Alphavantage Quantitative Hackathon **Report**

Prepared By :

Team Jainwin

Team Members:

Hemang Jain
Saksham Jain
Kanishk Jain
Aman Kumar
Ridhwan Deshwal
Aarushi Anand
Kushagra Gupta
Arpit Sinha

Index

1. Introduction

Bitcoin (BTC) and Ethereum (ETH), the two largest cryptocurrencies by market capitalization, serve distinct roles in the crypto ecosystem - with BTC often considered a "digital gold" and ETH operating as the foundational asset of decentralized applications. Their USDT (Tether) pairs are among the most liquid globally, offering deep order books, tight spreads, and sufficient granularity in historical tick and minute-level data for pattern recognition and model training.

Key Observed Trends and Patterns:

- **Volatility Clusters**: Both BTC and ETH demonstrate periods of high price turbulence followed by relatively calm phases, a phenomenon akin to volatility clustering in traditional markets.

- **Momentum and Mean Reversion Cycles**: Depending on market regime, momentum strategies (e.g., breakout trading) often outperform during sustained trends, while mean-reversion logic (e.g., Bollinger Band or RSI-based signals) performs better in sideways markets.

- **Correlation and Divergence**: BTC and ETH generally maintain a strong positive correlation; however, temporary divergences in price movement and volatility levels offer opportunities for statistical arbitrage and pair trading strategies.

- **Volume-Driven Breakouts**: Volume surges - particularly near support/resistance levels - are frequently associated with significant price breakouts, suggesting the importance of volume-based features in predictive modeling.

- **Weekend Effects**: Due to the 24/7 nature of crypto, weekends often witness lower liquidity, leading to exaggerated price moves or anomalies that can be exploited algorithmically.

Cryptocurrency markets like BTC/USDT and ETH/USDT offer immense profit potential, but their high volatility, rapid regime shifts, and noise-laden data make algorithmic trading particularly challenging. Traditional rule-based systems, while intuitive, often fail to generalize across dynamic conditions. This project addresses those limitations by designing three robust strategies - each tailored to different market scenarios using a mix of classical indicators, statistical filters, and machine learning agents.

Our approach spans:

- **Enhanced Technical Strategy**: A rules-based algorithm that combines Exponential Moving Averages (EMAs), RSI, Bollinger Bands, ADX, VWAP, and ATR to generate high-confidence entries and exits. Filters like Bollinger Band width and directional movement (DI+/DI−) prevent entries in choppy markets. This strategy is focused on precision trend-following and risk-managed profit targets.

- **Regime-Aware Volatility Model**: This hybrid approach uses statistical tools such as the **Hurst Exponent** (for trend persistence), **CUSUM filters** (to detect breakouts/crashes), and **Kalman Filters** (for price smoothing) to identify shifts between trending and mean-reverting regimes. The strategy dynamically adapts its logic based on the underlying market state to avoid false signals and time entries during favorable volatility conditions.

- **Q-Learning Reinforcement Agent**: This model-free reinforcement learning agent interacts with a custom-built trading environment. Using discretized market states (from indicators like Aroon, EMA, RSI, and price changes), it learns to make profitable trading decisions through trial-and-error. The agent is trained using a Q-table that updates with each episode and is optimized to maximize cumulative rewards while managing risk via drawdown penalties and stop-loss logic.

- **Hybrid Technical + ML Strategy**: This layer of our system integrates **machine learning classifiers (Random Forests, Decision Trees)** with **Neural Networks** for enhanced signal validation. Technical indicators such as MACD, SMA/EMA crossovers, Stochastic Oscillators, and price-based features are used to detect local minima/maxima. Signals are further refined by candlestick pattern recognition (e.g., Doji, Hammer, Engulfing) and confirmed via ensemble model consensus, creating a highly adaptive and intelligent decision-making pipeline.

All four strategies were tested on hourly BTC/USDT and ETH/USDT price data from 2020–2024, incorporating realistic elements like slippage, trading fees, and drawdown tracking. This multi-tiered approach is aimed at creating **profitable, risk-aware, and adaptable** trading systems suitable for deployment in real-world algorithmic infrastructure.

1.1. Market Dynamics

When Bitcoin experiences significant price movements-whether due to macroeconomic factors, institutional investment, or regulatory news-it tends to trigger ripple effects across the broader market, including Ethereum and other altcoins. This makes Bitcoin a key market indicator, as its fluctuations often set the tone for investor sentiment and market direction across the entire crypto sector.

Ethereum, on the other hand, plays a foundational role in the crypto space by supporting decentralized applications (dApps) and smart contracts. While Bitcoin captures investor sentiment as a hedge or reserve asset, Ethereum drives innovation and real-world utility within blockchain ecosystems. The interconnected nature of BTC and ETH, combined with their high trade volumes and volatility, makes understanding their relationship crucial for interpreting broader market dynamics and managing risk effectively.

1.2. Indicators Used

1.2.1. Moving Averages (MA7 and MA14)

This crossover method helps traders spot trend changes early. When the short moving average rises above the long one, it often signals the start of an uptrend, and vice versa for a downtrend.

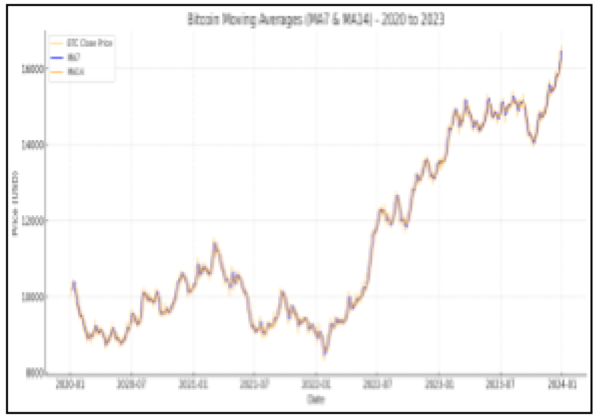

Fig 1.2.1.1. : Moving Averages

- **MA7:** 7-period simple moving average of the closing price.
- **MA14:** 14-period simple moving average of the closing price.

Interpretation:

- When **MA7 > MA14**, it indicates bullish (uptrend) → MA_Signal = 1
- When **MA7 < MA14**, it indicates bearish (downtrend) → MA_Signal = -1

$$SMA(t) = (P(t) + P(t-1) + \cdot \cdot \cdot + P(t-N+1))/N$$

$$P(t) = Price\ at\ time\ t$$

$$N = No.\ of\ periods$$

$$SMA(t) = Moving\ avg\ at\ time\ t$$

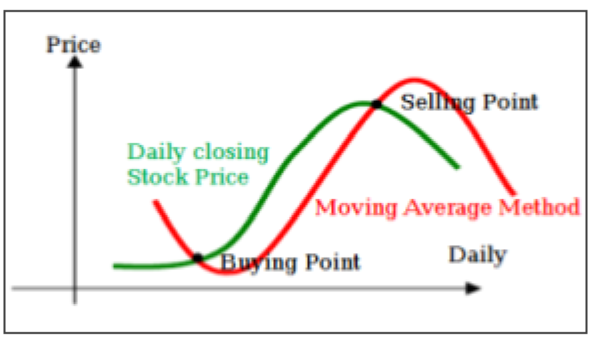

Fig. 1.2.1.2 :The signal points of trading rule

Fig. 1.2.1.3 : Simple Moving Average

Source: [Determination of Trading Points using the Moving Average Methods]

1.2.2. Aroon Indicator (Aroon_Up and Aroon_Down)

The Aroon indicator shows how strong and recent a trend is. A higher Aroon_Up means the price hit new highs recently-suggesting a strong bullish move. The closer the values are to 100, the stronger the trend.

