

## Program :-

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from sklearn.naive_bayes import GaussianNB
from sklearn.preprocessing import LabelEncoder

headers = ["buying", "maint", "doors", "persons", "lug_boot", "safety", "clas"]
df_car = pd.read_csv("car_evaluation.csv", names=headers)

label_encoder = LabelEncoder()
for col in df_car.columns:
    df_car[col] = label_encoder.fit_transform(df_car[col])

features = ["buying", "maint", "doors", "persons", "lug_boot", "safety"]
x = df_car[features]
y = df_car["clas"]

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.3, random_state=100)

for n_components in range(2, 6):
    pca = PCA(n_components)
    x_train_pca = pca.fit_transform(x_train)
    x_test_pca = pca.transform(x_test)

    clf = GaussianNB()
    clf.fit(x_train_pca, y_train)

    y_predict = clf.predict(x_test_pca)

    accuracy = accuracy_score(y_test, y_predict)

    print(f"Number of PCA components: {n_components}")
    print("Accuracy: {:.2f}%".format(accuracy * 100))

covariance_matrix = x.cov()
print("\nCovariance among original features:")
print(covariance_matrix)

plt.figure(figsize=(10, 8))
sns.heatmap(covariance_matrix, annot=True, fmt=".4f", cmap="coolwarm")
plt.title("Covariance Matrix Heatmap")
plt.show()
```

Output :-

```
(base) student@cseadmin:~/Desktop/sinan$ python3 pca.py
Number of PCA components: 2
Accuracy: 71.29%
Number of PCA components: 3
Accuracy: 71.29%
Number of PCA components: 4
Accuracy: 74.18%
Number of PCA components: 5
Accuracy: 73.99%

Covariance among original features:
      buying    maint    doors  persons  lug_boot  safety
buying  1.250724  0.000000  0.000000  0.000000  0.000000  0.000000
maint    0.000000  1.250724  0.000000  0.000000  0.000000  0.000000
doors    0.000000  0.000000  1.250724  0.000000  0.000000  0.000000
persons  0.000000  0.000000  0.000000  0.667053  0.000000  0.000000
lug_boot 0.000000  0.000000  0.000000  0.000000  0.667053  0.000000
safety   0.000000  0.000000  0.000000  0.000000  0.000000  0.667053
Warning: QT_DEVICE_PIXEL_RATIO is deprecated. Instead use:
  QT_AUTO_SCREEN_SCALE_FACTOR to enable platform plugin controlled per-screen factors.
  QT_SCREEN_SCALE_FACTORS to set per-screen DPI.
  QT_SCALE_FACTOR to set the application global scale factor.
(base) student@cseadmin:~/Desktop/sinan$
```

