Application of Alpha Testing on Deep Symbolic Regression-based Factor Zoo

Jongkook Choi
Department of Industrial & Management Engineering, Pohang University of Science and Technology

Woo-Sung Jung
Department of Industrial & Management Engineering, Pohang University of Science and Technology
Department of Physics, Pohang University of Science and Technology

In empirical asset pricing using a machine-generated factor zoo, signals are often contaminated with heavy noise and risk-premium exposures relative to genuine market anomalies. We verify that False Discovery Rate (FDR) controlled alpha selection of Giglio, Liao, and Xiu (2021) can be applied to a factor pool generated by Reinforcement Learning-based Deep Symbolic Regression. We demonstrate that FDR-controlled alpha selection improves out-of-sample shapre consistency of factor portforlio. (*JEL* G12, G31)

Received XXXX XX, XXXX; editorial decision XXXX XX, XXXX by Editor XXXXXXXXXXXX.

The construction of a "Factor Zoo" is an example of data-driven methods in empirical asset pricing (Goldstein, Spatt, and Ye (2021)). A factor zoo is a set of estimators for market anomaly α or risk exposure β for asset pricing. Quantitative alpha researchers evaluate candidate expressions by observing the backtested return performance of test portfolios that utilize these estimators as portfolio weights. Subsequently, portfolio researchers select and combine expressions from the factor zoo to build portfolios, for statistical arbitrage or controlled factor exposure.

I appreciate helpful comments from the referees. I also thank Damin Lee, Seonbin Jo, Sanggoo Kang, Hyunah Park, Hyunjin Cho and other members of Complexity in Social System Laboratory for helpful discussions. Send correspondence to Woo-Sung Jung, Department of Industrial & Management Engineering and Department of Physics, Pohang University of Science and Technology, Pohang 37673, Republic of Korea; telephone: (+82) 54-279-8255; email: wsjung@postech.ac.kr.

Published by Oxford University Press on behalf of The Society for Financial Studies 2026.
doi:10.1093/rfs/abcd123 Advance Access publication August 14, 2026

Symbolic regression is a relatively well-known method for automated factor zoo discovery. Originally introduced by Schmidt and Lipson (2009), symbolic regression searches for mathematical expressions that explain relationships among datasets by trying various symbolic expressions. The advantages of symbolic regression include the absence of an explicit prior model, relatively simple generalization for operators and data, and symbolic interpretability of the result.

Kakushadze (2016); Tulchinsky (2015) popularized symbolic regression based financial research. Conventionally, genetic programming-based heuristics are used for symbolic regression. More recently, Valipour et al. (2021); Yu et al. (2023) demonstrated the viability of applying Reinforcement Learning (RL) and Large Language Models (LLMs) for symbolic regression.

However, It is not trivial to expect for a machine-generated factor pool to consist of genuine market anomalies. Giglio, Liao, and Xiu (2021) demonstrate that funds with statistically significant alpha are sparse, based on cross-sectional factor modeling of return matrix from a fund pool. Therefore, machine-generated factor zoo trained solely on backtest metrics without knowledge of factor model, require statistical validation for alpha significances of each signal in the pool, compared to the pool.

The key contributions of our study are as follows:

First, we formulate the symbolic regression-based factor zoo mining process as a Markov Decision Process (MDP), demonstrating that factor search can be fully automated using RL-based sequence generation.

Second, in addition to standard syntax penalties used in RL-based symbolic regression, we show that symbolic agents can be trained using Sharpe ratio rewards. Transaction costs are omitted because estimation of turnover and trading cost after position netting is difficult during alpha search stage, and we focus on factor search.

Third, we show that our RL based symbolic regression produces spurious factors, and these factors harm out-sample performance of the factor zoo, like Harvey, Liu, and Zhu (2016) pointed out. Returns of each factors in the factor zoo, were combined to maximize in-sample Sharpe, and the weight was used to calculate the out-sample factor zoo performance. Inconsistency between the out-sample Sharpe and the in-sample Sharpe was observed, and rewarding agent with in-sample Sharpe Ratio did not mitigate this problem. We apply the False Discovery Rate (FDR) controlled alpha selection methodology of Giglio, Liao, and Xiu (2021) to our machine-generated pool, empirically confirming that the out-of-sample Sharpe ratios of portfolio returns are improved by introducing factor structure information.

