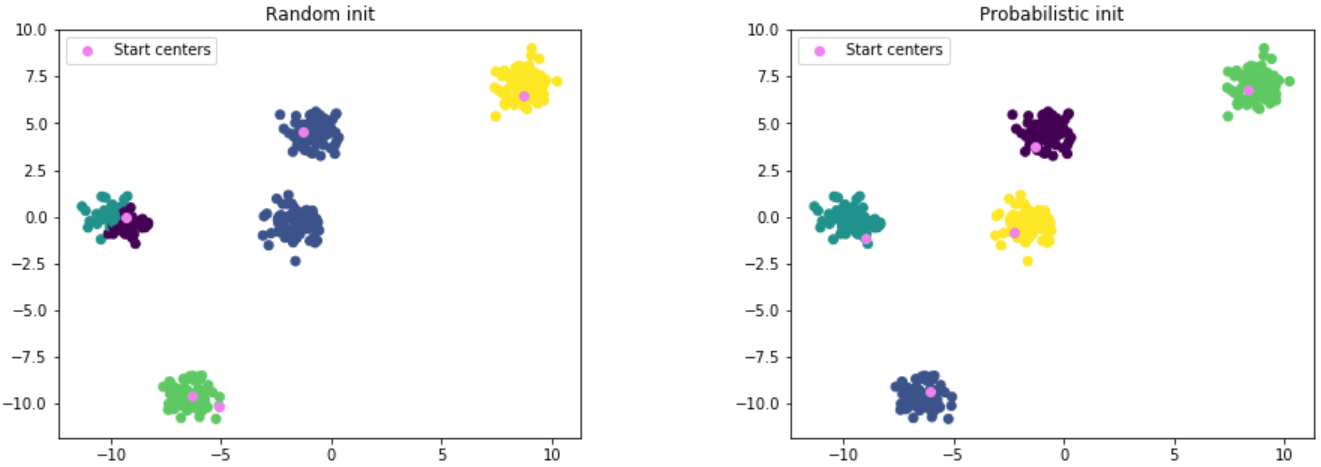
# Отчёт по заданию "Кластеризация"

In [42]:
```
1
```




Реализация kmeans++, основанная на выборе начальных центров с учём расстояния до уже выбранных, даёт относительно малую вероятность выбора близких центров.

