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

ОТЧЁТ

по лабораторной работе №2
по дисциплине «Машинное обучение»
Тема: Понижение размерности пространства признаков

Студент гр. 6304 _____ Корытов П.В.
Преподаватель _____ Жангиров Т.Р.

Санкт-Петербург
2020

1. Цель работы

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

2. Выполнение

2.1. Загрузка данных

1. Произведена загрузка и нормировка данных. Построенна диаграмма рассеяния для пар признаков; диаграмма представлена на 1.

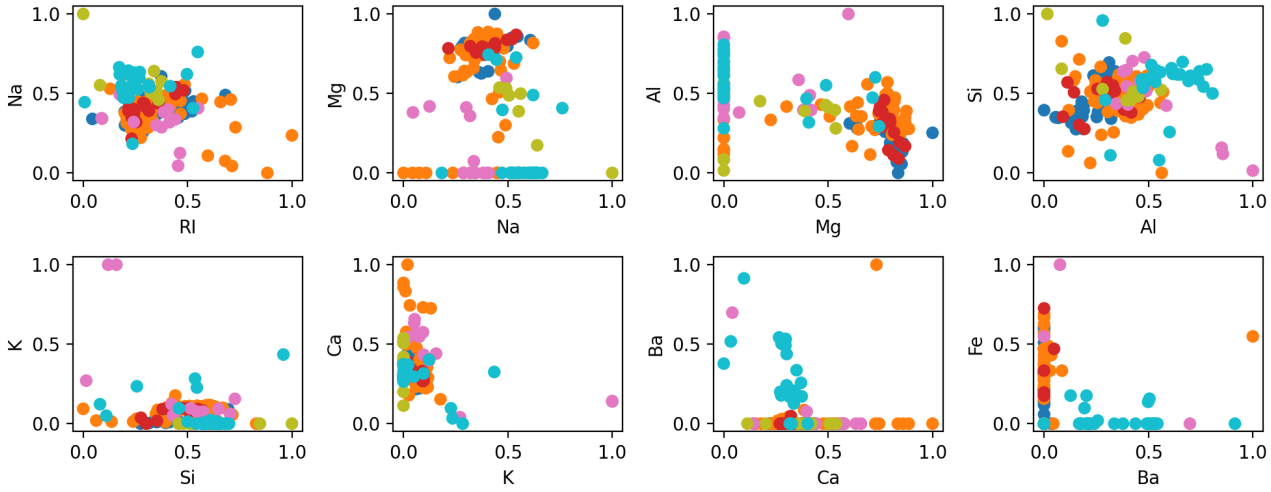

Рисунок 1 – Диаграмма рассеяния

2. Определено соответствие цвета на диаграмма и класса в наборе данных. Результат представлен на рис 2

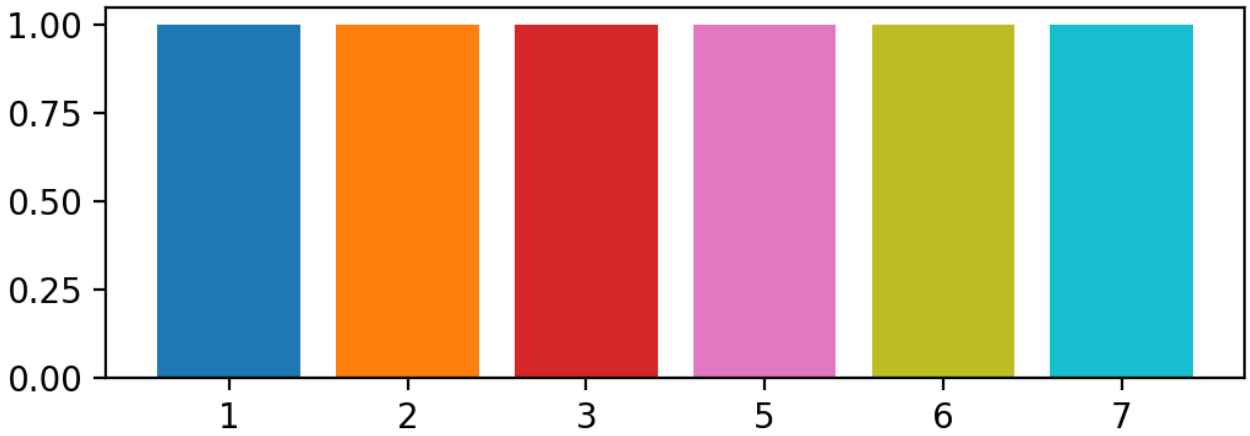

Рисунок 2 – Соответствие цвета и класса

2.2. Метод главных компонент

1. Произведено понижение размерности до 2. Значения объясненной дисперсии и сообственные числа:

```
explained_variance_ratio_: [0.45429569 0.17990097]
singular_values: [5.1049308  3.21245688]
```

2. Построена диграмма рассеяния после применения метода (рис. 3)

