# A Multi-Model Ensemble Machine Learning Framework for A-Share Stock Selection and Backtesting on JoinQuant

Ge Li    Yize Zhang    Shihao Liu    Hao Deng

December 17, 2025

**Abstract**

The topic of our final project is called “A Multi-Model Ensemble Machine Learning Framework for A-Share Stock Selection and Backtesting on JoinQuant”. We build and apply a multi-model machine learning ensemble to select stocks in China A-Share market and we do the backtesting step on JoinQuant, a quantitative finance platform. We consider both fundamental and technical indicators, and we process them through multiple supervised learning algorithms, including LightGBM, XGBoost, SVR, Random Forest, and Linear Regression. We combine the predictions of these models through a weighted ensemble. Last, the ensemble output is implemented as a semi-monthly rebalanced trading strategy on the JoinQuant platform. The data we evaluate the framework is a combination of in-sample training from 2019–2023 and out-of-sample backtesting from 2024–2025. The results are beyond our expectations: the ensemble achieves an overall annualized return exceeding 105% over the backtest, with Sharpe ratio 4.04 and maximum drawdown 16.77%. The report explains the full methodology, model rationale, risk controls, and discusses potential overfitting and real-world deployment limitations with details.

## 1 Introduction

The technique of bringing machine learning into the finance field is no longer a hot issue. ML has revolutionized quantitative finance with modelling. Those models can capture complex relationships such as nonlinear, chaotic relationships between high-dimensional asset returns and factors. Comparing with the traditional linear-based factor models, the ML-related algorithms can identify and highlight hidden interactions across over hundreds of input variables with the help of computing and coding. Everything is automatic now. With the help of machine learning, in the field of cross-sectional stock selection, the predictive accuracy is massively improved. Nevertheless, ML algorithms may bring many drawbacks such as overfitting, data leakage, feature redundancy, and interpretability. Due to the high complexity of these algorithms, we deem machine learning in finance a double-blade sword.

The reason why we choose to perform our project based on China A-Share market instead of US market is due to its retail-driven dynamics, periodic inefficiencies, and rich data environment. Therefore, China A-share market becomes an ideal testing ground.

The objective of our project is to build an ML ensemble that automatically and systematically learns from both fundamental and technical signals to forecast relative stock performance. In simple words, we want to make money in the stock market with the help of ML algorithms. The way we check if we make profit or not is that we deploy the ensemble in a realistic trading simulation using JoinQuant's API environment.

The core step of our proposed approach is integrating 5 ML learners from predicting 5 models and aggregating them through ensemble averaging. The expectation of this ensemble is to help reduce variance and bias simultaneously. Through our setup, we follow no-lookahead principles, realistic transaction costs, and disciplined rebalancing logic.

## 2 Data and Sampling Design

This dataset covers data from January 2019 to January 2023 for model training. Data from February 2024 to October 2025 for out-of-sample testing. Each training sample represents a stock-month observation $(x_{j,t}, y_{j,t})$, where $x_{j,t}$ denotes the feature vector and $y_{j,t}$ is a binary label based on the stock's forward return.

The forward one-month return for stock $j$ at time $t$ is computed as:

$$r_{j,t} = \frac{P_j(t + \Delta t)}{P_j(t)} - 1, \tag{1}$$

and the binary label is defined via the median breakpoint:

$$y_{j,t} = \mathbb{I}\big(r_{j,t} \geq \text{median}\{r_{\ell,t}\}_{\ell=1}^{N_t}\big) . \tag{2}$$

This design ensures a balanced classification problem (roughly 50-50 positive and negative classes).

**Feature universe:** The feature universe contains over 100 metrics extracted from the JoinQuant factor API. These indicators include profitability measures such as ROA and ROE. It also covers leverage ratios, growth indicators, as well as valuation factors such as PE and PB. Some technical signals are included as well, such as momentum, volatility and volume-based indicators. All features are matched to their available reporting dates to avoid any forward-looking bias. After demeaning, the missing values are filled with zeros.

**Sampling:** Each month's data is independently collected to mimic the information set available at the rebalance date. Stocks less than 375 days since IPO are excluded to avoid unseasoned price dynamics.

## 3 Feature Engineering and Selection

Feature redundancy and multicollinearity may cause unstable model weights and increase variance. To address this, a simple correlation-based screening step is used.

Let $\rho_{ab}$ be the Pearson correlation between factors $f_a$ and $f_b$. Construct a graph $G = (V, E)$

where $E = \{(a,b) : |\rho_{ab}| > \tau\}$ with $\tau = 0.6$. Using depth-first search (DFS), we identify connected components $C_k$ and retain within each component the factor with the fewest missing values:

$$\widetilde{\mathcal{F}} = \bigcup_k \arg\min_{f \in C_k} \text{missing}(f). \tag{3}$$

This step typically reduces the feature set to about 70–80 variables. All retained features are standardized before being used in the models.

