Лабораторная работа №2

Понижение размерности пространства признаков

Цель:

Ознакомиться с методами понижения размерности данных из библиотеки *Scikit Learn*

Выполнение:

Загрузка данных

1. Загрузить датасет (прилагается, файл с именем glass) . Данные представлены в виде csv таблицы.
2. Создать Python скрипт. Загрузить датасет в датафрейм, и разделить данные на описательные признаки и признак отображающий класс.

```
import pandas as pd
import numpy as np

df = pd.read_csv('glass.csv')

var_names = list(df.columns) #получение имен признаков

labels = df.to_numpy('int')[:,-1] #метки классов
data = df.to_numpy('float')[:,:-1] #описательные признаки
```

3. Провести нормировку данных к интервалу [0 1]

```
from sklearn import preprocessing

data = preprocessing.minmax_scale(data)
```

4. Построить диаграммы рассеяния для пар признаков. Самостоятельно определите соответствие цвета на диаграмме и класса в датасете

```
import matplotlib.pyplot as plt

fig, axs = plt.subplots(2,4)

for i in range(data.shape[1]-1):
    axs[i // 4, i % 4].scatter(data[:,i],data[:,(i+1)],c=labels,cmap='hsv')

    axs[i // 4, i % 4].set_xlabel(var_names[i])
    axs[i // 4, i % 4].set_ylabel(var_names[i+1])

plt.show()
```

Метод главных компонент

1. Используя метод главных компонент (PCA). Проведите понижение размерности пространства до размерности 2

```
from sklearn.decomposition import PCA

pca = PCA(n_components = 2)

pca_data = pca.fit(data).transform(data)
```

2. Выведите значение объясненной дисперсии в процентах и собственные числа соответствующие компонентам

```
print(pca.explained_variance_ratio_)
print(pca.singular_values_)
```

3. Постройте диаграмму рассеяния после метода главных компонент

```
plt.scatter(pca_data[:,0],pca_data[:,1],c=labels,cmap='hsv')
plt.show()
```

4. Проанализируйте и обоснуйте полученные результаты
5. Изменяя количество компонент, определите количество при котором компоненты объясняют не менее 85% дисперсии данных