RL-Based Explainable Factor Generation for Market Risk Analysis

Shuxiao Chen(✉), Ruyao Xu, Chenyu Wang, Mingjun Ma, and Zhongchen Miao

Innovation Lab, Shanghai Financial Futures Information Technology Co., Ltd, Shanghai, China
chensx@cffex.com.cn

Abstract. Financial markets are characterized by complex interdependencies and regime shifts. A critical yet underexplored aspect lies in the generation of interpretable factors for various financial tasks, such as risk prediction. These factors should adapt to evolving market conditions while maintaining transparency.

We propose a general two-stage framework, which separates factor generation from prediction through a modular pipeline consisting of feature-to-factor transformation and factor-to-forecast integration. The RL-based factor generator combines symbolic optimization with Proximal Policy Optimization (PPO) to produce factors grounded in market microstructure indicators. Our framework maintains explicit mathematical formulations, enabling a deeper understanding of economic causality.

Experimental results demonstrate two key advantages: 1) RL-generated factors exhibit superior temporal stability and predictive performance, with a stronger correlation to target variables compared to raw features, and 2) downstream prediction models universally benefit from these RL-enhanced factors, improving overall performance without compromising interpretability.

Keywords: Reinforcement Learning · Explainable AI · Financial Risk Prediction · Factor Generation

1 Introduction

Financial markets exhibit complex interdependencies and regime shifts, posing challenges for risk predcition and prevention [8,18,25]. For regulatory authorities, risk can be defined as extreme price fluctuations. Accurately identifying precursors to price volatility, along with understanding the causes of cross-market spillovers and risk transmission at the microstructural level, are two critical tasks. This problem can be described as a time series forecasting, and it must reconcile key requirements: achieving high prediction accuracy, and maintaining interpretable relationships.

In addition to classical models such as logistic regression and decision trees, recent advancements in time series forecasting have achieved state-of-the-art

T. Taniguchi et al. (Eds.): ICONIP 2025, LNCS 16311, pp. 396–409, 2026.
https://doi.org/10.1007/978-981-95-4381-6_27

accuracy through neural networks based on Transformer [14,16,29] architectures and temporal foundation models inspired by LLMs [3,26]. However, their decision logic remains opaque. Hence in various financial scenarios requiring high interpretability, symbolic regression(SR) [11,12] techniques are widely discussed. Symbolic approaches like genetic programming(GP) [11,12] generate formulas but suffer from inefficient search strategies and single-factor myopia. Recent research introduces reinforcement learning(RL) [33] and diffusion models [22], which show promise in generating interpretable factors for cross-sectional prediction tasks in quantitative finance.

The limitations crystallize in three dimensions. 1) **Rigid Task Coupling**: The application of factor generation to financial time series forecasting is still underexplored. Most systems entangle factor generation with prediction models, limiting extensibility to new data modalities or model selection. 2) **Temporal Adaptability**: Existing symbolic methods lack mechanisms to handle concept drift in temporal dimension. 3) **Interpretability-Accuracy Tradeoff**: Deep neural networks and temporal foundation models suffer from black-box opacity.

Our framework addresses these limitations through three key innovations:

(1) **Decoupled Modular Framework:** A modular pipeline separating factor generation from prediction, enabling plug-and-play integration of data modalities and diverse models; Additionally, we have designed an extensible evaluation scheme that includes distinct metrics across three key dimensions: factor quality, factor generation capability and prediction performance.
(2) **RL-Based Factor Generation:** We redesign RL-based factor generator to jointly optimize factor relevance and predictive ability, improving the temporal generalization ability.
(3) **Causal Economic Mapping:** The generated factors retain explicit mathematical formulations, providing insights into economic causality and enabling precise risk attribution.

Experimental results demonstrate the framework's effectiveness: 1) The analysis shows that the RL-generated factors exhibit superior temporal stability compared to the raw features. 2) Downstream prediction tasks benefit substantially. The incorporation of RL-generated factors significantly enhances the predictive performance. In 70.83% comparative experiments, the RL-generated factors outperform the baseline factors in terms of predictive accuracy.

Also, RL-generated factors not only reveal meaningful economic relationships but also offer valuable insights for downstream financial tasks.

