ZeltaLabs

Kharagpur Data Science Hackathon

Team- CognitiveCryptos

KSHITIJ 19TH-21ST JAN 2024 TECHNO-MANAGEMENT FEST IIT KHARAGPUR

KHARAGPUR DATA ANALYTICS GROUP

OUR TEAM

THE FOUR MEMBERS OF OUR TEAM ARE

Shuvraneel Mitra

Agnij Biswas

Nabayan Saha

Nabarup Ghosh

OUR **OBJECTIVES:**

Developing Advanced Machine Learning Models for Algorithmic Trading in BTC/USDT

- **Forecasting Accuracy:** Create predictive models to accurately forecast BTC/USDT price movements.
- **Risk Management:** Implement strategies to maximize returns while minimizing risks.
- **Method Comparison:** Compare performance across various machine learning techniques - Time Series Analysis, Regression, Deep Learning, Evolutionary Algorithms.
- **Back-testing for validity:** Validate model effectiveness through rigorous back-testing against historical data.
- **Innovation Emphasis:** Utilize state-of-the-art techniques to drive innovation in predictive modeling.
- **Benchmark Outperformance:** Aim to surpass existing market benchmarks with advanced algorithmic strategies.

we want to leverage machine learning methods to establish new benchmarks in the dynamic and challenging crypto trading arena.

