1. Problem Statement

Imbalanced datasets, especially in healthcare, pose a critical challenge to machine learning models. The underrepresentation of the minority class often results in models that are biased and perform poorly in predicting critical events like strokes. This project explores the use of Generative Adversarial Networks (GANs)—including Vanilla GAN, Conditional GAN (CGAN), and Wasserstein GAN (WGAN)—to generate synthetic data for the minority class and improve classification performance using a Multilayer Perceptron (MLP).

2. Description of Dataset & Imbalance Analysis

- Dataset: Stroke Prediction Dataset
- Features:
    - Demographics: gender, age, marital status, work type, residence type, smoking status
    - Health metrics: hypertension, heart disease, avg glucose level, BMI
    - Target Variable: **stroke** (0 = No, 1 = Yes)
- Preprocessing:
    - Dropped rows with missing BMI values
    - One-hot encoded categorical variables
    - Normalized numerical features
- Class Distribution:
    - Stroke = 1 (Minority): 209 samples
    - Stroke = 0 (Majority): 4700 samples
    - Imbalance Ratio: ~1:22

3. Details of GAN Architectures & Training

- **Vanilla GAN**
    - Generator: MLP (noise vector → synthetic features)
    - Discriminator: MLP (input features → real/fake)
    - Loss: Binary Cross-Entropy
    - Epochs: 200
    - Batch Size: 20

- **Conditional GAN (CGAN)**
    - Conditioning on class label (stroke = 1)
    - Generator & Discriminator both receive class labels
    - Helps target specific class generation

- **Wasserstein GAN (WGAN)**
  - Uses Wasserstein loss and weight clipping
  - Stabilizes training and improves gradient flow
  - Produces more diverse and realistic synthetic samples

4. Classifier Setup and Evaluation

- **Classifier Selection**

  We used the **MLPClassifier** from the **scikit-learn** library to perform binary classification on the stroke dataset. MLPClassifier is a feedforward neural network that supports backpropagation and is well-suited for modeling complex relationships in tabular data.

- **Dataset Preparation**

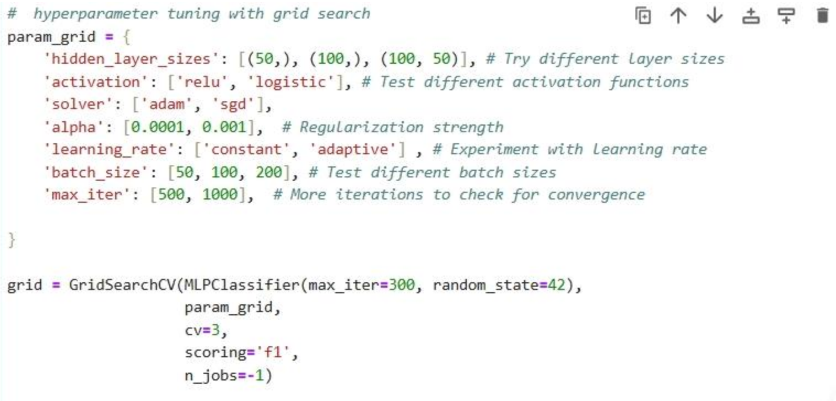
  The classifier was trained and evaluated on the following datasets:

  - Original dataset (after dropping records with missing BMI values).
  - Synthetic dataset generated by Vanilla GAN**.**
  - Synthetic dataset generated by WGAN**.**
  - Synthetic dataset generated by CGAN**.**

  Each dataset had the same feature structure and binary target variable stroke.

- **Hyperparameter Tuning using Grid Search**

To select the best hyperparameters for the classifier, we applied **Grid Search** using GridSearchCV with 3-fold cross-validation



