Московский государственный технический
университет им. Н. Э. Баумана
Факультет «Информатика и системы управления»

Кафедра «Системы обработки информации и управления»

Курс «Технологии машинного обучения»

Рубежный контроль №2

Группа: РТ5-61

Студент: Савушкин Д. А.

Преподаватель: Гапанюк Ю.Е.

Москва, 2020 г.

Тема: Задача 2. Кластеризация данных.

Задание:

Кластеризуйте данные с помощью двух алгоритмов кластеризации (MeanShift, иерархическая кластеризация).

Сравните качество кластеризации с помощью следующих метрик качества кластеризации (если это возможно для Вашего набора данных):

1. Adjusted Rand index
2. Adjusted Mutual Information
3. Homogeneity, completeness, V-measure
4. Коэффициент силуэта

Сделайте выводы о том, какой алгоритм осуществляет более качественную кластеризацию на Вашем наборе данных.

▾ **Дятленко Елена Александровна** Группа ИУ5-62Б

```
[38] import numpy as np
     import pandas as pd
     import seaborn as sns
     import matplotlib.pyplot as plt
     from sklearn.cluster import MeanShift, AgglomerativeClustering
     from itertools import cycle, islice
     import warnings
     warnings.simplefilter(action='ignore', category=FutureWarning)
     from sklearn.metrics import silhouette_score
     from sklearn.cluster import KMeans
```

```
[39] df = pd.read_csv('Admission_Predict_Ver1.1.csv')
     df.shape
```

```
(500, 9)
```

```
[40] df.head()
```

	Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
0	1	337	118	4	4.5	4.5	9.65	1	0.92
1	2	324	107	4	4.0	4.5	8.87	1	0.76
2	3	316	104	3	3.0	3.5	8.00	1	0.72
3	4	322	110	3	3.5	2.5	8.67	1	0.80
4	5	314	103	2	2.0	3.0	8.21	0	0.65

```
[41] ss_list = []
     for k in range(2, 20):
         kmeans_result = KMeans(n_clusters=k, random_state=1).fit_predict(df)
         ss_list.append([k,silhouette_score(df,kmeans_result)])
     ss_list
```

```
[[2, 0.6127654855568136],
 [3, 0.563379470709112],
 [4, 0.5311456169984072],
 [5, 0.5045717462773318],
 [6, 0.4809261927433612],
 [7, 0.4610748969934055],
 [8, 0.440502082290444],
 [9, 0.42512815857864755],
 [10, 0.40930520069430393],
 [11, 0.39114816574227623],
 [12, 0.3801581075387568],
 [13, 0.368304528940199],
 [14, 0.362077459778748],
 [15, 0.3502702301710579],
 [16, 0.3364489264219557],
 [17, 0.3309511702406532],
 [18, 0.3285303074846944],
 [19, 0.3433452349664484]]
```

Выберем кол-во кластеров 2

```
[42] #MeanShift и Иерархическая кластеризация
     MeanShift_2 = MeanShift()
     MeanShift_2_result = MeanShift_2.fit_predict(df)
```

```
AgglomerativeClustering_2 = AgglomerativeClustering(n_clusters=2)
AgglomerativeClustering_2_result = AgglomerativeClustering_2.fit_predict(df)
```

```
[44] #Сравнение моделей
     silhouette_score(df,MeanShift_2_result)
```

```
0.5100475697000604
```

```
[45] silhouette_score(df,AgglomerativeClustering_2_result)
```

```
0.5264163472424273
```

Иерархическая кластеризацияосуществляет более качественную кластеризацию данных