МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

Кафедра информатики и систем управления

ОТЧЕТ

по лабораторной работе №2

по дисциплине

Предиктивная аналитика

РУКОВОДИТЕЛЬ:

_____ Санников А.Н.

(подпись) (фамилия, и.,о.)

СТУДЕНТ:

_____ Напылов Е.И.

(подпись) (фамилия, и.,о.)

_М22-ИВТ-1_____

(шифр группы)

Работа защищена «___» _____

С оценкой _____

Нижний Новгород 2023

Содержание

1. Постановка задачи

В данной работе требуется решить задачу кластеризации с помощью алгоритмов k-средних и k-средних++.

Был выбран датасет, содержащий анонимные данные пациентов, имеющих различные сердечные заболевания. Данные получены из медицинского центра V.A. в Лонг-Бич, Калифорния. Допустим, что всех пациентов необходимо вылечить. Индивидуально подбирать методику лечения сложно, поэтому можно попробовать кластеризовать пациентов на группы на основании симптомов / сердечных метрик. Это должно облегчить работу специалистов в этой области.

Датасет выглядит следующим образом:

	id	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope
0	1	63	1	1	145	233	1	2	150	0	2.3	3
1	2	67	1	4	160	286	0	2	108	1	1.5	2
2	3	67	1	4	120	229	0	2	129	1	2.6	2
3	4	37	1	3	130	250	0	0	187	0	3.5	3
4	5	41	0	2	130	204	0	2	172	0	1.4	1
...	...	...	...	...	...	...	...	...	...	...	...	...
298	299	45	1	1	110	264	0	0	132	0	1.2	2
299	300	68	1	4	144	193	1	0	141	0	3.4	2
300	301	57	1	4	130	131	0	0	115	1	1.2	2
301	302	57	0	2	130	236	0	2	174	0	0.0	2
302	303	38	1	3	138	175	0	0	173	0	0.0	1

303 rows × 12 columns

Всего имеется 303 пациента и 12 признаков:

1. id - айди, его уберем
2. age - возраст
3. sex - пол
4. cp - тип боли в груди
5. trestbps - артериальное давление в состоянии покоя (в мм рт. ст. при поступлении в больницу)
6. chol - сывороточный холестерин в мг/дл
7. fbs - уровень сахара в крови натощак > 120 мг/дл (1 / 0)
8. restecg - результаты ЭКГ в состоянии покоя
9. thalach - максимальный пульс
10. exang - появляется ли стенокардия под физической нагрузкой (1 / 0)
11. oldpeak - депрессия ST, вызванная физической нагрузкой по сравнению с отдыхом (что-то из ЭКГ)
12. slope - наклон сегмента ST пикового упражнения (что-то из ЭКГ)

2. K-means и K-means++

Алгоритм кластеризации K-средних (K-means) - это один из самых популярных методов кластеризации в машинном обучении. Он относится к методам без учителя, то есть не требует размеченных данных. Его цель - разделить множество объектов на заранее определенное число кластеров (обычно обозначаемое как K), так чтобы объекты внутри кластеров были максимально похожи друг на друга, а объекты между кластерами были максимально различны.

Алгоритм K-средних работает следующим образом:

1. Инициализация. Выбираются K случайных объектов в качестве центров кластеров.
2. Присвоение объектов кластерам. Каждый объект относится к ближайшему к нему центру кластера.
3. Пересчет центров кластеров. Вычисляются новые центры кластеров как среднее арифметическое всех объектов, принадлежащих данному кластеру.
4. Повторение шагов 2-3 до сходимости. Этот процесс повторяется до тех пор, пока наблюдается небольшое изменение центров кластеров.

Хотя алгоритм K-средних прост в реализации и имеет хорошую масштабируемость, он чувствителен к начальной инициализации центров кластеров. Если инициализировать центры кластеров случайным образом, то можно получить разные результаты каждый раз при запуске алгоритма на том же наборе данных.

