REVISITING LOG-TRANSFORMATION IN DEEP REGRESSION: RECTIFYING METRIC-WARPED INTERPOLATION BIAS VIA TRANSIENT VARIANCE GUIDANCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Log-transformation is a standard paradigm for deep regression on heavily skewed data. We find that, for data whose underlying structure is additive-dominant in the original space, the network's smoothness preference in log space drives its predictions away from the data's intrinsic regularity, inducing a systematic interpolation bias during generalization. We define this issue as metric-warped interpolation bias, which arises from the joint effect of the smooth inductive bias of neural networks and the strict convexity of the exponential mapping. Unlike the widely studied pointwise inverse-transform error, this issue has remained unexplored in previous work. Although directly introducing linear intervention to mitigate this bias appears intuitive, it easily destabilizes optimization. To address this dilemma, we propose Transient Variance Guidance (TVG), which repurposes the transient variance–error coupling of Negative Log-Likelihood (NLL), typically regarded as undesirable, as an endogenous proxy for fitting lag. TVG uses this transient variance-based fitting-lag signal to safely inject linear intervention into the learned representation, incurs zero computational overhead during inference, and consistently outperforms mainstream log-regression baselines across five cross-domain skewed datasets.

1 INTRODUCTION

In many real-world regression applications, target variables often exhibit skewed, long-tailed, or zero-inflated distributions (Belitz & Stackelberg, 2021; Chen et al., 2024; Weng et al., 2024), including settings such as weather forecasting and traffic flow prediction. In such scenarios, directly optimizing neural networks in the target space (also named as original space) can cause gradient explosions and optimization instability due to extreme values. To stabilize training, applying concave transformations such as the log-transformation has become standard practice. However, target transformation changes not only the numerical scale of optimization, but also the metric under which neural networks express their smoothness bias. While the bias induced by logarithmic retransformation has been studied in classical statistics (Finney, 1941; Duan, 1983) and recently revisited in deep learning (Yu et al., 2025; Zhao et al., 2026), these works focus on correcting pointwise expectation shifts. They overlook how metric transformations structurally distort continuous interpolation between samples when the underlying structure is additive-dominant in the original space.

Metric-Warped Interpolation Bias. Unlike expectation correction methods that treat predictions as isolated points and rely on post-hoc calibration, this work examines an inductive mismatch in log-transformed regression. For data whose underlying structure is additive-dominant in the original space, the network's smoothness preference in the log space can bias predictions away from the additive regularity of the original space. Due to the strict convexity of the exponential mapping, smooth affine transitions learned in log space conflict with the original space's additive structure, inducing a metric-warped interpolation bias during generalization. Thus, the same transformation that stabilizes optimization can impose a mismatched interpolation geometry.

Transient Variance Guidance (TVG). Mitigating this bias by directly introducing linear intervention is non-trivial: naive linear intervention easily destabilizes optimization through dual-space gradient conflicts. To address this dilemma, we propose the Transient Variance Guidance (TVG) framework.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Our key observation is that the transient variance–error coupling in NLL optimization, typically regarded as undesirable, also exposes an endogenous signal of fitting lag before convergence. Rather than directly fitting unstable residual signals, TVG uses this transient variance as a smooth proxy for fitting lag to guide linear intervention. A stop-gradient operator on the log-mean head $\mu(\mathbf{X})$ isolates it from gradients induced by linear intervention, while the variance and modulation branches (collectively referred to as the auxiliary branches) reshape the shared representation during early training. At inference, these auxiliary branches are removed, yielding predictions with zero additional computational overhead. Our primary contributions and empirical findings are summarized as follows:

- **Theoretical Formulation:** We formalize the metric-warped interpolation bias in log-transformed deep regression, showing the conflict between log space smoothness and original space additivity under the convexity of the exponential mapping, and derive a closed-form approximation of the resulting distortion.
- **Methodology:** We reinterpret transient variance inflation in NLL optimization as an endogenous proxy for fitting lag, and develop TVG to inject linear intervention without directly fitting unstable residuals, while preserving stable log space optimization and zero inference overhead.
- **Empirical Validation:** Across five cross-domain skewed benchmark datasets and multiple network backbones, TVG consistently outperforms mainstream log-regression baselines and retransformation corrections.

2 Metric Warping in Log-Transformed Regression

2.1 Geometric Origin of Metric-Warped Interpolation Bias

Unlike in classical statistics, where logarithmic transformations are primarily employed to model fundamentally multiplicative data-generating processes, their adoption in deep regression typically represents an engineering compromise to stabilize gradient dynamics under skewed distributions. In a system governed by observational constraints or localized sparsity, assuming that the underlying local data exhibits linear additivity, the observed target value Y can be decomposed as $Y = Y^* + R(\mathbf{X})$, where Y^* denotes the conditional expectation trajectory dictated by this linear additive structure, and $R(\mathbf{X})$ represents an implicit residual. In the absence of explicit prior knowledge regarding this residual, the canonical unbiased assumption is $\mathbb{E}[R(\mathbf{X}) \mid \mathbf{X}] = 0$, thereby establishing Y^* as the conditional-mean reference trajectory in the original space.

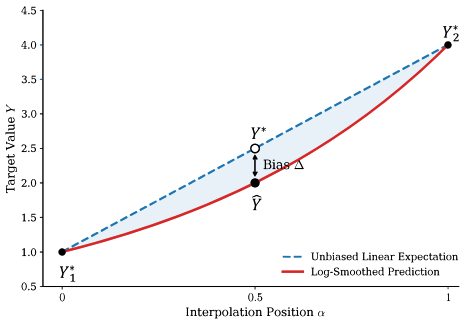

Figure 1: The origin of metric-warped interpolation bias in the original space.

Under the log-preprocessing paradigm, influenced by the smooth inductive bias of neural networks (Jacot et al., 2018; Rahaman et al., 2019), the model tends to form continuous transitions between adjacent samples In log space. To conceptually characterize this behavior, consider a local transition path that approximately satisfies linear interpolation:

$$Z_\alpha = (1-\alpha)Z_1 + \alpha Z_2, \tag{1}$$

where $Z_1 = \ln Y_1^*$ and $Z_2 = \ln Y_2^*$ denote the log space coordinates corresponding to two local reference anchors, and $\alpha \in [0, 1]$ parameterizes the interpolation path between them. While practical neural representations may exhibit higher-order variations, this affine formulation captures the low-complexity behavior favored by minimal-variation inductive biases (Savarese et al., 2019; Parhi & Nowak, 2021), serving as an idealized path model to isolate the structural distortion induced purely by metric warping (a formal analysis of non-affine curvature is provided in Appendix B.5).

Here, $Y^*(\mathbf{X})$ serves as the original space reference trajectory for isolating the pathwise geometric effect, while stochastic pointwise retransformation effects are treated separately in Appendix D.

Under the local log space affine path in Equation equation 1, exponential inversion yields the geometric interpolation $\hat{Y} = \exp(Z_\alpha) = (Y_1^*)^{1-\alpha}(Y_2^*)^{\alpha}$. In contrast, the unbiased baseline under linear additivity in the original space is the arithmetic mean $Y^* = (1-\alpha)Y_1^* + \alpha Y_2^*$. Due to the strict convexity of the exponential function, an algebraic distortion $\Delta = Y^* - \hat{Y}$ exists between the two (as illustrated in Figure 1). Originating from the non-linear mapping itself, this error is independent of the system's implicit residual $R(\mathbf{X})$ and cannot be neutralized by zero-mean noise, constituting a structural interpolation bias that empirical sample averaging cannot eliminate.

To quantitatively estimate the magnitude of this distortion, let $\delta = Z_2 - Z_1$ denote the local logarithmic span. Performing a second-order Taylor expansion around $\delta = 0$ on both $Y^* \approx Y_1^*\left(1 + \alpha\delta + \frac{1}{2}\alpha\delta^2\right)$ and $\hat{Y} \approx Y_1^*\left(1 + \alpha\delta + \frac{1}{2}\alpha^2\delta^2\right)$ yields the closed-form approximation:

$$\Delta \approx \frac{1}{2} Y_1^* \alpha(1-\alpha)\delta^2. \tag{2}$$

The complete derivation, together with a non-perturbative analysis confirming the sign structure of the distortion beyond the second-order approximation, is provided in Appendix B. Equation equation 2 reveals two structural properties: **(i) Path-dependent curvature:** the distortion is governed by $\alpha(1-\alpha)$, strictly positive inside the interpolation interval ($\alpha \in (0,1)$, peaking at $\alpha = 0.5$), vanishing at the anchors ($\alpha \in \{0,1\}$), and reversing sign under extrapolation ($\alpha \notin [0,1]$). **(ii) Quadratic sensitivity:** the distortion scales as δ^2, so regions with sharp transitions or sparse observations suffer the most pronounced interpolation distortion. This closed-form formulation also clarifies why metric-warped interpolation bias differs from classical log-retransformation corrections. Classical methods such as Duan's smearing (Duan, 1983) correct for aleatoric noise dispersion at isolated points ($\sigma^2 > 0$). In contrast, the bias in Equation equation 2 is a deterministic geometric distortion that persists even in noise-free systems ($\sigma^2 = 0$). Because this distortion varies continuously along interpolation paths, it cannot be eliminated by post-hoc scalar recalibration, motivating in-training constraints on the learned representation manifold. A detailed comparison with classical retransformation bias, including their distinct statistical objectives and behavior in the noise-free limit, is provided in Appendix D.

2.2 The Dilemma of Training-Time Linear Intervention

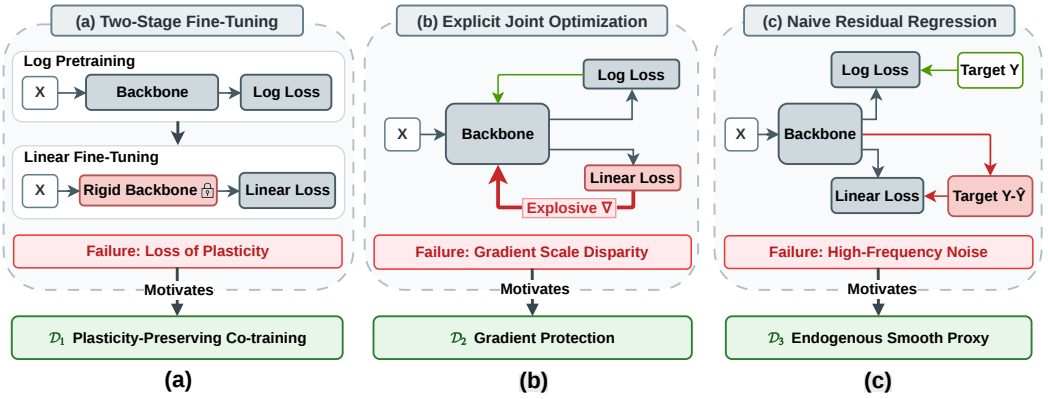

Figure 2: Failure modes of three intuitive linear-intervention paradigms and the corresponding design desiderata of TVG: (a) two-stage fine-tuning suffers from loss of plasticity, motivating $\mathcal{D}_1$ (plasticity-preserving co-training); (b) explicit joint optimization suffers from gradient-scale disparity, motivating $\mathcal{D}_2$ (asymmetric gradient protection); and (c) naive residual regression is prone to high-frequency noise memorization, motivating $\mathcal{D}_3$ (endogenous smooth proxy).

Although Section 2.1 provides the closed-form approximation for the metric-warped interpolation bias Δ in equation 2, the local geometric parameters of unseen test samples remain unobservable a priori at inference time, making static post-hoc calibration ineffective. Addressing this bias necessitates dynamic intervention during representation learning. The structural bias Δ in equation 2 is intrinsically induced by the smooth inductive bias under the logarithmic metric. If linear intervention

can be injected during learning, the prediction trend can be shifted away from the log space smoothness induced by the logarithmic metric and toward the original additive regularity, thereby suppressing the interpolation distortion at its root. However, directly introducing linear intervention during training easily destabilizes optimization. As illustrated in Figure 2, an in-depth analysis of the failure modes of three intuitive paradigms motivates three core design desiderata:

(a) Two-Stage Fine-Tuning: This paradigm pretrains the backbone f_θ via logarithmic supervision ($\mathcal{L}_{\text{log}}$) until convergence, subsequently switching to linear intervention ($\mathcal{L}_{\text{linear}}$). This sequential decoupling can induce representation hardening and plasticity loss (Lyle et al., 2023; Dohare et al., 2024). Once the representation locks into the smooth logarithmic basin, linear gradients struggle to provide sufficient momentum to reshape local features, resulting in stagnant updates. This limitation directly motivates $\mathcal{D}_1$ **(Co-training):** linear intervention should operate concurrently with logarithmic supervision during the early-to-middle transient phase to leverage the network's plasticity.

(b) Explicit Joint Optimization: Simultaneously supervising a shared backbone $f_\theta(\mathbf{X})$ with logarithmic ($\mathcal{L}_{\text{log}}$) and linear ($\mathcal{L}_{\text{linear}}$) losses causes gradient scale conflicts (Chen et al., 2018). Gradients induced by extreme values can heavily dominate the updates ($\|\nabla_\theta \mathcal{L}_{\text{linear}}\| \gg \|\nabla_\theta \mathcal{L}_{\text{log}}\|$), disrupting the stable manifold geometry established by log-preprocessing (Sener & Koltun, 2018; Yu et al., 2020). This scale disparity necessitates $\mathcal{D}_2$ **(Gradient Protection):** the design should prevent large linear-scale gradients from corrupting the logarithmic representation.

