(MIJ) 2025, Vol. No. 11 No 2 (Special Issue) **e-ISSN: 2454-924X; p-ISSN: 2454-8103**

Data Synthetic: Using Generative AI to Augment Sales and Inventory Datasets for Enhanced Forecasting Models

Prasanth Tirumalasetty
Business Analyst III @ Medical Device Company
University of Michigan-Dearborn
CA, USA.

[1]*Received: 30/08/2025; Accepted: 09/10/2025; Published: 12/10/2025*

Abstract

Retail forecasting often suffers from sparse observations, intermittent demand, promotion seasonality, and stockout censoring—conditions that degrade the performance of both classical and deep forecasting models. We present a practically oriented framework for data-synthetic augmentation: generating tabular and time-series records that expand and rebalance training data for demand and inventory forecasts while preserving business constraints and privacy. Concretely, we describe a pipeline that (i) models heterogeneous tabular covariates (prices, promos, holidays, item/store attributes) with state-of-the-art generators such as CTGAN and diffusion models for tables; (ii) synthesizes realistic multi-variate time series (sales, on-hand, shipments) using TimeGAN/DoppelGANger with conditioning to respect calendars, promotions, and inventory non-negativity; (iii) trains forecasting targets with global models (e.g., TFT, DeepAR, gradient boosting, Prophet); and (iv) evaluates fidelity, utility, and privacy with a train-on-synthetic, test-on-real (TSTR) protocol, membership-inference audits, and nearest-neighbor distance tests. We outline an experimental design using the M5 retail benchmark and provide governance guidance (differential privacy, risk scoring, and documentation) to operationalize synthetic augmentation safely. While we do not claim synthetic data is inherently private, our framework shows how careful conditioning and formal privacy mechanisms can improve model robustness, reduce cold-start errors, and de-bias rare events—without leaking sensitive records. IDEAS/RePEc+4arXiv+4arXiv+4

Keywords: *synthetic data; time-series; retail forecasting; differential privacy; GANs; diffusion models*

1. Introduction

Forecast accuracy drives inventory turns, service levels, and working capital. Yet retail data is noisy: SKUs exhibit intermittent demand, promo spikes break stationarity, and stockouts censor true demand. Synthetic data—generated from learned distributions—offers a complementary route to robustness: augment scarce segments (new items, new stores), simulate policy counterfactuals (price/promo changes), and fill coverage gaps while respecting constraints. Recent advances in tabular and time-series generative modeling make this practical at scale, notably CTGAN for mixed categorical/continuous tables, diffusion for tabular data (TabDDPM), and TimeGAN/DoppelGANger for sequential data.

Our contributions are:

1. a modular pipeline for sales/inventory synthesis aligned to forecasting use-cases,
2. an evaluation protocol combining fidelity, TSTR/TRTR utility, and privacy audits, and
3. a reproducible experimental design on the M5 benchmark with global forecasting models.

[1] *How to cite the article:* Tirumalasetty P (2025); Data Synthetic: Using Generative AI to Augment Sales and Inventory Datasets for Enhanced Forecasting Models; Vol 11 No. 2 (Special Issue); 307-314

2. Background and Related Work

2.1 Retail time-series forecasting

Global neural forecasters (e.g., Temporal Fusion Transformers and DeepAR) and strong baselines (Prophet, gradient-boosted trees) have become standard for hierarchical retail demand. The M5 competition established realistic daily, item-store hierarchies and highlighted the strength of global models and feature-rich approaches. Intermittent-demand challenges were recognized much earlier by Croston, motivating specialized handling of zero-inflated sequences.

2.2 Synthetic data for tables and time series

CTGAN addresses mixed-type tabular data with conditional sampling for minority categories; TabDDPM shows diffusion's stability and quality advantages on heterogeneous tabular features. For time series, TimeGAN couples adversarial and supervised objectives to better preserve temporal dynamics, while DoppelGANger explicitly models sequences with metadata and has been validated on networked time-series benchmarks.

