WHEN AND WHY ADVERSARIAL TRAINING IMPROVES TIME-SERIES FORECASTING

Anonymous authors
Paper under double-blind review

ABSTRACT

While adversarial training exhibits contrasting clean-accuracy effects across vision and language, whether and why it benefits unperturbed data in time-series forecasting remains an open question. In this work, we uncover an unexpected within-domain clean-effect reversal: standard raw-history adversarial training (AT) reduces unperturbed test MSE by up to 6.9% for expressive nonlinear forecasters, yet systematically degrades compact linear models. Through exact finite-network risk identities and asymptotic expansions, we trace this divergence to an intrinsic competition between prediction-variance contraction and parameter displacement: while loss-guided updates act as a frequency-selective regularizer that dampens low-frequency prediction variance, fitting perturbed histories pulls parameters away from the clean empirical minimum, incurring a displacement penalty that can outweigh variance gains in low-capacity architectures. To govern this trade-off, we propose Displacement-Guided Adversarial Training (DGA). DGA introduces an elastic parameter anchor to a pre-trained empirical risk minimization (ERM) checkpoint, effectively suppressing weakly constrained parameter drift while preserving beneficial variance regularization. Across 56 forecasting configurations spanning eight architectures and seven benchmarks, validation-selected DGA outperforms paired ERM in 51 settings ($p = 1.4 \times 10^{-5}$), turning linear-model degradations into consistent clean gains (e.g., -3.70% MSE on DLinear). Furthermore, DGA transfers effectively to language fine-tuning and vision classification, establishing parameter displacement control as a general stabilizer for adversarial optimization.

1 INTRODUCTION

Can training against adversarial perturbations improve model performance on ordinary, unperturbed data? In deep learning, this question has long exposed a **sharp cross-modal divide**. In computer vision, standard pixel-space adversarial training (Goodfellow et al., 2015; Madry et al., 2018) reliably induces a well-documented trade-off between robustness and clean generalization (Tsipras et al., 2019; Zhang et al., 2019). Conversely, in natural language processing, perturbing word representations serves as an effective regularizer that consistently boosts clean benchmark accuracy (Miyato et al., 2017; Zhu et al., 2020; Jiang et al., 2020). However, because these two domains differ along multiple confounding axes simultaneously—discrete symbolic tokens versus continuous pixels, raw input-space versus representation-space perturbations, and classification margins versus semantic embeddings—isolating whether this clean-effect discrepancy originates from data modality, perturbation space, or model capacity has proven remarkably difficult.

Time-series forecasting offers a uniquely controlled testbed to examine these factors. Like computer vision, forecasting operates directly on continuous raw inputs, allowing input-space perturbations without relying on latent representation-level proxies. Unlike vision classification governed by complex decision boundaries, forecasting is a continuous-to-continuous regression mapping where errors can be evaluated under transparent squared-error objectives. Crucially, this setting allows us to hold the forecasting task, the raw-input perturbation protocol, and the data distribution fixed, while systematically evaluating architectures that span the full spectrum of capacity—from compact linear predictors (Zeng et al., 2023; Xu et al., 2024) to expressive deep attention and convolution networks (Liu et al., 2024; Nie et al., 2023; Wang et al., 2026; Ma et al., 2025).

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.

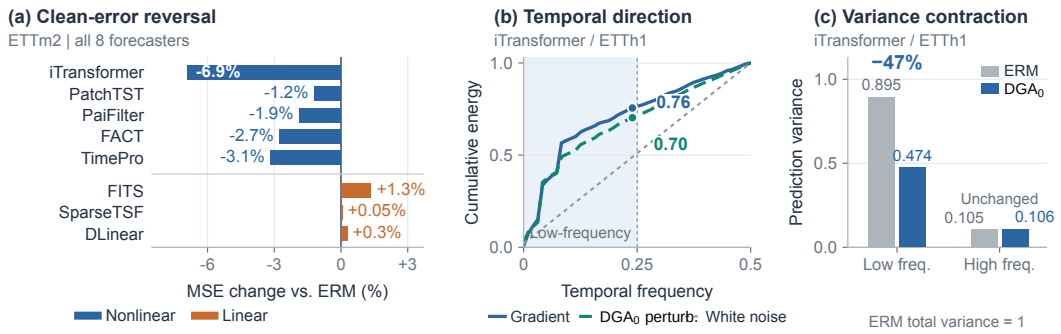

Figure 1: **Dynamics of adversarial training (AT) in time series.** **(a)** Within-domain reversal: AT lowers clean MSE for expressive models (up to 6.9%) but degrades linear baselines. **(b)** Gradients and perturbations concentrate in slow temporal variations. **(c)** Prediction variance contracts selectively in the low-frequency band.

Within this controlled setting, we uncover an unexpected **within-domain clean-effect reversal**: applying standard raw-history adversarial training (AT) reduces clean test MSE by up to 6.9% for expressive nonlinear forecasters, yet systematically degrades compact linear baselines (Figure 1(a)). This empirical divergence demonstrates that the clean generalization effect of adversarial training is not an immutable property of data modality or perturbation domain, but is fundamentally governed by an intrinsic interaction between perturbation geometry and model capacity. Investigating standard adversarial training (AT), we find that loss gradients and perturbations naturally concentrate in slow temporal variations, carrying over 70% of their spectral energy in low frequencies (Figure 1(b)), where phase-matched controls confirm that this error-sensitive gradient direction—rather than generic low-frequency smoothing—drives the gain. Across training runs, AT acts as a frequency-selective regularizer, contracting low-frequency prediction variance by up to 50% (Figure 1(c)). However, fitting perturbed histories simultaneously pulls parameters away from the clean empirical risk minimization (ERM) solution, incurring an ensemble-mean displacement penalty. In expressive models, variance reduction outweighs this displacement cost to yield net clean gains; in compact linear models, the displacement penalty dominates, deteriorating clean accuracy. A width intervention on a Transformer ($d_{\text{model}} \in \{4, \ldots, 256\}$) directly reproduces this capacity-driven crossing from harmful to beneficial regimes, confirming that capacity arbitrates this balance.

To resolve this dilemma between clean bias and adversarial variance reduction, we propose *Displacement-Guided Adversarial Training* (DGA). Rather than treating ERM and adversarial training as mutually exclusive regimes, DGA formulates a trajectory-anchored optimization problem:

$$\mathcal{L}_{\text{DGA}}(\theta; \lambda) = (1-\beta)\mathcal{L}(f_\theta(x), y) + \beta\mathcal{L}(f_\theta(x+\delta^*), y) + \lambda\|\theta - \theta_{\text{ERM}}\|^2, \tag{1}$$

where θ_{ERM} is a pre-trained clean checkpoint and δ^* is an inner K-step adversarial history perturbation ($\beta = 0.5$). The penalty $\lambda\|\theta - \theta_{\text{ERM}}\|^2$ establishes an elastic anchor to the clean solution, whose local Hessian geometry preserves directions strongly constrained by the adversarial loss while suppressing weakly constrained drift along flatter directions. Crucially, DGA subsumes standard adversarial training as an unconstrained special case: setting $\lambda = 0$ exactly recovers conventional AT (DGA_0), whereas $\lambda > 0$ defines a continuous family of objectives that constrains parameter displacement when a smaller departure from ERM is optimal. Rather than treating adversarial training as a rigid, all-or-nothing heuristic, DGA provides a unified framework that adaptively governs the optimization trajectory according to model capacity. Our main contributions are threefold:

- **Observation of Within-Domain Reversal:** We discover that identical input-space adversarial training produces opposite clean accuracy effects across time-series models, demonstrating that adversarial generalization is governed by model capacity rather than data modality.
- **Mechanistic and Theoretical Formulation of DGA:** We formalize the competition between variance contraction and parameter displacement via asymptotic expansions and an exact risk identity, and develop DGA to adaptively govern this trade-off through parameter anchoring.
- **Comprehensive Benchmarks and Cross-Modal Transfer:** Across an exhaustive 56-setting forecasting grid (8 models $\times$ 7 datasets, 5 paired seeds), validation-selected DGA achieves lower clean MSE than paired ERM in 51 of 56 settings ($p = 1.4 \times 10^{-5}$), converting linear-model degrada-

tions into positive gains (e.g., -3.70% MSE on DLinear). Furthermore, DGA transfers effectively to language (improving BERT's three-task GLUE mean by +0.41 points over 20 seeds) and vision (recovering clean accuracy on CIFAR and stabilizing SVHN).

2 Related Work

We position our study at the intersection of three lines of literature: (1) adversarial training across modalities, (2) anchored optimization and parameter trajectory control, and (3) spectral dynamics and bias-variance trade-offs.

Adversarial Training: From Robustness to Clean Generalization. Adversarial training was originally developed to defend deep networks against worst-case input perturbations (Goodfellow et al., 2015; Madry et al., 2018; Shafahi et al., 2019). In computer vision, extensive studies have documented that pixel-space adversarial training typically induces an accuracy-robustness trade-off, where gaining adversarial resilience comes at the cost of degraded clean accuracy (Tsipras et al., 2019; Zhang et al., 2019; Stutz et al., 2019). In contrast, natural language processing frameworks inject perturbations into latent word representations, treating adversarial training as a smoothness regularizer that improves clean benchmark generalization (Miyato et al., 2017; Zhu et al., 2020; Jiang et al., 2020). In time series, prior literature has established adversarial resilience as an important forecasting objective, primarily evaluating attacks and defenses for classification (Ismail Fawaz et al., 2019) or robust probabilistic forecasting under distribution shifts (Yoon et al., 2022; Liu et al., 2023). Complementary to these robustness-oriented defenses, our work investigates whether and why raw-history adversarial perturbations can actively enhance or degrade clean, unperturbed time-series forecasting across different model classes.

Parameter Trajectory Control. Penalizing parameter deviation from a reference checkpoint has proven effective in transfer and continual learning, such as L_2-SP (Li et al., 2018) in vision transfer and SMART (Jiang et al., 2020) in language fine-tuning. However, these methods are designed for cross-task adaptation to preserve source-domain knowledge. In contrast, DGA employs the same-task clean ERM solution as an anchor, providing a single-task control mechanism that prevents parameters from drifting away from the clean empirical minimum during adversarial training.

Spectral Dynamics and Bias-Variance Trade-offs. Frequency analysis offers valuable perspectives on neural optimization, spanning general spectral bias (Rahaman et al., 2019; Yin et al., 2019), theoretical adversarial linear regression (Javanmard et al., 2020; Roth et al., 2020), and frequency-based forecasting architectures (Zhou et al., 2022; Wu et al., 2023; Yi et al., 2024; Wang et al., 2025). However, existing literature focuses either on architectural decomposition or stylized linear models, leaving unaddressed how temporal adversarial gradient frequencies govern the empirical bias-variance trade-off across varying model capacities.

3 Preliminaries

Time-Series Forecasting. Given an observed continuous historical window $x \in \mathbb{R}^{L\times d}$ spanning L lookback steps over d variates, the objective of time-series forecasting is to predict future values $y \in \mathbb{R}^{H\times d}$ across a forecast horizon H. A model $f_\theta : \mathbb{R}^{L\times d} \to \mathbb{R}^{H\times d}$ parameterized by θ maps the history to predictions $\hat{y} = f_\theta(x)$. Standard Empirical Risk Minimization (ERM) minimizes the mean squared error (MSE) across N sliding training pairs (x_i, y_i):

$$\mathcal{L}_{\mathrm{ERM}}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \ell(f_\theta(x_i), y_i), \quad \text{where} \quad \ell(\hat{y}, y) = \frac{1}{Hd}\|\hat{y} - y\|_2^2. \tag{2}$$

Clean risk refers to the test MSE evaluated on unperturbed historical inputs.

Adversarial Training and the Zero-Anchor Endpoint (DGA_0). DGA's zero-anchor endpoint, DGA_0, exposes a model to loss-increasing input perturbations during optimization (Goodfellow et al., 2015; Madry et al., 2018). In time-series forecasting, these perturbations act directly on the continuous historical input x. For each instance (x, y), the inner problem seeks an additive perturbation $\delta^* \in \mathbb{R}^{L\times d}$ within an ℓ_∞ budget $\|\delta\|_\infty \le \epsilon$:

$$\delta^* = \arg\max_{\|\delta\|_\infty \le \epsilon} \ell(f_\theta(x + \delta), y), \tag{3}$$

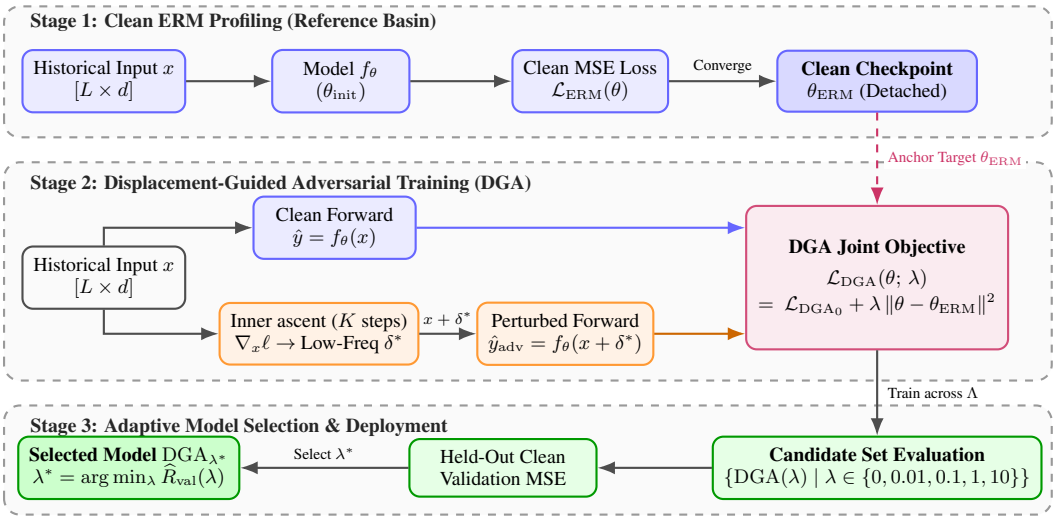

Figure 2: Overview of Displacement-Guided Adversarial Training (DGA). Stage 1 fits the clean reference θ_{ERM}. Stage 2 adds the displacement anchor $\lambda\|\theta-\theta_{\text{ERM}}\|^2$ to the mixed DGA_0 objective. Stage 3 selects λ^* on clean validation error: $\lambda^* = 0$ retains DGA_0, whereas a positive value limits learned displacement.

which we approximate with K projected sign-gradient ascent steps: $\delta_{k+1} = \Pi_{\|\delta\|_\infty \le \epsilon}\big(\delta_k + \alpha \cdot \text{sign}(\nabla_x \ell(f_\theta(x+\delta_k), y))\big)$ initialized at $\delta_0 = 0$, taking $\delta^* = \delta_K$. Here Π projects onto the ℓ_∞ ball and α is the inner step size. The outer objective mixes clean and perturbed historical sequences:

$$\mathcal{L}_{\text{DGA}_0}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\Big[(1-\beta)\ell(f_\theta(x_i), y_i) + \beta\ell(f_\theta(x_i + \delta_i^*), y_i)\Big], \tag{4}$$

where $\beta \in (0, 1]$ balances the clean and perturbed objectives (standardly $\beta = 0.5$).

Frequency Decomposition. To characterize the temporal frequency properties of gradients and perturbations, we apply the one-sided discrete Fourier transform $\mathcal{F}_+$ along the time dimension. For any sequence $z \in \mathbb{R}^{L\times d}$, we define its low-frequency energy ratio as:

$$r_g(z) = \frac{\|M_L \mathcal{F}_+ z\|_2^2}{\|\mathcal{F}_+ z\|_2^2} \in [0, 1], \qquad z \neq 0. \tag{5}$$

where M_L selects the lowest half of the frequency bins, $\lfloor (L/2+1)/2 \rfloor$ (yielding an expected baseline of $24/49 \approx 0.490$ for white noise at $L = 96$). Stored one-sided bins receive equal weight, while prediction-frequency probes replace L by H. We denote the real-space orthogonal projectors onto the low- and high-frequency subspaces as P_L and $P_H = I - P_L$, respectively, whose time-domain reconstructed energy accounts for Parseval weights (Appendix E.2).

4 Method

To diagnose and resolve the clean-effect reversal in time-series adversarial training, we establish a unified framework connecting finite-network optimization to asymptotic theory. First, we derive an exact finite-network risk identity (§4.1) that partitions clean risk change into beneficial residual correction and an adverse movement cost. Guided by this diagnostic, we introduce Displacement-Guided Adversarial Training (DGA, §4.2), which anchors the optimization trajectory to govern parameter drift along Hessian curvature directions. Finally, we establish an asymptotic polarity theory (§4.3) in a projected temporal subspace, formalizing the competition between covariance contraction and population shift (Theorem 1).

4.1 Finite-Network Diagnosis: The Parameter Displacement Dilemma

To uncover why the zero-anchor endpoint produces opposite clean generalization effects across model architectures, we first establish an exact quantitative diagnostic for finite networks. For any

model θ, let $R(\theta) = \mathbb{E}[\ell(f_\theta(x), y)]$ denote its clean test MSE. We formalize the relative performance change of DGA_0 over ERM via the *empirical clean-risk polarity*:

$$\mathrm{Pol} = 100 \cdot \frac{R(\theta_{\mathrm{ERM}}) - R(\theta_{\mathrm{DGA}_0})}{R(\theta_{\mathrm{ERM}})}, \tag{6}$$

where $\mathrm{Pol} > 0$ indicates clean accuracy improvement, and $\mathrm{Pol} < 0$ marks performance degradation.

To mathematically diagnose what governs the sign of Pol, consider a clean instance (x_c, y_c) with ERM residual $e_c = f_{\theta_{\mathrm{ERM}}}(x_c) - y_c$. Any optimization update shifting parameters by displacement $d = \theta - \theta_{\mathrm{ERM}}$ induces a prediction change $\Delta f_c(d) = f_{\theta_{\mathrm{ERM}}+d}(x_c) - f_{\theta_{\mathrm{ERM}}}(x_c)$. Under squared loss, expanding the clean test error yields an exact finite-network identity without approximation:

$$R(\theta_{\mathrm{ERM}} + d) - R(\theta_{\mathrm{ERM}}) = B_{\mathrm{disp}}(d) + S_{\mathrm{disp}}(d), \tag{7}$$

where $B_{\mathrm{disp}}(d) = \frac{2}{Hd}\mathbb{E}[e_c^\top \Delta f_c(d)]$ represents the signed coupling between prediction changes and ERM residuals, and $S_{\mathrm{disp}}(d) = \frac{1}{Hd}\mathbb{E}[\|\Delta f_c(d)\|_2^2] \geq 0$ denotes the non-negative prediction movement cost.