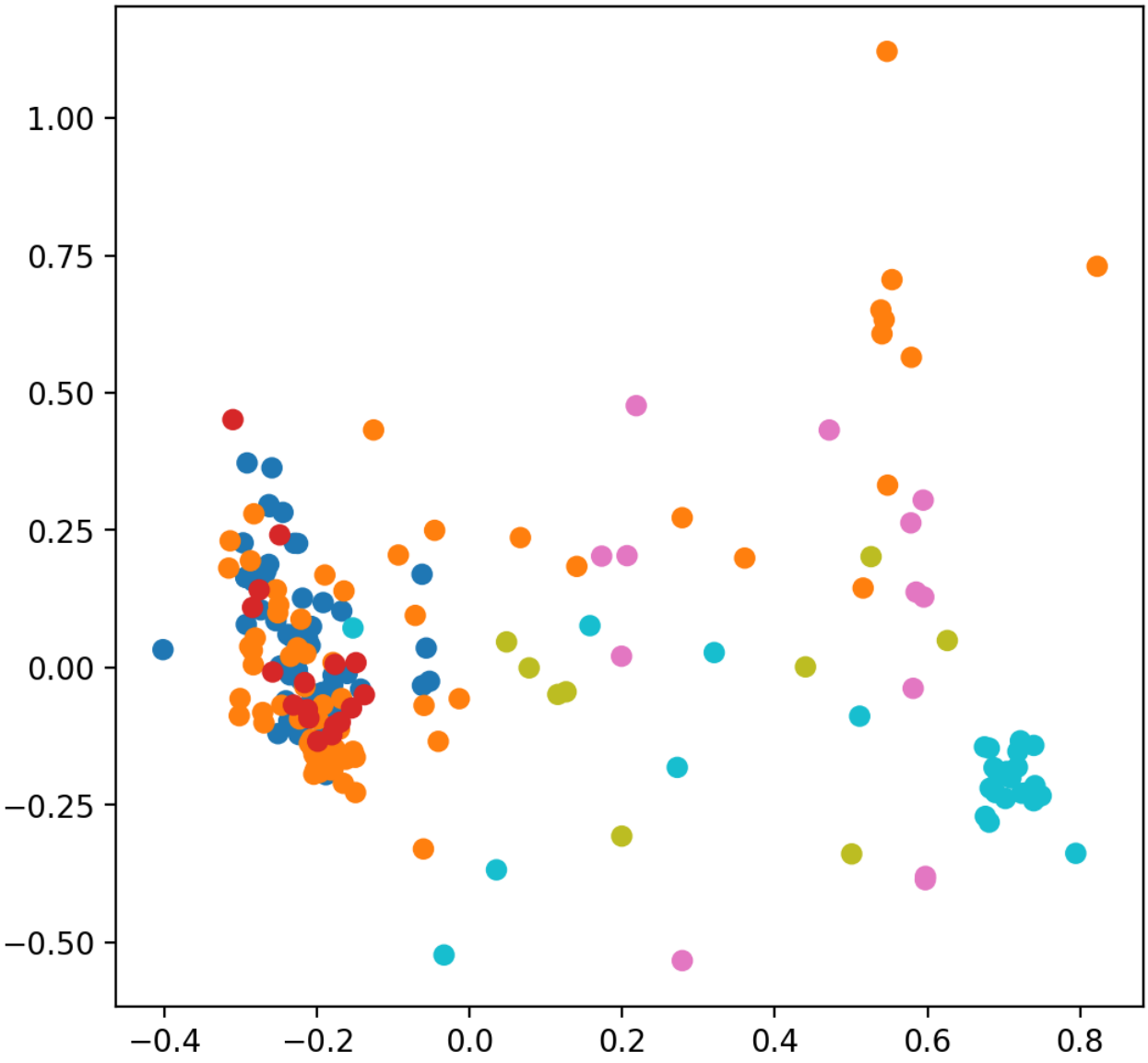

Рисунок 3 – Диаграмма рассеяния

3. Установлено число компонент, необходимое для описание 85% дисперсии. Результат на таблице 1.
4. Произведено обратное преобразование. Построена диаграмма рассеяния, аналогичная рис 3. Результат на рис. 4.

Таблица 1. Зависимость объясненной дисперсии от числа компонент

n_components	variance
2	0.634197
3	0.760691
4	0.85867
5	0.927294
6	0.969435
7	0.995533
8	0.999861
9	1

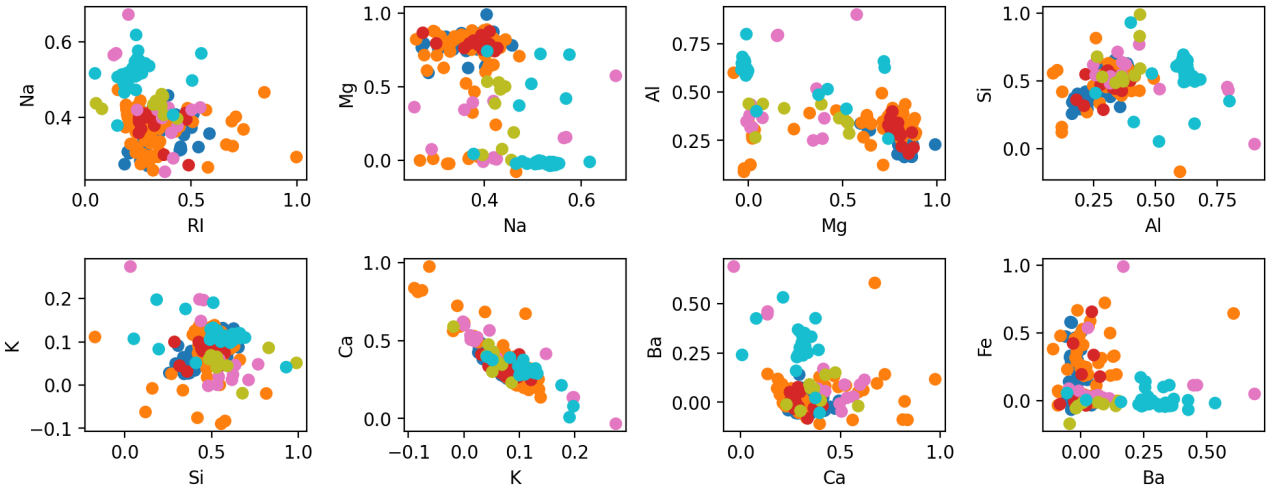

Рисунок 4 – Диаграмма рассеяния для восстановленных данных

Визуально заметны различия между рис. 3 и рис. 4.

5. Произведено исследование метода при различных параметрах `svd_solver`. Результаты представлены в таблице 2.

Таблица 2. Зависимость объясненной дисперсии от числа компонент и `svd_solver`

n_components	auto	full	arpack	randomized
2	0.634197	0.634197	0.634197	0.634197
3	0.760691	0.760691	0.760691	0.760691
4	0.85867	0.85867	0.85867	0.85867
5	0.927294	0.927294	0.927294	0.927294
6	0.969435	0.969435	0.969435	0.969435
7	0.995533	0.995533	0.995533	0.995533
8	0.999861	0.999861	0.999861	0.999861

Различия в результатах между различными методами на обнаружено. Предположительно, это вызвано малым размером набора данных.

2.3. Модификации метода главных компонент

2.3.1. Ядерный PCA

Произведено исследование ядерного PCA с различными ядрами и параметрами.

1. При использовании линейного ядра `KernelPCA` работает также, как `PCA`.

