Optimización de carteras sobre computación cuántica

DANEL ARIAS - APLICACIONES DE CIENCIAS DE DATOS Y TECNOLOGÍAS INTELIGENTES

Finanzas – Trabajo Final

Contenido

Optimización de carteras

La optimización de la cartera es el proceso de seleccionar la mejor cartera (distribución de activos), del conjunto de todas las carteras consideradas, de acuerdo con algún objetivo. El objetivo suele maximizar factores como la rentabilidad esperada y minimizar costes como el riesgo financiero, lo que da lugar a un problema de optimización multiobjetivo

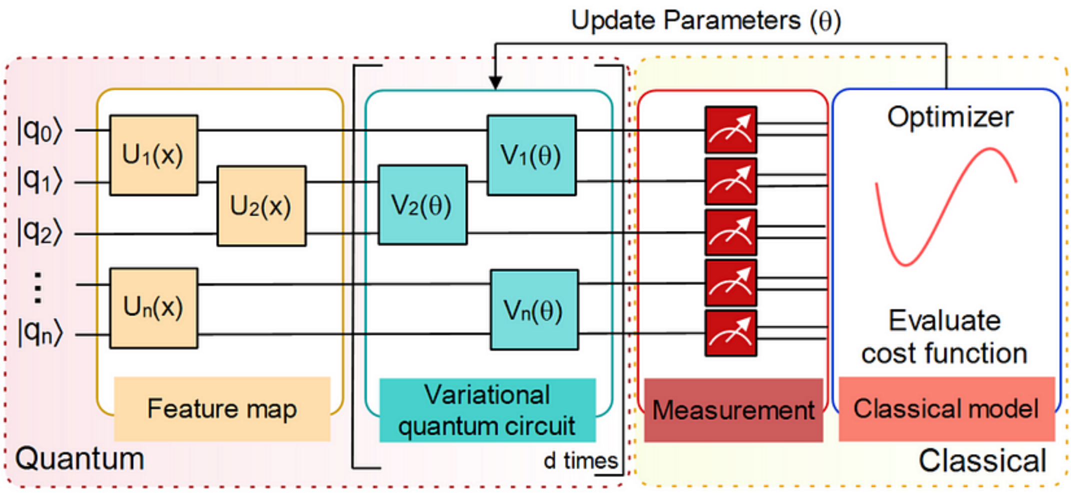
La teoría moderna de carteras fue introducida por Harry Markowitz en su tesis doctoral de 1952, en la que se definió por primera vez el modelo de Markowitz, que parte del supuesto de que un inversor trata de maximizar la rentabilidad esperada de una cartera en función de un determinado nivel de riesgo. Las carteras que cumplen este criterio, es decir, que maximizan la rentabilidad esperada dada una cantidad de riesgo prescrita.

El problema de optimización de la cartera se especifica como un problema de maximización de la utilidad con restricciones. Las formulaciones habituales de las funciones de utilidad de la cartera la definen como el rendimiento esperado de la cartera (neto de costes de transacción y financiación) menos un coste de riesgo.

Algoritmo cuántico

Para resolver este problema se utilizará un algoritmo variacional, en concreto VQE (Variational Quantum Eigensolver). Los algoritmos cuánticos variacionales son un enfoque mixto cuántico-clásico en el que el procesador cuántico prepara los estados cuánticos y se realizan las mediciones y la optimización corre a cargo de un ordenador clásico. En concreto para el VQE dado un ansatz, el procesador cuántico calcula el valor esperado del sistema con respecto a un observable, a menudo el hamiltoniano, y se utiliza un optimizador clásico para mejorar el ansatz. El algoritmo se basa en el método variacional de la mecánica cuántica.

El ansatz, por lo tanto, consta de un circuito parametrizado (ansatz), un optimizador clásico y un backend cuántico que permita la evaluación de circuito sobre el operador de coste.



(Imagen extraída de https://www.researchgate.net/publication/351813135_Variational_Quantum_Classifiers_Through_the_Lens_of_the_Hessian/figures?lo=1)

Qiskit Finance proporciona la siguiente construcción para resolver el siguiente problema de optimización de cartera de *media-varianza* para *n* activos:

$$\min_{x\in\{0,1\}^n} q\, x^T\Sigma x - \mu^T x$$

$$\text{sujeto a} : 1^T x = B$$

donde se usa la siguiente notación:

- $x \in \{0,1\}^n$ denota el vector de variables de decisión binarias, que indican qué activos elegir ($x[i] = 1$) y cuáles no elegir ($x[i] = 0$)
- $\mu \in R^n$ define el retorno esperado de los activos
- $\Sigma \in R^{n\times n}$ especifica las covarianzas entre los activos
- $q > 0$ controla la propensión al riesgo del responsable de la toma de decisiones
- *B* designa el presupuesto, es decir, el número de activos que deben seleccionarse de entre *n*

La restricción de igualdad $1^T x = B$ se asigna a un término de penalización $(1^T x - B)^2$ que se escala mediante un parámetro y se resta de la función objetivo. El problema resultante puede asignarse a un Hamiltoniano cuyo estado fundamental corresponde a la solución óptima.

De esta manera construimos el operador de coste que tras minimizar nos indicaría la solución óptima al problema presentado.

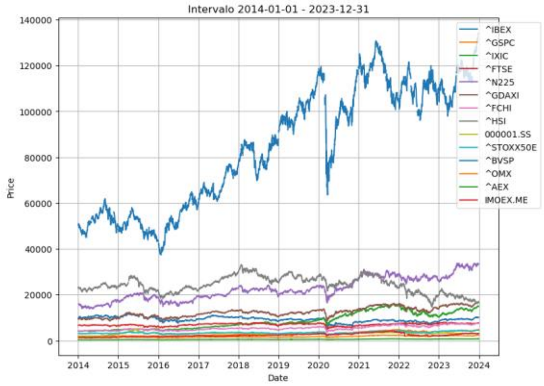
Experimento planteado

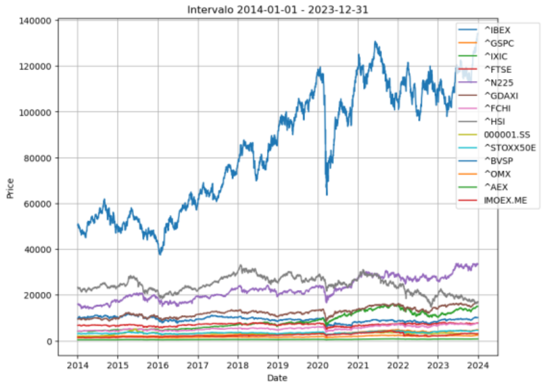
Datos seleccionados

Los datos para este proyecto constan de los precios de 14 índices bursátiles en la última década. En concreto se ha utilizado el periodo iniciando el 1 de enero de 2014 y finalizando el 31 de diciembre de 2023. Los datos han sido extraídos sobre la API de Yahoo Finance. Los índices seleccionados son los siguientes:

- IBEX 35 (^IBEX): Índice bursátil de referencia de la bolsa española.
- S&P 500 (^GSPC): Índice bursátil de referencia de la bolsa estadounidense.
- Nasdaq (^IXIC): Índice bursátil de referencia de la bolsa tecnológica estadounidense.
- FTSE 100 (^FTSE): Índice bursátil de referencia de la bolsa británica.
- Nikkei 225 (^N225): Índice bursátil de referencia de la bolsa japonesa.
- DAX (^GDAXI): Índice bursátil de referencia de la bolsa alemana.
- CAC 40 (^FCHI): Índice bursátil de referencia de la bolsa francesa.
- Hang Seng Index (^HSI): Índice bursátil de referencia de la bolsa de Hong Kong.
- Shanghai Composite (000001.SS): Índice bursátil de referencia de la bolsa china.
- Eurostoxx 50 (^STOXX50E): Índice bursátil de referencia de la bolsa europea.
- Bovespa (^BVSP): Índice bursátil de referencia de la bolsa brasileña.
- OMX Stockholm 30 (^OMX): Índice bursátil de referencia de la bolsa sueca.
- AEX (^AEX): Índice bursátil de referencia de la bolsa holandesa.
- MOEX (IMOEX.ME): Índice bursátil de referencia de la bolsa rusa.

Tras la extracción de las series temporales, encontramos que hay datos faltantes, para ello se ha realizado un tratamiento de interpolación lineal entre los valores conocidos para completar la serie.



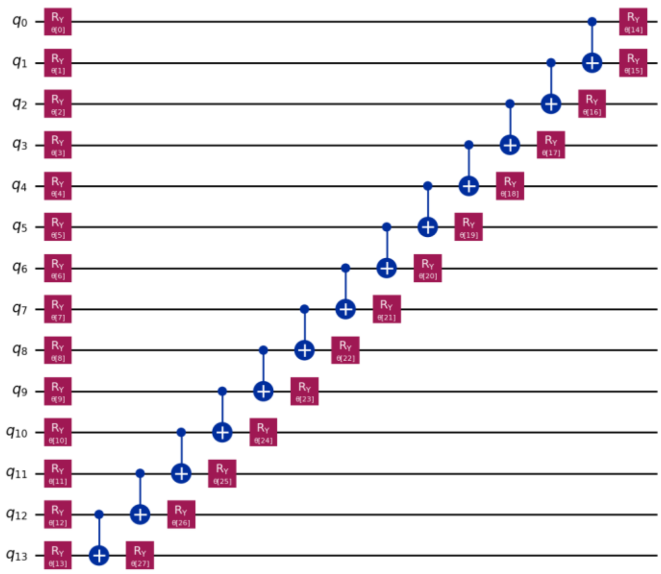


A la izquierda observamos los datos extraídos y a la derecha los datos tras la interpolación.

