**-Jaideep**

**Options Trading Strategy Backtest System – How It Works**

**1. Project Overview**

This project is a comprehensive Python-based backtesting system for options trading strategies, specifically tailored for Indian equity options (NIFTY). It combines technical indicators, machine learning (ML), and systematic risk management to simulate and analyze trading performance using historical data. The system is modular, allowing users to test, analyze, and optimize both individual components and the complete strategy.

**2. Project Structure & Components**

backtest.py: The main engine that orchestrates the backtest, manages trades, and generates results.

indicators.py: Implements technical indicators such as MACD, RSI, SuperTrend, EMA crossover, and Bollinger Bands.

model.py: Contains ML models (XGBoost, Random Forest, Logistic Regression) for signal enhancement.

signal\_engine.py: Combines signals from various sources and applies custom weights to generate composite trading signals.

utils.py: Handles data loading, preprocessing, and utility functions.

data/: Contains historical spot and options data in CSV format.

results/: Stores output files like equity curves, drawdown charts, trade logs, and performance metrics.

**3. Workflow & Data Flow**

A. Data Preparation

The user places historical spot and options data CSV files in the data/ folder.

utils.py loads and preprocesses this data, ensuring it contains all required fields (OHLC prices, in-house signals, expiry info for spot; full options chain for options).

B. Signal Generation

In-house signals (precomputed Buy/Sell/Hold) are loaded from the spot data.

Technical indicators are computed using indicators.py. Each indicator (MACD, RSI, SuperTrend, EMA crossover, Bollinger Bands) is calculated with standard parameters and produces its own Buy/Sell signals.

Signal weighting: The SignalEngine combines these signals using customizable weights (e.g., SuperTrend 20%, MACD 15%, etc.), producing a composite signal for each time step.

C. Machine Learning Enhancement

The ML model (default: XGBoost) is trained on historical data, using features like price momentum, indicator values, and time-based features.

The model predicts the next-period price direction. If the model’s confidence exceeds a threshold (e.g., 70%), its prediction can override the composite signal.

The ML model can be retrained, evaluated, and saved for future use.

D. Backtesting Logic

The OptionsBacktester class in backtest.py simulates trades based on generated signals.

Trade Entry: On a Buy signal, the system sells an ATM (At-The-Money) PUT option; on a Sell signal, it sells an ATM CALL option. The closest expiry is selected.

Risk Management: Each trade is managed with stop loss, take profit, and EOD (end-of-day) exit rules. Position sizing, max concurrent trades, and capital allocation are enforced.

Trade Exit: Positions are closed when stop loss or take profit is hit, or at a specified EOD time (e.g., 15:15 IST).

E. Results & Analysis

After the backtest, results are saved in the results/ folder:

ometrics.csv: Key performance metrics (total return, Sharpe ratio, drawdown, win rate, etc.).

oequity\_curve.png: Portfolio value over time.

odrawdown.png: Visualizes periods of loss.

otrades.csv: Detailed trade log (entry/exit times, signals, premiums, P&L, exit reasons).

Users can analyze these outputs to assess strategy performance, risk, and robustness.

**4. Customization & Extensibility**

Signal Weights: Users can adjust the contribution of each indicator in signal\_engine.py to optimize for different market conditions.

Backtest Parameters: Capital, stop loss, take profit, and other risk controls can be customized in backtest.py.

ML Models: The system supports multiple ML algorithms; users can switch models or tune hyperparameters in model.py.

Component Testing: Individual modules (data loading, indicators, signal engine, ML model) can be tested independently using simple Python commands.

**5. Best Practices & Limitations**

Data Quality: The accuracy of results depends on the quality and completeness of input data.

No Slippage/Brokerage: The system assumes trades are executed at historical prices with no slippage or brokerage costs.

Liquidity Assumption: Assumes sufficient liquidity for all trades.

Validation: Users are encouraged to run out-of-sample tests and adjust parameters based on changing market conditions.

**6. How to Use**

1.Install dependencies:

pip install -r requirements.txt

2.Add data:

Place your spot and options CSV files in the data/ folder.

3.Run backtest:

python backtest.py

4.Analyze results:

Review output files in the results/ folder for performance and risk analysis.