(c) Naive Residual Regression: To bypass direct gradient conflicts, an alternative is deploying an independent residual head $\hat{r}(\mathbf{X})$ to compensate for the base prediction. However, the pointwise residual target $r(\mathbf{X}) = Y - \hat{Y}(\mathbf{X})$ is fundamentally a mixture of systematic epistemic bias and unpredictable stochastic perturbations. Because its sign can vary rapidly across nearby inputs with substantial variance, forcing an unconstrained prediction head to directly fit this oscillatory signal from scratch readily triggers optimization oscillations and exacerbates the memorization of high-frequency noise (Arpit et al., 2017; Rahaman et al., 2019). This optimization pathology motivates our third desideratum, $\mathcal{D}_3$ **(Endogenous Smooth Proxy):** if residual-like compensatory signals are to be leveraged, the compensatory mechanism should avoid directly fitting discrete residuals with high-frequency sign oscillations.

3 METHODOLOGY

3.1 FROM NLL PATHOLOGY TO ENDOGENOUS FITTING-LAG SIGNAL

To fulfill the endogenous smooth proxy requirement ($\mathcal{D}_3$) without regressing non-stationary targets from scratch, we investigate the optimization dynamics of logarithmic regression under heteroscedastic uncertainty. In log space, the baseline probabilistic representation is supervised via the Negative Log-Likelihood (NLL) loss (Nix & Weigend, 1994; Kendall & Gal, 2017):

$$\mathcal{L}_{\text{NLL}} = \frac{(\ln Y - \mu(\mathbf{X}))^2}{2\sigma^2(\mathbf{X})} + \frac{1}{2}\ln \sigma^2(\mathbf{X}). \tag{3}$$

While prior work typically treats predictive variance–error coupling in NLL as an undesirable optimization pathology (Seitzer et al., 2022), we repurpose it. Evaluating the first-order optimality condition $\frac{\partial \mathcal{L}_{\text{NLL}}}{\partial \sigma^2(\mathbf{X})} = \frac{1}{2\sigma^2(\mathbf{X})}\left[1 - \frac{(\ln Y - \mu(\mathbf{X}))^2}{\sigma^2(\mathbf{X})}\right] = 0$ reveals that, if $\mu_t(\mathbf{X})$ were held fixed, the instantaneous conditional-risk optimum of the NLL objective for the variance branch would satisfy:

$$\sigma_t^{*2}(\mathbf{X}) = \mathbb{E}\left[(\ln Y - \mu_t(\mathbf{X}))^2 \mid \mathbf{X}\right] = \sigma^2_{\text{aleatoric}}(\mathbf{X}) + (Z^*(\mathbf{X}) - \mu_t(\mathbf{X}))^2, \tag{4}$$

where $Z^*(\mathbf{X}) \triangleq \mathbb{E}[\ln Y \mid \mathbf{X}]$ denotes the true conditional mean in log space, and $\sigma^2_{\text{aleatoric}}(\mathbf{X}) \triangleq \text{Var}(\ln Y \mid \mathbf{X})$. Governed by the spectral bias of deep neural networks (Rahaman et al., 2019), the mean prediction $\mu_t(\mathbf{X})$ prioritizes low-frequency modes and systematically lags behind local high-frequency transitions and sparse regions during early training. Consequently, the squared fitting lag $(Z^*(\mathbf{X}) - \mu_t(\mathbf{X}))^2$ surges dramatically, inducing a phenomenon of passive transient inflation wherein $\sigma_t^2(\mathbf{X})$ temporarily absorbs the instantaneous approximation error.

The 1D synthetic example in Figure 3 illustrates this transient coupling: $\sigma^2(\mathbf{X})$ inflates where the mean predictor $\mu(\mathbf{X})$ lags behind the target and subsides toward the aleatoric floor as fitting improves. Crucially, TVG does not require calibrated uncertainty; it exploits this temporary error-induced variance inflation as the endogenous smooth proxy for desideratum $\mathcal{D}_3$.

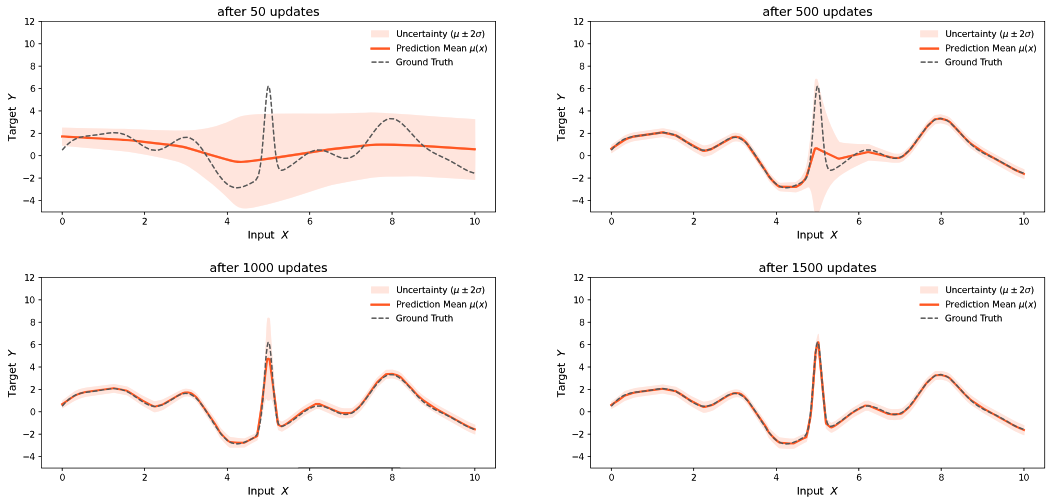

Figure 3: Transient evolution on a 1D synthetic dataset. Influenced by low-frequency spectral bias, predictive mean $\mu(X)$ underfits the high-frequency pulse early on, triggering transient inflation in variance $\sigma^2(X)$, which subsides upon convergence.

3.2 From Fitting-Lag Signal to Linear Intervention

This subsection addresses how to translate the statistical proxy into a signal for linear-scale intervention and instantiates the TVG framework. Although $\sigma^2(\mathbf{X})$ effectively localizes regional fitting lag, directly adopting it as a linear compensation term is hindered by three intrinsic bottlenecks: **(1) Scale mismatch:** Statistical variance and linear geometric distortion Δ operate on disparate numerical scales; **(2) Noise entanglement:** $\sigma^2(\mathbf{X})$ remains coupled with an irreducible aleatoric noise floor; and **(3) Static misalignment:** As $\mu(\mathbf{X})$ converges, static compensation without modulation cannot track the decaying variance dynamics, risking distortion from residual noise in later training stages (Stirn & Knowles, 2020).

To address these challenges, we introduce an input-dependent adaptive modulation head $\beta(\mathbf{X})$ to convert the statistical proxy into an adaptive log-domain modulation for original-scale correction, while enforcing gradient isolation ($\mathcal{D}_2$) on $\mu(\mathbf{X})$ via a stop-gradient operator $\mathrm{sg}[\cdot]$ (Chen & He, 2021). The auxiliary-branch prediction is formulated as:

$$\hat{Y}_{\text{aux}} = \exp\left(\mathrm{sg}[\mu(\mathbf{X})] + \beta(\mathbf{X}) \cdot \sigma^2(\mathbf{X})\right). \tag{5}$$

In this formulation, $\sigma^2(\mathbf{X})$ provides the smooth proxy basis, while $\beta(\mathbf{X})$ is intended to align local scales and adaptively decouple aleatoric noise so that the log-domain relative distortion $\ln(1 + \Delta/\hat{Y})$ can be absorbed within the exponent. This design aims to transform volatile residual fitting into smoother parameter regression ($\mathcal{D}_3$). Appendix B.4 further relates the modulation term $\beta(\mathbf{X})\sigma^2(\mathbf{X})$ to the ideal log-domain correction and details the resulting gradient flow through the shared representation.

As illustrated in Figure 4, the shared backbone first maps the input $\mathbf{X}$ to a latent representation $\mathbf{h}(\mathbf{X}) = f_\theta(\mathbf{X}) \in \mathbb{R}^d$, which branches into three parallel heads: (i) the Log-Mean Head $\mu(\mathbf{X})$; (ii) the Transient Variance Head $\sigma^2(\mathbf{X})$; and (iii) the Adaptive Modulation Head $\beta(\mathbf{X})$. Supervised by the original-scale target Y via the Mean Absolute Error (MAE), the composite objective of TVG is formulated as:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{NLL}}(\mu, \sigma^2; \ln Y) + \lambda \text{MAE}(\hat{Y}_{\text{aux}}, Y), \tag{6}$$

where λ balances the dual-space supervision strengths.

Asymmetric Gradient Shunting and Manifold Reshaping. Under equation 6, the parameter updates of the log-mean head $\mu(\mathbf{X})$ are driven exclusively by the stable logarithmic loss $\mathcal{L}_{\text{NLL}}$. This design achieves task decoupling, insulating the primary head from gradient directional conflicts induced by dual-space optimization objectives. Conversely, because the auxiliary (σ^2, β) pathway

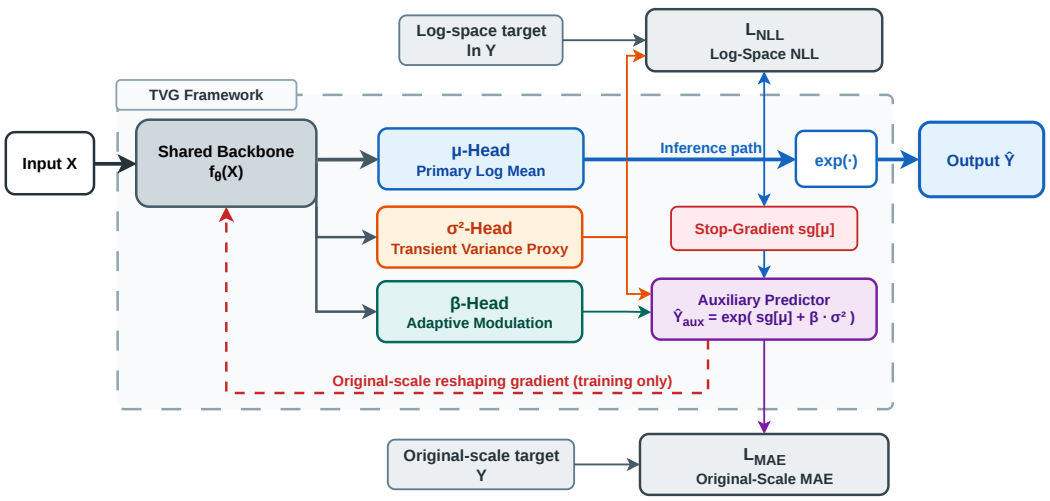

Figure 4: Architecture of the Transient Variance Guidance (TVG) framework. The shared backbone feeds three parallel heads. The log-mean head yields the primary-branch prediction $\hat{Y}$, while the variance and modulation heads with $\mathrm{sg}[\mu]$ form the auxiliary-branch prediction $\hat{Y}_{\mathrm{aux}}$. During training, linear intervention on $\hat{Y}_{\mathrm{aux}}$ unidirectionally reshapes the shared manifold via β and σ^2 while μ is shielded by $\mathrm{sg}[\cdot]$. At inference, the auxiliary branch is pruned for zero additional overhead.

learns a relative multiplicative modulation, the corrective gradients induced by the linear loss can propagate to the shared backbone through the $\beta(X)$ and $\sigma^2(X)$ heads. Within the early plasticity window (satisfying $\mathcal{D}_1$), this asymmetric gradient flow drives the shared manifold to internalize linear additive geometry. As $\mu(\mathbf{X})$ operates directly on this reshaped representation space, the standard inference mapping $\hat{Y} = \exp(\mu(\mathbf{X}))$ effectively suppresses the metric-warped interpolation bias (Ruder, 2017). Crucially, the auxiliary branches (σ^2, β) are completely pruned during inference, incurring zero additional computational overhead.

4 Experiments

4.1 Experimental Setup

Our empirical evaluation is organized around four research questions (RQs), covering overall performance, mechanism validation, training dynamics and efficiency, and cross-domain generalization.

RQ1 (Section 4.2): How does TVG compare with transformation and retransformation approaches on skewed regression tasks? We evaluate TVG on the CIKM2017 radar forecasting benchmark using ConvLSTM as the backbone, and compare against baseline families: (1) standard target transformations, (2) log space regression formulations, and (3) classical retransformation corrections.

RQ2 (Section 4.3): Are the core design components of TVG necessary, and do they support the desiderata $\mathcal{D}_1$–$\mathcal{D}_3$ introduced in Section 2.2? We compare TVG against the alternative intervention paradigms illustrated in Figure 2, including two-stage fine-tuning, parallel dual-branch training, and explicit compound optimization, and further ablate the stop-gradient operator and the transient-variance proxy through direct residual regression.

RQ3 (Section 4.4): Is the correction induced by the auxiliary branches transient, can these branches be removed at inference, and how sensitive is TVG to computational cost and the loss weight λ? We track the discrepancy between the primary-branch prediction $\hat{Y}$ and the auxiliary-branch prediction $\hat{Y}_{\mathrm{aux}}$ throughout training, profile training and inference efficiency, and evaluate sensitivity to λ.