Configuración del algoritmo cuántico

Para proceder, es necesario establecer la configuración usada para resolver el problema de optimización propuesto. En concreto, al tratarse de un algoritmo variacional, se requiere especificar el ansatz elegido (el circuito cuántico a optimizar), el optimizador clásico utilizado y el backend cuántico.

Como ansatz, por simplicidad para este proyecto, se ha decidido utilizar un Real Amplitudes con una sola repetición extendido a 14 qubits (los necesarios para 14 activos). Este circuito presenta las características necesarias como puertas uni-qubit parametrizadas al igual que puertas de entrelazamiento a los primeros vecinos para poder optimizar no solo la influencia de cada activo sino la interacción entre estos. Usualmente se suele diseñar un circuito específico que se ajuste de mejor manera a las necesidades del problema y la estructura y forma de los datos de este.



Para la optimización se ha decidido utilizar COBYLA (*Constrained Optimization BY Linear Approximation*) con un máximo de iteraciones de 200. Este optimizador, suele obtener buenos resultados, siendo el recomendado por Qiskit en varios tutoriales de la documentación, aunque, al igual que con el ansatz, lo ideal sería probar diferentes optimizadores y evaluar los resultados, la velocidad y la convergencia de estos para los diferentes problemas.

Por último, toca decidir el backend cuántico donde se hará la evaluación del circuito. IBM ofrece de manera gratuita el acceso a su hardware cuántico a través de la nube. Sin embargo, para cada evaluación, dado que esos sistemas están demandados, hay que esperar colas que pueden llegar a 7h. Una pequeña multiplicación, asumiendo que se realizan las 200 iteraciones del optimizador nos indica que es inviable. Por esto se ha decidido utilizar simulador local para la optimización del circuito. Esto elimina los problemas de ruido y errores que tiene el hardware real, lo cual simplifica el análisis de los resultados.

Evaluación y partición de la serie temporal

Dado que el problema propuesto estudia la optimización de forma estática, se ha decidido realizar un estudio seccionando las series temporales por periodos para ver la evolución en cuanto a la cartera optimizada.

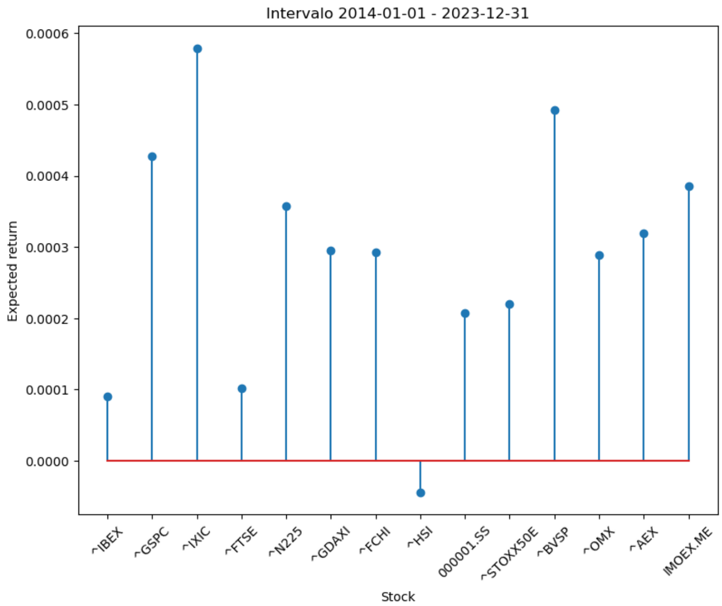
Lo ideal, sería poder realizar un estudio sobre periodos más cortos y que la optimización dependa directamente de la inversión anterior (el estado anterior de la cartera). Esto convertiría el problema en un problema de Optimización de Carteras Dinámico, el cuál implica un cambio en el operador de coste evaluado y otros ajustes que no se discutirán en este proyecto pero que la comunidad científica está investigando.

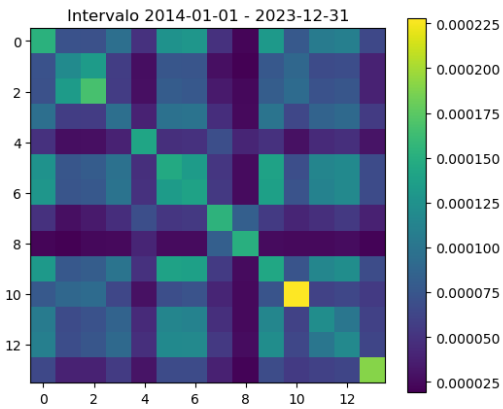
Para este proyecto se realizará la optimización con el periodo completo, con el periodo separado anualmente, en secciones de 6 meses y en sección cuatrimestrales.

Construcción del problema cuadrático

Para construir el operador de coste que se evaluará como función objetivo a optimizar, es necesario construir el problema cuadrático correspondiente primero. Este, tal y como se indica en la sección de algoritmo cuántico se construye recibiendo (1) El vector medio de retorno, (2) La matriz de covarianza del retorno, (3) El factor de apetito al riesgo y (4) El presupuesto.

El vector medio de retorno y la matriz de covarianza son extraíbles de los datos iniciales, por lo que queda definir el factor de riesgo y el presupuesto.





Para este proyecto se ha decidido utilizar un factor de riesgo de 0.5 y un presupuesto de 4. Esto indica que la optimización se hará teniendo en cuenta un riesgo moderado y que se hará una selección de 4 activos de los 14 presentados. Estos valores son más dependientes de la especificación del problema que de la optimización, pero sería razonable explorar otras vías con otros valores. Para este proyecto, sin embargo, se han establecido de manera arbitraria.

Con los datos segmentados, se construye un problema para cada segmento manteniendo el factor de riesgo y el presupuesto. En estrategias más avanzadas, no exploradas en este documento, se plantea el uso de diferentes valores en los diferentes segmentos, o construir una estrategia dinámica para estos.

Implementación con Qiskit y Python

Para la implementación de este proyecto se ha utilizado Qiskit, el lenguaje basado en Python de IBM para computación cuántica. En concreto, Qiskit proporciona un sub-paquete Qiskit Finance que proporciona herramientas para problemas de finanzas. Por un lado, se ha utilizado la herramienta YahooDataProvider para extraer los datos iniciales, ya que de esta manera se gestionan y ajustan fácilmente al resto de herramientas.

La herramienta, quizás más importante, es la clase PortfolioOptimization que ofrece el paquete de Qiskit Finance, ya que aporta una construcción automática del problema cuadrático, por tanto, el operador de coste, el vector medio de retorno, la matriz de covarianza del retorno, el factor de apetito al riesgo y el presupuesto.

Con esto, y utilizando en este caso las herramientas del sub-paquete Qiskit Optimization, podemos hacer uso de la clase VQE o SamplingVQE (una versión específica para el tipo de medición).

Posteriormente, la clase PortfolioOptimization ofrece funciones para la interpretación de los resultados, extrayendo el valor esperado y la varianza.

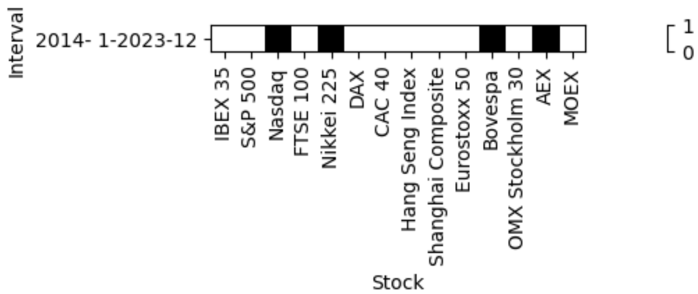
Todo el código utilizado se encuentra en un repositorio de GitHub (https://github.com/DanelArias-Dreyton257/Aplicaciones-Finance) donde se pueden observar la ejecución y los resultados del trabajo en los diferentes notebooks utilizados.

Resultados

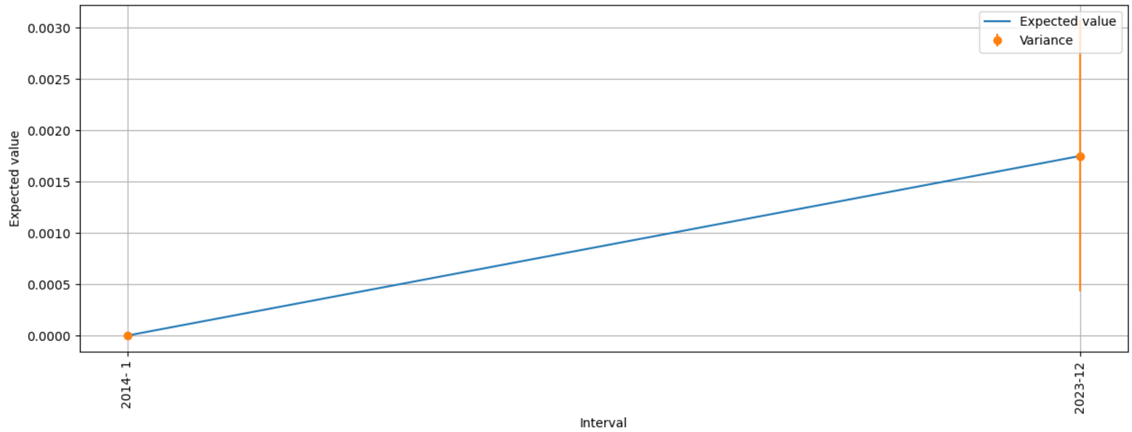
En este apartado se muestran los resultados de cada uno de los experimentos. Hay que tener en cuenta que los resultados obtenidos pueden no tener el resultado óptimo pues una optimización de 200 iteraciones puede no haber convergido. Sin embargo, aumentar el número de iteraciones se vuelve muy costosa ya que la simulación de circuitos cuánticos lo es.

