LogicSR: A Unified Benchmark for Logical Discovery from Data

Zimeng Zhang*, Xin Zheng*, Feifei Zhang*, Yunxin Liu, Yuanchun Li 📧

Paper : https://neurips.cc/virtual/2026/loc/sydney/poster/149309

Code and Dataset: https://github.com/deepsymbol/LogicSR

Introduction

- Task: Discover logical expressions from seen/unseen data.
- Logical Symbolic Regression ≈ Symbolic Regression (focuses on continuous functions) + Logic Synthesis (mainly targets exact and noise-free specifications).
- LogicSR: A Unified and Scalable Benchmark for logical SR.
  - 3 datasets (2 real + 1 synthetic, with data imperfection control), 18 methods, 5 metrics.
  - A scalable data generator.

Definitions

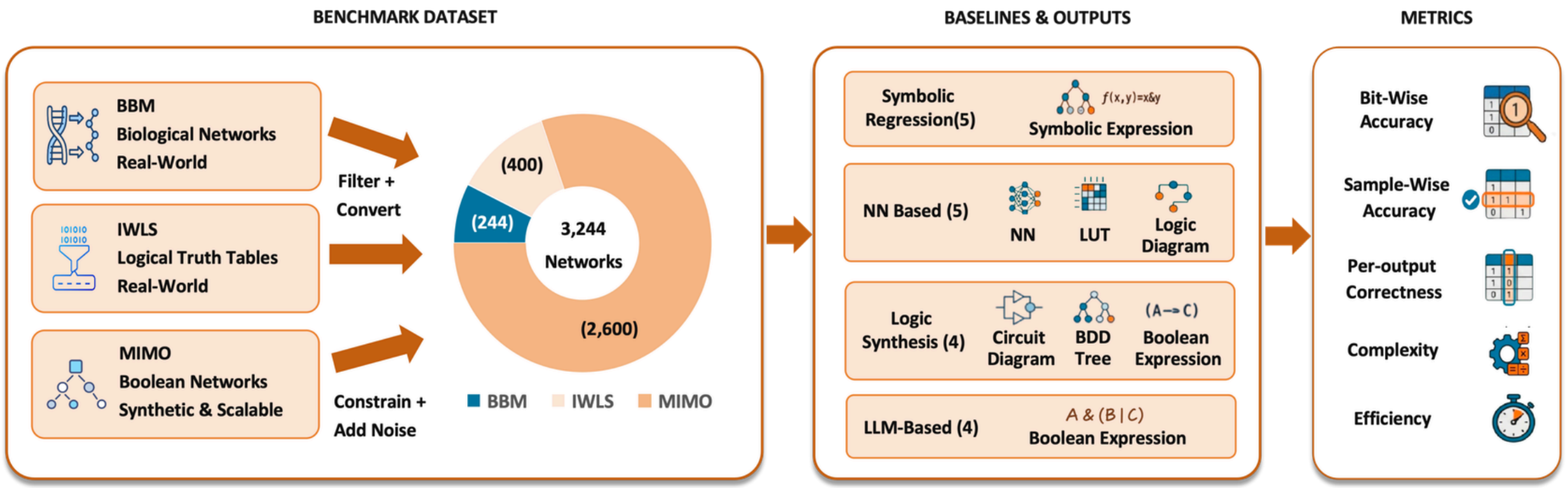
Definition 1 (Logical Symbolic Regression Task).

- Input: Dataset D = {(xi,yi)} with N samples
- Target: Discover Boolean function f that generalizes to seen and unseen inputs.

Definition 2 (Logic Network).