Equation 7 establishes the exact operational criterion for positive DGA_0 polarity: residual correction $-B_{\mathrm{disp}}(d)$ must exceed the movement cost $S_{\mathrm{disp}}(d)$. Locally, a first-order expansion $\Delta f_c(d) \approx J_c d$ (with Jacobian $J_c = \nabla_\theta f_\theta$) reveals a fundamental scaling asymmetry: residual correction $-B_{\mathrm{disp}}$ scales linearly with displacement $\|d\|_2$, whereas movement cost S_{disp} scales quadratically ($\propto \|d\|_2^2$). In expressive models, higher capacity enables prediction changes to align effectively against ERM residuals, letting linear correction outweigh quadratic cost. In compact models, unconstrained updates inflate movement cost S_{disp} faster than any residual benefit, deteriorating clean risk. This measured trade-off motivates the anchored formulation in §4.2.

4.2 Displacement-Guided Adversarial Training (DGA)

The diagnostic identity reveals that DGA_0 degrades compact models when parameter displacement d is unconstrained. Rather than discarding adversarial perturbations, we introduce *Displacement-Guided Adversarial Training* (DGA) to govern this drift by anchoring optimization to a fitted clean reference:

$$\mathcal{L}_{\mathrm{DGA}}(\theta; \lambda) = \mathcal{L}_{\mathrm{DGA}_0}(\theta) + \lambda\|\theta - \theta_{\mathrm{ERM}}\|_2^2, \tag{8}$$

where $\mathcal{L}_{\mathrm{DGA}_0}(\theta)$ is the mixed objective in Equation 4, and $\lambda \geq 0$ controls the anchor strength.

Curvature-Selective Contraction. A natural question is whether the isotropic anchor indiscriminately suppresses useful adversarial updates. Expanding the stationary condition around the unanchored minimizer $\hat{\theta}_{\mathrm{DGA}_0}$ reveals that the penalty naturally acts through the loss Hessian H_0 (Proposition 1 in Appendix B.1):

$$d_{\mathrm{DGA}}(\lambda) \approx (H_0 + 2\lambda I)^{-1} H_0 \, d_{\mathrm{DGA}}(0). \tag{9}$$

Along each Hessian eigendirection with eigenvalue $\mu_i > 0$, displacement contracts by $\frac{\mu_i}{\mu_i + 2\lambda}$. In stiff directions ($\mu_i \gg 2\lambda$), coordinates strongly constrained by the adversarial loss are largely preserved ($\frac{\mu_i}{\mu_i+2\lambda} \approx 1$); in flatter directions ($\mu_i \ll 2\lambda$), the anchor quenches weakly constrained drift. Because quadratic movement cost S_{disp} contracts faster than linear residual correction, this curvature-selective filtering allows a positive anchor to recover lower clean risk whenever unanchored displacement is excessive.

Two-Stage Optimization and Adaptive Selection. In practice, DGA first trains ERM to convergence and freezes the anchor θ_{ERM}. By default, each DGA candidate is trained from the same seed-matched random initialization as its ERM counterpart; the ERM checkpoint serves as the penalty center rather than a warm start. We search the compact grid $\Lambda = \{0, 0.01, 0.1, 1.0, 10\}$ and select $\lambda^* = \arg\min_{\lambda \in \Lambda} \widehat{R}_{\mathrm{val}}(\theta_{\mathrm{DGA}}(\lambda))$ on clean validation MSE. When unconstrained training is already beneficial, $\lambda^* = 0$ recovers DGA_0; otherwise, a positive λ^* limits displacement. Matched ERM-initialized continuation yields nearly identical solutions, confirming that the objective and anchor—rather than starting weights—determine the recovered operating point (Appendix E.4).

4.3 Asymptotic Statistical Foundation

While the finite-network identity characterizes prediction geometry on a fixed dataset, it does not isolate estimation variability. We therefore formalize this trade-off via an asymptotically tractable *projected linear M-estimator* in the low-frequency temporal subspace U_L defined in §3.

Let $z = U_L^\top \mathrm{vec}(x) \in \mathbb{R}^q$ denote low-frequency inputs, and consider the forecaster $f(z) = w_L^\top z$. Under standard mixing and regularity conditions (Assumptions A1–A3 in Appendix B.2), the estimator's sandwich covariance expands around the clean population optimum w_0^* as $C_\epsilon = C_0 + \beta\epsilon C_1 + \mathcal{O}(\epsilon^2)$. Evaluating the asymptotic clean risk under potential temporal shift yields our core theoretical result:

Theorem 1 (Asymptotic Polarity Decomposition)**.** *For n sliding training windows and perturbation budget ϵ, the asymptotic clean-risk polarity satisfies:*

$$\mathrm{Pol}_n^{\mathrm{asy}}(\epsilon) \;=\; \beta\epsilon\left(\frac{V_1}{n} - D_1\right) + \mathcal{O}(\epsilon^2) + \mathcal{O}(\epsilon^2/n)\,, \tag{10}$$

where $V_1 = -\mathrm{tr}(\Sigma_{\mathrm{test}} C_1) > 0$ represents covariance contraction under curvature dominance (Assumption A4'), and $D_1 = 2(w_0^* - w^*)^\top \Sigma_{\mathrm{test}} b_1$ captures population displacement under distribution shift ($d_0 = w_0^* - w^*$).

Actionable Insights for Algorithm Design. Theorem 1 isolates two competing first-order channels in the projected model:

1. **The Variance Dividend ($\frac{V_1}{n} > 0$):** Adversarial updates contract estimator covariance at rate $\mathcal{O}(1/n)$, directly corroborating the selective low-frequency variance reduction observed in neural models (Figure 1(c)).
2. **The Displacement Toll (D_1):** Concurrently, adversarial perturbations induce an $\mathcal{O}(1)$ parameter shift away from the clean optimum, penalizing clean performance whenever temporal distribution shifts occur ($w_0^* \neq w^*$).

Thus covariance contraction alone cannot guarantee clean gains: it competes directly with population shift. While D_1 is governed by underlying distributions, DGA provides the operational remedy by governing empirical displacement $B_{\mathrm{disp}} + S_{\mathrm{disp}}$ during training, effectively preserving adversarial variance reduction while mitigating destructive parameter drift.

5 Experiments

We conduct extensive experiments to evaluate DGA and empirically validate the theoretical mechanisms established in §4. Our investigation addresses four key questions: (1) Does DGA consistently improve clean forecasting accuracy across diverse architectures and datasets? (§5.2); (2) Does the adversarial gradient carry genuine predictive signals beyond low-frequency noise? (§5.3); (3) Does model capacity govern the net bias-variance balance, and can parameter anchoring restore positive polarity? (§5.3); and (4) Does displacement control generalize to language and vision? (§5.4).

5.1 Experimental Setup

Datasets and Tasks. We evaluate seven standard forecasting benchmarks (Zhou et al., 2021): the four Electricity Transformer Temperature variants (ETTh1, ETTh2, ETTm1, ETTm2), Weather, Electricity, and Traffic. They span diverse domains, sampling rates (10-minute to hourly), and variate dimensions ranging from compact systems ($d = 7$) to high-dimensional sensor networks ($d = 862$). Following standard protocol, all datasets are split chronologically into training, validation, and test partitions (Appendix A.1). We evaluate default lookback length $L = 96$ and prediction horizon $H = 96$, extending to long horizons $H \in \{192, 336, 720\}$ in §5.2.

Forecasting Backbones. To systematically examine the influence of model capacity on adversarial training, we benchmark eight representative architectures spanning the capacity spectrum: **Expressive Nonlinear Forecasters:** iTransformer (Liu et al., 2024), PatchTST (Nie et al., 2023), PaiFilter (Yi et al., 2024), FACT (Wang et al., 2026), and TimePro (Ma et al., 2025). **Compact Linear Forecasters:** DLinear (Zeng et al., 2023), FITS (Xu et al., 2024), and SparseTSF (Lin et al., 2024).

Training and Evaluation Protocol. DGA_0 uses $K = 3$ projected sign-gradient ascent steps with step size $\alpha = \epsilon/4$ and mixing weight $\beta = 0.5$. Fixed dataset-level ℓ_∞ budgets are applied across all channels: $\epsilon = 0.1$ for ETT, 0.01 for Weather and Electricity, and 0.003 for Traffic. DGA evaluates $\lambda \in \{0, 0.01, 0.1, 1.0, 10\}$ and selects λ^* strictly using the five-seed mean clean validation MSE per configuration ($\lambda = 0$ recovers DGA_0). All test MAE evaluations directly adopt the MSE-selected models without metric-specific re-tuning. All models are trained with AdamW across five paired random seeds. We report clean test MSE and MAE, with seed-paired macro t-tests, Holm, and Benjamini–Hochberg (BH) adjustments in Appendix D.1.

5.2 Main Forecasting Benchmarks

Table 1 reports clean forecasting performance across all 56 configurations at H=96 (complete three-way comparisons including unanchored DGA_0 appear in Appendix Table 9). Validation-selected $\mathrm{DGA}_{\lambda^\star}$ achieves lower clean MSE than paired ERM in **51 of 56 settings** (seed-paired macro $p = 1.4 \times 10^{-5}$), and lowers MAE in 44 of 56 settings using the identical MSE-selected checkpoints.

Table 1: Clean forecasting MSE/MAE at H=96 over five paired seeds. Five-seed mean validation MSE selects $\mathrm{DGA}_{\lambda^\star}$; MAE follows the same choice. Bold marks the lower mean.

Model	Method	ETTh1	ETTh2	ETTm1	ETTm2	Weather	ECL	Traffic
iTransformer	ERM	0.382/0.402	0.301/0.350	0.332/0.366	0.183/0.267	0.168/0.212	0.157/0.252	0.461/0.319
	$\mathrm{DGA}_{\lambda^\star}$	**0.376/0.396**	**0.292/0.343**	**0.316/0.355**	**0.171/0.252**	**0.163/0.208**	**0.156/0.251**	**0.459/0.317**
PatchTST	ERM	0.383/0.400	0.295/0.344	0.329/0.366	0.176/0.260	**0.164/0.210**	0.172/0.270	0.518/0.323
	$\mathrm{DGA}_{\lambda^\star}$	**0.379/0.398**	**0.289/0.339**	**0.323/0.361**	**0.174/0.256**	0.165/0.210	**0.171/0.268**	**0.514/0.319**
PaiFilter	ERM	0.388/0.402	0.298/0.349	0.332/0.369	0.176/0.258	**0.167/0.212**	0.167/0.258	0.494/0.329
	$\mathrm{DGA}_{\lambda^\star}$	**0.377/0.397**	**0.290/0.341**	**0.323/0.362**	**0.173/0.255**	0.168/0.213	**0.167/0.257**	**0.492/0.326**
FACT	ERM	0.416/0.421	0.303/0.350	0.334/0.369	0.182/0.264	0.162/0.208	0.137/0.234	0.436/0.268
	$\mathrm{DGA}_{\lambda^\star}$	**0.396/0.408**	**0.293/0.343**	**0.321/0.359**	**0.172/0.255**	**0.159/0.204**	**0.136/0.233**	**0.431/0.264**
TimePro	ERM	0.380/0.401	0.299/0.349	0.331/0.368	0.178/0.260	0.169/0.212	0.156/0.251	0.456/0.309
	$\mathrm{DGA}_{\lambda^\star}$	**0.372/0.395**	**0.293/0.342**	**0.322/0.360**	**0.172/0.254**	**0.166/0.209**	**0.156/0.250**	**0.455/0.308**
FITS	ERM	0.383/0.395	0.339/0.393	0.345/**0.372**	0.193/0.291	**0.196/0.256**	0.195/**0.278**	0.653/**0.399**
	$\mathrm{DGA}_{\lambda^\star}$	**0.381/0.395**	**0.337/0.391**	**0.345**/0.372	**0.192/0.290**	0.196/0.257	**0.195**/0.278	**0.652**/0.400
SparseTSF	ERM	0.390/0.394	0.305/0.347	**0.386/0.394**	0.206/**0.284**	0.196/0.237	0.202/0.278	0.661/**0.395**
	$\mathrm{DGA}_{\lambda^\star}$	**0.388/0.394**	**0.305/0.347**	0.386/0.394	**0.206**/0.284	**0.196/0.236**	**0.202/0.278**	**0.660**/0.395
DLinear	ERM	0.391/**0.405**	0.305/0.362	0.345/**0.371**	0.184/0.275	**0.198/0.262**	0.195/0.277	0.649/0.398
	$\mathrm{DGA}_{\lambda^\star}$	**0.389**/0.406	**0.294/0.349**	**0.343**/0.371	**0.182/0.271**	0.200/0.265	**0.195/0.277**	**0.649/0.398**

The selected anchors reveal a clear, capacity-linked split (Appendix C.1 and Table 7). Among 35 nonlinear cells, validation selects $\lambda^\star = 0$ in 15 and the minimal positive anchor 0.01 in 14. Conversely, among 21 linear cells, validation activates a positive anchor in 18, with 17 falling in $[0.1, 10]$. Crucially, the selected anchor improves upon DGA_0 in 17 of these 18 linear cells and in 36 of all 38 positive-anchor cells ($p = 4.7 \times 10^{-4}$). Thus, displacement control is most active precisely where unconstrained adversarial updates drift too far from the clean basin, turning prior linear-model degradations into positive gains (e.g., a 3.70% MSE reduction on DLinear/ETTh2).

Multi-Horizon Generalization. Table 2 evaluates DGA across horizons $H \in \{96, 192, 336, 720\}$ on iTransformer. It lowers both MSE and MAE in 13 of 16 settings and improves both metrics at every ETTm1 and ETTm2 horizon, including a 5.2% MSE reduction at $H = 720$ (Appendix D.2).

Table 2: **Multi-horizon forecasting on ETT ($L = 96$).** Clean MSE and MAE of iTransformer across horizons $H \in \{96, 192, 336, 720\}$. Bold marks the lower mean over paired seeds.

H	Method	ETTh1		ETTh2		ETTm1		ETTm2	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
96	ERM	0.3819	0.4017	0.3010	0.3500	0.3322	0.3665	0.1832	0.2667
	$\mathrm{DGA}_{\lambda^\star}$	**0.3757**	**0.3965**	**0.2922**	**0.3433**	**0.3161**	**0.3549**	**0.1706**	**0.2519**
192	ERM	0.4333	0.4313	0.3770	0.3981	0.3829	0.3938	0.2508	0.3110
	$\mathrm{DGA}_{\lambda^\star}$	**0.4276**	**0.4262**	**0.3734**	**0.3944**	**0.3645**	**0.3815**	**0.2365**	**0.2956**
336	ERM	0.4734	0.4532	0.4175	**0.4299**	0.4217	0.4169	0.3157	0.3510
	$\mathrm{DGA}_{\lambda^\star}$	**0.4684**	**0.4475**	**0.4171**	0.4306	**0.4036**	**0.4055**	**0.2965**	**0.3352**
720	ERM	**0.4744**	0.4743	**0.4247**	**0.4444**	0.5034	0.4598	0.4138	0.4069
	$\mathrm{DGA}_{\lambda^\star}$	0.4746	**0.4713**	0.4285	0.4455	**0.4771**	**0.4446**	**0.3967**	**0.3938**

5.3 Empirical Mechanism: Deconstructing the Reversal

We now examine the three mechanistic pillars established in §4:

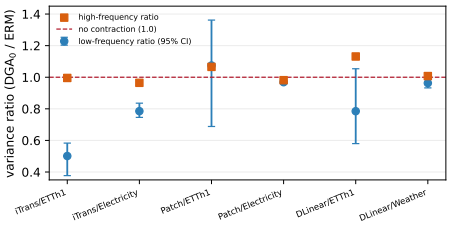

Figure 3: Across-seed prediction-variance ratio (AT/ERM) by frequency band. Ratios below 1.0 indicate contraction. Low-frequency variance contracts in four of six settings (blue circles, 95% CIs below 1.0); high-frequency response varies by setting, from near one to moderate contraction.

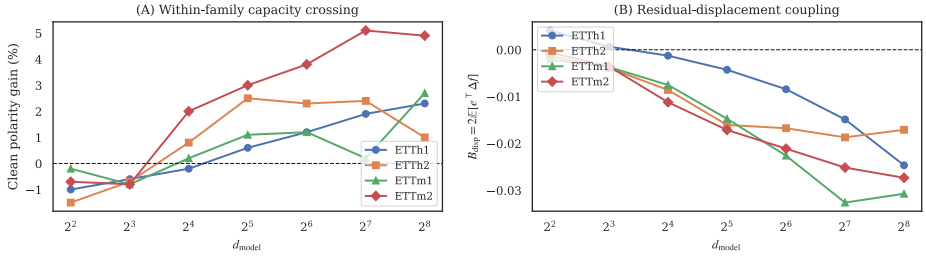

Figure 4: **Width-controlled capacity intervention on iTransformer.** (A) Expanding hidden dimension flips clean polarity from negative to positive across four datasets. (B) The signed coupling B_{disp} becomes increasingly negative, allowing residual correction to overtake movement cost S_{disp}.

Direction-Specific Predictive Alignment. Spectral analysis confirms that loss gradients and adversarial perturbations concentrate heavily in slow temporal dynamics, with low-frequency energy ratios of 0.71–0.77 for gradients and 0.65–0.82 for perturbations, compared with the 0.490 bin-count reference (Appendix E.1). To verify that clean gains require this specific gradient trajectory rather than generic low-frequency smoothing, we evaluate matched controls with identical magnitude spectra but randomized phases, alongside low-pass filtered uniform noise (Appendix E.1). While retraining with low-pass filtered adversarial perturbations preserves clean generalization gains, low-pass random noise tracks ERM. Thus, clean benefits stem strictly from optimization guided by error-sensitive gradient directions.

Prediction-Variance Contraction. To test whether the theoretical covariance contraction of Theorem 1 (V_1/n) operates in finite networks, we evaluate prediction variance across frequency bands (Figure 3). Across six matched settings, low-frequency prediction variance contracts in four, with ratios from 0.50 to 1.07 and 95% bootstrap CIs below one in four settings. The high-frequency response is setting-dependent: it is nearly unchanged on iTransformer/ETTh1 (ratio 0.995), iTransformer/Electricity (0.965), and PatchTST/Electricity (0.981), while the corresponding low-frequency ratios range from 0.501 to 0.969. This establishes a selective low-frequency contraction rather than a frequency-independent shrinkage, further corroborated under block-bootstrap sequence resampling (Appendix E.2).

Capacity Crossing and Displacement Control. The risk identity $\Delta R = B_{\text{disp}} + S_{\text{disp}}$ (Equation 7) posits that compact models degrade because unconstrained updates inflate movement cost S_{disp} beyond residual correction $-B_{\text{disp}}$. To verify that capacity governs this balance, we evaluate a controlled width intervention by scaling iTransformer's hidden dimension $d_{\text{model}} \in \{4, \ldots, 256\}$ across four datasets (Figure 4). Scaling width alone flips clean polarity from negative to positive across all datasets (Fig. 4A) as B_{disp} becomes increasingly negative (Fig. 4B), confirming that higher capacity enables residual corrections to overcome movement costs.