Periodo completo

Para el caso usando el periodo completo de 10 años, la selección de activos tras la optimización es: ^IXIC, ^N225, ^BVSP y ^AEX con un valor esperado de 0.00175. Podemos visualizar la selección en el gráfico siguiente mostrando en negro los seleccionados y en blanco sin seleccionar. Es interesante observar cómo los activos seleccionados son algunos de los que mayor retorno medio tenían. Quedan de lado el S&P 500 y el índice MOEX, mientras que Nikkei 225 y AEX son seleccionados. Es posible que MOEX, especialmente parece tener más variabilidad por lo que es posible que el factor de riesgo haya sido decisivo en este caso.



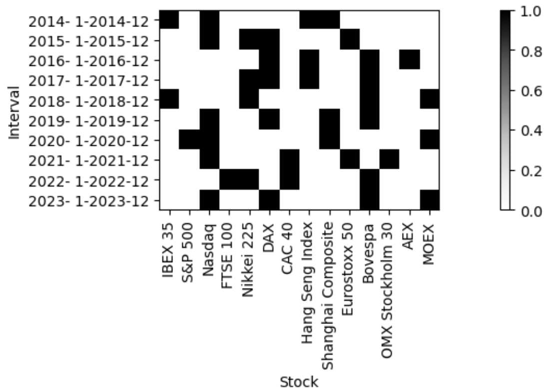
Podemos ver la progresión en cuanto a valor esperado para ese periodo, junto a la varianza asociada. Como podemos observar hay gran variabilidad.



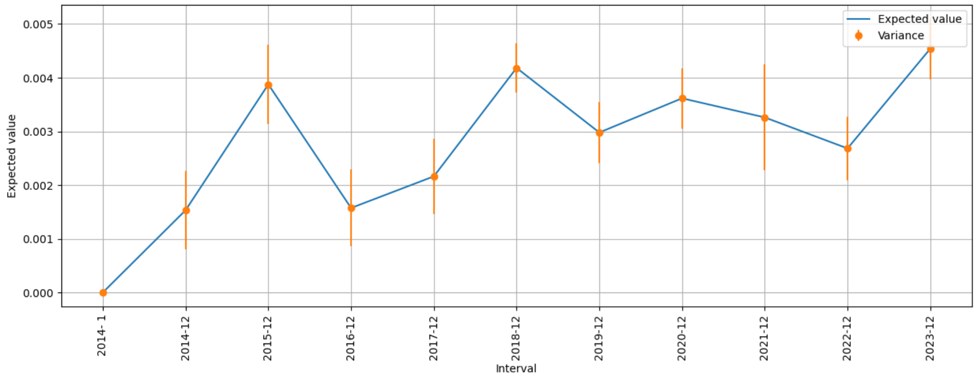
El valor esperado nos indica que, con una inversión de 1, obtendríamos un retorno de 1,002 aproximadamente.

Periodo anual

Para esta optimización se ha separado el data set por años y se ha realizado la optimización en cada caso. Podemos observar en la distribución elegida de activos seleccionados para cada año. Observamos que no hay ningún activo que se haya quedado al margen y no haya sido seleccionado en ninguno de los casos. Eso sí, los índices S&P 500, FTSE 100, OMX Stockholm 30 y AEX solo fueron seleccionados en un periodo cada uno. Es interesante observar también que los índices Nasdaq y Bovespa parecen ser los más seleccionados.



Si observamos la progresión del valor esperado para cada periodo, vemos que parece haber un ligero crecimiento pero que no supera el 0.005 anual. Observamos como incluso el valor esperado más pequeño de un periodo ya es superior al valor esperado por el caso anterior completo.



Observamos también el gráfico de valor esperado acumulado sobre una inversión inicial de 1.

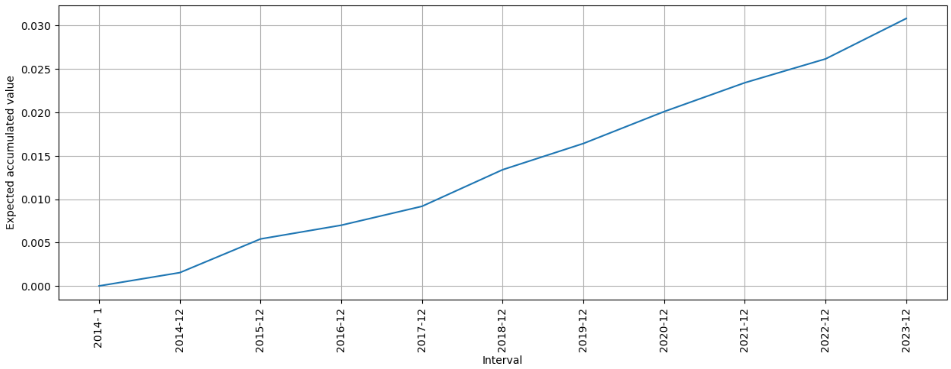
Este cálculo se realiza siguiendo:

$$Cumulative\ Value_n = Initial\ Investment \times \prod_{i=1}^{n} (1 + Expected\ Return_i)$$

Y por tanto el valor esperado acumulado:

$$Expected\ Return_n = Cumulative\ Value_n - Initial\ Investment$$

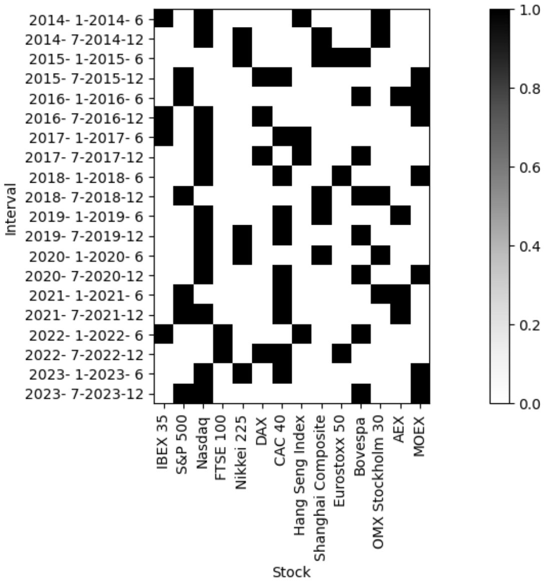
Para este caso el valor obtenido es cercano a 0.031. Este indicaría que, por esa inversión de 1, obtendríamos un retorno de 1,031 aproximadamente.



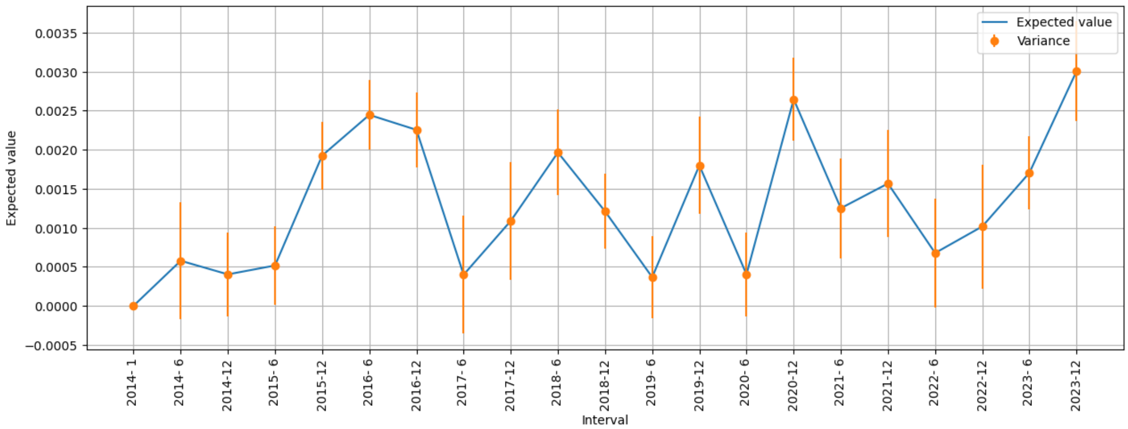
Este resultado curiosamente aumenta en un orden de magnitud el resultado obtenido por el estudio del periodo completo. Esto seguramente se deba a la capacidad de adaptarse al mercado y las fluctuaciones y variabilidades de este.

