LEVEL-SHIFT FAILURES IN SPATIOTEMPORAL KRIGING WITH TARGET-NODE HISTORY: DIAGNOSIS AND REPAIR

Anonymous authors
Paper under double-blind review

ABSTRACT

We study level-shift failures in spatiotemporal kriging when target nodes' current values are masked but their histories remain available. KRIN subtracts each node's recent mean and restores it after prediction. Under a common reconstruction protocol, it reduces MAE in 37 of 40 configurations using released implementations, but none of the centred released models outperforms seasonal persistence with current common-mode correction (+CM) on Chicago or Subway under the COVID-era evaluation condition. A matched no-shift control separates observation-placement difficulty from its additional effect under a local shift. After normalising each run's error by seasonal persistence on the same scored entries, both a simplified SPIN model and an adapted released IGNNK implementation show an increased outside-observation penalty in two cities. KRIN reduces this increase in all 15 combinations of three training seeds and five observation placements for each backbone–dataset comparison; the mean remaining increase is close to zero. Immediately after a uniform shift, +CM achieves lower error than the learned configurations, while shift alarms have limited precision for identifying failures after centring. These findings distinguish broad error reduction from superiority over a strong baseline and identify conditions under which restoring recent target-node levels reduces sensitivity to observation location.

1 INTRODUCTION

We study spatiotemporal kriging when historical observations remain available for the target nodes. Because only the target-day values are masked, copying the value from one week earlier, $\hat{x}_t^{(i)} = x_{t-7}^{(i)}$, is a valid training-free baseline. Its common-mode-corrected version (+CM) incorporates current changes among observed nodes and provides a stronger test of a learned model's predictive value.

Under a common protocol, five simplified models and STGNN have higher error than the seasonal baseline in 10 of 72 Normal configurations. In 2020, they have higher error in 49 of 54 configurations across three datasets whose levels move outside the previously observed range, but in none of 18 configurations on Bikeshare. The corresponding system-average shifts are 2.406–3.845 and 0.797, measured in units of node-specific training standard deviations. A graph-free affine model also fails against the baseline, showing that spatial aggregation is not necessary for this failure.

Removing and restoring the level follows principles used in RevIN (Kim et al., 2022), NLinear (Zeng et al., 2023) and SIN (Han et al., 2024). Our question concerns the observations that supply level information. KRIN uses each node's recent historical mean, +CM estimates the current common change, and spatial aggregation uses observed neighbours. Their usefulness depends on which nodes are observed and how recently the level has changed.

Our analysis focuses on three findings. First, we distinguish error reduction from superiority over a stronger baseline when evaluating released implementations. Second, matched no-shift and local-shift conditions hold targets and observation sets fixed, testing the shift-related outside-observation penalty in SPIN-s and released IGNNK-R. Third, we use errors immediately after a transition and the precision of shift alarms to distinguish shift detection from model-failure identification. Simplified

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.

variants (-s) support analyses under a common protocol; adapted released implementations (-R) provide evidence about the original architectures.

2 Related Work

Kriging estimates values at unobserved locations from observations elsewhere (Cressie, 1993). Graph-based methods include KCN's nearest-neighbour subgraphs (Appleby et al., 2020), IGNNK's random-subgraph training (Wu et al., 2021a), SATCN's masked aggregation (Wu et al., 2021b), and KITS's virtual nodes (Xu et al., 2025). Related imputation methods include GRIN's graph recurrence (Cini et al., 2022; Cho et al., 2014) and SPIN's spatiotemporal attention (Marisca et al., 2022; Vaswani et al., 2017). Their observation protocols determine which training-free baselines are admissible.

Protocol A reconstructs entirely unobserved nodes without their histories, as in IGNNK, SATCN and KITS. Node-specific persistence and history means are undefined, so their omission is appropriate. Protocol B imputes missing node–time pairs, as in GRIN and SPIN. Other observations of the same sensor can remain available, making last observation carried forward (LOCF) (Little & Rubin, 2019) computable without training. Neither paper's baseline list includes temporal persistence (Appendix B.2.6). Our protocol fixes the target-node set within a run but masks only target-day values. Both HA and `Naive-k` are therefore admissible (§3). Baseline comparisons must follow these information constraints.

Simple baselines can outperform neural methods in recommendation (Ferrari Dacrema et al., 2019), long-horizon forecasting (Zeng et al., 2023) and graph learning (Shchur et al., 2018). We examine when failures relative to persistence become widespread across kriging backbones and characterise the associated level shifts (§4).

Removing an input level and restoring it at the output follows RevIN (Kim et al., 2022) and NLinear (Zeng et al., 2023). We evaluate this operation for node reconstruction with available history (§5.1). DRIK (Yang et al., 2026) addresses spatial distribution shifts in inductive kriging; our focus is temporal level shifts with target-node history.

SIN (Han et al., 2024) uses historical–future covariance to select normalisation statistics and transformations. Berthelier et al. (2026) distinguish temporal, spatial and conditional output shifts, showing that some RevIN components can be unnecessary or harmful. Our contribution is consequently empirical: we examine how observation location and history availability affect recovery from level shifts, rather than claim a new mean-removal principle or benchmark new normalisation methods against these studies.

3 Problem Setting

Let $X \in \mathbb{R}^{T \times N}$ contain daily signals for N nodes over T days, with geographic coordinates available for each node. At evaluation time t, nodes are partitioned into observed and hidden sets, $\mathcal{O}$ and $\mathcal{H}$, with $|\mathcal{H}| = \lceil rN \rceil$. The model receives the history $H_t = [x_{t-K}, \ldots, x_{t-1}] \in \mathbb{R}^{K \times N}$, masked current observations $x_t \odot m$, and an observation-mask channel $m \in \{0,1\}^N$. The target is $\{x_t^{(i)} : i \in \mathcal{H}\}$; losses and evaluation metrics are computed only on hidden nodes.

Masking applies only to the target-day values of each evaluation sample. Although the hidden-node set is fixed within a run, target-day masks are not accumulated into the history masks of subsequent samples. A true observation masked for evaluation on day t can therefore enter the history on day $t+1$. This is daily reconstruction of a one-day gap, without recursively filling future input windows with predictions. It differs from reconstructing nodes that have never been observed or have stopped reporting indefinitely. The available history makes HA, `SNaive-7` and the KRIN window mean well defined. Experiments with pointwise or contiguous missingness apply a separate history mask (Appendix C.1).

Short reporting gaps also occur in the raw data. Among 3,326 observed reporting gaps in Airports, 74% last exactly one day; among 4,032 episodes in Subway, 99% last at most one week. Bikeshare and Chicago have no internal reporting gaps, so this empirical motivation applies to two of the four datasets (Appendix B.2.7).

We evaluate two conditions, Normal and Shock. Normal uses a chronological 70/10/20 split. Shock trains on the pre-pandemic period, reserving its final 15% for validation. Shock evaluation covers all 366 days of 2020 for Bikeshare, Chicago and Subway. For Airports, we distinguish `shock2020` (366 days in 2020), `shock2021` (365 days in 2021) and `all` (731 days in 2020–21). The main simplified-model results use `shock2020`, while the released implementations use `all`; each table states its window. Direct comparisons between these evaluations use `all` for every Airports column in Table 56 (Appendix B.1.1).

4 Baseline Failures under Level Shift

4.1 Baselines and Evaluation Criteria

We consider four training-free baselines. `HA` averages a target node's K-day history; `Naive-1` copies its value from the previous day; `SNaive-7` copies its value from one week earlier; and `KNN-geo` takes a weighted mean over observed neighbours in A^{geo}. Persistence is standard in forecasting (Hyndman & Athanasopoulos, 2021), and LOCF is standard in missing-data analysis (Little & Rubin, 2019). Our concern is whether these admissible baselines are reported under the protocols discussed in §2. Distinguishing `Naive-1` from `SNaive-7` matters because weekly seasonality can make the lag choice consequential. In Airports `shock2020`, their MAEs are 14.9 and 12.6, respectively; in the more weather-sensitive Bikeshare data, they are nearly identical.

We also use `SNaive-7`+CM, a training-free common-mode correction. For a hidden target node i,

$$\log(1+\hat{x}_t^{(i)}) = \log(1+x_{t-7}^{(i)}) + \text{median}_{j\in\mathcal{O}}\left[\log(1+x_t^{(j)}) - \log(1+x_{t-7}^{(j)})\right]. \tag{1}$$

The correction adds the median current-versus-weekly log change among observed nodes to the target's weekly reference. It therefore requires the target node's value at $t-7$, as well as observed nodes' values at t and $t-7$. Its use of current common changes explains its low error immediately after the uniform-shift intervention.

Table 1: Simplified variants under a common protocol: failures relative to seasonal persistence and level-shift diagnostics at $r = 0.5$. Errors are Shock MAE; counts compare six backbones with `SNaive-7` in each condition. Shock errors, failure counts and diagnostics all use 2020, including Airports (366 days). Normal failure counts use the Normal window. The diagnostic and below-minimum fraction use the system-level series defined in §4.2.

Dataset	`SNaive-7`	Best backbone without KRIN	|shift|	Eval days below the seen minimum	Shock: worse backbones	Normal: worse backbones
Bikeshare DC	10.8	6.4 (SPIN-s)	0.797	0.0%	0 / 6	0 / 6
NYC Subway	1095.6	1253.9 (SATCN-s)	2.406	56.8%	6 / 6	1 / 6 (GRIN-s)
Airports	12.6	10.6 (IGNNK-s)	3.845	37.7%	5 / 6	1 / 6 (GRIN-s)
Chicago 'L'	169.7	188.2 (IGNNK-s)	3.124	23.2%	6 / 6	1 / 6 (GRIN-s)

In Chicago and Subway, even the best uncentred backbone has higher error than the seasonal baseline. At $r = 0.5$, all five backbones other than GRIN-s outperform the baseline under Normal conditions, but fail to do so in 14 Shock configurations. GRIN-s's failures under Normal conditions require a separate interpretation: they disappear when its released implementation is used. We therefore do not attribute this behaviour of the simplified variant to the original GRIN method (Appendices B.2.5 and E). Full model and baseline errors are reported in Appendices A.1 and A.2.1.

Raw demand declines alone do not explain the difference across datasets. April 2020 demand was 24.1% of its pre-shock level in Bikeshare, 12.4% in Chicago, 11.9% in Airports and 10.1% in Subway, whereas the numbers of backbones with higher error than the seasonal baseline were 0 / 6 / 5 / 6. We therefore measure the level shift in the input space used by the models.

4.2 Level-Shift Diagnostic

We compute the diagnostic after applying `log1p` and nodewise z-standardisation fitted only on the training rows of the corresponding condition. Under Shock, one unit corresponds to one pre-pandemic standard deviation of the transformed signal at that node.

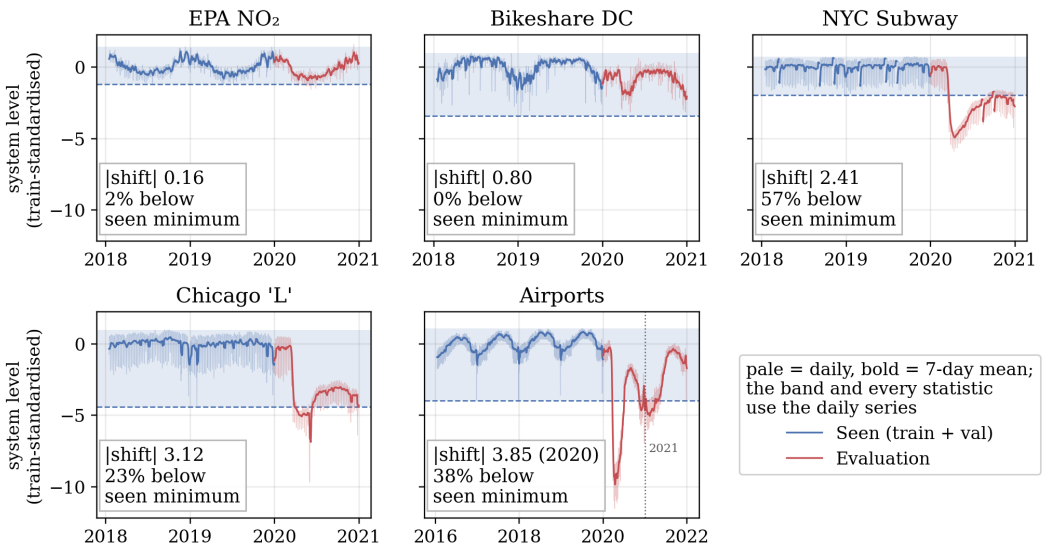


Figure 1: System-level trajectories in training-standardised space. Pale curves are daily values; bold curves are seven-day averages. Blue and red mark the seen and evaluation periods; the band spans daily seen levels. Diagnostics and below-minimum percentages use 2020 for all four mobility datasets, matching Table 1. Airports additionally displays 2021 for context. NO_2 is the held-out low-shift comparison.

Define the daily system level as the node average, $\ell(t) = \frac{1}{N}\sum_i \tilde{x}_t^{(i)}$. The diagnostic is

$$|\text{shift}| = \left| \bar{\ell}_{\text{test}} - \bar{\ell}_{\text{seen}} \right|$$

Here, "seen" combines the training and validation periods, both of which precede evaluation; the standardisation statistics themselves use training rows only. The diagnostic measures the change in the system-average level in node-specific training-standard-deviation units. Because a mean shift does not fully describe departures from the previously observed range, we also report the fraction of evaluation days below the seen minimum. Nodewise standardisation allows comparison across datasets with different raw scales. Further dividing the diagnostic by the training-period level range did not improve alignment in a prespecified test, so we do not use that additional denominator (Appendix C.2).

We define failure relative to the seasonal baseline as a model having higher Shock MAE than `SNaive-7`. This criterion identifies configurations that provide no improvement over persistence computed from the available input. Replacing `SNaive-7` with `HA` changes the failure counts and their ordering. However, `SNaive-7` is the stronger comparator on all three datasets whose levels leave the previously observed range (Appendix B.2.8).

Table 1 reports the shift diagnostic and the fraction of days below the seen minimum for matched 2020 windows; Figure 1 provides temporal context over the available evaluation periods. Airports uses the same `shock2020` window for MAE and diagnosis, giving $|\text{shift}| = 3.845$ and a below-minimum fraction of 37.7%. Its shift exceeds those of Chicago (3.124) and Subway (2.406), yet SPIN-s + KRIN has the smallest improvement over the best uncentred backbone among these three datasets. The diagnostic therefore does not monotonically predict the size of the improvement. Signed shifts and training-period ranges appear in Appendix A.2.5.

4.3 Level-Dependent Mappings as a Failure Mechanism

Nodewise z-standardisation fitted on training data does not remove a level shift at evaluation. A learned mapping that depends on absolute levels can therefore be a source of failure outside the training-observed range. To probe this mechanism, we measure the level-equivariance gap: how far a model's prediction departs from the same additive change applied to its inputs. Mean gaps across

the six backbones range from 0.086 to 0.118, so none is exactly equivariant at the tested inputs and offset. SPIN-s has the smallest gap, however, which does not support the stronger explanation that more expressive aggregators necessarily depend more on absolute levels (Appendix C.3).

If level-dependent learned mappings contribute to the failure, similar behaviour should be possible without graph aggregation. A 17-parameter affine model with no communication between nodes indeed has higher Shock MAE than `SNaive-7` on Chicago and Subway, and centring substantially reduces its error (Appendix C.4). Spatial aggregation is therefore not necessary for this failure. This control neither rules out architectural effects nor establishes that centring always suffices to outperform the baseline. By copying a past observation, `SNaive-7` directly carries forward the level present one week earlier, although this reference can lag immediately after a transition (Appendix C.2.3). These observations motivate separating the level component from the learned prediction pathway.

5 METHOD: KRIN

5.1 ADAPTING LEVEL REMOVAL TO NODE RECONSTRUCTION

We apply the level-removal and restoration principle of RevIN and NLinear to node reconstruction with available history. In a four-way comparison using the same SPIN-s backbone, removing the recent window mean yields lower error than removing the last value in seven of eight configurations. Additional mean/variance normalisation provides inconsistent gains, so we use mean-only centring in these experiments (Appendix B.2.2).

Separating node-specific levels also affects spatial aggregation. Under Normal conditions, with neighbours, weights and graph held fixed, adding nodewise centring to global scaling reduces the MAE of a weighted neighbour mean by factors of 3.2 in Airports, 2.3 in Chicago, 2.0 in Subway and 1.3 in Bikeshare (Appendix B.2.3). This comparison separates the effect of global scaling from the additional effect of nodewise centring.

5.2 KRIN

KRIN (Kriging Regime-Invariant Normalization) subtracts each node's mean over the previous K days and restores it after prediction. Below, $\tilde{H}_t = [\tilde{x}_{t-K}, \ldots, \tilde{x}_{t-1}]$ and $\hat{\tilde{x}}_t$ denote the history and prediction in the preprocessed space. Evaluation metrics are computed on the original scale after inversion.

$$\mu_i^{(t)} = \frac{1}{K} \sum_{k=1}^{K} \tilde{x}_{t-k}^{(i)},$$
$$\hat{\tilde{x}}_t = f_\theta\Big(\tilde{H}_t - \mu^{(t)}, (\tilde{x}_t - \mu^{(t)}) \odot m, m\Big) + \mu^{(t)}.$$

The mean uses only observations preceding the target day, never its hidden current value. KRIN introduces no trainable parameters: a backbone has the same parameter count with and without the wrapper (17,793 for IGNNK-s, for example). Retraining on centred inputs is required. Although mean-only restoration avoids multiplying predictions by a window standard deviation, it does not guarantee the stability of arbitrary network outputs or the exponential inverse transformation.

Level equivariance. For brevity, write the mapping in preprocessed space as $G(H, x, m) = f_\theta(H - \mu(H), (x - \mu(H)) \odot m, m) + \mu(H)$. For a fixed mask m, adding the same constant c to the history and current input gives

$$G(H + c, x + c, m) = G(H, x, m) + c$$

Since $\mu(H+c) = \mu(H)+c$, both centred inputs and the mask remain unchanged, while the restored mean increases by c. The equivariance gap is therefore zero in exact arithmetic. Appendix C.3 reports the numerical probe.

This guarantee requires the same shift to be present in the history. It does not cover windows mixing pre- and post-shift observations, stale histories, changes in seasonal patterns or dependencies, or

errors in predicting the centred residual. Moreover, multiplying the raw signal by a uniform factor is not exactly equivalent to adding a constant in the preprocessed space. We study transitions and masking of recent history separately through empirical comparisons (§7.3, 7.4).

Mean validation rank selects SPIN-s + KRIN as the representative configuration. Dataset-specific selection chooses SPIN-s in three cases and KITS-s in Airports (Appendix A.2.8); this does not establish a universally best combination. Simplified variants share a hidden dimension, but total parameter counts differ. We isolate KRIN's effect by comparing the same backbone before and after centring.

6 Experimental Setup

Table 2: Daily mobility datasets and raw-data coverage. Signal levels are per-node daily totals; April 2020 percentages use the corresponding pre-shock mean. Missing rates describe the input data before artificial target masking. No origin–destination flows are used.