For capacity-constrained models where width cannot be scaled, DGA's displacement anchor provides the operational remedy: penalizing parameter distance from θ_{ERM} suppresses quadratic movement cost S_{disp} faster than linear residual correction. On DLinear/ETTh2, a prediction decomposition at $\lambda = 0.1$ shows concurrent reductions in ensemble-mean bias ($0.301 \to 0.291$) and variance ($0.006 \to 0.0035$). Crucially, on DLinear/ETTh2, the validation-selected DGA solution attains an MSE 0.0097 (3.21%) below the best point along the entire post-hoc ERM–DGA_0 checkpoint interpolation path (Appendix E.4), confirming that anchored optimization explores a superior trajectory that cannot be replicated by simple post-hoc weight averaging.

Objective, Initialization, and Cost Controls. Fresh five-seed reruns reproduce negative DGA_0 polarity in all eight diagnostic linear cells. Relative to ERM, selected DGA and a clean-only anchored control reduce mean MSE by 0.79% and 0.83%, respectively, showing that proximal anchoring independently regularizes the estimator in this regime. Crucially, while clean anchoring stabilizes compact models, DGA's primary strength is establishing a unified adversarial pipeline: it preserves unconstrained adversarial gains on expressive architectures ($\lambda^* = 0$) while adaptively engaging the anchor to prevent destructive drift on compact ones. Furthermore, initializing DGA from the ERM checkpoint differs from cold start by only -0.04% in selected MSE, confirming that the objective and anchor—rather than warm-starting—determine the recovered solution. Across eight architectures, DGA incurs a modest overhead of $1.25\times$ peak memory and $4.19\times$ training seconds per epoch on geometric average (Appendix E.4).

5.4 Cross-Modal Generality: Language and Vision

To test whether displacement-guided trajectory control generalizes beyond time series, we evaluate standard language and vision benchmarks under task-specific perturbations (Table 3).

Table 3: Cross-modal clean accuracy (%, mean ± std). Evaluated on BERT-base across 20 paired seeds (NLP) and PreActResNet-18 across 5 seeds (CV). Bold marks the highest clean performance.

Benchmark	ERM	AT	Selected DGA (DGA_{λ^*})	p vs. ERM
Language: BERT-base fine-tuning on GLUE development sets (20 seeds)				
SST-2	92.84±0.45	92.98±0.34	**93.04**±0.49	0.0066
MRPC	84.34±1.16	**85.21**±1.47	84.62±1.51	0.182
RTE	66.44±2.62	**67.44**±2.26	67.18±2.23	0.063
Vision: PreActResNet-18 image classification (5 seeds)				
CIFAR-10	**91.55**±0.19	83.35±1.65	85.41±0.57	> 0.999
CIFAR-100	**69.62**±0.28	59.48±0.19	60.31±0.25	> 0.999
SVHN	**96.65**±0.09	61.85±38.78	94.34±0.54	> 0.999

Natural Language Processing. Fine-tuning BERT-base (Devlin et al., 2019) on three GLUE tasks (Wang et al., 2019) with word-embedding perturbations shows that validation-selected DGA consistently improves clean accuracy over ERM across 20 paired seeds ($+0.20$ on SST-2, $+0.28$ on MRPC, and $+0.74$ on RTE). The 20-seed macro average achieves a statistically significant gain of $+0.41$ percentage points ($p = 0.016$), confirming that parameter anchoring effectively preserves adversarial smoothness benefits during language model fine-tuning.

Computer Vision. For image classification with PreActResNet-18 (He et al., 2016), positive displacement anchors recover 2.06 percentage points on CIFAR-10 and 0.83 points on CIFAR-100 over unanchored **AT**. Most notably, on SVHN, standard adversarial training (AT) experiences severe optimization collapse on two of five seeds (collapsing to 19.6% accuracy, resulting in a degraded mean of 61.85%), whereas selected DGA completely stabilizes optimization across all five seeds ($94.34\% \pm 0.54\%$). These results demonstrate that parameter displacement control operates as a domain-general stabilizer for adversarial optimization beyond time-series forecasting.

6 Conclusion

In this work, we investigated when and why adversarial training improves clean generalization, uncovering a capacity-driven within-domain reversal in time-series forecasting: while input perturbations act as a frequency-selective regularizer that contracts prediction variance, they induce parameter displacement that degrades compact linear models. To resolve this trade-off, we introduced Displacement-Guided Adversarial Training (DGA), which anchors the parameter trajectory to a clean ERM checkpoint along Hessian curvature directions, enabling forecasters across the capacity spectrum to harvest variance reduction while preserving low clean bias. Extensive evaluations across 56 forecasting configurations, as well as downstream language and computer vision benchmarks, establish that parameter displacement control operates as a domain-general stabilizer for adversarial optimization. Future work can test displacement-guided trajectory control in large-scale pre-training and time-series foundation models, where its interaction with richer temporal representations and longer deployment horizons remains to be evaluated.

LLM Usage Statement. Generative AI was used for language polishing and writing assistance. The authors designed and conducted all experiments, verified all claims, and take full responsibility for the final content.

REFERENCES

Donald W. K. Andrews. Empirical process methods in econometrics. In *Handbook of Econometrics, Vol. 4*, pp. 2247–2294. North-Holland, 1994.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, 2019.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*, 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision (ECCV)*, 2016.

Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. Adversarial attacks on deep neural networks for time series classification. In *International Joint Conference on Neural Networks (IJCNN)*, 2019.

Adel Javanmard, Mahdi Soltanolkotabi, and Hamed Hassani. Precise tradeoffs in adversarial training for linear regression. In *Conference on Learning Theory (COLT)*, pp. 2034–2078, 2020.

Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao. SMART: Robust and efficient fine-tuning for pre-trained natural language models through principled regularized optimization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 2177–2190, 2020. doi: 10.18653/v1/2020.acl-main.197. URL https://aclanthology.org/2020.acl-main.197/.

Taesung Kim, Jinheon Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo. Reversible instance normalization for accurate time-series forecasting against distribution shift. In *International Conference on Learning Representations (ICLR)*, 2022.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Xuhong Li, Yves Grandvalet, and Franck Davoine. Explicit inductive bias for transfer learning with convolutional networks. In *International Conference on Machine Learning (ICML)*, pp. 2825–2834, 2018. URL https://proceedings.mlr.press/v80/li18a.html.

Shengsheng Lin, Weiwei Lin, Wentai Wu, Haojun Chen, and Junjie Yang. SparseTSF: Modeling long-term time series forecasting with 1k parameters. In *International Conference on Machine Learning (ICML)*, 2024.

Linbo Liu, Youngsuk Park, Trong Nghia Hoang, Hilaf Hasson, and Jun Huan. Robust multivariate time-series forecasting: Adversarial attacks and defense mechanisms. In *International Conference on Learning Representations (ICLR)*, 2023.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In *International Conference on Learning Representations (ICLR)*, 2024.

Xiaowen Ma, Zhenliang Ni, Shuai Xiao, and Xinghao Chen. TimePro: Efficient multivariate long-term time series forecasting with variable- and time-aware hyper-state. In *International Conference on Machine Learning (ICML)*, 2025.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*, 2018.

Takeru Miyato, Andrew M. Dai, and Ian Goodfellow. Adversarial training methods for semi-supervised text classification. In *International Conference on Learning Representations (ICLR)*, 2017.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. Technical report, Stanford University, 2011.

Whitney K. Newey and Daniel McFadden. Large sample estimation and hypothesis testing. In *Handbook of Econometrics*, volume 4. Elsevier, 1994.

Whitney K. Newey and Kenneth D. West. A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. *Econometrica*, 55(3):703–708, 1987.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations (ICLR)*, 2023.

Ariel Pakes and David Pollard. Simulation and the asymptotics of optimization estimators. *Econometrica*, 57(5):1027–1057, 1989.

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

Kevin Roth, Yannic Kilcher, and Thomas Hofmann. Adversarial training is a form of data-dependent operator norm regularization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Ali Shafahi, Mahyar Najibi, Amin Ghiasi, Zheng Xu, John Dickerson, Larry S. Davis, Micah Goldblum, and Tom Goldstein. Adversarial training for free! In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

David Stutz, Matthias Hein, and Bernt Schiele. Disentangling adversarial robustness and generalization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations (ICLR)*, 2019.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*, 2019.

Hao Wang, Licheng Pan, Zhichao Chen, Degui Yang, Sen Zhang, Yifei Yang, Xinggao Liu, Haoxuan Li, and Dacheng Tao. FreDF: Learning to forecast in the frequency domain. In *International Conference on Learning Representations (ICLR)*, 2025.

Huiqiang Wang, Jieming Shi, and Qing Li. FACT: Fine-grained across-variable convolution for multivariate time-series forecasting. In *International Conference on Learning Representations (ICLR)*, 2026.

Halbert White. Maximum likelihood estimation of misspecified models. *Econometrica*, 50(1):1–25, 1982.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. TimesNet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations (ICLR)*, 2023.

Zhijian Xu, Ailing Zeng, and Qiang Xu. FITS: Modeling time series with 10k parameters. In *International Conference on Learning Representations (ICLR)*, 2024.

Kun Yi, Jingru Fei, Qi Zhang, Hui He, Shufeng Hao, Defu Lian, and Wei Fan. FilterNet: Harnessing frequency filters for time series forecasting. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Dong Yin, Raphael Gontijo Lopes, Jonathon Shlens, Ekin D. Cubuk, and Justin Gilmer. A fourier perspective on model robustness in computer vision. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Taeho Yoon, Youngsuk Park, Ernest K. Ryu, and Yuyang Wang. Robust probabilistic time series forecasting. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2022.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *AAAI Conference on Artificial Intelligence (AAAI)*, 2023.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. Theoretically principled trade-off between robustness and accuracy. In *International Conference on Machine Learning (ICML)*, 2019.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In *International Conference on Machine Learning (ICML)*, 2022.

Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. Freelb: Enhanced adversarial training for natural language understanding. In *International Conference on Learning Representations (ICLR)*, 2020.

APPENDIX

This appendix is organized into six sections:

- **Appendix A: Experimental Setup and Reproducibility Protocols** detailing dataset splits, model training configurations, and the formal DGA algorithm.
- **Appendix B: Theoretical Proofs and Mathematical Foundations** containing the derivations of the exact finite-network risk identity, the proof of Proposition 1, and the asymptotic M-estimation proof of Theorem 1.
- **Appendix C: Detailed Forecasting Benchmarks and Statistical Significance** presenting full-grid MSE/MAE tables, the $\lambda^\star$ distribution breakdown, multi-horizon evaluations, and multiplicity corrections.
- **Appendix E: Empirical Mechanism Analysis and Diagnostic Probes** covering directional alignment controls, block-bootstrap variance tests, capacity sweeps, and loss-surface visualizations.
- **Appendix F: Cross-Modal Generalization in Language and Vision** providing complete experimental protocols, per-seed results, and paired tests for BERT and PreActResNet-18.
- **Appendix G: Limitations and Discussion** discussing the scope of the analysis, high-dimensional forecasting, and cross-modal training responses.

A EXPERIMENTAL SETUP AND REPRODUCIBILITY PROTOCOLS

A.1 DATASET STATISTICS AND PARTITIONING

All benchmarks follow standard long-term forecasting protocol (Zhou et al., 2021). Dataset statistics are summarized in Table 4. We use lookback length $L = 96$ across all experiments, with default forecast horizon $H = 96$ and extended horizons $H \in \{96, 192, 336, 720\}$. Following standard practice, ETT datasets adopt a 12/4/4-month train/validation/test split; Weather, Electricity, and Traffic follow a 7:1:2 chronological ratio.

Table 4: Dataset statistics. Samples indicate raw series lengths; split counts are time points before constructing lookback/forecast windows.

Dataset	d	Frequency	Samples	Train / Val / Test (Time Points)
ETTh1	7	hourly	17,420	8,640 / 2,880 / 2,880
ETTh2	7	hourly	17,420	8,640 / 2,880 / 2,880
ETTm1	7	15-min	69,680	34,560 / 11,520 / 11,520
ETTm2	7	15-min	69,680	34,560 / 11,520 / 11,520
Weather	21	10-min	52,696	36,887 / 5,270 / 10,539
Electricity	321	hourly	26,304	18,412 / 2,632 / 5,260
Traffic	862	hourly	17,544	12,280 / 1,756 / 3,508

A.2 MODEL CONFIGURATIONS AND HYPERPARAMETERS

Table 5 lists the per-model default configurations for the main forecasting grid. Optimization uses AdamW with constant learning rate, weight decay 0, and dropout 0.1. Early stopping monitors clean validation MSE with model-specific patience. The batch size is 32 (16 for TimePro). The mixed objective weight is $\beta = 0.5$; the inner perturbation uses $K = 3$ projected sign-gradient ascent steps with $\alpha = \epsilon/4$.

Table 5: Per-model default training configurations.

Model	d_{model}	Layers	Learning Rate	Normalization	Max Epochs	Patience
iTransformer	128	2	1×10^{-4}	RevIN	10	3
PatchTST	128	3	1×10^{-4}	RevIN	10	3
PaiFilter	256	2	5×10^{-3}	none	15	5
FACT	512	1	1×10^{-3}	RevIN	15	5
TimePro	16	2	5×10^{-5}	RevIN	10	3
FITS	128	1	5×10^{-4}	none	30	5
SparseTSF	128	2	1×10^{-3}	none	30	5
DLinear	128	1	1×10^{-3}	none	100	10

A.3 DGA Training and Reproducibility Protocol

Algorithm 1 Displacement-Guided Adversarial Training (DGA).

Require: batch (x, y); model f_θ; ERM checkpoint θ_{ERM}; budget ϵ; step size α; steps K; mixing weight β; anchor λ
Inner ascent on history: $\delta \leftarrow 0$;
for $k = 1, \ldots, K$ **do**
  $g \leftarrow \nabla_\delta \ell(f_\theta(x+\delta), y)$
  $\delta \leftarrow \Pi_{\|\delta\|_\infty \le \epsilon}(\delta + \alpha\,\text{sign}(g))$
end for
$\delta^* \leftarrow \delta$
Displacement-guided update: $\mathcal{L}_{\text{DGA}} \leftarrow (1-\beta)\ell(f_\theta(x), y) + \beta\ell(f_\theta(x+\delta^*), y) + \lambda\|\theta - \theta_{\text{ERM}}\|_2^2$
Update θ via gradient descent on $\nabla_\theta \mathcal{L}_{\text{DGA}}$

Two-Stage Workflow. DGA is implemented in two strictly separated stages:

1. *Stage 1 (Clean ERM Baseline):* Train the model under standard ERM to convergence (early-stopping on clean validation MSE), and save the detached parameter vector θ_{ERM}.
2. *Stage 2 (Anchored Training):* Re-initialize the model from scratch using the exact same random initialization seed as Stage 1. Train under $\mathcal{L}_{\text{DGA}}(\theta; \lambda) = (1-\beta)\mathcal{L}_{\text{clean}} + \beta\mathcal{L}_{\text{adv}} + \lambda\|\theta - \theta_{\text{ERM}}\|^2$ with $\beta = 0.5$.

Perturbation and Normalization Protocol. The iterative inner maximization operates on the training-standardized historical series before internal instance normalization (RevIN (Kim et al., 2022)). It starts from $\delta_0 = 0$ and applies $K = 3$ sign-gradient ascent steps of size $\alpha = \epsilon/4$, followed by projection onto $\|\delta\|_\infty \le \epsilon$. With this step size, the effective maximum radius is $\|\delta^*\|_\infty \le 0.75\epsilon < \epsilon$. Fixed ℓ_∞ budgets are: $\epsilon = 0.1$ for ETT, 0.01 for Weather and Electricity, and 0.003 for Traffic. For each model–dataset pair at $H = 96$, candidate anchors $\Lambda = \{0, 0.01, 0.1, 1, 10\}$ are selected by

$$\lambda^\star = \arg\min_{\lambda \in \Lambda} \frac{1}{5} \sum_{s=1}^{5} \widehat{R}_{\text{val}}(\theta_{s,\lambda}).$$

The five test runs use this shared selection; MAE is evaluated at the same anchor chosen for MSE.

B Theoretical Proofs and Mathematical Foundations

B.1 Finite-Network Risk Identity and Curvature Contraction

This subsection provides the rigorous algebraic derivation of the finite-network risk identity in §4.1, the formal statement and proof of Proposition 1, and its geometric interpretation.

Step-by-Step Derivation of the Exact Risk Identity. Consider a clean evaluation pair (x_c, y_c) where $x_c \in \mathbb{R}^{L\times d}$ and $y_c \in \mathbb{R}^{H\times d}$. Let $f_{\theta_{\text{ERM}}}$ denote the baseline predictor, and let $\theta = \theta_{\text{ERM}} + \Delta\theta$ be an updated model with displacement vector $\Delta\theta$. Define the unperturbed ERM prediction residual

as $e_c = f_{\theta_{\mathrm{ERM}}}(x_c) - y_c \in \mathbb{R}^{H\times d}$, and the prediction change as $\Delta f_c(\Delta\theta) = f_{\theta_{\mathrm{ERM}}+\Delta\theta}(x_c) - f_{\theta_{\mathrm{ERM}}}(x_c) \in \mathbb{R}^{H\times d}$. The displaced prediction satisfies $f_\theta(x_c) - y_c = e_c + \Delta f_c(\Delta\theta)$. Expanding squared error yields:

$$\|f_\theta(x_c) - y_c\|_2^2 = \|e_c + \Delta f_c(\Delta\theta)\|_2^2 = \|e_c\|_2^2 + 2e_c^\top \Delta f_c(\Delta\theta) + \|\Delta f_c(\Delta\theta)\|_2^2.$$

Subtracting the baseline ERM squared error $\|e_c\|_2^2 = \|f_{\theta_{\mathrm{ERM}}}(x_c) - y_c\|_2^2$, normalizing by Hd, and taking expectations over the clean test distribution, we obtain the exact identity:

$$R(\theta_{\mathrm{ERM}} + \Delta\theta) - R(\theta_{\mathrm{ERM}}) = B_{\mathrm{disp}}(\Delta\theta) + S_{\mathrm{disp}}(\Delta\theta), \tag{11}$$

where the signed residual-coupling term $B_{\mathrm{disp}}(\Delta\theta)$ and the non-negative movement cost $S_{\mathrm{disp}}(\Delta\theta)$ are:

$$B_{\mathrm{disp}}(\Delta\theta) \triangleq \frac{2}{Hd}\mathbb{E}\big[e_c^\top \Delta f_c(\Delta\theta)\big], \qquad S_{\mathrm{disp}}(\Delta\theta) \triangleq \frac{1}{Hd}\mathbb{E}\big[\|\Delta f_c(\Delta\theta)\|_2^2\big] \geq 0. \tag{12}$$

Equation 11 holds algebraically for arbitrary architectures f_θ without Taylor approximation. When f_θ has a locally Lipschitz parameter Jacobian $J_c = \nabla_\theta f_\theta(x_c)|_{\theta_{\mathrm{ERM}}}$, Taylor's theorem, with integrable bounds on the residual-weighted remainder, implies:

$$B_{\mathrm{disp}}(\Delta\theta) = \frac{2}{Hd}\mathbb{E}\big[e_c^\top J_c \Delta\theta\big] + \mathcal{O}(\|\Delta\theta\|_2^2), \qquad S_{\mathrm{disp}}(\Delta\theta) = \frac{1}{Hd}\Delta\theta^\top \mathbb{E}\big[J_c^\top J_c\big]\Delta\theta + \mathcal{O}(\|\Delta\theta\|_2^3).$$

Along a fixed direction $\Delta\theta = tv$, the leading residual-coupling term scales as t and the squared prediction change as t^2. An anchor can also change direction, so we evaluate its risk response using both measured terms.

