

MÉTODOS NUMÉRICOS 2

Particle swarm optimization (PSO)

Por Ximena Loarca y Montserrat Gil



Índice:

Algoritmos evolutivos
Particle Swarm Optimization
Nuestro objetivo
Nuestro Dataset
Pyswarm
Nuestra implementación
Resultado y conclusiones
Recomendaciones

Algoritmos evolutivos

Son métodos de optimización basados en la teoría de la evolución de especies biológica.

Sus principales elementos son:

- Población
- Selección
- Reproducción
- Cruce y mutación
- Repetir para la siguiente generación

Algunos algoritmos evolutivos:

- **PARTICLE SWARM OPTIMIZATION**

Basado en el comportamiento de bandadas de animales.

- **ANT COLONY OPTIMIZATION**

Basado en la idea de recolección de las hormigas por comunicación utilizando feromonas.

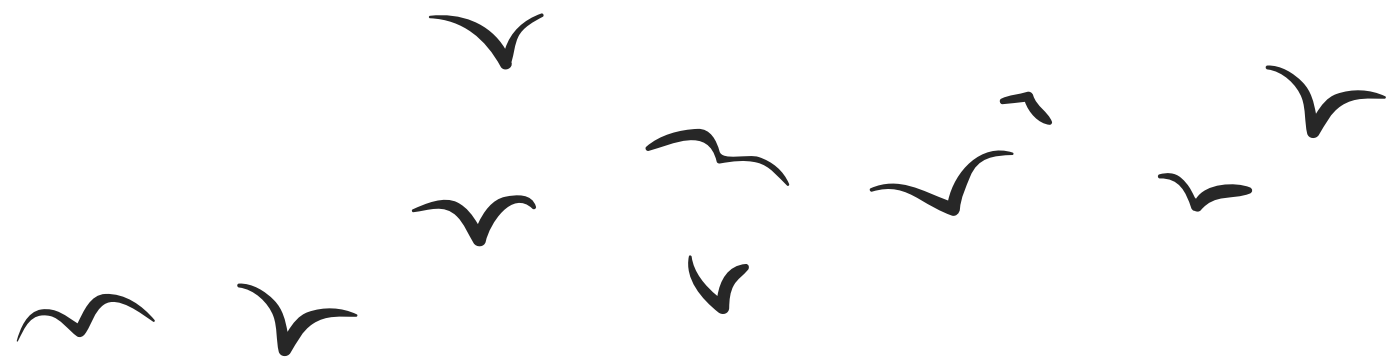
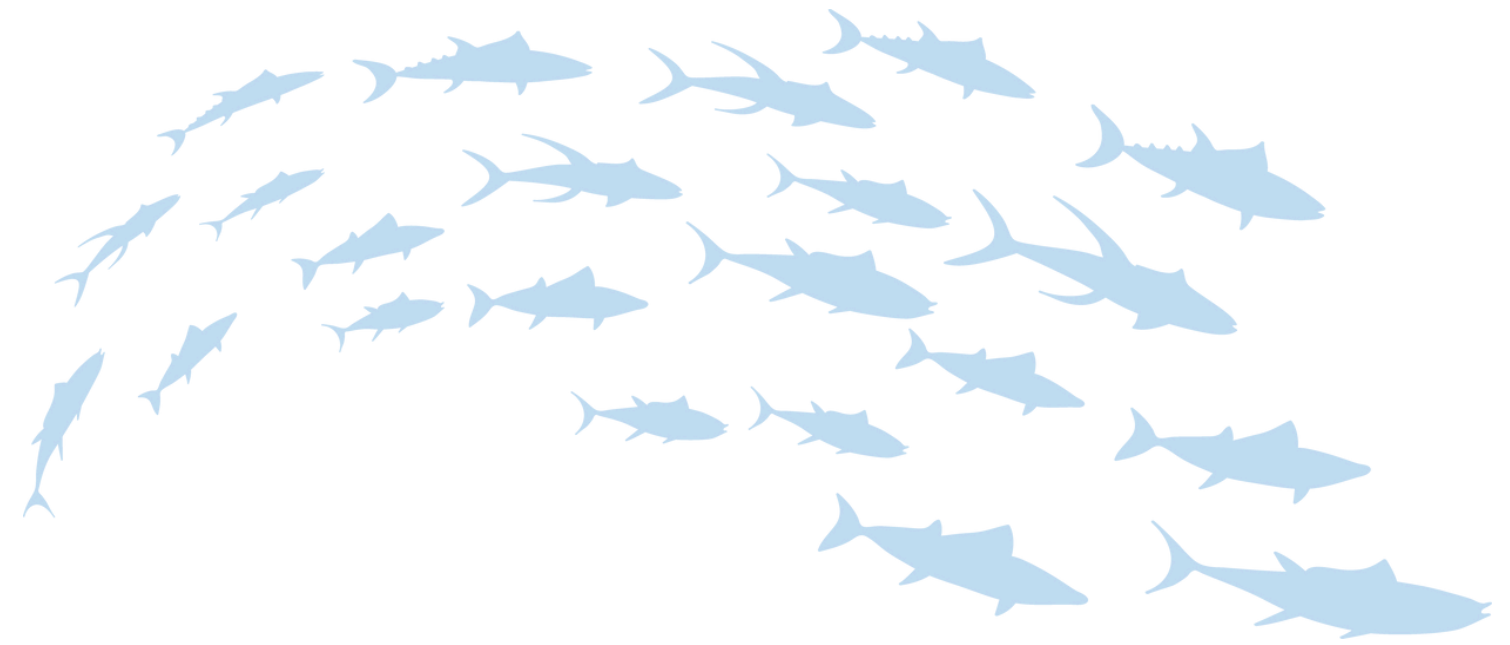
- **ALGORITMOS GENÉTICOS**

Basados en la evolución genética de las especies.

- **CUCKOO SEARCH**

Inspirado en el comportamiento de los pájaros cucos.

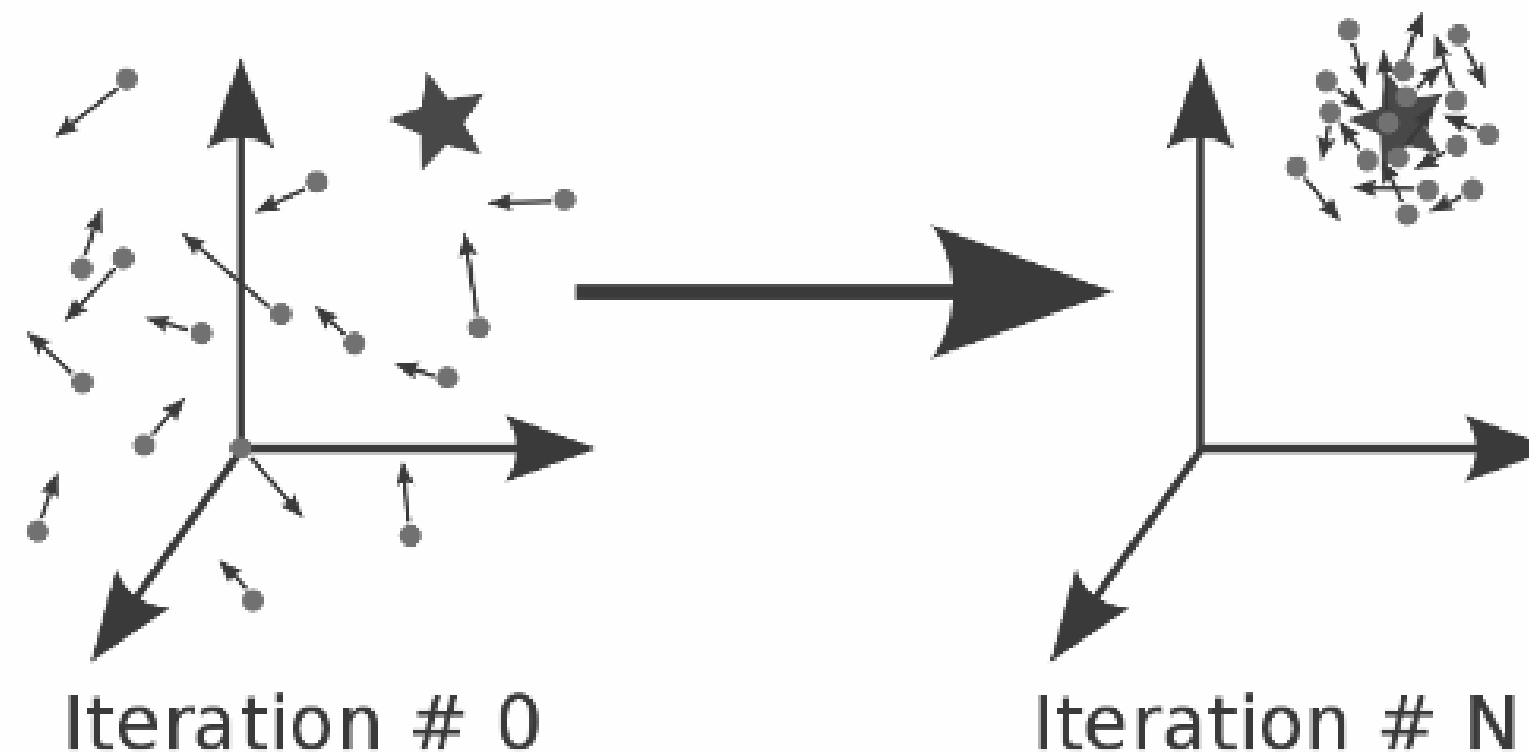
Optimización por enjambre de partículas (PSO)



Esta es inspirado en la
inteligencia colectiva
observada en la naturaleza,
en animales como bandadas
de aves o bancos de peces.

