InterIIT Tech Meet 12.0

Final Documentation

Team ID: 80

ZELTALABS - CRYPTO TRADING CHALLENGE

Contents

Abstract

This report presents a comprehensive analysis of an algorithmic trading strategy developed through a combination of price action analysis and a suite of market strategies, including the Relative Strength Index (RSI), Relative Vigour Index (RVI), Supertrend, Moving Average Convergence Divergence (MACD), and Weighted Bar Strength. The approach is designed to take advantage of market inefficiencies and trends, and to maximize performance in the BTC/USDT market, a thorough backtesting process was used to pick the combination of strategies. The report elaborates upon the reasoning behind each strategy's inclusion and clarifies how each one contributes to the strategy as a whole. We have used hypothesis testing to confirm the durability of our methodology, guaranteeing statistical significance in a variety of historical scenarios. Our approach places a strong emphasis on combining dynamic risk management strategies with different kinds of stop-loss mechanisms to reduce losses and safeguard money. The objective of adaptive risk management is to efficiently handle shifting market conditions and reduce negative risks. The strategy delivers high returns with minimal drawdown, showcasing its resilience in varying market scenarios of the BTC/USDT market.

Market Insights

- **Insight 1: Comparing BTC-USD and BTC-USDT**

  At the heart of cryptocurrency market stability is Tether (USDT), a stablecoin tethered to the stability of the US Dollar. Unlike its volatile counterparts, **Tether maintains a steady 1:1 value with the USD**, providing a semblance of stability in the highly dynamic cryptocurrency space. Trading in the BTC-USDT market diverges from the BTC-USD counterpart due to the stabilizing influence of Tether. The BTC-USDT market exhibits **reduced volatility** as compared to the BTC-USD market, offering a **smoother yet dynamic ebb and flow**. This characteristic attracts both cautious traders seeking relative stability and those leveraging Tether's liquidity for shorter-term moves. While Tether's influence tempers volatility, it does not eradicate the inherent complexities and risks of trading Bitcoin as we shall see in the next point. BTC/USDT is still far more volatile than ordinary stocks. Supply, demand, and market sentiment continue to play pivotal roles. Moreover, Tether's controversies and transparency concerns introduce an additional layer of uncertainty

- **Insight 2: Nevertheless BTC/USDT is a highly volatile market**

  The BTC/USDT market is **highly volatile as compared to the regular stock market**. We defined the coefficient of variation to compare the volatility between the stock market and the BTC/USDT market. We have defined the coefficient of volatility as:

$$\text{Coefficient of Volatility} = 100 \times \frac{\text{Standard Deviation}}{\text{Mean of Prices}}$$

  Division by mean is done to make the comparison of volatility independent of the magnitude of the prices.

  Comparing the relative coefficient of variation for BTC/USDT with Apple stocks shows a large difference in the volatility of the two

```python
import pandas as pd

# Download historical data
btc_data = pd.read_csv('/content/daily_data.csv')

# Extract the 'close' prices
btc_prices = btc_data['close']

# Calculate the mean and standard deviation
mean_price = btc_prices.mean()
std_deviation = btc_prices.std()

# Calculate the coefficient of variation (CV)
cv = (std_deviation / mean_price) * 100  # Multiply by 100 to express as a percentage

print(f"Coefficient of Variation for Bitcoin Prices: {cv:.2f}%")
```

```
Coefficient of Variation for Bitcoin Prices: 96.18%
```

```python
import yfinance as yf
import pandas as pd

# Define the ticker symbol for Bitcoin (BTC-USD)
ticker_symbol = "AAPL"

# Download historical data

df=yf.download('AAPL',start='2018-01-01',end='2022-01-13')
# Extract the 'Close' prices
stock_prices = df['Close']

# Calculate the mean and standard deviation
mean_price = stock_prices.mean()
std_deviation = stock_prices.std()

# Calculate the coefficient of variation (CV)
cv = (std_deviation / mean_price) * 100  # Multiply by 100 to express as a percentage

print(f"Coefficient of Variation for  Apple Stock: {cv:.2f}%")
```

```
Coefficient of Variation for  Apple Stock: 48.50%
```

- **Insight 3: BTC/USDT is a trend following market**

  We found that the BTC/USDT market is a **trend-following one**. To identify this, we have used the Hurst Exponent. The Hurst value is used as a measure of the long-term memory of the time series. A Hurst value greater than 0.5 translates to a trending market, a Hurst value less than 0.5 indicates a sideways market and a Hurst value of 0.5 indicates would indicate a random walk or a market where prediction of the future based on past data is not possible.

```python
import pandas as pd
import numpy as np
from hurst import compute_Hc

# Read the CSV file
data = pd.read_csv('data.csv')

# Convert the 'close' column to numeric (assuming 'close' is the relevant column)
data['close'] = pd.to_numeric(data['close'], errors='coerce')

# Drop rows with NaN values
data = data.dropna()

# Calculate the Hurst exponent
H, c, data_range = compute_Hc(data['close'], kind='price', simplified=True)

# Print the Hurst exponent
print("Hurst Exponent:", H)
```

```
Hurst Exponent: 0.869786186748709
```

- **Insight 4: Shorter timeframes offer more volatility and hence are difficult to predict**

  We tried to implement a variety of strategies involving scalping on shorter time frames to secure small profits using take profit and stop losses through intraday trading. However, we were unable to beat the benchmark returns without significant risk due to multiple reasons.

  As mentioned earlier, BTC/USDT is a highly volatile market. Hence, prices can fluctuate dramatically, even within minutes. This means even seemingly well-executed trades can turn against you quickly, leading to unexpected losses. News announcements, social media trends, or technical glitches can trigger these swings, making it difficult to predict future price movements. The volatility can work both ways, offering the potential for rapid gains as well. However, the **risk of large losses outweighs the potential for quick profits**. Even a small percentage swing against your position can be significant due to the volatile nature of the market, leading to increased drawdowns. Identifying optimal entry and exit points becomes challenging due to the unpredictable nature of price movements. Since we wanted our strategy **to minimize risks and have low drawdowns throughout, we finally chose the daily time frame**. Another challenge we faced was the transaction charges levied due to the large number of trades. **Hence, we believe trading on shorter time frames is riskier and less efficient**.

- **Insight 5 : Importance of Price Action Strategies:**

  Price action strategies hold paramount importance in trading for several compelling reasons:

  – Real-Time Insights:

    Price action strategies **provide traders with immediate and up-to-date information**. This real-time data reduces the time lag associated with traditional technical indicators, enabling traders to make timely and informed decisions based on the current market conditions.

  – Market Sentiment Reflection:

    Price action is a **direct reflection of market sentiment and participant behaviour**. Analyzing price charts allows traders to gauge the prevailing mood in the market. This insight into sentiment is crucial for anticipating potential market movements and adjusting strategies accordingly.

  – Supply and Demand Dynamics:

    Price action strategies offer a **nuanced understanding of supply and demand dynamics**. By closely examining price charts, traders can identify key levels, trend reversals, and areas of interest. This knowledge helps in making more informed decisions about entry and exit points, as well as understanding the potential for trend continuation or reversal.

- Simplicity and Clarity:

  Price action is presented unambiguously on charts. This simplicity makes it **accessible to traders of all experience levels**. Unlike complex indicators, price action allows for a straightforward interpretation of market dynamics, reducing the risk of misinterpretation.

- Risk Management:

  Understanding price action aids in **effective risk management**. Traders can identify potential support and resistance levels, helping them set appropriate stop-loss and take-profit orders. This proactive risk management approach is essential for preserving capital and enhancing overall trading success.

- **Insight 6: Drawdown and dip are the main enemy**

  During the exploration of this problem statement, a notable insight emerged. In the midst of the heightened volatility within the BTC-USDT market, numerous strategies demonstrated the capability to outperform the benchmark and yield favorable returns. However, the central challenge lay in confronting the inevitable drawdowns and dips.

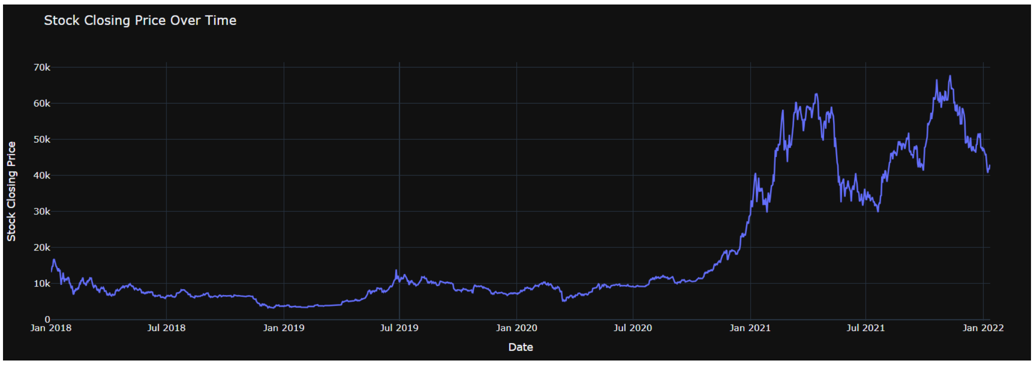
  The hallmark of an efficient and robust strategy lies not only in its capacity to generate returns but, more critically, in its ability to achieve this with minimal risk. Navigating the turbulent waters of a volatile market requires a delicate balance, where the quest for profitability is harmonized with a strategic approach to mitigating potential downsides.

  Therefore, the primary adversaries in this endeavor proved to be the drawdowns and dips. Overcoming these challenges demanded a considerable amount of effort and strategic finesse. The battle for sustained success in the volatile market landscape centered around effectively addressing and minimizing the impact of drawdowns and dips – a task that required diligence and a well-thought-out approach.

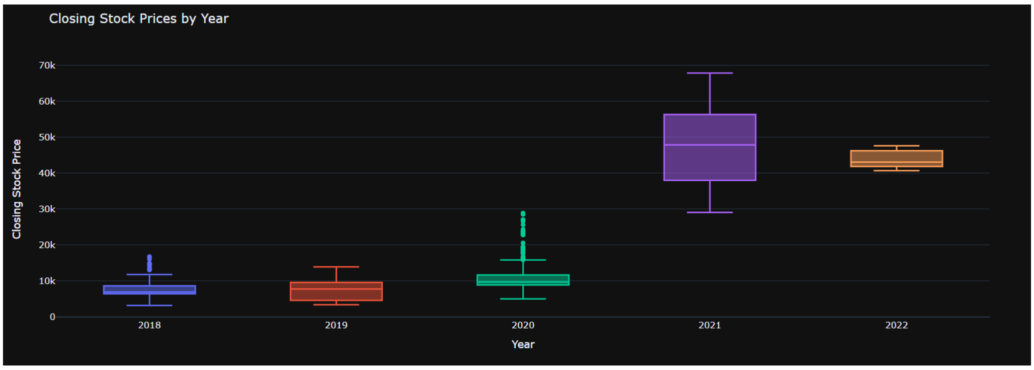
Analysing Historical Data

Just like a cricketer studies the pitch before stepping up to bat, a strategist must familiarize themselves with the market to formulate a successful strategy. The following methods were instrumental in enhancing our understanding of the market before moving forward with strategy development.

