidad de evaluaciones funcionales, lo que resulta finalmente en un proceso de optimización excesivamente lento, uso ineficiente de la información extraída en cada evaluación funcional e incluso necesidad de intervenir manualmente en ciertos algoritmos. Los autores presentan la optimización bayesiana basada en un proceso Gaussiano y demuestran que resuelven cada una de estas desventajas y evitando efectivamente los mínimos locales. Demuestran que este enfoque no solo es efectivo en problemas de alta dimensionalidad, con ruido y con funciones no-convexas, sino que también requiere sustancialmente de menos evaluaciones funcionales. Concluyen que esto se logra con mínima intervención en el ajuste de parámetros, demostrando una actuación efectiva incluso cuando se usan supuestos iniciales arbitrarios, la intervención de un experto en el campo de estudio o incluso al introducir errores voluntariamente.

Siguiendo con los aportes al *Machine Learning*, el trabajo de Snoek, Larochelle y Adams (2012) estudia ciertos problemas que acarrea la programación de algoritmos de aprendizaje automatizado y cómo el ejercicio de parametrización del modelo termina resultando, en ocasiones, lo que ellos llaman un trabajo de “magia negra”. Proponen la optimización bayesiana como método candidato para solucionar estos problemas y utilizan un proceso Gaussiano para la función de adquisición. Muestran que ciertas elecciones generadas por la naturaleza del proceso Gaussiano pueden jugar un rol crucial en la generación de un optimizador que pueda superar el rendimiento de un experto en la materia. Incluyen en proceso de optimización una variable que cuantifica la demora computacional y también estudian el rendimiento de la paralelización de los cálculos en distintas unidades computacionales. El modelo de optimización que alcanzan al final de su trabajo reduce significativamente la tardanza de los cálculos y supera de forma consistente el rendimiento de los expertos en la materia en un 3%.

Por último, Bergstra y Bengio (2012) marcan algunos lineamientos para el desarrollo de este trabajo. Si bien los autores estudian un problema específico sobre la parametrización de un modelo de redes neuronales orientadas al reconocimiento de imágenes, repasan un proceso de comparación de modelos de optimización que resulta conveniente para el problema que se aborda en este trabajo. Para parametrizar sus redes, los autores comparan en detalle los resultados de usar un algoritmo básico de búsqueda en grilla contra el de búsqueda en grilla aleatoria usando varios conjuntos de información de alta dimensionalidad (32 dimensiones) que no comparten siempre la misma relevancia entre variables. De esta forma, al cambiar la información utilizada en el modelo, no solo hay que buscar el óptimo en una alta dimensionalidad sino que también no se sabe con antelación qué variables son relevantes y cuáles no lo son. Concluyen en que el algoritmo de búsqueda en grilla aleatoria es superior en resultado y más eficiente en la búsqueda del óptimo. El algoritmo clásico de búsqueda en grilla asigna demasiadas pruebas en la exploración de dimensiones que no son relevantes y sufre de una pobre cobertura de las dimensiones más importantes. También enumeran en sus conclusiones varias ventajas del algoritmo ganador respecto a sus rivales.

### **3 Descripción del problema**

#### **3.1 Algoritmo de optimización**

El objetivo de este trabajo es estudiar el efecto que tiene la elección de la función de adquisición a utilizar y si influye significativamente en los retornos generados por el algoritmo de *Machine Learning* y su demora computacional. Particularmente se estudiará las diferencias que

implica elegir una función de adquisición diferente en el ámbito de optimización bayesiana. En primer lugar por la eficacia en encontrar la cesta de parámetros óptima para estrategia de negociación que repercute en los retornos obtenidos, y en segundo lugar por los insumos computacionales que demanda su ejecución en escenarios de muchas dimensiones, lo que se traduce en demoras en la implementación de los ajustes a la estrategia.

Se tendrá un conjunto  $AD$  que agrupará las tres funciones de adquisición que se estudiarán en este trabajo y se tendrá otro conjunto  $G$  que agrupará seis estrategias de negociación diferentes para estudiar el impacto de las funciones de adquisición en cada una de ellas. Las expresiones de estos dos conjuntos son las siguientes:

$$AD = \begin{bmatrix} ad_1 \\ ad_2 \\ ad_3 \end{bmatrix}, \quad G = \begin{bmatrix} g_1 \\ \vdots \\ g_6 \end{bmatrix}$$

Utilizando estos conjuntos más un conjunto estático  $S^*$  de simulaciones, el algoritmo de optimización tendrá como objetivo maximizar los retornos de cada estrategia  $g$  usando cada una de las tres funciones de adquisición  $ad$ . De esta forma se podrá estudiar el impacto en los resultados de cada función de adquisición en cada una de las diferentes estrategias. Si bien el algoritmo optimizará retornos, también se analizarán los tiempos de demora.

En definitiva, la expresión funcional del algoritmo de este trabajo que maximizará los retornos de cada estrategia variando la función de adquisición será la siguiente:

$$\forall g \in G, ad \in AD: \max_g R_{(g,ad|S^*)}$$

## 3.2 Estrategias de negociación

El conjunto  $G$  de simulaciones estará compuesto por seis estrategias que podrían diferenciarse en dos subgrupos diferentes según su nivel de desarrollo matemático. Se decidió utilizar más de una estrategia de negociación para ampliar los resultados de este trabajo y analizar las funciones de adquisición en distintas configuraciones. De esta forma se podrá ver si el rendimiento y demora de las funciones de adquisición tienen consistencia con diferentes estrategias o si los resultados son más bien dispersos.

### 3.2.1 Estrategias de medias móviles

El primer conjunto está compuesto por tres estrategias que responden a condiciones relativamente simples usando únicamente medias móviles.

La primera estrategia se llamará *TwoStandardMovingAverage* y calculará únicamente dos medias móviles. La estrategia entrará al mercado cuando la media móvil corta esté por encima de la larga y se venderá cuando ocurra lo contrario.

La segunda y la tercera estrategia contarán con tres medias móviles. La estrategia *ThreeStandardMovingAverage* contará con dos medias móviles cortas diferentes y una larga. Entrará al mercado cuando una media corta esté por encima de la larga y venderá en corto cuando la otra media corta esté por debajo de la larga. Esto apunta a darle diferentes velocidades de compra y de venta.

La última estrategia de este subgrupo se llamará *ThreeStandardMovingAverageAlternative* y tendrá una media corta, una media y una larga. Comprará cuando la media corta esté por encima de

la media y la media por encima de la larga. Venderá cuando la corta esté por debajo de la media y la media debajo de la larga.

### 3.2.2 Estrategias con indicadores

Este segundo subconjunto estará compuesto por otras tres estrategias que responderán a condiciones cuantitativas un poco más complejas.

La primera se llamará *ChandelierExitStrategy*. *Chandelier Exit* es un indicador basado en la volatilidad de los precios que identifica posibles puntos de salida para posiciones largas y cortas. Se basa principalmente en el indicador *Average True Range* (ATR) y está diseñado para mantener a los operadores en la tendencia hasta que se detecte una posible inversión de la tendencia. Se basa en el principio de que un cambio de tendencia es altamente probable cuando el precio de un activo se mueve en contra de una tendencia existente hasta tres veces la volatilidad promedio. Utiliza los precios máximos y mínimos durante un período de tiempo definido.

También se contará con la estrategia *VWAPvsSMA* que será la primera en utilizar la información sobre el volumen operado en cada observación. Se calculará compararán dos medias, una media móvil simple y una ponderada por volumen. La estrategia entrará al mercado cuando se detecte una acumulación de volumen negociado mayor a un coeficiente en las últimas observaciones y la media móvil ponderada por volumen este por encima de la simple.

La última estrategia de este subconjunto será la *BBvolumeStrategy* que usará las bandas de Bollinger más un filtro de volumen. Este indicador se compone de tres líneas. Una media móvil simple y otras dos que son la suma y resta de un número determinado de desvíos estándar. Se busca identificar condiciones en las que el mercado presenta sobrecompra o sobreventa basándose en el concepto de reversión a la media del precio. La reversión a la media asume que, si el precio se desvía sustancialmente de la media, se podría esperar una reversión hacia ella.

### 3.3 Generación de simulaciones

Al optimizar las estrategias no se utilizarán sólo los datos del pasado, sino que será necesario ampliar el conjunto de datos de entrenamiento con simulaciones. El ajuste a los datos históricos implicaría asumir que se espera ver un comportamiento en el futuro del mercado exactamente igual al ocurrido en el pasado y haría que los modelos no actúen bien en el mercado real porque regularmente no ocurre de tal forma.

Para solucionar este inconveniente, será necesario desarrollar un modelo complementario que tenga como finalidad la construcción de un conjunto de simulaciones de series temporales que emulen el comportamiento estimado del mercado. Llamaremos  $S^*$  a este conjunto de simulaciones. De esta forma, no sólo se soluciona el problema de sobre ajuste al pasado, sino que también se podría adaptar, a gusto del operador, el comportamiento de las simulaciones para alinearlos con las expectativas futuras del mercado.

El modelo clásico de Black-Scholes es ampliamente utilizado en la práctica por su facilidad de parametrización, pero no ajusta convenientemente a la curva generada por los retornos de los activos financieros en general, y del Bitcoin en particular. Por esto mismo, este trabajo utilizará un modelo de difusión con saltos que permite la desviación de los retornos simulados respecto a una distribución Normal.



## 4 Herramientas básicas

Para entender la investigación que se llevará a cabo a continuación, se detallarán a continuación algunos conceptos de estadística y la teoría financiera. La primera sección contemplará las definiciones financieras necesarias para entender el trabajo. Se hablará del tipo de activo que se utilizará y la forma de calcular los retornos financieros que en definitiva son la variable que utilizará el optimizador para encontrar una cesta de parámetros óptima. En la segunda sección se revisarán definiciones matemáticas que son necesarias para enunciar el método de optimización bayesiano.

### 4.1 Definiciones Financieras

**Definición 1** (Criptoactivo). *Son activos virtuales basados en la criptografía. Se caracterizan principalmente por no necesitar una autoridad central y su administración es llevada a cabo a través de un consenso distribuido. Existen varias razones por qué poseerlos y darles valor: poseen valor de mercado y se puede generar ingresos al comercializarlos, dan acceso a la gobernanza de proyectos tecnológicos a través de su derecho a voto, pueden ser utilizados para pagar comisiones de servicios financieros y pueden utilizarse como dinero, entre otras.*

**Definición 2** (Retorno logarítmico). *Sea  $S_t$  el precio de un instrumento financiero en tiempo  $t$ . Durante el transcurso de tiempo  $\Delta t$ , el retorno logarítmico  $R_{\Delta t}$  del instrumento en cuestión está definido como:*

$$R_{\Delta t} = \ln\left(\frac{S_{t+\Delta t}}{S_t}\right).$$

Resulta importante observar que cuando los incrementos temporales  $\Delta t$  son relativamente pequeños, no existe necesidad de obtener el logaritmo de los retornos ya que matemáticamente son prácticamente idénticos a los aritméticos. Pero como este trabajo analizará criptoactivos con volatilidad superior al de las acciones, se utilizarán de ahora en más los logarítmicos.

De la ecuación expuesta, puede verse que los retornos dependen del tiempo y del paso temporal. Pero los modelos que se detallan no suponen dependencia con el primer factor y mantienen su distribución constante. De este modo, los retornos únicamente dependen del espacio temporal  $\Delta t$ .

### 4.2 Definiciones estadísticas

**Definición 3** (Función de distribución). *Para una variable aleatoria  $X$ , la función de distribución acumulada  $F_X(x)$  (CDF, por sus siglas en inglés), está definida por:*

$$F_X(x) = P(X \leq x).$$

Es decir que la función de distribución acumulada será igual a la probabilidad que la variable aleatoria  $X$  adopte resultados iguales o inferiores a un determinado valor, dada la distribución de probabilidad de  $X$ .

