Parameter optimization of a technical indicator with a genetic algorithm in the Mexican stock exchange

*Angel Omar Campa Cantu*

*Abstract*—Trying to predict future stock prices is a difficult task. Investors can try fundamental or technical analysis, but both can end up relaying in human bias, like the selections of parameters for a specific technical indicator. This work focuses on the use of a genetic algorithm to optimize the selection of parameters for a technical indicator, Relative Strength Index (RSI) and it’s effectiveness to increment the profits of a trading strategy using it’s parameters. This study found that using a genetic algorithm to optimize parameters yielded a 57% annual profit, versus a 6% annual profit gained just by using the default parameters.

*Index Terms*—

# INTRODUCCIÓN

L

a Bolsa Mexicana de Valores (BMV) es el principal lugar donde se reúnen compradores y vendedores de valores en México [1].

Los activos que en ella se encuentran son afectados por factores como el desempeño de la empresa o el estado general de la economía.

Para predecir el precio de acciones, existen dos tipos de análisis: fundamental y técnico. El primero se enfoca en los fundamentos de la empresa, como son el crecimiento, el margen de ganancias, la liquidez, entre otros. El segundo es un análisis matemático de los valores pasados del activo [2] para pronosticar tendencias futuras, empleando herramientas llamadas indicadores técnicos.

En este estudio se emplea el indicador técnico Índice de Fuerza Relativa (RSI) para encontrar momentos adecuados de compra o venta, el cual se explicará a detalle en la sección II. Este indicador técnico tiene un parámetro de entrada, al cual se les aplica una fórmula matemática y da como resultado un valor entre 1 y 100 que nos indicara si el valor se encuentra sobrecomprado o sobrevendido [3]. El valor de este parámetro de entrada es seleccionado por el usuario de forma subjetiva, así como los 2 valores que tomara como referencia para comprar o vender la acción. Estas decisiones impactan en las ganancias obtenidas.

Es aquí donde se encuentra la utilidad de emplear un Algoritmo Genético (AG), que también se explicara a detalle en la sección

II. Los AG están inspirados en la evolución biológica [4] para encontrar los valores óptimos de entrada que maximicen o minimicen una función objetivo, en este caso se busca optimizar los parámetros del indicador RSI para generar ordenes de compra o venta y maximizar las ganancias.

Si bien ya se han empleado AG para optimizar parámetros de indicadores técnicos anteriormente [5] no se encontraron artículos referentes a este uso en México. Por lo que el presente estudio busca comprobar la eficacia de un AG para optimizar parámetros de entrada y decisión en una estrategia de operación usando el indicador RSI sobre diez acciones de la BMV, y compararlo con las ganancias obtenidas por una estrategia usando los parámetros comúnmente utilizados.

# Definiciones fundamentales

## Índice de Fuerza Relativa (RSI)

El Índice de Fuerza Relativa (RSI) es un oscilador que mide la velocidad con la que los precios cambian a través del tiempo [3]. Se calcula de la siguiente manera:

RS = Ganancia promedio / Perdida promedio

Ganancia promedio = Suma de las ganancias en los n periodos anteriores / n

Perdida promedio = Suma de las perdidas en los n periodos anteriores / n

Para calcular el RSI, se debe elegir un numero de periodos a tomar en cuenta, n, el cual comúnmente es 14. Con ello se aplica la formula sumando por un lado las ganancias de n periodos anteriores entre las pérdidas de n periodos anteriores, obteniendo un valor entre 1 y 100 [6].

Este valor tiene una interpretación subjetiva, pues se considera que mientras más bajo, el valor analizado se encuentra sobrevendido y mientras más alto, sobrecomprado. Típicamente se considera como umbral un RSI menor o igual que 30 para indicar sobreventa (y por lo tanto hay que comprar el valor), y un RSI mayor o igual que 70 indica sobrecompra (por lo que hay que vender el valor).

