1 ASSIGNMENT OBJECTIVES

Data preprocessing is a fundamental step in any data mining or machine learning pipeline. Real-world data comes in various formats and often requires different preprocessing techniques depending on its type and structure. This assignment aims to provide hands-on experience with preprocessing four major types of data:

- **Image Data**: Visual information requiring spatial and pixel-level transformations
- **Tabular Data**: Structured data in rows and columns with mixed data types
- **Textual Data**: Unstructured text requiring linguistic processing
- **Temporal Data**: Time-series data with sequential dependencies

By completing this assignment, students will develop practical skills in applying appropriate preprocessing techniques for different data modalities and understanding the impact of these techniques on downstream analysis tasks.

2 ASSIGNMENT DESCRIPTION

Students will work in groups of 3 members to preprocess and analyze four different types of datasets. Each type of data requires specific preprocessing techniques that must be appropriately applied and thoroughly documented.

2.1 Part 1: Preprocessing Digital Image Data (Required)

2.1.1 Dataset Selection

Students are required to choose one image dataset to perform this part. You can select from the following options:

- CIFAR-10 or CIFAR-100 (general object recognition)
- Fashion-MNIST (clothing item classification)
- Chest X-Ray Images (medical imaging): Chest X-Ray Images, NIH Chest X-rays
- Plant Disease Dataset (agricultural applications): New Plant Diseases Dataset

2.1.2 Required Preprocessing Techniques

Students must implement and demonstrate the following preprocessing techniques:

a) **Loading and Resizing**: First, load images from the selected dataset. Then, resize images to a consistent dimension (e.g., 224×224, 128×128), and explain the rationale for choosing the specific dimensions. Finally, discuss the trade-offs between image size and computational efficiency.

b) **Grayscale Conversion**: Convert color images to grayscale where appropriate, then compare information retention between color and grayscale representations. Discuss when grayscale conversion is beneficial versus detrimental.

c) **Normalization**: Apply pixel normalization (e.g., scaling to [0,1] or [-1,1]) and implement standardization (zero mean, unit variance). Then, compare different normalization techniques and their effects on the data distribution.

d) **Edge Detection (Optional Bonus)**: Apply edge detection algorithms (Sobel, Prewitt, Canny), extract edge features from images, and then visualize detected edges and discuss their significance for your chosen dataset.

2.2 Part 2: Preprocessing Tabular Data (Required)

2.2.1 Dataset Selection

Students are required to choose one tabular dataset to perform this part. You can select from the following sources:

- UCI Machine Learning Repository
- Kaggle datasets, such as: Credit Card Fraud Detection, Credit Card Transactions Fraud Detection Dataset.

Dataset Requirements: The selected dataset must have at least five attributes and 10,000 records.

2.2.2 Required Preprocessing Techniques

Students must implement and demonstrate the following preprocessing techniques:

a) **Handling Missing Values**: Identify patterns of missing data (MCAR, MAR, MNAR). Then apply appropriate imputation techniques (mean, median, mode, forward/backward fill, K-NN imputation). After that, compare the impact of different imputation strategies on data quality and distribution.

b) **Data Normalization**: Apply Min-Max scaling, standardization (Z-score normalization), and robust scaling for data with outliers. Then compare distributions before and after normalization using appropriate visualizations.

c) **Categorical Encoding**: Identify categorical variables requiring encoding. Apply suitable encoding for each type of categorical variable, such as one-hot encoding for nominal variables and ordinal encoding for ordinal variables. Discuss strategies for handling high-cardinality categorical features.

d) **Feature Selection**: Choose a suitable feature selection method. For example, calculate the correlation matrix, use variance threshold, apply feature importance from tree-based models, or implement recursive feature elimination (RFE). Then compare the selected feature sets and justify your final selection.

2.3 Part 3: Preprocessing Textual Data (Bonus - Choose Part 3 OR Part 4)

Note: If you choose Part 3, you do not need to complete Part 4.

