PATCHTRANSFERNET: A MECHANISM-DRIVEN DEMAND FORECASTING MODEL

Anonymous authors
Paper under double-blind review

ABSTRACT

Classical time series forecasting models are mostly designed for slowly varying, smooth, continuous signals (M1). By contrast, retail demand is discrete and non-negative, often intermittent or sparse. We model retail demand as an observation-driven counting process (M2), and then develop PATCHTRANSFERNET (PTNET), which translates the M2 recursion directly through mean-preserving convex decoding and aggregate-consistent supervision. Sparse series put the two targets in conflict. Per-day medians can be all zero under a positive window total, so daily supervision misses the total and aggregate supervision cannot distribute it. An idealized mixed-Poisson window turns this tension into a two-fingerprint rule. Firstly, the zero-mass share gates aggregate supervision on sparse populations, and then the benefit grows with the count autocorrelation ρ, the idealization places the crossing at $\rho_0 = 1/(W+1)$ for sales cycle W. The rule places the optimum correctly on all three demand datasets. Two of them sit at opposite ends of the λ range, and the third splits by the reporting metric, where the daily caliber selects λ=0 and the official aggregate metric selects λ=0.5. On FreshRetailNet-50K, PTNET reaches 27.94% WAPE, beats every single-stage forecaster in all five seeds by at least 1.4 pt (worst seed 29.60 vs. best baseline 31.06), and shrinks its under-forecast to -2.2%, matching the best two-stage pipeline. On the public M5 benchmark it ties the best published store-item-level score (0.905). When transplanted into a linear backbone on ten classical benchmarks, the anchor's gain tracks R^2_{last}, so each data type rewards its own inductive bias.

1 INTRODUCTION

Retail demand forecasting is a high-impact problem, yet the deep-learning revolution in time series has largely passed it by. The architectures that dominate classical benchmarks, from patching (Nie et al., 2023) and trend–seasonal decomposition (Wu et al., 2021; Zeng et al., 2023) to frequency-domain mixing (Zhou et al., 2022), assume that a good forecast tracks the shape of a smooth, slowly varying, and periodically repeating continuous signal. They try to reconstruct the waveform of series, which is the right bias for traffic flow or electricity load but not for retail demand, a discrete counting process driven by historical sales, inventory, promotions, and holidays. Classical statistics built specialized tools for this regime (Croston, 1972; Weiß, 2018), and the deep architectures ignore them.

This paper argues that *demand forecasting needs its own inductive bias*. We characterize retail demand as an *observation-driven counting process* (M2, Sec. 3), whose conditional mean is updated from past observations. The classical benchmarks instead hold smooth continuous series (M1), and a latent-driven boundary type (M3) sits at the low-autocorrelation extreme. Two design commitments follow. Forecasts should be anchored as convex combinations of historical patch means, the sufficient statistic for the latent rate, and the training objective should include a windowed-total term when the population is persistent enough that daily counting noise cancels at the sales-cycle horizon. PATCHTRANSFERNET (PTNET) realizes both, with components that map one-to-one onto the M2 recursion. On FreshRetailNet-50K (Wang et al., 2025), 50,000 public retail series with stock-out annotations, it reaches 27.94% WAPE, beating every single-stage forecaster by a seed-robust margin and matching the best two-stage recovery-then-forecast pipeline (27.94 vs 29.02).

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.

The larger contribution is the decision rule the mechanism yields, and the rule is falsifiable. Two structural facts pull the training loss in opposite directions. The pinball loss is subadditive, so for a given forecast the aggregate error is bounded above by the daily error (Proposition 1(i)). We also invert the test. If the M2 mechanism is genuinely demand-specific, its bias should *hurt* where the data carry no level information. Transplanting PTNET's anchor mechanism into a fixed linear backbone (DLinear) over ten M1 benchmarks produces exactly that switch. The gain over DLinear is governed by R^2_{last}, the R^2 of regressing Δx_{t+1} on x_t, and is positive above 0.04 and negative below 0.02. Each data type rewards its own inductive bias, and the fingerprints make the choice measurable rather than a matter of taste.

Contributions. We characterize demand as an observation-driven counting process and derive its two fingerprints (Sec. 3); the decision rule that follows is stated and tested on three demand datasets (§5.3); PATCHTRANSFERNET realizes both biases, and its numbers on FreshRetailNet-50K, SPR, and M5 appear in §5.2; and the test is inverted on ten classical benchmarks to locate where the bias stops helping (§5.4).

2 RELATED WORK

Deep architectures for time-series forecasting. The transformer lineage (Zhou et al., 2021; Wu et al., 2021; Zhou et al., 2022; Liu et al., 2022; Nie et al., 2023; Lim et al., 2021), its efficiency counter-line (Zeng et al., 2023; Das et al., 2023; Liu et al., 2024; Wang et al., 2024a; Wu et al., 2023), and instance normalization (Kim et al., 2022) are all designed for, and benchmarked on, smooth continuous series (M1). Our results quantify both their failure mode on counting data and the regime in which they dominate (Sec. 5.4).

Counting-process and intermittent-demand models. Croston's method (Croston, 1972) and its bias-corrected SBA variant (Syntetos & Boylan, 2005) forecast the inter-demand interval and mean size separately and remain standard in inventory practice (Hyndman et al., 2014). Our statistical backbone is the INGARCH class of observation-driven count models (Ferland et al., 2006; Fokianos et al., 2009; Weiß, 2018), with retail demand-forecasting practice studied by Ferreira et al. (2015). We add no new count model, only elementary structural facts about counting processes, namely median gaps and allocation unidentifiability, that constrain what a forecasting loss can achieve on its own.

Probabilistic forecasting, reconciliation, and model selection. Likelihood-based deep demand forecasting (Salinas et al., 2020) and censored-demand correction (Tobin, 1958) address estimation, while a separate line enforces *coherence* across aggregation levels (Athanasopoulos et al., 2017; Wickramasuriya et al., 2018; Rangapuram et al., 2021). Our aggregate-supervision term is not a hierarchy reconciliation but a single windowed-total constraint whose usefulness is regime-dependent (Lemma 2, Proposition 3). Choosing a forecaster from measurable series properties is classical, our contribution is a pair of decision rules, ρ versus $1/(W+1)$ on the demand side and R^2_{last} thresholds on the continuous side, validated by controlled transplant experiments. The closest benchmark-specific study is the FreshRetailNet-50K paper (Wang et al., 2025), whose two-stage pipelines motivate our single-stage comparison.

3 THE DEMAND MECHANISM: DATA TYPES, FINGERPRINTS, AND THE LOSS CHOICE

The design of the loss follows from what generates the data. We fix a three-type taxonomy. The smooth continuous type (M1) sits behind the classical benchmarks, the observation-driven counting process (M2) behind intermittent retail demand, and its latent-driven boundary (M3) is where the observation-driven feedback decays into noise. The section then develops the loss choice, through the fingerprints (Sec. 3.2), the trade behind them (Sec. 3.3), and the resulting criterion (Sec. 3.4).

3.1 THREE GENERATIVE MECHANISMS

Type M2: observation-driven counting process. Let $\{y_t\}_{t\geq 1}$ be a nonnegative integer count series (daily sales). We model it as an *observation-driven* counting process,

$$y_t \mid \mathcal{F}_{t-1} \sim \text{NegBin}(\lambda_t, \phi), \qquad \lambda_t = \omega + \alpha y_{t-1} + \beta\lambda_{t-1} + \sum_{\ell\in\mathcal{S}} \gamma_\ell\, y_{t-\ell} + \boldsymbol{\delta}^\top \mathbf{z}_t, \tag{1}$$

where $\mathcal{F}_{t-1}$ is the information set up to time $t-1$, λ_t is the conditional mean (rate), $\mathcal{S}$ is a set of seasonal lags, and $\mathbf{z}_t$ collects exogenous covariates (promotions, holidays, stock-out flags). The Poisson analyses below take the limit $\phi \to \infty$, where the variance equals the mean. The defining property is *observation-driven*. The rate is latent, but it carries no independent innovation of its own, so it is a deterministic function of the observed past. This is the INGARCH class (Ferland et al., 2006; Fokianos et al., 2009; Weiß, 2018). Equivalently, the latent rate is a signal-plus-slow-level process $\lambda_t = \mu\, u_t\, s(t)$, where the level u_t is persistent but *updated by past observations* through the recursion.

