Group No.

Leader:

Members :

1. Sean Kieran Sain
2. Vinz Bleik Ibay
3. Kylene Varona
4. Vien Bernales

Laboratory Activity

Title: Dataset Generation and Exploration for Neural Network Projects

Objective:

- To generate a custom dataset relevant to the student's proposed neural network project.
- To perform data exploration using Python tools to understand and prepare the dataset for training a neural network.

Instructions:

Part 1: Dataset Generation

Tasks:

1. **Identify the Problem Domain:**
    - Define the objective of your Neural Network project (e.g., image classification, sentiment analysis, price prediction).
2. **Data Source Selection:**
    - Choose how to generate or gather your dataset:
        - Web scraping
        - API data (e.g., Twitter, OpenWeather)
        - Public datasets (e.g., Kaggle, UCI)
        - Synthetic data using scikit-learn, Faker, or data augmentation techniques.

3. **Data Collection:**
    - Collect or simulate **at least 500 records**.
    - Save your dataset in .csv format.
    - Include labels if it's for supervised learning.
4. **Document your Dataset:**
    - Variables and their data types.
    - How data was collected or generated.
    - Any preprocessing done.

Part 2: Data Exploration (Week 2)

Tools to Use:

- Python with Pandas, Matplotlib, Seaborn, and NumPy.

Tasks:

1. **Load Dataset:**
    - Use Pandas to load and view your dataset.

```
# 1. Load Dataset
import pandas as pd

df = pd.read_csv('updated_fake_news_dataset.csv')
print("--- Dataset Info ---")
print(df.info())
```

2. **Summary Statistics:**
    - Use .describe(), .info() to understand the structure.

```
# 2. Summary Statistics
print("\n--- Summary Statistics ---")
print(df.describe(include='all'))
```

3. **Handle Missing or Duplicate Values:**
    - Identify and clean as needed.

```
# 3. Handle Missing or Duplicate Values
print("\n--- Missing Values ---")
print(df.isnull().sum())

print("\n--- Duplicate Rows ---")
print(df.duplicated().sum())

# Drop duplicate rows
df_cleaned = df.drop_duplicates()
print("\nAfter cleaning:")
print("Total rows:", len(df_cleaned))
print("Unique titles:", df_cleaned['title'].nunique())
print("Unique texts:", df_cleaned['text'].nunique())
```

    -

4. **Visualize Your Data:**

    - Create at least three types of visualizations:

```
# 4. Visualize Your Data
import matplotlib.pyplot as plt
import seaborn as sns
```

        - Histogram

```
# Histogram
plt.figure(figsize=(10, 5))
sns.histplot(data=df_cleaned, x='text_length', hue='label', kde=True, palette="Set2", multiple="stack")
plt.title("Text Length Distribution by News Label")
plt.xlabel("Text Length")
plt.ylabel("Frequency")
plt.tight_layout()
plt.show()
```

        - Correlation heatmap

```
# Correlation Heatmap
numeric_cols = ['text_length', 'title_length', 'num_words', 'num_sentences', 'label_numeric']
correlation_matrix = df_cleaned[numeric_cols].corr()

plt.figure(figsize=(8, 6))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', vmin=-1, vmax=1)
plt.title("Correlation Heatmap of Text Features")
plt.tight_layout()
plt.show()
```

- Boxplot or scatter plot

```
# Boxplot
plt.figure(figsize=(6, 5))
sns.boxplot(x='label', y='text_length', data=df_cleaned)
plt.title("Boxplot of Text Length by News Label")
plt.xlabel("Label")
plt.ylabel("Text Length")
plt.show()
```

5. **Feature Analysis:**
    - Identify key features relevant to your neural network input layer.

```
# 5. Feature Analysis
print("\n--- Average Text Length by Label ---")
print(df_cleaned.groupby('label')['text_length'].mean())
```

Expected Output / Submission:

- Jupyter Notebook or Python script with:
    - Data generation/collection code
    - Data exploration code
    - Visualizations
- .csv file of the dataset
- 1–2-page documentation report:
    - Dataset description
    - Observations from exploration
    - Challenges encountered

Lab Activity 2 Report

Title: Dataset Generation and Exploration for Neural Network Projects

- **Objective:**

  The objective of this lab activity is to create a dataset suitable for training a neural network and perform data exploration using Python libraries. The project focuses on classifying news articles as **FAKE** or **REAL** based on textual features.

