Notebook Summary: Forecasting vs Data

Libraries Used

The following Python libraries were imported and used throughout the notebook:

- **pandas** → Data manipulation and preprocessing
- **numpy** → Numerical operations
- **matplotlib.pyplot** → Plotting and visualizations
- **seaborn (sns, sls)** → Data visualization and styling
- **sklearn (scikit-learn)**
    - `train_test_split` → Splitting train/test data
    - `classification_report`, `confusion_matrix` → Evaluation metrics
    - `accuracy_score`, `precision_score`, `recall_score`, `f1_score`, `roc_auc_score`, `roc_curve` → Model evaluation
    - `mean_absolute_error` → Error metric (time-series validation)
- **xgboost (xgb)** → Core machine learning model (predict refill / no refill)
- **prophet** → Time-series forecasting
- **os, pathlib, datetime** → File handling and date utilities

---

Functions Defined

- **`get_metrics(y_true, y_pred, label)`**
  → Custom evaluation function to compute and return metrics (accuracy, precision, recall, F1) for model predictions.

What Was Done in the Notebook

1. **Data Loading & Preprocessing**
    - Imported multiple datasets: `claims`, `prescribers`, `market`, `calendar`.
    - Cleaned and structured patient refill claims data.
    - Encoded categorical variables (specialty, region) using one-hot encoding.
2. **Feature Engineering**
    - Created features like **gap_days, fill_count, days of supply (dos), adherence flags, time frames**.
    - Mapped patients to quarters and computed refill-related features.
3. **Train-Test Split**
    - Used `train_test_split` with stratification to handle imbalanced classes.
    - Multiple variations were tried (`test_size=0.2, 0.1, 0.4`) for small and large datasets.
4. **Modeling with XGBoost**
    - Baseline model with default threshold (0.5).
    - Threshold-tuned model (0.4).
    - Class-weighted model (handling imbalance with `scale_pos_weight`).
5. **Evaluation**
    - Accuracy, precision, recall, F1 score computed.
    - Confusion matrix visualized.
    - ROC curves and AUC scores compared.
    - Patient-level prediction probabilities binned into 10% intervals (0–10%, 10–20%, …).

6. **Experimentation on Data Size**
    - **Small Dataset** (~500 patients, 2 years) vs **Large Dataset** (~5,000 patients, 9 years).
    - Results compared across baseline, threshold-tuned, and class-weighted models.
7. **Forecasting with Prophet**
    - Applied Prophet on refill counts for time-series prediction (experimental part).
8. **Final Deliverables**
    - Patient-level prediction table with refill probability and label.
    - Comparative results across dataset sizes and methods.
    - Insights into **how data size, thresholds, and class weighting impact results**.

In short:
This notebook is a **forecasting + classification experiment** using **synthetic patient refill data**, comparing **different dataset sizes** and **modeling approaches** (baseline, threshold-tuned, class-weighted).