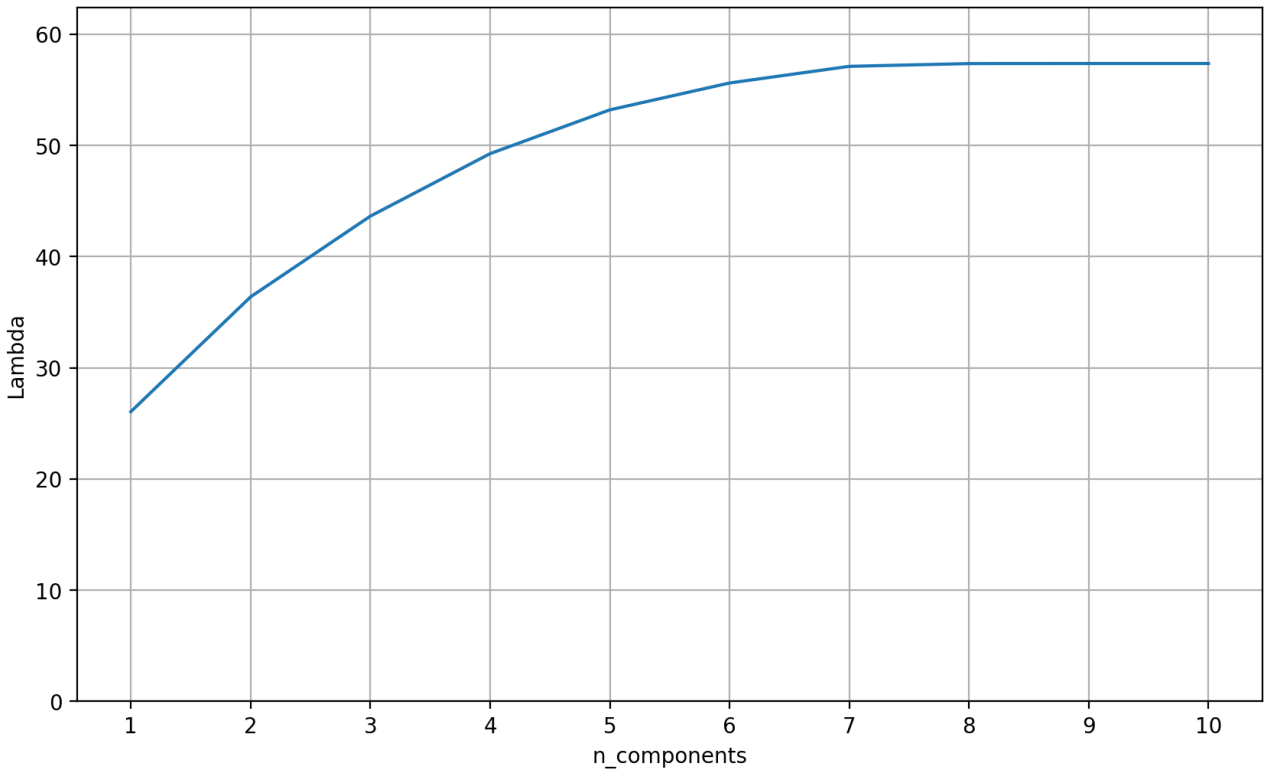

Рисунок 5 – Кумулятивная сумма собственных чисел

На рис 5 изображена кумулятивная сумма собственных чисел ядерной матрицы для ядерного PCA с линейным ядром.

Собственное число показывает количество объясненной дисперсии для данной главной компоненты. Таким образом, структура роста выбранной метрики поможет определить правильность подбора параметров.

В данном случае метрика принимает максимальное значение на 9 компонентах, т.к. линейное ядро неспособно добавить размерности к подаваемым данным.

2. Аналогичная метрика вычислена для полиномиального ядра (рис. 6).

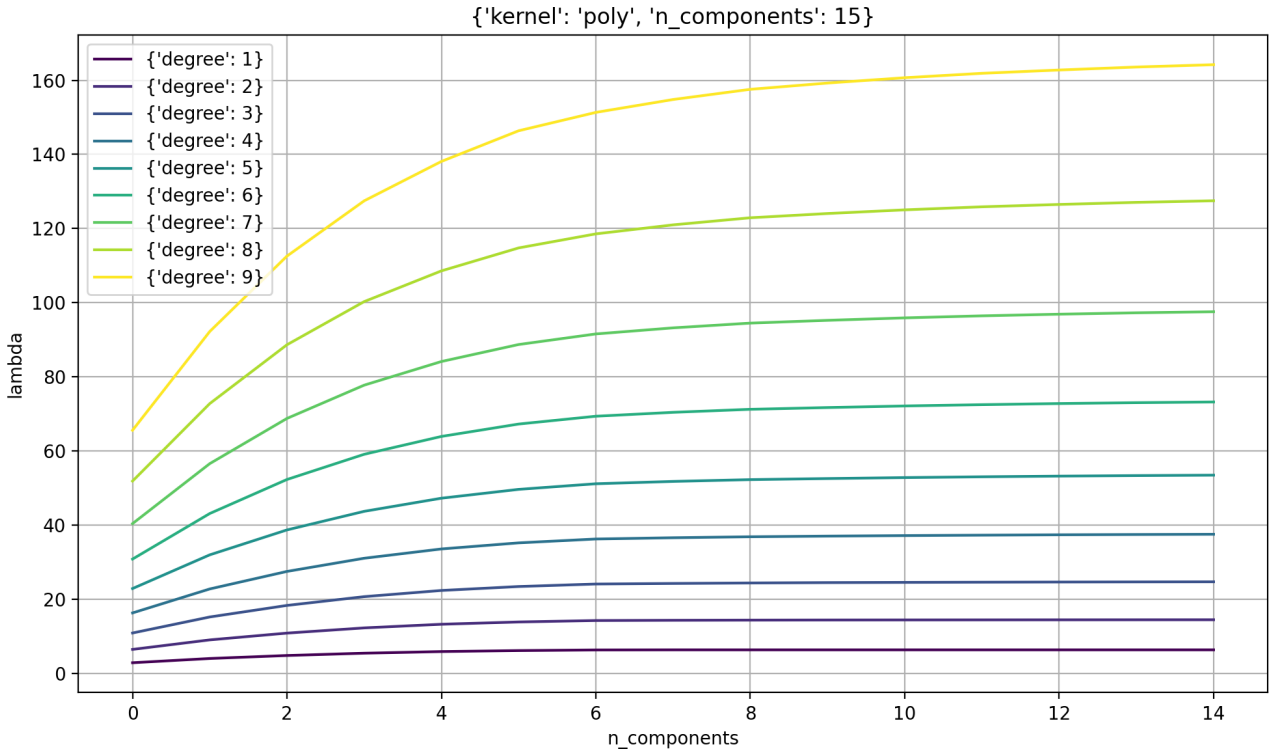

Рисунок 6 – Кумулятивная сумма собственных чисел для полиномиального ядра

Как можно заметить, метрика продолжает рост после 9 исходных компонент при повышении степени ядра. Таким образом, полиномиальное ядро может быть использовано для повышения размерности данных.

Чтобы было проще сравнить данных, указанная сумма нормализована. Результат на рис. 7.

Как видно, в данном случае увеличение степени привело к ухудшению результата; даже если предположить, что во всех случаях на 14 компонентах объяснено 100% дисперсии, важность первых компонент несколько ниже. Таким образом, отсечение нескольких компонент приведет к потере боль-

