LAB REPORT 10

SUBMITTED TO: MR. MUBASHIR IQBAL

SUBMITTED BY: AHMED SALEEM RANA

REG ID: 24-CYS-023

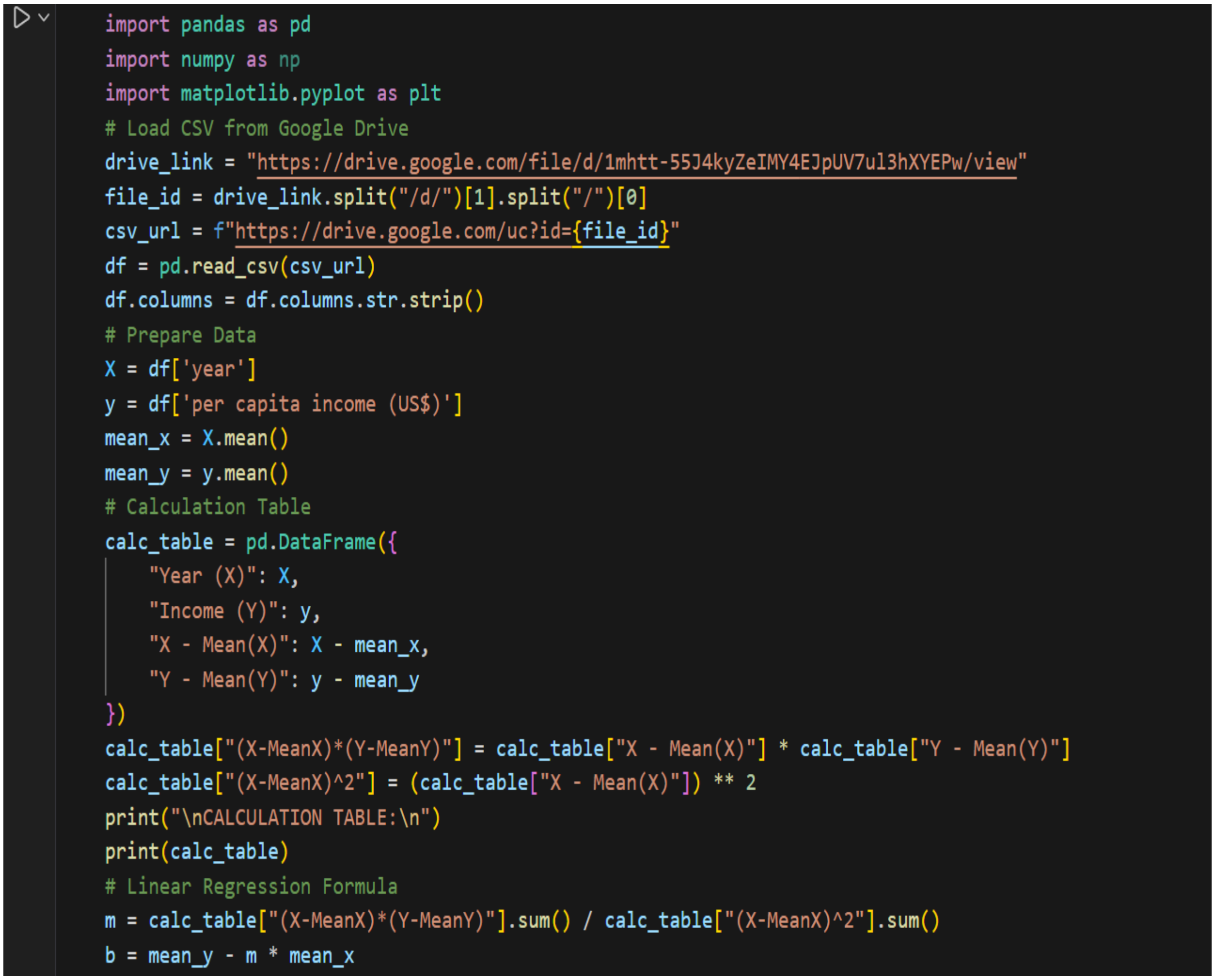
SECTION: B

HITEC UNIVERSITY TAXILA, CANTT

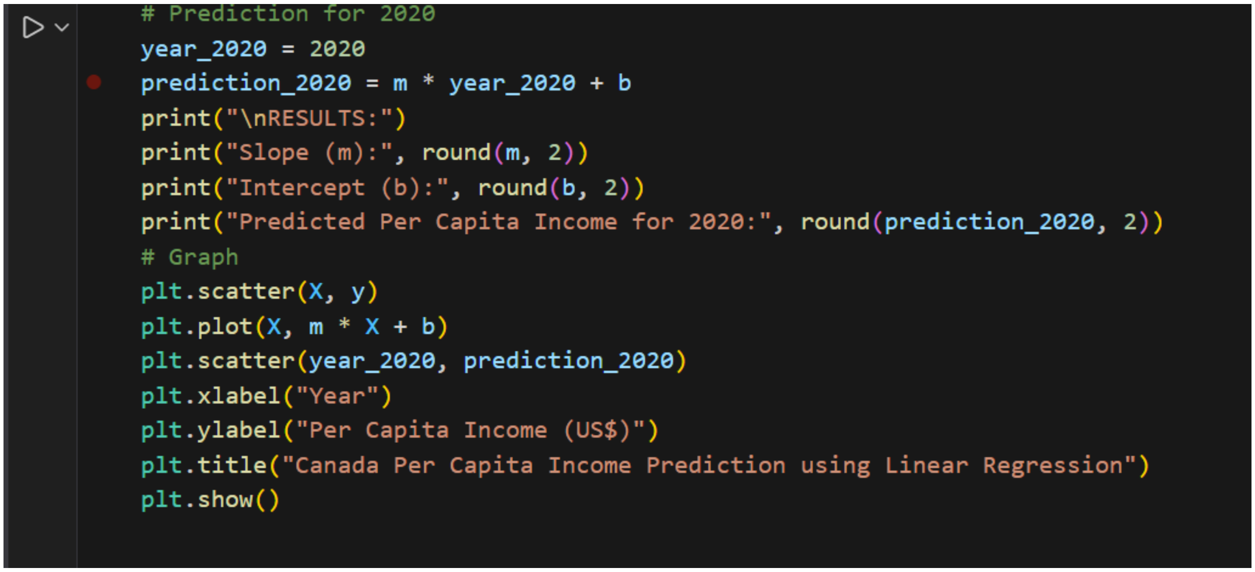
TASK 01

Predict Canada's per capita income in the year 2020 using the canada_per_capita_income.csv file. Using this, build a regression model and predict the per capita income of Canadian citizens in the year 2020. Here is the link to the CSV file: https://drive.google.com/file/d/1mhtt-55J4kyZeIMY4EJpUV7ul3hXYEPw/view?usp=sharing

CODE:

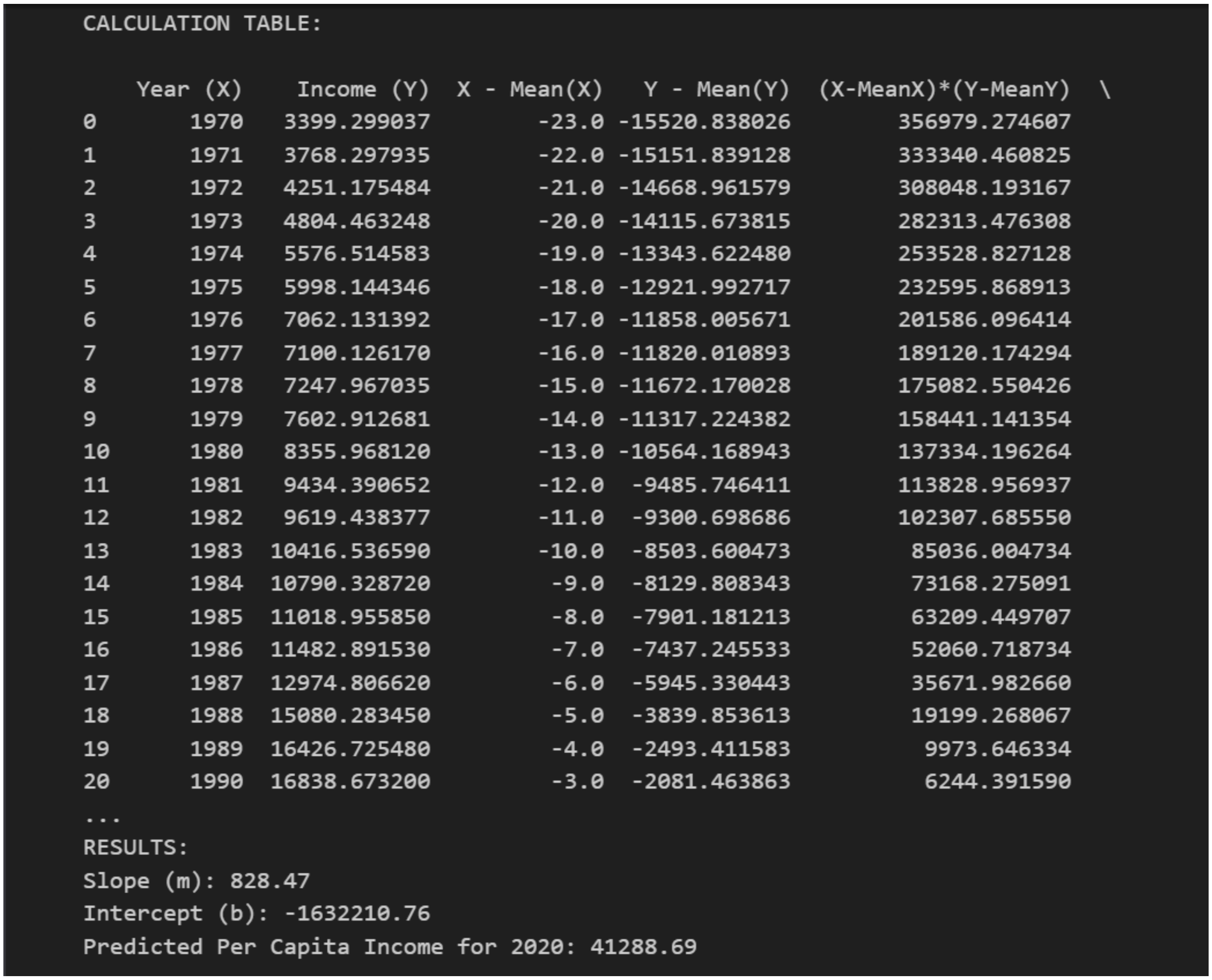


```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
# Load CSV from Google Drive
drive_link = "https://drive.google.com/file/d/1mhtt-55J4kyZeIMY4EJpUV7ul3hXYEPw/view"
file_id = drive_link.split("/d/")[1].split("/")[0]
csv_url = f"https://drive.google.com/uc?id={file_id}"
df = pd.read_csv(csv_url)
df.columns = df.columns.str.strip()
# Prepare Data
X = df['year']
y = df['per capita income (US$)']
mean_x = X.mean()
mean_y = y.mean()
# Calculation Table
calc_table = pd.DataFrame({
    "Year (X)": X,
    "Income (Y)": y,
    "X - Mean(X)": X - mean_x,
    "Y - Mean(Y)": y - mean_y
})
calc_table["(X-MeanX)*(Y-MeanY)"] = calc_table["X - Mean(X)"] * calc_table["Y - Mean(Y)"]
calc_table["(X-MeanX)^2"] = (calc_table["X - Mean(X)"]) ** 2
print("\nCALCULATION TABLE:\n")
print(calc_table)
# Linear Regression Formula
m = calc_table["(X-MeanX)*(Y-MeanY)"].sum() / calc_table["(X-MeanX)^2"].sum()
b = mean_y - m * mean_x
```