Periodo cada 6 meses

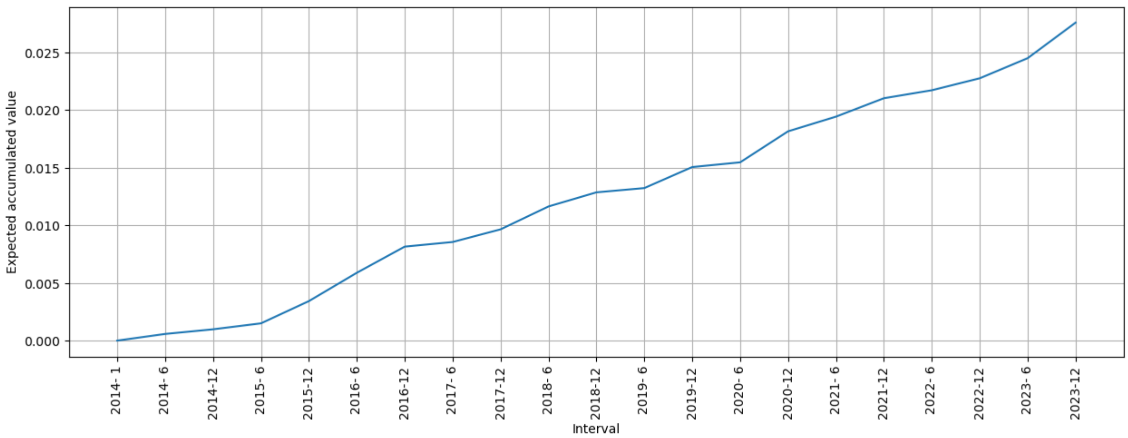
Para este caso se ha separado cada medio año, o sea, cada 6 meses. Observando la distribución de activos seleccionados, vemos ciertas preferencias. Por un lado, Nasdaq es el índice más seleccionado, siendo el elegido en la mayoría de los tramos. En cambio, el índice FTSE 100 es el menos seleccionado, siendo sólo escogido en 2022. El resto de la selección parece estar distribuida entre los demás activos.



Si observamos la progresión del valor esperado para cada periodo, vemos que parece haber periodos de solo 0.0005 de valor esperado mientras que en otros periodos se llega al 0.03. Y parece haber una ligera tendencia al crecimiento.



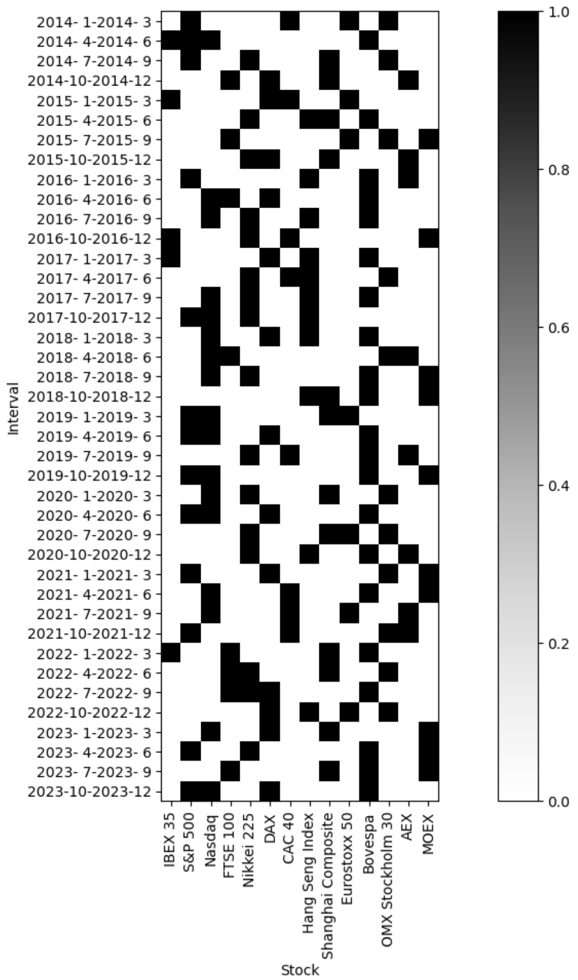
Observamos también el gráfico de valor esperado acumulado sobre una inversión inicial de 1. Para este caso el valor obtenido es cercano a 0.028. Este indicaría que, por esa inversión de 1, obtendríamos un retorno de 1,028 aproximadamente.



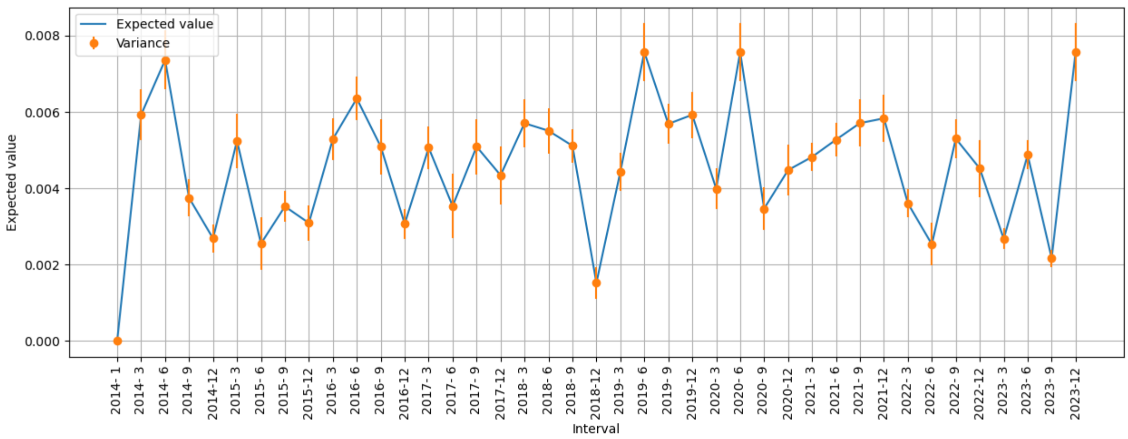
El resultado obtenido es similar al obtenido anteriormente en el caso anual, curiosamente no obtenemos una mejora, aunque es posible que esto sea dado por que la optimización no sea la óptima. También es cierto que el número de divisiones solo ha crecido en un factor de 2 frente a la división anterior, mientras que, en el anterior caso, fue un factor de 10.

Periodo cuatrimestral

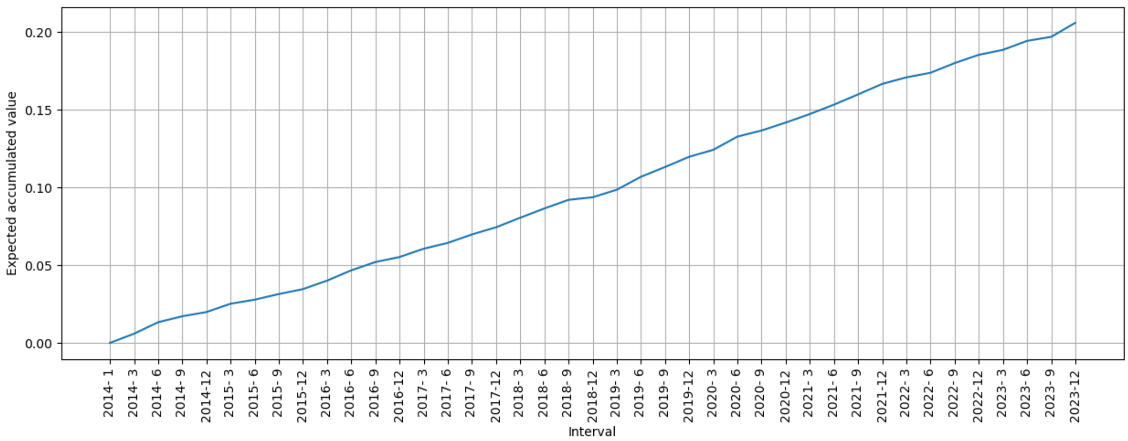
En este caso se ha separado por cuatrimestres, si observamos la distribución de activos seleccionados, vemos una gran variedad de ellos y no parece haber ningún activo que haya sido completamente ignorado. Observamos, eso sí, que durante largos periodos ciertos activos no son seleccionados, como, por ejemplo, el IBEX 35 desde 2017 hasta 2021, o el Shanghái Composite desde 2016 hasta finales del 2018. Otro activo que llama la atención es Bovespa que es quizá el más seleccionado durante todos los periodos.



Si observamos la progresión del valor esperado para cada periodo, vemos que aproximadamente se mantiene estable sobre 0,04 para cada periodo, no bajando del 0,02 pero en ocasiones llegando al 0,08. Es interesante observar cómo los valores en un solo periodo ya son mayores que los valores en los periodos más largos.



Observamos también el gráfico de valor esperado acumulado sobre una inversión inicial de 1. Para este caso el valor obtenido es cercano a 0.21. Este indicaría que, por esa inversión de 1, obtendríamos un retorno de 1,21 aproximadamente.



Este valor esperado supera con creces el esperado en la división cada medio año, creciendo en un orden de magnitud frente a ese. Siguiendo el mismo tren lógico de la partición anual sobre el periodo completo, este se deberá a la adaptabilidad que se consigue al dividir la serie temporal en extractos.

