МИНОБРНАУКИ РОССИИ

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ

ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)

Кафедра МОЭВМ

ОТЧЕТ

по лабораторной работе №6

по дисциплине «Машинное обучение»

Тема: Кластеризация (DBSCAN, OPTICS)

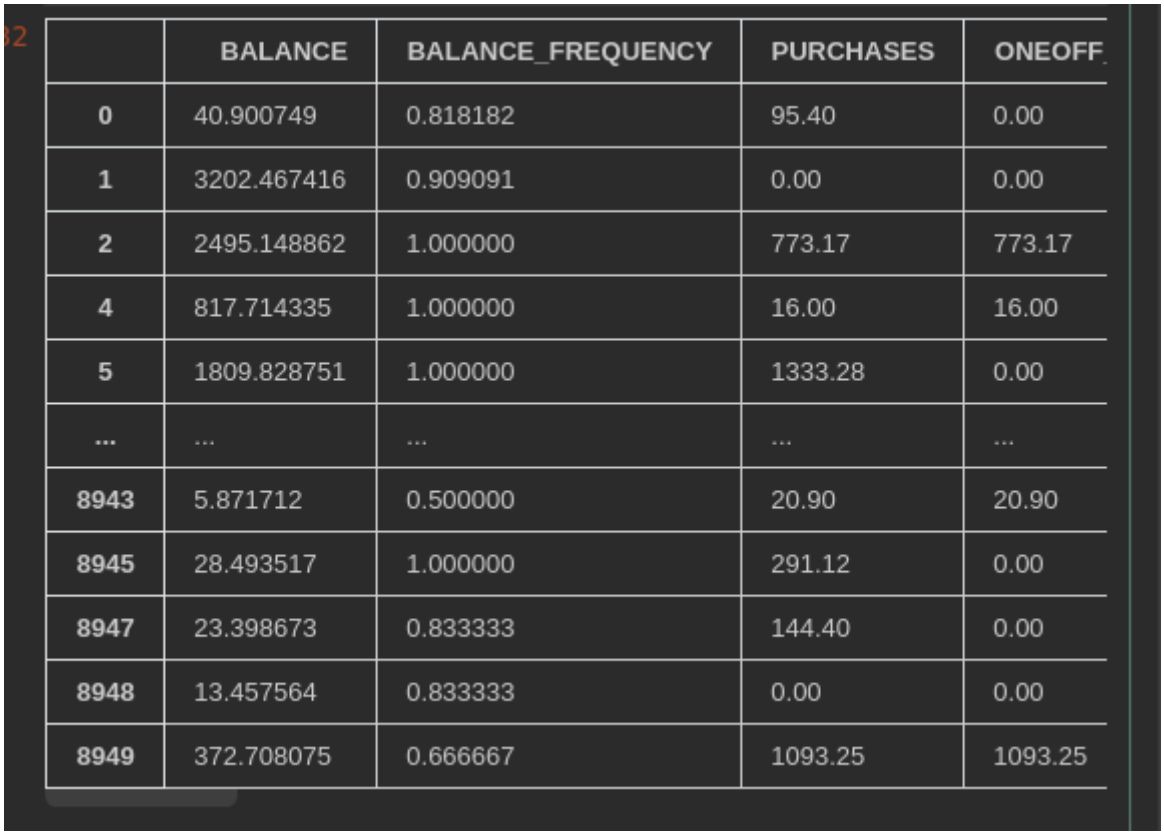
Студент гр. 6307 _____ Новиков Б.М.

Преподаватель _____ Жангиров Т.Р.

2020

Загрузка данных

1. загрузить датасет с сайта, потом загрузить его в датафрейм.



	BALANCE	BALANCE_FREQUENCY	PURCHASES	ONEOFF
0	40.900749	0.818182	95.40	0.00
1	3202.467416	0.909091	0.00	0.00
2	2495.148862	1.000000	773.17	773.17
4	817.714335	1.000000	16.00	16.00
5	1809.828751	1.000000	1333.28	0.00
...	...	...	...	...
8943	5.871712	0.500000	20.90	20.90
8945	28.493517	1.000000	291.12	0.00
8947	23.398673	0.833333	144.40	0.00
8948	13.457564	0.833333	0.00	0.00
8949	372.708075	0.666667	1093.25	1093.25