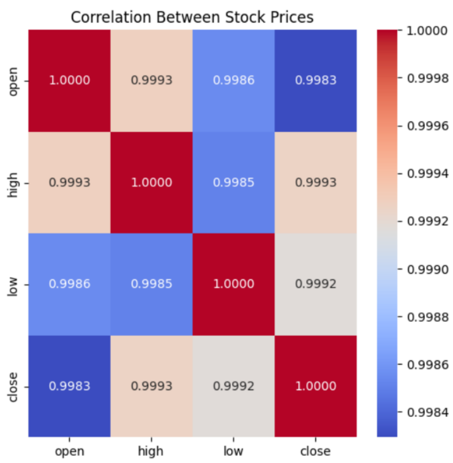
- **Data Visualization**
  - Closing Values



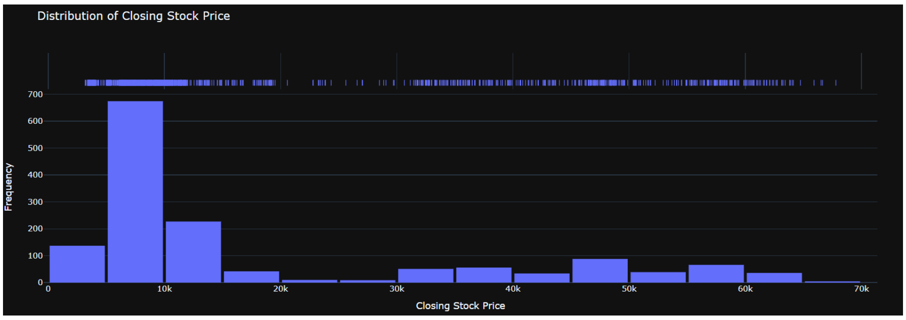
  - Closing prices per year



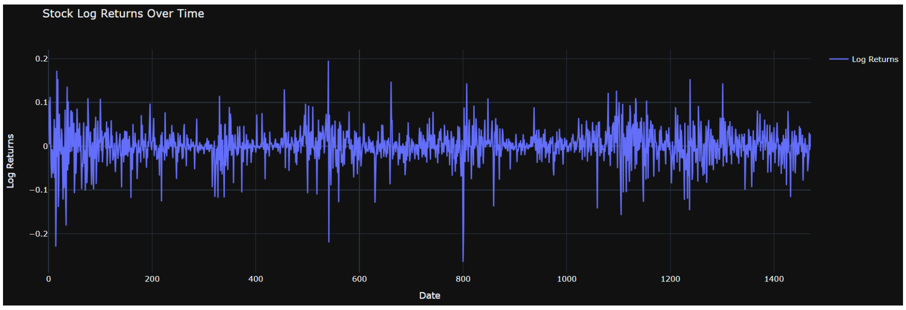
  - Correlation between OHLC prices



  - Distribution of closing stock price

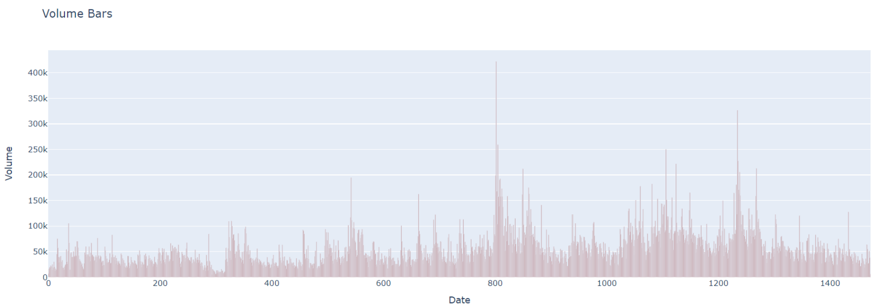


– Line chart of log of daily returns



– Candlestick plot with volume bars

– Plot of Volume Bars



- **Statistical Moment Analysis**

Statistical moment analysis is a mathematical and statistical technique used to quantify the shape and characteristics of a probability distribution. Moments provide numerical measures that summarize various aspects of the distribution, helping to describe its central tendency, spread, and shape.

The primary purposes of statistical moment analysis include:

– Describing the Distribution:

  Moments, such as the **mean** (first moment) and **variance** (second moment), provide **quantitative information about the central tendency and variability of a distribution**.

– Characterizing Shape:

  Higher-order moments (**skewness** and **kurtosis**) offer **insights into the asymmetry and tail behavior of a distribution**. Skewness measures the degree and direction of skew (departure from symmetry), while kurtosis measures the shape and heaviness of the tails.

– Risk and Uncertainty Assessment:

  In finance and risk analysis, **moments are used to assess the risk and uncertainty associated with a set of financial returns**. Skewness and kurtosis, for example, provide information about the potential for extreme events.

– Data Preprocessing:

  In data preprocessing, statistical moment analysis, such as computing mean, standard deviation, skewness, and kurtosis, helps in understanding the distributional properties of the data. This information aids in detecting outliers, guiding the development of robust models and ensuring that the data is properly scaled and centered for more effective analysis.

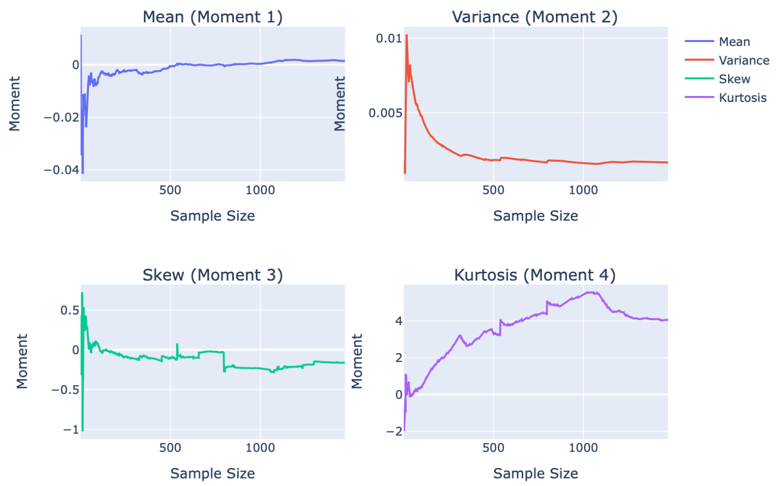
– Estimating Volatility:

  In estimating volatility, statistical moment analysis, particularly the second moment (variance or standard deviation), is crucial. By calculating the historical volatility of a financial time series using standard deviation, you gain insights into the level of variation or dispersion in the data. This information is valuable for estimating future volatility and is commonly used in financial modeling and risk management.

– Plots:

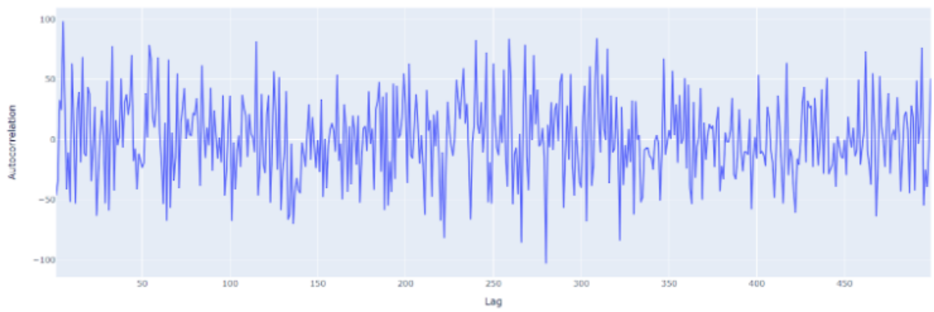
  Thus, Statistical Moment Analysis plays a pivotal role in the examination of historical data. The accompanying graphs illustrate the four key moments: Mean, Variance, Skewness, and Kurtosis.

.



- **Augmented Dickey-Fuller (ADF) Test**
  - The Augmented Dickey-Fuller (ADF) test is a statistical test used to assess whether a given time series data has a unit root, indicating it is non-stationary. Stationarity is a crucial assumption in many time series models. The ADF test helps determine if differencing the series (making it stationary) is necessary. The null hypothesis of the ADF test is that a unit root is present, implying non-stationarity. A rejection of the null hypothesis suggests that the time series is stationary.
  - Initially we applied the ADF test on the closing values of the daily data which gave a p-value of 0.805(greater than 0.05) making the closing values nonstationary.
  - However on applying the ADF test on the consecutive differences of the closing price we got a p-value of 0.0000(less than 0.05) making it stationary.

- **Autocorrelation Function**
  - Autocorrelation is a measure of the correlation between a time series and its own lagged values. Autocorrelation tests, such as the autocorrelation function (ACF) or the Durbin-Watson statistic, help assess whether there is a pattern or correlation between observations at different time points. If autocorrelation is present, it may indicate that the time series is not entirely random and may exhibit some degree of predictability. Identifying autocorrelation is crucial for time series analysis, as it can impact the performance of forecasting models. The Durbin-Watson statistic, for instance, tests for first-order autocorrelation in the residuals of a regression model.
  - The below shows a plot of autocorrelation values against the Lags.



Strategy Developement

- **Usage of Daily Data in our final strategy**

  We decided to formulate our trading method using daily data after conducting thorough research of the BTC-USDT market over several time frames, from three to sixty minutes. As we moved to shorter time frames, our observations showed that market unpredictability was increasing. This increased volatility had a significant effect on transaction profitability, which was mostly caused by the increased frequency of trades. Furthermore, the additional transaction expenses that came along with more frequent trades greatly reduced the possible earnings. Consequently, we decided to concentrate on using daily data to lessen these difficulties and create a more reliable and practical trading strategy.

- **Types of Strategies**

  Initially, we experimented with a range of different trading approaches, including Trend Following, Mean Reversion, Volatility-Based, Price Action, and Volume-Based strategies. Every strategy type was carefully examined and studied to understand its effectiveness, advantages, and disadvantages in the constantly shifting market environment. This thorough investigation sought to discover the strategy that best fits the market's natural dynamics while attempting to produce reliable consistent trading results.

- **Moving Average CrossOvers**

  Firstly we started by using Moving Average Crossovers that first showed promising returns. But it presented a big issue due to its innate lagging. This trailing characteristic frequently caused delayed reactions, which reduced its efficacy—particularly in sideways markets. Because of the high frequency of crosses, the strategy was unable to operate at its best under these circumstances, which resulted in erroneous interpretations and a rise in drawdowns. This restriction made us think about how to improve the strategy's accuracy and adaptability in a variety of market scenarios to overcome its inefficiency in sideways markets.

- **Mean Reverting Techniques**

  We then looked into mean-reverting techniques like Bollinger Bands, Stochastic oscillator, and RSI. But most of the time, these tactics produced erroneous trade signals. Following a series of tests and delving further into the market data, comprehensive visualizations of the trading view revealed an important conclusion: the market followed trends instead of returning to mean values.

