Датасет – Спрос на бронирование отелей:

https://www.kaggle.com/datasets/jessemostipak/hotel-booking-demand

Метод К-средних

Использование GridSearchCV для подбора наилучшего параметра по количеству кластеров. Оценка – силуэт кластера.

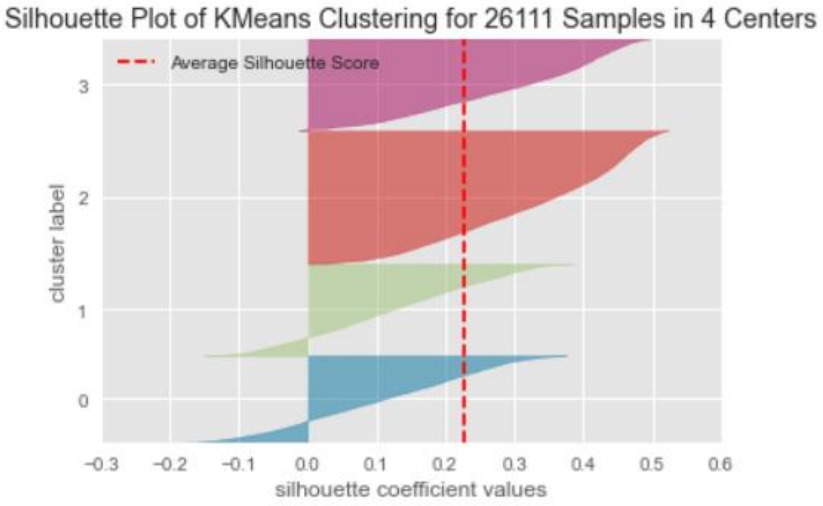
Ввод [103]:
```
from sklearn.cluster import KMeans
no_cv = [(slice(None), slice(None))]
kmeans = KMeans()
param_grid = dict(n_clusters=range(2, 5))
grid_search = GridSearchCV(kmeans, param_grid, cv=no_cv, scoring=silhouette_score)
grid_search.fit(train_df[train_columns])
train_df['cluster'] = grid_search.best_estimator_.labels_

print(f"Наилучшие параметры: {grid_search.best_params_}")
print(f"Коэффициент силуэта: {grid_search.best_score_}")
```

```
Наилучшие параметры: {'n_clusters': 4}
Коэффициент силуэта: 0.22774501562803573
```

```
train_df['cluster'].value_counts()
```

```
2    8597
3    5962
1    5945
0    5607
Name: cluster, dtype: int64
```