DBSCAN

1. Стандартизировать данные.

```
min_max_scaler = preprocessing.StandardScaler()
scaled_data = min_max_scaler.fit_transform(data)
```

```
array([[-0.74462486, -0.37004679, -0.42918384, ..., -0.30550763,

        -0.53772694,  0.35518066],
       [ 0.76415211,  0.06767893, -0.47320819, ...,  0.08768873,
         0.21238001,  0.35518066],
       [ 0.42660239,  0.50540465, -0.11641251, ..., -0.09990611,
        -0.53772694,  0.35518066],
       ...,
       [-0.75297728, -0.29709491, -0.40657175, ..., -0.32957217,
         0.30614422, -4.22180042],
       [-0.75772142, -0.29709491, -0.47320819, ..., -0.34081076,
         0.30614422, -4.22180042],
       [-0.58627829, -1.09958965,  0.03129519, ..., -0.32709767,
        -0.53772694, -4.22180042]])
```

2. Провести кластеризацию методом DBSCAN. Вывести метки кластеров, количество кластеров, процент некластеризированных наблюдений. Опишите все параметры, которые принимает DBSCAN

```
clustering = DBSCAN().fit(scaled_data)
```

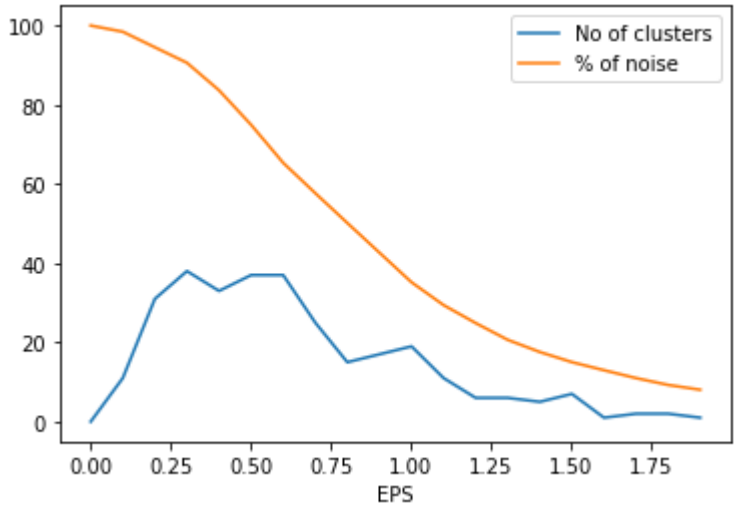
```
Метки: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19,
20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, -1}
```

```
Количество меток: 36
```

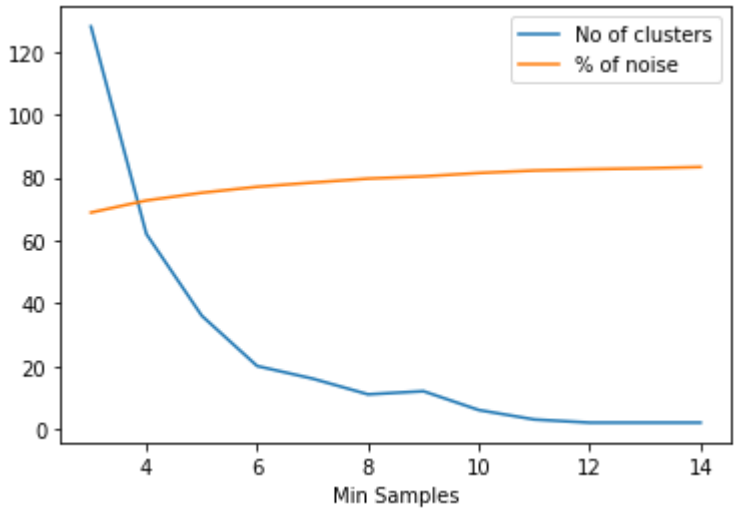
```
Процент некластер. Наблюд.: 0.7512737378415933
```

параметр	описание
eps	Максимальное расстояние между двумя сэмплами, означающее что точка в окрестности у другой
min_samples	Количество сэмплов в окрестности, чтобы считать точку основной
metric	Метрика, используемая для вычисления расстояния
metric_params	Дополнительный параметр для метрики
algotithm	Алгоритм, используемый в модуле NearestNeighbors
leaf_size	Размер листа, параметр, который передается в BallTree или cKDTree
p	Степень метрики Минковского
n_jobs	Количество потоков для исполнения

3. График количества кластеров и процента некласт. наблюдений в зависимости от максимальной дистанции.



4. График количества кластеров и процента некласт. наблюдений в зависимости от минимального количества точек, образующих кластер.