  As a result, we reorganized our strategy and used these indicators to match the direction of the market's development. For example, we took advantage of purchasing chances when the price of BTC-USDT signified overbought situations and we started selling when it signaled oversold ones. This modified strategy showed encouraging outcomes, especially when it came to RSI's higher returns as compared to earlier approaches.

- **Volume Based Indicator**

  We studied volume-based techniques and looked at many indicators, including the money flow index, On Balance Volume (OBV), and the Positive and Negative Volume Index. One important finding from our research was that volume-based methods by themselves were not very successful when it came to trading cryptocurrencies.

  The information showed that it was not possible to consistently turn a profit when making decisions only based on volume-based indications. We have tried many strategies and all ended up having less than half winning trades, Though they had potential, these techniques didn't match the volatile character of the cryptocurrency market well, which made them less reliable on their own. This insight encouraged us to investigate more thorough and integrated methods in our search for a robust trading strategy.

- **Volatility based strategies**

  We then shifted our attention to volatility-based strategies, presenting Supertrend, Volty Expansion, Keltner channels, and a variety of other techniques. We sought to take advantage of volatility, which we saw as a unique feature that set the BTC-USDT market apart from traditional stock markets. These tactics showed great promise in terms of performance, providing remarkable returns with comparatively smaller drawdowns than our earlier efforts.

  One noteworthy finding emerged: these techniques demonstrated an impressive capacity to recognize market patterns throughout a range of periods. This finding was critical since it demonstrated the methods' flexibility and dependability in routinely identifying market trends. Furthermore, we saw how these volatility-based techniques could increase entry and exit locations while enhancing and complementing the effectiveness of other strategies.

- **Price Action Strategies**

  We have implemented several price action strategies - IRB, IBS, RVI, pin bar, Sequence of highs and lows, etc. Candlesticks exhibit distinct characteristics that function as a miniature representation of market dynamics, providing intricate insights into fluctuations in price.

- **The Path forward**

  While implementing strategies we implemented them in 3 major fashions:

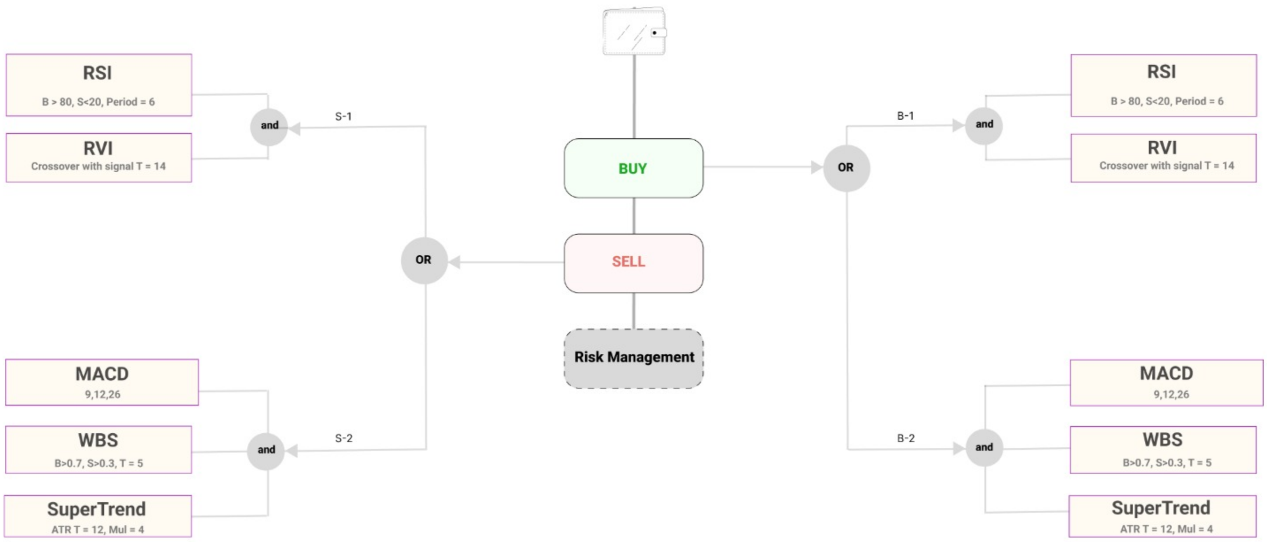
  - *Implementing The strategy independently*
  - *Implementing the strategy with other already implemented strategies*
  - *Implementing the strategy in an alternative fashion*.

    The alternative implementation of the strategy involved a sequential approach where we initially executed trades based on one specific strategy. Subsequently, we conducted an evaluation to determine if the application of another strategy would result in better performance. If the alternative strategy demonstrated improved outcomes during the testing period, we transitioned to its implementation. This method allowed for the systematic exploration and comparison of multiple strategies, enabling a comprehensive assessment of their effectiveness over time.

    Moreover, even when focusing on the implementation of a single strategy at a given time, the design of this approach facilitated the concurrent testing of multiple strategies within the overarching backtesting period. This adaptive framework ensured that we could explore and adapt to various market conditions, enhancing the overall robustness of our trading approach. The seamless transition between strategies based on performance evaluations allowed for continuous refinement and optimization of our trading methodology.

  All these tactics performed admirably together, particularly when applied in concert. These methods were able to regularly detect trends with higher likelihoods, which led us to base our strategy on them. These tactics demonstrated robustness and reliability in a variety of market scenarios, which prompted a more thorough ideation process to fully utilize their advantages.

Strategy Logic



In implementing our trading strategies, we employ two distinct approaches for market entry and exit. The first approach integrates the Relative Strength Index (RSI) and Relative Vigor Index (RVI) to effectively navigate continuous market swings. RSI, with its sensitivity to overbought and oversold conditions, coupled with RVI, providing insights into the vigor of the trend, collaboratively enhances our ability to capitalize on sustained market movements. The second approach involves the synergistic use of Moving Average Convergence Divergence (MACD), Super Trend, and Weighted Bar Strength (WBS). This strategy is applied to identify and confirm shifts in market swings. MACD and Super Trend offer trend direction insights, while WBS acts as a confirming metric. The utilization of these approaches is contingent upon prevailing market conditions, with the first approach excelling in continuous swings and the second strategically deployed for detecting changes in market dynamics. This dual-strategy framework aims to optimize our market entry and exit decisions, providing a nuanced and adaptable approach to navigate diverse market scenarios.

- **Rejection of Reverting Strategies and using Trend following strategies**

  Through statistical testing, it has been established that the prevailing market dynamics exhibit a trend-following nature. In contrast, the default settings of mean reversion strategies yielded unfavorable results. This empirical evidence underscores the importance of aligning trading strategies with the predominant trend in the market, emphasizing the inadequacy of mean reversion indicators in such a trend-following context. The formalization of this finding underscores the need for adaptive strategies that account for and capitalize on the underlying trend dynamics to enhance the efficacy of technical indicators in optimizing trading performance. Statistical tests which help us in coming to this outcome.

  – Autocorrelation function
  – Binomial testing
  – T - statistics
  – ADF test

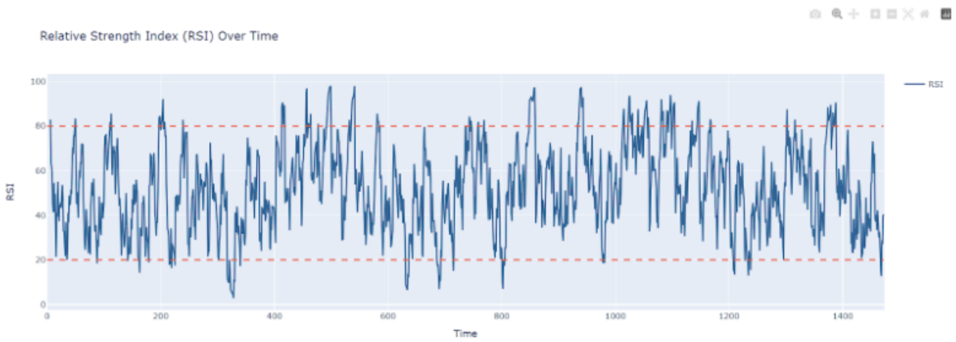
- **Relative Strength Index(RSI)**

  The relative strength index compares a security's strength on days when prices go up to its strength on days when prices go down.

$$\text{RSI} = 100 - \frac{100}{1 + \frac{(\text{Previous Average Loss} \times 13) + \text{Current Loss}}{(\text{Previous Average Gain} \times 13) + \text{Current Gain}}}$$

  Through a comprehensive analysis of (Relative Strength Index) RSI values during notable price movements, it is evident that RSI readings surpassing 80 consistently correspond to uptrend conditions, while RSI values below 20 consistently align with downtrends. This observation is substantiated by statistical measures, including time series momentum and standard deviation, which contribute to the rationale that extreme RSI values are indicative of sustained trends in trend-following markets. The chosen thresholds (buy above 80 and sell below 20) are thus deemed strategically appropriate for identifying favorable entry and exit points. The formalization of this analysis is reinforced by backtesting the strategy across diverse market conditions, providing empirical evidence of its efficacy in capturing significant price reversals and supporting its application as a reliable indicator for trend identification in dynamic market environments.

  The choice of **6 days** for calculating the Relative Strength Index (RSI) is informed by the specific characteristics of a volatile and fluctuating market. In such environments, where short-term trends may foreshadow longer-term movements, a shorter RSI period is strategically advantageous. A 6-day RSI is more reactive to recent price changes, allowing it to capture and signal early indications of shifts in market sentiment. This heightened reactivity aligns with the dynamic nature of the market, enabling the identification of shorter trends that may serve as precursors to more significant and enduring trends. Consequently, the 6-day RSI period enhances the sensitivity of the analysis, providing a nuanced understanding of evolving market dynamics and reinforcing its role as a valuable tool for trend identification in the context of a volatile market.



- **Supertrend using Average True Range(ATR)**

  - *TrueandAverageTrueRange*

    The true range of the day is taken as the greatest of the following: current high less the current low; the absolute value of the current high less the previous close; and the absolute value of the current low less the previous close. The Average True Range is a moving average, using 12 days, of the true ranges.

