Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

JUNAID GIRKAR
60004190057
TE COMPS A4

EXPERIMENT - 7

AIM: Implement Principal Component Analysis (PCA).

THEORY:

Principal Component Analysis is an unsupervised learning algorithm that is used for dimensionality reduction in machine learning. It is a statistical process that converts the observations of correlated features into a set of linearly uncorrelated features with the help of orthogonal transformation. These new transformed features are called the **Principal Components**. It is one of the popular tools that is used for exploratory data analysis and predictive modelling. It is a technique to draw strong patterns from the given dataset by reducing the variances.

PCA generally tries to find the lower-dimensional surface to project the high-dimensional data.

PCA works by considering the variance of each attribute because the high attribute shows the good split between the classes, and hence it reduces the dimensionality. Some real-world applications of PCA are ***image processing, movie recommendation systems, and optimising the power allocation in various communication channels.*** It is a feature extraction technique, so it contains the important variables and drops the least important variable.

Steps for PCA algorithm

1. **Getting the dataset**

   Firstly, we need to take the input dataset and divide it into two subparts X and Y, where X is the training set, and Y is the validation set.

2. **Representing data into a structure**

   Now we will represent our dataset into a structure. Such as we will represent the two-dimensional matrix of independent variable X. Here each row

corresponds to the data items, and the column corresponds to the Features. The number of columns is the dimensions of the dataset.

3. **Standardizing the data**

   In this step, we will standardize our dataset. Such as in a particular column, the features with high variance are more important compared to the features with lower variance.

   If the importance of features is independent of the variance of the feature, then we will divide each data item in a column with the standard deviation of the column. Here we will name the matrix as Z.

4. **Calculating the Covariance of Z**

   To calculate the covariance of Z, we will take the matrix Z, and will transpose it. After transpose, we will multiply it by Z. The output matrix will be the Covariance matrix of Z.

5. **Calculating the Eigen Values and Eigen Vectors**

   Now we need to calculate the eigenvalues and eigenvectors for the resultant covariance matrix Z. Eigenvectors or the covariance matrix are the directions of the axes with high information. And the coefficients of these eigenvectors are defined as the eigenvalues.

6. **Sorting the Eigen Vectors**

   In this step, we will take all the eigenvalues and will sort them in decreasing order, which means from largest to smallest. And simultaneously sort the eigenvectors accordingly in matrix P of eigenvalues. The resultant matrix will be named as P*.

7. **Calculating the new features Or Principal Components**

   Here we will calculate the new features. To do this, we will multiply the P* matrix to the Z. In the resultant matrix Z*, each observation is the linear combination of original features. Each column of the Z* matrix is independent of each other.

8. **Remove less or unimportant features from the new dataset.**

   The new feature set has occurred, so we will decide here what to keep and what to remove. It means, we will only keep the relevant or important features in the new dataset, and unimportant features will be removed out.

Applications of Principal Component Analysis

- PCA is mainly used as the dimensionality reduction technique in various AI applications such **as computer vision, image compression, etc.**
- It can also be used for finding hidden patterns if data has high dimensions. Some fields where PCA is used are Finance, data mining, Psychology, etc.

CODE:

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
from sklearn.datasets import load_wine
df=load_wine()
df=pd.DataFrame(data=np.c_[df['data'],df['target']], columns=df['feature_names']+['target'])
df.head()
df.shape
df.columns
freq = df['target'].value_counts()
plt.xlabel('Parameter')
plt.ylabel('Counts')
plt.title('Count of Each features')
freq.plot(kind='bar',color="green")
df = df[['alcohol','magnesium','color_intensity','target']]
sns.pairplot(df,hue='target',palette="crest")
from sklearn.preprocessing import StandardScaler
features = ['alcohol', 'magnesium','color_intensity']
# Separating out the features
x = df.loc[:, features].values
# Separating out the target
y = df.loc[:,['target']].values
# Standardizing the features
```