- **Aroon_Up:** Measures how long it's been since the highest high over the past 15 periods.
- **Aroon_Down:** Measures how long it's been since the lowest low over the past 15 periods.

Interpretation:

- **Aroon_Up > Aroon_Down** → uptrend → Aroon_Signal = 1
- **Aroon_Up < Aroon_Down** → downtrend → Aroon_Signal = -1

$$Aroon\ up\ =\ 100\ \times\ ((N\ -\ Periods\ since\ highest\ high)\ \div\ N)$$

$$Aroon\ down\ =\ 100\ \times\ ((N\ -\ Periods\ since\ lowest\ low)\ \div\ N)$$

Fig.1.2.2.1 : Aroon up and Aroon down

Fig. 1.2.2.2: Aroon uptrend and downtrend

Source: [A thermodynamic description of the time evolution of a stock market index]

1.2.3. RSI (Relative Strength Index)

The **Relative Strength Index (RSI)** is a momentum oscillator that measures the speed and magnitude of recent price changes to evaluate whether an asset is **overbought** or **oversold**

- **RSI_14**: Measures the magnitude of recent price changes to evaluate overbought or oversold conditions over a 14-period window.

Interpretation:

- **RSI >75** →overbought→ potential price drop → RSI_Signal = 1
- **RSI <35** → oversold→ potential price rise → RSI_Signal = -1

$$RSI\ =\ 100\ -\ (100/(1\ +\ RS))$$

$$RS\ =\ Average\ gain\ over\ N\ periods/Average\ loss\ over\ N\ periods$$

Fig 1.2.3.2 : Relative Strength Index

Fig. 1.2.2.3: RSI Analysis

1.2.4. Exponential Moving Averages(EMA7,EMA14,EMA28)

Exponential Moving Averages (EMAs) are trend-following indicators that assign more weight to recent prices, making them more responsive to market changes compared to simple moving averages.

- **EMA7**: Short-term EMA
- **EMA14**: Mid-term EMA
- **EMA28**: Long-term EMA

EMA_Signal:

- **EMA7 > EMA14 > EMA28**→ strong bullish trend→EMA_Signal = 1
- **EMA7 < EMA14 < EMA28**→ strong bearish trend→EMA_Signal = -1

$$EMA(t) = \alpha \cdot P(t) + (1 - \alpha) \cdot EMA(t - 1)$$

$$EMA(t) = EMA\ at\ time\ t$$

$$P(t) = Price\ at\ time\ t$$

$$\alpha = 2/N + 1 = Smoothing\ factor$$

$$N = Number\ of\ periods\ (ex - 10\ day\ EMA\ period)$$

Fig. 1.2.4.1 : 4- Exponential Moving Average

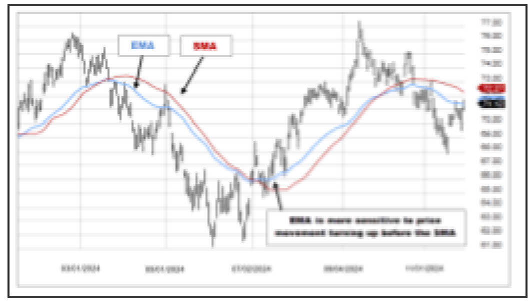

Fig 1.2.4.2 : EMA and SMA comparision

1.2.5. Hurst Exponent (for trend strength)

The **Hurst Exponent (H)** is a statistical measure used to evaluate the **trend strength and predictability** of a time series. The Hurst exponent ‘H’ is the **slope of the plot of each range’s log(R/S)** versus each range’s log(size). To **avoid trading in choppy markets** by identifying whether a market has a **strong trend (H > 0.5)** or is **mean-reverting/noisy (H ≤ 0.5)**.

- If the Hurst value is **more than 0.5** then it would indicate a persistent time series (roughly translates to a trending market).
- If the Hurst Value is **less than 0.5** then it can be considered as an anti-persistent time series (roughly translates to sideways market).
- If the Hurst value **is 0.5** then it would indicate a random walk or a market where prediction of future based on past data is not possible.

1.2.6. Kalman Filter (for noise reduction)

Kalman Filters are recursive statistical algorithms used to estimate the true value of a variable in the presence of noise. In trading, they are applied to **price data smoothing**, helping filter out market noise and better identify actual trends, especially in volatile assets like BTC/USDT.

- **Purpose**: Smooths out random fluctuations in price and emphasizes underlying trends
- **Why it's useful**: Reduces false signals caused by high-frequency noise, improving trend clarity
- **Behavior**: Reacts quickly in stable markets, slows down during sudden volatility spikes

Kalman_Signal:

- **Filtered_Price > Previous_Filtered_Price** → upward trend → `Kalman_Signal = 1`
- **Filtered_Price < Previous_Filtered_Price** → downward trend → `Kalman_Signal = -1`
- **No major change** → flat market → `Kalman_Signal = 0`

1.2.7. ATR (for volatility)

ATR is a **non-directional** indicator that measures how much an asset moves, on average, over a set period.It considers **gaps** and **intraday volatility** for a fuller picture of price action. Higher ATR = more volatile = higher risk.

$$ATR = 1/n \sum_{i=1}^{n} TR_i$$

$$TR_i \ is \ the \ true \ range \ for \ the \ i \ th \ period \ ATR = 1/n \sum_{i=1}^{n} TR_i$$

$$n \ is \ the \ number \ of \ periods \ over \ which \ the \ ATR \ is \ calculated$$

1.2.8. Rolling Hurst Exponent Indicator

The Rolling Hurst Exponent (H) is a statistical measure used to analyze the persistence, randomness, or mean-reverting behavior of a time series, such as stock prices. It helps identify whether a market is trending, random, or reverting.

- **H > 0.5**: Persistent behavior (trending market)-price movements tend to continue in the same direction.
- **H = 0.5**: Random walk (no trend)-price movements are unpredictable.
- **H < 0.5**: Mean-reverting behavior (anti-persistent)-price movements tend to reverse.

CALCULATION:

1. **Select a rolling window**: Choose a lookback period N N N (e.g., 50 days) to compute the Hurst Exponent over a rolling window.
2. **Logarithmic returns**: Compute the log returns of the price series:

$$R(t) = ln(P(t)) - ln(P(t - 1))$$

$$P(t) = Price \ at \ time \ t$$

3. **Rescaled Range (R/S) Analysis**:
    - Compute the mean of the returns over the window.
    - Calculate the cumulative deviation from the mean.
    - Find the range R R R (max deviation - min deviation) and standard deviation S S S.
    - Compute the rescaled range: R/S R/S R/S.

$$E\left[\frac{R(n)}{S(n)}\right] = Cn^H \qquad \text{as } n \to \infty$$

Where:

- $\left[\frac{R(n)}{S(n)}\right]$ is the Rescaled Range.
- $E[x]$ is the expected value.
- n is the time of the last observation (e.g., it corresponds to X_n in the input time series data).
- h is a constant.

1.2.9. Supertrend Indicator

The Supertrend Indicator is a trend-following tool used to identify the direction of the market trend and generate buy/sell signals. It plots a line above or below the price, acting as a dynamic support or resistance level, and is particularly effective in trending markets.