Силуэт кластера

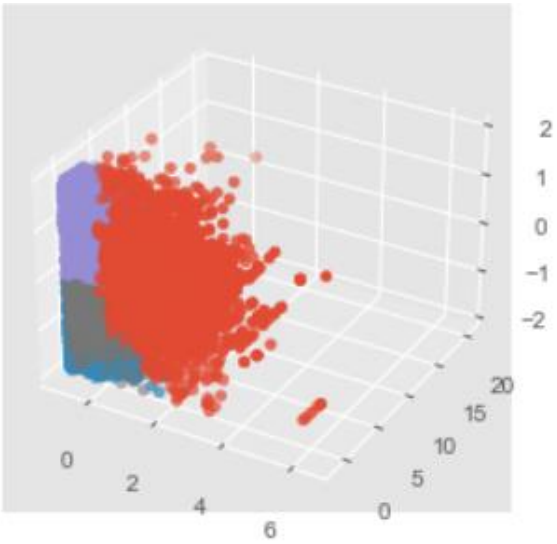


Кластеры:

```
Cluster 0 ((5607, 28))
         lead_time  stays_in_weeks_nights  arrival_date_week_number
count  5607.000000            5607.000000               5607.000000
mean      1.441212               0.878752                  0.172486
std       0.935357               1.424202                  0.703727
min      -0.931430              -1.320191                 -1.887793
```

```
25%       0.851774             -0.231560                 -0.276158
50%       1.321038              0.494194                  0.163379
75%       1.989740              1.219949                  0.602916
max       6.447749             20.452436                  1.921527
0    3474
1    2133
Name: is_canceled, dtype: int64
Cluster 2 ((8597, 28))
         lead_time  stays_in_weeks_nights  arrival_date_week_number
count  8597.000000            8597.000000               8597.000000
mean     -0.467893              -0.316655                 -1.026443
std       0.494289               0.629876                  0.493351
min      -0.931430              -1.320191                 -1.887793
25%      -0.872772              -0.957314                 -1.448256
50%      -0.638140              -0.231560                 -1.081975
75%      -0.192339               0.131317                 -0.642438
max       1.860692               3.760088                  0.016867
0    6460
1    2137
Name: is_canceled, dtype: int64
Cluster 3 ((5962, 28))
         lead_time  stays_in_weeks_nights  arrival_date_week_number
count  5962.000000            5962.000000               5962.000000
mean     -0.482969              -0.318658                  1.150457
std       0.483084               0.638849                  0.502518
min      -0.931430              -1.320191                  0.016867
25%      -0.872772              -0.957314                  0.749429
50%      -0.661603              -0.594437                  1.188965
75%      -0.227534               0.131317                  1.555246
max       1.802034               3.034334                  1.921527
0    4671
1    1291
Name: is_canceled, dtype: int64
Cluster 1 ((5945, 28))
         lead_time  stays_in_weeks_nights  arrival_date_week_number
count  5945.000000            5945.000000               5945.000000
mean     -0.198308              -0.051312                  0.167902
std       0.656550               0.720303                  0.642551
min      -0.931430              -0.957314                 -1.887793
25%      -0.743724              -0.594437                 -0.276158
50%      -0.391776              -0.231560                  0.236636
75%       0.218267               0.131317                  0.529660
max       3.350606               4.122966                  1.921527
0    4354
1    1591
Name: is_canceled, dtype: int64
```

Визуализация



Сравнение с реальными данными

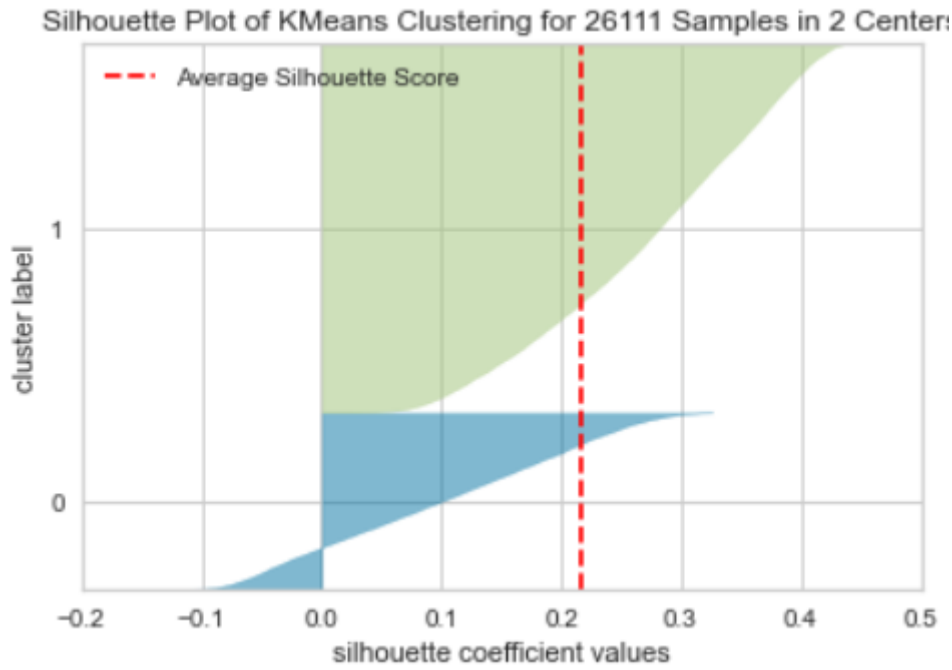
Попробуем разбить на 2 кластера. Реальные данные имеют ключевой параметр – отмена бронирование (0 или 1).

