CS787 : Generative AI

Modeling Tabular Data using Conditional GAN

Group Members:
Kshitij Pratap Singh (220554)
Sumit Neniwal (221103)
Kumar Gaurav Prakash (220560)
Akshit Shukrawal (220107)
M.K.S. Roshan (220633)

Professors:
Arnab Bhattacharya
Subhajit Roy

06.10.2025

1 Objective

After reviewing multiple ideas and research papers, we selected the NeurIPS 2019 paper titled *"Modeling Tabular Data using Conditional GAN"*. This work addresses the difficulty of applying deep generative models to tabular data, which typically includes discrete + continuous features, multi-modal distributions, and imbalanced categories.

To overcome these challenges, the paper proposes the **Conditional Tabular GAN (CTGAN)**, which includes:

- Conditional generator with training-by-sampling for imbalanced discrete variables.
- Mode-specific normalization using Gaussian Mixture Models.
- Use of WGAN-GP + PacGAN to reduce mode collapse.

Relevant links:

- CTGAN Repository: `https://github.com/sdv-dev/CTGAN/tree/main/ctgan`
- Research Paper: `http://papers.neurips.cc/paper/8953-modeling-tabular-data-pdf`

2 Introduction

Synthetic tabular data generation is critical for privacy-preserving machine learning and data-limited scenarios. However, unlike images or text, tabular data is challenging due to:

- Mixed categorical and continuous variables
- Non-Gaussian and multi-modal distributions
- Imbalanced categorical classes

CTGAN addresses these issues with:

- Mode-specific normalization using GMMs
- Conditional sampling for minority categories
- Fully connected GAN architecture with WGAN-GP and PacGAN

3 Model

CTGAN consists of a generator and a critic. Its key innovations include:

Mode-Specific Normalization

Continuous features are normalized using Gaussian Mixture Models instead of min–max normalization, enabling multi-modal modeling.

Conditional Generator + Training-by-Sampling

The generator is conditioned on discrete column values. During training, minority classes are oversampled to avoid mode collapse.

Network Architecture

- Fully connected generator and critic networks
- WGAN-GP loss ensures stable training
- PacGAN helps prevent mode collapse

4 Work Done

4.1 Novality

4.1.1 Exploration of Kernel Density Estimation (KDE) for Normalization

The baseline CTGAN model employs a Variational Gaussian Mixture Model (VGMM) for its mode-specific normalization. This is a parametric approach that inherently assumes each mode within a continuous variable's distribution can be effectively modeled by a Gaussian (bell-curve) shape.

Our team hypothesized that this assumption might be overly restrictive for our specific dataset, which features several continuous columns with heavily skewed or complex, non-Gaussian distributions.

As a novel exploration, we implemented an alternative normalization strategy using Kernel Density Estimation (KDE). The primary rationale was:

Distributional Flexibility: KDE is a non-parametric method. It makes no prior assumptions about the shape of the distributions and can faithfully model complex, arbitrary, and multimodal shapes. We believed this would capture the true data characteristics more accurately.

Results and Analysis Counter to our hypothesis, our experimental results showed that using the KDE-based normalization led to a slight degradation in performance on our downstream machine learning efficacy metrics compared to the original VGMM implementation.

We attribute this outcome to a fundamental incompatibility with the CTGAN generator's architecture:

1. Lack of Direct Parameterization: The CTGAN generator is explicitly designed to receive a two-part input for each continuous value: a one-hot vector identifying the mode and a normalized scalar representing the value within that mode. The VGMM directly provides the necessary parameters for this (mode centers η_k, standard deviations ϕ_k, and probabilistic mode assignments).

2. Heuristic Failure: KDE only produces a smooth density curve. To extract the "modes" required by the generator, we had to add a secondary, heuristic process to find local peaks (mode centers) and valleys (mode boundaries). This process proved less stable and robust than the direct, model-based clustering of the VGMM.

In conclusion, while the VGMM's Gaussian assumption is a limitation, its ability to provide clear, parameterized clusters is more critical for the CTGAN's learning process than the high-fidelity density representation offered by KDE.

5 Experiment Setup

5.1 Dataset

Adult Dataset (UCI Repository):

- Rows: $\sim$48,000
- 6 continuous and 8 categorical columns
- Binary income classification: $>$50K / $<$50K

5.2 Evaluation Method

We adopted the **Machine Learning Efficacy** metric from the paper:

- Train classifier on synthetic data
- Test classifier on real test split

If synthetic data is realistic, accuracy/F1 should match models trained on real data.

6 Results

6.1 Reproduction

After training CTGAN on the Adult dataset:

- Synthetic data preserved key statistical properties
- Generated samples successfully modeled categorical and continuous distributions

We trained the following classifiers:

- Logistic Regression
- Random Forest
- AdaBoost
- MLP
- Decision Tree

6.2 Evaluation Metrics

Metric	LR	RF	AdaBoost	MLP	DT
Accuracy	0.8386	0.8218	0.8269	0.8148	0.7844
F1 Score	0.6096	0.5495	0.5685	0.5986	0.5322
Precision	0.7297	0.7019	0.7109	0.6260	0.5572
Recall	0.5234	0.4515	0.4736	0.5736	0.5094
ROC AUC	0.8850	0.8722	0.8753	0.8497	0.6905

6.3 Comparison to Reported Results

Compared to the original CTGAN paper:

- Our classifier results trained on synthetic data show slightly lower F1.
- Accuracy and ROC-AUC remain high (>0.84), except DT.
- Trends match the paper's claims.

CTGAN successfully captures core structure but misses some fine-grained patterns, explaining the F1 drop.

7 Conclusion

Through this reproduction, we verified:

- CTGAN effectively models mixed-type tabular data.
- Synthetic data enables reasonable model performance on real test sets.
- Results closely follow the original paper's trends, validating CTGAN as a strong baseline.

CTGAN remains a powerful model for synthetic tabular data generation, especially under privacy constraints.