- **Supertrend Line Above Price**: Indicates a downtrend (bearish).
- **Supertrend Line Below Price**: Indicates an uptrend (bullish).

CALCULATION:

- **Upper Band** = $((High + Low)/2) + Multiplier \times ATR$
- **Lower Band** = $((High + Low)/2) - Multiplier \times ATR$
- **Supertrend(t)** =

$$\{Lower\ Band(t),\ if\ price(t) > Supertrend(t-1),\}$$

$$\{Upper\ Band(t),\ if\ price(t) < Supertrend(t-1),\}$$

CUSUM_Signal:

- **CUSUM > Threshold → Upward shift detected → CUSUM_Signal = 1**
- **CUSUM < -Threshold → Downward shift detected → CUSUM_Signal = -1**

Formula:

$$C^+_t = max\ (0,\ C^+_{t-1} + (P_t - \mu - \delta))$$

$$C^-_t = min\ (0,\ C^-_{t-1} + (P_t - \mu + \delta))$$

1.2.10. Bollinger Band

Bollinger Bands are **volatility-based indicators** that consist of a **moving average** and **two standard deviation bands**. They help identify **overbought/oversold conditions**, **price breakouts**, and **choppy/ranging markets**.

Bollinger_Signal:

- **Price > Upper Band** → Overbought / Breakout → **Bollinger_Signal = -1**
- **Price < Lower Band** → Oversold / Reversal Zone → **Bollinger_Signal = 1**
- **Price between Bands** → Neutral → **Bollinger_Signal = 0**

Formula:

$$Middle\ band\ =\ MA(N)$$

$$Upper\ Band\ =\ MA(N)\ +\ k\ \cdot\ \sigma$$

$$Lower\ Band\ =\ MA(N)\ -\ k\ \cdot\ \sigma$$

2. Strategy 1 - Q-Learning Based Strategy

Q-Learning is a model-free reinforcement learning algorithm that helps an agent learn the optimal action-selection policy by maximizing cumulative rewards over time. The agent interacts with the environment, updates a Q-table of state-action values, and learns which actions yield the highest rewards-without needing a model of the environment. By balancing exploration and exploitation, Q-Learning is effective in dynamic, uncertain settings like financial markets.

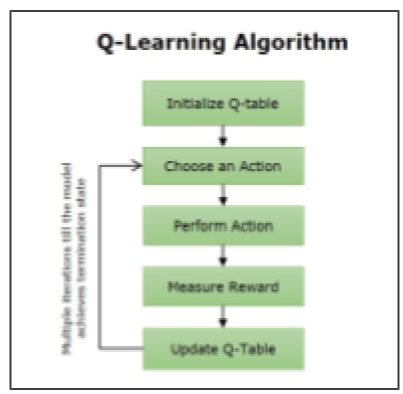

Fig 2.1 : DQN Algorithm

2.1. Why Q-Learning?

- No prior knowledge of the environment needed (model-free).
- Learn from interaction and experience.
- Suitable for sequential decision-making (e.g., trading).
- Handles exploration (trying new things) and exploitation (using known good actions).
- Converges to the optimal policy under the right conditions.

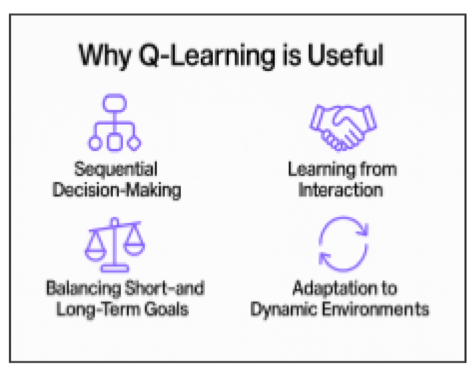

Fig 2.1.1 : Q-Learning

2.2. Q-Learning Reinforcement Agent

This strategy uses a Q-learning-based reinforcement learning agent for algorithmic trading. It leverages technical indicators like Aroon, RSI, and EMA, converting them into discrete bins to form a manageable market state space. The agent learns optimal trading actions by updating a Q-table with expected rewards, enabling it to make smarter decisions over time. Without needing prior market knowledge, it adapts to changing conditions, balancing exploration and exploitation to continuously improve and find better trading opportunities.

INDICATORS	PURPOSE
MOVING AVERAGE	A simple indicator that averages past prices over a set period to smooth out fluctuations and reveal the overall trend.
AROON SIGNAL	Consists of Aroon Up and Aroon Down lines to show how recently a stock hit highs or lows, helping to detect trend changes and strength.
EXPONENTIAL MOVING AVERAGE	A moving average that reacts faster to price changes by giving more weight to recent data, often used for short-term trend analysis.
RELATIVE STRENGTH INDEX	A momentum oscillator that ranges from 0 to 100, used to spot overbought (>70) or oversold (<30) market conditions.
KEY TECHNICAL INDICATORS USED	

Fig 2.2.1 : DQN Indicators

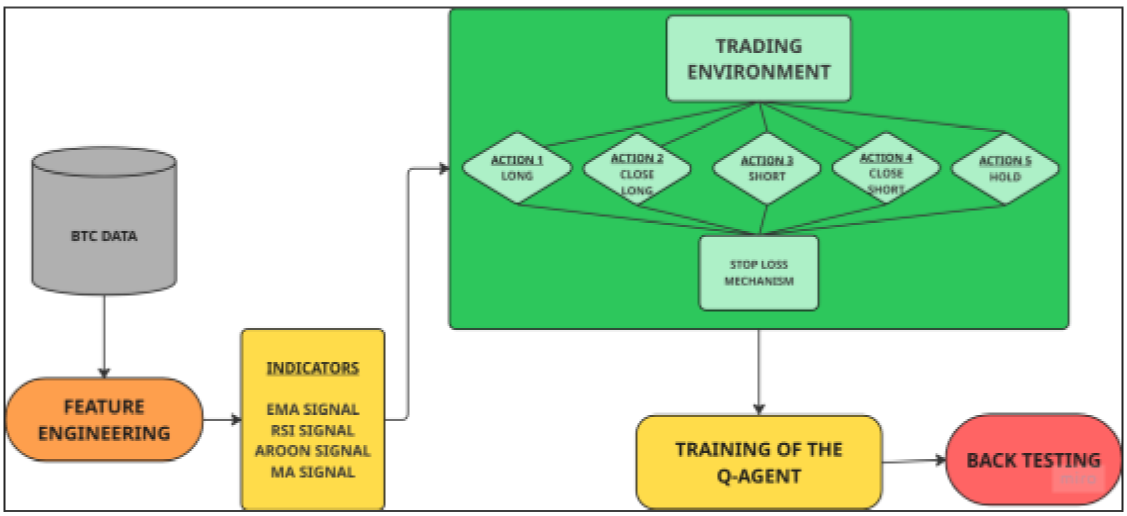

Fig 2.2.2 : DQN Workflow Flowchart

2.3. Trading Environment

The **TradingEnvironment** class simulates a realistic trading scenario for training reinforcement learning agents. Starting with a $10,000 balance, it lets the agent act based on indicators like **Aroon, RSI, EMA**, and **price change**. It tracks key metrics such as balance, net worth, positions, and trade history. The **reset()** method reinitializes the environment for new training episodes, offering a controlled setup for developing and testing Q-learning-based strategies.In our custom **TradingEnvironment**, we defined a total of **four trading actions** that the reinforcement learning agent can take at each step. These actions represent practical decisions a trader might make, and they are crucial for simulating a realistic trading strategy. Here's a breakdown of each action:

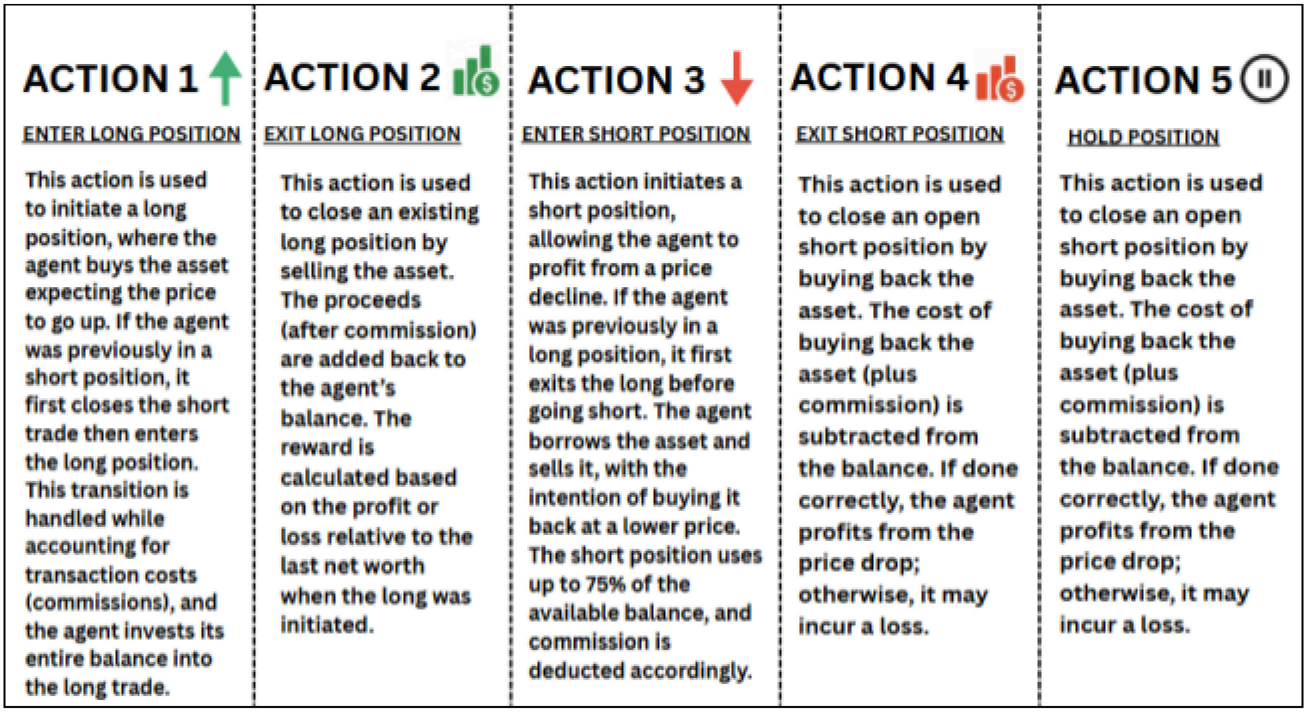

Fig 2.3.1 : Trading Environment

2.4. Stop-Loss Logic

To protect the trading agent from excessive losses and to simulate a realistic trading behavior, our team integrated a **stop-loss mechanism** into the trading environment. This mechanism automatically exits trades when certain loss thresholds are breached. The logic is built upon two primary conditions, one for long positions and the other for short positions.

2.4.1. Stop-Loss for Long Position

This condition triggers when the agent is in a long position and the current actual_price falls below a predefined percentage **(STOP_LOSS_PERCENT)** of the entry_price.

When triggered:

- The agent sells off all holdings.
- The commission is deducted from the sale amount.
- The net proceeds are added to the agent's balance.
- The trade is logged as a 'long_close'.
- The reward is adjusted based on the profit or loss made relative to the last known net worth.
- The position is reset to neutral (0).

2.4.2. Stop-Loss for Short Position

This condition activates when the agent is in a short position and the price rises beyond a defined loss threshold.

In this scenario:

- The agent buys back the asset at a higher price to exit the short.
- Commission is applied and subtracted from tbalance.
- The profit or loss is calculated in a similar manner.
- If the balance falls below a minimum trade amount, the agent is penalized heavily (e.g., reward += -100000000) and the episode is terminated (done = True).
- Otherwise, the trade is recorded as a 'short_close', and the position is reset.

Q-Learning Hyperparameters

Parameter	Value
action_size	5.0
alpha	0.05
gamma	0.95
epsilon	1.0
epsilon_decay	0.99
epsilon_min	0.15
ep	1300.0
n_pct_bins	20.0
n_signal_bins	3.0
n_holdings_states	3.0

Fig 2.4.2.1: Q Learning Hyperparameters

We have defined a Q-Learning framework for a trading agent by specifying key hyperparameters such as the learning rate (alpha), discount factor (gamma), and exploration parameters (epsilon, decay, and minimum). To manage the complexity of the state space, we discretized it using bins for technical indicators - Aroon, RSI, EMA - and percent price change, along with the agent’s holding status. Each unique combination of these features corresponds to a specific state index, which is used to access and update the Q-table.

2.5. State Space Design

The agent defines a **multi-dimensional discrete state space** using five features:

- **Aroon Signal** (n_signal_bins = 3): Trend signal (-1, 0, 1).
- **RSI Signal** (3 bins): Overbought/neutral/oversold.
- **EMA Signal** (3 bins): Moving average direction.
- **Holdings State** (3 states): Whether the agent is holding a position (long/flat/short).
- **Percent Change Signal** (n_pct_bins = 20): Binned return from price movement.

These are encoded into a **unique state index** using a function get_state_index(...) to look up values in the Q-table.

2.6. Q-table Initialization

A **Q-table** of shape (state_size, action_size) is initialized with zeros. Each row corresponds to a discrete state, and each column represents an action (e.g., buy, sell, hold, etc.). The goal is to learn the **expected future rewards** for each state-action pair.

2.6.1. Training Loop (Episodes)

The agent interacts with a custom `TradingEnvironment` across 1300 episodes, using an epsilon-greedy strategy to balance exploration and exploitation. At each step, it observes the current state, selects an action, and receives the next state, reward, and a done flag. The Q-table is updated using the Bellman equation, factoring in immediate and discounted future rewards ($\gamma = 0.95$).

2.7. Backtesting Results

1. Trade Statistics

Metrics	Value
Total Trades	37
Long trades	37
Short Trades	0
Winning Trades	22
Losing Trades	15
Win Rate (%)	59.46%
Largest Win ($)	319.12
Average Win ($)	82.67
Largest Loss ($)	-75.81
Average Loss ($)	-55.57
Winning Streak	7
Losing Streak	3

2. Performance Ratios & Benchmarks

Metrics	Value
Sharpe Ratio	**5.96**
Sortino Ratio	15.15
Benchmark Return (%)	157.08%
Benchmark Profit on $1000 ($)	1570.81
Strategy Profit (%)	131.11%
Max Portfolio Balance ($)	2419.15
Min Portfolio Balance ($)	1000.00

3. Risk & Drawdown

Metrics	Value
Maximum Drawdown (Static) (%)	**13.38%**
Average Drawdown (Static) (%)	3.93%
Maximum Drawdown (Compound)(%)	**20.61%**
Average Drawdown (Compound) (%)	6.60%
Max Adverse Excursion (%)	**11.98%**
Avg. Adverse Excursion (%)	4.23%

4. Time Based Statistics

Metrics	Value
Max Holding Time	52 days
Avg. Holding Time	9 days
Time to Recovery (Max)	**41.38 days**
Avg. Time to Recovery	16.85 days

5. Profitably Metrics

Metrics	Value
Gross Profit ($)	1040.82
Net Profit ($)	985.32
Average Profit per Trade ($)	26.63
Final Balance ($)	2311.13

3. Strategy 2 - Enhanced Technical Strategy

3.1. Overview

This strategy is a technically-driven trading algorithm that combines widely-used momentum and volatility indicators with disciplined entry, exit, and risk management rules. It aims to enhance profitability by applying stricter entry conditions, wider stop-loss ranges, simplified exit logic, and optional reversal handling.