RQ4 (Section 4.5): Do the gains of TVG generalize across datasets and model architectures? We evaluate TVG on four additional skewed datasets—Webtraffic, PM2.5, CAMELS, and Rossmann—and further test CNN-ResNet and Transformer backbones on CIKM2017.

Unless otherwise stated, all comparative methods use the same training budget and hyperparameter-search protocol, and reported results are averaged over five independent runs. Full architecture, optimization, preprocessing, and numerical-stabilization details are provided in Appendix C.

4.2 Comparison with Log-Regression and Retransformation Baselines

Table 1 reports the quantitative benchmarking results on the CIKM2017 dataset (Tianchi, 2018). TVG achieves the best performance across all evaluation metrics, reducing MAE by 9.97% and RMSE by 7.59% compared to the standard Log-MSE baseline, while yielding a 10.78% reduction in MAE relative to the Log-NLL baseline.

Table 1: Main performance comparison on the CIKM2017 dataset.

Method	**MAE**	**RMSE**
Z-Score Transformation	7.0515 ± 0.3088	10.2040 ± 0.3335
Box-Cox Transformation (Sakia, 1992)	7.0656 ± 0.2110	9.9923 ± 0.2661
Log-Space MSE Baseline	6.8930 ± 0.0780	10.2653 ± 0.1719
Log-Space NLL Baseline	6.9556 ± 0.1883	10.2441 ± 0.1898
Duan's Smearing (Duan, 1983)	7.0947 ± 0.1628	10.0872 ± 0.1635
TranSUN (Yu et al., 2025)	6.9810 ± 0.2551	10.3511 ± 0.3750
TVG (Ours)	$\mathbf{6.2059 \pm 0.0992}$	$\mathbf{9.4859 \pm 0.1623}$

Notably, pointwise retransformation corrections do not yield noticeable improvements over log space baselines. Furthermore, the standard Log-Space NLL baseline exhibits no performance gain over the MSE baseline. In contrast, by integrating supervision from both the log-transformed space and the original scale into a unified training objective, TVG consistently achieves superior performance across both MAE and RMSE, indicating that the improvement stems from the dual-space structural correction rather than the statistical properties of the NLL objective itself. To isolate this mechanism, controlled 1D experiments confirm TVG mitigates bias under locally additive structures, but its benefit strictly reverses under multiplicative ones. This matches our predicted boundary, proving TVG enforces targeted geometric constraints rather than generic regularization (Appendix A).

4.3 Ablations And Validation Of Design Desiderata

Table 2 details the quantitative ablation results for empirically assessing the three design desiderata ($\mathcal{D}_1$–$\mathcal{D}_3$).

Table 2: Ablation study and failure analysis of fusion paradigms (Backbone: ConvLSTM).

Category	**Variant**	**Desiderata**	**MAE**	**RMSE**
Reference Baseline	Log-Space MSE Baseline	–	6.8930 ± 0.0780	10.2653 ± 0.1719
Traditional Fusion	Two-Stage Fine-Tuning	$\mathcal{D}_1$	6.9094 ± 0.2541	10.3093 ± 0.2905
	Parallel Dual-Branch Training	$\mathcal{D}_2$	7.0691 ± 0.2414	10.2135 ± 0.3713
	Explicit Compound Loss	$\mathcal{D}_2$	6.9669 ± 0.1149	10.0239 ± 0.2133
Internal Ablation	Without Stop-Gradient	$\mathcal{D}_2$	6.6288 ± 0.1645	9.7482 ± 0.1832
	Direct Residual Regression	$\mathcal{D}_3$	6.8661 ± 0.2291	10.3119 ± 0.2911
Full Framework	**TVG (Ours)**	–	$\mathbf{6.2059 \pm 0.0992}$	$\mathbf{9.4859 \pm 0.1623}$

$\mathcal{D}_1$ **Verification:** Evaluation metrics of two-stage fine-tuning (MAE 6.9094) are essentially on par with the naive baseline (MAE 6.8930) and fall substantially behind the complete TVG framework (MAE 6.2059). This indicates that disjointed sequential optimization struggles to effectively transfer scale-complementary information across stages, since the representation may already have hardened into the smooth logarithmic basin before linear intervention begins.

$\mathcal{D}_2$ **Verification:** Removing the stop-gradient operator $\mathrm{sg}[\cdot]$ from TVG degrades the MAE from 6.2059 to 6.6288, while parallel dual-branch and compound loss structures exhibit even more severe

degradation despite possessing equivalent or even larger parameter capacity than TVG. This directly rules out increased parameter count as the source of improvement. Furthermore, although these multi-task variants could in principle induce implicit adaptive regularization, landscape curvature smoothing, or Hessian spectral conditioning, their inferior performance demonstrates that such generic optimization benefits are insufficient to explain TVG's gains. Instead, the results are consistent with bidirectional gradient flow introducing interference into the backbone representations, and they support the importance of unidirectional blocking in preventing negative transfer from the auxiliary branch.

$\mathcal{D}_3$ **Verification:** Replacing the transient variance basis of TVG with direct residual regression leads to an MAE degradation to 6.8661, yielding only marginal improvements over the baseline. This variant also forms a multi-task structure with an auxiliary residual head, and thus could potentially benefit from feature decoupling or improved representation robustness. Its poor performance indicates that neither multi-task decoupling nor direct residual learning alone provides effective auxiliary supervision. The adaptive, smooth proxy based on transient variance is therefore crucial for the observed gains.

4.4 Training Evolution , Efficiency, and Sensitivity

Transient Correction and Auxiliary-Branch Removal at Inference. Figure 5 traces the evolutionary trajectory of the absolute discrepancy between the primary-branch prediction $\exp(\mu(\mathbf{X}))$ and the auxiliary-branch prediction prediction $\hat{Y}_{\text{aux}}$ across training epochs. During early training, a pronounced deviation exists between the two predictions; as optimization proceeds, this discrepancy exhibits a steady downward trend and gradually converges to an extremely low level. This trend illustrates the transient nature of the auxiliary correction: as NLL training converges, the modulation term $\beta(\mathbf{X})\sigma^2(\mathbf{X})$ shrinks, bringing the primary-branch and auxiliary-branch predictions into agreement. This transient behavior is why the auxiliary branch can be discarded at inference.

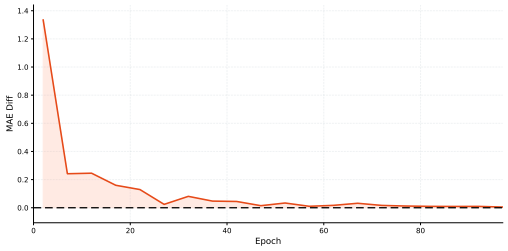

Figure 5: Evolution of the mean absolute discrepancy between $\hat{Y}$ and $\hat{Y}_{\text{aux}}$.

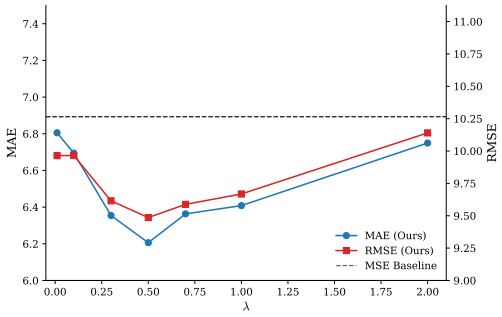

Figure 6: Hyperparameter sensitivity analysis of the loss balancing weight λ.

Computational Efficiency Analysis. Table 3 compares the computational and empirical overhead of TVG with the ConvLSTM baseline on CIKM2017. TVG introduces only 0.0217 M additional parameters during training, while increasing peak training memory by merely 0.33 MB. At inference, the auxiliary heads are fully pruned, leaving the backbone and theoretical inference FLOPs unchanged; empirical memory usage and throughput also remain essentially identical to the baseline. Thus, TVG adds negligible training overhead and requires no additional inference-time modules or FLOPs.

Table 3: Computational and empirical efficiency analysis.

Method	**Params (M)**	**Infer FLOPs (G)**	**Train VRAM (MB)**	**Infer VRAM (MB)**	**Throughput (samples/s)**
Baseline	1.0471	6.5755	4135.13	1601.75	744.77
TVG	1.0688	6.5755	4135.46	1601.83	743.93
Difference (Δ)	*+0.0217*	*≈ 0*	*+0.33*	*+0.08*	*-0.84*

Sensitivity Evaluation of Loss Weight λ. As shown in Figure 6, TVG consistently outperforms the baseline across $\lambda \in [0.01, 2.0]$, exhibiting broad parameter tolerance against tuning.

4.5 Cross-Domain and Cross-Architecture Generalization

Cross-Domain Skewed Dataset Generalization. As summarized in Table 4, TVG consistently outperforms the standard baseline across four benchmark skewed datasets from diverse domains: Webtraffic (Salinas et al., 2020), PM2.5 (Liang et al., 2015), CAMELS (Addor et al., 2017), and Rossmann (Guo & Berkhahn, 2016). These results indicate that TVG is not confined to specific radar echo tasks, but extends effectively to diverse continuous time-series forecasting scenarios characterized by heteroscedasticity or skewed distributions.

Table 4: Cross-domain generalization performance across multiple datasets (Baseline: standard Log-MSE).

Dataset	Backbone	Baseline (MAE)	TVG (MAE)	Baseline (RMSE)	TVG (RMSE)
Webtraffic	LSTM	527.27 ± 14.51	$\mathbf{504.37 \pm 9.45}$	20352.99 ± 1303.9	$\mathbf{16795.86 \pm 787.3}$
PM2.5	ConvLSTM	10.1278 ± 0.0723	$\mathbf{9.6421 \pm 0.0182}$	20.0517 ± 0.2696	$\mathbf{19.0804 \pm 0.1193}$
CAMELS	LSTM	0.4649 ± 0.0054	$\mathbf{0.4510 \pm 0.0034}$	1.6281 ± 0.0330	$\mathbf{1.6060 \pm 0.0122}$
Rossmann	LSTM	2091.0646 ± 26.8715	$\mathbf{1904.0893 \pm 12.1554}$	2844.1661 ± 48.5640	$\mathbf{2761.4195 \pm 30.3302}$

Architecture Agnosticism. Table 5 reports the performance of TVG when applied to CNN-ResNet and Transformer backbones on CIKM2017. Across different backbone designs, TVG yields consistent and substantial MAE reductions ranging from 4.05% to 9.97%. These empirical findings indicate that the performance gains of TVG do not rely on implicit smoothing assumptions tied to specific network architectures, demonstrating that this method, functioning as a lightweight training constraint, exhibits strong compatibility and complementarity with mainstream spatiotemporal backbones.

Table 5: Plug-and-play performance gains across diverse network backbones on CIKM2017.

Backbone	Baseline (MAE)	TVG (MAE)	Gain	Baseline (RMSE)	TVG (RMSE)	Gain
ConvLSTM	6.8930 ± 0.0780	$\mathbf{6.2059 \pm 0.0992}$	9.97%	10.2653 ± 0.1719	$\mathbf{9.4859 \pm 0.1623}$	7.59%
CNN-ResNet	8.1213 ± 0.1824	$\mathbf{7.5656 \pm 0.2516}$	6.84%	11.5695 ± 0.1464	$\mathbf{11.0168 \pm 0.3587}$	4.78%
Transformer	7.5939 ± 0.0668	$\mathbf{7.2860 \pm 0.1826}$	4.05%	10.7138 ± 0.1391	$\mathbf{10.4822 \pm 0.2120}$	2.16%

5 Conclusion and Limitations

In this paper, we address the metric-warped interpolation bias in log-transformed deep regression. We analyze this bias under a local affine interpolation assumption and show that the smooth inductive bias of neural networks, combined with the strict convexity of the exponential mapping, induces a systematic distortion that is deterministic, path-dependent, and orthogonal to classical pointwise retransformation corrections. To address this problem, we propose Transient Variance Guidance (TVG), which repurposes the transient variance inflation of NLL optimization as an endogenous smooth proxy and couples it with asymmetric gradient shunting, allowing the representation to absorb linear-scale constraints during early training while incurring zero inference overhead. To evaluate TVG, we conduct experiments on the CIKM2017 benchmark against transformation, log space, and retransformation baselines; ablation studies targeting the three design desiderata ($\mathcal{D}_1$–$\mathcal{D}_3$); efficiency and sensitivity analyses; and cross-domain evaluations on four additional skewed datasets and two additional backbones. TVG reduces MAE from 6.8930 to 6.2059 and RMSE from 10.2653 to 9.4859 on CIKM2017, and consistently outperforms baselines across all evaluated datasets and architectures. These empirical gains provide indirect support for the hypothesized mechanism of reducing metric-warped interpolation bias.

Limitations and Future Work. TVG assumes that the system possesses local additivity and relatively stationary noise. If the underlying mechanism is intrinsically purely multiplicative, or if extreme non-stationary noise and sharp discontinuities are present, the guidance mechanism will be limited. Future work may explore adaptive disentanglement and discontinuity-aware operators to extend its applicability boundary. Because the relevant interpolation occurs in the learned latent representation, the interpolation bias itself is not directly observable or estimable in our setting; we therefore interpret the empirical gains as indirect support for the hypothesized mechanism rather than as direct causal evidence.

Reproducibility Statement

We provide controlled synthetic validation in Appendix A, complete theoretical derivations and analyses in Appendix B, and detailed architecture, optimization, preprocessing, and numerical-stabilization settings in Appendix C. Unless otherwise stated, all comparative results are averaged over five independent runs. Anonymous source code and experimental configurations are provided in the supplementary material.

