**Final Report: Predicting Hospital Readmission within 30 Days**

**1. Key Preprocessing Steps Taken:**

Our primary focus was to prepare the dataset for modeling by ensuring that all features were usable for the Logistic Regression model. Below are the key preprocessing steps:

* **Data Exploration**: We thoroughly explored the dataset, checking for any missing values and inspecting the distribution of key features like age and readmission status. Fortunately, no missing data was found.
* **Handling Categorical Variables**:
  + **Label Encoding**: For binary categorical variables like "Gender" and "Smoking Status," we applied Label Encoding to convert these categories into numeric values (0 or 1).
  + **One-Hot Encoding**: For multi-class categorical variables such as "Family History," "Activity Level," and "Recommended Health Tips," we applied One-Hot Encoding. This step created new binary columns representing each unique category, ensuring the model could handle these features.
* **Multi-label Binarization**: The "Medical History" column contained lists of medical conditions for each patient. We applied **MultiLabelBinarizer** to split this column into multiple binary columns, one for each possible condition (e.g., "Hypertension," "Diabetes").
* **Feature Scaling**: We used **StandardScaler** to scale numerical features like "Age," "Number of Previous Admissions," and "Days Since Last Discharge" to ensure they were on a comparable scale. This is important for Logistic Regression, which is sensitive to the magnitude of feature values.

**2. Model Choice:**

We selected **Logistic Regression** for the task of predicting whether a patient would be readmitted within 30 days. Here’s why:

* **Binary Classification Problem**: Logistic Regression is specifically designed for binary outcomes, making it a natural choice for this problem, where the goal is to classify patients as either "Readmitted" (1) or "Not Readmitted" (0).
* **Interpretable Model**: Logistic Regression provides easily interpretable results, with each feature’s coefficient representing its contribution to the likelihood of a patient being readmitted. This transparency is crucial in healthcare contexts, where understanding why a model makes specific predictions is important.
* **Handling Class Imbalance**: Given that most patients in the dataset were not readmitted, we used the **class\_weight='balanced'** option in Logistic Regression. This approach automatically adjusted the importance of the minority class (readmitted patients), improving the model’s sensitivity to detecting readmissions.

**3. Performance Metrics of the Model:**

The model’s performance was evaluated using key metrics such as **accuracy**, **precision**, **recall**, and **F1-score**. We assessed the model both before and after threshold tuning to prioritize recall, which is critical for this task.

**Model Performance (With Class Weighting)**:

* **Accuracy**: 46.67%
* **Precision**: 25.17%
* **Recall**: 42.53%
* **F1-Score**: 31.62%

**4. Suggested Improvements to the Model:**

1. **Further Threshold Tuning**:
   * Additional fine-tuning of the classification threshold could further improve recall without significantly affecting precision. This is particularly useful if we want to focus on reducing false negatives (missed readmissions) while balancing false positives.
2. **Feature Engineering**:
   * Creating new features or interaction terms (e.g., combining age and specific medical conditions) could help the model capture more complex relationships between variables, potentially improving performance.
3. **Cross-Validation**:
   * Using cross-validation would ensure that the model’s performance is consistent across different subsets of the data and reduce the risk of overfitting to the specific train-test split.

**Conclusion:**

We successfully built and evaluated a **Logistic Regression** model to predict hospital readmission within 30 days. After applying class weighting and threshold tuning, the model achieved better recall, making it effective at identifying patients likely to be readmitted. Further improvements could be made by fine-tuning the threshold, applying cross-validation, and exploring advanced techniques for handling class imbalance. The model is interpretable, efficient, and flexible, making it a suitable solution for this task.