- $SuperTrendIndicator$

  The indicator combines the average true range (ATR) with a multiplier to calculate its value. This value is then added to or subtracted from the asset's closing price to plot the supertrend line. Supertrend can help identify trends, manage risk, and confirm market tendencies.The SuperTrend is calculated using the following steps:

  * **Calculate the Average of High and Low Prices:**

  $$\text{High-Low Average (HL Avg)} = \frac{\text{High} + \text{Low}}{2}$$

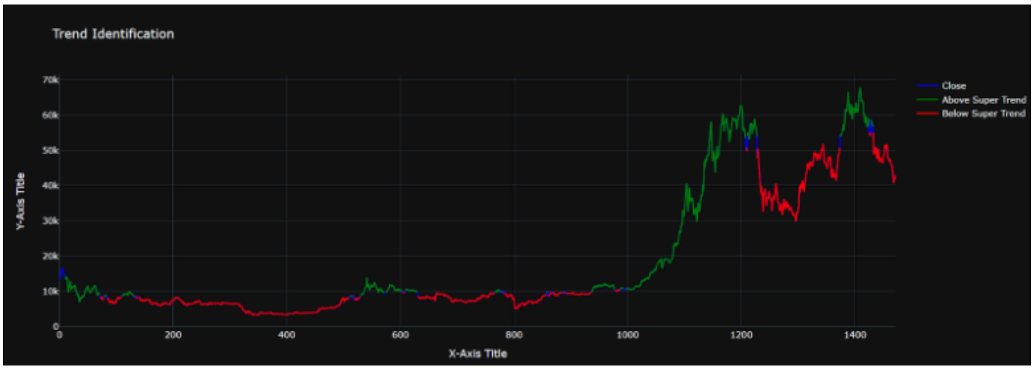
  * **Calculate the Upper and Lower Bands of the SuperTrend:**

  $$\text{Upper Band} = \text{HL Avg} + \text{Multiplier} \times \text{ATR}$$

  $$\text{Lower Band} = \text{HL Avg} - \text{Multiplier} \times \text{ATR}$$

  * The SuperTrend indicator uses these bands to determine the trend direction. It is commonly used in technical analysis to identify trend changes and potential entry or exit points.

In the context of identifying trends in a volatile market, employing a 12-period rolling calculation with a 4-multiplier based on the Average True Range (ATR) in the Super Trend indicator has proven to be a strategically effective approach. The choice of a 12-period rolling window and a 4-multiplier is informed by statistical analysis, specifically tailored for volatile market conditions. This configuration enables the Super Trend indicator to dynamically adapt to the market's inherent volatility, providing a responsive measure of trend identification. Backtesting results further support the efficacy of this parameter setting, substantiating its ability to accurately capture and delineate trends amidst market volatility. The formalization of this methodology underscores its empirical foundation, contributing to a robust framework for trend identification in dynamic and unpredictable market environments.



- **Relative Vigor Index(RVI)**

  The Relative Vigor Index (RVI) is a trend-based indicator designed to assess the strength and sustainability of a price trend by measuring the deviation between opening and closing prices. The indicator's core principle involves evaluating the "vigor" or energy of the trend. The following are the steps to calculate Relative Vigor Index(RVI)

  - $Calculatethepricedifference(PD):$

  $$PD = \text{Close} - \text{Open}$$

– $\underline{Calculate the price range (PR):}$

$$PR = \text{High} - \text{Low}$$

– $\underline{Shift the price difference and price range by one, two, and three time periods:}$

$$PD_i = \text{PD}_{\text{shifted by i}}$$

$$PR_i = \text{PR}_{\text{shifted by i}}$$

$$\text{Numerator} = \frac{\text{PD} + 2 \times PD_1 + 2 \times PD_2 + PD_3}{6}$$

$$\text{Denominator} = \frac{\text{PR} + 2 \times PR_1 + 2 \times PR_2 + PR_3}{6}$$

$$\text{RVI} = \frac{\text{Numerator}_{\text{rolling sum}}}{\text{Denominator}_{\text{rolling sum}}}$$

– $\underline{Computing the Signal line}$

$$\text{Signal Line} = \frac{\text{RVI} + 2 \times \text{RVI}_{\text{shifted by 1}} + 2 \times \text{RVI}_{\text{shifted by 2}} + 2 \times \text{RVI}_{\text{shifted by 3}}}{7}$$

In the context of trend determination, when the RVI crosses above the signal line, it signifies an uptrend, prompting a buy signal from the strategy. Conversely, when the RVI crosses from above to below the signal line, it signals a downtrend, prompting a sell signal. This formalization establishes a clear framework for interpreting RVI dynamics, facilitating informed decision-making based on trend strength and potential reversals.

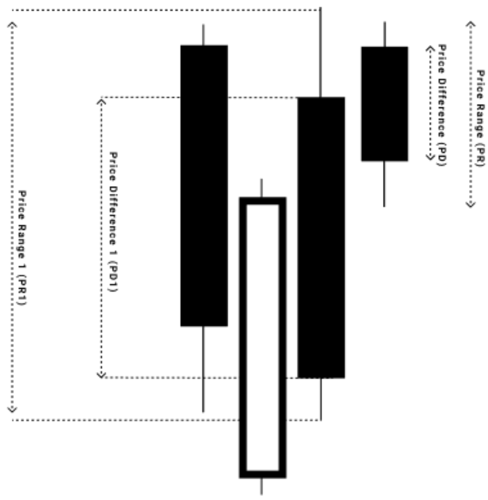


Figure: RVI

- **Weighted Bar Strength**

  In the pursuit of refining trend identification strategies in financial markets, a novel metric termed ”weighted bar strength” is introduced. This metric, defined as the ratio of,

$$\frac{(close - low)}{(high - low)}$$

undergoes an optimization process through the computation of an exponentially weighted average over a 5-day period. A buy signal is prompted when the weighted bar strength surpasses 0.7, indicating a potential uptrend, while a sell signal is triggered upon its descent below 0.3, suggesting a potential downtrend. This parameterization has undergone comprehensive statistical examination, affirming its effectiveness in trend identification within the context of a trend-following market. Backtesting results consistently demonstrate its proficiency in capturing sustained trends, highlighting the adaptability and utility of the weighted bar strength metric in fortifying trend identification strategies.

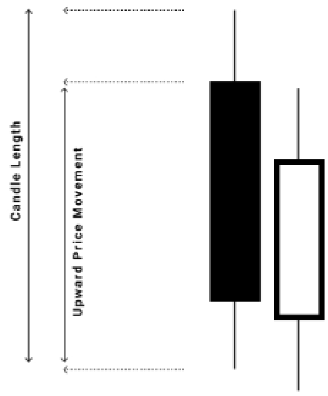
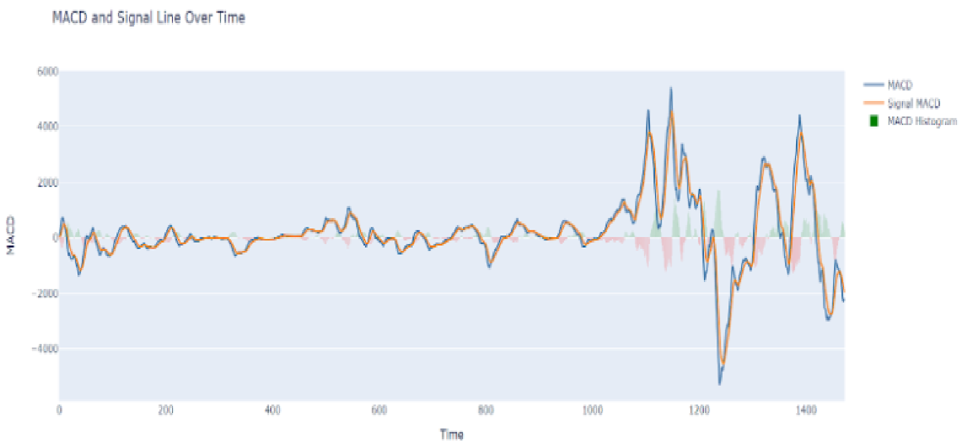


Figure: IBS

- **Moving Average Convergence Divergence(MACD)**

  In systematically confirming trends within trend-following markets, employing the Moving Average Convergence Divergence (MACD) indicator with standard parameters—specifically, a short period of 12, a long period of 26, and a signal line period of 9—has proven to be a discerning choice, substantiated through rigorous statistical analysis. This parameter configuration, aligning with the intrinsic features of trend-following markets, effectively distinguishes short-term and long-term trends and utilizes the signal line for smoothing and identifying significant crossovers. Rigorous backtesting unequivocally supports the efficacy of the MACD with these parameters in confirming and capturing sustained trends, underscoring the importance of parameter adaptability for traders navigating the dynamic terrain of trend-following markets.



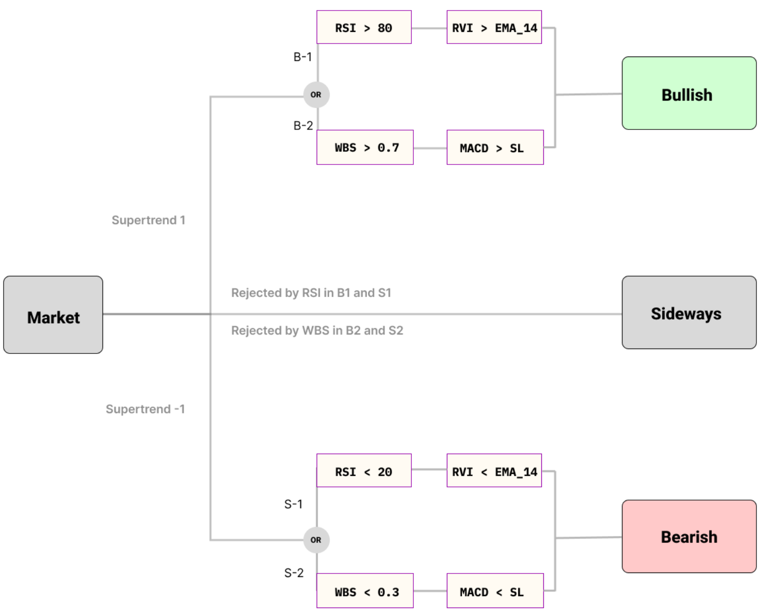
- **Generating Signals Based on Combination**

  Signals from the combination of MACD, Weighted Bar Strength, and SuperTrend is generated from the code shown below

```python
data['signals'] = 0

# Loop through the data starting from the second row
for i in range(1, len(data)):
    # Check if conditions for a bullish signal are met
    if (data['weight_avg'][i - 1] > 0.7 and
        (data.MACD[i] > data.Signal_MACD[i] and data.trend[i] == 1)):
        # Set signal to 1 for a bullish signal
        data.signals[i] = 1

    # Check if conditions for a bearish signal are met
    elif (data['weight_avg'][i - 1] < 0.3 and
          (data.MACD[i] < data.Signal_MACD[i] and data.trend[i] == -1)):
        # Set signal to -1 for a bearish signal
        data.signals[i] = -1
```

  Signals from the combination of RSI,RVI, and above mentioned combination are generated from the code shown below

```python
data['signals'] = 0

# Loop through the data starting from the second row
for i in range(1, len(data)):
    # Check if conditions for a bullish signal are met
    if (data['weight_avg'][i - 1] > 0.7 and
        (data.MACD[i] > data.Signal_MACD[i] and data.trend[i] == 1)):
        # Set signal to 1 for a bullish signal
        data.signals[i] = 1

    # Check if conditions for a bearish signal are met
    elif (data['weight_avg'][i - 1] < 0.3 and
          (data.MACD[i] < data.Signal_MACD[i] and data.trend[i] == -1)):
        # Set signal to -1 for a bearish signal
        data.signals[i] = -1
```

