Experiment 2

Name: Asad Shaikh
Branch: MTech CE
Registration Id: 242050023

Aim: To write R/Python program to Read data set from any online website, excel file and CSV file and to perform
a) Linear regression and logistic regression on iris dataset.
b) K-means clustering.

Theory:

In Big Data Analytics, the first step of any data analysis process is data collection or ingestion. Data can come from various sources such as:

- Online websites (like Wikipedia, financial data sites, etc.)
- Excel files used in business environments
- CSV (Comma Separated Values) files, a standard format for tabular data

Python, with libraries like pandas, provides powerful tools to read and manipulate these data formats efficiently.

1. Reading Data from an Online Website (HTML Table):

- Websites often contain tables of structured data (e.g., stock prices, country statistics).
- **pandas.read_html(url)** reads all tables from a webpage and returns a list of DataFrames.
- You need an internet connection and a well-structured HTML table for this to work.

2. Reading Data from an Excel File:

- Excel is commonly used in data reporting and storage.
- Python uses **pandas.read_excel()** to load .xlsx files.
- Requires the openpyxl library (for .xlsx format).

3. Reading Data from a CSV File:

- CSV is one of the most popular text-based formats for storing tabular data.
- Each row in the file is a data record, and each field is separated by a comma.
- Python uses **pandas.read_csv()** to read CSV files easily and efficiently.

Machine Learning Models:

1. Linear Regression:

- A supervised learning method for predicting a continuous outcome.
- In the Iris dataset, we might predict *Petal Length* based on *Sepal Length*.

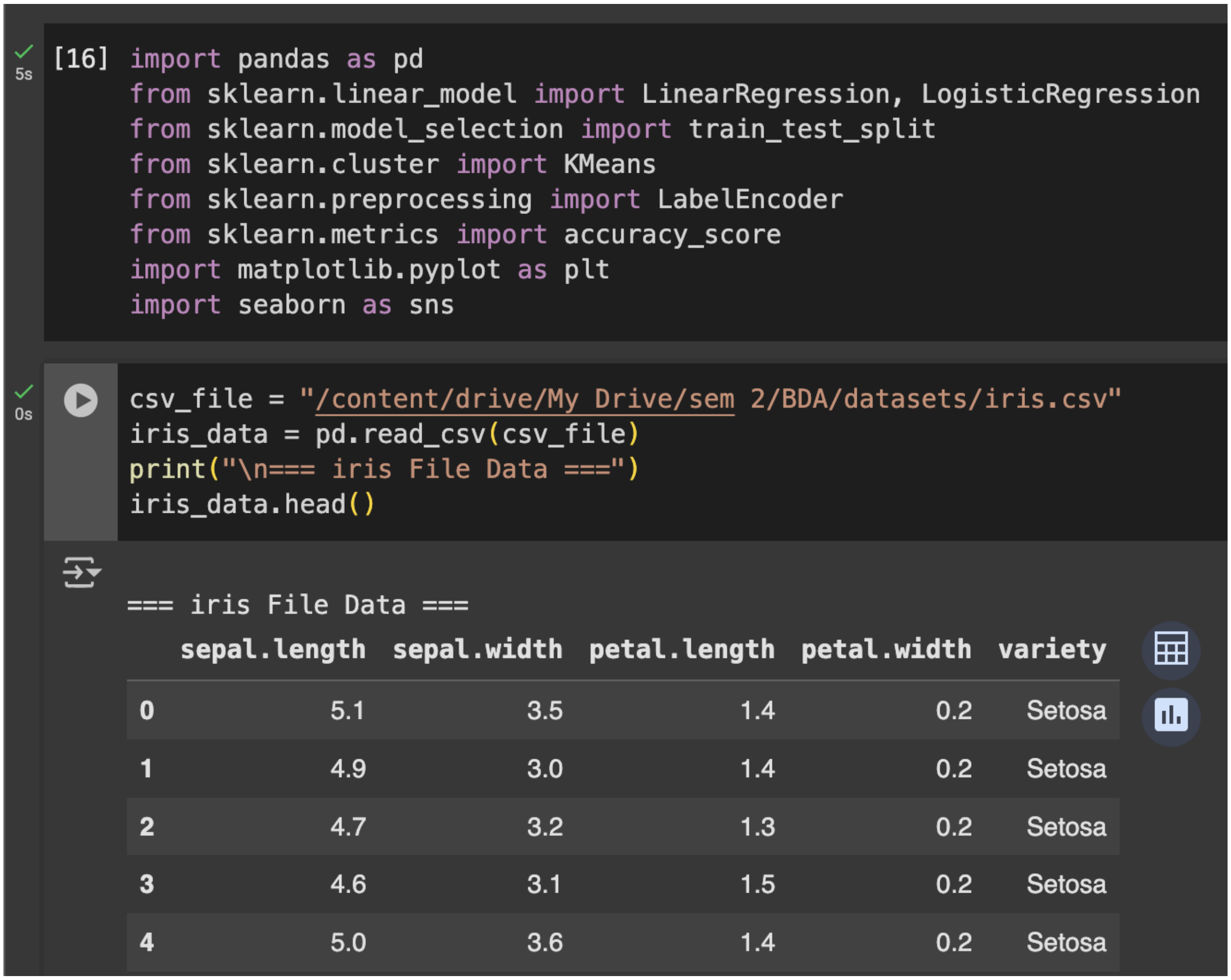
2. Logistic Regression:

- A supervised classification algorithm used for binary or multiclass classification.
- Used here to classify iris species.

3. K-Means Clustering:

- An **unsupervised learning** algorithm that groups similar data points into **K clusters**.
- Commonly used for exploratory data analysis.

Code / Output:
importing libraries and reading data from iris.csv:

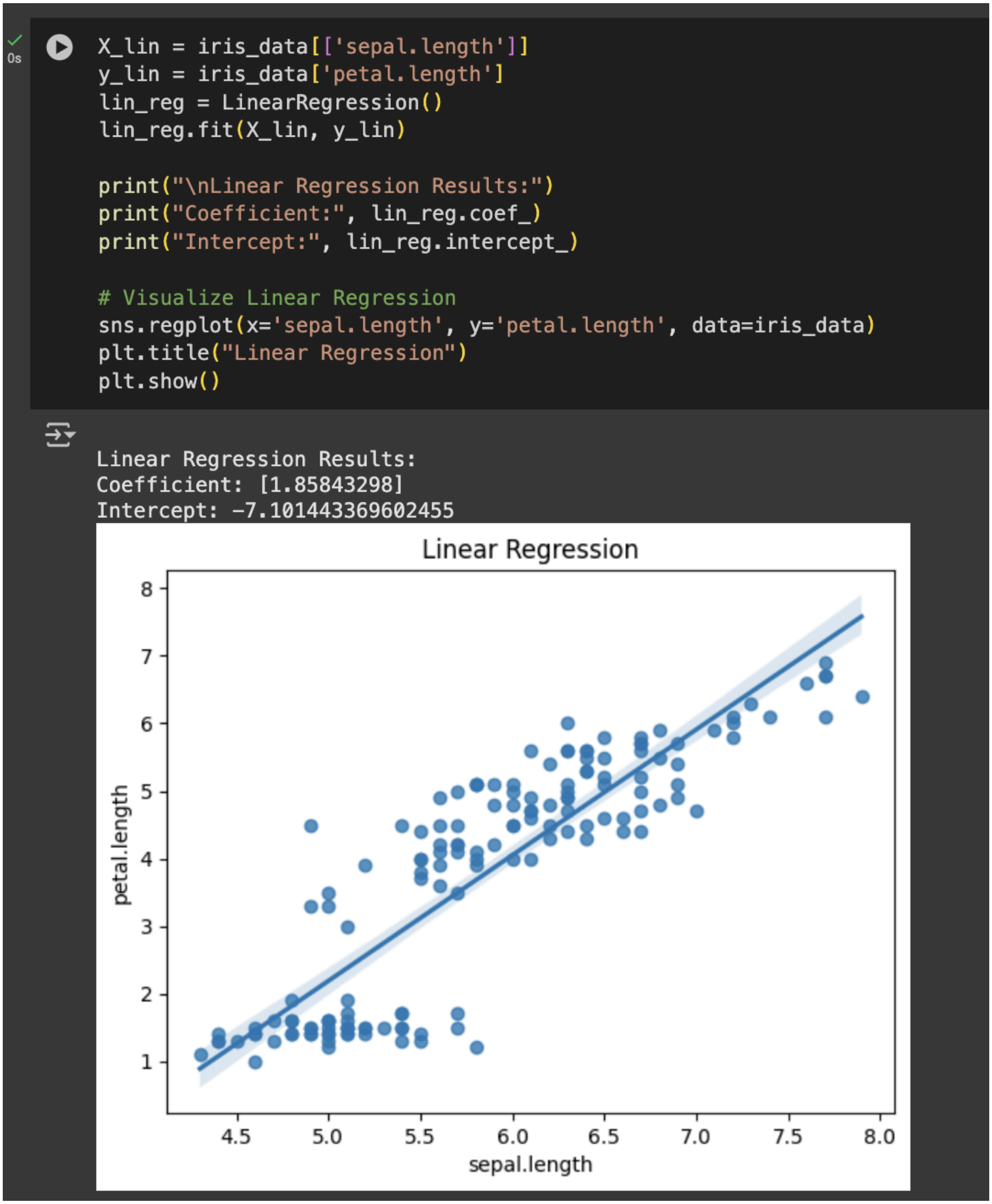


```
import pandas as pd
from sklearn.linear_model import LinearRegression, LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.cluster import KMeans
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import accuracy_score
import matplotlib.pyplot as plt
import seaborn as sns
```

```
csv_file = "/content/drive/My Drive/sem 2/BDA/datasets/iris.csv"
iris_data = pd.read_csv(csv_file)
print("\n=== iris File Data ===")
iris_data.head()
```

```
=== iris File Data ===
   sepal.length  sepal.width  petal.length  petal.width  variety
0           5.1          3.5           1.4          0.2   Setosa
1           4.9          3.0           1.4          0.2   Setosa
2           4.7          3.2           1.3          0.2   Setosa
3           4.6          3.1           1.5          0.2   Setosa
4           5.0          3.6           1.4          0.2   Setosa
```

