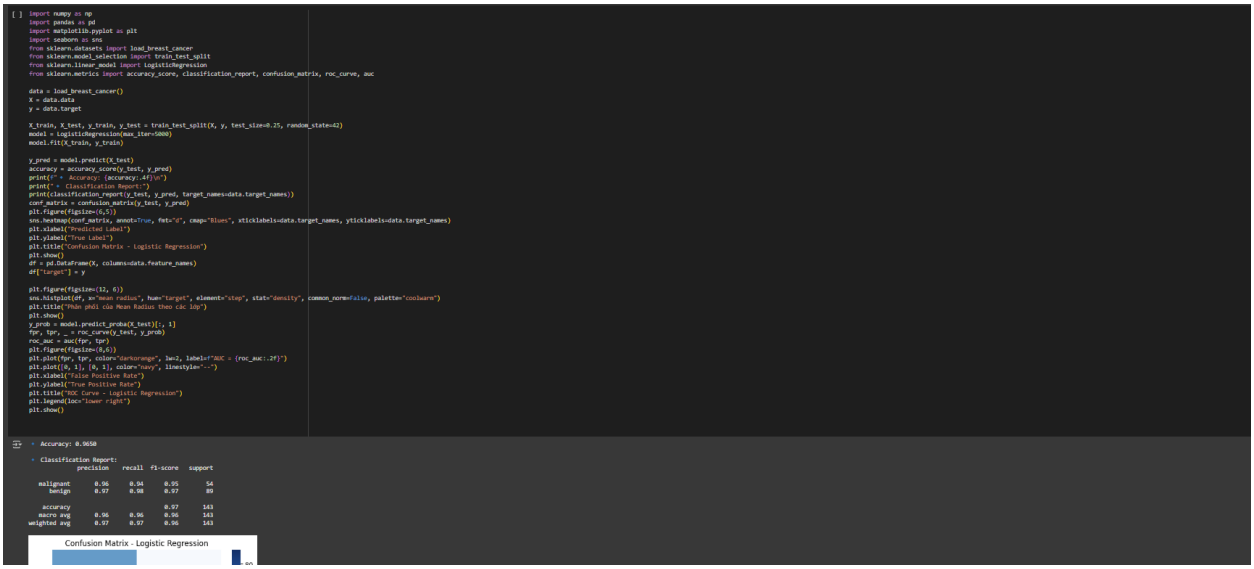
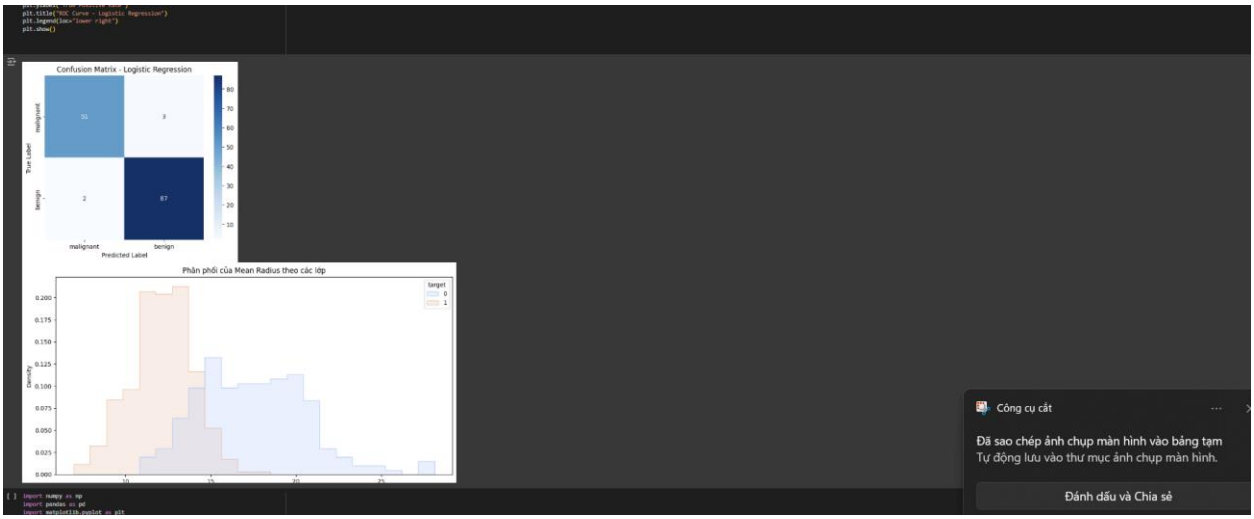
SỐ hóa và quản trị thông tin số

