

Hierarchical Ensembling of Heterogeneous Time Series Models for Retail Inventory Decisions: A Local Empirical Study on Two Public Datasets

Jiexiao Zhang

discoveryopen.org

Correspondence: jiexiaozhang1@outlook.com

Abstract

Whether improvements in retail forecasting translate into inventory benefits must be tested under common temporal splits and continuously evolving inventory states. This study develops InventoryFusion, a heterogeneous forecast ensemble that combines four lightweight baselines with Tiny Time Mixers R2.1, Chronos-2, and TimesFM 2.5 through a common P10/P50/P90 interface. Ensemble weights are estimated by shrinking validation losses across global, demand-segment, and individual-series levels. The evaluation uses 120 series from each of FreshRetailNet-50K and UCI Online Retail, arranged into three disjoint sampling cohorts per dataset. A 14-day horizon, a 7-day rolling step, and separated weight-fitting and test periods yield 1,560 test forecast windows and 21,840 series-origin-horizon observations. Hierarchical ensembling achieves WAPE values of 36.66% and 89.02%, compared with 36.51% and 93.81% for equal weighting. On UCI, hierarchical WAPE is 5.115% lower than equal weighting but 3.800% higher than validation-selected Chronos-2. Its central 80% interval covers only 54.720% of observations, exposing lower-bound miscalibration at zero sales. Paired whole-series cluster bootstrap intervals based on 2,000 resamples are complemented by hierarchy ablations, delayed-label online updates, stockout-flag sensitivity analysis, and strictly separated residual calibration. A continuous lost-sales simulation uses a 7-day review interval and a 7-day lead time to compare cumulative-residual inventory targets with sums of daily P90 forecasts across three holding-to-shortage cost ratios. The findings show that ensemble utility depends on the data distribution, the granularity of weight estimation, and protection-period calibration. Adding models or improving point forecasts does not automatically reduce inventory cost. Version-pinned local execution, series-level outputs, and traceable figures support examination of both the benefits and the limitations of the ensemble.

Keywords: retail forecasting; time series foundation models; forecast combinations; hierarchical shrinkage; inventory simulation; probabilistic calibration

1 Introduction

Inventory management converts uncertainty about future demand into present purchasing and holding decisions. Underprediction can increase shortages, whereas overprediction can tie up capital. For a given replenishment interval and lead time, the relevant outcome is the joint behavior of service and inventory cost. A reduction in forecast error therefore cannot be equated directly with an inventory benefit. Using rolling inventory simulations, Petropoulos et al. showed that forecasting methods should be evaluated jointly in terms of error, holding and shortage costs, and customer service [1]. This principle remains relevant as foundation models enter retail forecasting: neither larger models nor a greater number of models replace an examination of their decision consequences.

Retail time series are heterogeneous. Fast-moving products may exhibit stable within-week patterns, whereas infrequently purchased products contain many zeros. Promotions, seasonal changes, and limited availability

further alter the observed distribution. Croston separately estimated nonzero demand sizes and inter-demand intervals, providing a classical approach to intermittent demand [2]. The TSB method updates the probability of demand occurrence each period, allowing extended zero-demand runs to reduce forecasts [3]. Research on classification using average inter-demand intervals and the variability of nonzero demand also indicates that different demand patterns call for different modeling preferences [4]. These findings motivate retaining computationally inexpensive, interpretable traditional baselines when assessing modern models and examining performance across product groups.

Time series foundation models provide additional candidate forecasters. Pretraining across datasets can learn transferable temporal representations and reduce the need to train a model for every series. Nevertheless, pretraining does not remove distributional differences in the target data, and several models may make highly correlated errors on the same product. This study asks a testable question: under local computing constraints, can combining heterogeneous foundation models with statistical baselines using historical validation losses improve retail forecasts and inventory outcomes under a specified replenishment mechanism? The evaluation examines alternative weighting granularities, simple combination baselines, out-of-time testing, probabilistic calibration, and continuous inventory dynamics to identify the benefits, costs, and scope of ensembling.

2 Related Work

2.1 Forecast Combinations and Adaptive Weights

Forecast combination has a long research history. The review by Wang et al. spans simple averages that require no weight estimation, time-varying weights, nonlinear combinations, and cross-series learning, while discussing the limitations imposed by weight-estimation error and correlated model errors [5]. With limited validation data, complicated weights need not outperform averaging. Simple averaging is consequently an important reference for testing adaptive mechanisms. FFORMA trains a meta-model on time series features to allocate weights to candidate forecasting methods, demonstrating an established approach to learning method suitability from series characteristics [6].

The present method shares historical error information across global, demand-segment, and individual-series levels to balance cross-series evidence with local adaptation. When local evidence is limited, shrinkage toward a higher-level estimate can reduce the influence of a few unusual errors. Local evidence receives more influence as observations accumulate. This is an interpretable forecast combination, not an addition or concatenation of parameters from different neural networks. Its utility is assessed against averaging, fixed global weights, validation-based expert selection, and ablations that remove either the local or segment level.

2.2 Time Series Foundation Models and Heterogeneous Ensembling

Chronos scales and quantizes continuous time series into discrete tokens and uses language-model architectures for probabilistic forecasting [7]. Chronos-2 uses an encoder architecture and group attention to support univariate, multivariate, and covariate-informed forecasting within a unified framework [8]. Tiny Time Mixers reduces computation through compact mixer networks and multiresolution pretraining [9]. TimesFM learns cross-series patterns using patched inputs and a decoder architecture [10], and version 2.5 provides longer-context and continuous-quantile output options [11]. Architectural diversity supplies potential ensemble diversity, but whether it produces useful complementarity must be determined from errors on the target data.

A directly related study is the dual-strategy demand-forecasting ensemble of Yang et al., which already combines semantic hierarchies and forecasts from different model backbones [12]. Hierarchical ensembling and combinations of heterogeneous foundation models are therefore not claimed as inventions of this work. The

contribution is a reproducible empirical evaluation oriented toward local deployment. Lightweight baselines and public foundation models are connected under common temporal splits, input restrictions, and output interfaces, then assessed through point forecasts, quantile calibration, and continuous inventory decisions. Both improvements and failure cases are relevant findings within this scope; the ensemble is not assumed to win every metric.

2.3 Censoring, Uncertainty, and Evaluation Design

Recorded sales differ from latent demand. FreshRetailNet-50K provides hourly stockout information that helps identify supply-constrained observations [13]. UCI Online Retail provides transaction quantities, dates, and cancellation indicators, but lacks inventory states that would directly support latent-demand recovery [14]. The datasets consequently support different conclusions: stockout sensitivity can be examined in the former, while the latter primarily supports observed-sales forecasting. Cleaning, zero filling, and sampling rules must be fixed before examining test outcomes and retained in an audit trail.

Quantile forecasts describe risk, but a quantile label does not establish calibration. Research on conformal prediction for time series shows that coverage under dependence requires additional conditions and dedicated methods [15]. Here, empirical residual calibration is treated as a postprocessing step that requires measurement. Pinball loss, coverage, and interval width are evaluated jointly. Protection-period uncertainty is constructed at the cumulative-demand level rather than treating a sum of daily high quantiles as the same quantile of cumulative demand.

Temporal splitting follows rolling-origin evaluation: each forecast uses only information available at its origin, and tuning and weight fitting precede an out-of-time test period [16]. Overlapping windows and repeated forecasts for one product are not independent observations. Block-bootstrap research motivates retaining temporal dependence [17,18], although the actual resampling unit must be specified and paired comparisons must preserve the same series and origins for both methods. Joint forecast and inventory evaluation distinguishes gains in accuracy, probabilistic calibration, and decision utility instead of collapsing them into a single claim.

3 Data, Task, and Study Design

3.1 Data Sources and the Meaning of Observations

Two public datasets represent different retail settings. FreshRetailNet-50K contains store-product-date records, hourly sales, and stockout indicators from fresh retail [13]. The dataset is pinned to revision 08c1fab7f9257bc73679d415d65d644165d351d4. Combining the official training and evaluation files yields 97 consecutive days. The released `sale_amount` field is globally scaled, and its published scale is retained. It is not interpreted as physical units, currency, or directly measurable business profit. A daily stockout indicator is true when at least one source hourly interval is flagged. It indicates some supply-constrained time within the day, not necessarily complete censoring of the entire day.

UCI Online Retail records transactions from a UK online retailer [14]. Of 541,909 original rows, 483,971 remain after requiring the United Kingdom, positive quantities and prices, valid product codes, and noncancelled invoices. Quantities are aggregated by product and calendar day, and dates without retained transactions are filled with zero, producing 374-day series. Because no unique invoice-line key is available, 5,268 completely identical rows are retained to avoid automatically deleting genuine repeated purchases. The target is gross completed-sales quantity, not quantity net of returns. A no-transaction day does not establish the absence of latent demand. UCI has no true stockout field: an all-false array in the common computational

interface is only a structural placeholder and is not evidence of continuous product availability.

Both datasets are released under CC BY 4.0. Source identities, versions, downloaded-file SHA256 values, and filtering procedures are retained. Customer identities and personal contact fields are not used in the analysis. The final UCI day, December 9, 2011, contains transactions only until 12:50. The main results retain the released observations, while a sensitivity slice removes forecast targets on that calendar day. Missing observations, zero sales, and stockouts are distinct concepts. No test-period information is used to impute latent demand.

Table 1. Data and out-of-time evaluation design

Item	FreshRetailNet sample	UCI Online Retail sample
Series unit	Store-product	Product in UK market
Series / cohorts	120 / 3 x 40	120 / 3 x 40
Observation dates	2024-03-28 to 2024-07-02	2010-12-01 to 2011-12-09
Calendar days	97	374
Sampling history prefix	First 41 days	First 200 days
Tuning / test origins	3 / 3	6 / 10
Test forecast windows	360	1,200
Forecast-observation pairs	5,040	16,800
Continuous inventory days	21	70
Numerical units	Released normalized sales	Completed-sales quantity

One forecast window is a 14-day forecast for one series at one origin. Forecast-observation pairs are not independent samples. Datasets are analyzed separately; raw sales are not pooled.

3.2 Prespecified Sampling and Demand Segments

The sample was selected to permit auditable local comparisons, not to favor test performance. Eligibility uses only the first 41 days of FreshRetail and the first 200 days of UCI, requiring finite, nonnegative observations and at least 8 nonzero days. The eligible populations contain 49,991 and 2,520 series, respectively. Series are ranked by their initial zero-sales proportion and divided into four strata. Within each stratum, SHA256 ordering of the series identifier and seeds 17, 29, and 43 selects 10 previously unselected series per seed. Each cohort contains 40 series, yielding 120 series per dataset. This deterministic procedure creates disjoint cohorts spanning relatively dense to relatively sparse products.

