



Optimización de un portafolio con Python

Portfolio optimization with Python

Martínez-Sánchez, JF.; Cruz-García, S.; López-Castillo, J.

 **JF. Martínez-Sánchez**
jmartinez@uaeh.edu.mx
Universidad Autónoma del Estado de Hidalgo, México

 **S. Cruz-García**
s.c.garcia.fr@gmail.com
Universidad Autónoma del Estado de Hidalgo, México

 **J. López-Castillo**
july.pinky.yopii@gmail.com
Universidad Autónoma del Estado de Hidalgo, México

Pädi Boletín Científico de Ciencias Básicas e Ingenierías del ICBI

Universidad Autónoma del Estado de Hidalgo, México
ISSN-e: 2007-6363
Periodicidad: Semestral
vol. 9, núm. 17, 132-135, 2021
sitioweb@uaeh.edu.mx

Recepción: 02 Diciembre 2020
Aprobación: 12 Mayo 2021
Publicación: 05 Julio 2021

URL: <http://portal.amelica.org/ameli/jatsRepo/595/5952727024/index.html>

DOI: <https://doi.org/10.29057/icbi.v9i17.6807>

Autor de correspondencia: jmartinez@uaeh.edu.mx

Resumen: Motivación: Un inversor racional tiene como objetivo maximizar el rendimiento del portafolio y minimizar su riesgo. **Metodología:** Se utiliza la teoría de optimización de portafolios de Markowitz para obtener mediante el lenguaje de programación abierto Python el portafolio óptimo sobre la frontera eficiente. Este portafolio corresponde al punto donde el Sharpe ratio es el más alto. Para generar la frontera eficiente con los portafolios factibles se simulan múltiples portafolios de inversión mediante la implementación de un código en Python con Jupyter Notebook. **Resultados:** La composición óptima de un portafolio que contiene Bonos Gubernamentales Mexicanos, ETF's y acciones de Wal-Mart de México, está determinada por un Sharpe ratio de 0.8; el portafolio tiene un rendimiento esperado del 16% con una volatilidad del 15%.

Palabras clave: Teoría de Markowitz, portafolio óptimo, frontera eficiente, Python, Jupyter Notebook.

Abstract: Motivation: The goal of a rational investor is to maximize portfolio return and minimize portfolio risk. **Methodology:** Markowitz portfolio optimization theory is used to obtain by the Python programming language the optimal portfolio on the efficient frontier. This optimal portfolio corresponds to the point with the highest Sharpe ratio. To generate the efficient frontier with the feasible portfolios, multiple investment portfolios are simulated by implementing a code in Python with Jupyter Notebook. **Results:** The optimal composition of a portfolio constructed with Mexico Government Bonds, ETF's, and shares of Wal-Mart Inc's Mexico, is determined by a Sharpe ratio of 0.8; the portfolio has an expected return of 16% with a volatility of 15%.

Keywords: Markowitz theory, optimal portfolio, efficient frontier, Python, Jupyter Notebook.

1. INTRODUCCIÓN

De acuerdo con la teoría de Markowitz (Markowitz, 1952) (Dolci & Maçada, 2012), el inversor puede elegir distintas combinaciones con rendimiento esperado μ y varianza del rendimiento σ (riesgo) que dependen de

NOTAS DE AUTOR

jmartinez@uaeh.edu.mx

su elección de las proporciones $\#_1, \dots, \#_n$ de los activos $\# = 1, \dots, n$ en el portafolio. Entre todo el conjunto de posibles combinaciones rendimiento- riesgo $(\#, \#)$, el inversor debería querer seleccionar uno de aquellos portafolios que dan origen a combinaciones consideradas eficientes, —aquellas con una $\#$ mínima para una $\#$ dada, o una $\#$ máxima para un $\#$ dada—. Según Bernstein (Bernstein, 2005, p. 92), un portafolio eficiente es aquel que desde el punto de vista del inversor representa el menor riesgo a un cierto nivel de rendimiento esperado, o bien, el mayor rendimiento esperado a una cierta tasa de riesgo. Los portafolios eficientes de Markowitz caen en la frontera del conjunto de rendimientos esperados y riesgos $(\#, \#)$.