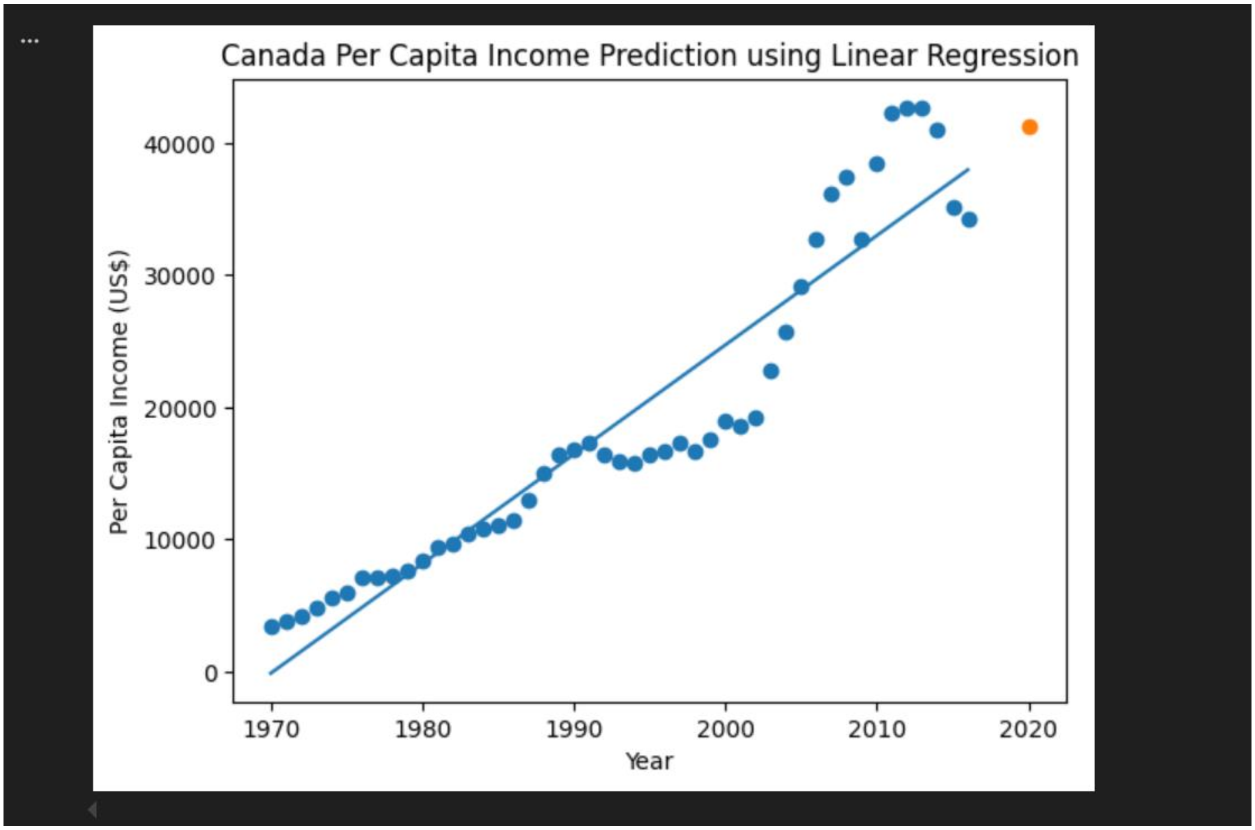
```python
# Prediction for 2020
year_2020 = 2020
prediction_2020 = m * year_2020 + b
print("\nRESULTS:")
print("Slope (m):", round(m, 2))
print("Intercept (b):", round(b, 2))
print("Predicted Per Capita Income for 2020:", round(prediction_2020, 2))
# Graph
plt.scatter(X, y)
plt.plot(X, m * X + b)
plt.scatter(year_2020, prediction_2020)
plt.xlabel("Year")
plt.ylabel("Per Capita Income (US$)")
plt.title("Canada Per Capita Income Prediction using Linear Regression")
plt.show()
```