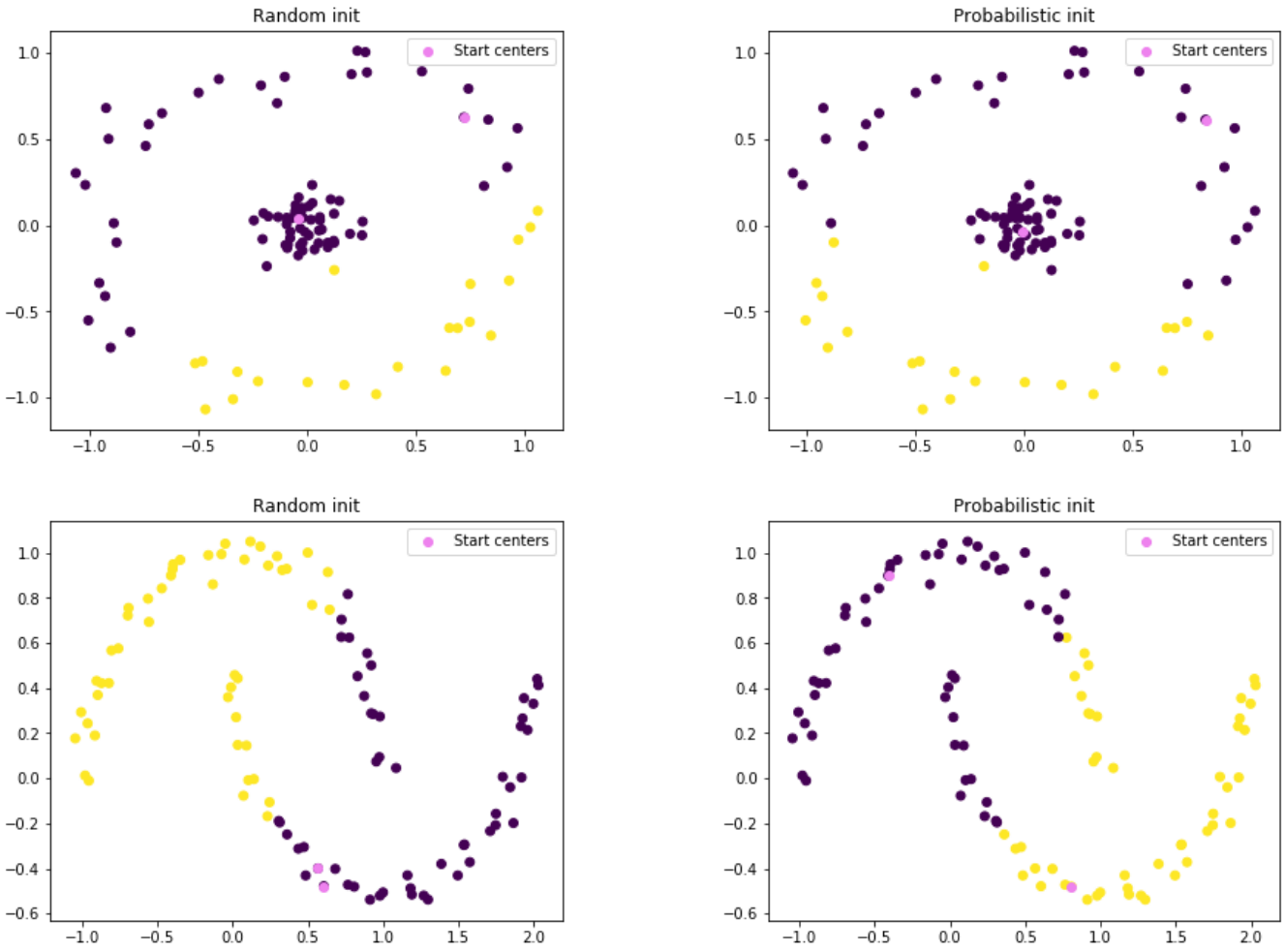
Алгоритм для kmeans++:

1a. Выбираем первый центр $c_1$, случайно из $\mathcal{X}$.

1b. Выбираем новый центр $c_i \in \mathcal{X}$ с вероятностью $\frac{D(x)^2}{\sum_{x \in \mathcal{X}} D(x)^2}$

1c. Повторяем шаг 1b. до тех пор пока не наберём $k$ центров.

In [0]:
```
1
```




Как видно это не самая лучшая задача для Kmeans. Этот алоритм хорошо определяет "шарообразные" кластеры с большой плотностью.

Kmeans также используется в кластеризации документов, выявлении криминогенных зон, сегментации клиентов, выявлении страховых мошенничеств, анализа данных общественного транспорта, кластеризации ИТ-оповещений.

Реализовано два метода:

1. Метод Локтя, где:

$$WSS = \sum_{C_k \in C} \sum_{x_i \in C_k} \|x_i - \mu_k\|^2$$

2. Метод силлуэтов, где:

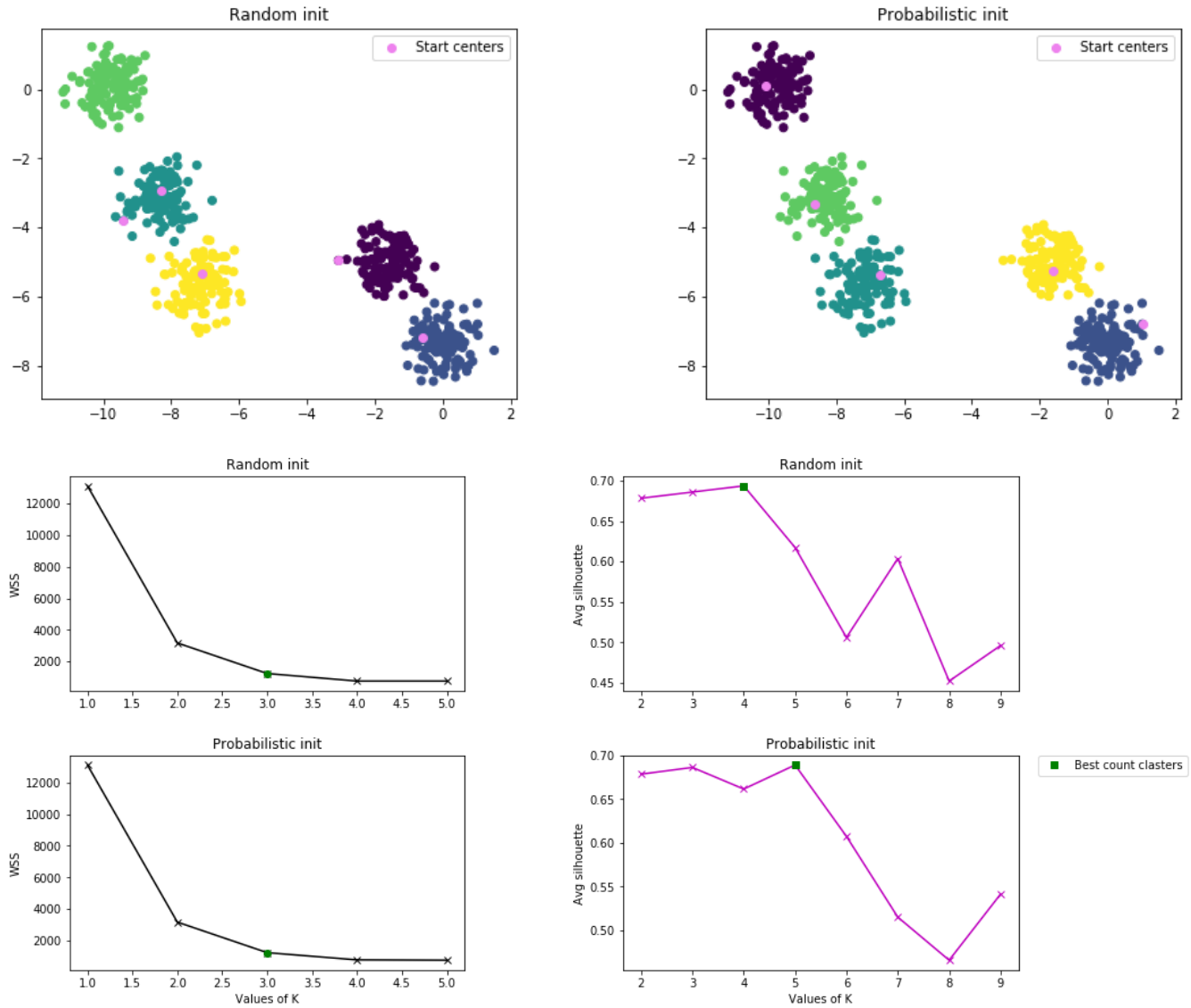
$$a(i) = \frac{1}{|C(i)|-1} \sum_{C(i), i \neq j} d(i,j)$$

$$b(i) = \min_{i \neq j} \left( \frac{1}{|C(j)|} \sum_{j \in C(j)} d(i,j) \right)$$

$$s(i) = \frac{b(i)-a(i)}{\max(a(i),b(i))}$$

$$\text{Avg silhouette} = \frac{\sum_{i=1}^{n} s(i)}{n}$$

In [45]:
```
1
```




Исходя из эксперимента, при больших K "elwob_method" теряет свою предсказательную способность, в отличии от "silhouette_method"

