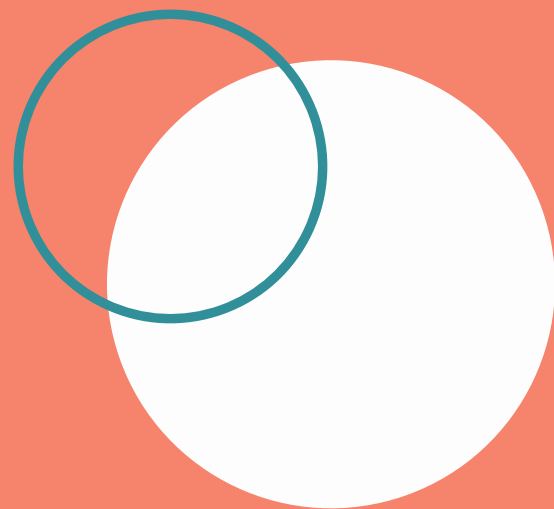




# AN AUTOMATIC STOCK TRADING SYSTEM USING PARTICLE SWARM OPTIMIZATION

Rafel Fernández Ortiz  
2022



# Index

## 1. INTRODUCTION

## 2. THE PROPOSED ALGORITHM

*2.1. TRADING SIGNALS FROM TECHNICAL INDICATORS*

*2.2. OPTIMIZATION WITH PSO*

*2.3. TRADING DECISION*

## 3. EXPERIMENTATION

*3.1. PARAMETERS*

*3.1. PERFORMANCE, RESULTS AND DISCUSSIONS*

## 4. CONCLUSION AND FUTURE WORK .

# Introduction



**The idea:** Resolve the trading signal inconsistencies, proposing a trading strategy based on a learning method to combine a set of technical trading signals.



**BUY**

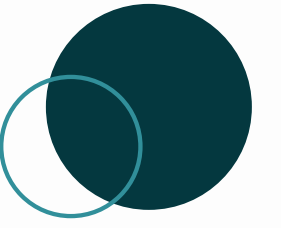


**SELL**

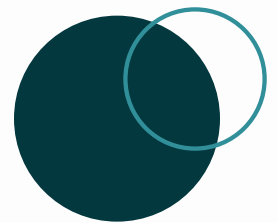


**HOLD**

**Keywords:** *Particle Swarm Optimization, Trading strategy, Technical indicators*



# The proposed algorithm



## FACTS

- Namely **PSOTraiding**
- **Required inputs:** daily closing prices, open prices, volumes, high prices and low prices of the stock in
- **Source:** Yahoo finance



# The proposed algorithm

## Trading Signals from Technical Indicators

- **Simple Moving Average (SMA)**. Here, we use 100 days moving average.
- **Moving Average Convergence Divergence (MACD)** (with EMA12 and EMA26)
- **Relative Strength Index (RSI)**  
value ranges in 0 to 100.  
Buy < 30. Sell > 70
- **Stochastic Oscillator (STO)**  
help identify trend velocity and movement.
- **William Percent Range (%R)**  
value ranges in 0 to 100.  
Overbought < 30. Oversell > 70



# The proposed algorithm

## Optimization with PSO

- Vector of **weights** for trading signals is represented in the PSO as the position vector of each particle

- Decision function

$$Decision_d = (w_1s_1 + w_2s_2 + \dots + w_ns_n) / \sum_i^n w_i$$

where trading signals  $s_1, s_2, ..$  are generated from each indicator. And  $w_1, w_2, ...$  are weights for each corresponding signal

- These weights are to be **optimized by a modified PSO** and thus they are the **decision variables of the optimization problem**
- Each value that Decision function returns, determines the trading decision for day  $d$

# The proposed algorithm

## Trading Decision

- We consider a parameter Threshold ( $t$ ) between -1 and 1
- If threshold is next to 0, frequency trading increase.