Ввод [43]:
```
from sklearn.cluster import KMeans
no_cv = [(slice(None), slice(None))]
kmeans = KMeans(n_clusters = 2)

param_grid = dict(n_clusters=range(2, 3))
grid_search = GridSearchCV(kmeans, param_grid, cv=no_cv, scoring=silhouette_score)
grid_search.fit(train_df[train_columns])
train_df['cluster'] = grid_search.best_estimator_.labels_
```

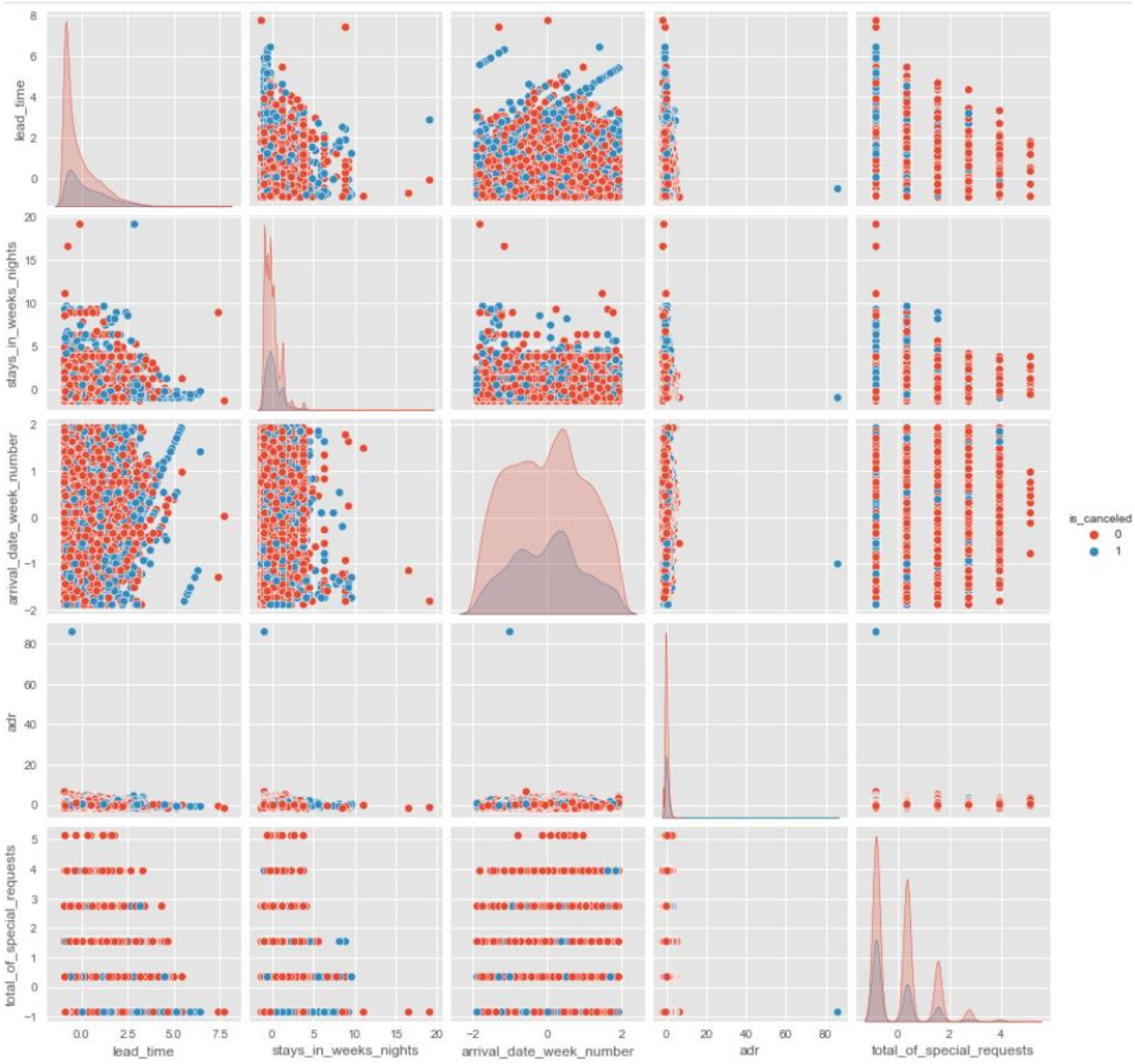
```
C:\Users\admin\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cl
ault value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
  warnings.warn(
C:\Users\admin\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cl
ault value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
  warnings.warn(
C:\Users\admin\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cl
ault value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_in
  warnings.warn(
```