1. Formulation

1.1 Portfolio Formation using Symbolic Regression

For a set of stocks at time t, the return vector $\mathbf{r}_t$ can be decomposed into linear factor components as follows:

$$r_t = \alpha + \sum_i \beta_{i,t}\lambda_i + \beta_{i,t}(f_{i,t} - \lambda_i) + \epsilon$$

where α is statistical arbitrage returns from market anomalies, $f_{i,t}$ is the factor return for the i-th factor at time t, $\beta_{i,t}$ is the factor exposure, λ_i is the risk premium for factor i, and ϵ_t is idiosyncratic specific risk.

Assuming that expected factor excess returns equal zero in the long run ($E[f_{i,t} - \lambda_i] = 0$) and $E[\epsilon_t] = 0$, the expected return of a test portfolio taking positions proportional to a factor estimator $\hat{\beta}_{j,t}$ in a factor zoo is given by:

$$E[R_p] = E[\hat{\beta}_{j,t}^T r_t] = E[\hat{\beta}_{j,t}^T \alpha] + E[\sum_i \hat{\beta}_{j,t}^T \beta_{i,t}\lambda_i] \tag{1}$$

The first term is statistical arbitrage generated from predicting market anomalies α, while the second term is the return from risk premium exposure of the test portfolio.

1.2 Symbolic Regression-based Factor Zoo

As mentioned before, factor zoo is a methodology for constructing a set of estimators for α or $\beta_{i,t}$. In symbolic regression-based factor discovery, each candidate estimator $\hat{\beta}_{j,t}$ is represented as a sequence of mathematical symbols. This formalism allows automated factor zoo construction by sequence generation algorithms.

Similar to Astudillo et al. (2020), our symbol set is typed. In other word, The category of our symbol is composed of parameter introduction (e.g., 'open', 'high'), constant introduction (e.g., 1,5), and computational(reduction) operators (e.g., addition, multiplication). Only some of the sequences are syntactically valid. In our setup, the maximum symbol sequence length is 32.

A stack machine $\mathcal{I}$ maps a valid symbol sequence to a computable function:

$$\mathcal{I} : \Sigma^{32} \to (\mathcal{D}_t \to \hat{\beta}_t)$$

where Σ^{32} denotes the space of symbol sequences, and $\mathcal{D}_t$ represents market information available at time t.

For instance, an token sequence with reverse Polish notation $\sigma =$ ["data('close')","rank","5","delta","[pad]",...] is interpreted by the stack machine $\mathcal{I}$ as $f(\text{close}) := \text{rank}(\text{close}_t) - \text{rank}(\text{close}_{t-5})$, generating a factor estimator $\hat{\beta}_{j,t} = \mathbb{E}[\beta_{j,t} \mid \mathcal{F}_t] \approx f(\mathcal{D}_t)$.

To summarize, the factor zoo consists of an estimator set $\hat{\beta}_{j,t}$ for α and β_t. If symbolic regression is used, then each estimator can be represented as a valid symbolic sequence $\sigma \in \Sigma^{32}$, and the stack machine $\mathcal{I}$ interpret the sequence and compute it using market data $\mathcal{D}_t$ to produce factor output $\hat{\beta}_t$.

1.3 RL-based Symbolic Regression

When incorporating diverse market data and mathematical operators, the search universe of symbolic sequence space Σ^N grows exponentially and, only a fraction of the symbolic sequence space is syntactically valid, and an even smaller subset represents genuine market anomalies. Consequently, efficient search algorithms are necessary to explore the expression state space.

Genetic programming algorithms have conventionally been used to heuristically define boundaries of meaningful search space subsets and construct machine-generated factor pools like Kakushadze (2016).

On the other hand, equation generation can be framed as a Markov Decision Process (MDP). Deep Symbolic Regression (DSR; Petersen et al. (2021)) formulated symbolic regression as a reinforcement learning, which is exploration and exploitation problem and risk preference problem.

In our study, the financial factor writing process was formulated as MDP as follows:

- **Environment**: Stack Machine $\mathcal{I}: \Sigma^{32} \to (\mathcal{D}_t \to \hat{\beta}_t)$
- **Action**: Selection of Token(symbol) Σ to be appended to the state
  - Similar to Astudillo et al. (2020), tokens Σ are categorized into data introduction and reduction operators.
  - Intermediate sequence Σ^{32} is processed by transformer layers to generate policy probability distribution π for the next token action.
- **State**: Token(symbol) sequence Σ^{32}
  - Only syntactically valid state subsequences $\Sigma^{32}_{\text{valid}}$ produce computable factor estimators $\hat{\beta} = \mathcal{I}(\Sigma^{32}_{\text{valid}})$.
- **Reward**:
  (Improvement by $\hat{\beta}$) − (Syntax Penalty) − (Length penalty)
  - Equation-level backtested Sharpe ratio metric is being used for the reward, instead of point-wise estimation error loss.
  - Improvement is measured by the estimator's contribution to the pool Sharpe. Refer to Appendix B.1 for detailed specification of the RL reward.