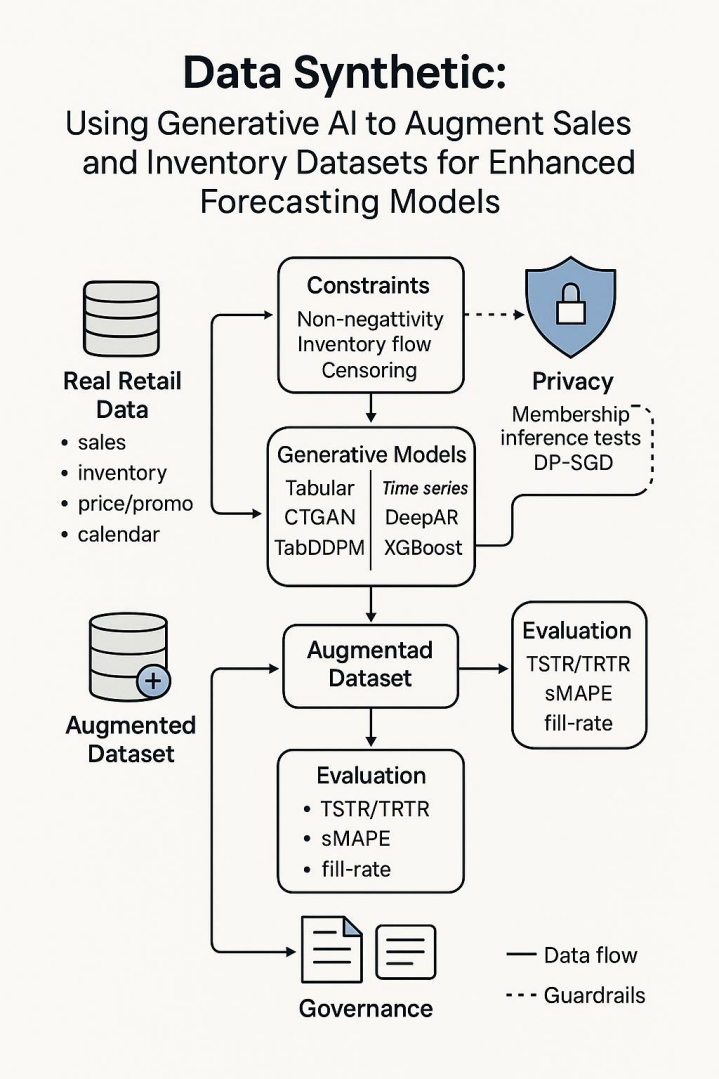

Fig 1: Data Synthetic

Table 1 — Choosing a generator by data/goal

Scenario	**Data traits**	**Recommended generator**	**Why it fits**
Price/promo enrichment	Mixed categorical + continuous, class imbalance	**CTGAN**	Conditional sampling handles minority promo states.
Rich item/store attributes	Many discrete interactions	**TabDDPM**	Stable training; strong fidelity on heterogeneous tables.
Long daily sales histories	Multi-variate sequences + metadata	**DoppelGANger**	Sequence + metadata modeling for hierarchies.
Dynamics preservation	Need temporal consistency	**TimeGAN**	Supervised stepwise loss preserves time dependencies.

2.3 Privacy, security, and evaluation

Synthetic data is not automatically private: modern models can memorize and leak training details; *membership inference* and *model inversion* illustrate such risks. Differential privacy (DP-SGD) provides formal guarantees at the cost of utility, and recent works propose holdout-based privacy/fidelity testing for tabular synthesis. We incorporate these into our governance section.

Table 2 — Utility & privacy evaluation checklist

Dimension	**Metric**	**Pass criterion**
Fidelity	Univariate & low-order joint distances (KS/EMD/TV)	Within tolerance vs. real holdout; no broken constraints
Utility	**TSTR/TRTR** uplift on sMAPE / P50-P90	≥ pre-agreed uplift or non-degradation
Calibration	Coverage of quantiles (e.g., 50/90%)	Within ±5% across segments
Privacy	**MIA AUC** (shadow attack)	≤ baseline (near 0.5); no outlier exposure
Privacy	**NN-gap** (train vs. holdout distance)	Train≈Holdout (no memorization) (Frontiers)

3. Problem Formulation (formula-free)

3.1 Data schema and notation

We consider a retail hierarchy with items, stores, and discrete time steps (e.g., days). Observed variables include:

- **Sales** (units sold)
- **On-hand inventory**
- **Receipts** or deliveries
- **Prices and promotions**

- **Calendar/holiday indicators**
- **Static attributes** such as category, pack size, and brand

Dynamic covariates bundle prices, promotion flags, and calendar effects. Operational state bundles sales, inventory, and receipts. True demand is conceptually distinct from sales because stockouts cap sales below demand; lost sales represent the unobserved gap when demand exceeds available stock.