	Bikeshare DC	Chicago 'L'	Airports	NYC Subway
Nodes N	413	143	221	437
Signal	Departures + arrivals	Station entries	Total flights	Turnstile entries + exits
Pre-shock mean	44.5	3,489.0	192.4	18,627.3
2020-04 (vs the pre-shock mean)	10.7 (24.1%)	431.1 (12.4%)	22.9 (11.9%)	1,879.0 (10.1%)
Missing rate	0.06%	0.00%	0.75%	3.95%

Bikeshare, Chicago and Subway cover 1,461 days from 2018 through 2021; Airports covers a longer period of 3,804 days starting in 2016. We use public data from Capital Bikeshare (Capital Bikeshare, 2026), the Chicago Transit Authority (Chicago Transit Authority, 2026), EUROCONTROL (EUROCONTROL, 2026) and the Metropolitan Transportation Authority (Metropolitan Transportation Authority, 2026). Graph-based experiments use the same construction for A^{geo}: a Gaussian kernel of haversine distance, sparsified to the top $k = 10$ neighbours and then symmetrised (Li et al., 2018). This standardises graph construction across datasets using coordinates alone. How well the resulting graph represents actual relationships can still vary by dataset.

The main experiments use $r \in \{0.25, 0.5, 0.75\}$ and 3 training seeds $\times$ 10 mask seeds, giving 30 evaluations per configuration. Models minimise hidden-node MSE in standardised space; evaluation uses the original scale. All receive the same history, masked current values and mask channel. The main protocol applies no additional history mask.

We distinguish the two evaluations by model names. `IGNNK-s`, `SATCN-s`, `GRIN-s`, `SPIN-s` and `KITS-s` are *simplified variants under a common protocol*, with STGNN as a standard control. They share a hidden dimension of 64 and a training procedure, allowing analysis of the simplified architectures and level handling. Total parameter counts differ, but each backbone's count is unchanged by KRIN. The suffix `-R` denotes models adapted from the authors' released code and provides evidence about the original architectures. This evaluation does not reproduce each source paper's full data, task and training strategy. In particular, KITS uses the common training procedure instead of its complete incremental training strategy (Appendix E).

The global/local-shift, transition, alarm and history-sensitivity analyses use $r = 0.5$ and 2 training seeds $\times$ 3 mask seeds. The redesigned observation-location control instead crosses training seeds $\{0, 1, 2\}$ with placement seeds $\{0, 1, 2, 3, 4\}$, giving 15 matched evaluations per configuration for SPIN-s and IGNNK-R, with and without KRIN. Evaluations sharing a training seed reuse a fitted model; they are not 15 independent training runs. Controlled shifts are applied only at evaluation, with weights fixed across interventions.

In the matched-shift experiment, the local intervention multiplies signals in clusters of 43 Chicago nodes or 131 Subway nodes by 0.25. Bisection selects the global multiplier to match the realised system-average shift. The location experiment then varies which nodes supply current observations while preserving the scored targets and observation count. These controls separate the spatial distribution of the shift from the location of available measurements (Appendix C.7).

MAE pools hidden-cell errors and is sensitive to high-volume nodes. NMAE(macro) divides each node's MAE by its mean signal level and weights nodes equally. Cohen's d measures standardised

effect size. For the 30-evaluation comparisons, paired 95% bootstrap intervals resample mask-seed clusters; evaluations share 3 training runs and are not independent. A hierarchical bootstrap that first resamples training seeds gives the same main conclusions (Appendix B.2.11).

7 Results and Analysis

7.1 Released Implementations and the Strengthened Baseline

Table 3 distinguishes error reduction from superiority over the strengthened baseline for the released implementations.

Table 3: Released implementations adapted to the common reconstruction protocol (-R): Shock MAE at $r = 0.5$. Arrows show uncentred $\rightarrow$ KRIN, averaged over 3 training seeds $\times$ 10 mask seeds. Airports uses 2020–21 (`all`); the other datasets use 2020. Both training-free baselines use the same evaluation windows.

Method	Bikeshare	Chicago	Airports (all)	Subway
`SNaive-7`	10.8	169.7	11.35	1,095.6
`SNaive-7`+CM	8.3	86.7	9.46	543.4
IGNNK-R	7.9 $\rightarrow$ 6.8	192.5 $\rightarrow$ 95.4	11.7 $\rightarrow$ 8.6	1,783.4 $\rightarrow$ 743.2
SATCN-R	9.3 $\rightarrow$ 6.7	265.6 $\rightarrow$ 111.9	43.6 $\rightarrow$ 12.1	2,467.8 $\rightarrow$ 1,103.4
GRIN-R	7.1 $\rightarrow$ 6.7	170.6 $\rightarrow$ 121.1	21.5 $\rightarrow$ 8.7	1,892.2 $\rightarrow$ 574.6
SPIN-R	6.7 $\rightarrow$ 6.3	569.2 $\rightarrow$ 267.8	21.7 $\rightarrow$ 15.0	3,330.6 $\rightarrow$ 738.4
KITS-R	9.8 $\rightarrow$ 7.0	296.9 $\rightarrow$ 126.3	42.2 $\rightarrow$ 12.2	2,207.3 $\rightarrow$ 1,149.7

Across Normal and Shock, centring reduces MAE in 37 of 40 configurations; all three increases in MAE occur for SPIN-R under Normal conditions. Under Shock, the number of configurations with higher error than `SNaive-7` falls from 15/20 to 6/20 (Appendix E.1). However, the best centred released-code models in Chicago and Subway have MAEs of 95.4 and 574.6, respectively, above the +CM values of 86.7 and 543.4. This ordering also holds for NMAE(macro); Table 57 reports these results and variability across training seeds.

The comparison with +CM varies across datasets: all five centred released implementations outperform it on Bikeshare, whereas only IGNNK-R and GRIN-R do so on Airports. Centring also reduces variation across fitted SPIN-R models: the training-seed standard deviation falls from 292.3 to 154.9 in Chicago and from 1,798.8 to 28.6 in Subway. The remaining Chicago variability and the Normal-condition increases show that reduced mean error does not imply uniform stability. These comparisons support evaluating the mean error, variation across training runs and performance relative to a strong baseline separately.

7.2 Observation Location with a Matched No-Shift Control

We hold targets and both observation sets fixed between no-shift ($z = 0$) and local-shift ($z = 1$, $\alpha = 0.25$) conditions. Each placement uses 21 targets and 22 observations in Chicago, and 65 and 66 in Subway. To account for the intervention's change in demand scale, we first divide each run's MAE by `SNaive-7` MAE on the same scored targets, obtaining Q. We then form the outside penalty $P_z = Q_{\mathrm{out},z} - Q_{\mathrm{in},z}$, excess penalty $D = P_1 - P_0$, and KRIN reduction $R = D_{\mathrm{base}} - D_{\mathrm{KRIN}}$. Normalisation precedes averaging over runs.

Table 4: Matched coverage control: mean normalised excess penalty D and KRIN reduction R, in units of same-condition `SNaive-7` MAE. Means use 3 training seeds crossed with 5 placements. The 95% intervals use 20,000 crossed-factor bootstrap draws (RNG seed 0), shared across paired conditions. All 15 combinations have $R > 0$ for each comparison.

Dataset	Backbone	D_{base}	D_{KRIN}	R [95% interval]
Chicago	SPIN-s	1.0583	0.000118	1.0581 [0.5607, 1.4708]
Chicago	IGNNK-R	0.4126	−0.000096	0.4127 [0.3527, 0.4838]
Subway	SPIN-s	1.3793	−0.000022	1.3793 [1.1607, 1.5682]
Subway	IGNNK-R	0.4084	−0.000003	0.4084 [0.2437, 0.5318]

The normalised outside penalty increases under the shift for both uncentred backbones. After KRIN, its mean change is close to zero in all four comparisons, and each interval for that change includes zero (Table 48). Thus, KRIN largely preserves the no-shift relative penalty in these conditions, whereas the uncentred models incur an additional penalty. This is not evidence that the location penalty itself vanishes, nor an equivalence test.

The mean reduction is positive for every training seed after averaging its five placements (Table 49). All four crossed-bootstrap intervals exclude zero, but only three fitted models per configuration inform training variability; the intervals remain exploratory, and the 15 combinations are not independent training replicates. Subway uses newly fitted models scored on targets with valid weekly references (Appendix C.7.3).

7.3 Transition Errors and Same-Day Alarms

Immediately after a transition, +CM benefits from incorporating the current common change. In Table 40, seasonal persistence has large errors during the first 7 days, when its weekly reference remains in the pre-shift period. By day 10, its MAE falls to 9.9 in Chicago and 83.8 in Subway. KRIN also has higher error than uncentred SPIN-s on the first day. Averaging over the full evaluation window obscures this temporal structure. The uniform multiplicative shift used here closely matches +CM's design; under the actual Chicago shock, SPIN-s + KRIN instead achieves lower MAE than +CM (70.0 versus 86.7; Appendix C.2.4).

The interval averages quantify the cost of this initial mismatch. Over the first 7 days, +CM has MAE 12.2 in Chicago and 70.4 in Subway, compared with 27.1 and 234.5 for SPIN-s + KRIN. Over the first 14 days, KRIN has lower average error than uncentred SPIN-s in both datasets: 20.8 versus 24.2 and 168.6 versus 191.6 (Table 41). This benefit is not present on every day: KRIN is worse on day 1, better on day 3 and slightly worse again on day 7. Recovery should therefore be assessed over explicit time intervals.

Detecting a level shift does not necessarily identify model failures after centring. Over the 366 days of the actual shock, alarm precision for same-day failures relative to a baseline ranges from 0.81–0.94 for uncentred SPIN-s, but only 0.13–0.35 with KRIN. Alarms occur on 83.1%–85.0% of evaluation days, and failures of the uncentred model are themselves common. High precision alone therefore does not establish prospective predictive value. The analysis distinguishes identifying a shift from identifying a model failure (Appendix A.2.10).

7.4 Simplified Variants and Improvement Magnitudes

Figure 2 contrasts Normal and Shock errors at $r = 0.5$ and shows how centring changes SPIN-s relative to seasonal persistence.

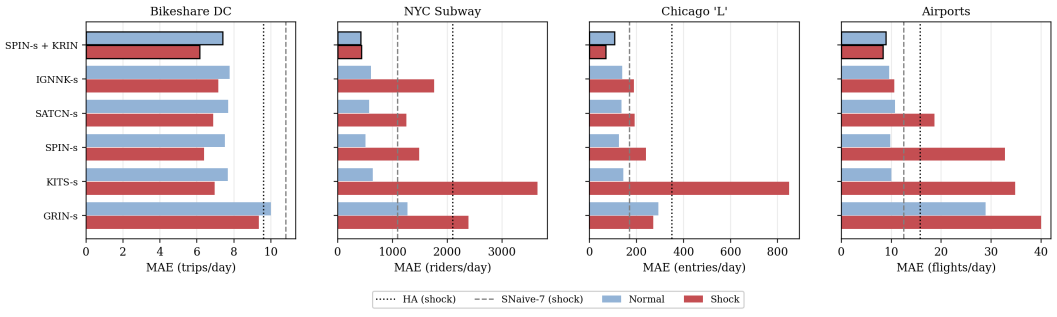


Figure 2: MAE of five simplified variants under a common protocol and SPIN-s + KRIN at $r = 0.5$, in Normal and 2020 Shock conditions (Airports: `shock2020`). The dotted and dashed lines show Shock MAE for HA and `SNaive-7`, respectively. Only Shock bars are compared with these reference lines; all rows except SPIN-s + KRIN are uncentred.

At the reported precision, the 144 simplified-model configurations contain 141 improvements, 3 ties and no increases in MAE. All three displayed ties are Normal configurations of SATCN-s on

Bikeshare. Unrounded values are unavailable in the accompanying materials, so these displayed ties are not classified as exact equality. The number of configurations with higher error than `SNaive-7` falls from 59 to 7; the remaining cases all involve GRIN-s (four in Subway and three in Airports). Full errors and rankings appear in Appendices A.1 and A.2.4.

For Shock at $r = 0.5$, averaging paired per-run relative changes across the six backbones gives -10.0% in Bikeshare, -60.1% in Chicago, -55.7% in Airports and -61.9% in Subway. These changes compare each backbone before and after centring. Separately, selecting a centred backbone by validation MAE yields margins of -8.5% to -25.7% against +CM, with Airports evaluated on `shock2020` (Appendix A.2.8). Together with the released implementations' remaining failures against +CM, these results distinguish broad error reduction from additional predictive value beyond a strong baseline.

Masking the most recent d slots of the learned models' 14-day target-node histories reveals a distinct failure when no observations remain. The masked slots are filled with the retained-slot mean; when $d = 14$, all slots instead become zero in the standardised space, corresponding to the training-period mean level. KRIN MAE rises from 126.1 to 1,182.8 in Chicago and from 621.1 to 6,580.9 in Subway between $d = 13$ and $d = 14$. This final jump reflects loss of all within-window observations, not the use of an older complete window. Both learned configurations exceed the fixed +CM comparison value at every tested $d \geq 3$ (Appendix C.7.4).

The baseline inputs, including the target's weekly reference, remain fixed in this sensitivity analysis. Their constant errors do not imply independence from target history. A comparison retaining equal numbers of recent versus old observations has not been run (Appendix C.7.4).

7.5 Limitations

Our task masks only target-day values and provides historical observations. It does not reproduce kriging at never-observed nodes or prolonged reporting outages. The location control covers SPIN-s and IGNNK-R in two cities at one shift magnitude and one cluster construction, with only three training seeds; transition analyses use SPIN-s. The real-world large-shift cases come from three mobility networks affected by the same pandemic event. NO_2 and METR-LA provide low-shift comparisons. The diagnostic characterises shifts but has not been validated as a predictor of improvement magnitude or as an automatic model-switching rule.

The released implementations are adapted to a common reconstruction task and training procedure; in particular, the original KITS incremental training strategy is not fully reproduced. SPIN-R's increases in MAE under Normal conditions and substantial variation across training seeds show that outcomes depend on implementation and evaluation conditions (Tables 55 and 57). The history-masking sweep jointly changes recency and the number of retained slots, so it does not isolate their individual effects. The auxiliary variance-normalisation failure involves the GRU–graph-convolution `KRINStd` configuration. Its comparison with the SPIN-s-based `spin_cs` configuration does not isolate an architectural cause of the difference in stability (Appendix B.2.9).

8 Conclusion

Centring reduces error across multiple backbones, but its additional value over a strong training-free baseline depends on the evaluation conditions. With targets and observation sets matched across no-shift and local-shift conditions, both SPIN-s and IGNNK-R show an increased baseline-normalised outside-observation penalty. KRIN reduces the increase in every combination of three training seeds and five placements, leaving mean changes close to zero; this does not establish the absence of a location penalty. Immediately after the uniform shift, current common-mode correction achieves lower error than the learned configurations, whereas shift alarms identify failures after centring only imprecisely.

Deployment requires checking the target's available history and the spatial representativeness of observed nodes. Baselines must share the same information constraints; +CM also requires the target's weekly reference. These results motivate validation under the deployment conditions, rather than model switching at a fixed history-age threshold.

AI USE STATEMENT

We used generative AI assistants for English translation, drafting and editing, literature and citation checks, LaTeX preparation, feedback on experimental design and interpretation, and assistance with experiment and analysis code. The authors are responsible for the final methods, analyses, claims, citations and AI-assisted content.

ETHICS STATEMENT

This study analyses publicly released aggregate mobility and environmental measurements and existing benchmarks. It does not collect individual travel trajectories or involve interventions with human participants. Reconstruction errors can nevertheless affect downstream transport planning and service allocation, particularly when poorly observed locations are underrepresented. The reported average improvements do not establish equitable performance across neighbourhoods or justify automatic operational decisions. Applications should assess errors under their own missingness patterns and demand conditions, retain human oversight, and respect the source datasets' access and redistribution terms. We report failures, incomplete evidence and the scope of the controlled interventions to support responsible interpretation.

REPRODUCIBILITY STATEMENT

Code and dataset configurations are accessible through the anonymous repository at `https://anonymous.4open.science/r/KRIN-91E4/`. Sections 3 and 6, Appendix B, and Appendix F specify the task, preprocessing, model identifiers and experiment entry points. The accompanying Table 4 package includes the final analysis code, Chicago and Subway run-level CSVs, paired contrasts, bootstrap settings and training-seed means. Running `analysis/analyse_coverage.py` with `-input-source matched` reproduces its normalised effects and crossed-factor intervals without model training. Appendix F.2 maps the inputs and outputs and distinguishes this statistical replay from the recorded data-level audit. The latter requires processed observations and saved predictions, which are not included; raw datasets, checkpoints and the complete set of other experiment outputs are also outside the scoped release. Tables 49 and 50 report five-placement means for each training seed. Appendix C.7.4 specifies the separate history-masking intervention.

REFERENCES

Gabriel Appleby, Linfeng Liu, and Li-Ping Liu. Kriging convolutional networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

Gaspard Berthelier, Tahar Nabil, Etienne Le Naour, Richard Niamke, Samir Perlaza, and Giovanni Neglia. On the role of reversible instance normalization. *arXiv preprint arXiv:2603.11869*, 2026.

Capital Bikeshare. Capital bikeshare system data. `https://capitalbikeshare.com/system-data`, 2026. Accessed 2026.

Chicago Transit Authority. CTA ridership: 'L' station entries, daily totals. `https://data.cityofchicago.org`, 2026. Chicago Data Portal. Accessed 2026.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

Andrea Cini, Ivan Marisca, and Cesare Alippi. Filling the gaps: Multivariate time series imputation by graph neural networks. In *International Conference on Learning Representations (ICLR)*, 2022.

Jacob Cohen. *Statistical Power Analysis for the Behavioral Sciences*. Lawrence Erlbaum Associates, 2nd edition, 1988.

Noel A. C. Cressie. *Statistics for Spatial Data*. Wiley, 1993.

EUROCONTROL. Aviation data for research. https://www.eurocontrol.int/dashboard/rnd-data-archive, 2026. Accessed 2026.

Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. Are we really making much progress? a worrying analysis of recent neural recommendation approaches. In *Proceedings of the 13th ACM Conference on Recommender Systems (RecSys)*, 2019.

Lu Han, Han-Jia Ye, and De-Chuan Zhan. SIN: Selective and interpretable normalization for long-term time series forecasting. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235, pp. 17437–17453, 2024.

Rob J. Hyndman and George Athanasopoulos. *Forecasting: Principles and Practice*. OTexts, 3rd edition, 2021.

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.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*, 2017.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International Conference on Learning Representations (ICLR)*, 2018.

Roderick J. A. Little and Donald B. Rubin. *Statistical Analysis with Missing Data*. Wiley, 3rd edition, 2019.

Ivan Marisca, Andrea Cini, and Cesare Alippi. Learning to reconstruct missing data from spatiotemporal graphs with sparse observations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Metropolitan Transportation Authority. MTA subway turnstile and hourly ridership data. https://data.ny.gov, 2026. New York State Open Data. Accessed 2026.

Art B. Owen. The pigeonhole bootstrap. *The Annals of Applied Statistics*, 1(2):386–411, 2007. doi: 10.1214/07-AOAS122.

Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. In *NeurIPS Relational Representation Learning Workshop*, 2018.