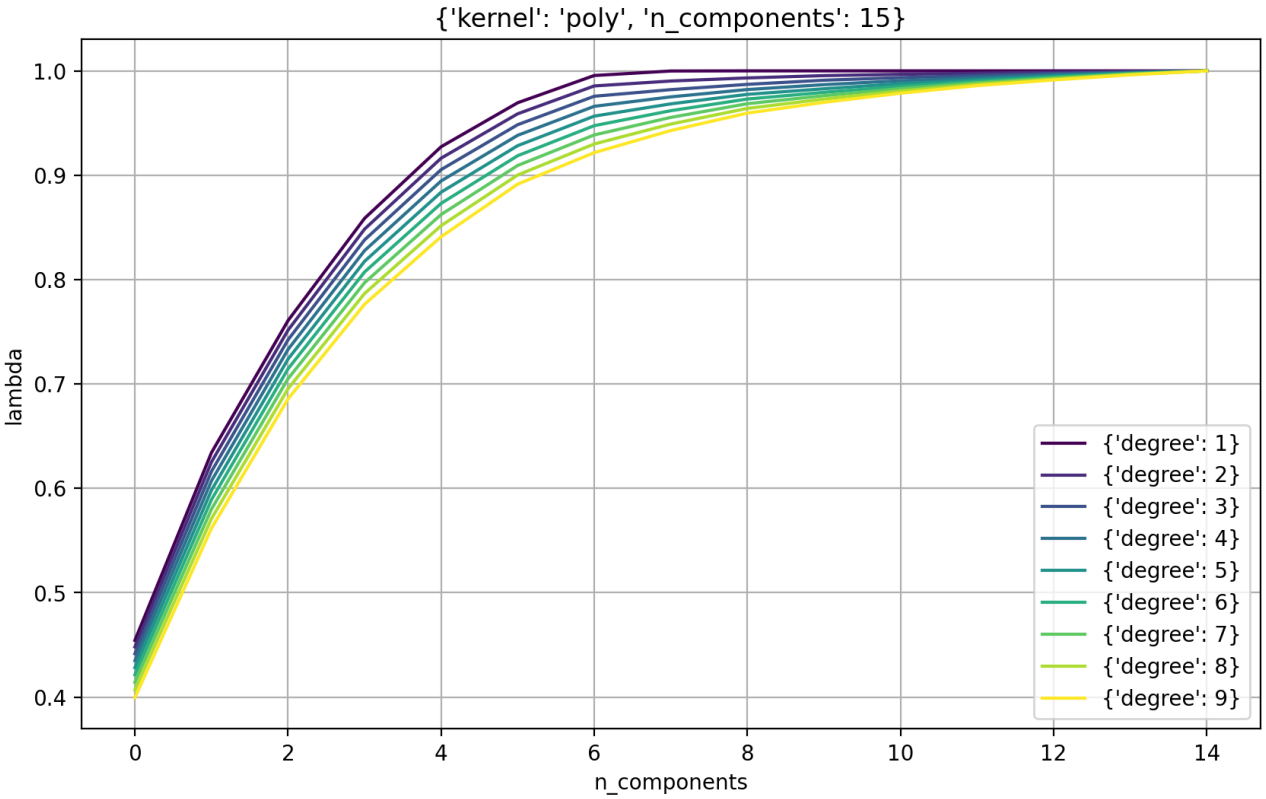

Рисунок 7 – Нормализованная кумулятивная сумма собственных чисел для полиномиального ядра

ших данных.

3. Аналогичный график построен для параметров `gamma` и `coef0`. Результат на рис. 8 и рис. 9

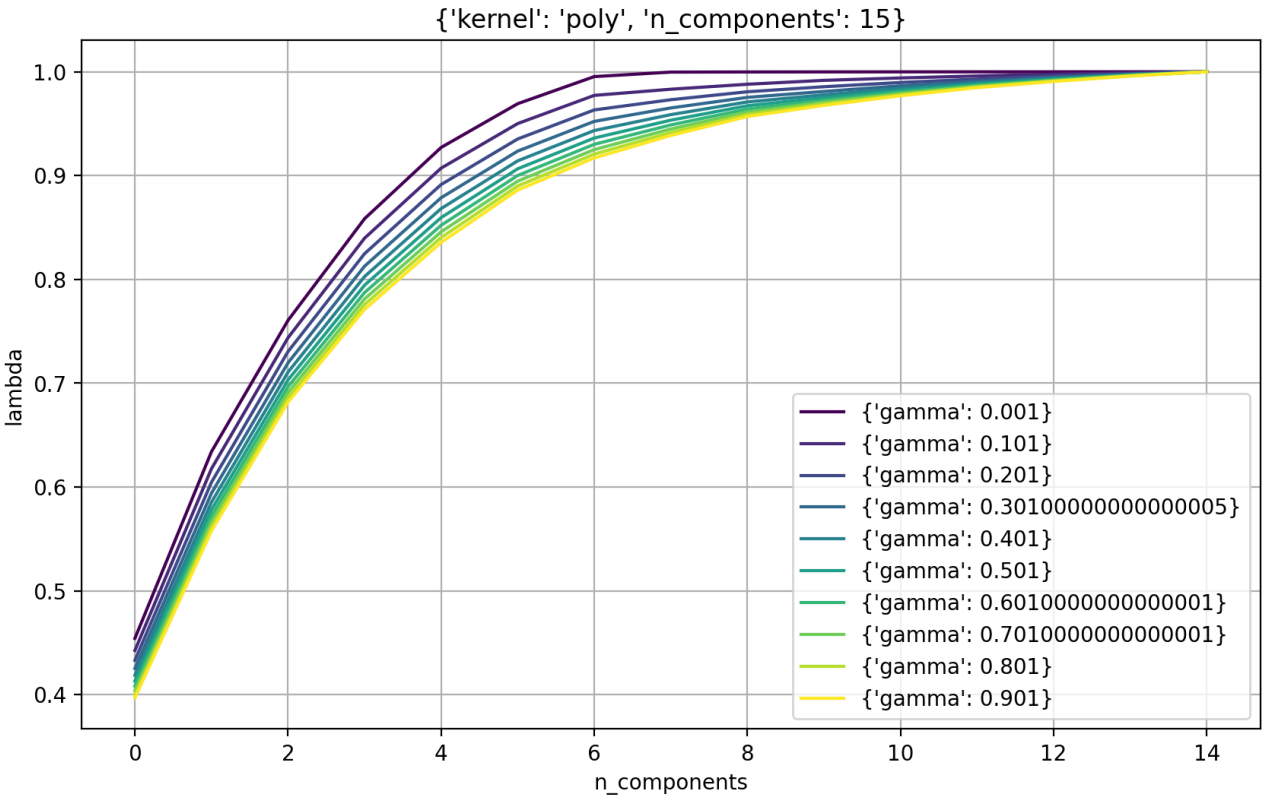

Рисунок 8 – Нормализованная кумулятивная сумма собственных чисел для полиномиального ядра для изменения параметра `gamma`

