Title: Prediction of House Rent Using Regression Models

Objective

To develop and evaluate a regression model that predicts house rent based on various housing features such as location, number of rooms, area (in square feet), and furnishing status.

Theory

Regression is a supervised machine learning technique used to model the relationship between a dependent variable and one or more independent variables. In this lab, we apply **linear regression** and optionally other regression techniques (e.g., Decision Tree Regression, Random Forest, etc.) to predict the rent of a house.

The goal is to identify a functional relationship between house features (like size, number of bedrooms, location) and the rent, allowing us to predict rent prices for new data.

Key Concepts:

- **Linear Regression**: Assumes a linear relationship between inputs and output.
- **Mean Squared Error (MSE)**: A common evaluation metric that measures average squared difference between predicted and actual values.
- **R^2 Score**: Indicates how well the model explains the variability of the target variable.

Procedure

1. **Data Collection**
    - Obtain a dataset related to house rentals. This can be downloaded from Kaggle or any open dataset source.
    - Example features: Location, Size (sqft), Bedrooms, Bathrooms, Furnishing Status, Rent.
2. **Data Preprocessing**
    - Load the dataset using Python (preferably with Pandas).
    - Handle missing values.
    - Convert categorical variables into numerical form using techniques like one-hot encoding.
    - Normalize or scale the data if needed.
3. **Exploratory Data Analysis (EDA)**
    - Plot graphs to understand relationships between features and rent.
    - Use correlation matrices to identify strongly related variables.

4. **Model Building**
    - Split the dataset into training and testing sets (e.g., 80:20).
    - Apply linear regression using scikit-learn.
    - Train the model on the training set.
5. **Model Evaluation**
    - Use the test set to make predictions.
    - Calculate performance metrics: MSE, RMSE, and R^2 score.
    - Optionally compare with other regression algorithms like Random Forest or Ridge Regression.
6. **Visualization**
    - Plot actual vs. predicted rents to visualize the accuracy.
    - Display regression line for linear models if appropriate.

Conclusion

In this lab, we successfully implemented a regression model to predict house rent. The model's performance was evaluated using R^2 and RMSE, and it demonstrated the ability to make accurate predictions based on housing features. This practical exercise highlighted the usefulness of regression analysis in solving real-world problems like property price estimation.

Title: Medical Diagnostics for Disease Detection Using Machine Learning

---

Objective

To design and implement a machine learning model for the detection of diseases based on patient data, using classification algorithms such as Logistic Regression, Decision Trees, or Support Vector Machines.

Theory

Machine learning (ML) in healthcare enables the development of predictive models that assist in early diagnosis and decision-making. Medical diagnostics using ML often involve **classification problems**, where the goal is to determine the presence or absence of a disease based on various clinical features.

Key Concepts:

- **Classification Algorithms**: Used to categorize input data into predefined labels (e.g., disease or no disease).

    - **Logistic Regression**: A statistical model for binary classification.
    - **Decision Trees**: A tree-like model used to make decisions based on feature values.
    - **Support Vector Machines (SVM)**: A classifier that finds the best boundary between classes.
- **Confusion Matrix**: Used to visualize the performance of classification models.
- **Evaluation Metrics**:
    - **Accuracy**: Percentage of correct predictions.
    - **Precision, Recall, F1-Score**: Useful for imbalanced datasets.
    - **ROC-AUC Curve**: Measures model performance across thresholds.

Procedure

1. **Data Collection**
    - Obtain a publicly available medical dataset (e.g., diabetes, breast cancer, heart disease from UCI or Kaggle).
    - Example features: age, blood pressure, glucose level, BMI, etc.
2. **Data Preprocessing**
    - Load data using Pandas.
    - Check for and handle missing values.
    - Encode categorical variables using label encoding or one-hot encoding.
    - Normalize numerical data if necessary.
3. **Exploratory Data Analysis (EDA)**
    - Analyze class distribution (e.g., number of positive vs. negative cases).
    - Visualize features using histograms, boxplots, and correlation heatmaps.
4. **Model Building**
    - Split the dataset into training and testing sets (e.g., 80:20).
    - Choose appropriate classification algorithms (e.g., Logistic Regression, Decision Tree).
    - Train the model using scikit-learn.
5. **Model Evaluation**
    - Make predictions on the test set.
    - Generate confusion matrix.

- Compute metrics: Accuracy, Precision, Recall, F1-Score, and ROC-AUC.
- Optionally compare different models.

6. **Visualization**
    - Plot confusion matrix.
    - Plot ROC curve.
    - Use bar charts to compare metrics across models.

Conclusion

In this lab, we developed a machine learning-based diagnostic model to detect diseases using patient data. By applying classification algorithms, we were able to predict the presence of a disease with measurable accuracy. This experiment demonstrates how machine learning can aid in early diagnosis and improve healthcare outcomes when properly applied to medical datasets.