- Dataset Generation

  The dataset was synthetically generated to simulate real and fake news articles. Each record includes a headline, article body, and a label (FAKE or REAL). Additional numeric features were derived from the text to support analysis and modeling. The final dataset includes the following columns:

  - title – News headline
  - text – Full article body
  - label – Target classification (FAKE or REAL)
  - text_length – Number of characters in the article body
  - title_length – Number of characters in the headline
  - num_words – Word count of the article
  - num_sentences – Estimated sentence count
  - label_numeric – Encoded label for modeling (0 = REAL, 1 = FAKE)

  The dataset was saved as **lab2.csv**.

- Data Exploration

  Observations from Data Exploration

  The dataset was thoroughly examined using Pandas and Seaborn. Out of the original 500 entries, 62 duplicates were removed, resulting in 438 unique records. No missing values were present across the dataset. Summary statistics showed that most articles contained 24–26 words and around 3 sentences. Histogram and boxplot visualizations revealed that REAL articles generally had slightly longer text compared to FAKE ones, with an average text length of 160.96 vs. 157.02 respectively. A correlation heatmap showed strong relationships among text_length, num_words, and num_sentences, supporting their relevance as input features for a classification model. These insights indicate a well-structured dataset suitable for training a neural network.

- Challenges Encountered

The initial version of the dataset had many duplicated entries due to repetitive text templates. To resolve this, additional variety was introduced in the text generation process. Another issue was the gray heatmap caused by a lack of numeric features, which was addressed by engineering new columns like word count and sentence count. Minor styling issues in visualizations (e.g., legend warnings) were also fixed by adjusting Seaborn parameters.

Whole Code

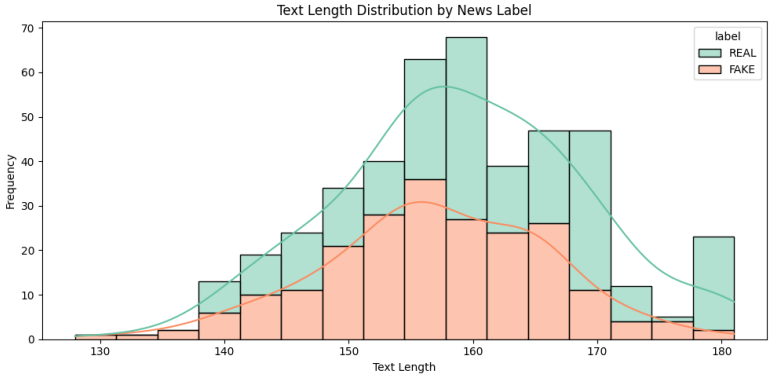
```python
# 1. Load Dataset
import pandas as pd

df = pd.read_csv('lab2.csv')
print("--- Dataset Info ---")
print(df.info())

# 2. Summary Statistics
print("\n--- Summary Statistics ---")
print(df.describe(include='all'))

# 3. Handle Missing or Duplicate Values
print("\n--- Missing Values ---")
print(df.isnull().sum())

print("\n--- Duplicate Rows ---")
print(df.duplicated().sum())

# Drop duplicate rows
df_cleaned = df.drop_duplicates()
print("\nAfter cleaning:")
print("Total rows:", len(df_cleaned))
print("Unique titles:", df_cleaned['title'].nunique())
print("Unique texts:", df_cleaned['text'].nunique())

# 4. Visualize Your Data
import matplotlib.pyplot as plt
import seaborn as sns

# Histogram
plt.figure(figsize=(10, 5))
sns.histplot(data=df_cleaned, x='text_length', hue='label', kde=True, palette="Set2", multiple="stack")
plt.title("Text Length Distribution by News Label")
plt.xlabel("Text Length")
plt.ylabel("Frequency")
plt.tight_layout()
plt.show()

# Correlation Heatmap
numeric_cols = ['text_length', 'title_length', 'num_words', 'num_sentences', 'label_numeric']
correlation_matrix = df_cleaned[numeric_cols].corr()

plt.figure(figsize=(8, 6))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', vmin=-1, vmax=1)
plt.title("Correlation Heatmap of Text Features")
plt.tight_layout()
plt.show()

# Boxplot
plt.figure(figsize=(6, 5))
sns.boxplot(x='label', y='text_length', data=df_cleaned)
plt.title("Boxplot of Text Length by News Label")
plt.xlabel("Label")
plt.ylabel("Text Length")
plt.show()

# 5. Feature Analysis
print("\n--- Average Text Length by Label ---")
print(df_cleaned.groupby('label')['text_length'].mean())
```

