

Benchmark Results: Interpretation

QCardEst/QCardCorr

December 20, 2025

1 Overview

This document provides a detailed interpretation of the JOB-light and STATS benchmark results comparing QCardEst (cardinality estimation) and QCardCorr (cardinality correction) approaches across different classical post-processing layers.

The comparison shows that **correction-based approaches (QCardCorr) consistently outperform estimation-based approaches (QCardEst)** across all classical layers and both benchmarks. This is evident from the significantly lower error values for correction compared to estimation.

2 JOB-light Benchmark Results

2.1 Overview

For JOB-light, the threshold layer achieves the best correction performance with a mean error difference of **0.39**, outperforming all other layers including the MSCN baseline (1.35) and PostgreSQL baseline (2.48).

2.2 Why Threshold Correction Wins

The **threshold** layer achieves the best correction performance with a mean error difference of **0.39**, outperforming all other layers including the MSCN baseline (1.35) and PostgreSQL baseline (2.48).

2.2.1 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
```

2.2.2 Why It Works Well for Correction

1. **Selective Application:** The threshold mechanism only applies corrections when the quantum output exceeds 0.25, 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. **Baseline Leverage:** Correction approaches benefit from leveraging existing knowledge (PostgreSQL's estimates) and only correcting when necessary. The threshold mechanism aligns perfectly with this philosophy by only activating corrections above a confidence threshold.

The threshold layer's success in correction (0.39 error) compared to estimation (5.19 error) demonstrates that **correction is fundamentally easier than full estimation** when a reasonable baseline exists.

2.3 Why PlaceValueNeg Variants Matter

The **PlaceValueNeg** variants show interesting and contrasting behaviors between estimation and correction:

- **PlaceValueNeg (estimation):** 12.97 error - the worst performing layer for estimation
- **PlaceValueNeg (correction):** 1.54 error - moderate performance for correction
- **PlaceValueNeg8 (estimation):** 8.62 error - still poor for estimation
- **PlaceValueNeg8 (correction):** 0.42 error - second best for correction after threshold

2.3.1 Mechanism

PlaceValueNeg uses a weighted sum where values are encoded as powers of $(1 + \text{scalar})$:

```
factors = [(1+scalar), (1+scalar)1, ..., (1+scalar)(n/2-1)]  
if negativ:  
    factors = [factors, -factors] # Includes both positive and negative  
result = sum(x * factors)
```

2.3.2 Why They Matter

1. **Representational Capacity:** The place-value encoding allows the model to represent corrections at different scales. The negative variant is crucial because it enables **bidirectional corrections** - the model can both increase and decrease the baseline estimate.
2. **Scale Sensitivity:** For estimation tasks, the place-value system struggles because it needs to represent absolute cardinalities spanning many orders of magnitude. However, for correction, the system only needs to represent **multiplicative factors** (typically close to 1.0), which is a much easier problem.
3. **Granularity Trade-off:** PlaceValueNeg8 (with 8 inputs) performs better than PlaceValueNeg (with 4 inputs) for correction (0.42 vs 1.54), showing that more granular encoding helps. However, this comes at a cost for estimation where PlaceValueNeg8 still performs poorly (8.62 vs 12.97).
4. **Correction Symmetry:** The negative factors in PlaceValueNeg variants allow symmetric corrections - the model can represent both “the baseline is $2\times$ too high” and “the baseline is $2\times$ too low” with similar representational complexity. This symmetry is less important for absolute estimation where values are always positive.

The dramatic improvement of PlaceValueNeg8 for correction (from 8.62 to 0.42) compared to estimation highlights that **the correction task is well-suited to multiplicative place-value representations**.

2.4 JOB-light Performance Summary

2.4.1 Correction (QCardCorr) Ranking:

1. **threshold:** 0.39 Best
2. PlaceValueNeg8: 0.42
3. thresholdRatio: 0.49
4. linear: 0.45
5. rational: 0.48
6. PlaceValue: 0.61
7. rationalLog: 0.93
8. PlaceValueNeg: 1.54
9. PlaceValue8: 6.28 (outlier - likely configuration issue)

2.4.2 Estimation (QCardEst) Ranking:

1. **linear**: 1.43 Best
2. PlaceValue8: 1.78
3. thresholdRatio: 3.15
4. rational: 5.11
5. threshold: 5.19
6. PlaceValue: 8.12
7. PlaceValueNeg8: 8.62
8. rationalLog: 10.53
9. PlaceValueNeg: 12.97

3 STATS Benchmark Results

3.1 Overview

For STATS, the threshold layer again achieves the best correction performance with a mean error difference of **1.96**, outperforming the PostgreSQL baseline (2.77). While absolute errors are higher than JOB-light, the relative patterns and layer rankings remain largely consistent.

3.2 Key Findings

The STATS benchmark shows similar patterns to JOB-light but with higher absolute error values across all layers:

1. **Threshold remains best for correction:** Despite higher absolute errors (1.96 vs 0.39), threshold still leads correction performance.
2. **Linear leads estimation:** Linear maintains its position as the best estimation layer (1.65 vs 1.43 in JOB-light).
3. **Consistent layer rankings:** The relative performance ordering of layers is largely preserved between benchmarks, suggesting the layer characteristics are robust across different query workloads.
4. **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.

3.3 STATS Performance Summary

3.3.1 Correction (QCardCorr) Ranking:

1. **threshold:** 1.96 Best
2. PlaceValueNeg8: 2.11
3. rationalLog: 2.31
4. rational: 2.46
5. thresholdRatio: 2.53
6. PlaceValueNeg: 2.53
7. linear: 2.73
8. PlaceValue: 3.38
9. PlaceValue8: 13.65 (outlier - likely configuration issue)

3.3.2 Estimation (QCardEst) Ranking:

1. **linear:** 1.65 Best
2. PlaceValue8: 2.97
3. threshold: 4.54
4. thresholdRatio: 5.02
5. rational: 7.37
6. PlaceValue: 8.26
7. PlaceValueNeg8: 10.37
8. rationalLog: 12.72
9. PlaceValueNeg: 13.23

3.4 STATS-Specific Observations

1. **Tighter correction gap:** The gap between best correction (1.96) and best estimation (1.65) is smaller in STATS than in JOB-light (0.39 vs 1.43), suggesting that correction advantage may be benchmark-dependent.
2. **RationalLog performance:** RationalLog performs relatively better in STATS correction (2.31, ranking 3rd) compared to JOB-light (0.93, ranking 7th), indicating it may be better suited for certain query characteristics.
3. **PlaceValue8 outlier:** Like in JOB-light, PlaceValue8 shows an outlier performance in correction (13.65), suggesting a potential configuration or implementation issue with this specific variant.

4 Comparison: JOB-light vs STATS

4.1 Absolute Performance Differences

Metric	JOB-light	STATS	Difference
Best Correction (threshold)	0.39	1.96	+1.57 (STATS harder)
Best Estimation (linear)	1.43	1.65	+0.22 (STATS harder)
PostgreSQL Baseline	2.48	2.77	+0.29 (STATS harder)
Correction Advantage	1.04	0.31	-0.73 (smaller gap)

4.2 Key Differences

1. **Error Magnitude:** STATS shows consistently higher errors across all layers, suggesting the benchmark is inherently more challenging. This could be due to:
 - More complex query patterns
 - Different data distributions
 - Larger cardinality ranges
 - More join combinations
2. **Correction Advantage:** The advantage of correction over estimation is smaller in STATS (0.31) compared to JOB-light (1.04). This suggests that:
 - STATS baseline (PostgreSQL) may be more accurate relative to the query difficulty
 - Correction has less room for improvement when baseline error is already relatively low
 - Estimation approaches may be more competitive when queries are inherently harder
3. **Layer Robustness:** Despite absolute differences, the relative ranking of layers is largely preserved:
 - **Correction:** threshold and PlaceValueNeg8 remain top performers in both
 - **Estimation:** linear and PlaceValue8 maintain their leading positions
 - This consistency suggests layer characteristics are robust across benchmarks
4. **Baseline Quality:** Both benchmarks have similar PostgreSQL baseline errors (2.48 vs 2.77), but STATS shows less room for improvement through correction, possibly indicating that the baseline errors are distributed differently.

4.3 Similarities

1. **Threshold dominance:** Threshold layer consistently performs best for correction in both benchmarks, confirming its robustness.
2. **Linear estimation:** Linear layer consistently performs best for estimation, validating its simplicity and effectiveness.
3. **PlaceValueNeg8 strength:** PlaceValueNeg8 consistently ranks second for correction, showing its reliability.
4. **Estimation struggles:** PlaceValueNeg variants consistently perform poorly for estimation in both benchmarks.

5 When to Pick Which Layer: Rule of Thumb

Based on the combined results from both benchmarks, here is a practical guide for selecting classical layers:

5.1 For Correction Tasks (QCardCorr - rowFactor)

Primary Recommendation: threshold

- Best overall performance across both benchmarks (0.39 JOB-light, 1.96 STATS)
- Robust and interpretable
- Works well when you have a reasonable baseline
- Use when: Correction is the primary goal and you want the best accuracy

Alternative: PlaceValueNeg8

- Consistently second best (0.42 JOB-light, 2.11 STATS)
- More expressive than threshold
- Use when: You need fine-grained control over correction scales

Avoid for correction:

- PlaceValue variants (non-negative): Poor performance (0.61-3.38)
- Linear: Moderate but not competitive (0.45 JOB-light, 2.73 STATS)
- PlaceValue8: Shows outlier behavior (6.28 JOB-light, 13.65 STATS)

5.2 For Estimation Tasks (QCardEst - rows)

Primary Recommendation: linear

- Best performance for estimation in both benchmarks (1.43 JOB-light, 1.65 STATS)
- Simple and efficient
- Use when: You must do direct estimation without a baseline

Alternative: PlaceValue8

- Good performance (1.78 JOB-light, 2.97 STATS)
- More expressive than linear
- Use when: You need more representational capacity than linear provides

Avoid for estimation:

- PlaceValueNeg variants: Poor performance (8.62-13.23) - negative factors add complexity without benefit
- RationalLog: Poor performance (10.53 JOB-light, 12.72 STATS) - logarithmic ratio encoding doesn't work well for absolute values
- Threshold: Moderate (5.19 JOB-light, 4.54 STATS) but significantly worse than linear

5.3 General Principles

1. **Correction is easier than estimation:** If possible, use correction (QCard-Corr) rather than estimation (QCardEst). Correction leverages baseline knowledge and only adjusts when needed. This advantage is stronger in JOB-light than STATS.
2. **Threshold for selective corrections:** The threshold mechanism excels when corrections should only be applied when the model has high confidence. This works consistently across benchmarks.
3. **Negativity matters for correction:** PlaceValueNeg variants (with negative factors) are important for correction because they enable symmetric, bidirectional adjustments to baseline estimates.
4. **Simplicity for estimation:** For estimation, simpler layers (linear, PlaceValue8) work better because they don't introduce unnecessary complexity that hurts when trying to estimate absolute values.
5. **Benchmark considerations:** STATS appears more challenging overall, with higher absolute errors. However, the relative performance patterns remain consistent, suggesting that layer choice should be based on the task (correction vs estimation) rather than the specific benchmark.

6. **Baseline quality matters:** The effectiveness of correction depends on baseline quality. While both benchmarks have similar PostgreSQL baseline errors, the correction advantage is larger in JOB-light, suggesting different error distributions or query characteristics.

6 Conclusions

The results demonstrate several key findings:

1. **Correction consistently outperforms estimation** across both benchmarks, though the advantage is larger in JOB-light.
2. **Threshold and PlaceValueNeg8 are robust choices** for correction, performing well across different query workloads.
3. **Linear is the clear winner for estimation**, providing the best balance of simplicity and performance.
4. **Layer characteristics are benchmark-agnostic:** Despite absolute differences, relative performance rankings are preserved, indicating that layer properties are robust across different query patterns.
5. **PlaceValueNeg variants show task-specific behavior:** They excel at correction but struggle with estimation, highlighting the importance of matching layer architecture to the task at hand.