## Algoritmos Genéticos

Un Algoritmo Genético (AG) busca maximizar o minimizar una función objetivo a través de un proceso iterativo inspirado en la evolución biológica [4]. Se compone de los siguientes pasos:

1. Generación de población inicial: Se genera una población inicial de manera aleatoria. Estos valores están compuestos por cromosomas, que son el resultado convertir el valor del individuo a números binarios (cada número binario es un cromosoma).
2. Evaluación: Se aplica la función objetivo a cada uno de los individuos y se obtiene una aptitud, que dependiendo del problema (maximizar o minimizar) será la interpretación de su valor, es decir, si estamos maximizando una mayor aptitud será benéfica y viceversa si estamos minimizando.
3. Selección: Se ordenan los individuos en base a su aptitud y se decide cuantos de ellos pasaran a la siguiente etapa.
4. Cruzamiento: Se elige un método para cruzar a los individuos, con una probabilidad elegida por el investigador, lo cual se hará en parejas, llamados padres, y dará como resultado otros 2 individuos, llamados hijos.
5. Mutación: Los hijos tendrán una probabilidad de mutación, elegida por el investigador. Con esto uno de sus cromosomas mutará (cambiará de 0 a 1 y viceversa).
6. Terminación: Se elige un número de individuos más aptos entre los padres y los hijos, los cuales pasaran a la siguiente generación, y se repite el ciclo una cantidad determinada de generaciones.

Para el presente estudio se utilizará el método del torneo para el cruzamiento, que consiste en ordenar aleatoriamente los padres, y ponerlos a competir en base a su aptitud: el primero con el segundo, el tercero con el cuarto, compitiendo de esta manera todos los padres seleccionados [7].

# Metodología

El Algoritmo Genético (AG) se codificó en Python, utilizando la librería pública fastquant elaborada por Lorenzo Ampil [8] para probar una estrategia de operación basada en el Índice de Fuerza Relativa (RSI) sobre 10 acciones representativas de la bolsa mexicana de valores en el año 2020: Cemex, América móvil, Grupo México, Bimbo, Arca Continental, Grupo Elektra, Grupo Televisa, Fomento Económico Mexicano, Genomma Lab, Quálitas Controladora.

El algoritmo funciona de la siguiente manera:

1. Se obtienen los precios de cierre de cada día operativo durante el año 2020 para cada una de las 10 acciones y se inicializa el valor del portafolio con $100,000 pesos mexicanos.
2. Se decide el valor de n, el cual será tanto la población inicial como el número de individuos que pasaran a la siguiente generación.
3. Se decide el número de generaciones (G), que será el número de iteraciones del algoritmo.
4. Se genera la población inicial. Cada individuo de la población inicial estará compuesto por 3 valores: el número de periodos utilizado para el RSI (entre 3 y 100), el valor de sobrecompra que indicará cuando se deba vender la acción (entre 51 y 100) y el valor de sobreventa que indicara cuando se debe comprar la acción (entre 3 y 50). Cada individuo de la población estará compuesto por estos 3 valores, por lo que cada uno tendrá 3 conjuntos de cromosomas. Ejemplo: Individuo 1 = [14, 70, 30]
5. Se aplica la función de aptitud en cada uno de los individuos, que en este caso es introducir los 3 valores de cada individuo a una función de la librería fastquant que prueba la estrategia utilizando el indicador RSI sobre los precios de cierre de la acción. Si el valor del RSI en un momento determinado está por debajo del valor de sobreventa que tiene el individuo con el que se está trabajando, el algoritmo comprara las acciones con todo el capital disponible y si el valor del RSI se encuentra por encima del valor de sobrecompra, el algoritmo venderá las acciones que posee. El resultado es un valor final del portafolio (aptitud), que reflejará perdidas o ganancias.
6. Se inicia el torneo, se reordenan aleatoriamente para que los padres compitan en base a su aptitud uno a uno, y se itera este proceso hasta lograr n padres.
7. Después de seleccionar por el método del torneo a los padres, va a haber n/2 cruzas. En cada cruza se seleccionan 2 padres aleatoriamente de la lista de padres y se verifica que no tengan los mismos cromosomas. Se selecciona una probabilidad de cruza de 0.5 y se genera un numero aleatorio entre 0 y 1. Si este número es mayor a 0.2, los padres se cruzarán, dando como resultado 2 hijos.
8. Para la cruza se convierten cada uno de los 3 valores de cada padre a números binarios con 8 dígitos. Estos se cruzarán tomando los primeros 2 dígitos del primer valor del primer padre, junto con los 4 dígitos siguientes del primer valor del segundo padre y los últimos 2 dígitos del primer valor del primer padre. Esto se realiza para cada uno de los 3 valores y se obtiene el primer hijo.
9. El segundo hijo se obtiene tomando los primeros 2 dígitos del primer valor del segundo padre, junto con los 4 dígitos siguientes del primer valor del primer padre y los últimos 2 dígitos del primer valor del segundo padre. Esto se realiza para cada uno de los 3 valores y se obtiene el segundo hijo.
10. Después de realizar n/2 cruzas, sigue la mutación. Se elige una probabilidad de mutación de 0.2, y se iterara sobre la lista de hijos. En cada uno se seleccionará un valor entre 0 y 1, y si este valor es mayor a 0.2, este hijo mutará
11. Para cada mutación se convertirá cada uno de los 3 valores del individuo a números binarios, y se seleccionará aleatoriamente un digito en cada valor, el cual se cambiará de 0 a 1 o de 1 a 0.
12. Se juntan las listas de padres e hijos en una sola, y se desechan aquellos que tengan un valor de sobrecompra o sobreventa mayor a 100 o menor a 3 para que el algoritmo pueda operar en la bolsa.
13. Se aplica nuevamente la función de aptitud a la lista combinada de padres e hijos, obteniendo el valor final del portafolio al operar en el año 2020.
14. Se ordenan los valores de padres e hijos en función de su aptitud, y se seleccionan los n individuos mas altos, que pasarán a la siguiente generación.
15. Se repiten los pasos 5 a 14 durante el número de generaciones seleccionado.

