**# Healthcare Readmission Prediction Pipeline**

This project builds a robust machine learning pipeline to predict patient readmission using the UCI Diabetic dataset. It includes preprocessing, model training, calibration, and interpretability analysis—designed for both technical and clinical stakeholders.

**## Dataset Overview**

- \*\*Source\*\*: UCI Machine Learning Repository – Diabetic Readmission Dataset

- \*\*Path\*\*: `C:/Users/USP/OneDrive/Desktop/Tulasi/Medical Assignment/diabetic\_data.csv`

- \*\*Initial Shape\*\*: ~100,000 rows × 50+ columns

**## Data Preprocessing**

**1.** **\*\*Missing Value Handling\*\***

- Replaced `'?'` with `np.nan`

- Dropped high-missing or low-utility columns: `weight`, `payer\_code`, `medical\_specialty`

- Dropped identifier columns: `encounter\_id`, `patient\_nbr`

- Imputed `race` with mode

- Dropped remaining rows with missing values

**2.** **\*\*Encoding\*\***

- Applied `LabelEncoder` to all categorical columns

**3. \*\*Final Shape\*\***

- ~70,000 rows × reduced feature set

**## Train-Test Split**

- \*\*Split Ratio\*\*: 80% training, 20% testing

- \*\*Stratification\*\*: Ensured class balance on `readmitted`

- \*\*Target Variable\*\*: `readmitted`

**## Model Training**

- \*\*Model\*\*: `XGBClassifier` (XGBoost)

- \*\*Parameters\*\*:

- `use\_label\_encoder=False`

- `eval\_metric='logloss'`

- `random\_state=42`

- \*\*Outcome\*\*: Model trained successfully on encoded features

**## Evaluation**

- \*\*Metric\*\*: Brier Score (measures calibration quality)

- \*\*Uncalibrated Brier Score\*\*: Printed after prediction

**## Calibration Curve**

- Used `calibration\_curve` from `sklearn`

- Visualized predicted vs. observed probabilities

- Included bin annotations for interpretability

- Compared against perfect calibration line

**## Interpretability with SHAP**

- Used `shap.Explainer` on trained XGBoost model

- Generated SHAP summary plot for test set

- Highlights top features influencing readmission predictions

**## Notes**

- Warnings suppressed for cleaner output

- All steps reproducible and trainer-proof

- Designed for clarity across technical and clinical audiences

**## Future Enhancements**

- Try `KNNImputer` for partial missing data

- Add model comparison (Random Forest, SVM)

- Integrate `CalibratedClassifierCV` for post-hoc calibration

- Package pipeline into reusable module