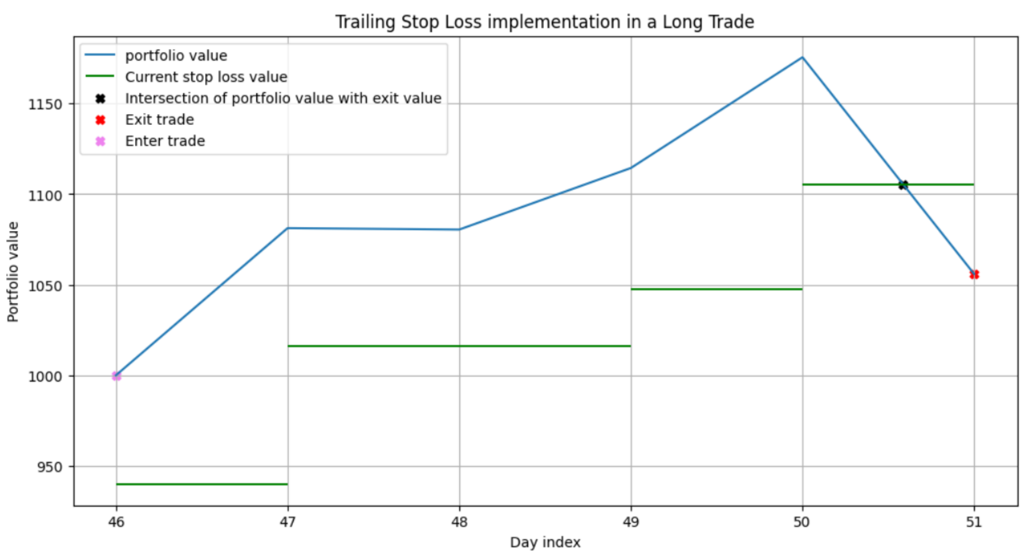


Risk Management Plan

We adopted several risk management measures to increase the robustness of our strategy across different types of markets by increasing returns and cutting down the drawdowns as well accelerating our other parameters.

- **Trailing Stop Loss Strategy(For Long Trades)**

A stop-loss is a pre-set order to sell a security when it reaches a specified price, minimizing potential losses for an investor or trader. A trailing stop loss is a dynamic risk management strategy where the stop loss level adjusts automatically based on the asset's price movement, helping to lock in profits or limit losses. The trade is closed when the market price decreases by more than a defined percent from the current high.



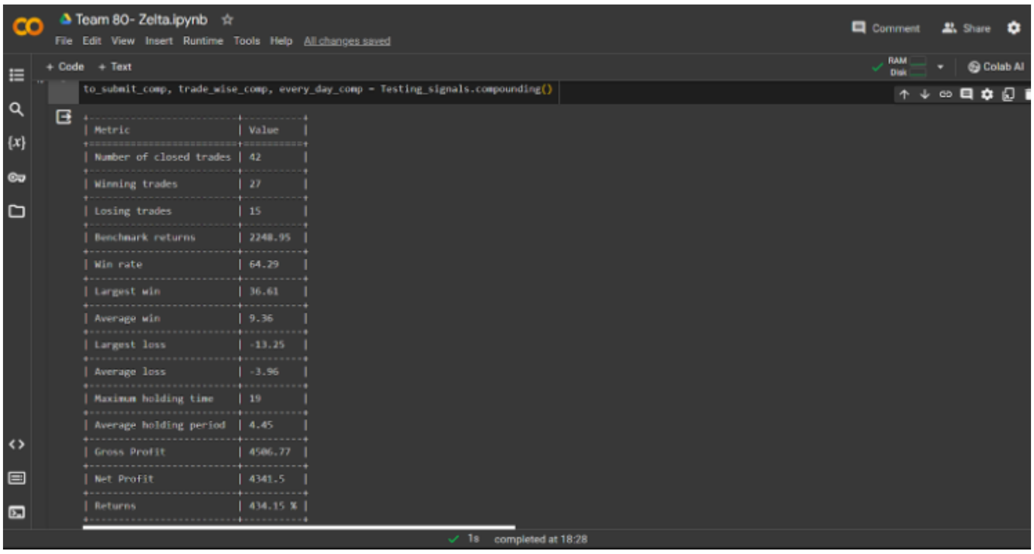
The above is an example of how the trailing stop loss is being implemented. The green line indicates the current stop-loss value. The trade started on day index 46, with the initial stop loss as 0.94 times the entry price as the stop loss percent has been set to 6 percent. The current maxima is the price on day index 46. Then as the price rises above the current maxima on day index 47 the current maxima is updated to portfolio value on day index 47 and the stop loss level is also raised. This process decreases our maximum loss in the trade. Now as the price drops slightly on day index 48, the stop loss is kept constant. The price rise continues on days 49 and 50 and hence the stop loss rises. It is on day 51 that the price level falls steadily but thanks to the trailing stop loss we were able to extract a substantial profit.

We found that an optimum trailing stop loss is extremely important to minimize losses and drawdowns, and secure profits in the highly volatile BTCUSDT markets. Through backtesting, we found that a trailing stop loss of 6 percent is optimum with our strategy throughout the history of the BTCUSDT market. A smaller stop loss such as 2 percent decreased returns while also increasing the drawdown. A similar comment could be made about larger stop losses of around ten percent.

Through meticulous backtesting, we have identified the optimal trailing stop loss for our strategy in the BTCUSDT market. The results indicate that a trailing stop loss set at six percent has consistently demonstrated its effectiveness throughout the historical data. This carefully calibrated percentage strikes a balance, allowing for profit maximization while minimizing drawdowns.

A smaller stop loss, such as 2 percent, was found to diminish returns while simultaneously increasing drawdowns. Conversely, larger stop losses, around ten percent, also exhibited unfavorable outcomes. Thus, our findings underscore the importance of an optimal trailing stop loss percentage, as it plays a pivotal role in achieving a resilient and effective risk management strategy in the dynamic BTCUSDT market.

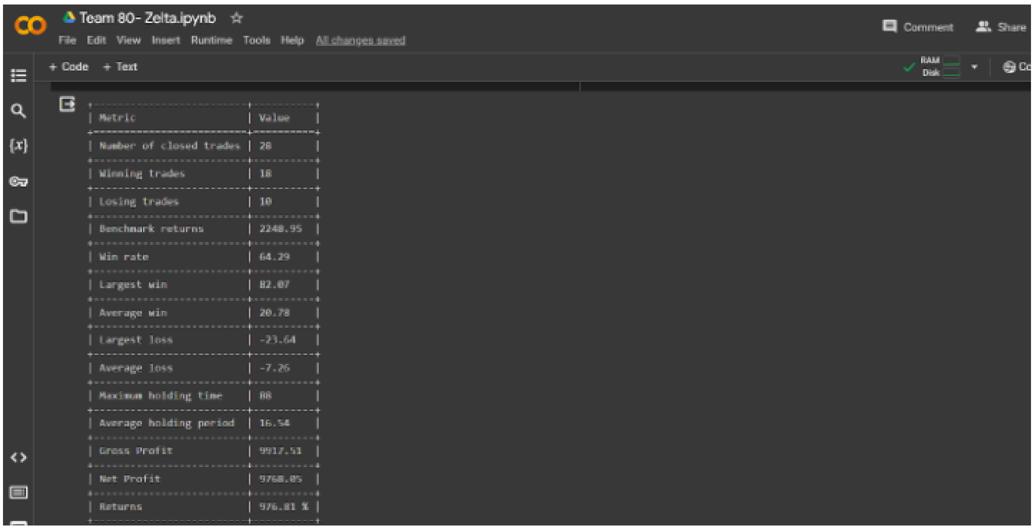
Here are the backtesting results when we took a stop loss percentage as 2 percent.


Team 80- Zelta.ipynb

```
to_submit_comp, trade_wise_comp, every_day_comp = Testing_signals.compounding()
```

Metric	Value
Number of closed trades	42
Winning trades	27
Losing trades	15
Benchmark returns	2248.95
Win rate	64.29
Largest win	36.61
Average win	9.36
Largest loss	-13.25
Average loss	-3.96
Maximum holding time	19
Average holding period	4.45
Gross Profit	4506.77
Net Profit	4341.5
Returns	434.15 %

Backtesting results were achieved through the compounding approach of taking a trailing stop loss of 10 percent.



Metric	Value
Number of closed trades	28
Winning trades	18
Losing trades	10
Benchmark returns	2248.95
Win rate	64.29
Largest win	82.07
Average win	20.78
Largest loss	-23.64
Average loss	-7.26
Maximum holding time	88
Average holding period	16.54
Gross Profit	9917.51
Net Profit	9768.05
Returns	976.81 %

These are meager returns as compared to the results obtained through our strategy of keeping a trailing stop loss of 6 percent.

- **Dyanamic Exit condition for Short Trades**

  We initially used a trailing stop loss for both long and short trades. However, we encountered challenges with short trades, where the portfolio value often decreased rapidly, triggering the stop loss too soon and reducing returns.

  To address this, we developed a dynamic exit condition for short trades. Here's how it works: Imagine the initial portfolio value is x dollars. We set an exit condition at x*(1+0.06), or the 1.06x level. If the portfolio value decreases, the exit condition adjusts proportionally. For instance, if the portfolio value becomes 0.5x dollars, the exit condition would be set at 0.5x*(1.06), or 0.53x.

  This dynamic exit condition proved advantageous by capturing rebounds in short trades, minimizing losses, and even generating profits. The goal was to avoid prolonged holding times and significant dips in portfolio value, common challenges in short trades.

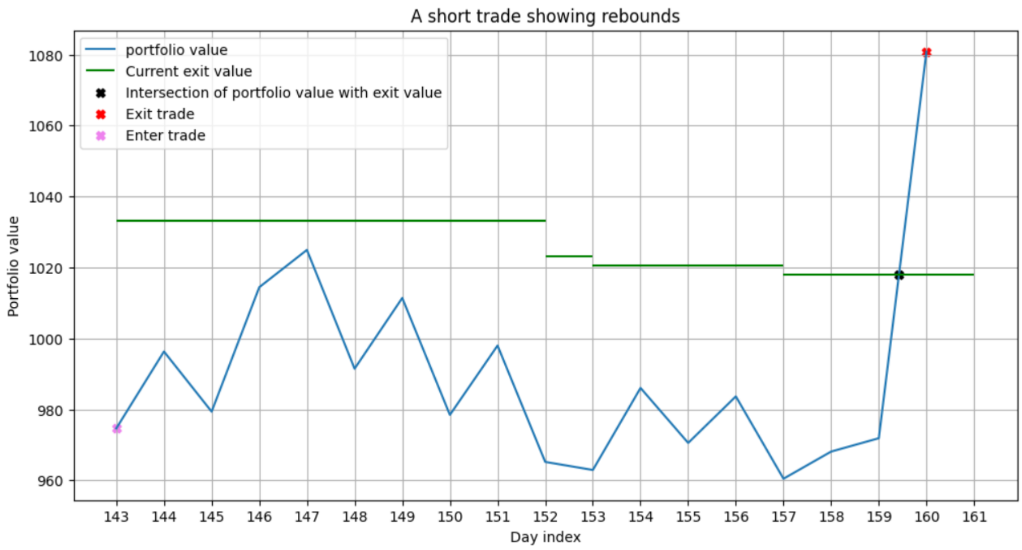
  A question arises: If short trades often result in a decrease in portfolio value, why not focus on a long-only strategy? The answer lies in the effectiveness of our dynamic exit condition, which allowed us to extract profits and minimize losses significantly. This approach also reduced the holding period compared to a long-only strategy, where extended dips and holding times are more prevalent. Consequently, we opted to implement this dynamic exit condition for its ability to enhance the performance of short trades.