- G = (V, E, I, O)
- V: Nodes (inputs/internal), E $\subseteq$ V × V: Edges (with optional inversion), I $\subseteq$ V: Inputs, O$\subseteq$ V: Outputs.
- Constraints: Connectivity and acyclicity (DAG)

LogicSR Components

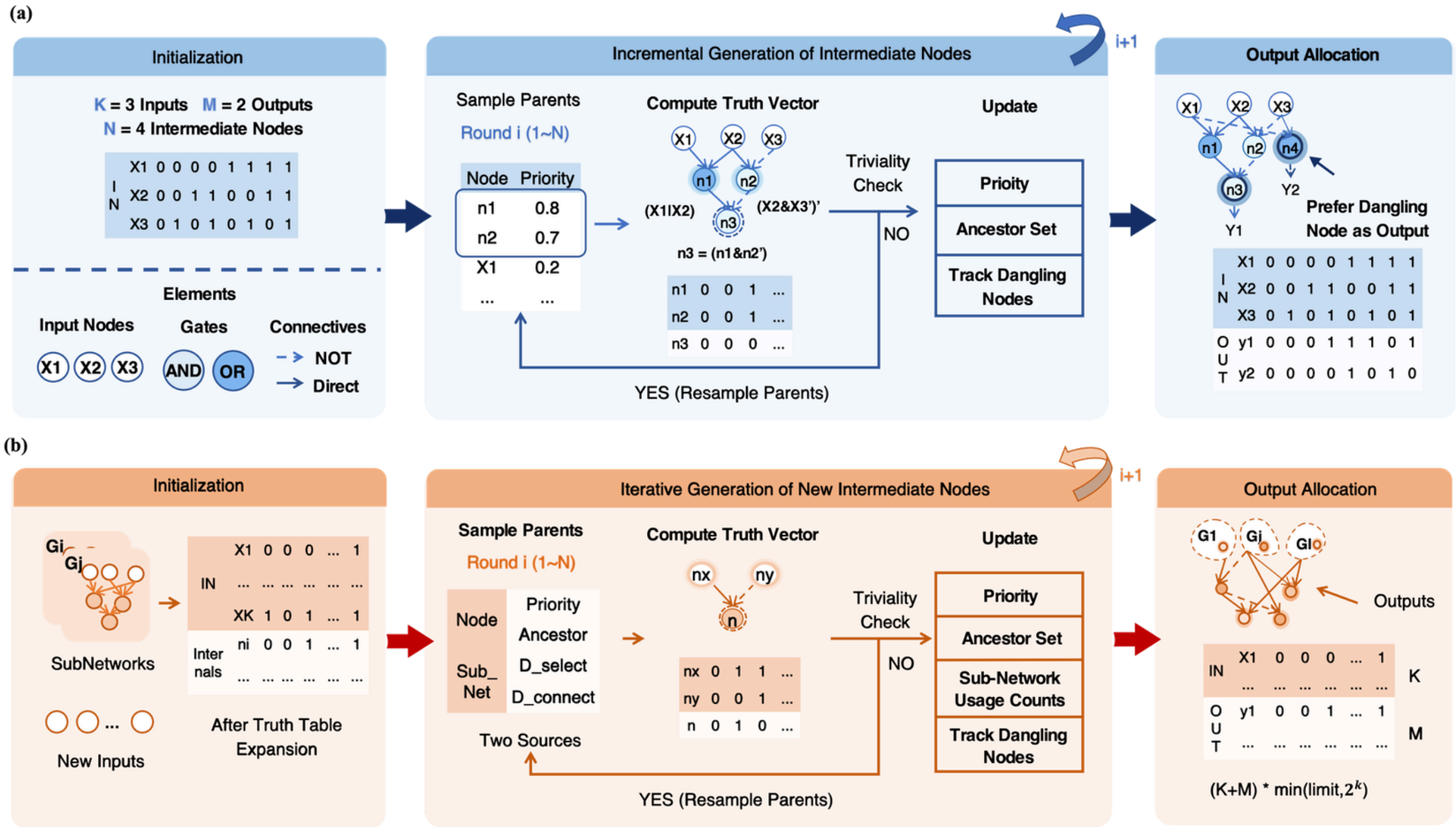


	BBM	IWLS (22'–25')	MIMO Small	MIMO Large
Dataset Type	Real-World	Real-World	Synthetic	Synthetic
Field	Biological Networks	Logic Circuits	Logical Networks	Logical Networks
Number of Samples	244	400	300/300	300/250/150
Input Range	0–223	5–16	5/10	20/40/80
Output Range	0–257	1–115	5/10	20/40/80
Number of Gates	4–853	Around 10–8,000	10–160	160–2,560
Noise Level	0, 0.01, 0.05	0	0, 0.01, 0.05	0

Total: 3,244 networks / 6,132 truth tables (noisy variants).