These three seeds control sample cohorts, not three neural-network training runs. Foundation-model weights remain fixed, with neither LoRA nor full-parameter fine-tuning. Cohort analyses are therefore not described as replication across training seeds. All 120 series in a dataset contribute to the global and segment-level loss estimates. Cohort results are subgroup robustness summaries within this shared fitting procedure, not three wholly independent training experiments.

The ensemble segments differ from the sampling strata. Segment labels are frozen before the first weight-fitting origin. Series with a zero proportion of at least 0.45 are classified as intermittent. Among the remainder, a series is trending when its absolute fitted linear slope multiplied by history length exceeds 0.35 times its mean sales. A remaining series is volatile if the squared coefficient of variation of its nonzero observations exceeds 0.7, and

regular otherwise. These are fixed operational rules, not a claimed replication of established ADI-CV-squared thresholds. Hierarchy ablations test whether sharing information through these segments is useful.

3.3 Temporal Separation and Forecast Targets

At origin o , a model can read only observations with indices strictly below o and predicts the 14 days from o through $o+13$. FreshRetail weight-fitting origins are 41, 48, and 55, and test origins are 69, 76, and 83. UCI weight-fitting origins are 248, 255, 262, 269, 276, and 283, and test origins are 297, 304, 311, 318, 325, 332, 339, 346, 353, and 360. Indices are zero-based. The final tuning target ends before the first test origin, preventing 14-day labels from crossing into the weight-selection stage. A 7-day stride within each stage allows adjacent target windows to overlap.

All methods obey the same univariate input restriction. Realized future prices, promotions, weather, and stockouts are unavailable to the forecasters. Calendar ridge regression uses only date features known in advance. As the test period progresses, sales already observed can enter subsequent forecast histories, but static ensemble weights remain frozen. The online variant updates only after the complete 14-day target of an earlier test forecast has matured. This sequential design permits newly observed history while preventing access to future labels.

The primary target is observed sales, not unobserved unconstrained demand. FreshRetail additionally reports conditional errors on target days without recorded stockout flags. This slice tests whether flagged supply constraints change the comparison, but may still reflect unobserved availability and selection effects and is not ground-truth latent demand. Both public datasets predate the pretrained checkpoints. Public documentation cannot completely exclude their inclusion in upstream pretraining. The out-of-time separation established here therefore applies to local weight fitting and evaluation; it does not establish complete independence from foundation-model pretraining data.

4 Ensembling and Inventory Decision Methods

4.1 A Common Interface for Heterogeneous Experts

Let $y(i,t)$ denote observed sales for series i at time t , and $q(m,i,o,h,\tau)$ the τ -quantile from expert m at origin o and horizon h . Every expert supplies 14-day forecasts for τ in $\{0.1, 0.5, 0.9\}$. Outputs are clipped to be nonnegative, followed by a cumulative maximum along the quantile axis to remove crossings. P50 supplies the main point forecast. A weighted average of corresponding quantiles combines quantile functions; it neither merges neural-network parameters nor necessarily equals the corresponding quantile of a mixture distribution. Coverage must still be assessed after combination.

Table 2. Forecasting experts and fixed configurations

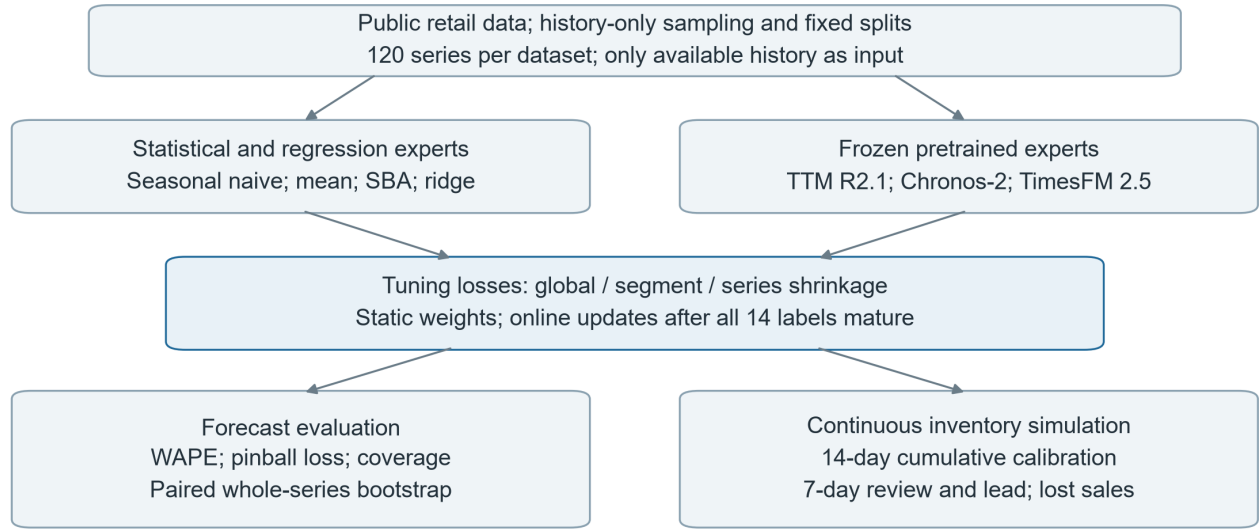
Expert	Point forecast or backbone	Quantiles and key settings
Seasonal naive	Repeat the last 7 days	Empirical quantiles of historical 7-day differences
28-day mean	Mean of the last 28 days	Historical rolling-mean residuals
Croston-SBA	Recursive nonzero size and interval	Alpha=0.15; SBA correction; historical one-step residuals
Calendar ridge	Trend and weekly/annual sine-cosine terms	Ridge penalty 8; at most 365 days; fitted residuals
TTM R2.1	Fixed 512-to-48 daily checkpoint	First 14 outputs; historical weekly-difference quantiles
Chronos-2	Fixed pretrained encoder	Native P10/P50/P90; context at most 512
TimesFM 2.5	Fixed 200M checkpoint	Continuous quantile head; maximum context 512

Quantiles describe the construction used here, not a claim of independently verified calibration. Residual-median shifts can make a baseline P50 differ from its original point forecast.

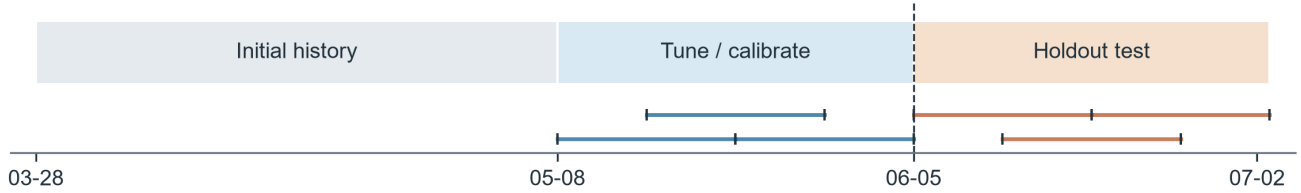
TTM uses daily frequency token 8. Histories shorter than 512 days are left-padded with their earliest observation. This adapter changes the input representation, especially for the short FreshRetail histories, so the results do not establish the performance ceiling of TTM with long histories or domain fine-tuning. TTM and the traditional baselines obtain quantiles by shifting point forecasts using historical residuals, which are not strictly independent multistep forecast errors. Chronos-2 disables cross-series learning and assigns each input to its own group. TimesFM enables input normalization, flip invariance, positivity inference, and quantile-crossing repair. All three models load pinned local weights and run in float32.

No new large backbone is trained. The fitted components are the ensemble weights and empirical cumulative-residual corrections used for inventory targets. This separation permits expert replacement, weight tracing, and resource measurement while keeping pretrained forecasting capabilities distinct from downstream replenishment rules. The engineering system supports an optimizer interface, but the experiments assess a specified periodic-review policy. Multiwarehouse, budget, and minimum-order-quantity optimization are not presented as completed experiments.

(a) Observed sales to forecasts, ensembles, and inventory



(b) FreshRetailNet: short lines denote 14-day windows



(c) UCI Online Retail: short lines denote 14-day windows



Figure 1. System architecture and temporally separated evaluation. (a) Seven experts output P10, P50, and P90 for a common horizon. Foundation weights are frozen, traditional forecasters are re-estimated at each origin, and the ensemble learns only from available labels. (b,c) Initial history, tuning/calibration, and the local out-of-time test are ordered chronologically. Each short line is a 14-day forecast, with origins 7 days apart. Holdout status refers to local weight fitting, not proven independence from upstream pretraining. Dataset timelines use separate scales. The main analysis shares tuning labels between weighting and calibration; a sensitivity analysis separates their target dates.

4.2 Normalized Loss and Hierarchical Shrinkage

Sales scales vary substantially across products. For each series, $a(i)$ is the mean sales before its first weight-fitting origin, floored at 10^{-10} , and remains fixed. An expert loss for one tuning window is the average pinball loss across three quantiles and 14 horizons, divided by $a(i)$. This scale uses no future actuals and prevents large-volume products from dominating weight fitting solely through their units. The aggregate normalized pinball score reported for testing instead divides total loss by total sales; these are deliberately distinct quantities.

$$\rho_\tau(u) = \max \{ \tau u, (\tau - 1)u \}, \quad \ell_{m,i,o} = \frac{\text{mean}_{h,\tau} \rho_\tau(y - q)}{a_i} \quad (1)$$

Let $L(m,i)$ be the average tuning-window loss for series i , and $G(m)$ the average across all series and tuning origins. For segment g , the mean loss $L(m,g)$ over its series-origin pairs is shrunk toward $G(m)$. If the segment contains $N(g)$ series and K tuning origins, its evidence count is $N(g)K$ and the segment shrinkage strength is fixed at 8. The series-level evidence count is K and its shrinkage strength toward the segment estimate is fixed at 4.

$$P_{m,g} = \frac{N_g K L_{m,g} + 8 G_m}{N_g K + 8} \quad (2)$$

$$L_{m,i}^{\text{adj}} = \frac{K L_{m,i} + 4 P_{m,g(i)}}{K + 4} \quad (3)$$

The adjusted losses are converted to exponential weights using temperature 7. Losses are normalized by the best adjusted loss for the series, reducing the weight of relatively worse experts. Any expert weight above 0.88 is capped, with the excess redistributed in proportion to the remaining expert weights. The cap retains diversity but does not guarantee better performance. Hyperparameters were fixed before test scoring and were not searched again to match the test ranking of either dataset.

$$w_{m,i} \propto \exp \left[-7 \left(\frac{L_{m,i}^{\text{adj}}}{\min_j L_{j,i}^{\text{adj}}} - 1 \right) \right], \quad \sum_m w_{m,i} = 1 \quad (4)$$

$$q_{i,o,h,\tau}^F = \sum_m w_{m,i} q_{m,i,o,h,\tau} \quad (5)$$

4.3 Combination Baselines, Ablations, and Causal Online Updates

Seven static combination strategies distinguish the availability of multiple experts from the benefit of sophisticated weighting: equal weighting; per-series selection of the lowest-loss tuning expert; global selection by mean normalized tuning loss; global inverse-loss weighting; full hierarchical ensembling; segment weighting without the SKU level; and shrinkage directly toward the global estimate without the segment level. Together with seven individual experts and one online hierarchical variant, there are 15 methods. Validation-based selection uses only tuning data. The best expert identified retrospectively on the test set describes a ranking, not an implementable decision with advance knowledge.

