VERSION2LABS

TECHNICAL REPORT 001

Same Data, Different Representation

An Empirical Study of Target Transformation in Rate-of-Penetration Prediction

Author	Heberto G Salinas

Protocol	v1.0
Primary well	Utah FORGE Well 56-32
Replications	Utah FORGE Well 78B-32; Utah FORGE Well 16B(78)-32
Date	2026-08-29

Executive summary

The question

This study asks a narrow engineering question:

> *Do models predict rate of penetration, or ROP, better when trained directly on raw ROP, or when trained on* `log1p(ROP)` *and converted back to ft/hr with* `expm1`*?*

Scope of the study. This report is not intended to identify the best prediction model, the best hyperparameter settings, or the best set of input variables for predicting ROP. Its purpose is narrower: holding the data, predictors, preprocessing, model settings, and validation procedure fixed, the study tests whether training on `log1p(ROP)` and converting predictions back to ft/hr produces lower original-unit prediction error than training directly on raw ROP.

The experiment changes only the target representation. Raw and transformed models use the same wells, rows, predictors, preprocessing structure, model settings, seeds, training blocks, and test blocks. Every comparison is evaluated in the original unit of ft/hr.

The study uses **Utah FORGE Well 56-32** as the primary well, with **Utah FORGE Well 78B-32** and **Utah FORGE Well 16B(78)-32** as replications. After this introduction, the report refers to them as 56-32, 78B-32, and 16B.

The answer

Under the frozen ten-block, five-fold protocol, `log1p` target training usually lowered prediction error, especially mean absolute error and especially for linear and ridge regression. It did not improve every model, metric, fold, or well.

Across 15 well-model comparisons:

- `log1p` lowered fold-balanced macro MAE in **14 of 15** cases;
- `log1p` lowered fold-balanced macro RMSE in **11 of 15** cases;
- both co-primary metrics favored `log1p` in **11** cases;
- both favored raw training in **1** case;
- MAE and RMSE disagreed in **3** cases.

Across 75 individual well-model-fold pairs, `log1p` lowered MAE in 62 and RMSE in 53.

The necessary qualification

All 30 fold-balanced macro R^2 values were negative, ranging from -4.061 to -0.172. Therefore, this study does not show that either target treatment generalized successfully across unseen depth blocks. It shows that `log1p` was often the less-poor of two frozen alternatives.

This distinction is central:

> *Relative improvement is not the same as absolute predictive success.*

Primary-well result

For 56-32, `log1p` lowered macro MAE for all five model families and macro RMSE for four. Linear, ridge, and XGBoost favored `log1p` on both co-primary metrics. Random forest showed a small and fold-sensitive advantage. MLP produced a direct disagreement: `log1p` improved MAE by 3.69 ft/hr but worsened RMSE by 15.02 ft/hr.

The lowest observed primary-well macro errors were produced by `log1p` XGBoost:

- MAE: 17.92 ft/hr;
- RMSE: 29.43 ft/hr;
- macro R^2: -0.214.

The negative R^2 prevents interpreting this pair as successful generalization.

Practical conclusion

The evidence supports a conditional result:

> *Under fixed preprocessing, fixed models, and equal-depth expanding-window validation,* `log1p` *target training usually reduces original-unit MAE and often reduces RMSE. The effect is strongest and most consistent for linear and ridge regression. It is model-dependent for tree ensembles and unreliable for the fixed MLP, particularly under RMSE and at high observed ROP.*

No composite winner is reported. MAE and RMSE answer different questions and remain separate.

1. Engineering question and mathematical idea

1.1 Why target representation matters

ROP is nonnegative and strongly right-skewed. Many observations occur at low or moderate rates, while a smaller number can be much larger. A learning algorithm trained on raw ROP sees the numerical distance between 20 and 200 ft/hr as 180 ft/hr. After a logarithmic transformation, the same difference is compressed.

For observed ROP $y \geq 0$, the transformed target is

$$z = \log(1 + y)$$

A model is trained to predict z. Its transformed prediction $\hat{z}$ is returned to original units by

$$\hat{y} = \exp(\hat{z}) - 1$$

In implementation, these are `log1p` and `expm1`. They are numerically stable near zero.

The transformation is one-to-one for nonnegative ROP. It does not add observations, predictors, or physical information. It changes the geometry of the loss seen during training.

1.2 A basic numerical example

Consider three observed ROP values:

ROP, ft/hr	log1p(ROP)
1	0.693
10	2.398
100	4.615

On the raw scale, moving from 10 to 100 adds 90 ft/hr. On the transformed scale, it adds about 2.217. Large values are compressed more than small values.

This creates a plausible tradeoff:

- transformed training may better represent the dense low-to-moderate ROP region;
- it may reduce the influence of extreme target values during fitting;
- after inversion, modest errors in log space can become large errors in ft/hr at high ROP.