Indicator	Purpose	Why It Matters
EMA 50 & EMA 200	Identify long-term vs short-term trend	- EMA 50 > EMA 200 → Bullish (buy bias) - EMA 50 < EMA 200 → Bearish (sell bias) - Prevents trades against the major trend
Bollinger Bands (20, 2 std)	Measure volatility; detect range vs breakout	- Tight bands → Sideways market (avoid) - Expanding bands with breakout → High probability trend - BB width filter avoids fakeouts
VWAP	Institutional fair price level	- Longs only above VWAP → buyer strength - Shorts only below VWAP → seller dominance - Avoids choppy entries near VWAP
RSI (14)	Gauge momentum & OB/OS levels	- RSI ≥ 55 → favor longs - RSI ≤ 45 → favor shorts - Avoids 50 ± 5 zone (low momentum)
ADX (14)	Detect strength of trend (not direction)	- ADX > 28 → strong trend (enter) - ADX < 20–25 → avoid (sideways market) - Confirms valid directional moves
ATR (14)	Set dynamic SL/TP/TSL based on volatility	- Avoids tight fixed SL/TP in volatile markets - Risk management adapts to volatility - SL = Entry - (X × ATR) - TP = Entry + (Y × ATR) - TSL adjusts with volatility

Table 3.1.1 : Indicators and their purpose for this Strategy

Fig 3.1.1 : Bollinger Bands

Fig 3.1.2 : ADX

Fig 3.1.3 : VWAP

Fig 3.1.4 : ATR

Fig 3.1.5 : ETH Signals of Our Strategy

Fig 3.1.6 : BTC signals by our stratergy

3.1.1. Entry Conditions

Long Entry

1. **EMA 50 > EMA 200 – Bullish Trend Confirmation**
 Enter only when short-term momentum is above long-term trend, confirming overall market strength.

2. **Close > VWAP – Price Above Fair Value**
 Indicates buying pressure and that the market is favoring higher prices.

3. **Bollinger Band Width > 0.015 – Avoid Sideways Market**
 Confirms sufficient volatility to support momentum-based entries.

4. **ADX > 28 – Strong Trend Strength**
 Filters out weak or directionless price movements by ensuring a robust trend.

5. **RSI ≥ 55 – Confirming Upward Momentum**
 Signals that bullish momentum has started but is not yet overextended.

6. **+DI > -DI – Bullish Directional Strength**
 Shows upward price movements are dominating, reinforcing a bullish bias.

Short Entry

1. **EMA 50 < EMA 200 – Bearish Trend Confirmation**
 Validates a downtrend where short trades align with the dominant market direction.

Type	Indicator(s)	Role
Trend Filter	EMA 50 > EMA :	Confirms market is in uptrend
Momentum	RSI ≥ 55, +DI > ·	Ensures buying strength & momentum
Volatility	Bollinger Band V	Avoids flat/sideways market
Strength	ADX > 28	Confirms there's a powerful trend, not just noise
Fair Value	Close > VWAP	Ensures you're trading above institutional fair price (buyers lead)

Fig 3.1.1.1 : Indicators for Long Entry Conditions

2. **Close < VWAP – Below Fair Price**
   Indicates selling pressure and potential further downside.

3. **Bollinger Band Width > 0.015 – Enough Volatility**
   Ensures the market isn't consolidating and has room for price moves.

4. **ADX > 28 – Strong Trend Strength**
   Supports entries only when the trend has sufficient strength.

5. **RSI ≤ 45 – Bearish Momentum**
   Signals early bearish momentum without being overly oversold.

6. **-DI > +DI – Bearish Directional Strength**
   Reinforces the bearish trend with dominant downward price movement.

Type	Indicator(s)	Purpose
Trend Filter	EMA 50 < EMA :	Confirms bearish structure
Fair Price	Close < VWAP	Below institutional average — sellers control
Volatility	BB Width > 0.01	Avoids false signals in low volatility
Momentum	RSI ≤ 45	Confirms downside momentum
Trend Strength	ADX > 28	Ensures price is strongly trending, not choppy
Direction	-DI > +DI	Bears dominate — downward price action is stronger

Fig 3.1.1.2 : Indicators for Short Entry conditions

3.1.2. Exit Conditions (For Both LONG & SHORT)

1. **Stop Loss = 2.5 × ATR**
   Adapts to market volatility and limits downside while avoiding tight stopouts in

choppy markets.

2. **Take Profit = 3.0 × ATR**
   Locks in meaningful gains with a favorable risk-reward ratio.

3. **Trailing Stop Loss = 1.5 × ATR (Dynamic)**
   Protects profits while allowing room for further trend development.

4. **Trend Weakness Exit – EMA 50 Crosses EMA 200**
   Exits the trade if the underlying trend structure changes direction.

5. **No Direct Reversals**
   Requires positions to be closed before taking the opposite side, reducing noise-driven flips.

Limitations & Notes

- Requires minimum 200 candles of data
- Sensitive to illiquid or low-volume assets
- All indicators are calculated using pandas_ta

Exit Rule	Type	Why It's Important
2.5× ATR	Stop Loss	Protects against large losses, adapts to volatility
3.0× ATR	Take Profit	Locks in healthy gains with decent RRR
Trailing Stop (TSL)	Profit Locking	Lets profits run while protecting gains
EMA Cross (Trend Char	Trend Invalidatio	Ensures you're only in valid directional setups
No Reversals	Discipline Rule	Prevents emotional trades and overtrading

Fig 3.1.2.1 : Indicators for Exit Conditions

3.2. Conclusion

This strategy strikes a balance between risk management and trend-following precision. The use of dynamic SL/TP via ATR, combined with ADX & RSI filtering, aims to improve profitability while reducing false signals in weak markets. Ideal for traders who prefer structured rule-based systems.