2 Related Work

2.1 Interpretable Time Series Forecasting

The development of interpretable models in financial time series forecasting has evolved from traditional statistical approaches to sophisticated hybrid architectures that balance predictive power with transparency. Early interpretable models, such as linear regression and decision trees, established foundational principles by offering direct insights through feature coefficients [15,32,36] and hierarchical decision rules [19]. These models remain pivotal due to their inherent simplicity and decomposability [10], enabling users to trace predictions to specific input features or logical conditions. However, the nonlinearity and complexity of financial data necessitated advancements integrating interpretability into more powerful frameworks. Attention mechanisms [26] emerged as a transformative solution, embedding feature and temporal importance directly within neural architectures. Models like AT-LSTM [35] and MAGNN [5] leverage attention weights to highlight influential features and critical time windows, while graph-based approaches [6,7] map relational dependencies between market entities, enhancing contextual interpretability. Hybrid techniques further bridge accuracy and transparency, such as adaptive linear regressions guided by neural networks [15] and fuzzy logic systems quantifying prediction uncertainty through membership functions [28]. Current research emphasizes evaluation frameworks assessing simulability, decomposability, and transparency [10]. It also explores domain-adaptive architectures that align model outputs with financial theories [37].

2.2 Financial Factor Generation

The automated generation of formulaic factors is grounded in SR, a methodology for uncovering interpretable mathematical relationships. Traditionally, SR assumes the existence of a ground-truth formula, but its adaptation to financial forecasting addresses market complexity through hybrid approaches, including neural-symbolic networks [21] autoregressive generators [17], and pretrained expression language models [27]. Landajuela et al. [13] further integrated neural and symbolic paradigms, facilitating structured formula discovery. These SR advancements complement genetic programming (GP)-based factor mining, where early frameworks [11,12] introduced time-series operators and mutual information to model nonlinear returns. AlphaEvolve [4] extended this approach by representing factors as computation graphs, enabling matrix-wise operations but sacrificing interpretability. Meanwhile, machine learning methods, such as SFM [34], and heterogeneous data fusion techniques [30,31] have improved trend modeling. The latest advancements in the field include the application of reinforcement learning [33], which goes beyond conventional GP by dynamically exploring collaborative factor sets. Another cutting-edge development is AlphaForge [22], a framework that leverages diffusion techniques to dynamically mine and combine formulaic alpha factors, enhancing performance in quantitative investment.

3 Methodology

3.1 Problem Formulation

Given a target stock index (e.g., CSI 300), we formulate providing early warning of market turbulence risk as a structured forecasting task. Let $\{(X_t, y_t)\}_{t=1}^{T}$ represent the observational dataset, where each trading day t contains:

Input: $X_t \in \mathbb{R}^{m\times\tau}$, a normalized feature matrix comprising m raw features (e.g., opening prices, futures basis spreads, trading volumes) across a τ-day window.

Target: $y_t \in \mathbb{R}$, normalized price change for day t.

The objective is twofold: **1.** Discover n interpretable risk factors $\{f_i\}_{i=1}^n$ as symbolic functions of raw features. **2.** Construct a linear combination to predict y_{t+1}:$\hat{y}_{t+1} = \sum_{i=1}^n \omega_i f_i(X_t)$. where each $f_i : \mathbb{R}^{m\times\tau} \rightarrow \mathbb{R}$ maps input features to scalar risk indicators. In practice, the selection of forecasting models is flexible, and any model can be applied.

This formulation decouples the problem into two interdependent stages (Fig. 1):

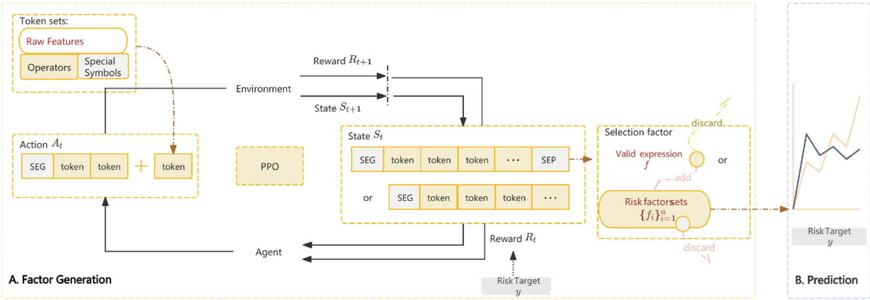

Fig. 1. Methodological Framework: feature-to-factor transformation and factor-to-forecast integration, enabling modular optimization of interpretability and predictive power.