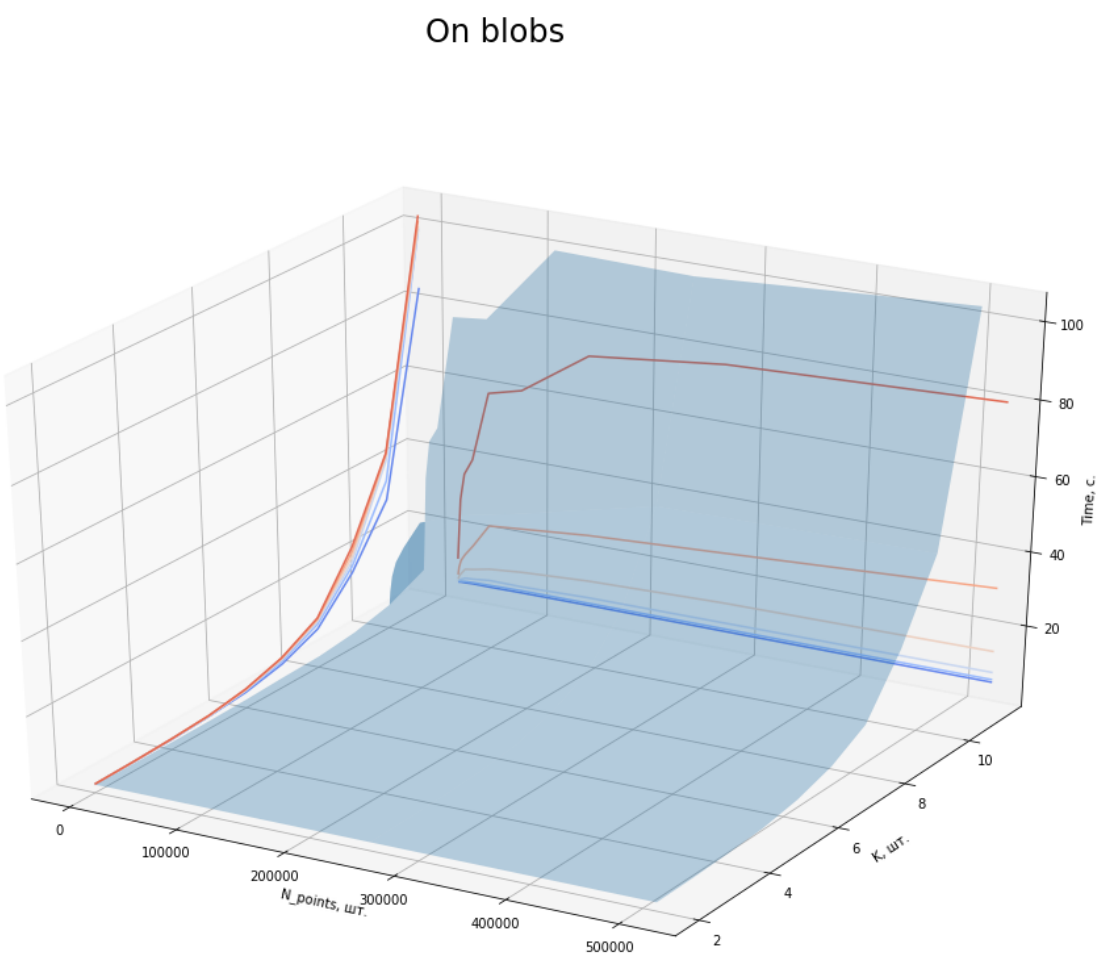
Время выполнения алгоритма Ллойда является $O(nkdi)$, где:

- n - число d-мерных векторов (подлежащих кластеризации)
- k - число кластеров
- i - число итераций, необходимых до сходимости.
- d - размерность пространства

