9-Copy1

October 28, 2018

1 9 Clustering

1.1 1. DBSCAN

Using DBSCAN iterate (for-loop) through different values of `min_samples` (1 to 10) and `epsilon` (.05 to .5, in steps of .01) to find clusters in the road-data used in the Lesson and calculate the Silohouette Coeff for `min_samples` and `epsilon`. Plot ***one*** line plot with the multiple lines generated from the min_samples and epsilon values. Use a 2D array to store the SilCoeff values, one dimension represents `min_samples`, the other represents epsilon.

In [60]:
```
import pandas as pd
# allow plots to appear in the notebook
%matplotlib notebook
import matplotlib.pyplot as plt
import seaborn
from mpl_toolkits.mplot3d import Axes3D
plt.rcParams['font.size'] = 14
plt.rcParams['figure.figsize'] = (10.0, 5.0)


X = pd.read_csv('../data/3D_spatial_network.txt.gz', header=None, names=['osm', 'lat','
X = X.drop(['osm'], axis=1).sample(10000)
X.head()
```

Out[60]:

	lat	lon
108514324	57.116136	7.935940
145155209	57.323512	28.719403
139894123	57.491219	42.535347
80477583	57.418888	19.245527
103953351	56.871157	20.010514

In [61]:
```
# K-means with N clusters
N = 7
from sklearn.cluster import KMeans
km = KMeans(n_clusters=N, random_state=1)
km.fit(X)
```

```
Out[61]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=300,
             n_clusters=7, n_init=10, n_jobs=1, precompute_distances='auto',
             random_state=1, tol=0.0001, verbose=0)
```

```
In [62]: # review the cluster labels
         set(km.labels_)
```

```
Out[62]: {0, 1, 2, 3, 4, 5, 6}
```

```
In [63]: X['cluster'] = km.predict(X)
```

```
In [64]: XX = X.copy()
         XX['lat'] = (X.lat - X.lat.mean())/X.lat.std()
         XX['lon'] = (X.lon - X.lon.mean())/X.lon.std()
```

```
In [65]: # calculate SC for K=7
         from sklearn import metrics
         metrics.silhouette_score(XX[['lon', 'lat']], XX.cluster)
```

```
Out[65]: 0.10349585363140311
```

```
In [107]: from sklearn.cluster import DBSCAN
          from tqdm import tqdm
          import numpy as np

          start   = 0.0
          stop    = 0.45
          step    = 0.01
          my_list = np.arange(start, stop+step, step)

          startb   = 1
          stopb    = 10
          stepb    = .2 # To scale proportionately with epsilon increments
          my_listb = np.arange(startb, stopb+stepb, stepb)


          my_range = range(45)

          one = []
          for i in tqdm(my_range):
              dbscan = DBSCAN(eps = .05 + my_list[i] , min_samples = 1 + my_listb[i])
              XX.cluster = dbscan.fit_predict(XX[['lat','lon']])
              one.append(metrics.silhouette_score(XX[['lat', 'lon']], XX.cluster))

          two = []
          for i in tqdm(my_range):
              dbscan = DBSCAN(eps = .05 + my_list[i] , min_samples = 2 + my_listb[i])
              XX.cluster = dbscan.fit_predict(XX[['lat','lon']])
              two.append(metrics.silhouette_score(XX[['lat', 'lon']], XX.cluster))
```

```python
three = []
for i in tqdm(my_range):
    dbscan = DBSCAN(eps = .05 + my_list[i] , min_samples = 3 + my_listb[i])
    XX.cluster = dbscan.fit_predict(XX[['lat','lon']])
    three.append(metrics.silhouette_score(XX[['lat', 'lon']], XX.cluster))

four = []
for i in tqdm(my_range):
    dbscan = DBSCAN(eps = .05 + my_list[i] , min_samples = 4 + my_listb[i])
    XX.cluster = dbscan.fit_predict(XX[['lat','lon']])
    four.append(metrics.silhouette_score(XX[['lat', 'lon']], XX.cluster))

five = []
for i in tqdm(my_range):
    dbscan = DBSCAN(eps = .05 + my_list[i] , min_samples = 5 + my_listb[i])
    XX.cluster = dbscan.fit_predict(XX[['lat','lon']])
    five.append(metrics.silhouette_score(XX[['lat', 'lon']], XX.cluster))

six = []
for i in tqdm(my_range):
    dbscan = DBSCAN(eps = .05 + my_list[i] , min_samples = 6 + my_listb[i])
    XX.cluster = dbscan.fit_predict(XX[['lat','lon']])
    six.append(metrics.silhouette_score(XX[['lat', 'lon']], XX.cluster))

seven = []
for i in tqdm(my_range):
    dbscan = DBSCAN(eps = .05 + my_list[i] , min_samples = 7 + my_listb[i])
    XX.cluster = dbscan.fit_predict(XX[['lat','lon']])
    seven.append(metrics.silhouette_score(XX[['lat', 'lon']], XX.cluster))

eight = []
for i in tqdm(my_range):
    dbscan = DBSCAN(eps = .05 + my_list[i] , min_samples = 8 + my_listb[i])
    XX.cluster = dbscan.fit_predict(XX[['lat','lon']])
    eight.append(metrics.silhouette_score(XX[['lat', 'lon']], XX.cluster))

nine = []
for i in tqdm(my_range):
    dbscan = DBSCAN(eps = .05 + my_list[i] , min_samples = 9 + my_listb[i])
    XX.cluster = dbscan.fit_predict(XX[['lat','lon']])
    nine.append(metrics.silhouette_score(XX[['lat', 'lon']], XX.cluster))

ten = []
for i in tqdm(my_range):
    dbscan = DBSCAN(eps = .05 + my_list[i] , min_samples = 10 + my_listb[i])
    XX.cluster = dbscan.fit_predict(XX[['lat','lon']])
    ten.append(metrics.silhouette_score(XX[['lat', 'lon']], XX.cluster))
```

```
0%|          | 0/45 [00:00<?, ?it/s]

2%|          | 1/45 [00:10<07:34, 10.32s/it]

4%|          | 2/45 [00:17<06:41,  9.33s/it]

7%|          | 3/45 [00:23<05:55,  8.46s/it]

9%|          | 4/45 [00:30<05:23,  7.89s/it]
```