Generation Framework

The procedure of logic network generation.

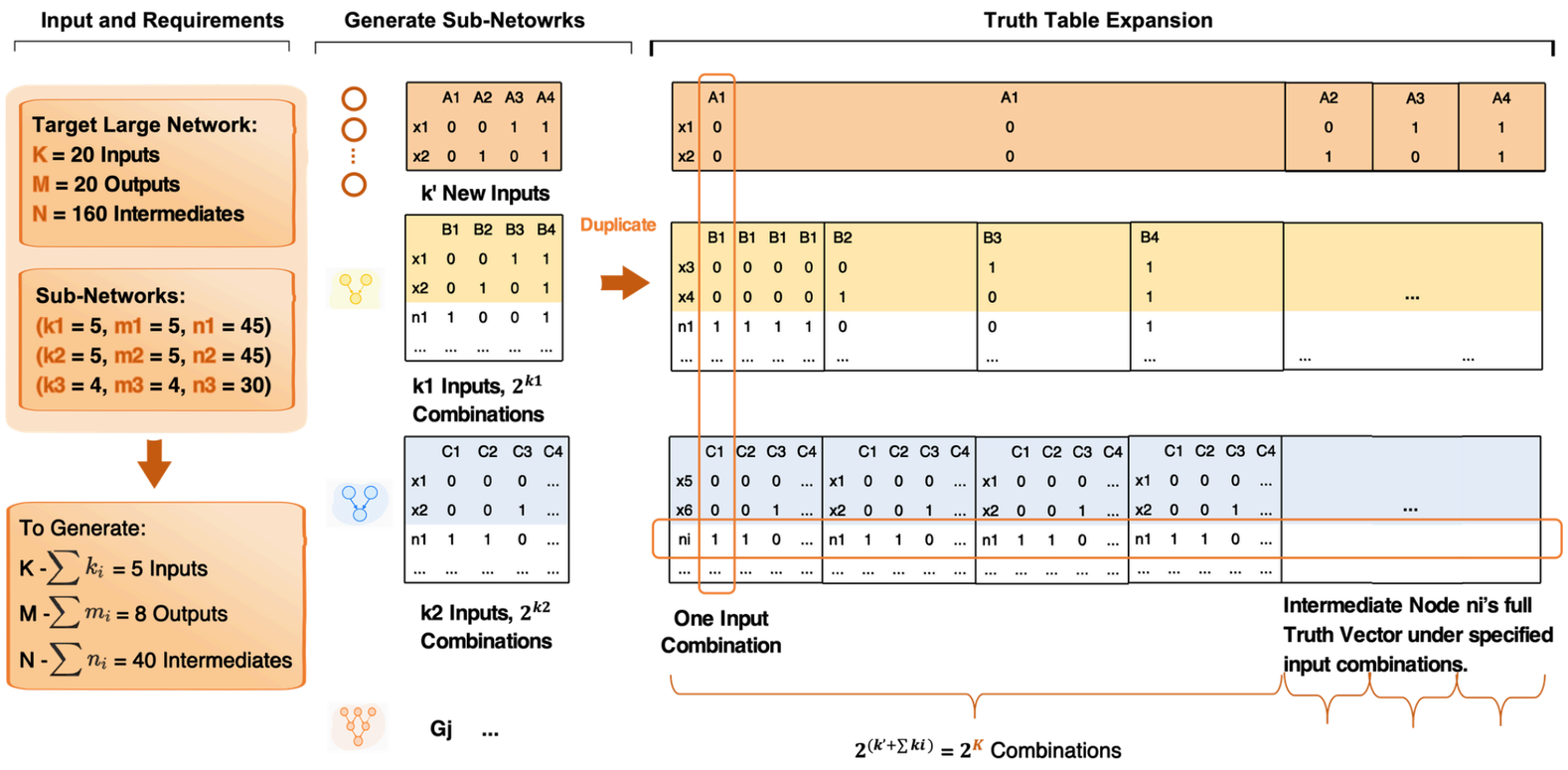


(a) Stage 1 (small-scale generation): produces the building blocks used in large-scale synthesis

(b) Stage 2 (large-scale generation): merges them into complex networks.