¿CÓMO FUNCIONA?

Resuelve un problema iniciando con una población de las posibles soluciones. Estas se llaman partículas y se mueven a través del espacio de búsqueda según una fórmula con la posición y la velocidad de cada partícula, en busca de la mejor solución.



Elementos del PSO

PARTÍCULAS EN VUELO

Las posibles soluciones del problema con una posición y memoria personal que guarda la mejor posición encontrada.

VELOCIDAD Y POSICIÓN

La velocidad de cada partícula se ajusta en cada iteración del algoritmo utilizando la mejor posición individual y global.

Elementos del PSO

FITNESS FUNCTION

Función que evalúa que tan buena es la solución según el objetivo de nuestro algoritmo.

PARÁMETROS

Factor de inercia que controla la tendencia de la partícula a continuar con su velocidad actual.

Coeficientes cognitivo y social que ajustan la influencia de la mejor posición individual y global.

FUNCIONAMIENTO BÁSICO

- **Iniciación:**

Cada partícula comienza en una posición aleatoria con una velocidad inicial.

- **Evaluación de aptitud:**

La función fitness evalúa la posición de cada partícula y actualiza los valores de mejor posición individual y global.

- **Actualización de velocidad:**

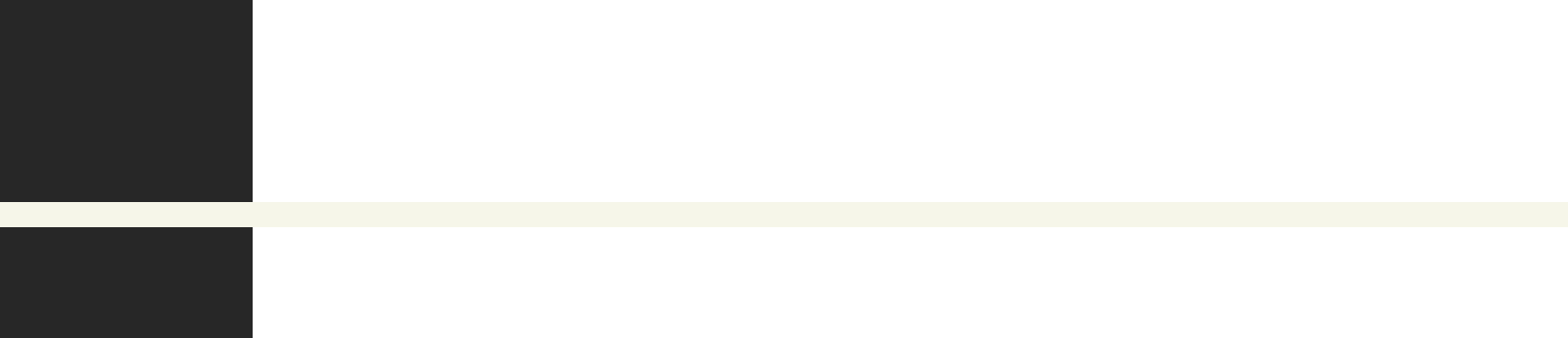
La velocidad de cada partícula se ajusta según tres factores:

- La velocidad actual
- El coeficiente cognitivo
- El coeficiente social

- **Actualización de la posición:**

Se modifica la posición de las partículas en función de los factores como su posición actual, su velocidad y la mejor posición individual y global.

- **Iteración hasta llegar a una solución**



Ventajas del PSO

Simple implementación

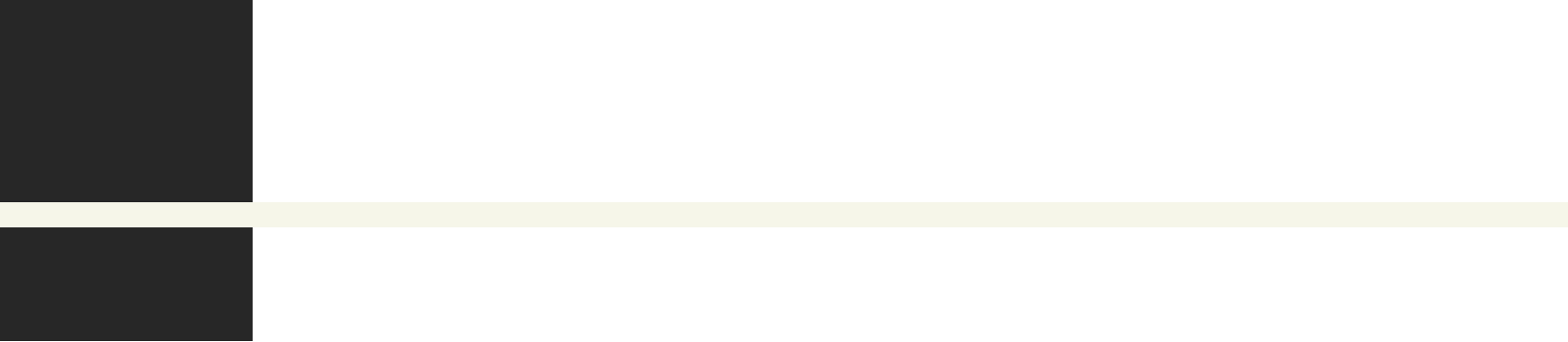
Convergencia rápida

Adaptabilidad

Independencia de derivadas

Optimización multiobjetivo





Limitaciones o desventajas del PSO

**Propenso a caer en óptimos
locales**

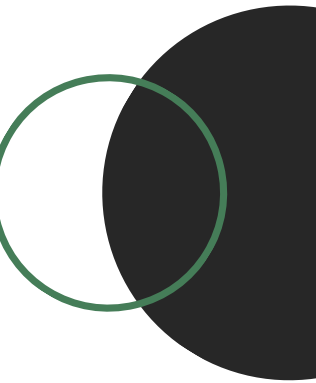
**Dependencia de los
parámetros**

Escalabilidad limitada

**Dificultad en problemas
discretos**



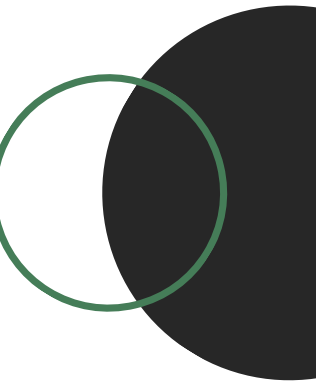
ECUACIONES DE MOVIMIENTO



ECUACIÓN DE ACTUALIZACIÓN DE LA VELOCIDAD DE UNA PARTÍCULA

$$v_i^{t+1} = w \cdot v_i^t + c_1 \cdot r_1 \cdot (p_i^{\text{best}} - x_i^t) + c_2 \cdot r_2 \cdot (g^{\text{best}} - x_i^t)$$

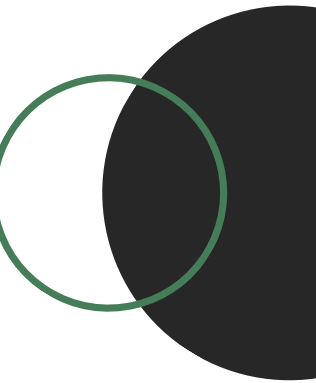
- v_i^{t+1} es la velocidad de la partícula i en el instante $t + 1$.
- w es el factor de inercia que controla la tendencia de la partícula a mantener su velocidad.
- c_1 y c_2 son los coeficientes cognitivo y social, respectivamente.
- r_1 y r_2 son valores aleatorios entre 0 y 1.
- p_i^{best} es la mejor posición personal de la partícula i .
- g^{best} es la mejor posición global encontrada por el enjambre.
- x_i^t es la posición actual de la partícula i en el instante t .