  The selection of the value 0.06 was based on a comprehensive comparison across various values, similar to our approach in determining the optimal value for the previous trailing stop loss risk management condition. After evaluating a range of values, 0.06 emerged as the most optimal choice. It demonstrated effective risk management, proving to be well-suited for our strategy.

  Consider Trade Number 6 in our strategy(below is the graph), illustrating the effectiveness of our dynamic exit condition. If a standard trailing stop loss were applied, the outcome would likely have been a modest profit or even a loss. Unlike traditional approaches, our dynamic exit condition showcases its power by adapting to market conditions.

As the portfolio value rises, the exit condition remains constant. However, if the portfolio value falls below the minimum achieved so far, the exit condition is promptly updated (depicted by the green lines in the graph below). In this specific case, a substantial rebound led to a profitable outcome, yielding a return of 10.89 percent. This result contrasts with what might have occurred with a conventional stop loss, potentially turning a winning trade into a losing one.

Therefore, this risk management measure significantly contributed to optimizing and enhancing the robustness of our strategy.

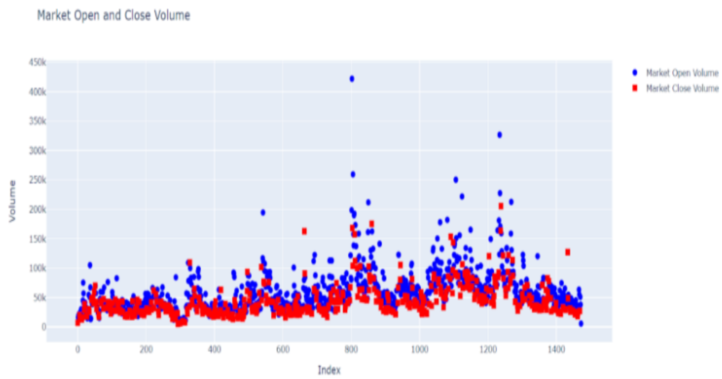


- **Market Volatility based stop loss**

  Recognizing the influence of external factors on the volatility of the BTC-USDT market, we have incorporated a dynamic adjustment to our exit condition as well as the trailing stop loss strategy. Extensive research has revealed a noteworthy correlation between heightened volatility in the BTC-USDT market and the days when global stock markets were open. In response to this insight, we have implemented a strategic modification to our trailing stop loss and the exit condition mechanism to better accommodate the increased market fluctuations during these periods.

  To capture the nuanced relationship between BTC-USDT market volatility and stock market activity, we introduced a multiplier within our framework. This multiplier serves as a dynamic factor that scales the trailing stop loss and exit condition percentage based on the prevailing volatility. Specifically, on days when stock exchanges worldwide are open, the multiplier is activated to proportionally increase the trailing stop loss and exit condition percentage.

  This adaptive approach allows our strategy to respond effectively to the dynamic market conditions associated with global stock market operations. By aligning our trailing stop loss and exit condition with the observed fluctuations during these periods, we aim to enhance risk management and optimize the balance between profit-taking and loss limitation. This nuanced adjustment reflects our commitment to staying ahead of market dynamics and underscores the importance of a flexible and data-driven approach in navigating the complexities of the BTC-USDT market.



- **Average True Range(ATR) Based Stop Loss**
  - *TrueRange*
    The true range of BTCUSDT for the day is the greatest of the following: current high less the current low; the absolute value of the current high less the previous close; and the absolute value of the current low less the previous close.
  - *AverageTrueRange*
    The Average True Range is a moving average of the true ranges.

  When volatility increases, the ATR value rises, and the stop-loss widens to accommodate larger price swings. Conversely, during periods of lower volatility, the stop-loss tightens. ATR is designed to measure market volatility. Using ATR allows us to set levels that are proportional to the current volatility, helping to account for the varying ranges of price movement.

  - **For Long Trades**
    Stop Loss price = closing price of bitcoin on the day of entry into trade- multiplier * present ATR value
  - **For Short Trades**
    Stop Loss price = closing price of bitcoin on the day of entry into trade + multiplier * present ATR value

- **Inventory Retracement Bars(IRB)**
  Inventory Retracement Bars are a certain type of candlestick that we have used as a signal to exit a long trade in an uptrend and as a signal to exit short trades in a downtrend to secure profits before the market changes direction. In an uptrend, we are looking for candlestick bars that open and close 95 percent or more off their high. In a downtrend, we are looking for candlestick bars that open and close 95 percent or more of their low. These bars work well during an uptrend or downtrend in the market but may fail in the sideways market. Hence, it is imperative to identify uptrends and downtrends effectively.

  An inventory Retracement Bar in a downtrend

Figure: IRB

We found that inventory retracement bars effectively decreased the drawdowns and helped secure profits in bullish and bearish markets. We have derived the idea of Inventory Retracement Bars from the famous **Hoffmann Trading Strategy** and modified the idea to make it more apt for the BTCUSDT market.

All of these Risk Management Methods have been implemented to increase the robustness of our strategy. A combination of optimization of parameters in the main strategy supplemented with These risk management measures helped us generate great returns that too with minimal drawdown.

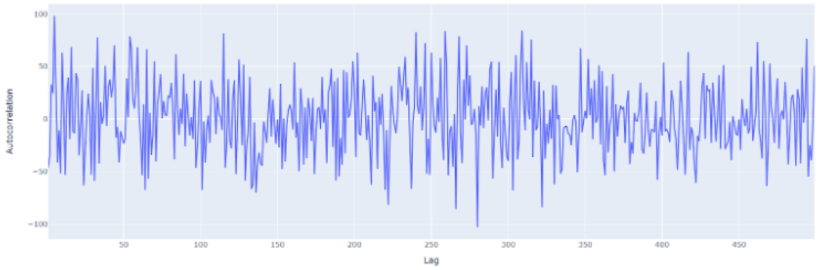
Statistical Testing for the Strategy

Statistical testing is crucial for any strategy as it provides a systematic and objective evaluation of the strategy's performance, helping distinguish genuine patterns or effects from random fluctuations and ensuring robust and reliable decision-making.Below are the statistical tests that we have adopted for our strategy.

- **Auto Correlation**

  Below is the code and the graph of autocorrelation vs Lag.

```
autocorrelation_values = []
max_lag = 100

for lag in range(1, max_lag):
    autocorr = np.correlate(returns[:-lag], returns[lag:]) / (np.std(returns[:-lag]) *
np.std(returns[lag:]))
    autocorrelation_values.append((lag, autocorr[0]))

autocorrelation_df = pd.DataFrame(autocorrelation_values, columns=['Lag', 'Autocorrelation'])
```



- **ADF TEST**

  The ADF test is carried out to find out the stationarity of the data. The below is the code and the results obtained on applying this to the close values of BTC-USDT daily data generated from the hourly data.

```
def adf_test(df):
    result = adfuller(df.values)
    print('ADF Statistic: %f' % result[0])
    print('p-value: %f' % result[1])
    print('Critical Values:')
    for key, value in result[4].items():
        print('\t%s: %.3f' % (key, value))
    if  result[1]>0.05:
      print('p-value>0.05 : data is non-stationary.')
    else:
      print('p-value<0.05 : data is stationary')

adf_test(data['close']) #Before Differencing
```

  We get that this is nonstationary. However, applying it on the difference of the closing values gives us stationarity.

```
df_trans=data.close.diff().dropna()
adf_test(df_trans) #After Differencing
```

- **Z test**

  The Z-test is a statistical method used to determine if there is a significant difference between sample and population means when the population standard deviation is known. The Z-test is adopted instead of the T-test when the population standard deviation is known, and the sample size is large (typically considered as $n > 30$). In such cases, the Z-test provides accurate results and is a suitable choice for hypothesis testing and comparing means.

  Below is the code along with the interpretation:

```python
Sharpe_ratio = Calculate_Parameters.sharpe_ratio(trade_wise_static['returns'])
number_of_years = len(data) // 365
t_statistic = Sharpe_ratio * number_of_years
from scipy.stats import t

def calculate_p_value(t_statistic, degrees_of_freedom):
    p_value = 2 * (1 - t.cdf(np.abs(t_statistic), df=degrees_of_freedom))
    return p_value

degrees_of_freedom=len(data)-1

p_value = calculate_p_value(t_statistic, degrees_of_freedom)
if p_value>0.05:
  print(f'P-Value: {p_value} > 0.05, strategy does not have a strategic importance')
if p_value<0.05:
  print(f'P-Value: {p_value} < 0.05, strategy has a strategic importance')
```

```
P-Value: 0.0 < 0.05, strategy has a strategic importance.
```

- **Binomial Test**

  The binomial test is employed in statistical analysis to determine if the observed proportion of binary outcomes significantly deviates from a specified expected proportion, offering a reliable method for assessing the significance of proportions in non-normally distributed data scenarios. It is a crucial tool for hypothesis testing, particularly in situations involving binary or categorical data.

  In our case, the alternate hypothesis is that the Strategy returns are consistently better than the benchmark. Below is the code showing how we rejected the null hypothesis(or adopted the alternate hypothesis)

```python
benchmark = []
for i in range(len(data)):
    benchmark.append(data['close'][i]*1000/data['close'][0]-1000)

import numpy as np
from scipy.stats import binom_test

# Define the threshold for superiority
threshold = 0.001

# Calculate the number of instances where strategy outperforms benchmark
superiority_count = np.sum(every_day_comp['portfolio value'] - benchmark > threshold)

# Set up the binomial test
num_trials = len(every_day_comp)
p_value = binom_test(superiority_count, num_trials, p=0.5, alternative='greater')

# Print the results
print(f"Superiority count: {superiority_count}")
print(f"P-value: {p_value}")

# Check if the null hypothesis is rejected at a significance level of 0.05
alpha = 0.05
if p_value < alpha:
    print("Reject the null hypothesis: Strategy returns are consistently better than the benchmark.")
else:
    print("Fail to reject the null hypothesis: No evidence that strategy returns are consistently better than the benchmark.")
```

```
Superiority count: 1473
P-value: 0.0
```

Abstract Strategy Testing

Strategy Parameters

- **Returns:**
Returns can be expressed nominally as the change in the dollar value of an investment over time. A return can also be expressed as a percentage derived from the ratio of profit to investment.

$$\text{Returns} = \frac{Value_f - Value_i}{Value_i} * 100$$

On our investment of 1000$ we have generated Net Profit of 18753.32$ which gives 1875.33 % on Compounding/ Similarly on Static we have generated net profit of 3557.51$ which gives 355.75% returns

- **Sharpe Ratio:**
Sharpe ratio measure of an investment's risk-adjusted performance, calculated by comparing its return to that of a risk-free asset.

$$\text{Sharpe Ratio} = \frac{R_s - R_f}{\sigma_p}$$

Sharpe ratio for our strategy comes out to be 10.92 which suggests there is good reward assosicated with a given risk.