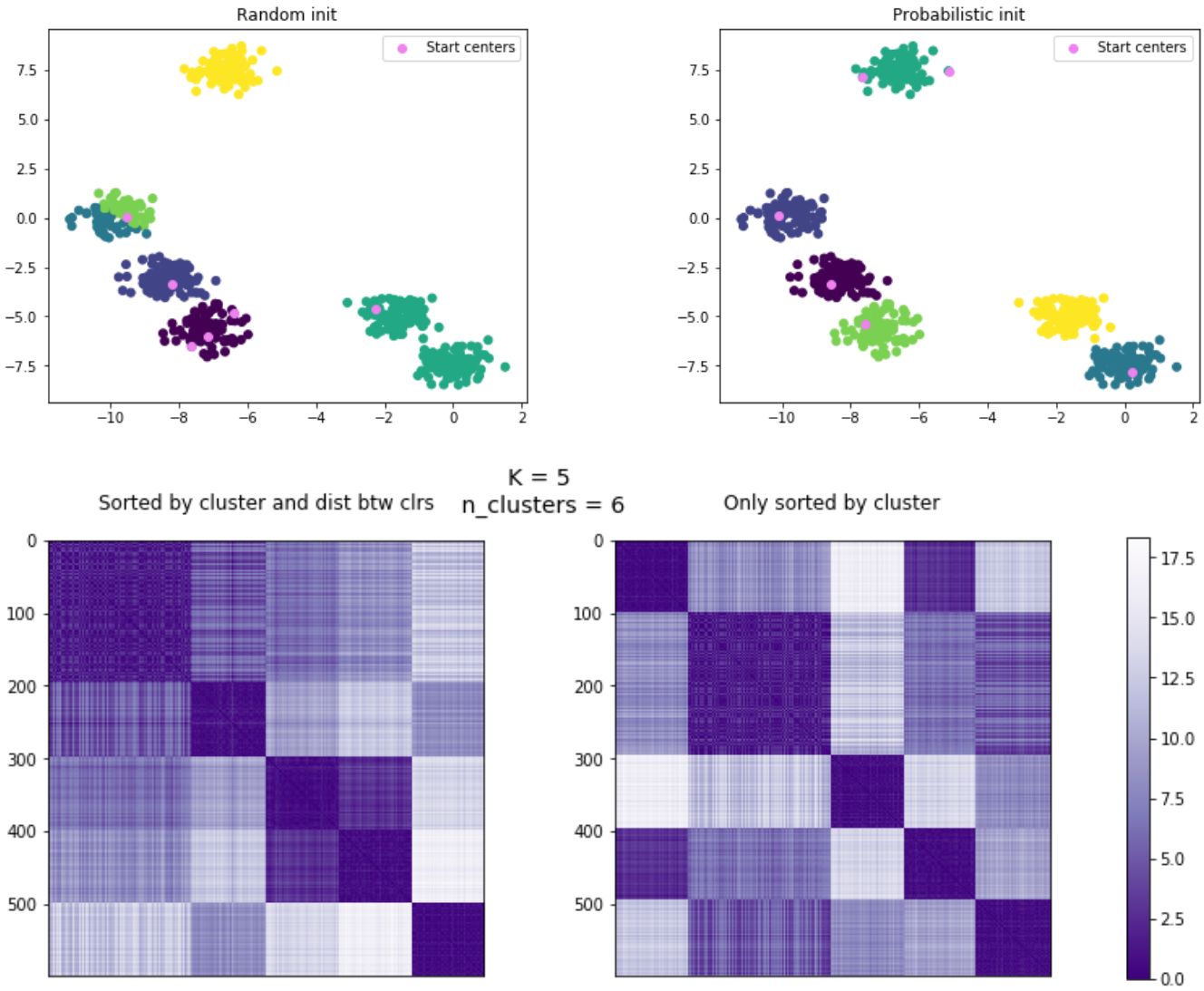
На данных, которые имеют кластеризованную структуру, число итераций до сходимости часто невелико, и результаты только немного улучшаются после первого десятка итераций.

В худшем случае, алгоритма Ллойда требует $i = 2^{\Omega(\sqrt{n})}$ итераций

In [0]:
```
1
```


On blobs
0
100000
200000
300000
400000
500000
N_points, шт.
2
4
6
8
10
K, шт.
20
40
60
80
100
Time, c.

