Evaluation of QCardEst/QCardCorr on JOB-light and STATS Benchmarks

Katrin Schwab

21.12.2025

Abstract

This report reviews and extends the evaluation of the hybrid quantum–classical models QCardEst and QCardCorr proposed by Winker et al. [5]. We summarize the experimental setting and compare performance across two established workloads: JOB-light and STATS. The analysis focuses on how the choice of post-processing layer affects accuracy for direct estimation (QCardEst) versus correction (QCardCorr), and highlights which layer families are most robust under different workload characteristics.

Contents

1 Scope and Constraints

1.1 Quantum Simulation Environment

First, the results are obtained using a quantum circuit simulator rather than real quantum hardware. This has several important implications:

- **No hardware noise**: Simulators produce ideal quantum states. The results reflect model capacity and the effectiveness of the hybrid approach, not hardware limitations or noise characteristics.
- **Prediction quality focus**: The evaluation focuses on prediction quality, not execution time. We do not claim quantum speedup or runtime advantages. The work evaluates whether quantum models can improve cardinality estimation accuracy.

1.2 Experimental Setup

Queries join up to 6 tables, matching 6 qubits (one qubit per table; Compact encoding), which keeps the simulation realistic and feasible on today's quantum hardware, while still being complex enough to stress the model. [5]

The training process involves a hybrid quantum-classical optimization loop:

- A Variational Quantum Circuit (VQC) is executed repeatedly
- After each execution, parameters are updated using the Adam optimizer with a decaying learning rate
- The optimization runs for 8000 episodes

Each training episode involves:

- Feature encoding
- Simulating a **6-qubit, 16-layer VQC**
- Measuring the circuit to obtain probability distributions
- Classical post-processing through various layer types
- Computing gradients and updating parameters

This hybrid optimization loop makes training computationally expensive, but enables end-to-end learning of the quantum-classical pipeline.

2 Benchmark Descriptions

Before examining the actual results, it is important to understand what kind of data the model is tested on, because this has a big impact on how difficult the task actually is.

In database systems, not all workloads are equally complex. Some queries follow patterns that optimizers can solve efficiently, while others are intentionally designed to break assumptions and

expose weaknesses. To cover both cases, the research group evaluated the QCardEst/QCardCorr models on two different benchmarks: JOB-light and STATS.

Both JOB-light and STATS are widely used benchmarks for evaluating cardinality estimation methods and query optimizers, but they differ significantly in their underlying data and query complexity.

2.1 JOB-light

JOB-light [3] is derived from the real-world IMDB dataset. It contains:

- Natural correlations between attributes
- Skewed distributions that reflect realistic data characteristics
- Real-world query patterns

Despite being realistic, these natural patterns are often handled reasonably well by modern optimizers like PostgreSQL, which is why JOB-light is generally considered the easier of the two benchmarks. [1]

2.2 STATS

STATS [2], in contrast, is based on a synthetic but carefully designed dataset. Its purpose is not realism, but control:

- Data distributions are constructed to systematically test different types of predicates
- Designed to test various selectivities and filter combinations
- Provides controlled stress testing scenarios

Because of this controlled design, STATS is typically much harder for traditional optimizers and simpler learned models. It reveals weaknesses that might never appear when working only with naturally correlated real-world data. [4]

2.3 Evaluation Strategy

Using both benchmarks provides a balanced evaluation, combining realistic workloads with controlled stress testing. We use the same hybrid quantum–classical pipeline for both benchmarks, so any performance differences observed are due to workload characteristics rather than changes in the model architecture or training procedure.

3 Experimental Results

This section presents the experimental results comparing QCardEst (cardinality estimation) and QCardCorr (cardinality correction) approaches across nine different classical post-processing layers.

3.1 JOB-light Benchmark Results

Figure 1 shows the comparison of all classical layers for the JOB-light benchmark.