11%| | 5/45 [00:36<05:00, 7.51s/it]

13%| | 6/45 [00:43<04:45, 7.31s/it]

16%| | 7/45 [00:50<04:35, 7.25s/it]

18%| | 8/45 [00:58<04:30, 7.31s/it]

20%| | 9/45 [01:04<04:15, 7.10s/it]

22%| | 10/45 [01:11<04:02, 6.93s/it]

```
24%|          | 11/45 [01:18<03:58,  7.01s/it]

27%|          | 12/45 [01:25<03:47,  6.88s/it]

29%|          | 13/45 [01:31<03:36,  6.76s/it]

31%|          | 14/45 [01:38<03:32,  6.84s/it]

33%|        | 15/45 [01:45<03:23,  6.79s/it]

36%|        | 16/45 [01:52<03:15,  6.75s/it]
```

```
38%|      | 17/45 [01:58<03:06,  6.64s/it]
```

```
40%|      | 18/45 [02:04<02:56,  6.55s/it]
```

```
42%|     | 19/45 [02:11<02:48,  6.48s/it]
```

```
44%|     | 20/45 [02:17<02:40,  6.44s/it]
```

```
47%|     | 21/45 [02:24<02:35,  6.48s/it]
```

```
49%|     | 22/45 [02:33<02:48,  7.34s/it]
```

```
51%|     | 23/45 [02:40<02:37,  7.18s/it]
```

```
53%|    | 24/45 [02:46<02:26,  6.98s/it]
```

```
56%|    | 25/45 [02:53<02:17,  6.90s/it]
```

```
58%|    | 26/45 [03:00<02:11,  6.90s/it]
```

```
60%|    | 27/45 [03:07<02:03,  6.88s/it]
```

```
62%|   | 28/45 [03:14<01:56,  6.88s/it]
```

```
64%|   | 29/45 [03:20<01:49,  6.86s/it]

67%|   | 30/45 [03:26<01:39,  6.63s/it]

69%|   | 31/45 [03:33<01:31,  6.54s/it]

71%|   | 32/45 [03:40<01:26,  6.68s/it]

73%| | 33/45 [03:46<01:19,  6.66s/it]

76%| | 34/45 [03:53<01:13,  6.70s/it]
```