Generation Framework

Truth-table Expansion



Noise Injection /
Partial Data Sampling

$$\tilde{y}_j(x) = \begin{cases} y_j(x) \oplus 1, & \text{if } x \in S_j, \\ y_j(x), & \text{otherwise.} \end{cases}$$

	X1	X2	X3	Y1	Y2
Visible	0	0	0	0	1
	0	0	1	1	0
	0	1	0	0	0
	0	1	1	1	1 *
Unseen	...	...	...	...	...

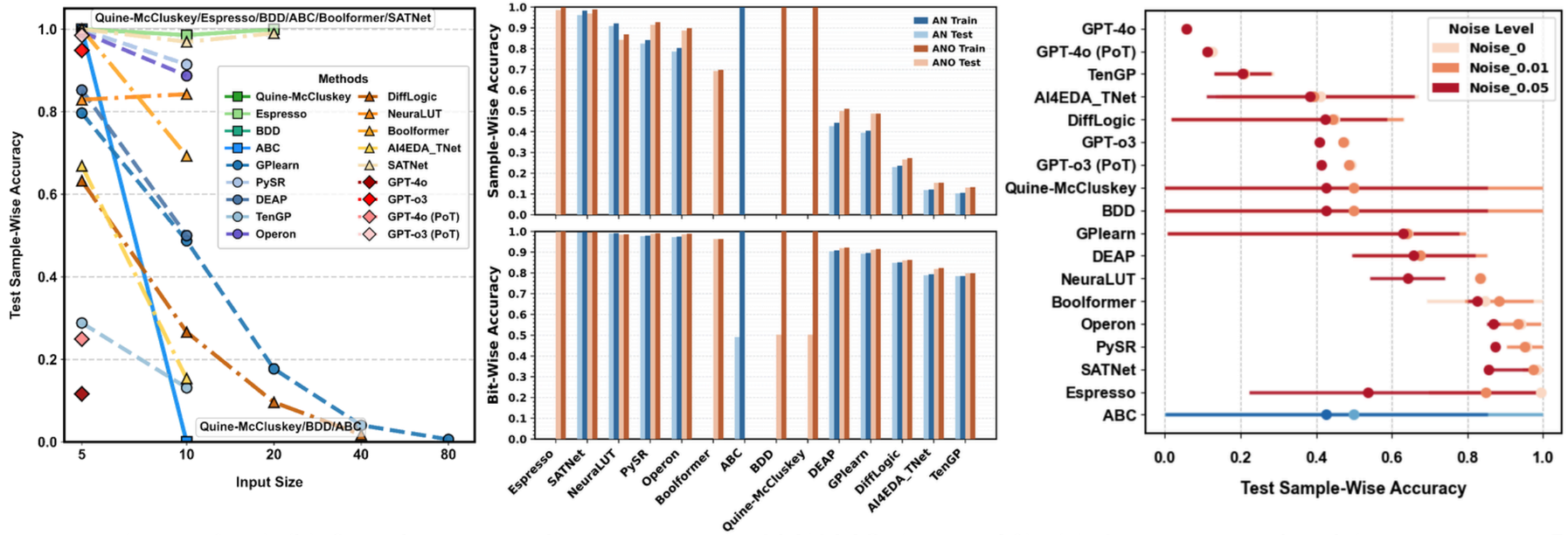
Experiment Results

Methods' performance on datasets (without noise).

