Do global forecasting models require frequent retraining?

Marco Zanotti, Matteo Pelagatti

University of Milano-Bicocca

International Symposium on Forecasting

June 29 – July 2 2025, Beijing, China

Outline

Motivation

- Global models are widely used for large-scale time series forecasting
- Retraining a forecasting model on a large dataset is costly

Research question

- Is frequent retraining necessary in the context of global forecasting models?

Experimental Design

Datasets

The most recent and comprehensive time series datasets related to retail demand forecasting: the M5 and the VN1 competition datasets.

Dataset	Frequency	Period	N. Series	T	h
M5	Daily	2011-2016	28.298	364	28
VN1	Weekly	2020-2024	15.053	52	13

Table 1: Characteristics of the different datasets used.

Forecasting Models

10 different global forecasting models, 5 classical machine learning methods and 5 deep learning architectures.

Machine Learning	**Deep Learning**
Linear Regression (LR)	MLP
Random Forest (RF)	LSTM
XGBoost	TCN
LGBM	NBEATS
CatBoost	NHITS

Table 2: Global forecasting models used in the experiment.

Evaluation Strategy: Rolling Origin

- **Why?** Out-of-sample testing assesses generalization
- **How?**
  - Train/test split, preserving temporal order
  - Train model on initial training set
  - Forecast h steps ahead
  - Shift origin forward by p steps
  - Repeat until the test set is exhausted
- **Window types:**
  - Fixed window: Only the most recent n observations used
  - Expanding window: Include all historical data up to the current origin
- **In our study:** Expanding window with a step size $p = 1$

Performance Metrics

- **Point**: Root Mean Squared Scaled Error (RMSSE) (1)
- **Probabilistic**: Scaled Multi-Quantile Loss (SMQL) (2),(3)
- **Cost:** Computing Time (CT) in seconds

Simulation of the forecasting costs of each retraining scenario for a large retailer (200,000 SKUs and 5,000 stores) assuming some standard costs for computing services ($3.5/hour).

Retraining Scenarios

- **Retrain window (r)**: the frequency at which the model is re-trained or updated.
- **Scenarios based on data frequency:**
  - Daily: $r = \{7, 14, 21, 30, 60, 90, 120, 150, 180, 364\}$
  - Weekly: $r = \{1, 2, 3, 4, 6, 8, 10, 13, 26, 52\}$
- **Three strategies:**
  - Continuous retraining ($r = 7$ or $r = 1$): most expensive, most accurate. Baseline scenarios.
  - Periodic retraining ($7|1 < r < T$): balance between cost and accuracy.
  - No retraining ($r = T$): least expensive, least accurate.

Empirical Results

Point Forecasting Accuracy

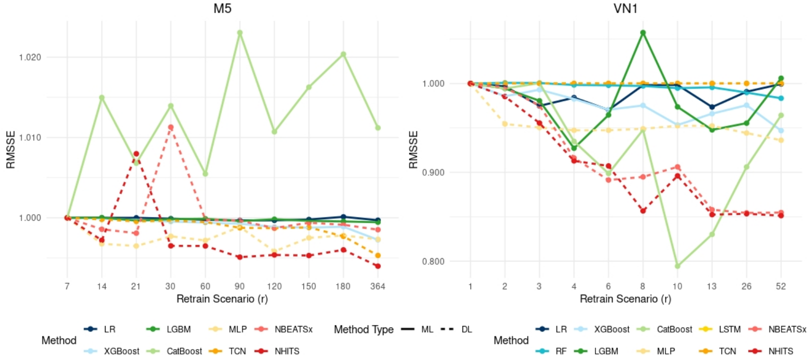

Figure 1: RMSSE results for each method and retrain scenario combination in relative terms with respect to the baseline scenarios ($r = 7$ and $r = 1$ respectively).

Probabilistic Forecasting Accuracy

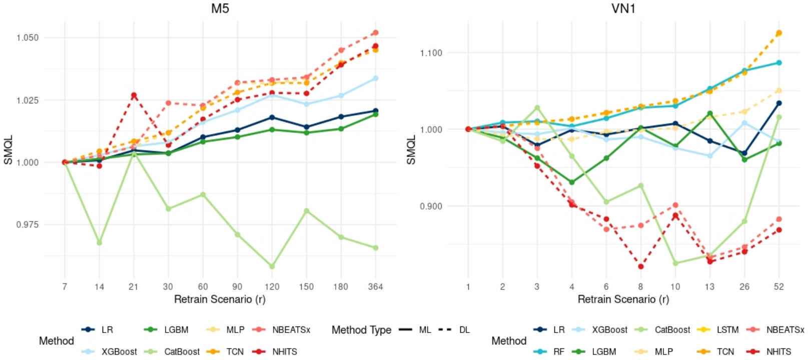

Figure 2: SMQL results for each method and retrain scenario combination in relative terms with respect to the baseline scenarios ($r = 7$ and $r = 1$ respectively).

