Assignment 4: Adult Income Dataset - Data Preprocessing

Overview

This assignment focuses on comprehensive data preprocessing techniques for the Adult Income dataset, including handling missing values, feature encoding, scaling, and data splitting for machine learning.

Score: 100/100 ⭐⭐

Objectives

- Handle missing values in real-world datasets
- Perform feature encoding (one-hot and label encoding)
- Apply feature scaling using StandardScaler
- Split data for training and testing
- Implement stratified k-fold cross-validation

Dataset

Adult Income Dataset: Census data containing demographic information to predict whether an individual's income exceeds $50K/year.

Task Requirements

Step 1: Handle Missing Values

- Load data using `pandas.read_csv()`
- Display rows with missing values
  - **Expected**: 3,620 rows with missing values
- Remove rows containing missing values
- Print the shape after dropping
  - **Expected shape**: (45,222, 15)

Step 2: Convert Target Column

Convert the `income` column to integer labels:

- Assign `0` to `'<=50K'`
- Assign `1` to `'>50K'`

Step 3: Feature Encoding

One-Hot Encoding

- Encode the `gender` column using one-hot encoding
- Use `drop_first=True` to represent gender with one column

Label Encoding

- Convert all remaining text columns to integers using label encoding

Step 4: Separate Features and Target

- Use `DataFrame.pop()` to separate feature columns from target column
- Assign feature columns to `x`
- Assign target column (income) to `y`

Step 5: Feature Scaling

- Use `StandardScaler()` to rescale all feature columns in `x`
- Display the shape of `x`
- Show the scaled results

Step 6: Train-Test Split

Use `train_test_split()` to divide data into:

- **Training data**: 80%
- **Test data**: 20%

Display:

- Training data and its shape
- Test data and its shape

Step 7: Stratified K-Fold Cross-Validation

Use `StratifiedKFold()` with the following parameters:

- `n_splits=3`
- `shuffle=True`
- `random_state=100`

Split the training data into 3 splits and display:

- Data for each split
- Shape for each split

Implementation Guidelines

- Use the provided template to write your code
- Fill out all required fields
- Ensure all results are displayed in the notebook

Deliverables

- Jupyter notebook containing:
  - All 7 steps implemented
  - Results displayed for each step
  - Proper shape verification
  - Clear output for all operations

Important:

- ✅ Include all results in the notebook
- ❌ Do NOT compress the notebook file
- ✅ Use the provided template

Requirements

```python
pandas
numpy
scikit-learn
```

Installation

```bash
pip install pandas numpy scikit-learn
```

Key Libraries Used

- `pandas`: Data manipulation and CSV loading
- `sklearn.preprocessing.StandardScaler`: Feature scaling
- `sklearn.preprocessing.LabelEncoder`: Label encoding
- `sklearn.model_selection.train_test_split`: Train-test splitting
- `sklearn.model_selection.StratifiedKFold`: Stratified k-fold CV

Expected Outputs Summary

1. Missing values: 3,620 rows
2. Clean dataset: (45,222, 15)
3. Encoded features and target
4. Scaled features
5. 80-20 train-test split
6. 3-fold stratified cross-validation splits

Results

Successfully completed all data preprocessing steps including missing value handling, feature encoding, scaling, and proper data splitting for machine learning model training and evaluation.

Assignment completed as part of Deep Learning coursework