That last point is important. If

$$\hat{y} = \exp(\hat{z}) - 1$$

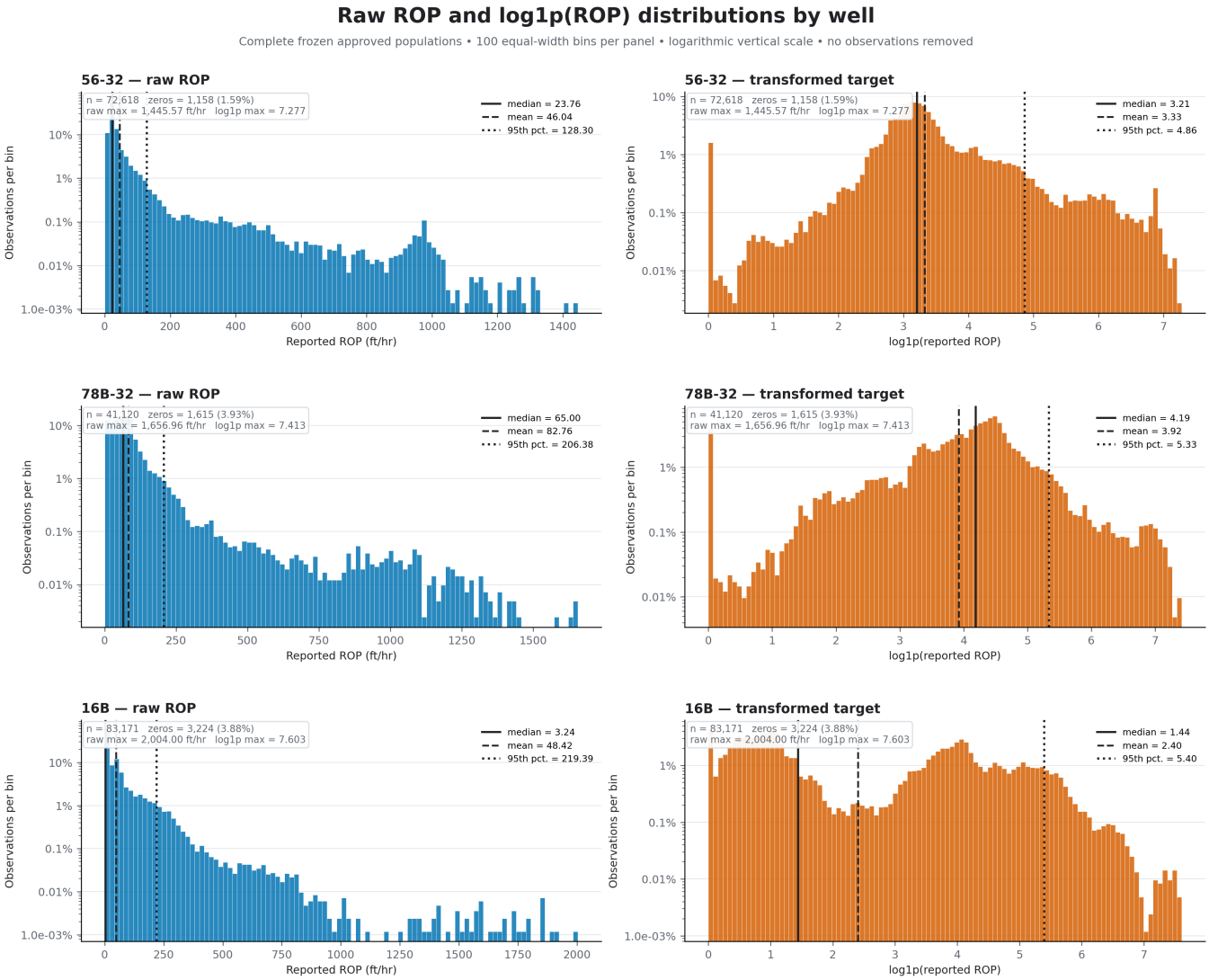
then the sensitivity of original-unit prediction to a transformed prediction is

$$\frac{d\hat{y}}{d\hat{z}} = \exp(\hat{z}) = 1 + \hat{y}$$

The same small error in $\hat{z}$ produces a larger ft/hr change when predicted ROP is high. This simple derivative explains why `log1p` can improve MAE while worsening RMSE for a model with occasional large misses.

1.2.1 Observed distributions before and after transformation

The following figure compares the complete frozen approved ROP population for each well before and after applying `log1p`. It uses all 196,909 approved observations, retains every finite upper-tail value, and uses a logarithmic vertical axis so low-frequency bins remain visible.



Raw and log1p ROP distributions for all three wells

The raw distributions are strongly right-skewed. The transformation compresses the long upper tails and makes the low-to-moderate ROP structure easier to see, especially in 16B. It preserves zero and the ordering of observations, but it does not make the three wells distributionally identical.

1.3 What is being estimated

The study estimates a paired target-representation effect under fixed conditions. It does not estimate the best separately tuned raw pipeline or best separately tuned transformed pipeline.

For each well, fold, and model, the comparison is

```
Same rows, predictors, folds, and model structure
+ { raw ROP target or log-transformed ROP target }
```

This paired structure isolates target representation better than comparing independently tuned systems.

2. Study population and predictors

2.1 Well hierarchy

The wells were analyzed independently:

1. **56-32:** primary well;
2. **78B-32:** replication 1;
3. **16B:** replication 2.

No metric pools observations across wells. Agreement across wells strengthens a pattern; disagreement remains part of the result.

2.2 Approved populations

Well	Approved rows	Out-of-sample rows, B6–B10	Out-of-sample coverage
56-32	72,618	56,324	77.56%
78B-32	41,120	23,545	57.26%
16B	83,171	67,592	81.27%
Total	**196,909**	**147,461**	**74.89%**

Each out-of-sample row is evaluated once per model and target treatment. B1–B5 form the initial training history and do not receive out-of-sample predictions.

2.3 Active drilling definition

An observation was eligible only when the independently approved active-on-bottom condition held:

$$|H_t - B_t| \le 1.0 \text{ ft}, \quad \text{WOB}_t > 0.5 \text{ klbf}, \quad \text{Flow}_t > 5.0\%$$

Here H_t is hole depth and B_t is bit depth. Missing formula inputs produce unknown state rather than a false state.

The source data remained read-only. Rows were not repaired, interpolated, winsorized, or removed after inspecting model errors.

2.4 Predictor set

The frozen predictor set contains six level variables and five causal 60-second changes:

Levels

1. hole depth;
2. signed bit-to-bottom gap;
3. hook load;
4. weight on bit;
5. rotary speed;
6. flow.

Causal changes

1. 60-second hook-load change;
2. 60-second WOB change;
3. 60-second RPM change;
4. 60-second flow change;
5. 60-second torque change.

For a ten-second source interval, a causal change is

$$\Delta_{60} x_t = x_t - x_{t-6}$$

It is available only when the six intervening intervals are exactly ten seconds and both endpoints are finite. This prevents differences from crossing missing observations or telemetry gaps.

