ML Project: Predicting Bank Term Deposit Subscriptions — Synthetic vs Real Data

Objective

The objective of the project is to explore and model customer subscription behavior for term deposits using both **synthetic** and **real** datasets. You will compare model performance when trained only on synthetic data, trained on a small amount of real data, and optionally when using **transfer learning** from synthetic to real data.

Datasets

The datasets for this project are part of an ongoing **Kaggle competition**:

- **Kaggle Competition:** Playground Series - Season 5, Episode 8
- **Real dataset:** Bank Term Deposit Subscription Dataset (Real)
- **Synthetic dataset:** Available in the competition's Dataset Link. Train and Test dataset are given. You can combine both the dataset for training
- **Test Dataset:** extract 20 % of dataset from Real Dataset for testing the prediction accuracy of your model.

Tasks

1. Data Understanding & Exploration

- Load both datasets and explore:
  - Feature types (categorical, numerical)
  - Missing values
  - Target distribution
- Compare synthetic vs real:
  - Summary statistics for numerical features
  - Category counts for categorical features
- Plot distributions side-by-side.

2. Data Preprocessing

Perform only those steps which are required.

- Handle missing values.
- Encode categorical features (One-Hot Encoding or Label Encoding).
- Scale numerical features (StandardScaler / MinMaxScaler).

3. Your Experimental Setup(s)

You should do the following experiments:

1) **Synthetic → Real (Direct Test)**
    - Train a model using only the synthetic dataset.
    - Test on the test dataset extracted from real dataset.
    - Purpose: See how well a model trained on synthetic data generalizes to real data.
2) **Small Real Data**
    - Train a model on the real dataset considering different data sizes.
    - Test on the test dataset.
    - Purpose: See how accuracy improves in adding more data.
3) **Transfer Learning**
    - Step 1: Train a model on the synthetic dataset.
    - Step 2: Fine-tune it using the small portion of real data. You can apply any transfer learning method.
        - For tree models: reuse engineered features and tuned hyperparameters from synthetic model, retrain on small real dataset.
        - For neural nets: load trained weights from synthetic training, continue training with small real dataset.
    - Step 3: Test on the test dataset.
    - Purpose: See if pretraining on synthetic helps when you have little real data.

4. Model Training

- You may choose Logistic Regression, Random Forest, XGBoost, CatBoost, LightGBM, or Deep Neural Network.
- You can choose more than one model for each setup.
- Evaluate using:
    - Accuracy

  - Precision
  - Recall
  - F1-score
  - ROC-AUC

5. Insights & Reporting

- How do synthetic and real datasets differ?
- How does performance change across chosen setups?
- Does transfer learning (if tried) improve results compared to direct training?

Deliverables

1. **Code (.py file)** with:
   - Data exploration
   - Preprocessing
   - Model training and evaluation
2. **Short Report** (2–3 pages) with:
   - Dataset comparison findings
   - Model results table
   - Key observations

Learning Outcomes

- Understand differences between synthetic and real data
- Work with imbalanced binary classification
- See the effects of dataset shift
- Practice minimal-data training and transfer learning
- Interpret model performance in a real-world context