# A CONSTRAINED GENERATIVE FRAMEWORK FOR SCALING METROLOGY DATA IN MULTI-COMPONENT AUTOMOTIVE ASSEMBLIES

**Faizan Ali[1*] – Ameya Punekar[1] – Juergen Bock[1]**

[1]Technische Hochschule Ingolstadt, AImotion Bavaria, Esplanade 10, 85049 Ingolstadt, Germany

**ARTICLE INFO**

***Article history:***
*Received: 20. 4. 2026.*
*Received in revised form: 11. 5. 2026.*
*Accepted: 29. 7. 2026.*

***Keywords:***
Synthetic data generation
Manufacturing metrology
Constraint enforcement
Rejection sampling
Hyperparameter optimisation
Tabular generative models

*DOI: 10.30765/er.3431*

***Abstract:***

Automotive pilot production yields limited metrology data, which hinders the development of robust quality-assurance models. This paper presents a constrained generative framework that synthesises multi-component assembly data while ensuring adherence to production boundaries. The pipeline combines Bayesian hyperparameter optimisation with rejection sampling and evaluates three generative architectures: a conditional tabular generative adversarial network, a tabular variational autoencoder, and a Gaussian copula model. The framework was tested on a roller-slide door assembly comprising eight components and approximately 1,200 real measurements, across 360 experimental conditions spanning three generative models, three scaling factors and five random seeds. Constraint enforcement was implemented as a post-hoc filter applied outside the generator, so that the same compliance layer wraps all three architectures without modification. Full adherence to the encoded empirical bounds and the torque-ordering rule was confirmed in every condition. Hyperparameter optimisation proved critical, increasing the mean Kolmogorov–Smirnov pass rate from 49.0 % to 94.0 % for the adversarial model, from 63.5 % to 89.0 % for the variational autoencoder, and from 55.2 % to 79.0 % for the copula model. The results confirm that tuned generative models can successfully scale scarce metrology datasets.



## 1. Introduction

Quality-assurance workflows in automotive manufacturing increasingly rely on machine learning, but the models behind them are only as good as the data they train on. During pilot production, metrology campaigns yield at most a few hundred measurements per component, nowhere near enough to cover the full range of dimensional variation that drives assembly failures [1]. Destructive tests are expensive, measurement cycles are slow, and proprietary restrictions block data sharing between suppliers [1], [2]. For assemblies built from multiple parts, each with its own measurement protocol, the data gap widens further.

Generating synthetic data is one way around this bottleneck. If the synthetic data preserves the statistical fingerprint of the real measurements, downstream models can train on the augmented set without sacrificing fidelity [3], [4]. Several tabular generators now exist, such as CTGAN and TVAE from Xu et al. [5] and copula-based methods in the Synthetic Data Vault [6], but none of them enforce domain-specific manufacturing constraints out of the box. A CTGAN trained on torque data, for instance, may readily generate a record in which the minimum torque exceeds the maximum. That record is statistically plausible, given what the generator learned, yet it describes a physically impossible configuration [1].

* Corresponding author
*E-mail address: faizan.ali@thi.de*

The proposed framework addresses this gap. Each generator is wrapped in a rejection-sampling layer that filters out constraint-violating records, and the entire pipeline is paired with Bayesian hyperparameter optimisation to extract more fidelity from each architecture. The contributions are: (1) a preprocessing pipeline for common metrology issues such as nested position columns, quantised weight readings and bimodal batch effects; (2) a model-agnostic compliance layer that enforces empirical range bounds and one torque-ordering rule via post-hoc rejection sampling, with no changes to generator internals; (3) a Bayesian optimisation stage (Optuna TPE [7], 15 trials) that tunes each model per component; and (4) a multi-seed evaluation protocol (five seeds, 360 conditions) that guards against lucky-seed artefacts. The framework is evaluated on an automotive roller-slide door assembly with eight components and roughly 1,200 real measurements.

## 2. Related work

CTGAN and TVAE [5], [8] handle mixed-type tabular data through mode-specific normalisation; extensions such as CTAB-GAN [9], MargCTGAN [10] and PSVAE [11] target semantic consistency, small-sample marginal fidelity and post-selection filtering, respectively. The Synthetic Data Vault [6] implements copula-based generation via Sklar's theorem [3], [12], separating marginal estimation from dependence modelling, which is a useful property when training data is scarce.