ECUACIÓN DE ACTUALIZACIÓN DE LA POSICIÓN DE UNA PARTÍCULA

$$x_i^{t+1} = x_i^t + v_i^{t+1}$$

- x_i^{t+1} es la nueva posición de la partícula i en el instante $t + 1$.
- v_i^{t+1} es la velocidad actualizada de la partícula i .





Parámetros importantes

**NÚMERO DE
PARTÍCULAS**

**FACTOR DE
INERCIA**

**COEFICIENTE
COGNITIVO**

**COEFICIENTE
SOCIAL**

**NÚMERO DE
ITERACIONES**

Nuestro objetivo

Optimizar la composición de un
portafolio de inversión utilizando el
algoritmo PSO para maximizar el
retorno esperado mientras se minimiza
el riesgo.

Nuestros datos

Dataset de 5 años de información de acciones del Stock market.

Contaba con 619040 entradas y 505 acciones diferentes.

date	open	high	low	close	volume	Name
Fecha del registro.	Precio de apertura de la acción.	Precio máximo alcanzado durante el día.	Precio mínimo alcanzado durante el día.	Precio de cierre de la acción.	Número de acciones negociadas ese día.	Nombre de la acción.

Nuestros datos

Se limpió el dataset eliminando nulos y duplicados.

Se seleccionaron 50 acciones.

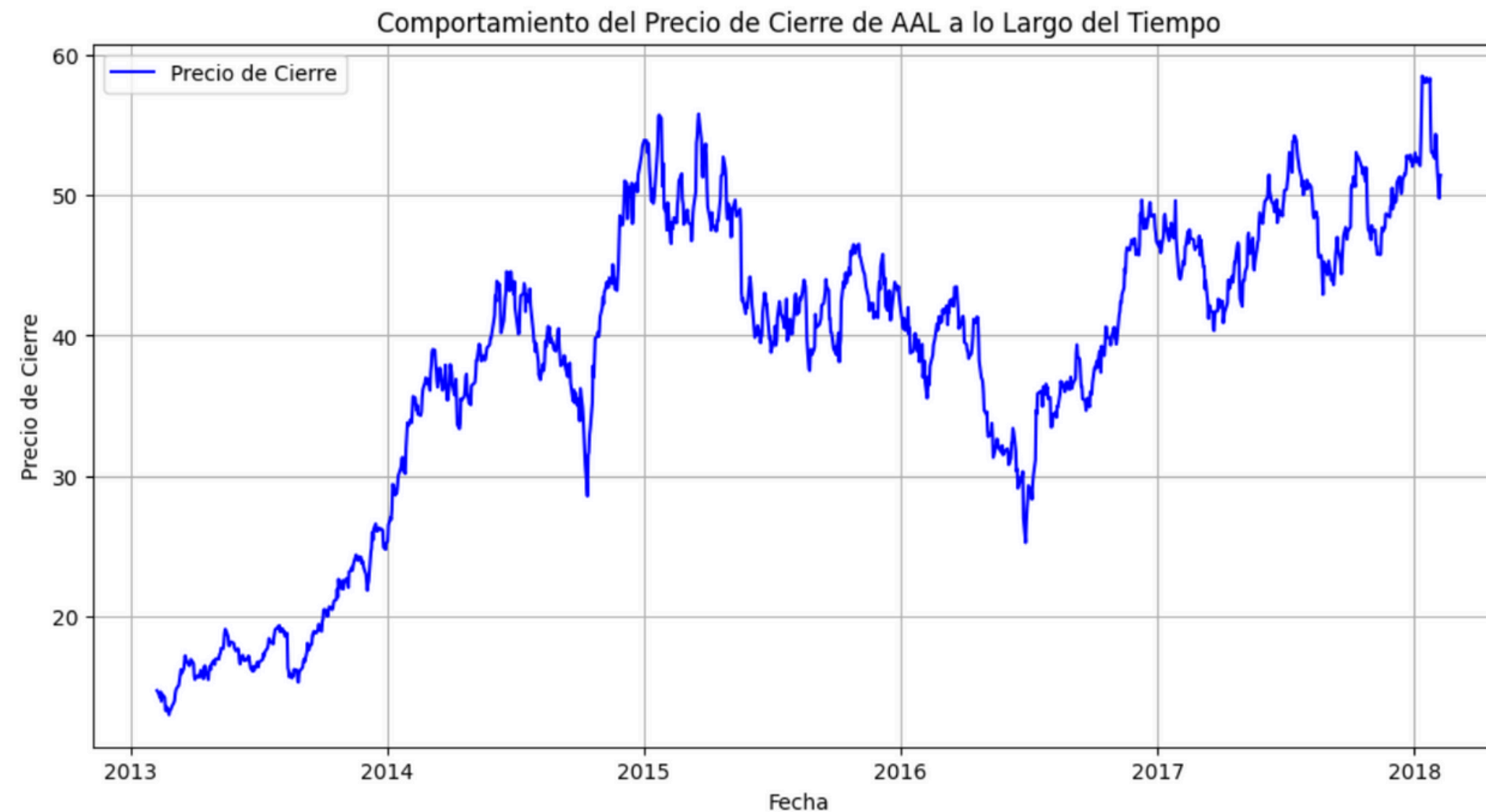
Tamaño del dataset: 61539 filas, 8 columnas

Rango de fechas: 08/02/2013 - 07/02/2018

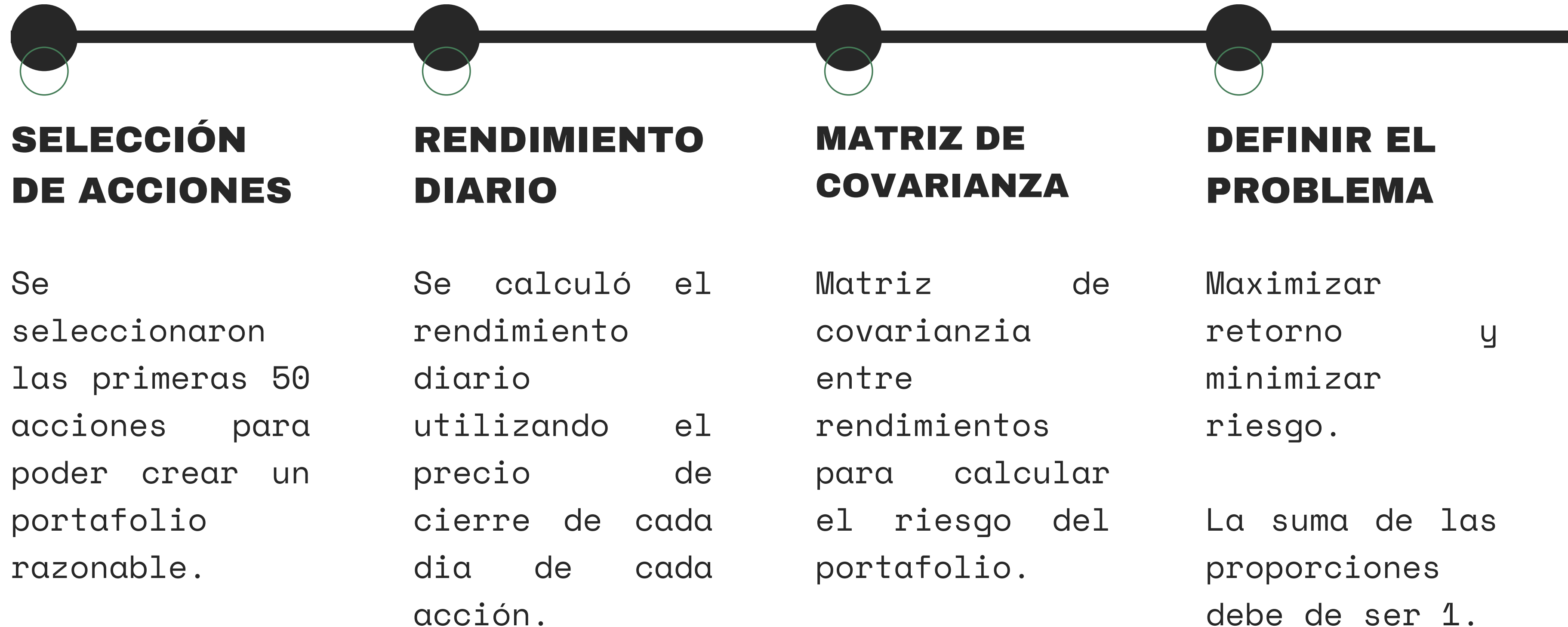
date	open	high	low	close	volume	Name
2013-02-08	15.07	15.12	14.63	14.75	8407500	AAL
2013-02-11	14.89	15.01	14.26	14.46	8882000	AAL
2018-02-07	72.70	75.00	72.69	73.86	4534912	ZTS