**Definición 4** (Variable aleatoria continua). *Una variable aleatoria  $X$  se dice que es continua si su CDF puede ser escrita de la siguiente forma integral:*

$$F_X(x) = \int_{-\infty}^x f_X(t) dt.$$

Esta es una *integral de Lebesgue*, que es una extensión de la clásica *integral de Riemann*.

**Definición 5** (Esperanza matemática). Para una variable aleatoria continua  $X$ , la esperanza  $E(X)$  está definida por:

$$E(X) = \int_{-\infty}^{\infty} x f_X(t) dx.$$

**Definición 6** (Varianza). Para una variable aleatoria continua  $X$ , su varianza  $Var(X)$  está definida como:

$$Var(X) = E(x - E(X))^2 = \int_{-\infty}^{\infty} (x - E(X))^2 f_X dx$$

**Definición 7** (Distribución gaussiana). La función de probabilidad de densidad (PDF, por sus siglas en inglés) de una variable aleatoria normal  $X$ , denotada como  $X \sim N(\mu, \sigma^2)$ , está definida por:

$$f_X(x) = \varphi(\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x - \mu)^2}{2\sigma^2}\right), \quad -\infty < x < \infty.$$

**Definición 8** (Distribución normal multivariada). Un vector aleatorio  $x = [x_1, \dots, x_n]^T$  sigue una distribución normal multivariada si satisface las siguientes condiciones equivalentes:

- 1) Toda combinación lineal  $Y = a_1 x_1 + \dots + a_n x_n$  está normalmente distribuida.
- 2) Hay un vector  $\mu$  y una matriz semi-definida positiva simétrica  $\Sigma$ . Si  $\Sigma$  es una matriz no singular, entonces la distribución se describe por la función de densidad

$$f_X(x_1, \dots, x_n) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu)\right),$$

donde  $|\Sigma|$  denota el determinante de la matriz  $\Sigma$ , el vector  $\mu$  es la esperanza del vector  $x$  y la matriz  $\Sigma = AA^T$  es la matriz de covarianza de las componentes  $x_j$ . Las matrices  $\mu$  y  $\Sigma$  se definen respectivamente como

$$\mu = [\mu_1, \dots, \mu_n]^T = [E[x_1], \dots, E[x_n]]^T \text{ y}$$

$$\Sigma = [\sigma^2(x_1, x_1) \ \dots \ \sigma^2(x_1, x_n) \ : \ \vdots \ : \ \sigma^2(x_n, x_1) \ \dots \ \sigma^2(x_n, x_n)].$$

**Definición 9** (Proceso estocástico). Suponga que  $\Omega$  es un espacio muestral,  $F$  es un conjunto de todos los eventos y  $P$  es una medida de probabilidad. El conjunto  $(\Omega, F, P)$  sería un espacio probabilístico y  $X$  es algún tipo de índice. Supongamos además que para cada  $x \in X$ , hay una variable aleatoria  $f_x: \Omega \rightarrow R$ . La función  $f: X \times \Omega \rightarrow R$  definida como  $f(x, \omega) = f_x$  es llamada proceso estocástico.

Nótese que para un  $x \in X$ , la función  $f(x, \cdot): \Omega \rightarrow R$  es simplemente una variable aleatoria, mientras que para un  $\omega \in \Omega$  la función  $f(\cdot, \omega): X \rightarrow R$  es una función determinística de  $x$  conocida como camino muestral o realización del proceso estocástico.

**Definición 10** (Movimiento browniano). Un proceso estocástico  $\{W_t\}_{t \geq 0}$  es llamado un movimiento browniano estándar, si las siguientes condiciones son satisfechas:

- 1)  $W_0 = 0$
- 2) Incrementos independientes: para todo tiempo  $0 \leq t_1 \leq t_2 \leq \dots \leq t_n$ , los incrementos de  $W$ :  $W_{t_n} - W_{t_{n-1}}, W_{t_{n-1}} - W_{t_{n-2}}, \dots, W_{t_2} - W_{t_1}$  son variables aleatorias independientes.

3) *Incrementos normalmente distribuidos: para todo tiempo  $0 \leq t_1 \leq t_2 \leq \dots \leq t_n$ , los incrementos de  $W$ :  $W_{t_n} - W_{t_{n-1}}, W_{t_{n-1}} - W_{t_{n-2}}, \dots, W_{t_2} - W_{t_1}$ , están normalmente distribuidos con media cero y varianza  $(t_n - t_{n-1}), (t_{n-1} - t_{n-2}), \dots, (t_2 - t_1)$  respectivamente.*

4) *Con probabilidad igual a 1, la función  $t \rightarrow W_t$  es una función continua en todo  $t$ .*

Nótese que el incremento  $\Delta W = W_{t+\Delta t} - W_t \sim N(0, \Delta t) \sim \sqrt{\Delta t} N(0, 1)$ , lo que significa que  $(W_{t+\Delta t} - W_t) / \sqrt{\Delta t} \sim N(0, 1)$ . Entendiendo esto, se puede simular fácilmente un movimiento browniano simulando una distribución normal con media cero, varianza uno y multiplicarla por la raíz cuadrada del paso temporal, que para nuestro caso sería una hora (se tomarán 8760 horas en un año).

**Definición 11** (Proceso Gaussiano). *Un proceso estocástico  $f(x): x \in X$  es llamado proceso Gaussiano si todas sus distribuciones de dimensión finita son distribuciones normales.*

**Definición 12** (Proceso de Poisson simple). *Sea  $\{t_i\}_{i \geq 1}$  una secuencia de variables aleatorias exponenciales con parámetro  $\lambda$ . Sea  $T_n = \sum_{i=1}^n t_i$ . Entonces el proceso de Poisson  $\{N_t\}_{t \geq 0}$  está definido como:*

$$N_t = \sum_{n \geq 1} 1_{t \geq T_n},$$

donde la intensidad  $\lambda$  es la cantidad esperada de saltos por unidad de tiempo.

Nótese que  $\Delta N_t = N_{t+\Delta t} - N_t$  está distribuido según Poisson y con media  $\lambda \Delta t$ , por lo que la probabilidad de un evento sería

$$P(\Delta N_t = k) = \frac{(-\lambda \Delta t)^k}{k!} e^{-\lambda \Delta t}, \text{ para } k = 1, 2, \dots.$$

**Definición 13** (Proceso de Poisson compuesto). *Sea  $\{Q_i\}_{i \geq 1}$  una secuencia de variables aleatorias independientes e idénticamente distribuidas, y  $\{N_t\}_{t \geq 0}$  un proceso de Poisson con intensidad  $\lambda$ . Entonces un proceso de Poisson compuesto  $\{Y_t\}_{t \geq 0}$  está definido por:*

$$Y_t = \sum_{i=1}^{N_t} Q_i.$$

Si  $N_t = 0$ , entonces  $Y_t$  está definido como  $Y_t = \sum_{i=0}^0 Q_i = 0$ .

**Teorema 1** (Teorema bayesiano). *Dada la función de probabilidad de densidad  $p(x)$  de un vector aleatorio  $X$  y la función de probabilidad de densidad  $p(y)$  de un vector aleatorio  $Y$ , la función de densidad condicional  $p(x|y)$  está definida por:*

$$p(y) = \frac{p(x, y)}{p(y)},$$

donde  $p(x, y)$  es la función de probabilidad multivariada de  $X$  e  $Y$ .

## 5 Optimización bayesiana

### 5.1 Introducción al modelo

La optimización bayesiana es uno de los enfoques más eficientes en términos del número de evaluaciones funcionales requeridas cuando se habla de problemas de optimización global. Es un enfoque secuencial basado en modelos probabilísticos que resuelven de una forma computacionalmente eficiente problemas de todo tipo y, en particular, los problemas de caja negra.

Esta técnica de programación matemática será la elegida en este trabajo para disminuir el consumo de recursos computacionales en la etapa de ajuste del modelo de un problema de *Machine Learning* (*model tuning*, en inglés). Mockus (1975) explica que la optimización bayesiana descansa sobre la idea de acumular la información extraída de cada iteración y combinarla con un modelo probabilístico de una función  $f$  para buscar la ubicación del óptimo global en la próxima iteración.

Por un lado, está compuesta de un modelo probabilístico que parte de una combinación inicial aleatoria o en base a una creencia previa que secuencialmente se va refinando a medida que el modelo lee más información. Si bien la función de costo es desconocida, es razonable asumir que existen conocimientos previos sobre la naturaleza del problema y sea más razonable modelarlo con algunas expresiones que con otras.

Por el otro lado, también se compone de una función de adquisición que en cada interacción evalúa el intercambio entre el costo de reproducir el experimento con una combinación de parámetros diferente y la probabilidad de alcanzar un resultado superior al máximo obtenido hasta el momento. Estas funciones en conjunto analizan la probabilidad de mejorar la función objetivo y guían la experimentación hasta realizar un número deseado de experimentos o hasta no poder generar variaciones del resultado superiores a un determinado nivel de aceptación.

En definitiva, se consigue que la búsqueda se vaya redirigiendo en cada iteración hacia las regiones de mayor interés. El objetivo final es reducir el número de combinaciones de hiperparámetros con las que se evalúa el modelo, eligiendo únicamente los mejores candidatos. Esto significa que, la ventaja frente a las otras estrategias mencionadas, se maximiza cuando el espacio de búsqueda es muy amplio o la evaluación del modelo es muy lenta. Exactamente lo que ocurre en los problemas de operatoria algorítmica de muchas dimensiones.

Como en otros tipos de algoritmos de optimización, la optimización bayesiana busca encontrar el máximo de una función  $f(x)$  de un conjunto acotado  $X$ . Lo que hace a este tipo de optimización diferente de otros procedimientos es que ésta construye un modelo probabilístico para  $f(x)$  y lo utiliza para hacer decisiones sobre en qué otro punto del conjunto  $X$  hacer la siguiente iteración del algoritmo. Lo esencial en el modelo probabilístico es utilizar toda la información disponible de las evaluaciones anteriores de  $f(x)$  e incorporarlas en la próxima decisión. El mecanismo también evita los máximos locales evaluando al conjunto  $X$  en todo su dominio. Esto resulta en un algoritmo que puede encontrar el máximo global de funciones no-convexas que no son derivables con relativamente pocas evaluaciones, ya que compara el costo computacional de cada evaluación con el de la mejora potencial del resultado antes de cada evaluación.

Para llevar esto a cabo es necesario hacer una serie de supuestos para hiperparametrizar el modelo subyacente. También es necesario elegir qué distribución van a tener los datos con los que trabaje el modelo predictivo. Se deben seleccionar, entre un set de funciones potenciales, cuál expresará la distribución de la función a ser optimizada. En este trabajo se utilizará el proceso Gaussiano por su flexibilidad y reproducibilidad. Se necesita también elegir una *función de adquisición*,

la cual será usada para construir la función de utilidad del modelo posterior, permitiéndonos determinar el próximo punto a evaluar.

La optimización bayesiana comúnmente trabaja asumiendo que la función objetivo, que es desconocida, siguió un proceso Gaussiano al momento de recolectar los datos y que seguirá el mismo proceso a medida que nuevas observaciones vayan apareciendo de modo tal que la distribución permanece inafectada por el obrar del investigador. Para elegir los hiperparámetros para la próxima iteración del experimento uno podría optimizar la mejora esperada (ME) como lo hacen Mockus, Tiesis y Zilinskas (1978) sobre el mejor resultado de la última iteración o se podría estimar el límite de confianza superior (LCS) del proceso Gaussiano subyacente como en Srinivas, Krause y Kakade (2009). Los métodos de ME y LCS han mostrado ser eficientes en cuanto al número de evaluaciones funcionales en problemas multimodales de caja negra en Bull (2011).