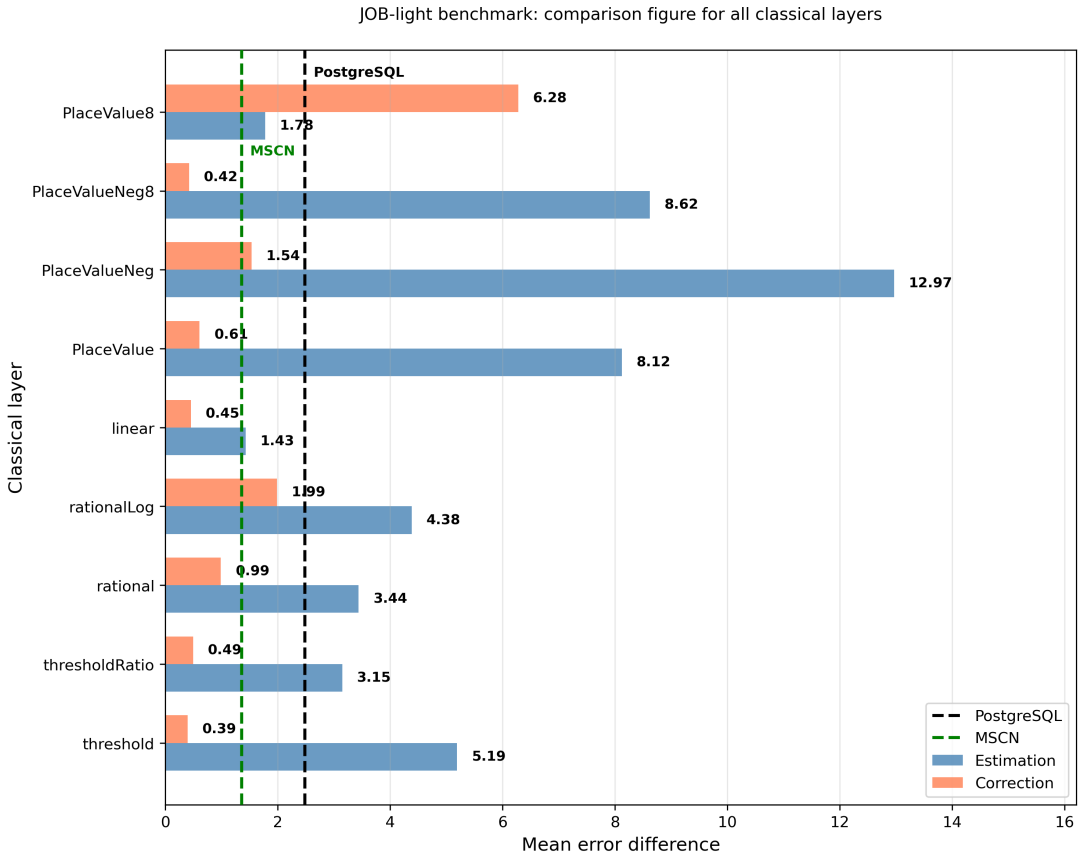

Figure 1: JOB-light benchmark: comparison figure for all classical layers. The chart shows mean error difference for Estimation (blue bars) and Correction (orange bars) approaches. Dashed lines indicate PostgreSQL (black, 2.48) and MSCN (green, 1.35) baseline errors.

4 JOB-light Benchmark: Detailed Analysis

4.1 Overview

For JOB-light, the **Threshold** layer achieves the best correction performance with a mean error difference of **0.39**, representing a remarkable improvement over both baseline estimators. This performance is **6.37 times better than the PostgreSQL baseline** (2.48) and **3.47 times better than the MSCN baseline** (1.35). The second-best performer is **PlaceValueNeg8** with an error of **0.42**, which is **5.91 times better than PostgreSQL** and **3.22 times better than MSCN**.

When examining all correction approaches, the research group observes that in **6 out of 8 cases where cardinality correction outperforms PostgreSQL, it also outperforms MSCN**. This pattern suggests that the quantum correction mechanisms are capable of systematically improving upon both classical estimators, with the improvement being more pronounced relative to PostgreSQL's estimates.

4.2 Why Threshold achieves the best performance compared to all other layers for JOB-light

4.2.1 Mechanism

The threshold layer (`SecondValueThreshold`) uses a gating mechanism based on ReLU activations with a threshold at 0.25:

```
posChange = 1 + ReLU(x[0] - 0.25) * x[1] * scalar²
negChange = 1 + ReLU(x[2] - 0.25) * x[3] * scalar²
result = posChange - negChange
```

4.2.2 Why It Works Well for Correction

1. **Selective Application**: The threshold mechanism only applies corrections when the quantum output exceeds the threshold, effectively filtering out noise and only making corrections when there is sufficient confidence in the quantum model's output.
2. **Bidirectional Corrections**: By combining positive and negative changes through subtraction, the threshold layer can handle both overestimations and underestimations from the baseline (PostgreSQL) estimator.
3. **Stability**: The threshold acts as a regularization mechanism, preventing the model from making unnecessary corrections when the baseline is already accurate. This is particularly important for correction tasks where many queries may already have reasonable estimates.