TIME-SERIES

ANALYSIS

TIME SERIES ANALYSIS

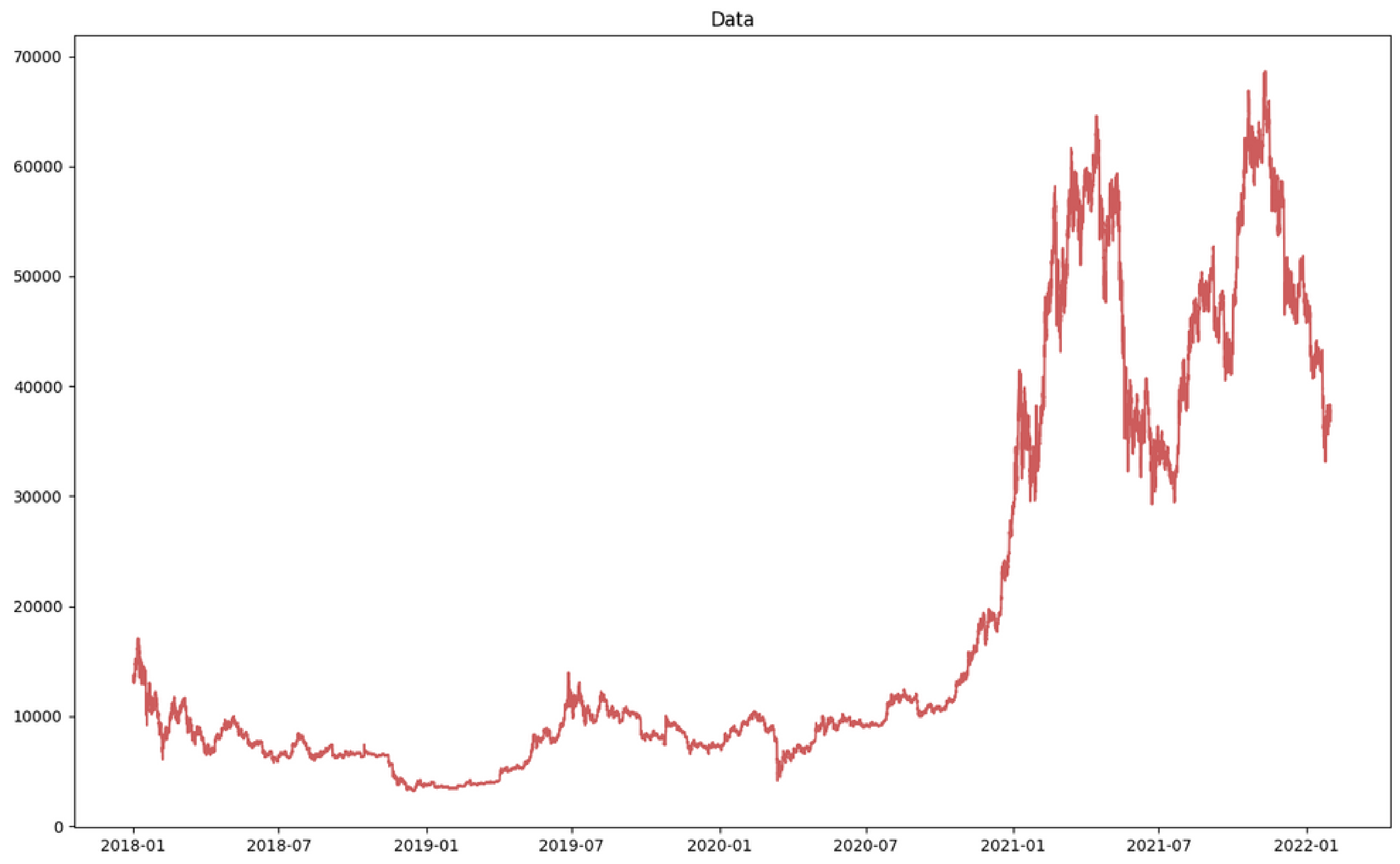
TSA

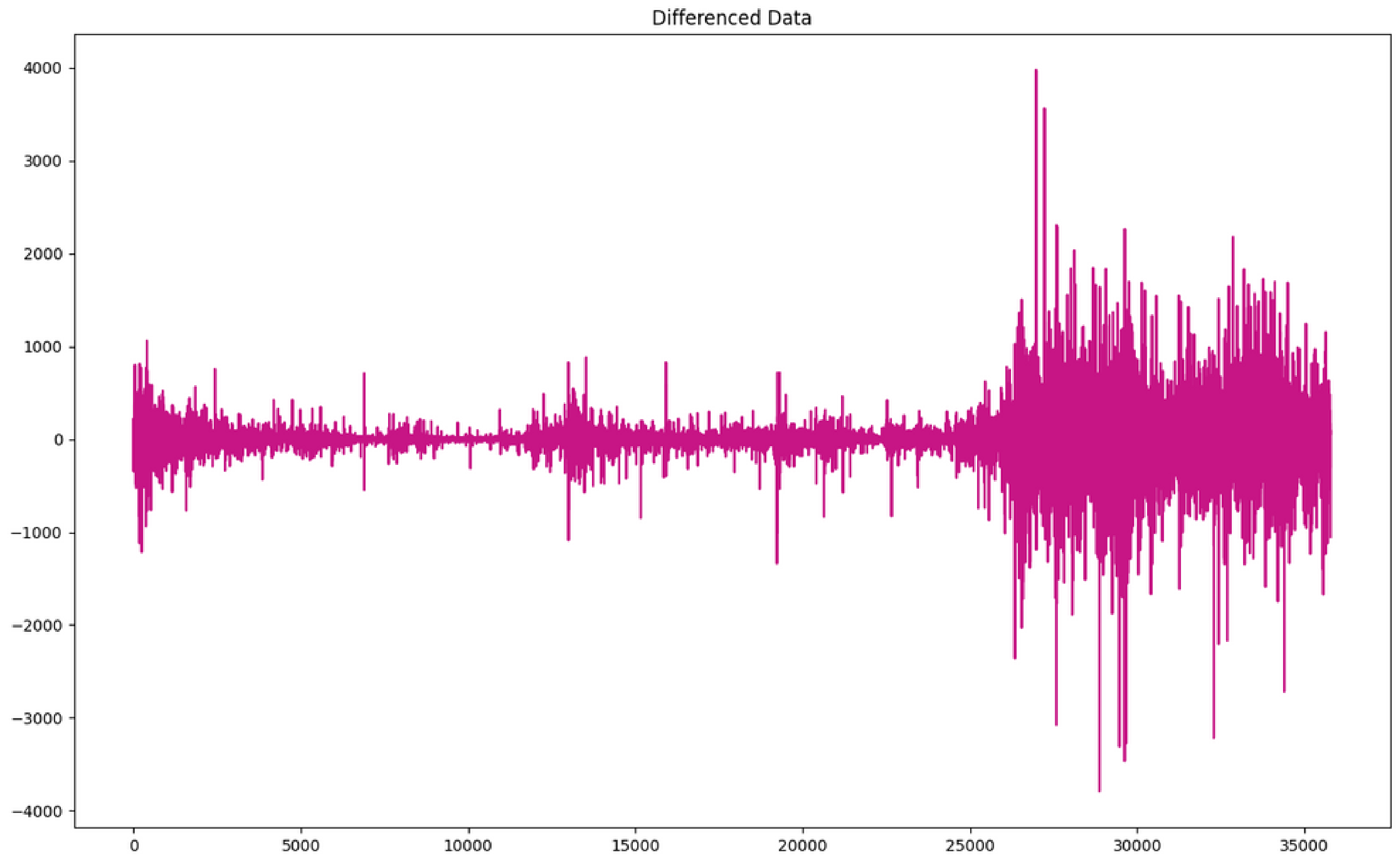
Utilizing Time Series Analysis to predict BTC/USDT price movements, employing models like ARIMA and STL to capture underlying trends and seasonality in the cryptocurrency market.