On the manufacturing side, Buggineni et al. [1] surveyed synthetic-data methods and flagged two gaps: data scarcity and the absence of built-in constraint handling. Constrained Deep Generative Models [13] and Physics-Aware Normalizing Flows [14] close the second gap by embedding constraints as differentiable layers. This places constraint enforcement inside the model, with three consequences: arbitrary differentiable constraints can be expressed exactly; training cost increases because the constraint layer participates in optimisation; and the compliance mechanism is tied to a specific architecture, so swapping generators requires re-deriving the layer. The black-box rejection approach adopted here trades the first property for the other two: constraint expressivity is restricted to predicates that can be evaluated post-hoc on samples, while training is unchanged and the same layer wraps CTGAN, TVAE and Gaussian Copula without modification. For the constraints in this dataset the rejection overhead stayed low, as quantified in Section 6.

Evaluation of synthetic tabular data usually covers statistical fidelity, machine-learning utility and privacy [15], [16]. The two-sample Kolmogorov–Smirnov test remains the standard univariate fidelity check [15], [16], yet none of the published frameworks incorporate constraint-satisfaction metrics tailored to manufacturing tolerances. This work adds that component.

## 3. Methodology

### *3.1 Framework overview*

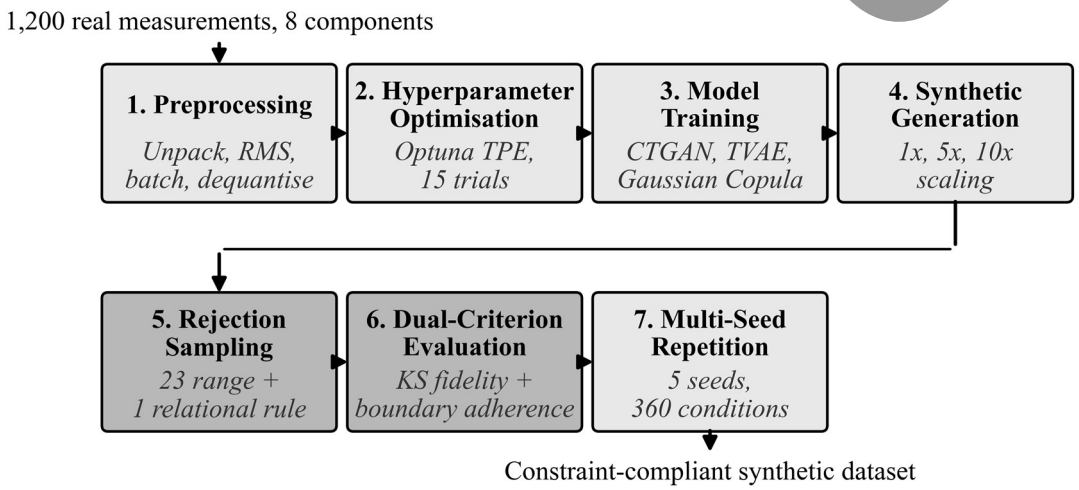


*Figure 1. Constrained generative framework pipeline.*

The pipeline illustrated in Figure 1 has seven stages: (1) preprocessing of raw metrology data files, (2) Bayesian hyperparameter optimisation per model–component pair, (3) model training with the best-found parameters, (4) synthetic generation at configurable scaling factors (1×, 5×, 10×), (5) rejection sampling for constraint enforcement, (6) dual-criterion evaluation combining Kolmogorov–Smirnov fidelity with boundary adherence, and (7) multi-seed repetition across five seeds to quantify variance. Stages 5 and 6 are model-agnostic: the same constraint engine and evaluation code wrap CTGAN, TVAE and Gaussian Copula without modification.

