**МИНОБРНАУКИ РОССИИ**
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**
**Кафедра МОЭВМ**

# ОТЧЕТ

**по лабораторной работе №2**

**по дисциплине «Машинное обучение»**

**Тема: Понижение размерности пространства признаков**

Студент гр. 6304 Ковынев М.В.

Преподаватель Жангиров Т.Р.

Санкт-Петербург
2020

**Цель**

Ознакомиться с методами понижения размерности данных из библиотеки Scikit Learn

**Ход работы**

1. Загружен датасет по ссылке: https://www.kaggle.com/uciml/glass. Данные представлены в виде csv таблицы.
2. Создан Python скрипт. Загружен датасет в датафрейм, и разделены данные на описательные признаки и признак отображающий класс
3. Проведена нормировку данных к интервалу [0 1].
4. Построить диаграммы рассеяния для пар признаков.

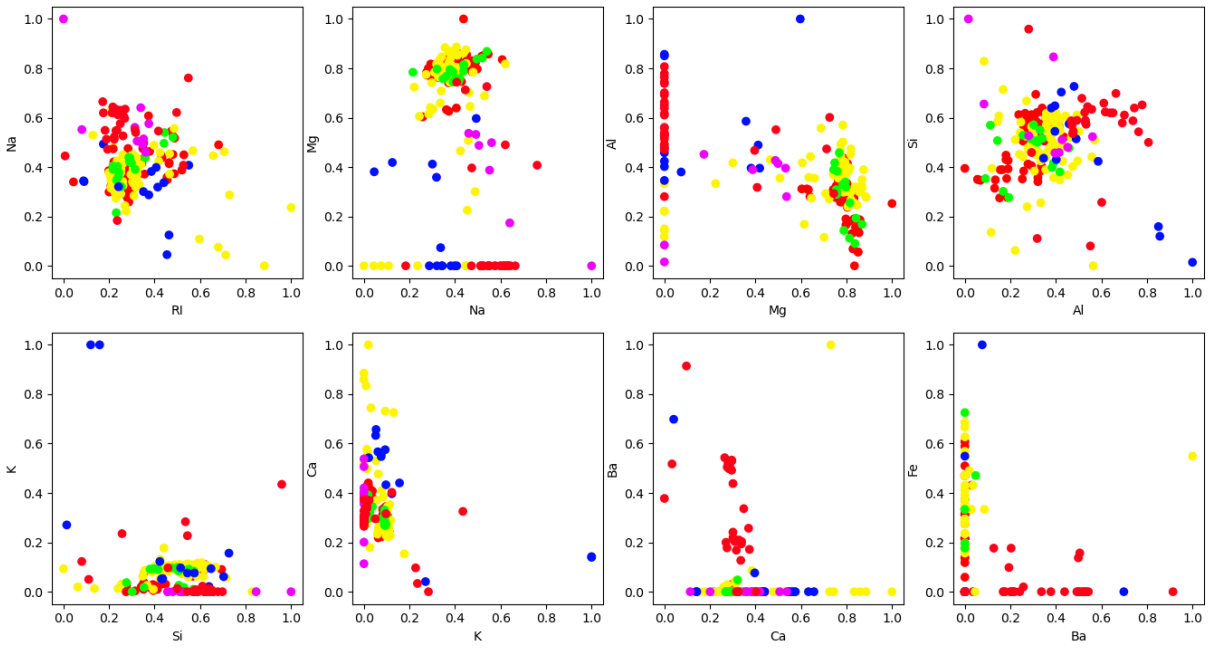


Рисунок 1 — Диаграммы рассеяния

5. Соответствие цвета и класса

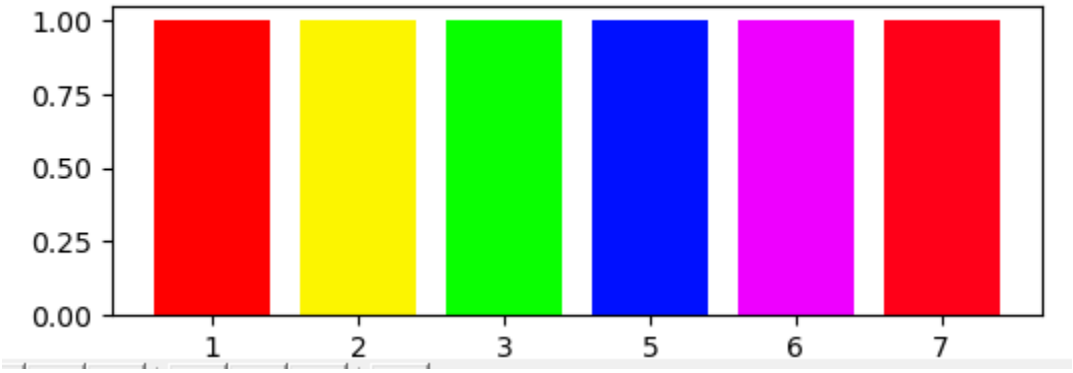


Рисунок 2 — Соответствие

6. Используя метод главных компонент (PCA). Проведено понижение размерности пространства до размерности 2.
7. Выведены значение объясненной дисперсии в процентах и собственные числа:
    - объясненная дисперсия - [0.45429569 0.17990097]
    - собственные числа - [5.1049308  3.21245688]

   По двум компонентам дисперсия 63.3% - недостаточно.
8. Построена диаграмма рассеяния после метода главных компонент