78%| | 35/45 [04:00<01:06, 6.68s/it]

80%| | 36/45 [04:07<01:02, 6.91s/it]

82%| | 37/45 [04:16<00:58, 7.30s/it]

84%| | 38/45 [04:22<00:49, 7.07s/it]

87%| | 39/45 [04:29<00:41, 6.97s/it]

89%| | 40/45 [04:36<00:34, 6.96s/it]

```
 91%| | 41/45 [04:44<00:29,  7.41s/it]
```

```
 93%|| 42/45 [04:51<00:22,  7.33s/it]
```

```
 96%|| 43/45 [04:58<00:14,  7.23s/it]
```

```
 98%|| 44/45 [05:05<00:07,  7.08s/it]
```

```
100%|| 45/45 [05:12<00:00,  6.92s/it]
```

```
  0%|          | 0/45 [00:00<?, ?it/s]

  2%|         | 1/45 [00:07<05:13,  7.12s/it]

  4%|         | 2/45 [00:13<04:56,  6.89s/it]

  7%|         | 3/45 [00:19<04:39,  6.65s/it]

  9%|         | 4/45 [00:26<04:32,  6.65s/it]

 11%|         | 5/45 [00:32<04:23,  6.59s/it]
```

```
13%|          | 6/45 [00:39<04:15,  6.56s/it]
```

```
16%|          | 7/45 [00:45<04:07,  6.50s/it]
```

```
18%|          | 8/45 [00:52<04:00,  6.50s/it]
```

```
20%|          | 9/45 [00:58<03:52,  6.44s/it]
```

```
22%|          | 10/45 [01:04<03:47,  6.49s/it]
```

```
24%|          | 11/45 [01:11<03:40,  6.47s/it]
```

```
27%|          | 12/45 [01:17<03:31,  6.42s/it]
```

```
29%|          | 13/45 [01:24<03:24,  6.41s/it]
```

```
31%|          | 14/45 [01:30<03:18,  6.41s/it]
```

```
33%|        | 15/45 [01:36<03:11,  6.39s/it]
```

```
36%|        | 16/45 [01:43<03:06,  6.44s/it]
```

```
38%|        | 17/45 [01:49<03:00,  6.43s/it]
```

```
40%|      | 18/45 [01:56<02:53,  6.43s/it]
```

```
42%|     | 19/45 [02:02<02:46,  6.41s/it]
```

```
44%|     | 20/45 [02:08<02:39,  6.38s/it]
```

```
47%|     | 21/45 [02:15<02:37,  6.55s/it]
```

```
49%|     | 22/45 [02:22<02:31,  6.59s/it]
```

```
51%|     | 23/45 [02:28<02:23,  6.52s/it]
```

```
53%|     | 24/45 [02:35<02:17,  6.56s/it]

56%|     | 25/45 [02:41<02:10,  6.51s/it]

58%|     | 26/45 [02:48<02:02,  6.47s/it]

60%|     | 27/45 [02:54<01:56,  6.47s/it]

62%|    | 28/45 [03:01<01:49,  6.45s/it]

64%|    | 29/45 [03:07<01:42,  6.43s/it]
```

```
67%|   | 30/45 [03:13<01:36,  6.42s/it]
```

```
69%|   | 31/45 [03:20<01:30,  6.47s/it]
```

```
71%|   | 32/45 [03:27<01:24,  6.50s/it]
```

```
73%|  | 33/45 [03:33<01:17,  6.47s/it]
```

```
76%|  | 34/45 [03:39<01:11,  6.46s/it]
```

```
78%|  | 35/45 [03:46<01:05,  6.56s/it]
```

```
80%|  | 36/45 [03:53<00:58,  6.50s/it]
```

```
82%| | 37/45 [03:59<00:51,  6.49s/it]
```

```
84%| | 38/45 [04:06<00:45,  6.50s/it]
```

```
87%| | 39/45 [04:12<00:39,  6.62s/it]
```

```
89%| | 40/45 [04:20<00:33,  6.76s/it]
```

```
91%| | 41/45 [04:26<00:26,  6.67s/it]
```

```
 93%|| 42/45 [04:33<00:20,  6.68s/it]
```

```
 96%|| 43/45 [04:39<00:13,  6.64s/it]
```

```
 98%|| 44/45 [04:46<00:06,  6.57s/it]
```

```
100%|| 45/45 [04:52<00:00,  6.58s/it]
```

```
  0%|          | 0/45 [00:00<?, ?it/s]
```

```
  2%|          | 1/45 [00:06<04:40,  6.37s/it]
```

```
  4%|          | 2/45 [00:12<04:29,  6.27s/it]