### *3.2 Data preprocessing*

Manufacturing metrology data rarely arrives in a form that generators handle well. Four transforms are applied. First, nested measurement unpacking: four of the eight components store paired nesting-position columns, so 100 rows × 2k columns are unpacked into 200 rows × k columns. Second, root-mean-square compression: the WindingTube's 30 straightness-deviation points collapse to three root-mean-square values, one per measurement position, cutting dimensionality by 90 %. Third, batch conditioning: Spring and SpringCover come from two production batches with noticeably different distributions, so a binary batch indicator is added. Fourth, dequantisation: weight columns recorded at 0.5 g precision receive a small Gaussian jitter ($\sigma = 0.125$ g) to remove the discrete spikes that impair density estimators.

### *3.3 Hyperparameter optimisation*

Default hyperparameters are a poor fit for 200-row metrology datasets. Optuna's Tree-structured Parzen Estimator (TPE) [7] is run for 15 trials on each of the 24 model–component combinations. The objective maximises the composite score given in Eq. (1):

$$S = 0.7p_{KS} + 0.2(1 - D_{KS}) + 0.1(1 - min(MAE_{\rho}, 1.0)) \quad (1)$$

where $S$ is the composite score, $p_{KS}$ the Kolmogorov–Smirnov pass rate, $D_{KS}$ the mean Kolmogorov–Smirnov statistic and $MAE_{\rho}$ the mean absolute error between the Pearson correlation matrices of the real and synthetic data. The pass rate dominates because distributional fidelity matters most for downstream use.

For CTGAN, the search space covers epochs (100–1000), batch size, PAC parameter, generator and discriminator layer counts and widths, and both learning rates. TVAE trials vary epochs, compression and decompression architectures, embedding dimension, L2 regularisation scale and loss factors. Gaussian Copula trials explore per-column distribution families (normal, beta, gamma, truncated Gaussian, uniform). The full search-space definitions are given in the configuration files for each model.

### *3.4 Constraint enforcement*

Two rule types govern every synthetic record. Across the eight components, 23 per-feature range constraints set upper and lower bounds drawn from observed extremes. One relational constraint encodes torque ordering for the AxleDamper, Eq. (2):

$$\tau_{min} \leq \tau_{mean} \leq \tau_{max} \quad (2)$$

where $\tau_{min}$, $\tau_{mean}$ and $\tau_{max}$ denote the minimum, mean and maximum torque of a record. At generation time the candidate pool is oversampled by 3×, all rules are evaluated vectorially, compliant rows are retained, and the result is truncated to the target count. Because the filter sits outside the generator, it works identically for all three architectures: substituting a different model leaves the constraint layer unchanged.

### *3.5 Evaluation metrics*

Each synthetic dataset is assessed on three axes. Distributional fidelity: a two-sample Kolmogorov–Smirnov test per numeric feature at $\alpha = 0.05$, reporting both the mean Kolmogorov–Smirnov statistic and the fraction of features that pass, termed the pass rate. Boundary adherence: the share of synthetic records satisfying every constraint rule, for which the target is 1.0 after filtering. Correlation preservation: mean absolute error between the Pearson correlation matrices of the real and synthetic data.

### *3.6 Multi-seed protocol*

Earlier work used a single seed, leaving open the question of whether results were seed dependent. Every evaluation is repeated across five seeds (42, 123, 456, 789, 2024), reporting mean ± standard deviation for each metric. The internal seeding mechanism of the Synthetic Data Vault was found to silently override the global NumPy seed with its own fixed constant, so naively setting a seed before sampling produces identical outputs. Only by calling the library's own reset_sampling() and _set_random_state() methods do per-seed randomness take effect.

## 4. Experimental setup

The dataset comes from production-qualification measurements of an automotive roller-slide door mechanism. Table 1 gives a per-component summary.

*Table 1. Dataset summary.*

| Component | Clean rows | Preprocessed rows | Features | Key pathology |
|---|---|---|---|---|
| AxleDamper | 200 | 200 | 3 | Torque correlations |
| EndCapLeft | 100 | 200 | 3 | Constant weight |
| EndCapRight | 100 | 200 | 3 | Quantised weight |
| GegenLagerFeder | 100 | 200 | 2 | Narrow range |
| LagerFeder | 100 | 200 | 3 | Extreme kurtosis |
| Spring | 200 | 200 | 4 | Bimodal (2 batches) |
| SpringCover | 200 | 200 | 3 | Bimodal (2 batches) |
| WindingTube | 200 | 200 | 4 | 30-point profile to 3 RMS |
| Total | 1,200 | 1,600 | 25 | |

Two configurations are compared. The baseline uses Synthetic Data Vault default hyperparameters, or minimal per-component tuning for CTGAN, with a single seed, evaluated at three scales and yielding 72 conditions. The optimised configuration uses Optuna-tuned parameters with five seeds at three scales, yielding 360 conditions. All three models are implemented through the Synthetic Data Vault library v1.30 [6].

