Licenciatura en Administración

“OPTIMIZACIÓN DE CARTERA DE INVERSIÓN CON PYTHON”

Autor

Débora Natalí Severiche

Reg. N°: 26384

Correo electrónico: deboraseveriche@gmail.com

Tutores

Alejandro Bartolomeo

Gustavo Machín Urbay

Mendoza – 2024

ÍNDICE

RESUMEN

Los inversionistas compran y venden títulos de valores, con el fin de obtener ingresos producto de sus inversiones, por lo que necesitan crear carteras o portafolios eficientes, teniendo por interés maximizar el rendimiento esperado o minimizar el riesgo asociado.

El presente trabajo se dirigirá a modelar un portfolio conformado por acciones del índice S&P500, teniendo por objetivo analizar el modelo matemático desarrollado por Harry Markowitz, que se utiliza para la optimización de carteras y busca minimizar la varianza o riesgo de los activos con la condición de obtener un rendimiento o rentabilidad con ponderaciones positivas o iguales a cero para formar la frontera eficiente encontrando así la mejor combinación de activos con los parámetros establecidos que deben cumplir las carteras. Con esto se podrá comparar si la cartera obtenida es más eficiente incluso que el propio índice de referencia (S&P500).

Utilizando un enfoque cuantitativo se llevará a cabo un procedimiento para generar carteras óptimas usando como objetivo el modelo de optimización de Markowitz, además de las condiciones que deben cumplir las carteras.

Actualmente las tecnologías de información cumplen un rol importante en las finanzas. Python es uno de los lenguajes de programación más versátiles para implementar el modelo y así lograr obtener la formación de una cartera de acciones optimizada.

Llevando a cabo distintas simulaciones y analizando los resultados obtenidos, finalmente se encuentra la cartera o portafolio óptimo para cada perfil de inversor, se verifica la efectividad del modelo y la semejanza con respecto a su índice de referencia.

INTRODUCCIÓN

1. GENERALIDADES DEL TRABAJO

"Un Mercado de Valores o Mercado Financiero, se puede definir como un mecanismo que reúne a compradores y vendedores de activos financieros para facilitar las negociaciones". (Alexander, Sharpe y Baily 2003, pág. 35)

Se puede decir que invertir en la Bolsa, es como un casino al revés. Si usted va todos los días al casino, tal vez a veces gane, pero a cabo del tiempo, acabará perdiendo todo su capital ya que siempre las probabilidades se encuentran a favor del casino. En la Bolsa de Valores la situación es al revés, ya que, invirtiendo en una cartera diversificada, como puede ser un índice de acciones y reinvirtiendo las ganancias, la evidencia empírica demuestra que a largo plazo la Bolsa recompensa al inversor.

Al tomar decisiones dentro del mercado financiero, todo gira alrededor de una Inversión, ya que es la primera acción que se realiza dentro del mercado. Inversión, en el sentido económico, es una colocación de capital para adquirir una ganancia futura, se cambia un beneficio inmediato por uno futuro, el cual es casi siempre improbable.

Al invertir en la bolsa de valores, las empresas como los gobiernos tienen la capacidad de financiar proyectos que sirvan de desarrollo, a su vez los oferentes de estos recursos reciben la oportunidad de invertir en una variedad de instrumentos que les permite diversificar su riesgo y optimizar su rendimiento.

En el presente trabajo se realizará una introducción a los conceptos básicos de las inversiones en el mercado de valores, definiciones de términos y principios que se emplean.

Las Bolsas de Valores son mercados organizados y especializados, en los que oferentes (unidades excedentarias) y demandantes (unidades deficitarias), realizan por medio de intermediarios autorizados (cobran comisiones), negociaciones de una serie de instrumentos financieros, genéricamente llamados "Títulos de Valores", como los instrumentos financieros de renta fija, entre los que se encuentran las obligaciones negociables, bonos corporativos, fideicomisos, cheques; y los de renta variable como acciones, futuros, opciones, divisas, etc. con la finalidad de obtener un mayor rendimiento sobre sus ahorros o disponer de recursos para proyectos de inversión.

Una de las bolsas más importante de los mercados de valores es la de Nueva York (NYSE), compuesto por varios mercados bursátiles, como Dow Jones, Nasdaq, S&P 500. Este último conformado por 11 sectores, representa las acciones de las 500 principales empresas que cotizan en la bolsa de valores de EE. UU., y es en el cual se enfoca esta investigación con la intención de obtener una cartera óptima a *largo plazo*.

Los portfolios están constituidos por un conjunto de activos, en el presente trabajo estará representado por acciones, los cuales son activos riesgosos que tienen resultados inciertos debido a sus variaciones. Por lo que el problema al que se enfrenta el inversor es determinar en qué títulos y en qué proporción de cada uno se conformará su cartera, teniendo en cuenta que el enfoque de inversión del presente trabajo será a largo plazo.

A lo largo de los capítulos se efectuará un análisis sobre el Modelo matemático de Markowitz, estableciendo tanto el objetivo del modelo como las condiciones que deben cumplir las carteras. En otro apartado se incluirá los procedimientos para generar carteras óptimas utilizando el modelo de optimización.

La teoría de Markowitz, se aplica principalmente al mercado de valores, pero en realidad se puede aplicar a cualquier tipo de inversiones donde se pueda estimar una rentabilidad y las medidas de dispersión correspondientes.

La rentabilidad se puede obtener de distintas formas, una de ellas es a través de la esperanza a partir de los históricos de los valores de los activos.

En cuanto a las medidas de dispersión, para aplicar el método de Markowitz hace falta conocer la varianza del valor de los activos como su covarianza, las cuales serán explicadas en los siguientes capítulos. Estas medidas de dispersión forman lo que se conoce como riesgo.

Por último, se realizará una breve introducción al lenguaje de programación Python, empleando dicha metodología cuantitativa para ejecutar el Modelo de Markowitz, incluyendo una descripción de dicha implementación. Se llevarán a cabo simulaciones con el fin de poder analizar los resultados obtenidos y poder dar una repuesta a la siguiente hipótesis: ¿es posible mediante el uso de tecnología obtener un portafolio óptimo que pueda superar el índice tomado como referencia?

El proyecto será factible en la medida que pueda encontrarse, mediante la utilización de tecnologías, la mejor combinación de variables, cuyo resultado sea el máximo rendimiento a distintos niveles de riegos. Se buscará que el rendimiento sea superior al del índice de referencia S&P 500.

La aplicación de tecnología hará posible el proceso de datos y la reducción de márgenes de error en lo que se incidiría al realizar cálculos manualmente. Por lo que brindaría información precisa para obtener los resultados deseados.

2. PLANTEAMIENTO DEL PROBLEMA

La bolsa de valores, está recibiendo inversores de diversas partes del mundo, frente a la globalización, la era de la tecnología y la comunicación inmediata, permite que los participantes recopilen información e inviertan en los mercados desde cualquier parte del mundo, incluso a través de un celular.

Pero el problema que enfrenta cada inversionista, es determinar qué títulos riesgosos (el presente trabajo se centrará en las acciones) poseer en la cartera y en qué proporciones para lograr obtener rendimiento.

Al momento de elegir un portfolio, los inversionistas buscan una combinación de Activos que les permita alcanzar los rendimientos esperados con un riesgo mínimo. Todas las acciones tienen resultados inciertos, por lo tanto, son riesgosas, por lo que una de las formas de reducir el riesgo de un portfolio es mediante la diversificación. A través del modelo de optimización se intenta determinar la mejor combinación de variables con el fin de obtener los resultados deseados, para resolver este problema se propone crear una cartera optimizada la cuál responda la pregunta ¿El modelo de Markowitz proveerá una decisión que a largo plazo sea factible, óptima y eficiente al inversor?, ¿Será esta cartera óptima mejor que el índice de referencia?

3 LIMITACIONES

Mediante este trabajo se buscará contribuir a la toma de decisiones en la selección de carteras de activos altamente cotizables en el mercado de valores. Específicamente, mediante el uso de lenguaje de programación Python, encontrando un portfolio óptimo de acciones que pueda contener un rendimiento superior a los índices de referencia a largo plazo, en este caso se tomará como referencia el índice S&P 500.

Asimismo, se recopilará datos históricos mediante la plataforma de Yahoo Finance, la misma muestra la evolución de los mercados en el tiempo para conocer los precios de cierre de las empresas que pertenecen al índice S&P 500, se trata de una plataforma amigable que permite descargar y obtener fácilmente la información necesaria. Como Yahoo Finance existen diversos sitios web similares, como por ejemplo Invertironline, Investing, Bloomberg, entre muchas otras.

El período de tiempo que se adoptará para la realización del presente trabajo, abarcará desde 2020-12-01 al 2023-12-01.

Después de conocer los precios de cierre se espera poder aplicar mediante el lenguaje de programación Python el modelo matemático de Markowitz y Sharpe y así poder analizar los datos para dar solución al problema del inversor, sin que esto sea un consejo de inversión, ya que cada inversor según sus preferencias (se verá más adelante) tomará sus propias conclusiones.

Finalmente, se verifica la eficiencia de la teoría de Markowitz y Sharpe en la actual selección de un portfolio óptimo o cartera óptima, realizando una comparación con el Índice de referencia con el fin de buscar una respuesta a la pregunta ¿existe una cartera más eficiente y óptima que el índice?

4 OBJETIVOS DEL TRABAJO

Los principales objetivos del presente trabajo son:

- Implementar los conceptos de optimización de cartera, aplicando la teoría de Markowitz y Sharpe a los mejores títulos de valores (acciones) de la Bolsa de Valores de Nueva York, determinando la mejor combinación de acciones en los cuales se debe invertir, para obtener un máximo rendimiento dado un nivel de riesgo o un mínimo riesgo dado un nivel de rendimiento. Esto en virtud de los datos históricos recopilados.

- Implementar, utilizando el lenguaje Python, el modelo de Makowitz por el cual se obtiene las cantidades optimizadas a invertir en cada uno de los activos seleccionados que se consideren de interés y verificar la efectividad de la estructuración de un portfolio óptimo de inversión con las acciones previamente seleccionadas.

- Comprobarla efectividad del modelo matemático identificando el portfolio óptimo (punto de tangencia ente la recta de mercado de capitales y la curva de la frontera eficiente) y realizar una comparación con el índice S&P 500 con la finalidad de poder cotejear si existe una cartera que sea más óptima que un índice a largo plazo.

CAPÍTULO I: MARCO TEÓRICO

1.1 EVOLUCIÓN DE LA TEORÍA DE MARKOWITZ

Markowitz es considerado el padre de la Teoría Moderna de cartera, fue un analista financiero americano premiado con el premio Nobel de Economía en el año 1990, en virtud a su aporte sobre la selección de cartera en el año 1950. Sus estudios se basaron en cómo los inversionistas concilian el riesgo y el rendimiento dentro de las inversiones riesgosas.

En el año 1952, publica el artículo titulado *"Portfolio Selection",* en el mismo especifica un modelo matemático que señala ¿cómo los inversionistas pueden obtener el menor riesgo posible a partir de una determinada tasa de rendimiento. (Rubinstein, 2006).

La evolución de la Teoría de la inversión se divide en 3 etapas: el Período Antiguo (antes de 1950), Periodo clásico (de 1950 a 1980) y el Período Moderno (desde 1980). Es justamente durante el período Clásico donde surgen las ideas de Markowitz sobre la Teoría del Portfolio, luego de haber establecido el riesgo por aversión dado por Huggens del período antiguo.

Los conceptos básicos de la teoría de la cartera fluyeron en Markowitz, a través de la lectura del libro "The theory of investmets value" de John B.William. En ese momento se percata que William no considera un aspecto importante que interviene en el proceso de inversión de cartera, el riesgo.

Además, Markowitz señala que a los inversionistas no les gusta la varianza del retorno de la cartera desde que están adversos al riesgo.

En el año de1959,Markowitz especifica el desarrollo del portafolio, indicando las matemáticas de la diversificación y tomando en cuenta una vía para conciliar su criterio de Media – Varianza con la maximización de la utilidad logarítmica esperada de rendimiento y además, demuestra una eficaz aproximación cuadrática para la estrategia que permite al inversionista escoger portafolios o carteras basados en la media y la varianza; es aquí donde se recomienda la semi varianza como una medida de riesgo dentro de los campos, dándole importancia a sus propiedades y procedimientos computacionales de la cartera óptima. Markowitz, asegura que el análisis basado en la tendencia de la semi varianza produce mejores carteras que aquellos basados en la varianza, donde se consideran retornos extremadamente altos o bajos. Un análisis basado en la varianza busca eliminar análisis extremos, sin embargo, un análisis basado en la semi varianza se concentra en reducir pérdidas. En este punto, Markowitz hace referencia a al modelo de mercado o diagonal, pero no fue hasta 1963 cuando es mejorado por William Sharpe y es quien considera maximizar las rentabilidades conalternativas de medidas de riesgo como: desviación estándar, el valor esperado de pérdida, la desviación absoluta esperada, las probabilidades de pérdida y la máxima pérdida.

Por otra parte, James Tobin afirma que el coeficiente de correlación de los activos de la cartera seleccionados mediante la diversificación, crean un portafolio libre de riesgo, es decir, aquel en el cual los riesgos individuales de cada activo se anulen y genere una cobertura ante el riesgo. Cuando dos activos se mueven en perfecta armonía, es decir que la correlación es igual a uno, el riesgo del portfolio es el promedio ponderado de los desvíos de los activos individuales. En estos casos la diversificación no produce ningún beneficio y es el único caso donde el riesgo del portfolio puede representarse mediante una función lineal. Siempre que el coeficiente de correlación sea menor a uno, la diversificación reducirá el riesgo por debajo del promedio ponderado.

Justamente a esto se refería James Tobin (Premio Nóbel de Economía) en relación con la teoría del portafolio: "... simplemente no es bueno poner todos los huevos en la misma canasta".

Hacia 1964 sobresale la figura de William Sharpe, quien se preguntó:¿qué pasaría si todos en la economía siguieran los mismos consejos que daba en su teoría Markowitz?, esta interrogante dirige a Sharpe a su primera publicación: "Capital Asset Pricing Model" (CAPM), cuyo significado en español es: "Modelo de Valoración de Activos de Capital". Este modelo indica que los valores de retorno μ_j, equivalen a la suma de los retornos sin riesgo $r$sumado con el producto de aversión al riesgo de todo el mercado y la covarianza de los valores de retorno con el retorno de la cartera del mercado (θ>0).Así:

$$\boldsymbol{\mu_j = r + \theta Cov(r_j, r_M)}$$

En la actualidad, la fórmula de Sharpe se describe así: $Q_j = \boldsymbol{Cov(r_j, r_g)/\sigma^2{}_G}$

La ecuación de CAPM puede ser interpretada como una descripción para discontinuar un incierto flujo de caja recibido al final de un solo periodo. El modelo CAPM construido sobre el de Markowitz, Royy Tobin(1952-1958),supone que todos los inversores escogen su portafolio o cartera de valores considerando únicamente la media de la cartera y la varianza del retorno, en el sentido de los deseos que el inversionista prefiera, tanto un retorno apropiado como una varianza retenida igual, adicional a un riesgo de aversión en el sentido que no sea del agrado del inversionista la varianza y el retorno esperado retenido igual.

El descubrimiento del CAPM es uno de los eventos más misteriosos en la historia de la teoría de inversión.

Con el pasar de los años, el desarrollo de las teorías de inversión ha tenido su mayor aporte en lo planteado por Harry Markowitz, cuya teoría da paso a nuevas ideas que se han ido complementando en la ejecución de inversiones pasando por las ideas de Tobin, seguido por Sharpe con el establecimiento del CAPM, y cuyos fundamentos se han mantenido como la base fundamental desde distintos puntos de vista para tomar decisiones en inversión de carteras dentro de un mercado de valores bursátil. Hoy en día se dan otras alternativas de inversión, motivadas por el surgimiento de los productos derivados y su efecto sobre el riesgo. Aun así, el CAPM y la Teoría de Markowitz todavía impactan a la teoría de inversiones.

1.2 TEORÍA DE CARTERA

La teoría de cartera es un modelo de optimización que se basa en los aportes realizados por Markowitz y Sharpe, mediante los cuales se procura determinar cuál es la cartera óptima o la mejor combinación de activos para que el inversor obtenga los rendimientos deseados. Si bien esta teoría nació con Harry Markowitz, fue posteriormente Sharpe quien la concluyó.

Ante el problema principal de la selección de cartera, que es determinar que títulos y en qué proporción poseer en la cartera, H. Markowitz presentó una solución que se basa en dos suposiciones: primero es la **insaciabilidad** en la cual el inversionista preferirá los niveles más altos de rendimiento ante niveles equivalentes de riesgo, y el segundo supuesto es la **aversión al riesgo**, se elegirá la cartera con menor nivel de riesgo ante un mismo nivel de rendimiento.

El modelo de Markowitz se basa en 2 hipótesis, basadas en:

- Mercado financiero y activos

- Los mercados están conformados únicamente por activos financieros volátiles o arriesgados (acciones), no existiendo ningún activo libre de riesgo.
- Los mercados financieros son perfectos. Es decir, que toda la información está al alcance de los inversores; no existen costes de transacción de compraventa; los títulos son infinitamente divisibles (se puede poseer cualquier porcentaje de los mismos); no existe inflación ni impuestos y los agentes son precio aceptantes (cualquiera cantidad demandada no afecta al precio, siendo este constante).
- Los activos disponen de liquidez inmediata al final del periodo.
- No se permiten las ventas en corto.

- Comportamiento y método de elección del inversor.

- Los individuos se comportan racionalmente, buscando maximizar su función de utilidad.
- Los inversores tienen aversión al riesgo, por lo que, para igual rendimiento, se busca la cartera de menor riesgo.
- La función de utilidad depende únicamente del rendimiento esperado como medida de la rentabilidad, y de la varianza como medida del riesgo.
- Las funciones de utilidad son monótonas crecientes, por lo que a igual varianza (riesgo) se busca un mayor rendimiento.

En este capítulo veremos el modelo de selección de cartera óptima de ambos escritores, que están desarrollados sobre la base de algunos conceptos fundamentales que trataremos previamente, como las curvas de indiferencia del inversor, el rendimiento y el riesgo, la forma de medirlos tanto para un solo activo como para una cartera.

1.2.1 RENDIMIENTO

Comprender la relación entre rendimiento y riesgo cumple un papel fundamental para proyectar y diseñar un portfolio. Un inversionista deberá analizar las carteras alternativas de acuerdo a sus rendimientos esperados y riesgo.

El rendimiento y riesgo de una cartera dependerá del rendimiento y riesgo de cada activo de la cartera. Por lo que la suma invertida en cada valor también es significante.

