Exploring GAN Variants for Balancing Imbalanced Datasets

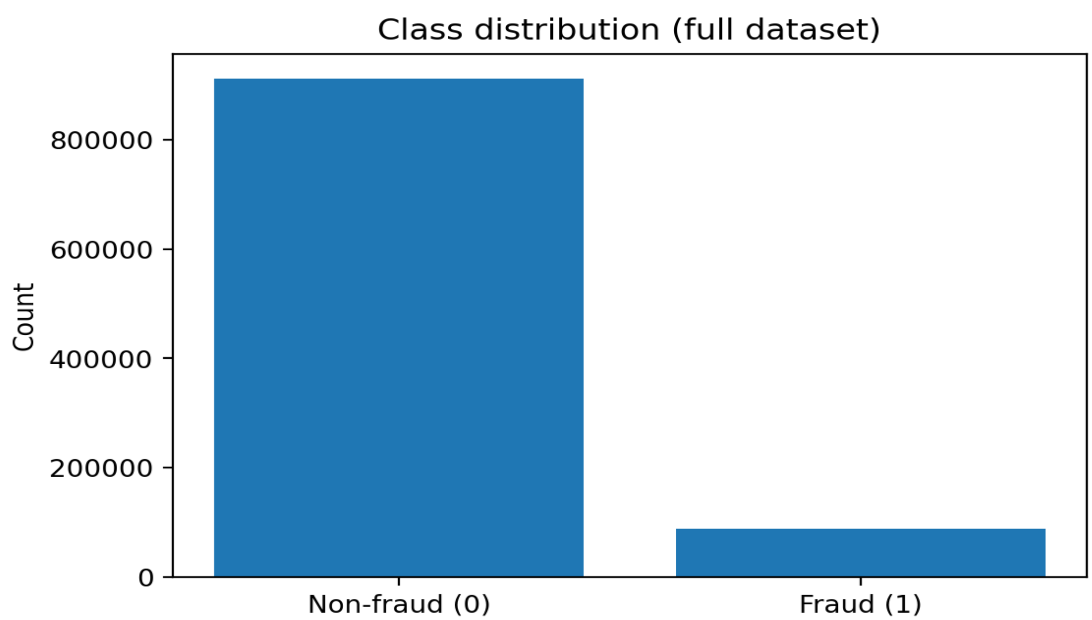
Dataset: card_transdata.csv (Fraud Detection) • Generated: 2026-01-09

1. Problem Statement

Class imbalance is common in fraud detection: legitimate transactions (majority class) far outnumber fraudulent ones (minority class). Standard classifiers may achieve high accuracy while still missing many fraud cases. This project evaluates whether GAN-based minority oversampling can improve minority-class detection by generating synthetic fraud samples and balancing the training data.

2. Dataset and Imbalance Analysis

The dataset contains 1,000,000 transactions with 7 input features and a binary label (fraud). Non-fraud = 912,597 (91.26%), Fraud = 87,403 (8.74%).



3. Methodology

We train GANs using only the minority-class (fraud=1) samples from the training split. After training, each GAN generates synthetic fraud samples to match the number of majority samples, producing a balanced training set. A downstream classifier is trained and evaluated under: (1) original imbalanced data, (2) balanced with Vanilla GAN, and (3) balanced with a GAN variant (WGAN-GP). LSGAN is also reported as an extra variant.

Preprocessing.

Features are scaled to [-1, 1] (MinMax). The generator outputs are inverse-scaled back to the original feature space; binary features are rounded back to {0,1}. The test set is never augmented (prevents leakage).

GAN variants.