## 5.2 Optimización con procesos Gaussianos

El proceso Gaussiano es una herramienta conveniente para estudiar el comportamiento de la distribución de las funciones objetivos, las cuales tienen la forma  $f : X \rightarrow R$ . Está definido por la propiedad de que cualquier subconjunto finito de  $N$  puntos  $\{x_n \in X\}_{n=1}^N$  inducen a una distribución gaussiana multivariada sobre  $R^N$ . Por ello, hasta la última iteración del algoritmo tendrá una determinada función  $f(x_n)$  que, gracias a las propiedades de la distribución gaussiana, se podrán calcular sus derivadas en una forma cerrada.

El componente probabilístico de la optimización bayesiana es el supuesto de que la función objetivo es la realización de algún proceso Gaussiano subyacente como se introdujo en la definición 9. Hay que recordar que en la configuración de la optimización bayesiana solo se tiene acceso a un número finito de observaciones, lo que significa que ni la distribución de los resultados ni el proceso subyacente son conocidos. Por ende, en la práctica solo se puede imponer los supuestos sobre los valores de la función objetivo. En otras palabras se los trata como evaluaciones del proceso Gaussiano subyacente en determinados puntos  $x_i \in X$ , los cuales son en sí variables aleatorias. Además, por la definición 8 cualquier vector finito de los resultados de la función objetivo tendrá una distribución gaussiana conjunta de dimensión  $N$  como la que se trabajó en Rasmussen (2006).

Formalmente, se supone que la muestra de  $n$  observaciones  $x = [x_1, \dots, x_n]^T \subset X^n$  y los valores de función objetivo asociada  $f = [f(x_1), \dots, f(x_n)]^T \subset R^n$  están dados. Siguiendo con el supuesto Gaussiano, se tiene que:

$$f \sim N(m, K),$$

donde  $m$  es el vector de las medias con dimensión  $n \times 1$  y  $K$  es la matriz de covarianzas con dimensión  $n \times n$ , tal y como lo expresa la definición 8.

Supongamos que se tiene un punto  $n + 1$  aún no calculado cuyo valor es  $x_* \in X$ . Ahora, en vez de evaluar la costosa función objetivo en ese punto podemos predecir su valor, es decir podemos calcular  $f_* = f(x_*)$  bajo el supuesto del proceso Gaussiano. Una vez más, por los supuestos Gaussianos y por la definición 11, se puede expresar que:

$$[f(x_*) \ f(x_1) \ \dots \ f(x_n)] = \begin{bmatrix} f_* \\ f \end{bmatrix}$$

sigue una distribución normal conjunta, definida por:

$$\begin{bmatrix} f_* \\ f \end{bmatrix} \sim N([m_* \ m], [k_{**} \ k_* \ k_*^T \ K]),$$

en donde se han separado tanto el vector de medias como la matriz de covarianzas para enfatizar qué partes de ellas representan la muestra anterior, el nuevo punto evaluado y cuál representa a ambos. Por lo tanto, tenemos que  $k_*$  es un vector de dimensión  $n \times 1$  que representa las covarianzas entre la muestra anterior y la nueva observación. Este vector se puede expresar como:

$$[k_*] = [k(x_1, x_*) : k(x_n, x_*)]$$

y similarmente se podrían expresar los vectores  $m_* = m(x_*)$  y  $k_{**} = k(x_*, x_*)$ .

### 5.2.1 Funciones de media y covarianza

Hay que considerar que el proceso Gaussiano está completamente definido por la función de su media  $m$  por un lado, y su función de covarianza  $k$ , por el otro. Es por ello que la elección de estos parámetros determinará ciertos atributos de la función objetivo. Como el proceso Gaussiano subyacente es desconocido, sus funciones de media y covarianza son, en definitiva, también desconocidas. Su elección será completamente a elección del investigador.

La estimación de la función de la media es realizada habitualmente a través de una regresión sobre la información disponible. Los métodos pueden variar desde estimar una constante, una regresión lineal o una regresión del tipo polinómica. Esto dependerá de cuáles son los supuestos sobre la función objetivo. Si se carece de mucha información, el enfoque más sencillo sería el de estimar una constante. En este caso, la media  $m_*$  de los puntos aún no observados  $x_*$  será igual a la media de los puntos anteriormente evaluados.

La función de covarianza es usualmente más interesante de estudiar porque codifica la relación entre los valores de la función modelada con el conocimiento extraído de las ejecuciones anteriores de la misma. El trabajo de Rasmussen (2006) también enuncia un gran número de propuestas para la función de la covarianza.

Para este trabajo se utilizará la exponencial cuadrática ya que es una de las propuestas más utilizadas. Esta se define como:

$$k(x, x') = \alpha \exp\left(-\sum_{i=1}^d \frac{|x_i - x'_i|^2}{\theta_i}\right),$$

para cualquier  $x = (x_1, \dots, x_d)$ ,  $x' = (x'_1, \dots, x'_d)$  y  $\alpha, \theta_1, \dots, \theta_d > 0$ . Si se ignora los parámetros por un momento, se puede obtener una buena interpretación de esta función que, tenderá a 1 para los puntos cercanos o similares y tenderá a 0 para los puntos lejanos. En este sentido, el parámetro  $\alpha$  puede ser interpretado como la magnitud de la correlación y el vector  $\theta = \{\theta_1, \dots, \theta_d\}$  como cuánto cambia la correlación en cada una de las  $d$  dimensiones.

Debido a que la elección manual de estos parámetros es bastante complicada, se los suele estimar a partir de la información disponible. Hasta este momento se ha supuesto que los parámetros son conocidos, pero si no se lo hiciese habría que elegir el mejor conjunto maximizando su probabilidad en base a la información disponible:

$$p(f) = \frac{p(f | \alpha, \theta) p(\alpha, \theta)}{p(f)}.$$

Como el denominador de esta expresión no depende de los parámetros en cuestión, se lo puede considerar como constante. Toda combinación de parámetros es igualmente probable, por lo

que asignarle una distribución uniforme no resulta tan descabellado. Es más, en ausencia de un supuesto fuerte de  $p(\alpha, \theta)$  sobre la distribución de los parámetros, se tendría simplemente:

$$p(\alpha, \theta | f) \propto p(f | \alpha, \theta).$$

El término de la derecha se refiere a la similitud y se interpreta como cuán bien los parámetros describen a la información disponible.

### 5.3 Función de adquisición

Esta será la función  $a : X \rightarrow R$  que se utilizará para determinar qué punto dentro del set  $X$  será el próximo a ser evaluado. La intuición detrás de ella es que los valores altos de la función de adquisición corresponden a valores altos de la verdadera función objetivo. Matemáticamente se formula esto como un problema de optimización sujeto a restricciones en los valores de las variables, es decir:

$$x_{next} = a(x).$$

La esencia de la optimización bayesiana es utilizar el hecho que optimizar  $a(x)$  es relativamente más sencillo que optimizar la función objetivo, ya que la evaluación del modelo probabilístico de  $a$  en cualquier punto es muchísimo menos costoso que reproducir la función objetivo verdadera.

Este modelo probabilístico es la distribución posterior derivada en la sección 3.2 y es utilizada como un reemplazo del valor de la función objetivo en un determinado punto. En toda esta sección se supone, por lo tanto, que  $f(x)$  para un determinado  $x \in X$  es una variable normal que tiene media  $\mu(x)$  y varianza  $\sigma^2(x)$ . Además, se usará el máximo valor de la verdadera función objetivo observado hasta el momento, el cual se notará como  $f(x^+)$ .

Hay tres métodos muy comunes en la literatura para trabajar con la función de adquisición. La propiedad clave que una función de adquisición debe satisfacer es un equilibrio entre la *explotación* y la *exploración*, en donde explotación concierne a preferir los puntos con mayor media posterior, mientras que exploración concierne a preferir los puntos con mayor varianza posterior. A continuación se detallarán las funciones de adquisición que constituirán el conjunto  $AD$  que tendrá un rol fundamental en el experimento de este trabajo.

#### 5.3.1 Probabilidad de mejora

Históricamente la primera función de adquisición introducida fue la *probabilidad de mejora* ( $PM$ ) en Kushner (1964). La idea principal es maximizar la probabilidad de mejorar el mejor valor obtenido hasta el momento  $f(x^+)$ . Esta función de adquisición es expresada matemáticamente como:

$$PM(x) = P[f(x) > f(x^+)].$$

Teniendo los supuestos sobre la distribución de  $f(x)$  todavía en mente, se puede obtener una expresión cerrada de  $PM$  de la siguiente forma:

$$\begin{aligned} PM(x) &= 1 - P[f(x) < f(x^+)] \\ &= 1 - P\left[\frac{f(x) - \mu(x)}{\sigma(x)} < \frac{f(x^+) - \mu(x)}{\sigma(x)}\right] \\ &= 1 - \Phi\left(\frac{f(x^+) - \mu(x)}{\sigma(x)}\right) \end{aligned}$$

$$= \Phi \left( \frac{\mu(x) - f(x^+)}{\sigma(x)} \right).$$

Una desventaja de este enfoque es que  $PM(x)$  va a ser grande cuando  $\sigma(x)$  es pequeña. Esto significa que las zonas exploradas con el actual máximo observado van a ser preferidas a las zonas inexploradas, lo que derivará en un comportamiento con demasiada explotación.

### 5.3.2 Mejora esperada

Otra función de adquisición es la *mejora esperada* ( $ME$ ) que fue introducida en Mockus, Tiesis y Zilinskas (1978) para remediar el problema de la explotación de la función anterior. Esta función se basa en la idea de tomar en cuenta la cantidad de mejora potencial y no solo la probabilidad como en la anterior. La función de mejora puede definirse como:

$$M(x) = \begin{cases} f(x) - f(x^+), & f(x) > f(x^+) \\ 0, & f(x) \leq f(x^+) \end{cases}$$

De ahora en más se reemplazará  $M(x)$ ,  $f(x)$ ,  $f(x^+)$ ,  $\mu(x)$  y  $\sigma^2(x)$  por  $M$ ,  $f$ ,  $f_+$ ,  $\mu$  y  $\sigma^2$  respectivamente para facilitar la notación.

Nótese que  $M$  es simplemente una función de una variable aleatoria, por lo que se podría calcular su esperanza matemática  $E(M)$ , la cual será en definitiva la función de adquisición  $ME$  querida. De la misma forma que en el método anterior, uno preferiría tener una expresión cerrada. Esta expresión se obtendría en primer lugar si se hacen los siguientes cambios de variable:

$$Z = \frac{f - \mu}{\sigma} \quad y \quad f'_+ = \frac{f_+ - \mu}{\sigma}.$$

La función de mejora puede ahora volverse a expresarse como:

$$M = \begin{cases} \sigma(Z - f'_+), & Z > f'_+ \text{ y } \sigma > 0 \\ 0, & \text{de otro modo} \end{cases}.$$

La mejora esperada para el caso en donde  $Z > f'_+$  puede expresarse por la siguiente simplificación:

$$ME = E(M) = \sigma \int_{f'_+}^{\infty} (z - f'_+) \phi_{(z)} dz,$$

Si se hace el cambio de variable  $z = -t$ , se tiene que:

$$\begin{aligned} ME &= -\sigma \int_{-f'_+}^{-\infty} (-t - f'_+) \phi_{(-t)} dt \\ &= \sigma \left( \int_{-\infty}^{-f'_+} -t \phi_{(t)} dt - f'_+ \int_{-\infty}^{-f'_+} \phi_{(t)} dt \right) \\ &= \sigma (A - f'_+ \Phi_{(-f'_+)}) \end{aligned}$$

en donde se define a  $A$  y a  $\Phi_{(-f'_+)}$  respectivamente como:

$$\int_{-\infty}^{-f'_+} -t \phi_{(t)} dt \quad y \quad \int_{-\infty}^{-f'_+} \phi_{(t)} dt.$$



La primera de estas últimas integrales podría reformularse si se escribe a  $\phi_{(t)}$  explícitamente como:

$$A = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-f'_+} -t \exp\left(-\frac{t^2}{2}\right) dt.$$

Usando ahora el cambio de variable  $u = -\frac{t^2}{2}$  se obtiene que:

$$A = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-\frac{(f'_+)^2}{2}} \exp(u) du = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(f'_+)^2}{2}\right) = \phi_{(f'_+)}.$$

Por último, la mejora esperada de la función de adquisición, en su forma cerrada, puede expresarse como:

$$\begin{aligned} ME &= \sigma \left( \phi_{(f'_+)} - f'_+ \Phi_{(-f'_+)} \right) \\ &= \sigma \phi_{\left(\frac{\mu - f_+}{\sigma}\right)} + (\mu - f_+) \Phi_{\left(\frac{\mu - f_+}{\sigma}\right)}. \end{aligned}$$

En el caso en el que la función de mejora sea cero, consecuentemente la mejora esperada también será cero.