## 5. Results

### *5.1 Boundary adherence*

As expected by design, every one of the 360 optimised conditions, and all 72 baseline conditions, achieves 100 % adherence to the encoded empirical bounds and torque-ordering rule after rejection sampling. The constraint filter operates independently of the raw generator output quality: at the rejection rates observed here, oversampling by 3× and filtering was always sufficient to fill the target count. This confirms that the rejection-

sampling wrapper is an effective, architecture-independent mechanism for enforcing manufacturing constraints.

### *5.2 Impact of hyperparameter optimisation*

This is the central finding. Figure 2 and Table 2 compare the baseline, using default parameters and a single seed, against the optimised configuration, using Optuna-tuned parameters averaged over five seeds, at 1× scale with matched target sizes.

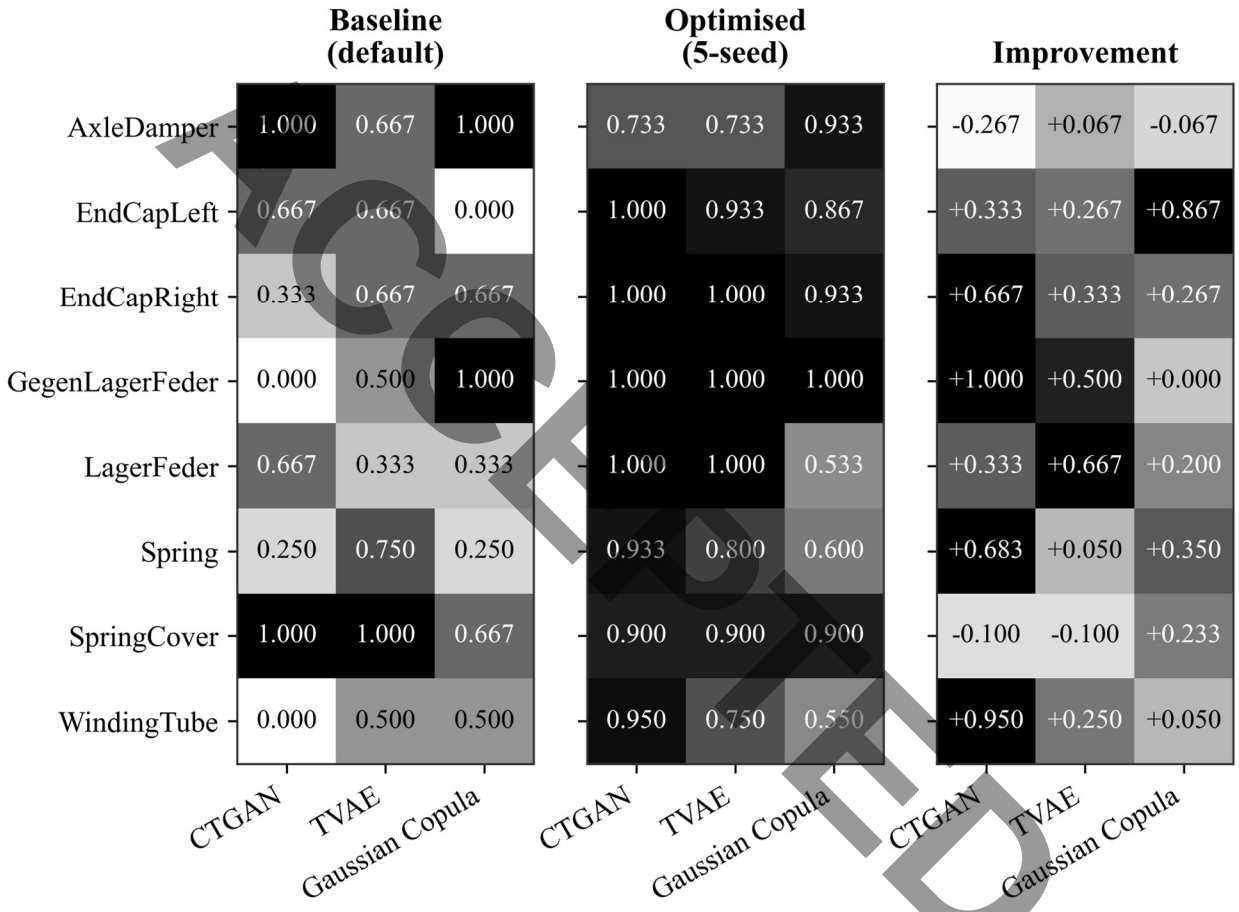