Computing Time Performance

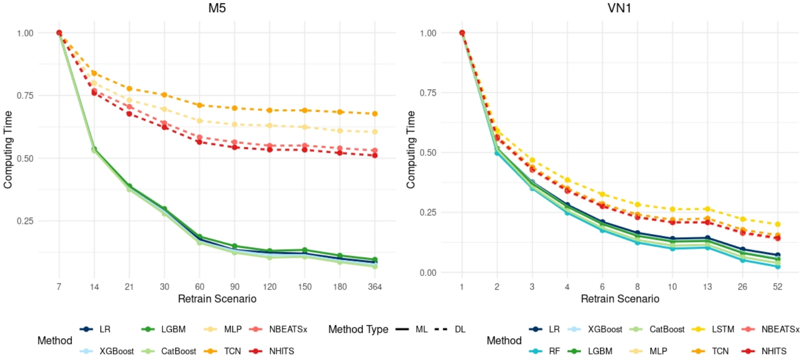

Figure 3: CT results for each method and retrain scenario combination in relative terms with respect to the baseline scenarios ($r = 7$ and $r = 1$ respectively).

Costs Analysis - Daily Data

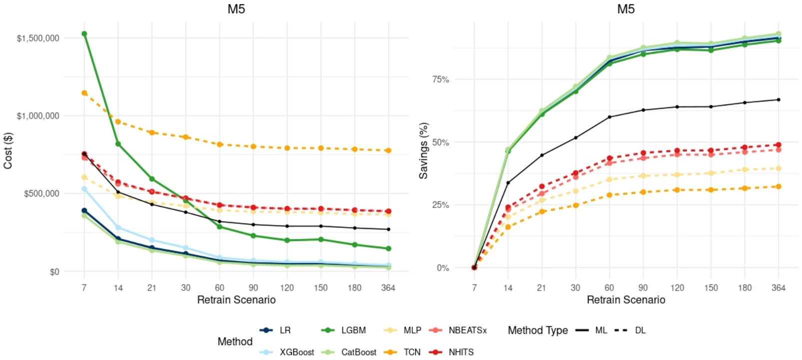

Figure 4: Estimated costs and savings for each method and retrain scenario combination. Average profile in black (from $750K to $250K).

Costs Analysis - Weekly Data

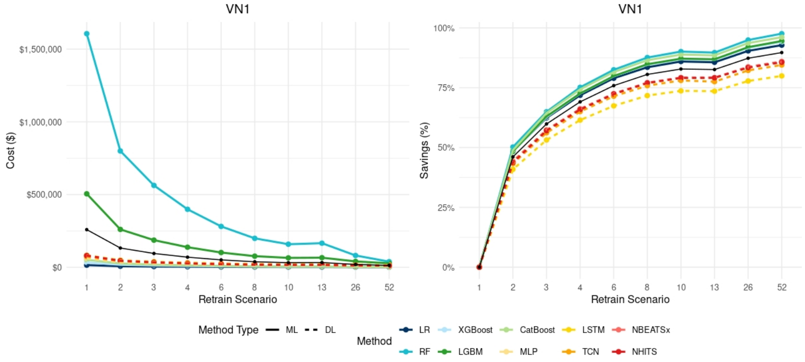

Figure 5: Estimated costs and savings for each method and retrain scenario combination. Average profile in black (from $250K to $15K).

Conclusions

Key Results

Accuracy

- Point accuracy is stable across retraining frequencies
- Probabilistic accuracy slightly degrades with less frequent retraining
- Periodic retraining often matches continuous retraining

Costs

- Computing time drops up to 90% with no retraining
- Cost savings: $500K (daily); $235K (weekly)
- ML models benefit more than DL from less frequent retraining as the frequency of the data increases

Key Takeaways

- Continuous retraining is unnecessary (under stable demand)
- Periodic retraining strategies balance cost with no effects on accuracy
- ML preferred over DL for high-frequency data under low retraining

Reducing the retraining frequency of global forecasting models allows to save cost and energy, often without harming accuracy and supporting more sustainable forecasting systems.

References

Zanotti, M. (2025). *Do global forecasting models require frequent retraining?*. arXiv. URL

Spiliotis, E., & Petropoulos, F. (2024). *On the update frequency of univariate forecasting models*. European Journal of Operational Research, 314, 111–121. URL

Thank you!

Appendix

Statistical Tests - Point Forecasting

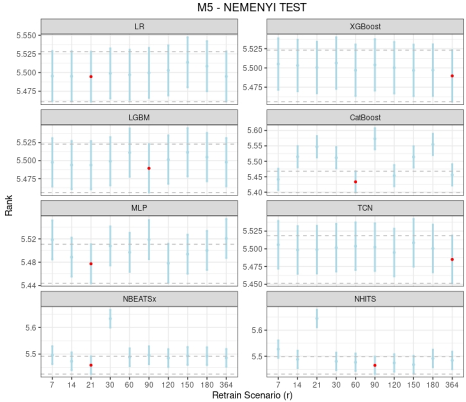

Figure 6: M5 Friedman-Nemenyi test results based on RMSSE.