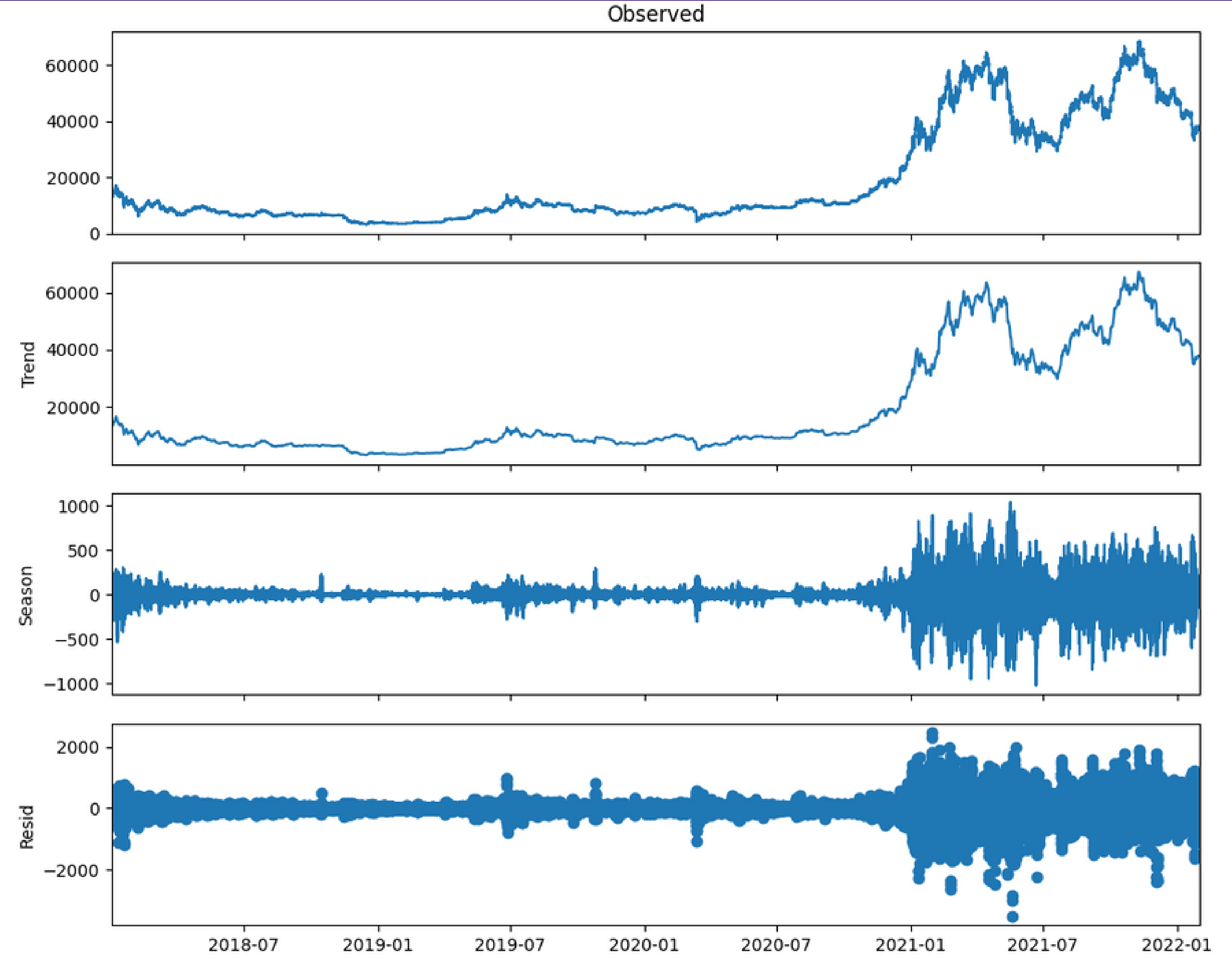
The attached plot shows the extreme voltatility that is typical to the crypto universe . The lower plot is that for the differenced closing price, i.e.

DIFFERENCE(t) = CLOSE(t + 1) - CLOSE(t).

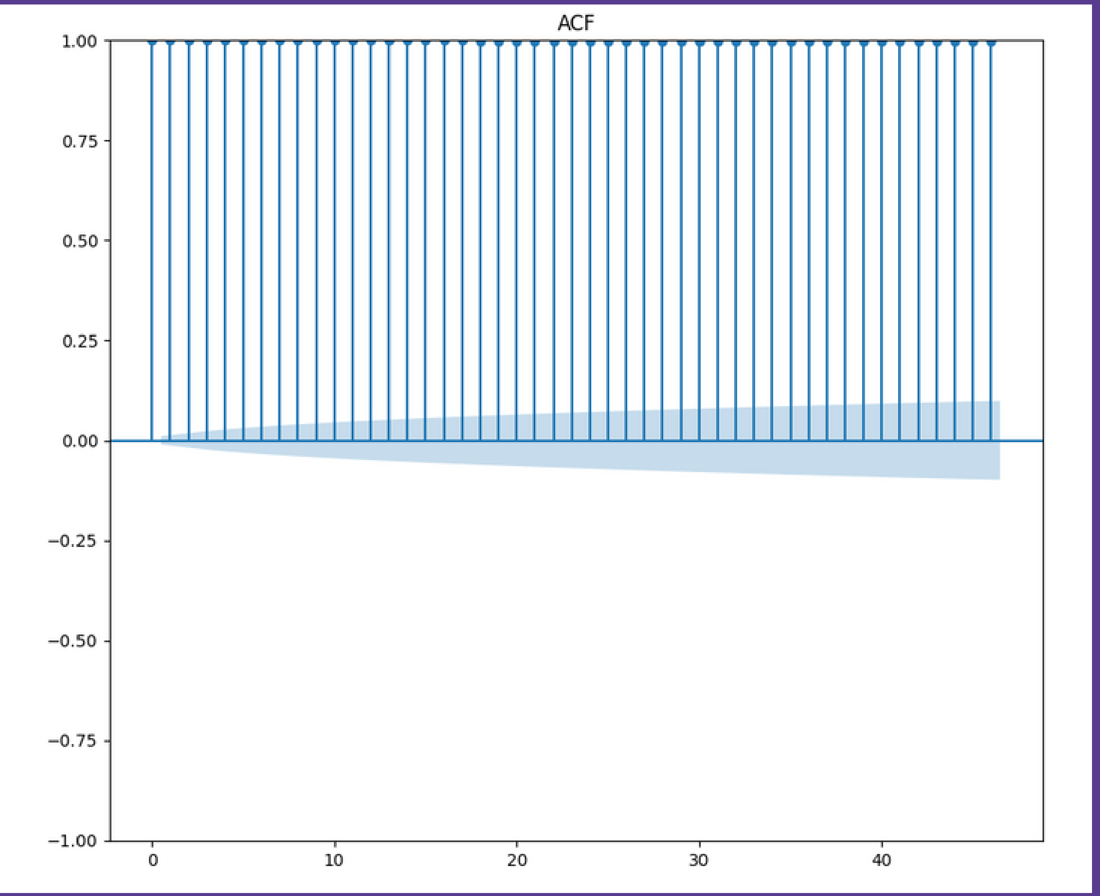
As is apparent, the first difference of the data is almost completely stationary, which is further evidenced by running the Augmented Dickey-Fuller test on the differenced data, giving p~0.0 .