Ввод [18]:
```
train_df['cluster'].value_counts()
```

Out[18]:
```
1    17653
0     8458
Name: cluster, dtype: int64
```




```
Cluster 1 ((17653, 28))
          lead_time  stays_in_weeks_nights  arrival_date_week_number
count  17653.000000           17653.000000              17653.000000
mean      -0.513399              -0.363607                 -0.129463
std        0.443425               0.548275                  1.069236
min       -0.931381              -1.307394                 -1.887925
25%       -0.873157              -0.950749                 -1.006664
50%       -0.663553              -0.594104                 -0.272280
75%       -0.279278               0.119187                  0.755858
max        1.828412               2.972349                  1.930873
0    13575
1     4078
Name: is_canceled, dtype: int64
Cluster 0 ((8458, 28))
         lead_time  stays_in_weeks_nights  arrival_date_week_number
count  8458.000000            8458.000000               8458.000000
mean      1.071535               0.758897                  0.270208
std       0.989232               1.268106                  0.770192
min      -0.931381              -1.307394                 -1.887925
25%       0.407759              -0.237458                 -0.198841
50%       0.978349               0.475832                  0.315228
75%       1.618808               1.189123                  0.682420
max       5.985569              20.091321                  1.930873
0    5407
1    3051
Name: is_canceled, dtype: int64
```


lead_time
stays_in_weeks_nights
arrival_date_week_number
adr
total_of_special_requests
is_canceled
0
1

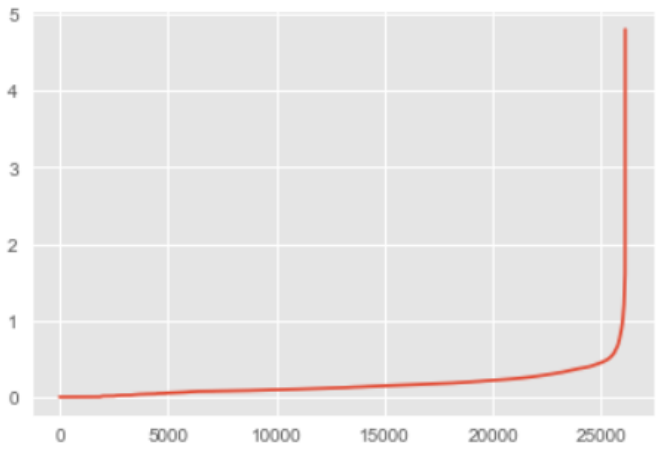

lead_time
stays_in_weeks_nights
arrival_date_week_number
adr
total_of_special_requests
cluster
0
1

DBSCAN

Вычисление оптимальной эпсилон для лучшей кластеризации при различных при различных min_samples.

Ввод [120]:
```
from sklearn.neighbors import NearestNeighbors

for min_samples in [i * len(train_columns) for i in range(1, 10)]:
    neighbors = NearestNeighbors(n_neighbors=min_samples)
    neighbors_fit = neighbors.fit(train_df[train_columns])
    distances, indices = neighbors_fit.kneighbors(train_df[train_columns])
    distances = np.sort(distances, axis=0)
    distances = distances[:,1]
    mp.plot(distances)
    mp.show()
```