Esta última expresión para  $ME$  es la suma de dos términos. El primero aumenta cuando  $\sigma$  aumenta, mientras que el segundo aumenta cuando es  $\mu$  la que lo hace. Intuitivamente, estos términos pueden ser interpretados como las propiedades de exploración y explotación, respectivamente.

### 5.3.3 Límite de confianza superior

Esta tercera función de adquisición es una expresión heurística simple que apunta a balancear el intercambio entre explotación y exploración. Está definida por

$$LCS(x) = \mu(x) + \beta \sigma(x),$$

Donde el término  $\beta$  es un número real no negativo que podría interpretarse como la cantidad preferida de exploración, la que es usualmente determinada por el investigador. La ventaja de incluir el factor de exploración  $\beta$  es que ofrece cierta flexibilidad y una correcta especificación puede derivar en una evidente mejora en la optimización. Aunque tiene la desventaja que si valor depende de la variable independiente, no será sencillo determinarla.

## 6 Difusión con saltos y simulación

Al optimizar las estrategias, además de utilizar los datos históricos, se ampliará el conjunto de datos de entrenamiento con simulaciones. En la sección 7 se detallarán los datos a utilizar en el trabajo. El ajuste a los precios históricos del Bitcoin en el año 2021 implicaría asumir que se espera ver un comportamiento en el futuro del mercado exactamente igual al ocurrido en el pasado y haría que los modelos no actúen bien en el mercado real porque regularmente no ocurre de tal forma.

Para solucionar este inconveniente, será necesario desarrollar un modelo complementario que tenga como finalidad la construcción de un conjunto de simulaciones de series temporales que emulen el comportamiento estimado del mercado. Llamaremos  $S^*$  a este conjunto y el código se programó siguiendo lo desarrollado en esta sección. De esta forma, no sólo se soluciona el problema

de sobre ajuste al pasado, sino que también se podrá ajustar, a gusto del operador, el comportamiento de las simulaciones para alinearlos con las expectativas futuras del mercado.

Este trabajo utilizará un modelo de difusión con saltos que permite la desviación de los retornos simulados respecto a una distribución Normal. Se toma esta elección debido a la evidente mejoría en el ajuste de la distribución de los retornos simulados con los observados. El modelo de Black-Scholes, si bien es ampliamente utilizado en la práctica por su facilidad de parametrización, no ajusta convenientemente a la curva generada por los retornos de los activos financieros en general, y del Bitcoin en particular. Para llegar al modelo de Merton, de todas formas es necesario exponer la teoría del modelo Black-Scholes.

En la teoría tradicional se utiliza simplemente una distribución gaussiana como en la definición 7, pero en el modelado del comportamiento de criptoactivos se requiere el uso de otro tipo de distribuciones. Los criptoactivos tienen explosiones de volatilidad muy regularmente que generan que las colas sean aún más pesadas que las acciones. Pasar este factor por alto implicaría subestimar el riesgo de saltos repentinos en el precio, a favor o en contra, y generar un entramado de simulaciones que no refleje la volatilidad a corto plazo del verdadero proceso subyacente.

En este sentido, se utilizarán como lineamientos los aportes de Cartea y Figueroa (2005) y Tang (2018) para descomponer el proceso estocástico en dos subprocesos con distribuciones diferentes que, al unirse generarían una función de densidad lo más similar posible al pasado inmediato recogido por los operadores.

## 6.1 Modelo de Black-Scholes

El precio de un activo financiero bajo este modelo se comporta de la siguiente forma:

$$S_t = S_0 \exp \left( \left( \mu - \frac{\sigma^2}{2} \right) t + \sigma W_t \right)$$

En donde  $S_0$  es el precio del criptoactivo en el tiempo cero,  $\mu$  es la media de los retornos (tendencia), y  $\sigma$  es la volatilidad de los retornos. Ambas griegas expresadas en términos anuales.

Conociendo cómo se calcula el retorno logarítmico gracias a la definición 2, se puede calcular que los precios logarítmicos en una primera instancia son:

$$\begin{aligned} (S_t) &= \ln \ln \left( S_0 \exp \left( \left( \mu - \frac{\sigma^2}{2} \right) t + \sigma W_t \right) \right) \\ &= \ln \ln (S_0) + \left( \mu - \frac{\sigma^2}{2} \right) t + \sigma W_t \end{aligned}$$

y que los retornos logarítmicos finalmente son:

$$\begin{aligned} R_{\Delta t} &= \ln \left( \frac{S_{t+\Delta t}}{S_t} \right) \\ &= \left( \mu - \frac{\sigma^2}{2} \right) \Delta t + \sigma \Delta W_t \\ &= \left( \mu - \frac{\sigma^2}{2} \right) \Delta t + \sigma \sqrt{\Delta t} N(0,1). \end{aligned} \quad (1)$$

Esto quiere decir básicamente que:

$$\frac{R_{\Delta t} - \left(\mu - \frac{\sigma^2}{2}\right)\Delta t}{\sigma\sqrt{\Delta t}} \sim N(0,1).$$

### 6.1.1 Estimación de los parámetros

Para utilizar el modelo en la práctica, es necesario estimar la tendencia y la volatilidad real de los datos con los que se está trabajando.

Sean  $E(R_{\Delta t})$  y  $Var(R_{\Delta t})$  la esperanza y la varianza de los retornos respectivamente, se puede extraer en base a las definiciones 5, 6 y a la fórmula (1) que ambos parámetros son los siguientes:

$$E(R_{\Delta t}) = \left(\mu - \frac{\sigma^2}{2}\right)\Delta t \quad Var(R_{\Delta t}) = \sigma^2\Delta t.$$

Despejando ahora las griegas, que son nuestras incógnitas, podemos ver que:

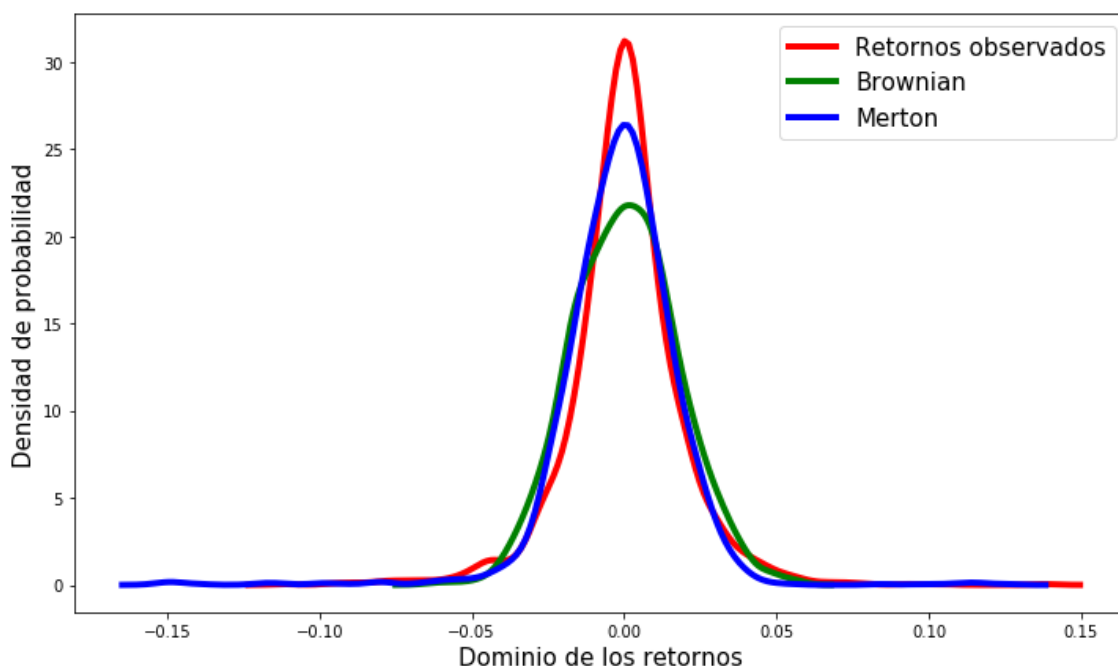
$$\hat{\mu} = \frac{2\hat{E}(R_{\Delta t}) + \widehat{Var}(R_{\Delta t})\Delta t}{2\Delta t} \quad \hat{\sigma}^2 = \frac{\widehat{Var}(R_{\Delta t})}{\Delta t},$$

Donde  $\hat{E}(R_{\Delta t})$  y  $\widehat{Var}(R_{\Delta t})$  son la media y varianza muestral, respectivamente, de los retornos empíricos. Aunque la optimización bayesiana podría estimar con precisión todos parámetros de las simulaciones, siempre es beneficioso partir de una primera estimación para utilizarla como vector inicial del proceso iterativo y ahorrarse de iteraciones que estén muy lejanas al punto óptimo. Dado que este trabajo tiene como objetivo principal la estimación precisa de los parámetros del modelo, se hará una comparación del resultado final con esta primera aproximación para evaluar si el beneficio de la optimización exacta compensa el costo computacional del modelo.

## 6.2 Modelo de difusión con saltos

Se ha comprobado repetidas veces que los retornos de las acciones no siguen un proceso estocástico normal como se expone en la sección anterior. Ocurre que en la práctica, las distribuciones suelen ser acampanadas como en la teoría pero sus colas son visiblemente más pesadas. Suponer que la distribución implícita es normal, entonces, implicaría subestimar la probabilidad de ocurrencia de retornos extraordinarios lejanos a la media. Debido a esto, se incurriría a subestimar el riesgo de la inversión, ya que en realidad hay más probabilidad de sufrir fuertes pérdidas en un solo paso temporal.

**Gráfico N°1: Comparación de densidades**



*Fuente: Retornos del Bitcoin en 4 horas durante el año 2021.*

En el caso de los retornos logarítmicos de los criptoactivos, los hechos se repiten. Se puede observar en el gráfico N°1 la inexistencia de una distribución normal en el proceso estocástico subyacente del Bitcoin durante todo el período, motiva la introducción e implementación práctica de los aportes realizados por Merton (1976). Este modelo es una ampliación del modelo de Black-Scholes tradicional que le agrega la posibilidad que los retornos tengan saltos discretos. Esto permitiría dividir la estocasticidad del proceso en dos distribuciones separadas cuya yuxtaposición dependerá de una variable de Poisson. Las dos tendrán retornos normales, una para modelar los retornos habituales del criptoactivo y la otra para explicar las explosiones que se dan de vez en cuando. Para analizar los detalles de este modelo, primero es necesario introducir el comportamiento de las variables aleatorias que se estarían agregando.