US Environmental Protection Agency. AQS daily summary data, parameter 42602 (nitrogen dioxide). https://aqs.epa.gov/aqsweb/airdata/download_files.html, 2026. Accessed 2026.

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

Yuankai Wu, Dingyi Zhuang, Aurelie Labbe, and Lijun Sun. Inductive graph neural networks for spatiotemporal kriging. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 2021a.

Yuankai Wu, Dingyi Zhuang, Mengying Lei, Aurelie Labbe, and Lijun Sun. Spatial aggregation and temporal convolution networks for real-time kriging. *arXiv preprint arXiv:2109.12144*, 2021b. URL https://arxiv.org/abs/2109.12144v1. Version 1.

Qianxiong Xu, Cheng Long, Ziyue Li, Sijie Ruan, Rui Zhao, and Zhishuai Li. KITS: Inductive spatio-temporal kriging with increment training strategy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 12945–12953, 2025.

Chen Yang, Changhao Zhao, Haoyang Zhao, Youquan He, Chen Wang, and Jiansheng Fan. Leakage-free evaluation and distribution-robust spatio-temporal graph learning for inductive kriging. *arXiv preprint arXiv:2509.23631v2*, 2026. URL https://arxiv.org/abs/2509.23631v2. Version 2, revised 22 August 2026; first posted in 2025.

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*, 2023.

A Complete Results

A.1 Results for All Simplified-Model Configurations

Table 5: Simplified variants under a common protocol, with STGNN as a standard control: Shock MAE at $r = 0.5$. Arrows show uncentred → KRIN after retraining; × marks an error above `SNaive-7`. All datasets use 2020 (Airports: `shock2020`); cells average 3 training seeds × 10 mask seeds.

	Bikeshare	Chicago	Airports	Subway
IGNNK-s	7.1 → 6.6	188.2 × → 91.6	10.6 → 8.4	1762.5 × → 695.8
SATCN-s	6.9 → 6.5	191.9 × → 71.3	18.6 × → 9.3	1253.9 × → 488.8
SPIN-s	6.4 → 6.2	240.8 × → 70.0	32.7 × → 8.4	1490.9 × → 432.6
KITS-s	7.0 → 6.3	849.7 × → 149.5	34.8 × → 8.8	3650.2 × → 656.7
GRIN-s	9.3 → 6.8	270.5 × → 131.4	40.0 × → 12.0	2390.3 × → 1120.7 ×
STGNN	6.9 → 6.5	188.5 × → 95.7	18.1 × → 8.7	1313.7 × → 551.2
`SNaive-7`	10.8	169.7	12.6	1095.6

Tables 6–9 report all configurations of the simplified-model evaluation. A model fails relative to the baseline when its MAE exceeds the corresponding `SNaive-7` MAE.

A.1.1 Capital Bikeshare, Washington DC

Table 6: Bikeshare MAE for simplified variants under a common protocol and STGNN. N and S denote Normal and Shock (2020); r is the hidden-node ratio. Each arrow reports uncentred → KRIN, averaged over 3 training seeds × 10 mask seeds. The last row is the training-free reference; rounding can hide small changes.

Backbone	N r=.25	N r=.5	N r=.75	S r=.25	S r=.5	S r=.75
IGNNK-s	7.5 → 7.4	7.8 → 7.6	8.0 → 7.9	6.9 → 6.3	7.1 → 6.6	7.7 → 7.0
SATCN-s	7.5 → 7.5	7.7 → 7.7	8.1 → 8.1	6.7 → 6.3	6.9 → 6.5	7.3 → 6.9
SPIN-s	7.4 → 7.3	7.5 → 7.4	7.5 → 7.4	6.2 → 6.0	6.4 → 6.2	6.3 → 6.1
KITS-s	7.5 → 7.3	7.7 → 7.4	7.6 → 7.5	6.8 → 6.2	7.0 → 6.3	7.1 → 6.5
GRIN-s	9.8 → 7.8	10.0 → 8.0	10.1 → 8.2	9.1 → 6.7	9.3 → 6.8	9.7 → 7.1
STGNN	7.5 → 7.4	7.7 → 7.6	7.9 → 7.8	6.7 → 6.3	6.9 → 6.5	7.3 → 6.8
`SNaive-7`	12.2	12.4	12.2	10.6	10.8	10.7

A.1.2 Chicago 'L'

Table 7: Chicago MAE for simplified variants under a common protocol and STGNN. N and S denote Normal and Shock (2020); r is the hidden-node ratio. Each arrow reports uncentred → KRIN, averaged over 3 training seeds × 10 mask seeds. The last row is the training-free reference.

Backbone	N r=.25	N r=.5	N r=.75	S r=.25	S r=.5	S r=.75
IGNNK-s	117.8 → 93.0	139.2 → 106.9	158.5 → 110.1	163.1 → 87.7	188.2 → 91.6	398.4 → 99.9
SATCN-s	121.1 → 95.1	136.5 → 107.7	148.3 → 114.4	141.0 → 66.3	191.9 → 71.3	192.4 → 80.6
SPIN-s	133.7 → 96.8	125.2 → 106.5	152.9 → 106.8	242.9 → 68.2	240.8 → 70.0	228.3 → 70.8
KITS-s	126.4 → 93.6	143.9 → 106.9	149.8 → 106.1	973.9 → 139.1	849.7 → 149.5	667.7 → 107.3
GRIN-s	280.7 → 136.0	293.2 → 154.9	300.4 → 160.9	259.1 → 111.3	270.5 → 131.4	364.1 → 123.6
STGNN	132.9 → 99.0	158.0 → 109.4	165.0 → 113.9	163.0 → 81.2	188.5 → 95.7	291.1 → 98.0
`SNaive-7`	172.0	189.0	188.1	168.0	169.7	172.9

A.1.3 AIRPORTS

Table 8: Airports MAE for simplified variants under a common protocol and STGNN. N denotes Normal and S denotes `shock2020`, not the 2020–21 pool used for released-code comparisons. Each arrow reports uncentred → KRIN at the indicated r, averaged over 3 training seeds × 10 mask seeds.

Backbone	N r=.25	N r=.5	N r=.75	S r=.25	S r=.5	S r=.75
IGNNK-s	9.8 → 9.5	9.6 → 9.5	10.4 → 10.0	9.9 → 8.3	10.6 → 8.4	13.4 → 9.3
SATCN-s	10.6 → 10.0	10.8 → 10.1	12.1 → 11.3	14.8 → 9.2	18.6 → 9.3	15.7 → 10.4
SPIN-s	9.3 → 8.8	9.8 → 8.9	10.0 → 9.5	28.1 → 8.2	32.7 → 8.4	32.0 → 8.9
KITS-s	10.4 → 9.4	10.0 → 9.3	10.7 → 9.8	31.8 → 8.8	34.8 → 8.8	41.1 → 9.5
GRIN-s	28.2 → 15.8	28.9 → 16.0	29.8 → 17.9	36.4 → 11.6	40.0 → 12.0	50.6 → 13.0
STGNN	10.3 → 9.5	10.3 → 9.5	11.0 → 10.1	16.7 → 8.5	18.1 → 8.7	27.1 → 9.6
`SNaive-7`	11.7	11.5	12.0	12.5	12.6	13.2

A.1.4 NEW YORK CITY SUBWAY

Table 9: NYC Subway MAE for simplified variants under a common protocol and STGNN. N and S denote Normal and Shock (2020); r is the hidden-node ratio. Each arrow reports uncentred → KRIN, averaged over 3 training seeds × 10 mask seeds. The last row is the training-free reference.

Backbone	N r=.25	N r=.5	N r=.75	S r=.25	S r=.5	S r=.75
IGNNK-s	554.8 → 432.2	608.1 → 470.2	688.1 → 501.0	1,232.6 → 610.2	1,762.5 → 695.8	3,068.1 → 829.3
SATCN-s	547.9 → 413.3	575.7 → 442.4	624.8 → 474.7	1,140.2 → 450.7	1,253.9 → 488.8	1,254.8 → 548.8
SPIN-s	500.9 → 401.0	506.8 → 424.0	535.5 → 424.6	1,176.9 → 405.4	1,490.9 → 432.6	1,202.3 → 430.5
KITS-s	625.8 → 436.4	641.9 → 464.2	636.4 → 472.0	2,764.6 → 601.3	3,650.2 → 656.7	5,647.3 → 750.2
GRIN-s	1,371.8 → 788.7	1,273.6 → 834.7	1,303.0 → 860.9	1,985.2 → 1,091.2	2,390.3 → 1,120.7	3,333.9 → 1,137.3
STGNN	538.3 → 432.3	608.7 → 471.2	695.3 → 509.0	1,038.6 → 510.2	1,313.7 → 551.2	2,662.2 → 616.7
`SNaive-7`	794.5	839.5	823.6	1,028.8	1,095.6	1,068.1

A.2 SUPPORTING COMPARISONS

A.2.1 TRAINING-FREE BASELINES

Appendix A.1 reports individual backbones, and Table 10 gives the training-free baselines under the same evaluation conditions.

Table 10: Training-free baseline MAE on the 2020 Shock windows at $r = 0.5$, including `shock2020` for Airports. `Naive-1` and `SNaive-7` copy the target node at lags 1 and 7; HA uses its history mean and KNN-geo uses observed geographic neighbours.

	Bikeshare DC	Chicago 'L'	Airports	NYC Subway
`SNaive-7`	10.8	169.7	12.6	1095.6
`Naive-1`	10.8	264.3	14.9	1557.7
HA	9.6	350.2	15.8	2104.5
KNN-geo	13.7	557.4	83.0	4415.9

A.2.2 THREE REFERENCE COMPARISONS ACROSS HIDDEN-NODE RATIOS

Table 11 separates the effect of centring from the performance differences that remain between centred backbones.

Table 11: SPIN-s + KRIN margins in the simplified-model evaluation on the 2020 Shock windows. A compares against the best uncentred backbone, B against uncentred SPIN-s, and C against the best other centred backbone. Entries are paired relative MAE differences (%) with 95% mask-cluster bootstrap intervals; negative values favour SPIN-s + KRIN.

r	Dataset	A: vs best uncentred	B: KRIN on the same backbone	C: vs best centred
0.25	Bikeshare	−3.1 [−4.0, −2.3]	−3.1 [−4.0, −2.3]	−2.2 [−2.6, −1.8]
0.25	Subway	−59.5 [−60.3, −58.6]	−65.0 [−66.9, −62.9]	−10.0 [−11.3, −8.9]
0.25	Airports	−16.8 [−18.6, −15.2]	−69.0 [−72.7, −65.2]	−1.0 [−1.9, +0.1]
0.25	Chicago	−50.1 [−53.0, −46.9]	−71.5 [−72.5, −70.5]	+2.9 [+0.8, +5.3]
0.5	Bikeshare	−3.5 [−4.2, −2.7]	−3.5 [−4.2, −2.8]	−2.5 [−2.7, −2.2]
0.5	Subway	−63.1 [−63.5, −62.7]	−70.0 [−70.6, −69.3]	−11.4 [−12.6, −10.4]
0.5	Airports	−20.6 [−22.3, −18.9]	−73.8 [−75.6, −72.0]	−0.1 [−0.6, +0.5]
0.5	Chicago	−60.7 [−62.3, −59.3]	−70.8 [−71.5, −69.9]	−1.9 [−4.1, +0.3]
0.75	Bikeshare	−2.8 [−3.2, −2.4]	−2.8 [−3.1, −2.4]	−5.2 [−5.5, −4.9]
0.75	Subway	−63.9 [−64.5, −63.3]	−63.9 [−64.5, −63.3]	−21.2 [−23.3, −19.2]
0.75	Airports	−33.5 [−34.5, −32.7]	−72.1 [−72.5, −71.7]	−4.8 [−5.3, −4.4]
0.75	Chicago	−61.9 [−63.1, −60.7]	−67.8 [−68.4, −67.2]	−11.7 [−15.0, −8.3]

A.2.3 Scale-Normalised Comparisons

Raw-scale `MAE` is sensitive to high-volume nodes. With `NMAE(macro)`, which normalises errors by each node's own level and then weights nodes equally, the margins remain negative in all eight configurations. Shock margins are -1.4% in Bikeshare, -57.7% in Subway, -9.6% in Airports and -54.0% in Chicago. In Airports, the margin decreases in magnitude from -20.6% under MAE to -9.6% under NMAE(macro).

A breakdown by node size helps explain the discrepancy. Improvements are concentrated at large airports, while both methods have relative errors of about 27% at airports with only around a dozen flights per day. Airports also has the greatest concentration of volume among the four datasets: the top 25% of nodes account for 71% of total volume. It therefore shows the largest difference between the two metrics.

On Bikeshare, SPIN-s + KRIN has margins of -3.5% for MAE and -1.4% for NMAE(macro). The paired confidence intervals exclude zero. The standardised effect size, $d = 0.73$, is smaller than the $d = 3.1$–3.6 values on the other three datasets. Averaging the paired per-run relative change for the same backbone before and after centring across all six backbones gives Shock improvements at $r = 0.5$ of -10.0% in Bikeshare, -61.9% in Subway, -55.7% in Airports and -60.1% in Chicago. These are distinct from the margins against the best uncentred backbone in Figure 3.

Bikeshare's 2020 level shift remains within the range of variation already observed during training: 0.0% of evaluation days fall below the seen minimum. None of its uncentred backbones has higher error than `SNaive-7` (0/6).

Table 12: SPIN-s + KRIN relative NMAE(macro) margin (%) against the best uncentred backbone at $r = 0.5$. Shock is evaluated in 2020, including Airports. Node-normalised errors receive equal weight; negative values favour the centred model.

Condition	Bikeshare	Chicago	Airports	Subway
Shock	−1.4	−54.0	−9.6	−57.7
Normal	−0.3	−17.2	−1.7	−14.2

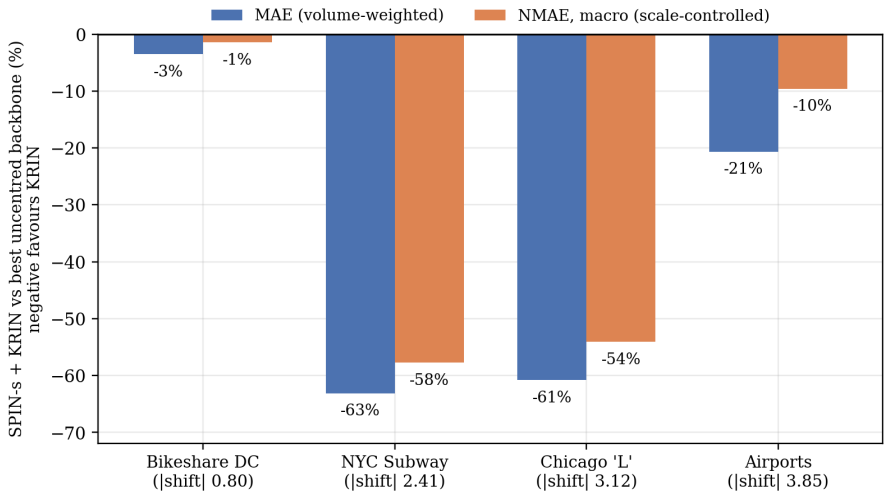


Figure 3: SPIN-s + KRIN relative error against the best uncentred backbone under MAE and NMAE(macro), at $r = 0.5$. Bars average paired per-run relative differences on the 2020 Shock windows; negative values favour KRIN. The Subway MAE margin is -63.1%. Diagnostics use the same evaluation windows, including Airports $|\text{shift}| = 3.845$. Datasets are ordered by increasing diagnostic value: Bikeshare, Subway, Chicago and Airports.

A.2.4 Ranking Before and After Centring

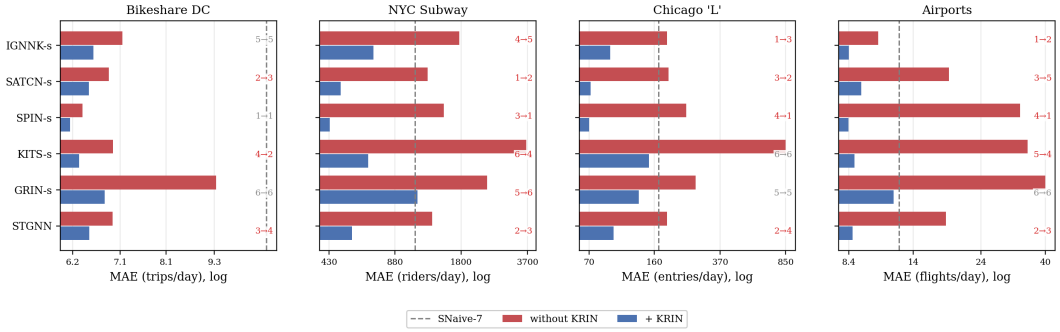


Figure 4: Backbone errors and ranks before and after KRIN in the simplified-model evaluation, using 2020 Shock MAE at $r = 0.5$. STGNN is the standard control. Each backbone is retrained with centring; rank 1 denotes the lowest error within each group. Side annotations give ranks before $\rightarrow$ after centring, with changes shown in red. Table 13 summarises the rank changes.

Figure 4 shows the changes in backbone errors and ranks. Because rank correlation can remain high despite several small rank changes, Table 13 also counts the backbones whose ranks change.

Table 13: Rank changes among the six backbones in the simplified-model evaluation after centring, using 2020 Shock MAE at $r = 0.5$. Rank 1 has the lowest error. The last column is the Spearman correlation between the uncentred and centred rankings.

	SPIN-s rank (before $\rightarrow$ after)	Backbones with a rank change	Rank corr. before vs after centring
Bikeshare	$1 \rightarrow 1$	3 / 6	+0.83
Chicago	$4 \rightarrow 1$	4 / 6	+0.49
Airports	$4 \rightarrow 1$	5 / 6	+0.54
NYC Subway	$3 \rightarrow 1$	6 / 6	+0.66

A.2.5 DIAGNOSTIC COMPONENTS

Dividing the diagnostic by the training-period amplitude was rejected by the prespecified test in Appendix C.2. Table 14 therefore includes amplitude only as supplementary descriptive information.

Table 14: Components of the 2020 Shock diagnostic. Signed shift and amplitude are measured in the transformed, training-standardised space; amplitude is the training-period system-level range. Monthly percentages compare each 2020 month with its pre-shock counterpart, then report the median and minimum across months.

Dataset	shift	|shift|	amplitude	2020 level (median month)	2020 trough (worst month)
Bikeshare DC	−0.797	0.797	4.421	57.6%	Apr 20.2%
NYC Subway	−2.406	2.406	2.713	29.9%	Apr 9.8%
Airports	−3.845	3.845	5.032	39.9%	Apr 9.8%
Chicago 'L'	−3.124	3.124	5.388	22.3%	May 11.5%

A.2.6 COMMON-MODE CORRECTION AND A GRAPH-FREE CONTROL

Table 15: Common-mode correction and removal of spatial connections: 2020 Shock MAE at $r = 0.5$ (Airports: `shock2020`). +CM adds the observed nodes' current common change to `SNaive-7`; the GRU control retains KRIN but has no graph. Learned-model cells average 3 training seeds × 10 mask seeds.