Оптимальный эпсилон – 0,55



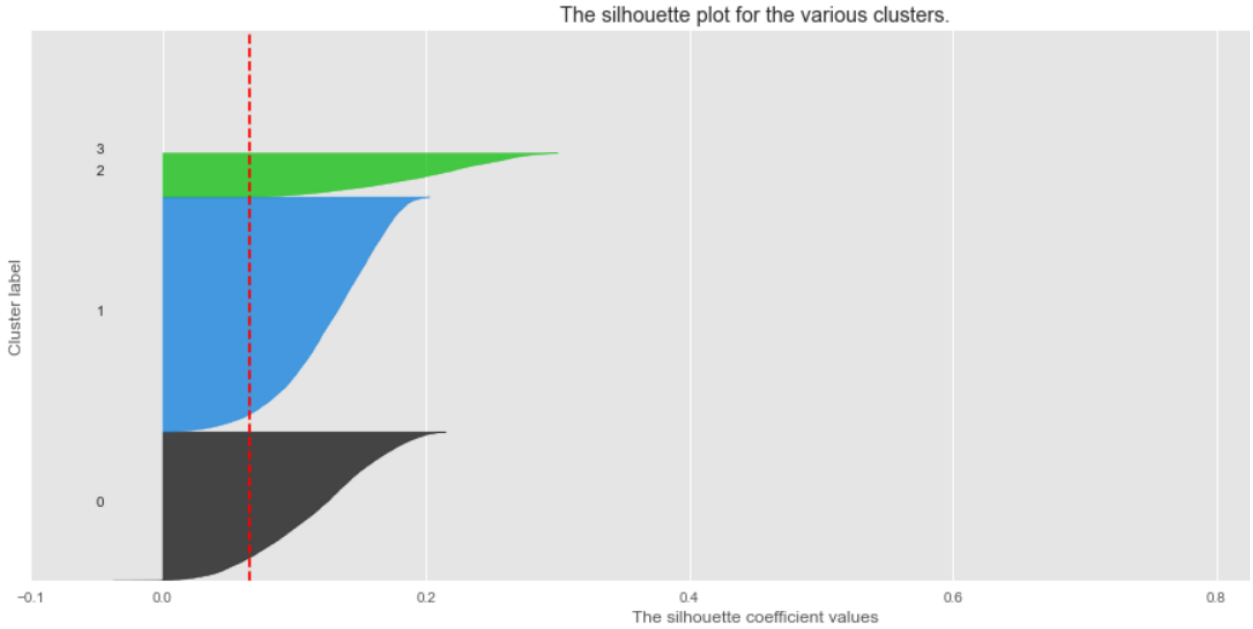
Использование GridSearchCV для подбора наилучшего параметра min_samples. Оценка силуэт кластера.

Ввод [123]:
```
from sklearn.cluster import DBSCAN

param_grid = dict(min_samples=[i * len(train_columns) for i in range(4, 9)])
dbscan = DBSCAN(eps=0.55)
grid_search = GridSearchCV(dbscan, param_grid, cv=no_cv, scoring=silhouette_score)
grid_search.fit(train_df[train_columns])
train_df['cluster'] = grid_search.best_estimator_.labels_

print(f"Наилучшие параметры: {grid_search.best_params_}")
print(f"Коэффициент силуэта: {grid_search.best_score_}")
```

```
Наилучшие параметры: {'min_samples': 40}
Коэффициент силуэта: 0.0657283055053541
```

```
Ввод [124]: train_df['cluster'].value_counts()

Out[124]:  1    11153
           0     7068
          -1     5814
           2     2076
          Name: cluster, dtype: int64
```