Formal Statement and Proof of Proposition 1.

Proposition 1 (Local Curvature-Selective Contraction)**.** *Let $\hat\theta_{\mathrm{DGA}_0}$ be a strict local minimizer of $\mathcal{L}_{\mathrm{DGA}_0}(\theta)$ with positive-definite Hessian $H_{\mathrm{DGA}_0} = \nabla^2\mathcal{L}_{\mathrm{DGA}_0}|_{\hat\theta_{\mathrm{DGA}_0}} \succeq \mu I$ for $\mu > 0$. Assume the ERM reference is sufficiently close to this minimizer and the connected stationary branch remains within a strongly convex neighborhood $\mathcal{N}$ on which $\mathcal{L}_{\mathrm{DGA}_0}$ has L_H-Lipschitz Hessian. Then, for any bounded anchor penalty $\lambda \in [0, \bar\lambda]$, the stationary branch $\hat\theta_{\mathrm{DGA}}(\lambda)$ continuously connected to $\hat\theta_{\mathrm{DGA}_0}$ satisfies:*

$$\Delta\theta_\lambda = (H_{\mathrm{DGA}_0} + 2\lambda I)^{-1} H_{\mathrm{DGA}_0}\, \Delta\theta_0 + \mathcal{O}_{\bar\lambda}\big(\|\Delta\theta_0\|_2^2\big), \tag{13}$$

where $\Delta\theta_\lambda = \hat\theta_{\mathrm{DGA}}(\lambda) - \theta_{\mathrm{ERM}}$ and $\Delta\theta_0 = \hat\theta_{\mathrm{DGA}_0} - \theta_{\mathrm{ERM}}$. Along each Hessian eigendirection v_i with eigenvalue $\mu_i > 0$, the displacement contracts according to:

$$v_i^\top \Delta\theta_\lambda \approx \frac{\mu_i}{\mu_i + 2\lambda}\big(v_i^\top \Delta\theta_0\big). \tag{14}$$

Proof. The stationary point $\hat\theta_{\mathrm{DGA}}(\lambda)$ satisfies $\nabla_\theta\mathcal{L}_{\mathrm{DGA}_0}(\hat\theta_{\mathrm{DGA}}(\lambda)) + 2\lambda(\hat\theta_{\mathrm{DGA}}(\lambda) - \theta_{\mathrm{ERM}}) = 0$. Linearizing around $\hat\theta_{\mathrm{DGA}_0}$, where $\nabla_\theta\mathcal{L}_{\mathrm{DGA}_0}(\hat\theta_{\mathrm{DGA}_0}) = 0$, gives

$$H_{\mathrm{DGA}_0}\big(\hat\theta_{\mathrm{DGA}}(\lambda) - \hat\theta_{\mathrm{DGA}_0}\big) + \mathcal{R}_H + 2\lambda\,\Delta\theta_\lambda = 0,$$

where $\|\mathcal{R}_H\|_2 \leq \frac{L_H}{2}\|\hat\theta_{\mathrm{DGA}}(\lambda) - \hat\theta_{\mathrm{DGA}_0}\|_2^2$. Substituting $\hat\theta_{\mathrm{DGA}}(\lambda) - \hat\theta_{\mathrm{DGA}_0} = \Delta\theta_\lambda - \Delta\theta_0$:

$$(H_{\mathrm{DGA}_0} + 2\lambda I)\,\Delta\theta_\lambda = H_{\mathrm{DGA}_0}\,\Delta\theta_0 - \mathcal{R}_H.$$

Since $H_{\mathrm{DGA}_0} \succeq \mu I$ and $\lambda \geq 0$, $(H_{\mathrm{DGA}_0} + 2\lambda I)$ is invertible with $\|(H_{\mathrm{DGA}_0} + 2\lambda I)^{-1}\|_2 \leq \frac{1}{\mu}$. Multiplying by $(H_{\mathrm{DGA}_0} + 2\lambda I)^{-1}$ and applying the implicit function theorem establishes Equation 13. The spectral decomposition $H_{\mathrm{DGA}_0} = \sum_i \mu_i v_i v_i^\top$ yields the coordinate shrinkage factor in Equation 14. □

Mechanism Interpretation. Proposition 1 describes the local stationary response to the Euclidean anchor. Directions with larger curvature of the zero-anchor objective retain a greater fraction of their displacement; flatter directions contract more strongly. Prediction changes additionally depend on J_c. The matched paths in Appendix E.4 measure how these changes affect residual correction and squared prediction movement.

B.2 Proof of Asymptotic Polarity Expansion (Theorem 1)

This subsection proves the asymptotic polarity expansion in projected linear regression under M-estimation regularity conditions. The expansion is sequential: $n \to \infty$ at fixed ϵ, followed by $\epsilon \to 0$ on population quantities. Write $z = U_L^\top \mathrm{vec}(x) \in \mathbb{R}^q$ and projector $P_L = U_L U_L^\top$. The projected model is $f = w_L^\top z$, residual $r = y - w_L^\top z$, and mixed ℓ_2 adversarial objective $L_\beta = \mathbb{E}[r^2 + 2\beta\epsilon|r|\|w_L\|_2 + \beta\epsilon^2\|w_L\|_2^2]$ for $\beta \in (0, 1]$.

Formal Assumptions.

- **A1 (Forecastability):** Omitted high frequencies contain no additional predictive information for squared loss: $\mathbb{E}[y \mid x] = \mathbb{E}[y \mid P_L \mathrm{vec}(x)]$.
- **A2′ (HAC Regularity):** Stationary α-mixing with $\sum_k \alpha(k)^{\delta/(2+\delta)} < \infty$ and $L^{2+\delta}$ score envelopes; long-run covariance uses a HAC estimator (Newey & West, 1987).
- **A3 (M-estimator Regularity):** $H_0 \succ 0$, $\|w_0^*\| \geq a_{\min} > 0$, continuous conditional residual density $f_{r|z}(0|z)$, local identification, and Bahadur expansion $\hat{w}_\epsilon - w_\epsilon^* = n^{-1}\sum_t \mathrm{IF}_{\epsilon,t} + o_p(n^{-1/2})$ with uniform integrability of $n\|\hat{w}_\epsilon - w_\epsilon^*\|^2$ and locally Lipschitz population derivatives and covariance derivatives along w_ϵ^* (Pakes & Pollard, 1989; Andrews, 1994).
- **A4′ (Curvature Dominance):** $V_1 = \mathcal{K}_{\mathrm{curv}} - \mathcal{I}_{\mathrm{score}} > 0$, where $\mathcal{K}_{\mathrm{curv}} = \mathrm{tr}[\Sigma_{\mathrm{test}}(H_0^{-1}H_1C_0 + C_0H_1H_0^{-1})]$ and $\mathcal{I}_{\mathrm{score}} = \mathrm{tr}[\Sigma_{\mathrm{test}}H_0^{-1}\Omega_1H_0^{-1}]$. Also assumes $\mathbb{E}[\mathrm{sign}(r)z] = 0$ at w_0^*. This curvature condition is separate from the gradient-frequency diagnostic r_g.
- **A5′ (Distributional Shift):** w_0^* and w^* denote training- and test-population minimizers; $d_0 = w_0^* - w^*$ represents the pseudo-true parameter shift (White, 1982).

Scores and covariance. For $r_t = y_t - w^\top z_t$, the clean score is $\psi_{0,t} = -2r_tz_t$, and the first perturbation score is

$$\psi_{1,t} = -2\,\mathrm{sign}(r_t)\|w\|_2 z_t + 2|r_t|\, w/\|w\|_2.$$

Thus $\psi_{\epsilon,t} = \psi_{0,t} + \beta\epsilon\psi_{1,t} + 2\beta\epsilon^2 w$. Define $H_\epsilon = \mathbb{E}[\nabla_w\psi_{\epsilon,t}]$ and the long-run score covariance $\Omega_\epsilon = \sum_{k\in\mathbb{Z}}\mathrm{Cov}(\psi_{\epsilon,t}, \psi_{\epsilon,t+k})$, evaluated at w_ϵ^*. The sandwich covariance is $C_\epsilon = H_\epsilon^{-1}\Omega_\epsilon H_\epsilon^{-1}$, with $H_0 = 2\Sigma_z$ and $\Sigma_z = \mathbb{E}[zz^\top]$. Differentiation along the population optimum gives

$$C_1 = -H_0^{-1}H_1C_0 - C_0H_1H_0^{-1} + H_0^{-1}\Omega_1H_0^{-1}.$$

The population risk is $R(w) = \mathbb{E}_{\mathrm{test}}[(y - w^\top z)^2]$, with $\Sigma_{\mathrm{test}} = \mathbb{E}_{\mathrm{test}}[zz^\top]$.

Step 1 (H_1): Let $a = \|w_L\|_2$, $u = w_L/a$, and $g = |r|a$. Direct differentiation yields $\nabla^2 g = -\mathrm{sign}(r)(zu^\top + uz^\top) + \frac{|r|}{a}(I - uu^\top) + 2a\delta_0(r)zz^\top$. Thus $H_1 = 2\mathbb{E}[\nabla^2 g]_{w_0^*} = \frac{2\mathbb{E}|r|}{a}(I - uu^\top) + 4a\mathbb{E}[f_{r|z}(0|z)zz^\top] \succeq 0$.

Step 2 (Ω_1, HAC): Let $q_t = \psi_{1,t} + A_tb_1$ where $b_1 = -H_0^{-1}\mathbb{E}[\psi_1]$ and $A_t = 2z_tz_t^\top$. Then $\Omega_1 = \sum_k[\mathrm{Cov}(q_t, \psi_{0,t+k}) + \mathrm{Cov}(\psi_{0,t}, q_{t+k})]$ evaluated at w_0^*, converging absolutely under A2′-A3.

Step 3 (Sequential Expansion): For fixed ϵ, as $n \to \infty$, asymptotic sandwich risk is $\mathcal{R}_n^{\mathrm{asy}}(\epsilon) = R(w_\epsilon^*) + n^{-1}\mathrm{tr}(\Sigma_{\mathrm{test}}C_\epsilon)$. Taking $\epsilon \to 0$, implicit differentiation yields $w_\epsilon^* = w_0^* + \beta\epsilon b_1 + \mathcal{O}(\epsilon^2)$, sandwich covariance $C_\epsilon = C_0 + \beta\epsilon C_1 + \mathcal{O}(\epsilon^2)$, and test risk $R(w_\epsilon^*) - R(w_0^*) = \beta\epsilon D_1 + \mathcal{O}(\epsilon^2)$ with $D_1 = 2d_0^\top\Sigma_{\mathrm{test}}b_1$. Combining yields Theorem 1:

$$\mathrm{Pol}_n^{\mathrm{asy}}(\epsilon) = \beta\epsilon\left(\frac{V_1}{n} - D_1\right) + \mathcal{O}(\epsilon^2) + \mathcal{O}(\epsilon^2/n), \qquad V_1 = -\mathrm{tr}(\Sigma_{\mathrm{test}}C_1).$$

This proves the sandwich-risk expansion. The next two results specify its expected-risk correction and a class with positive covariance contraction.

Expected-risk correction. Let $u_\epsilon = \hat{w}_\epsilon - w_\epsilon^*$. Quadratic test risk gives

$$\mathbb{E}R(\hat{w}_\epsilon) = R(w_\epsilon^*) + 2(w_\epsilon^* - w^*)^\top\Sigma_{\mathrm{test}}\mathbb{E}u_\epsilon + \mathrm{tr}\left[\Sigma_{\mathrm{test}}\mathbb{E}(u_\epsilon u_\epsilon^\top)\right].$$

Additionally assume $\mathbb{E}u_\epsilon = a_\epsilon/n + o(n^{-1})$ and $a_\epsilon = a_0 + \beta\epsilon a_1 + O(\epsilon^2)$ (Newey & McFadden, 1994). Subtracting the clean expression gives

$$\mathbb{E}R(\hat{w}_0) - \mathbb{E}R(\hat{w}_\epsilon) = \beta\epsilon\left(\frac{V_1 - Q}{n} - D_1\right) + O(\epsilon^2) + O(\epsilon^2/n) + o(n^{-1}),$$
$$Q = 2b_1^\top \Sigma_{\text{test}} a_0 + 2d_0^\top \Sigma_{\text{test}} a_1.$$

The $o(n^{-1})$ remainder is at fixed ϵ. Theorem 1 concerns the sandwich approximation; this additional mean-bias expansion gives the corresponding expected-risk coefficient.

Corollary 1 (Covariance contraction under correct Gaussian specification)**.** *Assume the regularity conditions above, correctly specified regression with a nonzero coefficient, Gaussian windows $z_t \sim \mathcal{N}(0, \Sigma_z)$, $\Sigma_z \succ 0$, and a zero-mean white Gaussian residual process independent of the entire input process. If $q \geq 2$ and $\Sigma_{\text{test}} \succ 0$, then $V_1 > 0$.*

Proof. Write $a = \|w_0^*\|_2$, $u = w_0^*/a$, $P_\perp = I - uu^\top$, $s_r^2 = \mathbb{E}r_t^2$, $m_1 = \mathbb{E}|r_t|$, and $f_0 = f_r(0)$. Gaussian residuals give $m_1 = 2s_r^2 f_0$. Independence and zero residual means make the nonzero-lag terms in Ω_0 and Ω_1 vanish, even when the input windows are dependent. Hence

$$C_0 = s_r^2 \Sigma_z^{-1}, \qquad H_1 = \frac{2m_1}{a} P_\perp + 4af_0\Sigma_z, \qquad \Omega_1 = 8am_1\Sigma_z.$$

In C_1, the Σ_z-proportional curvature contribution cancels the score-covariance contribution, leaving

$$C_1 = -\frac{2m_1 s_r^2}{a}\Sigma_z^{-1} P_\perp \Sigma_z^{-1}, \qquad V_1 = \frac{2m_1 s_r^2}{a}\text{tr}(MP_\perp), \quad M = \Sigma_z^{-1}\Sigma_{\text{test}}\Sigma_z^{-1}.$$

Since $M \succ 0$ and $P_\perp$ has rank $q - 1 \geq 1$, the trace is positive. □

This establishes the covariance channel for arbitrary positive-definite input covariance. Under general dependent residuals, A4′ states the required curvature–score balance. The expected-risk correction Q is specified separately above.

B.3 Projected Subspace Gradient Frequency

Table 6 connects the questions addressed by the projected theory to finite-network measurements and interventions.

Table 6: Formal quantities, neural measurements, and validation tests.

Aspect	Formal Theory	Neural Experiments	Validation Bridge
Model	Projected linear M-estimator	8 modern forecasters	Within-iTransformer width sweep
Objective	Mixed DGA_0 loss, $\beta \in (0, 1]$	$\beta = 0.5$	Multi-setting β-sweep (Table 33)
Dependence	Stationary mixing, HAC	Overlapping TS windows	Block-bootstrap variance (Table 24)
Perturbation	Exact ℓ_2 direction	3-step ℓ_∞ direction	Spectrum and norm controls
Regime	Sequential limit (n then ϵ)	Finite ϵ	Budget sweeps (Table 18)
Population shift	$d_0^\top \Sigma_{\text{test}} b_1$	Chronological test split	Width sweep leaves data split fixed
Learned drift	$w_\epsilon^* - w_0^* = \beta\epsilon b_1$	$\Delta\theta_\lambda \mapsto B_{\text{disp}} + S_{\text{disp}}$	Matched DGA displacement path

Proposition 2 (Gradient Frequency in the Projected Model)**.** *In the projected linear model $f = w_L^\top z$ ($z = U_L^\top \text{vec}(x)$), the loss gradient $\nabla_{\tilde{x}}\mathcal{L}$ lies in $\text{col}(U_L)$ whenever the gradient is nonzero at w_0^*, so $r_g(\nabla_{\tilde{x}}\mathcal{L}) = 1$. Consequently, the exact ℓ_2 perturbation $\delta^* = \epsilon\nabla_x\mathcal{L}/\|\nabla_x\mathcal{L}\|_2$ lies strictly within the low-frequency predictive subspace.*

Proof. For scalar squared loss, $\nabla_{\tilde{x}}\mathcal{L} = -2r\, U_L w_L$. Both this vector and its normalized ℓ_2 direction belong to $\text{col}(U_L)$. The projection fixes their frequency support. □

C Detailed Forecasting Benchmarks and Significance Analysis

C.1 Validation-Selected Anchor Strengths

Table 7 presents the complete distribution of validation-selected displacement penalties $\lambda^\star$ across all 56 forecasting configurations (8 models $\times$ 7 datasets at $H = 96$). The candidate grid is $\Lambda = \{0, 0.01, 0.1, 1.0, 10\}$, where $\lambda = 0$ is the unanchored endpoint DGA_0.

Table 7: **Validation-selected anchor penalty $\lambda^\star$ across all 56 forecasting configurations.** Structured into two tiers: expressive nonlinear forecasters (top) and compact linear baselines (bottom). Zero penalty ($\lambda^\star = 0$) retains unconstrained adversarial training; positive values activate displacement control.

Model Class / Architecture	ETTh1	ETTh2	ETTm1	ETTm2	Weather	ECL	Traffic
Expressive Nonlinear Forecasters (35 configurations)							
iTransformer	0	0	0	0	0	0.1	0.01
PatchTST	0	0	0.1	0	1	1	10
PaiFilter	0	0.01	0.01	0	0.1	0.01	0.01
FACT	0.01	0.01	0.01	0.01	0.01	0.01	0.01
TimePro	0	0	0	0	0	0.01	0.01
Compact Linear Forecasters (21 configurations)							
FITS	0	10	10	10	1	0.1	1
SparseTSF	0.01	0	10	0.1	1	1	1
DLinear	0	0.1	0.1	1	1	0.1	1

Capacity Dissection and Rescue Rates. Across the 56-cell grid:

1. **Prevalence of Strong Anchors in Linear Baselines (18/21 cells, 85.7%):** Compact linear models widely degrade under unanchored training ($\lambda = 0$) due to parameter drift (Table 11). Validation activates positive anchors in 18 of 21 configurations, with 17 selecting $\lambda^\star \in [0.1, 10]$. DGA outperforms DGA_0 in **17 of these 18 settings** (94.4% improvement rate relative to DGA_0), including a 3.70% clean-MSE reduction relative to ERM on DLinear/ETTh2.
2. **Concentration of Zero Anchors in Expressive Models (15/18 zero-anchors):** Expressive architectures account for 15 of the 18 $\lambda^\star = 0$ selections (83.3%), with iTransformer and TimePro retaining $\lambda^\star = 0$ across all ETT and Weather benchmarks. When positive anchors are selected for nonlinear models (primarily on high-dimensional Traffic with $d = 862$ and ECL with $d = 321$), validation overwhelmingly favors minimal damping ($\lambda^\star = 0.01$ in 14 of 20 cases).