Online ensembling starts from the static hierarchical weights. At forecast origin o , it processes only previously unused test windows with origins u satisfying $u+14 \leq o$. Their fully matured normalized losses update the weights multiplicatively with learning rate 0.25, followed by renormalization and the 0.88 cap. Each window is processed once. FreshRetail has a short test period, and only its final origin can use the first complete test window. The opportunity for online adaptation is consequently limited; the presence of an online mechanism alone does not establish its usefulness.

$$w_{m,i}^{\text{new}} \propto w_{m,i}^{\text{old}} \exp \{ -0.25 \ell_{m,i,u} \}, \quad u + 14 \leq o \quad (6)$$

4.4 Cumulative Demand Targets over the Protection Period

With review interval $R=7$ days and fixed lead time $L=7$ days, the protection period is $H=R+L=14$ days. Summing daily P90 forecasts generally does not produce the P90 of 14-day cumulative demand because cross-day dependence affects the sum distribution. The main policy therefore uses cumulative residuals, with the sum of daily P90 retained as a transparent comparator. For each tuning window, the sum of median forecasts is subtracted from cumulative observed sales and divided by $b_{i,o}$, defined as 14 times the preceding 28-day mean sales. The pooled normalized residuals yield an empirical 0.90 quantile c using the higher interpolation rule. The test order-up-to target is the cumulative median plus c times the current history scale, clipped to be nonnegative.

$$r_{i,o} = \frac{\sum_{h=1}^{14} y_{i,o+h-1} - \sum_{h=1}^{14} q_{i,o,h,0.5}^F}{b_{i,o}} \quad (7)$$

$$S_{i,o} = \max \left\{ 0, \sum_{h=1}^{14} q_{i,o,h,0.5}^F + Q_{0.90}(r_{\text{tune}})b_{i,o} \right\} \quad (8)$$

Each forecasting method receives its own correction estimated solely from tuning labels. In the main analysis, weight estimation and residual correction share the same tuning period. Residuals also have overlapping targets and cross-product dependence. The correction is thus empirical risk calibration, not a finite-sample conformal coverage guarantee established by this study. A strictly separated sensitivity analysis uses FreshRetail origin 41 for weights and 55 for calibration, and the first four UCI tuning origins for weights and origin 283 for calibration. Weight-fitting and calibration target dates do not overlap in either case, and evaluation uses the original test period.

4.5 Continuous Inventory Simulation and Cost Definition

Observed sales are replayed as a common exogenous demand sequence. Quantities are continuous to accommodate the normalized FreshRetail scale. Every method starts with on-hand inventory equal to 7 times the mean sales in the 28 days preceding testing and an empty pipeline. Each calendar day processes arrivals, inventory-position calculation and ordering on review days, demand fulfillment, and end-of-day holding in that order. Orders arrive exactly 7 days later. Unfulfilled demand is lost rather than backlogged. On-hand and pipeline inventory carry across review cycles, avoiding the artificial reset of each forecast window into a separate inventory experiment.

$$IP_{i,t} = I_{i,t} + O_{i,t}, \quad Q_{i,t} = \max \{0, S_{i,t} - IP_{i,t}\} \quad (9)$$

$$F_{i,t} = \min \{I_{i,t}^{\text{after arrival}}, y_{i,t}\}, \quad U_{i,t} = y_{i,t} - F_{i,t} \quad (10)$$

$$C = c_h \sum_{i,t} I_{i,t}^{\text{end}} + c_s \sum_{i,t} U_{i,t} \quad (11)$$

Here O is on-order inventory, F is fulfilled quantity, and U is lost sales. The unit lost-sales penalty is fixed at $c_s=1$, while daily unit holding cost c_h takes values 0.005, 0.02, and 0.05. These scenarios change cost evaluation without reoptimizing inventory targets or service levels. They represent cost sensitivity, not cost-optimal policies. Total cost divided by total replayed demand gives cost per observed demand unit,

interpreted separately for each dataset. Fill rate is total fulfilled quantity divided by total demand, not the proportion of in-stock days. Stockout series-days, mean total end-of-day inventory, and terminal pipeline inventory are also recorded.

Only the first 7 days following each test origin advance the inventory state, yielding continuous simulations of 21 days for FreshRetail and 70 days for UCI. Orders from the final review may remain in the terminal pipeline. They are reported without inventing future demand or salvage values. The main cost excludes purchasing, fixed ordering, transport, expiration, capacity, and budget constraints, as well as terminal disposal penalties. The simulation can therefore compare methods under a common simplified mechanism but cannot establish actual business return on investment or complete supply-chain optimization. The common initial state may materially affect absolute service levels over short horizons.

5 Metrics, Statistical Inference, and Implementation

5.1 Evaluation Metrics and Units of Comparison

The primary point-forecast metric is WAPE, computed as total absolute P50 error across all test series-origin-horizon positions divided by total observed sales. MAE, RMSE, and signed bias are also reported. Bias is forecast minus actual, with positive values indicating aggregate overprediction. The probabilistic score is total pinball loss averaged over the three quantiles, divided by total actual sales. It is not multiplied by 2, so comparisons with twice-pinball conventions require a conversion. Denominators and evaluation slices are retained with each result, preventing an arithmetic mean of SKU-level ratios from being mistaken for aggregate WAPE.

$$\text{WAPE} = \frac{\sum |q_{0.5} - y|}{\sum y}, \quad \text{Bias} = \frac{\sum (q_{0.5} - y)}{\sum y} \quad (12)$$

$$\text{nPB} = \frac{\sum \text{mean}_{\tau \in \{0.1, 0.5, 0.9\}} \rho_{\tau}(y - q_{\tau})}{\sum y} \quad (13)$$

Probabilistic diagnostics include empirical P10, P50, and P90 coverage and the coverage and width of the [P10,P90] interval. Quantile coverage is defined as the proportion satisfying $y \leq q_{\tau}$. With discrete sales and a point mass at zero, it need not equal τ exactly. Deviations can indicate conservative or insufficient forecasts, but their magnitude alone does not prove that every discrepancy is model misspecification. Cumulative-target coverage is evaluated separately rather than substituting daily coverage for protection-period risk.

5.2 Paired Whole-Series Cluster Bootstrap

Differences between the focal ensembles and comparators are evaluated with 2,000 paired whole-series cluster bootstrap resamples using seed 20260926. Each resample draws 120 complete series with replacement, preserving all test origins, 14-day targets, and the paired forecasts of the two methods. Aggregate WAPE is then recomputed. Absolute improvement is comparator WAPE minus ensemble WAPE, so positive values favor the ensemble. Relative improvement divides this difference by comparator WAPE. The 95% interval is the 2.5th to 97.5th percentile of the resampled statistic. Inventory comparisons similarly resample complete series-level cumulative costs.

This design preserves dependence among overlapping forecast windows within a series and does not treat the 21,840 forecast positions as independent samples. It still assumes approximate exchangeability across series clusters and does not add a second clustering level for shared store shocks or calendar blocks. Intervals are

conditional evidence for the sampled series and dates. Equal weighting and global validation-based selection are the principal comparators. Other comparisons are descriptive diagnostics, and intervals are unadjusted for multiplicity. An exploratory interval excluding zero is not treated as proof of a universal advantage.

5.3 Local Environment, Execution Audit, and Computational Cost

Data preparation, model inference, weight estimation, bootstrap resampling, and inventory simulation ran in the author's local Windows environment. The recorded GPU is an NVIDIA GeForce RTX 3070 Ti Laptop GPU, with PyTorch 2.10.0+cu130 and CUDA runtime 13.0. Batch size is 16 and CPU thread count is 8. Foundation models are loaded and released sequentially, so all networks need not reside in GPU memory simultaneously. Model inference uses locally stored weights, public data, and Python libraries without remote commercial inference services.

Before scoring, the runner records data hashes, forecast origins, model identities, floating-point precision, context settings, and script hashes. Caches retain complete expert quantile tensors. The analysis entry point checks IDs, dates, sample cohorts, actual labels, and stockout arrays against the fixed input data. Scientific unit tests cover temporal boundaries, quantile monotonicity, weight sums and caps, online label maturity, arrival lead times, inventory conservation, cumulative-quantile comparisons, and paired-bootstrap identity. Timing records separate loading from batch inference. GPU memory is the PyTorch peak allocated amount, not whole-system GPU usage or a strict peak-resident-memory benchmark.

Table 3. Local batch-inference time and GPU memory

Model	FreshRetail (s)	UCI (s)	Peak GPU (MiB)
Seasonal naive	0.101	0.369	0.0
28-day mean	0.148	1.372	0.0
Croston-SBA	0.416	5.523	0.0
Calendar ridge	0.143	0.525	0.0
TTM R2.1	0.645	0.747	13.8
Chronos-2	1.571	2.890	482.9
TimesFM 2.5	113.667	313.311	898.3

The datasets contain 720 and 1,920 tuning-plus-test windows, respectively. Model loading is excluded. CPU baselines use zero GPU memory. Single-run measurements from different implementations are not a standardized cross-hardware throughput benchmark.

6 Results

6.1 Overall Forecast Performance on the Two Datasets

Table 4. FreshRetail test forecast results

Method	WAPE	nPB	Bias	P90 coverage
Seasonal naive	43.63%	0.1414	-2.92%	86.49%
28-day mean	38.58%	0.1277	-5.97%	83.12%
Croston-SBA	36.85%	0.1218	-5.71%	84.88%
Calendar ridge	38.57%	0.1258	-6.94%	82.58%
TTM R2.1	39.30%	0.1284	-17.23%	83.35%
Chronos-2	37.27%	0.1208	-6.89%	86.27%
TimesFM 2.5	36.98%	0.1210	-8.95%	85.30%
Equal weight	36.51%	0.1189	-7.80%	85.73%
Per-series selection	37.80%	0.1237	-6.27%	84.40%
Global selection	36.98%	0.1210	-8.95%	85.30%
Global inverse loss	36.50%	0.1189	-7.88%	85.63%
Hierarchical	36.66%	0.1193	-7.58%	85.30%
No SKU level	36.57%	0.1192	-8.02%	85.18%
No segment level	36.67%	0.1193	-7.56%	85.32%
Online hierarchical	36.66%	0.1193	-7.58%	85.30%

Lower WAPE and nPB are better. Static weights are frozen from tuning; online weights use only matured labels. All methods share the same test sample.