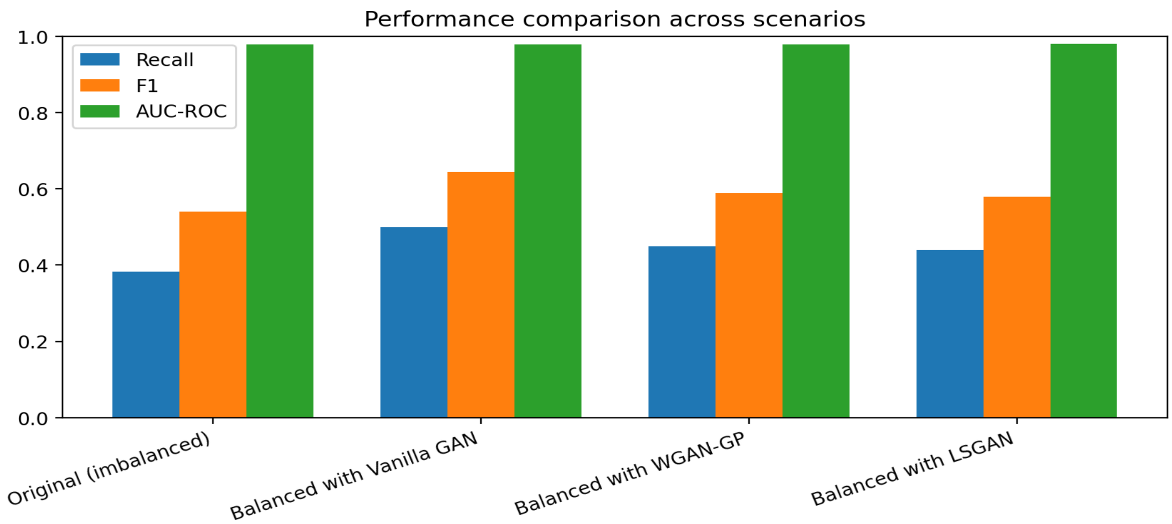
• Vanilla GAN: BCE loss (sigmoid discriminator) • WGAN-GP: Wasserstein loss + gradient penalty (critic) • LSGAN: least-squares loss (MSE)

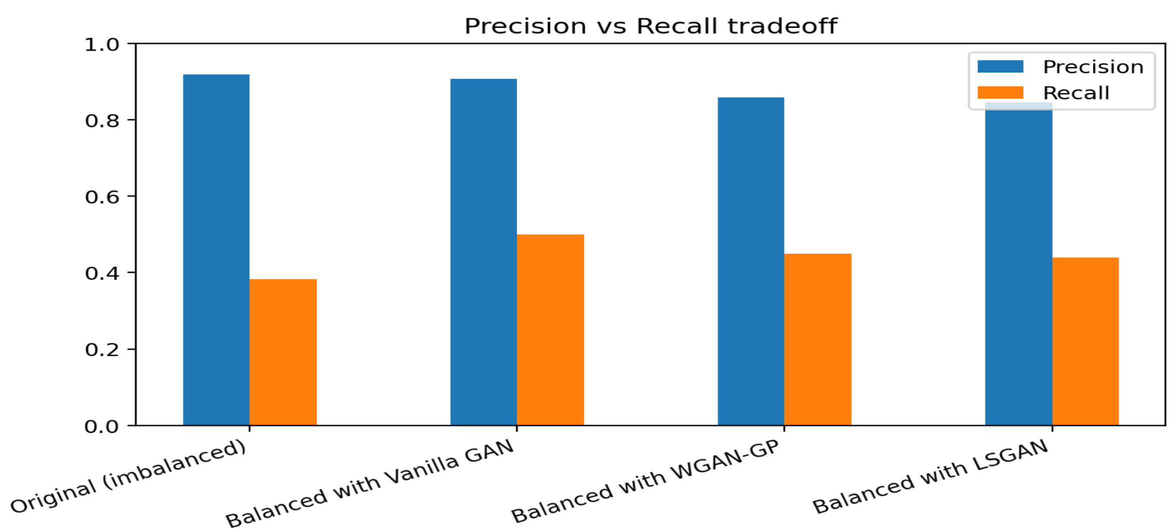
GAN	Main idea	Benefit
Vanilla GAN	Standard adversarial training	Baseline oversampling method
WGAN-GP	Wasserstein distance + gradient penalty	Stability, smoother training
LSGAN	Least-squares loss	More stable gradients than BCE

4. Experimental Setup and Evaluation

Downstream model: Logistic Regression (kept constant across scenarios). Metrics: Accuracy, Precision, Recall, F1-score, AUC-ROC, and Confusion Matrix.