3.2 RL-Driven Factor Mining

Risk Factors. Our approach to constructing risk factors $\{f_i\}_{i=1}^n$ follows the methodology of previous works [22,33]. Each factor mathematical formula is inherently equivalent to an expression tree, which in turn is mapped to a Reverse Polish Notation (RPN) expression. The process begins by constructing an expression tree, where the nodes represent either operands (such as features, constants, or variables) or operators (such as mathematical or statistical functions). A post-order traversal of this tree results in the equivalent RPN expression, where the operators follow their operands. In RPN, there is no need for parentheses or operator precedence rules, as the sequence of operations is unambiguously determined by the order of appearance of the operators and operands. In this approach, tokens are sequentially processed, with operands being pushed onto the stack and operators applied as soon as they are encountered (Fig. 2).

Factor Generation. We model factor generation as a non-stationary Markov Decision Process (MDP) to control the creation of valid mathematical expressions in RPN. This approach allows for sequential autoregressive generation while

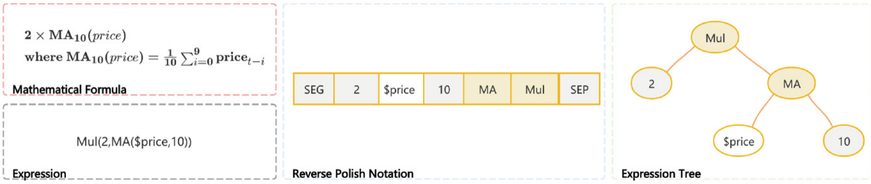

Fig. 2. Risk Factors: For example, consider the RPN expression: "BEG 2 $price 10 MA Mul SEP". This expression represents the calculation of the 10-day moving average of price, multiplied by 2.

ensuring algebraic interpretability. The MDP is defined as follows: *Token Representation.* The factor is expressed as a mathematical formula composed of tokens, including operators, features, and special symbols.

(1) **Operators**: Operators are classified into arithmetic, correlation and covariance analysis, and time-series operators.
(2) **Features**: Features represent selected raw variables, such as spot and futures prices, along with derived indicators like basis. In this study, we focus on market dynamics from the perspective of spot-futures interaction.
(3) **Special Symbols**: Special symbols, such as BEG and SEP, denote the beginning and end of a sequence, while constants also appear.

These tokens together form the structure of the factor expression.

State Space. The state s_t represents the current sequence of RPN tokens, initialized with a BEG token. Valid states follow the structure $[BEG, \tau_1, \tau_2, \ldots, \tau_n]$, where each τ_i is a token. The maximum sequence length is constrained to 20 tokens to maintain interpretability, and episodes are terminated if the sequence exceeds this threshold.

Action Space. The action space consists of valid RPN tokens, including operators, raw features, and constants. To ensure syntactically valid sequences, we enforce grammatical constraints via action masking, which eliminates invalid actions from the action space. At each state, the model selects an action a_t from the valid subset of tokens.

State Transition Dynamics. State transitions are deterministic: appending the action a_t to the current token sequence yields the next state s_{t+1}. Formally, $s_{t+1} = \text{concat}(s_t, a_t)$. When the terminal state is reached and it is a valid expression f, it will enter the selection phase to assess whether it will be retained as a risk factor.

Reward Structure. Rewards are provided only at terminal states. The final reward is based on the Mean Squared Error (MSE) between the input of the combination model, after incorporating the generated factor, and the prediction target y. Additionally, a correlation coefficient between the current expression and y is also considered.

$$Reward = -3 \cdot \frac{1}{T}\sum_{t=1}^{T}(y_t - \hat{y}_t)^2 + \frac{\mathrm{Cov}(f, y)}{\sigma_f \sigma_y} \tag{1}$$

where $f \in \mathbb{R}^T$ is expression at terminal states, and $\hat{y} \in \mathbb{R}^T$ is regression value in factor selection phase. We apply the following constraints on the reward structure:

(1) Sequences longer than 20 tokens receive a penalty $Reward = -1$.
(2) Redundant unary operations (e.g., rank(rank(x))) result in $Reward = 0$.
(3) Expressions with a correlation greater than 0.75 to the selected risk factors will receive a reward of $Reward = 0$.

Optimization Framework. We utilize Proximal Policy Optimization (PPO) [23] with a discount factor $\gamma = 1$ to avoid penalizing longer valid expressions. The objective function is defined as:

$$L_{\mathrm{CLIP}}(\theta) = \mathbb{E}_t\left[\min\left(r_t(\theta)\hat{A}_t, \mathrm{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)\right] \tag{2}$$

where $r_t(\theta) = \frac{\pi_{\mathrm{old}}(a_t|s_t)}{\pi_\theta(a_t|s_t)}$ and $\hat{A}_t$ is the generalized advantage estimate. To address action space constraints, we apply action masking [20] during policy sampling, ensuring that only valid tokens are selected at each step.