AI use statement

In this work, we used generative AI tools for language polishing and editing of the manuscript, and for debugging code used in our experiments. We have not used generative AI tools for generating research ideas, hypotheses, or scientific claims; producing or altering experimental data, results, or figures; or writing the core technical contributions, and the remaining required-disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for improving clarity and concision and for suggesting alternative phrasings. We have reviewed all AI-assisted work. Specifically, all AI-edited text was checked against our original results, references, and intended claims by the authors, all AI-suggested code changes were tested and verified for correctness by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

References

Nans Addor, Andrew J Newman, Naoki Mizukami, and Martyn P Clark. The camels data set: catchment attributes and meteorology for large-sample studies. *Hydrology and Earth System Sciences*, 21(10):5293–5313, 2017.

Devansh Arpit, Stanisław Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *International Conference on Machine Learning (ICML)*, pp. 233–242, 2017.

David G. T. Barrett and Benoit Dherin. Implicit gradient regularization. In *International Conference on Learning Representations*, 2021.

Peter L. Bartlett, Andrea Montanari, and Alexander Rakhlin. Deep learning: A statistical viewpoint. *Acta Numerica*, 30:87–201, 2021. doi: 10.1017/S0962492921000027.

Ronen Basri, David W. Jacobs, Yoni Kasten, and Shira Kritchman. The convergence rate of neural networks for learned functions of different frequencies. In *Advances in Neural Information Processing Systems*, volume 32, pp. 4761–4771, 2019.

Kenneth Belitz and PE Stackelberg. Evaluation of six methods for correcting bias in estimates from ensemble tree machine learning regression models. *Environmental Modelling & Software*, 139: 105006, 2021.

Guy Blanc, Neha Gupta, Gregory Valiant, and Paul Valiant. Implicit regularization for deep neural networks driven by an ornstein-uhlenbeck like process. In *Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pp. 483–513. PMLR, 2020.

Hsin-Yu Chen, Cheng-Te Li, and Ting-Yu Chen. Skewness-aware boosting regression trees for customer contribution prediction in financial precision marketing. In *Companion Proceedings of the ACM Web Conference (WWW)*, pp. 461–470, 2024.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15750–15758, 2021.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International Conference on Machine Learning (ICML)*, pp. 794–803, 2018.

Shibhansh Dohare, J Fernando Hernandez-Garcia, Qingfeng Lan, Parash Rahman, A Rupam Mahmood, and Richard S Sutton. Loss of plasticity in deep continual learning. *Nature*, 632(8026): 768–774, 2024.

Naihua Duan. Smearing estimate: a nonparametric retransformation method. *Journal of the American Statistical Association*, 78(383):605–610, 1983.

David J Finney. On the distribution of a variate whose logarithm is normally distributed. *Supplement to the Journal of the Royal Statistical Society*, 7(2):155–161, 1941.

Cheng Guo and Felix Berkhahn. Entity embeddings of categorical variables. *arXiv preprint arXiv:1604.06737*, 2016.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Xuan Liang, Tao Zou, Bin Guo, Shuo Li, Haozhe Zhang, Shuyi Zhang, Hui Huang, and Song Xi Chen. Assessing beijing's pm2.5 pollution: severity, weather impact, apec and winter heating. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 471(2182): 20150257, 2015.

Clare Lyle, Zeyu Zheng, Evgenii Nikishin, Bernardo Avila Pires, Razvan Pascanu, and Will Dabney. Understanding plasticity in neural networks. In *International Conference on Machine Learning (ICML)*, pp. 23190–23211, 2023.

Roei Mulayoff, Tomer Michaeli, and Daniel Soudry. The implicit bias of minima stability: A view from function space. In *Advances in Neural Information Processing Systems*, volume 34, pp. 17749–17761, 2021.

David A Nix and Andreas S Weigend. Estimating the mean and variance of the target probability distribution. In *IEEE International Conference on Neural Networks (ICNN)*, volume 1, pp. 55–60, 1994.

Rahul Parhi and Robert D. Nowak. Banach space representer theorems for neural networks and ridge splines. *Journal of Machine Learning Research*, 22(43):1–40, 2021.

Rahul Parhi and Robert D. Nowak. What kinds of functions do deep neural networks learn? insights from variational spline theory. *SIAM Journal on Mathematics of Data Science*, 4(2):464–489, 2022. doi: 10.1137/21M1418642.

Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *International Conference on Machine Learning (ICML)*, pp. 5301–5310, 2019.

Sebastian Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

Remi M Sakia. The box-cox transformation technique: a review. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 41(2):169–178, 1992.

David Salinas, Valentin Flunkert, Jan Gasthaus, and Tim Januschowski. Deepar: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3): 1181–1191, 2020.

Pedro Savarese, Itay Evron, Daniel Soudry, and Nathan Srebro. How do infinite width bounded norm networks look in function space?, 2019. URL `https://arxiv.org/abs/1902.05040`.

Maximilian Seitzer, Arash Tavakoli, Dimitrije Antic, and Georg Martius. On the pitfalls of heteroscedastic uncertainty estimation with probabilistic neural networks. In *International Conference on Learning Representations (ICLR)*, 2022.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

Andrew Stirn and David A Knowles. Variational variance: Simple, reliable, calibrated heteroscedastic noise variance parameterization. *arXiv preprint arXiv:2006.04910*, 2020.

Tianchi. CIKM AnalytiCup 2017 short-term quantitative precipitation forecasting dataset. `https://tianchi.aliyun.com/dataset/dataDetail?dataId=1085`, 2018.

Yunpeng Weng, Xing Tang, Zhenhao Xu, Fuyuan Lyu, Dugang Liu, Zexu Sun, and Xiuqiang He. Optdist: Learning optimal distribution for customer lifetime value prediction. In *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM)*, pp. 2523–2533, 2024.

Zhi-Qin John Xu, Yaoyu Zhang, Tao Luo, Yanyang Xiao, and Zheng Ma. Frequency principle: Fourier analysis sheds light on deep neural networks. *Communications in Computational Physics*, 28(5):1746–1767, 2020.

Jiahao Yu, Haozhuang Liu, Yeqiu Yang, Lu Chen, Jian Wu, Yuning Jiang, and Bo Zheng. Transun: A preemptive paradigm to eradicate retransformation bias intrinsically from regression models in recommender systems. In *Advances in Neural Information Processing Systems(NeurIPS)*, 2025.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *arXiv preprint arXiv:2001.06782*, 2020.

Mingyu Zhao, Zhaohan Li, Zhenxiong Miao, Xu Zhang, Dewei Leng, Yanan Niu, and Kun Gai. Pit-sun: A deployable empirical marginal transform framework with expectation-consistent recovery for regression in recommender systems, 2026. URL `https://arxiv.org/abs/2607.08202`.

A Controlled Validation of the Additive–Multiplicative Boundary

To isolate the mechanism from confounding factors in real data, we construct two idealized 1D regression problems that share the same training setup ($n_{\text{train}} = 5$ uniformly spaced points, 500 seeds, 2000 training epochs) but differ in the underlying data-generating structure. The additive setting (Appendix A.1) instantiates the metric-warped interpolation bias analyzed in Section 2, while the multiplicative setting (Appendix A.2) serves as a directional counterpart that probes the applicability boundary of TVG.

A.1 Additive Regime: Gain Scaling with Metric Warping

The true function is linear in the original space,

$$\mu(u) = a + bu, \qquad u \in [0, 1], \tag{7}$$

and the network is trained on only $n_{\text{train}} = 5$ uniformly spaced points. Under this setting, the unbiased baseline in the original space is the arithmetic interpolation, while log space training produces a geometric interpolation, yielding exactly the metric-warped bias analyzed in Section 2. We sweep the slope b to control the local logarithmic span δ between adjacent anchors.

Sanity checks. Table 6 confirms that the theoretical setting is correctly instantiated. The ground truth matches the arithmetic interpolation to machine precision (`arith_chk` $\sim 10^{-6}$), and the exact geometric–arithmetic bias aligns with its second-order approximation in Eq. equation 2 ($\rho_{ea} > 0.97$ across all configurations). This verifies that the bias in Eq. equation 2 is not an artifact of the derivation but a faithful description of the interpolation geometry in the idealized setting.

Gain scaling with local span. Table 7 reports TVG's gain as a function of b. The gain is positive in every configuration and increases monotonically with δ from $+0.005$ at $b = 10$ to $+0.363$ at $b = 100$. Two patterns are worth noting. First, at small b, the interpolation bias itself is small and the gain is correspondingly small; the statistical significance at $b = 10$ reflects the low variance of the gain estimator in that regime rather than a large effect. Second, the gain-to-δ^2 ratio grows with δ rather than staying constant, consistent with the interpolation bias being partially masked by network approximation noise when the bias is small. Taken together, these results provide controlled evidence that TVG mitigates the metric-warped interpolation bias when the bias is large enough to dominate network approximation error, and that the effect scales with the local logarithmic span in the direction predicted by the theory.

Table 6: Sanity checks for the idealized additive 1D experiment. `arith_chk` verifies that the ground truth equals the arithmetic interpolation when the true function is linear in the original space. ρ_{ea} measures the Spearman correlation between the exact geometric–arithmetic bias and its second-order approximation in Eq. equation 2.

b	`arith_chk`	ρ_{ea}
10	2.62×10^{-7}	$+0.998$
30	5.01×10^{-7}	$+0.991$
50	7.27×10^{-7}	$+0.986$
70	1.22×10^{-6}	$+0.984$
100	1.62×10^{-6}	$+0.973$

Table 7: Idealized additive 1D interpolation experiment. The true function is linear in the original space, and the network must interpolate between $n_{\text{train}} = 5$ uniformly spaced training points. Both Baseline MAE and TVG MAE are evaluated on a dense grid of interpolated test points. All results are averaged over 500 seeds with 2000 training epochs. δ denotes the local logarithmic span between adjacent anchors. TVG's gain increases monotonically with δ and is statistically significant at $b = 10$ ($p = 0.021$) and at $b \geq 50$ ($p \leq 0.012$), but not at $b = 30$ ($p = 0.086$).

b	δ^2	Baseline MAE	TVG MAE	Gain	Gain/δ^2	t	p
10	0.48	0.073 ± 0.038	0.068 ± 0.047	$+0.005 \pm 0.050$	0.011	2.31	0.021
30	1.9	0.34 ± 0.18	0.32 ± 0.14	$+0.015 \pm 0.20$	0.008	1.72	0.086
50	3.2	0.74 ± 0.38	0.69 ± 0.30	$+0.050 \pm 0.44$	0.016	2.53	0.012
70	4.5	1.23 ± 0.66	1.05 ± 0.38	$+0.185 \pm 0.70$	0.041	5.93	< 0.0001
100	5.8	2.07 ± 1.17	1.70 ± 0.78	$+0.363 \pm 1.32$	0.063	6.14	< 0.0001

A.2 Multiplicative Counterpart: Applicability Boundary

To verify that TVG's gains are specific to additively structured data rather than generic regularization, we conduct a multiplicative counterpart of the idealized 1D experiment. We set the true function to be

$$\mu(u) = \exp(a + bu), \qquad u \in [0, 1], \tag{8}$$

with $a = 1$ and sweep $b \in \{1, 2, 3, 4, 5\}$. Here b controls the strength of the multiplicative structure: at small b, the target is locally near-linear in the original space, while at large b, geometric interpolation becomes the correct ground truth.

Sanity checks. Table 8 confirms that the multiplicative setting is correctly instantiated. The geometric interpolation matches the true function to high precision (`geo_chk` $\sim 10^{-6}$), while the arithmetic interpolation deviates increasingly as b grows, with `arith_chk` rising from 2.56×10^{-2} at $b = 1$ to 1.07×10^{1} at $b = 5$. The exact geometric–arithmetic bias remains well approximated by its second-order form ($\rho_{ea} > 0.99$ across all configurations), confirming that the theoretical setting is consistent with the derivation in Eq. equation 2.

Gain transition. Table 9 reports TVG's gain. For $b \leq 3$, the gain remains positive ($+0.0038$ to $+0.0123$), indicating that when the multiplicative strength is moderate, the additive approximation

is still locally valid and TVG acts as a mild regularizer. For $b \geq 4$, the gain becomes negative, and at $b = 5$ the degradation is severe (-18.28). This transition coincides with the rapid growth of `arith_chk`, which indicates that the additive assumption underlying TVG's linear constraint is increasingly violated. These results show that TVG's effect is directional rather than generic. The sign change of the gain occurs precisely where `arith_chk` grows sharply (from 8.45×10^{-1} at $b = 3$ to 3.14×10^{0} at $b = 4$), i.e., where the additive approximation underlying TVG's linear constraint is no longer locally valid. A method that merely regularized the network uniformly would not exhibit such a sharp sign change at this boundary.