Esta *frontera eficiente* representa el conjunto de portafolios eficientes desde el menor riesgo hasta el mayor riesgo, o desde el menor rendimiento esperado hasta el mayor rendimiento esperado. La identificación de un portafolio eficiente corresponde a un problema de optimización: los valores de las variables $\#_i$ tienen que ser encontrados bajo las restricciones de que $0 \leq \#_i \leq 1$, para toda $i = 1, \dots, n$, que $\sum \#_i = 1$ y que el rendimiento del portafolio tenga un cierto valor esperado $\#$. El objetivo es encontrar los valores de las proporciones $\#_i$ de los activos que minimicen el riesgo $\#$ (Maringer, 2006, p. 38). Por otro lado, la optimización de un portafolio se refiere a elegir el portafolio sobre la frontera eficiente con el mayor Sharpe ratio (Carles, 2014).

El Sharpe ratio fue propuesto por Sharpe (Sharpe, 1963), se calcula restando la tasa de rendimiento libre de riesgo r_f del rendimiento esperado E del portafolio y se divide por la volatilidad σ (desviación estándar del rendimiento), es decir:

$$\text{Sharpe ratio} = \frac{E - r_f}{\sigma}.$$

El Sharpe ratio es una medida de rendimiento ajustada al riesgo que compara las inversiones con los activos libres de riesgo, la cual se usa frecuentemente para evaluar el comportamiento de un portafolio. En general, cuanto mayor es el valor del Sharpe ratio, más atractivo es el rendimiento ajustado al riesgo. Un Sharpe ratio negativo indica que el activo libre de riesgo se comportaría mejor que el portafolio con riesgo.

El portafolio eficiente corresponde al punto donde la Línea de Asignación de Capital es tangente a la frontera eficiente. La Línea de Asignación de Capital es la línea que representa gráficamente todas las combinaciones posibles de un activo libre de riesgo con un portafolio con riesgo dentro de la frontera eficiente. Su gráfico muestra el rendimiento que los inversores podrían ganar asumiendo un cierto nivel de riesgo con su inversión (CFA Institute, 2009, p. 161-165). Existen tantas Líneas de Asignación de Capital como portafolios con riesgo dentro de la frontera eficiente.

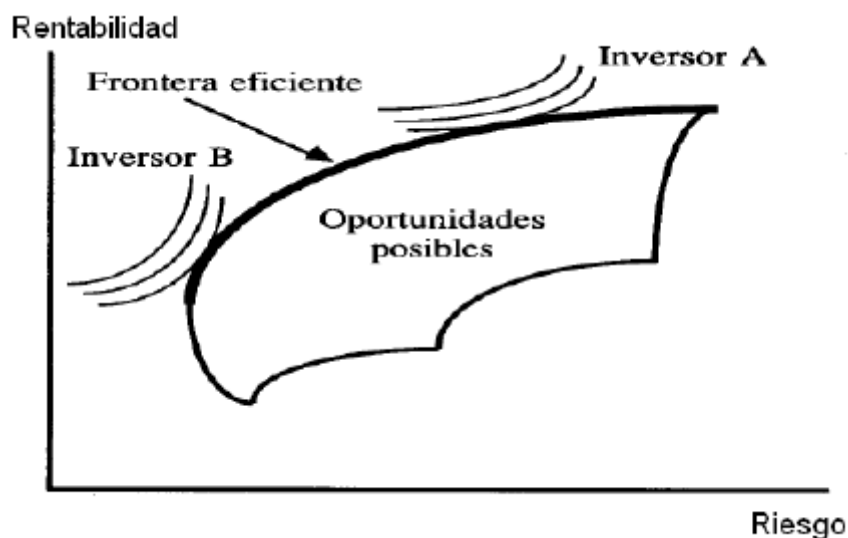


FIGURA 1
Frontera eficiente y asignación de capital
Universidad de Barcelona (tesis de grado)

La optimización de portafolios se presenta en diversos escenarios. Román et al. (Román, Pérez, & Estévez, 2012) simulan distintos portafolios de activos numismáticos de oro y metales nobles usando Toolkit financiera de MATLAB para obtener el portafolio óptimo, y así trazan la Línea del Mercado de Capitales para España, Estados Unidos, Gran Bretaña y Francia. Banda Ortiz et al. (Banda Ortiz, González García, & Gómez Hernández, 2014) aplican la teoría de Markowitz para crear portafolios óptimos de inversión de ahorros personalizados para trabajadores. Herrera (Herrera, 1999) aborda la teoría de Markowitz para mostrar una aplicación en la estimación del VaR (Valuación al Riesgo) para un portafolio de divisas. Dicho trabajo es una implementación teórica y no tanto práctica.