Conclusiones

Los resultados apuntan a que cuanto más pequeño sea el periodo estudiado, o sea que se realice la optimización de la cartera con mayor frecuencia, mayor es el retorno esperado acumulado. Sin embargo, estos datos acumulados solo tienen en cuenta el valor esperado y no su varianza. Esto último es importante porque esta se acumula de la misma manera en cuanto más periodos sea dividido el data set. Esto quiere decir que, los resultados finales también incrementan en gran medida su varianza, haciéndolos poco fiables.

Esto detalla algunas de las fallas de estos experimentos que se podrían mejorar para una mejora posterior de los resultados sobre todo en la fiabilidad y la correcta optimización de para llegar a estos.

Primero, para evitar lo de la varianza, se deberían realizar varias optimizaciones sobre cada periodo para asegurar una muestra grande. De esta manera también se mitiga el error introducido por la no convergencia del optimizador si fuera el caso. Por otro lado, como ya se ha indicado, un ajuste fino del ansatz seleccionado al igual que otros parámetros del optimizador son necesarios si se quiere alcanzar los mejores resultados.

De todas formas, la naturaleza errática de estos experimentos es posible que se deba a la naturaleza de las series temporales y que se está realizando una optimización de carteras estática. Por un lado, esto no tiene en cuenta el coste de cambio de un activo a otro, lo cual podría incurrir en resultados bastante diferentes, además de que la evaluación de cada selección debería ser influenciada por la del periodo anterior para corregir esa naturaleza temporal. Por tanto, convirtiéndolo en una estrategia dinámica.

Referencias

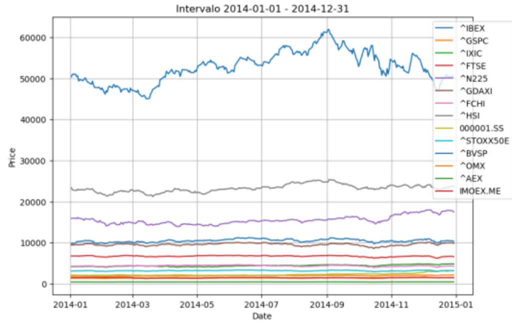
«Portfolio optimization», Wikipedia. 11 de mayo de 2024. Accedido: 9 de junio de 2024. [En línea]. Disponible en: https://en.wikipedia.org/w/index.php?title=Portfolio_optimization&oldid=1223339557

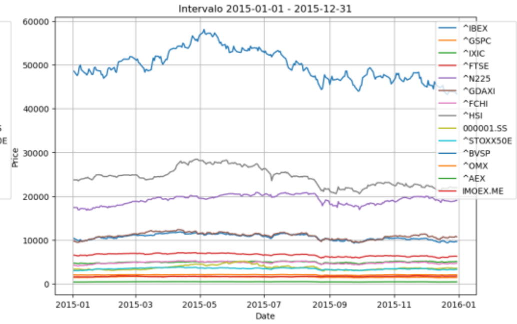
S. Mugel et al., «Dynamic Portfolio Optimization with Real Datasets Using Quantum Processors and Quantum-Inspired Tensor Networks», Phys. Rev. Research, vol. 4, n.º 1, p. 013006, ene. 2022, doi: 10.1103/PhysRevResearch.4.013006.

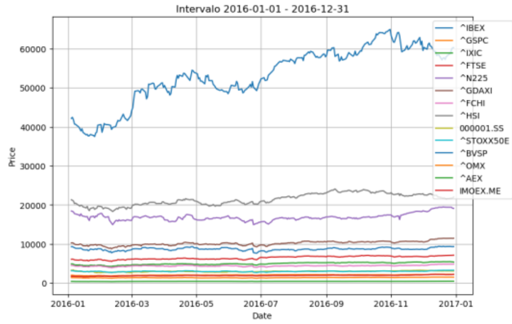
«Portfolio Optimization - Qiskit Finance 0.4.1». Accedido: 9 de junio de 2024. [En línea]. Disponible en: https://qiskit-community.github.io/qiskit-finance/tutorials/01_portfolio_optimization.html