2.3.1 Dataset Selection

Students are required to choose one textual dataset to perform this part. You can select from domains such as:

- Movie reviews (IMDB, Rotten Tomatoes): IMDB Dataset of 50K Movie Reviews
- News Articles
- Social media posts (Twitter, Reddit): Twitter 15 and 16
- Product reviews (Amazon, Yelp): Amazon Product Reviews Dataset

Dataset Requirements: The dataset should contain at least 1,000 text samples.

2.3.2 Required Preprocessing Techniques

Students must implement and demonstrate the following preprocessing techniques:

a) **Tokenization**: Apply basic techniques such as word tokenization, sentence tokenization, and subword tokenization (if applicable). Then compare different tokenization strategies and discuss their advantages for your dataset.

b) **Removing Stop Words**: Identify common stop words in the dataset. Then remove standard stop words and analyze the impact on vocabulary size and information retention.

c) **Stemming and Lemmatization**: Apply stemming algorithms (Porter, Snowball) and lemmatization techniques. Then compare the results of stemming versus lemmatization. Discuss the advantages and disadvantages of each approach for your specific dataset.

d) **Text Vectorization**: Apply techniques such as Bag-of-Words (BoW), TF-IDF vectorization, and Word2Vec or similar embeddings (if applicable). Then compare the dimensionality and sparsity of different representations.

2.4 Part 4: Preprocessing Temporal Data (Bonus - Choose Part 3 OR Part 4)

Note: If you choose Part 4, you do not need to complete Part 3.

2.4.1 Dataset Selection

Each group must select one time-series dataset. Recommended sources from Kaggle include domains such as:

- Stock market data: Stock Market Data (NASDAQ, NYSE, S&P500)

- Weather and climate data
- Energy consumption data
- COVID-19 time-series data

Dataset Requirements: The dataset should span at least 6 months with regular observations.

2.4.2 Required Preprocessing Techniques

Students must implement and demonstrate the following preprocessing techniques:

a) **Parsing Date and Time**: Parse datetime strings into proper datetime objects. Handle different datetime formats and extract components (year, month, day, hour, etc.). Set a proper datetime index for the time series.

b) **Handling Time Gaps**: Identify missing timestamps. Apply forward fill or backward fill methods. Use interpolation techniques (linear, polynomial, spline) and compare their effectiveness.

c) **Extracting Time-Based Features**: Extract cyclical features (day of week, month, season). Create indicator variables for special periods (holidays, weekends). Encode cyclical features using sine/cosine transformations. Extract time elapsed since specific events.

d) **Resampling Time Series Data**: Choose suitable techniques such as upsampling (interpolation), downsampling (aggregation), or different aggregation functions (mean, sum, max, min). Then, compare the effects of different resampling frequencies on data characteristics.

e) **Lag Features**: Create lag features ($t-1$, $t-2$, $\ldots$, $t-n$). Create rolling window statistics (moving average, rolling standard deviation). Create difference features. Discuss autocorrelation and the rationale for choosing appropriate lag values.

3 IMPLEMENTATION REQUIREMENTS

3.1 Technical Requirements

- All preprocessing must be implemented in Python using Jupyter Notebooks
- Required libraries: NumPy, Pandas, Matplotlib, Seaborn, scikit-learn, OpenCV/PIL (for images), NLTK/spaCy (for text)
- Code must be well-documented with clear comments explaining each step
- Each preprocessing technique must be in a separate, clearly labeled notebook cell
- Use markdown cells to provide explanations, interpretations, and insights

3.2 Documentation Requirements

For each data type and preprocessing technique, students must document:

1. **Dataset Description**: Source and download link, size and dimensions, brief description of the data, and potential use cases.
2. **Preprocessing Rationale**: Why this technique is necessary for this data, expected benefits of applying the technique, and discussion about potential drawbacks or limitations.
3. **Implementation Details**: Step-by-step explanation of the code, parameter choices and their justification, and computational considerations.
4. **Results and Analysis**: Quantitative metrics (where applicable), visual comparisons and interpretation of results (from visualized charts), and discussion of the impact on data quality.
5. **Language**: All documentation and explanations must be written in Vietnamese.
6. **Page Limit**: There is no page limit for the report.