Для улучшения начальной инициализации центров кластеров был разработан алгоритм K-средних++ (K-means++), который предлагает более умную инициализацию центров кластеров. Его алгоритм инициализации состоит из следующих шагов:

1. Выбирается первый центр кластера случайным образом из набора данных.
2. Для каждого объекта в наборе данных вычисляется расстояние до ближайшего центра кластера, уже выбранного на предыдущем шаге.
3. Следующий центр кластера выбирается случайным образом из набора данных с вероятностью, пропорциональной квадрату расстояния до ближайшего центра кластера.
4. Шаг 2-3 повторяется до тех пор, пока не будут выбраны K центров кластеров.

Таким образом, K-средних++ предлагает более интеллектуальную инициализацию центров кластеров, которая учитывает структуру данных и уменьшает вероятность того, что центры кластеров будут инициализированы в плохих местах. Это может привести к более устойчивым и качественным кластеризациям в сравнении с обычным K-средним.

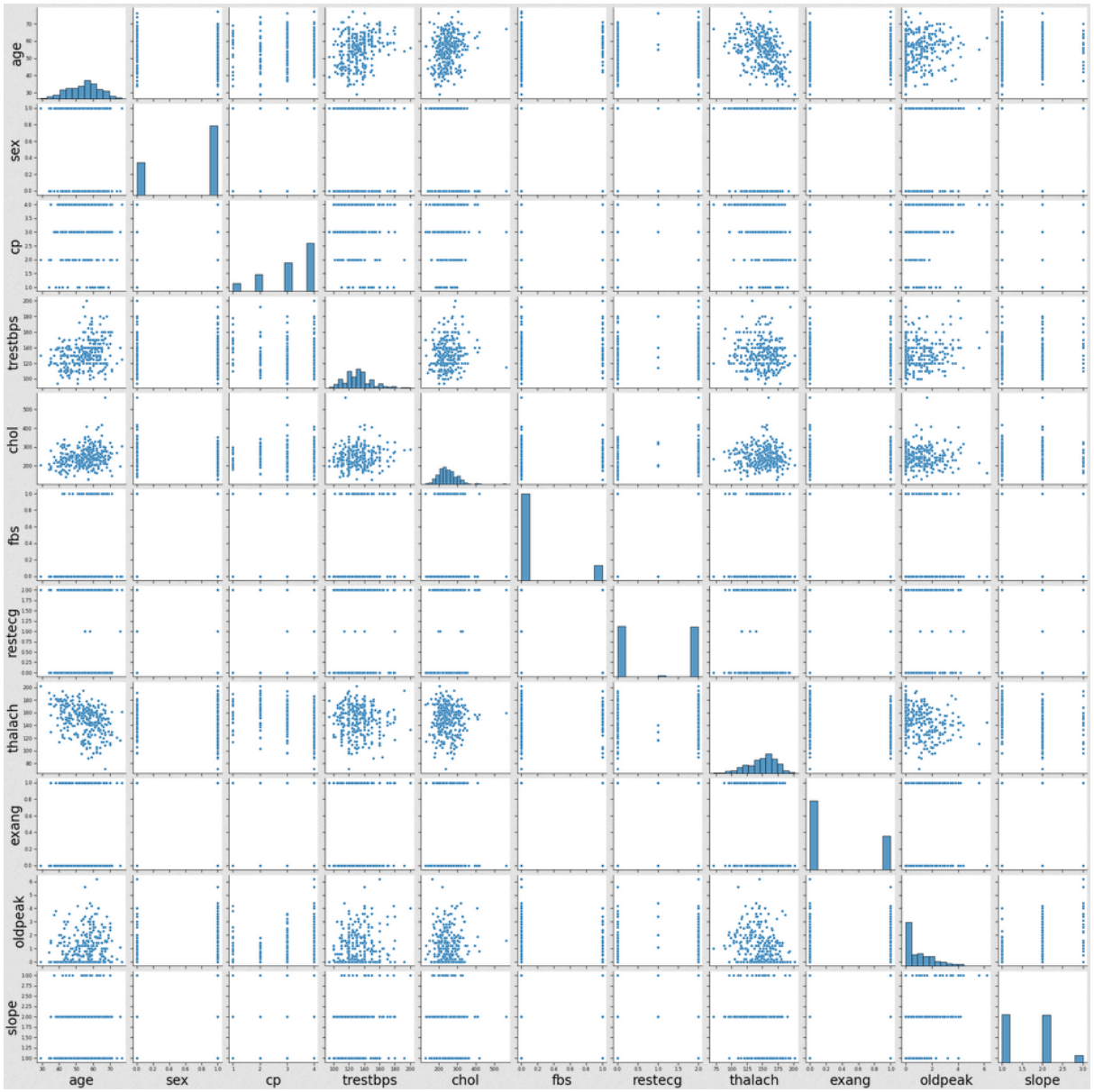
Несмотря на то, что алгоритм K-средних и K-средних++ имеют свои преимущества и широко применяются в машинном обучении, они также имеют некоторые недостатки, такие как чувствительность к начальной инициализации центров кластеров, необходимость задания числа кластеров K заранее и требование предобработки данных.