- Logic synthesis methods are highly accurate on small tasks, but lacks generalization and scalability.
- SR and NN methods generalize better at scale, but require more computation.
- LLMs struggle with context length on large tasks, though GPT-o3 works well on small ones.

		Synthetic ANO			BBM	Synthetic AN			IWLS
		in5_out5	in10_out10	in20_out20	(ANO)	in5_out5	in10_out10	in20_out20	(AN)
SR	DEAP	51.1 (7.5)	5.9 7.1 (9.2)	–	18.4 18.3 (6.5)	39.0 (5.8)	4.4 5.6 (7.8)	–	5.5 5.5 (17.1)
	GPlearn	36.7 (3.8)	5.7 6.9 (3.2)	4.2 4.3 (1.1)	15.5 15.3 (3.1)	28.5 (3.3)	3.7 4.8 (3.0)	3.3 3.3 (0.9)	12.5 12.6 (5.3)
	TenGP	0.1 (11.7)	0.0 0.0 (8.8)	–	0.0 0.0 (8.2)	0.1 (9.8)	0.0 0.0 (8.3)	–	0.1 0.3 (8.2)
	PySR	99.3 (3.1)	63.6 65.1 (6.6)	–	20.3 20.3 (1.8)	95.7 (3.1)	45.0 46.5 (5.9)	–	24.2 24.3 (6.6)
	Operon	96.9 (10.5)	58.3 59.0 (21.0)	–	25.2 25.2 (5.1)	90.4 (12.4)	42.4 42.9 (22.0)	–	26.7 26.6 (22.7)
NN	DiffLogic	18.2 (110.4)	1.1 2.2 (206.4)	2.3 2.4 (1821.4)	7.4 7.5 (256.2)	15.7 (110.4)	0.7 1.9 (206.4)	2.6 2.6 (1821.4)	6.3 6.3 (614.4)
	AI4EDA_TNet	26.0 (186.4)	2.0 2.8 (192.4)	–	8.7 8.7 (141.2)	28.3 (186.4)	0.3 1.6 (192.4)	–	6.1 6.2 (499.6)
	NeuraLUT	63.7 (6.4)	79.0 70.5 (102.4)	–	28.0 27.1 (51.2)	64.3 (6.4)	82.1 71.7 (102.4)	–	27.2 26.4 (409.6)
	Boolformer	100.0 (5.0)	83.1 80.1 (20.4)	–	18.2 18.2 (3.5)	N/A	N/A	N/A	N/A
	SATNet	100.0 (40.0)	78.1 72.7 (80.0)	72.7 73.9 (160.0)	41.3 40.2 (102.4)	99.8 (40.0)	71.5 66.9 (80.0)	73.2 74.4 (160.0)	32.7 31.8 (121.6)
Logic	ABC	N/A	N/A	N/A	N/A	100.0 (5.8)	100.0 0.8 (501.2)	–	43.3 7.4 (717.6)
	BDD	100.0 (56.8)	100.0 0.5 (5718.9)	–	21.5 10.9 (2416.9)	N/A	N/A	N/A	N/A
	Espresso	100.0 (6.2)	100.0 84.6 (22.4)	100.0 93.7 (66.3)	48.4 48.4 (10.8)	N/A	N/A	N/A	N/A
	Quine-McCluskey	100.0 (6.5)	100.0 0.5 (912.7)	–	21.5 10.8 (363.7)	N/A	N/A	N/A	N/A
LLM	GPT-o3	90.1 (6.9)	–	–	34.1 34.0 (2.8)	88.0 (9.5)	–	–	21.7 21.7 (89.9)
	GPT-4o	1.6 (6.4)	–	–	0.0 0.0 (5.9)	2.1 (5.5)	–	–	0.3 0.3 (23.3)
	GPT-o3 (PoT)	97.1 (41.0)	–	–	39.5 39.4 (30.2)	95.0 (44.8)	–	–	29.2 29.3 (102.6)
	GPT-4o (PoT)	16.8 (21.7)	–	–	5.1 5.1 (10.2)	17.4 (21.5)	–	–	5.4 5.3 (67.1)