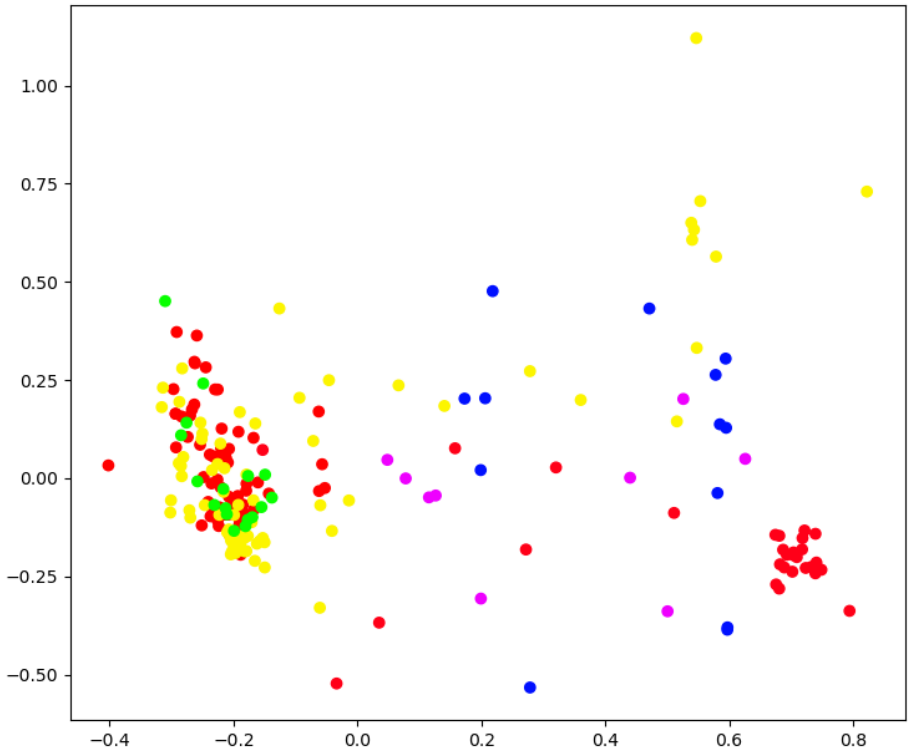


Рисунок 3 — Диаграмма рассеяния для 2х компонент

9. Изменяя количество компонент, определено количество, при котором компоненты объясняют не менее 85% дисперсии данных. Результат – 4.

```
1 0.4542956890746849 is bigger than 0.85 -  False
2 0.6341966621042779 is bigger than 0.85 -  False
3 0.7606912558548664 is bigger than 0.85 -  False
4 0.858669730510272 is bigger than 0.85 -  True
5 0.9272937149511479 is bigger than 0.85 -  True
6 0.9694347221994032 is bigger than 0.85 -  True
7 0.9955326243472863 is bigger than 0.85 -  True
8 0.9998605862637864 is bigger than 0.85 -  True
9 1.0 is bigger than 0.85 -  True
```

Рисунок 4 — Дисперсия от количества компонент.

10. Используя метод inverse_transform восстановлены данные. Для сравнения данных вычислено mse.

```
Al      0.005693
Ba      0.007382
Ca      0.000912
Fe      0.000916
K       0.009069
Mg      0.000439
Na      0.010299
RI      0.000866
Si      0.002308
```

11. Исследован метод главных компонент при различных параметрах svd_solver.