The two contrast types are the smooth continuous series (M1) of the classical benchmarks and the low-autocorrelation latent-driven Cox boundary (M3). App. A.1 defines both, with their fingerprints; M1 appears again only in the cross-domain diagnostic of Sec. 5.4.

3.2 TWO FINGERPRINTS: GAIN SIDE AND COST SIDE

Gain side: the count autocorrelation $\rho = \text{ACF}(1)$. The lag-one autocorrelation of the observed counts measures persistence strength and predicts the gain of aggregate supervision. A higher ρ means the W days of a window share more signal, so a window-total constraint can recover more of the level (Sec. 3.4).

Cost side: the zero-mass share. The second fingerprint measures sparsity, the share of the daily allocation that aggregate supervision can destroy (Proposition 1(ii)). For a window length W it is the share of series whose day-level median is 0 while the W-day total median is at least 1, both read over the last four sales cycles of the training history (the final $4W$ days); formally,

$$p_{\text{zero}}(W) \;=\; \Pr_{\text{series}}\Big[\, \text{med}(y_{1:W}) = 0 \;\wedge\; \text{med}\Big(\textstyle\sum_{h\leq W} y_h\Big) \geq 1 \,\Big]. \tag{2}$$

These are exactly the series on which Lemma 2's gap is *active*. The day-level median pins the forecast at zero while the window total is positive, so the daily loss is blind to the total on precisely this share, and forcing the total onto such series distorts their daily allocation. W is the sales cycle that also enters the ρ threshold, and it is weekly (W=7) in every demand dataset here, so both fingerprints use one window. Measured values appear in Table 3.

3.3 THE LOSS CHOICE: WHEN AGGREGATE SUPERVISION HELPS

A PTNET-style model (Sec. 4) can be trained with a day-level loss alone, or with an auxiliary *aggregate-supervision* term, the pinball loss on the windowed total. The objective becomes

$$\mathcal{L} = (1-\lambda)\,\mathcal{L}_{\text{day}} + \lambda\,\mathcal{L}_{\text{agg}}, \tag{3}$$

with $\lambda \in [0,1]$. Two opposing forces decide whether aggregate supervision helps.

Gain side: the total constraint carries level information.

Proposition 1 (Allocation is unidentifiable from the total)**.** *Let $\mathcal{L}^{\text{day}}_{\text{pin}}$ and $\mathcal{L}^{\text{agg}}_{\text{pin}}$ denote the pinball losses at a common level τ, the former applied per day and the latter to the W-day sums. Then the following holds.*

(i) ***Triangle inequality.*** *For any forecasts, $\mathcal{L}^{\text{agg}}_{\text{pin}}\big(\sum_h \hat{y}_h, \sum_h y_h\big) \;\leq\; \sum_{h=1}^{W} \mathcal{L}^{\text{day}}_{\text{pin}}\big(\hat{y}_h, y_h\big)$.*

(ii) ***Allocation unidentifiability.*** *For any two daily allocations $\{\hat{y}^{(1)}_h\}$, $\{\hat{y}^{(2)}_h\}$ with equal window sums $\sum_h \hat{y}^{(1)}_h = \sum_h \hat{y}^{(2)}_h$, the aggregate loss is identical. Under pure aggregate supervision ($\lambda = 1$) the objective therefore does not identify a daily allocation.*

Both parts follow from subadditivity of the pinball loss, $\rho_\tau(u_1+u_2) \le \rho_\tau(u_1) + \rho_\tau(u_2)$. Applied day by day it gives (i), which bounds the aggregate error above by the daily error but does not by itself make the total term free, while (ii) holds because $\mathcal{L}^{\text{agg}}_{\text{pin}}$ sees the forecasts only through their sum. This is *allocation unidentifiability*, and the zero-mass share measures it.

Cost side: when the gain-side bound corrupts the daily forecast.

Lemma 2 (Median incompatibility gap)**.** *Let $y_1, \dots, y_W \mid \mathcal{F}$ be conditionally independent counts with common zero probability $p = \Pr(y_h = 0 \mid \mathcal{F})$, the window-constant-rate idealization, and let $\text{med}(\cdot)$ denote a conditional median. If $\frac{1}{2} < p < 2^{-1/W}$, then*

$$\sum_{h=1}^{W} \text{med}(y_h \mid \mathcal{F}) = 0 \quad \textit{while} \quad \text{med}\Big(\sum_{h=1}^{W} y_h \,\Big|\, \mathcal{F}\Big) \ge 1.$$

Hence any forecaster calibrated at the conditional day median is *structurally* unable to see a positive window total on such series. The window-total correction can then only be applied by moving the daily forecast away from its median, which is where the gain side's bound turns costly.

3.4 A Mechanism-Inspired Scaling Law and the Decision Rule

Scaling. Whether the gain side or the cost side dominates is decided by how the window total's variance grows with the window length W, under an idealized *mixed-Poisson* window: a persistent per-series level μ with mean $m = \mathbb{E}[\mu]$ and variance $\sigma^2 = \text{Var}(\mu)$, around which the W days are conditionally independent Poisson counts (the zero-feedback, window-constant-rate limit of Eq. 1). The law of total variance gives $\text{Var}(S) = W^2\sigma^2 + Wm$: a *level footprint* $W^2\sigma^2$, against a *noise floor* Wm. Their ratio is the scaling law behind the decision rule. When the footprint dominates, the total constraint transmits genuine level information; when the floor dominates, it transmits only sampling noise that reaches the daily forecasts as a distortion.

Proposition 3 (Gain-side crossing in an idealized mixed-Poisson model)**.** *In this idealization, with level mean $m = \mathbb{E}[\mu]$, level variance $\sigma^2 = \text{Var}(\mu)$, and the level share $\sigma^2/(m+\sigma^2)$, the level footprint exceeds the noise floor, $W^2\sigma^2 > Wm$, if and only if*

$$\frac{\sigma^2}{m+\sigma^2} \;>\; \rho_0 \;=\; \frac{1}{W+1}. \tag{4}$$

Proof in App. A.2. In the idealization the level share coincides with this $\text{ACF}(1)$, and the crossing holds within-window persistence at its maximum, so it is the most favorable case for the gain side. In a finite window the observed $\text{ACF}(1)$ is further attenuated by level drift inside the window, so reading the threshold on it is conservative.

Corollary 4 (Two-fingerprint decision rule)**.** *Consider a population of demand series with sales cycle W, count autocorrelation ρ, and zero-mass share $z = p_{\text{zero}}(W)$, trained with Eq. 3. Then:*

- *(i)* ***Gate (cost side).*** *If z is high, aggregate supervision is vetoed regardless of ρ. The day-level median is pinned at zero on a large share of series, so forcing the window total corrupts the daily allocation (Lemma 2, Proposition 1(ii)). Use $\lambda = 0$ or a small λ.*
- *(ii)* ***Benefit (gain side).*** *If z is near zero, aggregate supervision is safe at any ρ, and its benefit grows with ρ. The idealized crossing of Proposition 3 separates a small benefit below from a decisive one above.*
- *(iii)* ***Gray zone.*** *When both fingerprints are intermediate* for the series being scored*, cross-validate. When the population is stratified* across aggregation levels *and the reporting metric re-weights across them, no single λ is implied and the metric decides. M5 is the worked case, its item×store level carries the fingerprint ($z = 42.9\%$, $\rho = 0.112$, gate shut) and is the dimension* PtNet *models, so daily WAPE selects λ=0; but every level above item has $z = 0$ with $\rho = 0.38$–0.54, far above ρ_0, so the official metric selects λ=0.5 on levels the cost-side gate never vetoed.*

Scope. *The rule is population-level, because the gain-side crossing is derived on a population level share and read on the observed ρ, and the zero-mass share is a probability over series. But there λ*

is a single scalar shared by every series, so a subgroup that would individually prefer another value cannot opt in.

4 PATCHTRANSFERNET: A NEURAL REALIZATION OF THE M2 MECHANISM

Section 3 yields a design contract. The M2 recursion says the forecastable object is a persistent rate reconstructed from past observations, so the model must (i) preserve the windowed latent rate through convex anchoring and (ii) allow a smooth inertial correction on top of it. The loss analysis of Sec. 3.3 adds a third requirement, to align with the windowed total only when Corollary 4 permits. PATCHTRANSFERNET realizes this contract in three components.