C.2 Complete 56-Setting Benchmark Tables

The complete three-way performance comparison across all 56 forecasting cells (clean MSE mean $\pm$ std over 5 seeds) is reported in Table 9. MAE evaluations at the MSE-selected anchor across all 56 settings appear in Table 10. Table 8 highlights the clean MSE degradations of standard unanchored training on DLinear.

Table 8: DGA_0 on DLinear (5 seeds, clean MSE, mean$\pm$std). The unanchored endpoint degrades clean accuracy on ETTh2, ETTm1, and ETTm2.

Model	Dataset	ERM	DGA_0	DGA_0 vs ERM
DLinear	ETTh1	0.3905$\pm$0.0019	0.3891$\pm$0.0024	-0.4%
DLinear	ETTh2	0.3048$\pm$0.0103	0.3057$\pm$0.0062	$+0.3\%$
DLinear	ETTm1	0.3447$\pm$0.0011	0.3483$\pm$0.0010	$+1.0\%$
DLinear	ETTm2	0.1843$\pm$0.0013	0.1848$\pm$0.0011	$+0.3\%$

Table 9: DGA results on all 56 forecasting cells (5 paired seeds, clean MSE). DGA_0 has zero anchor penalty; $\mathrm{DGA}_{\lambda^\star}$ is selected by validation MSE. Bold marks a selected mean below paired ERM; † marks Holm-adjusted $p < 0.05$ for $\mathrm{DGA}_{\lambda^\star}$ <ERM across all 56 paired tests.

Model	Data	$\lambda^\star$	ERM	DGA_0	$\mathrm{DGA}_{\lambda^\star}$	Δ vs ERM
iTransformer	ETTh1	0	0.3819±0.0014	0.3757±0.0018	**0.3757**±0.0018	$-1.62^{\dagger}\%$
iTransformer	ETTh2	0	0.3010±0.0027	0.2922±0.0010	**0.2922**±0.0010	$-2.92^{\dagger}\%$
iTransformer	ETTm1	0	0.3322±0.0039	0.3161±0.0015	**0.3161**±0.0015	$-4.86^{\dagger}\%$
iTransformer	ETTm2	0	0.1832±0.0009	0.1706±0.0011	**0.1706**±0.0011	$-6.89^{\dagger}\%$
iTransformer	Weather	0	0.1679±0.0026	0.1630±0.0009	**0.1630**±0.0009	-2.96%
iTransformer	ECL	0.1	0.1567±0.0008	0.1565±0.0005	**0.1561**±0.0005	-0.40%
iTransformer	Traffic	0.01	0.4606±0.0014	0.4607±0.0013	**0.4590**±0.0016	-0.34%
PatchTST	ETTh1	0	0.3829±0.0009	0.3789±0.0041	**0.3789**±0.0041	-1.06%
PatchTST	ETTh2	0	0.2949±0.0019	0.2894±0.0040	**0.2894**±0.0040	-1.88%
PatchTST	ETTm1	0.1	0.3285±0.0036	0.3241±0.0038	**0.3233**±0.0017	-1.59%
PatchTST	ETTm2	0	0.1760±0.0009	0.1739±0.0011	**0.1739**±0.0011	-1.21%
PatchTST	Weather	1	0.1642±0.0009	0.1652±0.0018	0.1646±0.0004	$+0.25\%$
PatchTST	ECL	1	0.1721±0.0011	0.1721±0.0011	**0.1712**±0.0008	-0.52%
PatchTST	Traffic	10	0.5180±0.0022	0.5177±0.0020	**0.5140**±0.0019	$-0.77^{\dagger}\%$
PaiFilter	ETTh1	0	0.3883±0.0055	0.3772±0.0035	**0.3772**±0.0035	-2.85%
PaiFilter	ETTh2	0.01	0.2976±0.0049	0.2923±0.0022	**0.2901**±0.0027	-2.51%
PaiFilter	ETTm1	0.01	0.3319±0.0048	0.3232±0.0046	**0.3230**±0.0031	-2.68%
PaiFilter	ETTm2	0	0.1764±0.0013	0.1731±0.0006	**0.1731**±0.0006	-1.89%
PaiFilter	Weather	0.1	0.1672±0.0010	0.1684±0.0021	0.1675±0.0014	$+0.21\%$
PaiFilter	ECL	0.01	0.1674±0.0008	0.1672±0.0007	**0.1666**±0.0014	-0.48%
PaiFilter	Traffic	0.01	0.4941±0.0010	0.4938±0.0012	**0.4917**±0.0011	$-0.48^{\dagger}\%$
FACT	ETTh1	0.01	0.4163±0.0111	0.3984±0.0096	**0.3956**±0.0070	-4.97%
FACT	ETTh2	0.01	0.3033±0.0030	0.2954±0.0050	**0.2930**±0.0025	-3.39%
FACT	ETTm1	0.01	0.3341±0.0037	0.3284±0.0038	**0.3210**±0.0016	$-3.92^{\dagger}\%$
FACT	ETTm2	0.01	0.1819±0.0020	0.1769±0.0066	**0.1721**±0.0020	$-5.37^{\dagger}\%$
FACT	Weather	0.01	0.1623±0.0026	0.1626±0.0024	**0.1586**±0.0017	-2.26%
FACT	ECL	0.01	0.1370±0.0005	0.1374±0.0006	**0.1360**±0.0009	-0.71%
FACT	Traffic	0.01	0.4357±0.0024	0.4378±0.0048	**0.4311**±0.0023	$-1.06^{\dagger}\%$
TimePro	ETTh1	0	0.3797±0.0023	0.3721±0.0006	**0.3721**±0.0006	$-2.01^{\dagger}\%$
TimePro	ETTh2	0	0.2987±0.0029	0.2933±0.0022	**0.2933**±0.0022	$-1.82^{\dagger}\%$
TimePro	ETTm1	0	0.3306±0.0022	0.3215±0.0036	**0.3215**±0.0036	-2.74%
TimePro	ETTm2	0	0.1779±0.0005	0.1723±0.0005	**0.1723**±0.0005	$-3.16^{\dagger}\%$
TimePro	Weather	0	0.1686±0.0011	0.1658±0.0032	**0.1658**±0.0032	-1.68%
TimePro	ECL	0.01	0.1559±0.0024	0.1559±0.0017	**0.1555**±0.0026	-0.24%
TimePro	Traffic	0.01	0.4563±0.0019	0.4560±0.0019	**0.4550**±0.0019	$-0.30^{\dagger}\%$
FITS	ETTh1	0	0.3832±0.0002	0.3842±0.0002	**0.3812**±0.0002	$-0.53^{\dagger}\%$
FITS	ETTh2	10	0.3392±0.0020	0.3441±0.0020	**0.3373**±0.0021	-0.55%
FITS	ETTm1	10	0.3452±0.0009	0.3502±0.0010	**0.3451**±0.0009	-0.03%
FITS	ETTm2	10	0.1931±0.0014	0.1957±0.0011	**0.1917**±0.0010	-0.73%
FITS	Weather	1	0.1958±0.0004	0.1957±0.0004	0.1959±0.0003	$+0.05\%$
FITS	ECL	0.1	0.1955±0.0000	0.1955±0.0000	**0.1954**±0.0000	-0.00%
FITS	Traffic	1	0.6526±0.0003	0.6525±0.0003	**0.6522**±0.0005	-0.06%
SparseTSF	ETTh1	0.01	0.3899±0.0067	0.3903±0.0068	**0.3878**±0.0065	-0.54%
SparseTSF	ETTh2	0	0.3053±0.0013	0.3048±0.0012	**0.3048**±0.0012	$-0.14^{\dagger}\%$
SparseTSF	ETTm1	10	0.3857±0.0017	0.3883±0.0019	0.3857±0.0015	$+0.01\%$
SparseTSF	ETTm2	0.1	0.2062±0.0004	0.2063±0.0004	**0.2062**±0.0003	-0.02%
SparseTSF	Weather	1	0.1964±0.0012	0.1963±0.0012	**0.1960**±0.0010	-0.18%
SparseTSF	ECL	1	0.2017±0.0001	0.2017±0.0001	**0.2016**±0.0000	-0.04%
SparseTSF	Traffic	1	0.6606±0.0002	0.6605±0.0002	**0.6603**±0.0002	-0.03%
DLinear	ETTh1	0	0.3905±0.0019	0.3891±0.0024	**0.3891**±0.0024	-0.37%
DLinear	ETTh1	0	0.3905±0.0019	0.3907±0.0024	**0.3891**±0.0024	-0.37%
DLinear	ETTh2	0.1	0.3048±0.0103	0.3057±0.0062	**0.2935**±0.0024	-3.70%
DLinear	ETTm1	0.1	0.3447±0.0011	0.3483±0.0010	**0.3426**±0.0012	-0.63%
DLinear	ETTm2	1	0.1843±0.0013	0.1848±0.0011	**0.1823**±0.0007	-1.12%
DLinear	Weather	1	0.1978±0.0023	0.1978±0.0024	0.1995±0.0026	$+0.89\%$
DLinear	ECL	0.1	0.1953±0.0002	0.1953±0.0002	**0.1951**±0.0002	-0.11%
DLinear	Traffic	1	0.6493±0.0009	0.6493±0.0009	**0.6486**±0.0009	-0.12%

D MAE AT THE MSE-SELECTED ANCHOR

Table 10 evaluates the clean-MSE-selected operating point under MAE. The value $\lambda^\star$ remains fixed by validation MSE, and test MAE serves as a transfer metric. Each row compares paired ERM, DGA_0, and the selected $\mathrm{DGA}_{\lambda^\star}$ under the same validation rule.

Table 10: MAE at all 56 validation-selected DGA points (5 paired seeds). $\lambda^\star$ is selected by validation MSE, not test MAE. Bold indicates lower MAE than paired ERM.

Model	Data	$\lambda^\star$	ERM	DGA_0	$\mathrm{DGA}_{\lambda^\star}$	Δ vs ERM
iTransformer	ETTh1	0	0.4017±0.0009	0.3965±0.0010	**0.3965±0.0010**	−1.31%
iTransformer	ETTh2	0	0.3500±0.0019	0.3433±0.0011	**0.3433±0.0011**	−1.92%
iTransformer	ETTm1	0	0.3665±0.0020	0.3549±0.0007	**0.3549±0.0007**	−3.15%
iTransformer	ETTm2	0	0.2667±0.0025	0.2519±0.0006	**0.2519±0.0006**	−5.55%
iTransformer	Weather	0	0.2121±0.0026	0.2076±0.0010	**0.2076±0.0010**	−2.13%
iTransformer	ECL	0.1	0.2523±0.0007	0.2519±0.0004	**0.2512±0.0003**	−0.43%
iTransformer	Traffic	0.01	0.3190±0.0013	0.3189±0.0012	**0.3174±0.0014**	−0.52%
PatchTST	ETTh1	0	0.3999±0.0014	0.3984±0.0014	**0.3984±0.0014**	−0.39%
PatchTST	ETTh2	0	0.3439±0.0013	0.3395±0.0019	**0.3395±0.0019**	−1.30%
PatchTST	ETTm1	0.1	0.3661±0.0028	0.3617±0.0027	**0.3613±0.0008**	−1.32%
PatchTST	ETTm2	0	0.2601±0.0010	0.2561±0.0012	**0.2561±0.0012**	−1.57%
PatchTST	Weather	1	0.2098±0.0009	0.2103±0.0016	0.2100±0.0001	+0.11%
PatchTST	ECL	1	0.2696±0.0019	0.2693±0.0019	**0.2676±0.0018**	−0.73%
PatchTST	Traffic	10	0.3230±0.0004	0.3229±0.0009	**0.3185±0.0015**	−1.38%
PaiFilter	ETTh1	0	0.4025±0.0032	0.3969±0.0016	**0.3969±0.0016**	−1.39%
PaiFilter	ETTh2	0.01	0.3489±0.0056	0.3407±0.0026	**0.3415±0.0030**	−2.14%
PaiFilter	ETTm1	0.01	0.3688±0.0031	0.3601±0.0026	**0.3618±0.0025**	−1.88%
PaiFilter	ETTm2	0	0.2585±0.0016	0.2550±0.0006	**0.2550±0.0006**	−1.37%
PaiFilter	Weather	0.1	0.2121±0.0009	0.2134±0.0022	0.2125±0.0017	+0.19%
PaiFilter	ECL	0.01	0.2577±0.0008	0.2575±0.0008	**0.2569±0.0017**	−0.30%
PaiFilter	Traffic	0.01	0.3291±0.0014	0.3292±0.0013	**0.3260±0.0019**	−0.93%
FACT	ETTh1	0.01	0.4214±0.0049	0.4090±0.0048	**0.4079±0.0047**	−3.21%
FACT	ETTh2	0.01	0.3504±0.0019	0.3454±0.0037	**0.3430±0.0029**	−2.12%
FACT	ETTm1	0.01	0.3686±0.0023	0.3644±0.0032	**0.3593±0.0017**	−2.52%
FACT	ETTm2	0.01	0.2637±0.0021	0.2568±0.0025	**0.2545±0.0018**	−3.48%
FACT	Weather	0.01	0.2079±0.0034	0.2095±0.0028	**0.2040±0.0016**	−1.86%
FACT	ECL	0.01	0.2342±0.0007	0.2347±0.0009	**0.2333±0.0018**	−0.39%
FACT	Traffic	0.01	0.2682±0.0013	0.2683±0.0013	**0.2636±0.0011**	−1.74%
TimePro	ETTh1	0	0.4012±0.0014	0.3954±0.0004	**0.3954±0.0004**	−1.46%
TimePro	ETTh2	0	0.3489±0.0013	0.3424±0.0021	**0.3424±0.0021**	−1.86%
TimePro	ETTm1	0	0.3685±0.0017	0.3597±0.0025	**0.3597±0.0025**	−2.37%
TimePro	ETTm2	0	0.2600±0.0004	0.2538±0.0006	**0.2538±0.0006**	−2.39%
TimePro	Weather	0	0.2122±0.0009	0.2093±0.0026	**0.2093±0.0026**	−1.36%
TimePro	ECL	0.01	0.2507±0.0018	0.2506±0.0010	**0.2502±0.0020**	−0.21%
TimePro	Traffic	0.01	0.3094±0.0011	0.3091±0.0009	**0.3078±0.0008**	−0.52%
FITS	ETTh1	0	0.3949±0.0001	0.3946±0.0001	**0.3946±0.0001**	−0.10%
FITS	ETTh2	10	0.3929±0.0017	0.3969±0.0015	**0.3913±0.0016**	−0.42%
FITS	ETTm1	10	0.3723±0.0006	0.3760±0.0008	0.3724±0.0007	+0.03%
FITS	ETTm2	10	0.2913±0.0020	0.2953±0.0017	**0.2895±0.0015**	−0.61%
FITS	Weather	1	0.2560±0.0009	0.2559±0.0009	0.2565±0.0004	+0.22%
FITS	ECL	0.1	0.2779±0.0003	0.2780±0.0004	0.2782±0.0001	+0.09%
FITS	Traffic	1	0.3992±0.0005	0.3992±0.0005	0.3996±0.0004	+0.08%
SparseTSF	ETTh1	0.01	0.3942±0.0065	0.3939±0.0061	**0.3936±0.0059**	−0.16%
SparseTSF	ETTh2	0	0.3474±0.0005	0.3470±0.0004	**0.3470±0.0004**	−0.10%
SparseTSF	ETTm1	10	0.3935±0.0012	0.3956±0.0013	0.3936±0.0010	+0.02%
SparseTSF	ETTm2	0.1	0.2835±0.0008	0.2838±0.0007	0.2836±0.0005	+0.01%
SparseTSF	Weather	1	0.2365±0.0010	0.2365±0.0009	**0.2361±0.0006**	−0.15%
SparseTSF	ECL	1	0.2781±0.0003	0.2781±0.0003	**0.2779±0.0003**	−0.08%
SparseTSF	Traffic	1	0.3945±0.0010	0.3945±0.0010	0.3945±0.0008	+0.01%
DLinear	ETTh1	0	0.4051±0.0035	0.4064±0.0039	0.4064±0.0039	+0.34%
DLinear	ETTh2	0.1	0.3619±0.0121	0.3632±0.0060	**0.3492±0.0022**	−3.51%
DLinear	ETTm1	0.1	0.3706±0.0033	0.3725±0.0018	0.3715±0.0030	+0.25%
DLinear	ETTm2	1	0.2753±0.0036	0.2750±0.0047	**0.2714±0.0006**	−1.44%
DLinear	Weather	1	0.2618±0.0050	0.2618±0.0050	0.2654±0.0044	+1.38%
DLinear	ECL	0.1	0.2773±0.0012	0.2773±0.0012	**0.2773±0.0012**	−0.00%
DLinear	Traffic	1	0.3983±0.0006	0.3983±0.0006	**0.3977±0.0007**	−0.16%

D.1 Statistical Significance and Multiplicity Correction

For each seed s, the global statistic is $g_s = 56^{-1} \sum_{c=1}^{56} \log(\mathrm{MSE}_{\mathrm{DGA},c,s}/\mathrm{MSE}_{\mathrm{ERM},c,s})$. We apply a one-sided one-sample t-test to the five g_s values against zero (alternative: lower DGA MSE). Each cell instead uses a paired t-test on its five MSE pairs. Holm and BH corrections are applied separately within each comparison family in Table 12.

Leave-one-seed-out (LOSO) selection chooses λ from the other four seeds' mean validation MSE, then evaluates the held-out seed's test predictor. Rotating the held-out seed gives five evaluations per setting. On the 24 settings with complete candidate records, LOSO lowers MSE relative to ERM in 20 settings and matches the all-seed anchor in 104/120 folds.

Table 11 details paired statistics across the 8 negative-polarity linear ETT settings, showing that $\mathrm{DGA}_{\lambda^\star}$ significantly improves over DGA_0 in 7/8 cells and over ERM in 6/8 cells under raw paired tests (retaining 7/8 and 4/8 under Holm correction).

Table 11: Paired statistics on the 8 negative-DGA_0 cells from the ETT subgrid (5 seeds; one-sided $^{**}p < 0.01$, $^{*}p < 0.05$).