1.4 Sharpe Reward for Training Symbolic Regression Agent

Because symbolic regression operates without prior knowledge of factor models, rewarding the agent for the Sharpe does not guarantee the discovery of the market anomaly, and there are possibilities of reward hacking for the in-sample risk premium or unwanted bias.

As shown in Equation 1, returns of test portfolios include risk premium exposures ($\sum_i \beta_{i,t}\lambda_i$). Even if we assume that the factor risk premiums converges to the average risk returns in the long run, $\hat{\beta}$ estimated based on in-sample factor return does not guarantee out-of-sample performance. Thus, optimizing raw backtest return metrics does not necessarily finds statistical arbitrage alpha.

In addition, biased signals are not executable. For instance, dividend strategies without proper ex-date adjustments introduces forward-looking bias. Another example is survivorship bias induced by selecting risky assets without proper delisting return adjustments. It can be difficult to catch these biases, based on out-of-sample performance tests , leading to non-executable live trading signals.

Therefore, robust portfolio research requires formal factor risk modeling. For example, instead of raw Sharpe ratios with risk premiums, residual returns from verified factor models Sheikh (1996) can be used as proxies for alpha and specific risk to calculate Information Ratios.

Acknowledging these limitations, we utilize Sharpe ratios as the RL reward for the following reasons:

First, individual symbolic representations are inherently proper for weak estimators rather than bias-free pure anomaly models. As a trade-off for high interpretability, it is difficult to represent pure market anomalies as a single equation. For example, intention of dividing fundamental data using market capitalization is often normalization but it introduces exposure to price reversion factor. Adding complex factor decomposition as operator set for symbolic regression is computationally prohibitive. Instead of that market neutralization was applied during the backtesting, which is computationally cost-effective, was used in our research.

Second, it is not easy to access stable factor models for an individual alpha researcher, while Sharpe ratio of backtest return is a conventional benchmark.

Furthermore, in this study, we empirically evaluate whether FDR-controlled selection can effectively select expressions with significant alpha, rather than beta exposure.

2. Methodology

2.1 FDR-controlled Alpha Screening

Alpha component(α) of the portfolio return can be tested for the significance, after the factor decomposition of the return matrix of the factor zoo. Alpha selection is performed by testing the null hypothesis ($H_0 : \alpha = 0$).

When evaluating large pools of candidate factors simultaneously, multiple hypothesis testing corrections are required. Giglio, Liao, and Xiu (2021) proposed a False Discovery Rate (FDR) controlled alpha selection method for large-scale fund return pools.

Giglio, Liao, and Xiu (2021) reports that bounding the FDR below 5% successfully selects a robust subset of skilled funds that outperform both in-sample and out-of-sample. The number of latent factors estimated in their framework is set to $K = 5$.

Goldstein, Spatt, and Ye (2021) noted that this FDR alpha selection framework can be extended to broader empirical asset pricing research. Based on this idea, we empirically verify the effect of FDR-controlled alpha selection on machine-generated factor zoos for more consistent out-of-sample performance.

Following Giglio, Liao, and Xiu (2021)'s method, t-statistics and p-values for each factor estimators in the factor zoo.

$$t_i = \frac{\hat{\alpha}_i}{\mathrm{SE}(\hat{\alpha}_i)}, \quad p_i = 1 - \Phi(t_i) \tag{2}$$

$\hat{\alpha}$ is the Fama-MacBeth two-pass regression on the matrix of return matrix of factor test portfolio returns Rj,t where j is the index of factor in the zoo, and t is the day index.

For the false discovery rate control, the adjusted p-value threshold p^* was estimated, utilizing the cross-sectional distribution of alpha in the zoo.

$$p^* = \left\{ p \in \{0,1\} \,\middle|\, p \leq \frac{\tau \sum_{i=1}^{N} 1\{i \leq N : p_i < p\}}{M} \right\} \tag{3}$$

Bootstrap algorithm can help when the return matrix from the factor zoo contains large amount of missing value. Contrary, the matrix coverage is high in our case, due to simplicity of operator and the dataset choice. In this reason, bootstrapping was not applied on p-values.