4.1 MULTI-SCALE PATCHES AND TRANSFER ATTENTION

The history is segmented into P overlapping patches at daily and aggregated scales, using a deterministic, non-parametric plan. With encoder length E, the plan stacks three bands: (i) the most recent d days as singleton patches; (ii) tiles of a set of lengths $\mathcal{S}$ laid from the window start with stride ℓ, giving $\lfloor E/\ell \rfloor$ patches each; and (iii) one global patch covering the full history, so that $P = d + \sum_{\ell \in \mathcal{S}} \lfloor E/\ell \rfloor + 1$. Each patch p covers $\mathcal{I}_p \subseteq \mathcal{E}$, carries a level anchor $b_p = |\mathcal{I}_p|^{-1} \sum_{t \in \mathcal{I}_p} \max(y_t, 0)$, the mean of valid sales over the patch, and a scale descriptor $(1 - z_p/E,\ \ell_p/E,\ \log(1 + z_p))$ recording recency, normalized length, and log age; patches reaching before the window start are masked. **Encoding.** Each patch is padded to $T_{\max}$ days; its daily sales, stockout and availability channels are concatenated with calendar embeddings, $z_p = [x_p \| \mathrm{Emb}_{\mathrm{cal}}(C_p)]$, processed by a 1-D convolution (kernel 3) with GELU and global average pooling, and fused with the scale descriptor by two linear branches and a variable-selection network (VSN) into a patch token $\boldsymbol{h}_p \in \mathbb{R}^H$ (H=128), zeroed on masked patches. The static columns (city, store, hierarchy, SKU) are embedded at dimension $\min(H, \max(2, \lceil v_i/2 \rceil))$ and mixed by a second VSN into a context $\boldsymbol{s}$. Gated residual networks (GRN) of TFT (Lim et al., 2021) modulate both into a patch key $\boldsymbol{k}_p = \mathrm{GRN}(\boldsymbol{h}_p; \boldsymbol{s})$ and a query $\boldsymbol{q}_h = \mathrm{GRN}([\mathrm{Emb}_{\mathrm{cal}}(\mathrm{dec_cal}_h) \| \mathrm{dec_cal_real}_h]; \boldsymbol{s})$ from future calendar features (weekday, holiday, promotion) alone, with no future sales. For each future day h, the query attends over the patch keys (Vaswani et al., 2017), producing simplex weights that average the score over attention heads with invalid patches masked before the softmax,

$$
\begin{aligned}
S_{h,p} &= \frac{1}{H_{\text{head}}} \sum_{j=1}^{H_{\text{head}}} \frac{[W_Q \boldsymbol{q}_h]_j^\top [W_K \boldsymbol{k}_p]_j}{\sqrt{d}}, \\
\alpha_{h,p} &= \frac{\exp(S_{h,p})}{\sum_{p'=1}^{P} \exp(S_{h,p'})}, \qquad \alpha_{h,p} \geq 0, \quad \textstyle\sum_{p=1}^{P} \alpha_{h,p} = 1,
\end{aligned}
\tag{5}
$$

with H_{head}=4 heads, $[\cdot]_j$ the width-d=H/H_{head} slice of head j, and m_p=0 patches set to a large negative value before the softmax. These are the neural analogue of the transfer coefficients in the M2 recursion (Eq. 1): attention relaxes the fixed lag structure into a learned, calendar-indexed weighting of multi-scale history, and the patch sizes expose the temporal resolution of the demand texture directly.

4.2 MEAN-PRESERVING CONVEX DECODING

The forecast anchor is a convex combination of raw anchors,

$$
\mu_h = \sum_{p=1}^{P} \alpha_{h,p}\, b_p. \tag{6}
$$

Here μ_h is bounded by the realized patch means, $\min_p b_p \leq \mu_h \leq \max_p b_p$, it stays inside the convex hull of observed sales, is in the same units as the target with no de-normalization, and offsets the systematic under-forecast of log-space decoders. The attention output, a context vector $\boldsymbol{c}_h = \sum_p \alpha_{h,p} \boldsymbol{h}_p$, drives three zero-initialized correction heads, a multiplicative gate $g'_h = \exp(\mathrm{clamp}(\cdot, -3, 3))$, an additive residual Δ_h, and a quantile spread $s_h = \mathrm{softplus}(\cdot)$, with final center and quantile forecasts

$$
\mathrm{center}_h = \mathrm{relu}\big(\mu_h\, g'_h + \Delta_h\big), \qquad \hat{y}_{h,q} = \mathrm{relu}\big(\mathrm{center}_h + s_h\,(q - \tfrac{1}{2}) \cdot 2\big). \tag{7}
$$

Because $s_h \geq 0$ and $2(q-\frac{1}{2})$ increases with q, the quantiles are monotone by construction, $\hat{y}_{h,0.25} \leq \hat{y}_{h,0.5} \leq \hat{y}_{h,0.75}$, so they never cross. Zero-initialization makes the network start exactly at the level-anchored μ_h for every q, and the three corrections are learned on top. The PTNET framework is shown in Figure 1.

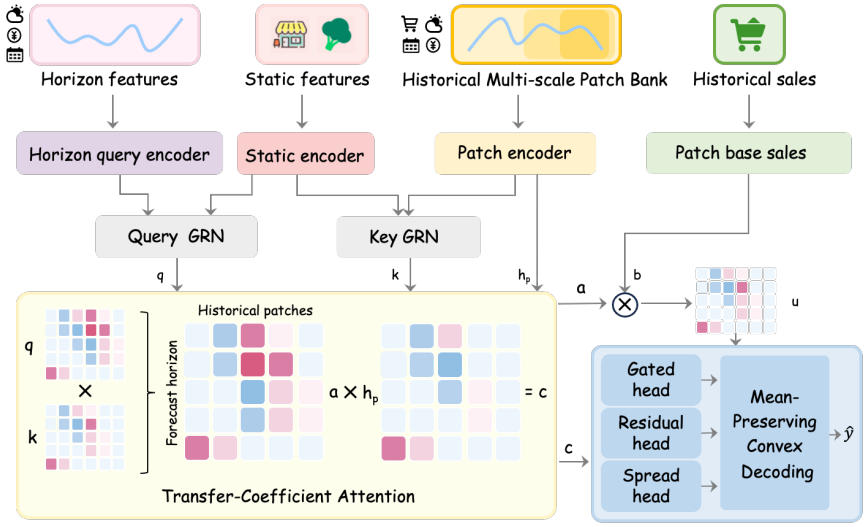

Figure 1: The PTNET framework. The history is tokenized into daily and aggregated patches, and a calendar-derived query mixes the patch tokens by simplex weights $\alpha_{h,p}$ into the level anchor μ_h; three zero-initialized heads (a multiplicative gate, an additive residual, and a quantile spread) then correct the center and expand it into daily quantiles, whose window sum forms the aggregate-consistent loss.

4.3 AGGREGATE-CONSISTENT SUPERVISION

The aggregate prediction sums the daily quantile forecasts over the forecast window, $\hat{Y}^{\text{agg}}_{i,q} = \sum_{h \in \mathcal{W}} \hat{y}_{i,h,q}$, and the full loss is the convex blend of Eq. 3. The window set $\mathcal{W}$ is the whole horizon in a single sum on FRN and SPR, and the W-day windows tiling the horizon on M5. Both the aggregate target and the aggregate prediction are summed over non-stockout days through the realized availability mask $m_{i,h} = 1 - \text{lack}_{i,h}$, used in the loss alone, which aligns the training objective with the evaluation caliber. The daily term is never masked, so the model still learns the per-day shape. The aggregate prediction is exactly the sum of the daily forecasts, so consistency holds by construction, with no separate reconciliation step and no extra parameters.

5 EXPERIMENTS

Three experiments test three commitments. **A** asks whether a forecaster built from the M2 biases wins (§5.2). **B** tests the two-fingerprint criterion, the mechanism's riskiest quantitative commitment, on three datasets placed at the threshold's two ends and in the gray zone (§5.3). **C** inverts the test, asking where the demand bias does *not* transfer (§5.4). Full details are in the appendix.