Ejemplo

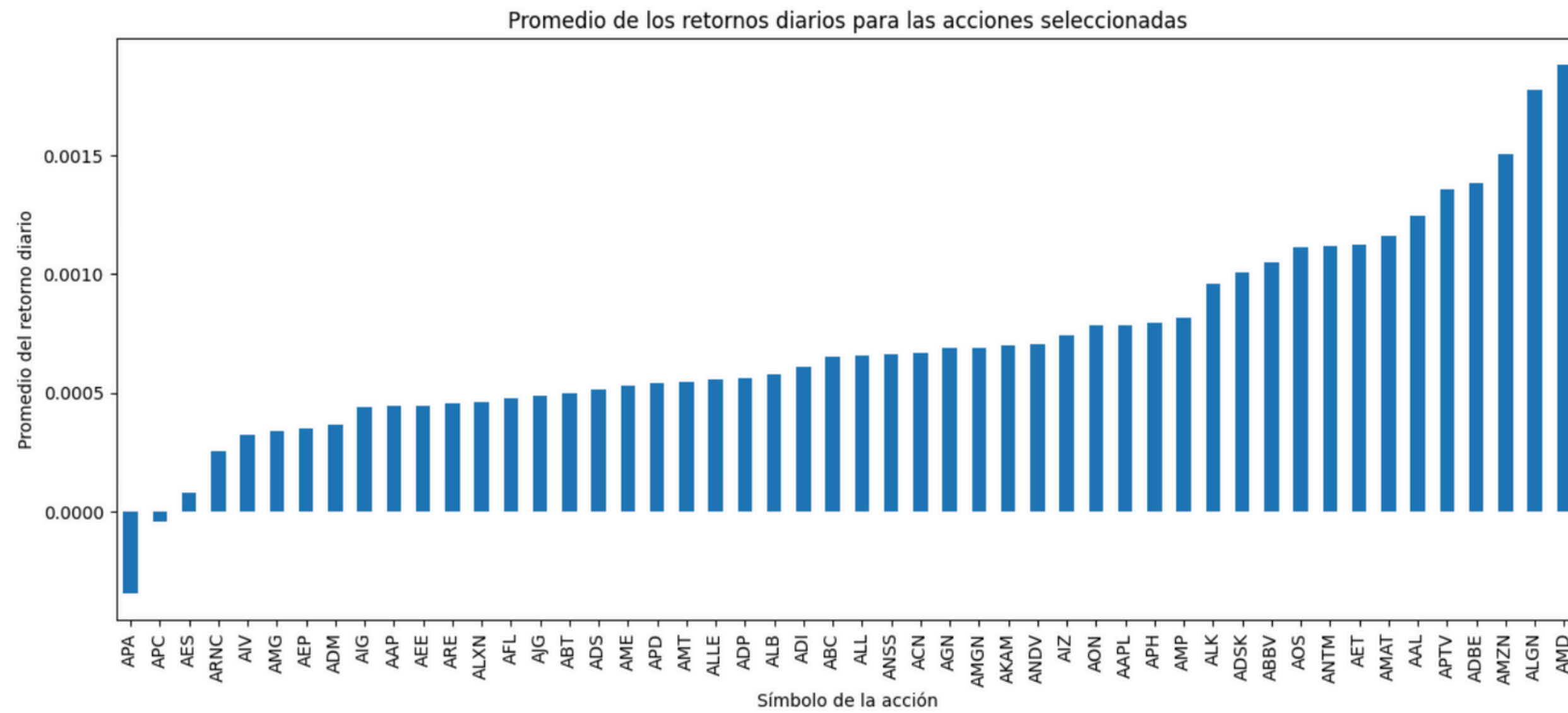
Comportamiento del precio de cierre a lo largo del tiempo de la acción “AAL”: American Airlines Group



Preparación de los datos



Promedio del retorno diario



Comparación

NUESTRA IMPLEMENTACIÓN

Realizamos nuestra propia implementación del algoritmo PSO diseñada para nuestro objetivo.

PYSWARM

Librería de Python que implementa el algoritmo PSO.

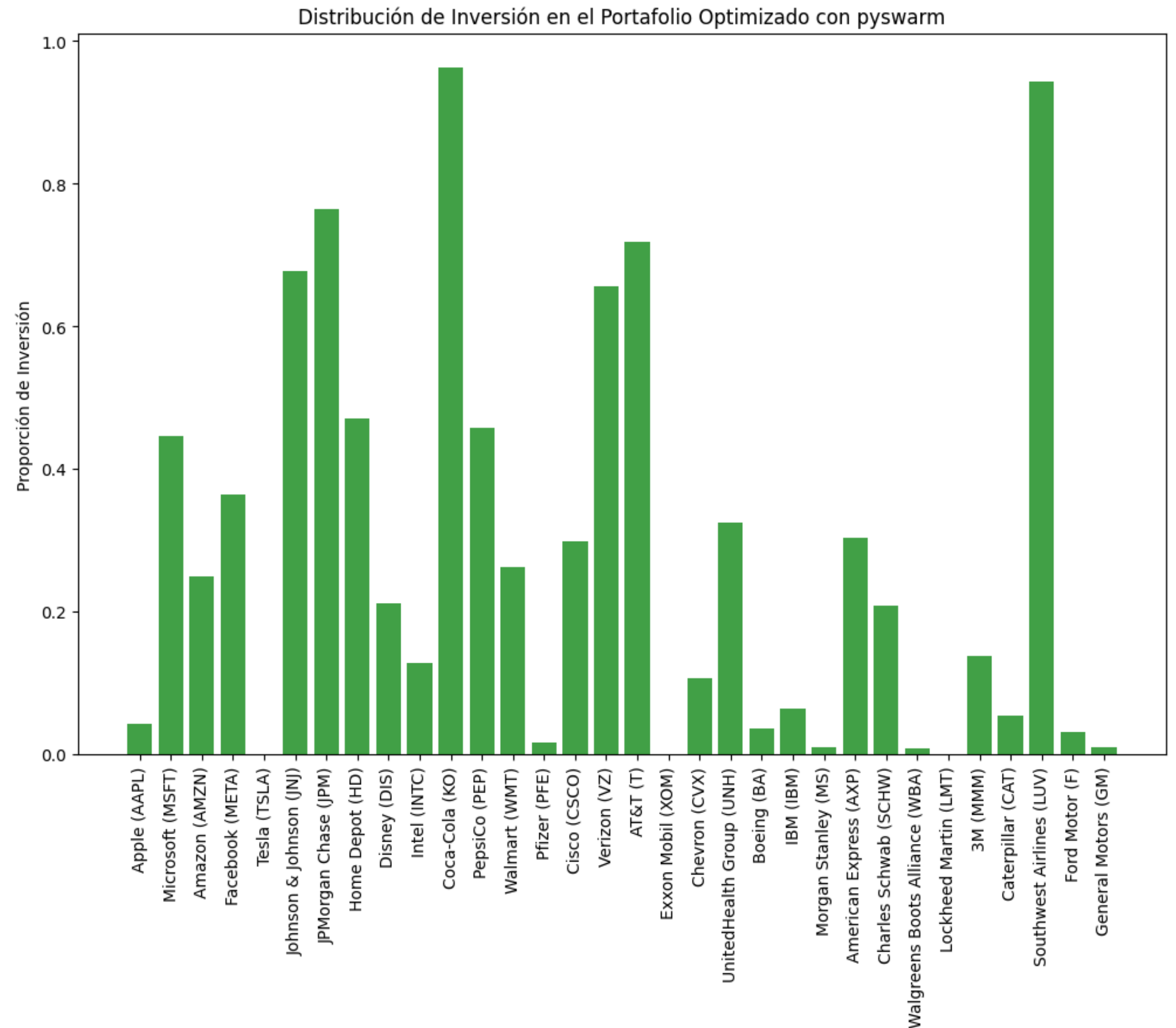
PYSWARM

Decidió invertir en
24 acciones.

Con un retorno
total de 0.71%.