	Bikeshare	Chicago	Airports	Subway
`SNaive-7`	10.8	169.7	12.6	1,095.6
`SNaive-7` + common mode	8.3	86.7	9.6	543.4
GRU + KRIN, no graph	8.7	152.2	10.3	1,086.1
SPIN-s + KRIN	6.2	70.0	8.4	432.6

A.2.7 CENTRED BACKBONES VERSUS THE STRENGTHENED BASELINE

Table 16: Centred simplified variants and STGNN against `SNaive-7`+CM. All backbone columns include KRIN. Entries are 2020 Shock MAE at $r = 0.5$; the Beat column counts centred configurations with lower MAE than +CM. Airports uses `shock2020`.

Dataset	`SNaive-7`+CM	IGNNK-s	SATCN-s	GRIN-s	SPIN-s	KITS-s	STGNN	Beat
Bikeshare	8.3	6.6	6.5	6.8	6.2	6.3	6.5	6 / 6
Chicago	86.7	91.6	71.3	131.4	70.0	149.5	95.7	2 / 6
Airports	9.6	8.4	9.3	12.0	8.4	8.8	8.7	5 / 6
Subway	543.4	695.8	488.8	1,120.7	432.6	656.7	551.2	2 / 6

Centring substantially reduces failures relative to persistence in these conditions, but does not by itself suffice to outperform the strengthened baseline. Only two of the six centred backbones outperform that baseline in each of Chicago and Subway. The benefit of centring must therefore be distinguished from the predictive performance of the resulting backbone.

A.2.8 BACKBONE SELECTION USING VALIDATION DATA

Choosing a backbone using evaluation results can overestimate its advantage. We instead select it using only the validation MAE already used for early stopping, then compare the selected model against the strengthened baseline (Table 17).

Table 17: Validation-selected KRIN configurations in the simplified-model evaluation, evaluated on 2020 Shock at $r = 0.5$ (Airports: `shock2020`). Selection uses validation MAE only. Margin is the relative test-MAE difference from `SNaive-7`+CM; the last column fixes SPIN-s on every dataset. Percentages retain the experiment summary's precision and need not equal ratios of rounded MAEs.

Dataset	Chosen on validation	Test MAE	`SNaive-7`+CM	Margin	Margin if SPIN-s is fixed
Bikeshare	SPIN-s	6.2	8.3	−25.7%	−25.7%
Chicago	SPIN-s	70.0	86.7	−19.3%	−19.3%
Airports	KITS-s	8.8	9.6	−8.5%	−12.7%
Subway	SPIN-s	432.6	543.4	−20.4%	−20.4%

Validation selects SPIN-s on three datasets and KITS-s on Airports. SPIN-s also has the best mean validation rank across datasets, at 1.50, followed by KITS-s (3.00), SATCN-s (3.25), STGNN (3.50), IGNNK-s (3.75) and GRIN-s (6.00). Thus, choosing one backbone for all four datasets using validation alone also selects SPIN-s.

The validation-selected configuration outperforms the strengthened baseline on all four datasets, so the observed advantage does not require selecting a backbone using evaluation errors. In Airports, however, validation selects KITS-s, whose margin is −8.5%, compared with −12.7% when SPIN-s is fixed in advance. The −8.5% to −25.7% range reported in the main text comes from this table. Fixing SPIN-s across all four datasets instead gives −12.7% to −25.7%.

A.2.9 Alarm Rates of the Rolling Monitor

An operational version of the diagnostic uses a rolling window. It compares the observed nodes' levels over the most recent 28 days with the training-period mean. The threshold is the maximum deviation observed during validation and is fixed without using the evaluation period. Under Shock, alarms occur on 74.8%–85.0% of evaluation days across the three large-shift datasets. Chicago and Subway have no alarms under Normal conditions (0%). Bikeshare also has a Normal alarm rate of 0%, but a Shock alarm rate of 29.2%, consistent with a level change that remains within the previously observed range. Airports retains a Normal alarm rate of 21.7%. Calling Normal alarms false alarms refers only to the calendar-defined condition. The next section instead defines positives by whether a model has higher error than a baseline on that day.

A.2.10 Alarms and Same-Day Baseline Failures

A level shift and a model's failure relative to a baseline are different events. Table 18 evaluates how the same alarms correspond to failures before and after centring.

Table 18: Rolling alarms versus same-day failures relative to the baselines over 366 Shock days per dataset: $r = 0.5$, 2 training seeds × 3 mask seeds. The 28-day monitor retains its validation threshold. A failure means that model MAE exceeds the listed reference on that day; precision is $\Pr(R_t = 1 \mid A_t = 1)$ and recall is $\Pr(A_t = 1 \mid R_t = 1)$.

Dataset	Model	Reference	Reversal days	Precision	Recall
Chicago	SPIN-s	`SNaive-7`	303 / 366	0.88	0.90
Chicago	SPIN-s	+CM	326 / 366	0.94	0.89
Chicago	SPIN-s + KRIN	`SNaive-7`	84 / 366	0.17	0.62
Chicago	SPIN-s + KRIN	+CM	148 / 366	0.35	0.74
Subway	SPIN-s	`SNaive-7`	260 / 366	0.81	0.95
Subway	SPIN-s	+CM	280 / 366	0.87	0.94
Subway	SPIN-s + KRIN	`SNaive-7`	53 / 366	0.13	0.77
Subway	SPIN-s + KRIN	+CM	123 / 366	0.35	0.87

Alarm rates are 85.0% in Chicago and 83.1% in Subway. For uncentred SPIN-s, precision ranges from 0.81–0.94 and recall from 0.89–0.95; with KRIN, precision falls to 0.13–0.35, while recall ranges from 0.62–0.87. The centred model has higher error than `SNaive-7` on only 84 days in Chicago and 53 in Subway. Against +CM, the corresponding counts are 148 and 123 days. The 53–84-day range therefore applies only to the seasonal baseline.

The diagnostic can monitor level changes and, in this evaluation, its alarms often coincide with failures of the uncentred model. Its precision for identifying failures after centring is low. Moreover,

the uncentred model's failure rates are already high: 82.8%/89.1% in Chicago and 71.0%/76.5% in Subway against SNaive-7/+CM. Precision and recall should therefore be interpreted alongside the alarm rate and the prevalence of failures. This analysis compares information and errors from the same day; it does not measure prospective failure prediction, detection delay or the net benefit of alarm-based model switching.

A.3 Relative Error Changes from Normal to Shock

Figure 5 and Table 19 compare raw-scale errors between Normal and Shock.

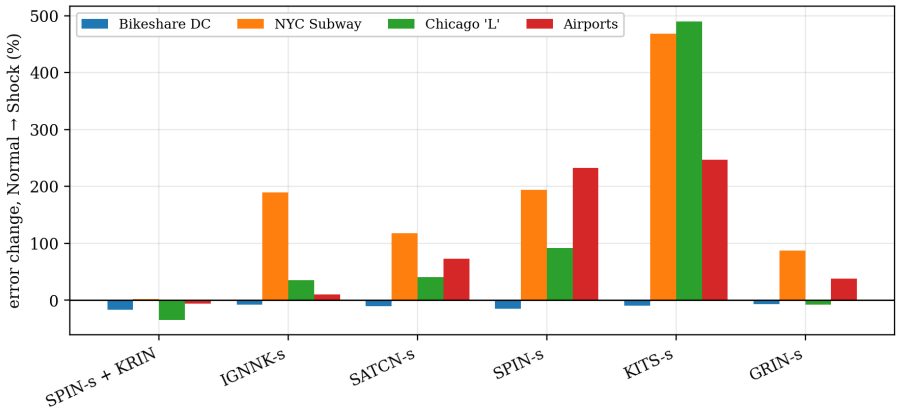


Figure 5: Normal-to-Shock relative MAE change at $r = 0.5$ for the simplified variants, with SPIN-s + KRIN shown separately. Shock covers 2020, including Airports. Zero denotes unchanged MAE; positive values denote larger Shock error. Table 19 gives the numerical values.

Table 19: Relative change in MAE from Normal to 2020 Shock, $100(\mathrm{MAE}_S/\mathrm{MAE}_N - 1)$, at $r = 0.5$. All models are from the simplified-model evaluation; only the first row includes KRIN. Airports Shock is `shock2020`. Negative values indicate lower absolute error, not necessarily lower scale-normalised error.

Model	Bikeshare	Chicago	Airports	Subway
SPIN-s + KRIN	−17	−34	−6	+2
IGNNK-s	−8	+35	+10	+190
SATCN-s	−11	+41	+73	+118
SPIN-s	−15	+92	+233	+194
KITS-s	−9	+490	+247	+469
GRIN-s	−7	−8	+38	+88

`SPIN-s + KRIN` shows no large Shock-related error increase on any of the four datasets. Negative changes in raw-scale MAE should not, however, be interpreted as improved robustness: absolute errors can decrease when demand falls. Scale-sensitive comparisons are discussed in §7.4.

B Protocol and Data Details

B.1 Split Sizes

Table 20: Numbers of usable target days in the training, validation and evaluation splits. Airports has a 731-day Shock pool; its 366-day `shock2020` subset is used in the main simplified-model tables. The released-implementation results and direct comparisons with simplified variants use the full pool.

Dataset	Condition	Train	Val	Eval
Bikeshare / Chicago / Subway	Normal	1,008	146	293
	Shock	606	110	366 (2020)
Airports	Normal	2,648	380	762
	Shock	1,227	220	731 (2020: 366, 2021: 365)

The Airports Shock evaluation pool contains 731 days. The main simplified-model results use `shock2020` (366 days), whereas the released implementations use `all` (731 days). The training and validation splits remain fixed, and direct comparisons use aggregates from the same evaluation window.

B.1.1 Airports Evaluation Windows

Differences in Airports MAE must be interpreted with their evaluation windows. Table 21 compares baselines after changing only the evaluation slice, with the training and validation split held fixed.

Table 21: Airports training-free baseline MAE by Shock evaluation window at $r = 0.5$. The training and validation split is fixed; only the evaluation slice changes. The two-decimal values distinguish `shock2020`, `shock2021` and their `all` pool.

Airports slice	Eval days	`SNaive-7`	`SNaive-7`+CM
`all` (2020–21)	731	11.35	9.46
`shock2020`	366	12.55	9.60
`shock2021`	365	10.20	9.32

The main simplified-model results and validation-selection table retain `shock2020`. The released implementations and direct comparisons between the two evaluations use `all`. Accordingly, the best simplified-model MAE of 8.2 in Table 56 is aggregated over 2020–21 and should be distinguished from the 2020-only main and validation-selection results. Diagnostics use the same window as their associated errors: `shock2020` gives $|\text{shift}| = 3.845$ and a below-minimum fraction of 37.7%. The values 3.102 and 31.9% describe the 2020–21 `all` window and are not paired with the main text's 2020 MAEs.

B.2 Protocol and Implementation Details

Figure 6 summarises the geographic graphs constructed by the common procedure in §6.

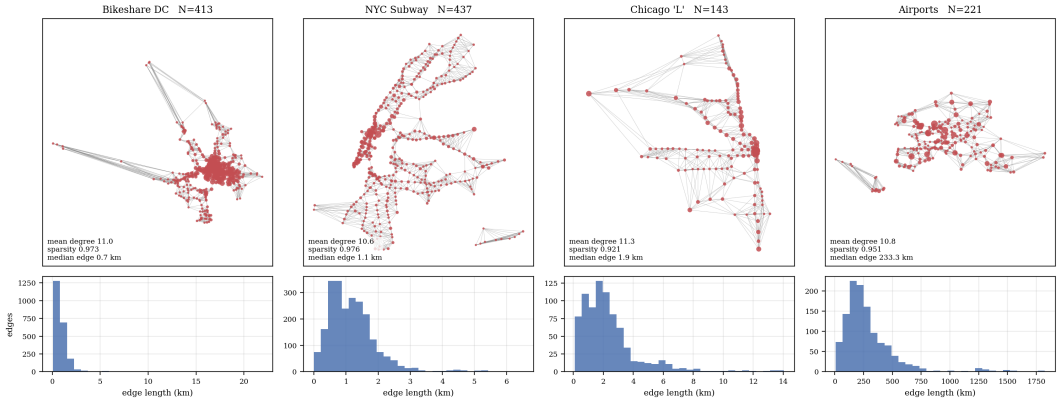


Figure 6: Geographic node layouts and A^{geo} edges; marker area scales with mean node activity. Map annotations report scalar summaries of mean degree, sparsity and median edge length; only edge lengths are shown as histograms, in kilometres. Graphs share the distance-kernel and symmetrised top-10 construction.

B.2.1 Parameter Counts

Table 22: Trainable parameter counts of the simplified variants under a common protocol and STGNN, before and after adding KRIN. Counts include the full backbone; a shared hidden dimension does not imply an equal total parameter count.

	Parameters	+ KRIN
IGNNK-s	17,793	17,793
STGNN	29,697	29,697
GRIN-s	29,505	29,505
SATCN-s	45,889	45,889
KITS-s	51,713	51,713
SPIN-s	54,913	54,913

KRIN adds zero parameters to every backbone. Section 5.2 discusses why parameter count alone does not determine performance in this setting.

B.2.2 Choice of Level Statistic

Table 23: Choice of level statistic on the same SPIN-s backbone with the same parameter count in each configuration. Cells report Shock / Normal MAE at $r = 0.5$; Shock covers 2020. The four rows compare no centring, last-value removal, window-mean removal, and mean/variance normalisation.

Centring	Bikeshare	Chicago	Airports (2020)	Subway
None	6.4 / 7.5	240.8 / 125.2	32.7 / 9.8	1,490.9 / 506.8
Last value (NLinear style)	6.2 / 7.5	71.5 / 114.5	8.7 / 9.1	440.3 / 448.6
Window mean (KRIN)	6.2 / 7.4	70.0 / 106.5	8.4 / 8.9	432.6 / 424.0
Mean + variance (full RevIN)	6.4 / 7.7	68.3 / 121.2	9.8 / 11.4	432.3 / 448.4

The mean/variance configuration is identified in the implementation as `spin_cs`. It retains the SPIN-s backbone used in the other rows. The separate `krin_std`/`KRINStd` stability experiment uses a GRU–graph-convolution backbone and is specified in Appendix B.2.9.

B.2.3 Effect of Centring on Spatial Aggregation

Table 24 separates the improvement from global scaling from the additional effect of nodewise centring.

Table 24: Weighted-neighbour mean MAE under Normal conditions with neighbours, weights and graph held fixed. Raw scale, global scaling and added per-node centring separate the two preprocessing effects. Centring effect is the ratio of globally scaled MAE to MAE after additional nodewise centring.

MAE of the weighted neighbour mean	Raw scale	Global scaling	+ per-node centring	Centring effect
Airports	92.8	46.2	14.4	3.2×
Chicago	533.8	259.8	112.6	2.3×
Subway	4,388.1	2,294.3	1,161.7	2.0×
Bikeshare	14.3	10.2	7.7	1.3×

B.2.4 Leakage Control

Standardisation statistics are fitted only on the training rows of the corresponding condition. The main graph A^{geo} depends only on coordinates and contains no learned component. Validation and evaluation indices are disjoint for both conditions of all four datasets, preventing early stopping from accessing evaluation examples (Yang et al., 2026). A target node's own history is restricted to $t-K,\ldots,t-1$, and current values are used only for observed nodes. Initial evaluation windows can include observations preceding the evaluation period, but neither the hidden target-day value nor future observations are used.

B.2.5 Differences from the Released Implementations

The five simplified backbones are analysis variants adapted to a common interface. Table 25 summarises their differences from the original architectures.

Table 25: Architectural changes defining the simplified variants under a common protocol (-s). The Released architecture column describes the original method; the final column lists the components retained or replaced in this study. The evaluation of released implementations is documented separately in Appendix E.

Variant	Released architecture	Simplification in this work
GRIN-s	Bidirectional BiGRIL; two-stage autoregressive imputation over the full window	Unidirectional recurrence
SPIN-s	Sparse graph attention with joint temporal and spatial tokens	Recurrent temporal encoding followed by attention across nodes
IGNNK-s	Diffusion convolution	A stack of symmetrically normalised GCN layers
SATCN-s	PNA-style multiple aggregators and scalers	A single masked-mean aggregator
KITS-s	Virtual nodes and incremental training	Virtual-node embeddings with the common training procedure

The simplified variants support analysis of level handling and architecture under a shared hidden dimension and training procedure. Results concerning the original architectures come from the released implementations; neither evaluation reproduces each source paper's complete task and training strategy. A shared hidden dimension also does not imply an equal total parameter count, so the comparison does not fully isolate the causal effect of aggregation architecture alone. Appendix E reports the full released-implementation results and variation across seeds.

We test whether directionality or parameter budget alone explains why GRIN-s is the only backbone to fail against the baseline under Normal conditions (Table 26).

Table 26: Partial restoration of GRIN-s: Normal / 2020 Shock MAE at $r = 0.5$. The bidirectional control restores directionality; the widened unidirectional control uses hidden dimension 80 to match its parameter budget. These controls do not restore every component of released GRIN.

GRIN-s variant	Bikeshare N / S	Chicago N / S	Airports N / S	Subway N / S
Unidirectional (used in the main tables)	10.0 / 9.3	293.2 / 270.5	28.9 / 40.0	1,273.6 / 2,390.3
Bidirectional, as in the official code	9.9 / 9.3	294.0 / 288.0	29.2 / 39.7	1,233.5 / 3,179.2
Unidirectional, widened to the same budget	10.0 / 9.3	294.3 / 281.8	28.5 / 38.7	1,254.3 / 2,284.0

Normal-condition failures remain after restoring bidirectionality and after matching the parameter budget with a unidirectional model. Changing either factor alone therefore does not eliminate the observed failures. This does not imply identical Shock errors across the three configurations, nor does it identify a particular remaining implementation difference as the cause. Results obtained by restoring the released architecture are reported in Appendix E.

B.2.6 Observation Protocols and Baselines in the Source Papers

Table 27: Observation protocols and baseline lists in the five source papers. A denotes kriging without target-node history; B denotes imputation with observations available at other times. The last column distinguishes an undefined per-node persistence baseline from an applicable baseline absent from the reported comparison.

Paper	Protocol	Baselines the paper lists	Per-node persistence
IGNNK	A	kNN, OKriging, KPMF, GLTL	Not definable
SATCN	A	kNN, OKriging, GLTL, KCN, IGNNK	Not definable
KITS	A	Mean (per timestep), OKriging, KNN, KCN, IGNNK, LSJSTN, INCREASE, GLTL, MPGRU, GRIN	Not definable
GRIN	B	MEAN, KNN, MICE, MF, VAR, rGAIN, MPGRU, BRITS	Applicable, not reported
SPIN	B	MEAN, KNN, MF, MICE, VAR, rGAIN, BRITS, SAITS, Transformer, GRIN	Applicable, not reported

The classification follows the observation protocols described in the source papers. IGNNK treats approximately 25% of nodes as unsampled virtual sensors and sets their inputs to $Y_t^u = 0$. SATCN similarly sets masked sensor values to zero and defines the task as interpolating signals at unsampled locations. KITS explicitly distinguishes nodewise means in imputation, where each node has observations across time, from kriging at unobserved nodes, where such a mean cannot be computed; the latter instead uses available observations across nodes at each time step. GRIN and SPIN simulate point missingness by randomly removing 25% of available values, and block missingness by combining 5% random removal with stochastic outages lasting 12 to 48 steps. Both papers also include virtual-sensor experiments that hide an entire node in their appendices, but their main protocols use missing node–time pairs. Their temporally informed baselines, VAR, MPGRU, BRITS and rGAIN, all require training. Their training-free baselines, MEAN and KNN, do not use a node's most recent observation. Neither baseline list therefore includes training-free temporal persistence.