5.1 DATASETS AND PROTOCOL

The three demand datasets place the criterion at its two ends and on the boundary, and the classical suite spans both sides of the anchor-transfer threshold. FreshRetailNet-50K (Wang et al., 2025) (FRN, public, 50,000 series, W=7) sits squarely in M2, SPR (internal, 53,552 series, 21-day horizon) drifts toward the M3 boundary, M5 (Makridakis et al., 2022) (30,490 public Walmart series, 28-day horizon) is the gray zone, and the ten TSLib benchmarks are all M1, with R^2_{last} on both sides of the crossing (Table 3, with the full list in App. C.1). Their fingerprints separate the two regimes cleanly. FRN has $\rho = 0.236$ with zero-mass 0.2%, while M5 and SPR have $\rho = 0.112/0.084$ with

Table 1: Main results on the three demand datasets. FRN and SPR are scored over their full forecast window under the same day-level stock-out mask, which drops the rows flagged out of stock and keeps each series' remaining days (%). M5 uses the official window d_{1914}–d_{1941}, scored over 28 days plus the official twelve-level WRMSSE and the item×store level L_{12} (the caliber of App. C.4). All rows are self-run under the shared single-run protocol (seed 42) except the † rows, whose FRN numbers quote the benchmark paper (Wang et al., 2025) (App. C.2). The two-stage pipeline is FRN-only. Bold marks the best entry per column and underline the second best (WPE by closeness to zero). The PTNET rows show the loss-weight transfer. λ=1 wins FRN, λ=0 wins SPR and M5 daily, and λ=0.5 minimizes the twelve-level M5 WRMSSE.

	FRN		SPR		M5 (official window)			
Model	WAPE	WPE	WAPE	WPE	WAPE	WPE	WRMSSE	L_{12}
INGARCH(1,1)+y_{t-7}	37.77	**+1.87**	92.32	**+7.70**	81.29	-34.6	2.726	1.07
INGARCH(1,1)+weekday	40.00	-5.19	86.39	+9.82	82.95	**+4.0**	1.188	0.96
DLinear†	31.56	-4.89	75.34	-31.17	70.66	-23.8	1.714	0.87
TFT†	31.75	-7.37	66.55	-45.45	**66.25**	-29.0	1.769	0.87
PatchTST	31.16	-4.99	70.68	-37.85	74.90	-36.5	2.377	0.98
TiDE	31.79	-5.71	77.18	-25.11	70.13	-28.9	1.910	0.88
iTransformer	34.55	-6.11	87.84	-63.49	74.33	-39.5	2.539	0.97
TimeMixer	31.06	-6.94	67.54	-32.97	67.80	-29.5	1.944	0.87
TimesNet	32.00	-8.14	71.25	-34.24	69.47	-18.3	1.425	0.86
TFT+TimesNet† (2-stage)	29.02	+2.58	—	—	—	—	—	—
PTNET (λ=0)	30.30	-7.50	**66.15**	-18.02	67.19	-34.5	2.011	0.88
PTNET (λ=0.5)	28.41	-4.70	71.52	-12.03	71.24	-6.7	**0.904**	**0.843**
PTNET (λ=1)	**27.94**	-2.19	71.80	-15.08	76.28	-6.2	1.771	0.93

zero-mass $42.9\%/37.3\%$. The ρ values are per-series lag-1 ACFs averaged over series (Sec. 3.2; M5 on its active 364-day window) and zero-mass follows Eq. 2 at W=7 (App. B).

5.2 EXPERIMENT A: MAIN RESULTS ON FRESHRETAILNET-50K

Experiment A tests whether a model built from the M2 biases of Sec. 4 beats architectures that import M1 biases onto demand data, and it does (Table 1). With the mechanism-predicted $\lambda = 1.0$, PTNET reaches 27.94% WAPE, ahead of every single-stage forecaster by a margin that is robust across five seeds, and at 1.33M parameters it is the smallest attention-based model in our comparison (App. C.2). Two rows support the mechanism's framing. The parametric member of the M2 class itself, INGARCH$(1, 1)$ with the best seasonal form, trails by roughly 10 pt, so the mechanism's value is realized through the architecture and loss rather than the parametric form. And the same PTNET trained without aggregate supervision under-forecasts at -7.5% WPE, shrinking to -2.2% at λ=1, so Lemma 2's under-forecast is a loss-caliber effect that the prescribed supervision corrects.

The five-seed spread of the champion configuration is $27.94/27.99/28.88/28.91/29.60$ (median 28.88). Every seed clears the best single-stage baseline by at least 1.4 pt, so the single-stage win is seed-robust. Against the best two-stage pipeline, TFT+TimesNet at 29.02, the median seed leads by 0.14 pt and the worst seed trails by 0.58 pt. The same five seeds give daily-WAPE spreads of 1.66 pt on FRN, 3.19 pt on SPR, and 0.09 pt on M5 (λ=0). SPR, the population drifting toward the M3 boundary, is the least seed-stable of the three. Table 7 (App. C.3) reports the full per-seed grid behind these spreads, including the aggregate caliber.

5.3 EXPERIMENT B: MECHANISM VALIDATION, THE λ TRANSITION

Experiment B sweeps λ on three datasets placed at the threshold's two ends and on the boundary, and asks whether each optimum lands where Corollary 4 says. It does in all three cells (endpoints in Table 1, full eleven-point grid in Table 9, App. C.5). FRN, decisively above the crossing with near-zero zero-mass, improves nearly monotonically to λ=1. SPR, decisively below the crossing, is optimal at λ=0 with no interior minimum, and a dense re-sweep between 0 and 0.1 confirms a true regime edge. M5, designated as the gray-zone population before any model was trained, splits its optimum along the two fingerprints, daily accuracy vetoes supervision through the 43% zero-mass

share, while the aggregate caliber demands it through the 34% high-ρ share. The split replicates in full on the competition's withheld final window (all eleven λ points), including the interior aggregate optimum at the same λ=0.5 (twelve-level WRMSSE, App. C.5). The WPE columns add the calibration dimension, where aggregate supervision shrinks the daily-median under-forecast of Lemma 2 (FRN $-7.50 \rightarrow -2.19$; M5 $-34.5 \rightarrow -6.7$).

The same fingerprints also predict *where inside each population* the effect concentrates, and Figure 3 (App. C.5) tests this at the series level. The gain of λ=1 over λ=0 is at most $+1.47$ pt in any bin below the crossing, reaches $+2.87$ and $+3.84$ pt in the two bins above it, and falls back to $+1.45$ pt at $\rho \geq 0.5$; the cost of supervision on SPR falls from 6.6–16.4 pt in every bin below the crossing to 2.3 and -0.2 pt above it. The transition belongs to the *aggregate decision* rather than to individual series. Per-bin gains form no sharp per-series phase transition, and the high-ρ minority on SPR cannot opt into supervision because the loss and parameters are shared.

The gate dominates the benefit. The subgroup *re-training* experiment (Table 10, App. C.6) tests whether the population-level rule is right for the right reason, since the full populations mix the two fingerprints. Re-training each extreme separately, split at the predicted boundary $\rho_0 = 1/8$, confirms that it is. FRN's low-ρ subgroup does *not* flip to λ=0, because its zero-mass share is 0.06% and the gate is therefore open; the below-threshold ρ only reduces the benefit ($+0.97$ pt versus $+2.39$ pt above). SPR's high-ρ minority, where ρ alone would license supervision, instead sides with the veto and pays for it (-3.08 pt). No subgroup flips, which is the ordering in Corollary 4. **The zero-mass share gates the decision, and ρ sets how much aggregate supervision adds once the gate is open.** Only on M5, where the two dimensions mix, does the joint criterion carry the decision.

5.3.1 M5 AS A PUBLIC BENCHMARK

The Table 1 also set as a public-benchmark comparison on M5. TFT (Lim et al., 2021) is the one baseline whose decoder is also a demand-quantity architecture with calendar-aware gating, and it wins daily WAPE (66.25 vs 67.19), while all remaining rows trail PTNET λ=0 (five seeds, daily WAPE spanning 0.09 pt). The aggregate caliber is where the mechanism separates from the field. On the competition's withheld private window, the same model's 0.726 beats 18 of the 24 published single-method benchmarks while tying the best item-level score (L_{12} 0.905); App. C.4 gives the table.

5.4 EXPERIMENT C: CROSS-DOMAIN DIAGNOSIS

The third layer inverts the test, asking where the demand bias does *not* transfer. If the M2 mechanism is genuinely demand-specific, transplanting PTNET's anchor-correction head into a DLinear-style backbone should *hurt* exactly where the level carries no information. The result on the ten TSLib classical benchmarks is a controlled switch. The anchor wins on the four datasets whose R^2_{last} clears the level-informativeness bar and loses on the rest, so the same PTNET that wins demand data loses to DLinear on six of ten classical datasets, while the level-removing PatchTST, dominant there, loses to PTNET on FRN. PTNET runs there as DLinear plus an anchor head, since classical benchmarks do not provide the full demand-side inputs; App. C.1 gives the table together with the capacity and anchor controls behind it.

