

Q: Stock Analyzer

Abdullah Al Marzouq – Akash Beh
Rayane Tarazi – Stanley Gunawan

Form

Q works through a **terminal** and outputs the results of the **profit-maximizing** algorithm in a given **timeframe** as such:

```
class Distribution in dataset:
target:
0 56.56488
1 18.55883
-1 15.888237
Name: proportion, dtype: float64
Training on 3638 samples, testing on 910 samples.
Training SVM model with RBF kernel...
Model training complete.

=====
Model Evaluation Results
=====
Accuracy: 94.29%

Classification Report:
precision    recall  f1-score   support

Sell (-1)    0.98    0.79    0.87    137
Hold (0)     0.93    1.00    0.96    682
Buy (1)      0.95    0.88    0.91    171

accuracy     0.96    0.89    0.94    910
macro avg    0.96    0.89    0.92    910
weighted avg 0.95    0.94    0.94    910

=====
-----BACKTESTING WITH 10% AGAINST BAH-----
----- Running SVM Strategy Simulation -----
----- Running Buy and Hold Simulation -----
----- Running Raw SVC Signal Simulation -----

Detailed Results per Ticker (Top 10 by SVM Profit)
-----
strategy  SVM Model  Raw SVC Signal  Buy and Hold
-----
SOFI      4.24      0.00      14.64
SNAP      3.84      0.25      11.62
SQM       2.17      1.99      32.51
TV        2.15      0.00      13.98
RKT       2.02      0.00      6.69
TSLA      1.80      0.50      -4.32
UAMY      1.80      0.00      -25.85
YDU       0.51      0.00      -2.81
HSH       0.36      0.00      2.88
TSM       0.33      -0.28      0.94

----- Average (Portfolio) Results Across All Tickers -----
final_value  profit  total_return_pct  risk_pct  sharpe_ratio
strategy
SVM Model    599.22    $-8.77      -0.77%    8.25%    -0.82
Raw SVC Signal  599.08    $-8.92      -0.82%    8.43%    -0.87
Buy and Hold   598.40    $-1.51     -1.51%    3.83%    0.45

-----
ticker      strategy  final_value  profit  total_return_pct  risk_pct  sharpe_ratio
-----
0  HOLA  SVM Model    94.711129    5.248802    0.055609    0.01837    2.774869
1  RKT   SVM Model   102.822879    2.822879    0.028229    0.00552    2.922754
2  HSH   SVM Model   100.000000    0.000000    0.000000    0.00000    0.000000
3  RR    SVM Model    77.725871   -22.274129   -0.222741    0.02296   -5.685911
4  RS    SVM Model   100.000000    0.000000    0.000000    0.00000    0.000000
-----
184 XPEV Raw SVC Signal  100.000000    0.000000    0.000000    0.00000    0.000000
185 XTC  Raw SVC Signal  100.000000    0.000000    0.000000    0.00000    0.000000
186 XYZ  Raw SVC Signal  100.000000    0.000000    0.000000    0.00000    0.000000
187 YDU  Raw SVC Signal  100.000000    0.000000    0.000000    0.00000    0.000000
188 ZIM  Raw SVC Signal  100.000000    0.000000    0.000000    0.00000    0.000000
```

Function

Q is a system that continuously scans major social-media platforms to measure real-time sentiment for thousands of publicly traded stocks. Leveraging natural-language processing and LLMs, it assigns each stock a **sentiment score** and identifies the top-ranked tickers showing unusually **strong positive momentum**.

These high-sentiment stocks are then cross-validated through two external data sources:

- **Historical price and volume** data to confirm market trends and detect volatility patterns, and
- **Political insider-trading** records to highlight potential influence from government officials' trades.

By combining sentiment trends, market fundamentals, and insider-trading signals, the system produces a transparent, **data-driven buy/sell recommendation**.

The platform also includes a fully automated **Autopilot mode**, which can execute trades on the user's behalf based on the **profit-maximizing algorithm**. This allows hands-free investing where the system monitors, analyzes, and acts in real time without requiring manual input from the user.

Resources

Q analyzes real-time social-media data using an integrated LLM-powered sentiment engine, along with **Python libraries** such as **PySentimiento** and **Pandas**. After identifying high-sentiment stocks, it cross-references them with historical market-data **APIs** and political insider-trading **APIs** to generate data-driven buy/sell recommendations. The system also includes an Autopilot mode that automatically executes trades based on these recommendations. All development is done in **Python** through **VS Code**, and the entire platform is hosted on a private **Clovux** server for secure, continuous operation.



What's next?

Moving forward, our primary focus is refining and optimizing the algorithm to reach a **higher accuracy standard**, especially in filtering noise and improving the reliability of sentiment-driven stock predictions. We plan to **expand our data** sources, integrate more robust market indicators, and experiment with additional LLM models to improve real-time interpretation of social-media trends. Future iterations will also include more rigorous back-testing, **enhanced risk-management** controls, and a more adaptive Autopilot system that can tailor strategies to individual user profiles. As the system matures, we aim to scale performance, improve automation stability, and transition toward a **fully production-ready platform**.