En el presente trabajo, se implementa la teoría de frontera eficiente de Markowitz para encontrar mediante simulaciones en Jupyter Notebook para Python la combinación óptima de algunos activos: Bonos Gubernamentales Mexicanos, ETF's y acciones de Wal-Mart de México. A diferencia de los trabajos previos, en este trabajo se utiliza un portafolio de activos, compuesto de bonos, acciones y los ETF's (Exchange-traded funds), los ETFs brindan acceso a mercados en todo el mundo, desde países específicos hasta una clase de activos, como bonos globales o incluso materias primas como oro. Invertir en un ETF facilita la inversión en mercados de difícil acceso, como los mercados emergentes. También, estos activos presentan una correlación negativa o cercana a 0, lo que permite la diversificación del portafolio. Por otro lado, destaca el uso de software libre y el empleo de la técnica de simulación MonteCarlo para generar la frontera eficiente. Otro de los resultados a destacar es su utilidad para el análisis de cualquier portafolio de activos financieros.

Jupyter Notebook es una aplicación web de código abierto (código abierto) del sitio Web <https://www.jupyter.org>, que permite integrar ecuaciones y bloques de texto narrativo (notebooks) junto con código en lenguaje Python (disponible en <https://www.python.org>) para realizar *data science*, cómputo científico y *machine learning*. Jupyter Notebook tiene incluidas numerosas librerías, por lo que no es necesario instalarlas manualmente, lo que hace el trabajo más eficiente.

El código de Python para el cálculo de la frontera eficiente de Markowitz es de acceso abierto (open access), fue desarrollado por Rojas (Rojas, 2018). En el presente documento presentamos las librerías de Python que ayudan a descargar, visualizar y hacer el análisis estadístico de los precios históricos de los activos. El propósito del trabajo es mostrar a los analistas financieros principiantes que es posible trabajar bases de datos y hacer análisis financiero con un simulador financiero basado en un streaming de código abierto.

Adicionalmente, este trabajo a diferencia del código de Rojas (Rojas, 2018), documenta en el mismo Notebook, el cual se adjunta como Anexo I, los procedimientos que se siguieron y los resultados de cada uno de ellos, lo cual hace una lectura fluida para los lectores de este trabajo.

2. METODOLOGÍA

Lo primero es elegir los activos que se desean incluir en el portafolio, tomando en cuenta los mercados en que cotizan y la diversificación de activos.

El portafolio por optimizar se conforma de los siguientes activos, los cuales fueron seleccionados debido a que tienen una baja correlación y eso como se mencionó al inicio de este artículo es bueno para la diversificación del portafolio. El período utilizado comprende las fechas del 6 de agosto de 2015 al 2 de octubre de 2019.

- Bonos Gubernamentales Mexicanos: iShares S&P/Valmer México CETETRAC
- ETF's: iShares Core S&P 500 ETF (IVV.MX)
- Acciones de Wal-Mart de México: Walmex

En el código en Python para el cálculo del portafolio óptimo, el manejo de los datos financieros se hace con las librerías *pandas* y *numpy*. La librería *numpy* tiene funciones estadísticas útiles para encontrar la media, desviación estándar, varianza, mínimo, máximo, percentil, etc. de los elementos dados en un array (arreglo). *pandas* es una librería que proporciona alto rendimiento, estructuras de datos y herramientas de análisis de datos; toma datos (como un archivo CSV o una base de datos SQL) y crea un *data frame* en Python con filas y columnas, que se parece mucho a una tabla en un software estadístico (como Excel o SPSS). Para graficar todos los posibles portafolios en forma *scatter* se utiliza la librería *matplotlib* de visualización 2D de datos. Para descargar los datos de precios históricos de los activos desde la página Web Yahoo Finance se usan las librerías *pandas_datareader* y *fix_yahoo_finance*. Se pueden extraer de una sola vez los precios de tantos activos como se quiera, requiriendo una lista de tickers (marcadores); los precios históricos son almacenados en el formato JSON.

Usando el código de Python hemos realizado la descarga de datos desde Yahoo Finance, indicando cuáles activos utilizaremos: iShares S&P/Valmer Mexico CETETRAC, iShares Core S&P 500 ETF (IVV) y Walmex; y las fechas de los datos. La salida del código es un arreglo con los datos de cierre ajustado de cada activo, como se muestra en la Tabla 1.