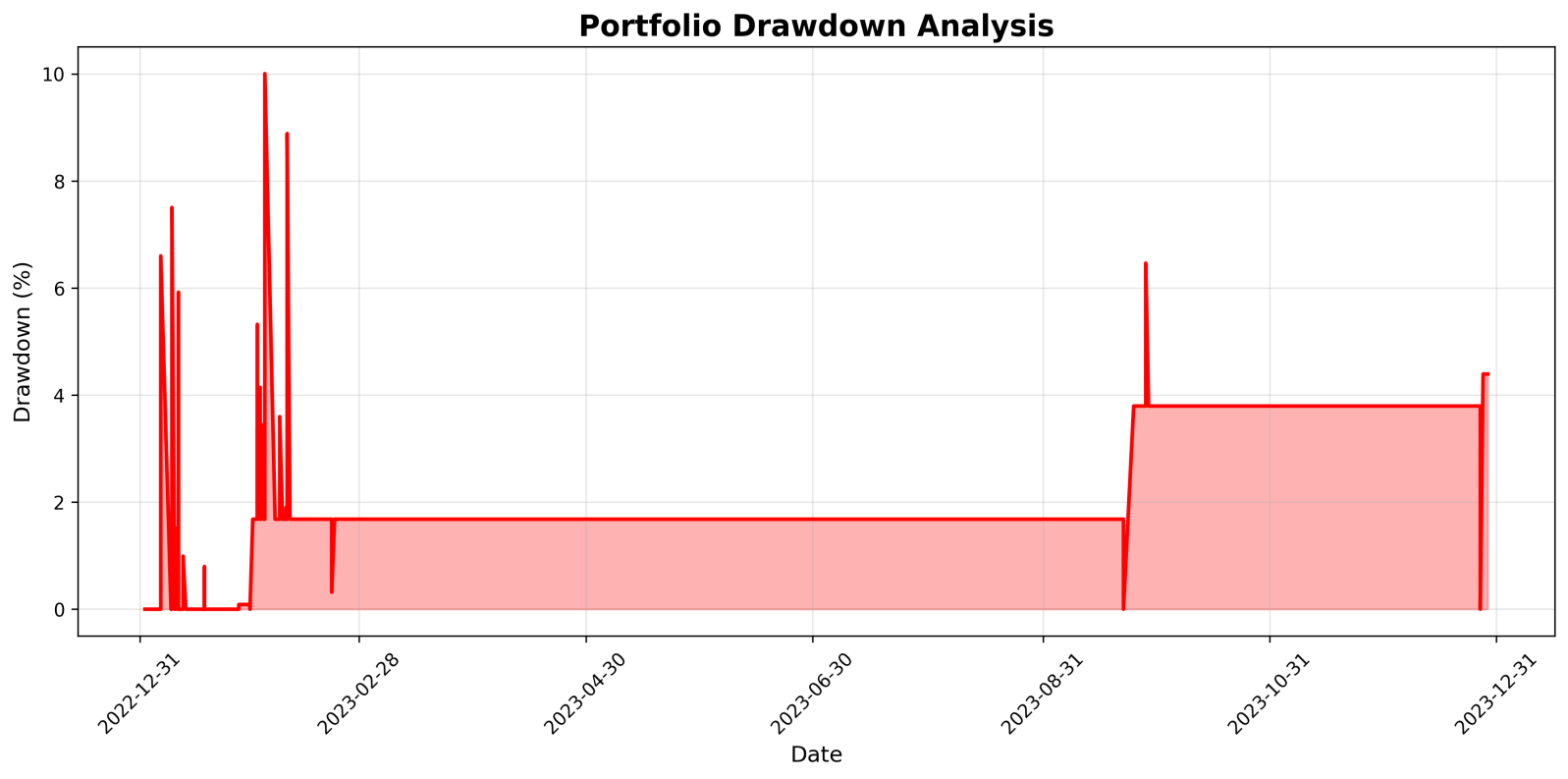
5.Customize:

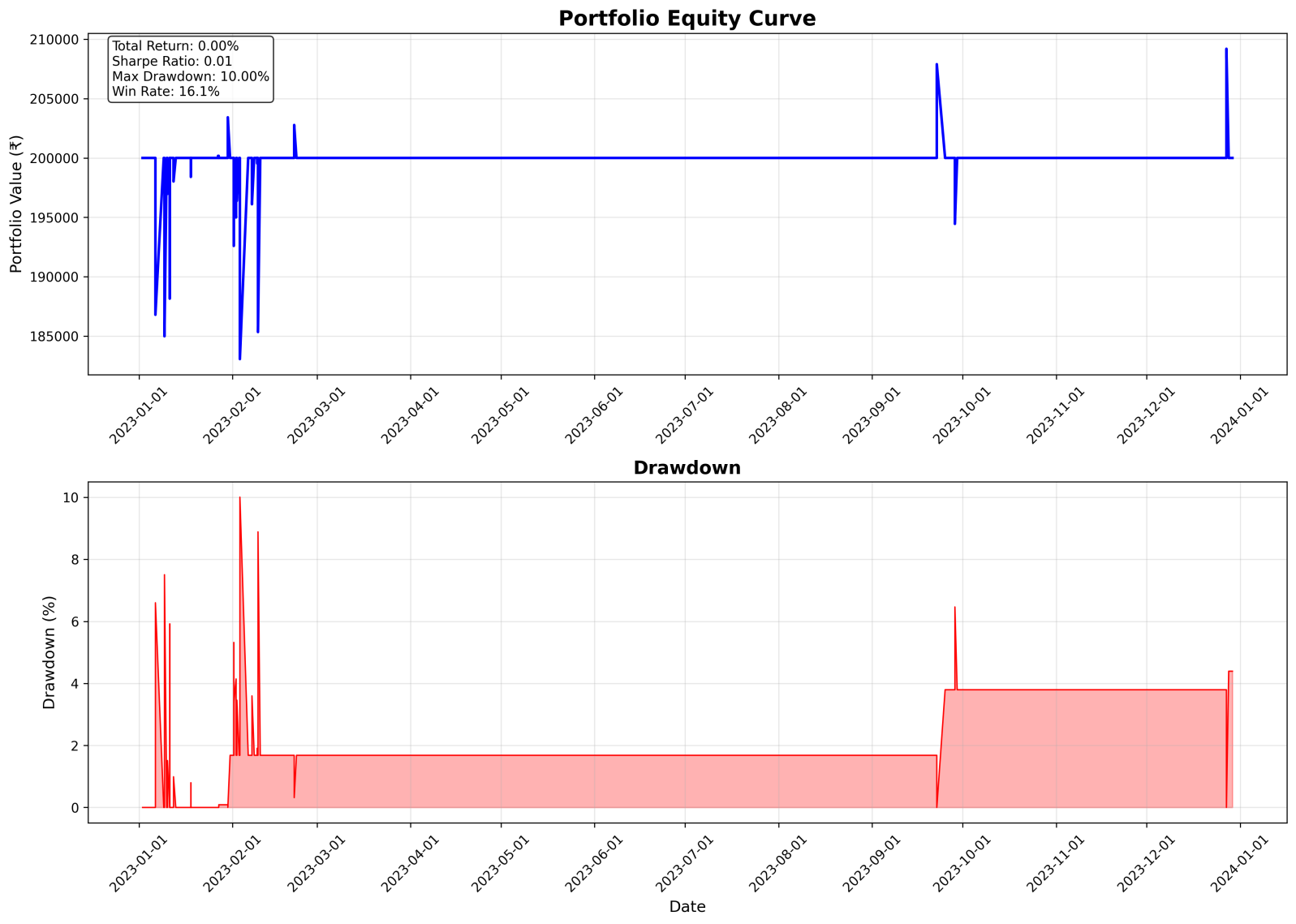
Adjust signal weights, backtest parameters, or ML models as needed.

**7. Conclusion**

This backtesting system provides a robust framework for developing, testing, and optimizing options trading strategies using both traditional technical analysis and modern machine learning. Its modular design, detailed analytics, and customization options make it suitable for both research and practical strategy development. Users can iterate quickly, analyze results, and adapt strategies to evolving market conditions—all within a single, well-documented Python project.