The network architecture utilized in our approach draws on that of previous works [33]. In line with PPO algorithm requirements, our model incorporates a value network and a policy network. Both networks share a base LSTM feature extractor that translates token sequences into dense vectors. Subsequently, distinct value and policy heads are appended post-LSTM. The value network assesses action values under given states, while the policy network selects actions based on current strategy and states. Both networks utilize a multi-layer perceptron (MLP) for further processing. The policy is trained to continuously generate novel factors, optimizing prediction performance without resetting between episodes.

Selection Factor. The valid expression f and the set of selected risk factors are integrated together as the independent variables $\widetilde{X} \in \mathbb{R}^{(n+1)\times T}$. A linear regression least squares problem is then established to predict $y \in \mathbb{R}^T$:

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^{n+1}}{\arg\min} \|\widetilde{X}^\top \beta - y\|_2^2 = (\widetilde{X}\widetilde{X}^\top)^{-1}\widetilde{X}y \in \mathbb{R}^{n+1} \tag{3}$$

The factor(expression) corresponding to the smallest absolute weight in $\hat{\beta}$ (i.e., $\arg\min_i |\hat{\beta}_i|$) will be discarded, as it represents the least significant factor in the model.

3.3 Interpretable Predictions

In data-driven decision-making, model interpretability is essential for ensuring that decisions are understandable and trustworthy. When decision-makers can comprehend how a model's predictions are generated, they are more likely to trust and adopt these predictions. Furthermore, interpretability helps to uncover biases within the data, enhancing the fairness and transparency of the model.

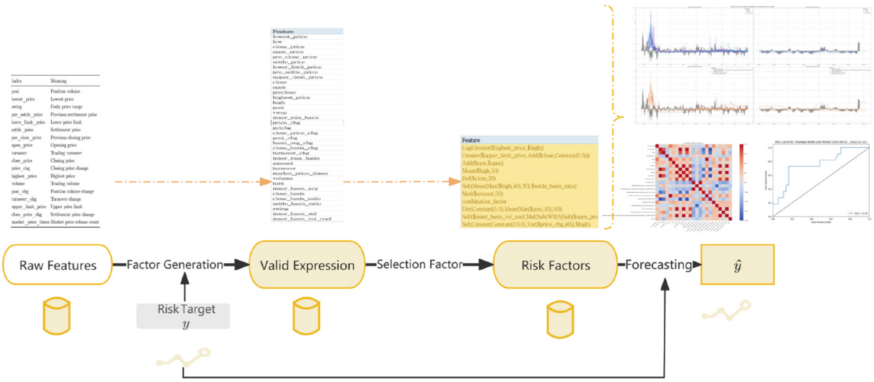

Fig. 3. Data Evolution Flow: the stagewise transformation of raw heterogeneous features into risk factor constructs

In this study, we use linear regression for prediction, with input features derived from the factors generated in the previous section. The prediction target is $y_t = \frac{\$close_{t+1} - \$close_t}{\$close_t} \in \mathbb{R}$. where $\$close_t$ is closing price of day t.

Figure 3 illustrates the data evolution flow of this work. Since the factor generation and prediction tasks are decoupled in the framework , various prediction models can be flexibly chosen during the prediction phase based on specific needs.

In the numerical experiments, we also used LightGBM(LGB) [9], Autoformer [29], Transformer [26], PatchTST [16] and Nonstationary Transformer [14] for comparison.

4 Experiments

4.1 Experiment Settings

The objective of this experiment is to evaluate the generated factors based on three key aspects: factor quality, model generation capability, and the ability to predict price fluctuations and provide market alerts.

Data. Our experiment focuses on the CSI 300 Index of the Chinese A-share market (ticker: SH000300), with the corresponding stock index futures code represented by IF. For the futures data, we selected the continuous main contract. The dataset is split into training, validation, and test sets, covering the following time periods (Table 1):

Table 1. Dataset Information

Dataset	#Training set	#Validation set	#Test set
SH000300-IF	1890 (2015-04-01,2022-12-31)	139 (2023-01-01,2023-07-31)	244 (2023-08-01,2024-07-31)

We set $m = 38$ and $n = 10$. The 38 selected indicators primarily encompass the underlying market conditions, futures market data, and basis. The data granularity is daily.