3.3. Backtesting Results

1. On BTC Market:

Fig 3.3.1 : Backtesting Results on BTC Market

2. On ETH Market:

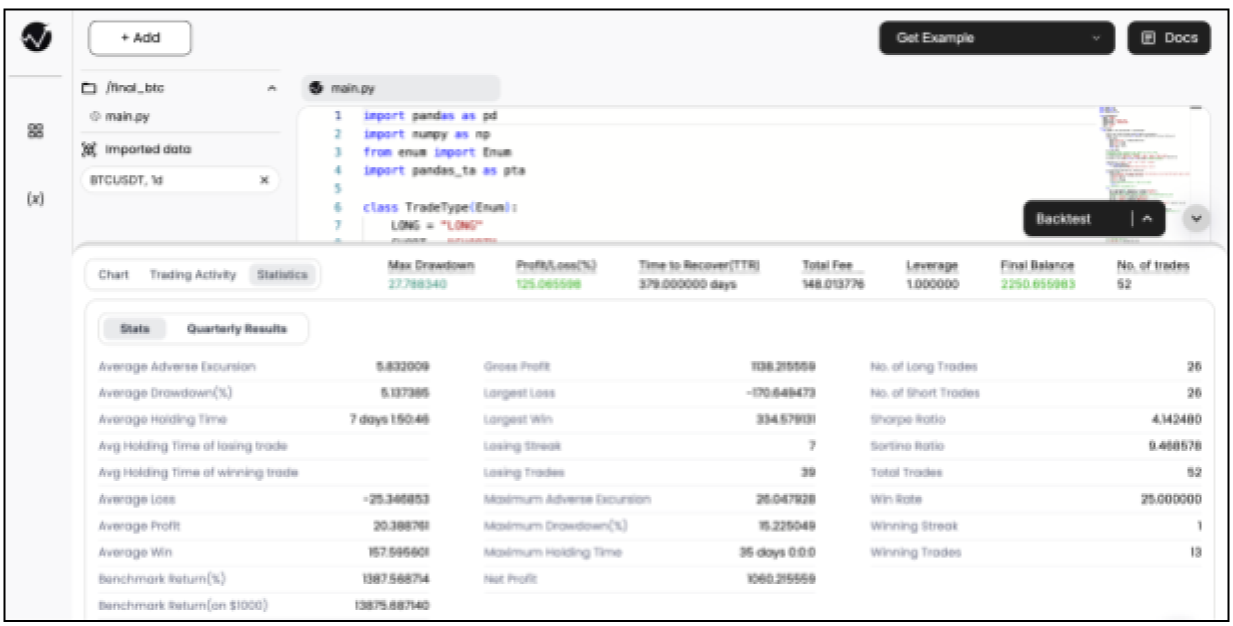

Fig 3.3.2 : Backtesting Results on ETH Market

4. Strategy 3 - ML Enhanced Hybrid BTC Strategy

4.1. Overview

This Bitcoin (BTC) trading strategy combines traditional technical analysis with machine learning to capture medium-term reversals and trend continuations. It starts by calculating key indicators such as SMA, EMA, MACD, RSI, Bollinger Band Width, ATR, and VWAP to evaluate trend direction, momentum, volatility, and key price levels. These indicators act as rule-based filters to ensure trades align with favorable market conditions.

The strategy's core innovation lies in its use of machine learning models—Random Forests and Decision Trees—to classify potential local tops and bottoms. A Neural Network further analyzes recent price differences to predict directional momentum, refining trade signals. Entry and exit decisions require alignment between the ML models and are supported by dynamic risk controls like ATR-based stop-loss, take-profit, and trailing stops. The system trades both long and short while avoiding whipsaws and emotional reversals, offering a disciplined, data-driven approach ideal for swing and medium-term trading.

4.2. Indicators Used and Their Purpose

Indicators	Purpose
Exponential Moving Averages (EMA)	Similar to SMAs but gives more weight to recent prices for quicker momentum detection.
Simple Moving Averages (SMA)	Used to identify short- and long-term trend direction by smoothing price over fixed windows.
Relative Strength Index (RSI)	Measures price momentum and signals overbought or oversold conditions to detect potential reversals.
MACD	Captures trend strength and direction through the difference between two EMAs.
Bollinger Bands Width	Quantifies market volatility by measuring the spread between upper and lower Bollinger Bands.
Average True Range (ATR)	Assesses market volatility and is used to dynamically set stop-loss and trailing stops.
Stochastic Oscillator	Detects potential reversals by comparing the closing price to its recent high-low range
VWAP (Volume Weighted Average Price)	Represents a fair value price based on volume, used to gauge institutional sentiment and entry zones.
Price Momentum Features	Custom features like diff and rolling returns are used to capture recent price strength and direction.

Table 4.2.1.1 : Indicators and their Purpose

4.3. Measures Taken to Prevent Overfitting

4.3.1. Data Cleaning Before Training

Before training any machine learning models, the strategy performs rigorous data cleaning. It removes rows that contain missing or infinite values in critical features. This ensures that the models are trained only on reliable, complete data and aren't influenced by anomalies or corrupted values. Clean data reduces noise and helps the model learn actual patterns instead of misleading ones caused by data artifacts.

4.3.2. Minimum Data Threshold Enforcement

To avoid learning from inadequate samples, the strategy checks whether there's a sufficient amount of clean data available. If the number of valid rows is too low (less than 50), the model training is skipped altogether. This prevents the model from overfitting to small, non-representative datasets and ensures it has enough historical context to make meaningful inferences.

4.3.3. Model Consensus Requirement (Ensemble Filter)

Rather than relying on a single prediction, the strategy uses both a Random Forest and a Decision Tree classifier to detect trading signals. It only acts on a signal when both models agree - a technique that acts like a built-in validation layer. This ensemble filter helps suppress false positives and provides more robust signal confirmation, reducing the chance of making decisions based on a biased or overfitted model.

4.3.4. Fallback Logic for Neural Network Predictions

The strategy includes intelligent fallback logic in case the neural network isn't fully initialized or its predictions fail. In such cases, it switches to a simpler, rule-based method - such as checking whether the majority of recent candles were positive or negative. This ensures the strategy continues functioning reliably, even when the neural network isn't ready or fails due to instability, and avoids over-reliance on a potentially overfit model.

4.3.5. Diverse Feature Set

A strong variety of technical indicators are used as model inputs - including momentum indicators like RSI and MACD, volatility measures like ATR and Bollinger Band Width, trend indicators like SMA and EMA, and volume-based features. This diversity allows the model to learn broader and more generalized market behaviors, reducing the risk that it fits too closely to any one specific condition or regime.

4.3.6. Trend Confirmation via SMA Crossovers

Trade entries are not taken blindly based on model predictions. Instead, the strategy checks whether shorter-term SMAs are above longer-term ones (e.g., `sma_10 > sma_30`) to confirm that the predicted signal aligns with the overall trend. This additional filter prevents the model from taking trades during choppy or range-bound periods and helps avoid signals that may be technically correct but poorly timed.

4.3.7. Behavior-Based Exit Validation

When considering an exit from a position, the strategy doesn't rely on a single factor. It checks for supportive behavior in recent candles - such as how many of the last 10 were bullish or bearish. This guards against exiting trades based on short-lived volatility or random fluctuations and ensures that exits are backed by consistent price behavior.

4.4 Strategy Workflow

- **Data Loading**
  We start by pulling historical cryptocurrency data from a CSV file, preparing it for analysis.

- **Feature Engineering**
  The script adds technical indicators like RSI, EMA, MACD, and candlestick patterns (e.g., Hammer, Hanging Man) to help understand price behavior.

- **Signal Generation**
  Based on these indicators, we label potential buy and sell zones using logic and thresholds.

- **ML Models & Neural Networks**
  We train a Random Forest and a Neural Network on the labeled data to predict trading signals.

- **Trade Execution**
  Using model predictions, we simulate trade entries and exits, applying stop loss and take profit logic.

- **Result Visualization**
  The performance of the strategy is visualized through metrics, equity curves, and charts showing trade points.

- **Backtesting Summary**
  Finally, we get a comprehensive breakdown of profit/loss, win rate, drawdowns, and risk-adjusted returns.