In [0]:
```
1
```

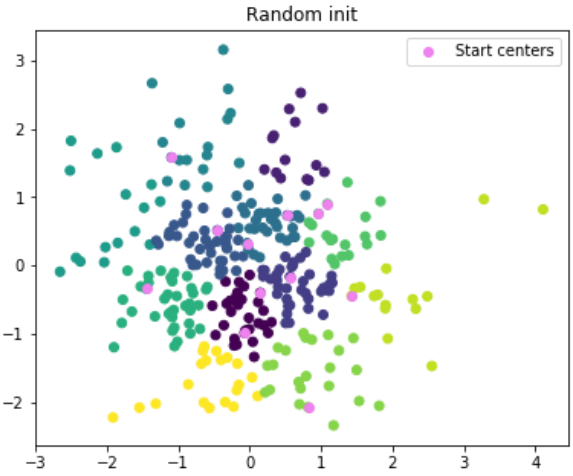


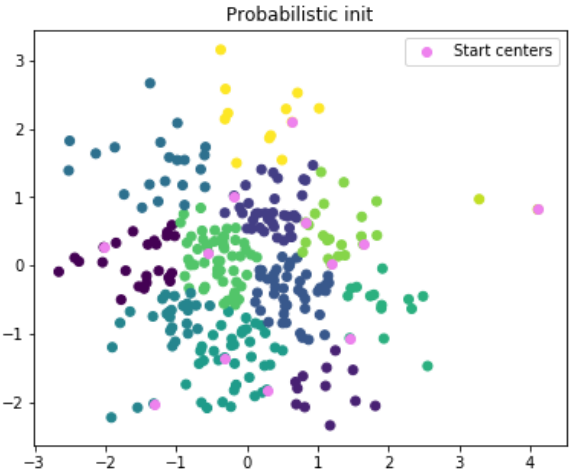

Как видно упорядочивание кластеров по расстоянию от наибольшего кластера даёт дополнительную информацию об расположении кластеров. Например, два кластера достаточно близкие к друг другу образуют в совокупности больший квадрат.