*Figure 2. Baseline versus optimised Kolmogorov–Smirnov pass rate by component and model.*

*Table 2. Kolmogorov–Smirnov pass rate: baseline versus optimised (1× scale).*

| Model | Baseline | Optimised (mean ± std) | Δ |
|---|---|---|---|
| CTGAN | 49.0 % | 94.0 % ± 9.2 % | +45.0 pp |
| TVAE | 63.5 % | 89.0 % ± 11.4 % | +25.5 pp |
| Gaussian Copula | 55.2 % | 79.0 % ± 19.4 % | +23.8 pp |

Optimisation improves the performance of every model by at least 20 percentage points. Gaussian Copula, which performed moderately at default settings, increases its average pass rate to nearly 80 %. CTGAN exhibits the most substantial improvement, increasing from a 49.0 % baseline pass rate to a 94.0 % average across all components.

### 5.3 Component-level analysis

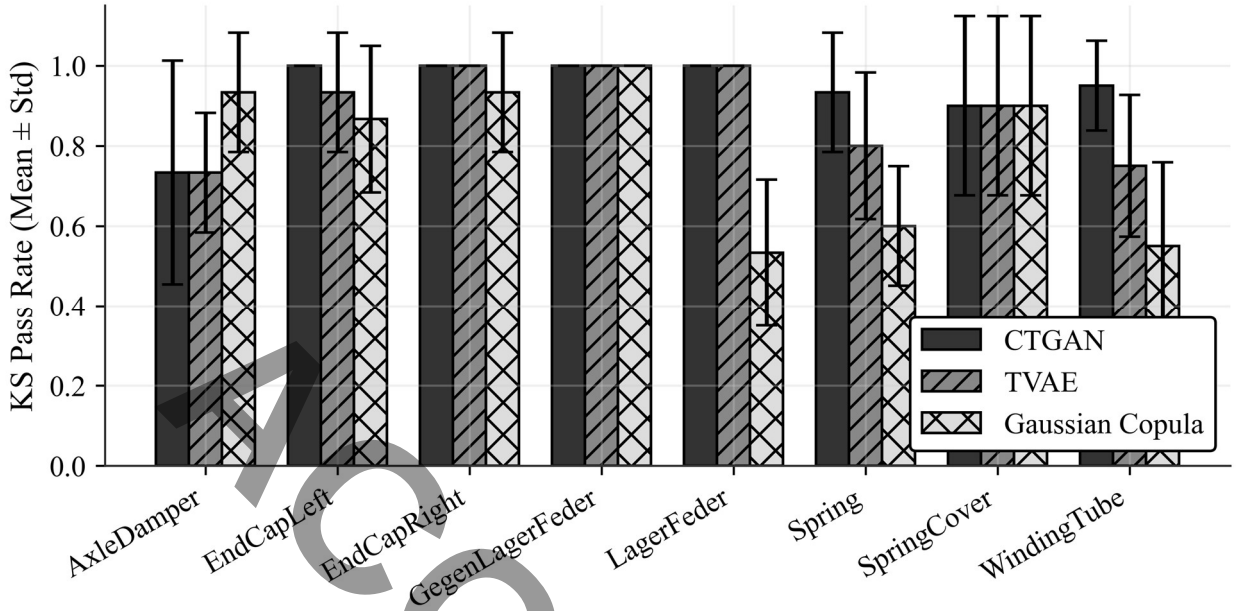


*Figure 3. Optimised Kolmogorov–Smirnov pass rate per component.*

*Table 3. Optimised Kolmogorov–Smirnov pass rate by component (1× scale, five-seed mean).*