```

```
  7%|          | 3/45 [00:18<04:23,  6.28s/it]
```

```
  9%|          | 4/45 [00:25<04:21,  6.39s/it]
```

```
 11%|          | 5/45 [00:31<04:14,  6.35s/it]
```

```
 13%|         | 6/45 [00:38<04:08,  6.38s/it]
```

```
16%|          | 7/45 [00:44<04:00,  6.33s/it]
```

```
18%|          | 8/45 [00:50<03:54,  6.33s/it]
```

```
20%|          | 9/45 [00:56<03:47,  6.31s/it]
```

```
22%|          | 10/45 [01:03<03:40,  6.31s/it]
```

```
24%|          | 11/45 [01:09<03:34,  6.30s/it]
```

```
27%|          | 12/45 [01:15<03:27,  6.30s/it]
```

```
29%|        | 13/45 [01:22<03:21,  6.31s/it]
```

```
31%|        | 14/45 [01:28<03:15,  6.31s/it]
```

```
33%|       | 15/45 [01:34<03:09,  6.32s/it]
```

```
36%|       | 16/45 [01:41<03:03,  6.32s/it]
```

```
38%|       | 17/45 [01:47<02:57,  6.32s/it]
```

```
40%|       | 18/45 [01:53<02:51,  6.35s/it]
```

```
42%|      | 19/45 [02:00<02:45,  6.35s/it]
```

```
44%|      | 20/45 [02:06<02:39,  6.36s/it]
```

```
47%|      | 21/45 [02:12<02:32,  6.36s/it]
```

```
49%|      | 22/45 [02:19<02:27,  6.40s/it]
```

```
51%|      | 23/45 [02:25<02:21,  6.43s/it]
```

```
53%|     | 24/45 [02:32<02:14,  6.42s/it]
```

```
56%|     | 25/45 [02:38<02:08,  6.41s/it]
```

```
58%|     | 26/45 [02:45<02:02,  6.43s/it]
```

```
60%|     | 27/45 [02:51<01:55,  6.43s/it]
```

```
62%|    | 28/45 [02:57<01:49,  6.42s/it]
```

```
64%|    | 29/45 [03:04<01:42,  6.41s/it]
```

```
67%|    | 30/45 [03:10<01:36,  6.40s/it]
```

```
69%|    | 31/45 [03:17<01:29,  6.41s/it]

71%|    | 32/45 [03:23<01:23,  6.41s/it]

73%|  | 33/45 [03:30<01:16,  6.41s/it]

76%|  | 34/45 [03:36<01:10,  6.41s/it]

78%|  | 35/45 [03:42<01:04,  6.40s/it]

80%|  | 36/45 [03:49<00:57,  6.41s/it]
```

```
82%| | 37/45 [03:55<00:51,  6.42s/it]
```

```
84%| | 38/45 [04:02<00:44,  6.42s/it]
```

```
87%| | 39/45 [04:08<00:38,  6.41s/it]
```

```
89%| | 40/45 [04:14<00:32,  6.42s/it]
```

```
91%| | 41/45 [04:21<00:25,  6.44s/it]
```

```
93%|| 42/45 [04:27<00:19,  6.45s/it]
```

```
 96%|| 43/45 [04:34<00:13,  6.50s/it]
```

```
 98%|| 44/45 [04:41<00:06,  6.54s/it]
```

```
100%|| 45/45 [04:47<00:00,  6.52s/it]
```

```
  0%|          | 0/45 [00:00<?, ?it/s]
```

```
  2%|         | 1/45 [00:06<04:36,  6.29s/it]
```

```
  4%|          | 2/45 [00:12<04:26,  6.20s/it]

  7%|          | 3/45 [00:18<04:22,  6.24s/it]

  9%|          | 4/45 [00:24<04:14,  6.20s/it]

 11%|          | 5/45 [00:30<04:07,  6.18s/it]

 13%|        | 6/45 [00:36<04:00,  6.16s/it]

 16%|        | 7/45 [00:43<03:53,  6.15s/it]
```

```
18%|          | 8/45 [00:49<03:51,  6.25s/it]
```

```
20%|          | 9/45 [00:55<03:45,  6.26s/it]
```

```
22%|          | 10/45 [01:02<03:39,  6.26s/it]
```

```
24%|          | 11/45 [01:08<03:33,  6.27s/it]
```

```
27%|          | 12/45 [01:14<03:27,  6.29s/it]
```

```
29%|          | 13/45 [01:21<03:23,  6.35s/it]
```