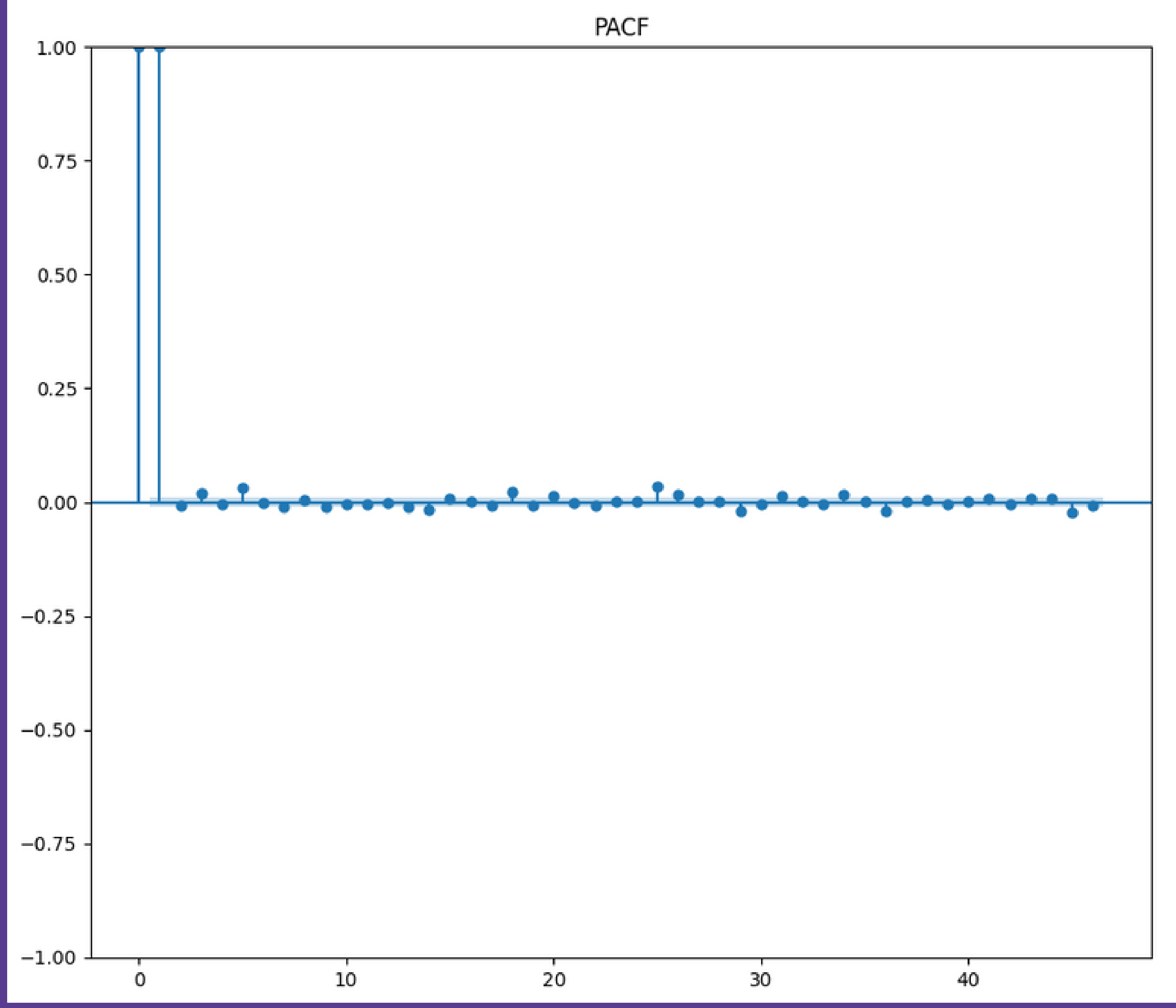




The seasonal decomposition (by STL) plot says a lot: the "trend" is just a de-noised version of the original data, while the "seasonal" and "residual" terms are almost pure white noise.



The ACF and PACF plots offer much deeper insight: the ACF plot of this data decays VERY slowly (1.00 to 0.97 in 46 lags) and this indicates a very steep trend. Also, the PACF plot cuts off sharply after lag 2, which might suggest modelling the data after an AR(2) process or a superset thereof.



REGRESSION

ANALYSIS

Random Forest
Regressor

Random Forest Regressor is an ensemble learning algorithm used for regression tasks. It belongs to the family of decision tree-based algorithms and is an extension of the Random Forest algorithm, which is commonly used for classification. Random Forest Regressor builds multiple decision trees during training and merges their predictions to obtain a more accurate and stable prediction.