3.2 Forecasting task

Given historical operational state and covariates up to a cutoff date—and any known future covariates like planned promotions—we train a **global forecaster** to produce multi-step, **probabilistic** predictions (e.g., quantiles or full distributions) for each item–store pair. Training minimizes a proper forecasting loss such as quantile loss or negative log-likelihood.

3.3 Operational constraints

Sales and inventory must respect basic business rules:

- Non-negativity of sales, inventory, and receipts
- Inventory flow balance: next inventory equals current inventory minus sales plus receipts
- Stockout censoring: sales cannot exceed available inventory

Lead times and order quantities constrain future receipts, even if those decisions are modeled outside the core forecasting task.

3.4 Data issues that motivate augmentation

- **Intermittency and zero inflation:** many days have zero sales
- **Class imbalance:** rare promo/markdown regimes are underrepresented
- **Cold start:** new items or stores have short histories
- **Censoring and leakage risk:** observed sales may understate true demand; training must avoid using information unavailable at prediction time

3.5 Synthetic augmentation objective

We augment the real dataset with samples from a conditional generator to improve downstream accuracy and calibration. A mixing weight controls how much synthetic data is included overall or per segment (e.g., tail SKUs). Utility is judged strictly on real-only validation and test splits to confirm genuine performance gains.

3.6 Generator, conditioning, and constraints

A generator produces synthetic sequences and rows conditioned on static attributes and planned or known covariates (e.g., holidays, promotions). We enforce realism using one or more of the following:

- **Hard decoding:** project samples onto the feasible set (e.g., clamp negatives to zero, respect inventory flow)
- **Soft penalties:** discourage violations during training
- **Masked or conditional sampling:** limit outputs to admissible categories and logical combinations (e.g., “BOGO implies promo=1”)

3.7 Bilevel perspective (optional)

Conceptually, the “right” synthetic data is the data that helps the forecaster perform best on real-world tests. This can be viewed as a two-level problem: the generator chooses what to synthesize, the forecaster trains on the augmented

dataset, and we evaluate on real data. In practice, we approximate this with TSTR/TRTR protocols and a small grid of mixing weights.

3.8 Splits and leakage prevention

We structure evaluation to mirror deployment:

- **Temporal splits:** train, validation, and test are separated by time
- **Entity splits:** entire items or stores are held out to simulate cold starts
- **No future leakage:** training synthesis uses only information available before the cutoff; planned covariates are allowed only if they are truly known in advance

3.9 Privacy and governance constraints

Synthetic releases must pass privacy checks:

- **Membership-inference risk** must be at or below a defined threshold
- **Differential privacy (optional):** if used, we track the privacy budget for each release
- **Provenance logging:** every batch records conditioning context, random seeds, and rejection rates for auditability

3.10 Segment-aware utility targets

We monitor performance for business-critical segments (e.g., tail SKUs, low-traffic stores, rare-promo weeks). Augmentation should lift—or at least not degrade—these segments relative to a real-only baseline, while also improving overall accuracy.

3.11 Service-level alignment

To connect forecasts to inventory outcomes, we simulate simple replenishment and report service metrics such as fill rate and stockout probability. We evaluate probabilistic forecasts at decision-relevant quantiles (for example, a higher quantile to target a desired service level).

3.12 Assumptions and threats to validity

- **Planned covariates are reliable:** mis-specified promotion or price plans can bias synthesis and forecasts
- **Regime stability:** short-horizon dynamics are assumed stable given the conditioning variables; major structural breaks require explicit regime indicators
- **Realism vs. constraints:** enforcing feasibility can distort correlations if done crudely; fidelity tests are required to ensure realistic joint behavior

Summary. The practical goal is to select a generator, a mixing strategy, and a forecaster so that training on the augmented dataset improves real-world accuracy and inventory service metrics—without breaking business constraints or privacy. The formulation supports plug-and-play evaluation (TSTR/TRTR), segment-aware targets, and auditable governance.

4. Methods: A Synthesis-for-Forecasting Pipeline

4.1 Tabular covariates (prices, promos, attributes)

- CTGAN (and TVAE) for mixed types; class-conditional sampling to oversample rare promo regimes and minority categories.