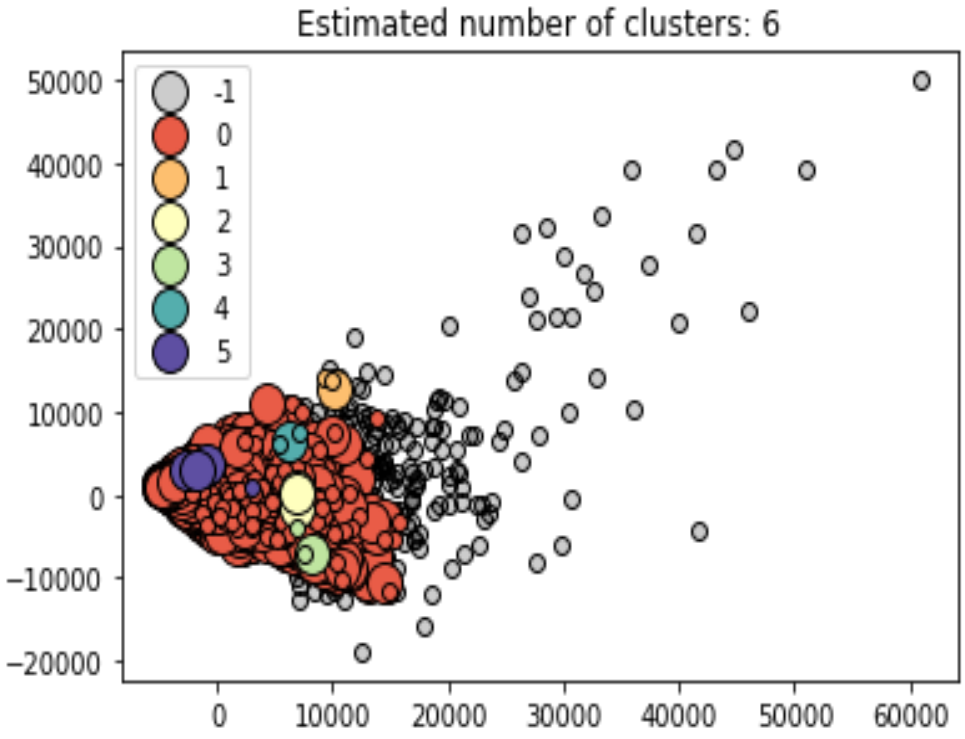


5. Значения параметров, при которых количество кластеров 5-7, процент некласт. 12%

```
clustering = DBSCAN(eps=2, min_samples=3).fit(scaled_data)

Количество: 6
Процент: 6.287633163501621
```

6. Понизить размерность данных до 2. Визуализировать результаты кластеризации.



OPTICS

1. Описать параметры и атрибуты OPTICS

Параметр	Описание
min_samples	Количество семплов в окрестности, чтобы точка считалась основной точкой
max_eps	Максимальное расстояние между двумя семплами, чтобы они считались в окрестности у друг друга
metric	Метрика для вычисления расстояния
p	Параметр для метрики Минковского
metric_params	Дополнительные параметры для метрик
cluster_method	Метод извлечения кластеров
eps	Тоже самое, что и max_eps, но для метода извлечения кластеров DBSCAN
xi	Определяет минимальную крутизну на графике достижения
predecessor_correction	Исправляет кластеры с учетом вычисленного предшественниками
min_cluster_size	Минимальное число сэмплов в кластере
algorithm	Алгоритм для вычисления ближайших соседей
leaf_size	Размер листа, параметр, который передается в BallTree или cKDTree
n_jobs	Количество потоков исполнения