Random Forest Regressor (RFR) gave a **R2 score** of **0.9734674817368255** on the dataset for a 80:20 split where we used the first 80% data for training and the last 20% for testing.

XGBoost
REGRESSOR

XGBoost (Extreme Gradient Boosting) is a popular machine learning algorithm that belongs to the family of gradient boosting algorithms. While gradient boosting is typically used for classification problems, XGBoost can also be applied to regression tasks. XGBoost Regressor is an implementation of XGBoost designed for regression problems.

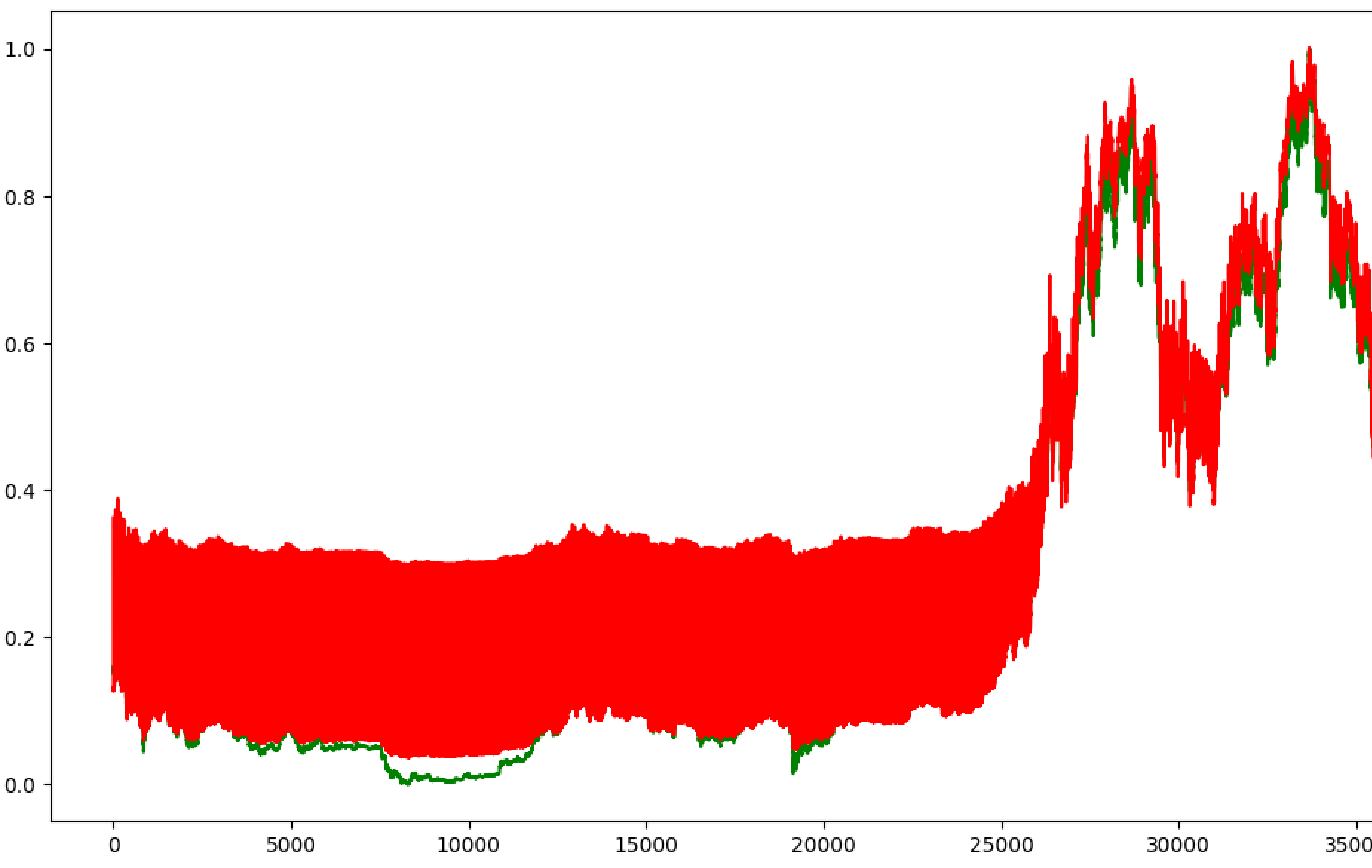
XGBoost Regressor gave a **R2 score** of **0.9137534427536891** on the dataset for a 80:20 split where we used the first 80% data for training and the last 20% for testing.

Long Short-Term Memory

LSTM

LSTM (Long Short-Term Memory) networks are a type of recurrent neural network (RNN) designed to handle sequence prediction problems. Unlike traditional RNNs, LSTM has the ability to remember information for long periods, which is achieved through its unique internal structure of gates that regulate the flow of information. These networks are particularly effective in applications like language modeling, speech recognition, and time series prediction

We have used the ResNLS architecture for designing this model from the paper: “ResNLS: An Improved Model for Stock Price Forecasting”



REJECTION OF
REGRESSION ANALYSIS

The model would not perform well with absence of some data. The model when trained with 80 percent of data and tested on 20% showed an R2 score > 0.9, but as soon as the training data was reduced to less than 75%, the R2 score fell to very low values

Even after training with 80% of the data, the classification accuracy of the model (to predict whether the price is supposed to go up or down was lower than 50%). Since our end goal is to predict whether the price would go up or down and take a position accordingly, we were forced to reject this approach.