El rendimiento periódico del título i durante un período de tiempo **t**, estará definido por la siguiente expresión:

$$R(t-1;t) = \frac{P_t}{P_t - 1} - 1 \qquad (1)$$

Donde P_t indica el precio del activo financiero al final del período t; P_{t-1} 1es el precio del activo al comienzo del período t; $R_{(t-1;t)}$es el rendimiento del título en el período t entre los momentos $t-1$ y t.

RENDIMIENTO ESPERADO

Se entiende por rendimiento esperado, al rendimiento de una acción que el inversionista espera obtener durante el lapso de tenencia del título.

El procedimiento para calcular el rendimiento esperado de una cartera, es el resultado del promedio ponderado de los rendimientos esperados de los valores que la componen y de su participación proporcional en el valor inicial de mercado de la cartera. Suele expresarse en % respecto a la cantidad invertida. Los inversionistas deben incluir más de un valor a sus carteras, puesto que esto conlleva a reducir el riesgo.

Para calcular el rendimiento esperado de *N* valores, se puede utilizar la siguiente ecuación o esperanza matemática:

$$E(R_i) = \overline{R_\iota} = \sum_{t=1}^{n} R_{it} P_{it} \qquad (2)$$

$$= R_1 p_1 + R_2 p_2 + \ldots\ldots\ldots\ldots + R_n p_n$$

Donde:

- Ri: rendimiento esperado de la cartera.
- $\sum$: signo de sumatoria.
- Rit: rendimiento esperado del activo *i* en el evento *t*.
- Pit: proporción del valor inicial invertido en el activo *i* en el evento *t.*
- *N*: número de valores de la cartera.

Un caso para la aplicación de esta ecuación es la determinación del rendimiento de una cartera de acciones, que tienen diferentes grados de participación.

1.2.1.1 DETERMINACIÓN DE TASA DE RENDIMIENTO DE UNA CARTERA

Siendo que un portfolio es un conjunto de activos, los cuales no tienen un valor fijo, estos pueden adquirirse en un determinado momento W_0y al venderse en el momento W_1pueden tener el mismo o un valor distinto que enW_0.

Por lo que para saber si una inversión es rentable o no, este dato se puede obtenerse de la siguiente manera a través de la tasa de retorno:

$$R_P = \frac{W_1 - W_0}{W_0} \qquad (3)$$

Donde:

- W_0: valor del compra en el momento 0.
- W_1: valor de mercado en el momento 1.

- R_P: tasa de rendimiento de una cartera.

Si esta relación es positiva, indica que el precio de compra ha sido menor al precio de venta. Por otro lado, si la relación es negativa es debido a que el precio de venta ha sido inferior al precio al cuál se compró.

1.2.2. RIESGO

La desviación estándar o riesgo es la probabilidad de lo que podría diferir un rendimiento real con relación al rendimiento esperado. Si esta estimación es alta, más dispersos estarán los rendimientos observados alrededor del promedio, es decir que se podría afirmar que la inversión a realizar es altamente riesgosa.

Una de las medidas más utilizadas para calcular la desviación estándar o volatilidad es a través de la varianza.

La varianza se calcula mediante la disparidad entre los rendimientos observados y el rendimiento promedio, elevada al cuadrado. Son cuadradas debido a que los resultados pueden variar por encima y por debajo del promedio, originando diferencias positivas y negativas.

$$\sigma^2 = \sum_{t=1}^{n} (R_{it} - \overline{R}_i)^2 \rho_{it} \qquad (4)$$

Donde p_{it} es la probabilidad de ocurrencia de cada uno de los valores.

En un ejemplo para dos acciones podemos obtener la ecuación desarrollada de la siguiente manera:

$$\sigma^2 = \sum_{t=1}^{n} [k_1^2 (R_{1t} - \overline{R_{1t}})^2 p_{1t} + k_2^2 (R_{2t} - \overline{R_{2t}})^2 p_{2t} + 2k_1 k_2 (R_{1t} - \overline{R_{1t}})(R_{2t} - \overline{R_{2t}}) p_{it}]$$

Los primeros sumandos representan la varianza del rendimiento multiplicado por la participación de cada título al cuadrado. Y el tercer apartado representa a la covarianza de los rendimientos de cada activo, multiplicado por la correspondiente participación al cuadrado.

La covarianza es una medida estadística que representa la relación de dos variables aleatorias, es decir cómo se mueven juntas, si la covarianza es positiva indica que las variables tienden a moverse en la misma dirección. Esto está muy relacionado con el coeficiente de correlación, ya que este último es el resultado de la división de la covarianza entre los valores y el producto de sus desvíos estándar.

$$\rho_{ij} = \frac{\sigma_{ij}}{\sigma_i \sigma_j} \qquad (5)$$

Donde $\boldsymbol{p_{ij}}$ representa la correlación de los rendimientos de los valores i, j y $\boldsymbol{\sigma_{ij}}$ es la covarianza de los rendimientos entre los valores i y j.

Este coeficiente siempre se encontrará en los valores de +1 y -1 y es muy importante para determinar el riesgo de un portfolio, ya que para construir la cartera de buscarán inversiones que estén correlacionadas negativamente, es decir que cuando una tienda a subir la otra bajará y viceversa.

La función objetivo del modelo con N acciones, consiste en minimizar el riesgo del portfolio.

$$Min \sum_{j=i}^{n} \sum_{i=1}^{n} x_i \, x_j \, \sigma_{ij} \qquad (6)$$

En donde x_i representa el peso de la inversión en el título i, x_j representa el peso de la inversión en el título j, σ_{ij} corresponde a la covarianza entre los títulos ij.

Esto sujeto a dos limitaciones, la primera que la suma de la proporción invertida en cada título sea igual a 1 (uno) y la segunda que la rentabilidad de la cartera sea igual al rendimiento exigido por el inversor.

$$\sum_{i=1}^{n} x_i = 1$$

$$\sum_{i=1}^{n} W_i \ r_i = R_p$$

Donde r_i corresponde al rendimiento esperado para el título i, R_p representa el rendimiento del portafolio esperada por el inversionista, esta sería la variable modificable y por la cual se decide el conjunto de carteras eficientes viables para invertir.

1.2.3. ACTITUD DEL INVERSIONISTA FRENTE AL RIESGO

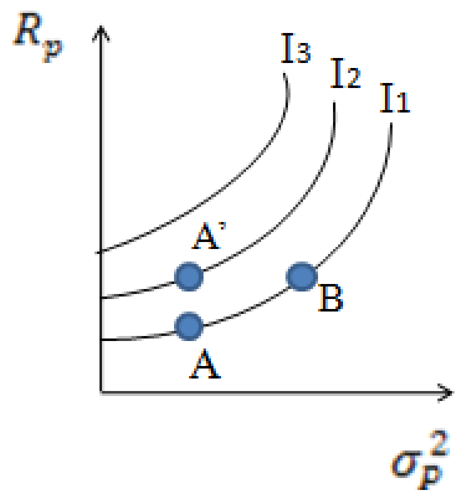
En el modelo de Markowitz se busca optimizar una cartera de inversión, en función del nivel de aversión al riesgo por parte del inversionista y su actitud ante los rendimientos esperados, por lo que se busca determinar cuál es la curva de indiferencia del inversor. Además, se puede decir que cada curva naturalmente será distinta para cada inversor y las mismas no se pueden intersecar.

Una curva de indiferencia es un conjunto de combinaciones de riesgo y rendimiento esperado que proporcionan al inversor la misma utilidad. Los inversores son indiferentes a la combinación de riesgo y rendimiento esperado para la misma curva de indiferencia.

Dado que las curvas de indiferencia indican las prioridades del inversor con respecto al riesgo y el rendimiento esperado, se pueden crear en una figura bidimensional donde el eje vertical indica los rendimientos esperados (R_p) y en el eje horizontal se plasman los riesgos medidos por la desviación estándar esperada (σ^2_p).

Por ejemplo, en la siguiente figura, al inversor le será indiferente elegir entre el punto A o el punto B en la curva de indiferencia I_1, pues, aunque B promete un mayor rendimiento que la cartera A, su riesgo es superior al de ésta última. Sin embargo, si tiene que elegir entre las carteras A y A´ elegirá ésta última, debido a que con el mismo riesgo obtiene un mayor rendimiento (A´>A).

Figura N°1 – Curvas de indiferencia



Fuente: Elaboración propia

Existen tres actitudes del inversionista frente al riesgo, tales como:

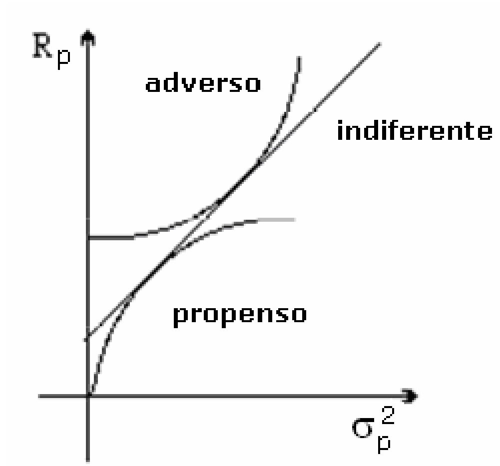
1. **Indiferente frente al riesgo**, significa que el inversionista se mantiene neutral o indiferente si tuviese que elegir entre dos alternativas con el mismo nivel de rendimiento esperado. Por cada unidad de riesgo adicional, se espera el mismo rendimiento.

2. **Adverso al riesgo**, para el cual "la rentabilidad requerida aumente ante un incremento del riesgo." (Gitman & Joehnk, 2005). Lo cual implica que por cada unidad de riesgo adicional hay que prometer un rendimiento cada vez más grande, ó entre dos alternativas de inversión con un mismo nivel de rentabilidad esperada, el inversionista elegirá la de menor riesgo. En el gráfico, cualquier cartera que esté más al noroeste en la curva será mayormente atractiva que otra que se encuentre menos al noroeste. Cuánto más adverso al riesgo sea la inversionista más inclinada será la curva de indiferencia.

3. **Propenso al riesgo**, para el cual "la rentabilidad requerida disminuye para un incremento

de riesgo." (Gitman & Joehnk, 2005). Es decir, el inversionista elegirá aquella inversión con mayor riesgo para un mismo nivel de rentabilidad esperado, por lo que su función de utilidad será de pendiente ascendente y convexa.

Figura N°2 – Actitud del inversionista frente al riesgo



Fuente: Elaboración propia

1.3 OPTIMIZACIÓN DE CARTERA

Los modelos de optimización, están compuestos por las siguientes variables:

- Variables de decisión: son aquellos valores controlables. Estas variables serán la participación (de acuerdo a su peso relativo) de cada una de las acciones en la cartera del inversor *(k1, k2…, kn).*

- Objetivo: es aquello que se pretende optimizar, algunos objetivos son *Maximizar el rendimiento* (dada una combinación de acciones, ofrezcan el máximo rendimiento esperado, para niveles variables de riesgo) o *Minimizar el riesgo* (dada una combinación de acciones, ofrezcan el mínimo riesgo para niveles variables de rendimiento).

- Restricciones: se refiere a las limitaciones en los valores que pueden tomar las variables de decisión. Un de las restricciones es que las participaciones de las acciones en el portfolio deben ser mayor o igual a 0 y a su vez deben sumar 1, es decir ser complementarias.

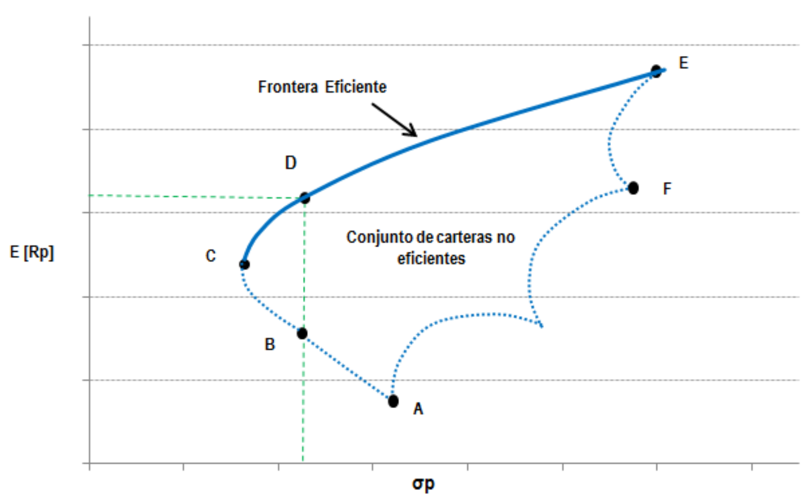
1.3.3 REGIÓN FACTIBLE

La región factible es una figura convexa conformada por títulos, en donde, en función de la proporción invertida en cada título, todas las combinaciones posibles de títulos que den como resultado un punto dentro de los límites de la figura convexa son carteras factibles y ,fuera de ella son carteras no factibles. El inversionista no calificará todas las carteras, sino que buscará el subconjunto de las mismas en función del teorema del conjunto eficiente. Este teorema establece, como hemos visto antes, que el accionista seleccionará dentro del conjunto de carteras su cartera óptima en base a aquella que maximice el rendimiento esperado para distintos valores de riesgos, o aquella que minimice su riesgo para distintos valores de rendimientos esperados.

Algo que se debe tener en claro, es que una cartera con mínimo riesgo no es sinónimo de cartera eficiente, por ello, dentro de la región factible también diferenciamos entre:

- Carteras eficientes o frontera eficiente, y
- Carteras no eficientes.

Figura N° 3- Frontera Eficiente

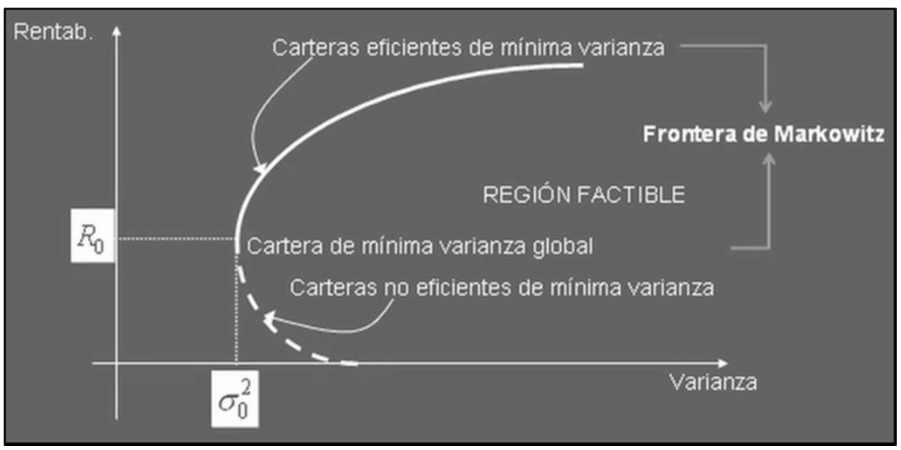


Fuente: Adaptado de "Diversificación Internacional de portafolio en los mercados accionarios de Argentina, Brasil, Chile, Colombia, México y Perú", 2019, (pág. 6) (https://www.researchgate.net/publication/320623350_DIVERSIFICACION_INTERNACIONAL_DE_PORTAFOLIO_EN_LOS_MERCADOS_ACCIONARIOS_DE_ARGENTINA_BRASIL_CHILE_COLOMBIA_MEXICO_Y_PERU)

1.3.4 FRONTERA EFICIENTE

Alexander, Sharpe y Bailey (2003) definen a la 'frontera eficiente' como el conjunto de todas las carteras eficientes, es decir aquellas que presentan menor riesgo dado un nivel de rentabilidad, por lo tanto, son válidas para el inversionista. En función de lo anterior, el punto de separación entre la frontera eficiente y la frontera no eficiente se denomina Cartera de mínima varianza global, como lo podemos observar representado en el siguiente gráfico (4).

Figura N° 4 – Frontera eficiente y carteras de mínima varianza



Fuente: Adaptado de "Análisis y selección de inversiones en mercados financieros", 2008, Brun & Moreno

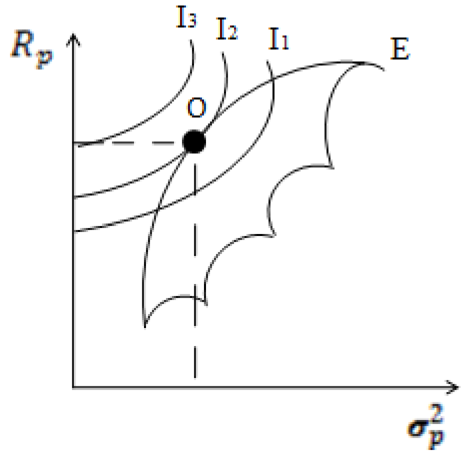
'La cartera de mínima varianza global' representada como el punto (σ^2; R_0) en la figura 4, corresponde a la cartera con el menor riesgo para un conjunto de activos financieros, es decir, no es posible obtener un menor riesgo para una combinación de activos financieros dada. Si bien, todas las carteras que se encuentran dentro de la región factible son opciones de inversión, para el inversionista, las carteras eficientes de mínima varianza representadas en el gráfico como una línea continua convexa, constituyen opciones de inversión válidas y, por lo tanto, las carteras no eficientes de mínima varianza representadas por la línea entrecortada en el gráfico, así como también las carteras no eficientes que se encuentran dentro de la región factible pueden ser descartables.

1.3.5 CARTERAS EFICIENTES

El modelo de selección de carteras de Markowitz busca obtener carteras eficientes, se entiende por cartera eficiente u optimizada a aquella que permite la mejor combinación de acciones que entreguen la más conveniente relación rentabilidad-riesgo dentro de los activos disponibles, en función de la conducta racional del inversionista, la cual establece dos supuestos: ante niveles equivalentes de riesgo siempre se preferirán los rendimientos más altos a los más bajos (INSACIABILIDAD), y el segundo supuesto es que ante el mismo rendimiento pero que difieran en el riesgo se escogerá la cartera con nivel de riesgo menor (AVERSIÓN AL RIESGO).

Podemos observar en el siguiente gráfico que la cartera óptima que cumple con las condiciones anteriores correspondería con el punto O, es decir, es la tangente de la curva de carteras eficientes con la curva de indiferencia I_2(gráfico 5). Dicha cartera se sitúa en este punto porque cualquier otro punto de la curva de carteras eficientes, correspondería con una curva de indiferencia de menor satisfacción.

Figura N° 5 – Cartera óptima según curva de indiferencia y frontera eficiente



Fuente: Elaboración propia

La figura representada es el conjunto de oportunidades de inversión posibles. La curva que une los puntos *O* y *E* es la curva o frontera de carteras eficientes; es decir, la representación de las carteras generadas por el planteamiento de Markowitz. El punto *O* se corresponde con la cartera de mínima varianza posible, mientras que el punto *E* es la cartera que ofrece la máxima rentabilidad posible.