2.2 Expression Search Space for Symbolic Regression

In this study, we construct a factor zoo using symbolic regression with daily price and volume data from the U.S. stock market as input data. Detailed descriptions of the dataset, input features, and mathematical operators are provided in Appendix 4.

2.3 Neural Net based Symbolic Regression Agent

As described in Section 1, stack machine $\mathcal{I}$ maps symbolic token sequence Σ^N to a computable factor estimator function $\hat{\beta}$, reducing the factor zoo construction task to sequence generation.

Standard machine learning method is employed for the sequence generation. Intermediate token sequence (state; $\sigma \in \Sigma^N$) is fed into the policy network to produce next-token probability distribution (policy; π). Next symbol (action) is iteratively sampled and appended until a finished sequence is formed. Reward is calculated using the stack machine, and the policy network is trained. The policy network is a decoder-only transformer with an embedding dimension of 8 attention heads, 4 layers, and a hidden dimension of 4. The value network is a 3-layer MLP with a hidden dimension of 8.

The RL policy is optimized using Proximal Policy Optimization (PPO), as detailed in Appendix B.1.

3. Results

In Section 1.3, we noted that while the symbolic regression search universe Σ^N grows exponentially, only a small fraction $(\Sigma^N)_{\text{valid}}$ is syntactically valid under the stack machine, and an even smaller subset $(\Sigma^N)_{\text{meaningful}}$ exhibits statistically significant factor estimators $\hat{\beta}$, highlighting the necessity for efficient exploration. reportsreportsreportsreports To achieve this, we implemented an RL-based Deep Symbolic Regression model following Petersen et al. (2021), with modified reward, which is based on Sharpe of the in-sample portfolio return.

Given that the RL state is represented as a token sequence without financial performance information, it need to be verified whether the agent merely learns valid syntax or capable of understanding the semantic relation for higher performance. To test this point, we evaluated agents trained under two distinct reward configurations:

- **Sharpe reward**:
  (Sharpe improvement) − (Syntax Penalty) − (Length penalty)
  - Sharpe improvement reflects the incremental test portfolio Sharpe ratio of a candidate factor, discounted by correlation with previously discovered factors.
  - Refer to Appendix B.1 for detailed specifications.
- **Syntax-only reward**:
  (Noise) − (Syntax Penalty) − (Length penalty)
  - The prediction target for the backtest was replaced with Gaussian noise, to remove the Sharpe reward.

Table 1
Agent Training Progress for the Equation Syntax

Episodes	Syntax rewarded agent		Sharpe rewarded agent	
	Trial 1	Trial 2	Trial 1	Trial 2
32,000	7.19%	3.52%	9.38%	5.97%
64,000	26.20%	6.57%	6.25%	8.39%
96,000	45.58%	22.33%	25.00%	17.27%
128,000	53.14%	41.68%	21.88%	44.35%
160,000	49.85%	50.79%	71.88%	45.80%

This table reports the probability of sampling valid and new equations for agents trained with syntax-only reward and additional Sharpe reward, across training episodes. As training episodes increase, agents learn to generate syntactically valid equations.

3.1 Efficiency of sampling syntactically valid equations

To evaluate the syntactic capability of the RL agent, we measured the success rate of generating valid equation across training episodes.

As presented in Table 1, across 160,000 sampled expressions for the training, the probability of generating syntactically valid sequences $(\Sigma^N)_{\text{valid}}$ increases.

For both Sharpe reward and syntax-only reward configurations, the success rate converges to approximately 50% after the training, increasing from the initially low success rate. Two independent runs are examined for reproducibility.

3.2 Diversity of generated alpha pool

To assess the diversity of the generated factor zoo, we evaluated the effective rank based on the explained variance of test portfolio returns.

The return $R_{j,t}$ of the j-th factor portfolio at time t can be expressed as:

$$R_{j,t} = \sum_k (\hat{\beta}^k_{j,t})^T \mathbf{r}^k_t = \sum_k \sum_i (\hat{\beta}^k_{j,t})^T \alpha^k + (\hat{\beta}^k_{j,t})^T \beta^k_{i,t} \lambda_i \tag{4}$$

where $(\hat{\beta}^k_{j,t})^T$ represents the portfolio weight assigned to stock k by the j-th factor estimator, $\beta^k_{i,t}$ is stock k's exposure to risk factor i, and λ_i is the risk premium.