Al final, el algoritmo arrojará al individuo más apto encontrado, con sus 3 valores (periodo, sobrecompra y sobreventa) y el valor del portafolio al final del periodo. Este algoritmo se aplicó a las 10 acciones antes mencionadas, y en cada una de las acciones se utilizó el algoritmo 4 veces, con los siguientes valores de n y G: 10 y 5, 10 y 10, 20 y 10, 50 y 10 respectivamente.

# Resultados

El resultado de cada una de las acciones se promedió y se agrupó según el numero de individuos de la población inicial (N) y el numero de generaciones o iteraciones del algoritmo genético (G), así como el beneficio de operación promedio sin ningún tipo de optimización, es decir, usando los parámetros normales. Esto se muestra en la tabla 1.



Tabla 1. Elaborada en Excel.

Finalmente, los resultados se dividieron y promediaron entre beneficios promedio sin optimización y beneficios promedio con alguna combinación de N y G para optimización, como se muestra en la tabla 2.



Tabla 2. Elaborada en Excel.

# Conclusiones

El uso de Algoritmos Genéticos para optimizar parámetros del indicador RSI para operación en la Bolsa Mexicana de Valores resultó ser exitoso, logrando una ganancia promedio del 57% anual comparado con un 6% anual cuando no se optimiza.

# Referencias

[1] Torres, O. A. (2014). *Análisis Economico.* Guadalajara: Astra Editorial.

[2] Montri Inthachot, V. B. (2016). Artificial Neural Network and Genetic Algorithm Hybrid Intelligence for Predicting Thai Stock Price Index Trend. *Computational Intelligence and Neuroscience*, 8.

[3] Dr. Bhargavi. R, D. S. (2017). Relative Strength Index for Developing Effective Trading Strategies in Constructing Optimal Protfolio. *International -Journal of Applied Engineering Research*, 10.

[4] Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning.* USA: Addison-Wesley Longman Publishing Co., Inc.

[5] Janko Straßburga, ,. C.-M. (2012). Parallel genetic algorithms for stock market trading rules. *International Conference on Computational Science* (pp. 1306 - 1313). Procedia Computer Science.

[6] Ţăran-Moroşan, A. (2011). The relative strength index revisited. *African Journal of Business Management*, 5855 - 5862.

[7] Reeves, C. R. (2010). *Handbook of Metaheuristics.* Coventry: Coventry University.

[8] Ampil, L. (2020). *fastquant*. Retrieved from Github: https://github.com/enzoampil/fastquant