TECHNICAL

INDICATORS

Moving Average Convergence Divergence

MACD

MACD (Moving Average Convergence Divergence) is a widely-used technical analysis indicator that helps traders identify trends and momentum in financial markets. It consists of two lines: the MACD line, which is the difference between a fast and a slow moving average of prices, and the signal line, which is an average of the MACD line itself. By observing the crossover of these lines and the divergence from the price chart, traders can make informed decisions about market entry and exit points.

Relative Strength Index

RSI

RSI (Relative Strength Index), and it is a popular technical indicator used in financial markets to assess the magnitude of recent price changes to evaluate overbought or oversold conditions in an asset's price. Developed by J. Welles Wilder, RSI is a momentum oscillator that measures the speed and change of price movements. It is typically used in the analysis of stocks, commodities, currencies, and other financial instruments.

RSI = 100−100/(1+(Avg. Gain/Avg. Loss))

RS (Relative Strength) is the average of 'n' days' up closes divided by the average of 'n' days' down closes. The most common period for RSI calculation is 14 days.

RESULTS BEFORE

LOOKBACK OPTIMIZATION

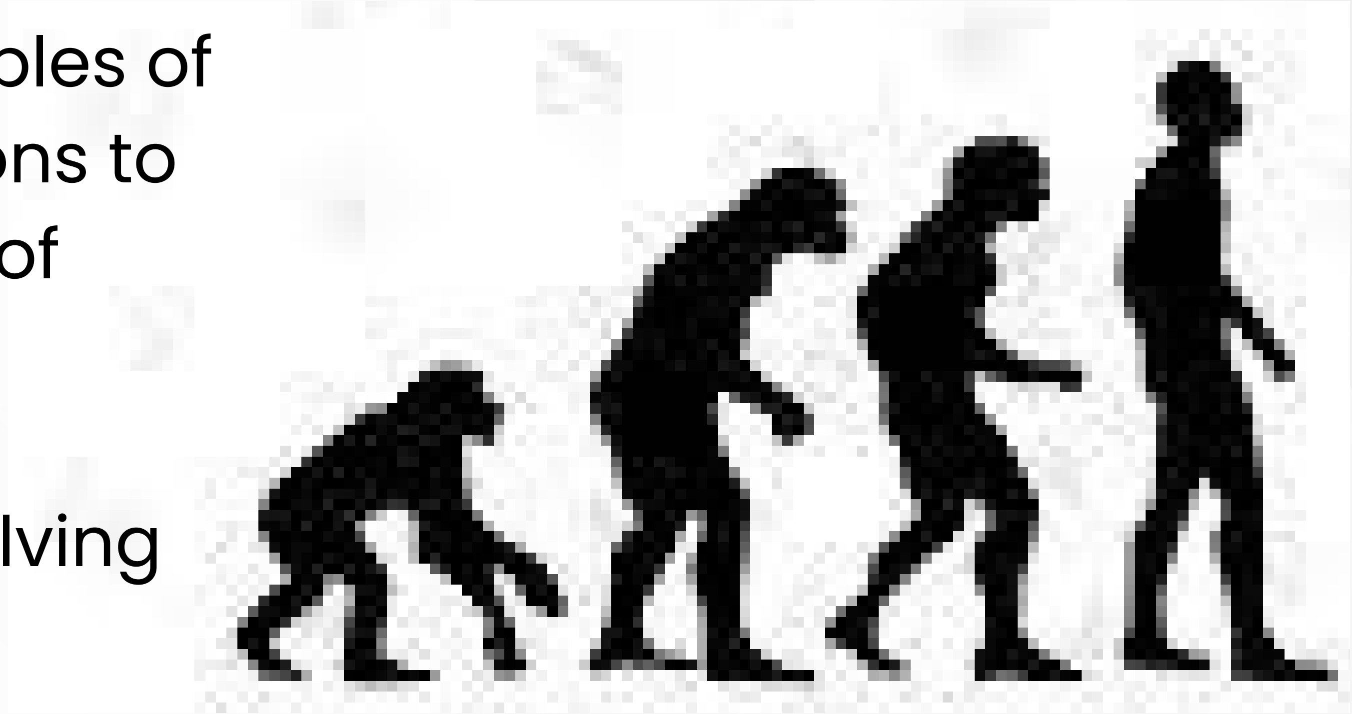
- **Start:** The strategy started on **2021-04-09** at **08:30:00**.
- **End:** It ended on **2022-01-31** at **05:30:00**.
- **Return :** The return percentage of the strategy is **0.40339 %**.
- **Max. Drawdown :** The maximum drawdown is **-12.099309%**.
- **# Trades:** The number of trades executed is **25**.
- **Win Rate:** The Win Rate is **40%.**

OPTIMIZATION

USING MOEA

WHAT ARE
EVOLUTIONARY ALGORITHMS?

Evolutionary algorithms (EAs) are a family of optimization algorithms inspired by the principles of natural evolution. They are used to find solutions to complex problems by mimicking the process of natural selection and genetic variation.