3. Данные и их обработка

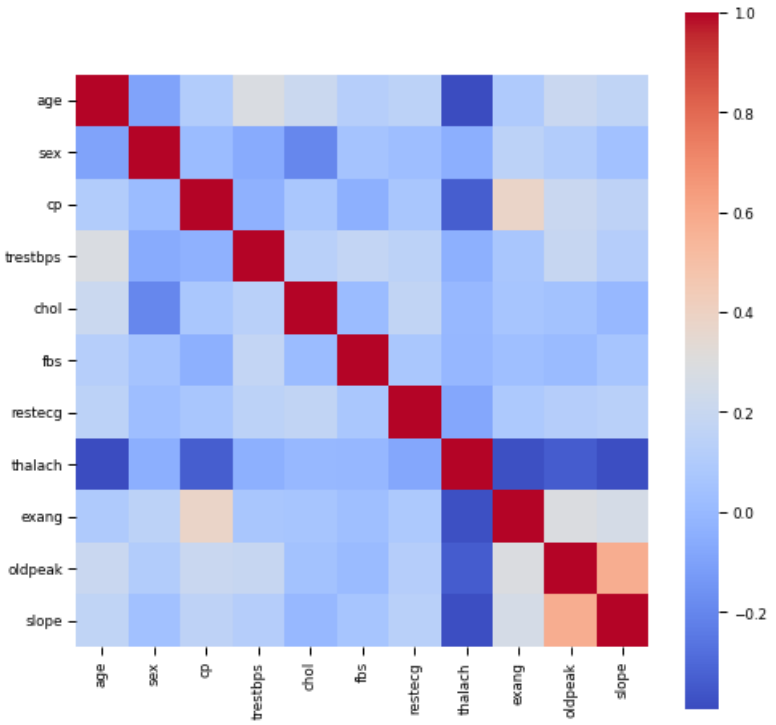
Датасет содержит 303 записи (пациента) и 12 признаков. Один из признаков - айди пациента, в дальнейшем он был удален.

	id	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope
0	1	63	1	1	145	233	1	2	150	0	2.3	3
1	2	67	1	4	160	286	0	2	108	1	1.5	2
2	3	67	1	4	120	229	0	2	129	1	2.6	2
3	4	37	1	3	130	250	0	0	187	0	3.5	3
4	5	41	0	2	130	204	0	2	172	0	1.4	1
...	...	...	...	...	...	...	...	...	...	...	...	...
298	299	45	1	1	110	264	0	0	132	0	1.2	2
299	300	68	1	4	144	193	1	0	141	0	3.4	2
300	301	57	1	4	130	131	0	0	115	1	1.2	2
301	302	57	0	2	130	236	0	2	174	0	0.0	2
302	303	38	1	3	138	175	0	0	173	0	0.0	1

Диаграммы рассеивания:



Корреляции - есть немного слабых корреляций и несколько "анти" корреляций:



Все данные являются числовыми и в целом достаточно чистые, следовательно, единственное, что необходимо сделать - нормализовать их, чтобы все числа принадлежали интервалу от 0 до 1.

```
normalized_data=(data-data.min())/(data.max()-data.min())
```

```
normalized_data
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope
0	0.708333	1.0	0.000000	0.481132	0.244292	1.0	1.0	0.603053	0.0	0.370968	1.0
1	0.791667	1.0	1.000000	0.622642	0.365297	0.0	1.0	0.282443	1.0	0.241935	0.5
2	0.791667	1.0	1.000000	0.245283	0.235160	0.0	1.0	0.442748	1.0	0.419355	0.5
3	0.166667	1.0	0.666667	0.339623	0.283105	0.0	0.0	0.885496	0.0	0.564516	1.0
4	0.250000	0.0	0.333333	0.339623	0.178082	0.0	1.0	0.770992	0.0	0.225806	0.0
...	...	...	...	...	...	...	...	...	...	...	...
298	0.333333	1.0	0.000000	0.150943	0.315068	0.0	0.0	0.465649	0.0	0.193548	0.5
299	0.812500	1.0	1.000000	0.471698	0.152968	1.0	0.0	0.534351	0.0	0.548387	0.5
300	0.583333	1.0	1.000000	0.339623	0.011416	0.0	0.0	0.335878	1.0	0.193548	0.5