    Значения svd_solder – auto, full, arpack, randomized.

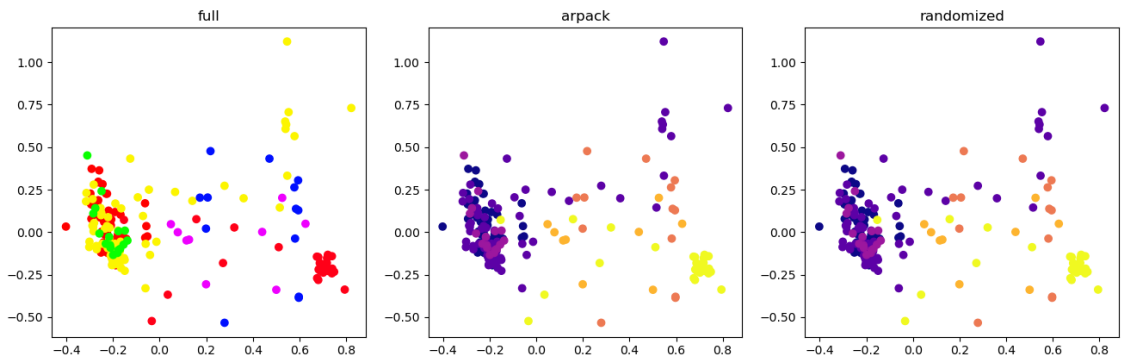


Рисунок 5 — Диаграммы рассеяния.

Дисперсия компонент:

- Full – 0.858669730510272
- Arpack – 0.8586697305102715
- Randomized - 0.8586697305102718

12. По аналогии с PCA исследован KernelPCA для различных параметров kernel и различных параметрах для ядра.

KernelPCA при kernel=linear ведет себя как PCA. В поле lambdas_ хранятся собственные числа. Сравним объясненные дисперсии для различных ядер на 4 компонентах.

| | **PC1** | **PC2** | **PC3** | **PC4** | **Кумул. сумма** | **Объясненные дисперсии** |
|---|---|---|---|---|---|---|
| *Liner* | 26.06 | 10.31 | 7.25 | 5.6 | 49.25 | 0.858 |
| *Poly* | 10.91 | 4.31 | 3.11 | 2.36 | 20.7 | 0.361 |
| *Rbf* | 5.35 | 2.01 | 1.49 | 1.11 | 9.97 | 0.173 |
| *Sigmoid* | 1.00 | 0.39 | 0.27 | 0.21 | 1.89 | 0.033 |
| *Cosine* | 18.31 | 6.47 | 4.69 | 3.57 | 33.06 | 0.576 |

13. SparcePCA

- Компоненты PCA:

  [

  [0.03420952 0.11044243 -0.90903503 0.24901968 0.05079549 -0.00269769 0.14094732 0.26682812 -0.06801349]