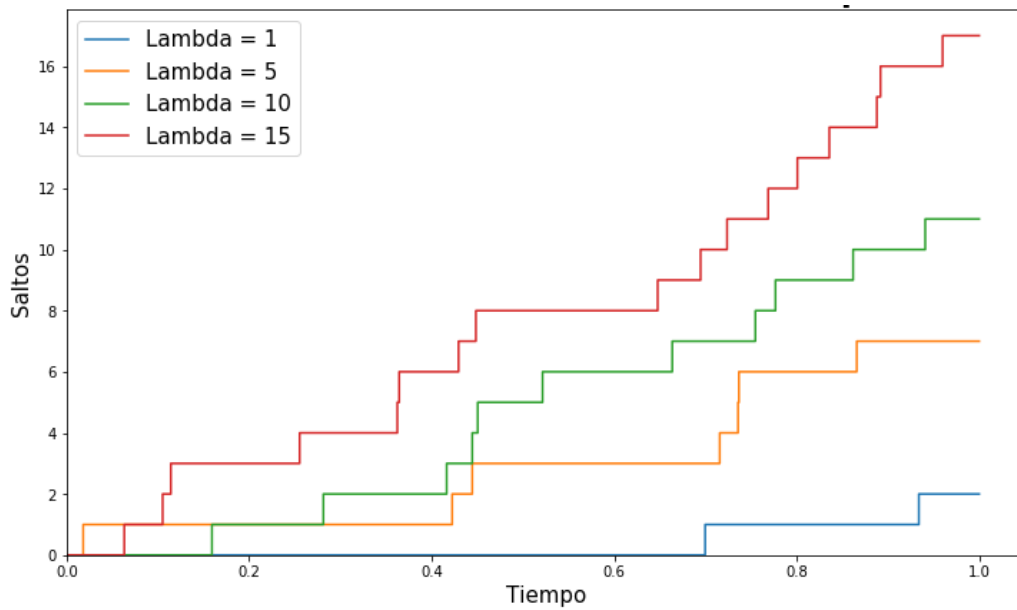
### **6.2.1 Proceso de Poisson simple**

Para modelar una sucesión de eventos extraordinarios se utiliza un proceso estocástico de tiempo continuo que consiste en contar eventos fuera de lo normal que ocurren a lo largo de un período determinado de tiempo. En este sentido podemos aplicar lo visto en la definición 12 para materializar un proceso de Poisson homogéneo.

En este primer caso, partimos de la base en la que estos saltos son simplemente contabilizaciones de los hechos extraordinarios. Por ahora no es importante su intensidad ni dirección, sólo interesa su ocurrencia. Así, podremos modelar una línea temporal continua que salta de vez en cuando con saltos unitarios siguiendo los parámetros con los que se genera el proceso, que en este caso es simplemente la cantidad de saltos que ocurren en un período de tiempo.

Definido esto, a continuación se realizarán cuatro simulaciones de un proceso de Poisson  $\{N_t\}_{t \geq 0}$  con intensidad  $\lambda$  diferentes. Se podrá ver que el factor de intensidad  $\lambda$  será aproximadamente igual a la cantidad de saltos que tiene cada proceso durante cada unidad de tiempo.

**Gráfico N°2: Proceso de Poisson simple**



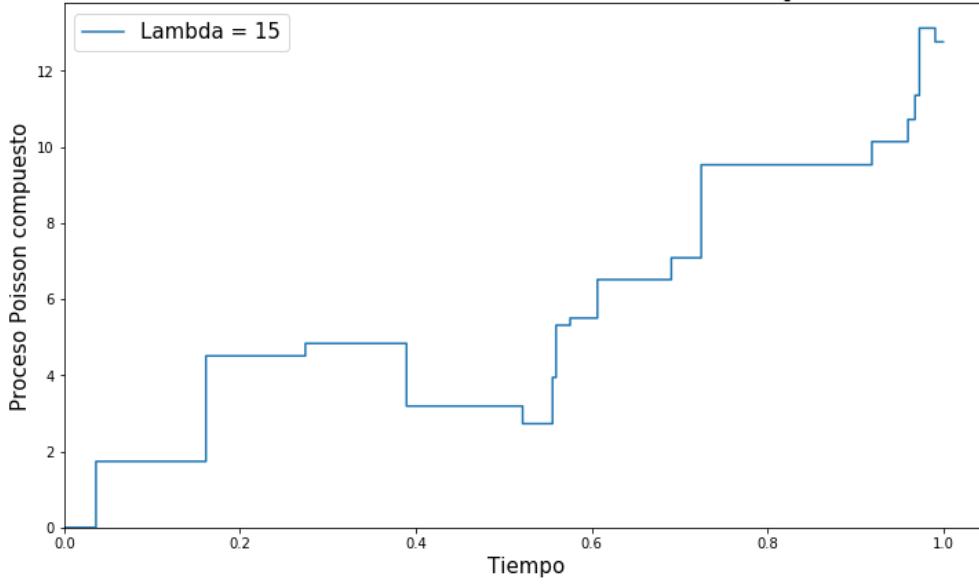
*Fuente: Elaboración propia en base a simulaciones.*

### 6.2.2 Proceso de Poisson compuesto

El caso particular del modelo de Merton que se estima implementar tiene una complejidad adicional. No solo la cantidad de saltos es aleatoria, sino también su intensidad que incluso puede ser positiva o negativa.

Para simular un proceso de Poisson compuesto expuesto en la definición 13, la suma de los saltos en  $[t, t + \Delta t]$  tienen la misma distribución como  $\sum_{i=1}^{\Delta N_t} Q_i$  donde se recuerda que la cantidad de saltos  $\Delta N$  está distribuida según Poisson con media  $\lambda \Delta t$ . Para simular  $\sum_{i=1}^{\Delta N_t} Q_i$  primero se genera  $\Delta N_t$  según una distribución de Poisson con intensidad  $\lambda \Delta t$ . Dado ese número aleatorio  $\Delta N_t = k$ , entonces se genera  $\sum_{i=1}^{\Delta N_t} Q_i = \sum_{i=1}^k Q_i$ . Si  $Q_i$  está distribuido normalmente, como en el modelo de Merton, entonces  $Q_i \sim N(\mu_j, \sigma_j^2)$ , donde  $j$  representa los saltos. Dicho esto, se demostraría que  $\sum_{i=1}^k Q_j \sim N(k\mu_j, k\sigma_j^2)$ .

**Gráfico N°3: Proceso de Poisson compuesto**



*Fuente: Elaboración propia en base a simulaciones.*

### 6.2.3 Introducción al modelo de Merton

Entendiendo la dinámica básica del modelo de Black-Scholes y los procesos de Poisson, se puede comenzar con el estudio del modelo de difusión con saltos. Este modelo toma exactamente la difusión exponencial de precios del anterior y le agrega un proceso de Poisson compuesto, entonces:

$$S_t = S_0 \exp \left( \left( \mu_d - \frac{\sigma_d^2}{2} \right) t + \sigma_d W_t + \sum_{i=1}^{\Delta N_t} Q_i \right),$$

Donde  $S_t$  es el precio del activo en tiempo  $t$ ,  $\mu_d$  y  $\sigma_d$  son la tendencia y la volatilidad respectivamente de la parte normal de la difusión,  $W_t$  es un movimiento Browniano estándar, y  $\sum_{i=1}^{\Delta N_t} Q_i$  es un proceso de Poisson compuesto con sus saltos distribuidos normalmente  $N(\mu_j, \sigma_j^2)$  con intensidad  $\lambda$ . De ahora en más, el proceso estará dividido en dos componentes: por un lado la difusión propiamente dicha cuyos parámetros serán denotados con la letra  $d$ , y por el otro lado un componente discreto que le dará la posibilidad de saltos a la difusión original cuyos parámetros serán denotados con la letra  $j$ .

Se puede demostrar que los precios logarítmicos del modelo de difusión con saltos son:

$$\begin{aligned} (S_t) &= \ln \ln \left( S_0 \exp \left( \left( \mu_d - \frac{\sigma_d^2}{2} \right) t + \sigma_d W_t + \sum_{i=1}^{\Delta N_t} Q_i \right) \right) \\ &= \ln \ln (S_0) + \left( \mu_d - \frac{\sigma_d^2}{2} \right) t + \sigma_d W_t + \sum_{i=1}^{\Delta N_t} Q_i \end{aligned} \quad (2)$$

Luego, por la Definición 2 se puede llegar a que los retornos logarítmicos de este modelo son:

$$R_{\Delta t} = \ln\left(\frac{S_{t+\Delta t}}{S_t}\right)$$

$$= \left(\mu_d - \frac{\sigma_d^2}{2}\right)\Delta t + \sigma_d \Delta W_t + \sum_{i=N_t}^{N_{t+\Delta t}} Q_i, \quad (3)$$

donde  $\Delta W_t = W_{t+\Delta t} - W_t$  es el incremento discreto del movimiento browniano,  $Q_i$  están independientemente distribuidos de forma normal con media  $\mu_j$  y varianza  $\sigma_j^2$ , y  $\Delta N_t = N_{t+\Delta t} - N_t$  es una variable aleatoria de Poisson con media  $\lambda \Delta t$ .

#### 6.2.4 Estimación de los parámetros

Partiendo de la ecuación (2) se podría demostrar que la esperanza y la varianza de los retornos logarítmicos de este nuevo modelo son:

$$\{E(R_{\Delta t}) = \left(\mu_d - \frac{\sigma_d^2}{2}\right)\Delta t + \mu_j \lambda \Delta t \quad Var(R_{\Delta t}) = \sigma_d^2 \Delta t + (\sigma_j^2 + \mu_j^2) \lambda \Delta t. \quad (4)$$

Una vez más, aunque la optimización bayesiana podría estimar con precisión todos parámetros de las simulaciones, siempre es beneficioso partir de una primera estimación para utilizarla como vector inicial del proceso iterativo y ahorrarse de iteraciones que estén muy lejanas al punto óptimo. Dado que este trabajo tiene como objetivo principal la estimación precisa de los parámetros del modelo, se hará una comparación del resultado final con esta primera aproximación para evaluar si el beneficio de la optimización exacta compensa el costo computacional del modelo.

#### 6.2.5 Estimación de lambda

La decisión de cuántos saltos incluir en el modelo es crucial para que esta adaptación tenga sentido. No existiría un cambio significativo en la bondad de ajuste del modelo si subestimamos este factor. Se asume que un salto ocurre cuando el valor absoluto de los retornos logarítmicos excede un determinado umbral a definir por el investigador. En este caso, se ha decidido implementar los aportes de Cartea y Figueroa (2005) los cuales establecen como umbral los tres desvíos estándar. Una vez que se identifica la cantidad de saltos que ocurrieron en base a esta decisión, se podría inferir que la cantidad de saltos por año será aproximadamente igual al total de saltos identificados dividido la longitud temporal de nuestra información medida en términos anuales. De esta forma, estamos asumiendo que la cantidad de saltos que podrían ocurrir a futuro está directamente relacionada con el índice de incidencia que estos eventos extremos ha tenido en el pasado. Aunque claro está, podría no ser el caso.

En el caso particular del Bitcoin, a través del algoritmo iterativo expuesto en las secciones anteriores, se ha encontrado que desde el comienzo de la muestra han ocurrido exactamente veintidós saltos. Esto hace que las observaciones iniciales, se dividan dos nuevas bases. Por un lado se tienen las observaciones de la difusión normal, compuesta por la mayoría de observaciones en las que no se evidenció una variación logarítmica que exceda nuestro umbral. Por el otro lado, se tiene la base en donde concentra todas las ocurrencias de variaciones excepcionales, compuesta por la menor parte de las observaciones.

