



Stock Market Forecasting Project Documentation

Welcome to the **Stock Market Forecasting** project wiki. This documentation provides detailed insights into the purpose, structure, algorithms, and methodology used to analyze and forecast stock trends using Python.



Table of Contents

1. [Overview](#)
 2. [Setup Instructions](#)
 3. [Data Collection](#)
 4. [Data Preprocessing & Visualization](#)
 5. [Technical Indicators](#)
 6. [Trading Strategy](#)
 7. [Backtesting](#)
 8. [Performance Metrics](#)
 9. [Results & Insights](#)
 10. [Future Work](#)
-



Overview

This project uses historical stock data and technical analysis to:

- Identify buy/sell signals
- Simulate a trading strategy
- Compare it with a basic buy-and-hold approach

We use **moving averages**, **RSI**, and **MACD** to trigger decisions and **backtest** the results.



Setup Instructions

1. Clone the repository:

```
git clone https://github.com/your-username/your-repo-name.git
```

1. Install dependencies:

```
pip install -r requirements.txt
```

1. **Run the notebook:** Open `Stock market forecasting summary.ipynb` in Jupyter Notebook.

2. **Add your Alpha Vantage API key:**

```
API_KEY = "your_alpha_vantage_api_key"
```



Data Collection

We use the **Alpha Vantage API** to fetch daily historical stock prices:

- Function used: `TIME_SERIES_DAILY`
- Output formats: `compact` (100 days) or `full` (entire history)
- Stored as a DataFrame and saved as CSV



Data Preprocessing & Visualization

- Convert raw JSON into a pandas DataFrame
- Handle missing values
- Plot the historical stock price
- Visualize moving averages and closing prices



Technical Indicators

Implemented indicators include:

1. Simple Moving Averages (SMA)

- `SMA_20`, `SMA_50`, `SMA_100`, `SMA_200`

2. Exponential Moving Average (EMA)

- `EMA_50`: More weight on recent prices

3. Relative Strength Index (RSI)

- Detect overbought (>70) or oversold (<30) conditions

4. MACD (Moving Average Convergence Divergence)

- MACD, MACD Signal Line
 - Identifies bullish or bearish momentum
-



Trading Strategy

Basic Strategy:

- Buy: SMA_50 crosses above SMA_200
- Sell: SMA_50 crosses below SMA_200

Enhanced Strategy:

- Buy: SMA_20 > SMA_100 AND RSI < 30 AND MACD > Signal
- Sell: SMA_20 < SMA_100 AND RSI > 70 AND MACD < Signal

Signals are stored in Signal, Crossover, Improved_Buy, and Improved_Sell columns.



Backtesting

Simulates portfolio growth:

- Initial capital: \$10,000
 - Buys full position on a buy signal
 - Sells all shares on a sell signal
 - Tracks portfolio value vs. Buy & Hold strategy
-



Performance Metrics

- **Final Portfolio Value**
- **Total Return:** % gain from starting capital
- **Win Rate:** % of profitable trades
- **Max Drawdown:** Worst drop from peak value

Stored in DataFrame columns:

- Portfolio Value
 - Improved Portfolio Value
 - Buy & Hold Value
 - Drawdown
-



Results & Insights

- The optimized strategy shows superior returns vs. buy-and-hold.
 - Drawdowns and risks are significantly reduced using combined indicators.
 - Signal confirmation improves decision accuracy.
-



Future Work

- Integrate with other APIs (e.g., Yahoo Finance)
 - Add support for multiple stocks & batch processing
 - Deploy with Streamlit or Flask for a UI
 - Apply machine learning models for pattern recognition
-

Feel free to contribute or suggest improvements via pull requests or issues.

Happy Trading! 