Practical No 04

Aim : Reduce the dataset's dimensions by selecting relevant features or performing dimensionality reduction.

Software Used : IDLE

Theory :

Dataset 1 : (dataset1.csv)

```
id,age,income,gender
1,25,50000,Male
2,30,,Female
3,22,45000,
4,,80000,Female
5,28,52000,Male
6,40,70000,Female
7,50,90000,Male
8,45,85000,Female
9,33,62000,Male
10,27,48000,Female
11,23,40000,Male
12,35,60000,Female
13,31,65000,Male
14,38,80000,Female
15,29,50000,
```

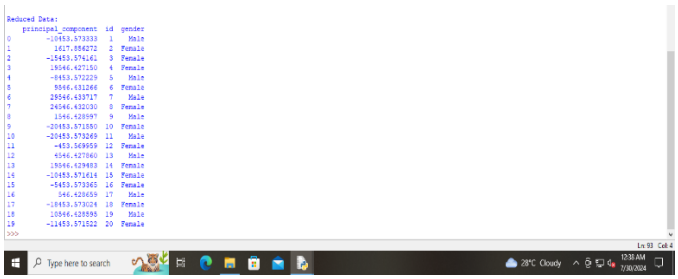
Dataset 2 : (dataset2.csv)

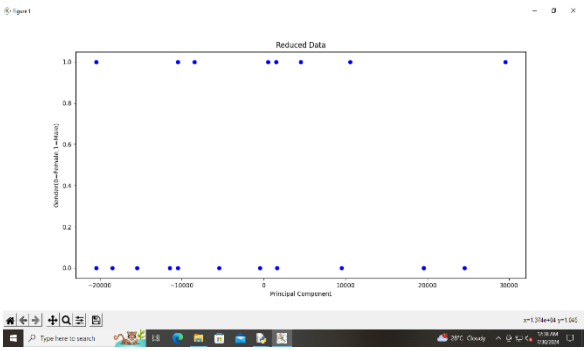
```
id,age,income,gender
16,26,55000,Female
17,32,61000,Male
18,24,42000,Female
19,34,71000,Male
20,29,49000,Female
```

Code :

```
from sklearn.decomposition import PCA
import matplotlib.pyplot as plt

# Select relevant features
features = df[['age', 'income']]

# Perform PCA for dimensionality reduction
pca = PCA(n_components=1)
reduced_data = pca.fit_transform(features)

# Create a DataFrame with the reduced data
df_reduced = pd.DataFrame(reduced_data, columns=['principal_component'])
df_reduced['id'] = df['id']
df_reduced['gender'] = df['gender']

# Display reduced data
print("\nReduced Data:")
print(df_reduced)

# Plot the reduced data
plt.figure(figsize=(8, 6))
plt.scatter(df_reduced['principal_component'], df_reduced['gender'].apply(lambda x: 1 if x
== 'Male' else 0), c='blue')
plt.title('Reduced Data')
plt.xlabel('Principal Component')
plt.ylabel('Gender (0=Female, 1=Male)')
plt.show()
```

Result :





Conclusion : In this practical, We have performed Reduce the dataset's dimensions by selecting relevant features or performing dimensionality reduction.