Model	Data	$\lambda^\star$	ERM	DGA_0	$\mathrm{DGA}_{\lambda^\star}$	$\mathrm{DGA}_{\lambda^\star} - \mathrm{DGA}_0$ / $-$ERM
DLinear	ETTh2	0.1	0.3048	0.3057	**0.2935**	-3.97^{**} / -3.70^{*}
DLinear	ETTm1	0.1	0.3447	0.3483	**0.3426**	-1.65^{**} / -0.63^{*}
DLinear	ETTm2	1.0	0.1843	0.1848	**0.1823**	-1.37^{*} / -1.12^{*}
FITS	ETTh2	10	0.3392	0.3441	**0.3373**	-1.96^{**} / -0.55^{**}
FITS	ETTm1	10	0.3452	0.3502	**0.3451**	-1.45^{**} / -0.03^{**}
FITS	ETTm2	10	0.1931	0.1957	**0.1917**	-2.05^{**} / -0.73^{*}
SparseTSF	ETTm1	10	0.3857	0.3883	0.3857	-0.67^{**} / $+0.01$
SparseTSF	ETTm2	0.1	0.2062	0.2063	0.2062	-0.05 / -0.02

Table 12 reports cellwise multiplicity-corrected tests. The fixed-grid macro test over all 56 cells $\mathrm{DGA}_{\lambda^\star}$ achieves $p = 1.43 \times 10^{-5}$ vs. ERM, with 15 improvements surviving strict Holm adjustment and 42 surviving Benjamini-Hochberg (BH) FDR control.

Table 12: Multiplicity-corrected significance counts per family of one-sided paired tests (5 seeds).

Family (contrast, scope)	m	Raw ($p < 0.05$)	Holm	BH ($q < 0.05$)
$\mathrm{DGA}_{\lambda^\star}$ vs ERM, all forecasting cells	56	46	15	42
$\mathrm{DGA}_{\lambda^\star}$ vs DGA_0, positive-anchor cells	38	23	6	22
$\mathrm{DGA}_{\lambda^\star}$ vs ERM, matched non-ETT	24	17	6	15
$\mathrm{DGA}_{\lambda^\star}$ vs DGA_0, matched non-ETT	24	15	6	14
LOSO $\mathrm{DGA}_{\lambda^\star}$ vs ERM, matched non-ETT	24	14	6	13
LOSO $\mathrm{DGA}_{\lambda^\star}$ vs DGA_0, matched non-ETT	24	12	6	11
$\mathrm{DGA}_{\lambda^\star}$ vs DGA_0, adverse ETT	8	7	7	7
$\mathrm{DGA}_{\lambda^\star}$ vs ERM, adverse ETT	8	6	4	5

D.2 Multi-Horizon Generalization Results

Table 2 presents full evaluations of iTransformer across $H \in \{96, 192, 336, 720\}$ on all four ETT benchmarks (76 matched seed pairs). Adversarial training achieves lower MSE and MAE in 14/16 comparisons each, jointly in 13/16. Improvements are particularly pronounced at long horizons on ETTm1 (-5.2% MSE at $H = 720$) and ETTm2 (-4.1% MSE at $H = 720$).

D.3 Adversarial Robustness Evaluation

Table 13 evaluates 20-step perturbed input test MSE ($\epsilon = 0.1, \ell_\infty$) across all 8 negative-polarity cells. Selected DGA lowers perturbed MSE in 7 of 8 cells relative to ERM while simultaneously improving clean MSE across all 8 cells compared to DGA_0.

Table 13: Clean and 20-step perturbed input test MSE (ε=0.1, ℓ_∞) for ERM/DGA_0/$\mathrm{DGA}_{\lambda^\star}$ on all 8 negative-polarity cells (5 seeds, mean; per-seed std < 0.012). Each entry reports clean/perturbed MSE (C/P), and ΔP is the perturbed-MSE change. The selected method lowers both metrics on seven cells. SparseTSF/ETTm1 has 0.01% higher clean MSE and ties perturbed input MSE at displayed precision. Against DGA_0, all eight clean means improve.

Cell	ERM (C/P)	DGA_0 (C/P)	$\mathrm{DGA}_{\lambda^\star}$ (C/P)	ΔP/ERM	ΔP/DGA_0
DLinear/ETTh2	0.3048/0.3841	0.3057/0.3680	0.2935/0.3685	-4.1%	$+0.1\%$
DLinear/ETTm1	0.3447/0.4246	0.3483/0.4151	0.3426/0.4207	-0.9%	$+1.4\%$
DLinear/ETTm2	0.1843/0.2516	0.1848/0.2362	0.1823/0.2487	-1.2%	$+5.3\%$
FITS/ETTh2	0.3392/0.4227	0.3441/0.4075	0.3373/0.4203	-0.6%	$+3.1\%$
FITS/ETTm1	0.3452/0.4276	0.3502/0.4153	0.3451/0.4274	0.0%	$+2.9\%$
FITS/ETTm2	0.1931/0.2579	0.1957/0.2488	0.1917/0.2562	-0.7%	$+3.0\%$
SparseTSF/ETTm1	0.3857/0.4488	0.3883/0.4512	0.3857/0.4488	0.0%	-0.5%
SparseTSF/ETTm2	0.2062/0.2544	0.2063/0.2539	0.2062/0.2540	-0.2%	0.0%

E Empirical Mechanism Analysis and Diagnostic Probes

E.1 Predictive Alignment and Matched Perturbation Controls

Spectral Concentration of Adversarial Gradients. Figure 5 and Table 14 confirm that loss gradients have $r_g = 0.71$–0.77 and perturbations have $r_g = 0.65$–0.82 on the four probes, above the 0.490 bin-count reference. Table 15 extends the gradient measurement to six models on ETTh2.

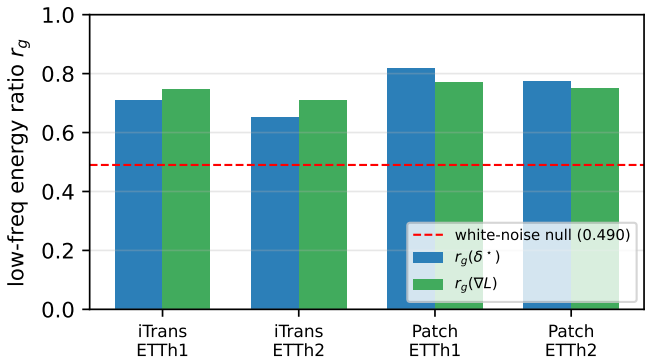

Figure 5: Gradient and perturbation low-frequency ratios r_g exceed the 0.490 white-noise reference.

Table 14: Frequency probe (5-seed mean$\pm$std). r_g is the low-frequency energy ratio (white-noise null ≈ 0.490). Both are low-frequency-enriched; A is the evaluation-MSE increment at a fixed ERM checkpoint; Q_{smooth} is normalized gradient alignment (Appendix E.1).

Model/Data	$r_g(\nabla L)$	$r_g(\delta^\star)$	A_{DGA_0}	$A_{\mathrm{DGA}_0}/A_{\mathrm{rand}}$	Q_{smooth}
iTransformer/ETTh1	0.747$\pm$0.019	0.711$\pm$0.012	0.0986	336$\times$	0.781$\pm$0.013
iTransformer/ETTh2	0.710$\pm$0.047	0.653$\pm$0.026	0.0582	74$\times$	0.638$\pm$0.010
PatchTST/ETTh1	0.771$\pm$0.011	0.819$\pm$0.007	0.0472	129$\times$	0.885$\pm$0.006
PatchTST/ETTh2	0.751$\pm$0.056	0.776$\pm$0.048	0.0597	90$\times$	0.837$\pm$0.011

Table 15: Gradient low-frequency ratio $r_g(\nabla_x \mathcal{L})$ per model (ETTh2). All > 0.490 null.

Model	iTransformer	PatchTST	DLinear	FITS	SparseTSF	PaiFilter
r_g	0.71–0.77	0.751	0.777	0.705	0.959	0.997

Direction vs. Phase Controls. For a fixed ERM checkpoint, A denotes the clean-to-perturbed evaluation MSE increment. With $g = \nabla_x \ell$ and the smoothed perturbation δ_{smooth}, define $Q_{\mathrm{smooth}} = \langle g, \delta_{\mathrm{smooth}} \rangle / (\epsilon \|g\|_1)$ for nonzero g. These are checkpoint sensitivity measurements.

In retraining controls, RAND samples uniform noise on $[-\epsilon, \epsilon]$. The **smooth** rule applies a temporal spectral taper after each sign-gradient step: the lowest half of stored bins retain weight one, and

higher weights decrease linearly to zero, followed by clipping to the input budget. **smoothrand** applies the same taper to uniform noise. The **spec_low** and **spec_high** rules instead update orthonormal Fourier coefficients in their gradient directions, constraining each coefficient magnitude to ϵ/w_k. With $F = L/2 + 1$ bins, the weights are $w_k = 1 + 3k/(F-1)$ and $w_k = 1 + 3(1 - k/(F-1))$, respectively. The **vat** control starts from uniform noise and maximizes squared prediction divergence from a detached clean prediction, then trains with clean loss plus a prediction-consistency penalty. Table 16 compares δ^* against spectrum-matched (identical magnitude spectrum, random phase), boundary-Rademacher, and norm-matched controls: the true gradient direction increases loss 14–39× more than spectrum-matched noise. Table 17 shows that retraining with low-pass filtered adversarial perturbations (**smooth**) preserves clean gains, whereas low-pass random noise (**smoothrand**) tracks standard random augmentation (RAND).

Table 16: Matched-control perturbation ablation (ε=0.1, 5 seeds, A = fixed-ERM-checkpoint MSE increment, 5-seed mean). The DGA_0 perturbation dominates boundary-random and spectrum-matched controls (identical spectrum, random phase). The projected and smoothed DGA_0 controls retain 70–87% of the original loss increase.

Model/Data	DGA_0	Bound.	Norm	Spect.	Proj.	Smooth	DGA_0/Spect.
iTransformer/ETTh1	0.0986	0.0012	0.0011	0.0025	0.0695	0.0667	39.2×
iTransformer/ETTh2	0.0582	0.0024	0.0021	0.0032	0.0418	0.0408	18.4×
PatchTST/ETTh1	0.0472	0.0012	0.0012	0.0034	0.0411	0.0383	14.0×
PatchTST/ETTh2	0.0597	0.0021	0.0020	0.0026	0.0489	0.0481	22.5×

Table 17: Perturbation comparison (5 seeds, clean MSE mean±sample SD). DGA_0 and **smooth** (low-pass zero-penalty update) give the lowest MSE; **smoothrand** tracks RAND. The gain follows the low-frequency loss-gradient direction, not low-frequency augmentation alone.

Model/Data	DGA_0	smooth	spec_low	spec_high	RAND	smoothrand	vat
iTransformer/ETTh1	0.376±0.002	0.376±0.002	0.379±0.002	0.380±0.001	0.383±0.001	0.383±0.001	0.379±0.002
iTransformer/ETTh2	0.292±0.001	0.293±0.001	0.295±0.002	0.295±0.001	0.300±0.003	0.300±0.003	0.295±0.001
PatchTST/ETTh1	0.379±0.004	0.379±0.004	0.379±0.004	0.382±0.004	0.385±0.003	0.385±0.003	0.386±0.005
PatchTST/ETTh2	0.289±0.004	0.290±0.004	0.295±0.001	0.296±0.001	0.297±0.004	0.297±0.004	0.292±0.005
PaiFilter/ETTh2	0.292±0.002	0.292±0.003	0.298±0.004	0.298±0.004	0.297±0.005	0.297±0.005	0.295±0.004
TimePro/ETTh1	0.372±0.001	0.373±0.002	0.375±0.001	0.376±0.002	0.380±0.002	0.380±0.002	0.377±0.002
TimePro/ETTh2	0.293±0.002	0.294±0.002	0.296±0.002	0.296±0.002	0.298±0.003	0.298±0.003	0.294±0.002

Perturbation Budget and Step Robustness. Table 18 sweeps $\epsilon \in \{0.01, 0.03, 0.05, 0.1, 0.2\}$: $\mathrm{DGA}_0 < \mathrm{ERM}$ in 38/40 nonlinear cells. Table 20 shows budget response on DLinear/FITS. Table 19 shows that gains remain stable across inner step counts $K \in \{1, 3, 5, 7, 10\}$.

Table 18: DGA_0 vs. ERM clean-MSE change across ε, with fixed K=3 and 3 matched seeds. The zero-penalty endpoint is lower in 38/40 cells; the two exceptions are $< 0.3\%$.

Model/Data	ε=0.01	0.03	0.05	0.1	0.2
iTransformer/ETTh1	-0.25%	-0.85%	-1.30%	-2.07%	-2.60%
iTransformer/ETTh2	-1.87%	-2.34%	-2.29%	-2.87%	-3.13%
PatchTST/ETTh1	+0.05%	-0.95%	-1.13%	-1.56%	-1.59%
PatchTST/ETTh2	-0.27%	-0.67%	-2.06%	-2.54%	-2.29%
PaiFilter/ETTh1	-0.45%	-0.94%	-0.16%	-1.67%	-0.97%
PaiFilter/ETTh2	-1.11%	+0.29%	-0.72%	-1.37%	-2.00%
TimePro/ETTh1	-0.70%	-1.33%	-1.63%	-2.30%	-2.69%
TimePro/ETTh2	-0.40%	-0.90%	-1.12%	-1.65%	-1.83%

Table 19: DGA_0 clean MSE vs. training-step count K (ε=0.1, 3 seeds, mean). The ERM-relative gain is stable across $K \in \{1, 3, 5, 7, 10\}$.

Model	Dataset	ERM	K=1	K=3	K=5	K=7	K=10	Δ%@K=3
iTransformer	ETTh1	0.3829	0.3796	0.3750	0.3737	0.3736	0.3735	-2.1%
iTransformer	ETTh2	0.3011	0.2944	0.2925	0.2918	0.2914	0.2912	-2.9%
PatchTST	ETTh1	0.3835	0.3797	0.3775	0.3773	0.3772	0.3772	-1.6%
PatchTST	ETTh2	0.2954	0.2933	0.2879	0.2878	0.2895	0.2895	-2.5%
PaiFilter	ETTh1	0.3851	0.3842	0.3787	0.3818	0.3806	0.3823	-1.7%
PaiFilter	ETTh2	0.2954	0.2945	0.2914	0.2953	0.2919	0.2912	-1.4%
TimePro	ETTh1	0.3813	0.3761	0.3725	0.3710	0.3711	0.3710	-2.3%
TimePro	ETTh2	0.2973	0.2944	0.2924	0.2917	0.2914	0.2912	-1.7%

Table 20: ϵ-sensitivity on DLinear/FITS $\times$ ETTh1/ETTh2 (5 seeds). Clean MSE gain vs. ERM.

Cell	$\epsilon = 0.01$	0.03	0.05	0.1	0.2
DLinear/ETTh1 (pos)	-0.09%	-0.24%	-0.37%	-0.37%	-0.44%
DLinear/ETTh2 (neg)	$+0.08\%$	$+0.25\%$	$+0.45\%$	$+0.84\%$	$+3.16\%$
FITS/ETTh1 (pos)	-0.08%	-0.21%	-0.32%	-0.53%	-0.67%
FITS/ETTh2 (neg)	$+0.03\%$	$+0.22\%$	$+0.49\%$	$+1.44\%$	$+4.33\%$

Cross-Loss Validation. Table 21 and Table 22 evaluate adversarial training under MSE, MAE, Huber, and FreDF (Wang et al., 2025), confirming clean gains under alternative loss functions.

Table 21: Matched-loss ablation: ERM vs. DGA_0 under each training loss (iTransformer/ETTh1, 5 seeds).

Training Loss	ERM (matched)	DGA_0	DGA_0 vs ERM
MSE	0.3832	0.3757	-1.9%
MAE	0.3785	0.3732	-1.4%
Huber	0.3812	0.3745	-1.8%
FreDF	0.3753	0.3722	-0.8%

Table 22: Cross-loss $\times$ perturbation (iTransformer, 5 seeds, clean MSE mean$\pm$std).

Dataset	Loss	DGA_0	smooth	RAND	DGA_0 vs RAND
ETTh1	MSE	0.3757$\pm$0.0018	0.3761$\pm$0.0017	0.3834$\pm$0.0008	-2.0%
ETTh1	FreDF	0.3722$\pm$0.0005	0.3722$\pm$0.0005	0.3750$\pm$0.0005	-0.8%
ETTh2	MSE	0.2922$\pm$0.0010	0.2925$\pm$0.0011	0.3001$\pm$0.0028	-2.6%
ETTh2	FreDF	0.2828$\pm$0.0003	0.2827$\pm$0.0003	0.2909$\pm$0.0021	-2.8%
ETTm1	MSE	0.3161$\pm$0.0015	0.3171$\pm$0.0026	0.3312$\pm$0.0030	-4.6%
ETTm1	FreDF	0.3167$\pm$0.0031	0.3160$\pm$0.0030	0.3291$\pm$0.0036	-3.8%

E.2 Prediction-Variance Contraction and Block-Bootstrap Probes

Across-seed probes measure training randomness at a fixed dataset; block resampling measures sensitivity to the dependent observations used for training. Table 23 reports frequency-band ratios, and Table 24 reports total and bandwise ratios over $R = 20$ training resamples. We split the training-window indices into $\lfloor N/96 \rfloor$ nonoverlapping blocks of 96 consecutive windows, sample that many blocks with replacement, and concatenate them; the incomplete trailing block is excluded. ERM and DGA_0 use the same resampled indices.

Mean-prediction error and variance. For S trained predictors, define $\bar{f} = S^{-1} \sum_s f_s$, $E_{\mathrm{mean}} = \mathbb{E}\|\bar{f} - y\|^2/(Hd)$, and $V = S^{-1} \sum_s \mathbb{E}\|f_s - \bar{f}\|^2/(Hd)$. Squared loss gives

$$\overline{R} = E_{\mathrm{mean}} + V, \qquad \Delta\overline{R} = \Delta E_{\mathrm{mean}} + \Delta V = \overline{B}_{\mathrm{disp}} + \overline{S}_{\mathrm{disp}}. \tag{15}$$

Bars on B_{disp} and S_{disp} average the paired-checkpoint terms over seeds. The two decompositions account for the same total risk change. E_{mean} uses observed targets and includes observation noise. For sample variance V_{sample} computed with divisor $S - 1$, the corresponding identity is $\overline{R} = E_{\mathrm{mean}} + (S-1)V_{\mathrm{sample}}/S$. Tables 25 and 30 retain that sample-variance convention; Table 26 uses the additive V.

Frequency-band convention. Prediction spectra use the forecast length H, whereas input-gradient spectra use the lookback length L. The band probes sum unweighted one-sided Fourier-bin variances; total variance is measured in the time domain. Parseval reconstruction weights the interior one-sided bins twice, so the reported band ratios do not directly sum to the time-domain total. The common seed-variance divisor cancels within each ratio.

Table 23: Across-seed prediction-variance probe (5 seeds, DGA_0 vs ERM). Ratios compare DGA_0 with ERM; values below one indicate contraction. The low-frequency 95% bootstrap confidence interval excludes one in 4/6 settings. High-frequency response is setting-dependent: it is near one in several settings and contracts more strongly in others.