Statistical Tests - Probabilistic Forecasting

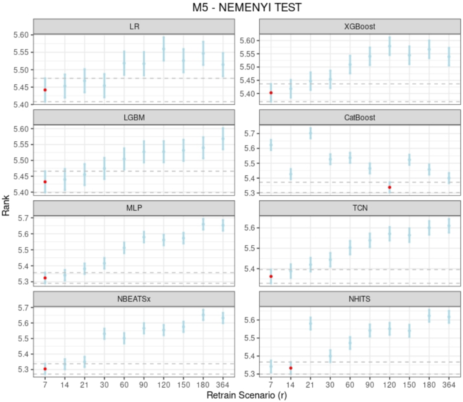

Figure 7: M5 Friedman-Nemenyi test results based on SMQL.

Cost Analysis

Method	7	14	21	30	60	90	120	150	180	364
LR	390,732	208,499	150,791	112,547	69,132	52,131	48,474	46,678	38,937	33,151
XGBoost	529,679	281,358	200,778	151,635	88,200	69,495	60,082	60,155	48,189	39,861
LGBM	1,526,424	818,569	593,676	455,075	286,262	228,698	199,337	204,972	170,802	146,252
CatBoost	358,123	189,714	134,258	99,699	58,167	44,164	37,206	38,266	30,622	24,362
MLP	604,120	482,505	441,981	419,823	392,046	383,287	380,697	376,968	368,023	365,525
TCN	1,146,256	960,626	890,924	862,432	815,006	801,612	791,801	791,348	784,201	776,181
NBEATSx	729,263	561,105	514,166	466,683	425,593	411,232	400,953	401,352	393,673	387,058
NHITS	754,783	573,233	510,655	470,077	425,684	409,844	402,731	402,394	393,071	385,470
Average	754,922	509,451	429,654	379,746	320,011	300,058	290,160	290,267	278,440	269,733

Table 3: M5 estimated costs (in $) for each method and retrain scenario combination.

Cost Analysis

Method	1	2	3	4	6	8	10	13	26	52
LR	15,234	7,839	5,731	4,292	3,205	2,508	2,140	2,190	1,463	1,099
RF	1,605,768	799,597	562,896	398,954	281,246	199,214	158,959	165,975	81,529	38,865
XGBoost	34,254	17,687	12,796	9,413	6,944	5,262	4,473	4,534	2,863	2,005
LGBM	505,304	260,721	186,904	138,460	101,775	76,738	65,013	66,335	40,721	27,701
CatBoost	52,021	26,594	18,515	13,366	9,611	7,022	5,798	5,977	3,311	2,023
MLP	62,102	34,489	26,431	21,149	17,121	14,391	13,027	13,015	10,255	8,891
LSTM	82,927	49,019	38,851	31,962	27,022	23,465	21,829	21,932	18,415	16,652
TCN	72,763	41,291	31,910	25,464	20,791	17,579	15,983	16,365	12,936	11,284
NBEATSx	80,337	44,837	34,283	27,179	22,057	18,387	16,686	16,702	13,085	11,277
NHITS	80,776	45,458	34,688	27,585	22,336	18,607	16,888	16,855	13,405	11,601
Average	259,149	132,753	95,301	69,782	51,211	38,317	32,080	32,988	19,798	13,140

Table 4: VN1 estimated costs (in $) for each method and retrain scenario combination.

Performance Metrics - Math

$$\mathrm{RMSSE} = \sqrt{\frac{\frac{1}{h}\sum_{t=n+1}^{n+h}(y_t - \hat{y}_t)^2}{\frac{1}{n-s}\sum_{t=s+1}^{n}(y_t - y_{t-s})^2}} \quad (1)$$

$$\mathrm{SQL} = \frac{1}{h}\frac{\sum_{t=n+1}^{n+h}\left(q \cdot (y_t - \hat{y}_t) \cdot \mathbb{I}_{y_t \geq \hat{y}_t} + (1-q) \cdot (\hat{y}_t - y_t) \cdot \mathbb{I}_{y_t < \hat{y}_t}\right)}{\frac{1}{n-s}\sum_{t=s+1}^{n}|y_t - y_{t-s}|} \quad (2)$$

$$\mathrm{SMQL} = \frac{1}{|\mathcal{Q}|}\sum_{q \in \mathcal{Q}} \mathrm{SQL}(q) \quad (3)$$

Computing Setup

- Microsoft Azure NC6s v3 VM
- 6 vCPUs, 1 GPU, 112 GB RAM
- Nixtla's libraries: `mlforecast`, `neuralforecast`