4.2.3 Key Observations for JOB-light

- **Correction dominance**: All correction approaches except PlaceValue8, PlaceValueNeg and rationalLog outperform both PostgreSQL and MSCN baselines. Since multiple independent correction layers show the same trend, this shows systematic improvement against the baselines.
- **Estimation limitations**: For direct cardinality estimation, only **Linear** and **PlaceValue8** perform better than the PostgreSQL baseline. All other layers fail to improve upon PostgreSQL's estimates.
- **PlaceValue8 anomaly**: PlaceValue8 shows dramatically different behavior between correction (6.28 error, worst performer) and estimation (1.78 error, second-best). This stems from a fundamental limitation: PlaceValue8 can **only produce positive numbers**, meaning it can only correct cardinalities by **increasing them**. Since PostgreSQL estimates often need downward corrections, PlaceValue8 is unable to make the necessary adjustments, leading to poor correction performance. However, for estimation, this constraint is less problematic since cardinalities are inherently positive values.
- **PlaceValueNeg8 contrast**: The variant PlaceValueNeg8, which allows for negative numbers, performs excellently for correction (0.42, second-best) but poorly for estimation (8.62). This inverse relationship between PlaceValue8 and PlaceValueNeg8 highlights the importance of **bidirectional correction capability** for improving baseline estimates, while simpler positive-only representations work better for absolute value estimation.

5 STATS Benchmark: Detailed Analysis

5.1 Overview

For STATS, the performance landscape differs significantly from JOB-light. The best correction performance is achieved by **RationalLog** with a mean error difference of **0.32**, which is **8.66 times better than the PostgreSQL baseline** (2.77). The second-best is **Rational** with an error of **1.04**, representing a **2.67 times improvement** over PostgreSQL. Notably, the Threshold layer, which dominated JOB-light, achieves 1.96 error (still better than PostgreSQL, but not the best).

This shift in optimal layer choice between benchmarks suggests that **the effectiveness of correction layers is sensitive to query workload characteristics**.

