MGSC410_Final

May 5, 2022

[189]:
```
import warnings
warnings.filterwarnings('ignore')

import pandas as pd
import numpy as np
from plotnine import *

from sklearn.preprocessing import StandardScaler
from sklearn.neighbors import NearestNeighbors

from sklearn.cluster import DBSCAN


from sklearn.cluster import KMeans
from sklearn.mixture import GaussianMixture

from sklearn.metrics import silhouette_score

%matplotlib inline
```

0.1 Understanding the subscriber segments present in the database

[202]:
```
subDF = pd.read_csv("https://raw.githubusercontent.com/tmoore-byte/MGSC-410/
 ↪main/subscriptionData.csv")

subDF = subDF.loc[subDF["subscriptionLength_months"] <= 60]
subDF.head()

subDF.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 255844 entries, 0 to 331659
Data columns (total 26 columns):
 #   Column                    Non-Null Count   Dtype
---  ------                    --------------   -----
 0   ID                        255844 non-null  int64
 1   App.Session.Platform      233633 non-null  object
```

```
 2   App.Activity.Type          250113 non-null  object
 3   Language                   255844 non-null  object
 4   Subscription.Type          255844 non-null  object
 5   Subscription.Event.Type    255844 non-null  object
 6   Purchase.Store             255844 non-null  object
 7   Purchase.Amount            255844 non-null  float64
 8   Currency                   255844 non-null  object
 9   Demo.User                  255844 non-null  int64
 10  Free.Trial.User            255844 non-null  int64
 11  Auto.Renew                 255844 non-null  int64
 12  Country                    255844 non-null  object
 13  User.Type                  255844 non-null  object
 14  Lead.Platform              255844 non-null  object
 15  Email.Subscriber           255844 non-null  int64
 16  Push.Notifications         255844 non-null  int64
 17  Send.Count                 255844 non-null  int64
 18  Unique.Open.Count          255844 non-null  int64
 19  Unique.Click.Count         255844 non-null  int64
 20  subscriptionLength_months  255844 non-null  int64
 21  UniqueOpenRate             255844 non-null  float64
 22  UniqueClickRate            255844 non-null  float64
 23  unique Currencies          21 non-null      object
 24  Exchange rates             21 non-null      float64
 25  Purchase.Amount.USD        255844 non-null  float64
dtypes: float64(5), int64(10), object(11)
memory usage: 52.7+ MB
```

[225]:
```
features = ["Purchase.Amount.USD", "UniqueOpenRate", "UniqueClickRate",␣
 ↪"subscriptionLength_months"]

X = subDF[features]

z = StandardScaler()
X[features] = z.fit_transform(X)
```

[226]:
```
X = X.sample(frac = 0.05)
type(X)
X.head()
```

[226]:

	Purchase.Amount.USD	UniqueOpenRate	UniqueClickRate	\
274630	-0.335289	-0.739027	-0.297699	
203871	-0.335285	-0.695312	-0.297699	
163297	-0.335287	-0.739027	-0.297699	
199311	-0.335288	-0.739027	-0.297699	
209925	-0.335289	1.883891	0.017189	

	subscriptionLength_months

```
274630                  -0.574319
203871                  -0.111010
163297                   0.815609
199311                   2.205538
209925                   0.352300
```

[ ]:

[193]:
```
#km = KMeans(n_clusters = 3) #2 clusters

#km.fit(X)

#membership = km.predict(X) #what class did each data point go in

#X["cluster"] = membership


kValuesKM = {}

for k in range(4,10):
    km = KMeans(n_clusters = k)
    km.fit(X)
    membership = km.predict(X)
    kValuesKM["K of " + str(k)] = (silhouette_score(X, membership).round(4))

max_key = max(kValuesKM, key=kValuesKM.get)

print(max_key)
```

```
K of 5
```

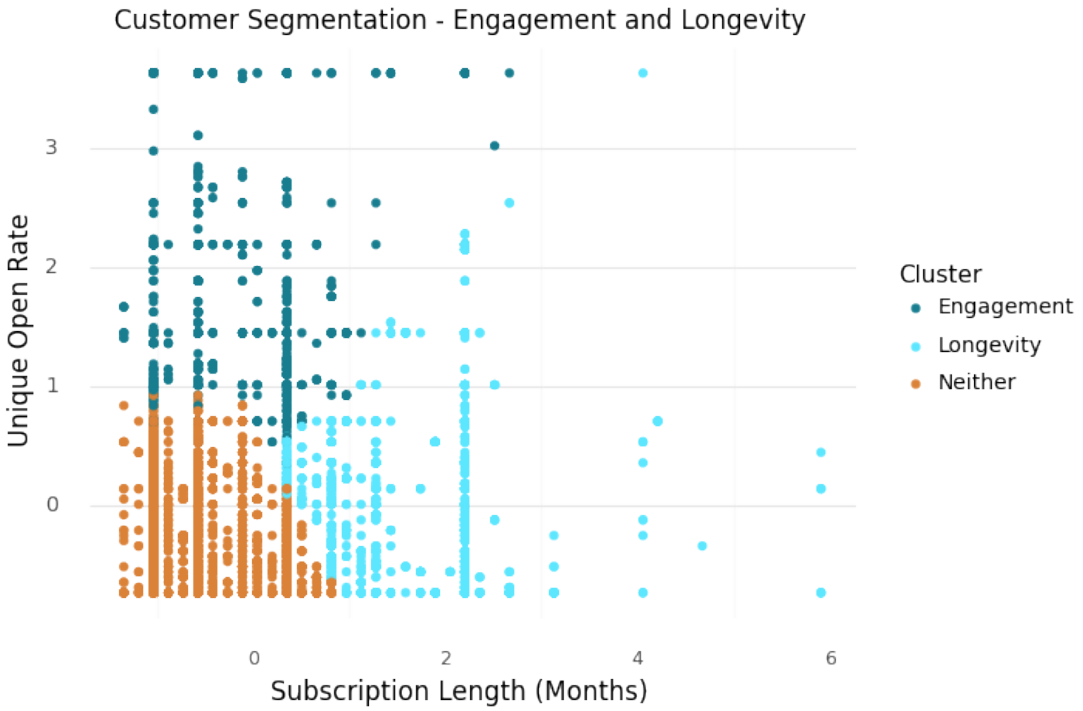
[228]:
```
km = KMeans(n_clusters = 5)
km.fit(X)
membership = km.predict(X)
X["cluster"] = membership

silhouette_score(X, membership)
```

[228]:
```
0.5518819147808789
```

[233]:
```
#X = X.loc[X["cluster"] != 3]
#X = X.loc[X["cluster"] != 2]
X["cluster"].replace({ 1 : "Longevity", 4 : "Engagement", 0 : "Neither"},␣
↪inplace =True)
```

[234]:
```
(ggplot(X, aes("subscriptionLength_months", "UniqueOpenRate", color =␣
↪"cluster")) +
```

```
 geom_point(size = 1.5) +
 scale_color_manual(["#177D8F", "#5CE7FF", "#DB8239"]) +
 labs(x = "Subscription Length (Months)", y = "Unique Open Rate", color =␣
 ↪"Cluster") +
 ggtitle("Customer Segmentation - Engagement and Longevity") +
 theme_minimal() +
theme(panel_grid_major_x = element_blank(),
      panel_grid_minor_y = element_blank(),
      axis_text_y = element_text(size = 10),
      axis_title_x = element_text(size = 12),
      axis_title_y = element_text(size = 12),
      plot_title = element_text(size = 12),
      legend_text = element_text(size = 10)))
```



[234]: <ggplot: (8778060167045)>