Se puede concluir que la frontera eficiente es objetiva ya que es igual para todos los inversores, mientras que las curvas de indiferencia son subjetivas por lo que son distintas para cada inversor, como así también la cartera óptima.

El modelo de selección de cartera viene dado por tres hipótesis:

- La rentabilidad de una cartera, viene dada por una variable aleatoria cuya distribución de probabilidad para el período de referencia es conocida por el inversor.
- El riesgo es medido a través de la varianza o la desviación estándar de la rentabilidad.
- Los inversores son adversos al riesgo y prefieren portfolios con la mayor rentabilidad y el menor riesgo posible.

Según esta teoría, se trata de buscar primeramente cuáles son las carteras que proporcionan el mayor rendimiento para un riesgo dado, al mismo tiempo que soportan el mínimo riesgo para un rendimiento conocido. A estas carteras se las denomina eficientes.

Analíticamente, podemos decir que, una cartera *p* es eficiente cuando:

- Cualquier otra cartea q de menor riesgo presente también menor rentabilidad que la cartera *p*.

$$\forall q \sigma_p^2 < \sigma_p^2 \rightarrow E\left(R_q\right) < E\left(R_p\right)$$

- O cualquier otra cartera q con igual riesgo que la cartera *p*, presente una rentabilidad menor o igual que la cartera *p*.

$$\forall q \sigma_q^2 = \sigma_p^2 \rightarrow E\left(R_q\right) \leq E\left(R_p\right)$$

Podemos concluir que el proceso de búsqueda de la cartera óptima se obtiene a partir de las siguientes tres etapas:

1- Determinación del conjunto de carteras eficientes

Este conjunto de carteras se puede hallar mediante el siguiente planteamiento matemático, en donde la función objetivo, consiste en proporcionar la máxima rentabilidad posible para un nivel de riesgo dado:

$$Max\ R_x = \sum_{i=1}^{n} X_i\, E[R_i]$$

Sujeto a:

$$\sigma_c{}^2 = \sum_{i=1}^{n} \sum_{j=1}^{n} X_i\, X_j \sigma_{ij} \qquad (7)$$

$$0 \leq X_1, X_2, \ldots, X_N \leq 1 \ (1.4)$$

$$X_1 + X_2 + \ldots + X_N = 1 \ (1.5)$$

La restricción expresada anteriormente es paramétrica, en el sentido de que $\sigma_c{}^2$ representa el nivel de riesgo que el inversor está dispuesto a soportar y se puede modificar en las distintas ejecuciones del modelo a fin de conseguir en cada una de ellas una cartera eficiente diferente. La ecuación (1.5) es la restricción presupuestaria y la (1.4) la de no negatividad de las incógnitas del problema. El programa busca obtener la combinación de valores X_i (composición de la cartera) que maximizan la rentabilidad (1.1), cumpliendo con las tres restricciones, para distintos valores de $\sigma_c{}^2$. Cada combinación de dichos valores nos proporciona la cartera que maximiza la esperanza matemática de la rentabilidad para cada valor de la varianza. Es necesario destacar que el problema también podría ser planteado de forma inversa, es decir, tomando como función objetivo $Min.\ \sigma_c{}^2$ (1.2). En este caso, cada combinación de valores proporcionaría la cartera de mínimo riesgo para cada valor de rentabilidad.

Rendimiento y riesgo son los parámetros que varían, lo que implica ir dándole valores a ambas variables para que el modelo nos diga en todo momento cuál es la mejor cartera para cada valor de ambas variables.

La varianza de un activo se calcula como:

$$\sigma_i{}^2 \sigma_j{}^2\ (R_i),$$

$$Donde: \sigma_i{}^2\ \frac{\sum_{i=1}^{n}(R_{it}-E_i)^2}{n} \qquad (8)$$

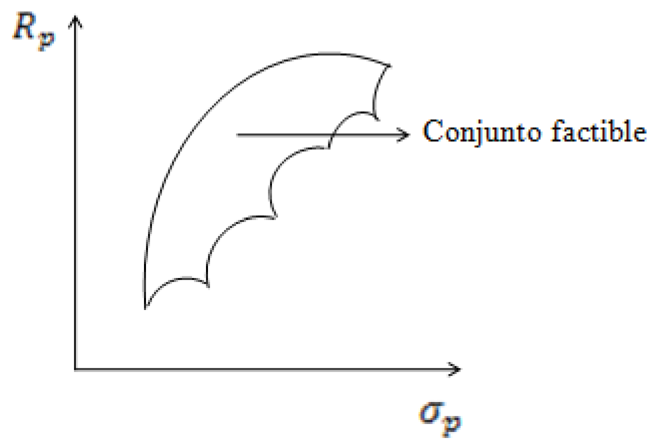
Y la desviación estándar es $\sqrt{\sigma_i{}^2} = \sigma_i = \sigma(R_i)$

La covarianza entre el activo i y j

$$\sigma_{ij} = \frac{\sum_{t=1}^{n}[(R_{it}-E(R_i))][(R_{jt}-E(R_j))]}{n} \qquad (9)$$

Por lo tanto, el modelo permite obtener el conjunto de carteras eficientes (figura 6)

Figura N° 6 – Frontera de carteras eficientes según el modelo de Markowitz



Fuente: Elaboración propia

2- La especificación de la actitud del inversor ante el riesgo.

Una vez obtenido el conjunto de carteras eficientes el inversor deberá elegir aquella que mejor satisfaga sus preferencias: su cartera óptima. Para ello es necesario especificar sus curvas de indiferencia entre rendimiento y riesgo. Como se describió con anterioridad las curvas serán diferentes para cada inversor, según sus propias funciones de utilidad o satisfacción, dadas por su grado de aversión al riesgo. Así, existirán inversores que busquen una mayor rentabilidad soportando un nivel de riesgo más elevado, mientras que otros se conformarán con una rentabilidad menor a cambio de un menor nivel de riesgo.

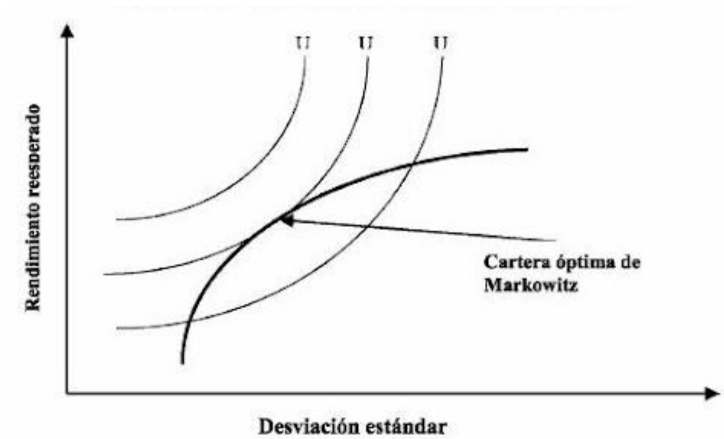
Estas curvas representan todas las posibles combinaciones de riesgo y rendimiento que proporcionan la misma utilidad al inversor. Es decir, que el inversor es indiferente a las combinaciones de riesgo-rendimiento que se encuentren dentro de la misma curva de indiferencia. La forma de la curva de indiferencia dependerá de su función de utilidad y esta será, naturalmente, distinta para cada inversor.

En la práctica de gestión de carteras no se determinan las curvas de indiferencia de utilidad de los inversores.

3- Obtención de la cartera óptima

El punto donde se intersecan las curvas de indiferencia y la frontera eficiente, constituirá la combinación óptima de títulos riesgosos para cualquier inversionista.

Figura N° 7 – Cartera óptima de Markowitz



Fuente: Adaptado de "Fundamentos de Inversiones" 3ra Edición, (pág. 149), por Alexander Sharpe y Bailey

Lo más importante del modelo de Markowitz, es demostrar que no es el riesgo del activo lo más significativo al invertir, sino la contribución de cada título al riesgo de la cartera (covarianza).

La decisión de poseer un activo no debe tomarse por comparación de su rendimiento y varianza con respecto a otros, sino que depende de los activos que desee poseer.

Esto se sustituye por la determinación del perfil de riesgo del inversor a través de cuestionarios. Así, una vez identificado el inversor como de perfil más o menos conservador, moderado o arriesgado, se puede estimar un valor de varianza de rentabilidad acorde a la categoría pertinente a fin de determinar su cartera óptima.

1.4 MODELO DE SHARPE

El ratio de Sharpe es una medida de performance desarrollada por el Nobel William Sharpe a través de su trabajo presentado en el año 1966. Este ratio presenta la relación entre el riego y la rentabilidad de una inversión, es decir expresa el exceso de rentabilidad o rendimiento obtenido por cada unidad de riesgo total de una inversión y produce el grado de deseabilidad de la inversión por parte de los inversores, ya que los mismos querrán saber si la mayor rentabilidad proporcional le compensa con el aumento de riesgo.

En su presentación, Sharpe realizó el cálculo de la desviación y la media de las rentabilidades anuales de distintos fondos de inversión americanos. A partir de esas rentabilidades, obtuvo la tasa libre de riesgo correspondiente a ese tiempo, y en consecuencia dedujo la prima de riesgo por cada fondo. Posteriormente dividió ese resultado por el riesgo (desviación)

Quedando su cálculo de la siguiente manera:

$$S_c = \frac{R_p - r_f}{\sigma_p} \qquad (10)$$

Donde:

- ✓ S_C es el ratio o índice de Sharpe.
- ✓ Rp es el rendimiento de la cartera.
- ✓ rf la tasa libre de riesgo.
- ✓ σp es el riesgo de la cartera.

Al ser el ratio de Sharpe una medida de rendimiento ajustada al riesgo, nos permite comparar las distintas inversiones con los activos libres de riesgo, y por medio de esto se puede evaluar el comportamiento de un portfolio.

Generalmente, cuanto mayor sea el valor del ratio, más atrayente será el rendimiento ajustado al riesgo, sin embargo un ratio negativo indicaría que el activo libre de riesgo se comportaría mejor que la cartera con riesgo. Entre dos carteras con diferentes rentabilidades esperadas y diferentes varianzas, se escogerá aquella con un mayor ratio de Sharpe, o en una situación donde un par de activos tengan la misma rentabilidad esperada (o riesgo), se podrá utilizar el valor de la razón de Sharpe de cada uno de los activos y comparar cuál es el que tiene mejor relación rentabilidad/riesgo.

1.5 COMPARACIÓN CON UN ÍNDICE DE REFERENCIA "BENCHMARK"

El benchmark es un índice o cartera utilizado para el análisis del progreso de un mercado y que además permite medir los resultados obtenidos por carteras o títulos similares. Por tanto, el benchmark es considerado como un elemento base para la comparación de resultados.

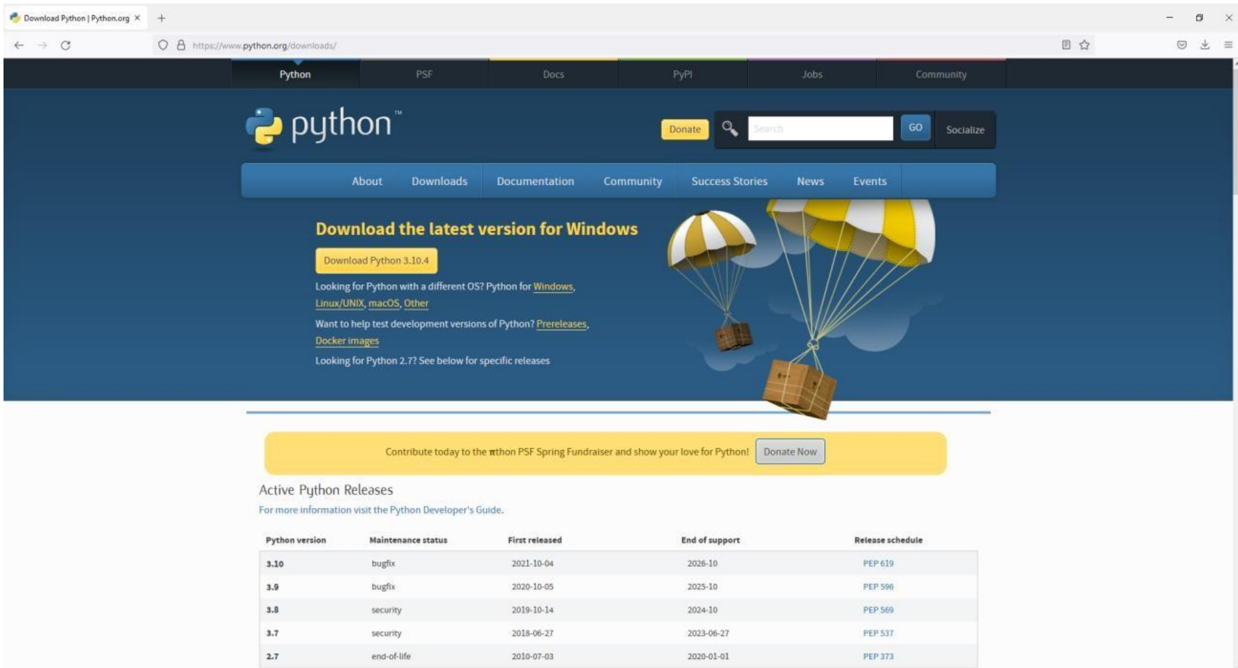
La característica fundamental es que admite relacionar los resultados obtenidos en una cartera, así como establecer los objetivos de la misma, considerado una cartera alternativa de referencia.

1.6 INTRODUCCIÓN DEL LENGUAJE DE PROGRAMACIÓN PYTHON

Para llevar a cabo la puesta en marcha del modelo de Markowitz, se hará uso del lenguaje de programación Python. El lenguaje Python es de propósito general, poderoso y flexible, a la vez que es sencillo y fácil de aprender. Es un lenguaje de alto nivel interpretado, dinámico y multiplataforma. Python permite procesar cualquier tipo de dato fácilmente, lo cual aporta al usuario mayor versatilidad. Asimismo, se trata de un software de código abierto, por lo que su uso es libre y gratuito. Este lenguaje se puede obtener a través de la página oficial de Python (ilustración 1). Todas estas características hacen

Que sea un lenguaje muy utilizado entre los programadores, de aquí se explica su gran popularidad.

Figura N° 8 – Página oficial de Python



Fuente: Adaptado de "https://www.python.org/"

Pero practicar en proyectos se convierte en una limitación, ya que se necesitan PC de gama alta para este tipo de cargas de trabajo. La respuesta a este problema es Google Colaboratory o Colab, para abreviar. Un desarrollador de programas Python puede utilizar este cuaderno para escribir y ejecutar códigos aleatorios de programas Python simplemente utilizando un navegador web. Para utilizar Colab, no se necesita instalar ni tiempo de ejecución ni actualizar el hardware del ordenador para cumplir los requisitos de carga de trabajo intensiva de CPU/GPU de Python.

Características de Google Colab o Colabority:

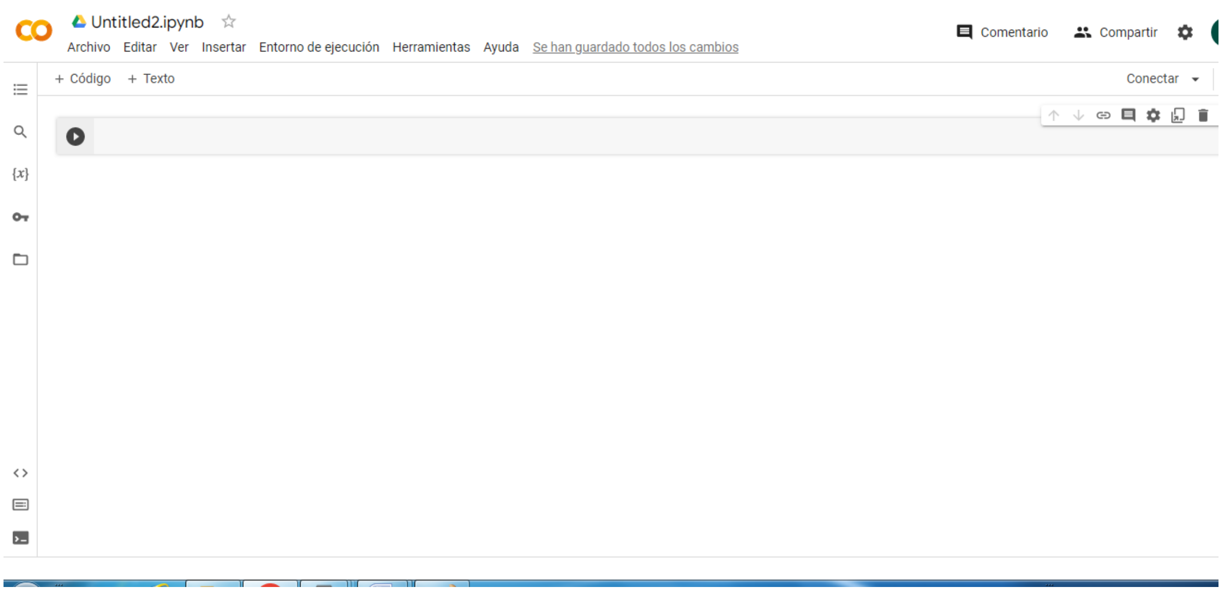
- Es una herramienta apoyada en la nube. Se puede empezar a codificar modelos de Machine Learning (rama de la inteligencia artificial) y ciencia de datos utilizando un Colab.

- Es gratuito con recursos limitados. Sin embargo, no se debe esperar que pueda almacenar modelos de inteligencia artificial o aprendizaje automático indefinidamente en la infraestructura gratuita.

- Cuenta con bibliotecas de ciencia de datos pre instaladas y populares.

- puede compartir fácilmente el cuaderno de código con colaboradores para codificar en tiempo real.

Figura N° 9 – Página Google Colaboraty



Fuente: Adaptado de "https://colab.research.google.com/drive/1YMjRbmNyG04jfwZ7-YFEk2Vo7v5vYjX-"

Retomando las características de Python, se trata de un lenguaje interpretado. Es decir que hace uso de un intérprete, lo cual genera una ventaja porque los compiladores no pueden traducir y ejecutar a medida, sino que los segundos tienen que compilar el código completo. Otra ventaja sería que el intérprete sí es capaz de guardar datos en memoria mientras que el compilador, una vez se ha ejecutado el programa, elimina los datos de la memoria una vez que se termina la ejecución. Por otra parte, es un lenguaje con un fuerte tipiado dinámico. Esto significa que las variables pueden contener distintos tipos de datos y que este tipo no cambia de forma repentina, sino que el usuario tiene que indicar al intérprete que quiere cambiar de un tipo a otro.

También se trata de un lenguaje multi paradigma, ya que permite distintos paradigmas de programación como por ejemplo la programación orientada a objeto, estructurada, imperativa y funcional, esta última en menor medida. Gracias a esto se considera que Python es un lenguaje muy versátil y por eso es de los más utilizados en la actualidad.