  [0.51327262 -0.19867029 -0.11710045 -0.34736315 -0.21642569 -0.12930091 0.50234458 -0.16429176 0.46883578]

  ]

- Компоненты Sparse Lars:

  [

  [ 0. 0. 0.99804243 -0.03718353 0. 0. 0. -0.0502861 0. ]

  [ 0. 0. 0. 0. 0. 0. 0. 0. 1. ]

  ]

- Компоненты Sparse cd:

  [

  [ 0. 0. 0.99804243 -0.03718353 0. 0. 0. -0.0502861 0. ]

  [ 0. 0. 0. 0. 0. 0. 0. 0. 1. ]

  ]

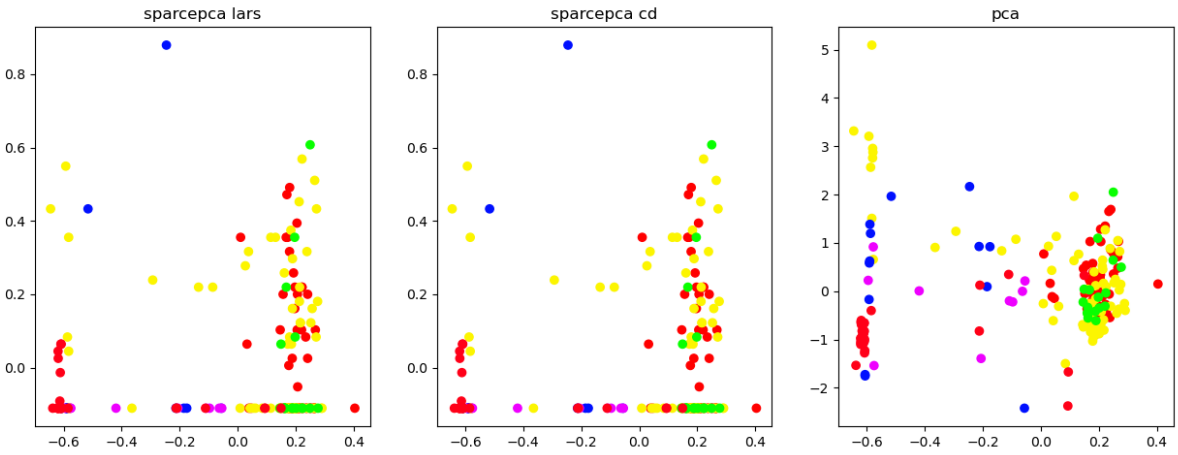


Рисунок 6 — Диаграммы рассеяния SparcePCA и PCA.

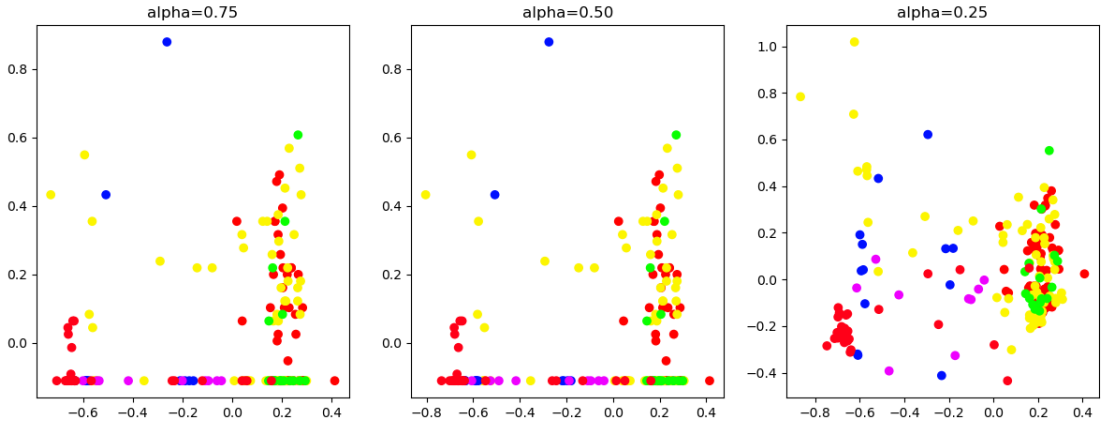


Рисунок 6 — Диаграммы рассеяния 2х компонент при разных alpha.