5.5 COMPONENT ABLATION

The FRN win is a mechanism effect rather than a component effect (full table in Table 11, App. C.7). Every removal that touches only the machinery around the mechanism still leaves the model ahead of the best single-stage baseline (31.06). In all of them the WAPE keeps its sign and only its magnitude moves (WPE -4.00 to -12.21 against Full's -2.19), so what the machinery buys is the calibration level, not the win. Removing the mechanism is the one change that costs it, collapsing the anchor-derived heads to a single mean head is the largest accuracy loss in the table (3.41 pt), the only variant that trails the best single-stage baseline (31.35 vs 31.06), and the only one whose bias reverses sign (WPE $-2.19 \rightarrow +2.17$).

6 DISCUSSION

A practical model-selection rule. Compute $\rho = \mathrm{ACF}(1)$ over the population, with W the sales cycle, and the active zero-mass share z over the last four training cycles. If z is high, keep the daily loss (λ=0) even when ρ is above threshold. If z is near zero, aggregate supervision is safe, and ρ above $1/(W+1)$ makes it the default (λ=1). Near the threshold or in a mixed population, cross-validate and let the metric decide, as on M5, where daily favors λ=0 and aggregate favors λ=0.5. For continuous series when R^2_{last} above 0.04 anchor-based models may help, below 0.02 prefer linear models, and in between cross-validate.

Limitations. The R^2_{last} thresholds and the operating threshold $\rho_0 = 1/8$ are empirically calibrated decision boundaries rather than universal constants. The fingerprints inherit the drift of the history they are read on, so a population whose persistence shifts after training would need them recomputed on newer history. On M5 our single model runs without competition machinery and is compared only against the organizers' published single-method benchmarks on the leaderboard's own withheld window and scorer (App. C.4).

7 CONCLUSION

We model retail demand as an observation-driven counting process (M2), and forecasting it needs the biases that process implies. PATCHTRANSFERNET realizes two of them, convex anchoring and aggregate supervision, and surpasses every single-stage forecaster on FreshRetailNet-50K with a seed-robust margin while nearly eliminating systematic under-forecast. The loss-choice rule, the zero-mass gate plus the ρ versus $1/(W+1)$ benefit, survived every empirical test we posed. Each data type rewards its own bias, and the fingerprints ρ, zero mass, and R^2_{last} tell the practitioner which one their data needs.

AI USE STATEMENT

AI-based assistants were used for routine writing and coding support. All research contributions were conceived, executed, and verified by the authors.

REPRODUCIBILITY STATEMENT

The method is specified in full in §4. All experiments follow the protocols of the benchmarks they use; complete configuration, hyperparameters, and dataset details are provided in the appendix.

REFERENCES

George Athanasopoulos, Rob J. Hyndman, Nikolaos Kourentzes, and Fotios Petropoulos. Forecasting with temporal hierarchies. *European Journal of Operational Research*, 262(1):60–74, 2017.

J. D. Croston. Forecasting and stock control for intermittent demands. *Journal of the Operational Research Society*, 23(3):289–303, 1972.

Abhimanyu Das, Weihao Kong, Andrew Leach, Shaan Mathur, Rajat Sen, and Rose Yu. Long-term forecasting with TiDE: Time-series dense encoder. In *Transactions on Machine Learning Research (TMLR)*, 2023.

René Ferland, Alain Latour, and Driss Oraichi. Integer-valued GARCH process. *Journal of Time Series Analysis*, 27(6):923–942, 2006.

Kris Johnson Ferreira, Bin Hong Alex Lee, and David Simchi-Levi. Analytics for an online retailer: Demand forecasting and price optimization. *Manufacturing & Service Operations Management*, 18(1):69–88, 2015.

Konstantinos Fokianos, Anders Rahbek, and Dag Tjøstheim. Poisson autoregression. *Journal of the American Statistical Association*, 104(488):1430–1439, 2009.

Rob J Hyndman, George Athanasopoulos, et al. *Forecasting: Principles and Practice*, volume 2. OTexts Melbourne, Australia, 2014.

Taesung Kim, Jinhee 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.

Bryan Lim, Sercan Ö Arık, Nicolas Loeff, and Tomas Pfister. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4): 1748–1764, 2021.

Yong Liu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Non-stationary transformers: Exploring the stationarity in time series forecasting. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 9881–9893, 2022.

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)*, volume 2024, pp. 11116–11140, 2024.

Spyros Makridakis, Evangelos Spiliotis, and Vassilios Assimakopoulos. The M5 accuracy competition: Results, findings, and conclusions. *International Journal of Forecasting*, 38(4):1346–1364, 2022.

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.

Syama Sundar Rangapuram, Lucien D. Werner, Konstantinos Benidis, Pedro Mercado, Jan Gasthaus, and Tim Januschowski. End-to-end learning of coherent probabilistic forecasts for hierarchical time series. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139, 2021.

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

Aris A. Syntetos and John E. Boylan. The accuracy of intermittent demand estimates. *International Journal of Forecasting*, 21(2):303–314, 2005.

James Tobin. Estimation of relationships for limited dependent variables. *Econometrica*, 26(1): 24–36, 1958.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y. Zhang, and Jun Zhou. TimeMixer: Decomposable multiscale mixing for time series forecasting. In *International Conference on Learning Representations (ICLR)*, pp. 38626–38652, 2024a.

Yangyang Wang, Jiawei Gu, Li Long, Xin Li, Li Shen, Zhouyu Fu, Xiangjun Zhou, and Xu Jiang. FreshRetailNet-50K: A stockout-annotated censored demand dataset for latent demand recovery and forecasting in fresh retail. *arXiv preprint arXiv:2505.16319*, 2025.

Yuxuan Wang, Haixu Wu, Jiaxiang Dong, Yong Liu, Mingsheng Long, and Jianmin Wang. Deep time series models: A comprehensive survey and benchmark. *arXiv preprint arXiv:2407.13278*, 2024b.

Christian H Weiß. *An Introduction to Discrete-Valued Time Series*. John Wiley & Sons, 2018.

Shanika L. Wickramasuriya, George Athanasopoulos, and Rob J. Hyndman. Optimal forecast reconciliation for hierarchical and grouped time series through trace minimization. *Journal of the American Statistical Association*, 114(4):1–45, 2018.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 22419–22430, 2021.

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.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 37, pp. 11121–11128, 2023.

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 *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 35, pp. 11106–11115, 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 *Proceedings of the 39th International Conference on Machine Learning (ICML)*, pp. 27268–27286. PMLR, 2022.

A Mechanism Details and Derivations

A.1 The M1 and M3 data types

Type M1: smooth continuous series (contrast class). Classical benchmarks inhabit the smooth continuous regime $x_t = \mu_t + \varepsilon_t$, continuous-valued and densely sampled, with additive continuous noise rather than the count noise of M2, and with a predictable component that is largely a function of the calendar rather than a recursion on the observed past. At one end of the suite sit trend-stationary members with clear deterministic trend and seasonality (Traffic, Weather, Electricity), at the other difference-stationary members close to a random walk (Exchange, Solar). Its fingerprint is R^2_{last}, the R^2 of regressing the last increment Δx_{t+1} on the level x_t, which locates a series on that span and measures empirically how much level information a historical anchor can carry. High values indicate mean-reverting (trend-stationary) structure and low values random-walk (difference-stationary) structure. It is used in Sec. 5.4 and App. C.1.

Type M3: a boundary regime, latent-driven Cox process. At the extreme of low autocorrelation the observation-driven feedback is too weak to exploit, and the series approaches a *latent-driven Cox process*, Poisson counts whose rate $u_t = \rho_u u_{t-1} + \eta_t$ evolves independently of past observations. This is what Eq. 1 becomes as the observation-driven coefficients $(\alpha, \{\gamma_\ell\})$ vanish, since the rate keeps evolving but no longer responds to observed counts. M3 is thus the *noise-dominated boundary* of the demand types. In this mechanism, past observations carry too little signal to align the total, so daily accuracy should be the sole objective (Sec. 3.3).

A.2 Derivation of the gain-side crossing (Proposition 3)

Setting. The window $y_1, \ldots, y_W \mid \mu \overset{\text{iid}}{\sim} \text{Poisson}(\mu)$ is a mixed-Poisson sample. A persistent level μ is drawn per series (the zero-feedback, window-constant-rate limit of Eq. 1), with $\mathbb{E}[\mu] = m$ and $\text{Var}(\mu) = \sigma^2$ across series. Then the following holds.