Absolute torque was excluded from 56-32 because its native unit was not certified. Torque entered only through its standardized causal change. Vendor on-bottom status, on-bottom ROP, time of penetration, and depth changes that approximate direct ROP were prohibited.

3. Ordered validation mathematics

3.1 Progress depth

Temporary upward bit movement should not move a later observation back into an earlier validation block. The split coordinate is therefore cumulative progress depth:

$$D_t^{\text{progress}} = \max_{s \le t} B_s$$

Because a cumulative maximum never decreases,

$$D_{t+1}^{\text{progress}} \ge D_t^{\text{progress}}$$

This monotonicity gives ordered block membership. Progress depth is used only to construct validation blocks. It is not a model predictor.

3.2 Equal-depth blocks

For each well, the progress span is divided into ten intervals of equal width:

$$w = \frac{D_{\max} - D_{\min}}{10}$$

The blocks are equal in depth, not in row count. A slow-drilling block can contain many more ten-second observations than a fast-drilling block. That imbalance is real process information and is not artificially equalized.

Well	Progress-depth range, ft	Block width, ft
56-32	182.00–9,145.00	896.30

Well	Progress-depth range, ft	Block width, ft
78B-32	391.40–9,500.00	910.86
16B	118.05–10,946.58	1,082.853

3.3 Five expanding folds

Fold	Training blocks	Test block
1	B1–B5	B6
2	B1–B6	B7
3	B1–B7	B8
4	B1–B8	B9
5	B1–B9	B10

Every test block is later in progress depth than its training history. A block tested in one fold may enter training in a later fold, which represents learning from accumulated history.

A 120-second training-side purge is applied immediately before each test block. Test observations are never purged. The purge is longer than the 60-second causal feature horizon.

3.4 Why the primary mean is fold-balanced

Let M_k be a metric computed on test block k, where $k=1,\ldots,5$. The primary macro summary is

$$\overline{M}_{\text{macro}} = \frac{1}{5}\sum_{k=1}^{5} M_k$$

Each equal-depth block contributes one-fifth, regardless of its number of ten-second rows.

A row-pooled metric answers a different question. If block k has n_k rows, row pooling effectively weights it by

$$w_k = \frac{n_k}{\sum_{j=1}^{5} n_j}$$

For 16B, the deep slow-drilling blocks contain many rows. A pooled result can therefore look better even when performance is poor across equal-depth regimes. The fold-balanced macro result remains primary.

4. Error mathematics

4.1 Residual direction

A residual is observed ROP minus predicted ROP. A positive residual means underprediction, while a negative residual means overprediction. Mean signed residual is retained only as a supporting diagnostic of average direction; positive and negative errors can cancel.

4.2 Mean absolute error

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|e_i|$$

MAE is the average absolute miss in ft/hr. Each additional ft/hr of error contributes linearly.

4.3 Root mean squared error

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}e_i^2}$$

RMSE gives greater weight to large misses because errors are squared before averaging. MAE and RMSE remain separate because they describe different error behavior.

4.4 Coefficient of determination

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$$

An R^2 of 1 is perfect prediction, 0 matches the constant test-block mean benchmark in squared error, and a negative value is worse than that benchmark. A treatment can have lower MAE than its pair while both treatments still have negative R^2.

4.5 Paired difference

For MAE and RMSE, the paired difference is defined as the `log1p` metric minus the raw-target metric. A negative difference favors `log1p`; a positive difference favors raw training; zero is a tie at the reported precision.

No MAE-RMSE composite is constructed. A composite would hide disagreements by imposing an unapproved weighting between typical and severe errors.

5. Computational experiment

The frozen scope was

```
3 wells × 5 folds × 5 models × 2 treatments = 150 fits
```

The model families were:

- ordinary linear regression;
- ridge regression with $\alpha=1$;
- random forest with 300 trees and minimum leaf size 5;
- XGBoost with 500 estimators, depth 6, and learning rate 0.05;
- an MLP with hidden layers 64 and 32 and a fixed 300-iteration budget.

No hyperparameter tuning, outcome-based model selection, or full-data refit occurred.

Feature means and population standard deviations were fitted on training rows only. The treated target was also standardized on training rows only. For `log1p`, target standardization was reversed before `expm1`.

Both treatments used the same physical nonnegative rule:

$$\hat{y}_{\text{final}} = \max(0, \hat{y})$$

Clipping was recorded rather than hidden.

6. Primary findings: 56-32

6.1 Fold-balanced macro results

Model	Raw MAE	`log1p` MAE	Δ MAE	Raw RMSE	`log1p` RMSE	Δ RMSE
Linear	30.82	17.95	**−12.87**	39.58	29.55	**−10.03**
Ridge	30.82	17.95	**−12.87**	39.58	29.54	**−10.03**
Random forest	30.16	28.13	**−2.03**	39.65	38.71	**−0.94**
XGBoost	27.16	17.92	**−9.24**	38.15	29.43	**−8.71**
MLP	29.35	25.66	**−3.69**	44.08	59.10	**+15.02**

6.2 Interpretation by model

Linear and ridge. `log1p` lowered both MAE and RMSE in all five folds for each model. Their near-identical results indicate that the fixed ridge penalty had little practical effect in this standardized setting. The target representation had a much larger effect than the difference between these two estimators.

Random forest. The macro differences are small. `log1p` lowered MAE in four of five folds but RMSE in only two. The slightly lower macro RMSE is caused by the sizes of the fold differences, not by a majority of fold wins.

XGBoost. `log1p` lowered MAE in four folds and RMSE in three. Its macro advantage is meaningful in size, but not universal across depth regimes.

MLP. `log1p` lowered both metrics in four folds, yet one large failure reversed macro RMSE. MAE improved by 3.69 ft/hr while RMSE worsened by 15.02 ft/hr. This is exactly why the study did not combine the two metrics.