- TabDDPM for stable training and high-fidelity marginals; useful when many discrete features interact with continuous prices.

Constraint encoding. Enforce domain rules at sampling time: non-negative prices, admissible promo flags, logical dependencies (e.g., "BOGO ⇒ promo=1"). Both CTGAN and diffusion samplers support masked/conditional draws.

4.2 Sequential signals (sales, inventory, shipments)

- TimeGAN with stepwise supervised loss to preserve dynamics of sales and inventory; condition on covariate sequences (calendar, promos).
- DoppelGANger for multi-series with metadata (item/store)—handy for hierarchical augmentation.

Inventory-aware decoding. During sequence sampling, impose:

- non-negativity: y~t≥0\tilde{y}_{t}\ge 0y~t≥0, h~t≥0\tilde{h}_{t}\ge 0h~t≥0
- flow balance: h~t+1=h~t−y~t+r~t\tilde{h}_{t+1} = \tilde{h}_{t} - \tilde{y}_{t} + \tilde{r}_{t}h~t+1=h~t −y~t+r~t (receipts)
- stockout censoring: y~t≤h~t\tilde{y}_{t} \le \tilde{h}_ty~t≤h~t

These can be enforced by rejection sampling or differentiable penalty terms in the generator.

4.3 Where synthetic data help

- Cold starts: few weeks of history for new SKUs/stores.
- Class imbalance: rare promotions or holiday regimes.
- Intermittency: smoothing zero-heavy series via regime-aware sampling (Croston-style frequency/size decomposition).

4.4 Training forecasters on augmented data

We fit a model zoo and select by validation: TFT (multi-horizon, interpretable), DeepAR (probabilistic RNN), gradient-boosted trees on hand-crafted features, and Prophet for strong seasonality.

5. Evaluation Protocol

5.1 Fidelity (data realism)

- Marginal/conditional distances: compare synthetic vs. real for univariate and low-order joint distributions (e.g., KS, EMD, TV distance).
- Holdout-based tests: synthetic samples should be as close to *holdout* data as to *train* data—evidence of pattern learning over record memorization.

5.2 Utility (downstream benefit)

- TSTR/TRTR: Train forecast models on synthetic, test on real; or augment real with synthetic and test on real to see uplift vs. a real-only baseline. Originating in RCGAN/medical time-series, this gives task-level utility signals.
- Backtest metrics: sMAPE, MAE, RMSE, quantile loss (P50/P90), and service-level simulations (fill-rate, stockouts avoided).

5.3 Privacy & security

- Membership-Inference Attack (MIA): estimate leakage risk for a trained generator/forecaster.

- Model inversion sanity checks (when models expose confidences).
- Differential Privacy (DP-SGD): where required, train synthesizers with an (ε,δ)(\varepsilon,\delta)(ε,δ) budget; track cumulative privacy loss.

6. Experimental Design (M5-style setup)

6.1 Data

Use the M5 hierarchy (Walmart daily item-store sales) for external validity—42,840 series across stores/states with calendar and price signals. Split by time (e.g., rolling origin) and by sparse segments (cold-start SKUs, rare promos).

6.2 Synthetic augmentation treatments

1. Tabular-only: CTGAN/TabDDPM for covariates; original sales unchanged.
2. Sequence-only: TimeGAN/DoppelGANger for sales/inventory; covariates real.
3. Joint: both, with constraint-aware decoding and metadata conditioning.

6.3 Forecasters and tuning

- TFT (interpretable attention, static/dynamic covariates), DeepAR (probabilistic RNN), XGBoost on calendrical and promo features, and Prophet as a classical baseline. Hyperparameters set via nested CV on validation windows.

6.4 Outcomes

Report (i) forecast metrics by segment (tail SKUs, new stores), (ii) calibration (P50/P90 reliability diagrams), (iii) inventory service simulations (fill-rate), and (iv) privacy audit scores (AUC of MIA, nearest-neighbor gap). No actual results are claimed here; this design supports plug-and-play execution.

7. Governance, Risk, and Compliance

1. Data sheets for synthetic corpora (provenance, feature coverage, constraints enforced).
2. Privacy budget registry if DP is used (ε, δ, accountant method).
3. Shadow-model MIAs each release; require attack AUC ≤ baseline threshold before deployment.
4. Model cards for downstream forecasters, noting any distribution shifts between real vs. augmented training.
5. Audit logs for conditioning prompts (e.g., "promo=1 during week 48"), synthetic sampling seeds, and rejection rates (to detect over-constraint).