Table 5. UCI Online Retail test forecast results

Method	WAPE	nPB	Bias	P90 coverage
Seasonal naive	130.75%	0.5437	2.01%	90.11%
28-day mean	102.08%	0.3954	-35.27%	89.43%
Croston-SBA	103.40%	0.4008	-33.46%	89.64%
Calendar ridge	94.81%	0.3622	-45.98%	89.20%
TTM R2.1	92.04%	0.3671	-54.92%	88.26%
Chronos-2	85.76%	0.3241	-63.59%	91.67%
TimesFM 2.5	87.42%	0.3345	-67.84%	89.30%
Equal weight	93.81%	0.3662	-42.72%	90.24%
Per-series selection	87.87%	0.3338	-57.32%	90.57%
Global selection	85.76%	0.3241	-63.59%	91.67%
Global inverse loss	92.52%	0.3592	-45.63%	90.26%
Hierarchical	89.02%	0.3390	-55.33%	90.54%
No SKU level	88.59%	0.3378	-57.05%	90.73%
No segment level	89.03%	0.3389	-55.40%	90.54%
Online hierarchical	88.47%	0.3359	-56.68%	90.62%

Lower WAPE and nPB are better. Static weights are frozen from tuning; online weights use only matured labels. All methods share the same test sample.

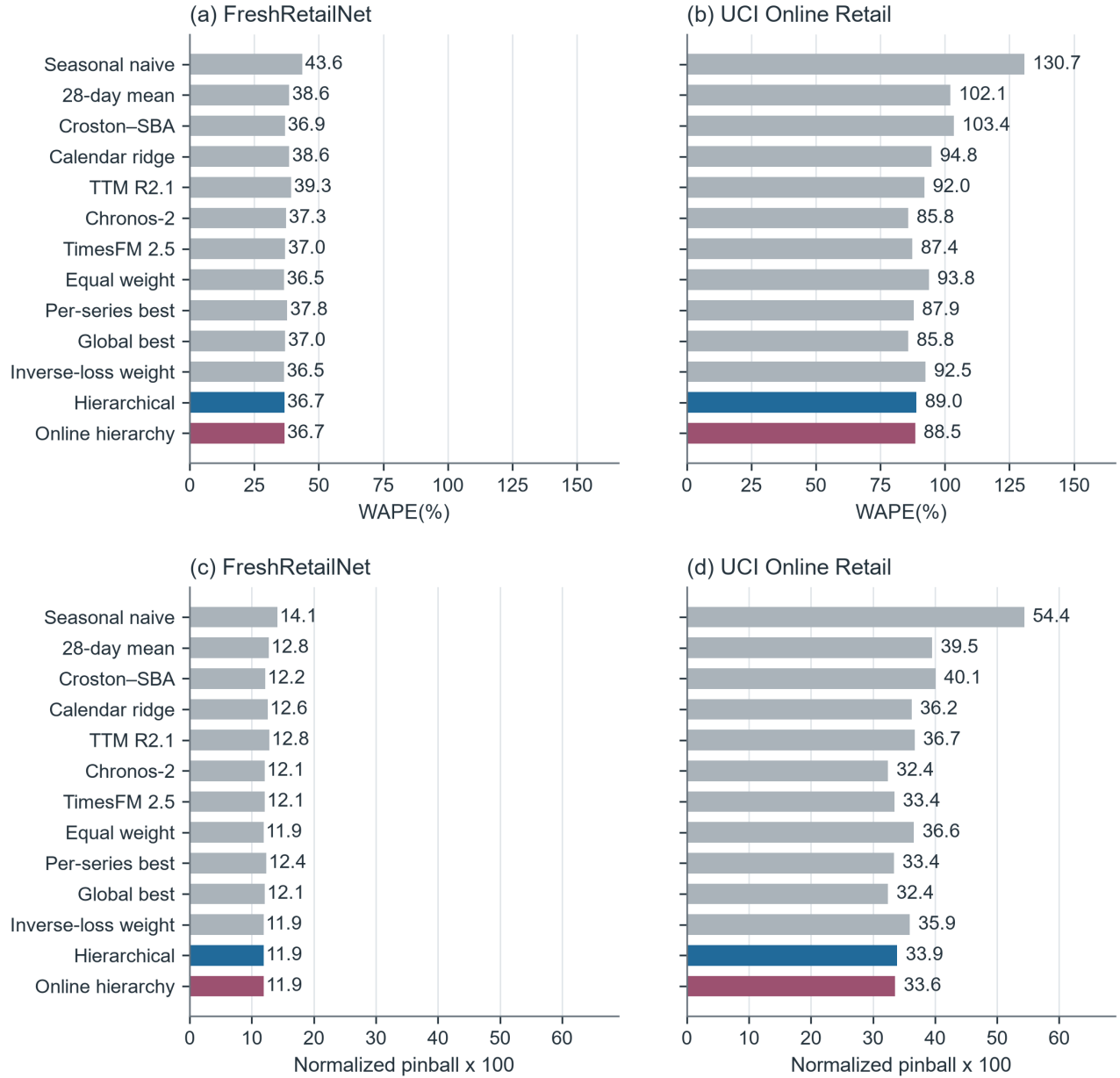


Figure 2. Forecast performance on the two test datasets. Methods share series, origins, a 14-day horizon, and history-only inputs. WAPE divides total absolute error by total observed sales. Normalized pinball loss divides total loss averaged across three quantiles by total sales, without a factor of 2. Both scores are lower-is-better and shown multiplied by 100 on zero-based axes. Blue denotes the frozen hierarchy and purple the online variant. Complete methods and ablations appear in the result tables.

6.2 Evaluation Scale and Main Findings

The study includes 120 series from FreshRetailNet-50K and 120 from UCI Online Retail. Each dataset has three disjoint 40-series sampling cohorts, used for sample sensitivity rather than neural-network training replication. With a 14-day horizon and 3 or 10 test origins, the evaluation comprises 21,840 forecast-observation pairs corresponding to 12,600 distinct series-date observations. Adjacent windows overlap. Paired 95% confidence intervals therefore use 2,000 resamples of complete series clusters rather than treating forecast-observation pairs as independent.

Ensemble performance depends on both the dataset and the comparator. Hierarchical ensembling improves on equal weighting for UCI but is worse than Chronos-2, which was selected during tuning; the paired intervals

support these directions. For FreshRetail, its intervals relative to equal weighting and the preselected TimesFM both include zero. Neither the SKU level nor the segment level provides a consistent advantage. The experiments establish an operational ensemble with conditional benefits, not general superiority over individual models or simple combinations.

6.3 Point Forecasts and Ablations

On FreshRetail, hierarchical WAPE is 0.366619, compared with 0.365060 for equal weighting and 0.365010 for global inverse-loss weighting. Individual experts achieve 0.368516 for Croston-SBA, 0.369837 for TimesFM, 0.372662 for Chronos-2, and 0.393027 for TTM. Relative improvement over equal weighting is -0.427%, with a 95% interval of [-1.249%, 0.304%]. Relative improvement over preselected TimesFM is 0.870%, with interval [-0.719%, 2.561%]. Both intervals include zero. Per-SKU validation selection yields WAPE 0.378017; hierarchical ensembling improves on it by 3.015%, with interval [1.202%, 4.950%]. This result is consistent with unstable selection of a single expert from short histories.

On UCI, hierarchical WAPE is 0.890155 versus 0.938138 for equal weighting, a relative improvement of 5.115% with interval [3.416%, 7.186%]. However, preselected Chronos-2 achieves 0.857570. The hierarchy is 3.800% worse relative to Chronos-2, with an improvement interval of [-6.210%, -2.022%]. Chronos-2 also has lower normalized pinball loss, 0.324053 versus 0.338985. Removing the SKU level produces WAPE values of 0.365738 and 0.885871, both slightly below the full hierarchy. Removing the segment level gives 0.366699 and 0.890297, close to the full method. These results do not provide sufficient evidence that either additional level improves forecasting.

Online updates lower UCI WAPE to 0.884689, a 0.614% relative improvement over the frozen hierarchy with interval [0.195%, 1.170%], while remaining worse than Chronos-2. FreshRetail online WAPE is 0.366645 and does not improve the frozen method. Only its final forecast can incorporate one fully matured window, so this short experiment cannot determine the potential of longer-term adaptation.

6.4 Probabilistic Forecasts and Cumulative Calibration

FreshRetail hierarchical P90 coverage is 85.298%, and central 80% interval coverage is 74.603%, indicating undercoverage in these diagnostics. UCI P90 coverage is 90.536%, but central 80% interval coverage is only 54.720%, compared with 80.732% for Chronos-2. Examining the upper quantile alone therefore misses a lower-bound failure. In UCI, 68.518% of test observations are zero, and the ensemble's positive lower bound excludes zeros accounting for 35.030% of all test observations. Linear quantile averaging does not automatically preserve the experts' point masses at zero, motivating specific treatment of discrete intermittent demand.

After correction by the empirical 90th percentile of tuning-period cumulative residuals, hierarchical 14-day cumulative coverage is 90.278% on FreshRetail and 91.583% on UCI. Direct sums of daily P90 forecasts instead give 95.278% and 84.750%. These opposite directions of deviation further demonstrate why daily quantile sums cannot be interpreted automatically as the same cumulative-demand quantile.

Strictly separating weight fitting from residual calibration and removing target-date overlap gives cumulative coverage of 87.500% and 93.750%. The UCI mean inventory target increases from 378.731 to 561.239, revealing sensitivity to short calibration samples. These are empirical corrections, without an established finite-sample coverage guarantee.

6.5 Continuous Inventory Replay and Cost Sensitivity

Inventory replay uses a 7-day review interval, a 7-day lead time, lost sales, and continuous state carryover over 21 and 70 days. All methods start from the same 7 days of stock based on the preceding 28-day mean and an empty pipeline. Daily processing consists of arrivals, ordering, sales fulfillment, and end-of-day holding. Total cost is holding cost plus a lost-sales penalty. It is a scenario cost, not realized monetary profit.