**Results**

****

****

|  |  |  |
| --- | --- | --- |
| Metric | Value |  |
| Initial Capital | 200000 |  |
| Final Capital | 200000 |  |
| Total Return % | 0 |  |
| Sharpe Ratio | 0.007414725441973054 |  |
| Max Drawdown % | 10.00492819702956 |  |
| Total Trades | 31 |  |
| Win Rate % | 16.129032258064516 |  |
| Profit Factor | 0.7207521284220209 |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| entry\_time | exit\_time | signal\_type | option\_type | strike\_price | entry\_premium | exit\_premium | quantity | pnl | exit\_reason |
| 2023-01-06 15:30:00+05:30 | 2023-01-09 09:20:00+05:30 | Buy | PE | 17850 | 49.8 | 115.8 | 200 | -13200 | SL |
| 2023-01-09 15:30:00+05:30 | 2023-01-10 09:20:00+05:30 | Sell | CE | 18100 | 206.25 | 518.95 | 48 | -15009.6 | SL |
| 2023-01-10 15:30:00+05:30 | 2023-01-11 09:20:00+05:30 | Sell | CE | 17900 | 350.55 | 458.6 | 28 | -3025.4 | SL |
| 2023-01-11 09:42:00+05:30 | 2023-01-11 09:43:00+05:30 | Buy | PE | 17900 | 59.35 | 129.85 | 168 | -11844 | SL |
| 2023-01-12 15:30:00+05:30 | 2023-01-13 09:20:00+05:30 | Sell | CE | 17850 | 423.5 | 509.65 | 23 | -1981.45 | SL |
| 2023-01-18 09:29:00+05:30 | 2023-01-18 09:30:00+05:30 | Sell | CE | 18050 | 307.5 | 357.4 | 32 | -1596.8 | SL |
| 2023-01-25 15:30:00+05:30 | 2023-01-27 15:15:00+05:30 | Sell | CE | 17900 | 458.6 | 458.6 | 21 | 0 | EOD |
| 2023-01-27 09:21:00+05:30 | 2023-01-27 15:15:00+05:30 | Sell | CE | 17900 | 518.9 | 509.65 | 19 | 175.75 | EOD |
| 2023-01-30 15:30:00+05:30 | 2023-01-31 09:20:00+05:30 | Buy | PE | 17700 | 117 | 76.75 | 85 | 3421.25 | TP |
| 2023-02-01 14:49:00+05:30 | 2023-02-01 14:50:00+05:30 | Sell | CE | 17900 | 518.9 | 782.5 | 19 | -5008.4 | SL |
| 2023-02-01 14:58:00+05:30 | 2023-02-01 14:59:00+05:30 | Buy | PE | 17500 | 87.15 | 137.6 | 114 | -5751.3 | SL |
| 2023-02-01 15:00:00+05:30 | 2023-02-01 15:01:00+05:30 | Buy | PE | 17500 | 87.15 | 137.6 | 114 | -5751.3 | SL |
| 2023-02-01 15:02:00+05:30 | 2023-02-01 15:03:00+05:30 | Buy | PE | 17500 | 87.15 | 122.95 | 114 | -4081.2 | SL |
| 2023-02-01 15:04:00+05:30 | 2023-02-01 15:05:00+05:30 | Buy | PE | 17500 | 87.15 | 137.6 | 114 | -5751.3 | SL |
| 2023-02-01 15:06:00+05:30 | 2023-02-01 15:07:00+05:30 | Buy | PE | 17500 | 87.15 | 152.1 | 114 | -7404.3 | SL |
| 2023-02-01 15:08:00+05:30 | 2023-02-01 15:09:00+05:30 | Buy | PE | 17600 | 119.5 | 152.1 | 83 | -2705.8 | SL |
| 2023-02-01 15:30:00+05:30 | 2023-02-02 09:20:00+05:30 | Sell | CE | 17900 | 518.9 | 700 | 19 | -3440.9 | SL |
| 2023-02-02 09:20:00+05:30 | 2023-02-02 09:21:00+05:30 | Sell | CE | 17900 | 518.9 | 782.5 | 19 | -5008.4 | SL |
| 2023-02-02 15:30:00+05:30 | 2023-02-03 09:20:00+05:30 | Buy | PE | 17600 | 119.5 | 163.05 | 83 | -3614.65 | SL |
| 2023-02-03 15:30:00+05:30 | 2023-02-06 09:20:00+05:30 | Sell | CE | 18500 | 262.3 | 707.85 | 38 | -16930.9 | SL |
| 2023-02-07 15:30:00+05:30 | 2023-02-08 09:20:00+05:30 | Buy | PE | 17500 | 135.1 | 187.75 | 74 | -3896.1 | SL |
| 2023-02-09 09:21:00+05:30 | 2023-02-09 09:22:00+05:30 | Buy | PE | 18000 | 252.25 | 264.1 | 39 | -462.15 | SL |
| 2023-02-09 15:30:00+05:30 | 2023-02-10 09:20:00+05:30 | Sell | CE | 18500 | 262.3 | 647.95 | 38 | -14654.7 | SL |
| 2023-02-21 15:30:00+05:30 | 2023-02-22 09:20:00+05:30 | Sell | CE | 17850 | 707.85 | 509.65 | 14 | 2774.8000000000006 | TP |
| 2023-02-23 15:30:00+05:30 | 2023-02-24 15:15:00+05:30 | Buy | PE | 17500 | 143.7 | 143.7 | 69 | 0 | EOD |
| 2023-03-24 15:30:00+05:30 | 2023-03-27 15:15:00+05:30 | Buy | PE | 17000 | 118.65 | 118.65 | 84 | 0 | EOD |
| 2023-03-27 15:30:00+05:30 | 2023-03-28 15:15:00+05:30 | Buy | PE | 17000 | 118.65 | 118.65 | 84 | 0 | EOD |
| 2023-03-28 15:30:00+05:30 | 2023-03-29 15:15:00+05:30 | Buy | PE | 17000 | 118.65 | 118.65 | 84 | 0 | EOD |
| 2023-09-22 15:30:00+05:30 | 2023-09-25 09:20:00+05:30 | Sell | CE | 19000 | 760 | 153 | 13 | 7891 | TP |
| 2023-09-28 15:30:00+05:30 | 2023-09-29 09:20:00+05:30 | Buy | PE | 19000 | 950.05 | 1505 | 10 | -5549.5 | SL |
| 2023-12-27 15:30:00+05:30 | 2023-12-28 09:20:00+05:30 | Sell | CE | 22000 | 90.55 | 7 | 110 | 9190.5 | TP |