All data used in the experiment is publicly available and can be accessed through platforms such as Wind and Baostock.

Model. Since our work focuses on framework design rather than modifications to model architecture, comparing with traditional baselines is not appropriate. Instead, we aim to optimize the prediction results of various models by integrating pre-processing RL. Baseline features is raw features.

We consider several models: machine learning approach like LightGBM (LGB) [9], Transformer [26], and several state-of-the-art models based on the Transformer architecture, such as Nonstationary Transformer [14], PatchTST [16]and Autoformer [29].

The experiments were conducted on an NVIDIA Tesla P100 GPU for computational support.

4.2 Evaluation Metrics

We have proposed a framework for evaluating the effectiveness of prediction tasks, incorporating four evaluation metrics across three dimensions.

Risk Factor Quality. First, we assess the ability of the generated factors to represent market risk. Referred to the design of the metric IC in factor investing, this is done by calculating the Spearman's correlation in time dimension between generated risk factors $\{f_i\}_{i=1}^{n}$ and the target indicator y over a test period.

The correlation is given by:

$$\mathrm{Corr}(f_i, y) = \frac{\frac{1}{T}\sum_{t=1}^{T}\left(f_{it} - \bar{f}_i\right)\left(y_t - \bar{y}\right)}{\sigma_{f_i}\sigma_y}, i = 1, 2, \cdots, n. \tag{4}$$

where $\bar{f}_i$ and σ_{f_i} are the means and standard deviations of f_i, respectively, and similarly for y. T is the sample size.

Additionally, we compute the weight of each risk factor in the predictive model. This helps us selectively focus on key factors for further financial analysis.

Factor Generation Ability. The second dimension evaluates the factor generation capability of the RL model, aiming to ensure that the generated factors outperform the raw ones. The generated risk factors $\{f_i\}_{i=1}^n$ are ranked alongside the raw features using the aforementioned correlation(Eq. 4). The generated risk factors and raw features are treated as positive and negative examples, respectively, for the calculation of the AUC value:

$$AUC = \frac{1}{m \times n}[\sum_{i=1}^{n} \text{rank}(f_i) - \frac{n \times (n+1)}{2}] \tag{5}$$

where rank(f_i) is the rank of generated risk factor f_i by correlation metrics, m is the number of raw features, and n is the number of generated risk factors.

The AUC is calculated to evaluate the likelihood that the factor generation model produces factors with stronger correlation to y compared to the raw features.

Prediction Performance. Finally, from the perspective of predicting price fluctuations in the regression task, we measure prediction accuracy using mean squared error(MSE), mean absolute error(MAE), median absolute error(MedianAE), and root mean squared error(RMSE).

For different prediction models, the RL factor will be calculated to assess the proportion of indicators on which it can improve the prediction results.

4.3 Main Results

Factor Evaluation Results. Table 2 presents the evaluation of the generated factors, including their weights, expressions, and correlations with the target indicator y.

To assess the temporal validity of factors, we performed cross-period validation through correlation analysis between RL-based factors and y, as well as between the raw features and y. The results can be observed in the Fig. 4.

The empirical results indicate that when examining the intersection of top decile factors ranked by absolute correlation coefficients across training (2015–2022) and testing (2023–2024) periods, only two RL-based factors demonstrated persistent significance. And none of raw features.

This persistence suggests that our RL-based factor generation method exhibits cross-period robustness, effectively capturing time-invariant market risk patterns.

Table 2. Factor Evaluation: Weights, Expressions, and Correlations with the Target

Weight	Factor Expression	Correlation
0.08	`Sum(Div($inner_basis_vol_coef$,Constant(−10.0)),20)`	0.03
−0.11	`Mul(Add(Constant(−1.0),$posi_chg),$settle_basis_ratio)`	−0.14
−0.14	`Div(Mul($close_basis_chg,$close_basis_chg), Constant(0.5))`	−0.06
0.16	`Max($settle_basis_ratio,20)`	0.01
−0.10	`Mul($inner_min_basis,$turnover_chg)`	−0.10
0.12	`Sub($turn,$close_basis_ratio)`	0.04
−0.25	`Add($close_basis_chg, Add(Less($close_basis_ratio, $inner_basis_avg), Sub($lowest_price, Mul($close_basis_chg,$pre_settle_price))))`	−0.15
−0.13	`Mad(Mul(Constant(−30.0), Med($close_basis_ratio,30)), 50)`	−0.06
−0.14	`Sub($close_basis,$inner_basis_vol_coef)`	−0.08
0.22	`Mul(Abs(Sub(Add($inner_basis_vol_coef,$turn) ,$close_basis_ratio)),$close_basis)`	0.03