Linear Regression on the iris.csv data:



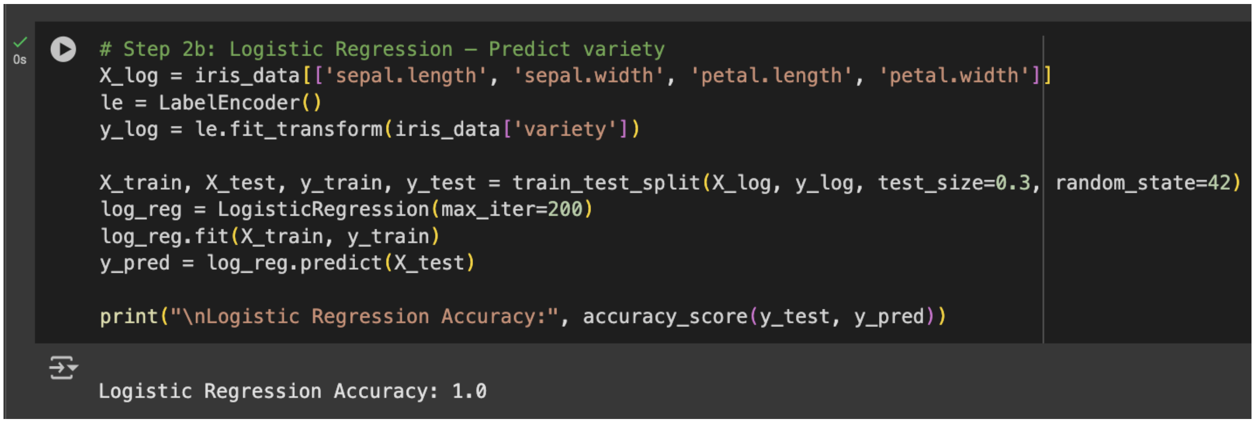
```
X_lin = iris_data[['sepal.length']]
y_lin = iris_data['petal.length']
lin_reg = LinearRegression()
lin_reg.fit(X_lin, y_lin)

print("\nLinear Regression Results:")
print("Coefficient:", lin_reg.coef_)
print("Intercept:", lin_reg.intercept_)

# Visualize Linear Regression
sns.regplot(x='sepal.length', y='petal.length', data=iris_data)
plt.title("Linear Regression")
plt.show()
```

```
Linear Regression Results:
Coefficient: [1.85843298]
Intercept: -7.101443369602455
```

Logistic Regression on the iris.csv data:



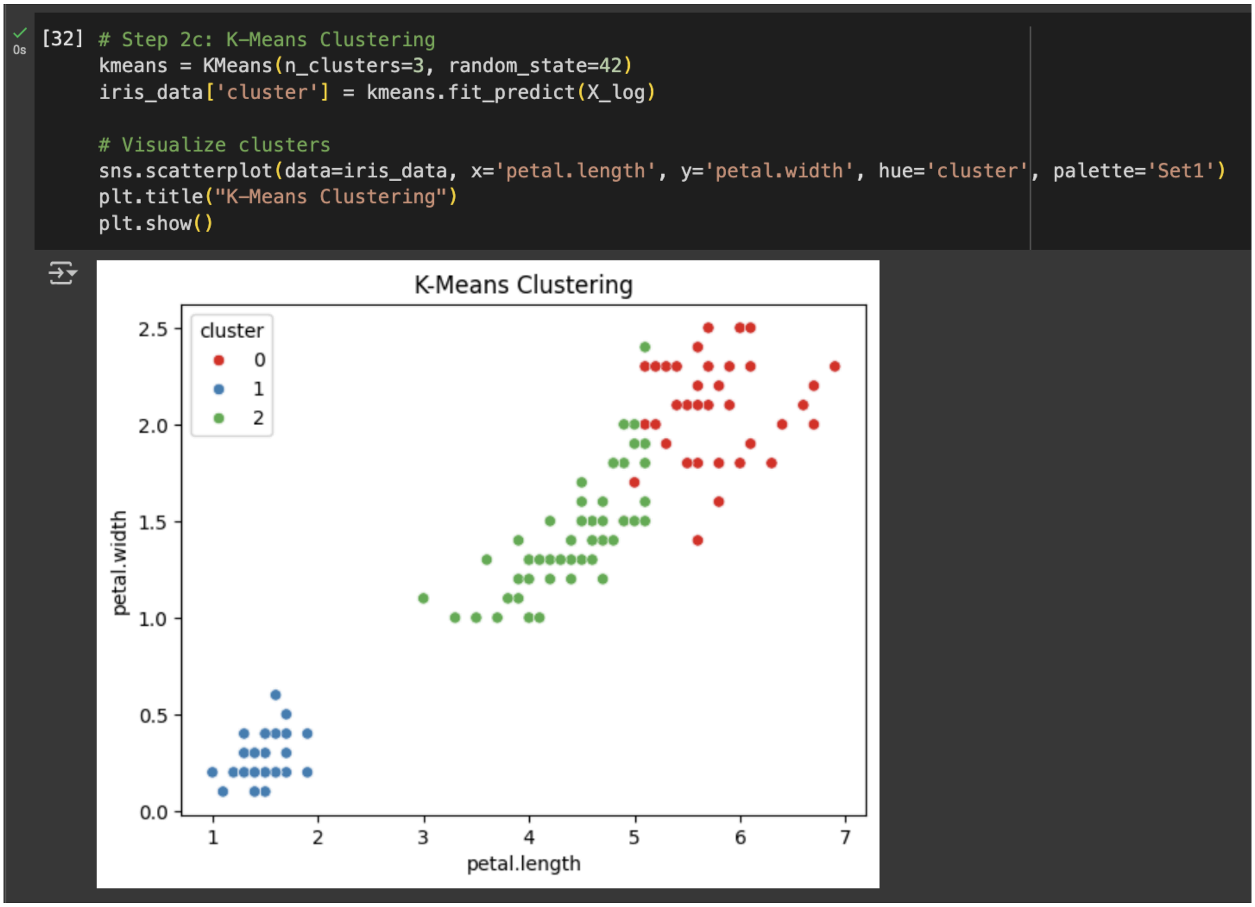
```
# Step 2b: Logistic Regression — Predict variety
X_log = iris_data[['sepal.length', 'sepal.width', 'petal.length', 'petal.width']]
le = LabelEncoder()
y_log = le.fit_transform(iris_data['variety'])

X_train, X_test, y_train, y_test = train_test_split(X_log, y_log, test_size=0.3, random_state=42)
log_reg = LogisticRegression(max_iter=200)
log_reg.fit(X_train, y_train)
y_pred = log_reg.predict(X_test)

print("\nLogistic Regression Accuracy:", accuracy_score(y_test, y_pred))
```

```
Logistic Regression Accuracy: 1.0
```

K-Means clustering on iris.csv data:



```
[32] # Step 2c: K-Means Clustering
kmeans = KMeans(n_clusters=3, random_state=42)
iris_data['cluster'] = kmeans.fit_predict(X_log)

# Visualize clusters
sns.scatterplot(data=iris_data, x='petal.length', y='petal.width', hue='cluster', palette='Set1')
plt.title("K-Means Clustering")
plt.show()
```

Conclusion:

In this experiment, we learnt about python and we also learnt about how to read data from websites, excel files and csv files and display them in tabular format. We also learned Linear, logistic regression and K-Means clustering in machine learning and implemented these machine learning techniques on the iris.csv data using Python.