

Moving Average Crossover – Quant Backtest & Optimization

A complete trading strategy implementation using Python, Differential Evolution, and a custom backtesting engine.

Overview

This project implements a Moving Average Crossover trading strategy that includes:

- Signal generation using EMA crossover
- A fully custom backtesting engine
- Parameter optimization using Differential Evolution
- Detailed performance metrics:
Sharpe Ratio, CAGR, Max Drawdown, Hit Ratio, Avg Profit/Loss, Total Trades
- Summary reports and visualizations

This project is part of my Quant Research Portfolio.

Project Structure

EMA_Crossover_DE_Optimize.py

trade_generator_BT.py

PerformanceAnalysis1.py

myutils.py

optimal_params.csv

EMA_Summary_Report.xlsx

RELIANCE.csv

README.pdf

requirements.txt

Strategy Logic

Buy: when $EMA_{fast} > EMA_{slow}$

Sell: when $EMA_{fast} < EMA_{slow}$

Supports long, short, and flat signals depending on configuration.

Optimization

Uses **Differential Evolution** (`scipy.optimize.differential_evolution`) to find the best:

- EMA Fast Period
- EMA Slow Period

Objective: **maximize Sharpe Ratio** on out-of-sample data.

Tech Stack

- Python
 - Pandas
 - NumPy
 - SciPy
 - Matplotlib
-

Author

Abhishek N

Quant Research | Python | Algorithmic Trading | CA

Email: nagarajabhishek3@gmail.com

LinkedIn: linkedin.com/in/ca-abhishek-n-47ba09242