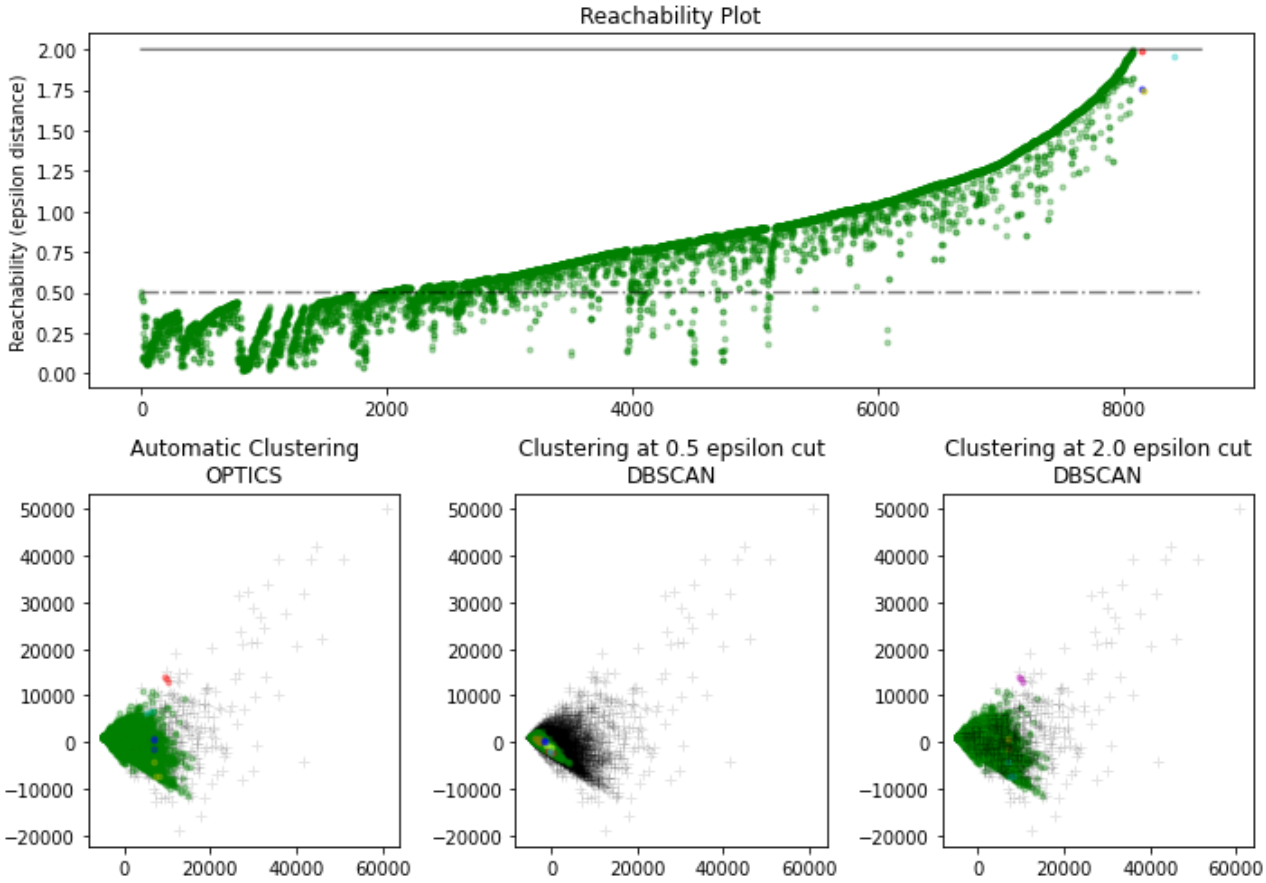
2. Найти параметры OPTICS, чтобы были близки к тем, что получены с DBSCAN. Описать различие методов.

```
clustering = OPTICS(max_eps=2, min_samples=3,
cluster_method='dbscan').fit(scaled_data)

Метки: {0, 1, 2, 3, 4, 5, -1}
Количество кластеров: 6
Процент некласт.: 6.310792033348773
```