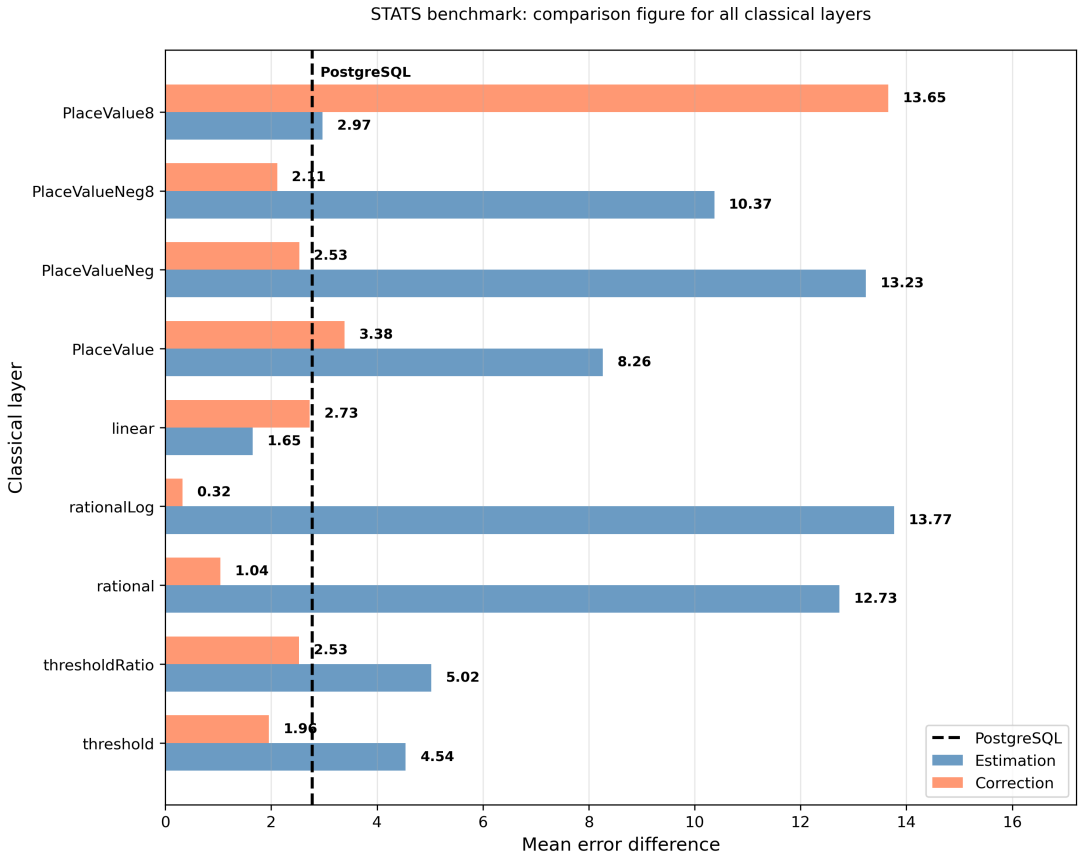

Figure 2: STATS benchmark: comparison figure for all classical layers. The chart shows mean error difference for Estimation (blue bars) and Correction (orange bars) approaches. Dashed line indicates PostgreSQL (black, 2.77) baseline error.

Figure 2 shows the comparison of all classical layers for the STATS benchmark.

5.2 Key Findings

The STATS benchmark reveals several important patterns:

1. **Rational layers excel**: RationalLog and Rational achieve the best correction performance, with RationalLog showing an exceptional 8.66× improvement over PostgreSQL. This represents the largest improvement factor observed across both benchmarks.

2. **Linear leads estimation**: As in JOB-light, Linear maintains its position as the best estimation layer (1.65 vs 1.43 in JOB-light), and closely followed by PlaceValue8 (2.97), it is the only layer that outperforms the PostgreSQL baseline for direct estimation.

3. **Higher baseline error**: PostgreSQL baseline error is 2.77 for STATS vs 2.48 for JOB-light, indicating that STATS queries may be inherently more challenging for classical estimators, yet still allowing for significant quantum-based improvements.

4. **Correction vs Estimation gap**: The gap between best correction (0.32) and best estimation in STATS is 1.33, which is larger than the gap in JOB-light (1.04). For STATS, the gap is much larger because this benchmark is harder for classical methods like PostgreSQL and they make bigger errors. Correcting bigger errors gives the largest learning benefit, which is why the correction approach achieves very low error on STATS.

6 Comparison: JOB-light vs STATS

6.1 Absolute Performance Differences

Metric	JOB-light	STATS	Difference
PostgreSQL Baseline	2.48	2.77	+0.29 (STATS harder)
Best Correction (threshold/rationalLog)	0.39	0.32	-0.07 (STATS better)
Best Estimation (linear)	1.43	1.65	+0.22 (STATS harder)
Gap (Best Correction - Best Estimation)	1.04	1.33	+0.29 (STATS larger gap)

6.2 Key Differences

1. **Error Magnitude**: STATS shows overall higher errors across all layers, suggesting the benchmark is inherently more challenging.

2. **Correction Advantage**: The advantage of correction over estimation is actually larger in STATS (1.33) compared to JOB-light (1.04). This suggests that:

   - Correction approaches can achieve even larger improvements in STATS (RationalLog: 8.66× better than PostgreSQL) compared to JOB-light (Threshold: 6.37× better)
   - The best correction in STATS (0.32) is actually better than the best correction in JOB-light (0.39), despite STATS being more challenging overall
   - Different layer architectures (RationalLog vs Threshold) are optimal for different workloads

3. **Layer Robustness**: The relative ranking of layers shows some variation between benchmarks:

   - **Correction**: Threshold excels in JOB-light while RationalLog dominates STATS, though PlaceValueNeg8 remains strong in both
   - **Estimation**: Linear and PlaceValue8 maintain their leading positions in both benchmarks
   - This variation suggests that optimal layer choice should consider workload characteristics

4. **Baseline Quality**: Both benchmarks have similar PostgreSQL baseline errors (2.48 vs 2.77), yet correction approaches achieve larger improvements in STATS compared to JOB-light, demonstrating that optimal layer choice can unlock significant correction potential.

6.3 Similarities

1. **Linear estimation dominance**: Linear layer consistently performs best for estimation in both benchmarks (1.43 JOB-light, 1.65 STATS), validating its simplicity and effectiveness.
2. **PlaceValueNeg8 reliability**: PlaceValueNeg8 consistently ranks high for correction, showing its reliability across different workloads.
3. **PlaceValue8 constraint**: PlaceValue8 consistently fails at correction due to its positive-only output constraint.
4. **Estimation challenges**: Only Linear and PlaceValue8 outperform or come very close to PostgreSQL for estimation, while most other layers struggle with absolute value estimation in both benchmarks.