Model/Data	Total ratio	Low-freq ratio	95% interval	Low reduction	High-freq ratio
iTransformer/ETTh1	0.653	0.501*	[0.377, 0.583]	49.9%	0.995
iTransformer/Electricity	0.821	0.785*	[0.746, 0.836]	21.5%	0.965
PatchTST/ETTh1	1.102	1.074	[0.688, 1.362]	-7.4%	1.065
PatchTST/Electricity	0.974	0.969*	[0.960, 0.979]	3.1%	0.981
DLinear/ETTh1	0.867	0.785	[0.580, 1.054]	21.5%	1.131
DLinear/Weather	0.970	0.963*	[0.933, 0.999]	3.7%	1.007

Table 24: Block-bootstrap sampling variance ($R = 20$ replicates per cell; iTransformer/ETTh2 validated at $R = 100$). Ratios below 1.0 indicate lower prediction variance for DGA_0 than ERM.

Cell	Regime	Total Ratio	Low-Freq Ratio	High-Freq Ratio	Clean MSE Gain
iTrans/ETTh2	positive	0.65	0.63	0.58	-3.1%
PatchTST/ETTh2	positive	0.76	0.75	0.85	-1.3%
iTrans/ETTh1	positive	0.83	0.75	0.92	-2.2%
DLinear/ETTh2	negative	0.98	0.96	0.91	$+0.8\%$

Table 25: Mean-prediction error and sample variance for DGA_0 on ETTh2 (5 seeds). Shares are E_{mean}/MSE, rounded from the measured values.

Model	Total MSE	E_{mean}	V_{sample}	Mean-error share
DLinear	0.306	0.301	0.006	98.0%
PaiFilter	0.292	0.283	0.011	97.0%
SparseTSF	0.305	0.302	0.003	99.0%
FITS	0.344	0.343	0.001	99.7%

Table 26: Two-channel risk decomposition on ETTh2 (5 seeds). $\widehat{\mathcal{V}} = V_{\mathrm{ERM}} - V_{\mathrm{DGA}_0}$ and $\Delta E_{\mathrm{mean}} = E_{\mathrm{mean,DGA}_0} - E_{\mathrm{mean,ERM}}$. Thus $\Delta\mathrm{MSE} = \Delta E_{\mathrm{mean}} - \widehat{\mathcal{V}}$.

Cell	$\widehat{\mathcal{V}}$	ΔE_{mean}	Polarity	Low-Freq Red.	High-Freq Red.	Net MSE Change
iTrans/ETTh2	$+0.0029$	-0.0059	positive	57.9%	43.5%	-2.92%
DLinear/ETTh2	$+0.0005$	$+0.0007$	negative	11.0%	4.8%	$+0.07\%$

E.3 Capacity Crossing and Norm-Matched Interventions

Table 27 reports norm-matched ℓ_2 adversarial training ($\epsilon_2 = 1.86$, median ℓ_2 norm of ℓ_∞ perturbations) across width extremes $d_{\mathrm{model}} \in \{4, 256\}$. All endpoint polarity signs match the ℓ_∞ sweep, confirming capacity crossing under norm-matched conditions.

Table 27: Norm-matched ℓ_2 replication of the capacity crossing (5 seeds, clean MSE mean$\pm$std).

Dataset	d_{model}	ERM MSE	$\mathrm{DGA}_0(\ell_2)$ MSE	Polarity	Consistent Seeds
ETTh1	4	0.604$\pm$0.033	0.610$\pm$0.034	-0.9%	5/5 negative
ETTh1	256	0.384$\pm$0.001	0.375$\pm$0.002	$+2.3\%$	5/5 positive
ETTm2	4	0.203$\pm$0.001	0.207$\pm$0.001	-1.6%	5/5 negative
ETTm2	256	0.182$\pm$0.002	0.172$\pm$0.001	$+5.5\%$	5/5 positive

E.4 Displacement Path and Weight Interpolation

Parameter Distance and Identity Residuals. Table 28 evaluates the displacement path on Weather checkpoints, showing that the anchor contracts parameter distance by 98.1% on FACT. Across 50 evaluated checkpoints, the algebraic identity residual $|\Delta R - (B_{\text{disp}} + S_{\text{disp}})|$ is at most 1.3×10^{-9} in these evaluations.

Table 28: Displacement path on Weather (5-seed means). Relative distance is $\|\Delta\theta_\lambda\|_2/\|\theta_{\text{ERM}}\|_2$; ΔR is DGA$_\lambda$ minus ERM clean MSE.

Cell	λ	Relative Distance	$B_{\text{disp}}(\lambda)$	$S_{\text{disp}}(\lambda)$	$\Delta R(\lambda)$
FACT/Weather	0	0.8917	-0.01388	0.01418	+0.00029
FACT/Weather	0.01	0.0169	-0.01012	0.00646	-0.00366
iTransformer/Weather	0	0.2628	-0.01216	0.00719	-0.00497
iTransformer/Weather	0.01	0.0180	-0.00284	0.00190	-0.00094

Anchor Sweeps, Objective Controls, and the Checkpoint Path. Training with an anchor changes the fitted solution as well as its distance from ERM. On DLinear/ETTh2, validation selects $\lambda^* = 1$ in the matched sweep (Table 29); this trained solution improves on the best evaluated ERM–DGA$_0$ interpolation point (Figure 6). An independent prediction decomposition shows reductions in both mean-prediction error and sample variance (Table 30). Having established that training with the anchor changes the attainable solution, the loss-mixing and objective controls examine which parts of training contribute to recovery (Tables 33 and 31).

Table 29: λ-sweep on DLinear/ETTh2 (5 seeds, clean MSE mean±std).

λ	Clean MSE	DGA$_\lambda$–ERM %	Optimization Regime
0	0.3057±0.0062	$+0.84\%$	Negative polarity (unanchored)
0.01	0.2998±0.0090	-1.11%	Anchor engages
0.1	0.2936±0.0033	-3.14%	Intermediate anchor
1.0	0.2933±0.0049	$\mathbf{-3.24}\%$	Validation-selected point
10	0.2980±0.0052	-1.71%	Strong anchor

Table 30: Mean-prediction error and sample variance per λ (DLinear/ETTh2, 5 seeds). Low-frequency variance ratios compare DGA$_\lambda$ with paired ERM. This decomposition and the sweep in Table 29 use separate five-seed experiments.

λ	E_{mean}	V_{sample}	var$_{\text{low}}$ Ratio	Total MSE
0	0.3008	0.0060	0.398	0.3057
0.01	0.2951	0.0061	0.482	0.3000
0.1	0.2907	0.0035	0.222	0.2935
1.0	0.2910	0.0044	0.285	0.2945

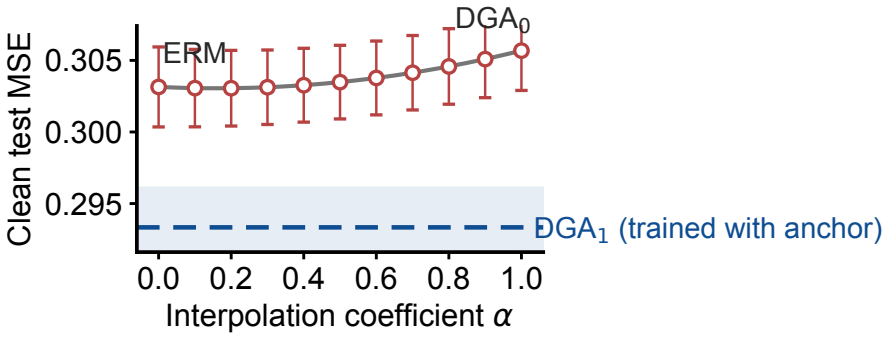

Figure 6: **The trained DGA solution lies below the complete ERM–DGA_0 checkpoint segment on DLinear/ETTh2.** Red points evaluate $\theta_\alpha = (1-\alpha)\theta_{\mathrm{ERM}} + \alpha\theta_{\mathrm{DGA}_0}$ for every $\alpha \in \{0, 0.1, \ldots, 1\}$; error bars are one standard error over five paired seeds, and the red curve is a quadratic fit ($R^2 > 0.9999$). The observed path minimum is 0.30306 at $\alpha = 0.2$ (fitted vertex $\alpha = 0.160$), whereas validation selects the separately trained DGA_1 solution at 0.29334. The 0.00972 MSE gap shows that training with the anchor reaches a better solution than rescaling the displacement along this checkpoint segment.

Table 31: **Matched objective and initialization controls on the eight negative-polarity cells.** Clean test MSE is mean±SD over five paired seeds. Each λ is selected by mean clean validation MSE; the clean-only control searches positive anchors, while both DGA variants also include $\lambda = 0$. Reported p-values are one-sided paired tests against ERM. Bold marks the lowest mean in each row.

Cell	ERM	DGA_0	Cold-start DGA	Clean + anchor	ERM-init. DGA
DLinear/ETTh2	0.3031 ± 0.0062	0.3057 ± 0.0062	$\mathbf{0.2933 \pm 0.0049}$ $\lambda = 1,\ p = .001$ 0.3427 ± 0.0011	0.2934 ± 0.0030 $\lambda = 0.1,\ p = .004$ $\mathbf{0.3426 \pm 0.0011}$	0.2934 ± 0.0049 $\lambda = 1,\ p = .001$ 0.3427 ± 0.0011
DLinear/ETTm1	0.3447 ± 0.0011	0.3483 ± 0.0010	$\lambda = 1,\ p = 3.0 \times 10^{-4}$ $\mathbf{0.1823 \pm 0.0007}$	$\lambda = 1,\ p = 2.5 \times 10^{-4}$ 0.1823 ± 0.0006	$\lambda = 1,\ p = 3.1 \times 10^{-4}$ 0.1823 ± 0.0008
DLinear/ETTm2	0.1843 ± 0.0013	0.1848 ± 0.0011	$\lambda = 1,\ p = .023$ 0.3373 ± 0.0021	$\lambda = 1,\ p = .021$ 0.3371 ± 0.0022	$\lambda = 1,\ p = .024$ $\mathbf{0.3363 \pm 0.0022}$
FITS/ETTh2	0.3392 ± 0.0020	0.3441 ± 0.0020	$\lambda = 10,\ p = .001$ 0.3451 ± 0.0009	$\lambda = 10,\ p = .001$ $\mathbf{0.3446 \pm 0.0009}$	$\lambda = 10,\ p = .012$ 0.3451 ± 0.0009
FITS/ETTm1	0.3452 ± 0.0009	0.3502 ± 0.0010	$\lambda = 10,\ p = .003$ $\mathbf{0.1917 \pm 0.0010}$	$\lambda = 1,\ p = .025$ 0.1917 ± 0.0010	$\lambda = 10,\ p = .001$ 0.1917 ± 0.0010
FITS/ETTm2	0.1931 ± 0.0014	0.1957 ± 0.0011	$\lambda = 10,\ p = .006$ 0.3857 ± 0.0015	$\lambda = 10,\ p = .005$ $\mathbf{0.3852 \pm 0.0015}$	$\lambda = 10,\ p = .007$ 0.3857 ± 0.0015
SparseTSF/ETTm1	0.3857 ± 0.0017	0.3883 ± 0.0019	$\lambda = 10,\ p = .541$ 0.2062 ± 0.0003	$\lambda = 10,\ p = .074$ $\mathbf{0.2061 \pm 0.0003}$	$\lambda = 10,\ p = .449$ 0.2061 ± 0.0002
SparseTSF/ETTm2	0.2062 ± 0.0004	0.2063 ± 0.0004	$\lambda = 0.1,\ p = .256$	$\lambda = 0.1,\ p = .037$	$\lambda = 0.1,\ p = .091$

Across the same eight cells, the zero-penalty endpoint has negative polarity in all eight, while validation-selected cold-start DGA improves seven and reduces mean MSE by 0.79% relative to ERM. The clean-only anchored objective improves all eight by 0.83% on average and attains the lowest row mean in five cells. The clean-only control identifies ERM-centered regularization as a source of recovery in these settings. It complements the direction controls in Appendix E.1: those compare the learning signal supplied by perturbations, whereas this comparison isolates the contribution of anchoring to a fitted reference. Initializing DGA from the fitted ERM checkpoint changes the mean selected MSE by -0.04% relative to cold start and gives lower MSE in three of eight cells. Recovery under both initializations supports treating the ERM solution as a constraint throughout training, rather than requiring it as the starting point.

Table 32: **Single-run training cost.** Peak reserved CUDA memory and mean training time per epoch for one complete ERM run and one run at the model's validation-selected DGA λ on ETTh2 (seed 42, the same otherwise-idle A100, `OMP_NUM_THREADS=16`, and four data-loader workers). Epoch time includes perturbation generation and optimization but excludes validation. Ratios are DGA/ERM.

Model	λ	Peak memory (GiB)			Train seconds/epoch		
		ERM	DGA	Ratio	ERM	DGA	Ratio
DLinear	1	0.02	0.02	1.04×	0.56	2.21	3.98×
iTransformer	0	0.05	0.06	1.13×	2.40	11.55	4.81×
PatchTST	0	0.10	0.17	1.66×	2.99	16.24	5.43×
FACT	0.01	0.54	1.02	2.13×	8.83	34.92	3.95×
PaiFilter	0.01	0.02	0.03	1.07×	0.81	3.55	4.40×
TimePro	0	0.04	0.05	1.28×	8.79	44.56	5.07×
FITS	10	0.02	0.02	1.03×	0.56	2.08	3.68×
SparseTSF	0	0.02	0.02	1.02×	0.54	1.53	2.83×
Geometric mean	–	–	–	1.25×	–	–	4.19×

Across the eight architectures, DGA uses $1.25\times$ the peak reserved GPU memory and $4.19\times$ the training time per epoch on geometric average. Per-model time ratios range from $2.83\times$ to $5.43\times$, consistent with the three inner input-update steps required by each outer optimization step. Memory grows more moderately because DGA_0 needs no reference-model copy; positive-anchor runs retain the frozen ERM checkpoint and reach at most $2.13\times$ in this profile.

Table 33: Mix-weight (β) control on the 8 negative-polarity cells (clean MSE, 5-seed mean; β=0 is ERM, β=0.5 is DGA_0). Best positive β per cell in bold; $\mathrm{DGA}_{\lambda^\star}$ beats the best positive β on all 8 cells, showing that anchoring differs from perturbed-loss mixing.

Model	Dataset	β=0	0.1	0.25	0.5	0.75	1.0	best $\beta > 0$	$\mathrm{DGA}_{\lambda^\star}$
DLinear	ETTh2	0.3048	**0.3038**	0.3039	0.3057	0.3085	0.3107	0.1	0.2935 (0.1)
DLinear	ETTm1	0.3447	**0.3455**	0.3462	0.3483	0.3505	0.3527	0.1	0.3426 (0.1)
DLinear	ETTm2	0.1843	**0.1840**	0.1843	0.1848	0.1856	0.1865	0.1	0.1823 (1)
FITS	ETTh2	0.3392	**0.3397**	0.3412	0.3441	0.3473	0.3507	0.1	0.3373 (10)
FITS	ETTm1	0.3452	**0.3462**	0.3477	0.3502	0.3523	0.3548	0.1	0.3451 (10)
FITS	ETTm2	0.1931	**0.1942**	0.1946	0.1957	0.1971	0.1985	0.1	0.1917 (10)
SparseTSF	ETTm1	0.3857	**0.3862**	0.3870	0.3883	0.3896	0.3909	0.1	0.3857 (10)
SparseTSF	ETTm2	0.2062	**0.2062**	0.2062	0.2063	0.2064	0.2063	0.1	0.2062 (0.1)

E.5 Input-Space Loss Surface Visualizations

Figures 8–9 depict the 2D input-loss surfaces along the gradient direction $\hat{g}$ and orthogonal direction $d_\perp$. Figure 10 quantitatively summarizes the directional curvature flattening in positive-polarity settings and loss recovery in negative-polarity settings. Figure 7 shows the 3D converged surfaces, Figure 11 plots 1D slices across all six diagnostic cells, and Figure 12 shows 3D optimization trajectory flattening.

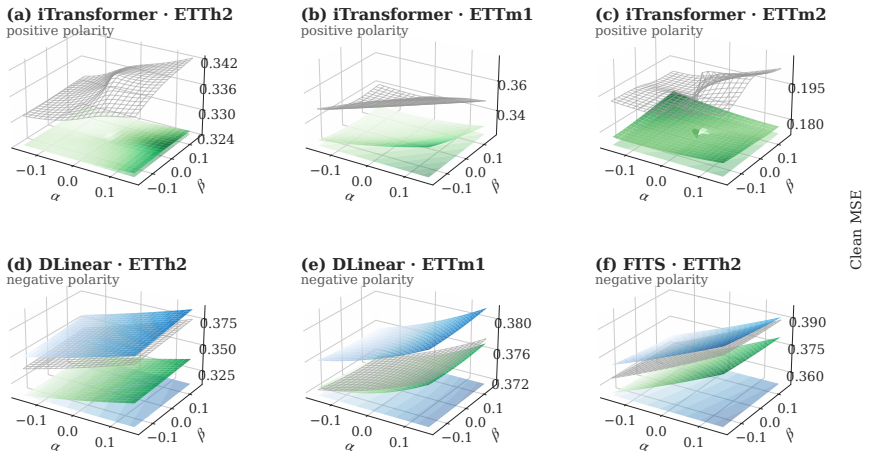

Figure 7: **Three-dimensional input-loss surfaces at convergence.** In positive cells (top), adversarial training flattens the surface along $\hat{g}$. In negative cells (bottom), unanchored training raises loss levels, while DGA_{λ^*} recovers low loss below ERM.

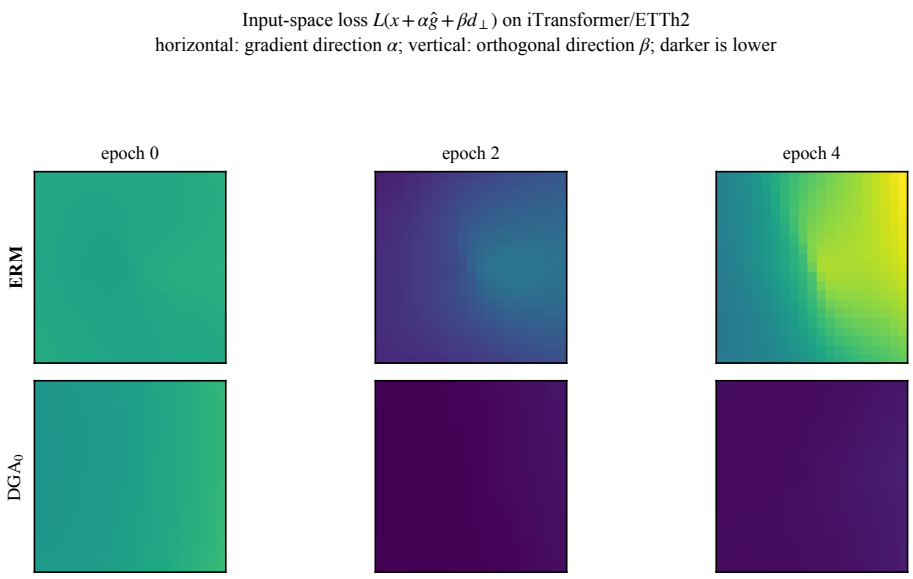

Figure 8: 2D loss surface on iTransformer/ETTh2 across epochs: adversarial training flattens directional curvature.