Output

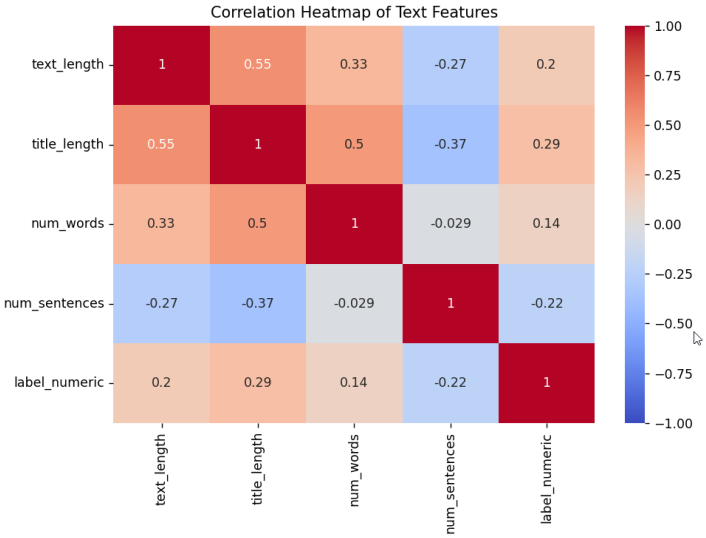
```
--- Dataset Info ---
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 500 entries, 0 to 499
Data columns (total 8 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   title           500 non-null    object
 1   text            500 non-null    object
 2   label           500 non-null    object
 3   text_length     500 non-null    int64
 4   title_length    500 non-null    int64
 5   num_words       500 non-null    int64
 6   num_sentences   500 non-null    int64
 7   label_numeric   500 non-null    int64
dtypes: int64(5), object(3)
memory usage: 31.4+ KB
None

--- Summary Statistics ---
                                                 title  \
count                                              500
unique                                              20
top     BFP nagsagawa ng fire drill sa Quezon City
freq                                                36
mean                                               NaN
std                                                NaN
min                                                NaN
25%                                                NaN
50%                                                NaN
75%                                                NaN
max                                                NaN

                                                 text label  text_length  \
count                                             500   500   500.000000
unique                                            438     2          NaN
top     Dahon ng mangga, ginawang cellphone charger. I...  REAL          NaN
freq                                                3   250          NaN
mean                                              NaN   NaN   159.056000
std                                               NaN   NaN     9.792479
min                                               NaN   NaN   128.000000
25%                                               NaN   NaN   153.000000
50%                                               NaN   NaN   159.000000
75%                                               NaN   NaN   166.000000
max                                               NaN   NaN   181.000000

       title_length   num_words  num_sentences  label_numeric
count     500.000000  500.000000     500.000000     500.000000
unique           NaN         NaN            NaN            NaN
top              NaN         NaN            NaN            NaN
freq             NaN         NaN            NaN            NaN
mean       48.606000   24.838000       3.044000       0.500000
std         5.681489    1.442191       0.205301       0.500501
min        39.000000   22.000000       3.000000       0.000000
25%        43.000000   24.000000       3.000000       0.000000
50%        48.000000   25.000000       3.000000       0.500000
75%        55.000000   26.000000       3.000000       1.000000
max        57.000000   31.000000       4.000000       1.000000

--- Missing Values ---
title            0
text             0
label            0
text_length      0
title_length     0
num_words        0
num_sentences    0
label_numeric    0
dtype: int64

--- Duplicate Rows ---
62

After cleaning:
Total rows: 438
Unique titles: 20
Unique texts: 438
```

Visualizations

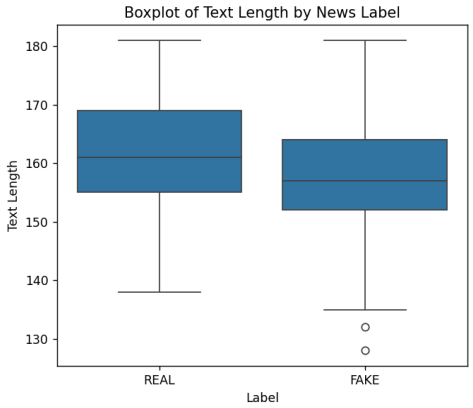
Histogram



Correlation Heatmap



Boxplot




```
--- Average Text Length by Label ---
label
FAKE    157.023364
REAL    160.964286
Name: text_length, dtype: float64
```

Evaluation Criteria:

Criteria	Points
Relevance and quality of dataset	20
Completeness of data exploration	30
Quality of visualizations	20
Clarity of documentation/report	20
Code quality and organization	10
Total	**100**