Estas características son consecuencia de una serie de principios recogidos en *The Zenof Python* (*El Zende Python* en español), los cuales se pueden encontrar a través de la web oficial de Python en la referencia. Estos principios son:

- Beautiful is better than ugly (Lo bonito es mejor que lo feo).
- Explicit is better than implicit (Explícito es mejor que implícito).
- Simple is better than complex (Simple es mejor que complejo).
- Complex is better than complicated (Complejo es mejor que complicado).
- Flat is better than nested (Plano es mejor que anidado).
- Sparse is better than dense (Disperso es mejor que denso).
- Readability counts (La legibilidad cuenta).
- Special cases aren't special enough to break the rules (Los casos especiales no son lo suficiente como para saltarse las normas).
- Although practicality beats purity (Aunque la practicidad le gana a la pureza).
- Errors should never pass silently (Los errores no deben pasar nunca silenciosamente).
- Unless explicitly silenced (A menos que sean silenciado explícitamente).
- In the face of ambiguity, refuse the temptation to guess (Ante la ambigüedad, rehúsa la tentación de adivinar).
- There should be one—and preferably only one—obvious way to do it. (Debe de haber una–y preferiblemente sólo una única–forma obvia de hacerlo)
- Now is better than never (Ahora es mejor que nunca).
- Although never is often better than right now (Aunque nunca es con frecuencia mejor que ahora mismo).
- If the implementation is hard to explain, it's a bad idea (Si la implementación es difícil de explicar, es una mala idea).
- If the implementation is easy to explain, it may be a good idea (Si la implementación es fácil de explicar, puede ser buena idea).
- Namespaces are one honking great idea--let's do more of those! (Los espacios de nombre son una gran idea--¡hagamos más cosas de esas!)

El lenguaje de programación Python, como se ha mencionado anteriormente, se trata de un lenguaje interpretado. Esto significa que las líneas que se escriban se ejecutan directamente, no requiere de una compilación previa para pasar las instrucciones del programa a instrucciones en lenguaje para que la máquina las entienda. Estas instrucciones se pueden pasar al intérprete de distintas formas. La primera de ellas es a través de la ventana de comandos o *Command Prompt.*

Una vez se ha iniciado Python se puede escribir directamente el código desde la consola. Una forma de introducir varias líneas de instrucciones es escribiendo en la ventana de comandos Python seguido por un espacio y el nombre de un archivo con extensión .py a continuación. Estos archivos se pueden crear a partir de un archivo de extensión .txt, seleccionando la herramienta "Guardar como" y añadiendo ".py" al final del nombre del documento. De esta forma se pueden ejecutar muchas líneas de código de una sola vez.

Sin embargo, esta forma de trabajar puede resultar difícil. La mayoría de los programadores eligen utilizar editores de texto o Entornos de Desarrollo Integrado (*Integrated Development Environment* o *IDE*). Estos presentan una interfaz más cómoda para el usuario. Pueden incluir ciertas funcionalidades como la predicción de la escritura, cierre de paréntesis o llaves automático, resalto de la escritura mediante colores, personalización de la interfaz, etc. Muchos de estos IDEs y editores de texto se pueden obtener de manera gratuita.

1.6.1 VARIABLES DE PYTHON

En primer lugar, la principal diferencia entre Python y otros muchos lenguajes de programación es que Python no requiere que el usuario declare la variable. Para declarar variables en otros lenguajes como C, es necesario escribir un comando con el tipo de la variable seguido del nombre que se le quiera dar a ésta. De esta manera, el sistema reserva espacio en la memoria para guardar el valor que se le quiera asignar a esa variable. Usando Python, no es necesario declarar las variables. La variable se crea directamente en el momento en el que se le asigna un valor. La variable creada tendrá el tipo de dato que más se ajuste al valor que se asigne en el momento de su creación. También permite Python una forma de especificar el tipo de variable que se quiere crear. Esto se hace escribiendo el nombre de la variable, seguido del signo de igual junto a la abreviación del tipo con el valor que se desea guardaren la variable entre paréntesis.

Los valores que se asignan a las variables pueden ser numéricos o una cadena de caracteres. En las variables se pueden guardar uno o varios valores. En función de qué valor se guarde y de cuántos, Python presenta distintos tipos de variables. A continuación, se describen los principales tipos que se encuentran, indicando a su vez algunos ejemplos.

1.6.1.1. VARIABLES DE TEXTO

Las variables que guardan textos on del tipo *string*(*str*). En este tipo de variable se puede guardar una cadena de caracteres. Las cadenas de caracteres pueden incluir números, espacios y otros signos de puntuación. Para indicar al intérprete que se quiere crear una variable de tipo *string* se colocan comillas a ambos lados del valor que se quiere guardar en la variable. En este caso, se han utilizado las dos formas de crear una variable *string*.

Figura N° 10 – Ejemplo variable de texto

```
# nombre_variable = tipo_variable ( valor )
# nombre_variable = valor

x = str("Hola Mundo")
y = "Hello World"
```

Fuente: Elaboración propia

1.6.1.2. VARIABLES NUMÉRICAS

El otro tipo de variable que tiene Python según su valor es el numérico. Dentro de las variables numéricas, existen 3 tipos distintos. El tipo entero (*int*), flotante (*float*), complejo (*complex*) y booleano (*bool*). En las variables de tipo entero, se puede almacenar cualquier número entero. En las variables tipo *float* se pueden guardar números que incluyan decimales. En Python se utiliza el punto para separar la parte entera de la parte decimal. Además, Python soporta las variables booleanas, que son aquellas que solo pueden tomar valor verdadero, cuando es distinto de 0 o falso en caso contrario.

1.6.1.3. VARIABLES DE TIPO SECUENCIA

Python incluye varios tipos de variable en el que se pueden almacenar varios valores a la vez. En este apartado se tratarán la *lista* y la *tupla*. El tipo *string* también puede recibir un trato bastante similar al tipo secuencia. Sin embargo, al ser el tipo *string* una cadena de caracteres, no se tratará en este apartado.

En primer lugar, Python introduce las listas. Se trata de un tipo de variable muy versátil. Para definir una lista, se escribe el nombre de la variable, seguido del signo de igualdad y a continuación unos corchetes. De esta forma se crea una lista vacía. Para incluir otros datos dentro de la lista al crearla, solo hay que escribir los valores que se deseen dentro de los corchetes, de la misma forma que si se estuviera creando la variable en la que se guarda dicho valor. A continuación, se muestra un ejemplo en el que se crea una lista con un número entero, un flotante, una cadena de caracteres y un booleano y se le pide al intérprete que muestre lo que contiene la lista por pantalla:

Figura N° 11 – Ejemplo variables de tipo secuencia

```python
# nombre_lista = [ valor1 , valor2 , ... ]

lista = [ 13 , 3.78 , "Hello World!", bool( 0 ) ]

# print es una función para mostrar su argumento por pantalla
# print ( variable )   o   print ( valor )
print(lista)
```

```
[13, 3.78, 'Hello World!', False]
```

Fuente: Elaboración propia

Las listas presentan un carácter heterogéneo y mutable. Esto significa que puede estar formada por varios elementos de distintos tipos, incluyendo otras listas, y que todos sus elementos son modificables una vez se ha creado la lista. Se puede acceder a las componentes de una lista usando el nombre de la variable seguido del número del hueco que ocupa dentro de la lista entre corchetes. La primera componente tendrá el índice 0 y la última será el tamaño de la lista menos1.

En segundo lugar, Python proporciona la posibilidad de crear *tuplas*. Las tuplas son como las listas, pero inmutables, es decir, una vez se ha creado una tupla, no se pueden modificar sus componentes. De esta manera, para crear una tupla, se procede de la misma manera que para una lista, pero usando, en lugar de los corchetes, paréntesis a ambos lados de los valores que se quieran incluir en la tupla.

1.6.1.4. VARIABLES DE TIPO MAPEO

En el lenguaje Python se incluyen los diccionarios. Los diccionarios son variables que recogen cualquier tipo de valor. Además, estos valores se guardan junto a una etiqueta que recibe el nombre de *key*.

Este tipo de variable es similar a las estructuras de datos del lenguaje C. Para crear este tipo de variable, se hace introduciendo las etiquetas y los valores entre unas llaves. Se muestra a continuación un ejemplo:

Figura N° 12 – Ejemplo variables de tipo mapeo

```
# nombre_variable = {
#   "etiqueta1": "valor1",
#   "etiqueta2": "valor2", ...
# }

fecha = {
    "Día": "25",
    "Mes": "Abril",
    "Año": 2022
}

print( fecha )
```

```
{'Día': '25', 'Mes': 'Abril', 'Año': 2022}
```

Fuente: Elaboración propia

Las características de los diccionarios son:

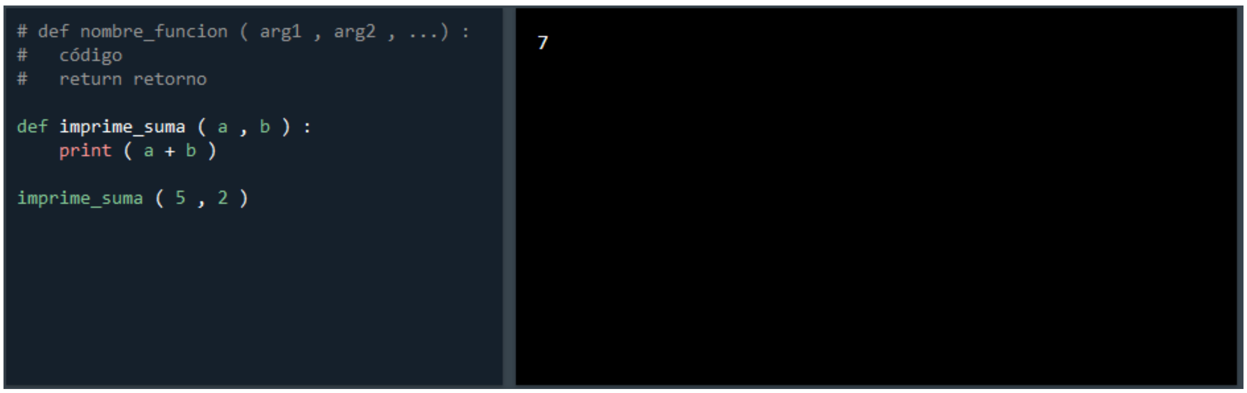
- Mantienen el orden en el que las etiquetas son escritas en el código.
- Son mutables.
- Son anidables, es decir, un diccionario puede contener a otro.

Esta variable es muy útil puesto que se le puede poner cualquier etiqueta, por lo que es más manejable que los índices de los vectores de C o las listas y las tuplas de Python.

1.6.2 FUNCIONES EN PYTHON

Las funciones en Python se definen con el comando *def*. Para definir una función, no es necesario hacerlo al principio del programa, se puede definir y llamar a esa función en la siguiente línea de código. La forma de crear una función será escribir el comando *def*, seguido del nombre que se le desee dar a la función junto con tantos argumentos como se le quieran introducir separados por comas y finalizando con dos puntos. Las sentencias que se asignen a la función deben estar identadas, ya que el intérprete no considera parte de la función lo que no esté identado de igual manera. Si se desea que la función devuelva un valor, se utiliza el comando *return*. Para utilizar una función, se escribe el nombre de esta seguido de tantos valores como argumentos se hayan escrito en el momento de la definición. Este resultado se puede guardar en una variable, aplicarlo en otra expresión matemática, imprimirlo por pantalla, etc.

Figura N° 13 – Ejemplo definición de una función en Python



Fuente: Elaboración propia

Por otra parte, Python ofrece la posibilidad de crear funciones *lambda*. Este tipo de funciones son funciones con un código pequeño, definen estas funciones como:*"Python lambdas are only ashor thand notation if you're too lazy to define a function".* Lo que se puede traducir como que las funciones lambda de Python son solo una notación abreviada si eres demasiado vago para definir una función. Estas funciones pueden tener tantos argumentos como se quiera pero solo puede tener una expresión. Estas funciones también son llamadas anónimas. Para definir una función lambda, se escribe el nombre de la función con el signo de igual y del comando *lambda* acompañado por sus argumentos separados por comas. En la misma línea, se escriben dos puntos, seguidos por la expresión que se le quiera asignar a la función.

Figura N° 14 – Ejemplo funciones lambda

```python
# nombre_función = lambda arg1 , arg2 , ... : expresion

cuadrado = lambda a : a*a

print ( cuadrado ( 7 ) )
```

```
49
```

Fuente: Elaboración propia

1.6.3 LIBRERIAS EN PYTHON

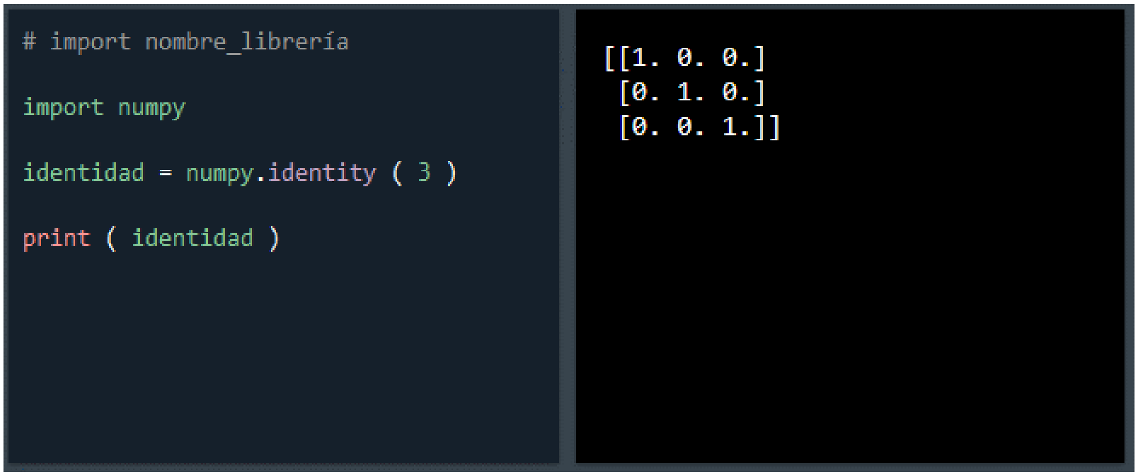
Una de las principales características de Python es que tiene una gran comunidad que es partícipe en el desarrollo del software. Esto significa que es muy fácil generar código y que este pueda ser utilizado por cualquier usuario. Cuando un usuario genera unas líneas de comandos que son de gran utilidad, incluyendo variables, funciones, etc., se guarda todo este en una biblioteca.

Los tipos de bibliotecas se pueden clasificar en:

- Deeplearning
- Machine learning
- Cálculo numérico
- Visualización
- Procesamiento del lenguaje natural

Para comenzar, generalmente se comienza por importar las bibliotecas necesarias para llevar a cabo las acciones siguientes. Cuando se importa se utiliza la palabra reservada *import*. Este comando se escribe seguido por el nombre de la biblioteca que se quiera utilizar. Para acceder a las funcionalidades (métodos) de la biblioteca hay que utilizar el nombre de la biblioteca (o su alias) y el nombre del método que se quiera utilizar con un punto entre ambos. Por ejemplo, para una matriz identidad de 3x3:

Figura N° 15 – Ejemplo importación de bibliotecas



Fuente: elaboración propia

Para el cálculo de la cartera óptima, el manejo de los datos financieros se realiza a través de bibliotecas. Entre las principales para realizar el presente trabajo se encuentran:

- PANDAS: esta biblioteca se utiliza más en el área de Data Science y Machine Learning, es especializada en el manejo y análisis de grandes bloques de datos. Panda dispone de tres bloque de datos diferentes, *Series* (estructura de una dimensión), *DataFrame*(estructura de dos dimensiones, tablas) y *Panel* (estructura de tres dimensiones, cubo). Podemos destacar las principales características de esta biblioteca, en cuanto permite leer y escribir en forma fácil ficheros en formato Excel, CSV y bases de datos SQL, permite trabajar con series temporales, brinda métodos para reordenar, dividir y combinar conjuntos de datos, entre otros

- NUMPY: se utiliza particularmente para la programación en Data Science, ya que se basa en cálculos científicos muy complejos. Numpy tiene su propia estructura de datos incorporada llamada Array que es similar a la lista normal de Python pero puede almacenar y operar con datos de manera mucho más eficiente. Dentro del Array podemos encontrar funciones estadísticas útiles para encontrar la desviación estándar, la media, la varianza, etc.

- MATPLOTLIB: es una biblioteca que permite generar gráficas estáticas, animadas e interactivas, a partir de datos contenidos en listas o arrays en Python. Es una herramienta muy completa que permite generar visualizaciones de datos muy detalladas.

- YFINANCE: es un módulo que permite volcar los datos directamente desde la web de referencia mediante la conexión a su Api pública, en este caso sería la página oficial de Yahoo Finance. Con esta web podemos obtener los datos históricos del valor de los activos que se utilizaran como variables, además de los principales indicadores como son el volumen negociado, pago de dividendos, precio de apertura, máximo, mínimo, etc.

- PYPORTFOLIOOPT: es una biblioteca que permite implementar los métodos de optimización de cartera de Markowitz, incluidas las técnicas clásicas de frontera eficiente de Black Litterman, así como desarrollos más recientes en el campo.

CAPÍTULO II

ESTUDIO EMPÍRICO

2.1 ÁMBITO DE APLICACIÓN

En el mundo de las Bolsas de Valores, la más importante en la actualidad es el NYSE (New York Stock Exchange), mercado que utilizará como referencia el presente trabajo. Este mismo no controla los precios de los títulos, sino que asegura la validez de un mercado ordenado. Entre los miembros que podemos ver interactuando dentro de estos mercados podemos encontrar, los intermediarios por comisión, intermediarios por contratación, especialistas, y operadores autorizados.

- **Intermediarios a comisión** se encargan de realizar las operaciones y ejecutan las órdenes de los clientes.
- **Intermediarios por contratación** ayudan a realizar su trabajo a otros miembros y operan solo para sí mismos, al no estar autorizados a tratar con el público directamente.
- **Especialistas** crean un mercado para valores cotizados asignados al punto de operaciones. Su propósito es mantener el mercado justo y ordenado. Sus ingresos provienen de comisiones o de la diferencia entre el precio de compra y venta cuando actúan como agentes.
- **Operadores autorizados** realizan operaciones por su cuenta y se ahorran comisiones al ser miembros de la bolsa.

En cada bolsa de mercado existen diferentes índices o promedios, los cuales pretenden indicar el comportamiento en el tiempo de los movimientos de los valores y títulos que están registrados en el mercado.