31%| | 14/45 [01:27<03:17, 6.36s/it]

33%| | 15/45 [01:34<03:10, 6.37s/it]

36%| | 16/45 [01:40<03:04, 6.35s/it]

38%| | 17/45 [01:46<02:57, 6.36s/it]

40%| | 18/45 [01:52<02:51, 6.34s/it]

42%| | 19/45 [01:59<02:45, 6.35s/it]

```
44%|     | 20/45 [02:05<02:38,  6.36s/it]
```

```
47%|     | 21/45 [02:12<02:32,  6.37s/it]
```

```
49%|     | 22/45 [02:18<02:25,  6.35s/it]
```

```
51%|     | 23/45 [02:25<02:21,  6.43s/it]
```

```
53%|    | 24/45 [02:31<02:15,  6.43s/it]
```

```
56%|    | 25/45 [02:37<02:08,  6.42s/it]
```

58%| | 26/45 [02:44<02:01, 6.42s/it]

60%| | 27/45 [02:50<01:56, 6.49s/it]

62%| | 28/45 [02:57<01:49, 6.46s/it]

64%| | 29/45 [03:03<01:43, 6.46s/it]

67%| | 30/45 [03:10<01:36, 6.45s/it]

69%| | 31/45 [03:16<01:30, 6.47s/it]

```
71%|   | 32/45 [03:24<01:27,  6.75s/it]
```

```
73%|  | 33/45 [03:30<01:19,  6.65s/it]
```

```
76%|  | 34/45 [03:36<01:12,  6.58s/it]
```

```
78%|  | 35/45 [03:43<01:05,  6.52s/it]
```

```
80%|  | 36/45 [03:50<00:58,  6.55s/it]
```

```
82%| | 37/45 [03:56<00:52,  6.60s/it]
```

```
84%| | 38/45 [04:03<00:45,  6.55s/it]
```

```
87%| | 39/45 [04:09<00:39,  6.53s/it]
```

```
89%| | 40/45 [04:16<00:32,  6.53s/it]
```

```
91%| | 41/45 [04:23<00:26,  6.66s/it]
```

```
93%|| 42/45 [04:29<00:20,  6.67s/it]
```

```
96%|| 43/45 [04:36<00:13,  6.68s/it]
```

```
 98%|| 44/45 [04:43<00:06,  6.67s/it]
```

```
100%|| 45/45 [04:50<00:00,  6.74s/it]
```

```
  0%|          | 0/45 [00:00<?, ?it/s]
```

```
  2%|         | 1/45 [00:06<04:27,  6.09s/it]
```

```
  4%|         | 2/45 [00:11<04:19,  6.04s/it]
```

```
  7%|          | 3/45 [00:17<04:12,  6.02s/it]

  9%|          | 4/45 [00:24<04:07,  6.04s/it]

 11%|          | 5/45 [00:30<04:08,  6.21s/it]

 13%|        | 6/45 [00:36<04:01,  6.19s/it]

 16%|        | 7/45 [00:43<03:55,  6.19s/it]

 18%|        | 8/45 [00:49<03:49,  6.20s/it]
```

```
20%|          | 9/45 [00:55<03:43,  6.21s/it]

22%|          | 10/45 [01:02<03:41,  6.32s/it]

24%|          | 11/45 [01:08<03:36,  6.37s/it]

27%|          | 12/45 [01:15<03:31,  6.41s/it]

29%|          | 13/45 [01:21<03:25,  6.44s/it]

31%|          | 14/45 [01:28<03:20,  6.45s/it]
```

```
33%|      | 15/45 [01:34<03:17,  6.58s/it]
```

```
36%|      | 16/45 [01:41<03:10,  6.56s/it]
```

```
38%|      | 17/45 [01:47<03:03,  6.56s/it]
```

```
40%|      | 18/45 [01:54<02:57,  6.56s/it]
```

```
42%|     | 19/45 [02:01<02:51,  6.58s/it]
```

```
44%|     | 20/45 [02:07<02:42,  6.51s/it]
```

```
47%|     | 21/45 [02:13<02:35,  6.47s/it]
```

```
49%|     | 22/45 [02:20<02:28,  6.44s/it]
```

```
51%|     | 23/45 [02:26<02:21,  6.41s/it]
```

```
53%|    | 24/45 [02:33<02:17,  6.55s/it]
```

```