4 PROJECT ORGANIZATION

The requirements for organization and implementation of the project:

- **Folder Organization**: Notebook files should be clearly separated for each part with descriptive names.
- **Answer Presentation**: All questions should be answered through visualization figures accompanied by students' contemplative explanations and interpretations.
- **Code Documentation**: There must be an explanation (in markdown cells) for every code cell in the Jupyter notebooks. Cells containing only code without explanation will be ignored during grading.

Recommended Project Structure:

```
project-root/
|-- README.md                          # Project overview and instructions
|-- requirements.txt                   # Python dependencies
|-- data/
|   |-- images/                        # Image dataset files
|   |-- tabular/                       # Tabular dataset files
|   |-- text/                          # Text dataset files (if applicable)
|   |-- temporal/                      # Time-series dataset files (if applicable)
|-- notebooks/
|   |-- 01_image_preprocessing.ipynb
|   |-- 02_tabular_preprocessing.ipynb
|   |-- 03_text_preprocessing.ipynb    # If Part 3 is chosen
|   |-- 04_temporal_preprocessing.ipynb # If Part 4 is chosen
|-- docs/                              # Put your document here
|   |-- Report.pdf
```

5 SUBMISSION

- **File Format**: All reports, code, and datasets must be submitted as a compressed file (.zip) named according to the format: `Group_ID.zip` (e.g., `Group_05.zip`).
- **Large Files**: If the compressed file exceeds 25MB, prioritize including the report and source code. Upload large files (images, datasets) to Google Drive and include a publicly accessible link in your README.md file.

- **README Requirements**: Include a README.md file with:
  - Group member names and student IDs
  - Brief description of datasets used
  - Instructions for running the notebooks
  - Links to any external resources (Google Drive, datasets, etc.)

6 CRITERIA FOR GRADING

No.	Criteria	Score
1	**Image Preprocessing (Required)**	**30%**
	Correct implementation of required techniques	10%
	Quality of visualizations and presentation	10%
	Analysis, interpretation, and insights	10%
2	**Tabular Data Preprocessing (Required)**	**30%**
	Correct implementation of required techniques	10%
	Quality of visualizations and presentation	10%
	Analysis, interpretation, and insights	10%
3	**Text Preprocessing (Bonus)**	**20%**
	Correct implementation of all techniques	10%
	Quality of visualizations and presentation	5%
	Analysis, interpretation, and insights	5%
4	**Temporal Data Preprocessing (Bonus)**	**20%**
	Correct implementation of all techniques	10%
	Quality of visualizations and presentation	5%
	Analysis, interpretation, and insights	5%
5	**Code Quality and Documentation**	**20%**
	Clear, readable, and well-commented code (AI assistance must be under 30%)	15%
	Proper project organization and structure	5%
6	**Presentation and Q&A with TAs**	**20%**
Total Score Formula: (1) + (2) + (5) + (6) + (3 OR 4)		**120%**

Note: Students choose either Part 3 or Part 4 for the bonus section. Maximum achievable score is 120%.

7 IMPORTANT NOTICES

Please pay attention to the following important notices:

- This is a **GROUP** assignment. Each group consists of 3 members.
- **Duration**: Approximately 3 weeks from the assignment date.
- **Submission Deadline**: 23:55 03/11/2025.

- **Academic Integrity**: Any form of plagiarism, cheating, dishonesty, or unauthorized collaboration will result in a score of 0 for the entire course.
- **AI Tool Usage**: Limited AI assistance is permitted (maximum 30% as assessed in code quality). However, students must understand and be able to explain all submitted code.
- **Late Submission Policy**: Not accepted

8 CONTACT

If you have any questions about this assignment, please contact the instructor:

Lê Nhựt Nam

Email: lnnam@fit.hcmus.edu.vn

The instructor will respond to questions as soon as possible.

End of Assignment