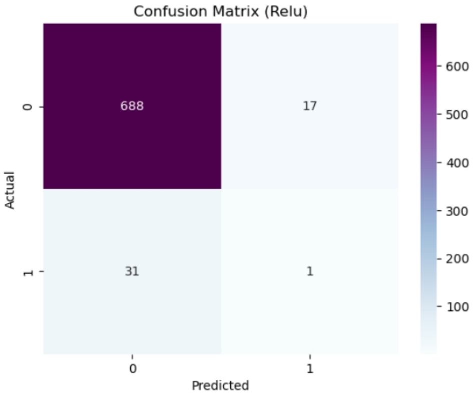
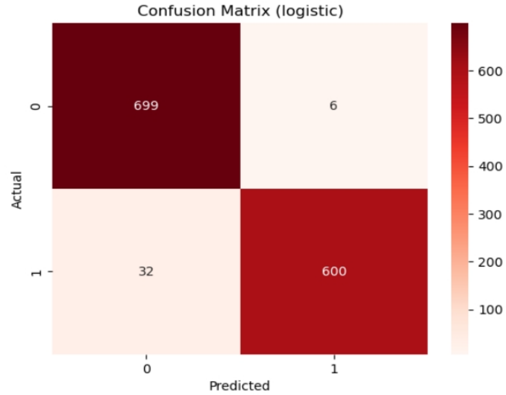
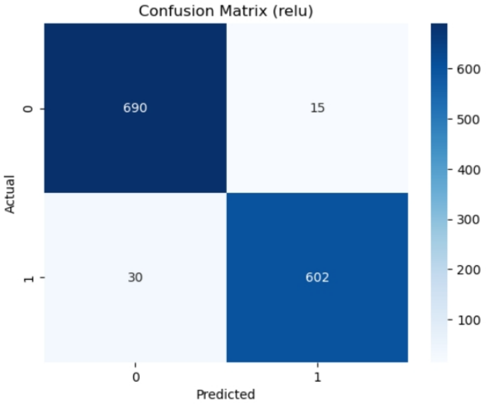
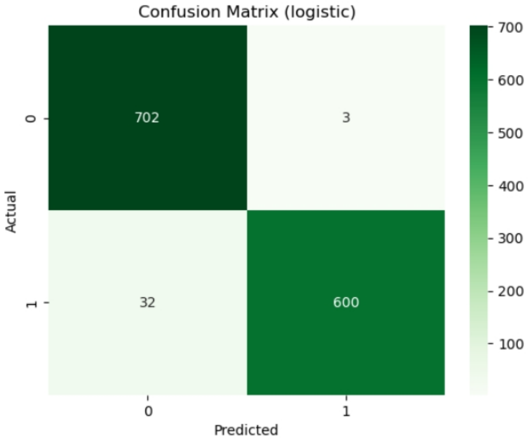
```
#  hyperparameter tuning with grid search
param_grid = {
    'hidden_layer_sizes': [(50,), (100,), (100, 50)], # Try different layer sizes
    'activation': ['relu', 'logistic'], # Test different activation functions
    'solver': ['adam', 'sgd'],
    'alpha': [0.0001, 0.001],  # Regularization strength
    'learning_rate': ['constant', 'adaptive'] , # Experiment with learning rate
    'batch_size': [50, 100, 200], # Test different batch sizes
    'max_iter': [500, 1000],  # More iterations to check for convergence

}

grid = GridSearchCV(MLPClassifier(max_iter=300, random_state=42),
                    param_grid,
                    cv=3,
                    scoring='f1',
                    n_jobs=-1)
```

- **Evaluation Metrics**: Accuracy, Precision, Recall, F1-Score

  Confusion Matrix

Original	
Vanilla GAN	
WGAN	
CGAN	

- **Training Scenarios:**

  Original imbalanced dataset

  Dataset augmented with Vanilla GAN

  Dataset augmented with WGAN

  Dataset augmented with CGAN

5. Results & Comparisons

Model	Accuracy	Precision	Recall	F1-Score
Original	93.49%	50.62%	50.36%	50.31%
Vanilla GAN	97.16%	97.32%	97.04%	97.14%
WGAN	96.63%	96.70%	96.56%	96.62%
CGAN	97.38%	97.57%	97.26%	97.37%

The much higher precision, recall, and F1 for all GAN-augmented models strongly suggest that the GANs produced useful synthetic minority data that improved classifier performance.

6. Observations and Conclusions

- All GAN-based data augmentation methods significantly enhanced the MLP classifier’s ability to detect stroke cases compared to using the original dataset.

- **CGAN** achieved the **highest overall performance**, with an accuracy of **97.38%**, precision of **97.57%**, recall of **97.26%**, and an F1-score of **97.37%**. This superior performance is likely due to CGAN's ability to generate stroke-specific samples conditioned on class labels, effectively addressing class imbalance.

- **Vanilla GAN** also yielded strong results (**97.16% accuracy**, **97.14% F1-score**), indicating its effectiveness despite lacking the conditional generation mechanism.

- **WGAN**, known for its stable training and sample quality, performed slightly lower than CGAN but still outperformed the baseline, achieving **96.63% accuracy** and **96.62% F1-score**.

- In contrast, the **original model** trained without synthetic data showed considerably weaker performance (**50.31% F1-score**), highlighting the challenges posed by the severe class imbalance.

- Preprocessing steps such as **dropping missing BMI values** and applying **one-hot encoding** contributed to improved data quality and model learning.

Conclusion:

GAN-based augmentation, particularly through **CGAN** and **WGAN**, proves to be a robust and effective solution for addressing extreme class imbalance in healthcare datasets, leading to more accurate and reliable stroke prediction.

.