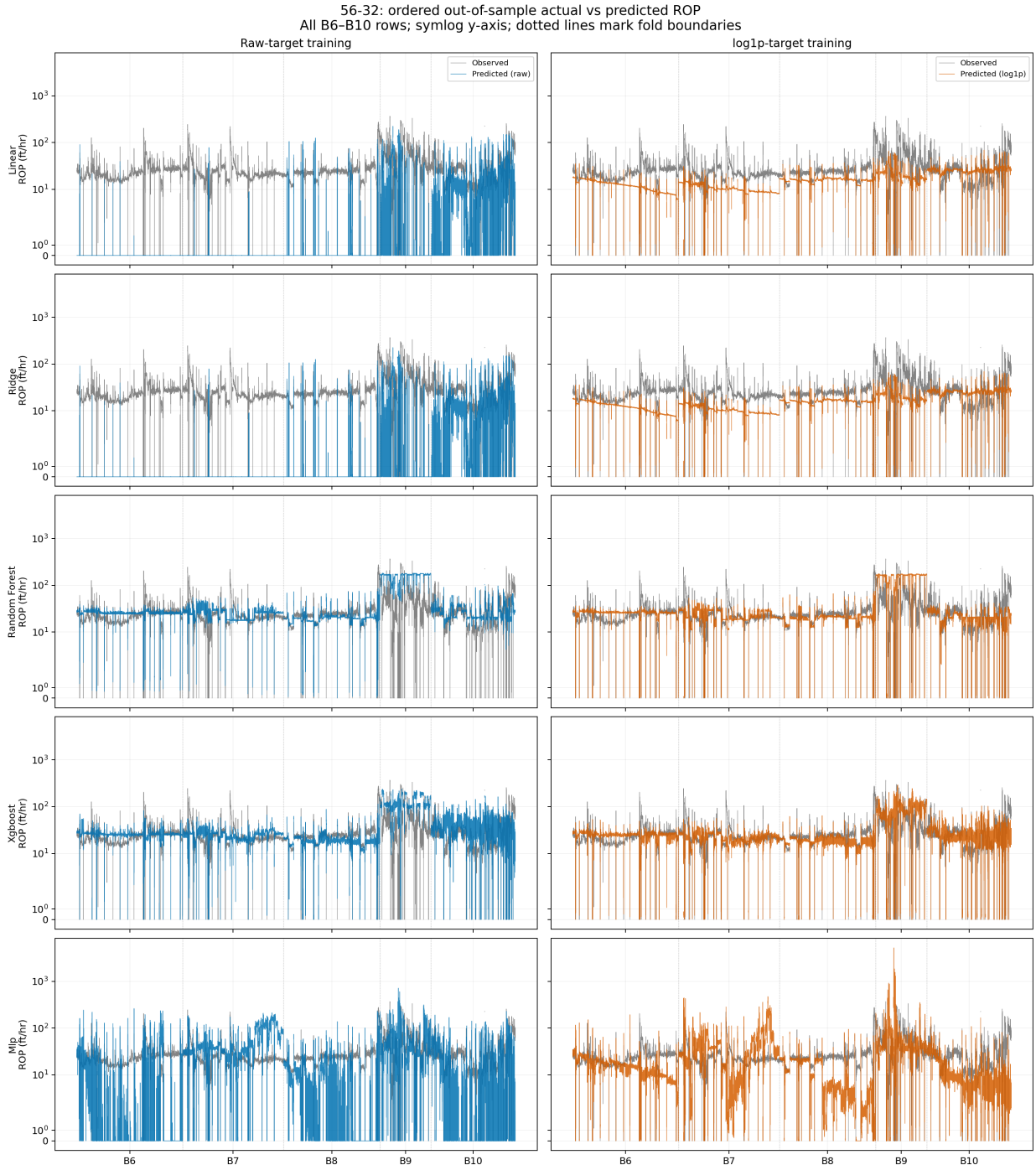
6.3 Least-poor observed pair

The lowest macro MAE and RMSE in the primary well were produced by `log1p` XGBoost:

Metric	Value
Macro MAE	17.92 ft/hr
Macro RMSE	29.43 ft/hr
Macro R^2	−0.214

The negative R^2 means this pair should not be called successful, best-in-class, or deployment-ready. It was the least-poor observed pair under the frozen comparison.