At holding rate 0.02 and unit lost-sales penalty 1, FreshRetail hierarchical cumulative-residual cost is 565.638 versus 572.355 for equal weighting, a reduction of 1.174% with interval [0.633%, 1.685%]. Fill rate also falls from 95.317% to 95.262%, so this is not a cost reduction at matched service. UCI costs are 60,223.066 and 57,195.213, meaning the hierarchy costs 5.294% more than equal weighting. Relative to Chronos-2 cost of 65,431.165, it costs 7.960% less, with interval [7.071%, 8.943%], while fill rate falls from 94.986% to 94.935%. Forecast-error rankings and inventory-cost rankings differ.

At holding rates 0.005 and 0.05, hierarchical cost relative to equal weighting falls by 0.210% and 1.556% on FreshRetail but rises by 4.609% and 5.454% on UCI. For UCI, cumulative-residual correction raises fill rate from 90.946% under the daily-P90-sum rule to 94.935%, while increasing cost from 23,955.939 to 60,223.066. Higher coverage must therefore be reported alongside the additional inventory cost.

FreshRetail online and frozen policies have identical realized costs. The online update occurs on the final review date, and its order arrives just beyond the scoring period, changing only terminal pipeline inventory. Both terminal on-hand and pipeline quantities are retained and reported rather than reset to zero or omitted.

6.6 Robustness, Scope, and Execution Checks

Hierarchical ensembling does not outperform equal weighting in any of the three FreshRetail cohorts. In all three UCI cohorts, it outperforms equal weighting but remains worse than Chronos-2. Excluding the incomplete final UCI day yields WAPE values of 0.889464, 0.937230, and 0.857094 for hierarchical, equal-weight, and Chronos-2 forecasts, preserving the principal ordering. Restricting FreshRetail targets to days without recorded stockout flags gives hierarchical and equal-weight WAPE values of 0.346835 and 0.344461, again preserving the direction.

Raw predictions were checked against source IDs, dates, cohorts, actual observations, and stockout flags. Tuning targets end no later than the first test origin. The maximum series-level inventory-flow conservation residual is below 9×10^{-14} for FreshRetail and 2×10^{-12} for UCI, and independent cost-recalculation error is below 4×10^{-12} . Eight analysis unit tests pass. Hashes are retained for input data, predictions, and analysis scripts.

The simulation replays observed sales and does not recover latent demand during stockouts. Budget, capacity, minimum order quantities, and actual procurement costs are not evaluated. Confidence intervals cluster by series without separately modeling shared store shocks, and multiple comparisons are unadjusted. Public data predate the model checkpoints, so upstream pretraining exposure cannot be excluded. Temporal separation ensures that local ensemble fitting does not use future test labels, not that the foundation models never encountered these public histories.

6.7 Paired Differences, Calibration, and Mechanism Displays

Table 6. Relative WAPE improvement of hierarchical ensembling

Dataset	Comparator	Relative gain	Paired-cluster 95% interval
FreshRetail	Seasonal naive	15.97%	[13.27%, 18.71%]
FreshRetail	Chronos-2	1.62%	[-0.07%, 3.45%]
FreshRetail	Equal weight	-0.43%	[-1.25%, 0.30%]
FreshRetail	Global selection	0.87%	[-0.72%, 2.56%]
UCI	Seasonal naive	31.92%	[26.79%, 36.39%]
UCI	Chronos-2	-3.80%	[-6.21%, -2.02%]
UCI	Equal weight	5.11%	[3.42%, 7.19%]
UCI	Global selection	-3.80%	[-6.21%, -2.02%]

Positive values favor hierarchical ensembling. Intervals are unadjusted for multiplicity. Equal weighting and global validation selection are the principal comparators.

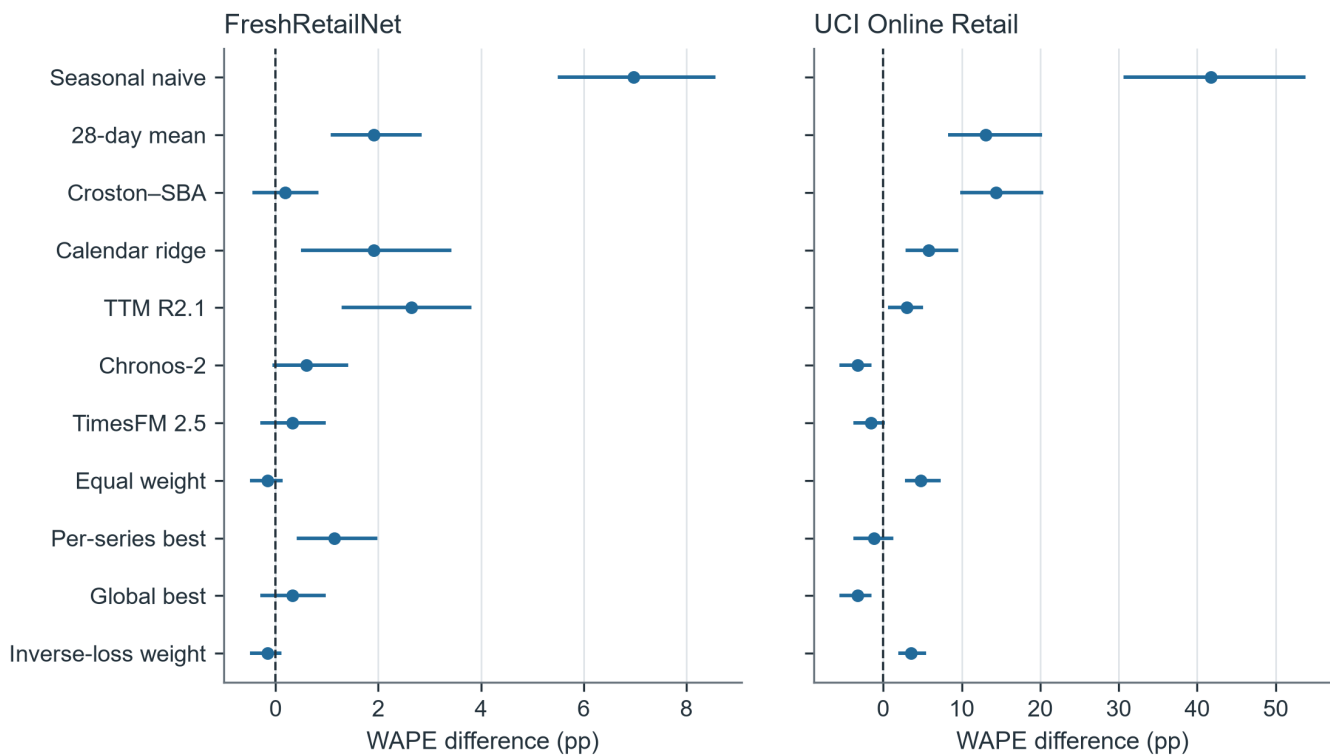


Figure 3. Paired differences and uncertainty relative to hierarchical ensembling. Points show comparator WAPE minus hierarchical WAPE; positive values favor the hierarchy. Segments are unadjusted 95% percentile intervals from 2,000 paired whole-series cluster bootstrap resamples. All overlapping windows of a sampled SKU-location series stay together. Shared store and calendar shocks are not separately clustered. Panels use different horizontal ranges to reveal small FreshRetail differences; both use percentage points. An interval crossing zero is not evidence of equivalence.

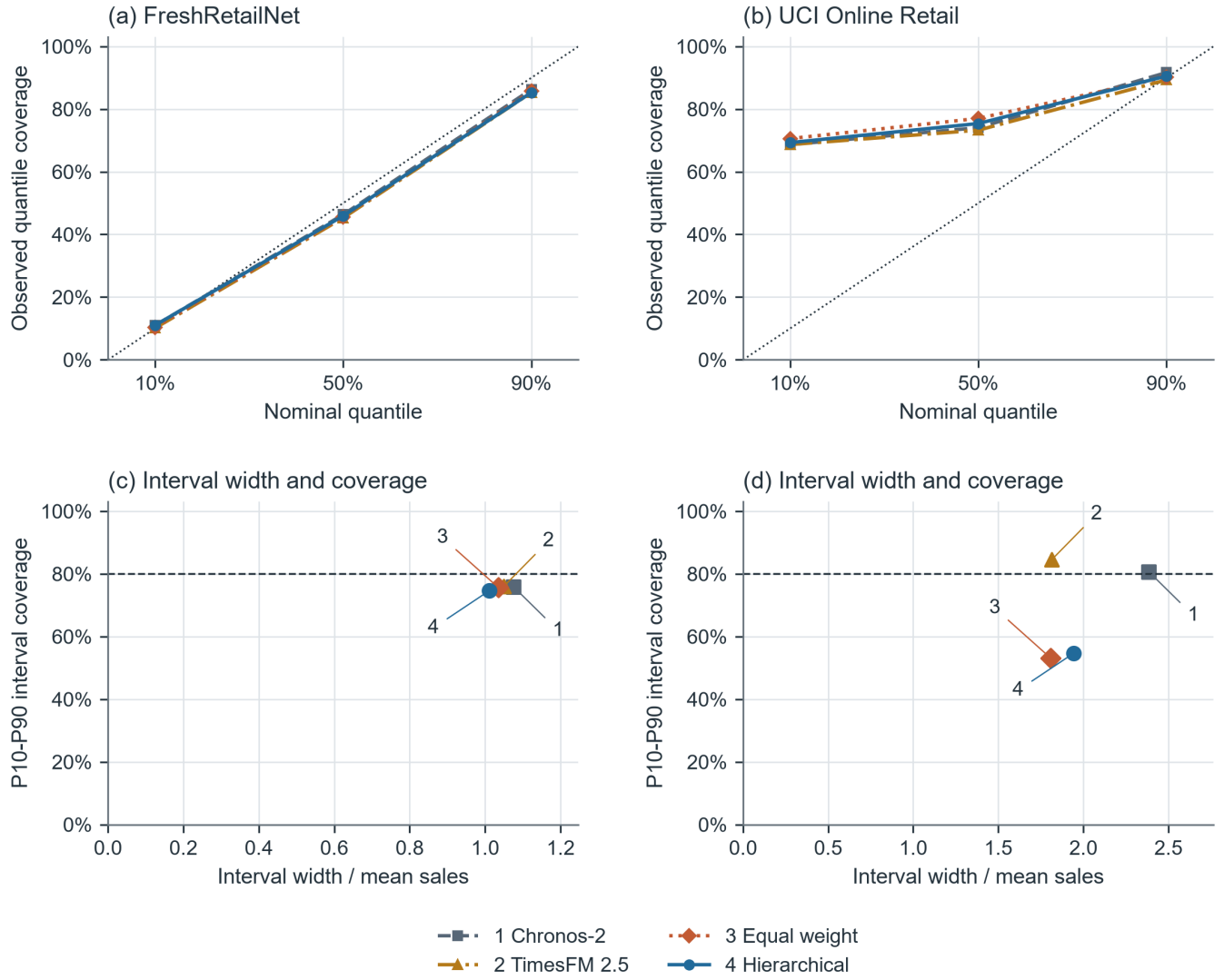


Figure 4. Daily quantile coverage and interval width. The top row shows the proportion of observed sales at or below P10, P50, and P90; the diagonal is a nominal reference. The bottom row compares central P10-P90 coverage and normalized width, with an 80% reference line. Numbers identify methods in the legend, selected by their roles as foundation models or simple/hierarchical ensembles. UCI hierarchical P90 coverage is 90.54%, but central coverage is 54.72%. Positive lower bounds exclude zero sales accounting for 35.03% of all observations. Near-nominal one-sided coverage does not imply a reliable two-sided interval. Normalized width is total interval width divided by total sales. Bottom panels use separate horizontal ranges.