Силуэт кластера

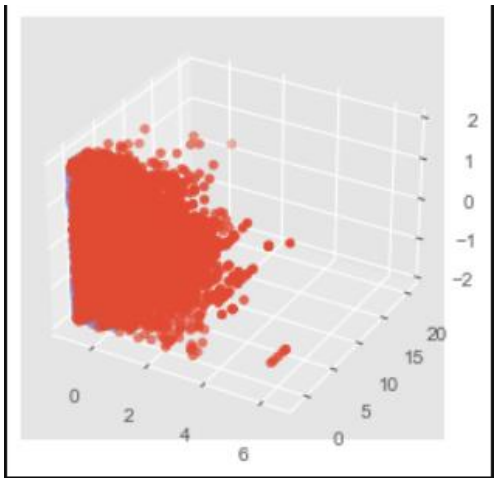


Кластеры:

```
Cluster -1 ((5814, 28))
         lead_time  stays_in_weeks_nights  arrival_date_week_number
count  5814.000000            5814.000000               5814.000000
mean      0.855632               0.810510                  0.163909
std       1.287566               1.498535                  0.921352
min      -0.931430              -1.320191                 -1.887793
25%      -0.239265              -0.231560                 -0.495926
50%       0.710995               0.494194                  0.236636
75%       1.766839               1.219949                  0.749429
max       6.447749              20.452436                  1.921527
0    3899
1    1915
Name: is_canceled, dtype: int64
Cluster 0 ((7068, 28))
         lead_time  stays_in_weeks_nights  arrival_date_week_number
count  7068.000000            7068.000000               7068.000000
mean     -0.230687              -0.202809                 -0.041868
std       0.725260               0.618791                  0.984472
min      -0.931430              -1.320191                 -1.887793
25%      -0.814114              -0.594437                 -0.862207
50%      -0.497361              -0.231560                  0.016867
75%       0.147878               0.131317                  0.676172
max       2.611515               1.945703                  1.921527
0    5649
1    1419
Name: is_canceled, dtype: int64
```

```
Cluster 1 ((11153, 28))
          lead_time  stays_in_weeks_nights  arrival_date_week_number
count  11153.000000           11153.000000              11153.000000
mean      -0.222580              -0.233089                 -0.081138
std        0.772323               0.665538                  1.026305
min       -0.931430              -1.320191                 -1.887793
25%       -0.861040              -0.957314                 -0.935463
50%       -0.520824              -0.231560                 -0.129645
75%        0.218267               0.131317                  0.749429
max        2.775757               1.945703                  1.921527
0    7694
1    3459
Name: is_canceled, dtype: int64
Cluster 2 ((2076, 28))
         lead_time  stays_in_weeks_nights  arrival_date_week_number
count  2076.000000            2076.000000               2076.000000
mean     -0.415081              -0.327173                  0.119412
std       0.520428               0.477631                  1.054911
min      -0.931430              -0.957314                 -1.887793
25%      -0.825845              -0.594437                 -0.788951
50%      -0.579482              -0.231560                  0.236636
75%      -0.180607               0.131317                  0.969197
max       1.743376               0.857072                  1.921527
0    1717
1     359
Name: is_canceled, dtype: int64
```

Визуализация



Иерархическая кластеризация

Подбор параметра linkage – критерий связи. Критерий связи определяет, какое расстояние использовать между наборами наблюдений. Алгоритм объединит пары кластеров, которые минимизируют этот критерий.

«ward» минимизирует дисперсию объединяемых кластеров.

«average» использует среднее значение расстояний каждого наблюдения из двух наборов.

«complete» или «maximum» связь использует максимальные расстояния между всеми наблюдениями двух наборов.

«single» использует минимальное расстояние между всеми наблюдениями двух наборов

Ввод [131]:
```
from sklearn.cluster import AgglomerativeClustering
agglomerative_clustering = AgglomerativeClustering(n_clusters=2)
param_grid = dict(linkage=['ward', 'complete', 'average', 'single'])
grid_search = GridSearchCV(agglomerative_clustering, param_grid, cv=no_cv, scoring=silhouette_score)
grid_search.fit(train_df[train_columns])
train_df['cluster'] = grid_search.best_estimator_.labels_

print(f"Наилучшие параметры: {grid_search.best_params_}")
print(f"Коэффициент силуэта: {grid_search.best_score_}")
```

```
Наилучшие параметры: {'linkage': 'average'}
Коэффициент силуэта: 0.8375503802771566
```

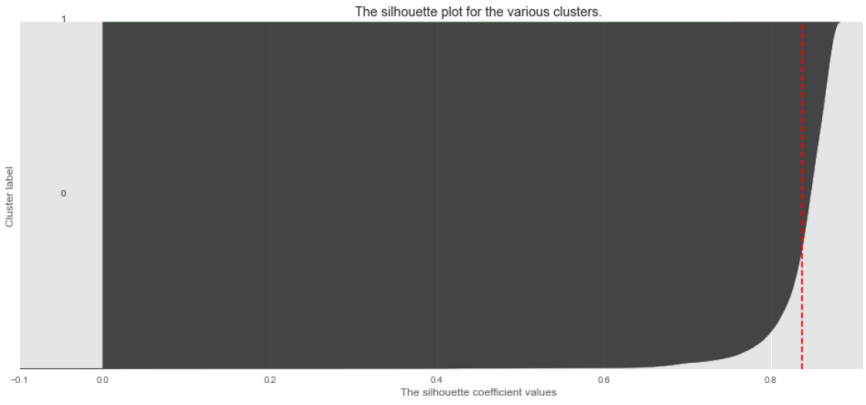
Ввод [133]:
```
train_df['cluster'].value_counts()
```