7 When to Pick Which Layer (Rule of Thumb)

7.1 For Correction Tasks

For JOB-light workloads: Threshold

- Best performance (0.39, 6.37× better than PostgreSQL)
- Robust and interpretable selective correction mechanism
- Use when: Working with JOB-light-style queries and PostgreSQL baseline

For STATS workloads: RationalLog

- Best performance (0.32, 8.66× better than PostgreSQL)
- Exceptional improvement factor
- Use when: Working with STATS-style queries and need maximum correction accuracy

7.2 For Estimation Tasks

Linear

- Best performance for estimation in both benchmarks (1.43 JOB-light, 1.65 STATS)
- Use when: You must do direct estimation without a baseline or when the baseline is not good enough

7.3 Cross-cutting Insights and Principles

- **Cardinality correction is more robust than direct estimation**: The consistent superiority of correction approaches across all layers and benchmarks demonstrates that leveraging baseline estimates and applying selective adjustments is fundamentally more effective than estimating absolute cardinalities from scratch.
- **Expressiveness of the classical post-processing layer is critical**: The dramatic performance differences between layers (e.g., Threshold vs. PlaceValue8 for correction) show that the choice of classical post-processing architecture is as important as the quantum circuit design itself.

7.4 Key Takeaways

- **Hybrid quantum–classical models outperform classical baselines**: The 6–8× improvements over PostgreSQL and 3–4× improvements over MSCN show that quantum-enhanced approaches can provide substantial accuracy gains when properly integrated with classical methods.
- **Cardinality correction is a realistic near-term application**: Since the hybrid models rely on compact encoding, shallow variational quantum circuits with a small number of qubits, quantum correction of classical cardinality estimates is theoretically feasible on current and near-term quantum hardware. QCardCorr aligns well with the capabilities of today's noise-prone quantum processors.

8 Future Work

Building on the insights from this evaluation, several promising directions for future research emerge across different time horizons.

8.1 Short-term

- **Evaluation on larger benchmarks**: Extending the evaluation to benchmarks with more complex query patterns, larger numbers of tables, and more diverse data distributions would provide further validation of the approach and help identify scalability limits.
- **Automatic learning of classical post-processing layers**: Rather than manually selecting from predefined layer architectures, developing methods to automatically learn or discover optimal classical post-processing layers could further improve performance.

8.2 Mid-term

- **Integration into real DBMS pipelines**: Moving from simulation to integration with actual database management systems would enable evaluation of end-to-end query optimization performance and assessment of practical deployment considerations.
- **Execution on real quantum hardware**: Transitioning from quantum simulators to real quantum hardware would reveal the impact of noise, gate errors, and other hardware limitations on model performance, providing crucial insights for practical deployment.

8.3 Long-term

- **Joint optimization of join order and cardinality estimation**: A quantum-enhanced model could, in principle, reason over entire query structures rather than individual estimates. This would enable joint optimization of join order selection and cardinality estimation, potentially leading to more globally optimal query plans.
- **Deeper and more expressive quantum models as hardware scales**: As quantum hardware continues to improve in terms of qubit count, gate fidelity, and coherence times, opportunities emerge for deploying deeper and more expressive quantum circuits. This could enable more sophisticated feature encodings and potentially unlock new capabilities for query optimization.

References

[1] *Comparison of JOB-light and STATS-CEB Benchmark Workloads*. https://www.researchgate.net/figure/Comparison-of-JOB-LIGHT-and-STATS-CEB-bench-mark-query-workload_tbl2_354571909. Accessed December 2025.

[2] Yuxing Han et al. "Cardinality Estimation in DBMS: A Comprehensive Benchmark Evaluation". In: *arXiv preprint arXiv:2109.05877* (2021). URL: https://arxiv.org/abs/2109.05877.

[3] Viktor Leis et al. "How Good Are Query Optimizers, Really?" In: *Proceedings of the VLDB Endowment* 9.3 (Nov. 2015), pp. 204–215. DOI: 10.14778/2850583.2850594.

[4] *PostgreSQL Query Planner and Statistics*. https://www.postgresql.org/docs/current/planner-stats.html. Accessed December 2025.

[5] Tobias Winker, Jinghua Groppe, and Sven Groppe. "QCardEst/QCardCorr: Quantum Cardinality Estimation and Correction". In: *Proceedings of the ACM SIGMOD Workshops*. New York, NY, USA: ACM, 2025.