301	0.583333	0.0	0.333333	0.339623	0.251142	0.0	1.0	0.786260	0.0	0.000000	0.5
302	0.187500	1.0	0.666667	0.415094	0.111872	0.0	0.0	0.778626	0.0	0.000000	0.0

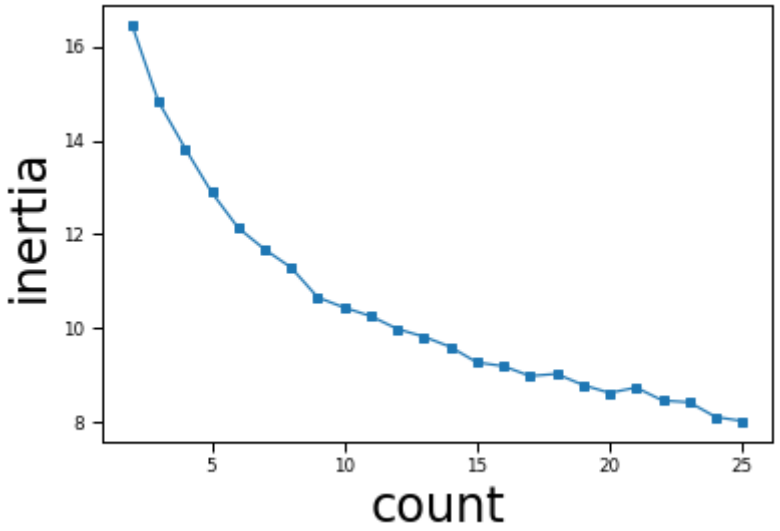
303 rows × 11 columns

4. Обучение K-means и K-means++. Результаты.

Первый вопрос, который возникает при кластеризации - сколько кластеров выбрать? Было решено перебрать все варианты в разумных пределах (от 2 до 25). Затем выбрать такое число кластеров, при котором происходит последнее сильное падение инерции - сумме квадратов расстояний от точек до центроидов кластеров, к которым они относятся.

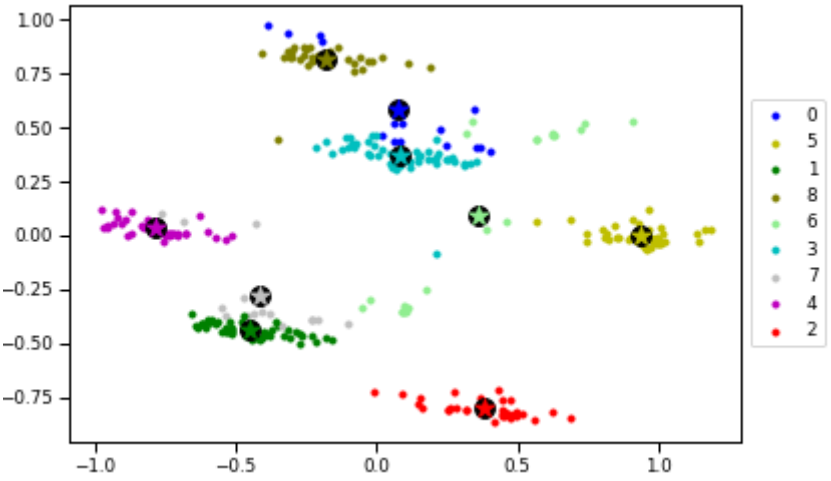
K-means

Падение инерции от числа кластеров:



На мой взгляд, можно взять 9 кластеров.

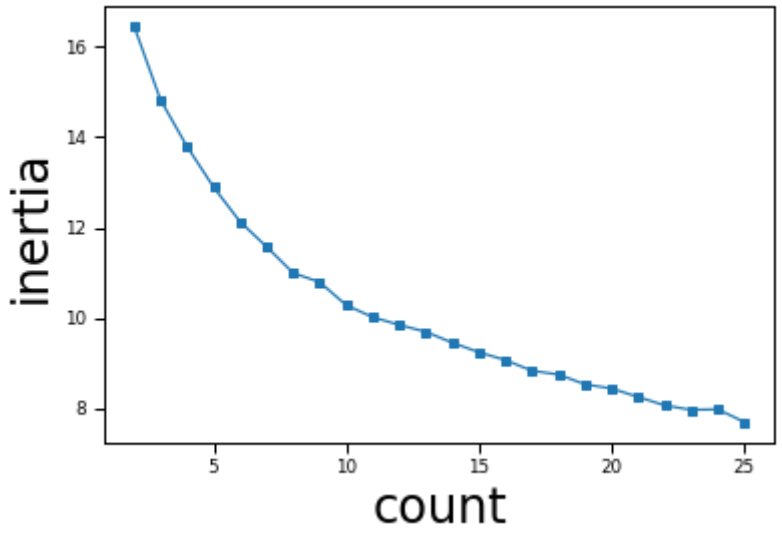
Для отображения результата кластеризации я воспользовался методом понижения размерности “Метод главных компонент” (PCA).