Los índices bursátiles sirven para dar referencia muy importante a los gestores de cartera, ya que se habla más de ellos que de los mercados a los que representan, puesto que logran medir el comportamiento de su mercado y así compararlo con la evolución de una cartera de valores determinada.

El índice Standard & Poor's 500, también conocido como S&P 500, es uno de los índices bursátiles más importantes de EEUU y se lo considera el índice más representativo de la situación real del mercado.

Se fundamenta en la capitalización bursátil de las 500 empresas más grandes que poseen acciones que cotizan en las bolsas NYSE. Se diferencia de otros índices de mercado, tales como el Dow Jones Industrial Average o el índice Nasdaq Composite, en la diversidad de rubros que lo componen y en su metodología de ponderación.

Los orígenes del S&P500 datan de 1923 cuando la empresa Standard Statistics establece un índice que cubría 233 compañías. Luego fue formulado en el año 1957 al incluir a las 500 compañías más grandes de Estados Unidos, incluyendo 400 empresas industriales, 20 del sector transporte, 40 de servicios y 40 financieras. Todas estas empresas para ser incluidas debieron cumplir ciertos requisitos, como son: superar los 8.200 millones de dólares como mínimo de capitalización, que al menos el 50% de los títulos estén en el mercado y tener cuatro trimestres seguidos con resultados positivos. Para que esto se sostenga, el índice es revisado periódicamente para quitare introducir compañías que dejen de cumplir o empiecen a cumplir estos requisitos.

La elección de las compañías que tributan dentro del S&P500 se realiza por capitalización. Este índice se pondera de acuerdo a la capitalización de mercado de cada una de las empresas, o sea los movimientos en los precios de las acciones con mayor capitalización (precio de cada acción multiplicado por el número de acciones emitidas) poseen una influencia mayor sobre el valor del índice que las empresas que tienen menor capitalización, por ejemplo, Apple tiene mucho más peso que una compañía con una menor valoración bursátil. Esto supone que el índice se ve más influenciado en su movimiento por aquellas empresas que pesan más dentro del índice. Por esto mismo expresa mejor la situación real del mercado y es más sensible que el Dow Jones.

La cotización del índice, se calcula mediante una media aritmética, por lo que se mide asignando una puntuación estándar a los instrumentos o empresas que son tomadas en cuenta para el cálculo del mismo, y luego realizando una sumatoria o resta simple a fin de encontrar las posibles variaciones.

El mundo de las finanzas se mueve alrededor de estos índices, por lo que tener un retorno de portfolio mayor al retorno del indicador es considerado un éxito.

En la siguiente figura podemos observar la evolución del índice S&P500 a lo largo de los años, y cómo su rendimiento a largo plazo siempre ha sido ascendente.

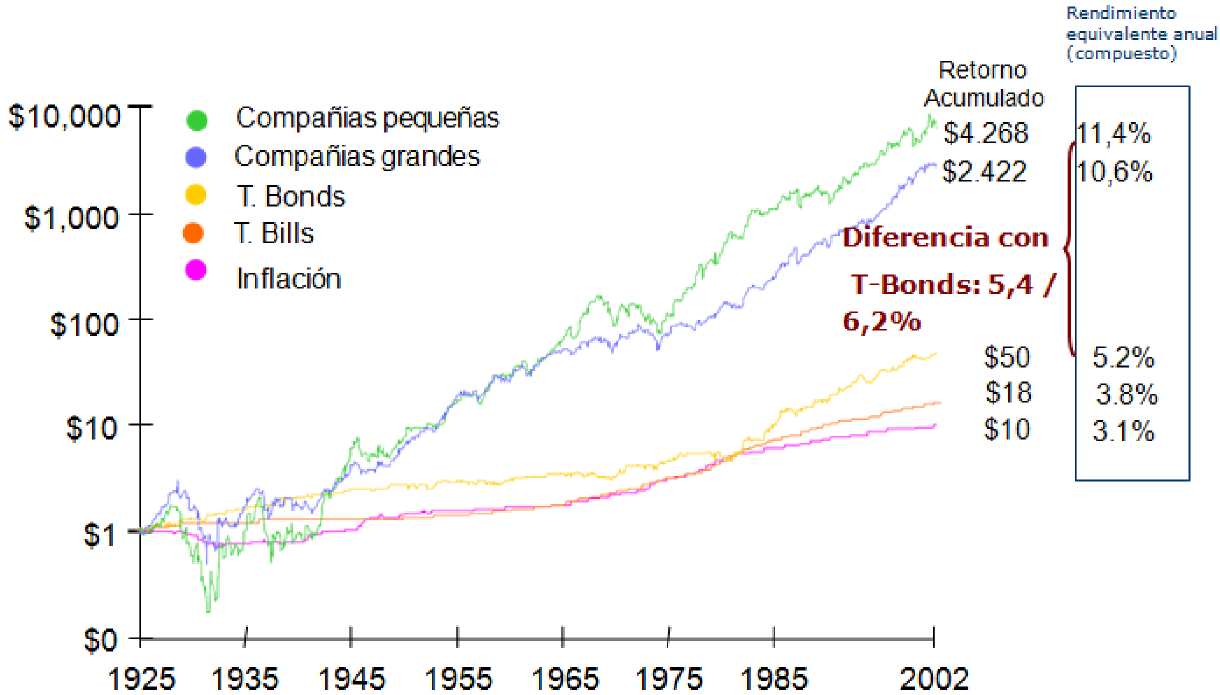
Figura N° 16 – Evolución de precios del índice S&P 500

Fuente: Yahoo Finance

Por otro lado, según el estudio que realizaron en 1982 Roger Ibbotson y Rex Sinquefeld sobre la evolución de los rendimientos anuales para activos financieros en Estados Unidos, en el cual los activos considerados fueron la Inflación, Letras del tesoro de Estados Unidos (*T-bills*) con vencimiento a tres meses, Bonos a largo plazo del Gobierno de Estados Unidos a veinte años *(T-bons)*, Acciones comunes de compañías grandes y Acciones comunes de compañías pequeñas (integrada por las acciones de 20% de las compañías pequeñas registradas en la Bolsa de Valores de Nueva York), todos calculados a una tasa equivalente anual.

En el siguiente gráfico podemos observar cómo evolucionó, a lo largo de los años tomados como referencia, la inversión de un dólar suponiendo la reinversión de sus intereses en el mismo activo. Y se puede concluir, que, a pesar de tener mayor volatilidad, los inversores han sido recompensados a larga con mayores rendimientos o retornos debido a las acciones, obteniendo una diferencia de retorno del 6,2% con los Bonos, sin dejar de lado que estos instrumentos también significan una buena inversión a largo plazo para hacer frente a la inflación.

Figura N° 17 – Rendimiento de los principales Activos de Estados Unidos durante el año 1925-2022

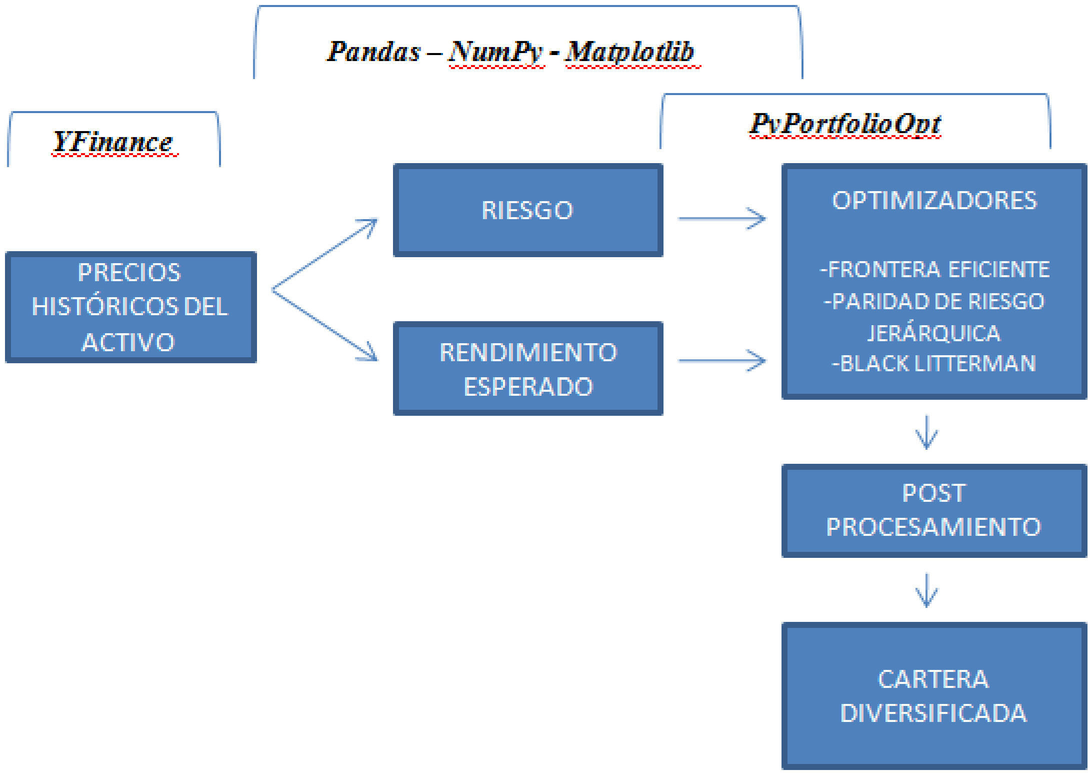


Fuente: Fuente: Libro "Finanzas corporativas, un enfoque latinoamericano". G.L Dumrauf. Pág. 187

2.2 APLICACIÓN DEL MODELO EN EL LENGUAJE DE PROGRAMACIÓN PYTHON

El gráfico que se presenta a continuación es una herramienta visual que concentra de manera integral y detallada todos los pasos inherentes al proceso de obtención de la cartera óptima. Además, este resumen brinda una exposición de cada una de las bibliotecas empleadas con el propósito de llevar a cabo dicho proceso de manera efectiva.

Figura N° 18 – Bibliotecas utilizadas para el proceso de optimización de cartera en Python



Fuente: Adaptado de "Optimización de cartera de acciones con Python", (pág. 8), 2022 por Bartolomeo Alejandro y Machín Gustavo

A continuación, pasaremos a detallar cada paso realizado en Python por medio de Google Colaboratory.

- **OBTENCIÓN DE DATOS**

Como primer punto se empleará la herramienta para obtener los precios de las acciones que componen el índice S&P500.

Para lograr lo anterior se instala e importa la biblioteca, luego se determinará una lista de los tickets (símbolos) de los títulos que componen el índice con los cuales formaremos nuestras carteras eficientes, estos datos serán obtenidos a través de la información proporcionada por la página de Wikipedia.

Posteriormente, con la función "*download*" podemos conectarnos y descargar la información necesaria como el precio de apertura y cierre de las acciones solicitadas previamente y según los parámetros establecidos como la fecha indicada.

```
pip install yfinance
importyfinanceasyf
data="https://en.wikipedia.org/wiki/List_of_S%26P_500_companies#S&P_
500_component_stocks"
tickers= lista["Symbol"]
df = yf.download(lista_indice, start='2020-12-01', end='2023-12-01',
rounding=True)
precios = df['AdjClose']
precios
```

Para concluir esta parte se selecciona una columna (*Adjusted Close*) que corresponde a precio de cierre ajustado y se define un *data frame* llamado precios.

Figura N° 19 – Precio de las acciones

Ticker	A	AAL	AAPL	ABBV	ABNB	ABT	ACGL	ACN	ADBE	ADI	...	WTW	WY	WYNN	XEL	XOM	XYL	YUM	ZBH	ZBRA	ZTS
Date																					
2020-12-01	112.80	14.27	120.27	90.13	NaN	102.01	33.02	240.18	479.15	132.10	...	203.87	25.06	101.06	61.52	33.15	93.43	100.22	138.21	372.17	157.06
2020-12-02	112.03	14.86	120.63	90.96	NaN	101.69	33.11	237.70	481.26	132.04	...	202.35	24.74	103.80	61.76	34.39	92.53	98.94	139.86	367.21	154.72
2020-12-03	111.06	16.09	120.49	90.74	NaN	101.11	32.59	237.95	484.28	131.76	...	202.66	25.40	107.11	60.53	34.63	93.12	98.50	138.40	372.37	152.89
2020-12-04	112.59	16.40	119.81	92.94	NaN	101.46	34.17	241.29	486.00	135.74	...	204.25	26.46	112.06	59.13	35.89	94.72	99.77	141.00	378.47	153.74
2020-12-07	111.86	17.21	121.28	91.69	NaN	100.78	33.25	237.35	492.25	135.50	...	199.45	26.04	109.91	59.90	35.22	94.51	99.18	137.70	373.83	154.23
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2023-11-24	126.20	12.31	189.47	136.09	128.37	101.87	86.60	331.54	619.43	181.34	...	242.64	31.43	87.87	59.38	102.77	102.96	127.27	112.99	228.92	179.29
2023-11-27	124.60	12.18	189.29	136.50	129.00	101.71	87.00	329.94	619.27	182.05	...	243.40	31.12	86.63	59.31	102.17	102.88	127.05	113.40	231.50	177.88
2023-11-28	124.30	12.17	189.90	135.51	127.56	101.07	84.45	330.07	623.32	181.49	...	240.05	31.04	84.52	59.82	102.11	102.34	125.76	112.65	233.89	176.07
2023-11-29	127.17	12.23	188.87	135.92	126.48	102.62	82.38	330.85	617.39	181.00	...	239.15	31.19	83.26	58.87	100.58	103.20	124.89	113.48	236.62	174.89
2023-11-30	127.38	12.43	189.45	139.74	126.34	103.28	83.69	330.65	611.01	181.66	...	244.65	31.04	84.00	59.69	100.97	104.82	124.92	115.86	236.98	175.77

755 rows × 503 columns

Fuente: Elaboración propia

A modo de realizar un análisis preliminar, para visualizar gráficamente la cotización de las acciones utilizaremos la biblioteca *matplotlib*. Al ser demasiadas empresas, en el gráfico se analizarán las acciones con mayores precios para evitar la aglomeración de líneas.

Figura N° 20 – Precio de las acciones a lo largo del tiempo

Fuente: Elaboración propia

- **CALCULAR RENDIMIENTO Y RIESGO DE LAS ACCIONES**

Para lograr obtener la cartera eficiente en base a este modelo, primeramente debemos averiguar los rendimientos esperados y la matriz de varianza y covarianza de las acciones en función de los datos obtenidos anteriormente.

Esto se puede calcular utilizando de la biblioteca *Pandas* la función *pct_change.* Esta es una función que calcula el cambio porcentual entre cada par de elementos consecutivos en los datos de precios, devolverá una serie de tiempo que representa los rendimientos diarios.

```
rendimientos = precios.pct_change()[1:]
rendimientos
```

Figura N° 21 – Rendimiento de las acciones

Ticker Date	A	AAL	AAPL	ABBV	ABNB	ABT	ACGL	ACN	ADBE	ADI	...	WTW	WY	WYNN	XEL
2020-12-02	-0.006826	0.041345	0.002993	0.009209	NaN	-0.003137	0.002726	-0.010326	0.004404	-0.000454	...	-0.007456	-0.012769	0.027113	0.003901
2020-12-03	-0.008658	0.082773	-0.001161	-0.002419	NaN	-0.005704	-0.015705	0.001052	0.006275	-0.002121	...	0.001532	0.026677	0.031888	-0.019916
2020-12-04	0.013776	0.019267	-0.005644	0.024245	NaN	0.003462	0.048481	0.014037	0.003552	0.030206	...	0.007846	0.041732	0.046214	-0.023129
2020-12-07	-0.006484	0.049390	0.012269	-0.013450	NaN	-0.006702	-0.026924	-0.016329	0.012860	-0.001768	...	-0.023501	-0.015873	-0.019186	0.013022
2020-12-08	0.022796	0.024404	0.005112	0.017559	NaN	-0.003572	-0.002105	0.002275	0.006155	0.016827	...	0.006718	0.018817	-0.000091	-0.008514
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2023-11-24	0.021201	0.000000	-0.007023	0.001472	-0.005192	0.001672	0.007563	0.002722	-0.000468	0.002100	...	-0.000782	-0.001588	0.008840	0.001518
2023-11-27	-0.012678	-0.010561	-0.000950	0.003013	0.004908	-0.001571	0.004619	-0.004826	-0.000258	0.003915	...	0.003132	-0.009863	-0.014112	-0.001179
2023-11-28	-0.002408	-0.000821	0.003223	-0.007253	-0.011163	-0.006292	-0.029310	0.000394	0.006540	-0.003076	...	-0.013763	-0.002571	-0.024356	0.008599
2023-11-29	0.023089	0.004930	-0.005424	0.003026	-0.008467	0.015336	-0.024512	0.002363	-0.009514	-0.002700	...	-0.003749	0.004832	-0.014908	-0.015881
2023-11-30	0.001651	0.016353	0.003071	0.028105	-0.001107	0.006431	0.015902	-0.000605	-0.010334	0.003646	...	0.022998	-0.004809	0.008888	0.013929

Fuente: Elaboración propia

Posteriormente con la siguiente función "*describe"*, la misma se utiliza comúnmente para obtener estadísticas descriptivas sobre un conjunto de datos de rendimientos. Estas estadísticas pueden incluir la media, la desviación estándar, el mínimo y máximo y percentiles.

```
rendimientos.describe()
```

Con esto podemos conseguir una tabla que muestra el rendimiento esperado *(mean)* y el riesgo *(std)* diario de cada una de los títulos.

Figura N° 21 - Tabla de estadísticos

Ticker	A	AAL	AAPL	ABBV	ABNB	ABT	ACGL	ACN	ADBE	ADI	...
count	754.000000	754.000000	754.000000	754.000000	747.000000	754.000000	754.000000	754.000000	754.000000	754.000000	...
mean	0.000317	0.000263	0.000759	0.000669	0.000384	0.000119	0.001363	0.000557	0.000581	0.000604	...
std	0.017711	0.029918	0.017688	0.013137	0.033726	0.014308	0.016099	0.016348	0.022613	0.019059	...
min	-0.065824	-0.119945	-0.058704	-0.079941	-0.134250	-0.093097	-0.068310	-0.059166	-0.167932	-0.078362	...
25%	-0.009038	-0.018554	-0.008912	-0.006366	-0.019802	-0.007726	-0.007279	-0.008320	-0.010885	-0.010881	...
50%	0.000517	-0.000598	0.000897	0.001128	-0.000874	0.000518	0.001628	0.000832	0.000869	0.001077	...
75%	0.010390	0.016840	0.011503	0.008185	0.018909	0.008299	0.011146	0.009722	0.013781	0.012000	...
max	0.087236	0.106246	0.088959	0.048956	0.133532	0.078174	0.096537	0.077254	0.103992	0.081865	...