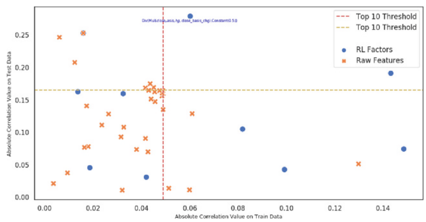

Fig. 4. Factor Persistence: absolute correlation comparison between two periods

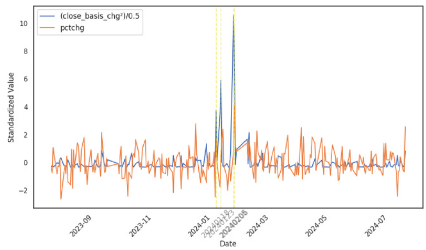

Fig. 5. Comparative Trends of Squared Closing Basis Changes and Index Price Changes: At specific junctures, the futures-spot linkage is evident

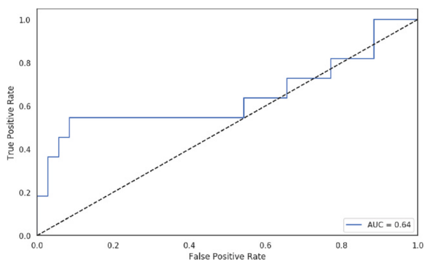

Fig. 6. ROC-AUC Curve for Train Data

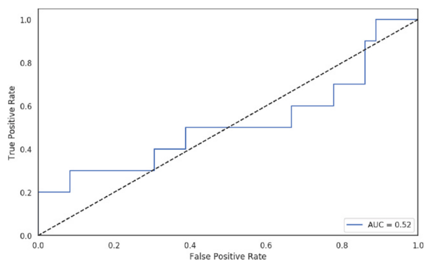

Fig. 7. ROC-AUC Curve for Test Data

ROC-AUC Evaluation. The ROC-AUC curves for the train period and test period are shown in Figs. 6 and 7, illustrating the performance of the factor generation model.

On the training data, our factor generation model has a 64% probability of producing factors with stronger correlations to y. Meanwhile, on the test set, this probability is 52% , indicating that the model demonstrates good generation quality while also exhibiting some degree of generalization.

Prediction Performance. The prediction performance of the generated factors was evaluated on six different models: Logistic Regression, LightGBM, Nonstationary Transformer, PatchTST, Transformer, and Autoformer. The results for both models are presented in Table 3, where the comparison is made between the raw features and the generated factors.

Table 3. Prediction Performance

Model	Factor Type	Metric			
		MSE	MAE	MedianAE	RMSE
Linear Regression	Raw Features	**0.9440**	0.7801	0.6839	**0.9716**
	Generated Factors	1.0605	**0.7641**	<u>0.6157</u>	1.0298
LGB	Raw Features	1.1068	0.8335	0.6832	1.0521
	Generated Factors	<u>0.9995</u>	<u>0.7667</u>	**0.6096**	<u>0.9998</u>
Autoformer	Raw Features	1.2175	0.8453	0.6402	1.1034
	Generated Factors	1.1213	0.8314	0.6440	1.0589
Transformer	Raw Features	1.0754	0.8055	0.6555	1.0370
	Generated Factors	1.0912	0.8149	0.6629	1.0446
PatchTST	Raw Features	1.1576	0.8521	0.6986	1.0759
	Generated Factors	1.1419	0.8416	0.6909	1.0686
Nonstationary Transformer	Raw Features	1.1744	0.8604	0.7612	1.0837
	Generated Factors	1.1231	0.8198	0.6728	1.0598

Easily obtainable, in 70.83%(17/24), comparative experiments, the RL-generated factors outperform the baseline factors in terms of predictive accuracy.

4.4 Case Study: Factor Interpretability and Economic Causality

It can be observed that factors(Table 2), such as the basis and other spot market metrics with linked contract periods, exhibit a high level of participation. This also reflects the contagion and spillover relationships between different markets to some extent.

`Div(Mul($close_basis_chg,$close_basis_chg),Constant(0.5))` suggests that the squared change in the closing basis and the index price change are closely related(Fig. 4).

As shown in Fig. 5, their trends converge, further indicating a tight connection between futures and spot markets. The volatility spillover between these markets represents the process of information transmission, reflecting the price discovery mechanism [1,2,24].