#### 6.2.6 Estimación de los parámetros de los saltos

Una primera aproximación a los parámetros podría ser asumir que existe solo un salto, en ese caso la expectación y la varianza de los retornos logarítmicos son:

$$E(R_{\Delta t}^J) = E(N_{t+\Delta t} - N_t = 1)$$

$$\begin{aligned}
&= E \left[ \left( \mu_d - \frac{\sigma_d^2}{2} \right) \Delta t + \sigma_d \Delta W + Q_i \right] \\
&= \left( \mu_d - \frac{\sigma_d^2}{2} \right) \Delta t + \mu_j, \\
\text{Var}(R_{\Delta t}^J) &= \text{Var}(N_{t+\Delta t} - N_t = 1) \\
&= \text{Var} \left[ \left( \mu_d - \frac{\sigma_d^2}{2} \right) \Delta t + \sigma_d \Delta W + Q_i \right] \\
&= \sigma_d^2 \Delta t + \sigma_j.
\end{aligned}$$

En resumen:

$$E(R_{\Delta t}^J) = \left( \mu_d - \frac{\sigma_d^2}{2} \right) \Delta t + \mu_j \quad \text{Var}(R_{\Delta t}^J) = \sigma_d^2 \Delta t + \sigma_j.$$

Finalmente, de las ecuaciones anteriores se extrae que los parámetros  $\mu_j$  y  $\sigma_j^2$  son:

$$\{\hat{\mu}_j = \hat{E}(R_{\Delta t}^J) - \left( \hat{\mu}_d - \frac{\hat{\sigma}_d^2}{2} \right) \Delta t \quad \hat{\sigma}_j^2 = \widehat{\text{Var}}(R_{\Delta t}^J) - \hat{\sigma}_d^2 \Delta t, \quad (5)$$

donde  $\hat{E}(R_{\Delta t}^J)$  y  $\widehat{\text{Var}}(R_{\Delta t}^J)$  son la media y varianza muestral de los retornos logarítmicos empíricos de la segunda base de datos compuesta únicamente por los saltos.

### 6.2.7 Estimación de los parámetros de la difusión normal

Cuando no existen saltos, la expectación y la varianza de los retornos logarítmicos son:

$$\begin{aligned}
E(R_{\Delta t}^D) &= E(N_{t+\Delta t} - N_t = 0) \\
&= E \left[ \left( \mu_d - \frac{\sigma_d^2}{2} \right) \Delta t + \sigma_d \Delta W \right] \\
&= \left( \mu_d - \frac{\sigma_d^2}{2} \right) \Delta t, \\
\text{Var}(R_{\Delta t}^D) &= \text{Var}(N_{t+\Delta t} - N_t = 0) \\
&= \text{Var} \left[ \left( \mu_d - \frac{\sigma_d^2}{2} \right) \Delta t + \sigma_d \Delta W \right] \\
&= \sigma_d^2 \Delta t.
\end{aligned}$$

Los parámetros  $\mu_d$  y  $\sigma_d$  pueden ser extraídos de las ecuaciones anteriores, como se ha procedido anteriormente al analizar el modelo de Black-Scholes. Operando del mismo modo:

$$\hat{\mu}_d = \frac{2\hat{E}(R_{\Delta t}^D) + \widehat{\text{Var}}(R_{\Delta t}^D)\Delta t}{2\Delta t} \quad \hat{\sigma}_d^2 = \frac{\widehat{\text{Var}}(R_{\Delta t}^D)}{\Delta t}, \quad (6)$$

donde  $\hat{E}(R_{\Delta t}^D)$  y  $\widehat{\text{Var}}(R_{\Delta t}^D)$  son la media y varianza muestral de los retornos logarítmicos empíricos de la primera base de datos compuesta únicamente por las observaciones que no fueron filtradas por el proceso iterativo de Cartea y Figueroa (2005).



## 7 Datos

Se utilizarán datos del par “BTC/USDT” que corresponde a la cotización del Bitcoin valuado en una criptomoneda indexada al valor del dólar norteamericano. Esto último es la valuación más cercana a la tradicional y no implica consideraciones adicionales más allá de cuestiones operativas propias de este mercado. La información podría descargarse manualmente de distintos portales, pero para contar con la mayor información posible, se desarrolló una interfaz de programación de aplicaciones (API, por sus siglas en inglés). La API descarga información de forma ilimitada de Binance, la plataforma de intercambio de criptomonedas con mayor volumen y número de usuarios del mundo. Se adjuntan extractos de los módulos más importantes de este desarrollo en el Anexo.

Como la información es descargada naturalmente en intervalos de un minuto, se desarrolló adicionalmente un módulo especial que agrupa las operaciones en cualquier intervalo que el usuario desee y lo expone en formato de velas japonesas (“velas” de ahora en más). En este trabajo, sólo se usarán los cuatro precios que forman a una vela: precio de apertura, precio de cierre, precio máximo y precio mínimo; además del volumen operado. Este código también se adjuntará en el Anexo.

Trabajar con información de criptomonedas ofrece la ventaja de contar con información sin cortes debido a su negociación continua. Este factor ahorra la necesidad de considerar el valor tiempo del dinero para decidir si mantenerse en el mercado los fines de semana. Por otro lado, en este mercado en particular existe la tendencia de compartir información abiertamente, así que cuenta con una gran cantidad de información disponible y la mayoría de *brokers* la ofrecen de forma pública y gratuita. Si bien no todos los activos cuentan con el mismo nivel de liquidez, siendo algunos considerablemente ilíquidos, se utilizará el activo más líquido del mercado, superando incluso a muchos instrumentos del mercado tradicional. Por último, si bien podría haberse elegido cualquier compresión temporal de la información, se optó por usar la compresión de cuatro horas ya que, en base a experiencia propia, se ha entendido que ofrece un balance entre el ruido de las bajas temporalidades y la tardanza de reacción de las temporalidades superiores.

Por último, se utilizarán datos que irán desde el 1 de diciembre del 2020 hasta el 1 de enero del 2022. Aunque en realidad se estudiará únicamente el año 2021 completo, pero se adiciona información para asegurar que se puedan calcular los indicadores usados en las estrategias de operatoria algorítmica. La elección del año a estudiar resulta conveniente para focalizar el estudio en el desempeño computacional de los modelos de optimización y no en los retornos anormales obtenidos por la estrategia de trading en sí, ya que en período seleccionado resulta evidente que la elección de una estrategia tendencial sería una buena decisión. En otros casos se debería alternar entre estrategias tendenciales y estrategias de rango para acompañar al mercado en su evolución, ya que las primeras no se desempeñan para nada bien en escenarios en donde las segundas son adecuadas.

**Tabla N°1: Muestra de datos**

Index	open	close	high	low	volume	exchange	pair
2020-12-01T00:00:00.000000000	19695.9	19419.7	19720	19340	12570.9	binance	BTC/USDT
2020-12-01T04:00:00.000000000	19419.7	19515.6	19546.8	19281.4	12160.7	binance	BTC/USDT
2020-12-01T08:00:00.000000000	19515.6	19425	19888	18886	28164.4	binance	BTC/USDT
2020-12-01T12:00:00.000000000	19425.4	19263.4	19489.3	18001.1	46986.2	binance	BTC/USDT
2020-12-01T16:00:00.000000000	19263.4	19069.8	19325.8	18611.9	18487.2	binance	BTC/USDT
2021-12-31T04:00:00.000000000	47201.8	47191.1	47550	46836.9	3618.28	binance	BTC/USDT
2021-12-31T08:00:00.000000000	47191.1	47975.8	48548.3	47065.3	8542.25	binance	BTC/USDT
2021-12-31T12:00:00.000000000	47975.8	48005.4	48170	47830.6	4864.55	binance	BTC/USDT
2021-12-31T16:00:00.000000000	48005.4	45728.3	48118.4	45679	9090.79	binance	BTC/USDT
2021-12-31T20:00:00.000000000	45728.3	46216.9	46570	45678	5543.96	binance	BTC/USDT

*Fuente: Elaboración con código propio y con datos de Binance.*

**Tabla N°2: Descripción de los retornos logarítmicos**

count	mean	std	min	25%	50%	75%	max
2375	0.000365076	0.0183243	-0.111249	-0.00809529	0.000578677	0.00907505	0.137577

*Fuente: Elaboración con código propio y con datos de Binance.*

## 8 Experimento

Se deberán cumplimentar varios pasos antes de tener toda la información necesaria para ejecutar el experimento. Antes de estar listo para ejecutar el algoritmo de optimización se necesitara:

1. Obtener los datos sobre precios y volumen de negociación,
2. Procesamiento de datos para poder consumirlos,
3. Separar el periodo en muestras,
4. Analizar los datos y generar las simulaciones, y
5. Ejecutar el algoritmo de optimización con sus variaciones.

### 8.1 Búsqueda de datos

Se programó una API para obtener toda la información para el experimento. El *endpoint* estará direccionado al *broker*, que en este caso será Binance. Se hará una *request* del tipo *get* a una base de datos en donde se almacenan los precios HOCL y el volumen operado en cada minuto, en donde precios HOCL se refiere al precio máximo, de apertura, de cierre y mínimo.

El *ticker* será "BTC/USDT" y se utilizarán datos que irán desde el 1 de Diciembre del 2020 hasta el 1 de Enero del 2022. Aunque en realidad se estudiará únicamente el año 2021 completo, pero se adiciona información para asegurar que se puedan calcular los indicadores usados en las estrategias

de operatoria algorítmica. La elección del año a estudiar resulta conveniente para focalizar el estudio en el desempeño computacional de los modelos de optimización y no en los retornos anormales obtenidos por la estrategia de trading en sí, ya que en período seleccionado resulta evidente que la elección de una estrategia tendencial sería una buena decisión. En otros casos se debería alternar entre estrategias tendenciales y estrategias de rango para acompañar al mercado en su evolución, ya que las primeras no se desempeñan para nada bien en escenarios en donde las segundas son adecuadas.

En este sentido, se asumirá que el mercado puede comportarse direccionalmente o de manera lateral. En estos términos, la hipótesis podría ser estudiada con dos posibles configuraciones. La primera sería tomar un período de tiempo en donde los precios se comportaron de manera lateral y aplicar una estrategia de negociación del estilo reversión a la media. La segunda sería elegir un período con precios del estilo direccional y aplicar estrategias del estilo tendencial. Para este trabajo, se elegirá un período que tenga evidente direccionalidad para aplicar una estrategia de estilo tendencial.

## **8.2 Procesamiento de datos**

Una vez descargada la información por minuto, se programó un método especial para procesar la información y condensarla en el intervalo correspondiente. Luego se programaron varios otros métodos específicos para consumir la información más fácilmente. En todos estos, se tendrá como variables de entrada al *ticker* del par, el bróker, las fechas de comienzo y final de la consulta, y la compresión temporal de cada observación.

La compresión horaria se refiere a la cantidad de minutos que poseerá cada registro en formato de vela japonesa. En este trabajo se utilizará una compresión de 240 minutos para formar así registros de cuatro horas. Cada registro contará el total de volumen operado durante esas cuatro horas, el primer y último precio operados en ese intervalo y los precios máximos y mínimos registrados. La decisión responde a experiencia en la utilización de esta información que ha concluido en la preferencia de este tipo de compresión, ya que compresiones inferiores conlleva trabajar con mucho ruido de mercado y compresiones superiores implican una considerable lentitud en la reacción de los algoritmos para aprovechar las oportunidades del mercado.