OUTPUT:



```
CALCULATION TABLE:

    Year (X)    Income (Y)  X - Mean(X)   Y - Mean(Y)  (X-MeanX)*(Y-MeanY)  \
0       1970   3399.299037        -23.0 -15520.838026        356979.274607
1       1971   3768.297935        -22.0 -15151.839128        333340.460825
2       1972   4251.175484        -21.0 -14668.961579        308048.193167
3       1973   4804.463248        -20.0 -14115.673815        282313.476308
4       1974   5576.514583        -19.0 -13343.622480        253528.827128
5       1975   5998.144346        -18.0 -12921.992717        232595.868913
6       1976   7062.131392        -17.0 -11858.005671        201586.096414
7       1977   7100.126170        -16.0 -11820.010893        189120.174294
8       1978   7247.967035        -15.0 -11672.170028        175082.550426
9       1979   7602.912681        -14.0 -11317.224382        158441.141354
10      1980   8355.968120        -13.0 -10564.168943        137334.196264
11      1981   9434.390652        -12.0  -9485.746411        113828.956937
12      1982   9619.438377        -11.0  -9300.698686        102307.685550
13      1983  10416.536590        -10.0  -8503.600473         85036.004734
14      1984  10790.328720         -9.0  -8129.808343         73168.275091
15      1985  11018.955850         -8.0  -7901.181213         63209.449707
16      1986  11482.891530         -7.0  -7437.245533         52060.718734
17      1987  12974.806620         -6.0  -5945.330443         35671.982660
18      1988  15080.283450         -5.0  -3839.853613         19199.268067
19      1989  16426.725480         -4.0  -2493.411583          9973.646334
20      1990  16838.673200         -3.0  -2081.463863          6244.391590
...
RESULTS:
Slope (m): 828.47
Intercept (b): -1632210.76
Predicted Per Capita Income for 2020: 41288.69
```