8 rows × 499 columns

Fuente: Elaboración propia

A partir de este punto, para seguir los pasos de obtención de la cartera eficiente, utilizaremos la biblioteca PyPortfolioOpt, utilizando la función *expected_returns*, la cual está diseñada específicamente para ayudar en el cálculo de los rendimientos esperados de activos financieros o carteras de inversión.

Esta función proporciona diferentes métodos de estimación que permiten calcular los rendimientos, entre estos métodos podemos encontrar el rendimiento histórico medio, modelos de valoración de activos, análisis de regresión, Capm, entre otros.

En este caso utilizaremos el rendimiento histórico medio. Por defecto, la salida son rendimientos esperados anuales, en los demás parámetros se toman los precios de las acciones, no sus rendimientos *(returns_data)*, se utiliza la frecuencia de 252 días hábiles para un año, por último se utiliza la media logarítmica e interés compuesto para calcular los rendimientos anuales de los títulos.

```
rendEsp1 = expected_returns.mean_historical_return(precios,
returns_data=False, compounding=True, frequency=252,
log_returns=False)
rendEsp1
```

```
Ticker
A        0.041464
AAL     -0.045089
AAPL     0.164002
ABBV     0.157850
ABNB    -0.044764
           ...
XYL      0.039194
YUM      0.076409
ZBH     -0.057249
ZBRA    -0.140030
ZTS      0.038332
Length: 499, dtype: float64
```

Como podemos observar en la lista anterior, hay empresas que componen el índice que tienen rendimiento negativo. Por lo que en el presente trabajo eliminaremos esas acciones, ya que al basarnos en la teoría de la cartera de Markowitz, una de las finalidades es obtener el mayor rendimiento, por lo que los rendimientos negativos no nos servirán para obtener una cartera eficiente.

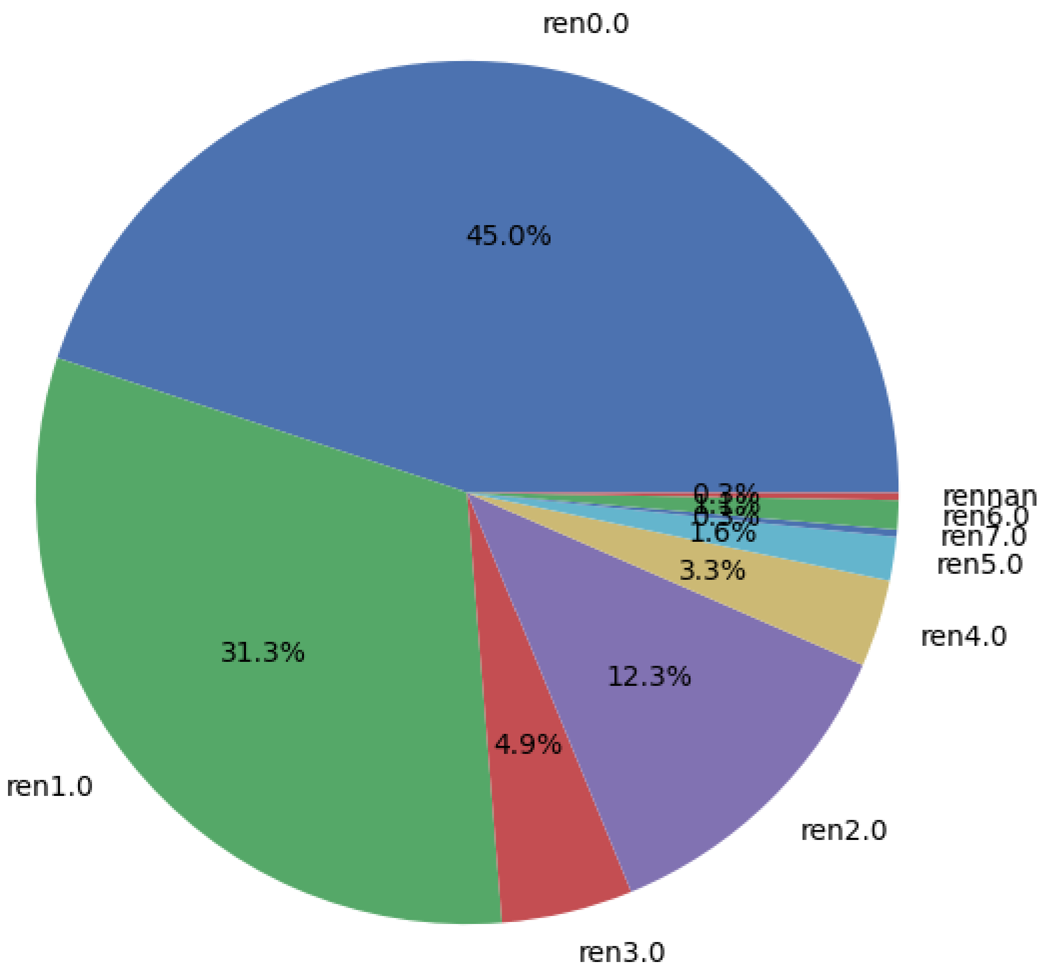
A través del siguiente código se filtra el DataFrame *"rendEsp1"* eliminando las filas que tienen valores negativos en una columna específica. Precisamente se comprueba si el valor en la columna correspondiente al índice **i** es menor que 0 se elimina.

```
foriin rendEsp1.index:
    if rendEsp1[i] <0:
```

```
Ticker
A       0.041464
AAPL    0.164002
ABBV    0.157850
ABT     0.004144
ACGL    0.364549
          ...
WY      0.074144
XOM     0.450988
XYL     0.039194
YUM     0.076409
ZTS     0.038332
Length: 367, dtype: float64
```

Aplicando los resultados obtenidos en un gráfico de torta, podemos analizar que la mayoría de las empresas tienen un rendimiento entre 0 (31.3%) y 1 (45%).

Gráfico N° 1 - Torta de rendimientos



Fuente: Elaboración propia

Para obtener el riesgo de las acciones utilizaremos el módulo *risk_models* de la biblioteca PyPortfolioOpt, se utiliza para calcular diferentes modelos de riesgo que son útiles en la gestión de carteras y la optimización de activos financieros.

Una de los métodos principales de este módulo es el que proporciona los métodos para calcular la matriz de varianza y covarianza de los rendimientos de los activos financieros en función de los datos históricos. Plasmando los riesgos individuales de cada una de las acciones y la covariación entre ellos, es decir cómo se relacionan entre sí las diferentes variables en un conjunto de datos. Esta matriz es esencial para medir el riesgo y la volatilidad de una cartera.

```
matrizCov1 = risk_models.sample_cov(precios1)
matrizCov1
```

Figura N° 22 - Matriz de varianza y covarianza de las acciones

Ticker	A	AAPL	ABBV	ABT	ACGL	ACN	ADBE	ADI	ADM	ADP	...	WMT
Ticker												
A	0.079056	0.037752	0.013522	0.034873	0.020780	0.040968	0.051553	0.046500	0.020404	0.032535	...	0.014527
AAPL	0.037752	0.078858	0.009037	0.026798	0.019428	0.044398	0.064422	0.052174	0.014960	0.034592	...	0.016094
ABBV	0.013522	0.009037	0.043493	0.015411	0.012308	0.013065	0.009580	0.011814	0.013222	0.012964	...	0.010126
ABT	0.034873	0.026798	0.015411	0.051606	0.013876	0.028294	0.034679	0.028178	0.012567	0.024206	...	0.009847
ACGL	0.020780	0.019428	0.012308	0.013876	0.065382	0.023132	0.018385	0.022188	0.024898	0.020267	...	0.009804
...	...	...	...	...	...	...	...	...	...	...	...	...
WY	0.033855	0.036308	0.010556	0.022076	0.022744	0.038891	0.039915	0.041050	0.028335	0.029769	...	0.014173
XOM	0.013471	0.014808	0.013521	0.005364	0.022486	0.016329	0.010179	0.019042	0.038333	0.016214	...	0.009893
XYL	0.041657	0.036600	0.010722	0.026570	0.028209	0.041830	0.043290	0.044710	0.024039	0.034615	...	0.016552
YUM	0.023503	0.023240	0.008511	0.017719	0.020085	0.024120	0.026393	0.025714	0.014584	0.020471	...	0.011224
ZTS	0.044559	0.038336	0.015207	0.031231	0.016534	0.039176	0.048877	0.038167	0.013294	0.031233	...	0.015626

367 rows × 367 columns

Fuente: Elaboración propia

- **OPTIMIZACIÓN DEL PORTFOLIO SEGÚN OBJETIVO**

En este punto, se deberá plasmar la solución al problema de optimización, el cual es elegir el objetivo a optimizar utilizando el método de Markowitz de rendimiento/riesgo, determinando la participación de cada acción en el portfolio y las restricciones al método.

FRONTERA EFICIENTE SEGÚN OBJETIVO

Para definir la frontera eficiente se puede utilizar la función *EfficientFrontier*, con esto obtenemos el conjunto de carteras que maximiza el rendimiento para determinado nivel de riesgo o minimiza el riesgo para determinado nivel de rendimiento. De la función antecedente podemos obtener distintas técnicas de optimización de cartera utilizadas para construir carteras de inversión con las características anteriormente citadas.

```
fe = EfficientFrontier(rendEsp1, matrizCov2, weight_bounds=(0, 1))
```

Los parámetros incluidos son la matriz de varianza y covarianza mediante la función *weigth_bounds* que corresponde al máximo y mínimo peso que puede tomar una acción en la cartera y los rendimientos esperados que se visualizan como vector.

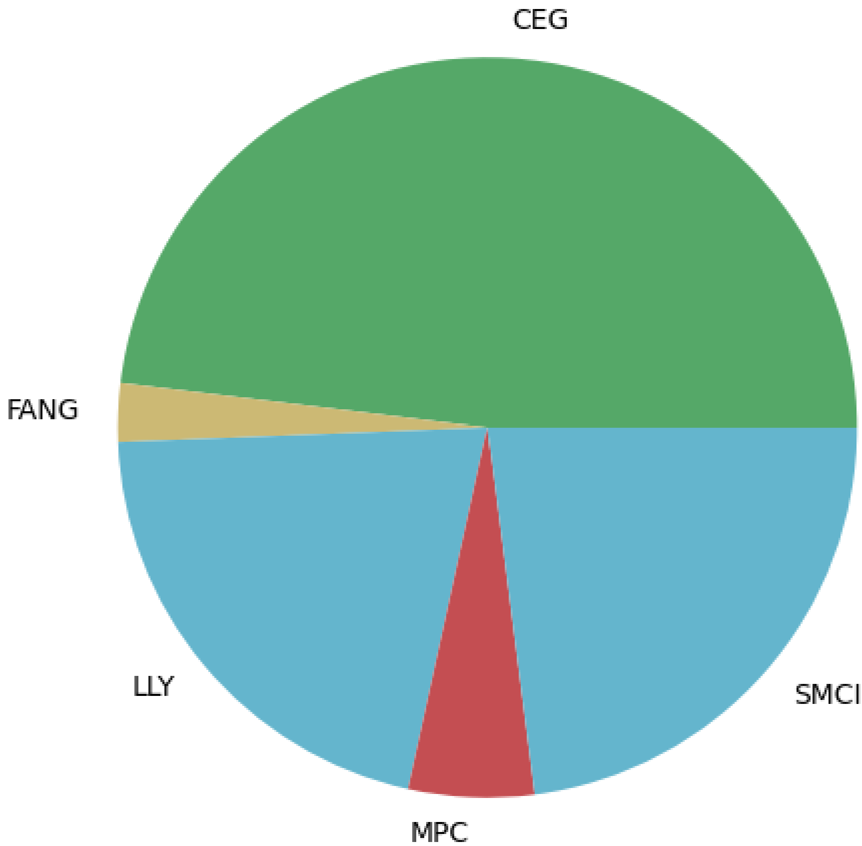
De las técnicas más comunes de optimización, correspondientes a los distintos objetivos a optimizar incluyen:

- PORTFOLIO DE MÁXIMO RENDIMIENTO "`efficient_risk()`": la función utiliza algoritmos de optimización para asignar de manera óptima los activos en una cartera de inversiones, maximizando el rendimiento esperado dado un nivel de riesgo, siguiendo el principio de *insaciabilidad* de Markowitz.

```
fe = EfficientFrontier(rendEsp1, matrizCov2)
fe.efficient_risk(target_volatility= 0.25)
pesos = fe.clean_weights()
pesos
```

```
OrderedDict([('CEG', 0.48094),
             ('FANG', 0.02529),
             ('LLY', 0.20937),
             ('MPC', 0.05471),
('SMCI', 0.22968)]
```

Gráfico N° 2 – Portfolio de máximo rendimiento



Fuente: Elaboración propia

Como se puede observar, la cartera resultante que maximiza el rendimiento para un determinado riesgo de 25%, estará compuesta por las siguientes acciones con sus respectivas participaciones:

'CEG' con un 48,09%

'FANG' con un 2,53%

'LLY' con un 20,93%

'MPC' con un 5,47%

'SMCI' con un 22,96%

Mediante la función *fe.portfolio_performance(verbose=True)* se puede calcular y mostrar el rendimiento de la cartera junto con otras métricas relevantes, como el rendimiento esperado, volatilidad y ratio de Sharpe.

```
Expected annual return: 81.4%
Annual volatility: 25.0%
Sharpe Ratio: 3.18
```

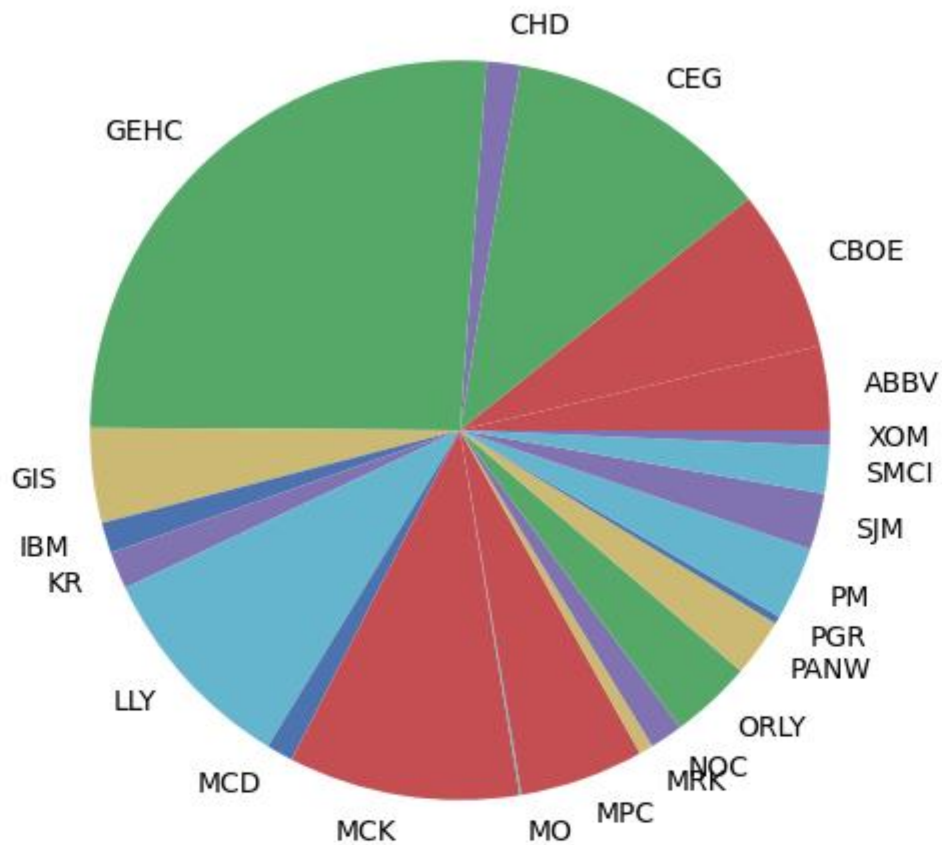
- PORTFOLIO DE MÍNIMO RIESGO "`efficient_return`": esta función asigna de manera óptima los activos en una cartera de inversiones minimizando el riesgo dado un nivel de rendimiento. Cumpliendo con el supuesto de Markowitz de *aversión al riesgo.*

```
fe = EfficientFrontier(rendEsp1, matrizCov2)
fe.efficient_return(target_return=0.35)
pesos = fe.clean_weights()
pesos
```

```
OrderedDict([('ABBV', 0.03666),
             ('CBOE', 0.07095),
             ('CEG', 0.11627),
             ('CHD', 0.01459),
             ('GEHC, 0.26044),
             ('GIS', 0.04144),
             ('IBM', 0.01345),
             ('KR', 0.01638),
             ('LLY', 0.09299),
             ('MCD', 0.0116),
             ('MCK', 0.10106),
             ('MO', 0.00081),
             ('MPC', 0.05425),
             ('MRK', 0.00612),
             ('NOC', 0.01436),
             ('ORLY', 0.0358),
             ('PANW', 0.02532),
             ('PGR', 0.00332),
             ('PM', 0.03207),
             ('SJM', 0.02468),
```

```
('SMCI', 0.02074),
('XOM', 0.00669),
```

Gráfico N° 3 – Portfolio de Mínimo Riesgo



Fuente: Elaboración propia

El resultado conseguido que minimiza el riesgo dado un rendimiento de 35%, es la cartera compuesta por las siguientes acciones y sus relativas participaciones:

‘ABBV’ con un 3,66%

‘CBOE’ con un 7,09%

‘CEG’ con un 11,62%

‘CHD’ con un 1,46%

‘GEHC’ con un 26,04%

‘GIS’ con un 4,14%

‘IBM’ con un 1,34%

‘KR’ con un 1,63%

‘LLY’ con un 9,3%

‘MCD’ con un 1,16%

‘MCK’ con un 10,10%

‘MO’ con un 0,08%

‘MPC’ con un 5,42%

‘MRK’ con un 0,61%

‘NOC’ con un 1,43%

‘ORLY’ con un 3,58%

‘PANW’ con un 2,53%

‘PGR’ con un 0,33%

‘PM’ con un 3,2%

‘SJM’ con un 2,46%

‘SMCI’ con un 2,07%

‘XOM’ con un 0,67%

Con respecto a las demás métricas relevantes, como el rendimiento esperado, volatilidad y ratio de Sharpe, obtenemos lo siguiente.

```
Expected annual return: 35.0%
Annual volatility: 11.5%
Sharpe Ratio: 2.87
```

