Datasets

Three imbalanced datasets are used for this study. The datasets are referred to as breast cancer, credit card fraud detection, and customer churn./

Data Cleaning

Dataset Overview

Loading Data: Each dataset was loaded into a DataFrame.

Missing Values: Checked for missing values and handled them using imputation or removal based on the extent of missingness.

Data Types: Ensured correct data types for each feature.

Outliers: Identified and addressed outliers using statistical methods or domain knowledge.

Exploratory Data Analysis (EDA)

Class Distribution

The class distribution in each dataset was visualized using count plots. This helped to understand the extent of imbalance in the datasets.

Feature Analysis

Descriptive Statistics: Calculated mean, median, standard deviation, and other statistics for numerical features.

Correlation Analysis: Examined correlations between features to identify potential multicollinearity.

Visualization: Used histograms, box plots, and scatter plots to visualize feature distributions and relationships.

Machine Learning Algorithms

Five classifiers were applied to the datasets to evaluate their baseline performance:

Random Forest: An ensemble method that combines multiple decision trees to improve performance.

Decision Tree: A simple yet interpretable model that splits data based on feature values.

Naive Bayes: A probabilistic classifier based on Bayes’ theorem with an assumption of independence between features.

Logistic Regression: A linear model used for binary classification problems.

K-Nearest Neighbors (KNN): A non-parametric method that classifies instances based on the majority class among the k-nearest neighbors.

Baseline Performance

For each algorithm, we assessed the performance using metrics such as accuracy, precision, recall, F1-score, ROC-AUC, and PR-AUC. Cross-validation was used to ensure robust performance estimates.

Balancing Techniques

To address the class imbalance, three techniques were applied:

SMOTE (Synthetic Minority Over-sampling Technique):

Objective: To generate synthetic examples for the minority class.

Implementation: Applied SMOTE to the training data to balance class distribution.

One-Class SVM:

Objective: To identify anomalies by training on the majority class only.

Implementation: Used One-Class SVM for anomaly detection and classification.

Feature Engineering:

Objective: To enhance model performance by creating polynomial features.

Implementation: Applied polynomial feature generation to capture interactions between features and improve classification performance.

Results

Class Distribution After Balancing

Class distribution was visualized before and after applying balancing techniques to assess their effectiveness in addressing imbalance.

Performance Evaluation with Balancing Techniques

Each classifier was re-evaluated after applying the balancing techniques. Key metrics (accuracy, precision, recall, F1-score, ROC-AUC, PR-AUC) were compared to baseline results to assess improvements.