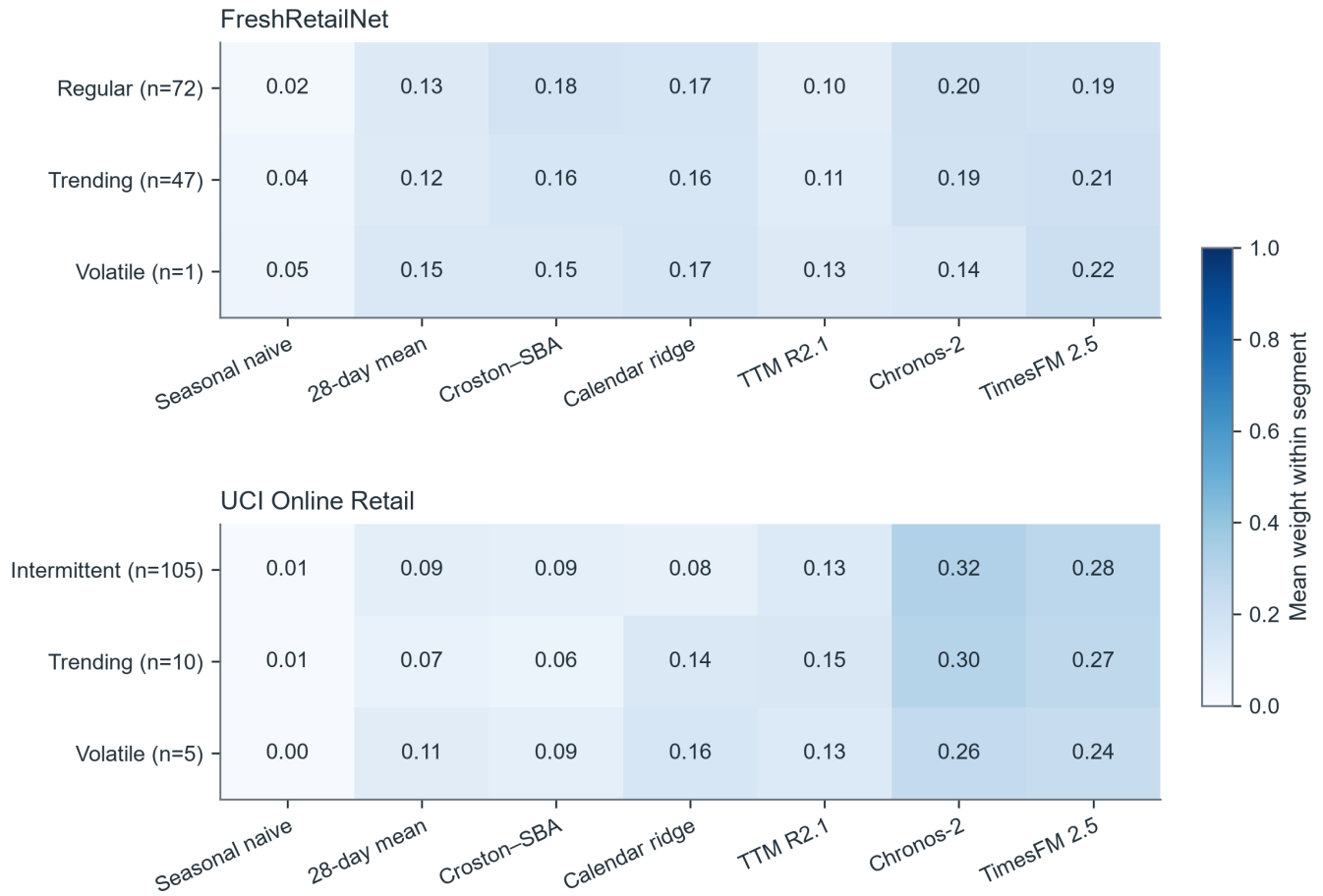


Figure 5. Expert weights learned from historical tuning data. Cells show mean frozen hierarchical expert weights within each history-defined demand segment. Rows sum to 1 and n is the number of series in a segment. These are descriptive average weights, not causal importance or segment contributions. Both datasets use the same 0-1 color scale. Absent segments are not filled with invented zero weights.

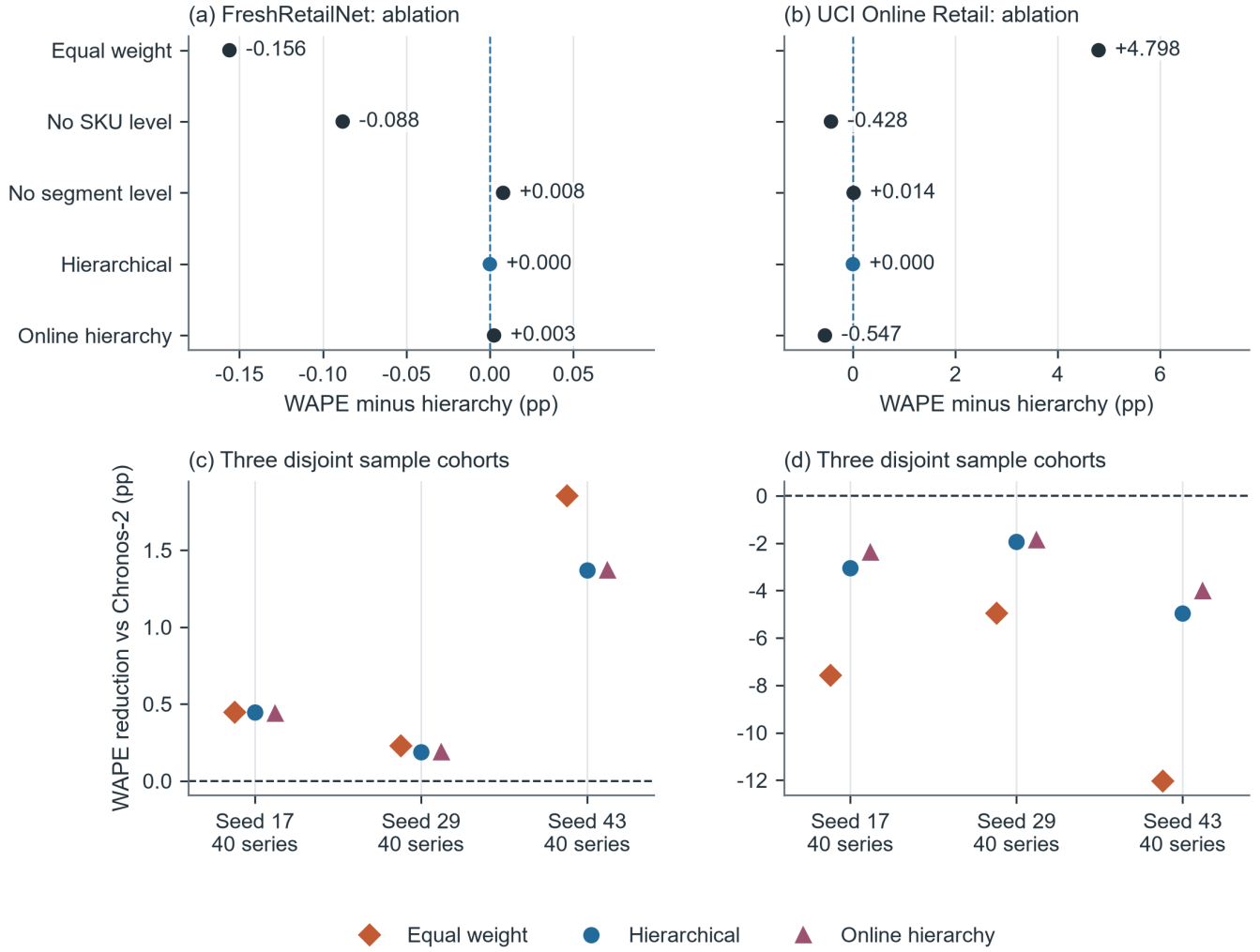


Figure 6. Ensemble ablations and sampling-cohort robustness. Top panels show method WAPE minus full hierarchical WAPE, so negative values favor the alternative. All methods reuse the same expert predictions without backbone retraining. Bottom panels show three disjoint deterministic 40-series cohorts. Seeds 17, 29, and 43 control sampling, not repeated neural-network training. Positive bottom-panel differences indicate lower error than Chronos-2. Panels use separate scales to display small ablation effects and cohort variation. Cohort points are descriptive, without confidence intervals, and do not count as additional independent training experiments.

6.8 Inventory Tradeoffs and Forecast Examples

Table 7. Continuous inventory simulation with cumulative-residual targets

Dataset	Method	Fill rate	Mean total EOD stock	Cost / demand
FreshRetail	Seasonal naive	95.15%	1017.39	0.1977
FreshRetail	Chronos-2	95.38%	1141.79	0.2136
FreshRetail	Equal weight	95.32%	1043.33	0.1998
FreshRetail	Global selection	95.38%	1053.85	0.2007
FreshRetail	Hierarchical	95.26%	1023.63	0.1975
FreshRetail	Online hierarchical	95.26%	1023.63	0.1975
UCI	Seasonal naive	95.03%	51630.68	1.4560
UCI	Chronos-2	94.99%	44895.89	1.2730
UCI	Equal weight	94.92%	38986.94	1.1128
UCI	Global selection	94.99%	44895.89	1.2730
UCI	Hierarchical	94.93%	41156.87	1.1717
UCI	Online hierarchical	94.93%	41006.90	1.1676

Daily holding cost is 0.02 and unit lost-sales cost is 1. The datasets have different units, so absolute stock levels are not comparable across datasets. Fill rate refers to replayed observed sales, not fulfillment of unobserved real-world demand.