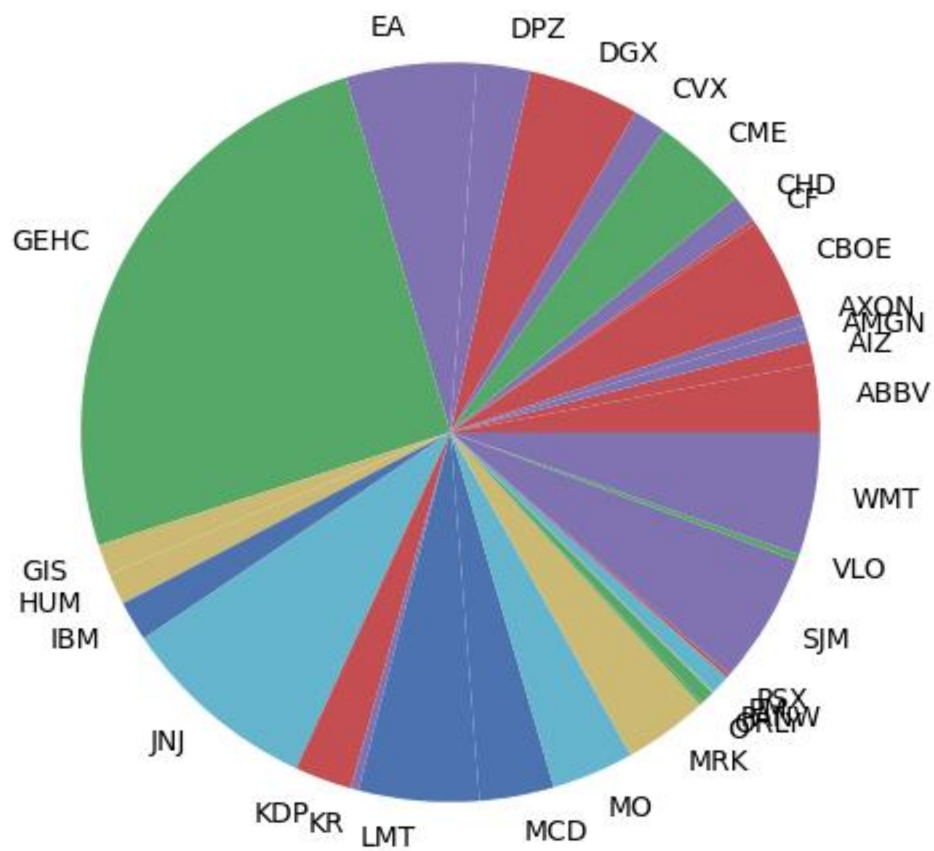
- CARTERA DE MÍNIMA VOLATILIDAD “`min_volatility ()`”: la misma es utilizada para calcular la cartera óptima que minimiza la volatilidad, dadas las rentabilidades esperadas de los activos y su matriz de covarianza. La idea es encontrar la cartera que tiene la menor medida de riesgo en términos de la dispersión de los rendimientos de los activos que la componen.

```
fe = EfficientFrontier(rendEsp1, matrizCov2)
varianza_minima = fe.min_volatility()
pesos = fe.clean_weights()
pesos
```

```
OrderedDict([('ABBV', 0.0294),
             ('AIZ', 0.01),
('AMGN', 0.00684),
('AXON', 0.00547),
('CBOE', 0.04386),
('CF', 0.00197),
('CHD', 0.01195),
('CME', 0.04248),
('CVX', 0.01447),
('DGX', 0.04863),
('DPZ', 0.02346),
('EA', 0.05705)
('GEHC', 0.25388),
('GIS', 0.01312),
('HUM', 0.01397),
('IBM', 0.01763),
('JNJ', 0.0873),
('KDP', 0.02495),
('KR', 0.00348),
('LMT', 0.05253),
('MCD', 0.03261),
('MO', 0.03598),
('MRK', 0.03701),
('O', 0.00139),
('ORLY', 0.00603),
('PANW', 0.0004),
('PM', 0.00866),
('PSX, 0.00159),
```

```
('SJM', 0.05729),
('VLO', 0.00269),
('WMT', 0.05377),
```

Gráfico N° 4 – Portfolio de Mínima volatilidad



Fuente: Elaboración propia

La cartera que minimiza la volatilidad estará conformada por los siguientes tickets con su adecuada proporción:

‘ABBV’ con un 2,93%

‘AIZ’ con un 1%

‘AMGN’ con un 0,68%

‘AXON’ con un 0,54%

‘CBOE’ con un 4,38%

‘CF’ con un 0,19%

‘CHD’ con un 1,19%

‘CME’ con un 4,24%

‘CVX’ con un 1,44%

‘DGX’ con un 4,86%

‘DPZ’ con un 2,34%

‘EA’ con un 5,7%

‘GEHC’ con un 25,39%

‘GIS’ con un 1,31%

‘HUM’ con un 1,4%

‘IBM’ con un 1,76%

‘JNJ’ con un 8,72%

‘KDP’ con un 2,49%

‘KR’ con un 0,34%

‘LMT’ con un 5,25%

‘MCD’ con un 3,26%

‘MO’ con un 3,59%

‘MRK’ con un 3,7%

‘O’ con un 0,13%

‘ORLY’ con un 0,6%

‘PANW’ con un 0,04%

‘PM’ con un 0,86%

‘PSX’ con un 0,15%

‘SJM’ con un 5,73%

‘VLO’ con un 0,27%

‘WMT’ con un 5,37%

El rendimiento esperado, volatilidad y ratio de Sharpe, para la cartera según el objetivo propuesto será:

```
Expected annual return: 10.4%
Annual volatility: 9.6%
Sharpe Ratio: 0.88
```

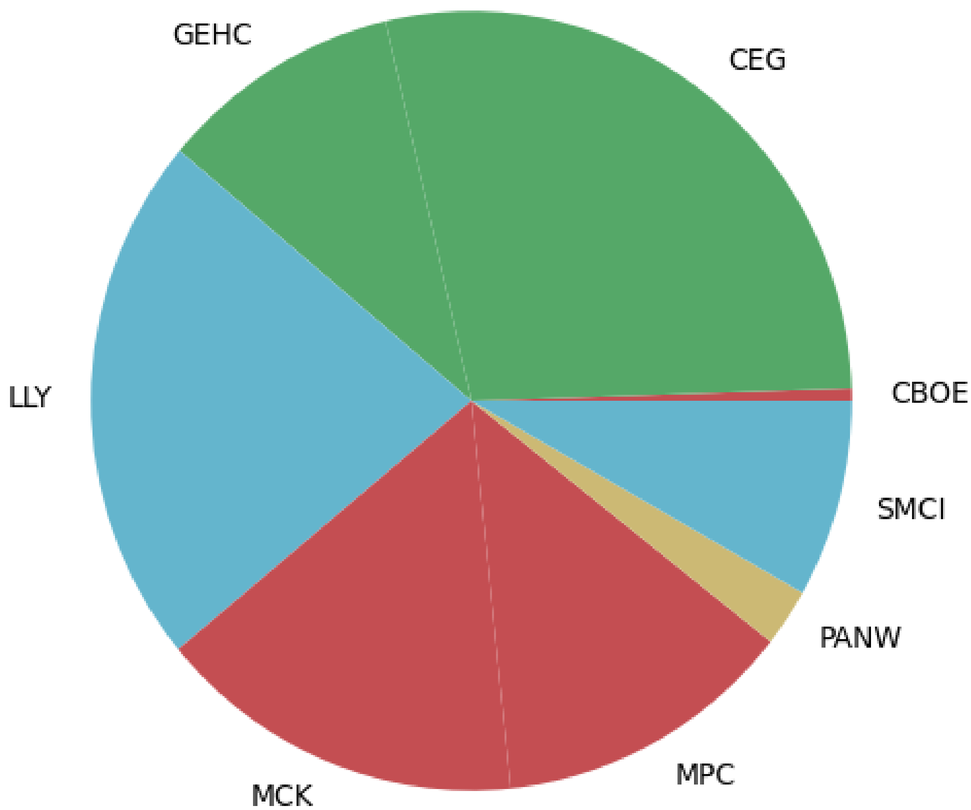
- CARTERA DE MÁXIMO RATIO DE SHARPE “`max_sharpe`”: esta función busca maximizar el índice de Sharpe de una cartera de inversiones. El índice de Sharpe es una medida de la rentabilidad ajustada al riesgo, que indica cuánto rendimiento adicional se está obteniendo por cada unidad adicional de riesgo asumido en una cartera de inversiones. Este portfolio representa la cartera de inversión más eficiente u óptima en términos de equilibrio entre riesgo y rendimiento según la teoría moderna de carteras. Es también nombrada como la cartera de tangencia, es decir donde la línea de mercado de capitales es tangente a la frontera eficiente.

```
fe = EfficientFrontier(rendEsp1, matrizCov2, weight_bounds=(0, 1))
max_sharpe = fe.max_sharpe(risk_free_rate=0.02)
pesos = fe.clean_weights()
pesos
```

En la función podemos percibir que se utiliza una tasa libre de riesgo de 2% anual.

```
OrderedDict([('CBOE', 0.00518),
             ('CEG', 0.2812),
             ('GEHC', 0.10272),
             ('LLY', 0.22089),
('MCK', 0.15638),
             ('MPC', 0.12758),
             ('PANW', 0.02407),
             ('SMCI', 0.08198),
```

Gráfico N° 5 – Portfolio de máximo Ratio de Sharpe



Fuente: Elaboración propia

Esta cartera estará compuesta por las siguientes acciones y sus participaciones:

‘CBOE’ con un 0,51%

‘CEG’ con un 28,12%

‘GEHC’ con un 10,27%

‘LLY’ con un 22,09%

‘MCK’ con un 15,64%

‘MPC’ con un 12,75%

‘PANW’ con un 2,4%

‘SMCI’ con un 8,19%

El rendimiento esperado, volatilidad y ratio de Sharpe, para la cartera según el objetivo de máximo Sharpe serán:

```
Expected annual return: 61.5%
Annual volatility: 17.3%
Sharpe Ratio: 3.45
```

- **OPTIMIZACIÓN SEGÚN RESTRICCIONES ESTABLECIDOS**

Según el objetivo del inversor, se pueden agregar restricciones al planteo para buscar la cartera que mejor se adapte al perfil. Por ejemplo, si se busca diversificar más, podemos incluir que cada activo no supere cierta cantidad de participación, en este caso hipotético sería del 1%.

```
fe = EfficientFrontier(rendEsp1, matrizCov2, weight_bounds=(0, 0.01))
fe.max_sharpe()
```

```
OrderedDict([('ABBV''ACGL''AFL''AJG''AMGN''ANET''AON''AVGO''AXON''AZO''BG'
'BLDR''BSX''CAH''CBOE''CDNS''CEG''CF''CHD’'CME’'COP’'COR’'COST’'DECK’'DVN’
'ED’'EG’'ELV’'EOG’'EQT’'ETN’'EXC’'FANG’'FE’'FICO’'GD’'GEHC’'GILD’'GIS’'GWW
’'HES’'HIG’'HSY’'HUBB’'HWM’'IBM’'IRM’'IT,'JBL’'JNJ’'KHC’'KR’'LLY’'LMT’'MCD
’'MCK’'MDLZ’'MMC’'MO’'MOH’'MPC’'MRK’'MRO’'MSFT’'MSI’'NOC’'NUE’'NVDA’'ORCL’
'ORLY’'OXY’'PANW’'PCAR’'PEP’'PG’'PGR’'PHM’'PM’'PSX’'PWR’'REGN’'RSG’'SLB’'S
MCI’'SNA’'SNPS’'SO’'STLD’'TRGP’'UNH’'VLO’'VRTX’'VST’'WM’'WMB’'WRB’'XOM’,0.
01),
             ('CMG', 0.0086723981892183),
             ('CVX', 0.0060378225621153),
             ('NTAP', 0.0092128431282311),
             ('TAP', 0.0022824575923044),
```

```
            ('WELL', 0.0037944785281305),

Expected annual return: 28.9%
Annual volatility: 15.2%
Sharpe Ratio: 1.77
```

Se pueden anexar un sinfín de restricciones a la función según el objetivo buscado por cada inversor. Para esto se puede utilizar el código *add_constraint ()*. Entre algunas de las limitaciones que pueden imponerse en las variables que se están optimizando podemos encontrar los límites de capacidad, restricciones de no negatividad, relaciones entre variables, restricciones de igualdad o desigualdad, restricciones de rango de valores, entre otras.

- **GRAFICAR LA FRONTERA EFICIENTE**

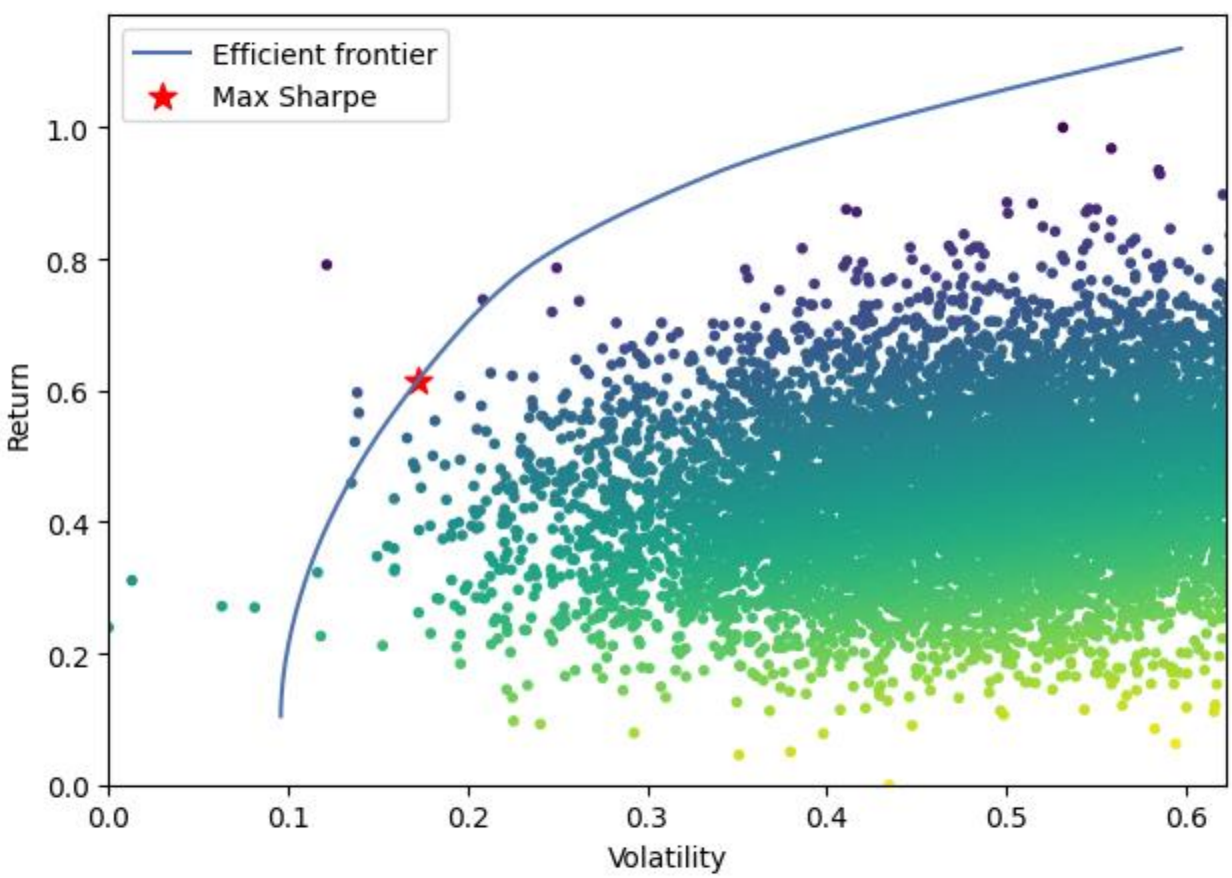
En el último paso, utilizando la simulación Montecarlo obtenemos las carteras aleatorias y con el módulo *Plotting* graficamos la frontera eficiente y la cartera de tangencia, esta cartera es la que maximiza el ratio de Sharpe

En esta gráfica utilizaremos la simulación de 10.000 carteras para obtener la frontera eficiente.

```
fe = EfficientFrontier(rendEsp1, matrizCov2)
fig, ax = plt.subplots()
ef_max_sharpe = fe.deepcopy()

# Gracfica la fronteraeficiente
plotting.plot_efficient_frontier(fe, ax=ax, show_assets=False)
ax.set_xlim(left=0)
ax.set_ylim(bottom=0)

# Genera carteras aleatorias
n_samples = 10000
w = np.random.dirichlet(np.ones(fe.n_assets), n_samples)
rets = w.dot(rendEsp1)
stds = np.sqrt(np.diag(w @ matrizCov2 @ w.T))
sharpes = rets / stds
#ax.scatter(stds, rets, marker=".", c=sharpes, cmap="viridis_r")
rets_normalized = (rets - min(rets)) / (max(rets) - min(rets))
```

```
stds_normalized = (stds - min(stds)) / (max(stds) - min(stds))
ax.scatter(stds_normalized, rets_normalized, marker=".", c=sharpes,
cmap="viridis_r")

# Encuentra la cartera de tangencia y la grafica
ef_max_sharpe.max_sharpe()
ret_tangent, std_tangent, _ = ef_max_sharpe.portfolio_performance()
ax.scatter(std_tangent, ret_tangent, marker="*", s=100, c="r",
label="Max Sharpe")

# Output
ax.set_title("Frontera eficiente con cartera de tangencia")
ax.legend()
plt.tight_layout()
plt.savefig("ef_scatter.png", dpi=200)
plt.show()
```

Figura N° 23 – Frontera eficiente con cartera de tangencia



Fuente: Elaboración propia

2.3 ANÁLISIS RENDIMIENTO DEL ÍNDICE S&P 500

En este apartado se analizará el comportamiento del rendimiento del índice. Esto se tomará como referencia para realizar la comparación con la cartera óptima y así obtener el resultado que busca el presente trabajo.

```
ÍNDICE = "^GSPC"
start = datetime.datetime(2020, 12, 1)
end = datetime.datetime(2023, 12, 1)
df = yf.download(ÍNDICE, start='2020-12-01', end='2023-12-01', 
actions='inline')
df
```

El módulo *datetime* nos proporciona una manera conveniente de trabajar con fechas y horas en programas. Tomamos como referencia el mismo período de tiempo con el que se calcularon los distintos parámetros de las acciones, que es del período 2020-12-01 al 2023-12-01.

Con la biblioteca de yahoo finance se descargan los datos financieros del índice a través de un data frame.