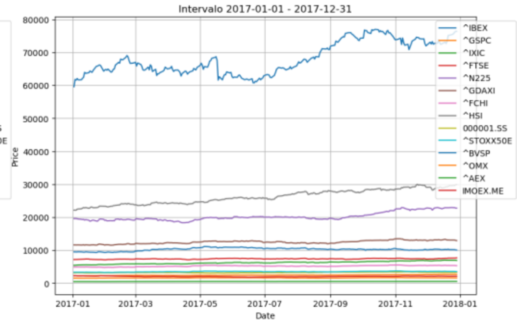
Apéndice A. Datos iniciales separados

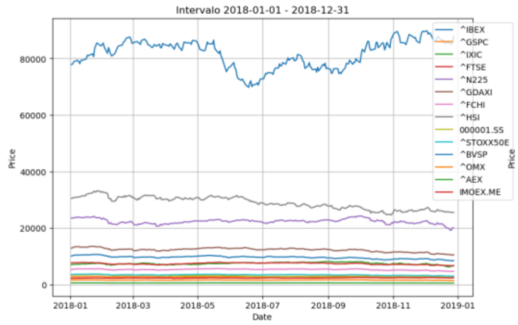
Datos periodos anuales

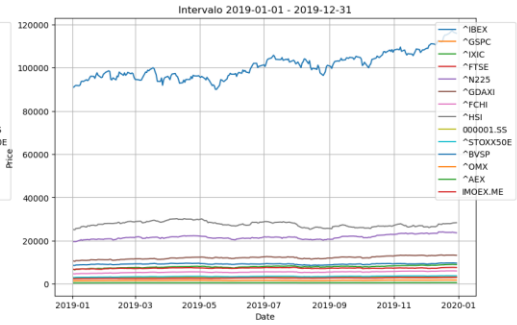


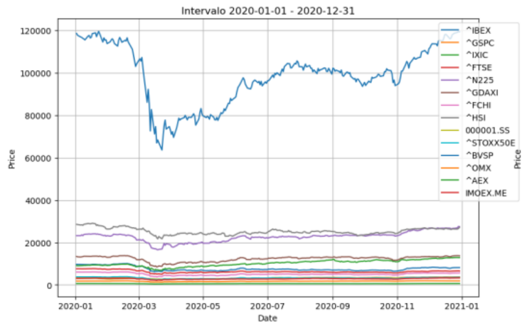


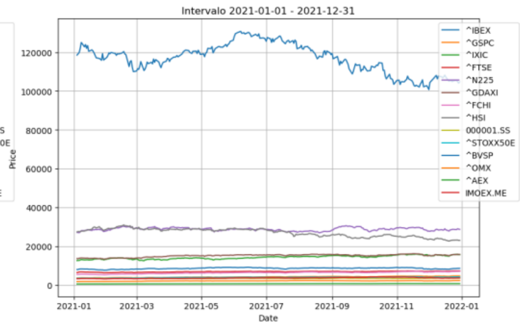


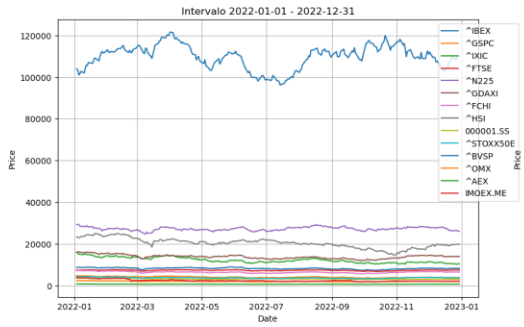


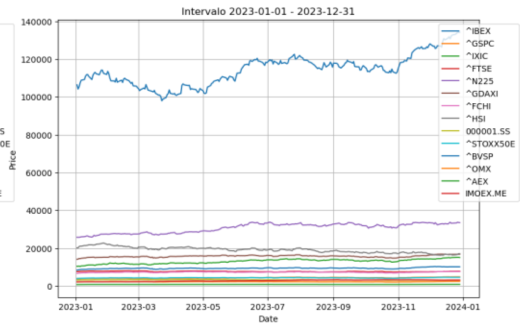




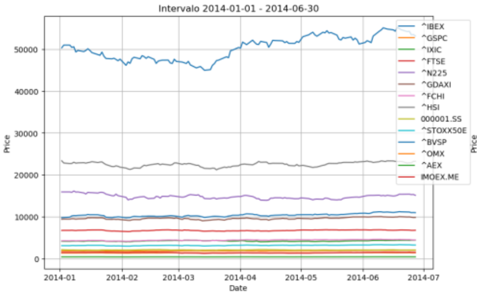


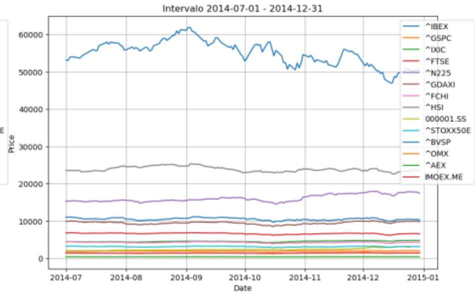


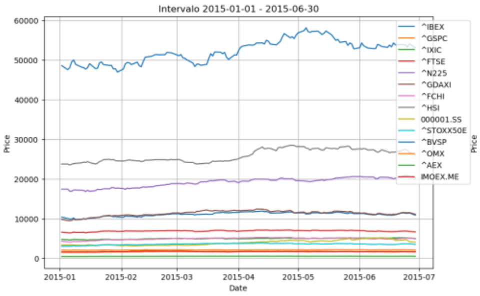


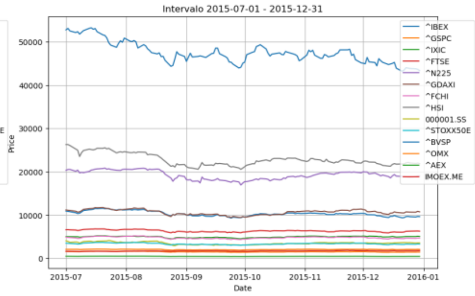


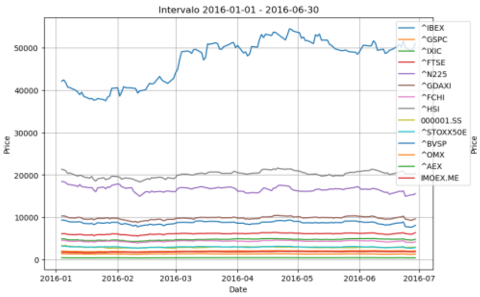
Datos cada 6 meses