- **Sortino Ratio:**
The Sortino ratio is a variation of the Sharpe ratio that differentiates harmful volatility from total overall volatility by using the asset's standard deviation of negative strategy returns—downside deviation—instead of the total standard deviation of portfolio returns.

$$\text{Sortino Ratio} = \frac{R_s - r_f}{\sigma_d}$$

Sortino ratio for our strategy comes out to be 98.53 which indicates exceptionally strong risk-adjusted returns for the investment portfolio.

- **Maximum Drawdown:**
A drawdown is a peak-to-trough decline during a specific period for an investment. Maximum Drawdown is maximum of all the drawdowns. We have calculate drawdown from the code given below

```python
def max_drawdown(returns):
        out = []
        cumulative = []
        start = 100
        cumulative.append(start)
        for ret in returns:
            cumulative.append(cumulative[-1] * (1 + ret / 100))
        max_return = [max(cumulative[:i + 1]) for i in range(len(cumulative))]
        drawdowns = [((cumulative[i] - max_return[i]) / max_return[i]) for i in range(len(cumulative))]
        out = min(drawdowns)
        return out
```

Maximum Drawdown for our strategy comes out to be 9.32% which is decent for an unpredictable market like BTC-USDT.

- **Dip :**
  Dip for a given trade is defined as maximum price movement which has gone against us.

$$\text{Dip} = \frac{EntryPrice - LowestPrice}{EntryPrice} * 100$$

  Maximum dip for our strategy comes out to be 22.28% and Average dip comes out to be 6.28% which could be a concern for our strategy

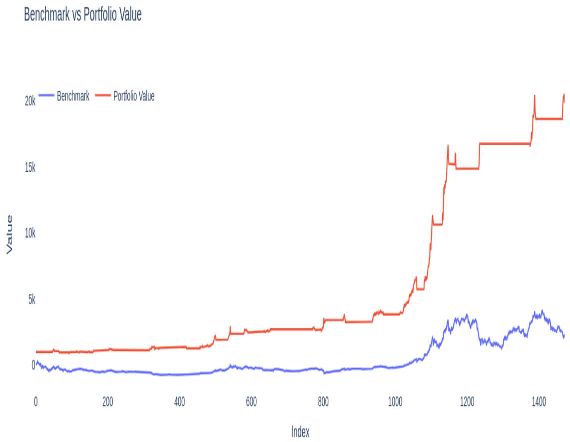
- **Return Over Max Drawdown(RoMaD):**
  Return over maximum drawdown (RoMaD) is a risk-adjusted return metric used as an alternative to the Sharpe Ratio or Sortino Ratio.

$$\text{RoMaD} = \frac{Returns}{MaxDrawdowns}$$

  RoMaD for our strategy comes out to be 201.21 which is very high from industry standards.

- **Transaction Costs:**
  Transaction costs are expenses incurred when buying or selling a commodity. We have taken 0.15% as Transaction Cost on each trade.

BTC-USDT, Daily 2018/01/01-2022/01/12, Ticker - 1D



Portfolio Value vs Benchmark

Static

3557.51$ Net Profit	3604.01$ Gross Profit	176.55$ Average Win
31 Total Trades	70.97% Win Rate	9.32% Max Drawdowns
10.92 Sharpe Ratio	98.53 Sortino Ratio	6.28% Average Dip
11.29 Average Holding (in days)	48 Max Holding (in days)	−36.29$ Average Loss

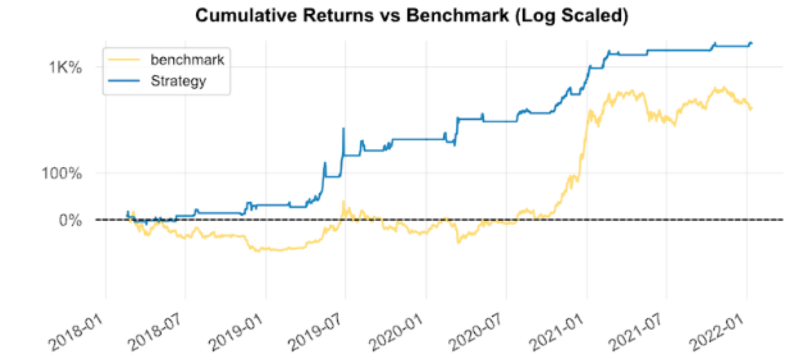
Compounding

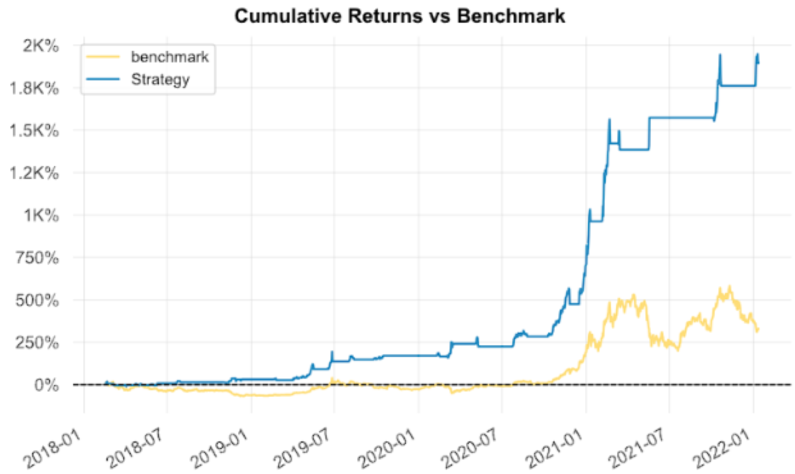
18753.32$ Net Profit	18975.79$ Gross Profit	1875.33% Returns

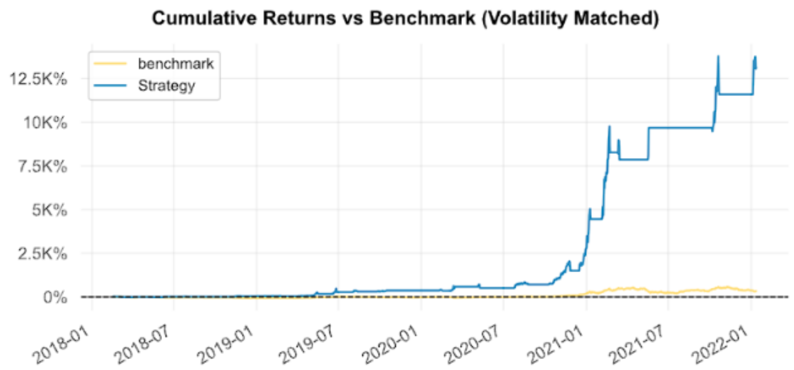
- Performance Report of our strategy

Key Performance Metrics

Metric	benchmark	Strategy
Risk-Free Rate	0.0%	0.0%
Time in Market	100.0%	25.0%
Cumulative Return	329.11%	1,875.33%
Sharpe	0.73	10.92
Sortino	1.05	98.53
Max Drawdown	-72.78%	-9.32%
Volatility (ann.)	60.98%	34.23%
Skew	-0.15	1.56
Kurtosis	3.95	29.55
Daily Value-at-Risk	-6.14%	-3.31%
Expected Shortfall (cVaR)	-6.14%	-3.31%
Payoff Ratio	1.18	1.17
Profit Factor	1.14	1.95
Common Sense Ratio	1.14	4.53
CPC Index	0.69	1.42
Tail Ratio	1.0	2.33
Outlier Win Ratio	3.39	17.72
Outlier Loss Ratio	3.29	3.12
MTD	-8.59%	7.33%
3M	-25.58%	13.5%
6M	25.66%	19.36%
YTD	-8.59%	7.33%
1Y	31.06%	88.03%
3Y (ann.)	79.04%	90.6%
5Y (ann.)	29.38%	69.82%
10Y (ann.)	29.38%	69.82%
All-time (ann.)	29.38%	69.82%
Best Day	21.54%	22.76%
Worst Day	-23.19%	-19.7%
Best Month	58.39%	43.31%
Worst Month	-36.6%	-9.32%
Best Year	302.75%	196.97%
Worst Year	-62.38%	7.33%
Avg. Up Month	28.35%	18.79%
Avg. Down Month	-15.77%	-3.12%
Win Days	50.91%	62.22%
Win Month	47.92%	76.67%
Win Quarter	52.94%	81.25%
Win Year	60.0%	100.0%
Beta	-	0.15
Alpha	-	0.52
Correlation	-	26.03%
Treynor Ratio	-	12989.14%







Backtesting Engines

We have developed two full backtesting frameworks: One in Python and another using the backtrader library.

Backtrader Library

```python
#Final backtrader Framework. Takes in a dataframe and gives various parameters.
import backtrader.analyzers as btanalyzers
class MyStrategy(bt.Strategy):

    def log(self, txt, dt=None):
      dt = dt or self.datas[0].datetime.date(0)
      print('%s, %s' % (dt.isoformat(), txt))

    def notify(self, order):
        if order.status in [order.Submitted, order.Accepted]:
            # Buy/Sell order submitted/accepted to/by broker - Nothing to do
            return

        # Check if an order has been completed
        # Attention: broker could reject order if not enougth cash
        if order.status in [order.Completed, order.Canceled, order.Margin]:
            if order.isbuy():
                self.log(
                    'BUY EXECUTED, Price: %.2f, Cost: %.2f, Comm %.2f' %
                    (order.executed.price,
                     order.executed.value,
                     order.executed.comm))

                self.buyprice = order.executed.price
                self.buycomm = order.executed.comm
                self.opsize = order.executed.size
            else:  # Sell
                self.log('SELL EXECUTED, Price: %.2f, Cost: %.2f, Comm %.2f' %
                         (order.executed.price,
                          order.executed.value,
                          order.executed.comm))

                gross_pnl = (order.executed.price - self.buyprice) * \
                    self.opsize


                net_pnl = gross_pnl - self.buycomm - order.executed.comm
                self.log('OPERATION PROFIT, GROSS %.2f, NET %.2f' %
                         (gross_pnl, net_pnl))

    def __init__(self):
        self.percent_of_cash_per_trade = 1
        self.in_position = None  # Flag to track position
        self.trades=0
        self.netreturns=0
        self.entry_price=0
        self.profits=0
        self.profitable_trades=0
        self.loss_trades=0
        self.losses=0
        self.sizes=0

    def next(self):
        #self.log(self.broker.cash)
        if self.in_position==None:
            if self.data.signal[0]==1:
                self.sizes = ((self.broker.cash)-0.0000001) / self.data.close[0]
                self.log(self.broker.cash)
                self.buy(size=self.sizes)
                self.log(f'BUY LONG:{self.data.close[0]}')
                self.in_position='Long'
                self.entry_price=self.data.close[0]

            elif self.data.signal[0]==-1 :
                self.sizes = (self.broker.cash) / self.data.close[0]
                self.log(self.broker.cash)
                self.sell(size=self.sizes)
                #self.log(f'SELL SHORT:{self.data.close[0]}')
                self.in_position='Short'
                self.entry_price=self.data.close[0]
        elif self.in_position=='Long':
            if self.data.signal[0]==-1:
                self.log(self.broker.cash)
                self.close(size=self.sizes)
                self.trades+=1
                self.log(f'Trade Number:{self.trades},CLOSING LONG POSITION:{self.data.close[0]}')
                self.sizes=0
                self.in_position = None

        elif self.in_position=='Short':
            if self.data.signal[0]==1:
                self.log(self.broker.cash)
                self.close(size=self.sizes)
                self.trades+=1
                self.log(f'Trade Number:{self.trades},CLOSING SHORT POSITION:{self.data.close[0]}')
                self.in_position = None
                self.sizes=0
```