| Component | CTGAN | TVAE | Gaussian Copula | Best performance |
|---|---|---|---|---|
| AxleDamper | 0.733 | 0.733 | 0.933 | GC |
| EndCapLeft | 1.000 | 0.933 | 0.867 | CTGAN |
| EndCapRight | 1.000 | 1.000 | 0.933 | CTGAN / TVAE |
| GegenLagerFeder | 1.000 | 1.000 | 1.000 | CTGAN / TVAE / GC |
| LagerFeder | 1.000 | 1.000 | 0.533 | CTGAN / TVAE |
| Spring | 0.933 | 0.800 | 0.600 | CTGAN |
| SpringCover | 0.900 | 0.900 | 0.900 | CTGAN / TVAE / GC |
| WindingTube | 0.950 | 0.750 | 0.550 | CTGAN |

Figure 3 and Table 3 illustrate that with optimised hyperparameters, both CTGAN and TVAE achieve a perfect 1.000 pass rate on several components, including EndCapRight, GegenLagerFeder and LagerFeder. Gaussian Copula performs best on the AxleDamper component, where its closed-form estimation maintains a slight advantage. CTGAN, which had the lowest baseline performance, now leads or ties on seven out of eight components, demonstrating the high sensitivity of adversarial architectures to hyperparameter tuning on small datasets.

LagerFeder proved difficult in the baseline evaluation, with an average pass rate of 0.19, due to extreme kurtosis ($k = 54.65$) in the weight measurements. After optimisation, both CTGAN and TVAE reach a pass rate of 1.000 on this component, likely due to appropriately tuned network capacities and regularisation parameters.

### 5.4 Scaling factor effects

Figure 4 shows that, as in the baseline, pass rates drop at larger scales (5× and 10×) while the Kolmogorov–Smirnov statistics improve.