Table 2 reports the cumulative explained variance of test portfolio returns across PCA components. In empirical financial asset pricing, alpha signals are inherently weaker than systematic risk premiums (approximately 15% of total variance relative to risk premiums in Fama and French (2015)). If an insufficient number of latent factors ($K=5$) is specified, residual risk premiums can dominate the factor pool. Applying the standard cumulative eigenvalue criterion targeting $>85\%$ explained variance (following Fama and French (2015)), the required number of latent factors for our machine-generated pool is estimated at $K=30$. Setting $K=30$ successfully controls noise or bias and enhances out-of-sample Sharpe ratios (Table 4).

Table 2
Explained Variance of Test Portfolio Returns Matrix

Factors	1	5	10	20	30	60	100
Sharpe reward	36.12%	60.47%	71.23%	78.62%	82.10%	87.76%	91.84%
Syntax only	56.94%	70.97%	77.07%	83.23%	85.80%	89.20%	91.56%

This table reports the cumulative explained variance of the test portfolio returns matrix across PCA components for factor pools generated by Sharpe-rewarded versus syntax-only agents.

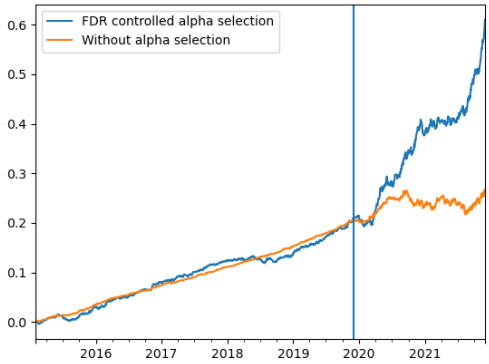


Figure 1
Effect of FDR Alpha Selection on Out-of-Sample Factor Portfolio Returns
Alpha pool was generated using Sharpe-rewarded symbolic regression agent, after training 16,000 episodes. 20 expressions were selected from pool size of 1,000 by FDR-controlled alpha selection with $K = 30$ and $\alpha = 0.01$. Expressions were combined with denoising covariance estimation (De Prado (2020)) and Sharpe maximization using Delatte, Nicolini, and Manzi (2024).

3.3 Alpha Selection on the Machine Generated Pool

Giglio, Liao, and Xiu (2021) proposed a False Discovery Rate (FDR) controlled alpha selection method for fund pools with consideration on multiple testing adjustment.

Goldstein, Spatt, and Ye (2021) highlighted that FDR selection applies naturally to factor research. Based on the idea, we conclude that fund performance can be replaced with the test portfolio returns $R_{j,t}$ of a factor estimator in pool, for alpha research.

We apply FDR-controlled alpha selection to our machine-generated factor pool. Following Section 3.2, the number of latent factors is set to $K = 30$, with significance threshold $\alpha = 0.01$. The result was evaluated with various parameters to test the robustness.

Figure 1 illustrates the cumulative test portfolio return of alpha zoo with 1,000 factors mined with symbolic regression agent, with FDR-controlled alpha selection ($K = 30$, $\alpha = 0.01$), and optimizing in-sample

Table 3
Alpha selection count across latent factor specifications

	Reward	Syntax only		Sharpe rewad	
K	α	Trial 1	Trial 2	Trial 1	Trial 2
0	1.0	1,000	1,000	1,000	1,000
5	0.01	128	153	109	291
	0.05	185	226	197	388
	0.1	243	279	238	451
10	0.01	76	153	74	0
	0.05	122	230	178	9
	0.1	145	294	240	88
30	0.01	22	97	25	30
	0.05	48	215	89	49
	0.1	78	286	129	104
50	0.01	2	22	5	11
	0.05	7	39	31	21
	0.1	9	63	59	37
100	0.01	0	0	14	9
	0.05	0	14	41	18
	0.1	14	23	81	24

This table reports the selection ratio/count of statistically significant alphas from an SR-generated alpha pool (initial pool size of 1,000 factors, mined after 16,000 training episodes) across various numbers of latent factors (K) and FDR significance thresholds (α).