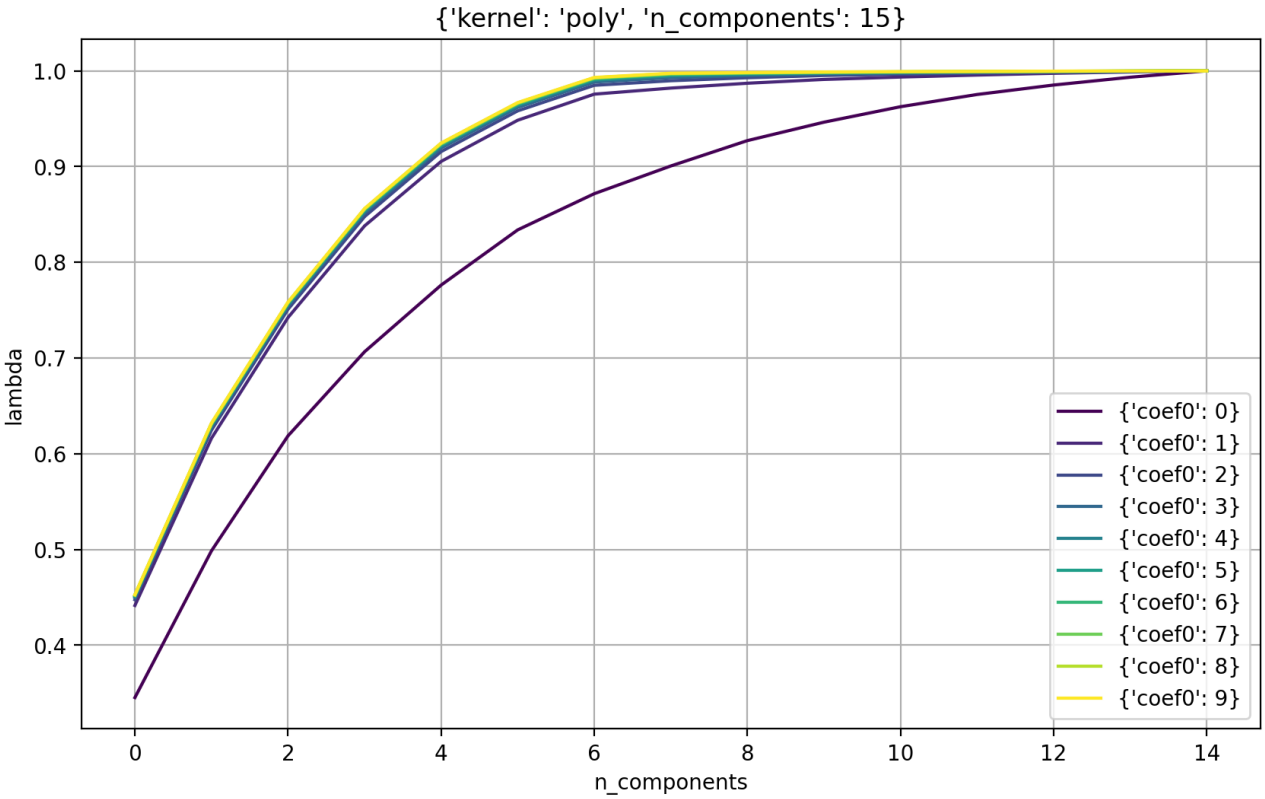

Рисунок 9 – Нормализованная кумулятивная сумма собственных чисел для полиномиального ядра для изменения параметра `coef`

Только 0-й `coef0` существенно ухудшает работу; остальные показывают близкие результаты. Увеличение `gamma` относительно стандартного значения также ухудшает результат.

4. Аналогичные действия произведены для ядра `rbf` и параметра `gamma`, сигмоидального ядра и параметров `gamma` и `coef0`, а также ядра `cosine`. Результаты представлены на рис. 10–13.