56%|    | 25/45 [02:39<02:10,  6.53s/it]
```

```
58%|    | 26/45 [02:46<02:03,  6.49s/it]
```

```
60%|    | 27/45 [02:52<01:56,  6.45s/it]
```

```
62%|   | 28/45 [03:01<02:03,  7.29s/it]
```

```
64%|   | 29/45 [03:10<02:02,  7.65s/it]
```

```
67%|   | 30/45 [03:17<01:51,  7.43s/it]
```

```
69%|   | 31/45 [03:25<01:49,  7.79s/it]
```

```
71%|   | 32/45 [03:33<01:40,  7.69s/it]
```

```
73%|  | 33/45 [03:40<01:29,  7.49s/it]
```

```
76%|  | 34/45 [03:47<01:19,  7.20s/it]
```

```
78%|  | 35/45 [03:53<01:09,  6.97s/it]
```

```
80%|  | 36/45 [04:00<01:01,  6.88s/it]
```

```
82%| | 37/45 [04:06<00:54,  6.81s/it]
```

```
84%| | 38/45 [04:13<00:47,  6.76s/it]
```

```
87%| | 39/45 [04:20<00:40,  6.72s/it]
```

```
89%| | 40/45 [04:27<00:34,  6.82s/it]
```

```
91%| | 41/45 [04:34<00:27,  6.86s/it]
```

```
93%|| 42/45 [04:40<00:20,  6.74s/it]
```

```
96%|| 43/45 [04:47<00:13,  6.68s/it]
```

```
98%|| 44/45 [04:53<00:06,  6.61s/it]
```

```
100%|| 45/45 [05:00<00:00,  6.59s/it]
```

```
  0%|          | 0/45 [00:00<?, ?it/s]
```

```
  2%|         | 1/45 [00:05<04:14,  5.79s/it]
```

```
  4%|         | 2/45 [00:11<04:09,  5.81s/it]
```

```
  7%|         | 3/45 [00:17<04:05,  5.85s/it]
```

```
  9%|          | 4/45 [00:23<04:02,  5.91s/it]

 11%|          | 5/45 [00:29<04:00,  6.01s/it]

 13%|         | 6/45 [00:36<03:57,  6.09s/it]

 16%|         | 7/45 [00:42<03:51,  6.08s/it]

 18%|         | 8/45 [00:48<03:46,  6.11s/it]

 20%|         | 9/45 [00:54<03:40,  6.14s/it]
```

```
22%|          | 10/45 [01:01<03:37,  6.23s/it]
```

```
24%|          | 11/45 [01:07<03:32,  6.25s/it]
```

```
27%|          | 12/45 [01:13<03:26,  6.27s/it]
```

```
29%|          | 13/45 [01:19<03:21,  6.29s/it]
```

```
31%|          | 14/45 [01:26<03:15,  6.30s/it]
```

```
33%|         | 15/45 [01:32<03:12,  6.41s/it]
```

```
36%|      | 16/45 [01:39<03:04,  6.37s/it]

38%|      | 17/45 [01:45<02:57,  6.36s/it]

40%|      | 18/45 [01:51<02:51,  6.33s/it]

42%|     | 19/45 [01:58<02:44,  6.33s/it]

44%|     | 20/45 [02:04<02:39,  6.37s/it]

47%|     | 21/45 [02:11<02:33,  6.38s/it]
```

```
49%|     | 22/45 [02:17<02:27,  6.39s/it]
```

```
51%|     | 23/45 [02:24<02:26,  6.66s/it]
```

```
53%|    | 24/45 [02:32<02:26,  7.00s/it]
```

```
56%|    | 25/45 [02:40<02:23,  7.17s/it]
```

```
58%|    | 26/45 [02:47<02:17,  7.22s/it]
```

```
60%|    | 27/45 [02:54<02:07,  7.10s/it]
```

```
62%|    | 28/45 [03:01<02:02,  7.19s/it]
```

```
64%|    | 29/45 [03:08<01:53,  7.07s/it]
```

```
67%|    | 30/45 [03:15<01:44,  6.96s/it]
```

```
69%|    | 31/45 [03:22<01:37,  6.93s/it]
```

```
71%|    | 32/45 [03:29<01:30,  6.96s/it]
```

```
73%|  | 33/45 [03:35<01:22,  6.89s/it]
```