Table 4
Out-of-sample annualized Sharpe ratios across selection parameters

	Reward	Syntax only		Sharpe	
K	α	Trial 1	Trial 2	Trial 1	Trial 2
0	1.0	-0.25	-1.45	0.83	-4.15
5	0.01	-1.46	0.11	-1.57	-2.33
	0.05	-3.59	0.39	-0.27	-3.84
	0.1	-2.68	0.16	-0.53	-2.97
10	0.01	-0.57	-2.36	-0.16	-
	0.05	-0.73	-3.18	-1.09	0.69
	0.1	-0.59	-2.77	-1.88	0.66
30	0.01	2.46	0.51	3.32	3.80
	0.05	2.47	-1.28	0.02	-0.61
	0.1	1.98	-1.07	1.38	-1.67
50	0.01	0.59	-0.19	4.92	-2.72
	0.05	1.92	0.62	1.53	3.06
	0.1	2.20	2.55	1.96	-1.75
100	0.01	-	-	3.62	-3.66
	0.05	-	0.36	0.77	2.45
	0.1	2.58	-1.31	0.11	1.93

This table reports the annualized out-of-sample Sharpe ratio of the combined factor portfolio (pool size of 1,000 factors before alpha selection, mined after 16,000 training episodes) across various numbers of latent factors (K) and FDR significance thresholds (α).

factor combination weights. Convex Sharpe maximization was applied directly across test portfolio return series $R_{j,t}$ of individual factors, without position netting consideration. Improvement of out-sample Sharpe is observed after the alpha selection.

Table 4 and Table 5 report empirical results for the alpha selection effect across diverse parameters and repeated trials. Without alpha

Table 5
Portfolio Sharpe out-of-sample to in-sample ratio

	Reward	Syntax only		Sharpe	
K	α	Trial 1	Trial 2	Trial 1	Trial 2
0	1.0	-0.03	-0.17	0.10	-0.51
5	0.01	-0.31	0.03	-0.44	-0.44
	0.05	-0.69	0.08	-0.06	-0.66
	0.1	-0.46	0.03	-0.10	-0.51
10	0.01	-0.14	-0.46	-0.06	-
	0.05	-0.17	-0.54	-0.24	0.23
	0.1	-0.12	-0.44	-0.40	0.13
30	0.01	0.71	0.11	1.15	1.22
	0.05	0.65	-0.23	0.01	-0.14
	0.1	0.44	-0.18	0.34	-0.34
50	0.01	0.40	-0.06	3.15	-2.29
	0.05	0.87	0.20	0.54	1.26
	0.1	0.99	0.61	0.62	-0.47
100	0.01	-	-	1.64	-2.71
	0.05	-	0.37	0.23	1.15
	0.1	1.07	-0.73	0.03	0.85

This table reports the ratio of out-of-sample Sharpe ratio to in-sample Sharpe ratio (OS/IS ratio) for factor portfolios selected from the factor zoo, across various numbers of latent factors (K) and FDR significance thresholds (α).

selection ($\alpha = 1.0$) or when choosing insufficient latent factors ($K < 30$), candidate factor pools consistently exhibit negative out-of-sample Sharpe ratios, driven by unmitigated risk premium exposures present in raw machine-generated expressions.

At $K = 30$ and $\alpha = 0.01$, out-of-sample Sharpe ratios and OS/IS Sharpe ratios show consistent improvement, with a selected pool size of around 30.

Table 3 presents the count of statistically significant alphas selected by the FDR-controlled significance testing method. At larger K values, agents trained with Sharpe rewards exhibit higher selection ratios compared to syntax-only agents, confirming that RL agents actively learn financially meaningful symbolic combinations.

To summarize, relying solely on backtest return rewards is insufficient to distinguish genuine market anomalies from risk premiums. Applying econometric FDR alpha selection to machine-generated factor pools is essential to eliminate false discoveries and enhance the out-of-sample robustness of the factor zoo.

reports

4. Conclusion

In this paper, alpha significance testing, which is econometric method, is required to increase out-sample consistency of machine-learning based factor pool.

To test our point, we formulated and implemented symbolic regression agent for factor zoo discovery in the U.S. daily equity market,

by introducing Sharpe rewards and syntax penalties. Following the convention of machine learning community, RL agents without an explicit prior factor risk model was used to construct factor zoo. The test portfolio of the factor zoo without alpha selection, exhibits low out-of-sample performance relative to in-sample metrics due to risk premium fitting and data snooping.

We demonstrate that the False Discovery Rate (FDR) controlled alpha selection methodology of Giglio, Liao, and Xiu (2021) can be successfully applied to machine-generated factor pools. As a result, out-of-sample Sharpe ratios and OS/IS performance ratios are increased.

While providing backtest Sharpe rewards during RL policy training does not guarantee that high out-of-sample Sharpe in average, it increases the yield of statistically significant, selected alpha signals within the pool, which implies more efficient factor search.