TABLA 1
Primeros cinco datos históricos por activo: Bonos Gubernamentales Mexicanos (CETETRAC), ETF's, y acciones de Wal-Mart de México (Walmart).

Date	CETETRAC	ETF's	Walmart
2015-08-06	102.980003	3426.429932	1181.924316
2015-08-10	102.900002	3372.97998	1153.038696
2015-08-11	102.900002	3418.169922	1147.869995
2015-08-12	102.989998	3423.649902	1163.469727
2015-08-13	103.000000	3421.189941	1169.493652

Elaboración propia

Si se requiere tener acceso a toda la base de información se debe modificar la instrucción `df.head(5)` en el código (Anexo 1), por `df`.

Para calcular el rendimiento esperado $\#$ del portafolio se requiere calcular el cambio porcentual diario de los precios de los activos. Con la instrucción `df.pct_change().dropna()` se obtienen los cambios porcentuales en el frame de Pandas. Lo anterior genera el rendimiento y la matriz de varianzas y covarianzas del portafolio con pesos iguales para cada activo $\#$.

Con el objetivo de calcular las proporciones $\#$ de cada activo $\#$, que den el Sharpe ratio más alto en las combinaciones de activos dentro de los portafolios eficientes, se restringen las $\#$ a ser no negativas, a alcanzar un valor máximo de 1, y a que sumen 1, esto es $0 \leq \# \leq 1$ para $\# = 1, \dots, \#$ y $\sum \# = 1$. El problema de optimización se resuelve con *SciPy optimize*, el cual proporciona funciones para minimizar (o maximizar) funciones objetivo que pueden estar sujetas a restricciones. El paquete *scipy.optimize* proporciona varios algoritmos de optimización de uso común. Lo anterior se ejecuta para encontrar la distribución óptima de pesos de cada activo que maximice el Sharp del portafolio.

Finalmente, con la distribución óptima se realizan simulaciones de portafolios para encontrar la relación riesgo- rendimiento que permita encontrar la frontera eficiente.

3. RESULTADOS

Con el código de Python realizamos la simulación de diez mil portafolios distintos para construir y graficar el conjunto de portafolios eficientes —los portafolios que proporcionan el menor riesgo para un cierto nivel de rendimiento esperado—, es decir la frontera eficiente de Markowitz. De acuerdo con las simulaciones, considerando una tasa libre de riesgo de 4%, el Sharpe ratio del portafolio óptimo es 0.8, y dicho portafolio tiene un rendimiento esperado del 16% con una volatilidad (desviación estándar del rendimiento) del 15%. La composición óptima del portafolio consiste en 0% en Bonos Gubernamentales Mexicanos (CETETRAC), 65% en ETF's y 35% en acciones de Wal-Mart de México (Walmart), para mayor detalle sobre los resultados anteriores revisar el Anexo 1.

En la Figura 2 graficamos el rendimiento esperado de todos los posibles portafolios versus la volatilidad, junto con la localización del portafolio óptimo, marcada con una X, sobre la frontera eficiente.

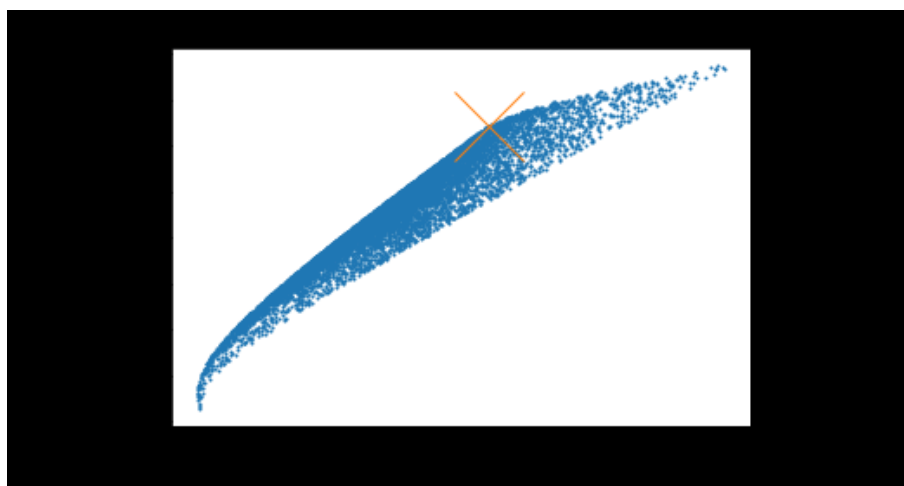


FIGURA 2:
Rendimiento esperado vs volatilidad del conjunto de todos los portafolios posibles.
El punto señalado sobre la frontera eficiente corresponde al portafolio óptimo.