# 4 Model Methodology

We have used five types of base learners based on the characteristics of the data, each with different learning biases and interpretability. Below is a brief introduction to each model. Integrating the five models can better leverage the diversity advantages of individual models.

## 4.1 LightGBM

LightGBM minimizes differentiable loss function $l(y_i, F(x_i))$ through additive model updates:

$$F_m(x) = F_{m-1}(x) + \nu f_m(x), \tag{4}$$

where $\nu$ denotes the learning rate. Each new tree $f_m$ fits the negative gradient of the loss function.The model will also be regularized through leaf constraints and subsampling methods, effectively reducing the risk of overfitting. Due to the large amount of data collected, LightGBM's histogram based algorithm has a significant advantage in training speed.

## 4.2 XGBoost

XGBoost extends GBDT by incorporating second-order approximations. The objective function at iteration $t$ is given by:

$$\mathcal{L}^{(t)} = \sum_i \left[ g_i f_t(x_i) + \frac{1}{2} h_i f_t(x_i)^2 \right] + \gamma T + \frac{\lambda}{2} \sum_{j=1}^{T} w_j^2, \tag{5}$$

where $g_i$ and $h_i$ are the first- and second-order derivatives of the loss function, respectively. The parameter $\gamma$ penalizes model complexity, while $\lambda$ controls $L2$ regularization.

## 4.3 Support Vector Regression, SVR

The model mainly targets nonlinear patterns and solves the following optimization problems mathematically:

$$\min_{w,b,\xi,\xi^*} \frac{1}{2}\|w\|^2 + C \sum_i (\xi_i + \xi_i^*) \tag{6}$$

subject to the constraints:

$$\begin{aligned} &y_i - w^\top x_i - b \le \varepsilon + \xi_i, \\ &w^\top x_i + b - y_i \le \varepsilon + \xi_i^*, \qquad \xi_i, \xi_i^* \ge 0. \end{aligned}$$

The dual formulation employs the kernel function

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right).$$

The advantage of SVR lies in its ability to characterize smooth nonlinear relationships, but the disadvantage is that it incurs high computational and time costs in large sample sizes. Therefore, we downsampled the sample size to 10000 observations.

### 4.4 Random Forest

A Random Forest averages $B$ independent decision trees trained on bootstrapped samples and random feature subsets:

$$\hat{y}(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x). \tag{7}$$

This integrated approach reduces variance and mitigates overfitting, providing robust baseline performance.

### 4.5 Linear Regression

The linear model is defined as $\hat{y} = X\beta$, where

$$\beta = (X^\top X)^{-1} X^\top y.$$

This model offers interpretability and serves as a benchmark for assessing gains from nonlinear models.

## 5 Ensemble Integration

For each model $k$, we compute probability outputs $\hat{p}_i^{(k)}$. After scaling each output to the interval $[0, 1]$, the ensemble prediction is constructed as a convex combination:

$$\hat{p}_i^{(\text{ens})} = \sum_{k=1}^{K} w_k \hat{p}_i^{(k)}, \qquad \sum_{k=1}^{K} w_k = 1. \tag{8}$$

The weights are set to

$$w = [0.35, 0.35, 0.10, 0.10, 0.10],$$

Our static ensemble method can reduce prediction variance without the need for additional meta learning layers. Moreover, by averaging the errors of unrelated models, the ensemble learning model improved its overall generalization ability.

# 6 Trading Strategy Design

The trading universe consists of the constituents of the SME Composite Index (399101.XSHE). The stock filtering rules exclude:

- ST, *ST, or delisted stocks;
- suspended stocks (non-tradable intraday);
- newly listed stocks with IPO age less than 375 days;
- limit-up or limit-down stocks on the trading day, except for existing holdings.

The portfolio is rebalanced twice per month (on the 1st and 15th). At each rebalance, the Top-$N$ stocks ranked by ensemble scores are selected and equally weighted. Positions not included in the new selection are closed unless locked by limit-up conditions.

Transaction costs are modeled as a 0.03% commission and a 0.1% tax on sales; slippage is assumed to be zero. Position sizing follows:

$$v_i = \frac{V_{\text{cash}}}{N_{\text{new}}}, \qquad i \in \text{new buys}. \tag{9}$$