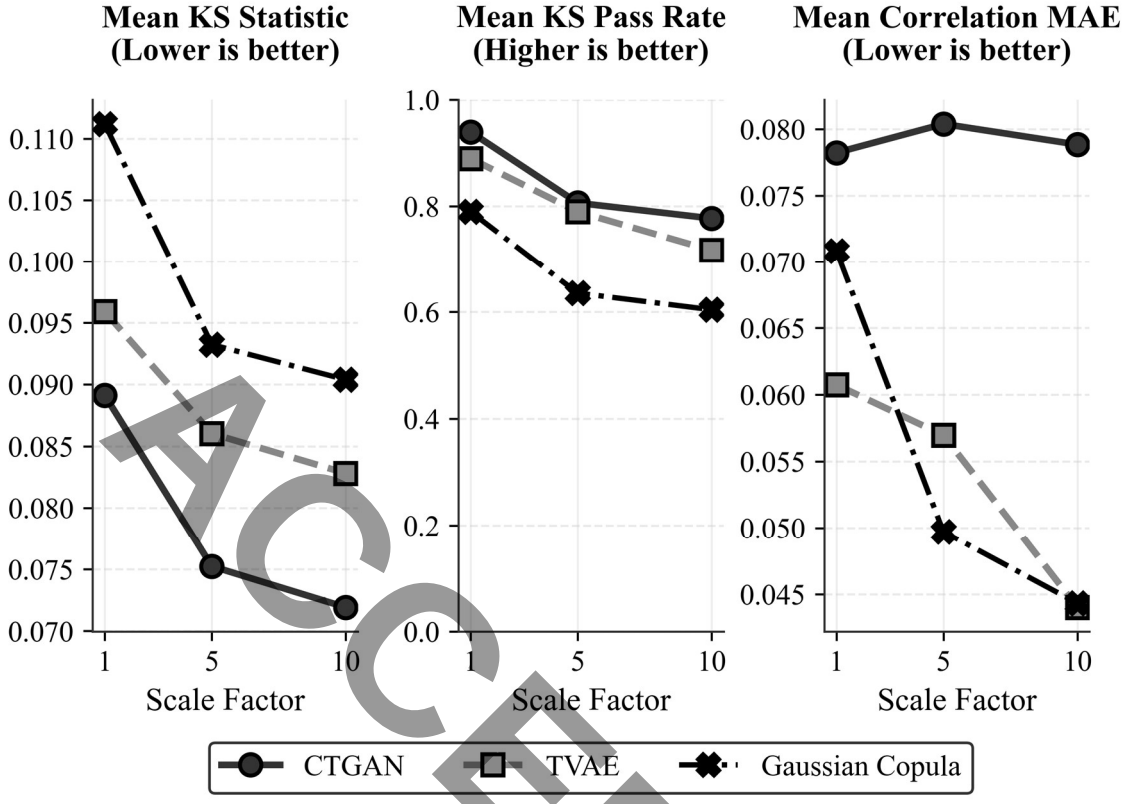


*Figure 4. Scaling effects on the evaluation metrics.*

This is a statistical-power artefact: the Kolmogorov–Smirnov test grows more sensitive with sample size, rejecting the null hypothesis of identical distributions for ever-smaller deviations. When comparing augmented datasets, practitioners should consider the magnitude of the Kolmogorov–Smirnov statistic rather than the binary pass or fail verdict.

### *5.5 Seed stability*

The multi-seed protocol reveals meaningful variance. The standard deviation of TVAE across seeds ranges from 0.000, indicating perfect consistency on EndCapRight, GegenLagerFeder and LagerFeder, to 0.200 on SpringCover, suggesting that seed sensitivity tracks component complexity. CTGAN shows the highest variance, up to 0.249 on AxleDamper, consistent with the well-known instability of adversarial training on small samples.

## 6. Discussion

The baseline evaluation indicated clear initial performance trends: TVAE models showed consistent performance, Gaussian Copula followed closely, while CTGAN clearly lagged. Hyperparameter optimisation complicates this perspective. When each model is parameterised optimally for the specific component, performance gaps narrow considerably, and CTGAN frequently overtakes the alternative models. This highlights that model selection without hyperparameter tuning can lead to suboptimal decisions, particularly for sample sizes below a few hundred observations.

From a deployment standpoint, the rejection-sampling approach remained robust across all 360 conditions. Because the filter guarantees constraint compliance by construction, boundary adherence is ensured without modifying internal generator architectures. For the empirical constraints in this dataset, comprising 23 range

rules and one relational rule, the processing overhead was small and concentrated on a single component: across the 360 optimised conditions, the seven components governed by range constraints alone showed 0 % rejection, while AxleDamper, the only component with a relational rule, averaged 0.53 % rejection, with a maximum of 4.87 % for CTGAN at 10× scale. The overall mean rejection rate was 0.066 %, so the 3× oversampling factor is more than sufficient for this constraint set.

Several limitations remain. The constraints are derived from observed data ranges plus one relational rule on AxleDamper torque, not from engineering drawings or tolerance specifications. Tighter design-sheet tolerances would increase rejection rates and may require re-tuning the oversampling factor. Cross-component tolerance-stackup constraints are also not encoded. A Train-on-Synthetic-Test-on-Real evaluation was not performed, so downstream task utility remains unverified and constitutes the primary direction for future work. The root-mean-square compression applied to WindingTube discards spatial structure that may matter for certain failure modes. The fidelity assessment relies on the univariate Kolmogorov–Smirnov test together with a Pearson correlation summary; incorporating multivariate metrics such as maximum mean discrepancy would strengthen the quality assessment. Finally, the results come from a single roller-slide door assembly with eight components, so generalisation to other automotive subsystems requires replication.

## 7. Conclusion

A constrained generative framework was built and tested on an eight-component automotive roller-slide door assembly, covering 360 experimental conditions across three models, eight components, three scales and five seeds. Bayesian hyperparameter optimisation proved essential: it lifted CTGAN from 49.0 % to 94.0 % Kolmogorov–Smirnov pass rate, TVAE from 63.5 % to 89.0 %, and Gaussian Copula from 55.2 % to 79.0 %, overturning baseline rankings that suggested simpler models always win on small data. Rejection sampling delivered 100 % adherence to the encoded empirical bounds and torque-ordering rule across every condition, without requiring changes to the generator architectures. Future work will focus on Train-on-Synthetic-Test-on-Real evaluation to validate downstream utility and on extending the constraint formulation to engineering tolerance specifications.