76%| | 34/45 [03:42<01:15, 6.88s/it]

78%| | 35/45 [03:49<01:07, 6.76s/it]

80%| | 36/45 [03:55<01:00, 6.77s/it]

82%| | 37/45 [04:03<00:55, 6.91s/it]

84%| | 38/45 [04:09<00:47, 6.78s/it]

87%| | 39/45 [04:16<00:40, 6.69s/it]

```
 89%| | 40/45 [04:22<00:33,  6.64s/it]
```

```
 91%| | 41/45 [04:29<00:26,  6.59s/it]
```

```
 93%|| 42/45 [04:36<00:20,  6.71s/it]
```

```
 96%|| 43/45 [04:42<00:13,  6.62s/it]
```

```
 98%|| 44/45 [04:48<00:06,  6.56s/it]
```

```
100%|| 45/45 [04:55<00:00,  6.54s/it]
```

```
0%|          | 0/45 [00:00<?, ?it/s]

2%|          | 1/45 [00:05<04:17,  5.85s/it]

4%|          | 2/45 [00:11<04:11,  5.85s/it]

7%|          | 3/45 [00:17<04:07,  5.89s/it]

9%|          | 4/45 [00:23<04:02,  5.92s/it]
```

```
11%|          | 5/45 [00:29<03:58,  5.96s/it]

13%|         | 6/45 [00:36<03:59,  6.14s/it]

16%|         | 7/45 [00:42<03:52,  6.12s/it]

18%|         | 8/45 [00:48<03:46,  6.11s/it]

20%|         | 9/45 [00:54<03:41,  6.14s/it]

22%|        | 10/45 [01:00<03:35,  6.17s/it]
```

```
24%|          | 11/45 [01:08<03:40,  6.48s/it]
```

```
27%|          | 12/45 [01:15<03:46,  6.87s/it]
```

```
29%|          | 13/45 [01:22<03:39,  6.85s/it]
```

```
31%|          | 14/45 [01:29<03:28,  6.74s/it]
```

```
33%|         | 15/45 [01:36<03:23,  6.78s/it]
```

```
36%|         | 16/45 [01:42<03:13,  6.67s/it]
```

```
38%|      | 17/45 [01:49<03:09,  6.76s/it]
```

```
40%|      | 18/45 [01:56<03:08,  6.99s/it]
```

```
42%|     | 19/45 [02:03<02:59,  6.91s/it]
```

```
44%|     | 20/45 [02:10<02:49,  6.77s/it]
```

```
47%|     | 21/45 [02:17<02:48,  7.01s/it]
```

```
49%|     | 22/45 [02:24<02:38,  6.89s/it]
```

```
51%|      | 23/45 [02:30<02:28,  6.74s/it]
```

```
53%|     | 24/45 [02:37<02:22,  6.80s/it]
```

```
56%|     | 25/45 [02:44<02:16,  6.84s/it]
```

```
58%|     | 26/45 [02:51<02:08,  6.77s/it]
```

```
60%|     | 27/45 [02:57<02:00,  6.72s/it]
```

```
62%|    | 28/45 [03:04<01:53,  6.71s/it]
```

```
64%|   | 29/45 [03:11<01:47,  6.73s/it]

67%|   | 30/45 [03:18<01:41,  6.75s/it]

69%|   | 31/45 [03:24<01:34,  6.74s/it]

71%|   | 32/45 [03:31<01:27,  6.73s/it]

73%|  | 33/45 [03:38<01:23,  6.94s/it]

76%|  | 34/45 [03:45<01:16,  6.93s/it]
```

```
78%|  | 35/45 [03:52<01:09,  6.95s/it]

80%|  | 36/45 [03:59<01:02,  6.97s/it]

82%| | 37/45 [04:07<00:57,  7.19s/it]

84%| | 38/45 [04:14<00:49,  7.01s/it]

87%| | 39/45 [04:21<00:42,  7.13s/it]

89%| | 40/45 [04:28<00:36,  7.23s/it]
```

```
 91%| | 41/45 [04:36<00:29,  7.27s/it]