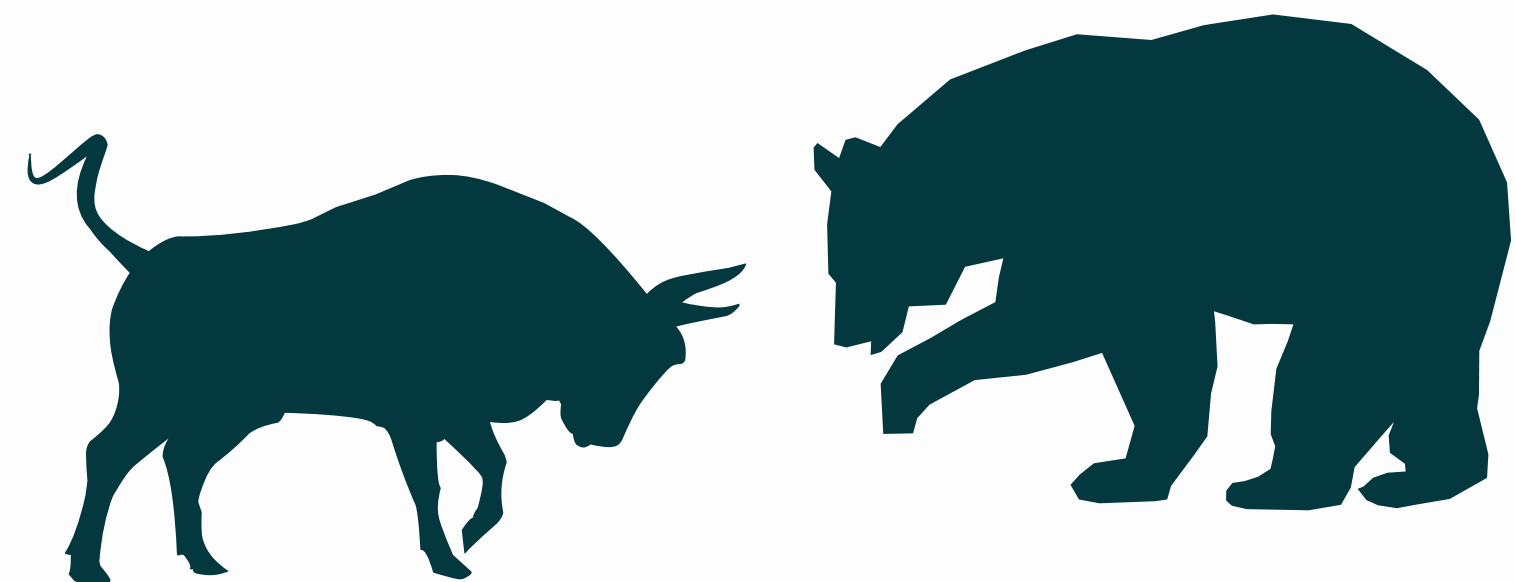
### Decision

- If decision is higher than  $t$ , then BUY
- if decision is lower than  $t$ , then SELL
- Otherwise, HOLD (or do nothing)

### Fitness Function

$\text{return} = (\text{final portfolio} - \text{initial portfolio}) / \text{initial portfolio}$

### VS Buy&Hold Strategy



# EXPERIMENTATION

- Used active stocks: **APPLE**
- Tested Period: **One year (2021)**
- Apache Spark Framework