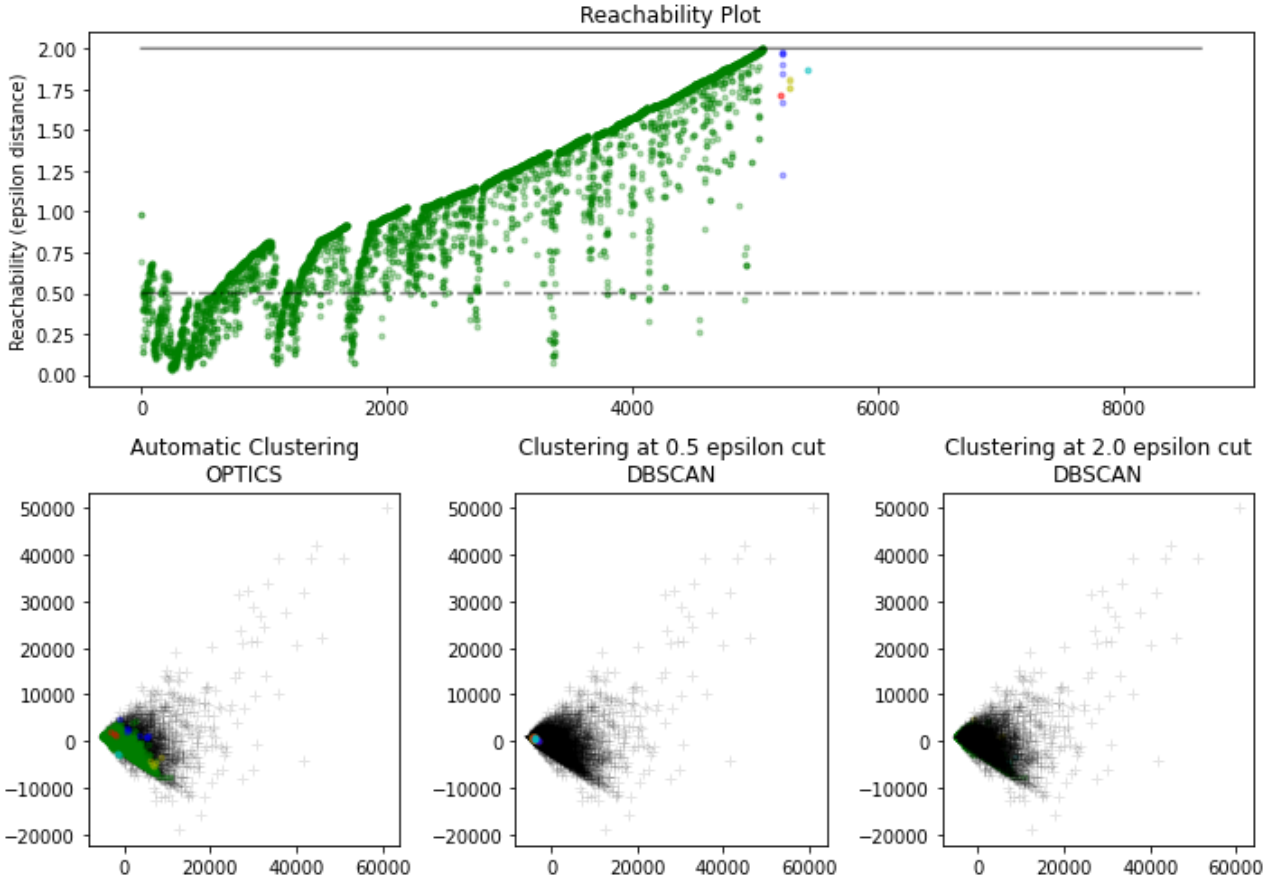
В отличии от DBSCAN, сохраняет иерархию кластеров для разных радиусов внутри окрестностей. Больше подходит для использования на больших датасетах.

3. Визуализировать полученный результат, а также построить график достижимости



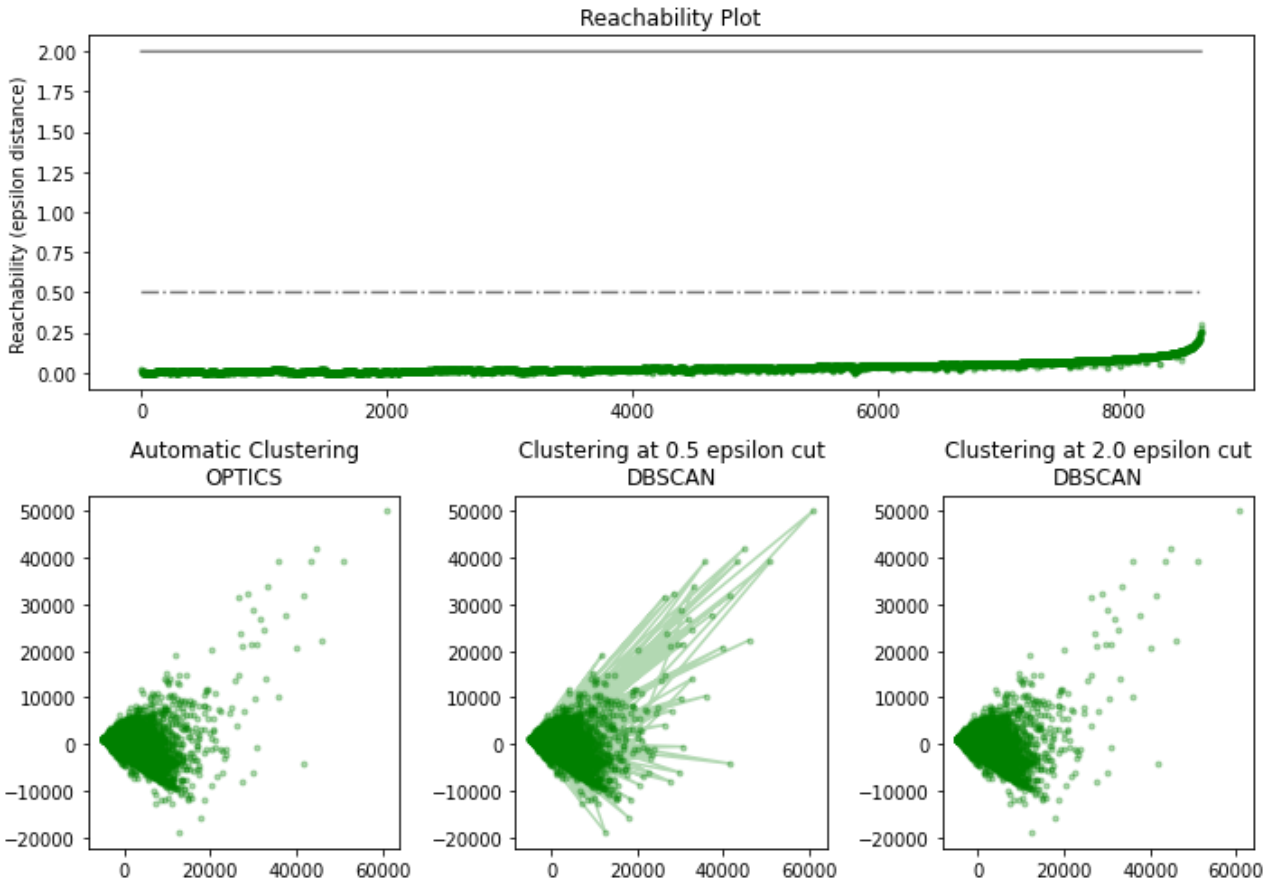
4. Исследовать работу метода с использованием различных метрик