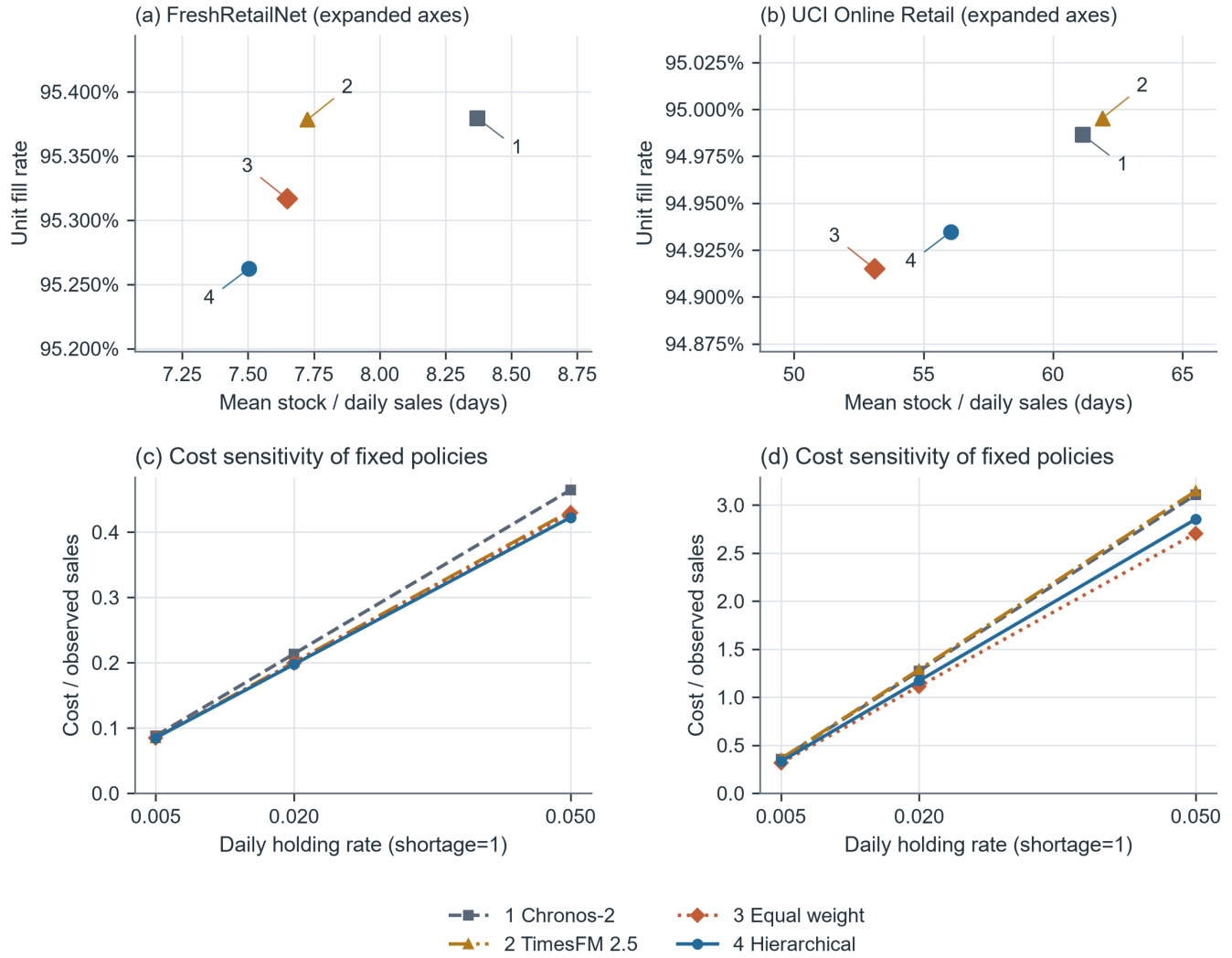


Figure 7. Service, holdings, and cost sensitivity in continuous inventory simulation. A 7-day review interval and a 7-day lead time govern a lost-sales simulation with continuous on-hand and pipeline states and common initial stock. Targets use empirical 90% calibration of 14-day cumulative errors. Top panels express mean holdings as days of observed average sales; numbers match the legend. Both axes are locally expanded, separately by dataset, to display nearby service levels. Bottom panels vary holding rates along the same policy trajectories rather than reoptimizing policies; cost axes also use dataset-specific ranges. Cost per total observed sales is not realized monetary profit. FreshRetail sales are globally normalized; UCI sales are quantities.

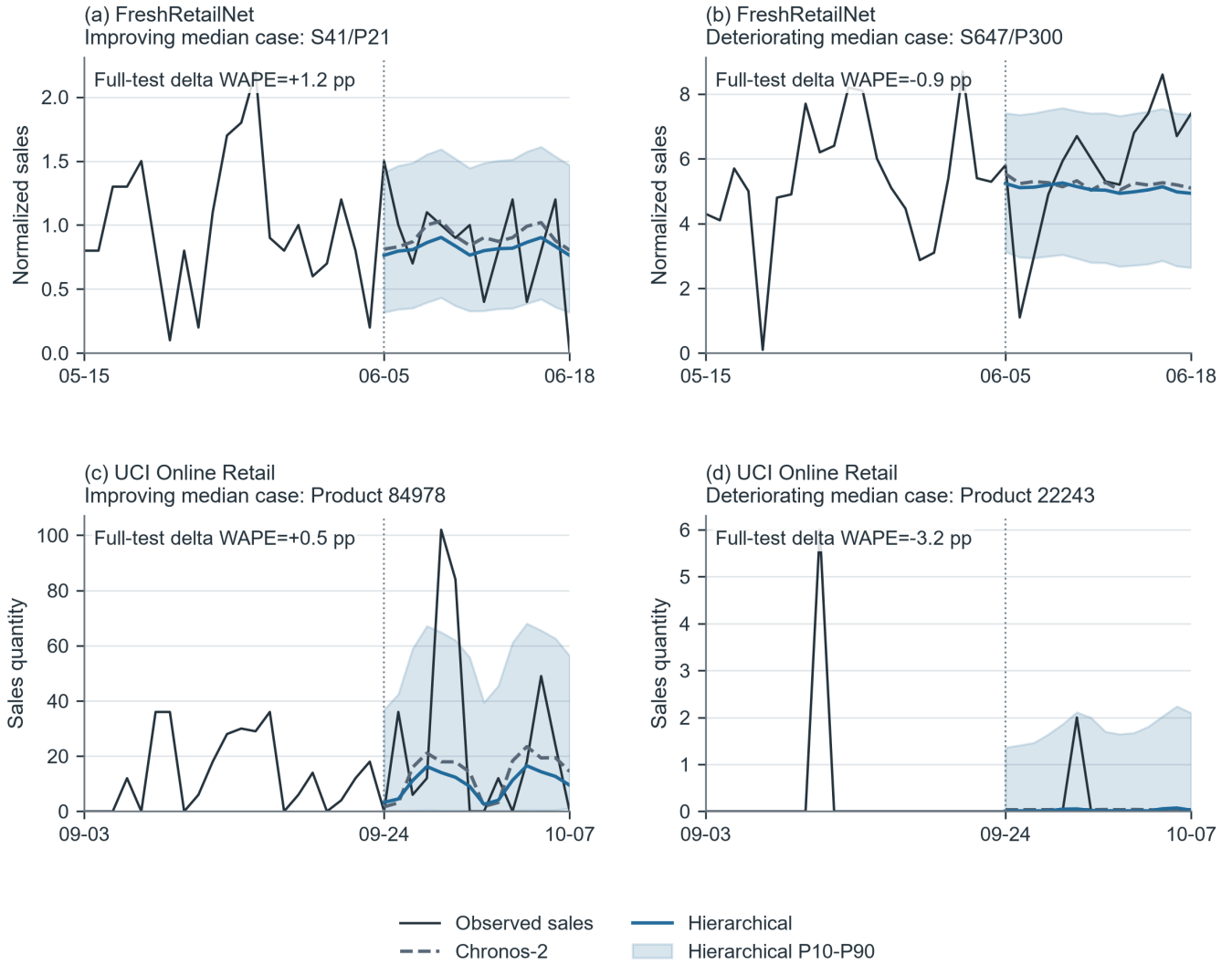


Figure 8. Improving and deteriorating cases selected by a fixed rule. For each dataset, the full-test per-series difference is Chronos-2 WAPE minus hierarchical WAPE. Positive and negative groups are separately sorted by difference and ID, and the middle-ranked series is selected. Each panel shows its first test origin and preceding 21 days. The displayed delta refers to the complete test period, not just the illustrated window. The vertical line marks the origin and the blue band is the hierarchical P10-P90 interval. Vertical scales differ across panels. These outcome-selected cases illustrate behavior and do not replace aggregate evidence or establish overall superiority.

7 Discussion

7.1 Inherited Forecasting Capabilities and Additional Complexity

The results support two separate interpretations. Combining complementary experts can retain useful forecasting structures within a common probabilistic interface without retraining the backbones. Adding hierarchical weights, series-specific adaptation, or online updates, however, does not guarantee improvement over equal weighting or validation selection. If averaging already offsets some biases, finely estimated weights from a short tuning period may add variance. Segment-level error patterns in short histories may also fail to persist. The evidence supports selecting the useful degree of complexity from actual comparisons rather than treating ensembling itself as a performance promise.

Hierarchical shrinkage aims to limit domination by a small number of local errors, but its benefit depends on whether series in the same segment genuinely share expert rankings and whether tuning-period advantages persist into testing. The no-segment ablation provides indirect evidence about the first condition, while rolling origins and online updates provide evidence about the second. Similar errors with intervals crossing zero indicate insufficient evidence for a clear advantage, not proof of equivalence. Equivalence or noninferiority testing would require prespecified business-relevant margins; post hoc thresholds are not used here.

7.2 Why Lower Point-Forecast Error Need Not Lower Inventory Cost

WAPE is dominated more strongly by high-volume products and penalizes overprediction and underprediction symmetrically through absolute error. Inventory policies impose asymmetric consequences on shortages and excess stock and are constrained by lead times, pipeline orders, and existing inventory. Lower P50 error can coexist with higher stock if P90 forecasts or cumulative correction are too conservative. Conversely, lower cost may reflect reduced service. Table 7 and the inventory figure consequently show fulfillment, holdings, and cost together rather than selecting one as evidence of overall improvement.

Summing daily high quantiles often yields a large protection-period target, but it is neither necessarily an unacceptable business rule nor automatically a valid quantile of cumulative demand. Cumulative-residual correction uses historical joint errors but depends on the transferability of calibration-period errors to testing. Strict separation reduces label reuse while leaving fewer observations for both weight and residual estimation, potentially increasing uncertainty. Its results expose a limited-data tradeoff. A single coverage estimate near 90% does not establish a general probabilistic guarantee.

