Faculdade de Tecnologia Baixada Santista Rubens Lara
Curso Superior de Tecnologia em Ciência de Dados

Anderson Portes do Nascimento
Kaylane Chavier Costa

PCA - Análise dos componentes principais

Santos, SP
2023

1 Introdução

O tema escolhido para desenvolvimento do trabalho foi vinhos. Através de uma pesquisa bibliográfica, foi encontrado um dataset, no site Kaggle, dos tipos de vinho e suas respectivas qualidades e, através dele, foi realizado o PCA (análise dos componentes principais).

2 Codificação

2.1 Passo 1 - Baixar o dataset via kaggle

Acessando o link https://www.kaggle.com/datasets/yasserh/wine-quality-dataset é possível visualizar as informações sobre a base de dados utilizada e baixar o arquivo csv clicando em "Download".

2.2 Passo 2 - Criar o arquivo .py ou .ipynb (notebook python)

Cria-se os arquivos com extensão .py e .ipynb para realizar a codificação do PCA.

2.3 Passo 3 - Importar as bibliotecas que serão utilizadas no desenvolvimento

Importou-se as bibliotecas que serão utilizadas no desenvolvimento no arquivo .ipynb:

import numpy as np: Biblioteca do numpy usada para realizar as operações matematicas dentro do código;

import pandas as pd: Biblioteca usada para importar a base de dados .csv;

import matplotlib.pyplot as plt: Biblioteca usada para plotar os gráficos;

from sklearn.preprocessing import StandardScaler: Biblioteca usada para normalizar os dados.

Figura 1: Código do 3º passo

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler
```

Fonte: Elaborada pelo autor.

2.4 Passo 4 - Importar o dataset e remover dados nulos

A função "pd.read csv"serve para importar o dataset dentro do código, neste caso, nomeado como "data.csv". A função "dropna"remove os dados nulos do dataset o parametro "inplce=true"indica que a própria variavel "data"será reescrita.

Figura 2: Código do 4º passo

```
data = pd.read_csv('data.csv')
data.dropna(inplace=True)
```

Fonte: Elaborada pelo autor.

2.5 Passo 5 - Criação de labels para, posteriormente, distinguir a qualidade dos vinhos

A variavel "mean quality"contém a media de qualidade dos vinhos, enquanto a variavel "is good"será um vetor contendo apenas "True"ou "False", indicando se a qualidade do vinho do índice está maior ou igual a média de qualidade, exemplificando caso o indice 0 dessa variável for "True", logo o primeiro vinho possuí uma qualidade maior ou igual a média.

Figura 3: Código do 5º passo

```
mean_quality = data.quality.mean()
is_good = (data.quality >= mean_quality).to_numpy()
```

Fonte: Elaborada pelo autor.

2.6 Passo 6 - Remover colunas desnecessárias

De início, o dataset contém duas colunas que não serão utilizadas na análise: a "quality- esse será o ponto de análise, por isso ela será retirada do dataset - e a "Id", que não contribui na análise.
a função drop recebe como primeiro parametro um array com o nome das colunas que serão removidas. O "axis=1"indica que todos os elementos dessa coluna serão removidos e o "inplace=True"indica que a variavel será reescrita.

Figura 4: Código do 6º passo

```
data.drop(['Id','quality'],axis=1,inplace=True)
```

Fonte: Elaborada pelo autor.

2.7 Passo 7 - Transformar o dataset do pandas em uma estrutura do numpy e realizar a normalização dos dados

A função "data.to numpy"transforma o dataframe do pandas em uma matriz do numpy, e, logo abaixo, usa-se a classe "StandarScaler", executando a função "fit transform"para normalizar os elementos da matriz onde a formula sera:

(valor do elemento - média) / desvio padrão

Com isso, os dados estarão na mesma escala.

Figura 5: Código do 7º passo

```
np_data = data.to_numpy()
np_data = StandardScaler().fit_transform(np_data)
```

Fonte: Elaborada pelo autor.

2.8 Passo 8 - Realizar a decomposição por valores singulares

a função "np.linalg.svd"retorna uma tripla, contendo as matrizes "U", "S"e "Vt". Neste caso, só precisará da matriz "Vt"dos componentes pricipais.

Figura 6: Código do 8º passo

```
U,S,Vt = np.linalg.svd(np_data)
```

Fonte: Elaborada pelo autor.

import streamlit as st: Para criar a interface;

import pickle: Para ler as variaveis em memória;

import pandas as pd: Para importar a base de dados.

2.9 Passo 9 - Projetar os 5 principais componentes na matriz inicial

A variável "principal components"será uma matriz contendo os 5 principais componentes. Já a variavel "pca data"será a projeção entre a matriz inicial e os componentes principais.

Figura 7: Código do 9º passo

```
principal_components = Vt[:5,:].T
pca_data = np_data @ principal_components
```

Fonte: Elaborada pelo autor.

2.10 Passo 10 - Plotar todas as combinações dos principais componentes usando o scatterplot 2d

Esse código realizará a comparação entre cada componente principal a partir de um gráfico "scatterplot"e, passando a variável "color"s dentro do plot, será possivel observar os vinhos com qualidade abaixo da média (pontos em vermelho) e os vinhos com qualidade acima ou igual a média (pontos azuis).

Figura 8: Código do 10º passo

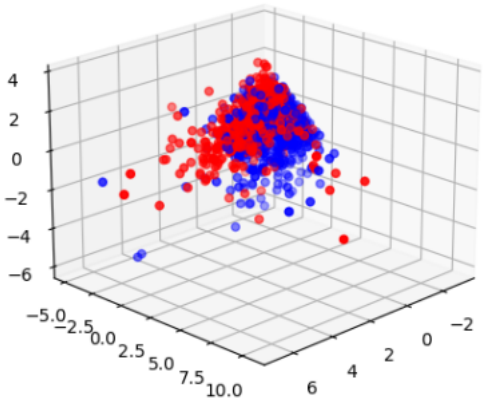
```
for i in range(5):
    for j in range(5):
        for k in range(5):
            if(i  != j and  j != k and k != i):
                print(f'i:{i}; j:{j};k:{k}')
                fig = plt.figure()
                ax = fig.add_subplot(111, projection='3d')
                ax.scatter(pca_data[:, i], pca_data[:, j], pca_data[:, k], c=colors)
                ax.view_init(elev=20, azim=45)
                plt.show()
```

Fonte: Elaborada pelo autor.

3 Resultado

O melhor padrão encontrado foi utilizando, respectivamente, os principais componentes: 2, 4 e 3, onde pode-se observar o agrupamento de vinhos com qualidade abaixo da média - vermelhos - e os vinhos com qualidade igual ou acima da média - azuis.

Figura 9: Resultado



Fonte: Elaborada pelo autor.

4 REPOSITÓRIO DO PROJETO

Anderson Portes do Nascimento: https://github.com/Anderson-Portes/wine-quality-pca

Kaylane Chavier Costa: https://github.com/kaychavier/wine-quality-pca

Dataset: https://www.kaggle.com/datasets/yasserh/wine-quality-dataset