We searched for the terms persistence, last observed, LOCF, carried forward, previous value, naive, seasonal naive, random walk and previous week. Venue information follows the official conference bibliographies; SATCN is cited as an arXiv preprint.

B.2.7 Reporting Gaps in the Raw Data

We count internal gaps in the raw data to assess whether temporary reporting interruptions occur in practice (Table 28).

Table 28: Internal reporting gaps in the raw data. An episode is a contiguous missing run between a node's first and last observations, excluding openings and closures. Percentages use episodes as the denominator; affected-node counts use all retained nodes. n/a denotes no episodes.

Dataset	Dropout episodes	Nodes affected	Exactly 1 day	≤ 7 days
Airports	3,326	150 / 221	74%	97%
Subway	4,032	437 / 437	11%	99%
Bikeshare	0	0 / 413	n/a	n/a
Chicago	0	0 / 143	n/a	n/a

Approximately three quarters of Airports reporting gaps last exactly one day, and 99% of Subway episodes last at most one week. These are interruptions at previously observed nodes, for which observations from other nodes on the same day can support reconstruction. Bikeshare and Chicago have no internal reporting gaps, so this empirical evidence comes from two of the four datasets.

B.2.8 HA AS AN ALTERNATIVE REFERENCE

Replacing `SNaive-7` with `HA` changes the failure counts and their ordering, with Airports showing the greatest deterioration relative to HA. The two baselines differ in strength across datasets. In Chicago and Subway, `SNaive-7` has approximately half the error of `HA` (170 versus 350, and 1096 versus 2105), whereas the Airports gap is smaller (12.6 versus 15.8). Thus, `SNaive-7` sets a more demanding reference on all three datasets whose levels leave the previously observed range. HA is stronger only in Bikeshare (9.6 versus 10.8), where both baselines yield zero failures among the backbones (0/6).

B.2.9 AN EXTREME ERROR IN THE KRINSTD CONFIGURATION

Table 29: Auxiliary Airports results for `krin_std` (class `KRINStd`), using the GRU–graph-convolution backbone specified below. Median and mean MAE summarise 3 training seeds $\times$ 10 mask seeds per condition. The last column counts individual evaluations above 10^6 MAE; Shock 2020 and Shock 2021 are evaluated separately. The SPIN-s-based `spin_cs` results are reported in Table 23.

Airports auxiliary condition	Median MAE	Mean MAE	Runs above 1e6
Normal	9.63	9.57	0 / 30
Shock 2020 (deep trough)	8.89	1e15	1 / 30
Shock 2021 (level recovered)	8.54	8.55	0 / 30

Configuration and implementation. The run identifier `krin_std` selects class `KRINStd` through `src/models/__init__.py`; its implementation in `src/models/networks.py` combines a GRU temporal encoder with graph convolution and has 29,761 trainable parameters. This is the specific mean/variance-normalised configuration evaluated in Table 29; KRIN itself remains a wrapper rather than the name of this backbone. The implementation places a lower bound of 10^{-4} on each node's history-window standard deviation. Writing $s_i^{(t)} = \max(\sigma_i^{(t)}, 10^{-4})$, normalised predictions $z_i^{(t)}$ are restored to the globally standardised space as

$$\hat{\hat{x}}_t^{(i)} = s_i^{(t)} z_i^{(t)} + \mu_i^{(t)}.$$

The standard-deviation floor prevents division by zero during normalisation; it does not bound the learned output or the restored prediction.

Difference from the SPIN-s ablation. The mean/variance row in Table 23 uses `spin_cs`, which applies centring and scaling to the SPIN-s backbone. The registry implements this as `CenteredStd(SPIN)`, using $s_i^{(t)} = \sigma_i^{(t)} + 10^{-5}$, whereas `KRINStd` clamps the standard deviation from below at 10^{-4}. That backbone uses recurrent temporal encoding followed by attention across nodes and has 54,913 trainable parameters (Tables 22 and 25). Thus, `krin_std` and `spin_cs` refer to different implementations: the former uses GRU and graph convolution, whereas

the latter uses the SPIN-s attention backbone. They must be distinguished when interpreting numerical stability. Their different backbones, parameter counts and scale stabilisation rules preclude attributing the observed stability difference to a single component.

Extreme error and scope of the result. The Shock 2020 mean and median differ because one of the 30 `krin_std` evaluations, using training seed 2 and mask seed 3, has MAE 3×10^{16}. The remaining 29 evaluations range from 7.9 to 9.6; excluding that single evaluation gives a mean of 8.87.

The restoration and inverse-preprocessing chain can amplify extreme outputs. After multiplication by the window scale and restoration of the mean, predictions are mapped back to the log-transformed space and then passed through `expm1`. Large positive values can therefore produce extremely large raw-scale errors even with the standard-deviation floor. Mean-only restoration removes the multiplication by the window scale from this pathway, but does not guarantee stability of the learned output or the exponential inverse transformation.

In contrast, `spin_cs` produces finite errors in all 30 Airports 2020 evaluations, with mean MAE 9.8. Under Chicago Shock, its MAE of 68.3 is lower than the window-mean configuration's 70.0. Table 29 therefore documents a failure of the identified `KRINStd` configuration, rather than evidence that variance normalisation is uniformly unstable across backbones. The comparison does not establish which implementation difference caused the extreme error.

B.2.10 Optimisation Settings

- Hidden-node ratios $r \in \{0.25, 0.5, 0.75\}$; random node masking reproducible from the mask seed.
- 3 training seeds $\times$ 10 mask seeds, giving 30 evaluations per configuration.
- Up to 60 training epochs; batch size 32; Adam (Kingma & Ba, 2015) with learning rate 10^{-3} and weight decay 10^{-5}; early-stopping patience 12; hidden dimension 64; dropout 0.1; history length $K = 14$. Graph convolution uses 2 layers by default and 3 layers for IGNNK-s, SATCN-s and KITS-s, whose source architectures use deeper propagation. Convolutions use the standard symmetrically normalised adjacency operator (Kipf & Welling, 2017).
- Training loss: masked MSE on hidden nodes in the standardised space.
- Evaluation metrics: errors on the original scale, computed only on hidden nodes.

B.2.11 Effect Sizes and Paired Uncertainty Estimates

We report Cohen's d, the difference between two means divided by their pooled standard deviation (Cohen, 1988), as a standardised effect size. Uncertainty is estimated by bootstrapping paired differences. Relative differences are first computed for models sharing the same training seed and mask seed, accounting for variation shared by the pair before resampling. We consider two resampling schemes. Clustering by mask seed measures sensitivity to mask sampling conditional on the fitted training runs. Hierarchically resampling training seeds first, and mask seeds within them, also accounts for variation across training runs. With only three training seeds, the latter gives wider intervals here. The main text and Appendix A.2.2 use mask-seed clustering; the table below compares both schemes.

Table 30: Bootstrap uncertainty for the three SPIN-s + KRIN comparisons in Table 11, restricted to $r = 0.5$. Point estimates and 95% intervals are paired relative MAE differences (%) from 4,000 resamples. Columns resample mask clusters only, or training seeds first and mask seeds within them.

Dataset	Comparison	Point	Mask clusters	Training seeds first
Bikeshare	A: vs best uncentred	−3.5	[−4.2, −2.8]	[−4.5, −2.4]
Bikeshare	B: KRIN on the same backbone	−3.5	[−4.2, −2.8]	[−4.5, −2.3]
Bikeshare	C: vs best centred	−2.5	[−2.7, −2.2]	[−3.3, −1.7]
Subway	A	−63.1	[−63.5, −62.7]	[−74.8, −56.8]
Subway	B	−70.0	[−70.6, −69.3]	[−76.4, −65.1]
Subway	C	−11.4	[−12.5, −10.4]	[−13.9, −9.7]
Airports	A	−20.6	[−22.3, −18.9]	[−24.4, −14.9]
Airports	B	−73.8	[−75.5, −72.1]	[−76.2, −70.7]
Airports	C	−0.1	[−0.6, +0.5]	[−2.2, +3.0]
Chicago	A	−60.7	[−62.3, −59.4]	[−69.4, −48.4]
Chicago	B	−70.8	[−71.5, −69.9]	[−72.6, −67.9]
Chicago	C	−1.9	[−4.2, +0.3]	[−4.1, +0.7]

Resampling training seeds first widens the intervals without changing the qualitative conclusions. All three Bikeshare comparisons continue to exclude zero, despite having the smallest effects. Comparison C for Airports and Chicago continues to include zero. The main conclusions therefore hold under both resampling schemes.

B.3 Preprocessing Errors and Corrections

All three preprocessing issues arose when processing the New York Subway source data. One selectively affected the Shock evaluation period. We document them because the uncorrected values change the results.

1. Incorrect date indexing. Assigning dates by file order caused cumulative-counter differences to span non-adjacent days. Replacing those identifiers with actual calendar ordinals reduced the anomaly rate from 5.1% to 0.1%.
2. Twelve-hour timestamps. For 14 days starting on 2021-07-24, the source used a 12-hour clock. The parser ignored AM/PM suffixes, merging 4 a.m. and 4 p.m. and disrupting temporal ordering. Counts over those 2 weeks were inflated by approximately a factor of 3 (system daily totals: 4.2M to 13.7M).
3. Accumulation across missing intervals. Eight feed outages lasted 7 days each. Counter differences spanning an outage were assigned entirely to the day reporting resumed. A magnitude filter detected these spikes under normal traffic volumes, but missed two resumption days in 2020, when ridership had fallen to one quarter of its previous level; their totals were inflated by factors of 4.8 and 4.0. Requiring counter readings to be no more than 24 hours apart resolved the issue.

The third issue selectively affected one dataset's evaluation period. Correcting it increased the measured advantage of centring, so retaining the erroneous values would have understated that advantage.

C Additional Experiments

C.1 Ranking Changes with Incomplete Histories

The ranking changes in Figure 4 are measured with available target-node history. In Protocol A, no such observations exist, so the node-specific level μ cannot be estimated and this operation is inapplicable (§7.5). Here, we introduce partial missingness into the reconstruction history. This tests sensitivity to missing node–time entries, as in Protocol B, while retaining the main experiment's target-day masking. It does not reproduce the source papers' full Protocol B.

Figure 7 summarises the point-missingness experiment. We independently remove each (day, node) history entry with probability 0.25, giving an observed missing rate of 25.0%. A single fixed (T, N)

history mask is shared by all models and applied during both training and evaluation. Target-day masking remains unchanged to permit direct comparison with Appendix A.2.4. The remaining settings match the main experiments: `A_geo`, Shock, $r = 0.5$, and 3 training seeds $\times$ 10 mask seeds.

Missing entries are filled with the mean of that node's observed values within the window. If $k > 0$ values remain and their mean is m, filling the $K - k$ missing entries with m leaves the full-window mean equal to m. Centring after imputation therefore uses the observed-value mean. If $k = 0$, the entire window is filled with zero in the standardised space, corresponding to the node's training-period mean in the transformed space. KRIN's μ then reverts to that training mean and no longer adapts to the recent level. The same fallback is applied to all methods. Table 36 reports how frequently it occurs.

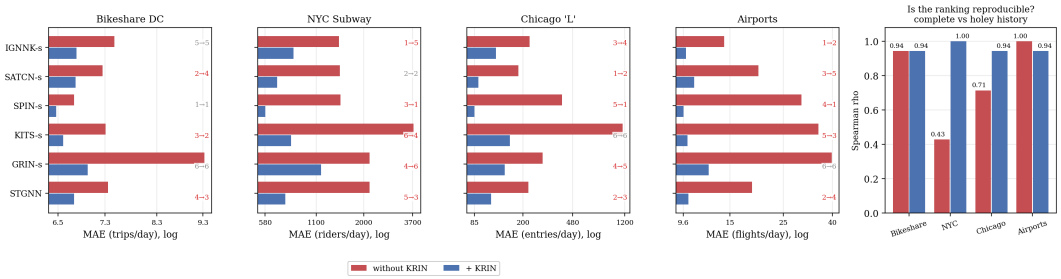


Figure 7: MAE and ranks with 25% pointwise missingness in the history for the simplified variants and the STGNN control: 2020 Shock, $r = 0.5$, 3 training seeds $\times$ 10 mask seeds. The rightmost panel reports Spearman rank correlation between complete and masked histories, separately before and after KRIN. Target-day masking is unchanged. Tables 31–34 give the values before and after KRIN.

Table 31: Bikeshare with 25% pointwise missingness in the history: 2020 Shock MAE and backbone ranks at $r = 0.5$. Simplified variants and STGNN use the same fixed history-missingness mask in training and evaluation; target-day masking is unchanged. Cells average 3 training seeds $\times$ 10 mask seeds; Change is the relative MAE change after KRIN.

Backbone	Without KRIN	Rank	+KRIN	Rank	Change
IGNNK-s	7.5	5	6.8	5	-8.9%
SATCN-s	7.3	2	6.8	4	-6.5%
GRIN-s	9.3	6	7.0	6	-25.0%
SPIN-s	6.8	1	6.5	1	-4.4%
KITS-s	7.3	3	6.6	2	-9.9%
STGNN	7.4	4	6.8	3	-8.0%

Table 32: Chicago with 25% pointwise missingness in the history: 2020 Shock MAE and ranks at $r = 0.5$. The simplified variants and STGNN share the history-missingness mask and retain the main target-day masking. Cells average 3 training seeds $\times$ 10 mask seeds; Change is the relative MAE change after KRIN.

Backbone	Without KRIN	Rank	+KRIN	Rank	Change
IGNNK-s	224.0	3	124.0	4	-44.7%
SATCN-s	183.9	1	91.1	2	-50.5%
GRIN-s	282.9	4	144.8	5	-48.8%
SPIN-s	397.2	5	84.7	1	-78.7%
KITS-s	1,154.7	6	159.0	6	-86.2%
STGNN	219.4	2	113.9	3	-48.1%

Table 33: Airports with 25% pointwise missingness in the history: `shock2020` MAE and ranks at $r = 0.5$. The simplified variants and STGNN share the history-missingness mask and retain the main target-day masking. Cells average 3 training seeds $\times$ 10 mask seeds; Change is the relative MAE change after KRIN.

Backbone	Without KRIN	Rank	+KRIN	Rank	Change
IGNNK-s	14.2	1	9.8	2	-30.8%
SATCN-s	19.7	3	10.7	5	-46.0%
GRIN-s	39.8	6	12.2	6	-69.2%
SPIN-s	29.8	4	9.6	1	-67.7%
KITS-s	35.0	5	10.0	3	-71.4%
STGNN	18.6	2	10.1	4	-45.6%

Table 34: Subway with 25% pointwise missingness in the history: 2020 Shock MAE and ranks at $r = 0.5$. The simplified variants and STGNN share the history-missingness mask and retain the main target-day masking. Cells average 3 training seeds $\times$ 10 mask seeds; Change is the relative MAE change after KRIN.

Backbone	Without KRIN	Rank	+KRIN	Rank	Change
IGNNK-s	1,466.7	1	826.7	5	-43.6%
SATCN-s	1,483.4	2	673.2	2	-54.6%
GRIN-s	2,154.1	4	1,172.2	6	-45.6%
SPIN-s	1,493.8	3	580.5	1	-61.1%
KITS-s	3,732.2	6	802.5	4	-78.5%
STGNN	2,155.9	5	747.0	3	-65.4%

KRIN improves all 24 configurations in this experiment, with no degradations. Its margins against the best uncentred backbone have the same sign as in §7.4: -4.4% in Bikeshare ($d = 0.9$), -53.9% in Chicago ($d = 3.4$), -32.3% in Airports ($d = 4.0$) and -60.4% in Subway ($d = 4.9$).

`SPIN-s + KRIN` ranks first among centred backbones on all four datasets. Its margins over the second-best centred backbone are smaller: -1.7% in Bikeshare ($d = 0.35$), -7.0% in Chicago ($d = 0.84$), -2.3% in Airports ($d = 0.55$) and -13.8% in Subway ($d = 3.67$). Only Subway retains a large standardised gap. This pattern resembles the complete-history setting.

Table 35 reports how strongly backbone rankings are preserved after changing the history-missingness pattern.

Table 35: Spearman correlation of backbone rankings between the main history protocol and 25% pointwise missingness in the history. Both use 2020 Shock MAE at $r = 0.5$, with the same target-day masking. Uncentred and KRIN configurations are ranked separately.

Rank corr. across missing structures	No centring	With KRIN
Bikeshare	+0.94	+0.94
Chicago	+0.71	+0.94
Airports	+1.00	+0.94
NYC Subway	+0.43	+1.00

Uncentred rankings can be sensitive to the missingness pattern, with the largest difference here occurring in Subway. After centring, the rank correlation rises to 1.00 in Subway and 0.94 in Chicago. In Airports, however, it falls from 1.00 to 0.94. The results therefore do not establish improved ranking stability on every dataset. This analysis was not prespecified and is reported as exploratory.

C.1.1 Contiguous Missingness

Independent point missingness usually leaves recent observations available for each node, whereas real reporting outages can persist for several days (§3). We therefore replace point missingness with contiguous blocks targeting the same 25% missing rate and a mean duration of 7 days. The realised missing rate is 22.1%, and the median block length is 5 days. The qualitative findings persist: KRIN improves all 24 configurations, with no degradations, and `SPIN-s + KRIN` ranks first on all four

datasets. SPIN-s's rank changes from 1 to 1 in Bikeshare, 5 to 1 in Chicago, 4 to 1 in Airports and 4 to 1 in Subway. The ranking changes thus persist under both isolated and contiguous missingness.

The target-day mask and history-missingness mask have distinct roles. An entry (t, i) selected for a contiguous history gap remains unavailable in every history window containing that entry. A true value hidden solely for target-day evaluation is not automatically removed from all later windows. Input tracing at Chicago node 40010 shows that the standardised day-t value, -1.7941, appears unchanged in the final history position on day $t + 1$ under both the main and point-missingness protocols. When the block mask covers that entry, it is replaced by the mean of the observed window entries, -0.9192. Each day is therefore scored as a separate one-day reconstruction; the protocol does not remove all historical observations of hidden nodes throughout evaluation.

Table 36: Availability within the $K = 14$ history window. Percentages use hidden-node/evaluation-day pairs as the denominator. An empty window contains no remaining observations; the last column includes every window with fewer than seven observations, including empty windows. Main denotes the protocol without additional history masking; Point and Block denote the added missingness patterns.

Dataset	History protocol	Empty window	Fewer than $K/2$ observations
Bikeshare	Main	0.00%	0.00%
Bikeshare	Point 25%	0.00%	1.10%
Bikeshare	Block 25%	4.39%	16.77%
Chicago	Main	0.00%	0.00%
Chicago	Point 25%	0.00%	0.88%
Chicago	Block 25%	4.23%	17.84%
Airports	Main	0.08%	0.76%
Airports	Point 25%	0.12%	2.71%
Airports	Block 25%	4.40%	17.74%
Subway	Main	0.00%	0.02%
Subway	Point 25%	0.00%	6.34%
Subway	Block 25%	4.77%	18.48%

Completely empty windows are rare in the main protocol, at no more than 0.08%, but occur in 4.23%–4.77% of cases under block missingness. The block-missingness results therefore combine windows that estimate recent levels with windows that revert to the training mean.