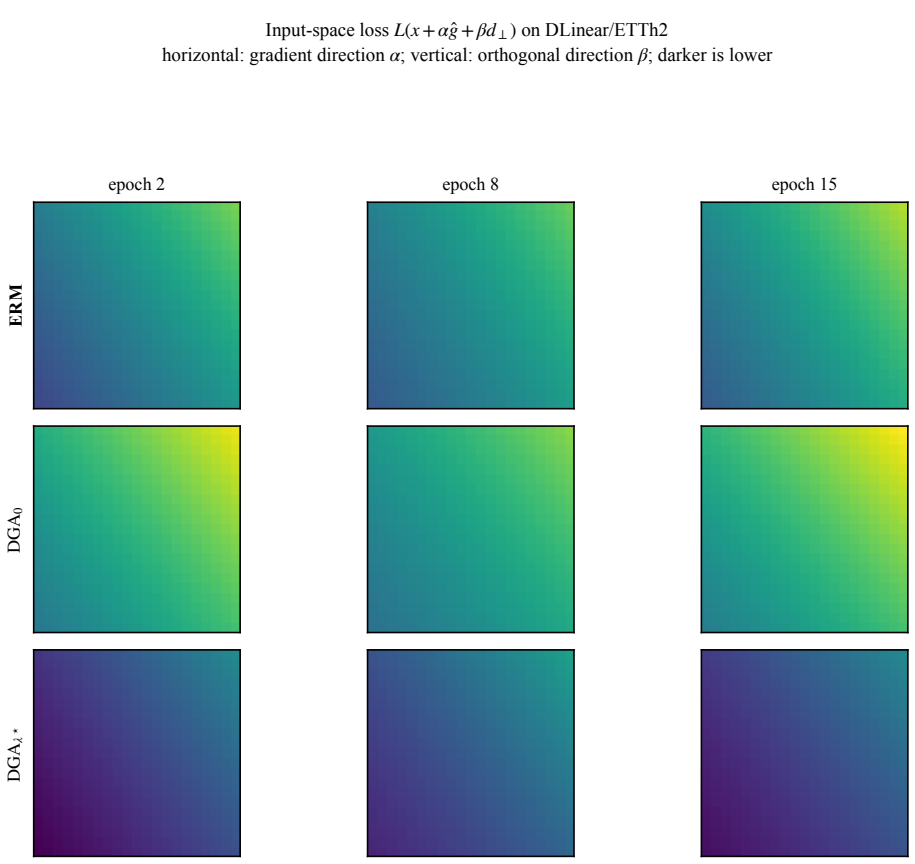

Figure 9: 2D loss surface on DLinear/ETTh2 across epochs: DGA_0 raises loss levels, while DGA_{λ^*} restores them.

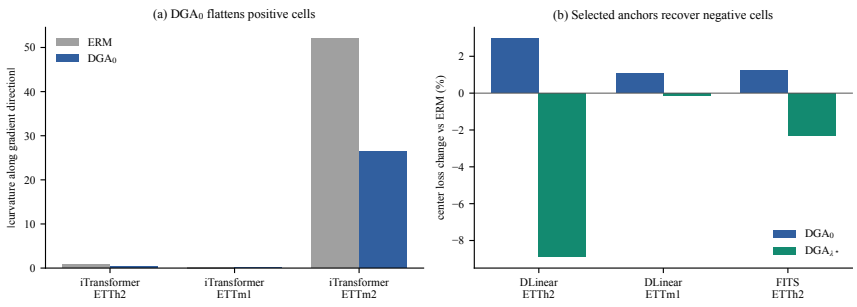

Figure 10: Quantitative curvature and loss metrics across six diagnostic cells.

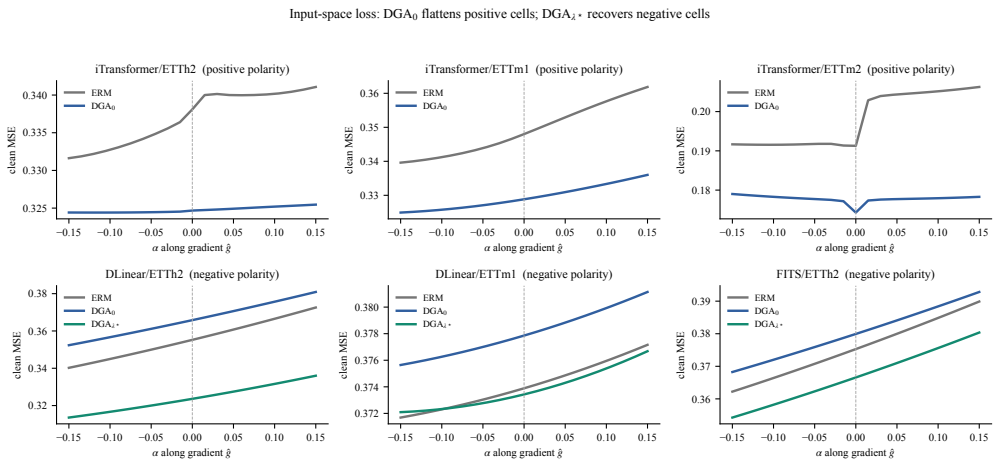

Figure 11: 1D loss slices along gradient direction $\hat{g}$ across all six diagnostic cells.

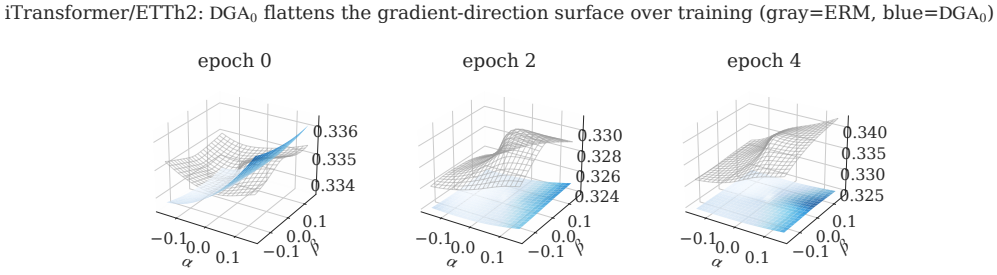

Figure 12: 3D loss-surface evolution across epochs on iTransformer/ETTh2 (gray=ERM, blue=DGA$_0$).

E.6 Synthetic Frequency-Phase Sweeps

Using an MLP forecaster on sinusoidal synthetic series $s(t) = \sin(2\pi ft/96)$, Figure 13 confirms that adversarial clean gains are strictly concentrated at low target frequencies ($f \leq 8$), approaching parity at high frequencies ($f = 32$). Figure 14 explores interaction with depth, noise level, and dataset size.

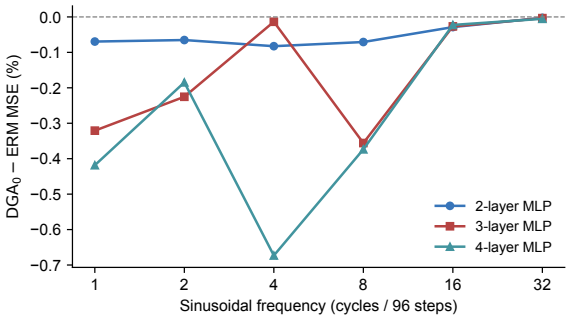

Figure 13: DGA$_0$ clean MSE gain vs. sinusoidal frequency by MLP depth.

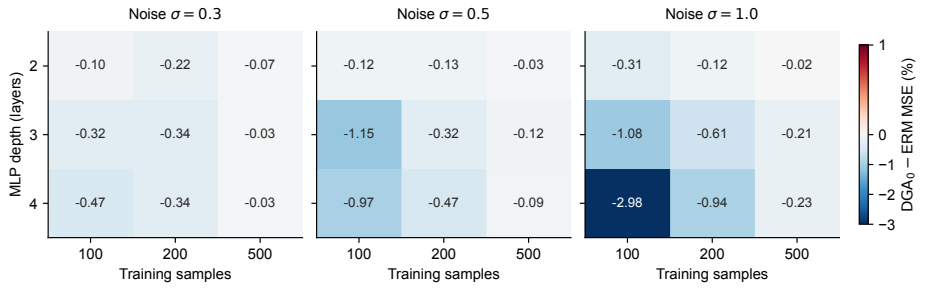

Figure 14: 4D parameter sweep at low target frequency ($f = 2$) across depth, noise, and sample size.

F Cross-Modal Generalization in Language and Vision

This section details the protocols, complete seed results, and statistical tests for cross-modal evaluations in language fine-tuning and computer vision.

F.1 Natural Language Processing: BERT Fine-Tuning on GLUE

Protocol. We fine-tune BERT-base-uncased (Devlin et al., 2019) on three GLUE benchmarks (Wang et al., 2019): SST-2 (sentiment), MRPC (paraphrase), and RTE (textual entailment). A label-stratified 10% training split selects $\lambda \in \{0, 10^{-5}, 10^{-4}, 10^{-3}, 0.01, 0.1, 0.3, 1\}$. Official development sets are evaluated only after selection. The anchor θ_{ERM} is the separately fitted same-seed clean model. Perturbations are applied to raw token embeddings using one gradient step (ℓ_2 radius 10^{-4}). AdamW uses learning rate 5×10^{-5}, weight decay 0.01, linear warmup (10%), and batch size 128 (SST-2, 3 epochs) or 64 (MRPC/RTE, 10 epochs).

Detailed Results. Table 34 lists the selected $\lambda^\star$ and evaluation accuracy across all 20 seeds. Table 35 presents paired t-tests: selected DGA improves the three-task macro average by $+0.41$ points over ERM ($p = 0.016$).

Table 34: Twenty-seed BERT accuracy (%) on official GLUE development sets.

Seed	42	123	456	2024	2025	1001	1002	1003	1004	1005
SST-2 ERM	92.43	93.00	92.55	92.78	94.04	93.23	92.55	92.43	92.78	93.46
DGA_0	92.66	93.46	93.00	92.89	93.46	93.00	93.00	92.78	92.55	92.89
$\mathrm{DGA}_{\lambda^\star}$	92.43	93.35	93.00	92.78	94.27	93.12	93.00	92.78	93.92	93.46
$\lambda^\star$	10^{-5}	10^{-5}	0	10^{-5}	1	10^{-3}	0	0	10^{-3}	0.3
MRPC ERM	84.80	86.03	85.29	85.05	83.09	85.05	82.35	85.29	83.09	84.07
DGA_0	86.03	87.01	84.07	86.76	86.03	86.76	84.80	85.54	85.54	83.33
$\mathrm{DGA}_{\lambda^\star}$	86.03	85.29	85.05	85.05	86.03	86.76	82.84	82.84	83.09	83.33
$\lambda^\star$	0	10^{-4}	0.01	0.01	0	0	0.01	10^{-3}	0.1	0
RTE ERM	66.79	67.51	68.59	68.95	61.01	69.31	67.87	63.54	64.98	67.87
DGA_0	64.26	66.79	68.23	66.06	66.06	68.95	67.51	68.23	68.23	70.40
$\mathrm{DGA}_{\lambda^\star}$	64.26	70.40	68.23	68.59	66.06	70.04	66.43	68.23	66.06	69.31
$\lambda^\star$	0	10^{-5}	10^{-5}	0.01	0	0.01	10^{-3}	0	10^{-3}	10^{-4}

Seed	1006	1007	1008	1009	1010	1011	1012	1013	1014	1015
SST-2 ERM	93.12	92.66	92.89	92.32	93.35	93.12	92.32	92.78	92.43	92.55
DGA_0	92.66	92.55	92.89	93.23	93.12	93.46	93.12	92.43	93.58	92.78
$\mathrm{DGA}_{\lambda^\star}$	93.00	92.89	92.89	92.43	93.35	93.46	93.00	92.43	92.55	92.66
$\lambda^\star$	10^{-5}	0.1	10^{-4}	10^{-4}	0.01	10^{-3}	10^{-5}	0	0.01	1
MRPC ERM	83.82	85.05	85.05	82.11	82.84	86.03	84.31	85.29	84.07	84.07
DGA_0	84.56	83.58	85.29	84.07	83.82	86.27	84.80	87.75	86.27	81.86
$\mathrm{DGA}_{\lambda^\star}$	83.33	82.35	86.03	82.84	83.09	86.27	86.03	86.27	85.78	84.07
$\lambda^\star$	0.3	10^{-3}	10^{-3}	10^{-3}	0.3	0	10^{-5}	0.1	10^{-5}	0.01
RTE ERM	69.31	66.06	67.51	63.54	62.82	64.26	63.90	66.43	71.12	67.51
DGA_0	70.76	67.87	68.59	64.26	61.73	65.34	68.59	67.87	70.04	68.95
$\mathrm{DGA}_{\lambda^\star}$	69.31	67.87	67.51	65.34	61.73	63.54	66.79	67.87	68.95	67.15
$\lambda^\star$	10^{-5}	0	0.01	0.01	0	10^{-5}	10^{-4}	0	10^{-4}	1

Table 35: Paired selected-DGA accuracy differences (percentage points). p_+ tests a positive difference; p_2 is two-sided. Holm_3 adjusts p_+ over the three tasks for this contrast; Holm_6 adjusts over both selected-DGA contrasts (against ERM and DGA_0) on all three tasks. Macro rows average tasks within each seed; the 1001–1015 row reports the corresponding 15-seed subset.

Cohort / Task	Reference	Difference	95% CI	p_+	p_2	$\text{Holm}_3/\text{Holm}_6$
20-seed SST-2	ERM	$+0.20$	$[+0.05, +0.35]$	0.0066	0.013	0.020/0.040
20-seed MRPC	ERM	$+0.28$	$[-0.35, +0.92]$	0.182	0.364	0.182/0.728
20-seed RTE	ERM	$+0.74$	$[-0.23, +1.71]$	0.063	0.125	0.125/0.313
20-seed macro	ERM	$\mathbf{+0.41}$	$[+0.04, +0.78]$	**0.016**	0.032	—
1001–1015 macro	ERM	$+0.35$	$[+0.02, +0.67]$	0.020	0.040	—

F.2 Computer Vision: PreActResNet-18 on Image Benchmarks

Protocol. CIFAR-10, CIFAR-100 (Krizhevsky, 2009), and SVHN (Netzer et al., 2011) use PreActResNet-18 (He et al., 2016) with iterative raw-pixel ℓ_∞ perturbations ($\epsilon = 8/255$, $\alpha = 2/255$, $K = 7$ training, $K = 20$ evaluation). The anchor θ_{ERM} is the same-seed clean model. Validation over a 10% split selects $\lambda^\star \in \{10^{-6}, 10^{-4}, 10^{-3}, 3 \times 10^{-3}, 10^{-2}\}$.

Detailed Results. Table 36 reports per-seed clean accuracy on CIFAR-10, where DGA recovers $+2.06\%$ clean accuracy over DGA_0. Table 37 provides multi-dataset results on CIFAR-100 and SVHN: on SVHN, DGA_0 collapses to 19.6% on 2 of 5 seeds (mean 61.85%), whereas DGA stabilizes all 5 seeds at $94.34\% \pm 0.54\%$.

Table 36: Cross-modal per-seed clean accuracy on CIFAR-10 (5 seeds).

Seed	42	123	456	2024	2025
ERM (paired)	0.9179	0.9148	0.9129	0.9166	0.9152
DGA_0	0.8541	0.8475	0.8199	0.8171	0.8289
Selected DGA ($\text{DGA}_{\lambda^\star}$)	**0.8518**	**0.8494**	**0.8612**	**0.8488**	**0.8591**
Selected $\lambda^\star$	10^{-3}	10^{-3}	10^{-6}	3×10^{-3}	10^{-6}

Table 37: Multi-dataset cross-modal per-seed accuracy (5 seeds; $\lambda^\star$ selected per seed on development/validation data). CV reports clean test accuracy and 20-step perturbed input accuracy on rows marked perturbed. ERM is the paired anchor and DGA_0 is the zero-penalty endpoint.

Seed	42	123	456	2024	2025
CIFAR-100 ERM	0.6983	0.6919	0.6989	0.6959	0.6961
DGA_0	0.5939	0.5975	0.5934	0.5961	0.5930
$\text{DGA}_{\lambda^\star}$	0.5988	0.6044	0.6054	0.6035	0.6034
ERM perturbed	0.000	0.000	0.000	0.000	0.000
DGA_0 perturbed	0.207	0.208	0.205	0.204	0.208
$\text{DGA}_{\lambda^\star}$ perturbed	0.183	0.182	0.179	0.179	0.182
CIFAR-100 $\lambda^\star$	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}
SVHN ERM	0.9666	0.9660	0.9657	0.9679	0.9663
DGA_0	0.1959	0.8346	0.9323	0.9336	0.1959
$\text{DGA}_{\lambda^\star}$	0.9362	0.9500	0.9444	0.9400	0.9466
ERM perturbed	0.005	0.003	0.003	0.002	0.004
DGA_0 perturbed	0.196	0.232	0.500	0.515	0.196
$\text{DGA}_{\lambda^\star}$ perturbed	0.243	0.405	0.401	0.482	0.350
SVHN $\lambda^\star$	10^{-6}	3×10^{-3}	3×10^{-3}	10^{-3}	10^{-2}

G Limitations and Discussion

Theoretical Limitations (Theory-to-Practice Gap). Theorem 1 provides an asymptotic sandwich-risk expansion for a projected linear M-estimator under continuous ℓ_2 perturbations and sequential limits ($n \to \infty$ followed by $\epsilon \to 0$), conditioned on curvature dominance (Assumption A4′). Our finite-network experiments test directional predictive alignment, variance contraction,

and learned displacement in deep architectures under multi-step ℓ_∞ input updates; the linear expansion is not an identity for those networks. Extending the analysis requires treating non-convex optimization, the second-order estimator-bias correction Q, and the non-smooth geometry of ℓ_∞ projections.

Empirical Scope and Cross-Modal Boundaries. On the empirical side, our controlled mechanism probes focus primarily on the ETT benchmarks, whereas on ultra-high-dimensional datasets (Electricity with $d = 321$ and Traffic with $d = 862$), DGA lowers clean MSE across all 16 settings by comparatively modest margins (0.1% to 0.7%). As budget sweeps indicate (Appendix E.1), simple scaling alone does not resolve this attenuation; applying uniform ℓ_∞ perturbations across hundreds of channels inevitably dilutes gradient signals across uninformative variates, warranting future exploration of channel-sparse or low-rank perturbation subspaces. Additionally, cross-modal evaluations employ one representative backbone per domain (BERT-base and PreActResNet-18), where vision reflects an inherent trade-off with perturbed input accuracy, and SVHN exhibits seed-dependent optimization collapse that DGA stabilizes. These transfer experiments evaluate ERM-centered parameter trajectory control across tasks rather than postulating a universal spectral law across all data domains.