Table 8: Sanity checks for the multiplicative 1D experiment. `geo_chk` verifies that the ground truth equals the geometric interpolation when the true function is multiplicative. `arith_chk` measures the deviation of the arithmetic interpolation, which grows with the multiplicative strength b. ρ_{ea} measures the Spearman correlation between the exact geometric–arithmetic bias and its second-order approximation in Eq. equation 2.

b	`arith_chk`	`geo_chk`	ρ_{ea}
1	2.56×10^{-2}	2.74×10^{-7}	$+0.998$
2	1.90×10^{-1}	2.44×10^{-7}	$+0.996$
3	8.45×10^{-1}	1.38×10^{-6}	$+0.995$
4	3.14×10^{0}	2.20×10^{-6}	$+0.996$
5	1.07×10^{1}	9.35×10^{-6}	$+0.996$

Table 9: Multiplicative 1D interpolation experiment. The true function is $Y = \exp(a + bu)$ with $a = 1$, and the network must interpolate between $n_{\text{train}} = 5$ uniformly spaced training points. Both Baseline MAE and TVG MAE are evaluated on a dense grid of interpolated test points. All results are averaged over 500 seeds with 2000 training epochs. TVG's gain transitions from positive to negative as the multiplicative strength b increases. At $b = 5$, the large standard deviation reflects a heavy-tailed distribution across seeds; the sign of the mean gain, rather than its magnitude, is the relevant signal.

b	Baseline MAE	TVG MAE	Gain	t	p
1	0.0086 ± 0.0050	0.0048 ± 0.0032	$+0.0038 \pm 0.0057$	14.859	< 0.0001
2	0.0250 ± 0.0157	0.0169 ± 0.0126	$+0.0081 \pm 0.0207$	8.776	< 0.0001
3	0.0654 ± 0.0431	0.0531 ± 0.0473	$+0.0123 \pm 0.0640$	4.288	< 0.0001
4	0.1536 ± 0.1256	0.2065 ± 0.4828	-0.0528 ± 0.5005	-2.359	0.0187
5	0.3628 ± 0.3269	18.6470 ± 20.9750	-18.2842 ± 21.0113	-19.439	< 0.0001

B Extended Theoretical Derivations and Analysis

In this section, we provide comprehensive mathematical derivations and extended theoretical analyses for the formulations presented in Section 2 and Section 3. Specifically, Appendix B.1 provides the complete step-by-step derivation of the metric-warped interpolation bias in Equation equation 2. Appendix B.2 presents a non-perturbative and higher-order truncation error analysis, rigorously proving that the structural concavity remains invariant regardless of truncation order or the magnitude of δ. Appendix B.3 details the first-order optimality condition and the stationary point derivation of the transient variance in Equation equation 4.

B.1 Complete Derivation of Equation equation 2

Let $Y_1^*, Y_2^* \in \mathbb{R}^+$ represent two localized conditional expectation anchors in the original space. Under the logarithmic transformation, the corresponding representations in the log space are defined as:

$$Z_1 = \ln Y_1^*, \quad Z_2 = \ln Y_2^*. \tag{9}$$

We define the local logarithmic span between the two anchors as $\delta \triangleq Z_2 - Z_1$, which implies:

$$Z_2 = Z_1 + \delta \quad \Longleftrightarrow \quad Y_2^* = Y_1^* \exp(\delta). \tag{10}$$

Assuming local linear interpolation in log space governed by the smooth inductive bias of neural networks, the intermediate prediction path parameterized by the interpolation coefficient $\alpha \in [0, 1]$ is given by:

$$Z_\alpha = (1-\alpha)Z_1 + \alpha Z_2 = Z_1 + \alpha(Z_2 - Z_1) = Z_1 + \alpha\delta. \tag{11}$$

When mapped back to the original space via the canonical inverse exponential transformation $\exp(\cdot)$, the model's actual predictive trajectory is the weighted geometric mean:

$$\hat{Y} = \exp(Z_\alpha) = \exp(Z_1 + \alpha\delta) = Y_1^* \exp(\alpha\delta). \tag{12}$$

Conversely, under the assumption of local linear additivity in the original space, the unbiased normative conditional expectation baseline is the arithmetic mean:

$$Y^* = (1-\alpha)Y_1^* + \alpha Y_2^* = (1-\alpha)Y_1^* + \alpha Y_1^* \exp(\delta) = Y_1^* \left[(1-\alpha) + \alpha \exp(\delta)\right]. \tag{13}$$

The algebraic discrepancy between the unbiased baseline Y^* and the model prediction $\hat{Y}$ is denoted by $\Delta \triangleq Y^* - \hat{Y}$. Substituting Equations equation 12 and equation 13 yields the exact expression:

$$\Delta(\alpha, \delta) = Y_1^* \left[1 - \alpha + \alpha \exp(\delta) - \exp(\alpha\delta)\right]. \tag{14}$$

To characterize the dominant geometric behavior around small to moderate logarithmic spans ($\delta \approx 0$), we perform a second-order Taylor series expansion of the exponential terms $\exp(\delta)$ and $\exp(\alpha\delta)$ around $\delta = 0$:

$$\exp(\delta) = 1 + \delta + \frac{1}{2}\delta^2 + \mathcal{O}(\delta^3), \tag{15}$$

$$\exp(\alpha\delta) = 1 + \alpha\delta + \frac{1}{2}(\alpha\delta)^2 + \mathcal{O}(\delta^3) = 1 + \alpha\delta + \frac{1}{2}\alpha^2\delta^2 + \mathcal{O}(\delta^3). \tag{16}$$

Substituting Equation equation 15 into the expression for Y^* in Equation equation 13:

$$\begin{aligned} Y^* &= Y_1^* \left[(1-\alpha) + \alpha\left(1 + \delta + \frac{1}{2}\delta^2 + \mathcal{O}(\delta^3)\right)\right] \\ &= Y_1^* \left[1 - \alpha + \alpha + \alpha\delta + \frac{1}{2}\alpha\delta^2 + \mathcal{O}(\delta^3)\right] \\ &= Y_1^* \left[1 + \alpha\delta + \frac{1}{2}\alpha\delta^2 + \mathcal{O}(\delta^3)\right]. \end{aligned} \tag{17}$$

Substituting Equation equation 16 into the expression for $\hat{Y}$ in Equation equation 12:

$$\hat{Y} = Y_1^* \left[1 + \alpha\delta + \frac{1}{2}\alpha^2\delta^2 + \mathcal{O}(\delta^3)\right]. \tag{18}$$

Subtracting Equation equation 18 from Equation equation 17, the zeroth-order and first-order linear terms cancel out identically:

$$\begin{aligned} \Delta &= Y^* - \hat{Y} \\ &= Y_1^* \left[\left(1 + \alpha\delta + \frac{1}{2}\alpha\delta^2\right) - \left(1 + \alpha\delta + \frac{1}{2}\alpha^2\delta^2\right) + \mathcal{O}(\delta^3)\right] \\ &= Y_1^* \left[\frac{1}{2}(\alpha - \alpha^2)\delta^2 + \mathcal{O}(\delta^3)\right] \\ &= \frac{1}{2}Y_1^*\alpha(1-\alpha)\delta^2 + \mathcal{O}(\delta^3). \end{aligned} \tag{19}$$

Neglecting higher-order residual terms $\mathcal{O}(\delta^3)$ completes the derivation of Equation equation 2 in the main text.

B.2 Higher-Order Truncation Error Analysis

A natural question arises: *does the conclusion of systematic underestimation ($\Delta > 0$) hold when the local logarithmic span δ is sufficiently large, or is it an artifact of truncating the Taylor series at the second order?*

Below, we rigorously prove that the core physical property of systematic underestimation within the interpolation interval ($\alpha \in (0, 1)$) is an immutable structural characteristic that is strictly invariant to the truncation order and holds globally for any $\delta \neq 0$. We establish this via two complementary perspectives: an analytical higher-order expansion and a non-perturbative global geometric proof.

B.2.1 Analytical Higher-Order Expansion

Recall the exact dimensionless discrepancy function from Equation equation 14:

$$g(\alpha, \delta) \triangleq \frac{\Delta(\alpha, \delta)}{Y_1^*} = (1 - \alpha) + \alpha \exp(\delta) - \exp(\alpha\delta). \tag{20}$$

Using the infinite Maclaurin series expansion of the exponential function $\exp(u) = \sum_{k=0}^{\infty} \frac{u^k}{k!}$, we express $g(\alpha, \delta)$ as:

$$\begin{aligned} g(\alpha, \delta) &= (1 - \alpha) + \alpha \sum_{k=0}^{\infty} \frac{\delta^k}{k!} - \sum_{k=0}^{\infty} \frac{(\alpha\delta)^k}{k!} \\ &= (1 - \alpha + \alpha - 1) + (\alpha - \alpha)\delta + \sum_{k=2}^{\infty} \frac{\alpha - \alpha^k}{k!} \delta^k \\ &= \sum_{k=2}^{\infty} C_k(\alpha)\delta^k, \end{aligned} \tag{21}$$

where the k-th order polynomial coefficient $C_k(\alpha)$ is defined as:

$$C_k(\alpha) \triangleq \frac{\alpha(1 - \alpha^{k-1})}{k!}. \tag{22}$$

For any interior interpolation weight $\alpha \in (0, 1)$ and for all orders $k \geq 2$, since $0 < \alpha < 1$, it strictly holds that:

$$\alpha^{k-1} < 1 \quad \Longrightarrow \quad 1 - \alpha^{k-1} > 0 \quad \Longrightarrow \quad C_k(\alpha) > 0, \quad \forall k \geq 2. \tag{23}$$

Now, expanding $g(\alpha, \delta)$ up to the fourth order:

$$g(\alpha, \delta) = \underbrace{\frac{\alpha(1 - \alpha)}{2!}\delta^2}_{\text{2nd-order (Dominant)}} + \underbrace{\frac{\alpha(1 - \alpha^2)}{3!}\delta^3}_{\text{3rd-order (Asymmetry)}} + \underbrace{\frac{\alpha(1 - \alpha^3)}{4!}\delta^4}_{\text{4th-order (Curvature)}} + \mathcal{O}(\delta^5). \tag{24}$$

Analyzing Equation equation 24 reveals two crucial insights:

1. **Directional Skewness from the 3rd-Order Term:** The coefficient of the cubic term is $C_3(\alpha) = \frac{\alpha(1-\alpha)(1+\alpha)}{6} > 0$. When $\delta > 0$ (i.e., transitioning from low to high target values), the cubic term is positive, compounding the underestimation error. When $\delta < 0$ (transitioning from high to low values), the cubic term partially counters the quadratic term, inducing a slight directional asymmetry.
2. **Positivity Preservation via Higher Even Terms:** Even orders ($k = 2, 4, 6, \dots$) satisfy $C_k(\alpha)\delta^k > 0$ universally for all $\delta \neq 0$. Because factorial growth in the denominator $k!$ ensures rapid convergence of the series, the positive quadratic core $\frac{1}{2}\alpha(1 - \alpha)\delta^2$ strictly dominates the sign of the summation over typical local transitions.

B.2.2 Non-Perturbative Proof via Strict Convexity (Jensen's Inequality)

To resolve any ambiguity regarding the global convergence or the sign of the infinite tail, we invoke the strict convexity of the exponential mapping directly without relying on series truncation.

Proposition 1 (Global Invariance of Underestimation Bias)**.** *For any non-trivial local logarithmic span $\delta \neq 0$ (i.e., $Z_1 \neq Z_2$) and any interior interpolation parameter $\alpha \in (0, 1)$, the metric-warped interpolation bias is strictly positive:*

$$\Delta(\alpha, \delta) = Y^* - \hat{Y} > 0. \tag{25}$$

Proof. Consider the real scalar function $f(u) = \exp(u)$ defined on $\mathbb{R}$. Its second derivative satisfies:

$$f''(u) = \exp(u) > 0, \quad \forall u \in \mathbb{R}. \tag{26}$$

Thus, $f(u) = \exp(u)$ is strictly convex on the entire real line.

By definition, for any two distinct points $Z_1, Z_2 \in \mathbb{R}$ ($Z_1 \neq Z_2$) and any convex combination weight $\alpha \in (0, 1)$, Jensen's inequality for strictly convex functions asserts that:

$$f\left((1-\alpha)Z_1 + \alpha Z_2\right) < (1-\alpha)f(Z_1) + \alpha f(Z_2). \tag{27}$$

Substituting $f(u) = \exp(u)$, $Z_1 = \ln Y_1^*$, and $Z_2 = \ln Y_2^*$:

$$\exp\left((1-\alpha)\ln Y_1^* + \alpha \ln Y_2^*\right) < (1-\alpha)\exp(\ln Y_1^*) + \alpha \exp(\ln Y_2^*). \tag{28}$$

Recalling that the left-hand side is precisely the model prediction $\hat{Y} = \exp(Z_\alpha)$ and the right-hand side is the unbiased arithmetic expectation $Y^* = (1-\alpha)Y_1^* + \alpha Y_2^*$, we obtain:

$$\hat{Y} < Y^* \quad \Longleftrightarrow \quad \Delta = Y^* - \hat{Y} > 0. \tag{29}$$

This confirms that the phenomenon of systematic underestimation is not a local approximation artifact of the second-order Taylor expansion, but a fundamental, non-perturbative physical consequence of the strict convexity of the exponential retransformation. □

B.3 Stationary Point Proof of Transient Variance

In this subsection, we derive the instantaneous first-order optimality condition of the variance branch under heteroscedastic Negative Log-Likelihood (NLL) optimization, proving Equation equation 4 from Section 3.1.