C.2 A Controlled Level-Shift Sweep

The diagnostic in §4.2 is based on four real datasets, with no observed diagnostic value between 0.797 and 2.406. We introduce controlled level shifts to fill this interval and test the diagnostic's form. The goal is to isolate how the diagnostic relates to baseline failure as the magnitude and direction of a uniform shift vary within the same datasets. This is not a test of robustness to arbitrary distribution shifts. We first apply the same multiplier to the history and evaluation period, then separately consider transitions whose new level is not yet represented throughout the history.

We multiply the evaluation period and the preceding K days by $\alpha \in [0.15, 5.0]$. Training always uses the unshifted data, $\alpha = 1$, and training rows are unchanged. Each fitted model is evaluated at all values of α. Failure at a given α means higher error than `SNaive-7`. The experiment covers 4 datasets $\times$ 12 model configurations $\times$ 3 training seeds; each fitted model is evaluated at 12 values of α with 3 mask seeds.

We specified three predictions in advance; one was contradicted by the results.

C.2.1 Results of the Sweep

Varying α directly changes the level-shift magnitude, but does not force a model to fail against persistence or make its failure onset align across datasets. We specifically test the diagnostic's denominator. The four datasets' amplitude values range from 2.713 to 5.388, nearly a factor of two. If normalising by amplitude is appropriate, failure onsets should align more closely in |shift|/amplitude coordinates. The experimental construction does not impose such alignment.

For each backbone–dataset pair, failure onset is the smallest |shift| at which model error exceeds `SNaive-7` error. Table 37 compares the alignment of these onsets across datasets.

Table 37: Cross-dataset alignment of baseline-failure onset in the controlled shift sweep. n counts backbone–dataset pairs with an observed onset. CV is the coefficient of variation of the first failing |shift|, with or without division by training amplitude; smaller CV means tighter alignment. Rows include all six backbones or exclude GRIN-s.

Slice	n	CV(shift/amp)	CV(|shift|)
Downward, all	17	0.645	0.592
Downward, excl. GRIN-s	13	0.458	0.385
Upward, all	21	0.482	0.444
Upward, excl. GRIN-s	17	0.311	0.252

The unnormalised |shift| aligns onsets more closely in all four analysis subsets. The direction also holds separately for all six backbones (amplitude-normalised versus unnormalised values: IGNNK-s 0.23 versus 0.12, SATCN-s 0.79 versus 0.70, GRIN-s 1.56 versus 1.53, SPIN-s 0.39 versus 0.34, KITS-s 0.39 versus 0.34, and STGNN 0.31 versus 0.10). This meets the prespecified falsification criterion for the amplitude denominator, so §4.2 uses the simpler |shift|. The numerical difference is modest (approximately 0.46 versus 0.39); the result concerns the lack of improvement from dividing by amplitude, rather than the irrelevance of amplitude itself.

Excluding GRIN-s, onsets range from $|\text{shift}| = 0.22$ to 2.06, with a median of 1.49. GRIN-s already has higher error than `SNaive-7` before an artificial shift on three of the four datasets, placing its onset near zero and increasing the coefficient of variation in analyses that include it.

C.2.2 Direction of the Shift

Table 38: Backbone–dataset pairs with higher MAE than `SNaive-7` at one or more points in the controlled multiplicative sweep, before and after KRIN. The denominator is six backbones $\times$ four datasets. The Either direction row counts the union of upward and downward failures.

Direction	Pairs worse than `SNaive-7`, without KRIN	+ KRIN
Downward (demand falls)	17 / 24	1 / 24
Upward (demand rises)	21 / 24	1 / 24
Either direction	21 / 24	1 / 24

Failures also occur under upward demand shifts and, in this sweep, affect more backbone–dataset pairs than under downward shifts. The observed failure is therefore not specific to falling demand. This is consistent with the explanation in §4.3 involving inputs outside the previously observed level range.

Across the sweep, KRIN reduces the number of backbone–dataset pairs exhibiting a baseline failure from 21 to 1; the remaining pair is GRIN-s on Airports. This extends the evidence across shift magnitudes and directions within the same four datasets, without establishing generalisation to other types of distribution shift or application domains.

C.2.3 Evaluation Beginning at Shift Onset

The preceding sweep applies the multiplier to both the evaluation period and its preceding K days, so every evaluation window already represents the new level. Immediately after a real transition, the history instead contains a mixture of old and new levels.

Moving shift onset to the first evaluation day initially leaves the entire history at the old level while the target is at the new level. Pre-shift observations are then replaced sequentially as the window advances. Table 39 summarises the full-window effect; Tables 40 and 41 show early recovery using a separate set of seeds.

Table 39: Full-window MAE for a controlled global multiplier $\alpha = 0.25$ at $r = 0.5$. Arrows change the onset from K days before evaluation to the first evaluation day. Models are fixed after Normal training; this sweep uses 3 training seeds × 3 mask seeds, distinct from the daily analysis below.

Dataset	SNaive−7	SPIN-s	SPIN-s + KRIN
Bikeshare	3.3 → 3.7	2.3 → 2.3	2.0 → 2.0
Chicago	43.6 → 58.5	44.1 → 41.3	24.8 → 25.3
Airports	2.9 → 4.1	4.3 → 4.4	2.2 → 2.3
Subway	211.3 → 328.8	180.6 → 190.7	106.7 → 111.8

Including the transition reduces the number of failures counted over the full evaluation window, from 6/6 to 2/6 in Airports and from 3/6 to 2/6 in Chicago. This does not imply that the learned models alone have improved. During the initial transition, SNaive−7 uses a pre-shift weekly reference to estimate a post-shift target, increasing its full-window MAE from 43.6 to 58.5 in Chicago and from 211.3 to 328.8 in Subway. The learned models' full-window errors change less. Because failure is defined relative to this baseline, increasing baseline error can reduce the counted failures even without a reduction in model MAE. Early accuracy should therefore be assessed using the daily errors and the MAE over the first h days in Appendix C.2.4.

C.2.4 Errors Immediately After the Transition

A full-window average can obscure failure and recovery immediately after a transition. We therefore report daily MAE and MAE over the first h evaluation days separately (Tables 40 and 41).

Table 40: Daily MAE following a global multiplier $\alpha = 0.25$ beginning on evaluation day 1. Models are fixed after Normal training; $r = 0.5$, 2 training seeds × 3 mask seeds. The first history window is entirely pre-shift and later windows mix levels. Day identifies an individual evaluation day.

Dataset	Day	SNaive−7	+CM	SPIN-s	SPIN-s + KRIN
Chicago	1	390.0	12.6	42.3	43.8
Chicago	3	696.0	11.6	21.4	19.8
Chicago	7	621.0	15.1	27.8	28.9
Chicago	10	9.9	9.7	21.5	13.6
Chicago	14	11.8	11.1	26.1	19.9
Subway	1	2,921.2	117.9	291.4	393.9
Subway	3	5,616.9	65.6	327.5	252.2
Subway	7	3,843.0	68.0	194.9	208.1
Subway	10	83.8	61.5	87.9	75.2
Subway	14	155.3	135.4	195.7	168.5

Table 41: MAE over the first $h \in \{1, 3, 7, 14\}$ days of the same runs as Table 40: global $\alpha = 0.25$, onset on day 1, $r = 0.5$, 2 training seeds × 3 mask seeds. The error is averaged over the stated interval, including days not displayed in the daily table; it is not a sum of daily MAEs.

Dataset	Days	SNaive−7	+CM	SPIN-s	SPIN-s + KRIN
Chicago	1	390.0	12.6	42.3	43.8
Chicago	1–3	590.4	12.0	35.6	39.5
Chicago	1–7	638.7	12.2	26.3	27.1
Chicago	1–14	326.6	11.1	24.2	20.8
Subway	1	2,921.2	117.9	291.4	393.9
Subway	1–3	4,620.3	81.9	336.6	302.1
Subway	1–7	5,000.1	70.4	251.0	234.5
Subway	1–14	2,551.8	76.3	191.6	168.6

Because SNaive−7 copies the value from one week earlier, its reference remains in the pre-shift regime for the first 7 days. The reference enters the post-shift period on day 8. By the reported day 10, MAE falls to 9.9 in Chicago and 83.8 in Subway. No day-8 measurement is reported here, so this table alone does not establish the exact empirical recovery time.

KRIN has higher error than uncentred SPIN-s on the first day in both datasets and lower error on day 3, but slightly higher error again on day 7. The results therefore do not imply a persistent reversal from day 3 onward, and they do not determine the unreported day-2 ordering. MAE over days 1–3 is also higher with KRIN in Chicago but lower in Subway, showing different patterns in early error.

+CM has the lowest error on every reported day and over every reported interval starting on day 1. The intervention applies the same multiplier to all nodes, however, and +CM is designed to estimate that common change from the median log difference among observed nodes. Its advantage should be interpreted in light of this match between the intervention and estimator; it does not establish near-optimality under local changes or real shocks. In the main simplified-model evaluation on Chicago's actual shock, SPIN-s + KRIN has lower MAE than +CM (70.0 versus 86.7). The transition analysis primarily reveals how the weekly reference and window mean incorporate the new level over time.

C.3 A Level-Equivariance Probe

Section 4.3 considers a level-dependent learned mapping as one failure mechanism. Here, we measure the mapping's level equivariance separately from predictive accuracy, testing whether adding a constant c to the entire input in preprocessed space shifts the prediction by exactly c.

$$\text{gap}(c) = \frac{1}{|c|}\,\mathbb{E}\big[\,\big|\, f(H + c,\, (x + c) \odot m,\, m) - f(H,\, x \odot m,\, m) - c \,\big|\,\big]$$

A value of gap $= 0$ means that no equivariance discrepancy is measured at the tested input and offset. A prediction that does not move at all gives gap $= 1$, although this value alone does not uniquely characterise the output's response. Table 42 measures the responses of fixed fitted models.

Table 42: Level-equivariance gap of fixed Normal-trained models for an additive shift $c = 0.5$ in transformed input space, averaged over 3 training seeds. The same offset is applied to history and current input. Zero denotes no measured gap at this offset; values are shown to three decimal places.

Backbone	Bikeshare	Chicago	Airports	Subway	Mean
KITS-s	0.117	0.112	0.117	0.127	0.118
GRIN-s	0.085	0.106	0.113	0.127	0.108
SATCN-s	0.104	0.093	0.097	0.115	0.102
IGNNK-s	0.107	0.106	0.062	0.124	0.099
STGNN	0.093	0.096	0.064	0.124	0.094
SPIN-s	0.081	0.082	0.102	0.080	0.086
SPIN-s + KRIN	0.000	0.000	0.000	0.000	0.000

Two observations follow.

First, all six backbones have nonzero equivariance gaps at the tested offset, with mean gaps ranging from 0.086 to 0.118. A probe at one offset does not justify linear extrapolation to the larger shifts in the actual Shock conditions. The measured gap for `SPIN-s + KRIN` is zero at the reported precision. Its exact equivariance follows from the proposition in §5.2, rather than from training; this row also serves as a numerical control for the probe.

Second, the probe does not support the explanation that more expressive aggregators encode absolute levels more strongly. SPIN-s has the smallest gap, and the rank correlations between the gap and Shock deterioration vary across datasets: $+0.31$ in Bikeshare, -0.09 in Chicago, $+0.66$ in Airports and -0.09 in Subway. We therefore report the observed ranking changes without assigning them a single established mechanism.

The proposition in §5.2 has a specific scope. Exact equivariance holds after `log1p` transformation and nodewise z-standardisation using training statistics. Adding a constant to $\log(1+x)$ corresponds to multiplying $1 + x$, not x itself, so this transformation differs from the raw-signal multiplication in Appendix C.2. The proposition does not establish equivariance to arbitrary additive or multiplicative changes on the original scale. It also requires the same time-constant offset in the history and current input; a shift that first occurs on day t and is not yet reflected in $\mu^{(t)}$ is outside its scope and is evaluated separately in Appendix C.2.3. Finally, equivariance does not guarantee accuracy: errors

in predicting the centred residual can remain. The zero measured by the probe is a numerical check of the specified transformation property, not a guarantee of accuracy under general regime changes.

This probe was not prespecified and is reported as an exploratory analysis.

C.4 Graph-Free Affine and MLP Controls

To test whether the level-dependent mapping mechanism discussed in §4.3 can arise without spatial aggregation, we evaluate simple learned controls with no communication between nodes.

The `linear` control applies an affine map to each node's own K-day history, masked current observation and mask, using 17 parameters and no information from other nodes. The `mlp` control uses the same input with a two-layer readout of hidden width 64 and 5,313 parameters. Neither uses a graph. Table 43 reports errors by condition.

Table 43: Graph-free affine and MLP controls: Normal / Shock MAE at $r = 0.5$, averaged over 3 training seeds $\times$ 10 mask seeds. Each node uses only its own history, masked current value and mask. Airports Shock uses 2020–21 (`all`); other datasets use 2020. Params denotes the number of trainable parameters.

Model	Params	Bikeshare	Chicago	Airports	Subway
`SNaive-7`	0	12.4 / 10.8	189.0 / 169.7	11.5 / 11.35	839.5 / 1,095.6
Linear	17	9.7 / 8.5	162.1 / 386.0	11.0 / 10.2	881.3 / 2,495.8
Linear + KRIN	17	9.7 / 8.4	158.7 / 170.6	11.1 / 10.1	877.0 / 1,287.4
MLP	5,313	9.9 / 8.9	173.6 / 375.4	10.1 / 13.2	822.1 / 2,355.9
MLP + KRIN	5,313	9.7 / 8.6	141.2 / 134.8	9.6 / 9.2	657.1 / 1,146.2

First, failure relative to persistence can occur without a graph. The 17-parameter affine model's MAE increases from 162.1 under Normal to 386.0 under Shock in Chicago, and from 881.3 to 2,495.8 in Subway, exceeding the Shock training-free baseline in both cases. Spatial aggregation is therefore not necessary for these failures and cannot be their sole explanation. The linear model also fails against the Normal baseline in Subway, so not all of its failures can be attributed to the level shift.

Second, the effect of centring varies by dataset and model. The linear control changes little in Bikeshare (Normal: 9.7 to 9.7; Shock: 8.5 to 8.4), whereas both controls improve substantially under Shock in Chicago and Subway. In Airports, the linear model's Shock MAE changes only from 10.2 to 10.1, while the MLP improves from 13.2 to 9.2. Large improvements therefore do not occur for every model under every level-shift condition.

Third, centring is not sufficient to outperform persistence. Both centred controls remain worse than the Subway baseline (1,287.4 and 1,146.2 versus 1,095.6), and `Linear + KRIN` remains slightly worse in Chicago. Error reduction and recovery of an advantage over persistence must be distinguished. The `GRU + KRIN` control in Appendix C.5 also fails to outperform the stronger common-mode-corrected baseline, indicating that prediction of the centred residual still matters.

C.5 Common-Mode Correction and Graph-Free Controls

The common-mode correction in Equation (1) uses current observations to update the target node's weekly reference, with neither learned parameters nor a graph. As a complementary control, we retain centring in a recurrent model but remove its spatial connections. Appendix A.2.6 reports the full Shock results at $r = 0.5$.

Common-mode correction substantially reduces baseline error: from 169.7 to 86.7 in Chicago and from 1,095.6 to 543.4 in Subway, and by approximately one quarter in the other two datasets. This gives a more demanding comparison than seasonal persistence alone. The graph-free `GRU + KRIN` model has higher error than the corrected baseline on all four datasets, showing that centring alone is insufficient. A spatial-model advantage nevertheless remains in the simplified evaluation: `SPIN-s + KRIN` has relative MAE margins of -12.7% to -25.7% against the corrected baseline and -19% to -60% against the centred graph-free model. Selecting the backbone on validation gives -8.5% to -25.7% against +CM (Appendix A.2.8), with Airports evaluated on `shock2020`.

These are validation-selected simplified-model results, not guarantees for the released implementations. `SPIN-s + KRIN` also avoids large raw-scale error increases from Normal to Shock across all four datasets (Appendix A.3).

C.6 Persistence on the Original METR-LA Benchmark

The preceding measurements use datasets constructed for this study. To examine the baseline issue raised in §1 on a source paper's own benchmark, we also apply a training-free predictor directly to the original data.

We follow the point-missingness setting of GRIN (Cini et al., 2022). METR-LA (Li et al., 2018) contains 34,272 time steps at 5-minute intervals for 207 sensors. Zeros indicate missing values. We randomly mask 25% of available entries and measure reconstruction MAE on the evaluation portion of a chronological 70/10/20 split. Three checks support protocol agreement: the original missing rate is 8.1%, matching the reported 8.10%; injected missingness accounts for 23.0% of all entries, matching 23.00%; and the `Mean` baseline yields 7.51 compared with the published 7.56, a difference of less than 0.7%.

Table 44: Pointwise imputation on METR-LA: MAE when 25% of available entries are masked within the chronological 70/10/20 split. Reported rows are transcribed from Cini et al. (2022); only LOCF is measured here. The comparison evaluates persistence on the source benchmark and contains no KRIN results.

Method	MAE	Trained	Source
KNN	7.88	No	reported
Mean	7.56	No	reported
MF	5.56	Yes	reported
MICE	4.42	Yes	reported
LOCF (previous step)	2.94	No	this work
rGAIN	2.83	Yes	reported
VAR	2.69	Yes	reported
MPGRU	2.44	Yes	reported
BRITS	2.34	Yes	reported
GRIN	1.91	Yes	reported

LOCF has lower error than four of the eight baselines reported in the source paper—`Mean`, `KNN`, `MF` and `MICE`—but is absent from its comparison table. The published learned methods nevertheless outperform LOCF on this benchmark: `GRIN` reports MAE 1.91 and `BRITS` 2.34, both below our measured LOCF MAE of 2.94.

This result is consistent with our conditional interpretation. Applying the diagnostic to METR-LA gives $|\text{shift}| = 0.069$, with 0.1% of evaluation time steps below the training-period minimum. The shift is about an order of magnitude smaller than Bikeshare's 0.797, and the published GRIN and BRITS errors are below our directly measured LOCF error. This is a low-shift comparison without observed baseline failure. We do not reproduce the learned models or evaluate their errors with and without KRIN in this experiment.

The METR-LA comparison preserves the evidence that GRIN and BRITS outperform simple persistence under low level shift. Our mobility experiments show that failures relative to persistence can instead become widespread under large shifts. Performance on a low-shift benchmark alone therefore does not establish robustness to large level changes.

C.7 Observation Information and Conditional Recovery

We apply controlled level shifts only at evaluation to the Normal scenarios in Chicago and Subway. The global/local comparison and history-sensitivity sweep use the earlier $r = 0.5$, 2 training seeds $\times$ 3 mask seeds design. The redesigned observation-location control in Appendix C.7.3 uses 3 training seeds $\times$ 5 placement seeds and scores a fixed subset within each placement. Models are trained once per training seed and their weights are held fixed across interventions. The two designs and their result files must not be pooled.