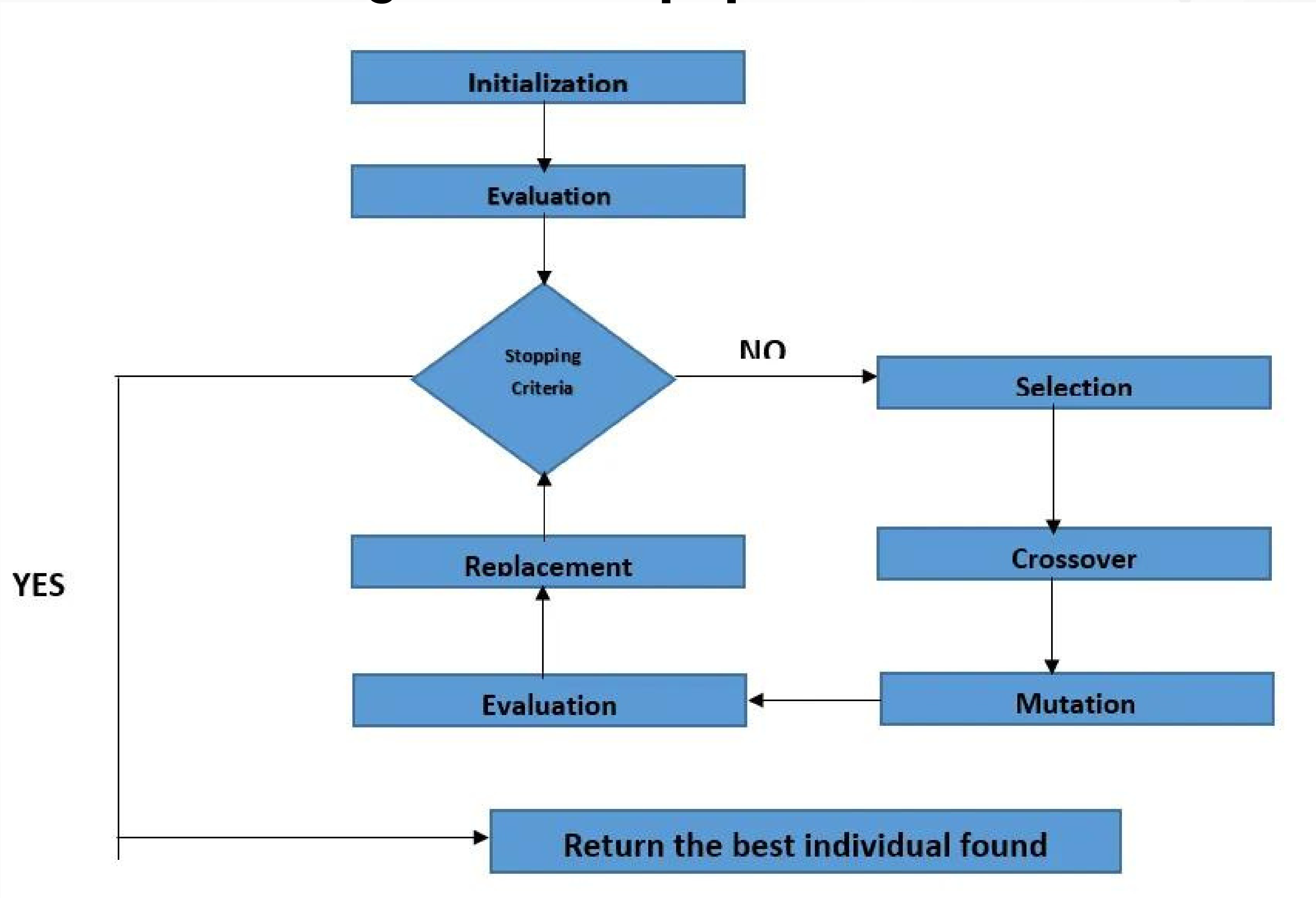


These algorithms are particularly useful for solving optimization and search problems where the solution space is large, complex, and not well understood.

ALGORITHMIC IMPLEMENTATION

GENETIC ALGORITHMS

Genetic Algorithms follow a pre-defined flow. A random population is initialized. Then, in each subsequent generation, the mating pool is selected, then crossover is performed to generate offspring population, followed by mutation to provide randomness to the model. This is followed by an elitist survival stage to get the new generation population.



Multi-objective Evolutionary Algorithms

Multi-objective evolutionary algorithms (MOEAs) are any of the paradigms of evolutionary computing (e.g., genetic algorithms, evolutionary strategies, etc.) used to solve problems requiring optimization of two or more potentially conflicting objectives, without resorting to the reduction of the objectives to a single objective by the means of a weighted sum.

OBJECTIVES

1. Maximize Win Rate
2. Maximize Returns
3. Maximize Sharpe
4. Maximize Number of Trades
5. Minimize Drawdowns

VARIABLES

1. Lookback of slower EMA
2. Lookback for faster EMA
3. Lookback for Signal line

VARIABLE CONSTRAINTS

Lookback within 1 and 90

SELECTION ALGORTIHMS

1. NSGA-II
2. SPEA-II
3. Best Selection

CROSSOVER AND MUTATION

1. Crossover done using Simulated Binary Crossover (SBX) Operator
2. Crossover probability given as 0.9
3. Mutation done using Polynomial Mutation Operator
4. Mutation probability set to (1/length of population)