Let the target variable be $Y \in \mathbb{R}^+$, and denote the logarithmic ground truth as $Z \triangleq \ln Y$. Under the assumption that the conditional distribution of Z given input $\mathbf{X}$ follows a Gaussian distribution $\mathcal{N}(Z^*(\mathbf{X}), \sigma^2_{\text{aleatoric}}(\mathbf{X}))$, the true conditional mean and irreducible aleatoric variance in log space are formally defined as:

$$Z^*(\mathbf{X}) \triangleq \mathbb{E}[Z \mid \mathbf{X}] = \mathbb{E}[\ln Y \mid \mathbf{X}], \tag{30}$$

$$\sigma^2_{\text{aleatoric}}(\mathbf{X}) \triangleq \text{Var}(Z \mid \mathbf{X}) = \mathbb{E}\left[(Z - Z^*(\mathbf{X}))^2 \mid \mathbf{X}\right]. \tag{31}$$

The model parameterized by weights θ outputs a predictive mean $\mu(\mathbf{X})$ and a predictive variance $\sigma^2(\mathbf{X})$ via two respective output heads. The heteroscedastic Gaussian Negative Log-Likelihood loss is defined as:

$$\mathcal{L}_{\text{NLL}}(\mu(\mathbf{X}), \sigma^2(\mathbf{X}); Z) = \frac{(Z - \mu(\mathbf{X}))^2}{2\sigma^2(\mathbf{X})} + \frac{1}{2}\ln \sigma^2(\mathbf{X}) + \frac{1}{2}\ln(2\pi). \tag{32}$$

Omitting the constant term $\frac{1}{2}\ln(2\pi)$, the conditional risk (expected loss) given input $\mathbf{X}$ at any specific training iteration step t is:

$$\mathcal{R}\left(\sigma^2(\mathbf{X}) \mid \mu_t(\mathbf{X}), \mathbf{X}\right) \triangleq \mathbb{E}_{Z|\mathbf{X}}\left[\frac{(Z - \mu_t(\mathbf{X}))^2}{2\sigma^2(\mathbf{X})} + \frac{1}{2}\ln \sigma^2(\mathbf{X})\right], \tag{33}$$

where $\mu_t(\mathbf{X})$ denotes the instantaneous state of the mean prediction head at training step t.

To analyze the optimal variance tracked by the network at step t, we decompose the numerator of the expectation term in Equation equation 33. Adding and subtracting the true conditional expectation $Z^*(\mathbf{X})$:

$$Z - \mu_t(\mathbf{X}) = (Z - Z^*(\mathbf{X})) + (Z^*(\mathbf{X}) - \mu_t(\mathbf{X})). \tag{34}$$

Taking the conditional expectation of the squared term:

$$\begin{aligned}\mathbb{E}_{Z|\mathbf{X}}\left[(Z-\mu_t(\mathbf{X}))^2\right] &= \mathbb{E}_{Z|\mathbf{X}}\left[((Z-Z^*(\mathbf{X}))+(Z^*(\mathbf{X})-\mu_t(\mathbf{X})))^2\right] \\ &= \mathbb{E}_{Z|\mathbf{X}}\left[(Z-Z^*(\mathbf{X}))^2\right] \\ &\quad + 2\left(Z^*(\mathbf{X})-\mu_t(\mathbf{X})\right)\underbrace{\mathbb{E}_{Z|\mathbf{X}}\left[Z-Z^*(\mathbf{X})\right]}_{=0 \text{ by Eq. equation 30}} \\ &\quad + \left(Z^*(\mathbf{X})-\mu_t(\mathbf{X})\right)^2 \\ &= \sigma^2_{\text{aleatoric}}(\mathbf{X}) + \left(Z^*(\mathbf{X})-\mu_t(\mathbf{X})\right)^2. \end{aligned} \tag{35}$$

Let $v \triangleq \sigma^2(\mathbf{X}) \in \mathbb{R}^+$ represent the variance variable. Substituting Equation equation 35 back into the conditional risk in Equation equation 33:

$$\mathcal{R}(v) = \frac{\sigma^2_{\text{aleatoric}}(\mathbf{X}) + \left(Z^*(\mathbf{X})-\mu_t(\mathbf{X})\right)^2}{2v} + \frac{1}{2}\ln v. \tag{36}$$

To find the stationary point of the expected risk with respect to the predictive variance v, we take the first-order partial derivative $\frac{\partial \mathcal{R}}{\partial v}$:

$$\frac{\partial \mathcal{R}}{\partial v} = -\frac{\sigma^2_{\text{aleatoric}}(\mathbf{X}) + \left(Z^*(\mathbf{X})-\mu_t(\mathbf{X})\right)^2}{2v^2} + \frac{1}{2v}. \tag{37}$$

Setting the derivative to zero to satisfy the first-order necessary condition for optimality:

$$\frac{\partial \mathcal{R}}{\partial v} = 0 \quad \Longleftrightarrow \quad \frac{1}{2v}\left(1 - \frac{\sigma^2_{\text{aleatoric}}(\mathbf{X}) + \left(Z^*(\mathbf{X})-\mu_t(\mathbf{X})\right)^2}{v}\right) = 0. \tag{38}$$

Since $v > 0$, the unique stationary point $v_t^*(\mathbf{X}) = \sigma_t^{*2}(\mathbf{X})$ is explicitly given by:

$$\sigma_t^{*2}(\mathbf{X}) = \sigma^2_{\text{aleatoric}}(\mathbf{X}) + \left(Z^*(\mathbf{X})-\mu_t(\mathbf{X})\right)^2. \tag{39}$$

To verify that this stationary point is a strict global minimum, we compute the second-order partial derivative at $v = \sigma_t^{*2}(\mathbf{X})$:

$$\frac{\partial^2 \mathcal{R}}{\partial v^2} = \frac{\sigma^2_{\text{aleatoric}}(\mathbf{X}) + \left(Z^*(\mathbf{X})-\mu_t(\mathbf{X})\right)^2}{v^3} - \frac{1}{2v^2}. \tag{40}$$

Substituting $v = \sigma_t^{*2}(\mathbf{X})$ from Equation equation 39:

$$\left.\frac{\partial^2 \mathcal{R}}{\partial v^2}\right|_{v=\sigma_t^{*2}(\mathbf{X})} = \frac{\sigma_t^{*2}(\mathbf{X})}{(\sigma_t^{*2}(\mathbf{X}))^3} - \frac{1}{2(\sigma_t^{*2}(\mathbf{X}))^2} = \frac{1}{2(\sigma_t^{*2}(\mathbf{X}))^2} > 0. \tag{41}$$

Because the second derivative is strictly positive for all valid variance values ($v > 0$), the stationary point is strictly convex and represents the unique global minimum of the risk landscape.

This concludes the proof that the instantaneous target tracked by the variance head $\sigma^2(\mathbf{X})$ at training step t dynamically encapsulates both the aleatoric noise floor $\sigma^2_{\text{aleatoric}}(\mathbf{X})$ and the transient fitting lag $(Z^*(\mathbf{X}) - \mu_t(\mathbf{X}))^2$, confirming Equation equation 4.

B.4 A Conceptual Rationalization of the TVG Design

In this subsection, we provide an idealized analysis of how the Transient Variance Guidance (TVG) framework is intended to compensate for the metric-warped interpolation bias Δ during training, and why the auxiliary branches may be removed at inference time under the assumption that the learned constraints remain in the backbone.

B.4.1 Compensatory Trajectory in the Logarithmic Manifold

As derived in Appendix B.1, the prediction generated by a standard log-space model along an interpolation path is the geometric mean $\hat{Y} = \exp(\mu(\mathbf{X}))$, whereas the physically consistent linear baseline is the arithmetic expectation $Y^* = \hat{Y} + \Delta$ with $\Delta > 0$.

To approximate this unbiased target Y^* upon exponential retransformation, the preferred prediction In log space is not the unconstrained geometric mean $\mu(\mathbf{X})$, but rather a compensated trajectory $\tilde{\mu}(\mathbf{X})$:

$$\begin{aligned}\tilde{\mu}(\mathbf{X}) \triangleq \ln(Y^*) &= \ln\left(\hat{Y}(\mathbf{X}) + \Delta(\mathbf{X})\right) \\ &= \ln\left(\hat{Y}(\mathbf{X})\left(1 + \frac{\Delta(\mathbf{X})}{\hat{Y}(\mathbf{X})}\right)\right) \\ &= \mu(\mathbf{X}) + \ln\left(1 + \frac{\Delta(\mathbf{X})}{\hat{Y}(\mathbf{X})}\right). \end{aligned} \tag{42}$$

Equation equation 42 indicates that mitigating the metric-warped interpolation bias mathematically corresponds to introducing a localized compensatory term:

$$C^*(\mathbf{X}) \triangleq \ln\left(1 + \frac{\Delta(\mathbf{X})}{\hat{Y}(\mathbf{X})}\right) \approx \frac{\Delta(\mathbf{X})}{\hat{Y}(\mathbf{X})}, \tag{43}$$

into the underlying representation, shifting the prediction manifold from a standard log-linear path to a curvature-corrected trajectory.

B.4.2 Auxiliary Bridging and Gradient Mechanics

Under the TVG framework, we define the composite modulation term as $\gamma(\mathbf{X}) \triangleq \beta(\mathbf{X}) \cdot \sigma^2(\mathbf{X})$. The auxiliary-branch prediction is formulated as:

$$\hat{Y}_{\text{aux}} = \exp\left(\text{sg}[\mu(\mathbf{X})] + \gamma(\mathbf{X})\right). \tag{44}$$

In the forward pass, $\text{sg}[\mu(\mathbf{X})]$ evaluates identically to $\mu(\mathbf{X})$. The auxiliary original-scale supervision $\mathcal{L}_{\text{linear}} = \text{MAE}(\hat{Y}_{\text{aux}}, Y) = |\hat{Y}_{\text{aux}} - Y|$ provides a samplewise corrective signal to the auxiliary pathway. In the idealized deterministic setting underlying the geometric analysis, where $Y = Y^*$, minimizing this loss favors $\hat{Y}_{\text{aux}}$ approaching the additive reference trajectory. Matching the auxiliary-branch prediction to this reference trajectory gives

$$\mu(\mathbf{X}) + \gamma(\mathbf{X}) \approx \ln(Y^*) = \mu(\mathbf{X}) + C^*(\mathbf{X}), \qquad \gamma(\mathbf{X}) \approx C^*(\mathbf{X}). \tag{45}$$

Under stochastic observations, we do not interpret the auxiliary MAE as an estimator of the conditional mean; rather, it supplies samplewise original-scale corrective gradients that reshape the shared representation during training. The stop-gradient operator $\text{sg}[\cdot]$ prevents the linear loss gradients from directly affecting the log-mean prediction head $\mu(\mathbf{X})$ (satisfying $\mathcal{D}_2$). Applying the chain rule, the gradient of the linear loss with respect to the shared backbone parameters θ is derived as:

$$\begin{aligned}\nabla_\theta \mathcal{L}_{\text{linear}} &= \frac{\partial \mathcal{L}_{\text{linear}}}{\partial \hat{Y}_{\text{aux}}} \cdot \frac{\partial \hat{Y}_{\text{aux}}}{\partial \gamma} \cdot \frac{\partial \gamma}{\partial \theta} \\ &= \text{sign}(\hat{Y}_{\text{aux}} - Y) \cdot \hat{Y}_{\text{aux}} \cdot \left[\sigma^2(\mathbf{X}) \frac{\partial \beta(\mathbf{X})}{\partial \theta} + \beta(\mathbf{X}) \frac{\partial \sigma^2(\mathbf{X})}{\partial \theta}\right]. \end{aligned} \tag{46}$$

Equation equation 46 illustrates the optimization dynamics of TVG:

1. **Magnitude Scaling:** The term $\hat{Y}_{\text{aux}}$ acts as a scale modulator, ensuring that regions with larger target magnitudes receive correspondingly scaled corrective gradients.
2. **Variance-Guided Gradient Scaling:** The gradient backpropagated through the modulation head β is explicitly scaled by the transient variance $\sigma^2(\mathbf{X})$. In early training iterations, when the predictive mean lags behind local data transitions, the inflated $\sigma^2(\mathbf{X})$ (as derived in Appendix B.3) allows a larger gradient flow to update the shared backbone parameters.

B.4.3 Dynamic Internalization and Inference Removal

The auxiliary branch in TVG functions as a dynamic training constraint rather than a static additive component.

As the unidirectional gradient flow $\nabla_\theta \mathcal{L}_{\text{linear}}$ updates the shared backbone $f_\theta(\mathbf{X})$, it adjusts the latent representation $\mathbf{h}(\mathbf{X}) = f_\theta(\mathbf{X})$. Under this adjusted representation, linear transitions in the latent space incorporate the curvature required to approximate arithmetic additivity in the original space. Because the log-mean head $\mu(\mathbf{X}) = \mathbf{w}_\mu^\top \mathbf{h}(\mathbf{X})$ operates directly on this representation, its forward trajectory $\mu_t(\mathbf{X})$ is guided toward the compensated path $\tilde{\mu}(\mathbf{X})$.

Consequently, as training progresses toward convergence, the shared backbone gradually internalizes the linear-scale correction, so the compensated trajectory is absorbed into the primary log-mean head:

$$\mu_t(\mathbf{X}) \to \tilde{\mu}(\mathbf{X}) \quad \Longrightarrow \quad \Delta_t(\mathbf{X}) \to 0 \quad \Longrightarrow \quad C_t^*(\mathbf{X}) = \ln\left(1 + \frac{\Delta_t(\mathbf{X})}{\hat{Y}_t(\mathbf{X})}\right) \to 0. \tag{47}$$

As the residual correction $C_t^*(\mathbf{X})$ vanishes, the auxiliary modulation head no longer needs to output a non-zero correction, driving $\beta_t(\mathbf{X}) \to 0$. Hence the composite modulation term decays:

$$\gamma_t(\mathbf{X}) = \beta_t(\mathbf{X})\sigma_t^2(\mathbf{X}) \to 0 \quad \Longrightarrow \quad |\hat{Y}_{\text{aux}} - \exp(\mu(\mathbf{X}))| \to 0. \tag{48}$$

Here $\sigma_t^2(\mathbf{X})$ settles toward the aleatoric floor $\sigma^2_{\text{aleatoric}}(\mathbf{X})$, which is generally non-zero, while the product γ_t still vanishes because $\beta_t \to 0$. This is precisely why the auxiliary branch can be safely removed at inference.