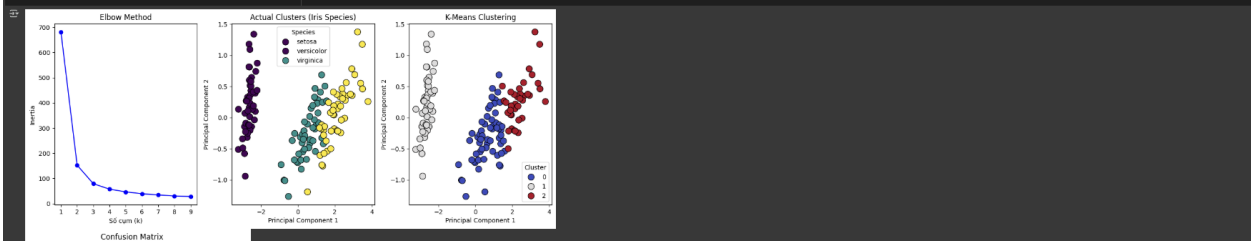
Lab4

LÊ Minh Tâm_2274802010781

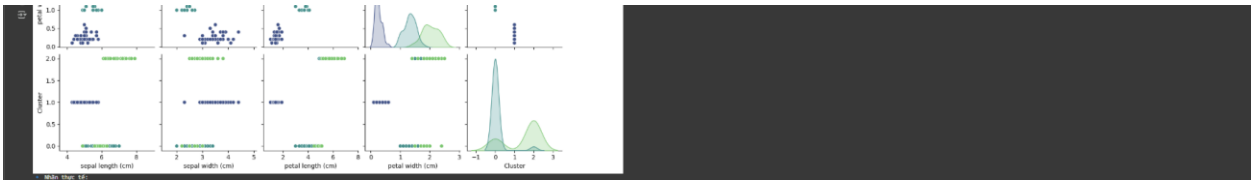




```
print(" * Nhãn thực tế:")
print(y[:20])
print("\n * Nhãn phân cụm K-Means:")
print(y_pred[:20])

#giải thích PCA để giảm chiều là PCA giúp chuyển dữ liệu từ không gian cao chiều xuống thấp chiều mà vẫn giữ được phần lớn thông tin quan trọngTrực quan hóa dữ liệu nhiều chiều (ví dụ: từ 4D về 2D).
#Tiền xử lý trước khi phân cụm (K-Means, DBSCAN,...).
#Giảm nhiễu trong mô hình Machine Learning.
#Ưu đo cần PCa
#Dữ liệu có quá nhiều đặc trưng gây quá tải tính toán.
#Khó trực quan hóa dữ liệu nếu có >3 chiều.
#Một số đặc trưng có thể không cần thiết hoặc tương quan cao.
#PCA giúp giảm nhiễu, tối ưu hóa mô hì\
## tổ ích
# PCA giúp tối ưu dữ liệu, giảm chiều nhưng vẫn giữ thông tin quan trọng.Thích hợp để trực quan hóa, phân cụm và cải thiện mô hình học máy.
```




```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from google.colab import files
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
import os
```

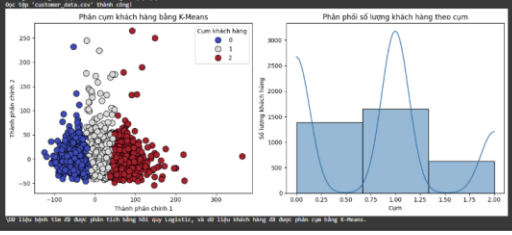



```
- Nhãn thực tế:
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]

- Nhãn phân cụm K-Means:
[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1]
```

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from google.colab import files
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA
import os
```

```
plt.subplot(1, 2, 2)
sns.histplot(df_customers['Cluster'], bins=3, kde=True)
plt.title("Phân phối số lượng khách hàng theo cụm")
plt.xlabel("Cụm")
plt.ylabel("Số lượng khách hàng")

plt.tight_layout()
plt.show()

print("\nDữ liệu bệnh tim đã được phân tích bằng hồi quy Logistic, và dữ liệu khách hàng đã được phân cụm bằng K-Means.")
```

```
Tệp 'framingham.csv' không tồn tại. Vui lòng tải lên.
Choose Files framingham.csv
- framingham.csv(text/csv) - 190950 bytes, last modified: 21/02/2025 - 100% done
Saving framingham.csv to framingham (2).csv
Đọc tệp 'framingham.csv' thành công!

Độ chính xác của mô hình Logistic Regression: 0.8360655737704918

Báo cáo phân loại:
              precision    recall  f1-score   support

           0       0.84      0.99      0.91       610
           1       0.56      0.08      0.14       122

    accuracy                           0.84       732
   macro avg       0.70      0.53      0.53       732
weighted avg       0.80      0.84      0.78       732


Ma trận nhầm lẫn:
 [[602   8]
 [112  10]]
Tệp 'customer_data.csv' không tồn tại. Vui lòng tải lên.
Choose Files framingham (1).csv
- framingham (1).csv(text/csv) - 190950 bytes, last modified: 27/02/2025 - 100% done
Saving framingham (1).csv to framingham (1) (1).csv
Đọc tệp 'customer_data.csv' thành công!
```




```
Dữ liệu bệnh tim đã được phân tích bằng hồi quy Logistic, và dữ liệu khách hàng đã được phân cụm bằng K-Means.
```