RAW DATA TO CLEAN DATA USING EDA

Project Overview

This project performs **Exploratory Data Analysis (EDA)** on an employee dataset using **Pandas**, **NumPy**, **Matplotlib**, and **Seaborn**. The dataset contains information about employees, such as their name, domain, age, location, salary, and experience. The project focuses on data cleaning, handling missing values, and performing statistical and graphical analyses to understand the underlying trends in the dataset.

Dataset Attributes

The dataset includes the following columns:

- **Name**: The employee's name.
- **Domain**: The field or domain in which the employee works.
- **Age**: The employee’s age.
- **Location**: The geographical location of the employee.
- **Salary**: The salary of the employee.
- **Experience**: The number of years of work experience the employee has.

Key Objectives

1. **Data Cleaning**: Handle missing values, detect and remove duplicates, and ensure the data is ready for analysis.
2. **Univariate Analysis**: Analyze each variable individually to understand its distribution and characteristics.
3. **Bivariate Analysis**: Analyze the relationships between two variables to find correlations and insights.
4. **Imputation**: Handle missing data through techniques such as mean, median, or mode imputation.
5. **Variable Identification**: Identify categorical and numerical variables to apply the appropriate analysis techniques.
6. **Variable Transformation**: Transform variables where necessary, such as normalizing salary or categorizing age.
7. **Slicing and Indexing**: Use slicing to filter data for specific domains, locations, or other attributes.

Libraries Used

- **Pandas**: For data manipulation, handling missing values, and exploratory data analysis.
- **NumPy**: For numerical computations and data transformation.
- **Matplotlib**: For basic plotting of graphs and data visualizations.
- **Seaborn**: For advanced statistical visualizations and graphs.

Features

1. Data Cleaning

- **Handling Null Values**: Identify and treat missing values using Pandas functions like isnull() and fillna(). Use imputation methods such as mean, median, or mode depending on the variable type.
- **Slicing and Filtering**: Slice and index the dataset to isolate specific employee groups based on attributes such as domain, location, or salary range.
- **Removing Duplicates**: Detect and remove duplicate records, if any, using drop_duplicates().

2. Univariate Analysis

- **Age Distribution**: Analyze the distribution of employee ages using histograms or density plots.
- **Salary Distribution**: Use boxplots and histograms to understand the distribution of employee salaries.
- **Experience Analysis**: Visualize the distribution of employee experience levels using Seaborn plots like distplot().

3. Bivariate Analysis

- **Salary vs Experience**: Use scatter plots to examine the relationship between salary and years of experience.
- **Domain vs Salary**: Use bar plots to compare the average salaries across different employee domains.
- **Location vs Salary**: Analyze salary distribution across different locations using boxplots.

4. Variable Transformation

- **Log Transformation**: Apply log transformation to salary data if it's highly skewed to normalize the distribution.
- **Categorizing Age**: Group employees into different age categories (e.g., Young, Middle-aged, Senior) for more granular analysis.

5. Imputation

- **Numerical Variables**: Impute missing numerical values (e.g., salary, age, experience) using mean or median values.
- **Categorical Variables**: Impute missing categorical data (e.g., domain, location) using the most frequent category (mode).

Data Analysis Process

1. Loading the Data:

- Load the employee dataset into a Pandas DataFrame and inspect its structure using head(), info(), and describe() functions.

2. Cleaning and Preprocessing:

- Check for null values and missing data using isnull(), handle missing values using imputation techniques, and remove duplicate entries.

- Filter and slice the dataset to focus on specific groups, such as employees in a particular location or domain.

3. Univariate Analysis:

- Use Seaborn and Matplotlib to plot the distribution of numerical variables like salary and age, as well as bar plots for categorical variables like domain and location.

4. Bivariate Analysis:

- Plot pairwise relationships between key variables, such as salary vs experience, or location vs salary, using scatter plots and correlation heatmaps.

5. Variable Transformation:

- Normalize skewed data, such as salary, using log transformation, and categorize continuous variables like age for clearer analysis.

Visualizations

1. **Univariate Plots**: Histogram, bar chart, and boxplot visualizations for individual variables like salary, age, and domain.
2. **Bivariate Plots**: Scatter plots, correlation heatmaps, and boxplots to understand relationships between two variables, such as salary vs experience.
3. **Correlation Matrix**: Visualize the correlation between numerical variables using a heatmap to identify strong relationships.