(i) For two days $h \neq h'$ within the window, conditional independence given μ gives $\text{Cov}(y_h, y_{h'} \mid \mu) = 0$, so $\text{Cov}(y_h, y_{h'}) = \text{Cov}(\mathbb{E}[y_h \mid \mu], \mathbb{E}[y_{h'} \mid \mu]) = \text{Var}(\mu) = \sigma^2$ (law of total covariance); and $\text{Var}(y_h) = \mathbb{E}[\text{Var}(y_h \mid \mu)] + \text{Var}(\mathbb{E}[y_h \mid \mu]) = m + \sigma^2$. Hence the level share is $\sigma^2/(m + \sigma^2)$.

(ii) $\text{Var}(S) = \text{Var}(\mathbb{E}[S \mid \mu]) + \mathbb{E}[\text{Var}(S \mid \mu)] = \text{Var}(W\mu) + \mathbb{E}[W\mu] = W^2\sigma^2 + Wm$.

(iii) The footprint $W^2\sigma^2$ exceeds the floor Wm iff $\sigma^2 > m/W$, i.e. iff $m/\sigma^2 < W$. Since $\sigma^2/(m + \sigma^2) = 1/(1 + m/\sigma^2)$, this is equivalent to $\sigma^2/(m + \sigma^2) > 1/(W + 1)$, the crossing ρ_0.

□

The difference in expected sales across series, which is present before any noise model is assumed, the Poisson form only sharpens the noise floor to m, so the signal-versus-noise comparison resolves into the footprint-to-floor ratio $\rho_0 = 1/(W + 1)$.

B EXPERIMENTAL SETUP

Datasets and windows. We evaluate on three demand datasets, and ten classical benchmarks (Table 2) for the cross-domain diagnostic.

Table 2: Datasets and evaluation protocols. FRN (Wang et al., 2025), SPR, and M5 (Makridakis et al., 2022) are the demand datasets; the classical benchmarks follow the TSLib protocol (Wu et al., 2023; Wang et al., 2024b). W is the sales cycle that enters both fingerprints and the aggregate-supervision term.

Dataset	Series	Input	Horizon	W	Metric
FreshRetailNet-50K	50,000	62	7	7	WAPE/WPE, day-level stock-out excl.
SPR (internal)	53,552	356	21	7	WAPE/WPE, day-level stock-out excl.
M5 Forecasting-Acc.	30,490	364	28	7	WAPE/WPE; official WRMSSE
Classical (10)	—	96 (ILI 36)	96 (ILI 24)	—	MSE/MAE

Table 3: Datasets and demand-side fingerprints (the table referenced from Sec. 5.1). ρ is the lag-1 ACF per series averaged over series and R^2_{last} is the R^2 of $\Delta x_{t+1} \sim x_t$, both read as in Sec. 3.2 (M5 on its active 364-day window; per-dataset windows in App. B).

Dataset	Domain	ρ	R^2_{last}	zero-mass	mechanism	role
FreshRetailNet-50K	retail	0.236	0.386	0.2%	M2	λ=1 regime (Exp. A)
M5 Forecasting-Acc.	retail	0.112	0.466	42.9%	level-mixed	gray zone (Exp. B)
SPR	retail	0.084	0.477	37.3%	M3 boundary	λ=0 regime (Exp. B)

Evaluation metrics. All targets are non-negative counts and every reported run predicts non-negative values, so each denominator below is a plain sum of actuals. WAPE and WPE form the ratio per forecast day and then average over the horizon $h = 1, \ldots, H$, with i indexing the series entering that day's denominator:

$$\text{WAPE} = \frac{1}{H}\sum_h \frac{\sum_i |\hat{y}_{ih} - y_{ih}|}{\sum_i y_{ih}}, \qquad \text{WPE} = \frac{1}{H}\sum_h \frac{\sum_i (\hat{y}_{ih} - y_{ih})}{\sum_i y_{ih}}, \tag{8}$$

One formula serves all three demand datasets, only the exclusion mask on i differs. FRN and SPR have a day-level mask that drops every row flagged out of stock and keeps each series' remaining days. M5 carries no stock-out annotation and so applies no mask. Days are weighted equally by construction, so a single high-volume day does not dominate the score.

For M5 we also report the competition's official WRMSSE, the equally weighted mean over its twelve aggregation levels of the dollar-sales-weighted RMSSE. At level ℓ and aggregate series g, with $T_{\ell g}$ the length of g's active period counted from its first non-zero day and Δ the first difference along it,

$$\text{RMSSE}_{\ell g} = \sqrt{\frac{\frac{1}{H}\sum_h (\hat{Y}_{\ell g h} - Y_{\ell g h})^2}{\frac{1}{T_{\ell g}-1}\sum_{t=2}^{T_{\ell g}} (\Delta Y_{\ell g t})^2}}, \qquad \text{WRMSSE} = \frac{1}{12}\sum_{\ell=1}^{12} \frac{\sum_g w_{\ell g}\, \text{RMSSE}_{\ell g}}{\sum_g w_{\ell g}}, \tag{9}$$

where $w_{\ell g}$ is the aggregate's dollar sales over the last 28 training days. The L_{12} column of Table 1 is the item×store level's own weighted RMSSE, and $L1$–$L9$ and $L10$–$L12$ in App. C.4 are the corresponding averages over the nine upper and the three item-level scores.

On the ten classical benchmarks we report the standard scale-dependent pair,

$$\text{MSE} = \frac{1}{H}\sum_h (\hat{y}_h - y_h)^2, \qquad \text{MAE} = \frac{1}{H}\sum_h |\hat{y}_h - y_h|, \tag{10}$$

taken as the mean over series and channels under the TSLib protocol.

Training schedules. PTNET uses $d = 128$, 4 heads, and quantiles $[0.25, 0.5, 0.75]$ on all three demand datasets; only the input scale and the schedule below differ (Table 4). Batch and epoch counts align with the TSLib comparison models on all three demand datasets.

Table 4: PTNET training schedule on the demand datasets. The M5 window cap is per series and independent of the 364-day input of Table 2: 44% of M5 series have more than a year of leading zeros, so the cap bounds the history fed while the input sets the look-back used (App. B).

Dataset	Batch	Epochs	lr	Note
FRN	1024	6	10^{-3}	—
SPR	768	5	2×10^{-3}	—
M5	768	5	2×10^{-3}	training window capped at 730 days

Every *self-run* model, which covers PTNET, the baselines we train ourselves (PatchTST, TiDE, iTransformer, TimeMixer, TimesNet, and DLinear on M5), and the INGARCH$(1, 1)$ reference rows of Table 1, is trained and evaluated once per configuration under a fixed seed (42) in the same harness. The FRN numbers for DLinear, TFT, and the two-stage TFT+TimesNet pipeline are quoted from the benchmark paper (Wang et al., 2025) and marked as published in Table 1; the classical-suite tables (App. C.1) are entirely self-run. Training schedules differ across datasets but never across models within a dataset, except on the baseline side. TimesNet runs at d_{16}/ff_{32}: TSLib sets TimesNet's width per dataset, spanning d_8–d_{768} across its shipped scripts, and 16/32 is the low-channel width it uses for ETTh1 and the same M-competition family's M4-Yearly. TimeMixer's M5 input is padded to 376 for its pooling alignment, with PTNET at its own 364. The learning rate is likewise shared within each dataset (10^{-3} on FRN, 2×10^{-3} on SPR and M5). PatchTST is the one outlier at 2×10^{-3} on M5: 74.90 WAPE, against 70.51 at 10^{-3}. The seed sensitivity of PTNET's champion configurations is reported in Sec. 5.2.

Input protocol and fairness. PTNET consumes the target history, future-known covariates (calendar, on FRN the dataset's weather and discount channels), and static hierarchy attributes (SKU/category/store IDs on FRN, state/store/category/department/item on M5). On FRN the TSLib baselines run in the benchmark's multivariate setting, with all six channels in the encoder and future calendar in the decoder marks, which is the protocol the benchmark paper prescribes for its own TFT baseline and PTNET follows it. The one remaining asymmetry is the static hierarchy IDs, which no TSLib baseline consumes, and a *no-static* ablation (all static context zeroed, architecture and parameter count unchanged) measures its contribution directly (Table 11, App. C.7). Static context is worth 1.44 pt of the FRN margin, while the stripped model still clears the best single-stage baseline by 1.68 pt, so the margin is mechanism rather than inputs.