---

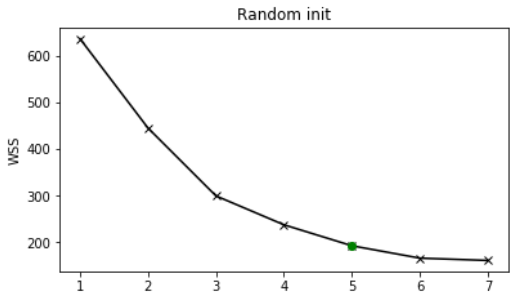
`Аналогично для гауссиан(K = 12)`

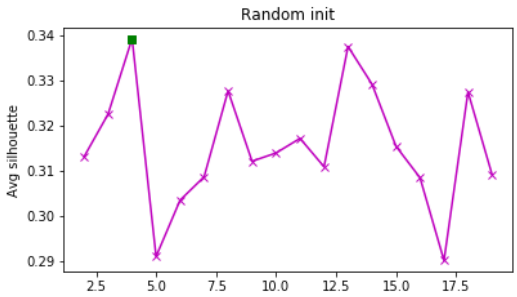
In [62]:
```
1
```

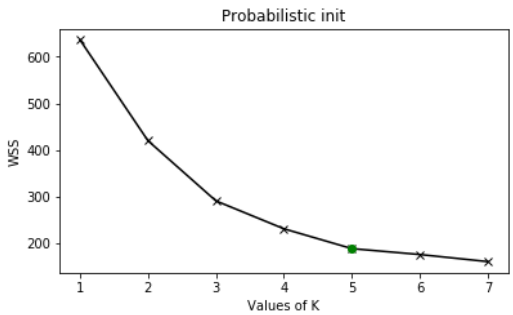

Random init
Start centers

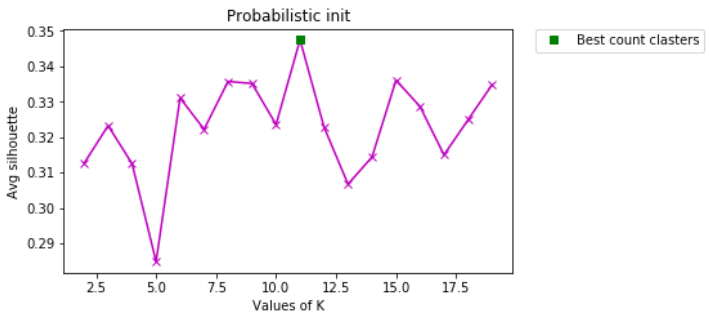

Probabilistic init
Start centers

In [19]:
```
1
```

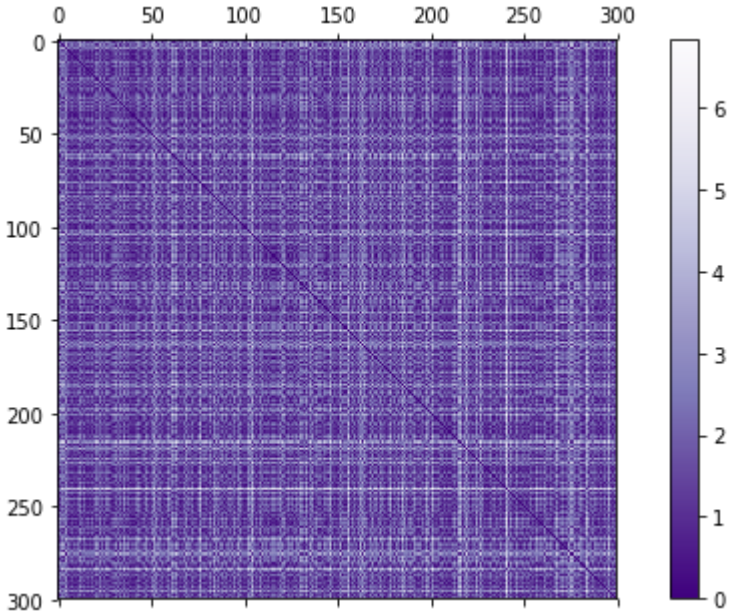

Random init
WSS


Random init
Avg silhouette

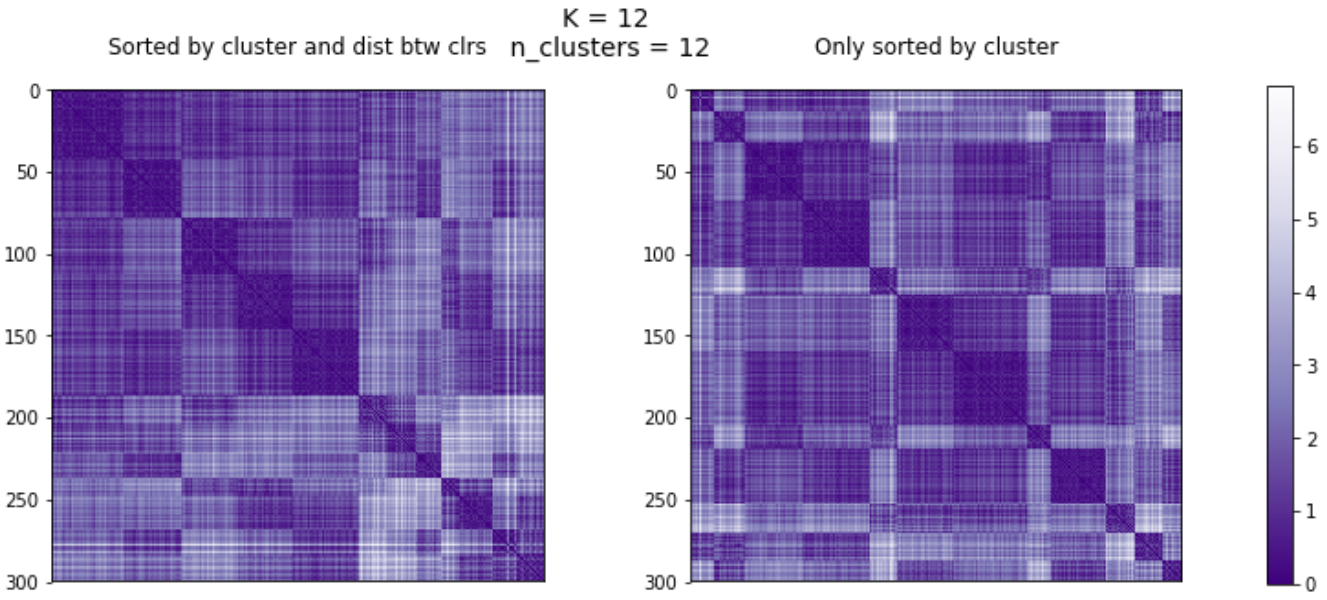

Probabilistic init
WSS
Values of K


Probabilistic init
Avg silhouette
Values of K
Best count clasters

In [20]:
```
1
```


0
50
100
150
200
250
300
0
50
100
150
200
250
300
0
1
2
3
4
5
6

In [32]:
```
1
```


K = 12
Sorted by cluster and dist btw clrs
n_clusters = 12
Only sorted by cluster
0
50
100
150
200
250
300
0
1
2
3
4
5
6