Y un riesgo de
7.8%.

Valor de costo de
0.0026.



Pseudocódigo

1. Inicializar parámetros de PSO (`num_particles`, `num_iterations`, `coeficientes`).
2. Cargar datos de entrada del portafolio (`mean_returns`, `cov_matrix`).
3. Definir función objetivo (`portfolio_objective`) que calcula costo basado en retorno y riesgo.
4. Inicializar posiciones y velocidades aleatorias para las partículas.
5. Establecer las posiciones personales y globales iniciales.

6. Para cada iteración:

Para cada partícula:

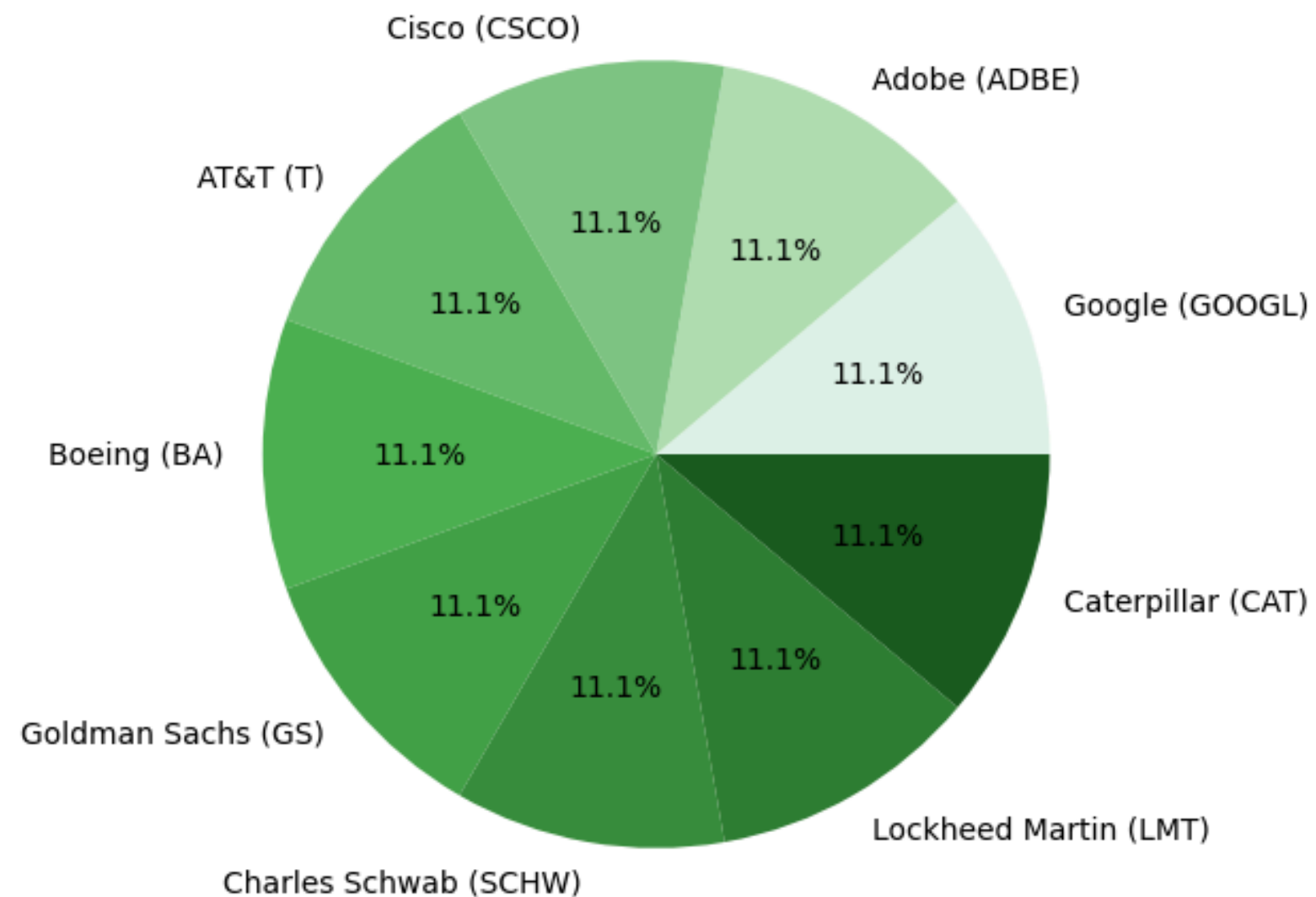
- Actualizar velocidad con inercia, componente cognitivo y social.
- Actualizar posición y normalizar para que los pesos sumen 1.
 - Aplicar restricciones de asignación máxima (si aplica).
 - Evaluar el costo de la posición actual.
- Si el costo actual es mejor, actualizar la mejor posición personal.
- Si el costo actual es mejor que la global, actualizar la mejor posición global.

7. Al finalizar, el mejor conjunto de pesos global representa el portafolio óptimo.

8. Calcular y mostrar retorno esperado, riesgo y otras métricas.

Nuestros resultados

Composición del Portafolio Optimizado



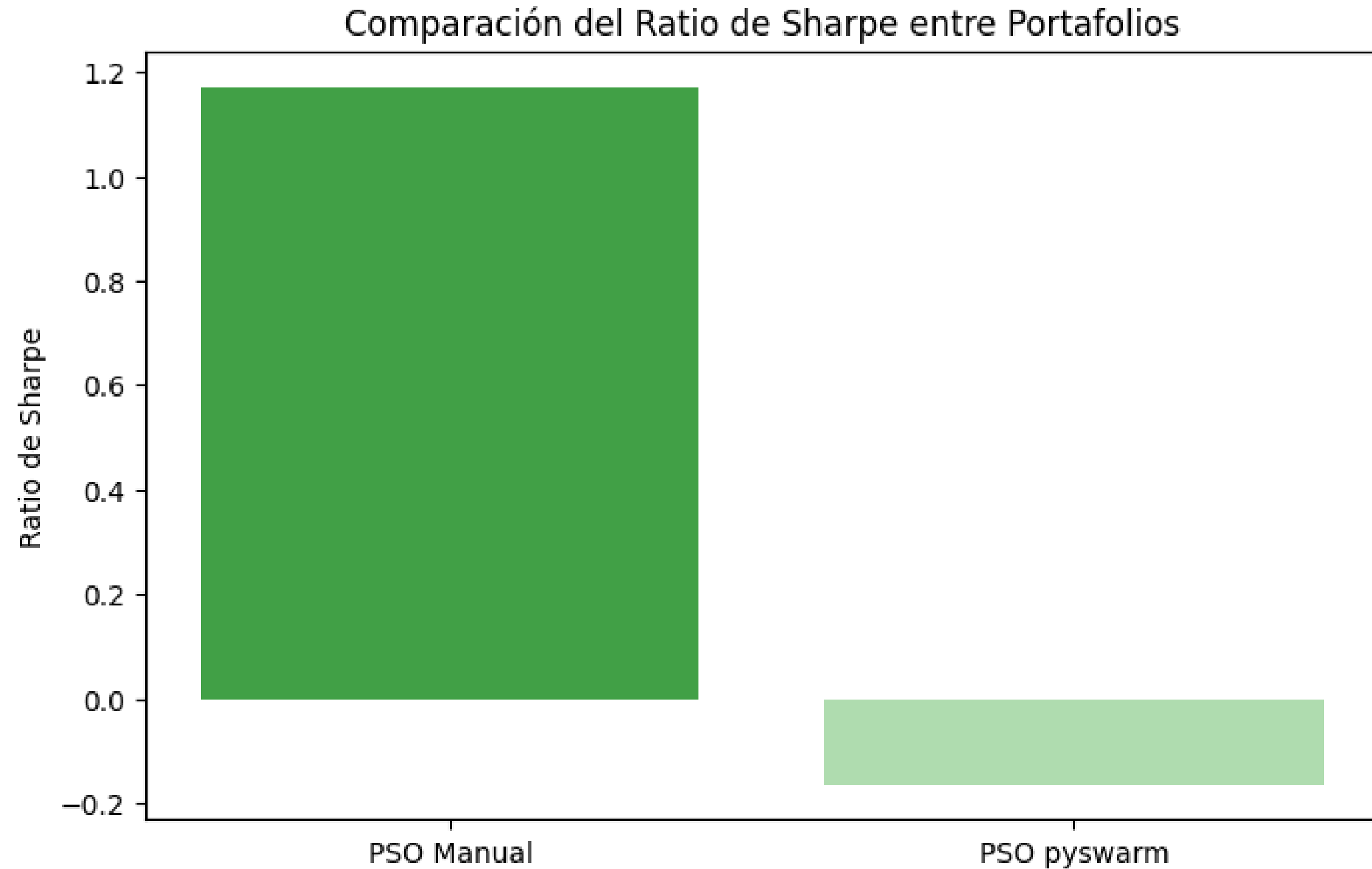
Decidió invertir en 9 acciones.