4.5. Backtesting Results

Start	0
End	37807
Period	1575 days 08:00:00
Start Value	100.0
End Value	7822114795.530767
Total Return [%]	7822114695.530767
Benchmark Return [%]	325.035
Max Gross Exposure [%]	100.0
Total Fees Paid	397366763.306167
Max Drawdown [%]	32.056545
Max Drawdown Duration	37 days 14:00:00
Total Trades	342
Total Closed Trades	341
Total Open Trades	1
Open Trade PnL	-7822114.79553
Win Rate [%]	85.043988
Best Trade [%]	67.458372
Worst Trade [%]	-17.110564
Avg Winning Trade [%]	7.586494
Avg Losing Trade [%]	-4.204339
Avg Winning Trade Duration	3 days 12:31:26.896551724
Avg Losing Trade Duration	4 days 15:25:52.941176470
Profit Factor	20.983691
Expectancy	22961691.525884
Sharpe Ratio	7.615933
Calmar Ratio	207.230644

Fig 4.5.1 : Backtesting Results

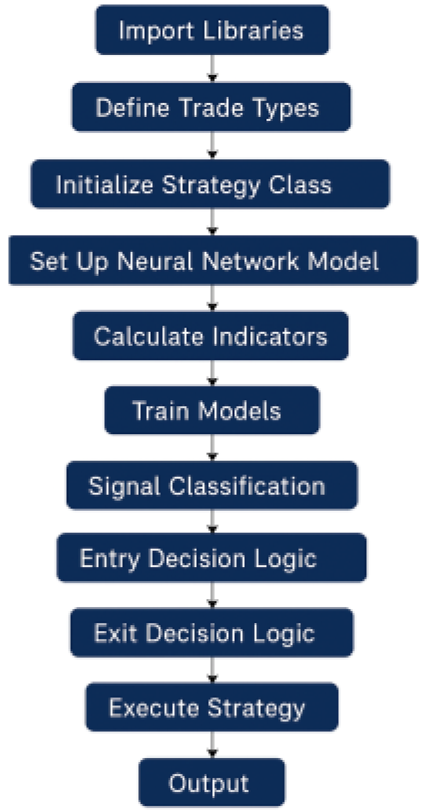

Fig 4.5.2. : Workflow Diagram

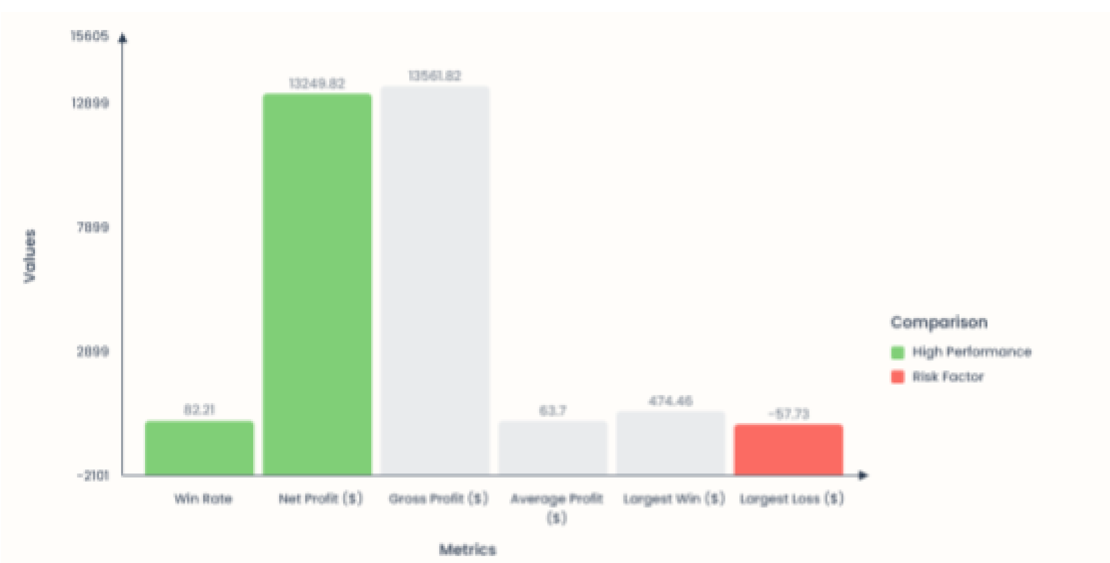

Fig 4.5.3. : Results

5. Strategy 4 - BTC-ETH Regime-Adaptive Pair Strategy

5.1 Solution

We designed a **trading strategy** that dynamically adapts to different market regimes using a combination of **technical indicators**, **statistical features**, and **volatility filters**. Our approach integrates **signal generation**, **risk management**, and **position control** tailored to BTC/USDT and ETH/USDT hourly data.

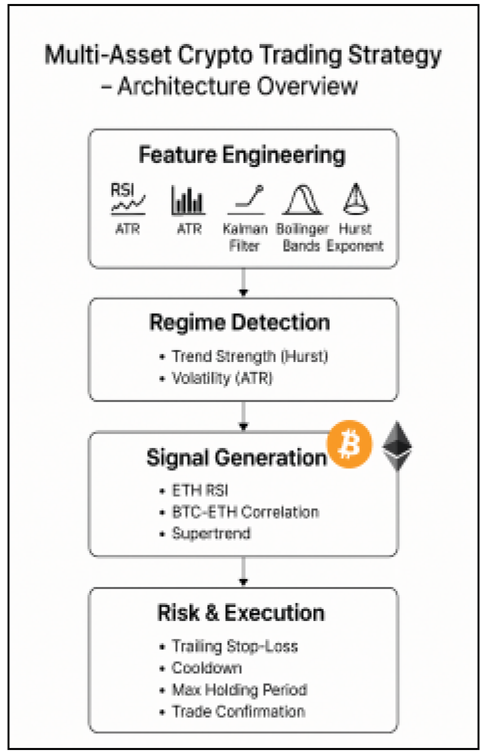

Fig 5.1.1 : Architecture Overview

5.2. CODE EXPLANATION:-

5.2.1. Code Structure & Components

1. Key Libraries

- pandas/numpy: Data manipulation
- hurst: Calculates Hurst exponent for trend detection
- pykalman: Applies Kalman Filter for price smoothing
- untrade.client: Backtesting SDK

2. Core Functions

FUNCTION	PURPOSE
process_data()	Loads, merges, and calculates indicators for BTC/ETH data
strat()	Implements trading logic and generates signals
perform_backtest()	Executes backtest using untrade SDK
perform_backtest_large_csv()	Handles large datasets via chunked processing
main()	Orchestrates execution flow

Table 5.2.1 : Functions and their purpose

5.2.2 Data Processing & Indicators

1. Data Sources

- **BTC Data**: 1-hour candles from BTC_2019_2023_1h.csv
- **ETH Data**: 1-hour candles from ETHUSDT_1h.csv

2. Key Technical Indicators

INDICATORS	PURPOSE
RSI	Measures momentum to identify overbought/oversold conditions.
ATR	Assesses volatility for stop-loss and entry/exit decisions.
Kalman Filter	Smooths price data for better trend identification.
Correlation	Evaluates BTC-ETH price relationship to identify trading regions.
Hurst Exponent	Detects market trends or mean-reverting behavior
Bollinger Bands	Highlights price volatility and potential reversal points.
Supertrend	Provides trend-following buy/sell signals.
CUSUM	Detects shifts in price trends using cumulative sums of deviations.
Regimes	Classifies market conditions as bullish or bearish for signal refinement.