Scenario	Acc	Prec	Recall	F1	AUC	TN	FP	FN	TP
Original (imbalanced)	0.94300	0.917808	0.382857	0.540323	0.978908	3638	12	216	134
Balanced with Vanilla GAN	0.95175	0.906736	0.500000	0.644567	0.978791	3632	18	175	175
Balanced with WGAN-GP	0.94525	0.857923	0.448571	0.589118	0.978524	3624	26	193	157
Balanced with LSGAN	0.94400	0.846154	0.440000	0.578947	0.979361	3622	28	196	154

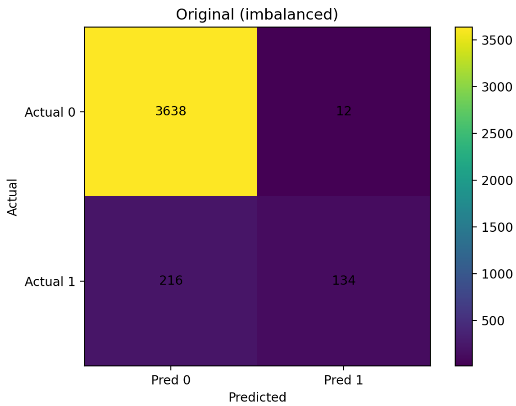




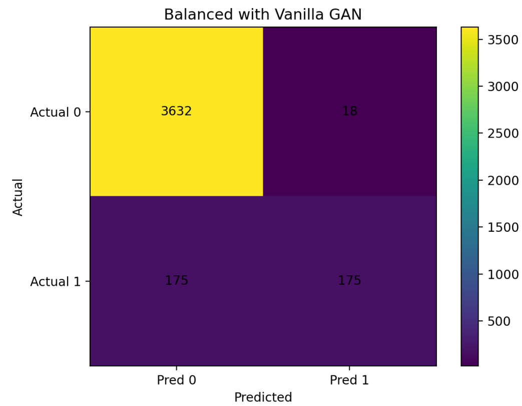
5. Confusion Matrix Comparison

For fraud detection, the most important change after balancing is reducing FN (missed fraud) and increasing TP (caught fraud). Below are confusion matrices for each scenario.

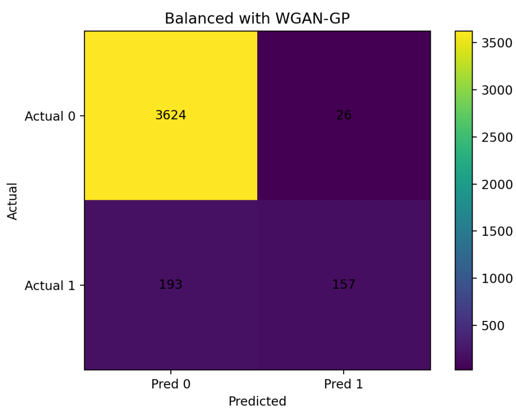
Original (imbalanced)



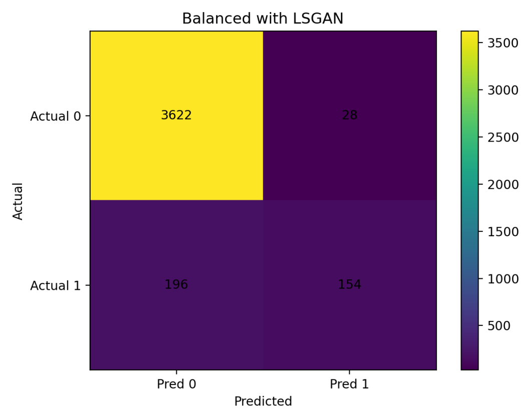
Balanced with Vanilla GAN



Balanced with WGAN-GP



Balanced with LSGAN



6. Discussion and Conclusion

Balancing with Vanilla GAN produced the best minority-class performance. Recall increased from 0.383 to 0.500 and F1 increased from 0.540 to 0.645. TP increased from 134 to 175, while FN decreased from 216 to 175.

The trade-off is a small increase in false positives (FP 12 → 18) and a slight precision decrease (0.918 → 0.907). Overall, the improved recall/F1 indicates better detection of fraud cases.

WGAN-GP and LSGAN also improved over the baseline (WGAN-GP F1=0.589, LSGAN F1=0.579) but were slightly below Vanilla GAN in this run.

Suggested improvements (optional):

• Increase GAN epochs and tune learning rates. • Try a stronger classifier (MLP / Gradient Boosting). • Add a traditional baseline such as SMOTE. • Repeat experiments with multiple random seeds for robustness.