6.4 Chronological evidence



Primary-well actual versus predicted ROP

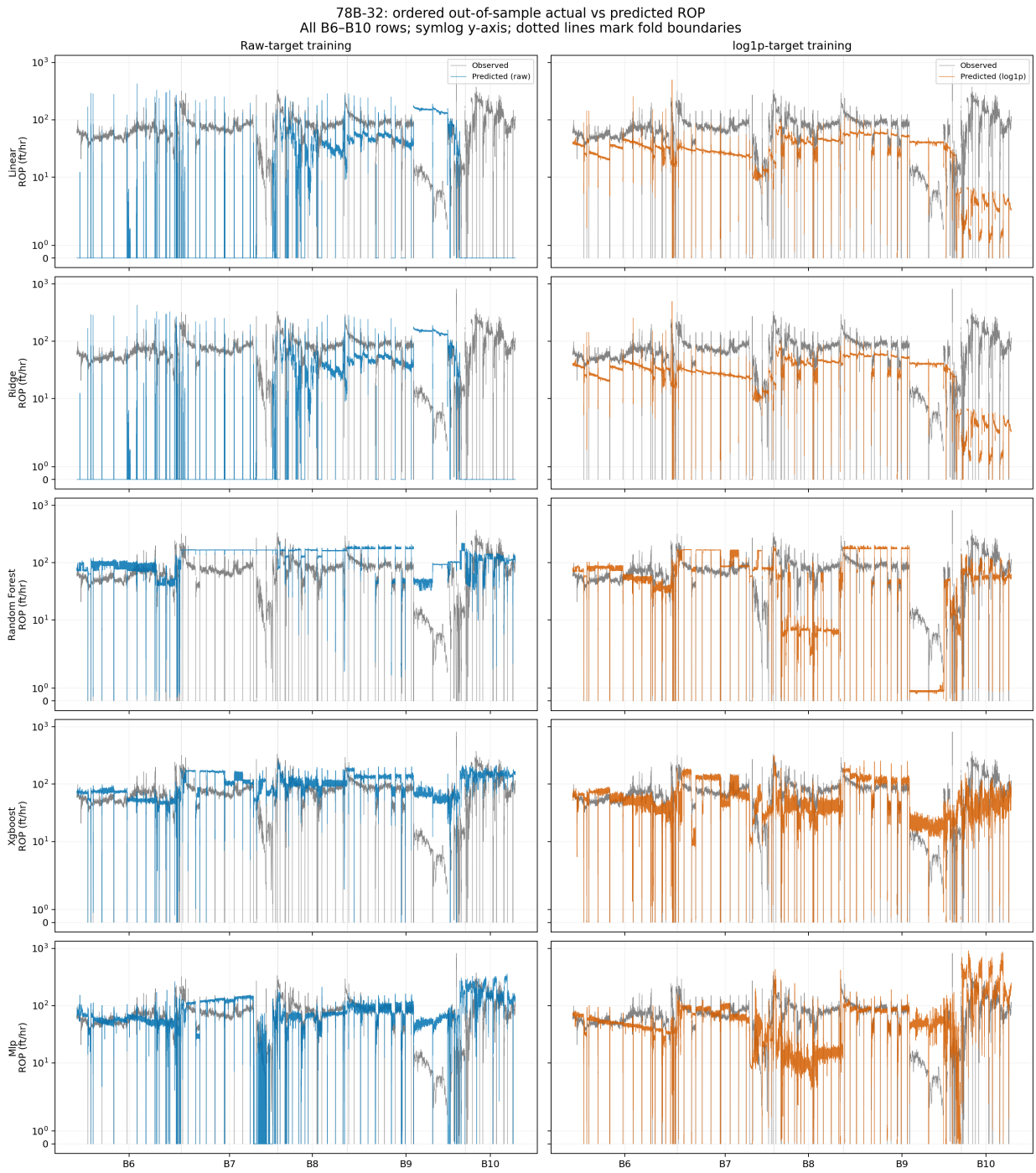
The figure contains only genuine out-of-sample predictions from B6–B10. Fold boundaries remain visible, and lines break across telemetry gaps. The symlog vertical axis preserves both low-rate and high-rate behavior without deleting extremes.

7. Replication 1: 78B-32

Model	Raw MAE	log1p MAE	Δ MAE	Raw RMSE	log1p RMSE	Δ RMSE
Linear	76.64	54.97	**−21.67**	89.45	67.23	**−22.23**
Ridge	76.64	54.96	**−21.68**	89.45	67.22	**−22.24**
Random forest	63.69	53.71	**−9.98**	72.37	66.38	**−5.99**
XGBoost	44.06	42.07	**−1.99**	52.28	54.71	**+2.43**
MLP	42.45	60.60	**+18.16**	58.17	81.16	**+22.99**

Linear, ridge, and random forest reproduce a same-direction log1p advantage on both metrics. XGBoost produces a metric disagreement: typical absolute error improves slightly, while tail-sensitive RMSE worsens. MLP is the only well-model case where raw training wins both co-primary metrics.

The lowest MAE was log1p XGBoost at 42.07 ft/hr. The lowest RMSE was raw XGBoost at 52.28 ft/hr. There is no single least-poor pair unless the operational metric is specified.



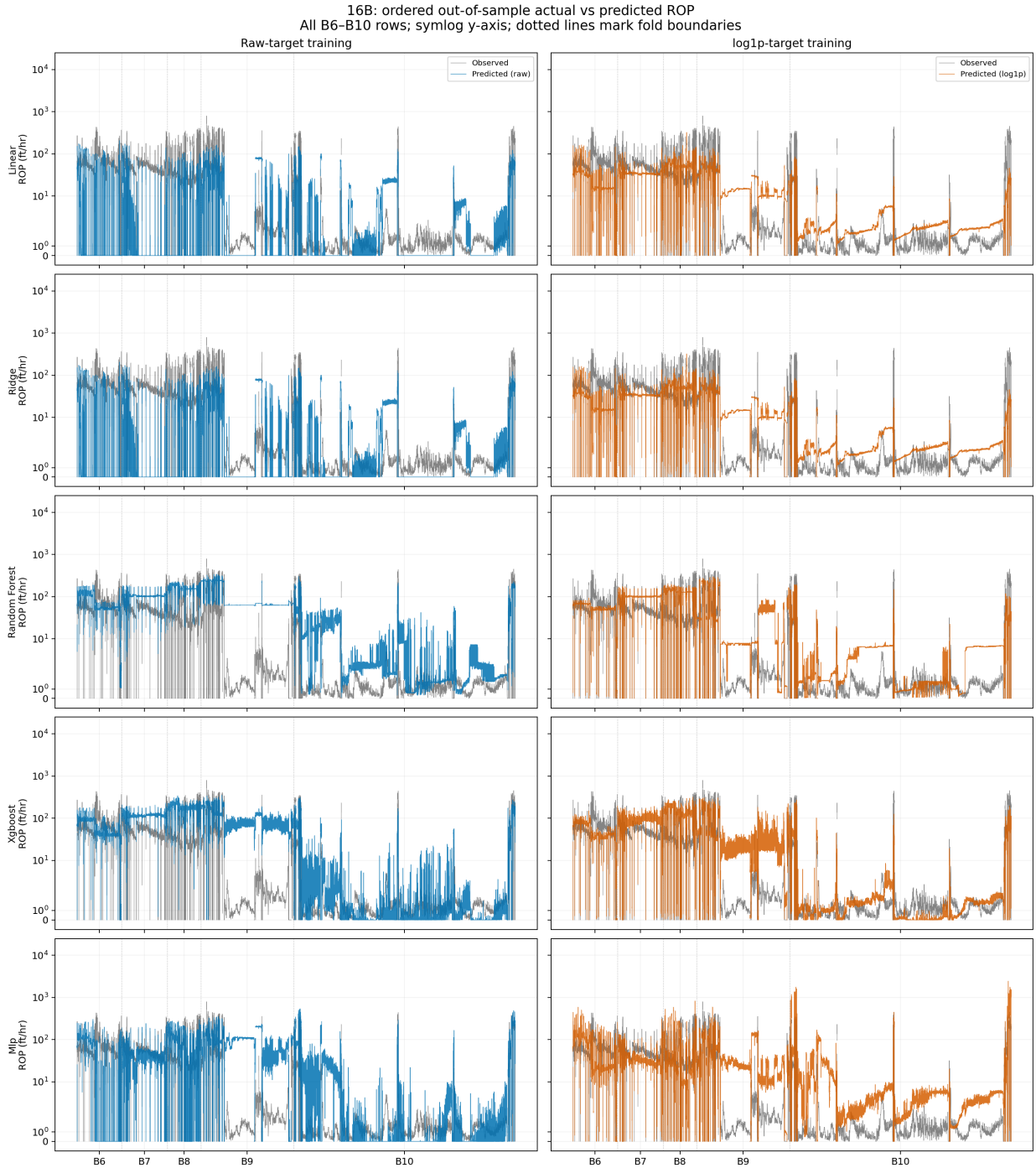
Replication 1 actual versus predicted ROP