7.3 Commercial Deployment, Version Pinning, and Maintenance

The model cards for the pinned IBM Granite TTM, Amazon Chronos-2, and Google TimesFM 2.5 weights identify Apache 2.0 licensing. Both datasets identify CC BY 4.0 licensing. These provide a basis for local modification and commercial use subject to the respective terms, not permission to remove copyright notices, NOTICE files, provenance, or required attribution. Later releases under the same model brand must not be assumed to have identical licenses. Appendix A records exact model IDs and revisions for version auditing. Model licenses, data licenses, dependency licenses, and rights in an enterprise's own data remain distinct considerations.

A deployment should retain lightweight baselines as interpretable fallbacks and separate quantile forecasting from order calculation. Daily jobs can cache expert outputs and update histories and matured error states as new observations arrive. Forecast failures should be logged and handled through a prespecified fallback, never by replacing predictions with future actuals. Procurement execution additionally requires enterprise lead-time distributions, pipeline reliability, pack multiples, budgets, shelf life, and human approval procedures. These are

deployment requirements, not evidence of field-validated business benefits in the present study.

7.4 External Validity and Scope

The public-data samples have a defined scope: 120 series per dataset, excluding products with extremely sparse initial histories. They do not represent the complete long tail, cold-start products, or all stores. FreshRetail contains only three test origins and a 21-day inventory period. UCI has longer histories but represents a single UK online-retail setting and ends with an incomplete calendar day. Cross-dataset differences demonstrate heterogeneity, not generality across companies, years, or promotion regimes.

Observed-sales replay ignores demand lost, substituted, or deferred under actual stockouts and does not model substitution or price elasticity. Foundation-model pretraining may include public benchmarks, so only local label isolation is established. Left-padding short histories, residual-based quantiles, fixed hyperparameters, and single-run resource measurements also limit interpretations of model capability. Inventory service is not strictly matched before holding costs are compared. Where fill rates differ, cost differences must be interpreted jointly with service changes. These conditions delimit the scope of the reported findings.

8 Conclusion

This study completes a unified local evaluation of seven experts, eight combination strategies, two real retail datasets, and continuous inventory simulation. Hierarchical WAPE is 36.66% on FreshRetail and 89.02% on UCI, compared with 36.51% and 93.81% for equal weighting. Temporal separation, fixed sampling, whole-series cluster bootstrap, segment and SKU ablations, delayed-label online updates, and cumulative-risk calibration distinguish interface integration from forecast accuracy and inventory utility.

Heterogeneous combinations offer locally executable, traceable forecasting capabilities, but finer weighting structures are not intrinsically superior. Simple combinations must remain substantive baselines. Inventory evaluation must preserve on-hand and pipeline continuity while checking service, holdings, and cost together. By presenting both successful and unsuccessful cases, the study provides a reproducible and testable ensemble decision framework rather than an unconditional claim of a stronger model.

Author Contributions and Local Computation

Jiexiao Zhang is the sole author and is responsible for conceptualization, methodology, software, investigation, formal analysis, visualization, and writing. Public data, pretrained models, and open-source software are attributed to their original sources. Data processing, model inference, ensemble-weight estimation, statistical analysis, and inventory simulation were executed in the author's local computing environment. AI assistance in implementation and manuscript preparation is disclosed separately below.

Data and Code Availability

The public datasets are identified in references [13,14]. The base project code and usage documentation are available at <https://github.com/JiexiaoZhang1/inventory-fusion-commercial-v1>, and the model release repository is <https://huggingface.co/NeoZJX/inventory-fusion-commercial-v1>. The accompanying reproduction archive is the exact execution snapshot for the additional experiments reported here. It includes data preparation, frozen forecast protocols, analysis and plotting scripts, selected-series lists, source hashes, prediction tensors, and detailed results. Earlier demonstration results in the public release repositories do not by themselves reproduce the manuscript numbers. Original public data can be retrieved from the official links in the manifests, and model weights from the pinned revisions.

Funding and Competing Interests

Funding: This research received no external funding or in-kind support. Competing interests: The author declares no competing interests.

Declaration of Generative AI and AI-Assisted Technologies

OpenAI Codex was used to assist with software implementation and debugging, experiment orchestration, statistical-check scripts, figure and table preparation, manuscript drafting, and Chinese-to-English translation and editing. Forecasts and reported numerical results were generated by the local computational pipeline from the cited public datasets; generative AI was not used to fabricate observations or experimental values. Research figures are programmatic visualizations of recorded outputs, not generatively synthesized evidence. Jiexiao Zhang is the sole author and retains responsibility for the scientific content and for reviewing and approving the submitted version. Codex is not an author.

References

- [1] Fotios Petropoulos; Xun Wang; Stephen M. Disney. The inventory performance of forecasting methods: Evidence from the M3 competition data. *International Journal of Forecasting*, 35(1), 251–265, 2019.
<https://doi.org/10.1016/j.ijforecast.2018.01.004>
- [2] J. D. Croston. Forecasting and Stock Control for Intermittent Demands. *Operational Research Quarterly*, 23(3), 289–303, 1972.
<https://doi.org/10.1057/jors.1972.50>
- [3] Ruud H. Teunter; Aris A. Syntetos; M. Zied Babai. Intermittent demand: Linking forecasting to inventory obsolescence. *European Journal of Operational Research*, 214(3), 606–615, 2011.
<https://doi.org/10.1016/j.ejor.2011.05.018>
- [4] Aris A. Syntetos; John E. Boylan; J. D. Croston. On the categorization of demand patterns. *Journal of the Operational Research Society*, 56(5), 495–503, 2005.
<https://doi.org/10.1057/palgrave.jors.2601841>
- [5] Xiaolian Wang; Rob J. Hyndman; Feng Li; et al. Forecast combinations: An over 50-year review. *International Journal of Forecasting*, 39(4), 1518–1547, 2023.
<https://doi.org/10.1016/j.ijforecast.2022.11.005>
- [6] Pablo Montero-Manso; George Athanasopoulos; Rob J. Hyndman; et al. FFORMA: Feature-based forecast model averaging. *International Journal of Forecasting*, 36(1), 86–92, 2020.
<https://doi.org/10.1016/j.ijforecast.2019.02.011>
- [7] Abdul Fatir Ansari; Lorenzo Stella; Caner Turkmen; et al. Chronos: Learning the Language of Time Series. *arXiv:2403.07815*, 2024.
<https://arxiv.org/abs/2403.07815>
- [8] Abdul Fatir Ansari; Oleksandr Shchur; Jaris Küken; et al. Chronos-2: From Univariate to Universal Forecasting. *arXiv:2510.15821*, 2025.
<https://arxiv.org/abs/2510.15821>
- [9] Vijay Ekambaram; Arindam Jati; Pankaj Dayama; et al. Tiny Time Mixers (TTMs): Fast Pre-trained Models for Enhanced Zero/Few-Shot Forecasting of Multivariate Time Series. *Advances in Neural Information Processing Systems*, 2024.
<https://arxiv.org/abs/2401.03955>
- [10] Abhimanyu Das; Weihao Kong; Rajat Sen; et al. A decoder-only foundation model for time-series forecasting. *Proceedings of the 41st International Conference on Machine Learning*, PMLR 235, 10148–10167, 2024.
<https://proceedings.mlr.press/v235/das24c.html>

- [11] Google Research. TimesFM 2.5: google/timesfm-2.5-200m-pytorch model card and release notes. Hugging Face model card and official software repository, 2025.
<https://huggingface.co/google/timesfm-2.5-200m-pytorch>
- [12] Wei Yang; Defu Cao; Yan Liu. Foundation Models for Demand Forecasting via Dual-Strategy Ensembling. arXiv:2507.22053, 2025.
<https://arxiv.org/abs/2507.22053>
- [13] Yangyang Wang; Jiawei Gu; Li Long; et al. FreshRetailNet-50K: A Stockout-Annotated Censored Demand Dataset for Latent Demand Recovery and Forecasting in Fresh Retail. arXiv:2505.16319; accompanying Dingdong-Inc dataset, 2025.
<https://arxiv.org/abs/2505.16319>
- [14] Daqing Chen. Online Retail. UCI Machine Learning Repository [Dataset], 2015.
<https://doi.org/10.24432/C5BW33>
- [15] Chen Xu; Yao Xie. Conformal prediction interval for dynamic time-series. Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 11559–11569, 2021.
<https://proceedings.mlr.press/v139/xu21h.html>
- [16] Rob J. Hyndman; George Athanasopoulos. Forecasting: Principles and Practice. 3rd edition. Melbourne: OTexts. Section 5.10, Time series cross-validation, 2021.
<https://otexts.com/fpp3/tscv.html>
- [17] Hans R. Künsch. The Jackknife and the Bootstrap for General Stationary Observations. The Annals of Statistics, 17(3), 1217–1241, 1989.
<https://doi.org/10.1214/aos/1176347265>
- [18] Dimitris N. Politis; Joseph P. Romano. The Stationary Bootstrap. Journal of the American Statistical Association, 89(428), 1303–1313, 1994.
<https://doi.org/10.1080/01621459.1994.10476870>

Appendix A Pinned Model Versions and Reproduction Entry Points

Table A1. Identities of pretrained checkpoints

Model ID	Pinned revision
ibm-granite/granite-timeseries-ttm-r2	caa1eb3d16f053859d81952a1f387d6aeb670076
amazon/chronos-2	29ec3766d36d6f73f0696f85560a422f50e8498c
google/timesfm-2.5-200m-pytorch	1d952420fba87f3c6dee4f240de0f1a0fbc790e3

Local directories are ttm-r2.1-512-48, chronos-2, and timesfm-2.5, respectively. Version identities are retained from the fixed release configuration and official sources.

The reproduction sequence is `prepare_data.py`, `run_forecasts.py`, `analyze_study.py`, then `make_figures.py`. Data manifests retain filtering counts and selected IDs. The prediction tensor axes are [expert, series, origin, horizon, quantile], with quantiles ordered as 0.1, 0.5, and 0.9. The analysis directory retains aggregate, per-SKU, per-origin, cohort, and segment metrics, bootstrap comparisons, cumulative calibration, strictly separated calibration sensitivity, and daily inventory ledgers. The Excel workbook supports checking headline results; it does not replace statistical computations in the scripts.

Forecast caches are reused within the fixed experiment directory. Changes to input data, origins, models, or code require a new run directory or removal of the relevant caches in a separate working copy, followed by recomputation and verification of protocols and hashes. Old caches must not be presented as results of a new protocol. Fixed sampling and bootstrap seeds do not imply bitwise agreement across all GPU floating-point implementations.