В целом, алгоритм K-means неплохо справился с задачей, однако получилось 2 кластера (0 и 6), которые пересекаются с остальными кластерами.

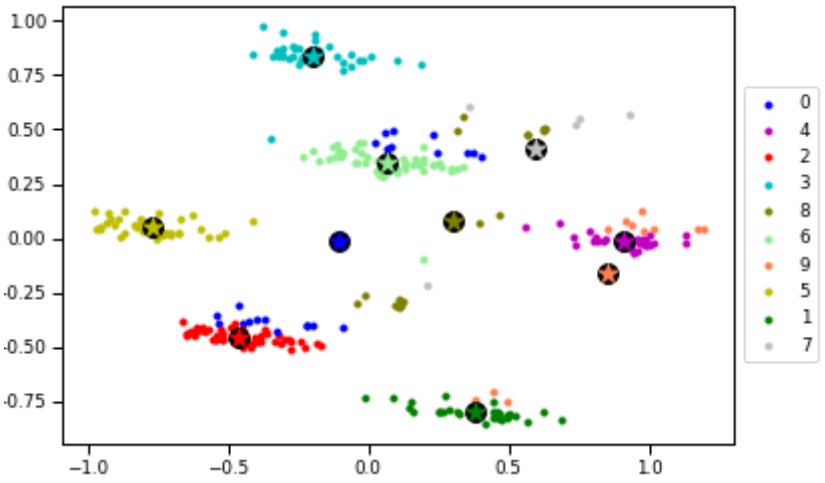
K-means++

Падение инерции от числа кластеров:



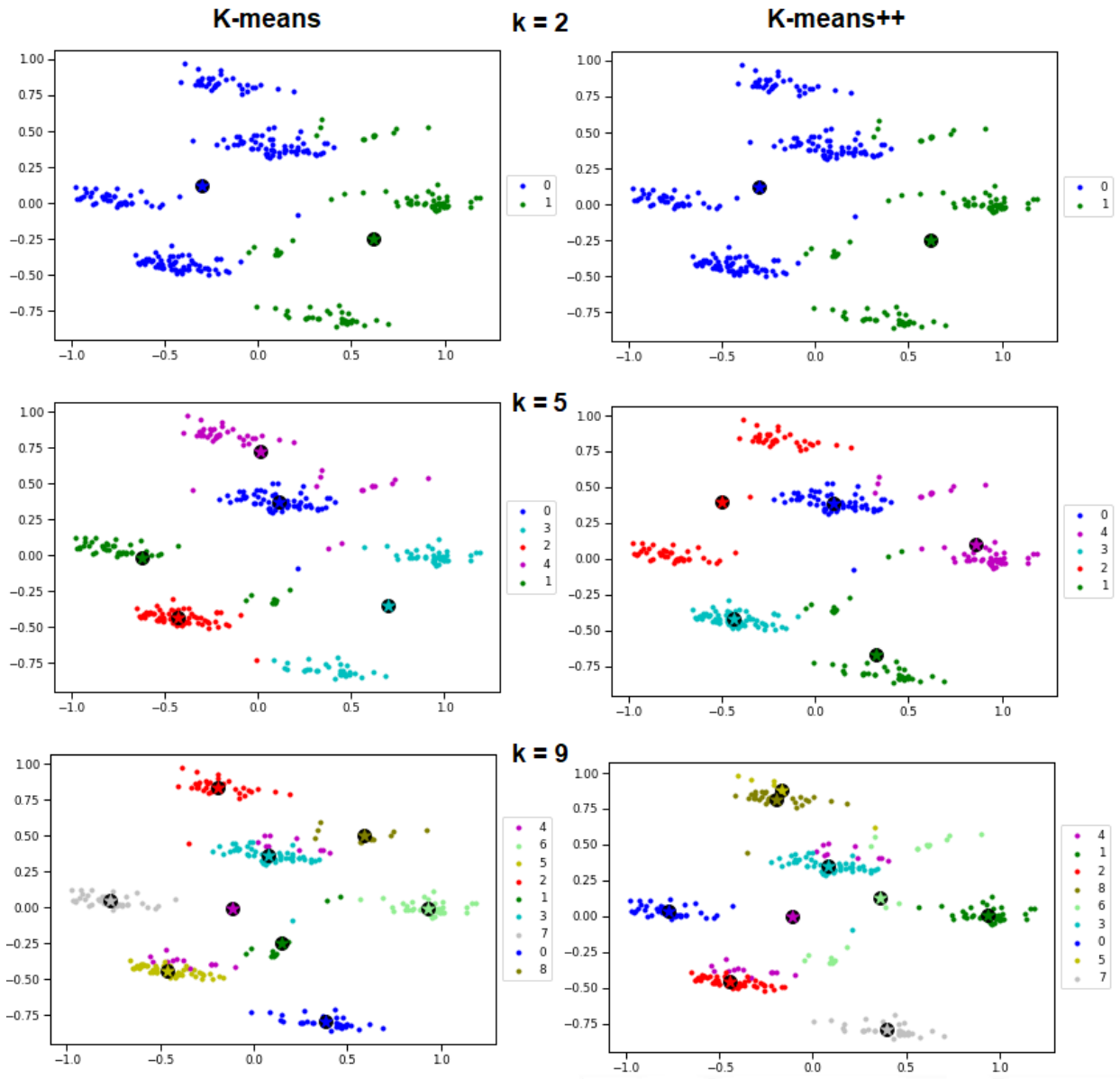
Здесь оптимальным выбором можно считать 10 кластеров.

PCA:



Этот алгоритм так же хорошо справился с задачей кластеризации.

Теперь сравним алгоритмы при одинаковом количестве кластеров.



K-means++ немного выигрывает у K-means, но классический алгоритм тоже неплохо работает.

5. Выводы

В результате работы была решена задача кластеризации пациентов на группы. Решение может стать полезным для медиков при лечении большого количества пациентов. Это позволяет назначать план лечения по группам вместо индивидуального подхода, что позволяет сократить временные расходы. Оба алгоритма - K-means и K-means++ показали хорошие результаты. K-means++ немного выигрывает по качеству у классического алгоритма за счет умной инициализации первых кластеров.