8. Replication 2: 16B

Model	Raw MAE	log1p MAE	Δ MAE	Raw RMSE	log1p RMSE	Δ RMSE
Linear	45.79	31.50	**−14.29**	67.95	55.58	**−12.36**
Ridge	45.79	31.50	**−14.29**	67.94	55.58	**−12.36**
Random forest	58.36	38.58	**−19.78**	70.23	54.66	**−15.57**
XGBoost	61.17	40.99	**−20.19**	73.02	56.68	**−16.35**
MLP	46.62	37.53	**−9.08**	62.71	84.77	**+22.06**

All five model families favor `log1p` on MAE. Four favor it on RMSE. MLP again shows that improving typical error can coexist with worsening severe-error behavior.

The lowest macro MAE was `log1p` ridge at 31.50 ft/hr. The lowest macro RMSE was `log1p` random forest at 54.66 ft/hr. The identity of the least-poor model still depends on the metric.



Replication 2 actual versus predicted ROP

9. Cross-well mathematical patterns

9.1 Direction counts

Evidence unit	`log1p` lower MAE	`log1p` lower RMSE	Total comparisons
Well-model macro cases	14	11	15
Individual paired folds	62	53	75

The counts are descriptive, not independent Bernoulli trials. Folds share wells, model families, and expanding training histories. An independence-based p-value would overstate the amount of independent evidence.

9.2 Linear and ridge consistency

For linear and ridge regression:

- `log1p` lowered MAE in all 30 paired folds across both model families;
- it lowered RMSE in 28 of 30 paired folds;
- macro MAE and RMSE favored `log1p` in every well.

This is the most stable pattern in the study.

9.3 Tree-model qualification

Random forest macro MAE and RMSE favor `log1p` in all wells, but fold directions are less stable. XGBoost macro MAE favors `log1p` in all wells, while RMSE reverses in 78B-32.

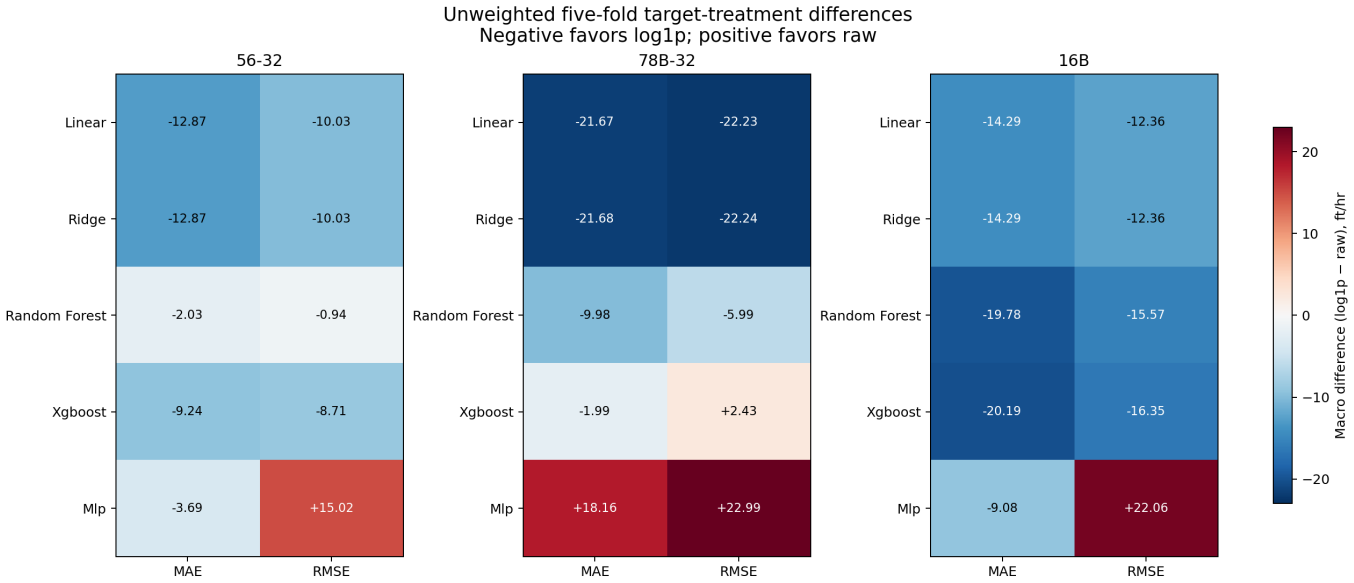
The transformation therefore has a broader MAE benefit than RMSE benefit for the tree ensembles.