Con un retorno total de 83.1%.

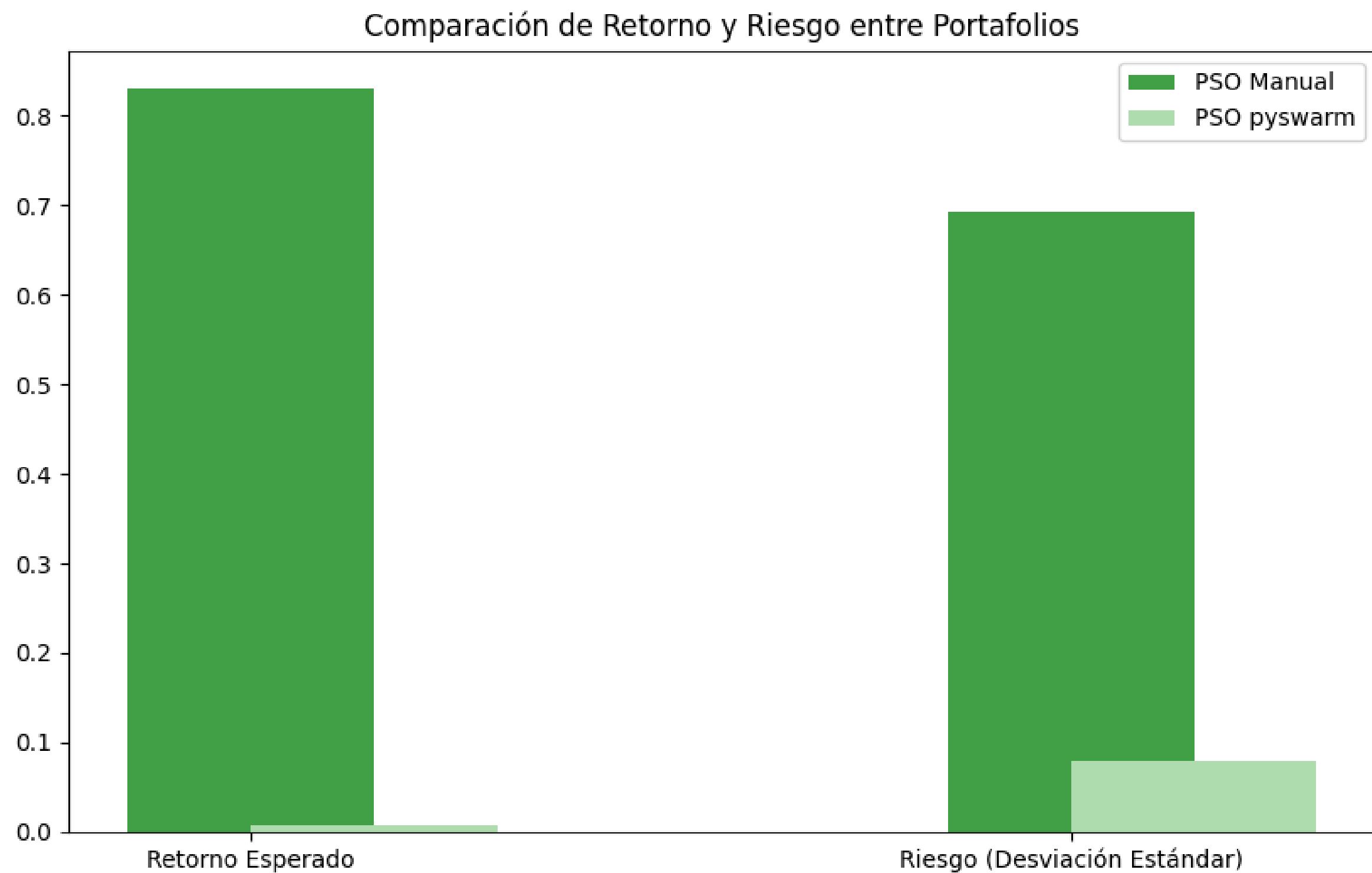
Y un riesgo de 69.3%.

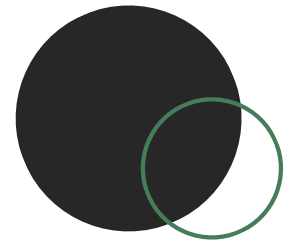
Valor de costo de -1.1823.

Comparación



Comparación





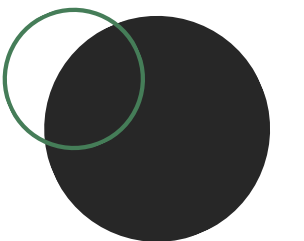
ANÁLISIS DE LOS PORTAFOLIOS

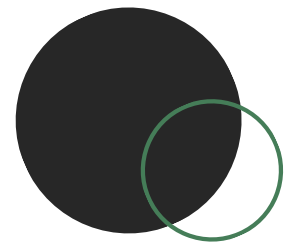
PORTAFOLIO DE PYSWARM

Ofrece una distribución más variada de inversiones proveyendo más estabilidad pero muy poco retorno.

Para inversores conservadores.

Priorizó reducir el riesgo a minimizar el valor de costo.





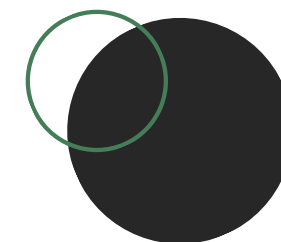
ANÁLISIS DE LOS PORTAFOLIOS

NUESTRO PORTAFOLIO

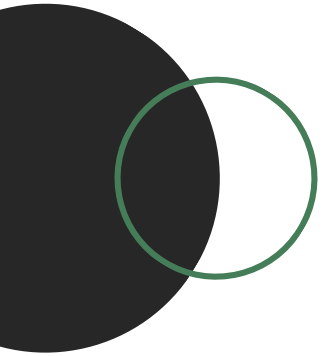
Ofrece una inversión arriesgada que prioriza el retorno frente al riesgo.

Ideal para inversores que prefieren crecimiento a estabilidad.