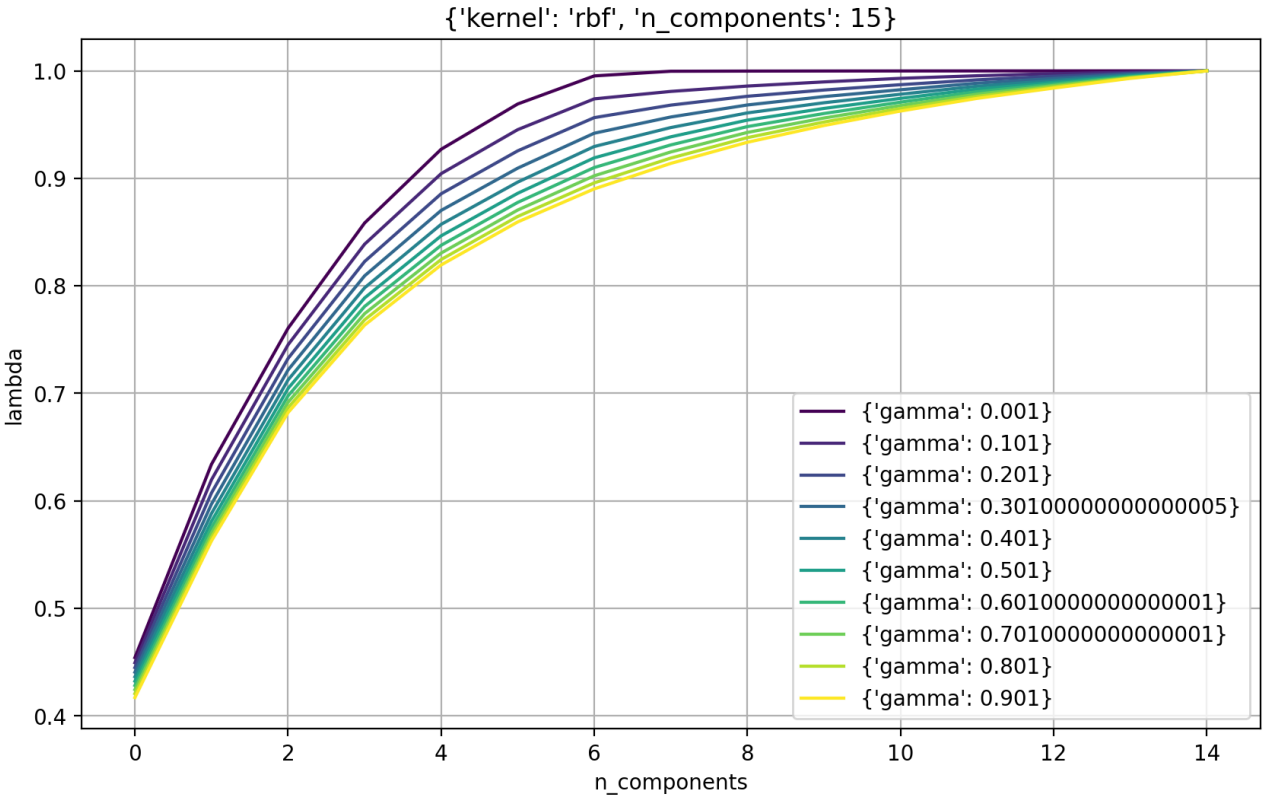

Рисунок 10 – Ядро `rbf`, параметр `gamma`

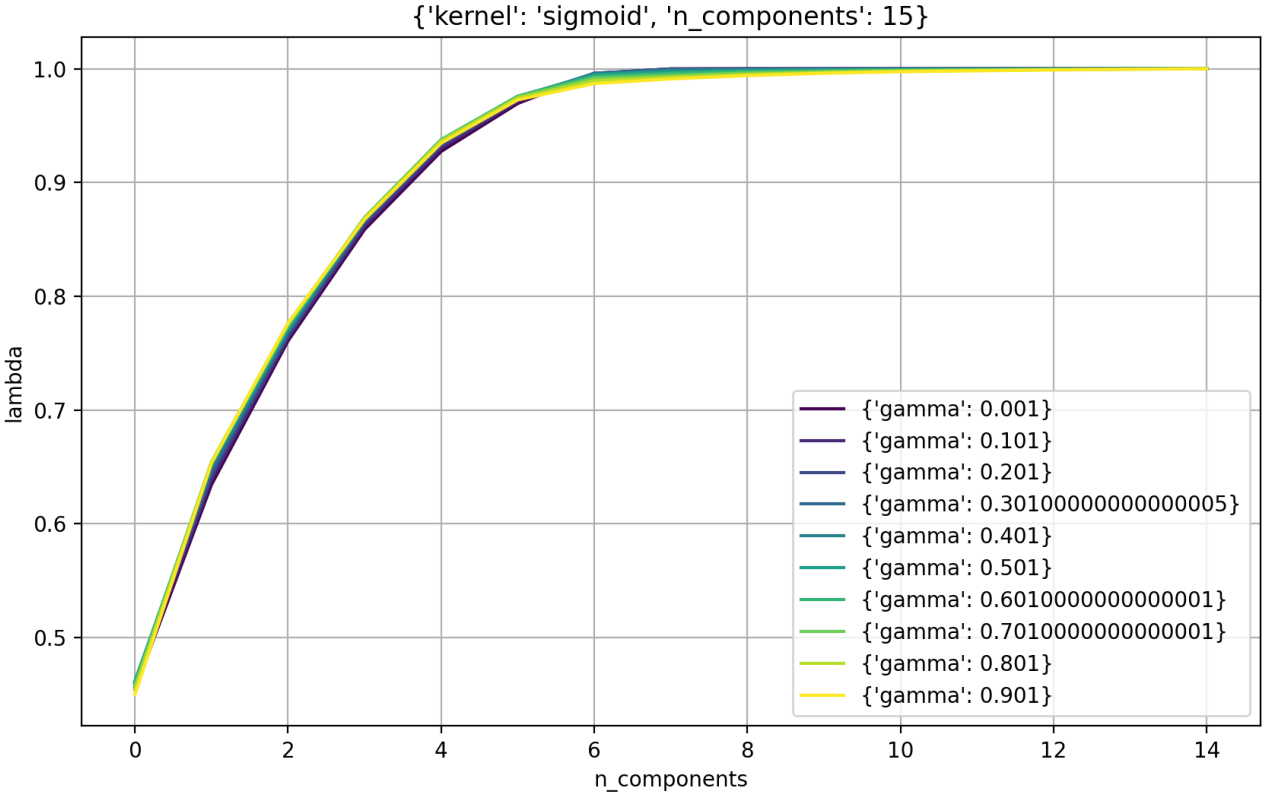

Рисунок 11 – Ядро `sigmoid`, параметр `gamma`

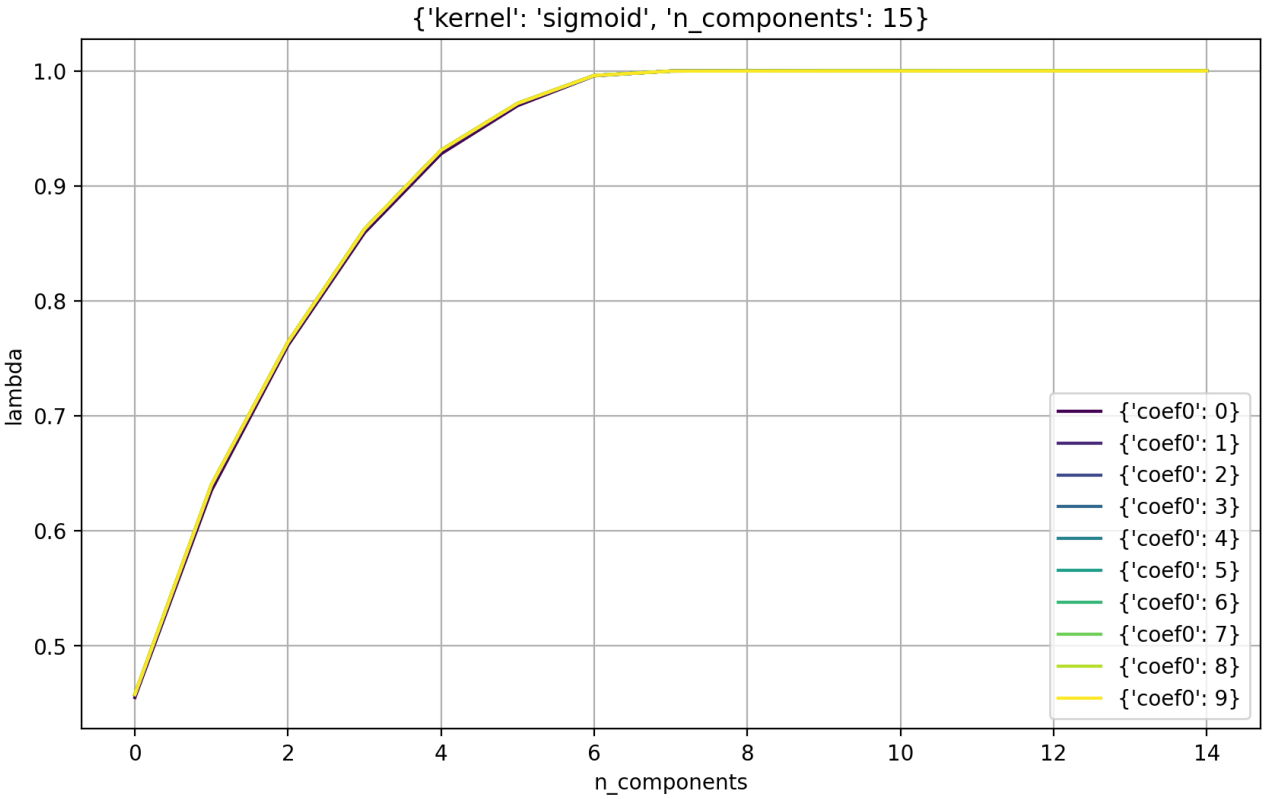

Рисунок 12 – Ядро `sigmoid`, параметр `coef0`

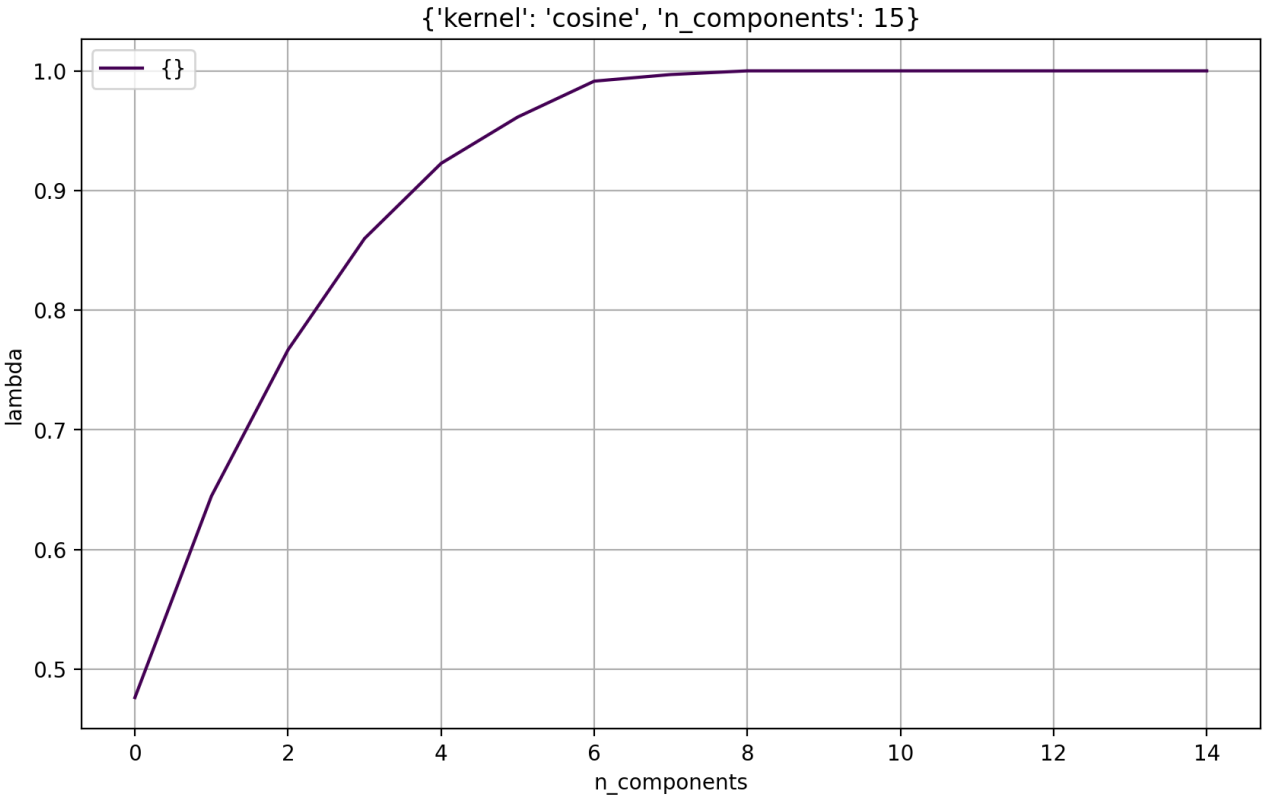

Рисунок 13 – Ядро `cosine`