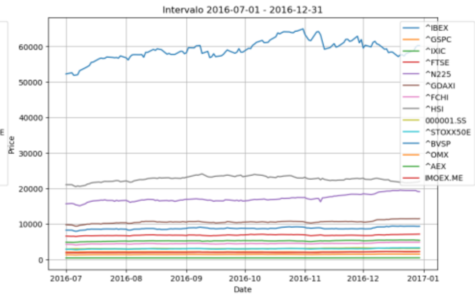


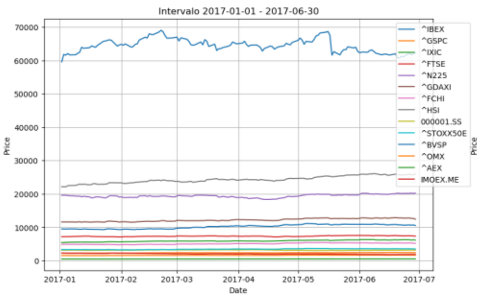


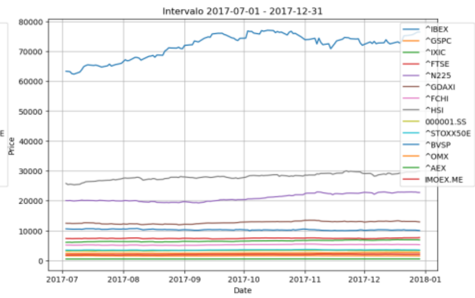


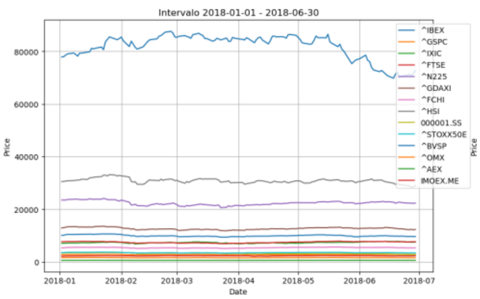


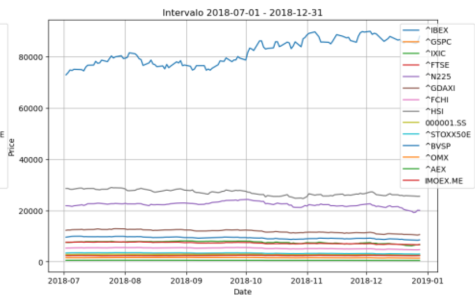


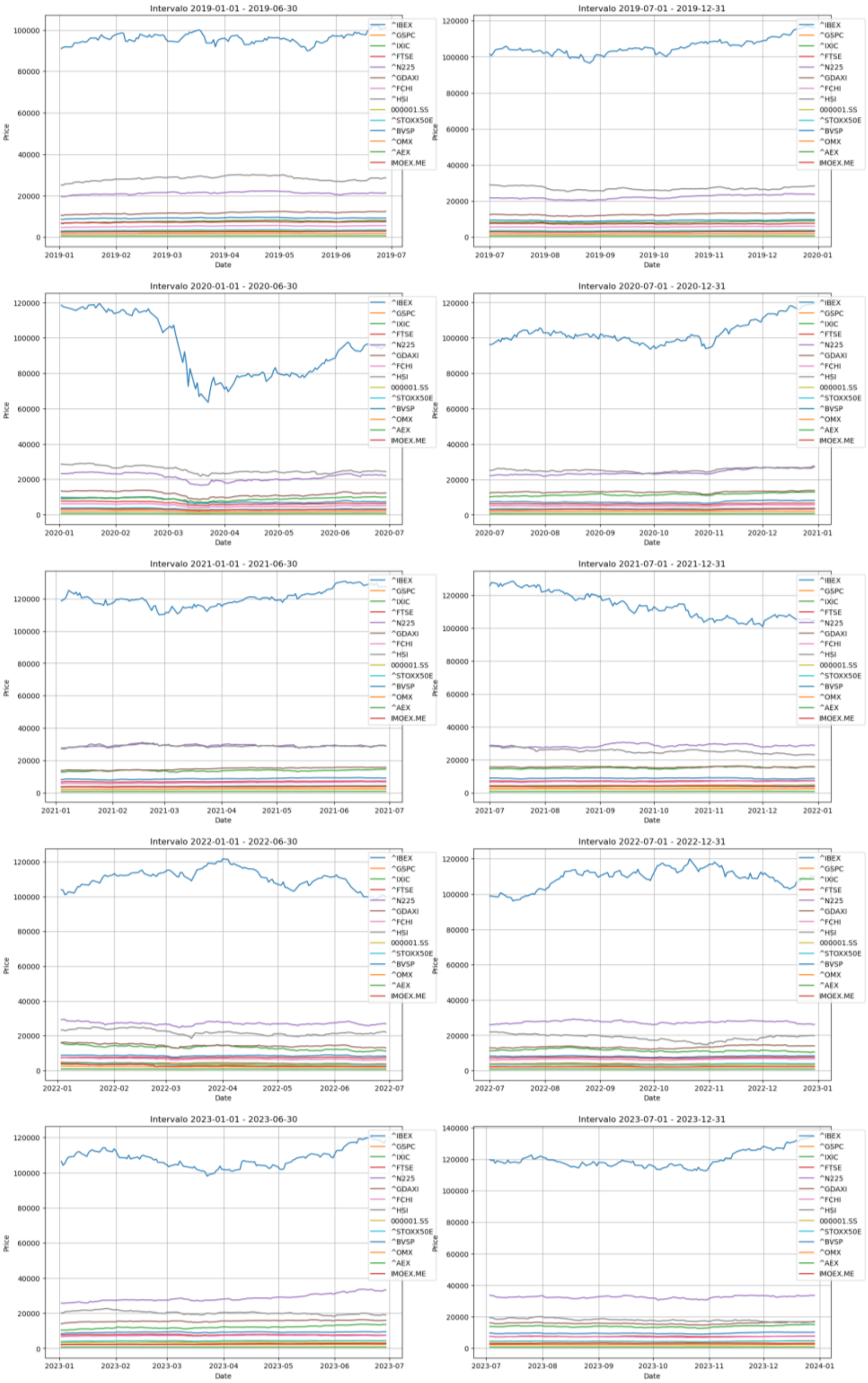










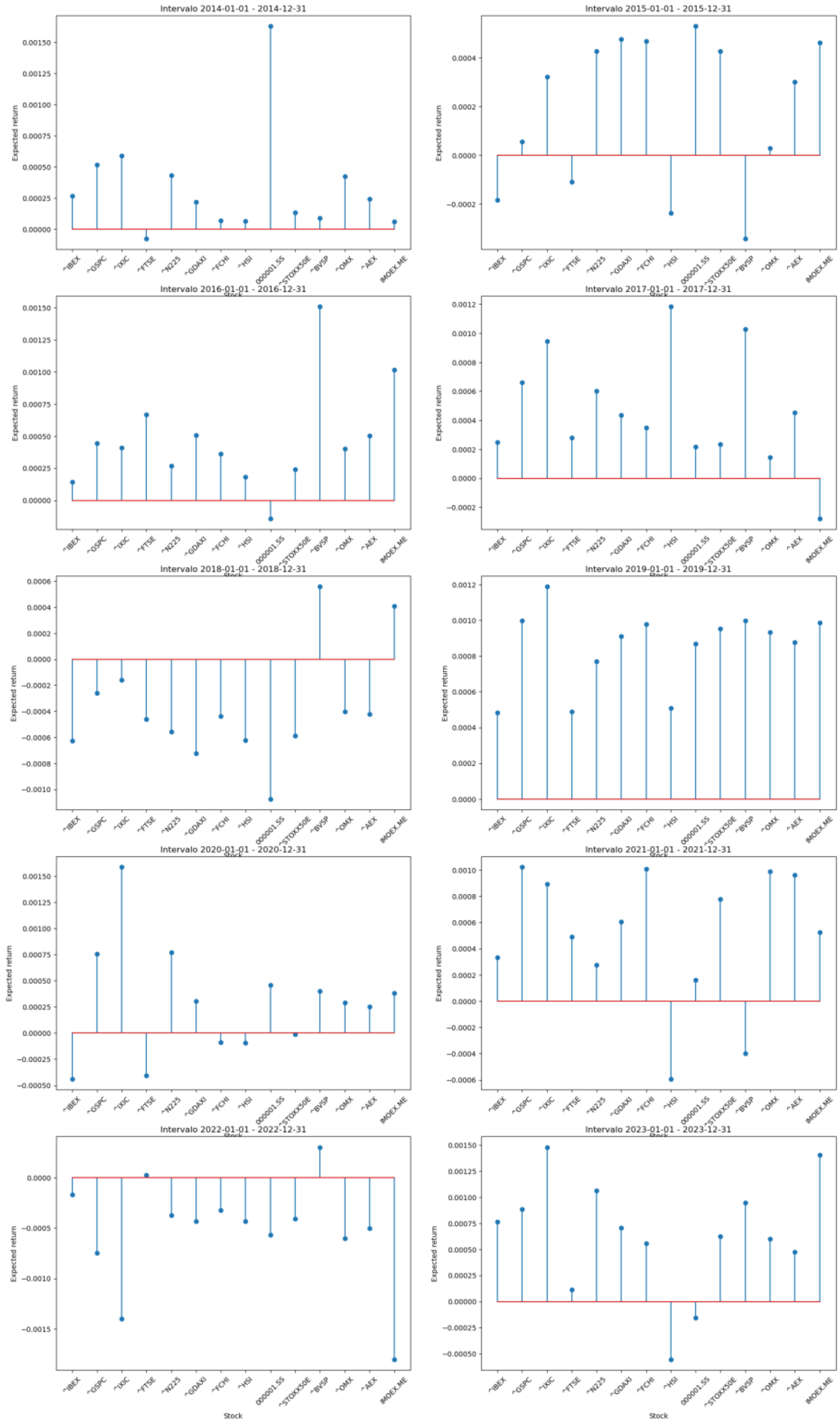


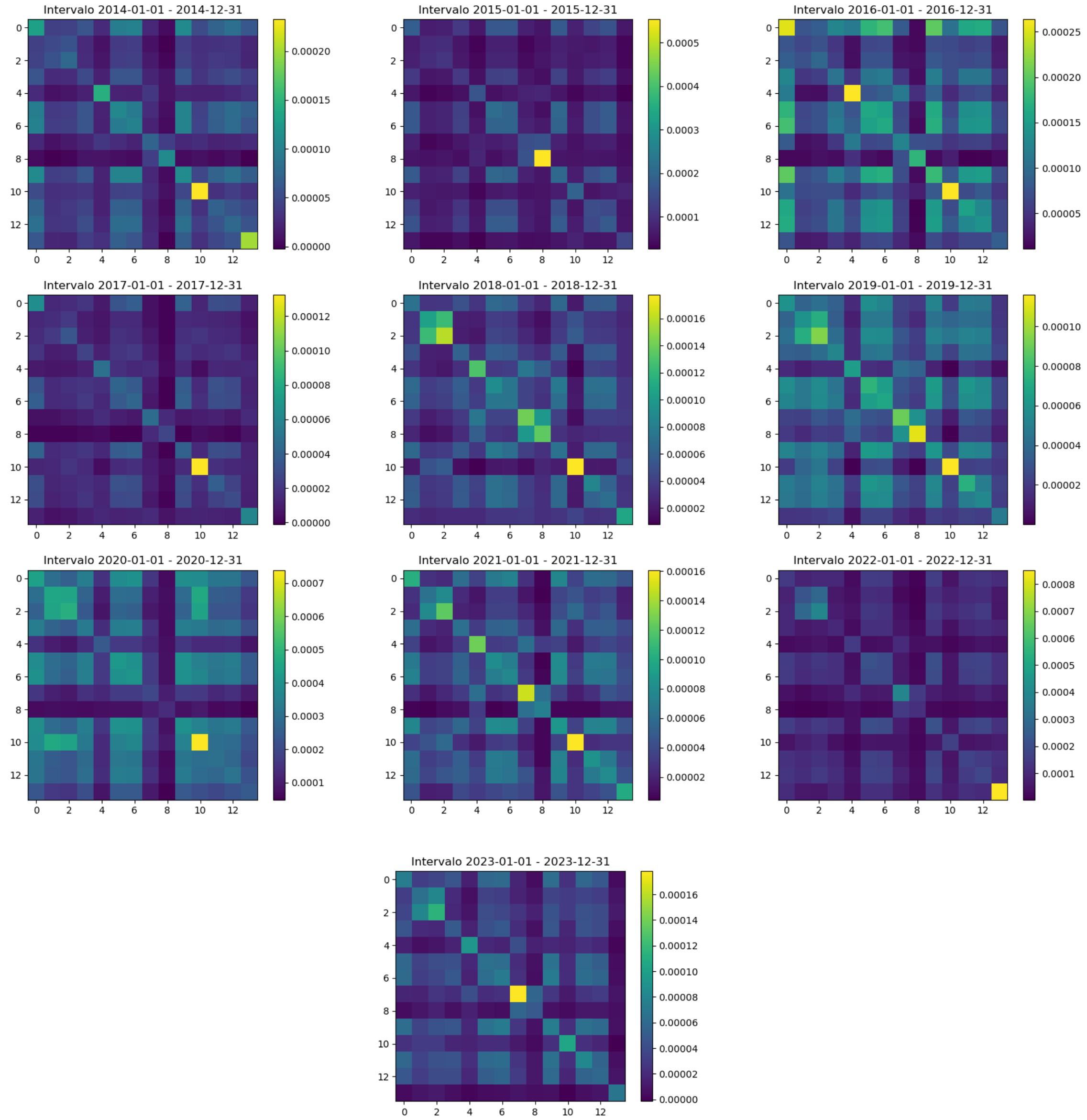
Periodos cuatrimestrales

No se presentan los datos de estos ya que los gráficos ocuparían muchas páginas. Aun así, ese contenido está accesible en el código proporcionado.

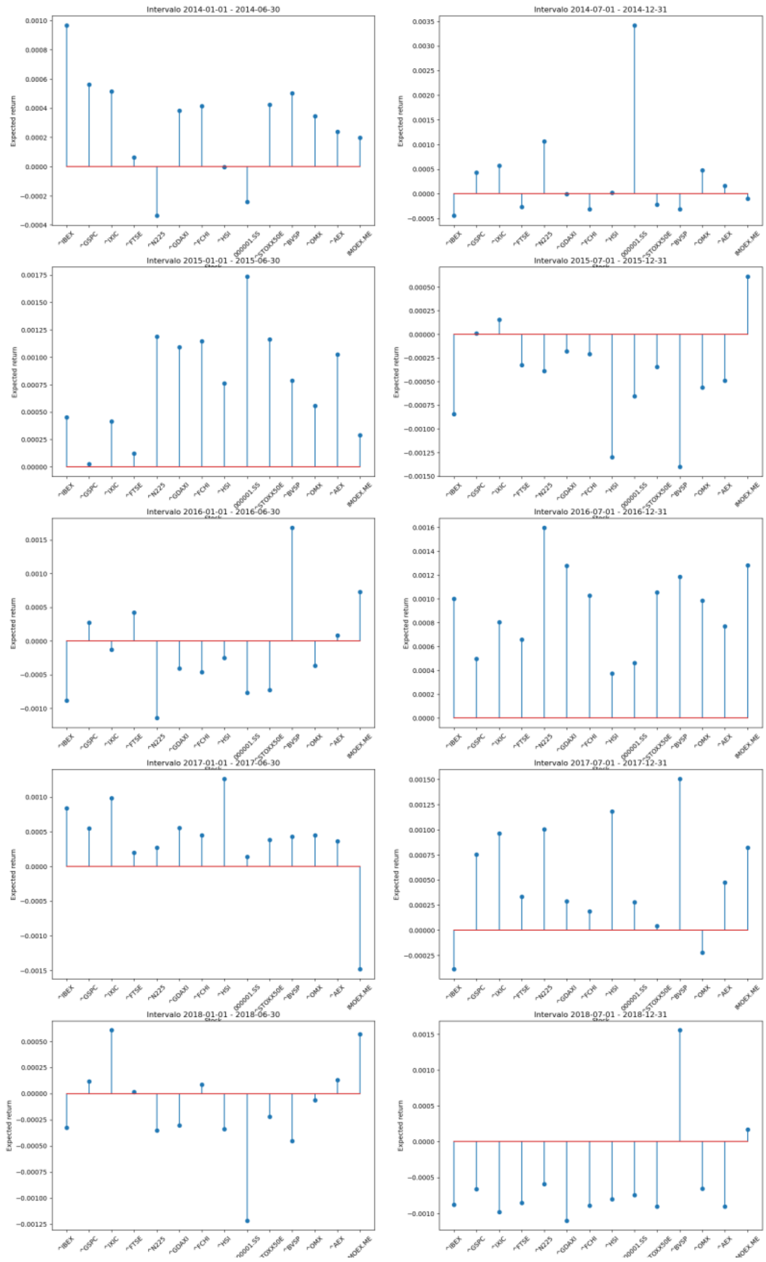
Apéndice B. Retorno esperado y matriz de covarianza