Значения компонент при разных alpha

- Alpha = 0.75

[[ 0.  0.  0.985 -0.117  0.  0.  -0.005 -0.129  0. ]
 [ 0.  0.  0.  0.  0.  0.  0.  0.  1. ]]

- Alpha = 0.50

[[ 0.  -0.001  0.965 -0.173  0.  0.  -0.061 -0.186  0. ]
 [ 0.  0.  0.  0.  0.  0.  0.  0.  1. ]]

- Alpha = 0.25

[[-0.008 -0.052  0.943 -0.203  0.  0.  -0.126 -0.228  0. ]
 [ 0.477 -0.169  0.  -0.288 -0.191 -0.022  0.437 -0.073  0.654]]

14.Факторный анализ

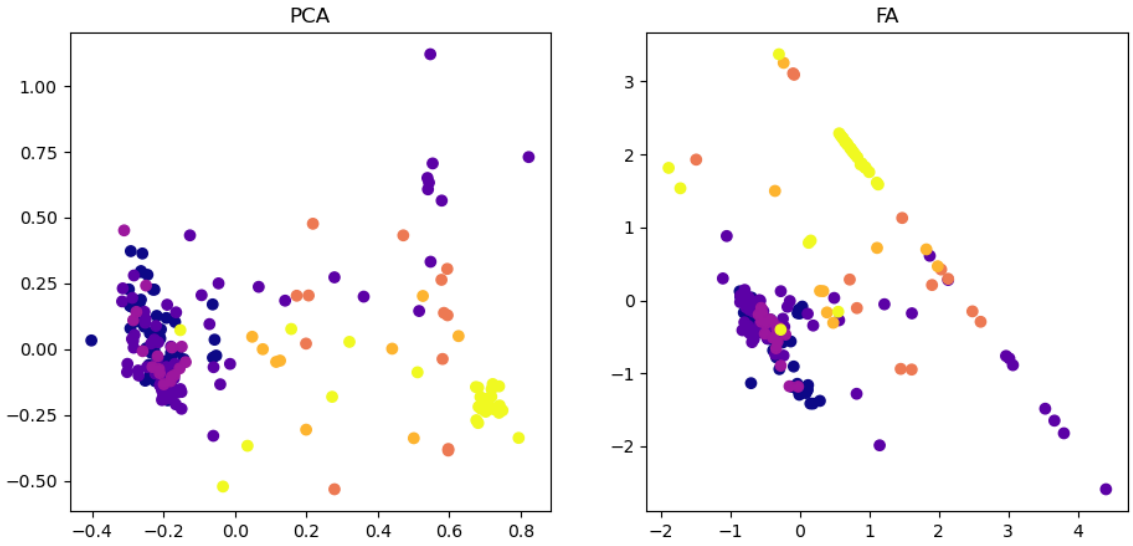


Рисунок 6 — Диаграммы рассеяния PCA, FA

1. PCA – компоненты ортогональны друг другу, FA – не обязательно
2. PCA – метод уменьшения размерности данных, FA – метод поиска скрытых переменных (факторов)
3. PCA – линейная комбинация наблюдаемой переменной, FA – наблюдаемые переменные есть линейная комбинация данных

**Вывод**

В ходе выполнения данной лабораторной работы было осуществлено ознакомление с методами понижения размерности данных из библиотеки Scikit Learn.

В ходе работы выявлено, что разное количество компонент в PCA объясняет разное количество дисперсии данных.

KernelPCA используется для поиска нелинейных зависимостей в данных. KernelPCA сводится к PCA с линейным ядром, SparsePCA – при alpha = 0 (параметр регуляризации)