Figura N° 24 - Datos históricos del precio del índice S&P500

	Open	High	Low	Close	Adj Close	Volume	Dividends	Stock Splits
Date								
2020-12-01	3645.870117	3678.449951	3645.870117	3662.449951	3662.449951	5418480000	0.0	0.0
2020-12-02	3653.780029	3670.959961	3644.840088	3669.010010	3669.010010	5041250000	0.0	0.0
2020-12-03	3668.280029	3682.729980	3657.169922	3666.719971	3666.719971	5065340000	0.0	0.0
2020-12-04	3670.939941	3699.199951	3670.939941	3699.120117	3699.120117	5099620000	0.0	0.0
2020-12-07	3694.729980	3697.409912	3678.879883	3691.959961	3691.959961	4804500000	0.0	0.0
...	...	...	...	...	...	...	...	...
2023-11-24	4555.839844	4560.310059	4552.799805	4559.339844	4559.339844	1639500000	0.0	0.0
2023-11-27	4554.859863	4560.520020	4546.319824	4550.430176	4550.430176	3403990000	0.0	0.0
2023-11-28	4545.549805	4568.140137	4540.509766	4554.890137	4554.890137	3586240000	0.0	0.0
2023-11-29	4571.839844	4587.640137	4547.149902	4550.580078	4550.580078	4418760000	0.0	0.0
2023-11-30	4554.870117	4569.890137	4537.240234	4567.799805	4567.799805	5399300000	0.0	0.0

755 rows × 8 columns

Fuente: elaboración propia

Posteriormente se calcula el rendimiento anual total del índice, teniendo en cuenta el precio inicial, el precio final y los dividendos recibidos durante un período de tiempo determinado.

```
precio_inicial = df['AdjClose'].iloc[0]
precio_final = df['AdjClose'].iloc[-1]
dividendos = df['Dividends'].sum()
retorno_total = (precio_final + dividendos) / precio_inicial-1
dias_transcurridos = (end - start).days
años_transcurridos = dias_transcurridos / 365.25
rendimiento_anual_equivalente = ((1 + retorno_total) ** (1 /
años_transcurridos) - 1) * 100
print(f"El rendimiento anual del índice {ÍNDICE} desde
{start.strftime('%Y-%m-%d')} hasta {end.strftime('%Y-%m-%d')} es del
{rendimiento_anual_equivalente:.2f}%")
```

Como resultado durante el periodo de tiempo elegido se consigue un rendimiento anual del índice *^GSPC* de **7,65%.**

2.4 COMPARACIÓN DE CARTERA ÓPTIMA CON ÍNDICE DE REFERENCIA

Una vez obtenido el rendimiento del índice y las carteras óptimas según el objetivo del inversor, para un periodo determinado de tiempo. Ahora iniciaremos con la comparación final para poder responder a la pregunta inicial de este trabajo, el cual es concluir si, ¿existe una cartera con más rendimiento que el índice?

Para esta comparación, se empleará la cartera de mínima volatilidad, por ser la que menor rendimiento tiene entre los demás portfolios, ya que su finalidad es conseguir una cartera con mínima volatilidad y por consiguiente a menor volatilidad, menor rendimiento.

Primeramente se descarga los precios de cierre ajustados para las acciones que componen la cartera de mínima volatilidad y para el índice (benchmark) desde el 1 de diciembre de 2020 hasta el 1 de diciembre de 2023 y los almacena en la variable *cartera_data*.

```
cartera_data = yf.download(cartera_symbols, start="2020-12-01",
end="2023-12-01")["Adj Close"]
indice_data = yf.download(indice_symbol, start="2020-12-01",
end="2023-12-01")["Adj Close"]
```

Posteriormente se calcular el rendimiento acumulado de la cartera y del benchmark.

```
cartera_cumulative_return = (1 + cartera_returns).cumprod() - 1
indice_cumulative_return = (1 + indice_returns).cumprod() - 1
```

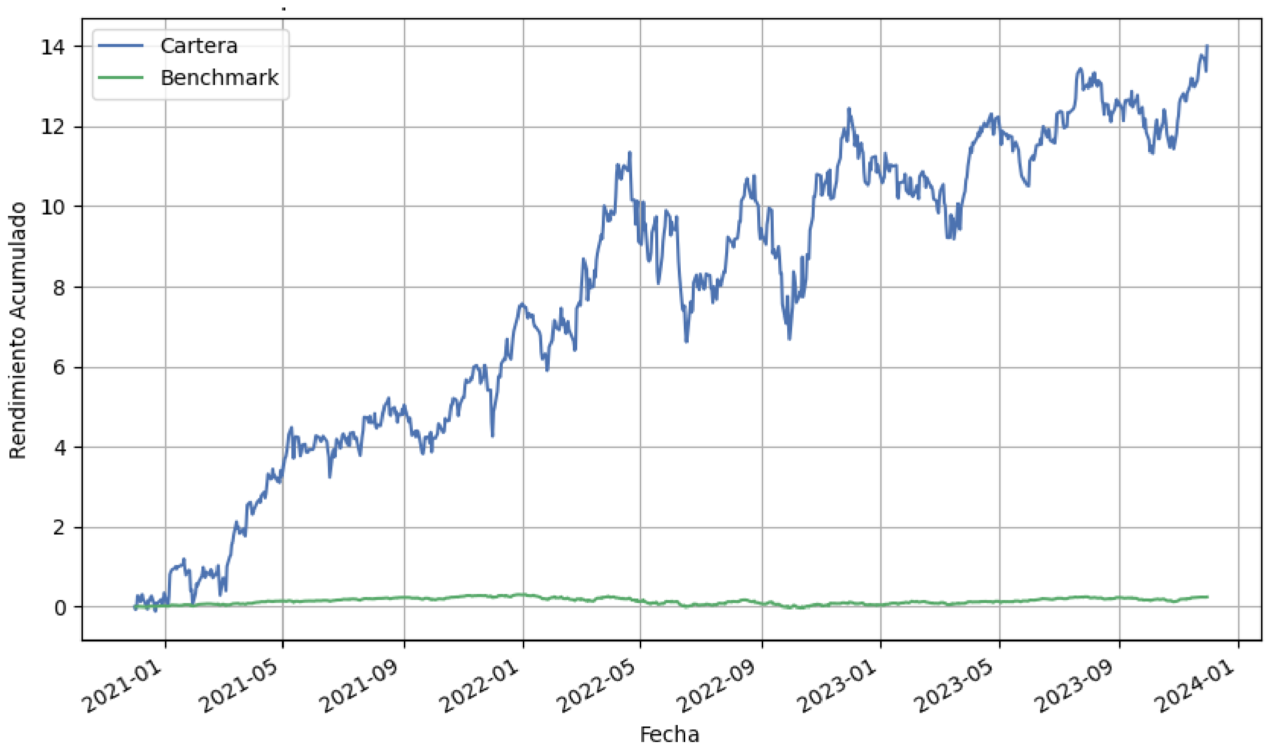
Donde *cartera_returns* es una serie que contiene los retornos diarios de la cartera e índice respectivamente. La suma (+1) se realiza para convertir los retornos diarios en factores de crecimiento acumulativo. Y con la función *cumprod* se logra acumular los retornos a lo largo del tiempo. Finalmente, se resta 1 para ajustar el retorno acumulado a una base de 0, por ejemplo, si el retorno acumulado es 1.10, significa que la cartera ha crecido un 10% en total.

Como último se crea un Data Frame que compara el rendimiento acumulado de la cartera de activos con el rendimiento acumulado del índice de mercado obtenido en el paso anterior.

Utilizamos la biblioteca Matplotlib para trazar el gráfico de dicha comparación.

```
comparacion = pd.DataFrame({
    "Cartera": cartera_cumulative_return.sum(axis=1),
     "Benchmark": indice_cumulative_return
    })

comparacion.plot(figsize=(10, 6))
plt.title('Comparación de Rendimiento: Cartera vs Índice de
Referencia S&P500')
plt.xlabel('Fecha')
plt.ylabel('Rendimiento Acumulado')
plt.grid(True)
plt.legend()
plt.show()
```

Figura N° 25 - Datos Comparación de rendimiento: Cartera de mínima volatilidad vs índice de referencia S&P500



Fuente: Elaboración propia

CONCLUSIONES

En este trabajo se ha analizado el Modelo de Markowitz con el fin de lograr responder a la pregunta si es posible obtener una cartera eficiente de acciones que pueda tener un mejor rendimiento que su índice de referencia.

En el proceso de análisis llevado a cabo se examinaron los datos a tener en cuenta para elegir las carteras eficientes según distintos objetivos, como la cartera de máximo rendimiento para un nivel de riesgo, cartera de mínimo riesgo para un nivel de rendimiento, la cartera de mínima volatilidad, la cartera de tangencia o de máximo Sharpe y una cartera con restricciones impuestas.

La longitud del período de tiempo en el que se tiene en cuenta los distintos valores de las acciones queda definida por el inversor, en el presente trabajo se tomó como período de comparación desde 2020-12-01 al 2023-12-01, pero pueden tomarse distintos parámetros de tiempo que conllevarán a modificar considerablemente los resultados, como por ejemplo si tomáramos el período de 2020-03, el rendimiento obtenido sería muy distinto, ya que el factor de la pandemia por Covid repercutió negativamente en la macroeconomía y por consecuencia resultó afectado el valor de las acciones, con una caída precipitosa en los valores de los activos de todo el mundo.

Por otra parte, hacemos uso de la tecnología de información a través de la herramienta Google Colaboraty, el cual nos permite ejecutar el lenguaje de programación Python para implementar de manera sencilla el modelo de optimización de Markowitz. Esta herramienta es de acceso gratuito para cualquier persona y brinda una variedad de bibliotecas con sus respectivos códigos que son necesarias para lograr los objetivos propuestos.

De acuerdo a los resultados obtenidos mediante las distintas simulaciones realizadas para cada objetivo de optimización de cartera, se arriba al siguiente cuadro de resumen:

Tabla N° 1 - Cuadro objetivos de Optimización

	MÁXIMO RENDIMIENTO (RIESGO 25%)	MÍNIMO RIESGO (RENDIMIENTO 35%)	MÁXIMO SHARPE	MÍNIMA VOLATILIDAD
RENDIMIENTO	78,40%	35%	59,10%	9,90%
RIESGO	25%	11,50%	17%	9,40%
RATIO SHARPE	3,06	2,86	3,36	0,84

Fuente: Elaboración propia

En cuanto al rendimiento anual del índice S&P 500 (^GSPC) desde 2020-12-01 hasta 2023-12-01 es del 7.65%.

Por lo que se concluye según el cuadro precedente que durante el mismo período de tiempo, la cartera óptima obtenida según los distintos objetivos del inversor, poseen un mayor rendimiento en comparación a la cartera que ofrece el índice de referencia S&P500.

Esto se debe a que el índice al estar constituida por una mayor diversificación de activos en su cartera, se pueden encontrar empresas con rendimiento negativo. A diferencia de la cartera óptima que solo toma las acciones con rendimiento positivos. Como sabemos las acciones no ponderan igual en el índice, es decir aquellas con una mayor capitalización bursátil tienen una mayor influencia por lo que por ejemplo un rendimiento negativo en la acción Apple que tiene una gran ponderación en el índice 5,9%, lo afectará más que la cartera de Sharpe.

REFERENCIAS

BIBLIOGRAFÍA

- ALEXANDER, SHARPE y BAYLE: “Fundamentos de Inversiones”, 3ra edición, México: Pearson.
- BARTOLOMEO, Alejandro y MACHIN URBAY, Gustavo: “Estrategias en el armado de cartera de bonos en el mercado argentino”. Anales de las XXXVIII Jornadas Nacionales de Profesores Universitarios de Matemática Financiera.
- BARTOLOMEO, Alejandro, MACHIN URBAY, Gustavo y SEGURA, María Verónica: “Formación de cartera de bonos en el mercado argentino”. Jornadas de Ciencias Eonómicas 2018
- BARTOLOMEO, Alejandro, MACHÍN URBAY, Gustavo (2022). “Algoritmos de clasificación para medir el desempeño académico de los alumnos universitarios: auto aprendizaje automático con PyCaret”. Anales de las 43° Jornadas Nacionales de Profesores Universitarios de Matemática Financiera. Asociación de Profesores Universitarios de Matemática Financiera (APUMF).
- DUMRAUF, GUILLERMO L.: “Finanzas Corporativas” un enfoque latinoamericano, 2da edición.
- ELBAUM, Marcelo: “Administación de Cartera de Inversión”, 1ra edición
- FREITAS, Wilson. A curated list of insanely awesome libraries, packages and resources for Quants (Quantitative Finance) 2021. https://github.com/wilsonfreitas/awesome-quant#data-sources
- HILPISCH, Yves: “Phyton for Finance”. Analyze Big Financial Data. O’ Reilly
- MARTIN, Robert Andrew. PyPortfolioOpt: optimización de cartera en Python. The Journal of Open Source Software. (2021). https://doi.org/10.21105/joss.03066

- MACHÍN URBAY, Gustavo. (2022) PortfolioOptimization_PyPortfolioOpt. https://github.com/gustavomachin/PortfolioOptimization_PyPortfolioOpt

- VAN HORNE, James: “Administración Financiera”. 10ma edición.

- WERNER, Charlotte (2021). Introduction to Portfolio analysis in python. In DataCamp. https://www.datacamp.com/courses/introduction-to-portfolio-analysis-in-python

OTROS MATERIALES

- Apuntes de la Cátedra Inversiones Financieras
- Apuntes de la Cátedra Matemática Financiera

PÁGINA WEB CONSULTADAS

- Rankia. (03 de Julio de 2024). ¿Cómo funciona la Bolsa de valores de Nueva York (NYSE)?. https://www.rankia.cl/blog/analisis-ipsa/3557639-como-funciona-bolsa-valores-nueva-york-nyse

- El índice **Standard & Poor's 500** (Standard & Poor's 500 Index). En Wikipedia. https://es.wikipedia.org/wiki/S%26P_500

- Crocker, B. (5 de Abril de 2019). Exploración y análisis del SP500con R, Parte 1. Medium. https://towardsdatascience.com/exploring-the-sp500-with-r-part-1-scraping-data-acquisition-and-functional-programming-56c9498f38e8

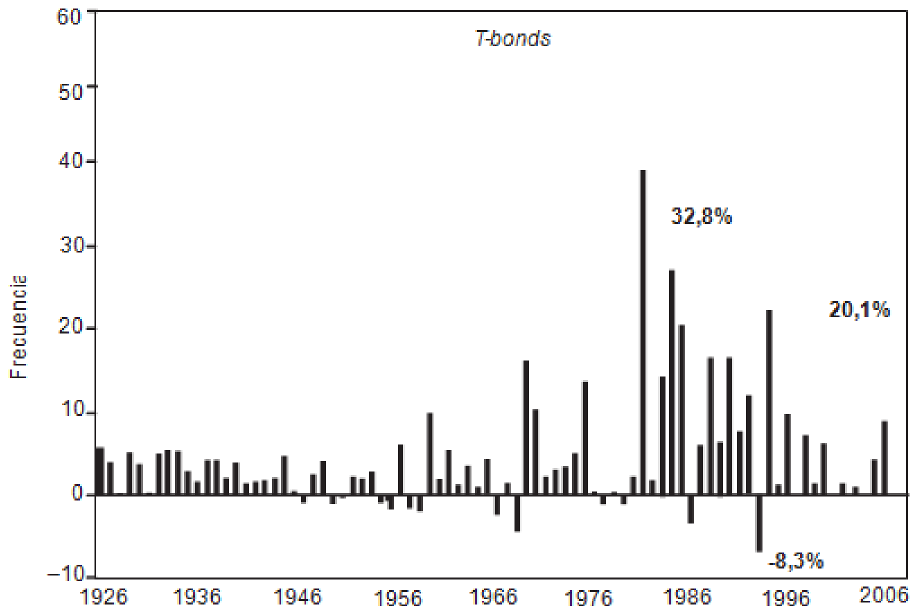
- Núñez, J., González, J. , Rueda, G., Díaz, S. (2017). Diversificación internacional de portafolio en los mercados accionarios de Argentina, Brasil, Chile, Colombia, México y Perú. [Archivo PDF]. https://www.researchgate.net/publication/320623350_DIVERSIFICACION_INTERNACIONAL_DE_PORTAFOLIO_EN_LOS_MERCADOS_ACCIONARIOS_DE_ARGENTINA_BRASIL_CHILE_COLOMBIA_MEXICO_Y_PERU

ANEXOS

ANEXOS 1-A

RENDIMIENTOS ANUALES DE LOS T-BOND (EEUU)

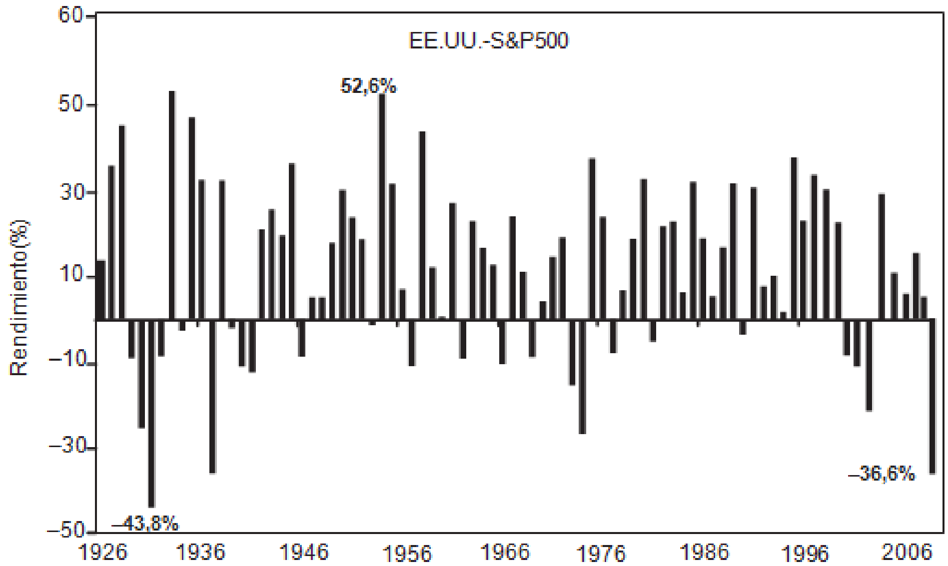
Fuente: Libro “Finanzas corporativas, un enfoque latinoamericano”. G.L Dumrauf. Pág. 195



ANEXOS 2 -B

RENDIMIENTOS ANUALES DE ACCIONES EN EEUU

Fuente: Libro “Finanzas corporativas, un enfoque latinoamericano”. G.L Dumrauf. Pág. 195


EE.UU.-S&P500
52,6%
–43,8%
–36,6%
Rendimiento (%)
60
50
30
10
–10
–30
–50
1926
1936
1946
1956
1966
1976
1986
1996
2006

DECLARACIÓN JURADA RESOLUCIÓN 212/99 CD

El autor de este trabajo declara que fue elaborado sin utilizar ningún otro material que no haya dado a conocer en las referencias que nunca fue presentado para su evaluación en carreras universitarias y que no transgrede o afecta los derechos de terceros.

Mendoza, 30 DE JULIO DE 2024

DÉBORA NATALÍ SEVERICHE
Firma y aclaración

26384
Número de registro

34748675
DNI