```
x = StandardScaler().fit_transform(x)
# Scaled Dataset
df_scaled = pd.DataFrame(x)
"""## Applying PCA"""
from sklearn.decomposition import PCA
pca = PCA(n_components=2)
principalComponents = pca.fit_transform(x)
principalDf = pd.DataFrame(data = principalComponents, columns = ['principal component
1', 'principal component 2'])
principalDf.head()
finalDf = pd.concat([principalDf, df[['target']]], axis = 1)
finalDf.head()
fig = plt.figure(figsize = (8,8))
ax = fig.add_subplot(1,1,1)
ax.set_xlabel('Principal Component 1', fontsize = 15)
ax.set_ylabel('Principal Component 2', fontsize = 15)
ax.set_title('2 component PCA', fontsize = 20)
targets = [0, 1, 2]
colors = ['r', 'g', 'b']
for target, color in zip(targets,colors):
  indicesToKeep = finalDf['target'] == target
  ax.scatter(finalDf.loc[indicesToKeep, 'principal component 1'],
finalDf.loc[indicesToKeep, 'principal component 2'], c = color, s = 50)
ax.legend(targets)
ax.grid()
```

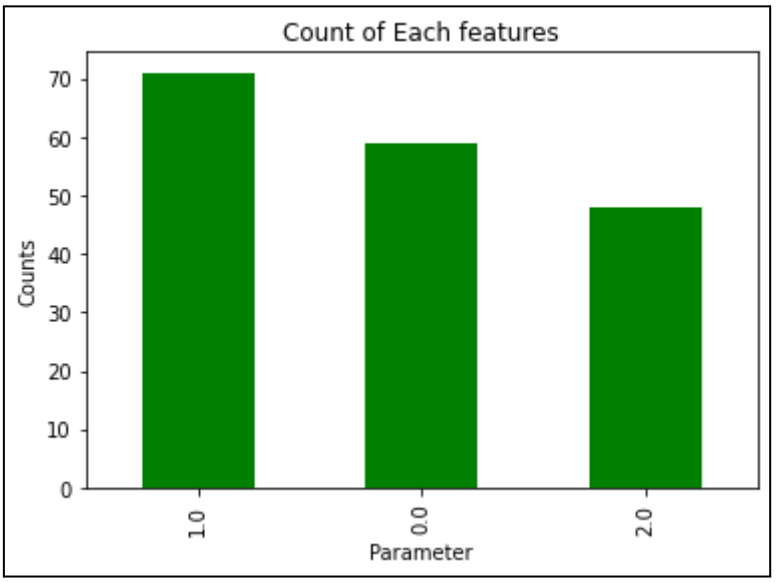
Output:

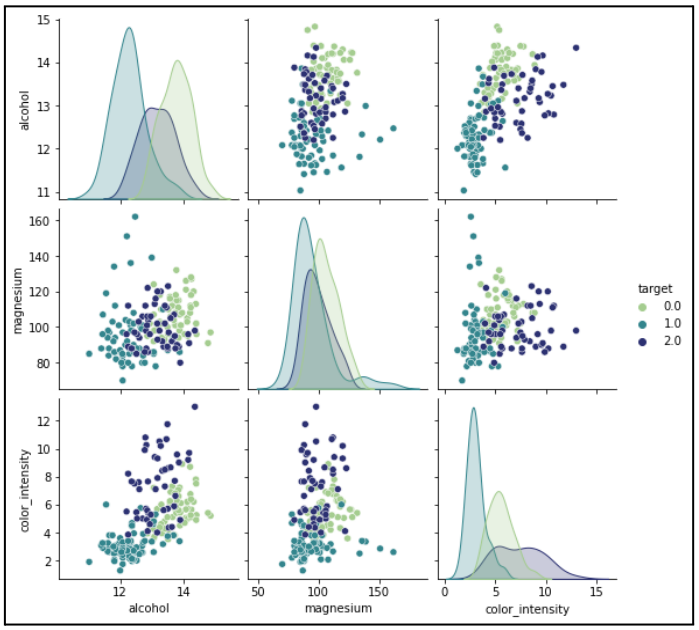
INITIAL

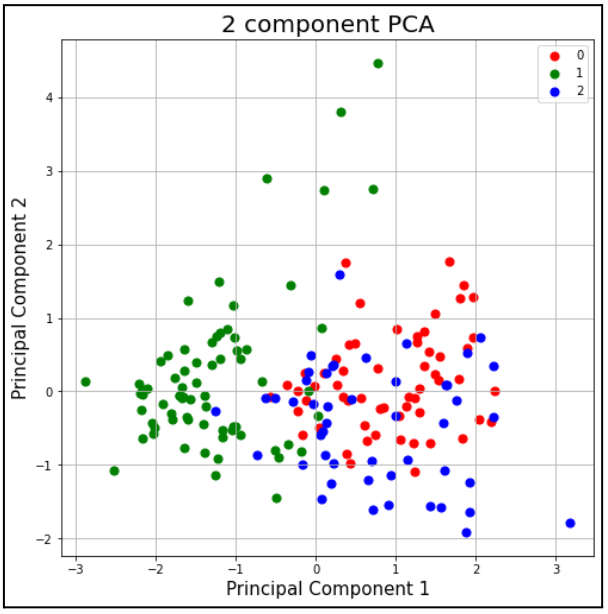
	alcohol	magnesium	color_intensity	target
0	14.23	127.0	5.64	0.0
1	13.20	100.0	4.38	0.0
2	13.16	101.0	5.68	0.0
3	14.37	113.0	7.80	0.0
4	13.24	118.0	4.32	0.0

FINAL

	principal component 1	principal component 2	target
0	1.966082	1.283706	0.0
1	-0.015728	0.073984	0.0
2	0.334609	-0.067353	0.0
3	2.243218	0.006116	0.0
4	0.541480	1.206477	0.0







Conclusion: We learnt about Principal Component Analysis and analysed the wine dataset from sklearn.