`cityblock – манхеттанское расстояние`



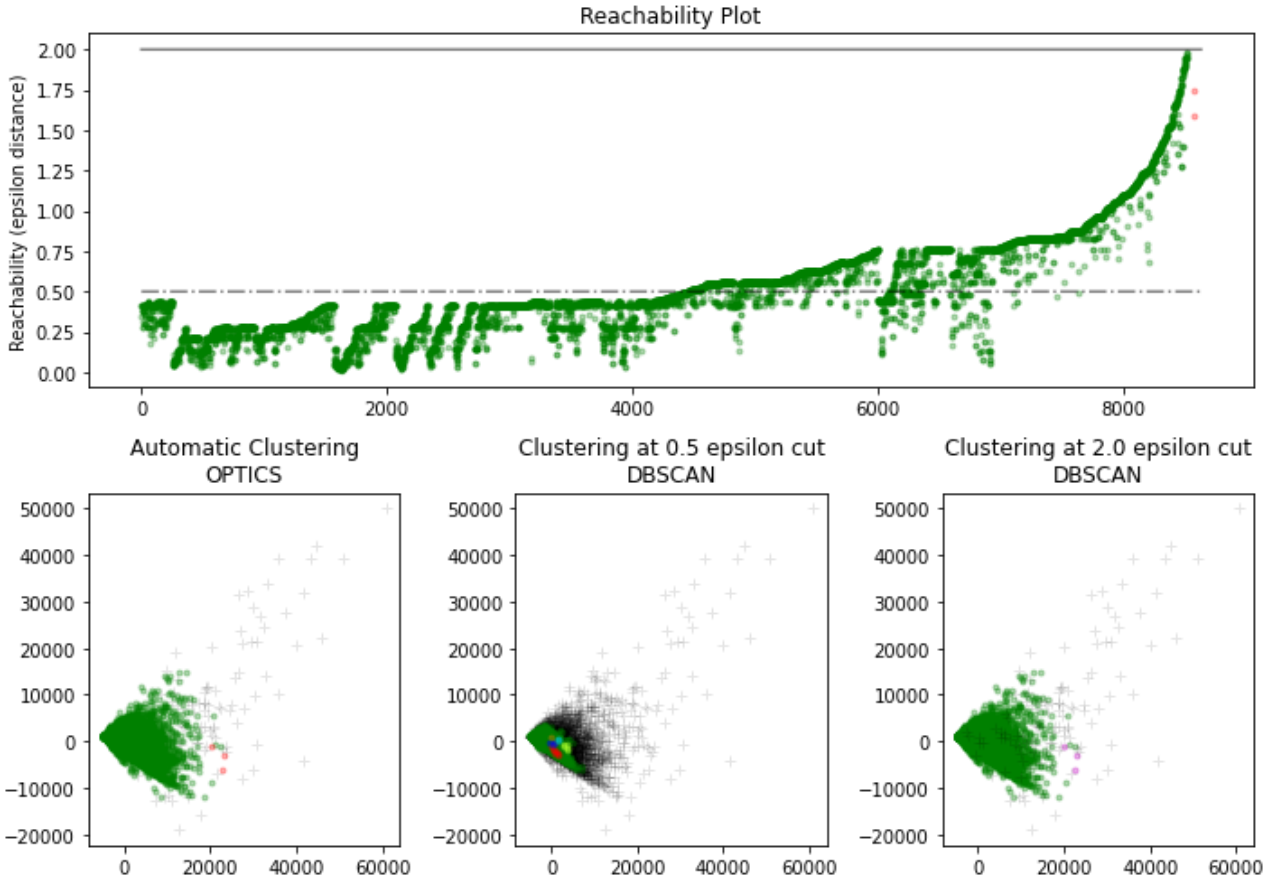

```
Метки: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19,
20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39,
40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, -1}
Кол-во меток: 55
Процент некласт: 39.49745252431681
```