Elaboración propia

4. CONCLUSIONES

Elegir un portafolio en el cual invertir capital monetario representa riesgos, por ello es necesario elegir con cautela el que mejor se acomode a las necesidades de los inversores.

Este trabajo presenta una forma práctica de construir portafolios óptimos de inversión. A pesar de ya existir implementaciones en software comerciales como MATLAB y DigiFundManager, se presenta una solución de aprendizaje e implementación en un software código abierto como Python. Actualmente el software de código abierto ha comenzado a dominar los negocios, ya que las licencias código abierto evitan las restricciones en el uso del software, enfatizando su accesibilidad y libertad.

AGRADECIMIENTOS

Este trabajo fue financiado por el Apoyo a la incorporación de NPTC PRODEP 2018 con la subvención UAEHPTC784.

REFERENCIAS

- Banda Ortiz, H., González García, L. M., & Gómez Hernández, D., (2014). Una aproximación de la teoría de portafolio a las SIEFORES en México. *Revista científica Pensamiento y Gestión* 36, 28–55. DOI: 10.14482/pege.36.5565
- Bernstein, P. L., 2005. *Capital ideas: The improbable origins of modern Wall Street*. John Wiley & Sons, Hoboken, New Jersey.
- Carles, P. G., (2014). Risk-Adjusted performance measurement. In: Editor (Ed.), *Investment risk management*. Oxford University Press, New York, NY, Ch. 19, pp. 365–386.
- CFA Institute, (2009). *2010 Level 2 Book 5: Derivatives and portfolio management*. Kaplan Schweser, United States of America.
- Dolci, P. C., & Maçada, A. C. G., (2012). Portfolio theory: The contribution of Markowitz's theory to information system area. In: Editor (Ed.), *Information systems theory: Explaining and predicting our digital society*. Vol. 1. Springer Science & Business Media, Heidelberg, New York, Dordrecht, London, Ch. 10, pp. 199–211. DOI: 10.1007/978-1-4419-6108-2
- Herrera, F. L., (1999). Aplicación del enfoque de Markowitz al cálculo del Valor en Riesgo (VaR) a un portafolio de divisas. *Revista Contaduría y Administración* 193(2), 53–60.
- Markowitz, H., (1952). Portfolio selection. *The Journal of Finance* 7(1), 77– 91. DOI: 10.1111/j.1540-6261.1952.tb01525.x
- Maringer, D. G., 2006. *Portfolio management with heuristic optimization*. Vol. 8. Springer Science & Business Media, Dordrecht, The Netherlands.
- Román, C. P., Pérez, J. L. C., & Estévez, P. G., (2012). Aplicación de la teoría de carteras con activos numismáticos y metales preciosos. *Cuadernos de Gestión* 12(1), 123–143. DOI: 10.5295/cdg.100201cp
- Rojas, O., (2018). Frontera eficiente en Python. LinkedIn. Recuperado de <https://www.linkedin.com/pulse/frontera-eficiente-en-python-oscar-rojas-cfa>
- Sharpe, W. F., (1963). A simplified model for portfolio analysis. *Management science* 9(2), 277–293. DOI: 10.1287/mnsc.9.2.277

NOTAS

8. Springer Science & Business Media, Dordrecht, The Netherlands. Román, C. P., Pérez, J. L. C., & Estévez, P. G., (2012). Aplicación de la teoría de carteras con activos numismáticos y metales preciosos. *Cuadernos de Gestión* 12(1), 123–

143. DOI: 10.5295/cdg.100201cp Rojas, O., (2018). Frontera eficiente en Python. LinkedIn. Recuperado de <https://www.linkedin.com/pulse/frontera-eficiente-en-python-oscar-rojas-cfa> Sharpe, W. F., (1963). A simplified model for portfolio analysis. *Management science* 9(2), 277–293. DOI: 10.1287/mnsc.9.2.277