Python

```
def printTradeAnalysis(analyzer):
    '''
    Function to print the Technical Analysis results in a nice format.
    '''
    #Get the results we are interested in
    total_open = analyzer.total.open
    total_closed = analyzer.total.closed
    total_won = analyzer.won.total
    total_lost = analyzer.lost.total
    win_streak = analyzer.streak.won.longest
    lose_streak = analyzer.streak.lost.longest
    pnl_net = round(analyzer.pnl.net.total,2)
    strike_rate = (total_won / total_closed) * 100
    #Designate the rows
    h1 = ['Total Open', 'Total Closed', 'Total Won', 'Total Lost']
    h2 = ['Strike Rate','Win Streak', 'Losing Streak', 'PnL Net']
    r1 = [total_open, total_closed, total_won, total_lost]
    r2 = [strike_rate, win_streak, lose_streak, pnl_net]
    #Check which set of headers is the longest.
    if len(h1) > len(h2):
        header_length = len(h1)
    else:
        header_length = len(h2)
    #Print the rows
    print_list = [h1,r1,h2,r2]
    row_format ="{:<15}" * (header_length + 1)
    print("Trade Analysis Results:")
    for row in print_list:
        print(row_format.format('',*row))

if __name__ == '__main__':
    cerebro = bt.Cerebro()

    cerebro.adddata(data)
    cerebro.broker.addcommissioninfo(CommInfoFractional())
    cerebro.addstrategy(MyStrategy)
    cerebro.broker.set_coc(True)
    #cerebro.addsizer(bt.sizers.PercentSizer, percents=10)
    #cerebro.broker.set_shortcash(True)
    #cerebro.addsizer(bt.sizers.SizerFix,stake=3)
    cerebro.addanalyzer(btanalyzers.SharpeRatio, _name='mysharpe', timeframe=bt.TimeFrame.Days,riskfreerate=0.03 )
    cerebro.addanalyzer(btanalyzers.DrawDown, _name='drawdown', fund=True)
    cerebro.addanalyzer(btanalyzers.TradeAnalyzer, _name='ta')
    cerebro.broker.set_cash(1000.0)
    cerebro.broker.setcommission(commission=0.00)

    print('Starting Portfolio Value: %.2f' % cerebro.broker.getvalue())
```

```
    print('Starting Portfolio Value: %.2f' % cerebro.broker.getvalue())

    thestrats=cerebro.run()
    thestrat = thestrats[0]
    printTradeAnalysis(thestrat.analyzers.ta.get_analysis())
    print('Sharpe Ratio:', thestrat.analyzers.mysharpe.get_analysis())
    print('Max Drawdown:', thestrat.analyzers.drawdown.get_analysis())
    #print('Net Returns:',thestrat.netreturns)
    #print('Profits:',thestrat.profits)
    #print('Losses:',thestrat.losses)
    #print('Profitable Trades:',thestrat.profitable_trades)
    #print('Loss Trades:',thestrat.loss_trades)
    #print('Total Closed Trades:',thestrat.trades)
    #print('Average Winning Trade:' ,thestrat.profits/thestrat.profitable_trades)
    #print('Average Losing Trade:' ,thestrat.losses/thestrat.loss_trades)
    print('')
    print('Ending Portfolio Value: %.2f' % cerebro.broker.getvalue())
    #print(MyStrategy.returns)
```

We have used the Python backtesting framework in our final code for the strategy. The Backtesting engine in Python was implemented using classes to make it more systematic and organized.

Limitations

- **Volatility and Uncertainty in BTC-USDT Market:**

  The effectiveness of our strategy in controlling market dips is hindered by the inherent volatility of the cryptocurrency market, exacerbated by the uncertainty in the BTC-USDT market. Despite Tether's stabilizing influence, sudden and unpredictable market fluctuations can pose challenges in effectively mitigating losses during downward trends. The absence of position sizing limits our ability to tailor trade sizes to the confidence ratio, potentially impacting our capacity to strategically manage dips.

- **Limited Profit Potential in Sideways Markets:**

  Our trading approach may encounter limitations in generating substantial profits during periods of market consolidation or sideways movement. The uncertainty in market direction during such phases poses challenges in capitalizing on significant price movements, potentially impacting the overall profitability of the strategy.

- **Potential for False Signals:**

  The utilization of indicators, while instrumental in guiding our trading decisions, introduces the possibility of false signals. Market dynamics are influenced by various factors, and indicators may not always accurately reflect impending trend changes. However, our risk management plan is designed to minimize losses during instances of false signals, safeguarding the overall integrity of our trading strategy.

- **Sensitivity to External Factors:**

  The strategy is inherently sensitive to external factors such as regulatory changes, macroeconomic events, and technological developments in the cryptocurrency space. Unforeseen external influences can introduce an element of unpredictability, challenging the adaptability of our strategy to evolving market conditions.

- **Limited Influence Over Market Variables:**

  Given the decentralized and dynamic nature of the cryptocurrency market, our ability to exert control over various market variables is constrained. Factors such as liquidity, market sentiment, and the evolving regulatory landscape may impact the efficacy of our strategy, necessitating constant monitoring and adjustments.

- **Conclusion:**

  While our trading strategy provides a structured framework for navigating the Bitcoin market, it is essential to recognize and account for its limitations. The multifaceted nature of the cryptocurrency market, coupled with external influences, underscores the need for an adaptive and risk-aware approach. Acknowledging these limitations enhances our capacity to refine and optimize our strategies in the ever-evolving landscape of BTC - USDT trading.

Conclusion

In conclusion, our journey through the development and analysis of this algorithmic trading strategy has been a dynamic exploration of market intricacies, risk management, and technical indicators. We have finally concluded this problem statement by incorporating a strategy comprising of Relative Strength Index (RSI), Relative Vigor Index (RVI), Supertrend, Moving Average Convergence Divergence (MACD), and Weighted Bar Strength. We believe that this strategy is robust and effective for the BTC/USDT market and the best that we found throughout the one month of solving this problem statement. However, we also recognize that this strategy is not perfect and still has scopes for improvement, just like any other strategy.

When we started off this problem statement, we struggled to beat the benchmark with drawdowns less than 40 percent. We could have never imagined that we would be able to craft a strategy that would beat the benchmark by a vast margin while also minimizing the drawdown to less than ten percent. It has been an amazing journey of improvement throughout and we have learned a lot. We have found promising ideas such as the two strategy framework, which we could not incorporate into our final strategy due to various reasons but we are sure to keep experimenting on these ideas and are certain that we would be able to craft better and more robust strategies in the coming future.

We are grateful to ZeltaLabs to provide us the opportunity to learn and grow through this problem statement.

Appendix

After trying out more strategies than flavors at an ice cream shop, we've landed on 'The One.' It's been a bit like a treasure hunt, with some wins, some losses, and a few moments of scratching our heads. Here's the story told in numbers – our strategy's journey from trial to triumph.We implemented over 50 strategies and tried to optimize them with risk management measures. Our final result is:

- Time Frame: daily
- Net Profit on Static: 3557.51
- Compounding: 1875.33 percent
- Drawdown: 9.32 percent
- Sharpe:10.92

The performance of some of the strategies can be viewed in the following Drive Link:
Drive Link

Strategy	Time-Frame	Net Profit on Static	Returns on Compounding
Macd of Kama + RSI	1h	4521	63x
Keltner Channel + Heikin Ashi + Piercing	1h	4019	92x
Sma of obv + macd	1h	3654	26x
Ichimoku cloud + ADX + IBS	daily	3510	6x
Macd of mcglinney average + ADX	daily	6432	22x
Turtle Channel	daily	2582	7x
Volume surge	daily	2304	8x
Vortex indicator + MACD + True Range	daily	927	2x
Williams R	daily	2422	12x
Trix Indicator	1h	215	1x
Volty Expan Strategy	daily	3306	23.5x
Cuppock+RSI(reverse)+50SMA	daily	2500	5.5x

Table 1 – Strategy along with metrics-1

Strategy	Max drawdown (in percent)	Sharpe ratio
Macd of Kama + RSI	40	3
Keltner Channel + Heikin Ashi + Piercing	24	3.75
Sma of obv + macd	39	3.04
Ichimoku cloud + ADX + IBS	31	4.18
Macd of mcglinney average + ADX	36	2.19
Turtle Channel	42	3.86
Volume surge	8	8.67
Vortex indicator + MACD + True Range	39	3.67
Williams R	42	3.54
Trix Indicator	48	0.759
Volty Expan Strategy	20	2.45
Cuppock+RSI(reverse)+50SMA	19.7	2.5

Table 2 – Strategy along with metrics-2

Apart from applying the above-mentioned strategies we also implemented them in an alternating fashion as has been explained in the subsection path forward in section **Strategy Developement**.

In the Drive Link we have attached the codes for some of our strategies along with the alternating method for both static and compounding.

References

Giving a nod to our strategy sidekicks – the books, articles, and smart folks who shared their wisdom. They're like the trusty guides leading us through the strategy jungle. Imagine them as the cool older siblings of our strategy, sharing all the tips and tricks. Big thanks to the references that helped us level up in our financial journey!

- Investopedia
- Trading View
- Backtrader: helped in verfying our backtesting framework
- Scalping
- Pine Script Documentation
- Neural Network based algo trading
- Forecasting in crypto market
- Hurst Exponent
- Technical Chart patterns
- Williams Fractal
- IBS
- Directory of Trading Strategies
- TA-Lib documentation
- Some more advanced strategies
- Parabolic SAR
- KNN based approach
- Statistical Test
- Plotly for interactive graphs