cosine - косинусовое



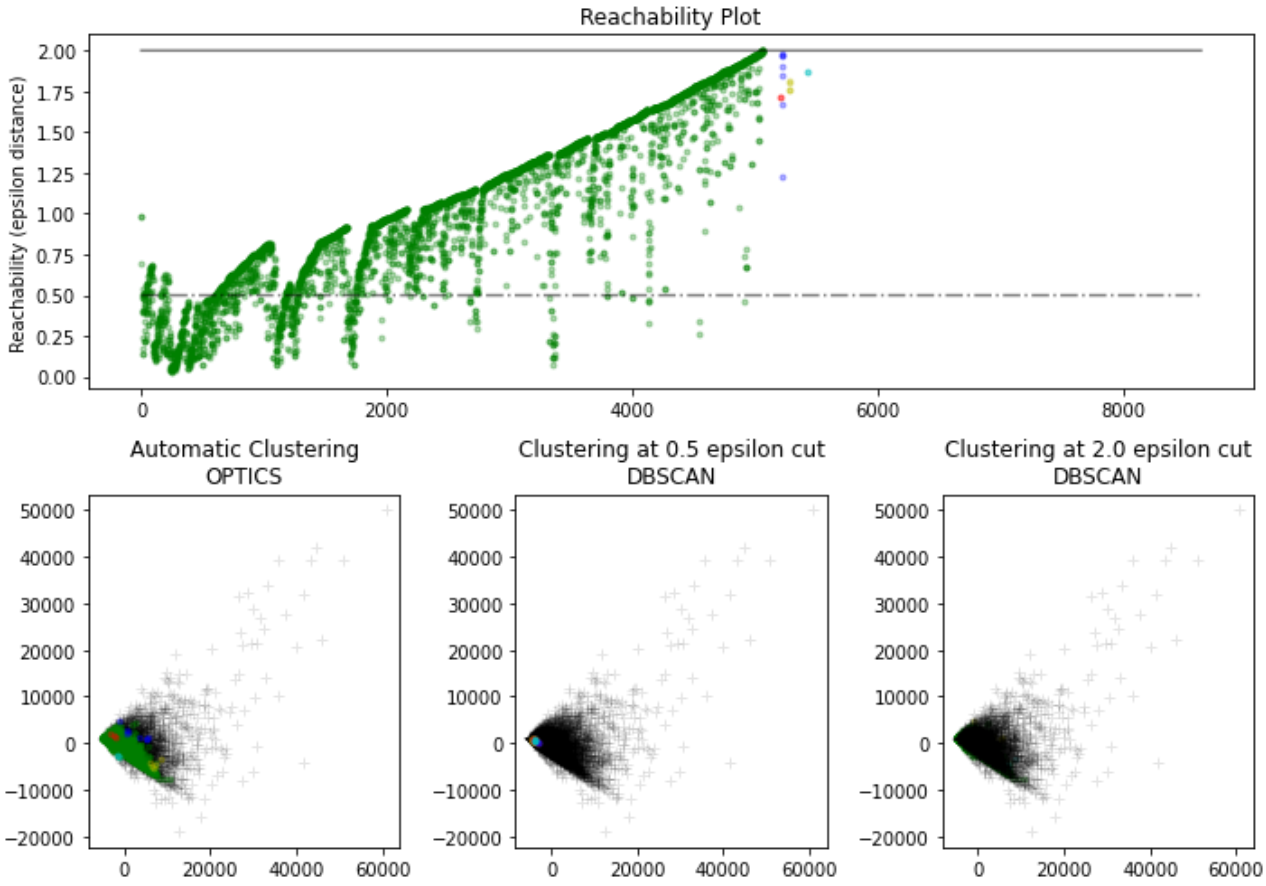
Метки: {0}
Кол-во меток: 0
Процент некласт: 0.0