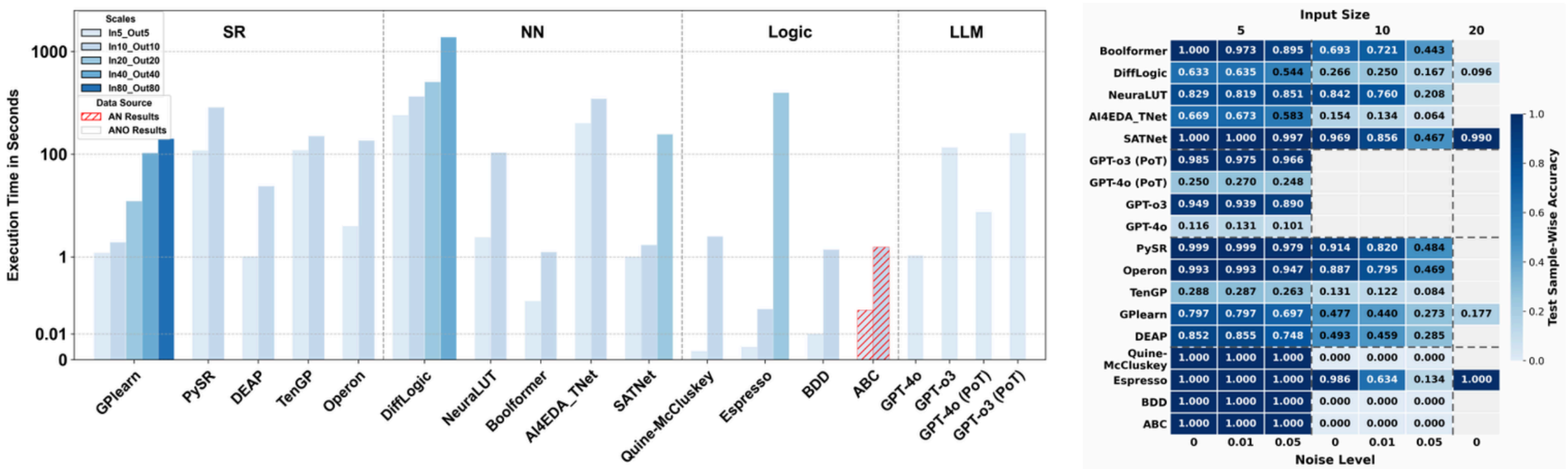
Experiment Visualizations



(a) Acc vs. input size (noise-free) (b) Sample-wise vs. bit-wise acc on AN/ANO (input=10). (c) Acc vs. noise level (avg. on input=5/10).

- Espresso stays accurate up to 20 inputs, whereas ABC, BDD, and Quine–McCluskey fail at scale.
- Bit-wise accuracy is higher than sample-wise accuracy, as a single bit error ruins the whole sample.
- Symbolic regression is robust to noise on larger inputs, but logic synthesis methods are highly sensitive.

Experiment Visualizations



(a) Computational efficiency under different scales; (b) Sample-wise fidelity under varying inputs and noise. (ABC* on AN, others on ANO.)

- SR methods face a massive runtime increase as input scales grow.
- Traditional logic synthesis is highly efficient on small tasks, achieving the shortest runtime.
- GP and NN methods struggle at scale, and traditional methods (ABC, BDD, QM) generalize poorly.

Conclusion

Conclusion

- LogicSR establishes a unified and scalable benchmark for evaluating logical symbolic regression across diverse methods and tasks.
- Reveals key challenges in scalability, noise robustness, and logical operator compatibility.

Limitations and Future Work

- Structural Constraints: Binary fan-in limits expressiveness. ➡️ Add multi-fanin nodes and industrial operators.
- Scalability Constraints: Limited to practical-scale tasks. ➡️ Enable benchmarking for ultra-large networks.
- Domain Applications: Synthetic data lacks domain-specific biases. ➡️ Add related logic (e.g., temporal logic).

THANK YOU!

Please reach out to us at: liyuanchun@air.tsinghua.edu.cn