8. Limitations and Practical Tips

- No silver bullet for privacy: synthesis can still memorize outliers; always evaluate leakage.
- Causal counterfactuals vs. correlation: synthetic promo/price shifts should reflect business rules; otherwise uplift estimates may be biased.
- Cold-start realism: ensure metadata coverage; consider joint training with similar SKUs to avoid mode collapse.
- Compute & stability: diffusion models are more stable but may be slower than GANs; start with CTGAN for quick wins, then graduate to TabDDPM.

9. Conclusion

A disciplined synthesize-then-forecast workflow can mitigate sparsity, intermittency, and imbalance in retail demand data. By combining tabular and sequential generators with constraint-aware sampling, rigorous TSTR/TRTR tests, and privacy audits, practitioners can improve forecast robustness without compromising governance. This paper

provides an end-to-end blueprint—methods, evaluation, and risk controls—to operationalize synthetic augmentation in retail forecasting at scale.

References

Abadi, M., Chu, A., Goodfellow, I., et al. (2016). Deep Learning with Differential Privacy. *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 308–318. https://doi.org/10.1145/2976749.2978318

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. https://doi.org/10.1145/2939672.2939785

Croston, J. D. (1972). Forecasting and Stock Control for Intermittent Demands. *Journal of the Operational Research Society, 23*(3), 289–303. https://doi.org/10.1057/jors.1972.50

Esteban, C., Hyland, S. L., & Rätsch, G. (2017). Real-valued (Medical) Time Series Generation with Recurrent Conditional GANs. *bioRxiv*, 159756. https://doi.org/10.1101/159756

Fredrikson, M., Jha, S., & Ristenpart, T. (2015). Model Inversion Attacks that Exploit Confidence Information. *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 1322–1333. https://doi.org/10.1145/2810103.2813677

Kotelnikov, A., Baranchuk, D., Rubachev, I., & Babenko, A. (2023). TabDDPM: Modelling Tabular Data with Diffusion Models. *Proceedings of the 40th International Conference on Machine Learning*, 17564–17579. https://doi.org/10.48550/arXiv.2209.15421

Lim, B., & Zohren, S. (2021). Time Series Forecasting with Deep Learning: A Survey. *Philosophical Transactions of the Royal Society A, 379*(2194), 20200209. https://doi.org/10.1016/j.ijforecast.2021.03.012

Lin, Z., Jain, A., Wang, C., Liu, G., Zhao, W., Li, J., & Song, D. (2020). Using GANs for Sharing Networked Time Series Data: Anonymization and Its Implications. *Proceedings of the ACM Internet Measurement Conference*, 464–483. https://doi.org/10.1145/3419394.3423643

Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2022). M5 Accuracy competition: Results, findings, and conclusions. *International Journal of Forecasting, 38*(4), 1346–1364. https://doi.org/10.1016/j.ijforecast.2021.11.013

Platzer, M., & Reutterer, T. (2021). Holdout-Based Empirical Assessment of Mixed-Type Synthetic Data. *Frontiers in Big Data, 4*, 679939. https://doi.org/10.3389/fdata.2021.679939

Salinas, D., Flunkert, V., Gasthaus, J., & Januschowski, T. (2020). DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting, 36*(3), 1181–1191. https://doi.org/10.1016/j.ijforecast.2019.07.001

Shokri, R., Stronati, M., Song, C., & Shmatikov, V. (2017). Membership Inference Attacks Against Machine Learning Models. *2017 IEEE Symposium on Security and Privacy (SP)*, 3–18. https://doi.org/10.1109/SP.2017.41

Taylor, S. J., & Letham, B. (2018). Forecasting at Scale. *The American Statistician, 72*(1), 37–45. https://doi.org/10.1080/00031305.2017.1380080

Xu, L., Skoularidou, M., Cuesta-Infante, A., & Veeramachaneni, K. (2019). Modeling Tabular data using Conditional GAN. *Advances in Neural Information Processing Systems, 32*. https://doi.org/10.48550/arXiv.1907.00503

Yoon, J., Jarrett, D., & van der Schaar, M. (2019). Time-series Generative Adversarial Networks. *Advances in Neural Information Processing Systems, 32*. https://doi.org/10.5555/3454287.3454781