chebyshev - чебышева



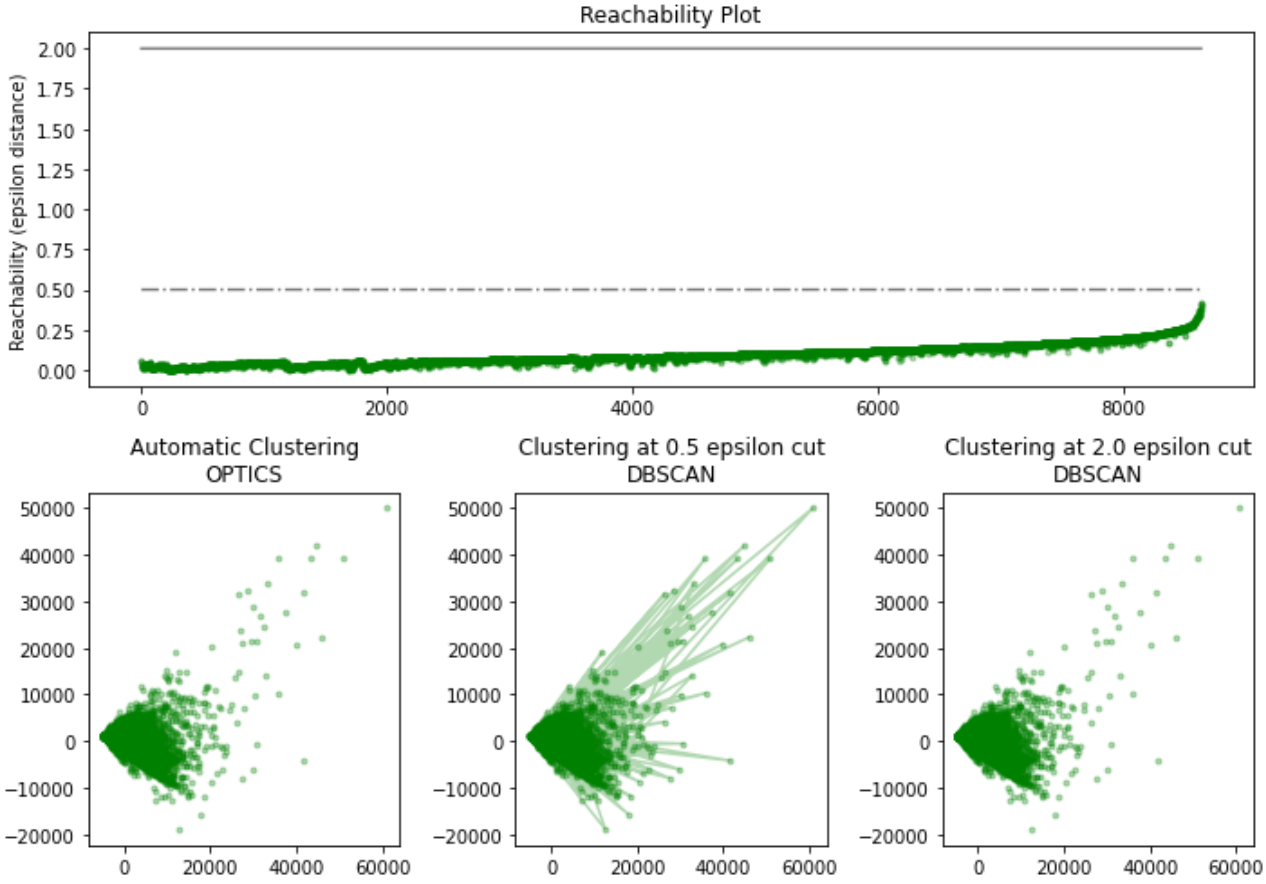
Метки: {0, 1, -1}
Кол-во меток: 2
Процент некласт: 1.331635016211209

l1



Метки: {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, -1}
Кол-во меток: 55
Процент некласт: 39.49745252431681

braycurtis



Метки: {0}
Кол-во меток: 0
Процент некласт: 0.0