# Experimentation

## Parameters

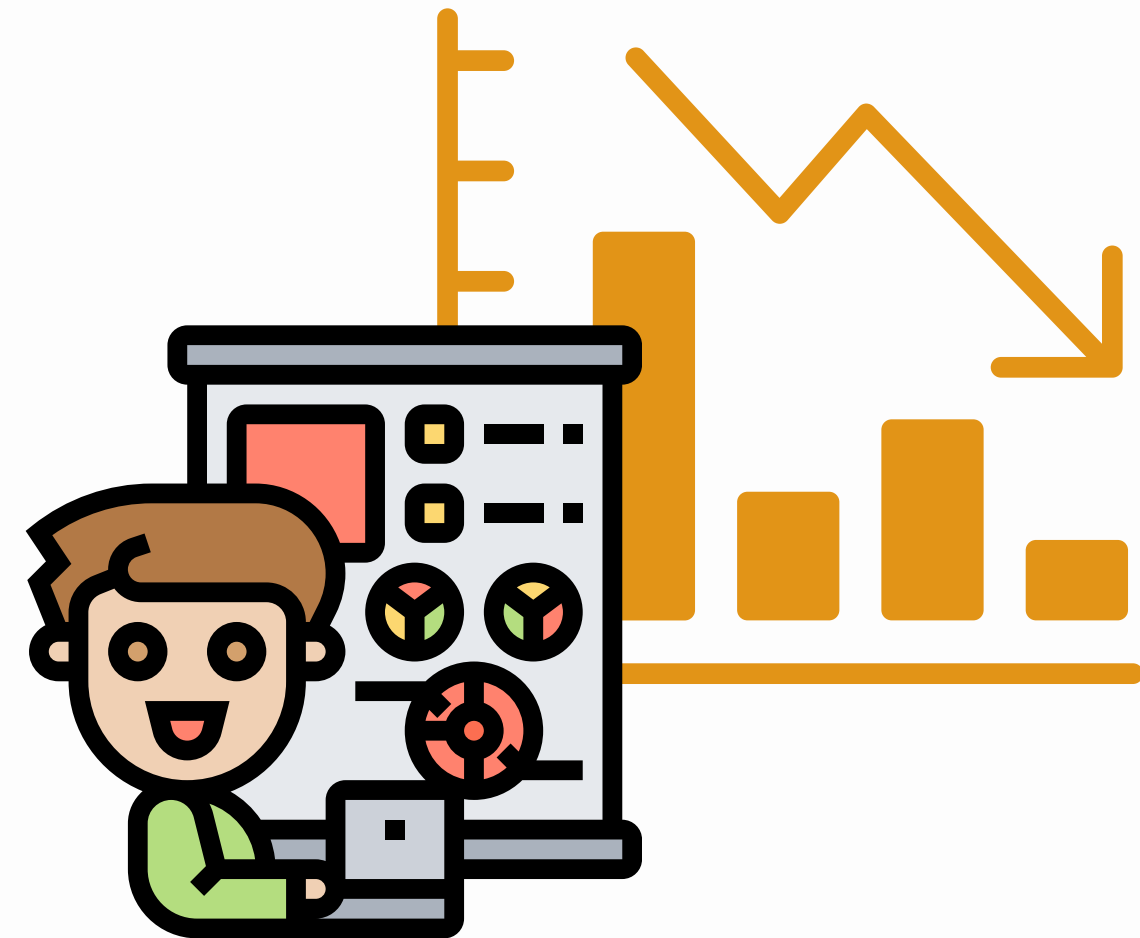
- **C1** from 2.5 to 0.5
- **C2** from 0.5 to 2.5
- Inertia **W<sub>i</sub>** from 0.9 to 0.4
- **Threshold** 0.1, 0.3, 0.5
- **Iteration** = 20
- **Swarm Size** = 10 particles
- Buying with **¼ of the available capital** and selling **¼ of the stocks we have** at that moment



# Experimentation

## Performance, Results and Discussions

- Buy&Sell sin comisiones
- Buy&Sell con comisiones
- Buy&Hold



# Conclusion & Future Work

## WHAT WE CAN DO TO IMPROVE?

- Cloud
- Improve Buy&Sell system
- Cryptocurrencies





# **AN AUTOMATIC STOCK TRADING SYSTEM USING PARTICLE SWARM OPTIMIZATION**

Rafel Fernández Ortiz  
2022