Table 5.3.1 : Indicators and their Purpose

3. Feature Engineering Pipeline

- Merge BTC and ETH datasets based on timestamp.
- Calculate all relevant indicators for both assets.
- Filter the dataset to include only the 2020–2023 time frame.
- Rename columns and structure data to be compatible with the backtesting engine.

5.3. Trading Strategy Logic

5.3.1 Core Principles

- Pair Trading Basis: Leverages BTC-ETH correlation (>0.6 threshold)

- Regime Filtering: Requires trending markets (Hurst > 0.5)

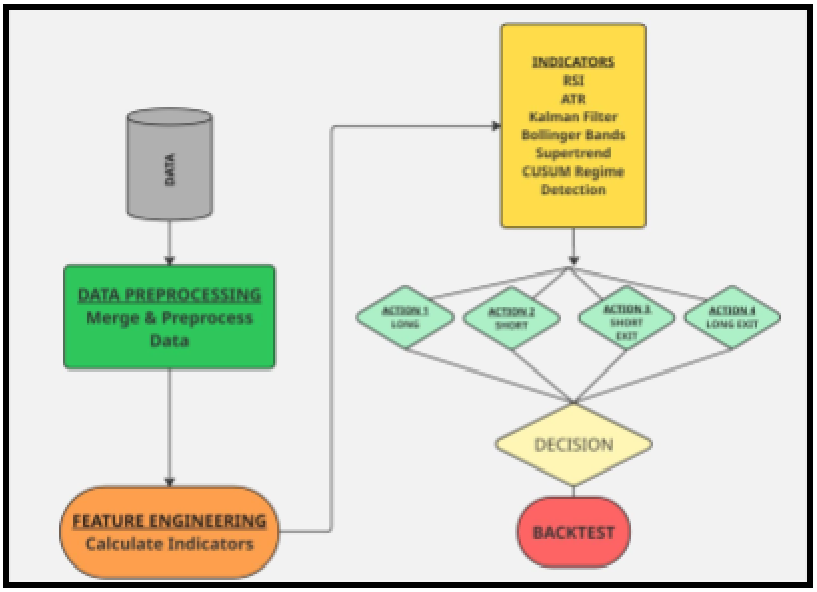

Fig 5.3.2.1 : Strategy Workflow

- Volatility Control: Uses BTC ATR for trade eligibility checks
- Multi-Indicator Confirmation: Requires alignment of RSI, Supertrend, and Bollinger Bands

5.3.2 Entry Conditions

Long Entry (ETH):

- BTC RSI > 70 (overbought)
- BTC in bullish regime
- BTC price above Bollinger midpoint
- ETH Supertrend indicates uptrend

Short Entry (ETH):

- BTC RSI < 30 (oversold)
- BTC in bearish regime
- BTC price below Bollinger lower band
- ETH Supertrend indicates downtrend

5.3.3 Exit Conditions

Long Exit:

- BTC RSI < 30
- ETH RSI decreasing

- BTC enters bearish regime
- ETH Supertrend reverses

Short Exit:

- BTC RSI > 70
- ETH RSI increasing
- BTC enters bullish regime

Fig 5.3.3.1 : Quarterly Performance

- ETH Supertrend reverses

5.3.4 Risk Management

Mechanism	Implementation
Trailing Stop-Loss	10% from peak/valley since entry
Volatility Stop	Force exit if BTC ATR > 2.5%
Time Stop	Max holding period: 28 days (672 hours)
Cooldown Period	24-hour pause after stop-loss trigger

Table 5.3.4.1 : Stop Loss Table

5.4 Backtesting Results

Metrics	**Value**
Benchmark Return	+1,687.59%
Strategy Net Profit	+5,242.45%
Total Profit Percentage	+7,684.52%
Max Drawdown (Portfolio)	**17.14%**
Max Drawdown (Single Trade)	8.16%
Sharpe Ratio	**6.02**
Sortino Ratio	19.68
Total Number of Trades	**162**
Win Rate	47.53%
Avg Win / Avg Loss Ratio	2.87:1
Long vs Short Ratio	91:71 (56% Long, 44% Short)
Total Trading Duration	376 days
Average Holding Time	6 days, 2h 47m
Maximum Holding Time	28 days

6. Risk Management Report: Strategy Assessment

6.1. Stop Loss (SL) & Take Profit (TP)

- **Stop Loss (SL):**
  Set at **2.5× ATR** to account for natural market volatility. This helps prevent early exits from normal fluctuations but may allow bigger losses if the trade fails.

- **Take Profit (TP):**
  Set at **3.0× ATR**, targeting a slightly higher gain than the potential loss, giving a **reward-to-risk ratio of ~1.2:1**.

ATR-based SL/TP dynamically adapts to market conditions, though a higher TP multiple may improve profitability.

6.2. Trailing Stop Loss (TSL)

- The TSL adjusts **only when a better stop level is found** in the direction of the trade.

- Helps **secure profits** while allowing trades to run longer during trends.

An effective mechanism to lock gains and minimize losses during reversals.

6.3. Position Management

- Only **one active position** allowed at a time (LONG, SHORT, or HOLD).

- Avoids rapid flipping between positions, reducing overtrading and noise-driven decisions.

Conservative strategy control; avoids unnecessary exposure and psychological trading traps.

6.4. Exit Conditions

- Trades are excited if:

  - Price hits **SL**, **TP**, or **Trailing Stop**

  - **EMA-50/EMA-200 crossover** indicates a trend reversal

Having multiple exit triggers helps protect against extended losses and avoids overstaying in weak trades.

6.5. Entry & Signal Filtering

- Entry conditions depend on:
    - **ADX & DI+/DI−** for trend strength
    - **RSI** for momentum and reversal signals
    - **Bollinger Band width** for volatility
    - **VWAP** for institutional price anchoring

Multi-indicator filtering reduces false entries in sideways/choppy markets.

6.6 Core Risk Management Indicators

Component	Role in Strategy	Risk Control Benefit
ATR (Average True Range)	Measures volatility	Dynamically adjusts SL/TP based on market conditions
RSI (Relative Strength Index)	Detects overbought/oversold zones	Prevents entries at extremes
Kalman Filter	Reduces market noise	Smoother trend estimates; fewer false signals
Bollinger Bands	Detects volatility boundaries	Entry/exit confirmation during breakouts/reversals
Supertrend + Custom Labels	Detects market regime (bullish/bearish/neutral)	Helps adapt strategy to market phase
CUSUM Filter	Detects anomalies and shifts	Identifies abnormal behavior and stops trading
Hurst Exponent	Reveals trending vs mean-reverting nature	Aligns strategy style with market structure
BTC-ETH Correlation	Monitors cross-asset exposure	Avoids doubling risk in correlated conditions

Table 6.6.1 : Indicators and their role in strategy

6.7 Practical Risk Mitigation Scenarios

Scenario	Indicator(s)	Action Taken
Market volatility spikes	ATR	Increase SL/TP distance or reduce position size
Sudden market shifts	CUSUM	Exit trades or pause new entries temporarily
Overheated/oversold zones	RSI + Bollinger Bands	Avoid entries or take early profits
Weak/no trend	Supertrend + Hurst Exponent	Avoid trend-following trades
High BTC-ETH correlation	Correlation Coefficient	Lower exposure or diversify into uncorrelated assets

Table 6.7.1 : Indicators and their scenario