## **8.3 Separación de muestras**

Una vez obtenida la información, se dividirán las 52 semanas del año de optimización cada 15 días para formar así una lista de 26 tablas en donde se harán las optimizaciones. Esto responde a que en la práctica, los modelos de operatoria algorítmica se ajustan regularmente. La decisión de ajustar los parámetros con esta frecuencia es, en cierta medida, arbitraria y analizar si conviene hacerlo más o menos regularmente podría representar un trabajo de investigación adicional ya que es otro factor ignorado normalmente en la literatura y afecta regularmente el rendimiento de los modelos. De hecho, en la práctica real se suelen implementar *triggers* que disparen la parametrización incluso si no se ha cumplido el plazo preestablecido ya que se pueden detectar cambios repentinos en el mercado.

## **8.4 Análisis de datos y generación de simulaciones**

El paso siguiente es seleccionar una de las tablas con información quincenal y generar las cincuenta simulaciones que conformarán el conjunto  $S^*$  que será usado en todas las estrategias. Para poder generar los dos procesos estocásticos que conformarán las difusiones con saltos al estilo

Merton, se seguirá el método planteado por Cartera y Figueroa (2005), en donde se filtran las variaciones anormales a través de un filtro de detecta aquellos que excedan tres desvíos estándares. Además, cabe destacar que se generan simulaciones de la tabla completa, incluyendo precios de apertura, de cierre, máximos y mínimos. También se simula el volumen de operatoria en cada observación. Sobre estas simulaciones se ejecutarán el optimizador bayesiano utilizando las tres funciones de adquisición estudiadas en este trabajo.

### **8.5 Ejecución del optimizador**

Una vez lista la matriz de simulaciones  $S^*$ , se aplicará el algoritmo de optimización bayesiano en cada una de ellas permitiéndole iterar unas cien veces para alcanzar la cesta óptima de parámetros para así maximizar la estrategia de operatoria algorítmica. Se considerarán costos de transacción equivalentes al 0.01% para no subestimar los costos reales de realizar exceso de transacciones.

Este proceso iterativo se repetirá para cada una de las tres funciones de adquisición estudiadas en este trabajo y además se estudiarán seis estrategias de operatoria algorítmica diferentes para estudiar cómo afectan estos factores en configuraciones diferentes, intentando demostrar cómo afecta la omisión de este estudio en la práctica real.

Resulta importante aclarar que se utilizarán los parámetros óptimos de la muestra  $m$  en la muestra  $m + 1$ . Es decir, los parámetros óptimos de la primera sección, se usarán en la segunda sección y será éste rendimiento el que se evaluara en última instancia. De este modo, se respetará el modo en que se hubieran usado en la práctica real, ya que se optimizaría el modelo en base al comportamiento del mercado del período anterior, ya que el actual se encuentra inconcluso.

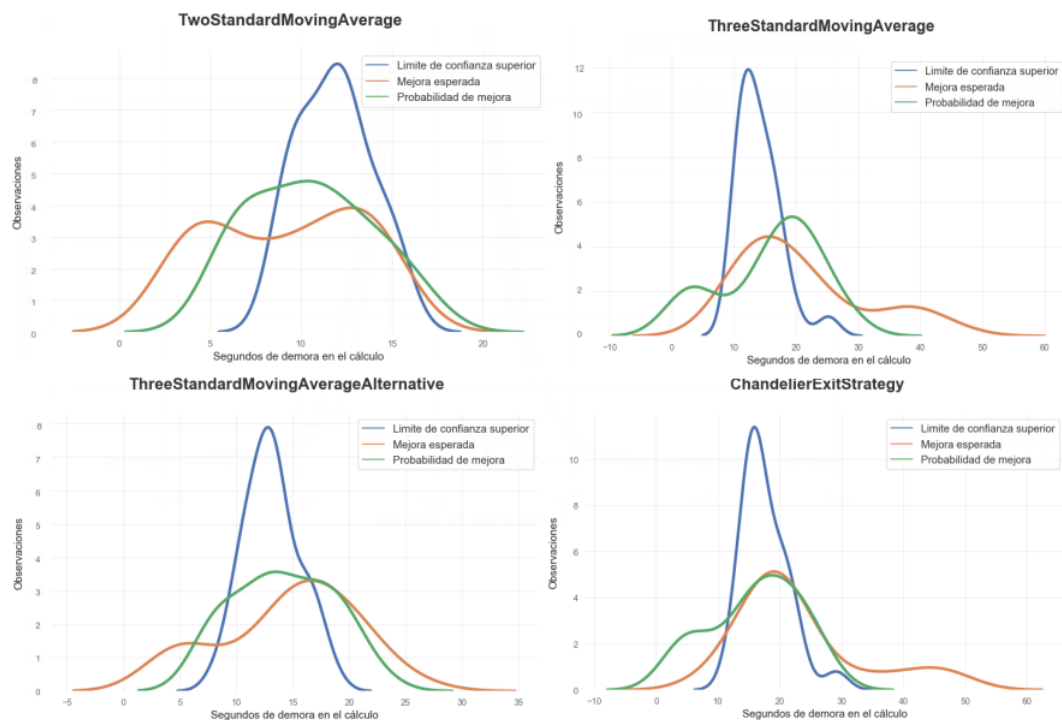
## **9 Resultados**

### **9.1 Demora computacional**

Si bien el trabajo del optimizador demora entre unos diez y veinte segundos en alcanzar la cesta óptima de parámetros de una serie temporal determinada, en este experimento el tiempo resultó mucho mayor. Recorrer todas las simulaciones fue el primer desafío ya que el tiempo de procesamiento se incrementó sustancialmente. Además, dado que el objetivo del trabajo es comparar el comportamiento de tres funciones de adquisición diferentes, esta problemática se terminó triplicando. De aquí se pudo extraer la primera conclusión y observar que las diferentes funciones de adquisición tienen un promedio de demora significativamente diferente.

Mientras que la función de adquisición de límite de confianza superior ha tenido resultados bastante homogéneos en términos de demora de los cálculos, las funciones de mejora esperada y probabilidad de mejora han tenido resultados bastante dispersos. La primera ha tenido resultados muy concentrados en torno a su media lo que implicaría cierta predictibilidad de su demora computacional. Por otra parte, las otras dos han tenido resultados a veces más rápidos que la anterior pero en otras ocasiones con mucho más demora. Esto terminaría implicando que, a pesar de una posible mejora en cuanto a actuación, las funciones de adquisición de mejora esperada y probabilidad de mejora implican una imprevisibilidad en la demora de sus cálculos ya que los resultados han sido muy heterogéneos.

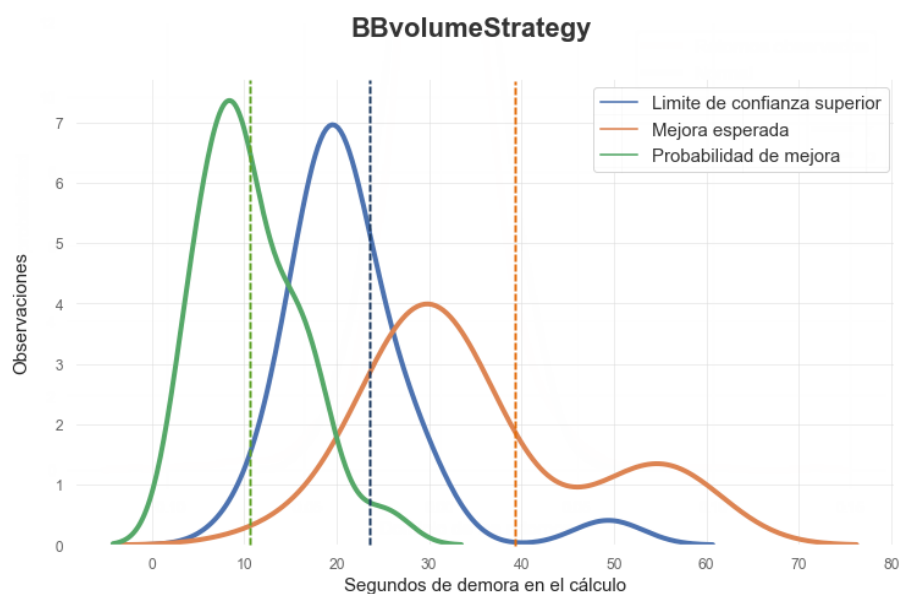
## Gráficos N°4-7: Dispersiones de demora computacional



*Fuente: Elaboración propia*

La homogeneidad de demora en los cálculos de la función de límite de confianza superior no necesariamente implica que esta función de adquisición sea computacionalmente más eficiente, ya en varias estrategias se observó que la media de todas las funciones es muy similar. De hecho, en algunas estrategias resultó ser mayor, concluyendo que en algunas estrategias resultaría mejor utilizar alguna de las otras dos. Un ejemplo de esto se da en el gráfico de la estrategia *BBVolumeStrategy*, en donde la función de probabilidad de mejora tiene una demora de cálculo evidentemente inferior.

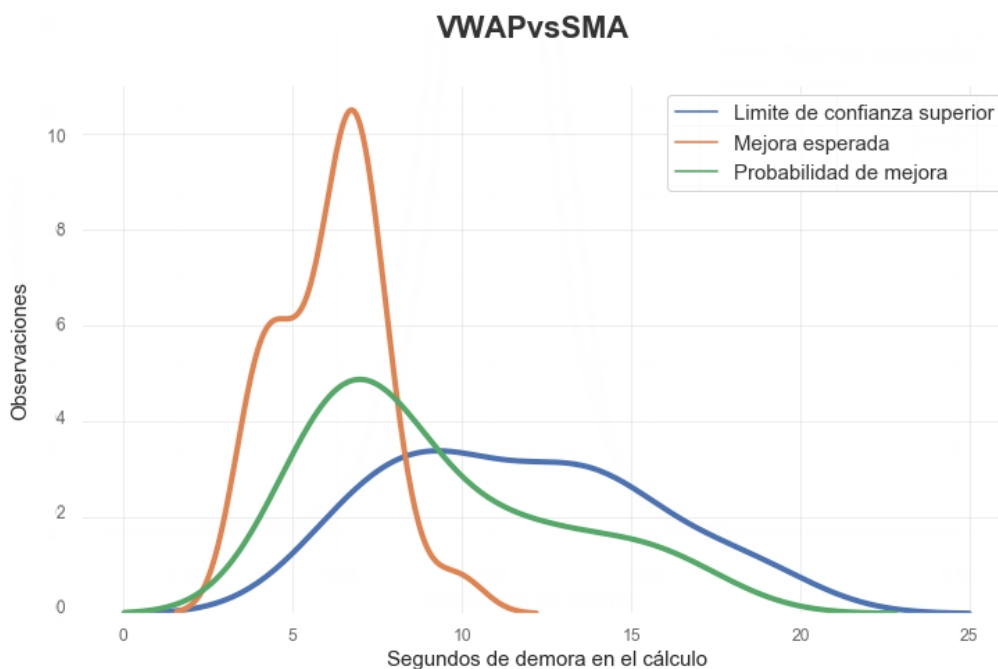
## Gráfico N°8: Comparación de medias de demora computacional



*Fuente: Elaboración propia*

La concentración de las demoras de la función de límite confianza superior y la dispersión de las otras dos se ha mantenido en casi todas las estrategias estudiadas. Sin embargo, se pudo observar que en la estrategia *VWAPvsSMA* se presentó un caso particular. En esta estrategia la función de mejora esperada tuvo resultados más concentrados en torno a su media y las otras dos tuvieron resultados más dispersos. Fue la única estrategia en donde la función de límite de confianza superior no tuvo una homogeneidad de resultados en cuanto al análisis de demora respecta.

