University of Tehran
College of Science
School of Computer Science

Data mining: HW2 Dr.Sajedi

Income Classification

1 Objective

This project aims to develop a data mining model to classify individuals into income brackets based on census data, specifically predicting whether an individual earns more than $50,000 per year.

2 Dataset Description

The dataset contains census data of individuals, including various features such as age, education level, occupation, etc. The target variable is binary, indicating whether an individual earns more than $50,000 per year.

3 Instructions

1. **Data Preprocessing**: Load the dataset and preprocess it to handle missing values, outliers, and categorical data encoding. Utilize libraries such as pandas and NumPy for efficient data manipulation.

2. **Feature Reduction (PCA)**: Implement Principal Component Analysis (PCA) to reduce the dimensionality of the data while preserving variance. This step aims to enhance model efficiency and reduce computational overhead.

3. **Model Implementation**:

   - **KNN**: Utilize the K-Nearest Neighbors (KNN) algorithm.
   - **SVM**: Implement Support Vector Machine (SVM).
   - **Naive Bayes**: Implement the Naive Bayes algorithm.

- **MLP**: Finally, implement a Multilayer Perceptron (MLP) neural network. Experiment with different parameters, such as the number of layers, then compare and report the results and effects.

4. **Parameter Tuning**: Experiment with different parameters for each model, such as hyperparameters for Naive Bayes, KNN, MLP, and SVM algorithms.

5. **Performance Evaluation**:

   - Analyze and compare the performance of SVM , KNN, Naive Bayes, and MLP algorithms using appropriate metrics such as accuracy, precision, recall, and F1-score.
   - Evaluate the impact of PCA on model performance and discuss its effectiveness in reducing dimensionality while preserving information.

6. **Code Cleanliness and Documentation**:

   - Write clean and well-documented code following best practices such as adhering to PEP 8 conventions, using meaningful variable names, and including comments where necessary.
   - Utilize Jupyter Notebook for documenting code, experimenting with different algorithms and parameters.

7. **Report Preparation**:

   - Prepare a comprehensive report detailing the methodology, findings, and insights derived from the analysis.
   - Structure the report with sections such as Introduction, Data Preprocessing, Model Implementation, Performance Evaluation, and Conclusion.
   - Include visualizations such as confusion matrices, precision-recall curves, and ROC curves to support analysis and conclusions.

4 Additional Guidance

- Ensure thorough documentation of code in .ipynb format.
- Utilize appropriate visualizations and statistical techniques to reinforce analysis and conclusions.
- Provide specific guidance on interpreting results and comparing different implementations effectively.