In summary, while machine learning algorithms can efficiently mine symbolic factor expressions, backtest metrics alone cannot distinguish genuine market anomalies from risk premiums without structural econometric risk modeling. Applying econometric FDR alpha selection to candidate factor pools is essential to eliminate false discoveries and ensure robust out-of-sample factor discovery in empirical asset pricing.

Appendix A. Data and Operators Used

A.1 Backtesting Universe

The stock universe for portfolio backtesting is constructed monthly from equities listed on NASDAQ and NYSE. Assets are included if their data coverage over the preceding 3 months exceeds 0.8 and excluded if coverage drops below 0.2. Stocks lacking ISIN mapping to price records are omitted. Over the evaluation period from February 2, 2015 to November 30, 2021 (1,782 trading days), a universe of 703 stocks was selected.

In-sample symbolic factor backtesting is conducted over the period from February 2, 2015 to November 29, 2019. Out-of-sample verification is evaluated over the period from December 2, 2019 to November 30, 2021.

The prediction target for backtesting is the two-day execution lag close-to-close return ($t+1$ to $t+2$).

A.2 Operators Used

Name	Arity	Input Variable	Input Constant
flip, rank, cs_zscore	Unary	1	0
+, -, /, *	Binary	2	0
delta, ts_nanmean	Binary	1	1

reportsreportsreports

A.3 Input Data

The input data variables available for agent selection are as follows:

```
close
high/close
low/close
volume/adv20
returns
```

Appendix B. Backtesting Methodology

B.1 Signal Normalization and Portfolio Neutralization

To remove systematic market factor risk premiums, cross-sectional demean and side-by-side dollar-neutral scaling are applied during backtesting.

The cross-sectional demean operation for computed symbolic signal x_i is defined as:

$$\tilde{x}_{i,t} = x_{i,t} - \langle x \rangle_t \tag{B1}$$

Long-short neutralization scales the long and short position weights separately:

$$w_i = w_i^{\text{long}} + w_i^{\text{short}}, \quad w_i^{\text{long}} = \frac{\max(\tilde{x}_i, 0)}{\sum_j \max(\tilde{x}_j, 0)}, \quad w_i^{\text{short}} = \frac{\min(\tilde{x}_i, 0)}{\sum_j |\min(\tilde{x}_j, 0)|} \tag{B2}$$

To account for realistic daily closing information availability, a 2-day execution lag is maintained between input feature calculation $(t-1)$ and target return realization.

Appendix C. Neural Net Training

To train the policy network using RL, standard reinforcement learning techniques are incorporated:

First, to address credit assignment and prevent early termination on overly short expressions, we implement Potential-Based Reward Shaping.

Second, to prevent the agent from repeatedly generating identical expressions, a tabu search mechanism discounts candidate Sharpe rewards based on maximum correlation with previously mined factors.

Third, to apply adaptive baselines according to expression complexity, a parameterized value network is used for Generalized Advantage Estimation (GAE).

To summarize, our reward is defined as follows:

$$r(s_t, a_t) = \phi(s_t) - \phi(s_{t-1}) - c_{\text{step}} - c_{\text{unfinished}} \tag{C3}$$

where ϕ is the potential function, $c_{\text{step}} = 0.02$ is the step penalty, and $c_{\text{unfinished}} = 10.0$ is the syntax penalty for failing to produce a valid closed-form expression.

C.1 Policy network loss

The potential function $\phi(s_t)$ measures the marginal Sharpe ratio contribution of expression s_t to test portfolio pool $\mathcal{P}$:

$$\phi(s_t) = (|\mathcal{P}| + 1) \cdot (\text{Sharpe}(\mathcal{P} \cup \{s_t\}) - \text{Sharpe}(\mathcal{P})) \tag{C4}$$

where $\mathcal{P}$ represents the pool of finished equations previously generated by the agent. For incomplete sequence states, mirroring researcher behavior in evaluating sub-expressions to enhance a primary signal, we assign the maximum potential achieved across valid subsequences (or 0 if no valid subsequence exists).

