Hospital Readmission Model

Overview

This notebook implements a machine learning pipeline for analyzing a healthcare dataset. The goal is to preprocess the dataset, handle missing values and duplicates, perform exploratory data analysis (EDA), and build classification models to predict patient conditions based on the given attributes.

1. Data Loading and Cleaning

- The dataset used is: `healthcare_dataset_with_new_rules.csv`
- Duplicates in the dataset are identified and dropped using `df.duplicated().sum()` and `df.drop_duplicates()`.
- Missing values are checked using `isnull().sum()` and appropriately handled.
- Unnecessary columns like `Unnamed: 0` and potentially others are dropped.

2. Exploratory Data Analysis (EDA)

- Various plots such as bar plots and count plots are used to understand the distribution of categorical variables like `smoking`, `alcohol`, `Stroke`, and `Diseases`.
- Relationships between features are explored using `seaborn` and `matplotlib`.
- Count plots help in visualizing class imbalance and feature correlations.

3. Data Preprocessing

- Categorical columns are label encoded using `LabelEncoder`.
- Features and target variable (`Diseases`) are separated.

- Dataset is split into training and testing using `train_test_split` (70% train, 30% test).

- Feature scaling is applied using `StandardScaler`.

4. Model Building

Two different models are built and evaluated:

Model 1: Logistic Regression

- Model is trained using `LogisticRegression()` from `sklearn.linear_model`.

- Performance is evaluated using accuracy, confusion matrix, and classification report.

Model 2: Random Forest Classifier

- Model is trained using `RandomForestClassifier()` from `sklearn.ensemble`.

- Similar evaluation metrics are used.

- Random Forest shows better performance compared to Logistic Regression.

5. Model Evaluation

- Accuracy scores, confusion matrices, and classification reports are generated for both models.

- Based on accuracy and F1-score, Random Forest performs better and is chosen as the preferred model.

6. Observations and Conclusion

- Data preprocessing and proper EDA helped in preparing a clean dataset for modeling.

- Random Forest is a more robust classifier for this problem.

- The notebook follows a standard ML workflow from data cleaning to model evaluation.

This documentation reflects the operations, decisions, and outcomes represented in the Learnathon_1.ipynb notebook.