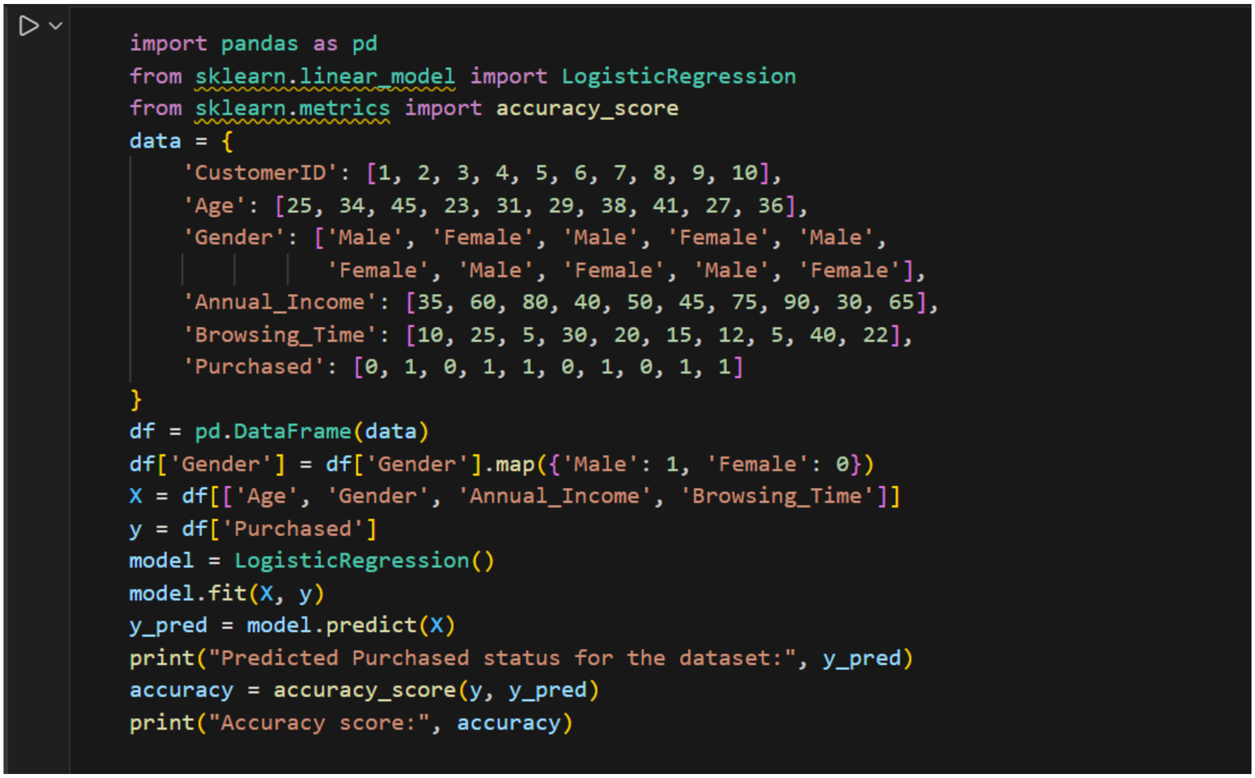


TASK 02

2. Using *the below dataset, build a logistic regression model to predict whether a customer will purchase a product or not (Purchased) based on the features Age, Gender, Annual_Income, and Browsing_Time. Perform the following tasks:*
   1. Preprocess the data (e.g., convert Gender to numerical).
   2. Split the data into features (X) and target (y).
   3. Fit a logistic regression model using scikit-learn.
   4. Predict the purchase status for the new sample data.
   5. Evaluate the model using the accuracy score.

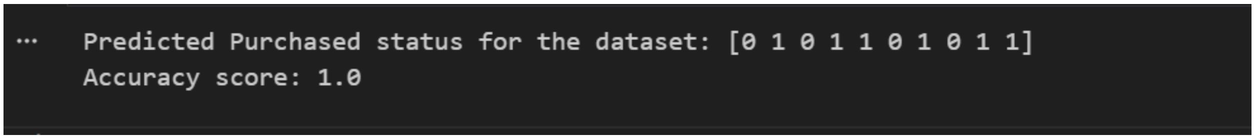
CustomerID	Age	Gender	Annual_Income (k$)	Browsing_Time (min)	Purchased (1=Yes, 0=No)
1	25	Male	35	10	0
2	34	Female	60	25	1
3	45	Male	80	5	0
4	23	Female	40	30	1
5	31	Male	50	20	1
6	29	Female	45	15	0
7	38	Male	75	12	1
8	41	Female	90	5	0
9	27	Male	30	40	1
10	36	Female	65	22	1

CODE:



```
import pandas as pd
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
data = {
    'CustomerID': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
    'Age': [25, 34, 45, 23, 31, 29, 38, 41, 27, 36],
    'Gender': ['Male', 'Female', 'Male', 'Female', 'Male',
               'Female', 'Male', 'Female', 'Male', 'Female'],
    'Annual_Income': [35, 60, 80, 40, 50, 45, 75, 90, 30, 65],
    'Browsing_Time': [10, 25, 5, 30, 20, 15, 12, 5, 40, 22],
    'Purchased': [0, 1, 0, 1, 1, 0, 1, 0, 1, 1]
}
df = pd.DataFrame(data)
df['Gender'] = df['Gender'].map({'Male': 1, 'Female': 0})
X = df[['Age', 'Gender', 'Annual_Income', 'Browsing_Time']]
y = df['Purchased']
model = LogisticRegression()
model.fit(X, y)
y_pred = model.predict(X)
print("Predicted Purchased status for the dataset:", y_pred)
accuracy = accuracy_score(y, y_pred)
print("Accuracy score:", accuracy)
```

OUTPUT:



```
Predicted Purchased status for the dataset: [0 1 0 1 1 0 1 0 1 1]
Accuracy score: 1.0
```

CONCLUSION:

First in these machine learning algorithms, we applied logistic regression, a binary classification algorithm, where we predicted if customers would make a purchase based on characteristics such as age, sex, annual income, as well as time spent browsing. However, after assigning a range of numeric values for sex, the algorithm produced 100% accuracy on a limited set of data. We also applied linear regression on forecasting the per capital income in Canada for the year 2020 based on previous data, after which we obtained a figure of between $40,000 and $45,000.