The policy is trained using Proximal Policy Optimization (PPO):

$$L_{\text{total}}(\theta, \phi) = L^{\text{Policy}}(\theta) + L^{\text{Value}}(\phi) \tag{C5}$$

The policy loss $L^{\text{Policy}}(\theta)$ is specified as:

$$L^{\text{Policy}}(\theta) = -\mathbb{E}_t\left[\min\left(\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)} A_t^{\text{GAE}}, \text{clip}\big(\frac{\pi_\theta(a_t \mid s_t)}{\pi_{\theta_{\text{old}}}(a_t \mid s_t)}, 0.2, 5.0\big) A_t^{\text{GAE}}\right)\right] \tag{C6}$$

$$A_t^{\text{GAE}} = \sum_{l=0}^{T-t-1} (\gamma\lambda)^l \delta_{t+l} \tag{C7}$$

$$\delta_t = r(s_t, a_t) + \gamma(1-d_t)V(s_{t+1}) - V(s_t) \tag{C8}$$

$$R_t = A_t^{\text{GAE}} + V(s_t) \tag{C9}$$

Standard PPO baseline $V(s) = E[R_t]$ is replaced by $E[R_t] + \sqrt{\text{Var}[R_t]}$. As noted in Petersen et al. (2021), symbolic factor mining prioritizes discovering rare top-tail alpha signals that surpass the benchmark frontier rather than average expressions. Setting an elevated baseline induces a risk-seeking search bias toward extreme positive Sharpe signals.

C.2 Value network loss

Because executing backtest evaluations across 703 stocks over 1,782 trading days imposes high sampling costs per episode, full sequence attention in the value baseline is replaced by a 5-dimensional manual embedding vector (token entropy, sequence length, data count, operator count, completion flag). The value network is parameterized as a lightweight 3-layer MLP with a hidden dimension of 8, predicting mean μ and standard deviation σ of returns R_t. The value loss L^{Value} is negative log-likelihood for Gaussian distribution.

$$L^{\text{Value}}(\phi) = \mathbb{E}_t\left[\frac{\big(R_t - \mu_\phi(s_t)\big)^2}{2\sigma_\phi(s_t)^2} + \log\sigma_\phi(s_t)\right] \tag{C10}$$

Following Petersen et al. (2021), action masking is applied during sampling:

$$\pi(a_t \mid s_t) = \text{Softmax}(\text{logits}(s_t) + \mathbf{1}_{\{a_t \notin \mathcal{M}(s_t)\}} \cdot (-\infty)) \tag{C11}$$

where $\text{logits}(s_t)$ is computed by the transformer policy layers, and masked action $\mathcal{M}(s_t)$ is stack underflow, which mean that the action is reduction(computation) operator on empty stack(no data introduced), or is a data type(constant or dataframe) mismatch.

References

Astudillo, R. F., M. Ballesteros, T. Naseem, A. Blodgett, and R. Florian. 2020. Transition-based parsing with stack-transformers. In *Findings of the association for computational linguistics: EMNLP 2020*, 1001–7.

De Prado, M. M. L. 2020. *Machine learning for asset managers*. Cambridge University Press.

Delatte, H., C. Nicolini, and M. Manzi. 2024. skfolio. doi:10.5281/zenodo.16148630.

Fama, E. F., and K. R. French. 2015. A five-factor asset pricing model. *Journal of financial economics* 116:1–22.

Giglio, S., Y. Liao, and D. Xiu. 2021. Thousands of alpha tests. *The Review of Financial Studies* 34:3456–96.

Goldstein, I., C. S. Spatt, and M. Ye. 2021. Big data in finance. *The Review of Financial Studies* 34:3213–25.

Harvey, C. R., Y. Liu, and H. Zhu. 2016. ... and the cross-section of expected returns. *The Review of financial studies* 29:5–68.

Kakushadze, Z. 2016. 101 Formulaic Alphas.

Petersen, B. K., M. Landajuela, T. N. Mundhenk, C. P. Santiago, S. Kim, and J. T. Kim. 2021. Deep symbolic regression: Recovering mathematical expressions from data via risk-seeking policy gradients. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*.

Schmidt, M., and H. Lipson. 2009. Distilling free-form natural laws from experimental data. *science* 324:81–5.

Sheikh, A. 1996. Barra's risk models. *Barra Research Insights* 1–24.

Tulchinsky, I. 2015. *Finding alphas: A quantitative approach to building trading strategies*.

Valipour, M., B. You, M. Panju, and A. Ghodsi. 2021. Symbolicgpt: A generative transformer model for symbolic regression .

Yu, S., H. Xue, X. Ao, F. Pan, J. He, D. Tu, and Q. He. 2023. Generating synergistic formulaic alpha collections via reinforcement learning. In *Proceedings of the 29th ACM SIGKDD conference on knowledge discovery and data mining*, 5476–86.