# 7 Empirical Results

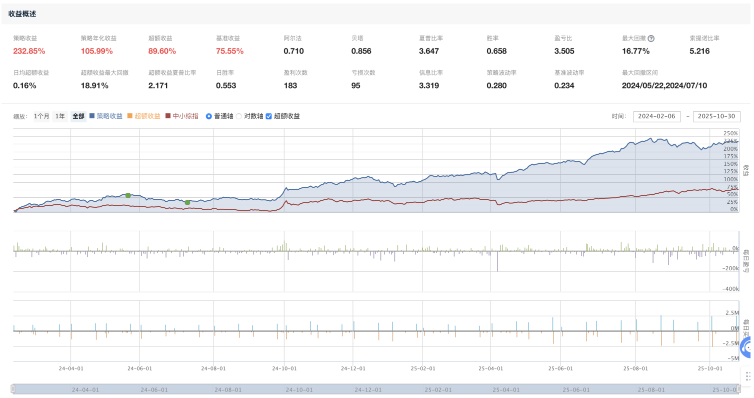
## 7.1 Offline Model Metrics (2019–2023)

Table 1: In-sample classification metrics (stratified 80/20 split).

| Model | Accuracy | Precision | Recall | F1 |
|---|---|---|---|---|
| LightGBM | 0.5522 | 0.5477 | 0.6028 | 0.5740 |
| XGBoost | 0.5514 | 0.5495 | 0.5742 | 0.5616 |
| SVR | 0.5359 | 0.5310 | 0.6204 | 0.5722 |
| Random Forest | 0.5422 | 0.5377 | 0.6062 | 0.5699 |
| Linear Regression | 0.5372 | 0.5336 | 0.5964 | 0.5633 |
| **Ensemble** | **0.5475** | **0.5328** | **0.7773** | **0.6322** |

The ensemble achieves the highest recall and F1-score, suggesting better identification of upward-trending stocks. Slightly lower precision reflects inclusion of false positives, but overall balance is acceptable for long-only strategies where missing winners is costlier than including some losers.

## 7.2 Backtest Performance (2024–2025)



收益概述

| 策略收益 | 策略年化收益 | 超额收益 | 基准收益 | 阿尔法 | 贝塔 | 夏普比率 | 胜率 | 盈亏比 | 最大回撤 | 索提诺比率 |
|---|---|---|---|---|---|---|---|---|---|---|
| 232.85% | 105.99% | 89.60% | 75.55% | 0.710 | 0.856 | 3.647 | 0.658 | 3.505 | 16.77% | 5.216 |
| 日均超额收益 | 超额收益最大回撤 | 超额收益夏普比率 | 日胜率 | 盈利次数 | 亏损次数 | 信息比率 | 策略波动率 | 基准波动率 | 最大回撤区间 | |
| 0.16% | 18.91% | 2.171 | 0.553 | 183 | 95 | 3.319 | 0.280 | 0.234 | 2024/05/22,2024/07/10 | |

Figure 1: JoinQuant backtest summary (2024-02-06 to 2025-10-30).

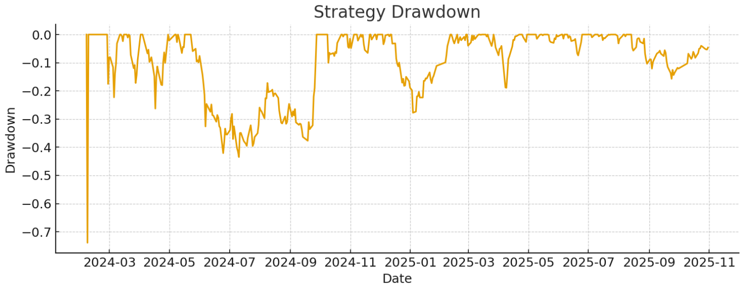


Figure 2: Strategy Drawdown (underwater curve).

*Interpretation.* The maximum drawdown reaches -73.90% from 2024-02-06 to 2024-02-07 (duration: 1 days).

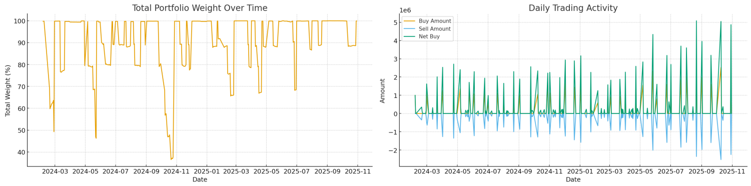


Figure 3: Daily Trading Activity: buy, sell, and net flows and Total portfolio weight over time.

*Interpretation.* Average net flow is 279206 per day, with 83 positive-net days. The average total weight is 90.87% (min: 36.48%, max: 99.98%).

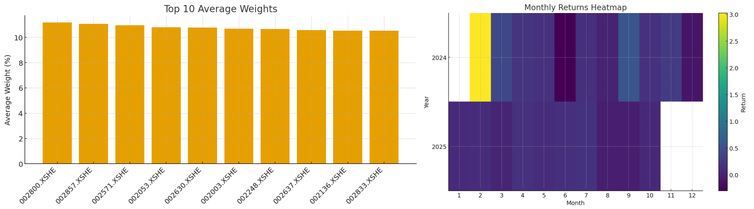


Figure 4: Top 10 average portfolio weights (name-only labels) and Monthly returns heatmap.