C Additional Results

C.1 Cross-domain diagnosis: full detail and controls

Sec. 5.4's cross-domain experiment in full. The design isolates the anchor, which means aggregate supervision is disabled, the loss is MSE, the backbone is fixed at DLinear style, and the anchor is the only component that differs from the DLinear baseline. Table 5 gives the controlled pair per dataset, with PatchTST and TimesNet as M1-native references.

Table 5: Cross-domain diagnosis (MSE/MAE, TSLib protocol). Bold = better of the controlled pair, judged on MSE (both cells marked), underline = best M1-native reference (PatchTST/TimesNet). R^2_{last} measures the level-informativeness of increments.

	PtNet		DLinear		PatchTST		TimesNet		
Dataset	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	R^2_{last}
Traffic	**0.571**	**0.376**	0.697	0.429	0.488	0.321	0.585	0.311	0.079
Weather	**0.189**	**0.261**	0.196	0.255	0.172	0.214	0.176	0.223	0.054
Electricity	**0.185**	**0.294**	0.210	0.302	0.184	0.276	0.171	0.274	0.046
ETTh1	0.417	0.445	**0.397**	**0.412**	0.382	0.405	0.426	0.433	0.034
ETTh2	1.046	0.751	**0.340**	**0.394**	0.318	0.361	0.333	0.369	0.021
ETTm1	0.362	0.399	**0.346**	**0.374**	0.331	0.368	0.379	0.399	0.012
ETTm2	0.218	0.327	**0.193**	**0.293**	0.182	0.265	0.193	0.268	0.008
Exchange	0.131	0.273	**0.098**	**0.233**	0.096	0.214	0.136	0.255	0.001
ILI	5.118	1.586	**4.794**	**1.658**	2.299	0.954	2.423	0.935	0.017
Solar	**0.212**	**0.323**	0.290	0.378	0.242	0.290	0.237	0.274	0.004

In Figure 2, the controlled pair's gain crosses zero where the fingerprint passes through the gray zone (the rightmost loss sits at R^2_{last}=0.034 and the leftmost gain at 0.046, so the 0.04 boundary bisects the gap), while the two references show no crossing of their own. PatchTST, which removes the level, dominates exactly where R^2_{last} is low. Solar is a low-R^2_{last} dataset where the anchor wins because the exogenous inputs (irradiance, day length) carry signal the history-only fingerprint cannot see. R^2_{last} prices the *history* channel alone, so it should be read jointly with any strong exogenous channel, as the demand-side criterion is read jointly over $(\rho,\ \text{zero mass})$.

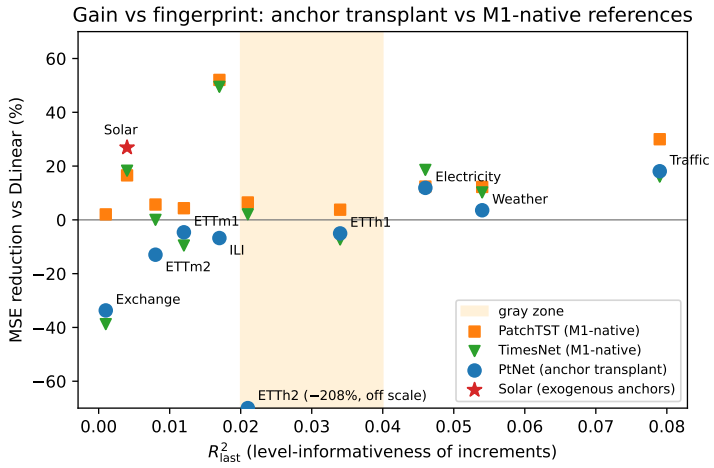

Figure 2: Gain (MSE reduction vs DLinear) as a function of R^2_{last}. PtNet (blue; Solar starred) is the only series whose gain tracks the fingerprint across the zero crossing, positive for all datasets above R^2_{last}=0.04 and negative for every dataset below 0.02 except the starred Solar, whose gain rides on an exogenous channel, while the M1-native references (PatchTST, TimesNet) show no dependence on R^2_{last}.

C.2 COMPUTE AND PARAMETER ACCOUNTING ON FRN

Table 6 reports parameter counts for PTNET and the five self-run deep baselines on FreshRetailNet-50K, with the TSLib configurations used in the runs; all models train for 6 epochs under the benchmark protocol on a single GPU. PTNET sits at the parameter-efficient end among attention-based models (1.33M vs PatchTST's 6.34M and iTransformer's 4.77M) and trains end-to-end in 59 minutes. The aggregate-supervision term adds no parameters by construction and negligible time, since the windowed sums are a reshape of tensors already on device.

Table 6: Parameter counts on FreshRetailNet-50K (self-run models, TSLib configurations).

Model	Parameters
PTNET (λ=1.0)	1.33M
PatchTST	6.34M
iTransformer	4.77M
TimesNet	0.59M
TiDE	0.44M
TimeMixer	0.04M

C.3 MULTI-SEED STABILITY

Table 7 re-trains each champion configuration under five seeds with nothing else changed, seed 42 being the published run of Table 1. Daily accuracy is tight on M5 and FRN and looser on SPR, the population drifting toward the M3 boundary, where less level information survives in the history for a seed to reproduce.

Table 7: Multi-seed stability of the champion configurations. Each configuration is re-trained under five seeds; seed 42 is the published run and no other setting changes. FRN runs at λ=1, SPR at λ=0, and M5 at λ=0 for the daily caliber and λ=0.5 for the aggregate one, so each M5 block is read in its own champion caliber. The spread row is $\max - \min$ within a column, in that column's own unit. Bold marks the best entry per column (WPE by closeness to zero).

	FRN (λ=1)		SPR (λ=0)		M5 (λ=0)		M5 (λ=0.5)	
seed	WAPE	WPE	WAPE	WPE	WAPE	WPE	WRMSSE	L_{12}
42 (published)	**27.94**	**-2.19**	66.15	-18.02	67.19	-34.55	0.904	0.843
7	29.60	-13.27	67.95	**-12.62**	67.18	-34.21	0.978	**0.837**
13	28.88	-8.16	65.44	-20.87	67.12	-36.86	1.187	0.853
99	27.99	+2.16	**64.76**	-28.68	**67.10**	-36.00	1.145	0.852
2026	28.91	+0.82	65.73	-23.49	67.19	**-28.28**	**0.895**	0.842
median	28.88	-2.19	65.73	-20.87	67.18	-34.55	0.978	0.843
spread	1.66	15.43	3.19	16.06	0.09	8.58	0.292	0.016

C.4 M5 OFFICIAL-BENCHMARK COMPARISON (SAME WINDOW)

The competition's organizers publish, alongside the final leaderboard, twenty-four benchmark methods scored on the private window d_{1942}–d_{1969}, the same window our Table 9 replication evaluates. These are single published forecasting methods like PTNET itself, unlike the competition's fifty finalist teams (gradient-boosted ensembles over engineered features with hierarchical reconciliation, 0.52–0.58), which are not comparable to a single model. Table 8 places PTNET's private-window aggregate champion (λ=0.5) against the published benchmarks.

The table shows the six suites ahead of PTNET on the average, and taking the best of the six level by level they lead on all of $L1$–$L11$, with PTNET taking the top spot only at the item–store level $L12$. All six obtain aggregate coherence by hierarchical reconciliation as a post-processing step. PTNET obtains coherence only partially, through the λ term inside a single model. λ=0.5 buys most of the aggregate-coherence benefit without abandoning per-series accuracy, which is the trade the gray-zone criterion prices. The rows below PTNET make the position concrete, from the seasonal-naive

Table 8: Same-window comparison on the M5 final private window (official WRMSSE caliber; lower is better). Benchmark rows are the competition's published final-leaderboard entries; the PTNET row is our private-window replication (never used for tuning). Best per column in bold (ties both bolded). $L1$–$L9$ is the mean of the nine upper aggregation levels and $L10$–$L12$ the mean of the three item-level ones, both taken over the level values at full precision. The six suites ahead of PTNET (0.671–0.723) all obtain aggregate coherence by hierarchical reconciliation as a post-processing step (top-down for ESX, ARIMAX, and ES_td; bottom-up for ES_bu and Com_b; Com_tb averages the two), and the two exogenous-variable variants (ESX, ARIMAX) additionally regress on the SNAP-day count and an event flag; PTNET applies no reconciliation, but consumes the same two signals as calendar inputs, alongside the log price level.