C.7.1 Information Available to Each Predictor

Table 45: Information available to the four predictors in the controlled recovery experiments. SPIN-s and SPIN-s + KRIN receive the same observations; KRIN changes how the target-history level is processed. +CM denotes `SNaive-7` with current common-mode correction.

Predictor	Information used
`SNaive-7`	Target node's value from one week earlier; no current observations
`SNaive-7`+CM	The same reference plus the current common change among observed nodes
SPIN-s	Node histories, current observations and masks; learned spatial aggregation
SPIN-s + KRIN	The same inputs; nodewise K-day means are removed and restored at the output

KRIN receives no additional observations. SPIN-s already has access to each node's history; KRIN explicitly removes and restores its mean outside the learned mapping. Similarly, +CM does not operate on current observations alone: it also requires the target node's value from one week earlier.

C.7.2 Global and Local Shifts at Matched System-Average Magnitude

A spatially uniform shift and a shift concentrated within a cluster can define different reconstruction problems even at the same system-average diagnostic value (Table 46).

Table 46: Global versus local shifts with matched system-average |shift|: MAE at $r = 0.5$, 2 training seeds $\times$ 3 mask seeds. Local $\alpha = 0.25$ affects 43/143 Chicago and 131/437 Subway nodes around high-degree centres. Bisection sets global $\alpha = 0.6773$ and 0.6613, yielding matched magnitudes 1.126 and 0.841, respectively. Parentheses give the ratio of the displayed mean MAE to the same-condition `SNaive-7` mean MAE, not mean paired relative differences or nodewise NMAE. Nodewise shift distributions remain different.

Dataset	Shift	`SNaive-7`	+CM	SPIN-s	SPIN-s + KRIN
Chicago	None	174.3	123.5 (0.709)	123.1 (0.706)	99.3 (0.570)
Chicago	Global	118.0	83.6 (0.708)	82.7 (0.701)	67.3 (0.570)
Chicago	Local	140.2	100.1 (0.714)	104.7 (0.747)	79.8 (0.569)
Subway	None	845.3	536.7 (0.635)	518.4 (0.613)	430.6 (0.509)
Subway	Global	559.0	354.9 (0.635)	318.2 (0.569)	284.8 (0.509)
Subway	Local	611.6	390.7 (0.639)	439.7 (0.719)	311.8 (0.510)

Raw-scale MAE is higher under the local shift, but the remaining demand also differs. Using the same-condition seasonal baseline as a reference, $R_{m,c} = \overline{\mathrm{MAE}}_{m,c}/\overline{\mathrm{MAE}}_{\mathrm{SNaive},c}$, SPIN-s changes from 0.701 to 0.747 in Chicago and from 0.569 to 0.719 in Subway, increases of approximately 6.6% and 26.3%. KRIN remains nearly unchanged at 0.570 to 0.569 and 0.509 to 0.510. Thus, at matched system-average shift, localisation changes the uncentred model's relative error while KRIN largely preserves its relative performance in these conditions. This supplementary ratio does not remove every scale effect. In particular, the raw-MAE increases of 18.6% and 9.5% do not by themselves establish vulnerability of KRIN to local shifts. The ratios use rounded aggregate errors; neither per-run uncertainty nor nodewise NMAE is reconstructed from those aggregates.

C.7.3 Observation Location with a Matched No-Shift Control

Pilot audit and matched design. The original pilot used a fixed `default_rng(1234)` for its Inside/Outside sets, so nominal mask seeds repeated one placement for each fitted model. We replace that pilot with training seeds $\{0, 1, 2\}$ crossed with placement seeds $\{0, 1, 2, 3, 4\}$. Each placement fixes the targets and both observation sets across no-shift and local-shift conditions. Chicago has a 43-node shifted cluster, 21 targets and 22 observed nodes; Subway has a 131-node cluster, 65 targets and 66 observed nodes. Clusters follow the high-degree construction in Appendix C.7.2, with local $\alpha = 0.25$. These counts define a target subset, not a 50% hidden fraction of the full graph.

SPIN-s and the released IGNNK adapter (registry `ignnk_off`) are evaluated with and without KRIN under Normal/`inregime` training. Each configuration has three fitted models evaluated at five placements and four shift–coverage combinations: 240 learned-model evaluations per dataset,

not 240 independent fits. The pairing keys have no missing or duplicate entries. Removing repetitions across model and training-seed labels gives 20 baseline rows per dataset (five placements, two shifts and two coverage arms).

Matching the scored entries. Chicago's original model and baseline scores use identical entries. In the original Subway runs, 38 of 95,184 target-valid entries across the five placements lacked a valid lag-7 reference and were excluded only from `SNaive-7`. These are placement-specific scoring counts: a date–node pair can occur in more than one placement. The final Subway evaluation restricts every method to the same 95,146 entries with valid targets and weekly references. Table 47 summarises the final inputs.

Existing Subway predictions and checkpoints were unavailable, so all four configurations were retrained with three training seeds using the current experiment code and training-input settings. The final analysis therefore combines original Chicago models with newly fitted Subway models. It does not obtain the new Subway MAEs by subtracting 38 errors from the old results. Separate scores on the original all-target-valid set track variation due to retraining (Table 51). Earlier Subway cell means and statistics on mismatched scoring sets are excluded from the final comparisons.

Table 47: Final coverage-control input audit. Learned rows count model–seed–placement–condition evaluations; baseline rows are deduplicated across models and training seeds. Mismatches count final model/baseline scoring-set discrepancies. File paths and audit outputs are identified in the Reproducibility Statement and accompanying file index.

Dataset	Final learned-model source	Learned rows	Baseline rows	Mismatches
Chicago	Original runs	240	20	0
Subway	Matched-support retraining	240	20	0

Normalisation within each matched run. Let $L_{mab,z,c}$ denote MAE for configuration m, training seed a, placement b, shift $z \in \{0, 1\}$ and coverage arm $c \in \{\text{in}, \text{out}\}$. Let $S_{b,z}$ be the matched `SNaive-7` MAE. This baseline uses the same targets and evaluation days; it is independent of training seed and identical across the two coverage arms. We calculate

$$Q_{mab,z,c} = L_{mab,z,c}/S_{b,z}, \qquad P_{mab,z} = Q_{mab,z,\text{out}} - Q_{mab,z,\text{in}},$$
$$D_{mab} = P_{mab,1} - P_{mab,0}, \qquad R_{ab} = D_{\text{base},ab} - D_{\text{KRIN},ab}.$$

These contrasts are formed before averaging across the 15 combinations. They are measured in multiples of same-condition baseline error, not percentages or nodewise NMAE. In general, an average of run-level ratios differs from a ratio of average errors. This distinction matters because all scored targets lie in the cluster whose demand is multiplied by 0.25. Negative raw-MAE excess penalties after KRIN, or raw reductions exceeding 100%, do not by themselves establish removal of the location effect.

Uncertainty under crossed dependence. We use 20,000 crossed-factor bootstrap draws with RNG seed 0, following the row/column resampling principle of Owen (2007). Each draw independently samples three training-seed IDs and five placement IDs with replacement and averages their Cartesian product. The same draws are shared across all models and conditions, preserving pairing. For sampled IDs $a_1^*, \ldots, a_3^*$ and $b_1^*, \ldots, b_5^*$,

$$\overline{R}^* = \frac{1}{15}\sum_{j=1}^{3}\sum_{k=1}^{5} R_{a_j^* b_k^*}, \qquad \overline{R}_a = \frac{1}{5}\sum_{b=1}^{5} R_{ab}.$$

The first quantity generates the bootstrap distribution; the second averages the five placement-specific reductions within one training seed. Only three fitted models per configuration inform training variability, so the intervals remain exploratory. The 15 combinations are not independent training replicates, and direction counts are not significance tests. Table 49 reports all twelve dataset–backbone–training-seed means; Table 50 gives the corresponding condition-specific MAEs. Both are computed from the released run-level records before rounding. The complete values and input hashes are provided with the final analysis code (Appendix F.2).

Table 48: Normalised excess outside penalty D on matched scoring sets: mean, 95% crossed-factor bootstrap interval, and observed positive combinations out of 3 training seeds $\times$ 5 placements. These counts concern D, not the KRIN reduction R in Table 4. All four centred intervals include zero; this is not an equivalence test. Six decimal places retain the small residual changes.

Dataset	Model	Mean D	95% interval	$D > 0$
Chicago	SPIN-s	1.058263	[0.560654, 1.471025]	15/15
Chicago	SPIN-s + KRIN	0.000118	[−0.000171, 0.000458]	8/15
Chicago	IGNNK-R	0.412556	[0.352517, 0.483631]	15/15
Chicago	IGNNK-R + KRIN	−0.000096	[−0.000381, 0.000184]	6/15
Subway	SPIN-s	1.379266	[1.160771, 1.568462]	15/15
Subway	SPIN-s + KRIN	−0.000022	[−0.000303, 0.000231]	9/15
Subway	IGNNK-R	0.408433	[0.243650, 0.531726]	15/15
Subway	IGNNK-R + KRIN	−0.000003	[−0.000180, 0.000211]	9/15

Table 49: Normalised effects by training seed for the final matched coverage control. Each row averages exactly five placement-specific contrasts within one fitted training seed: $\overline{D}_{m,a} = 5^{-1}\sum_b D_{mab}$ and $\overline{R}_a = 5^{-1}\sum_b R_{ab}$. Normalisation and condition pairing precede averaging. The source is `results/coverage/matched/training_seed_means.csv`; nine decimal places retain the small centred residuals. These are twelve descriptive means, not twelve independent replications of a single effect.

Dataset	Backbone	Training seed	$\overline{D}_{\mathrm{base},a}$	$\overline{D}_{\mathrm{KRIN},a}$	$\overline{R}_a$
Chicago	SPIN-s	0	1.207825409	0.000131512	1.207693897
Chicago	SPIN-s	1	0.549139578	−0.000023331	0.549162910
Chicago	SPIN-s	2	1.417822661	0.000246586	1.417576075
Chicago	IGNNK-R	0	0.396298375	−0.000032297	0.396330672
Chicago	IGNNK-R	1	0.457856634	−0.000155850	0.458012484
Chicago	IGNNK-R	2	0.383512271	−0.000099940	0.383612211
Subway	SPIN-s	0	1.149977674	0.000051900	1.149925774
Subway	SPIN-s	1	1.481966362	−0.000252571	1.482218933
Subway	SPIN-s	2	1.505853011	0.000134200	1.505718811
Subway	IGNNK-R	0	0.235682806	−0.000072469	0.235755275
Subway	IGNNK-R	1	0.510026775	−0.000099049	0.510125824
Subway	IGNNK-R	2	0.479588550	0.000162918	0.479425631

Table 50: MAE by training seed and condition, averaged over placement seeds 0–4, for the final matched coverage control. The same fitted model is evaluated in all four columns of a row; target and observation sets are paired across the two shift conditions. Chicago uses the original fits and Subway the matched-support rerun. These means retain demand-scale differences and do not replace the per-run normalisation used in Table 4. Full-precision MAEs and means of per-run MAE/`SNaive-7` ratios are in `results/coverage/matched/training_seed_mae.csv`.

			No shift		Local shift	
Dataset	Model	Seed	Inside	Outside	Inside	Outside
Chicago	SPIN-s	0	165.299	171.509	43.806	110.143
Chicago	SPIN-s	1	152.274	152.510	62.509	92.259
Chicago	SPIN-s	2	188.504	192.158	62.862	140.437
Chicago	SPIN-s + KRIN	0	127.997	130.117	32.003	32.543
Chicago	SPIN-s + KRIN	1	113.673	115.148	28.428	28.796
Chicago	SPIN-s + KRIN	2	118.761	120.982	29.688	30.259
Chicago	IGNNK-R	0	213.708	217.615	76.307	98.899
Chicago	IGNNK-R	1	189.060	195.561	82.586	109.292
Chicago	IGNNK-R	2	217.831	227.838	133.765	157.148
Chicago	IGNNK-R + KRIN	0	160.530	169.988	40.121	42.486
Chicago	IGNNK-R + KRIN	1	144.155	153.184	36.035	38.285
Chicago	IGNNK-R + KRIN	2	147.125	156.195	36.785	39.049
Subway	SPIN-s	0	643.607	670.080	170.821	463.498
Subway	SPIN-s	1	570.198	581.463	284.252	655.584
Subway	SPIN-s	2	639.426	663.281	190.678	571.060
Subway	SPIN-s + KRIN	0	515.947	518.245	129.095	129.683
Subway	SPIN-s + KRIN	1	492.438	497.921	123.219	124.526
Subway	SPIN-s + KRIN	2	499.417	500.822	124.938	125.322
Subway	IGNNK-R	0	1042.768	1044.907	363.736	422.720
Subway	IGNNK-R	1	900.216	901.543	399.338	526.150
Subway	IGNNK-R	2	924.989	923.135	344.736	463.390
Subway	IGNNK-R + KRIN	0	767.864	766.053	192.025	191.555
Subway	IGNNK-R + KRIN	1	791.640	793.875	197.891	198.425
Subway	IGNNK-R + KRIN	2	883.363	887.176	220.883	221.879

Interpretation and audit. The uncentred normalised excess is positive in every observed combination for both backbones and cities. KRIN reduces it in all 15 combinations in each comparison; all four intervals for the mean reduction have positive lower endpoints (Table 4). After centring, the mean excess is close to zero and the signs across combinations are mixed. This supports approximate preservation of the no-shift relative penalty in this design, not elimination of the penalty itself or proof of equivalence. It does not establish superiority over +CM, and generalisation beyond the tested evaluation windows, cluster construction and shift magnitude remains untested.

The recalculation audit checks all 240 new Subway prediction files. The maximum absolute difference between prediction-reaggregated and stored MAE is 1.4×10^{-5}, consistent with float32 accumulation-order effects; recomputing baseline errors from raw data gives a maximum difference of 2.27×10^{-13}. Final model/baseline scoring-set mismatches are zero in both datasets. The retraining comparison in Table 51 distinguishes these numerical checks from variation between fitted models: SPIN-s changes substantially even when evaluated under the original scoring rule.

Table 51: Subway retraining audit: maximum absolute MAE difference between the original and newly fitted models, both scored on the original all-target-valid entries. This comparison holds the scoring rule fixed, separating retraining variation from the final restriction to entries with valid weekly references. Zeros are at the displayed precision. These values are audit differences, not coverage penalties.

Model	Maximum absolute replay difference
SPIN-s	109.647543
SPIN-s + KRIN	15.503583
IGNNK-R	0.000000
IGNNK-R + KRIN	0.000000

C.7.4 Sensitivity to Missing Recent Target-Node History

We evaluate SPIN-s and SPIN-s + KRIN trained under Normal, applying a fixed global shift and recent-history masking only at evaluation. This is an input-sensitivity experiment: the history intervention affects only the learned models' hidden-node inputs. The baseline inputs are held fixed. In particular, `SNaive-7` and +CM still use the target node's weekly reference; their invariance here results from excluding them from the intervention, not from independence of target-node history.

Fixed window and preprocessing. With $K = 14$, the dataset constructs the history as

$$\tilde{H}_t = [\tilde{x}_{t-K}, \ldots, \tilde{x}_{t-1}] \in \mathbb{R}^{K \times N}.$$

The slice `bundle.scaled[t-K:t]` covers $t-14$ through $t-1$, inclusive, and excludes the target day. The window is not moved backwards. Preprocessing applies `log1p` followed by nodewise standardisation fitted only on training rows. Raw missing entries are replaced by zero after scaling. Thus, zero represents the node's training-period mean in log-transformed space, not its arithmetic mean on the raw scale.

The implementation is `make_dataset` in `src/kriging_data.py`, and `experiments/run_history.py` runs the expanded sweep. Raw-gap filling occurs when the bundle is built. At sample construction, recent-slot masking is applied before the optional Protocol B history-mask fill. That optional fill uses each node's observed-slot mean, or zero when no slots are observed; it is used in Appendix C.1 and is disabled in this sweep. A retained slot can already contain a value imputed for a raw calendar gap; the slot counts below describe retention under the additional intervention.

Recent-slot masking and the restored level. Let $m_i = 0$ identify a hidden target node and let $q = K - d$, with d capped at K. The intervention keeps the first q slots, dated $t-K, \ldots, t-d-1$, and replaces the last d slots with their retained-slot mean. It neither repeats the last observation nor retrieves an earlier complete window. The operation is:

```
h   = copy(bundle.scaled[t-K:t])  # Protocol B mask disabled here
d   = min(d, K)
q   = K - d
hid = (observed == 0)
fill = mean(h[:q, hid], axis=0) if q > 0 else 0.0
h[q:K, hid] = fill
```

Histories of observed nodes, current-day inputs and the target-day mask are unchanged. For $q > 0$, define $a_i = q^{-1} \sum_{j=0}^{q-1} \tilde{H}_t[j, i]$. The mean that KRIN computes from the filled history $\tilde{H}_t^{(d)}$ is exactly

$$\mu_i^{(t,d)} = \frac{1}{K} \sum_{j=0}^{K-1} \tilde{H}_t^{(d)}[j, i] = \begin{cases} \dfrac{qa_i + da_i}{K} = a_i, & q > 0, \\ 0, & q = 0. \end{cases}$$

Mean filling therefore preserves the mean of the retained preprocessed slots. It does not establish that this mean is an unbiased estimate of the current level. Increasing d changes both the number and the dates of retained slots. For one traced Chicago target node, the means at $d = 0, 3, 7, 14$ are -1.2396, -1.1616, -1.2419 and 0.0000, respectively. At $d = 14$, no slot remains: the hidden-node history and its KRIN mean both revert to zero, the training-mean reference in transformed space.

Table 52: Expanded history-masking sweep: raw-scale MAE under the fixed global-shift settings of Table 46, with $r = 0.5$ and 2 training seeds $\times$ 3 mask seeds. Each learned configuration is trained once per training seed and reused across d. All learned-model entries come from this separate expanded rerun. $q = 14 - d$ counts retained history slots, and "latest" gives the date of the most recent retained slot. +CM denotes `SNaive-7`+CM with its inputs held fixed. The separated final row in each panel removes every slot and invokes the training-mean fallback; it is not an older-observation condition.

History condition	q	Latest	+CM	SPIN-s	SPIN-s + KRIN
Chicago					
$d = 0$ (complete window)	14	$t - 1$	83.6	86.8	67.5
$d = 3$	11	$t - 4$	83.6	151.2	123.4
$d = 7$	7	$t - 8$	83.6	156.4	138.9
$d = 10$	4	$t - 11$	83.6	176.4	156.1
$d = 12$	2	$t - 13$	83.6	163.4	134.2
$d = 13$	1	$t - 14$	83.6	160.5	126.1
No observation in window ($d = 14$) (training-mean fallback)	0	None	83.6	1,008.6	1,182.8
Subway					
$d = 0$ (complete window)	14	$t - 1$	354.9	322.1	281.6
$d = 3$	11	$t - 4$	354.9	825.5	756.6
$d = 7$	7	$t - 8$	354.9	904.3	912.2
$d = 10$	4	$t - 11$	354.9	1,008.5	996.1
$d = 12$	2	$t - 13$	354.9	782.2	748.2
$d = 13$	1	$t - 14$	354.9	699.2	621.1
No observation in window ($d = 14$) (training-mean fallback)	0	None	354.9	6,249.1	6,580.9

An abrupt failure when no observations remain. Table 52 distinguishes two changes. First, masking the three most recent days increases KRIN MAE from 67.5 to 123.4 in Chicago and from 281.6 to 756.6 in Subway. Both learned configurations exceed the fixed +CM value at every tested positive d. Between $d = 3$ and $d = 13$, however, error is nonmonotonic; retaining fewer, older slots does not produce a steady increase. The aggregate means alone do not establish whether these fluctuations fall within seed variability, nor do they separate changes in level estimation from changes in the residual predictor's input.