Таким образом, во всех описанных случаях изменение параметров либо не приводит к существенным изменениям результата, либо изменяет его в худшую сторону. Результаты на рис. 14.

Осталось сравнить работу всех ядер:

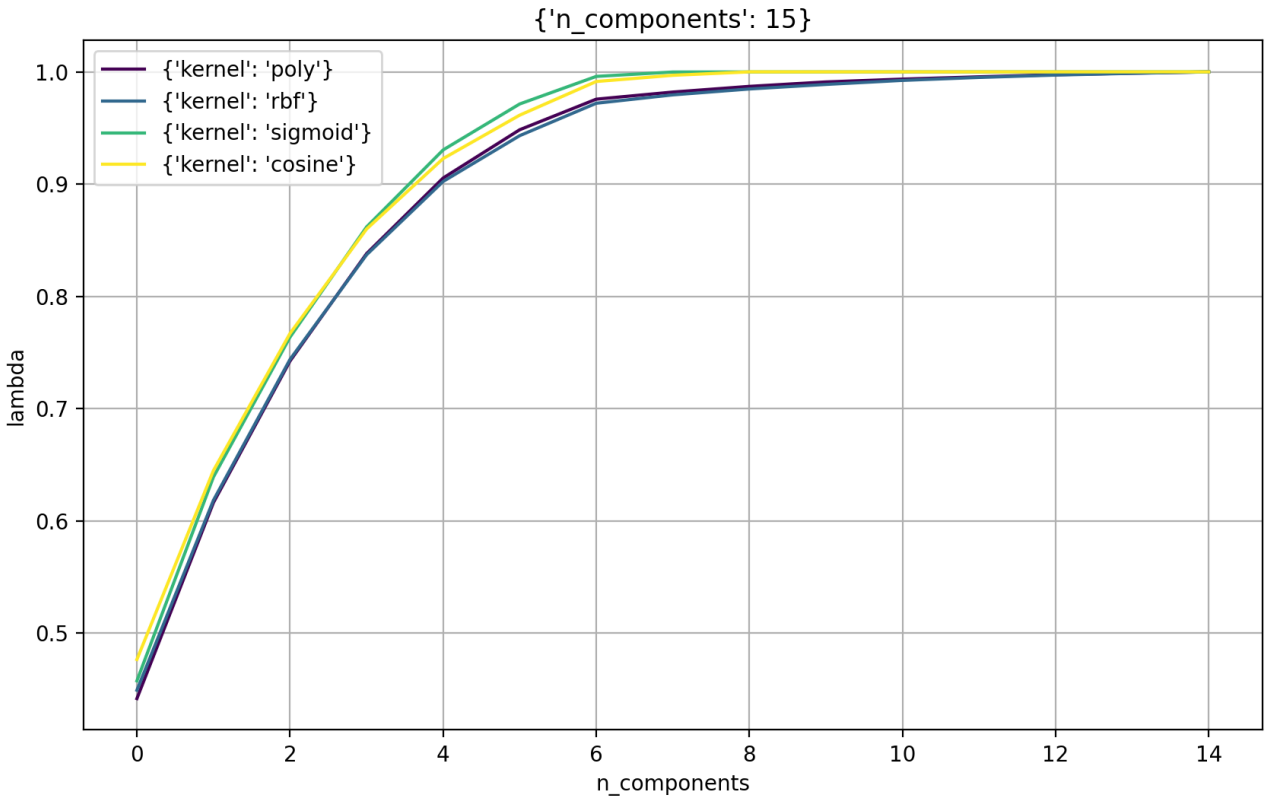

Рисунок 14 – Сравнение ядер

Результаты близки для всех вариантов. Ядро `cosine` смогло выделить лучшую первую и вторую главную компоненту.