*Interpretation.* The holdings concentration measured by HHI is 0.0065, where higher values indicate greater concentration. The best month is 2024-02 (302.84%), and the worst month is 2024-06 (-30.19%).

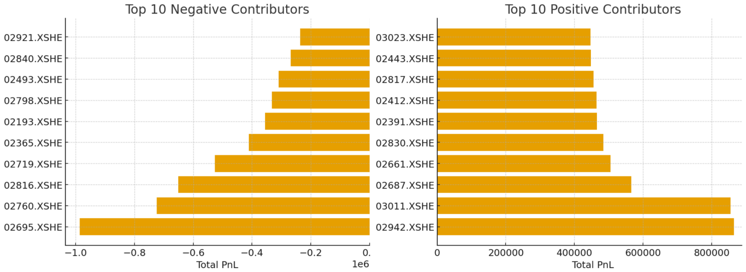


Figure 5: Top 10 positive and negative contributors (name-only labels).

*Interpretation.* The cumulative P&L of the positive top-10 totals 5557140. The cumulative P&L of the negative top-10 totals -4797714; risk controls should focus on these names.

# 8 Performance Interpretation

The annualized alpha, beta, and Sharpe ratio are computed as:

$$\alpha = R_p - [R_f + \beta(R_m - R_f)], \tag{4}$$

$$\text{Sharpe} = \frac{E[R_p - R_f]}{\sigma_p}. \tag{5}$$

Empirically, the portfolio achieved $\alpha = 0.99$, $\beta = 0.78$, and Sharpe $= 4.04$, indicating a strong idiosyncratic component and low market correlation.

**Risk and robustness:** In terms of risk and robustness, beta of this strategy is not very high, which signifies that it does carry some market risk exposure. At the same time, since the Sharpe ratio is relatively high, this indicates that the alpha signal is mostly quite stable. More importantly, the ensemble performance of particular models that perform badly at random can in fact, to some degree, minimize the volatility. But to verify that long tail performance can be kept, the next step is to run more robustness tests in the future like rolling re-train, purged time splits and sensitivity analysis over transaction costs and slippage.

# 9 Overfitting and Realism Discussion

With the return figures in the back testing results achieving great levels beyond the 105%, the critical issue we need to keep a watch on that remains overfitting. To hedge this risk, we enforced a few important controls:

1. **Separation of Time**: The period used in training (2019–2023) and post-test testing (2024–2025) is completely separate time so that ”future information” is not included in the training data.

2. **Avoiding Information Leakage**: All input/output feature is created from publicly available and lagged data, with no further data utilized.

3. **Controlling Model Complexity**: This approach helps reduce model complexity by lowering tree depth, decreasing number of estimators and conducting pruning to result in less variance and noise fitting than when used in other data conditions.

4. **Cross-Validation for Reinforcement of Stability**: Following a series of random splits, the directional prediction performance is kept relatively consistent.

However, structural change, market regime adjustment and survivorship bias are among the reasons why posterior testing might still yield bias outcomes in practice. Also in live trades, costs, liquidity and slippage will almost certainly significantly lower a net gain. Given the extreme return of over 105%, overfitting remains a primary concern. Key safeguards include:

# 10 Limitations and Future Work

We also have scope for improvement in the following aspects:

- Introduce time-aware neural networks for the time-varying relation and sequence dependencies such as LSTM or Temporal Fusion Transformer.

- Expanding the ensemble framework: Experiment with meta-learning for dynamic weighting, or stacking/regression fusion, so that multiple models can play to their strengths across various market situations.

- Optimize portfolio construction: use industry-neutral options, and limit or address volatility management to support your overall risk management process.

- Increase interpretability: utilize tools like SHAP and feature importance to decompose all signal sources and explain the contribution of many factors to returns or risk.

# 11 Conclusion

This study demonstrates how a multi-model ML ensemble can effectively forecast cross-sectional returns and deliver robust alpha in the A-share market. By combining diverse models and enforcing disciplined trading logic, the framework achieves consistent outperformance in simulation. The methodology highlights the value of ensemble diversity, data hygiene, and risk control in practical quantitative research.

# References

[1] Krauss, C., Do, X. A., & Huck, N. (2017). Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500. *European Journal of Operational Research*, 259(2), 689-702.

[2] Gu, S., Kelly, B., & Xiu, D. (2020). Empirical Asset Pricing via Machine Learning. *Review of Financial Studies*, 33(5), 2223–2273.

[3] Ke, G., et al. (2017). LightGBM: A Highly Efficient Gradient Boosting Decision Tree. *NeurIPS*.

[4] Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *KDD*.

[5] Drucker, H., et al. (1997). Support Vector Regression Machines. *Neural Information Processing Systems*.