## References

[1] V. Buggineni, C. Chen, and J. Camelio, “Enhancing manufacturing operations with synthetic data: A systematic framework for data generation, accuracy, and utility,” Front. Manuf. Technol., vol. 4, 2024, doi: 10.3389/fmtec.2024.1320166.

[2] Y. Zhou, M. R. Bouadjenek, J. R. Wells, and S. Aryal, “Handling incomplete heterogeneous data using a data-dependent kernel,” 2025, arXiv:2501.04300.

[3] D. Meyer, T. Nagler, and R. J. Hogan, “Copula-based synthetic data augmentation for machine-learning emulators,” Geosci. Model Dev., vol. 14, no. 8, pp. 5205–5215, 2021, doi: 10.5194/gmd-14-5205-2021.

[4] F. Benali, D. Bodénès, N. Labroche, and C. de Runz, “MTCopula: Synthetic complex data generation using copula,” in Proc. 23rd Int. Workshop on Design, Optimization, Languages and Analytical Processing of Big Data (DOLAP), CEUR Workshop Proc., vol. 2840, 2021.

[5] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, “Modeling tabular data using conditional GAN,” in Proc. Adv. Neural Inf. Process. Syst. (NeurIPS), vol. 32, 2019, pp. 7335–7345. [Online]. Available: arXiv:1907.00503

[6] N. Patki, R. Wedge, and K. Veeramachaneni, “The synthetic data vault,” in Proc. IEEE Int. Conf. Data Sci. Adv. Analytics (DSAA), 2016, pp. 399–410, doi: 10.1109/DSAA.2016.49.

[7] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, “Optuna: A next-generation hyperparameter optimization framework,” in Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD), 2019, pp. 2623–2631, doi: 10.1145/3292500.3330701.

[8] P. Yadav et al., “Rigorous experimental analysis of tabular data generated using TVAE and CTGAN,” Int. J. Adv. Comput. Sci. Appl., vol. 15, no. 4, 2024, doi: 10.14569/IJACSA.2024.01504125.

[9] Z. Zhao, A. Kunar, H. Van der Scheer, R. Birke, and L. Y. Chen, "CTAB-GAN: Effective table data synthesizing," in Proc. 13th Asian Conf. Mach. Learn. (ACML), PMLR, vol. 157, 2021, pp. 97–112. [Online]. Available: arXiv:2102.08369

[10] T. Afonja, D. Chen, and M. Fritz, "MargCTGAN: A 'marginally' better CTGAN for the low sample regime," 2023, arXiv:2307.07997.

[11] V. Shulakov, "High-quality tabular data generation using post-selected VAE," 2024, arXiv:2407.13016.

[12] K. Karra and L. Mili, "Hybrid copula Bayesian networks," Int. J. Optim. Control: Theories Appl., vol. 6, no. 2, pp. 107–125, 2016.

[13] M. C. Stoian, S. Dyrmishi, M. Cordy, T. Lukasiewicz, and E. Giunchiglia, "How realistic is your synthetic data? Constraining deep generative models for tabular data," in Proc. Int. Conf. Learn. Representations (ICLR), 2024. [Online]. Available: arXiv:2402.04823

[14] B. Schindler and T. Schmid, "Physics-aware normalizing flows: Leveraging electric circuit models in adversarial learning," in Proc. 32nd Eur. Symp. Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN), 2024, doi: 10.14428/esann/2024.es2024-177.

[15] W. Niu et al., "FEST: A unified framework for evaluating synthetic tabular data," in Proc. 11th Int. Conf. Inf. Syst. Security Privacy (ICISSP), 2025, doi: 10.5220/0013383700003899.

[16] M. Hernandez et al., "Comprehensive evaluation framework for synthetic tabular data in health: Fidelity, utility and privacy analysis of generative models with and without privacy guarantees," Front. Digit. Health, vol. 7, 2025, doi: 10.3389/fdgth.2025.1576290.