ALGORITHMIC IMPLEMENTATION

01

Generate an initial random population

02

Select the mating pool based on fitness. NSGA-II and SPEA-II selectors used

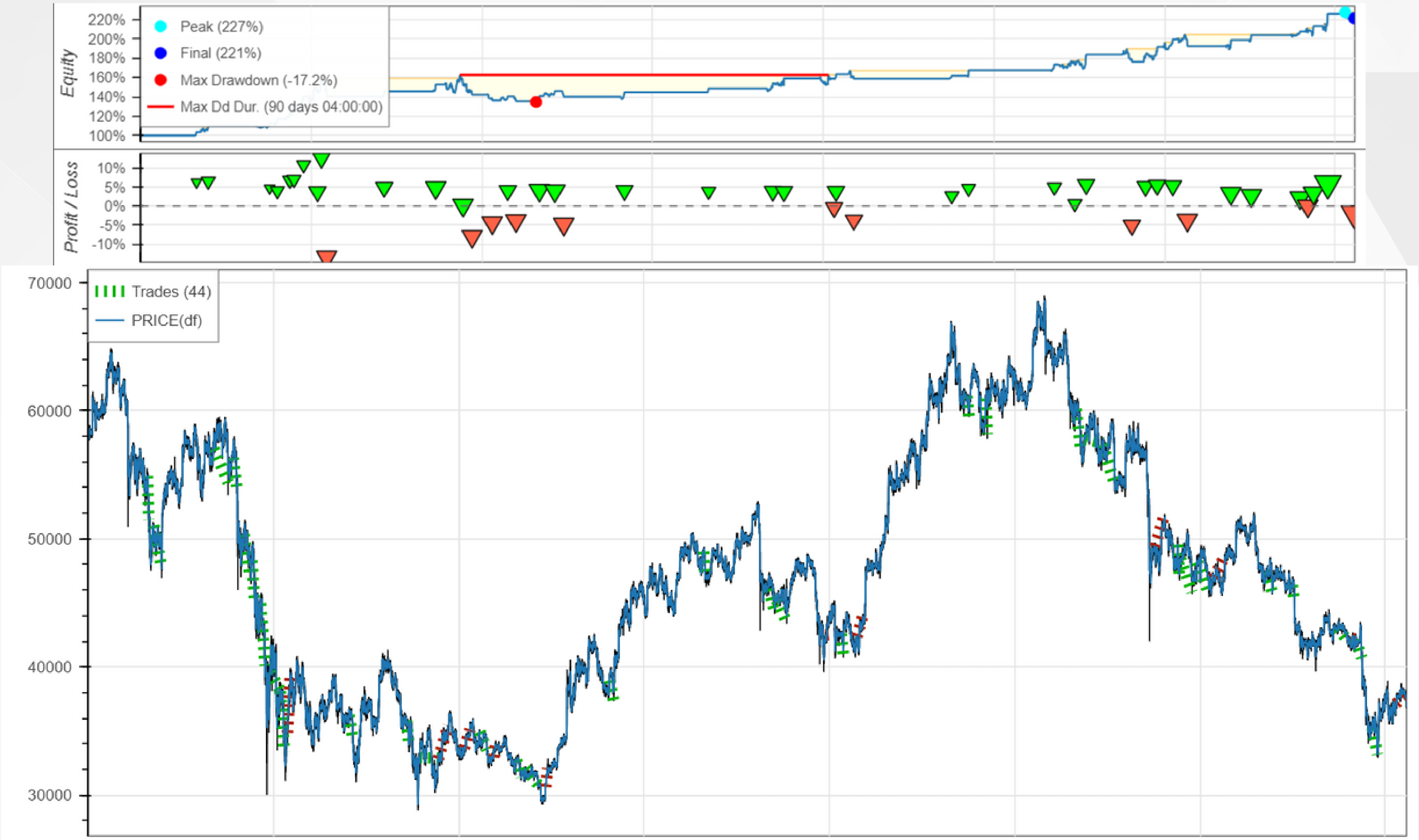
03

Perform Crossover and Mutation on the solutions

04

Use an Elitist selection algorithm for survivor

Final Strategy and PnL chart



- Use **Random Forest Classifier** to predict whether the price will go up or go down in the next tick
- Use Moving Average Convergence Divergence crossover to verify trades
- Go long if **MACD** line crosses over the signal line and the classifier predicts that the price would go up in the next tick
- Go short if **MACD** line crosses below the signal line and the classifier predicts that the price would go down

RESULTS

OBTAINED

- **Start:** The strategy started on **2021-04-09** at **08:30:00**.
- **End:** It ended on **2022-01-31** at **05:30:00**.
- **Return :** The return percentage of the strategy is **120.997826%**.
- **Volatility (Ann.):** The annualized volatility is **94.008247%.**
- **Sharpe Ratio:** The Sharpe ratio is **1.745589**.
- **Sortino Ratio:** The Sortino ratio is **8.692681**.
- **Calmar Ratio:** The Calmar ratio is **9.52136.**
- **Max. Drawdown :** The maximum drawdown is **-17.234912%**.
- **# Trades:** The number of trades executed is **44**.
- **Win Rate:** The Win Rate is 75**%.**

SUGGESTED

IMPROVEMENTS

1. **Enhance Model Accuracy:** Investigate advanced algorithms or fine-tune existing models to improve forecasting accuracy.
2. **Optimize Risk Management:** Explore advanced risk management techniques to further minimize risks and handle market volatility.
3. **Back-Testing with Varied Market Conditions:** Extend back-testing to include a wider range of market conditions, including extreme scenarios.
4. **Enhanced Feature Engineering:** Experiment with advanced feature engineering techniques to better capture market dynamics and improve predictive accuracy.

THANK YOU