Second, removing the final retained slot produces a discontinuity. With one slot from $t - 14$ still available ($d = 13$), KRIN MAE is 126.1 in Chicago and 621.1 in Subway. With no slots ($d = 14$), it reaches 1,182.8 and 6,580.9, respectively: increases by factors of approximately 9.4 and 10.6. Uncentred SPIN-s also deteriorates sharply, from 160.5 to 1,008.6 and from 699.2 to 6,249.1. The final row therefore measures complete removal of within-window level information with a training-mean fallback, rather than gradual ageing of an observed history. KRIN has higher error than uncentred SPIN-s in this condition, so centring does not prevent failure after that information is lost.

For $q > 0$, mean filling preserves a constant offset already present in the retained slots. At $q = 0$, the fixed zero fill cannot carry such an offset: the supplied hidden-node history no longer satisfies the premise of the equivariance result. This identifies where the level information is removed without attributing all resulting error to mean restoration alone. The comparison with fixed +CM values does not validate a model-switching threshold at $d = 3$.

D Evaluation on a Held-Out Non-Mobility Dataset

The four preceding datasets all measure mobility demand, and their real shock stems from the same pandemic event. The diagnostic's empirical range and analysis choices were established using those datasets and the sweep in Appendix C.2. We now hold those choices fixed and evaluate a fifth dataset that was not used to develop the analysis.

We use the US EPA's daily NO_2 monitoring network (US Environmental Protection Agency, 2026), containing 336 stations over 1,461 days. The signal is daily mean concentration in ppb. Its levels are

influenced by weather and emissions rather than mobility demand. The protocol, graph construction, masks, models and seeds match the main experiments.

We selected this dataset as a low-shift control and specified the selection rationale and predictions before running the experiment. Strong weather-related annual variation in NO_2 was expected to produce a broad training-period level range, a low diagnostic value and no failures relative to the training-free baseline. The measured |shift| is 0.159, below Bikeshare's 0.797 and below the failure-onset range observed after excluding GRIN-s in the sweep of Appendix C.2.

We prespecified two predictions: no backbone would have higher error than the training-free baseline, and KRIN would provide little additional improvement.

Table 53: Held-out EPA NO_2 network: Normal and 2020 Shock MAE in ppb at $r = 0.5$. Simplified variants under the common protocol and STGNN use the same graph construction and 3 training seeds $\times$ 10 mask seeds as the mobility experiments. Values are rounded to two decimals, which can hide small changes.

Backbone	Normal, without KRIN	Normal, + KRIN	Shock, without KRIN	Shock, + KRIN
IGNNK-s	1.58	1.58	1.67	1.67
SATCN-s	1.58	1.58	1.66	1.66
GRIN-s	1.89	1.72	1.99	1.79
SPIN-s	1.77	1.77	1.95	1.95
KITS-s	1.59	1.59	1.70	1.68
STGNN	1.58	1.58	1.67	1.67

Under Normal conditions, the training-free MAEs are 3.44 for `SNaive-7`, 2.52 for `Naive-1` and 2.64 for HA; under Shock they are 3.58, 2.75 and 2.82, respectively. All six backbones outperform all three baselines in both conditions (0/6 failures). KRIN's mean relative effect is -1.6% under Normal and -2.1% under Shock, with small degradations in four of six backbones under Normal and three of six under Shock. The 141 improvements and 3 rounded ties in the main mobility study (§7.4) therefore do not extend automatically to every domain. GRIN-s improves more substantially, by -8.9% under Normal and -10.3% under Shock. The results support a small average effect for most backbones, but a low diagnostic value does not guarantee negligible benefit for each individual backbone.

The predicted absence of baseline failures and the small average centring benefit are observed, but GRIN-s remains a clear exception to the small-effect pattern. Figure 1 shows the low NO_2 shift and the small fraction of evaluation days below the seen minimum (approximately 2%).

`Naive-1` is also stronger than `SNaive-7` here (2.52 versus 3.44), reversing the ordering associated with weekly patterns in the mobility data (§4.1). This is consistent with a weaker weekly persistence signal in NO_2.

Because no backbone fails relative to the baselines, this experiment provides a low-shift negative control. The real-data cases combining large shifts with baseline failures remain the three mobility networks described in §7.5; Appendix C.2 extends shift magnitudes and directions within the same four mobility datasets. A large-shift positive case in a non-mobility domain remains untested.

E Released Implementations under the Reconstruction Protocol

We use the suffix `-R` to distinguish these models from the simplified backbones (-s). The authors' model code is left unchanged, with an external adapter connecting it to our task. The adapter forms a length-$K + 1$ window $[H_t; x_t \odot m]$, applies the node mask at the final time step, and reads the output at that step as the prediction. Architectures and the hyperparameters listed below follow the released implementations, while training follows our common procedure. In particular, these results do not reproduce the full KITS incremental training strategy.

Table 54: Architecture and optimisation settings retained from the released implementations (-R). Each model is adapted to the same final-time-step reconstruction task; the common training procedure still applies. In particular, KITS' full incremental training strategy is not reproduced.

Backbone	Released architecture settings	Optimisation settings
IGNNK-R	hidden 100, diffusion order 1	lr 1e-4, batch 4
SATCN-R	channels 128, 1 layer, kernel 2	lr 1e-3, batch 8
GRIN-R	d_hidden 64, d_emb 8, d_ff 64, dropout 0	batch 32
SPIN-R	hidden 32, 4 layers, eta 3	lr 8e-4, batch 8 split into 2
KITS-R	d_hidden 64	batch 32

Performance differences between backbones remain interpretable in this evaluation. However, the models differ in size and optimisation hyperparameters, so those differences cannot be attributed solely to the aggregation architecture. The simplified variants provide complementary comparisons under a shared hidden dimension and training procedure, although they also have unequal total parameter counts. Both evaluations support comparisons of the same backbone with and without KRIN, as well as comparisons against training-free baselines.

E.1 Full MAE Results

Table 55: Released implementations: Normal / Shock MAE at $r = 0.5$ on A^{geo}, averaged over 3 training seeds $\times$ 10 mask seeds. Airports Shock is 2020–21 (`all`); other datasets use 2020. Five backbones $\times$ four datasets $\times$ two conditions give 40 paired comparisons. The +CM comparison using the same evaluation windows is in Table 56.

Method	Bikeshare	Chicago	Airports (all)	Subway
`SNaive-7`	12.4 / 10.8	189.0 / 169.7	11.5 / 11.35	839.5 / 1,095.6
IGNNK-R	8.3 / 7.9	186.5 / 192.5	10.4 / 11.7	751.0 / 1,783.4
IGNNK-R + KRIN	7.8 / 6.8	120.3 / 95.4	10.0 / 8.6	562.2 / 743.2
SATCN-R	9.9 / 9.3	287.9 / 265.6	34.3 / 43.6	1,290.6 / 2,467.8
SATCN-R + KRIN	7.9 / 6.7	155.8 / 111.9	16.0 / 12.1	890.9 / 1,103.4
GRIN-R	7.8 / 7.1	179.6 / 170.6	10.1 / 21.5	714.7 / 1,892.2
GRIN-R + KRIN	7.7 / 6.7	120.8 / 121.1	9.8 / 8.7	490.6 / 574.6
SPIN-R	7.4 / 6.7	181.1 / 569.2	9.5 / 21.7	683.7 / 3,330.6
SPIN-R + KRIN	8.4 / 6.3	268.4 / 267.8	11.6 / 15.0	498.9 / 738.4
KITS-R	10.2 / 9.8	364.2 / 296.9	36.2 / 42.2	1,350.5 / 2,207.3
KITS-R + KRIN	8.1 / 7.0	169.4 / 126.3	16.5 / 12.2	884.4 / 1,149.7

Of 40 before-and-after comparisons, 37 show lower MAE and 3 show higher MAE. All three increases in MAE occur for SPIN-R under Normal conditions: 7.4 to 8.4 in Bikeshare, 181.1 to 268.4 in Chicago and 9.5 to 11.6 in Airports. Under Shock, all five backbones improve on all four datasets. The number of Shock configurations with higher error than `SNaive-7` falls from 15/20 to 6/20. The median relative change computed from the rounded table entries is approximately -31.2%.

E.2 Direct Comparison and Interpretation

In Table 56, the best centred released-code model is SPIN-R in Bikeshare, IGNNK-R in Chicago and Airports, and GRIN-R in Subway. The best models outperform +CM in Bikeshare and Airports, but the strengthened baseline remains better in Chicago and Subway. The released implementations therefore support broad error reduction, with additional gains over the strengthened baseline depending on the dataset.

Table 56: Best centred configurations in each evaluation versus training-free baselines on matched Shock windows at $r = 0.5$. Airports uses 2020–21 (`all`) in every column; other datasets use 2020. Best denotes the lowest reported evaluation MAE among the models in that group, rather than selection on validation data. The validation-selected analysis in Table 17 uses Airports 2020 only.

Dataset	Shock slice	`SNaive-7`	+CM	Best -s + KRIN	Best -R + KRIN
Bikeshare	2020	10.8	8.3	6.2	6.3
Chicago	2020	169.7	86.7	70.0	95.4
Airports	2020–21 (all)	11.35	9.46	8.2	8.6
Subway	2020	1,095.6	543.4	432.6	574.6

GRIN-R does not fail against `SNaive-7` under Normal conditions, so GRIN-s's Normal failures should not be attributed to the original method. Together with the partial directionality and parameter-budget controls in Appendix B.2.5, this suggests an effect of simplification. However, adapting the released code changes several implementation components together and does not isolate a particular autoregressive stage as the cause.

E.3 Training-Seed Variability and Scale-Normalised Error

Table 57: Released-code Shock results with training-seed variability at $r = 0.5$, using 3 training seeds $\times$ 10 mask seeds. MAE is reported with the standard deviation across training seeds; a dash denotes a training-free method. NMAE(macro) is in percent. Airports uses `all`; 0.0 is a rounded standard deviation, not a claim of identical runs.

Dataset	Method	MAE	Training-seed SD	$\text{NMAE}_{\text{macro}}$ (%)
Chicago	`SNaive-7`+CM	86.7	–	7.6
Chicago	IGNNK-R + KRIN	95.4	5.0	8.2
Chicago	SPIN-R	569.2	292.3	41.9
Chicago	SPIN-R + KRIN	267.8	154.9	22.4
Subway	`SNaive-7`+CM	543.4	–	7.7
Subway	GRIN-R + KRIN	574.6	6.3	8.3
Subway	SPIN-R	3,330.6	1,798.8	36.3
Subway	SPIN-R + KRIN	738.4	28.6	9.6
Airports	`SNaive-7`+CM	9.46	–	19.4
Airports	IGNNK-R + KRIN	8.6	0.0	17.0

After centring, SPIN-R's standard deviation across training seeds falls from 292.3 to 154.9 in Chicago and from 1,798.8 to 28.6 in Subway. These cases show reduced variation between fitted models alongside lower mean error, although substantial variation remains in Chicago.

The Chicago and Subway ordering persists under NMAE(macro). +CM achieves 7.6 and 7.7, respectively, below the best released-code values of 8.2 and 8.3. The baseline advantage in these two datasets therefore cannot be explained solely by the influence of high-volume nodes on raw-scale MAE.

E.4 Rank Changes for the Released Implementations

Table 58: Released-code Shock MAE ranks before $\rightarrow$ after KRIN, computed from Table 55. Rank 1 is best; ties in displayed MAE receive average ranks. Bikeshare SATCN-R and GRIN-R both display 6.7 after KRIN and share rank 2.5. The final row counts models whose rank changes under this rounding-aware convention.

Model	Bikeshare	Chicago	Airports (all)	Subway
IGNNK-R	$3 \rightarrow 4$	$2 \rightarrow 1$	$1 \rightarrow 1$	$1 \rightarrow 3$
SATCN-R	$4 \rightarrow 2.5$	$3 \rightarrow 2$	$5 \rightarrow 3$	$4 \rightarrow 4$
GRIN-R	$2 \rightarrow 2.5$	$1 \rightarrow 3$	$2 \rightarrow 2$	$2 \rightarrow 1$
SPIN-R	$1 \rightarrow 1$	$5 \rightarrow 5$	$3 \rightarrow 5$	$5 \rightarrow 2$
KITS-R	$5 \rightarrow 5$	$4 \rightarrow 4$	$4 \rightarrow 4$	$3 \rightarrow 5$
Changed	3 / 5	3 / 5	2 / 5	4 / 5

After centring, ranks change for 3/5 backbones in both Bikeshare and Chicago, 2/5 in Airports and 4/5 in Subway. The Bikeshare count assigns average ranks to ties in the rounded errors and therefore does not resolve the small difference between SATCN-R and GRIN-R. Performance rankings can be compared, but differences between models with different sizes and optimisation settings cannot be attributed solely to their aggregation architectures.

F CODE AND EXPERIMENT-TO-RESULT MAPPING

The anonymous repository is available at `https://anonymous.4open.science/r/KRIN-91E4/`. Code links open this repository, and the displayed paths are relative to its root. The accompanying Table 4 package contains the final coverage-analysis code and run records listed below; other experiment output paths identify generated files rather than a complete archive of empirical results. Dependencies are listed in `requirements.txt`. The dataset configurations are `configs/bike.yaml` (Bikeshare), `configs/cta.yaml` (Chicago), `configs/base.yaml` (Airports), `configs/mta.yaml` (Subway) and `configs/aqs.yaml` (NO_2). Each configuration defines its raw-data, processed-data and result directories.

F.1 IMPLEMENTATION AND EXPERIMENT ENTRY POINTS

The input windows and separate target-day/history masks are implemented in `src/kriging_data.py`; `src/splits.py` fits the scaler on training rows, and `src/build_graphs.py` constructs the graphs. The registry in `src/models/__init__.py` maps `spin` to SPIN-s and `ignnk_off` to IGNNK-R. In general, `_off` selects a released-model adapter in `src/models/official_wrap.py`, while `_c` applies the mean-only `Centered` wrapper in `src/models/networks.py`. Thus, `spin_c` and `ignnk_off_c` denote the corresponding KRIN configurations. Shared optimisation and scoring are in `src/train.py` and `src/evaluate.py`. The +CM predictor is `predict_common_mode` in `src/baselines.py`.

Table 59: Experiment-to-code map. Experiment outputs are relative to each configuration's `paths.results`; final coverage-analysis paths are relative to the package root. The accompanying package includes the final coverage records and statistical outputs. Other rows identify generation procedures, not a complete release of all experiment results. Appendix F.2 gives the exact Table 4 command and input selection.

Analysis in the paper	Entry point	Output or input contract
Main and released-model evaluations	`experiments/run_main.py`	`runs_<tag>.csv`; `main_results_<tag>.csv`
Training-free baselines	`src/baselines.py`	`baselines.csv`
Original coverage design	`experiments/run_coverage.py`	`coverage_runs.csv`
Final matched coverage statistics	`analysis/analyse_coverage.py`	`results/coverage/matched/`
Matched global/local shifts and expanded history sweep	`experiments/run_history.py`	`infoexp.csv`; `exp=A` and `exp=C`
Initial transition errors	`experiments/run_transition.py`	`transition_daily.csv`
Shift alarms and reversals	`experiments/run_alarm.py`	`alarm_split.csv`
Validation-based model choice	`experiments/run_valselect.py`	`val_select.csv`
Released-model table	`analysis/fidelity_table.py`	Reads `runs_official2.csv` and `baselines.csv`
Figures and plotted values	`analysis/make_figures.py`	PDF/PNG/CSV under `results/figures/`

The main runner reads mask seeds from the configuration (0–9) and defaults to training seeds 0–2; these executable defaults take precedence over its older three-mask docstring. Set `-tag=geo3m10` for the main run-file names and `-tag=official2` for the filenames read by the released-model table script. Model names must be supplied explicitly when generating all uncentred/centred pairs: the runner's default model list is not that paired list. For Airports, `-split-shock-years` exports the separate 2020/2021 scores while retaining `all`. The Python function `diagnose` in `src/shift_amp.py` must likewise receive `sreg="shock2020"` for the main 2020 comparisons; its default is `all`.

F.2 Reproducing the Final Coverage Statistics

The accompanying package provides the final statistical implementation in `analysis/recalculate_coverage.py`, called by `analysis/analyse_coverage.py`. It forms each run's MAE/`SNaive-7` ratio before the paired contrasts and uses shared crossed-factor bootstrap draws. From the package root, the following command reproduces Table 4, Table 48, and the training-seed summaries using NumPy and pandas, without fitting a model:

```
python3 analysis/analyse_coverage.py --input-source matched \
  --output results/coverage/matched
```

The inputs are `results_cta/coverage_runs.csv` and `results_mta/coverage_matched_runs.csv`, each containing 240 learned-model evaluations and repeated same-condition baseline errors. The latter repetitions are checked and deduplicated; they do not create independent baseline observations. The alternative `-input-source original` uses the original Subway `coverage_runs.csv` and is retained only for sensitivity and provenance. It must not be substituted for the final table because its scoring support and fitted SPIN-s models differ.

The released outputs under `results/coverage/matched/` are `table4.csv` (normalised D and R, intervals and sign counts), `paired_contrasts.csv` (individual matched contrasts), `matched_snaive7.csv` (deduplicated baselines), `training_seed_means.csv` (twelve five-placement means), and `training_seed_mae.csv` (96 condition-specific means). The readable seed table is `training_seed_means.md`. A replay also writes normalised runs and detailed contrast summaries. The source choices, seed settings and SHA-256 input hashes are recorded in `audit.json`; `analysis/TABLE4.md` documents the output schema.

The CSV-only calculation reproduces the released numerical effects and bootstrap endpoints. It does not independently recheck which raw date–node entries contributed to each saved MAE. The recorded data-level audit is preserved as `support_audit.json`; a CSV-only replay labels its own scoring-support status `unverified`. Repeating the original support check requires `-verify-baselines` and processed observations. Adding `-verify-predictions` also requires the saved matched-rerun predictions. These processed data, predictions and model checkpoints are not included. This distinction separates a reproducible statistical calculation from a full rerun of training and per-entry scoring.

Four portable checks in `analysis/test_coverage_statistics.py` cover rejection of incomplete or duplicate pairs, normalisation under pure scale contraction, shared placement draws across training seeds, and five-placement seed summaries. They run with `python3 -m unittest discover -s analysis -p test_coverage_statistics.py -v`. The earlier `experiments/run_coverage.py` remains the original training/evaluation entry point; its original Subway scoring rule does not generate the matched rerun. The final table is tied explicitly to the released matched-run CSV, its provenance record and the analysis command above.