9.4 MLP counterexample

MLP results falsify a universal transformation claim:

- 56-32: MAE favors `log1p`; RMSE favors raw;
- 78B-32: both metrics favor raw;
- 16B: MAE favors `log1p`; RMSE favors raw.

One 78B-32 fold-3 `log1p` MLP fit reached the frozen 300-iteration limit. The finite fit was retained. The broader MLP reversals are not confined to that warning.



Macro paired differences

The heatmap uses the frozen sign convention `log1p − raw`. Blue cells favor `log1p`; orange cells favor raw. The MLP RMSE row visibly contains the largest reversals.

10. Why MAE and RMSE disagree

Suppose one treatment has errors *10,10,10,10,50* and another has *18,18,18,18,18*, all in ft/hr.

For the first treatment:

$$\mathrm{MAE} = 18, \quad \mathrm{RMSE} \approx 24.1$$

For the second:

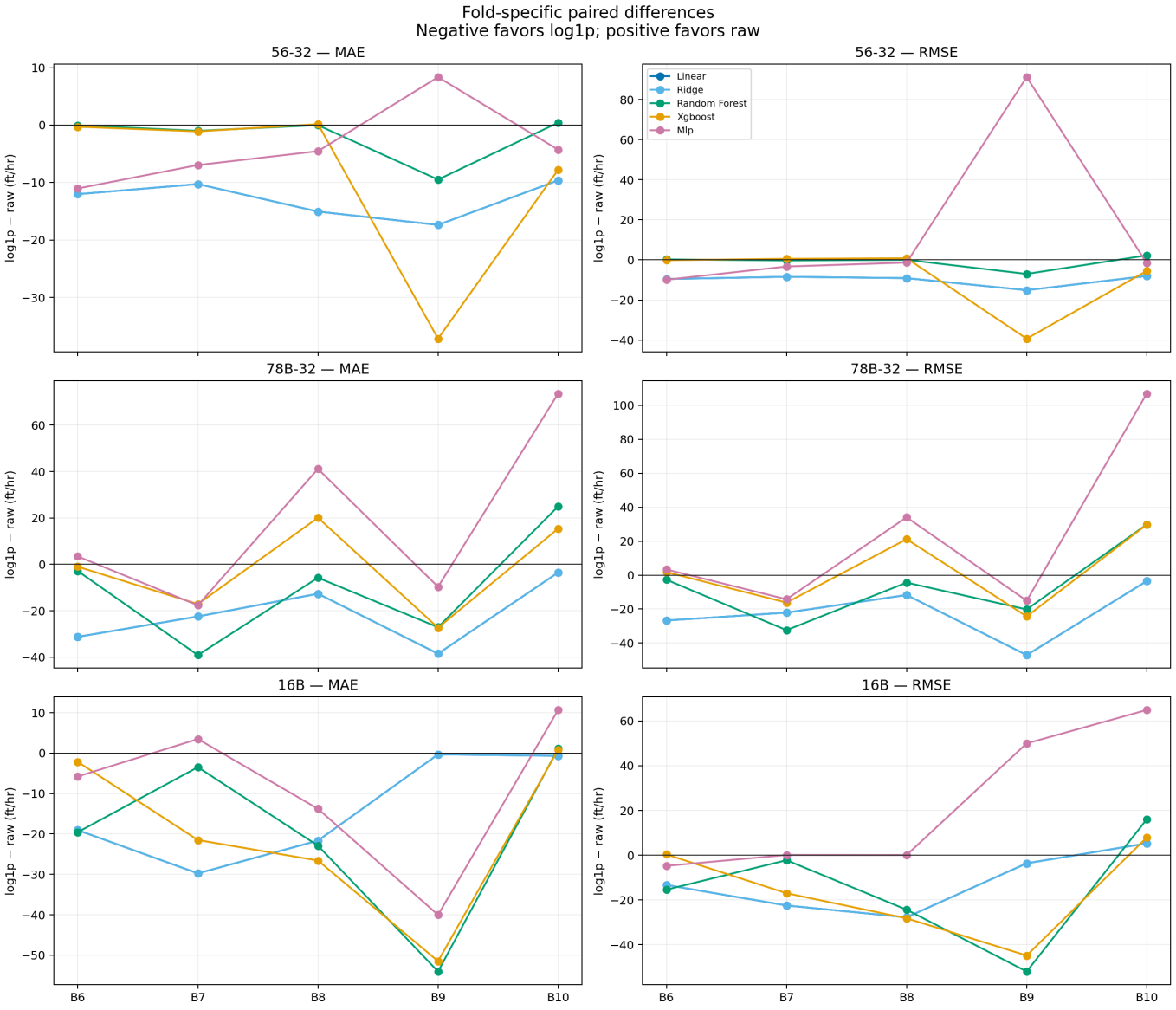
$$\mathrm{MAE} = 18, \quad \mathrm{RMSE} = 18$$

The MAEs tie, but RMSE rejects the one large miss. This simplified example mirrors the observed MLP behavior: many improved predictions can coexist with a small number of severe transformed-path failures.

Observed-ROP quintiles show this directly:

- in the highest ROP quintile of 78B-32, `log1p` worsened both MAE and RMSE for random forest, XGBoost, and MLP;
- in the highest quintile of 16B, `log1p` reduced MAE for random forest and XGBoost but increased RMSE;
- for 16B MLP's highest quintile, `log1p` increased MAE by 30.66 ft/hr and RMSE by 81.04 ft/hr;
- linear and ridge retained high-quintile improvements in all three wells.

The transformation often improves the center of the error distribution while retaining a risk of severe original-unit errors for some nonlinear models.