Symbolic factors like this can help to better understand economic causal relationships, thereby facilitating other financial analysis tasks.

5 Conclusion

This study presents a two-stage reinforcement learning framework designed to generate interpretable risk factors for forecasting financial market dynamics, with particular emphasis on the interactions between futures and spot markets. By integrating symbolic optimization and adaptive reward design (weighting prediction accuracy and factor relevance), our method simultaneously improves model interpretability and prediction accuracy. The main conclusions are as follows:

1) The analysis shows that the RL-generated factors exhibit superior temporal stability compared to the raw features. Through rigorous cross-period validation (across both the training and testing phases), two of the generated factors maintained statistically significant correlations with price changes. In contrast, no such correlations were observed with the raw features. 2) Downstream prediction tasks benefit substantially. The incorporation of RL-generated factors significantly enhances the predictive performance of nearly all models, particularly in terms of MAE and MedianAE. In 70.83% comparative experiments, the RL-generated factors outperform the baseline factors in terms of predictive accuracy.

Also, RL-generated factors not only reveal meaningful economic relationships but also offer valuable insights for downstream financial tasks.

This work establishes a paradigm for explainable AI in quantitative finance. Future research could refine reward functions to prioritize tail-risk correlations, improving early warning capabilities for extreme market events. Additionally, experiments on cross-domain and international financial market datasets will be essential to validate the broader applicability of this approach.

References

1. Cai, X.: Study on the price discovery function of stock index futures. Price: Theory Pract. **2**, 56–57 (2010)
2. Cai, X.: Research on the price discovery function of CSI 300 index futures. J. Financ. Dev. Res. **3**, 29–34 (2011)
3. Cao, D., et al.: Tempo: prompt-based generative pre-trained transformer for time series forecasting (2023)
4. Cui, C., Wang, W., Zhang, M. et al.: AlphaEvolve: a learning framework to discover novel alphas in quantitative investment. In: Li, G. et al. (eds.) SIGMOD 2021, pp. 2208–2216. ACM (2021)
5. Cheng, D., Yang, F., Xiang, S., Liu, J.: Financial time series forecasting with multi-modality graph neural network. Pattern Recognit. **121**, 108218 (2022)
6. Deng, S., Chen, J., Zhang, N. et al.: Knowledge-driven stock trend prediction and explanation via temporal convolutional network. In: WWW 2019 Companion, pp. 678–685. ACM, San Francisco (2019)
7. Huang, K., Li, X., Liu, F., et al.: ML-GAT: a multilevel graph attention model for stock prediction. IEEE Access **10**, 86408–86422 (2022)
8. Kuester, K., Mittnik, S., Paolella, M.S.: Value-at-risk prediction: a comparison of alternative strategies. J. Financ. Econom. **4**(1), 53–89 (2006). https://doi.org/10.1093/jjfinec/nbj002
9. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., et al.: LightGBM: a highly efficient gradient boosting decision tree. In: Proceedings 31st International Conference on Neural Information Processing Systems, pp. 3149–3157. Curran Associates (2017)
10. Lipton, Z.C.: The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. Queue **16**(3), 31–57 (2018). https://doi.org/10.1145/3236386.3241340
11. Lin, X., Chen, Y., Li, Z., He, K.: Revisiting stock alpha mining based on genetic algorithm. Technical Report HTSC-2019-10750101, Huatai Securities (2019)
12. Lin, X., Chen, Y., Li, Z., He, K.: Stock alpha mining based on genetic algorithm. Technical Report HTSC-2019-10750101, Huatai Securities (2019)
13. Landajuela, M., Lee, C.S., Yang, J. et al.: A unified framework for deep symbolic regression. In: Advances in Neural Information Processing Systems, vol. 35, pp. 33985–33998 (2022). https://doi.org/10.48550/arXiv.2205.14832
14. Liu, Y., Wu, H., Wang, J., Long, M.: Non-stationary transformers: exploring the stationarity in time series forecasting. In: Advances in Neural Information Processing Systems, vol. 35, pp. 9881–9893 (2022)
15. Munkhdalai, L., Munkhdalai, T., Pham, V.H., et al.: Recurrent neural network-augmented locally adaptive interpretable regression for multivariate time-series forecasting. IEEE Access **10**, 11871–11885 (2022)
16. Nie, Y., Nguyen, N. H., Sinthong, P., Kalagnanam, J.: A time series is worth 64 words: long-term forecasting with transformers (2022)
17. Petersen, B.K., Landajuela, M., Mundhenk, T.N. et al.: Deep symbolic regression: recovering mathematical expressions from data via risk-seeking policy gradients. In: ICLR 2021 (2021)
18. Peng, Y., Wang, G.X., Kou, G., Shi, Y.: An empirical study of classification algorithm evaluation for financial risk prediction. Appl. Soft Comput. **11**(2), 2906–2915 (2011). https://doi.org/10.1016/j.asoc.2010.11.028

19. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo (1993). https://books.google.ca/books?id=HExncpjbYroC
20. Huang, S., Ontañón, S.: A closer look at invalid action masking in policy gradient algorithms. In: Proceedings of the Thirty-Fifth Inter-national Florida Artificial Intelligence Research Society Conference, FLAIRS 2022,Hutchinson Island, Jensen Beach, Florida, USA (2022)
21. Sahoo, S.S., Lampert, C.H., Martius, G.: Learning equations for extrapolation and control. In: Dy, J.G., Krause, A. (eds.) ICML 2018. PMLR, vol. 80, pp. 4439–4447 (2018). https://proceedings.mlr.press/v80/sahoo18a.html
22. Shi, H., Song, W., Zhang, X. et al.: AlphaForge: a framework to mine and dynamically combine formulaic alpha factors. In: AAAI 2025, vol. 39(12), pp. 12524–12532. AAAI Press (2025)
23. Schulman, J., Wolski, F., Dhariwal, P. et al.: Proximal policy optimization algorithms. arXiv:1707.06347 (2017)
24. Sun, J., Zheng, L.: Research on the linkage relationship between stock index futures and spot prices based on VAR model. Shanghai Financ. **11**, 63–69 (2017)
25. Valaskova, K., Kliestik, T., Svabova, L., Adamko, P.: Financial risk measurement and prediction modelling for sustainable development of business entities using regression analysis. Sustainability **10**(7), 2144 (2018). https://doi.org/10.3390/su10072144
26. Vaswani, A., Shazeer, N., Parmar, N. et al.: Attention is all you need. In: NeurIPS 2017, vol. 30, pp. 5998–6008. Curran Associates (2017)
27. Valipour, M., You, B., Panju, M., Ghodsi, A.: SymbolicGPT: a generative transformer model for symbolic regression. arXiv:2106.14131 [cs.LG] (2021)
28. Wang, W., Lin, W., Wen, Y., et al.: An interpretable intuitionistic fuzzy inference model for stock prediction. Expert Syst. Appl. **213**, 118908 (2023)
29. Wu, H., Xu, J., Wang, J., Long, M.: Autoformer: decomposition transformers with autocorrelation for long-term series forecasting. In: Advances in Neural Information Processing Systems, vol. 34, 22419–22430 (2021)
30. Xu, W., Liu, W., Wang, L. et al.: HIST: A graph-based framework for stock trend forecasting via mining concept-oriented shared information. arXiv:2110.13716 [q-fin.ST] (2021)
31. Xu, W., Liu, W., Xu, C. et al.: REST: Relational event-driven stock trend forecasting. In: Leskovec, J. et al. (eds.) WWW 2021, pp. 1–10. ACM/IW3C2 (2021)
32. Xu, J., Zhou, J., Jia, Y. et al.: An adaptive master-slave regularized model for unexpected revenue prediction enhanced with alternative data. In: ICDE 2020, pp. 601–612. IEEE (2020)
33. Yu, S., Xue, H., Ao, X. et al.: Generating synergistic formulaic alpha collections via reinforcement learning. In: KDD'23, pp. 5476–5486. ACM, New York (2023). ISBN 979-8-4007-0103-0
34. Zhang, L., Aggarwal, C.C., Qi, G.J.: Stock price prediction via discovering multi-frequency trading patterns. In: Proceedings of the 23rd ACM SIGKDD, pp. 2141–2149. ACM, Halifax (2017)
35. Zhang, X., Liang, X., Zhiyuli, A. et al.: AT-LSTM: an attention-based LSTM model for financial time series prediction. In: IOP Conference Series: Materials Science and Engineering, vol. 569, no. 5, pp. 052037 (2019)
36. Zhang, X., Qu, S., Huang, J., et al.: Stock market prediction via multi-source multiple instance learning. IEEE Access **6**, 50720–50728 (2018)
37. Zheng, Y., Yang, Y.Y., Chen, B.: Incorporating prior financial domain knowledge into neural networks for implied volatility surface prediction. In: KDD 2021, pp. 3968–3975. ACM (2021)