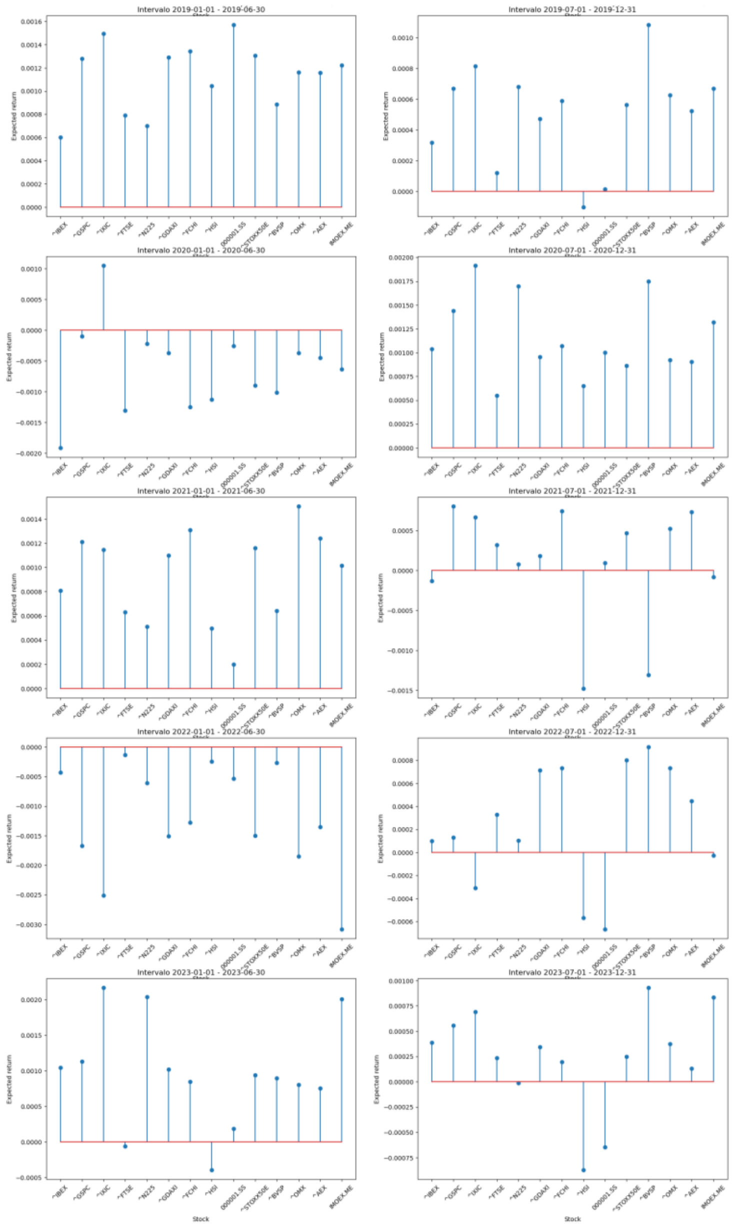
Periodos anuales



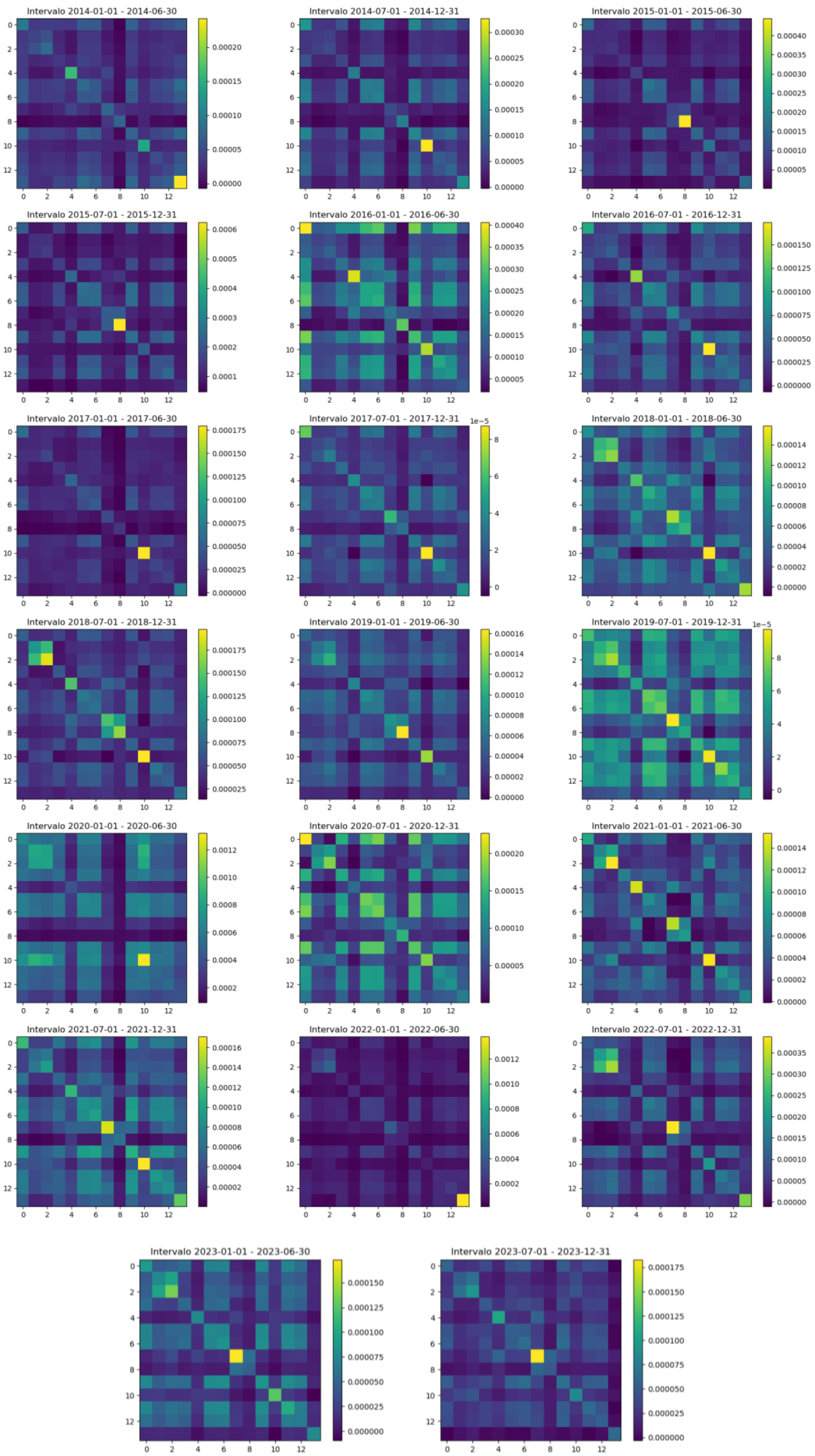

Intervalo 2014-01-01 - 2014-12-31
Intervalo 2015-01-01 - 2015-12-31
Intervalo 2016-01-01 - 2016-12-31
Intervalo 2017-01-01 - 2017-12-31
Intervalo 2018-01-01 - 2018-12-31
Intervalo 2019-01-01 - 2019-12-31
Intervalo 2020-01-01 - 2020-12-31
Intervalo 2021-01-01 - 2021-12-31
Intervalo 2022-01-01 - 2022-12-31
Intervalo 2023-01-01 - 2023-12-31

Periodos cada 6 meses





Intervalo 2019-01-01 - 2019-06-30
Intervalo 2019-07-01 - 2019-12-31
Intervalo 2020-01-01 - 2020-06-30
Intervalo 2020-07-01 - 2020-12-31
Intervalo 2021-01-01 - 2021-06-30
Intervalo 2021-07-01 - 2021-12-31
Intervalo 2022-01-01 - 2022-06-30
Intervalo 2022-07-01 - 2022-12-31
Intervalo 2023-01-01 - 2023-06-30
Intervalo 2023-07-01 - 2023-12-31
Expected return
Stock
^IBEX
^GSPC
^IXIC
^FTSE
^N225
^GDAXI
^FCHI
^HSI
000001.SS
^STOXX50E
^BVSP
^OMX
^AEX
IMOEX.ME


Intervalo 2014-01-01 - 2014-06-30
Intervalo 2014-07-01 - 2014-12-31
Intervalo 2015-01-01 - 2015-06-30
Intervalo 2015-07-01 - 2015-12-31
Intervalo 2016-01-01 - 2016-06-30
Intervalo 2016-07-01 - 2016-12-31
Intervalo 2017-01-01 - 2017-06-30
Intervalo 2017-07-01 - 2017-12-31
Intervalo 2018-01-01 - 2018-06-30
Intervalo 2018-07-01 - 2018-12-31
Intervalo 2019-01-01 - 2019-06-30
Intervalo 2019-07-01 - 2019-12-31
Intervalo 2020-01-01 - 2020-06-30
Intervalo 2020-07-01 - 2020-12-31
Intervalo 2021-01-01 - 2021-06-30
Intervalo 2021-07-01 - 2021-12-31
Intervalo 2022-01-01 - 2022-06-30
Intervalo 2022-07-01 - 2022-12-31
Intervalo 2023-01-01 - 2023-06-30
Intervalo 2023-07-01 - 2023-12-31

Periodos cuatrimestrales

No se presentan los datos de estos ya que los gráficos ocuparían muchas páginas. Aun así, ese contenido está accesible en el código proporcionado.