Out[133]:
```
0    26108
1        3
Name: cluster, dtype: int64
```

Силуэт кластера

Ввод [136]:
```
plot_silhouette(train_df[train_columns], train_df['cluster'])
```

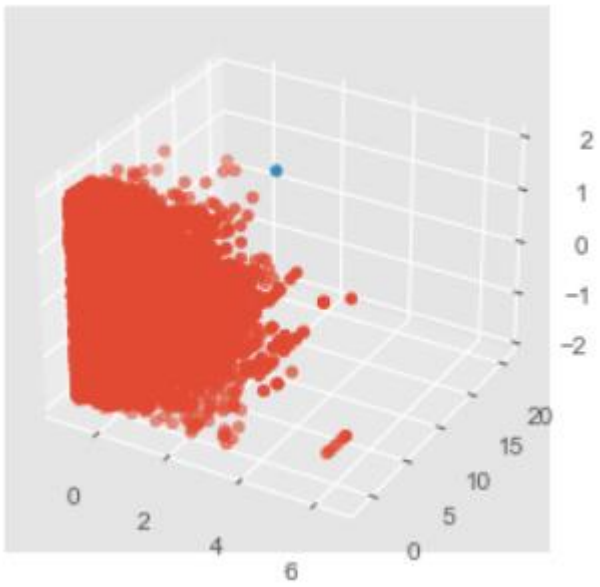
```
For n_clusters = 2 The average silhouette_score is : 0.8375503802771566
```



Кластеры:

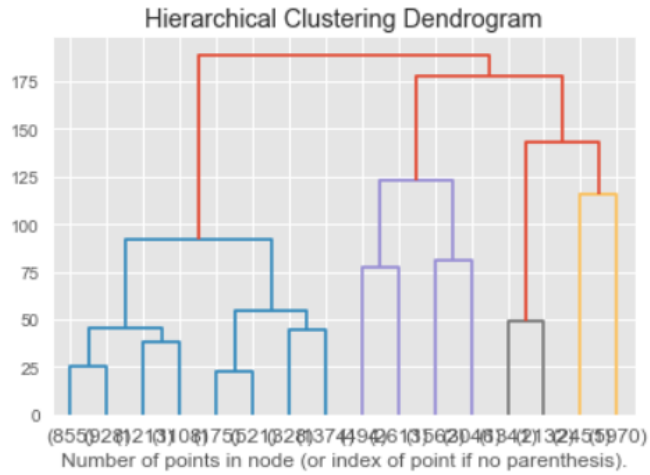
```
Cluster 0 ((26108, 28))
          lead_time  stays_in_weeks_nights  arrival_date_week_number
count  26108.000000           26108.000000              26108.000000
mean       0.000019              -0.002058                  0.000124
std        1.000061               0.981062                  0.999954
min       -0.931430              -1.320191                 -1.887793
25%       -0.790651              -0.594437                 -0.788951
50%       -0.368313              -0.231560                  0.016867
75%        0.535021               0.494194                  0.749429
max        6.447749              11.380508                  1.921527
0    18956
1     7152
Name: is_canceled, dtype: int64
Cluster 1 ((3, 28))
       lead_time  stays_in_weeks_nights  arrival_date_week_number
count   3.000000               3.000000                  3.000000
mean   -0.168876              17.912296                 -1.081975
std     0.601409               3.225322                  1.205948
min    -0.802382              14.283525                 -1.814537
25%    -0.450434              16.642226                 -1.777909
50%    -0.098486              19.000927                 -1.741281
75%     0.147878              19.726681                 -0.715694
max     0.394241              20.452436                  0.309892
0    3
Name: is_canceled, dtype: int64
```