El algoritmo priorizo minimizar el valor de costo a reducir el riesgo,



Recomendaciones

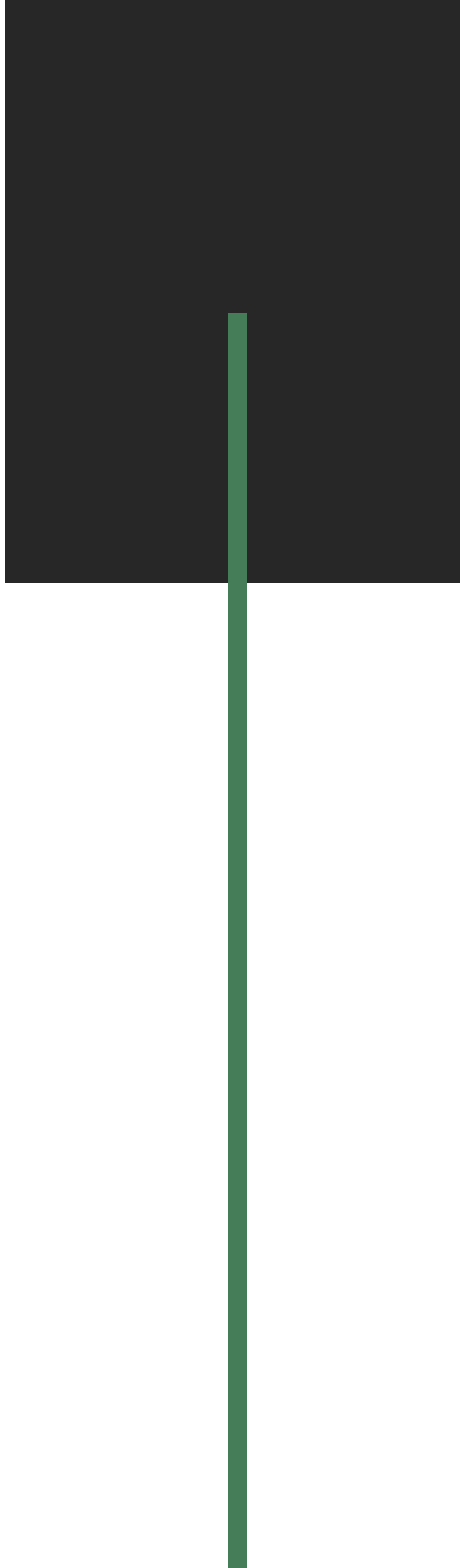


**JUGAR CON LOS
PARÁMETROS.**

**COMPARAR CON
OTROS ALGORITMOS
DE OPTIMIZACIÓN.**

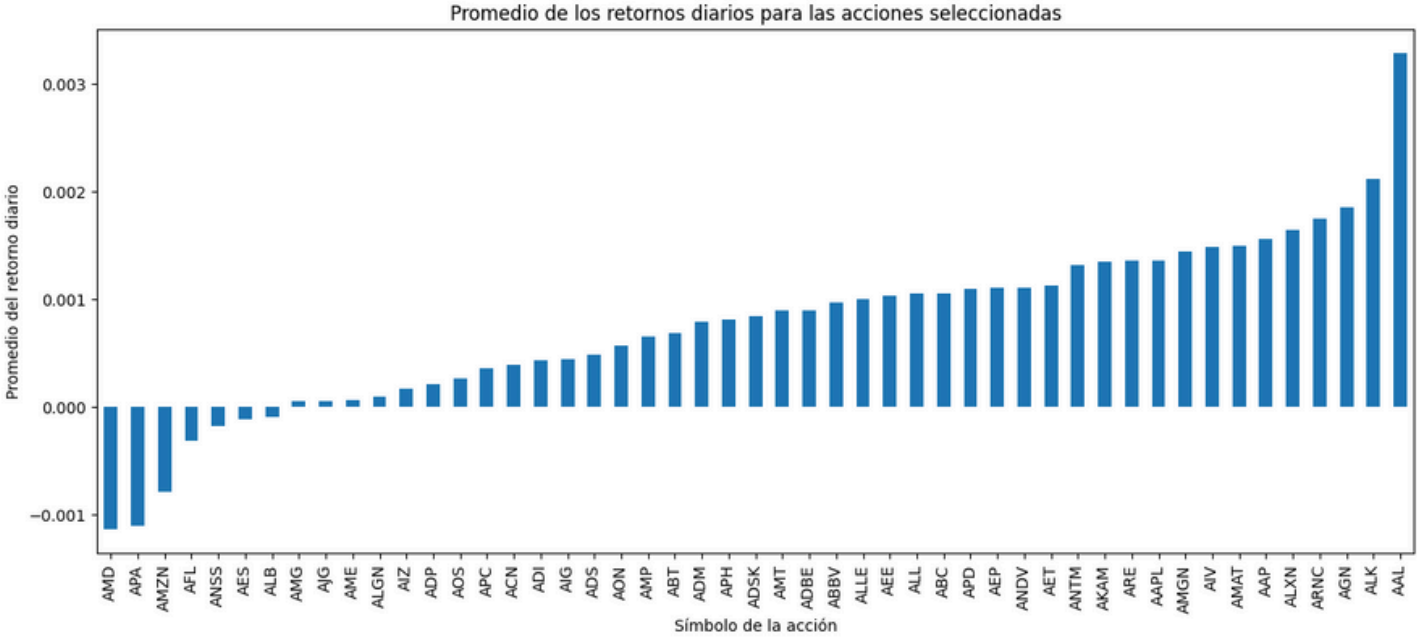
**INTENTAR CON MÁS
ACCIONES
DISTINTAS.**