2.3.2. SparsePCA

Произведено применение `SparsePCA` к исходным данным. Компоненты обычного PCA и `SparsePCA` представлены далее:

```
Компоненты SparcePCA:
-----  -----  -----  ------  -----  -----  -----  ------  -----
0.000  0.000  0.998  -0.037  0.000  0.000  0.000  -0.050  0.000
0.000  0.000  0.000   0.000  0.000  0.000  0.000   0.000  1.000
-----  -----  -----  ------  -----  -----  -----  ------  -----
Компоненты PCA:
-----  ------  ------  ------  ------  ------  -----  ------  ------
0.034   0.110  -0.909   0.249   0.051  -0.003  0.141   0.267  -0.068
```

```
0.513   -0.199   -0.117   -0.347   -0.216   -0.129   0.502   -0.164   0.469
-----   ------   ------   ------   ------   ------   -----   ------   ------
```

Как видно, `SparsePCA` произвел более разреженные главные компоненты, чем обычный `PCA`. Сравнение диаграмм рассеяния приведено на рис. 15.

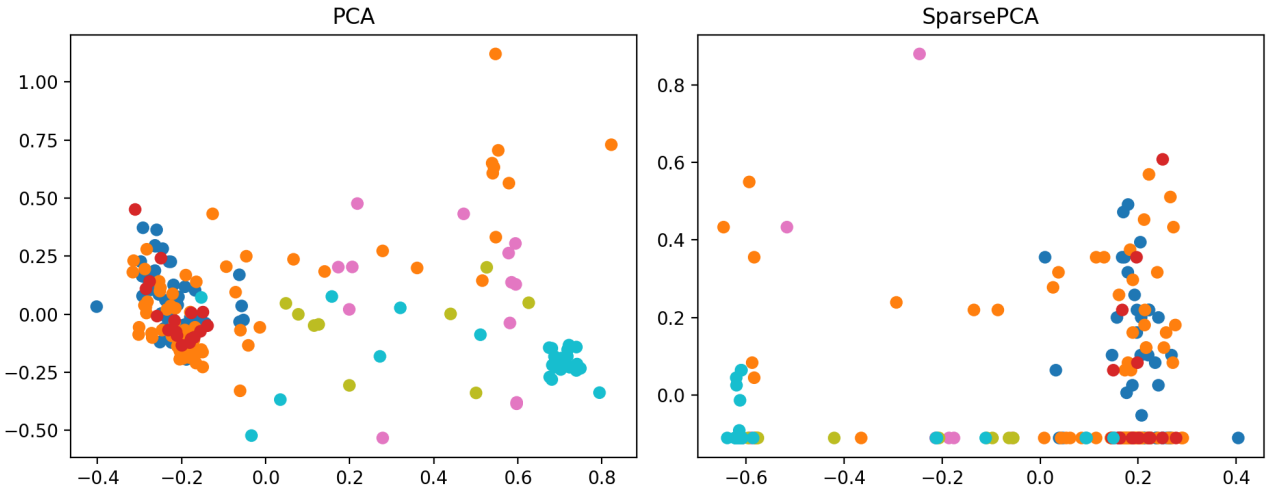

Рисунок 15 – Диаграммы рассеяния для `SparsePCA` и `PCA`

Изменение параметра `alpha` приводит к ослаблению регуляризации, как видно на таблице 3.

Таблица 3. Изменение главных компонент при изменении `alpha`

alpha	i	0	1	2	3	4	5	6	7	8
0.000	1	-0.034	-0.110	0.909	-0.249	-0.051	0.003	-0.141	-0.267	0.068
0.000	2	0.513	-0.199	-0.117	-0.347	-0.216	-0.129	0.502	-0.164	0.469
0.010	1	-0.100	-0.081	0.918	-0.200	-0.020	0.017	-0.205	-0.242	0.002
0.010	2	0.505	-0.210	0.000	-0.376	-0.220	-0.124	0.480	-0.195	0.478
0.100	1	-0.067	-0.071	0.928	-0.203	-0.010	0.000	-0.176	-0.238	0.000
0.100	2	0.505	-0.199	0.000	-0.365	-0.212	-0.095	0.476	-0.169	0.513
1.000	1	0.000	0.000	0.998	-0.037	0.000	0.000	0.000	-0.050	0.000
1.000	2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	1.000
10.000	1	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
10.000	2	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

При `alpha=0` `SparsePCA` работает также, как `PCA`.

2.4. Факторный анализ

Проведено понижение размерности с использованием факторного анализа. Диаграмма рассеяния для данных из факторного анализа представлена на рис. 16.

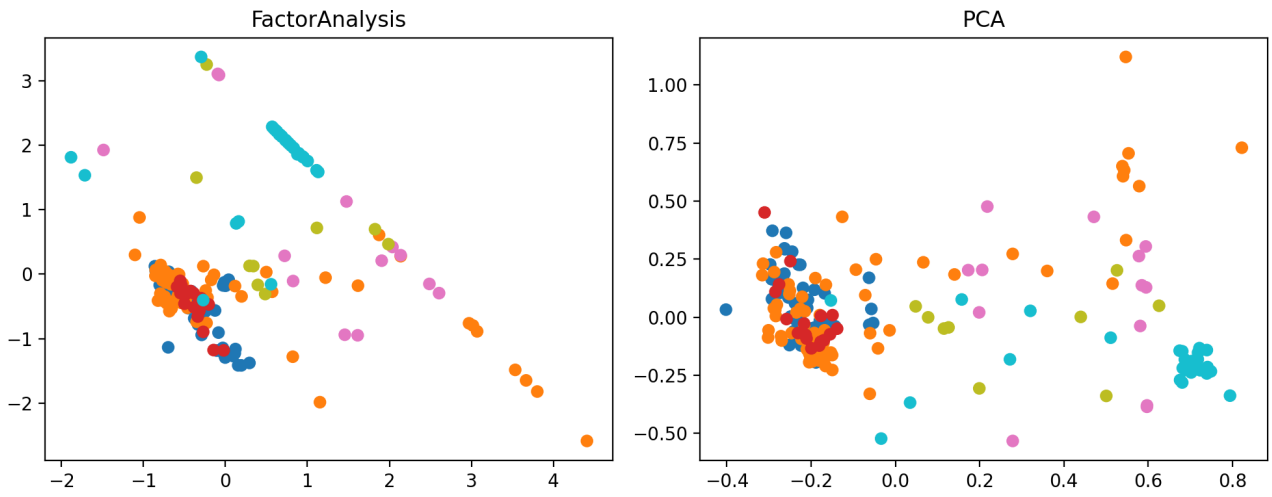

Рисунок 16 – Результаты факторного анализа

Отличия PCA и факторного анализа:

1. PCA направлен на поиск оптимальной линейной комбинации входных признаков; факторный анализ — на поиск скрытых переменных.
2. Главные компоненты ортогональны, факторы — не обязательно.

3. Выводы

Произведено знакомство с методами понижения размерности `PCA`, `KernelPCA`, `SparsePCA`, `FactorAnalysis`.

Установлено, что на данном наборе данных изменение параметров по умолчанию либо не изменяет результат, либо изменяет результат в худшую сторону. При параметрах по умолчанию различных ядра `KernelPCA` показывают приблизительно одинаковые результаты.

`KernelPCA` сводится к обычному `PCA` с линейным ядром, `SparsePCA` — с параметром регуляризации $\alpha = 0$.