	Avg	$L1$–$L9$	$L10$–$L12$	L_{12}
ES_bu (best benchmark)	**0.671**	**0.573**	0.965	0.915
ESX	0.679	0.581	0.973	0.917
Com_tb	0.682	0.590	**0.958**	**0.905**
ARIMAX	0.691	0.599	0.966	0.912
ES_td	0.720	0.636	0.972	0.915
Com_b	0.723	0.643	0.960	0.907
PTNET λ=0.5 (single model)	0.726	0.647	0.963	**0.905**
sNaive	0.847	0.729	1.201	1.176
Intermittent classics (range over members)	0.955–0.961	0.941–0.949	0.993–1.005	0.919–0.928

baseline down to the intermittent-demand classics family standardly recommended for sparse count demand.

C.5 ADDITIONAL λ-SWEEP DETAIL

Table 9 gives the full eleven-point grid. A dense SPR re-sweep between 0 and 0.1 adds four points (66.40, 67.70, 67.05, 69.47), none better than λ=0 at 66.15. M5's twelve-level WRMSSE is optimal at λ=0.5 on both the official and the withheld private window. The official window has one interior exception: λ=0.4 is worse than both of its neighbours on daily WAPE, WRMSSE, and L_{12} (WAPE +1.53 pt over λ=0.3). That exception does not replicate on the private window, where the WRMSSE falls monotonically from λ=0.3 to λ=0.5. The series-level gradient beneath that split is plotted in Figure 3.

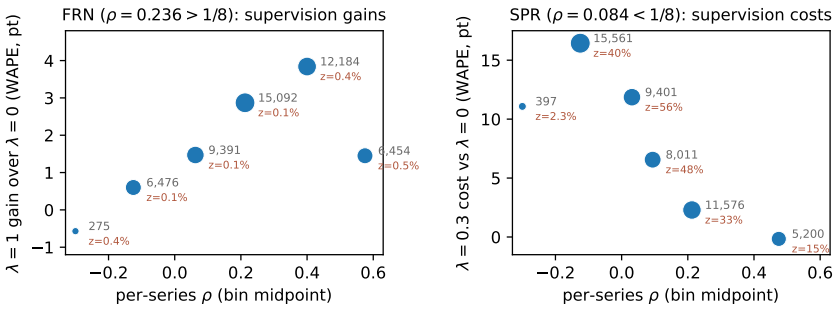

Figure 3: The series-level gradient beneath the population-level rule. Series are binned by their ρ of Sec. 3.2. The left panel shows FRN's per-series-ρ-bin gain of λ=1 over λ=0 (WAPE, pt) and the right panel SPR's per-series-ρ-bin cost of λ=0.3 relative to λ=0 (WAPE, pt). Point area encodes series count per bin; the annotation on each point gives the series count n and the bin's zero-mass share z (Eq. 2, W=7), the two dimensions the population-level rule weighs.

Table 9: The full λ grid across the three demand datasets. Metrics are no stockout daily horizon WAPE/WPE (%) for FRN and SPR, and daily WAPE/WPE, the twelve-level WRMSSE, and the item×store level L_{12} on the official window d_{1914}–d_{1941} for M5 (L_{12} is the finest of the twelve official levels and the dimension PTNET itself models), with the private window d_{1942}–d_{1969} replication of the full grid in parentheses, never used for tuning. Best per column in bold and second best underlined (WPE by closeness to zero, ties all marked). The populations order exactly as Corollary 4 requires.

	FRN (ρ=0.236)		SPR (ρ=0.084)		M5 (ρ=0.112)			
λ	WAPE (%)	WPE (%)	WAPE (%)	WPE (%)	WAPE (%)	WPE (%)	WRMSSE	L_{12}
0.0	30.30	-7.50	**66.15**	-18.02	**67.19** (**70.59**)	-34.5 (-28.3)	2.011 (1.745)	0.877 (0.927)
0.1	30.21	-6.73	76.25	+7.12	67.85 (72.34)	-23.1 (-13.5)	1.439 (1.064)	0.850 (0.905)
0.2	29.33	-6.96	73.70	**-0.53**	69.15 (74.15)	-14.3 (-7.0)	1.051 (0.829)	0.841 (0.906)
0.3	28.97	-8.44	69.52	-13.76	69.78 (74.53)	-12.1 (-6.1)	0.966 (0.820)	**0.840** (0.906)
0.4	28.58	-8.93	73.03	-2.22	71.31 (74.77)	-9.7 (-5.0)	1.238 (0.807)	0.855 (**0.903**)
0.5	28.41	-4.70	71.52	-12.03	71.24 (75.52)	-6.7 (-2.7)	**0.904** (**0.726**)	0.843 (0.905)
0.6	28.45	-4.90	73.17	-10.74	71.67 (75.63)	-6.1 (-3.7)	0.962 (0.814)	0.847 (0.905)
0.7	28.43	-4.80	71.99	-11.43	71.93 (76.52)	-5.5 (+1.7)	0.990 (0.800)	0.850 (0.909)
0.8	28.28	-3.42	71.13	-16.08	71.75 (76.51)	-7.1 (-1.6)	1.073 (0.954)	0.848 (0.909)
0.9	28.20	-2.79	73.85	-9.28	72.30 (76.03)	**-5.2** (-2.9)	1.100 (0.924)	0.853 (0.908)
1.0	**27.94**	**-2.19**	71.80	-15.08	76.28 (78.83)	-6.2 (**-1.1**)	1.771 (1.482)	0.931 (0.951)

C.6 SUBGROUP RE-TRAINING PROTOCOL

The experiment of Table 10 uses each dataset's champion configuration unchanged. The only manipulated variables are the training series population, selected by train-history $\mathrm{ACF}(1)$ relative to $1/8$, and λ. On SPR the swept endpoints are $\lambda \in \{0, 0.3\}$, where 0.3 is the best aggregate-side point of the stable plateau. The metric is the WAPE/WPE caliber of App. B over the full forecast window, under the day-level stock-out mask that FRN and SPR share. No population flips, including SPR's high-ρ subgroup, where ρ alone would license supervision but the joint criterion vetoes it.

Table 10: Subgroup re-training. Each row re-trains PTNET on one series population.

Population	λ_a / λ_b	# series	WAPE@λ_a	WAPE@λ_b	Δ (pt)	zero-mass
FRN low-ρ ($< 1/8$)	$0 / 1$	16,280	36.32	35.35	$+0.97$	0.06%
FRN high-ρ ($\geq 1/8$)	$0 / 1$	33,720	29.33	26.94	$+2.39$	0.31%
FRN full	$0 / 1$	50,000	30.30	27.94	$+2.36$	0.22%
SPR high-ρ ($\geq 1/8$)	$0 / 0.3$	16,485	59.79	62.87	-3.08	27.2%
SPR full	$0 / 0.3$	53,552	66.15	69.52	-3.37	37.3%

C.7 FULL COMPONENT ABLATION TABLE

Table 11 lists every component-removal variant behind Sec. 5.5, one row per ablation on FreshRetailNet-50K at $\lambda = 1.0$ under the full run's protocol. The no-static row carries the fairness argument, since no classical baseline consumes static IDs at all. Zeroing the static context leaves architecture and parameter count unchanged, costs 1.44 pt of WAPE, and still clears every single-stage forecaster without any static input.

Table 11: Full component ablation on FreshRetailNet-50K ($\lambda = 1.0$). Best per column in bold; Δ columns show the change from Full (positive ΔWAPE = degradation), and the WPE-column bold is by closeness to zero.

Variant	Overall	Low	High	WPE	ΔWAPE	ΔWPE
Full	**27.94**	35.47	**22.54**	-2.19	–	–
No static (all contexts zeroed)	29.38	36.88	23.99	-4.71	+1.44	-2.52
w/o gate	29.17	36.30	24.04	-12.21	+1.23	-10.02
w/o residual	28.40	**35.42**	23.38	-4.00	**+0.46**	**-1.81**
w/o spread	28.68	36.25	23.24	-9.35	+0.74	-7.16
Mean-only	31.35	38.86	25.96	**+2.17**	+3.41	+4.36
No VSN/GRN (all gating $\rightarrow$ linear, 0.49M params)	28.97	36.72	23.40	-5.88	+1.03	-3.69