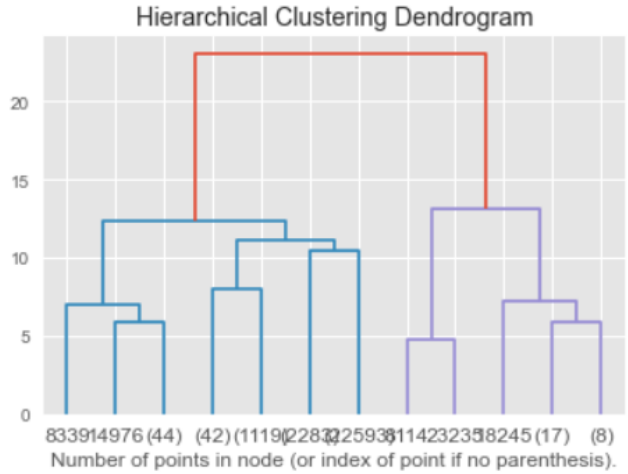
Визуализация

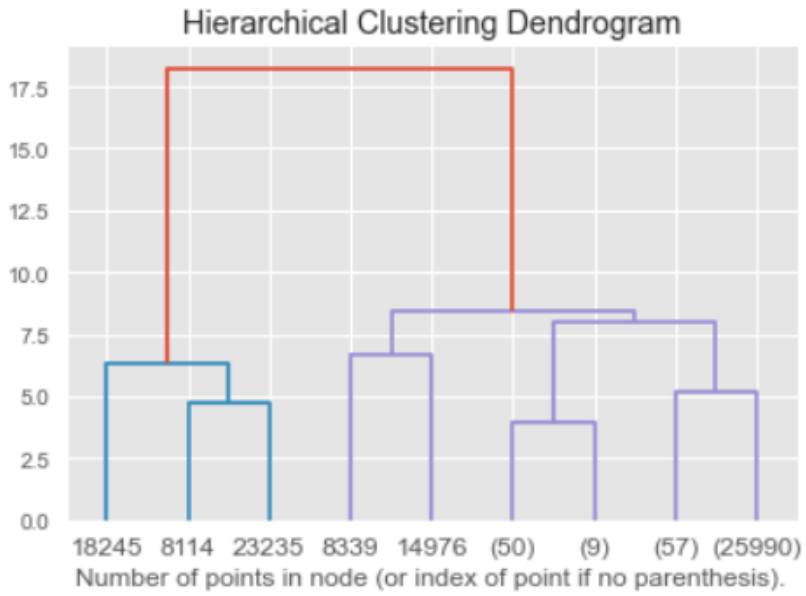


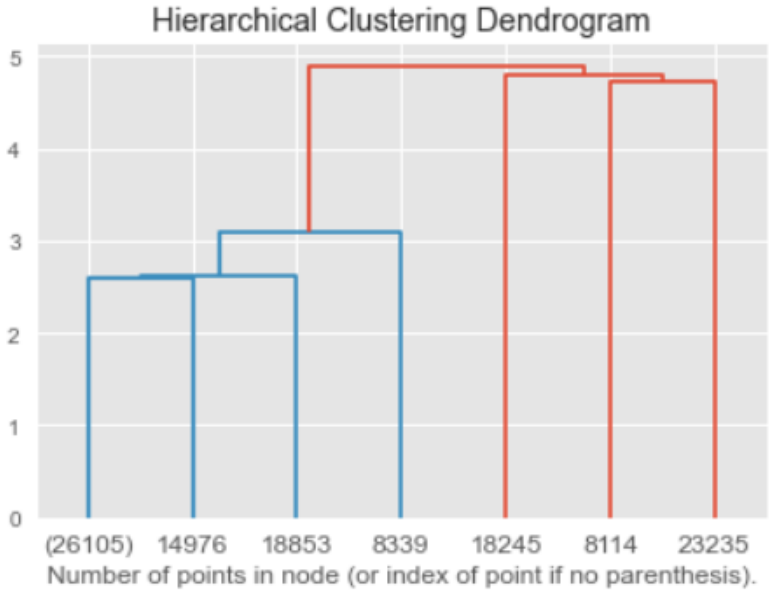
Дендограммы с разными линковками

Ввод [134]:
```
for linkage in ['ward', 'complete', 'average', 'single']:
    plot_dendrogram(train_df[train_columns], linkage=linkage)
```






Hierarchical Clustering Dendrogram
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
18245 8114 23235 8339 14976 (50) (9) (57) (25990)
Number of points in node (or index of point if no parenthesis).


Hierarchical Clustering Dendrogram
5
4
3
2
1
0
(26105) 14976 18853 8339 18245 8114 23235
Number of points in node (or index of point if no parenthesis).