B.5 Why Log-Space Affine Interpolation Is a Plausible Default

In Section 2, the analytical derivation of the metric-warped interpolation bias rests on the premise that deep neural networks predominantly construct affine (or near-affine) transition trajectories, $Z_\alpha = (1-\alpha)Z_1 + \alpha Z_2$, between adjacent conditioning states in the log space.

A natural theoretical inquiry is whether an overparameterized deep neural network—by virtue of the Universal Approximation Theorem—could spontaneously synthesize an intrinsically curved logarithmic trajectory:

$$\tilde{\mu}_\alpha = \ln\left((1-\alpha)Y_1^* + \alpha Y_2^*\right) = \ln\left((1-\alpha)\exp(Z_1) + \alpha\exp(Z_2)\right), \tag{49}$$

which would, upon exponential inversion, trivially match the linear expectation Y^* without requiring any auxiliary guidance.

The relevant question, however, is not one of representational capacity but of the inductive bias induced by finite-sample gradient-based optimization. While such nonlinear compensation is certainly contained within the infinite-capacity hypothesis class, standard training regimes strongly disfavor the realization of Equation equation 49. Below, we formalize why the affine interpolation path Z_α is the structurally favored solution through three foundational pillars of modern deep learning theory.

1. **Spline Theory and Minimal Variation in Functional Banach Spaces:** For single-hidden-layer infinite-width ReLU networks with bounded Euclidean parameter norm, Savarese et al. (2019) established that the minimal network norm required to represent a function $f : \mathbb{R} \to \mathbb{R}$ is bounded by $\max(\int |f''(x)|dx, |f'(-\infty) + f'(\infty)|)$, and consequently, the minimal-norm interpolant of a finite sample is exactly given by **linear spline interpolation**. This result, extended within the functional Banach space framework by Parhi & Nowak (2021; 2022), implies that under appropriate norm-based regularization, the optimal variational interpolants between scattered observation anchors are continuous piecewise linear functions. In locally unpopulated or data-sparse interpolation intervals between distinct sample clusters $\mathbf{X}_1$ and $\mathbf{X}_2$, the absence of intermediate training points means that introducing nonzero curvature does not reduce empirical risk but strictly increases the variational cost. Consequently, within this framework, the minimum-norm interpolant is the zero-second-derivative affine geodesic:

$$\frac{\partial^2 Z_\alpha}{\partial \alpha^2} = 0 \quad \Longrightarrow \quad Z_\alpha = (1-\alpha)Z_1 + \alpha Z_2. \tag{50}$$

2. **Spectral Bias and the Frequency Principle:** In the Neural Tangent Kernel (NTK) regime of infinite-width networks, or more generally in standard multilayer perceptron training with gradient descent, optimization exhibits a spectral hierarchy governed by the NTK eigenspectrum (Jacot et al., 2018; Rahaman et al., 2019; Xu et al., 2020; Basri et al., 2019). The convergence rate of each frequency component during training is linked to the corresponding eigenvalue of the NTK matrix: lower-frequency components correspond to larger eigenvalues, leading to exponentially faster convergence (Basri et al., 2019). Synthesizing the specific target curve $\tilde{\mu}_\alpha$ in Equation equation 49 requires the network to represent a localized, non-trivial harmonic component strictly aligned with the arbitrary boundary anchors (Z_1, Z_2). Under the Frequency Principle (Xu et al., 2020), gradient descent prioritizes the lowest-order harmonic mode connecting the boundary values, which corresponds identically to a flat linear transition.
3. **Implicit Regularization and Low-Curvature Preference:** Extensive literature on the implicit bias of gradient descent demonstrates that overparameterized models naturally gravitate toward flat minima. Barrett & Dherin (2021) showed via backward error analysis that the discrete steps of gradient descent implicitly regularize models by penalizing trajectories with large loss gradients, thereby biasing optimization toward flatter regions of the loss surface. More importantly, the implications of parameter-space flatness for the induced function in function space have been rigorously studied: Mulayoff et al. (2021) showed that SGD in univariate ReLU networks is biased toward functions whose second derivative with respect to the input has a bounded weighted ℓ_1 norm (encouraging smoothness), and showed that SGD regularizes Sobolev seminorms of the function. The ideal log-domain trajectory $\tilde{\mu}_\alpha$ in Equation equation 49 exhibits strict local concavity:

$$\frac{\partial^2 \tilde{\mu}_\alpha}{\partial \alpha^2} = -\frac{(Y_2^* - Y_1^*)^2}{((1-\alpha)Y_1^* + \alpha Y_2^*)^2} = -\left(\frac{\exp(\delta) - 1}{(1-\alpha) + \alpha \exp(\delta)}\right)^2 < 0, \quad \forall \delta \neq 0. \tag{51}$$

Because maintaining nonzero curvature in the prediction function incurs an implicit regularization penalty under these gradient mechanics (Barrett & Dherin, 2021; Bartlett et al., 2021), an unguided network tends to flatten the effective second-order derivative toward zero, collapsing the prediction back onto the biased geometric-mean path $\hat{Y}_\alpha = \exp(Z_\alpha)$.

The Compounding Curse of Tail Sparsity in Skewed Regression. This inductive preference for affine transitions is especially detrimental in continuous regression tasks involving heavy-tailed, zero-inflated, or skewed distributions (e.g., radar reflectivity, rainfall precipitation, or traffic peaks). In such distributions, the density of observations decays rapidly toward the tail, creating vast interpolation voids across the feature space. Because the logarithmic jump $\delta = Z_2 - Z_1$ across these sparse gaps is large, the resulting metric-warped distortion energy scales quadratically as $\mathcal{O}(\delta^2)$ (Equation equation 2).

Hence, absent an external inductive mechanism, a standard deep network under conventional finite-sample training is strongly biased by its own optimization dynamics against self-correcting this curvature deficit. This motivates the design of our TVG framework, which aims to counteract this default preference by leveraging early transient variance to inject linear geometric constraints into the representation during training.

Scope and Qualifications. The above argument characterizes the inductive bias of standard finite-sample gradient training rather than an absolute limitation on representational capacity. If training data are sufficiently dense, or if explicit high-frequency encodings and specially designed losses are employed, a network can indeed converge to the curved trajectory in Equation equation 49. Furthermore, while the precise mapping between parameter-space flatness and function-space curvature remains an active area of investigation (Blanc et al., 2020), the prevailing consensus strongly aligns with a penalty against high-curvature transitions. Therefore, the necessity of the TVG framework arises fundamentally from overcoming the default optimization preference in sparse, heavy-tailed regimes.

C Reproducibility and Implementation Details

In this section, we provide the implementation details, network configurations, and the numerical stabilization mechanisms required to ensure the full reproducibility of the TVG framework.

C.1 Network Architecture and Multi-Head Parameterization

For the primary spatiotemporal radar benchmark (CIKM2017), the shared backbone comprises a 4-layer residual convolutional encoder ($32 \to 64 \to 128 \to 64$ channels with stride-2 downsampling), followed by a 2-layer recurrent ConvLSTM (3×3 kernels, hidden dimension $d = 64$).

The recurrent hidden representation of the final time step is spatially flattened into a shared latent vector $\mathbf{h}(\mathbf{X})$. A single shared linear projection layer $\mathbf{W}_{\text{heads}}$ maps $\mathbf{h}(\mathbf{X})$ simultaneously into three parallel prediction heads:

$$[\mu_{\text{logit}}, s_{\text{logit}}, \beta_{\text{logit}}]^\top = \mathbf{W}_{\text{heads}}\mathbf{h}(\mathbf{X}) + \mathbf{b}_{\text{heads}}. \tag{52}$$

To enforce physical validity, numerical stability, and smooth optimization dynamics, domain-specific non-linear activations are applied to each head:

1. **Log-Mean Head (μ):** Constrained to be strictly non-negative via the Softplus activation:
$$\mu(\mathbf{X}) = \text{Softplus}(\mu_{\text{logit}}) = \ln\left(1 + \exp(\mu_{\text{logit}})\right). \tag{53}$$
2. **Transient Variance Head (σ^2):** Supervised in the log-variance domain $s \triangleq \ln \sigma^2$. To prevent variance collapse and numerical explosion, we bound the predictive log-variance from above using an inverted softplus with an upper ceiling $L_{\text{var}} = 1.5$:
$$\ln \sigma^2(\mathbf{X}) = L_{\text{var}} - \text{Softplus}\left(L_{\text{var}} - s_{\text{logit}}\right). \tag{54}$$
During NLL computation, a lower bound is enforced via clamping ($\ln \sigma^2 \geq -6.0$). The variance used in the auxiliary modulation branch is explicitly exponentiated via $\sigma^2(\mathbf{X}) = \exp\left(\ln \sigma^2(\mathbf{X})\right)$.
3. **Adaptive Modulation Head (β):** Modulated via a scaled hyperbolic tangent activation bounded within $[-B_{\text{limit}}, B_{\text{limit}}]$ with $B_{\text{limit}} = 1.0$:
$$\beta(\mathbf{X}) = B_{\text{limit}} \cdot \tanh(\beta_{\text{logit}}). \tag{55}$$
The linear bias corresponding to β_{logit} is initialized to 0, ensuring that training begins at a zero-curvature prior while maintaining smooth gradient flow across both positive and negative correction regimes.

C.2 Training Hyperparameters and Numerical Safeguards

Optimization Configuration. Unless otherwise stated, all models are trained end-to-end using the Adam optimizer with an initial learning rate of $\eta = 1 \times 10^{-4}$ and a mini-batch size of 32 for 200 epochs. Learning rate scheduling is managed via `ReduceLROnPlateau` (monitoring validation MAE with a decay factor of 0.8, patience of 25 epochs, cooldown of 5 epochs, and minimum learning rate of 1×10^{-6}). For the primary CIKM2017 benchmark, the auxiliary loss weight in Equation equation 6 is set to $\lambda = 0.5$. The default CIKM2017 configuration is summarized in Table 10.

Dataset-Specific Auxiliary Loss Weight. The appropriate magnitude of the auxiliary loss weight λ is dataset-dependent because the auxiliary objective is imposed after inverse transformation in the original target space. In particular, the gradient backpropagated through the `expm1` mapping contains a multiplicative factor proportional to the original-scale prediction magnitude. Consequently, datasets with substantially larger or more dispersed target values generally require a smaller λ to maintain a comparable strength of auxiliary intervention.

Empirically, an excessively small λ makes the auxiliary guidance too weak to meaningfully reshape the shared representation, whereas an excessively large λ causes the auxiliary objective to dominate optimization and prevents the discrepancy between the primary-branch and auxiliary-branch predictions from vanishing near convergence. We therefore use the late-stage reconciliation of the two predictions as a stability diagnostic to determine the appropriate order of magnitude of λ, and select the final value through hyperparameter search within the resulting stable regime. The values used in all datasets are reported in Table 11.

Numerical Safeguards Against Exponential Overflow. Because the target distributions often contain zero-inflated values, we employ the $Z = \ln(Y + 1)$ transformation for data preprocessing.

Table 10: Default hyperparameter settings and numerical safeguards for the primary CIKM2017 benchmark.

Category	Hyperparameter	Value
Architecture	ConvLSTM Hidden Dimension	64
	ConvLSTM Kernel Size	3×3
	Modulation Bound B_{limit}	1.0
	Variance Upper Limit L_{var}	1.5
Optimization	Optimizer	Adam ($\beta_1 = 0.9, \beta_2 = 0.999$)
	Base Learning Rate	1×10^{-4}
	Batch Size	32
	Epoch Budget	200
	Auxiliary Loss Weight λ	0.5
Numerical Safeguards	Gradient Clipping Norm	1.0
	Training Exponent Clamp	$\max = 10.0$

Table 11: Dataset-specific auxiliary loss weights used in TVG.

Dataset	Auxiliary Loss Weight λ
CIKM2017	0.5
CAMELS	0.5
PM2.5	0.1
Rossmann	0.005
Webtraffic	0.0005

Consequently, the auxiliary reconstruction branch optimizes the original-scale target via the inverse mapping (`expm1`), defined practically as $\hat{Y}_{\text{aux}} = \text{expm1}(\text{sg}[\mu] + \beta\sigma^2)$. Direct backpropagation through this operation risks generating extreme original-scale gradients and floating-point overflow (NaN/Inf). To guarantee stability, we implement three mandatory engineering safeguards:

1. **Norm-Based Gradient Clipping:** Parameter gradients are strictly clipped to a maximum global Euclidean norm of 1.0 before each optimizer step:
$$\mathbf{g} \leftarrow \mathbf{g} \cdot \min\left(1, \frac{1.0}{\|\mathbf{g}\|_2 + 10^{-6}}\right). \tag{56}$$
This is essential because the backward derivative $\frac{\partial \hat{Y}_{\text{aux}}}{\partial \gamma} = \hat{Y}_{\text{aux}}$ scales proportionally with target magnitude, which could otherwise destabilize updates during early epochs.
2. **Exponent Clamping:** The input to the exponential retransformation is safeguarded via upper clamping:
$$\hat{Y}_{\text{aux}} = \text{expm1}\left(\min\left(\text{sg}[\mu(\mathbf{X})] + \beta(\mathbf{X})\sigma^2(\mathbf{X}),\ M_{\text{train}}\right)\right), \tag{57}$$
where $M_{\text{train}} = 10.0$ is applied only to the auxiliary reconstruction branch during training. The final prediction is not clamped. This prevents the auxiliary exponential term from generating excessively large gradients during early training.
3. **Unidirectional Gradient Detachment:** Consistent with Equation equation 5, the primary log-mean head μ is detached prior to the auxiliary exponentiation (`mu.detach()`). This completely shields the NLL optimization from linear gradient shocks, ensuring that only the auxiliary modulation pathways (β, σ^2) transmit linear geometric constraints back to the backbone.