**AJUSTAR LA
FUNCIÓN
OBJETIVO.**

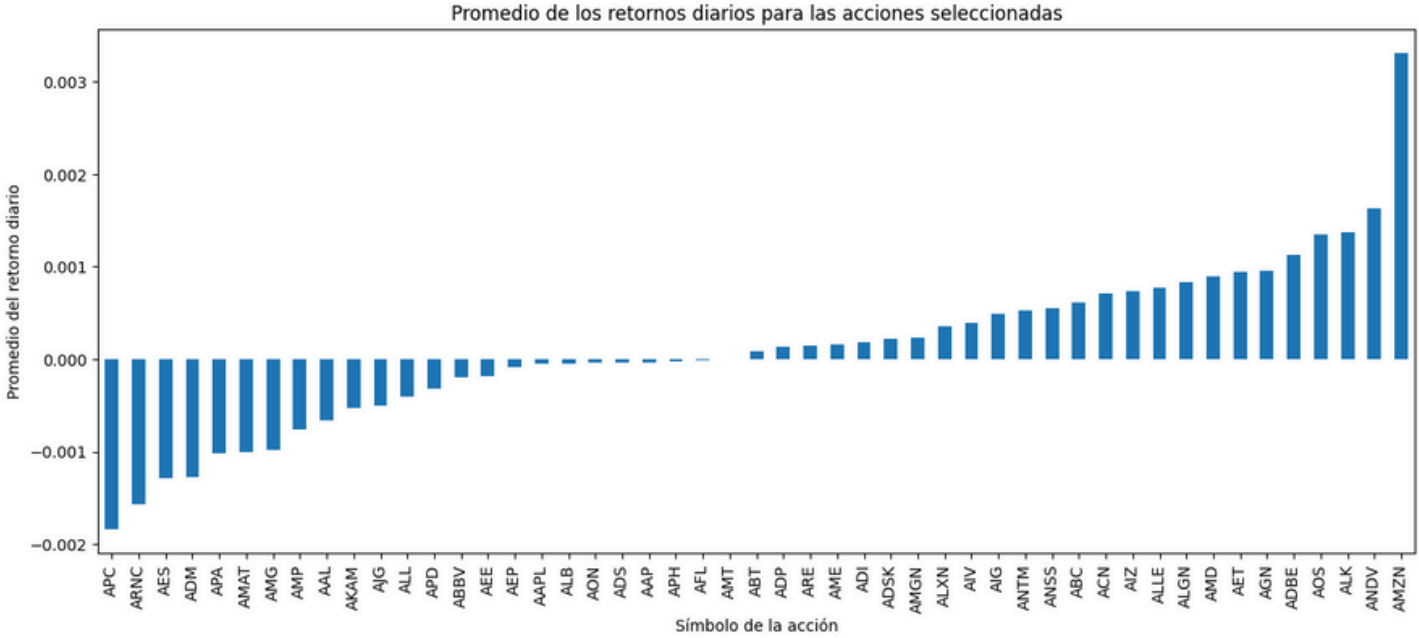


Distintas ventanas de tiempo

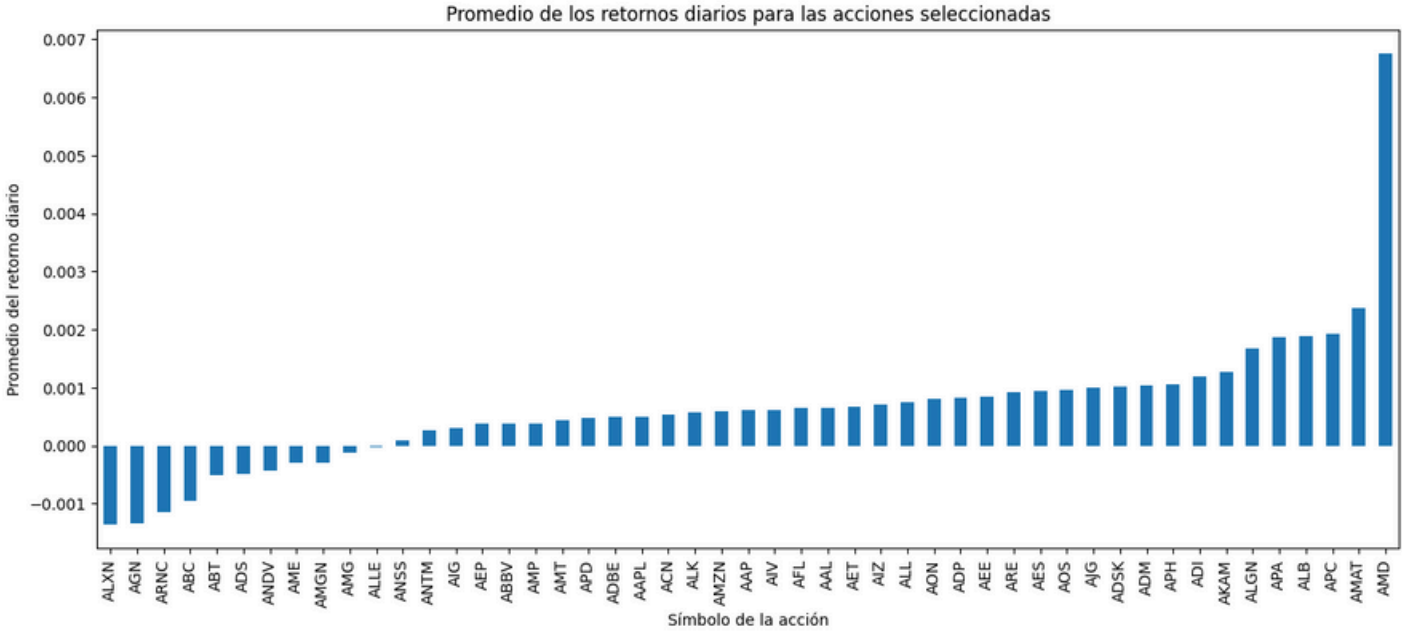
2014-01-01 α 2014-12-31



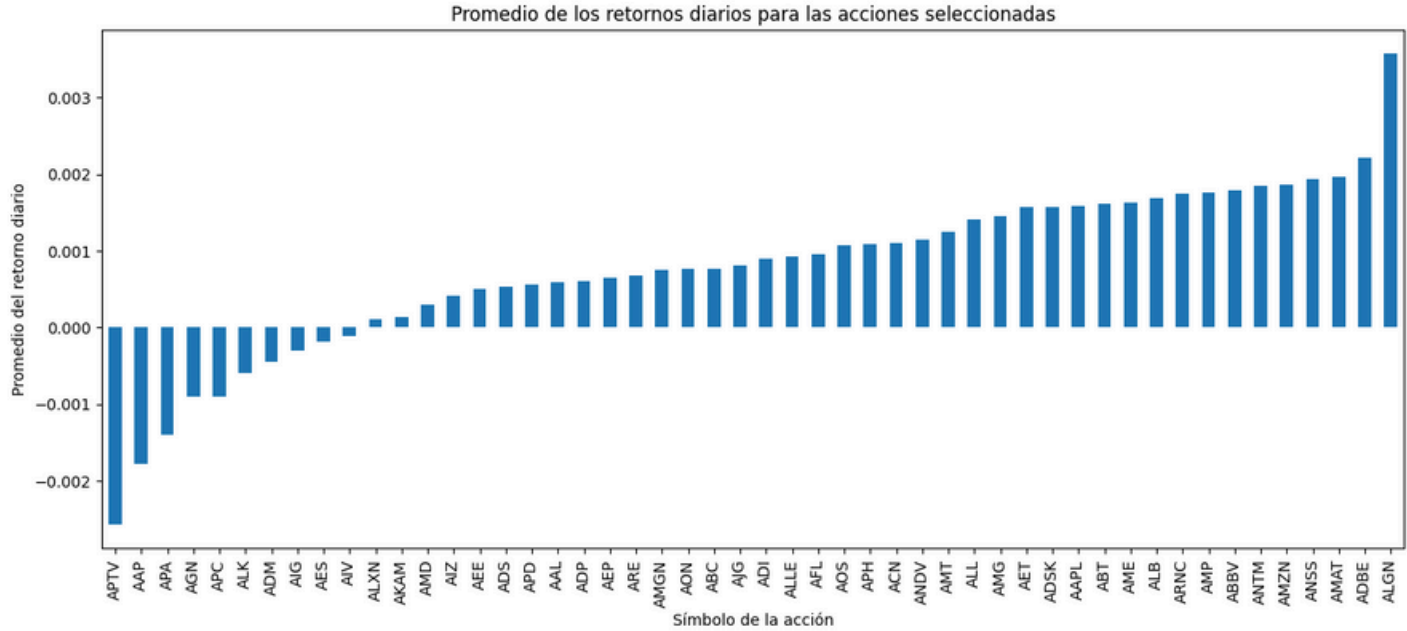
2015-01-01 α 2015-12-31



2016-01-01 α 2016-12-31



2017-01-01 α 2017-12-31

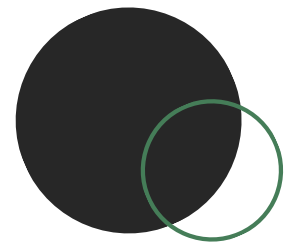


PYSWARM

Periodo	Cant acciones	Retorno total	Riesgo	Valor de costo
Todo	24	0.71%	7.8%	0.002608
'2014-01-01' - '2014-12-31'	22	0.72%	5.6%	-0.000439
'2015-01-01' - '2015-12-31'	23	0.59%	10.2%	0.007511
'2016-01-01' - '2016-12-31'	28	0.99%	9.3%	0.003625
'2017-01-01' - '2017-12-31'	34	1.98%	7.47%	-0.004349

NUESTRA

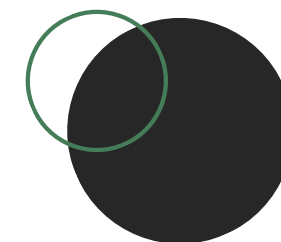
Periodo	Cant acciones	Retorno total	Riesgo	Valor de costo
Todo	9	83.1%	69.3%	-1.182254
'2014-01-01' - '2014-12-31'	10	85.7%	66.5%	-1.271731
'2015-01-01' - '2015-12-31'	10	82.8%	68.9%	-1.180665
'2016-01-01' - '2016-12-31'	9	81.0%	68.5%	-1.151395
'2017-01-01' - '2017-12-31'	10	82.4%	66.7%	-1.202546

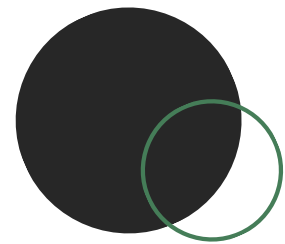


CONCLUSIONES

Se mantiene:

- Nuestra implementación da un portafolio más arriesgado con altos retornos (inversor con alta tolerancia al riesgo).
- La optimización con pyswarm es para un inversor conservador. Prioriza la estabilidad y un riesgo bajo, aunque da menores retornos.

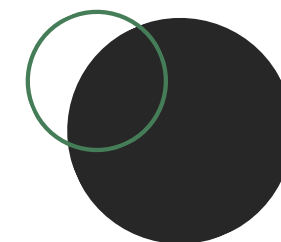




CONCLUSIONES

Además: ambos muestran variaciones en el retorno y el riesgo en cada período.

El contexto temporal impacta en la optimización.



Distintas 50 acciones

PYSWARM

Periodo	Cant acciones	Retorno total	Riesgo	Valor de costo
Todo	24	0.51%	7.0%	0.002356
'2014-01-01' - '2014-12-31'	19	0.68%	5.0%	-0.000846
'2015-01-01' - '2015-12-31'	14	0.23%	8.2%	0.005511
'2016-01-01' - '2016-12-31'	31	0.61%	8.2%	0.003596
'2017-01-01' - '2017-12-31'	28	1.81%	6.9%	-0.004182

NUESTRA

Periodo	Cant acciones	Retorno total	Riesgo	Valor de costo
Todo	9	83.1%	69.3%	-1.182254
'2014-01-01' - '2014-12-31'	14	81.7%	66.9%	-1.186853
'2015-01-01' - '2015-12-31'	9	88.7%	70.8%	-1.273279
'2016-01-01' - '2016-12-31'	12	87.4%	69.8%	-1.262128
'2017-01-01' - '2017-12-31'	14	82.1%	65.7%	-1.209748

Referencias

- Eberhart, R., & Shi, Y. (2001). Particle Swarm Optimization: Developments, Applications and Resources.
<http://silverio.net.br/heitor/disciplinas/ce/arquivos/bibliografia/PSOCEC.pdf>
- Kennedy, J., & Eberhart, R. (1995). "Particle Swarm Optimization."
- Golmakani, Hamid & Fazel, Mehrshad. (2011). Constrained Portfolio Selection using Particle Swarm Optimization. Expert Syst. Appl.. 38. 8327-8335.
10.1016/j.eswa.2011.01.020.
- Burcu Adıgüzel Mercangöz & Altaf Q. H. Badar, (2021). Optimal Portfolio Selection with Particle Swarm Algorithm: An Application on BIST-30
https://econpapers.repec.org/bookchap/sprisochnp/978-3-030-70281-6_5f9.htm
- Erik Cuevas, Miguel Cienfuegos, Daniel Zaldívar, Marco Pérez-Cisneros (2013) . A swarm optimization algorithm inspired in the behavior of the social-spider,
<https://www.sciencedirect.com/science/article/abs/pii/S0957417413003394>