Fold-level paired differences

11. Absolute performance

11.1 Fold-balanced R²

All 30 well-model-treatment macro R² values were negative:

$$-4.061 \leq \text{mean fold-balanced } R^2 \leq -0.172$$

At the individual-fit level:

Fold-level R² sign	Count
Positive	34
Zero	0
Negative	116
Total	**150**

This is the most important unfavorable result. It means no frozen configuration consistently beat a test-block mean benchmark when equal-depth blocks were given equal weight.

11.2 Row-pooled R^2

Seven of 30 row-pooled R^2 values were positive, all in 16B. None were positive in 56-32 or 78B-32.

The pooled and macro statements are not contradictory. They apply different weights. Row pooling lets blocks with more ten-second observations dominate. Macro averaging gives each equal-depth future regime one-fifth.

For this study's engineering question, fold-balanced generalization is primary.

12. Clipping as a model-behavior diagnostic

Both target treatments received the same nonnegative rule. Clipping counts were:

Model	Raw clipped rate	`log1p` clipped rate
Linear	70.112%	0.647%
Ridge	70.106%	0.647%
Random forest	0.000%	0.000%
XGBoost	10.208%	3.446%
MLP	31.981%	1.377%

Raw linear and ridge extrapolated below zero for most repeated test evaluations. `log1p` greatly reduced this behavior.

Clipping can affect metrics, but it does not fully explain the transformation result. Random forest required no clipping under either treatment and still had lower `log1p` macro MAE in every well.

No clipped row was removed. The same observed row remained in every paired evaluation.

13. What the study establishes

The experiment establishes the following empirical results under its frozen protocol:

1. `log1p` usually lowers original-unit MAE.
2. Its RMSE advantage is less consistent because severe high-ROP misses can reverse the result.
3. Linear and ridge show the strongest cross-well consistency.
4. Tree-model conclusions are conditional on well and metric.
5. The fixed MLP supplies clear counterexamples to universal superiority.
6. Both treatments frequently generalize poorly across unseen equal-depth blocks.
7. Equal-depth macro aggregation and row pooling can produce different absolute-performance impressions.

These are empirical statements, not mathematical theorems about every ROP dataset or every model configuration.

14. What the study does not establish

The experiment does not establish:

- that `log1p` is universally better than raw-target training;
- that lower MAE guarantees lower RMSE;
- that either target treatment is deployment-ready;
- that the same result holds after independent hyperparameter tuning;
- that the same result holds for other wells, basins, rigs, sensors, or operating definitions;
- that clipping caused the observed differences;
- that high ROP values are sensor errors;
- that negative R^2 can be repaired by deleting difficult rows;
- that 75 fold comparisons are 75 independent statistical experiments.

No later IQR, percentile, z-score, winsorization, target trimming, residual trimming, or outcome-based boundary change was introduced.

15. Falsification criteria and next research questions

A useful claim should state what could weaken it.

15.1 Claim tested here

> *Under this paired protocol,* `log1p` *usually improves MAE and often improves RMSE.*

This claim would be weakened by:

- a separately frozen set of wells where raw training lowers MAE across most models and folds;
- a sensitivity analysis showing that the advantage disappears when prediction-domain handling is changed symmetrically;
- model-specific tuning that reverses most paired results, while being reported as a different estimand;
- evidence that source-target corruption, rather than target geometry, explains the transformed-path advantage.

15.2 Claim already falsified

The stronger claim

> `log1p` *is always better than raw-target training*

is already falsified by:

- raw MLP winning both metrics in 78B-32;
- raw XGBoost winning RMSE in 78B-32;
- raw MLP winning RMSE in 56-32 and 16B;

- multiple fold-level reversals.

15.3 Recommended next protocol

A future study should be frozen separately. Reasonable questions include:

1. Does matched hyperparameter selection preserve the linear/ridge result?
2. Do section-aware or additional well-level replications preserve the MAE pattern?
3. Can a loss function designed directly for original-unit tail risk reduce the MAE-RMSE disagreement?
4. Can uncertainty intervals identify blocks where either treatment should not be trusted?
5. Does modeling ROP change over a causal horizon provide more stable generalization than predicting the level directly?

These are future protocols, not repairs to the completed experiment.

16. Execution and verification

16.1 Execution state

- 150 of 150 approved fits completed;
- 150 prediction artifacts persisted;
- all predictions and metrics finite;
- residual arithmetic independently recomputed;
- prediction identities matched within each paired well-fold comparison;
- 196,909 rows by 11 features reconstructed independently from retained raw sources;
- one finite MLP convergence warning retained;
- no result-driven refit performed.

16.2 Analysis verification

Independent Step 7 verification returned:

```
PASS
source metrics SHA-256: 0817404cda7d0ceadb8f606056966b610a123b82f434da46348d549ae8c2a39e
fold rows: 150
macro rows: 30
paired fold rows: 75
paired macro rows: 15
pooled rows: 30
residual rows: 150
PNG figures: 5
SVG figures: 5
artifact hashes: 16
```

The complete software test suite returned 11 passing tests before report construction.

16.3 Verification scope

Verification covered the frozen population, processed predictors, ordered fold identities, original-unit predictions, residual arithmetic, metric calculations, summary tables, and scientific figures. The report was built from the verified tables and figures without changing the approved population, protocol, model outputs, or findings.

17. Final conclusion

The target transformation question has a conditional answer.

`log1p(ROP)` training usually reduced original-unit MAE and often reduced RMSE under the frozen experiment. The evidence is strongest for linear and ridge regression, qualified for random forest and XGBoost, and unreliable for the fixed MLP. The transformation appears to improve typical error more consistently than it controls severe high-ROP misses.

At the same time, every fold-balanced macro R^2 was negative. The study therefore compares two frequently poor target treatments rather than demonstrating a solved prediction problem.

The mathematically honest conclusion is:

> *Use* `log1p` *as a justified candidate representation, not as a universal rule. Preserve MAE and RMSE separately, inspect high-ROP residuals, and require new ordered validation before making operational claims.*