Target transformation and inverse mapping across datasets. All five datasets are non-negative and contain zero-inflated or near-zero values. We therefore apply the shifted logarithmic transformation
$$Z = \ln(1 + Y) \tag{58}$$

to the target before training, and use the corresponding inverse mapping

$$\hat{Y} = \text{expm1}(\hat{Z}) = \exp(\hat{Z}) - 1 \tag{59}$$

during both auxiliary reconstruction and final evaluation. This convention is applied consistently across CIKM2017, Webtraffic, PM2.5, CAMELS, and Rossmann. The theoretical analysis in Section 2 is carried out on the transformed variable $Z = \ln(1 + Y)$, which is equivalent to applying the standard log-transformation to the shifted target $Y' = Y + 1$. Since the shift is a constant translation, the arithmetic-versus-geometric interpolation discrepancy retains the same form on Y', and the derived bias expression applies to the shifted target without modification.

D Connection and Contrast with Classical Statistics

A foundational question regarding log-transformed regression is how the proposed metric-warped interpolation bias relates to the long-standing problem of log-retransformation bias studied in classical statistics and econometrics (Finney, 1941; Duan, 1983). In this section, we distinguish the two problems along both their mechanism and statistical objective. Traditional methods address a pointwise expectation shift and typically target recovery of the conditional mean, whereas our work concerns a deterministic inter-manifold geometric distortion induced during generalization. Crucially, the two error modes remain distinct even in the noise-free limit, where classical retransformation corrections vanish while metric-warped interpolation bias remains active.

D.1 Classical Formulation: PointWise Expectation Shift and Smearing

In classical linear and generalized regression, log-transformation is typically introduced to model multiplicative data-generating processes or to stabilize residual variance. The canonical observational model is formulated as:

$$\ln Y = Z = \mu(\mathbf{X}) + \epsilon, \quad \text{with} \quad \mathbb{E}[\epsilon \mid \mathbf{X}] = 0, \tag{60}$$

where $\mu(\mathbf{X}) \triangleq \mathbb{E}[\ln Y \mid \mathbf{X}]$ denotes the true conditional mean In log space, and ϵ represents stochastic aleatoric noise with conditional variance $\text{Var}(\epsilon \mid \mathbf{X}) = \sigma^2_{\text{aleatoric}}(\mathbf{X})$.

When recovering the target Y in the original liner space, a naive exponential inverse mapping yields $\hat{Y}_{\text{naive}} = \exp(\mu(\mathbf{X}))$. However, by virtue of Jensen's inequality applied to the strictly convex exponential function over the random variable ϵ, the true conditional expectation in the linear domain systematically deviates from this naive prediction:

$$\begin{aligned}
\mathbb{E}[Y \mid \mathbf{X}] &= \mathbb{E}[\exp(Z) \mid \mathbf{X}] \\
&= \mathbb{E}[\exp(\mu(\mathbf{X}) + \epsilon) \mid \mathbf{X}] \\
&= \exp(\mu(\mathbf{X})) \cdot \mathbb{E}[\exp(\epsilon) \mid \mathbf{X}] \\
&\geq \exp(\mu(\mathbf{X})) \cdot \exp(\mathbb{E}[\epsilon \mid \mathbf{X}]) = \exp(\mu(\mathbf{X})) = \hat{Y}_{\text{naive}}.
\end{aligned} \tag{61}$$

To correct this pointwise expectation underestimation induced by the dispersion of ϵ, classical statistics introduces scalar correction factors:

1. **Parametric Correction (Finney, 1941):** Under the assumption of homoscedastic Gaussian residuals $\epsilon \sim \mathcal{N}(0, \sigma^2)$, the moment-generating function yields $\mathbb{E}[\exp(\epsilon)] = \exp\left(\frac{1}{2}\sigma^2\right)$, leading to the parametric corrected prediction:

$$\hat{Y}_{\text{param}} = \exp(\mu(\mathbf{X})) \cdot \exp\left(\frac{1}{2}\sigma^2\right). \tag{62}$$

2. **Non-Parametric Smearing Estimator (Duan, 1983):** When the error distribution is non-Gaussian or unknown, Duan (1983) proposed the empirical smearing factor $S \triangleq \frac{1}{N}\sum_{i=1}^{N} \exp(e_i)$, where $e_i = \ln Y_i - \hat{\mu}(\mathbf{X}_i)$ are sample residuals:

$$\hat{Y}_{\text{smearing}} = \exp(\mu(\mathbf{X})) \cdot S = \exp(\mu(\mathbf{X})) \cdot \left[\frac{1}{N}\sum_{i=1}^{N} \exp(e_i)\right]. \tag{63}$$

Both parametric and smearing corrections operate as post-hoc scalar recalibrations. Their core objective is strictly statistical: neutralizing the Jensen gap created by the dispersion of the random noise ϵ around a single observation anchor $\mathbf{X}$.

D.2 Retransformation Correction Is Estimand-Dependent

Importantly, retransformation "bias" is defined relative to a particular statistical estimand and should not be interpreted as a task-independent notion of predictive error. Most classical retransformation corrections aim to recover the conditional mean $\mathbb{E}[Y \mid \mathbf{X}]$. This target is optimal under squared-error risk:

$$\arg\min_a \mathbb{E}\big[(Y-a)^2 \mid \mathbf{X}\big] = \mathbb{E}[Y \mid \mathbf{X}]. \tag{64}$$

Under absolute-error risk, however, the optimal point prediction is instead a conditional median:

$$\arg\min_a \mathbb{E}[|Y-a| \mid \mathbf{X}] \in \mathrm{Med}(Y \mid \mathbf{X}). \tag{65}$$

For skewed conditional distributions, the mean and median can differ substantially. Consequently, a correction that successfully moves a prediction toward the conditional mean may improve squared-error metrics while leaving absolute error unchanged or even worsening it. Thus, mean-unbiased retransformation is statistically meaningful, but its desirability depends on the downstream prediction objective.

This distinction is orthogonal to the phenomenon studied in this work. Metric-warped interpolation bias does not arise from choosing an incorrect pointwise estimand, but from how a finite-sample neural predictor connects different conditioning states under the transformed metric. Pointwise retransformation correction constrains what statistic should be predicted at a given $\mathbf{X}$; it does not by itself constrain the geometry by which the predictor generalizes between sparsely observed states. The practical impact of either effect remains task-dependent, but they act on different axes of the prediction problem.

D.3 Deep Regression Formulation: Manifold-Induced Interpolation Distortion

In contrast to classical statistics, deep regression models are overparameterized continuous function approximators trained via gradient descent. Under the log-preprocessing paradigm, deep networks do not treat samples as disjoint points; instead, governed by their spectral bias and smooth inductive bias (Jacot et al., 2018; Rahaman et al., 2019), they construct a continuous, low-complexity representation manifold connecting neighboring training observations.

Consider two neighboring local conditioning states $\mathbf{X}_1$ and $\mathbf{X}_2$ with respective deterministic linear conditional expectations $Y_1^* = \mathbb{E}[Y \mid \mathbf{X}_1]$ and $Y_2^* = \mathbb{E}[Y \mid \mathbf{X}_2]$. Along a continuous transition path $\mathbf{X}_\alpha$ parameterized by $\alpha \in [0,1]$, the network naturally learns a linear interpolation in log space:

$$Z_\alpha = (1-\alpha)\ln Y_1^* + \alpha \ln Y_2^*. \tag{66}$$

Mapping back to the original space yields the geometric mean path $\hat{Y}_\alpha = (Y_1^*)^{1-\alpha}(Y_2^*)^\alpha$. However, if the underlying physical system satisfies local additive superposition, the reference trajectory implied by local additivity is the arithmetic path $Y_\alpha^* = (1-\alpha)Y_1^* + \alpha Y_2^*$.

As derived in Equation equation 2, the resulting discrepancy is:

$$\Delta(\alpha) = Y_\alpha^* - \hat{Y}_\alpha \approx \frac{1}{2} Y_1^* \alpha(1-\alpha)\delta^2. \tag{67}$$

Notice the structural difference: Equation equation 67 does not contain any random error variable ϵ or variance σ^2. It is entirely a path-dependent, geometric deformation between the logarithmic coordinate system and the original space .

D.4 Orthogonality and the Noise-Free Limit ($\epsilon \to 0$)

To formally establish that metric-warped interpolation bias is fundamentally distinct from classical retransformation bias, we investigate the asymptotic behavior of both formulations in the noise-free deterministic limit.

Proposition 2 (Decoupling of Smearing and Interpolation Bias in Deterministic Systems)**.** *Suppose a continuous physical system is completely deterministic, such that the aleatoric noise vanishes identically:* $\mathrm{Var}(\epsilon \mid \mathbf{X}) \to 0$*, implying* $\epsilon \equiv 0$ *almost surely. Then:*

Table 12: Conceptual and mathematical comparison between classical pointwise retransformation bias and metric-warped interpolation bias.

Dimension	Classical Statistics (e.g., Duan's Smearing)	Metric-Warped Interpolation Bias (Ours)
Error Origin	Jensen's inequality over random noise ϵ	Jensen's inequality over manifold paths Z_α
Underlying Mechanism	Aleatoric noise dispersion: $\mathbb{E}[\exp(\epsilon)] > 1$	Smooth inductive bias: $\exp(\mathbb{E}_\alpha[Z]) < \mathbb{E}_\alpha[\exp(Z)]$
Spatial Nature	Pointwise, isolated at individual $\mathbf{X}$	Inter-manifold, path-dependent between $(\mathbf{X}_1, \mathbf{X}_2)$
Noise Dependence	Strictly requires non-zero aleatoric noise ($\sigma^2 > 0$)	Persists even in noise-free systems ($\sigma^2 = 0$)
Correction Method	Post-hoc scalar multiplier (e.g., smearing factor S)	In-training representation manifold reshaping via TVG
Inference Behavior	Modifies predictions via scalar multiplication	Prunes auxiliary branches; zero inference overhead
Statistical Target	Conditional mean $\mathbb{E}[Y \mid \mathbf{X}]$	Original-space interpolation geometry

1. *Classical retransformation bias vanishes, and both parametric and non-parametric smearing factors collapse to unity:*

$$\lim_{\sigma^2 \to 0} \hat{Y}_{param} = \lim_{e_i \to 0} \hat{Y}_{smearing} = \exp(\mu(\mathbf{X})) = \hat{Y}_{naive}. \tag{68}$$

2. *Conversely, the metric-warped interpolation bias $\Delta(\alpha)$ remains strictly positive and entirely unaffected for all non-trivial transitions ($Y_1^* \neq Y_2^*$) and interior interpolation states $\alpha \in (0, 1)$:*

$$\lim_{\sigma^2 \to 0} \Delta(\alpha) = Y_\alpha^* - \hat{Y}_\alpha > 0. \tag{69}$$

Proof. In a deterministic system where $Y = \exp(Z^*) = Y^*$, the aleatoric residual satisfies $\epsilon = \ln Y - \ln Y^* = 0$ everywhere. For classical parametric correction (Equation equation 62), substituting $\sigma^2 = 0$ yields:

$$\hat{Y}_{\text{param}} = \exp(\mu(\mathbf{X})) \cdot \exp(0) = \exp(\mu(\mathbf{X})). \tag{70}$$

Similarly, for Duan's smearing factor (Equation equation 63), because every sample residual $e_i = 0$:

$$S = \frac{1}{N} \sum_{i=1}^{N} \exp(0) = 1 \quad \Longrightarrow \quad \hat{Y}_{\text{smearing}} = \exp(\mu(\mathbf{X})) \cdot 1 = \exp(\mu(\mathbf{X})). \tag{71}$$

Thus, classical statistical corrections exert zero effect ($S = 1$) in deterministic environments.

However, the metric-warped interpolation bias depends exclusively on the non-zero logarithmic span $\delta = \ln Y_2^* - \ln Y_1^*$ between distinct physical states. By Proposition 1, the strict convexity of $\exp(\cdot)$ guarantees that for any $Y_1^* \neq Y_2^*$ and $\alpha \in (0, 1)$:

$$\Delta(\alpha) = (1 - \alpha)Y_1^* + \alpha Y_2^* - (Y_1^*)^{1-\alpha}(Y_2^*)^\alpha > 0. \tag{72}$$

Because $\Delta(\alpha)$ is independent of $\sigma^2_{\text{aleatoric}}$, taking the limit $\sigma^2 \to 0$ leaves $\Delta(\alpha)$ strictly positive and unchanged. □