TP Assignment: Exploratory Data Analysis (EDA) Project

Introduction

In data science, **Exploratory Data Analysis (EDA)** is a critical step that allows you to understand data, detect patterns, spot anomalies, and form hypotheses before modeling. In this TP, you will study two blog articles on EDA, then replicate and adapt their methodology on a dataset of your choice. This exercise will improve your skills in reading data-science literature, applying EDA techniques, organizing code, and communicating insights.
The two guiding articles are:

1. **A Comprehensive Guide on Exploratory Data Analysis (EDA)** by Pabbakavya.
   Link: Click Here

   QR Code:

2. **Extensive EDA on Titanic Dataset** by Phineouse.
   Link: Click Here

   QR Code:

Task Description

Objective: Using the methodology demonstrated in the articles, perform a complete EDA on a dataset you select. You will explore the data, generate visualizations, interpret findings, and produce a written report.
Duration: 2 weeks

Deliverables:

- **Well-organized code:** Jupyter Notebook or Python script that runs without errors. Code must be clean, modular, and well-commented. Include markdown explanations for each step.
- **Written report:** 5–8 pages, structured as follows:
  1. **Introduction:** Brief overview of the dataset and your research question.
  2. **Data Description:** Summary of dataset features, data types, missing values, and any cleaning steps.
  3. **Methodology:** Explanation of EDA steps you performed.
  4. **EDA Results:** Include visualizations, statistics, and tables.
  5. **Interpretation & Discussion:** Insights, patterns, anomalies, and hypotheses.
  6. **Conclusion:** Key findings, limitations, and possible extensions.
  7. **References:** Cite the two guiding articles and any other sources.
- **Presentation (mandatory):** Prepare 10–12 slides summarizing your problem, dataset, methodology, main findings, and conclusions. The presentation is **required** but will **not be graded in this TP session**. It is intended for English test practice.

Tasks

1. **Read the Articles:** Summarize each article in half a page. Focus on the dataset used, the EDA techniques, visualizations, and main findings.
2. **Choose a Dataset:** Select one dataset to analyze. It must have at least 300 rows and include numerical and categorical variables. You can choose from the following datasets available in Python:
   - **From** `sklearn.datasets`**:** `load_iris`, `load_wine`, `load_breast_cancer`, `load_diabetes`, `load_digits`
   - **From** `seaborn`**:** `titanic`, `penguins`, `tips`, `diamonds`, `flights`, `exercise`, `planets`, `car_crashes`
   - Or you may select any other publicly available dataset of your choice, e.g., from Kaggle.

   Define a research question or problem for your analysis and justify your choice.
3. **Data Cleaning:** Check for missing values, duplicates, and inconsistent data. Clean the dataset as needed.
4. **Exploratory Data Analysis (EDA):**
   - Perform univariate analysis: histograms, bar charts, summary statistics.
   - Perform bivariate/multivariate analysis: correlations, scatterplots.
   - Use meaningful visualizations with titles, axis labels, and legends.
5. **Interpret Findings:** For each chart or statistic, explain what it shows and why it matters.
6. **Optional Extensions (Bonus):** You may include feature engineering, clustering (e.g., K-Means), dimensionality reduction (e.g., PCA), or simple predictive modeling.
7. **Report:** Write a structured report including:
   7.1. Introduction
   7.2. Data Description
   7.3. Methodology
   7.4. EDA Results
   7.5. Interpretation & Discussion
   7.6. Conclusion
   7.7. References (include the provided articles)
8. **Presentation (Mandatory for English Test):** Prepare 10–12 slides summarizing the problem, dataset, methodology, and findings. **This is required but not graded in this TP.**
9. **Code Organization:** Make sure your code is clean, modular, and well-commented. Use markdown to explain each step.

Note: The English-test presentation slides are **not graded** in this TP session.

Important Notes & Tips

- Work independently; discussion with peers is allowed.
- Use Python (Pandas, Seaborn, Matplotlib).
- Focus on insights, not just charts.
- For large datasets, you may sample a subset (justify your choice).
- Save your work regularly; version control (Git/GitHub) is recommended.