### Gráfico N°9: Excepción en demora computacional



*Fuente: Elaboración propia*

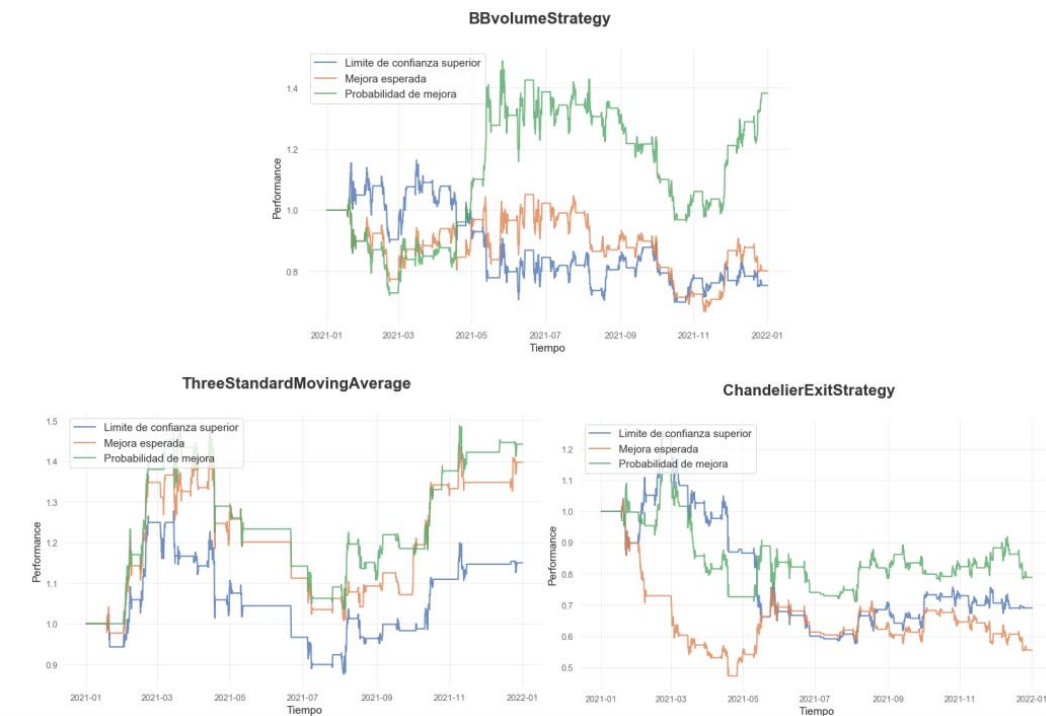
Otro desafío que implicó el experimento es que en la práctica los parámetros se optimizan una sola vez en base al período que acaba de terminar. En este experimento, al hacerse un *backtesting* de todo un año se incurrió en la necesidad de evaluar una cantidad significativamente mayor de períodos. Aquí, la división de periodos cada dos semanas fue totalmente arbitraria, pero se observa que la decisión del usuario repercute en gran medida en la demora del experimento.

## 9.2 Rendimiento obtenido

En cuanto al rendimiento de las diferentes funciones de adquisición, se pudo observar que han tenido resultados muy diferentes. Estos resultados son los de mayor relevancia ya que la diferencia que han tenido las tres funciones de adquisición son sustanciales. Cabe recordar que todas fueron optimizadas con las mismas simulaciones, por lo que el factor de aleatoriedad de los parámetros de entrada no ha influido en esta discrepancia.

La función de adquisición que ha tenido mejores resultados ha sido la de probabilidad de mejora, ya que en tres de las estrategias analizadas ha demostrado obtener mejores resultados que las otras dos. Si bien en general no se ha diferenciado en gran medida de las otras, en la estrategia *BBvolumeStrategy* ha tenido una diferencia de actuación de aproximadamente un 60%. De hecho fue la única función de adquisición que ha tenido resultados positivos con esa estrategia. Este caso en particular es una gran evidencia que la correcta parametrización del experimento podría implicar diferencias significativas en el modelo de operatoria algorítmica.

## Gráficos N°10-12: Rendimiento de Probabilidad de Mejora

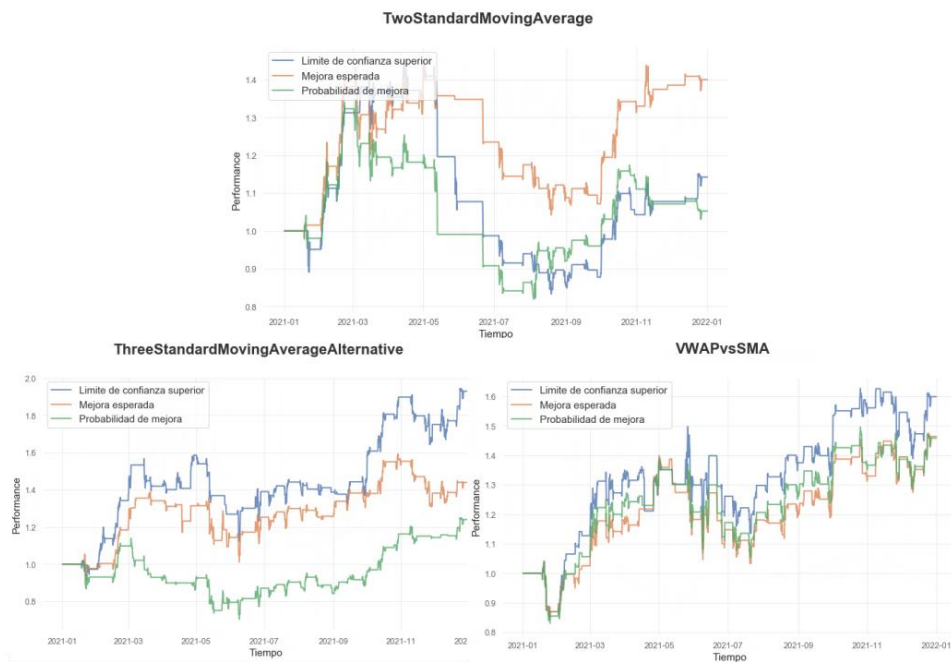


***Fuente:** Elaboración propia*

La segunda función de adquisición en orden de mejores resultados fue la de límite de confianza superior. Esta función, además de haber presentado una mayor homogeneidad de demoras en sus cálculos, ha destacado en actuación en dos de las seis estrategias y en una de ellas ha podido destacar con casi un 50% de actuación acumulada al final del período analizado. En la otra estrategia no ha tenido diferencia significativa con el resto de las funciones de adquisición. En último lugar se ubica la función de mejora esperada, que a pesar de haber destacado en una sola estrategia, ha podido superar la función de probabilidad de mejora por casi un 35%.

Resulta interesante observar también la dispersión de resultados de la función de probabilidad de mejora. Si bien ha tenido actuación sobresaliente en tres de las seis estrategias, también ha sido la de peor retorno acumulado en la misma cantidad de estrategias. Esto parece indicar que cada estrategia de operatoria algorítmica podría tener una función de adquisición óptima y que en base a un experimento como este se podría parametrizar su optimización en base a la función de adquisición que se adecue mejor.

## Gráfico N°13-15: Rendimiento de funciones de adquisición



*Fuente: Elaboración propia*

## 10 Conclusiones

Este trabajo buscó ejecutar variaciones en la función de adquisición de un método de optimización global y comparar sus resultados en el ámbito de operatoria algorítmica. Se buscó estudiar la existencia de diferencias estadísticamente significativas en su rendimiento acumulado al finalizar el período y en su tiempo de procesamiento. En particular, se estudiaron variaciones en las funciones de adquisición de la optimización bayesiana aplicada a la optimización de seis estrategias de operatoria algorítmica.

Los resultados han confirmado que la parametrización, en especial la elección de la función de adquisición, genera variaciones significativas en el rendimiento del algoritmo y también genera una variación en la tardanza y consumo de recursos computacionales del proceso de optimización. La literatura en torno a la negociación cuantitativa se centra principalmente en el desarrollo de estrategias de negociación capaces de ganarle al mercado de forma recurrente pero ignoran totalmente otras decisiones que deben tomar los profesionales de las finanzas en su día a día y se los deja improvisados de recursos para tomar decisiones, las cuales terminan siendo arbitrarias y posiblemente ineficiente.

La primera gran evidencia encontrada fue la diferencia en los requerimientos computacionales de las diferentes funciones de adquisición ya que la omisión de la correcta parametrización de esta variable dentro de un modelo de operatoria algorítmica conlleva una diferencia significativa en el rendimiento del modelo completo. La segunda evidencia es que el rendimiento acumulado en un año completo es diametralmente diferente según la función de adquisición seleccionada, llegando incluso a significar tener un rendimiento negativo cuando se podría haber obtenido otro resultado.

Es interesante observar que no hay una elección inequívoca y se debe analizar el caso en particular, ya que cada una de las seis estrategias analizadas ha presentado resultados dispares entre



sí. Si bien la función de adquisición de probabilidad de mejora ha tenido resultados excelentes tanto en rendimiento como en demora computacional en la estrategia *BBvolumeStrategy*, esta misma función ha tenido resultados negativos en otras estrategias. La elección de este parámetro, sin un análisis detallado del caso en particular a optimizar significará una variación significativa en el rendimiento de la estrategia, su demora computacional y, en última instancia, la rapidez en la que el operador podría ajustar su modelo.

Con estos resultados, hay evidencia para decir que en general las estrategias de negociación algorítmica están lejos de ser infalibles, ya que en general son programadas de forma ad-hoc. De esta forma tienen un buen rendimiento durante el período de entrenamiento pero en la práctica dependen más bien del azar. En definitiva, al menos con los resultados de este trabajo, parece que las estrategias de negociación algorítmicas son heurísticas ya que resultan de un procedimiento que intenta producir soluciones buenas a un problema de forma rápida pero que carece de garantías teóricas.

## 11 Referencias

<https://gitlab.com/fcucullu/tesis>

Bergstra, J., & Bengio, Y. (2012). *"Random search for hyper-parameter optimization"*. Journal of machine learning research, 13(2).

Bergstra, J., Bardenet, R., Bengio, Y., & Kégl, B. (2011). *"Algorithms for hyper-parameter optimization"*. Advances in neural information processing systems, 24.

Bull, A. D. (2011). *"Convergence rates of efficient global optimization algorithms"*. Journal of Machine Learning Research, 12(10).

Cartea, A. y Figueroa, M (2005). *"Pricing in electricity markets: a mean reverting jump diffusion model with seasonality"*. Applied Mathematical Finance, 12 (4).

Gelbart, M. A., Snoek, J., & Adams, R. P. (2014). *"Bayesian optimization with unknown constraints"*. ArXiv preprint arXiv:1403.5607.

Kushner, H. J. (1964). *"A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise"*. Cambridge, MA: MIT press.

Lizotte, D. J., Wang, T., Bowling, M. H., & Schuurmans, D. (2007). *"Automatic Gait Optimization With Gaussian Process Regression"*. In IJCAI (Vol. 7, pp. 944-949).

Merton, R. C. (1976). *"Option pricing when underlying stock returns are discontinuous"*. Journal of Financial Economics, 3, 125-144.

Mockus, J. (1975). *"On the Bayes methods for seeking the extremal point"*. IFAC Proceedings Volumes, 8(1), 428-431.

Mockus, J., Tiesis, V. & Zilinskas, A. (1978). *"The application of Bayesian methods for seeking the extremum"*. Towards Global Optimization, 2:117-129.

Rasmussen, C. E. & Williams, C. K. (2006). *"Gaussian processes for machine learning"* (Vol. 2, No. 3, p. 4). Cambridge, MA: MIT press.

Snoek, J., Larochelle, H., & Adams, R. P. (2012). *"Practical bayesian optimization of machine learning algorithms"*. Advances in neural information processing systems, 25.

Srinivas, N., Krause, A., Kakade, S. M., & Seeger, M. (2009). *"Gaussian process optimization in the bandit setting: No regret and experimental design"*. ArXiv preprint arXiv:0912.3995.

Tang, F. (2018). *"Merton Jump-Diffusion. Modeling of Stocks Price Data"*. Degree proyect, Linnaeus University, Suecia.