```

```
 93%|| 42/45 [04:42<00:21,  7.08s/it]
```

```
 96%|| 43/45 [04:49<00:13,  6.98s/it]
```

```
 98%|| 44/45 [04:56<00:07,  7.01s/it]
```

```
100%|| 45/45 [05:04<00:00,  7.11s/it]
```

```
  0%|          | 0/45 [00:00<?, ?it/s]

  2%|         | 1/45 [00:06<04:34,  6.24s/it]

  4%|         | 2/45 [00:11<04:21,  6.09s/it]

  7%|         | 3/45 [00:18<04:18,  6.15s/it]

  9%|         | 4/45 [00:25<04:22,  6.41s/it]

 11%|         | 5/45 [00:32<04:20,  6.50s/it]
```

```
13%|          | 6/45 [00:39<04:22,  6.74s/it]

16%|          | 7/45 [00:46<04:17,  6.78s/it]

18%|          | 8/45 [00:52<04:04,  6.61s/it]

20%|          | 9/45 [00:58<03:55,  6.54s/it]

22%|          | 10/45 [01:05<03:49,  6.55s/it]

24%|          | 11/45 [01:11<03:40,  6.47s/it]
```

```
27%|        | 12/45 [01:17<03:32,  6.43s/it]

29%|        | 13/45 [01:24<03:27,  6.48s/it]

31%|        | 14/45 [01:31<03:23,  6.57s/it]

33%|       | 15/45 [01:38<03:20,  6.68s/it]

36%|       | 16/45 [01:45<03:16,  6.79s/it]

38%|       | 17/45 [01:52<03:14,  6.94s/it]
```

```
40%|          | 18/45 [01:59<03:06,  6.91s/it]

42%|          | 19/45 [02:06<03:03,  7.06s/it]

44%|          | 20/45 [02:14<02:59,  7.19s/it]

47%|          | 21/45 [02:21<02:54,  7.29s/it]

49%|          | 22/45 [02:29<02:50,  7.40s/it]

51%|          | 23/45 [02:37<02:45,  7.52s/it]
```

```
53%|     | 24/45 [02:44<02:33,  7.31s/it]
```

```
56%|     | 25/45 [02:50<02:23,  7.17s/it]
```

```
58%|     | 26/45 [02:57<02:14,  7.09s/it]
```

```
60%|     | 27/45 [03:05<02:12,  7.36s/it]
```

```
62%|    | 28/45 [03:12<02:02,  7.21s/it]
```

```
64%|    | 29/45 [03:19<01:54,  7.14s/it]
```

```
67%|    | 30/45 [03:26<01:45,  7.02s/it]
```

```
69%|    | 31/45 [03:34<01:42,  7.34s/it]
```

```
71%|    | 32/45 [03:42<01:39,  7.65s/it]
```

```
73%|  | 33/45 [03:50<01:30,  7.55s/it]
```

```
76%|  | 34/45 [03:57<01:22,  7.50s/it]
```

```
78%|  | 35/45 [04:04<01:14,  7.46s/it]
```

80%| | 36/45 [04:12<01:05, 7.33s/it]

82%| | 37/45 [04:18<00:57, 7.16s/it]

84%| | 38/45 [04:26<00:50, 7.24s/it]

87%| | 39/45 [04:36<00:49, 8.18s/it]

89%| | 40/45 [04:43<00:38, 7.79s/it]

91%| | 41/45 [04:50<00:30, 7.55s/it]

```
 93%|| 42/45 [04:57<00:21,  7.31s/it]
```

```
 96%|| 43/45 [05:04<00:14,  7.37s/it]
```

```
 98%|| 44/45 [05:11<00:07,  7.16s/it]
```

```
100%|| 45/45 [05:17<00:00,  6.99s/it]
```

```
  0%|          | 0/45 [00:00<?, ?it/s]
```

```
  2%|          | 1/45 [00:05<04:11,  5.71s/it]

  4%|          | 2/45 [00:11<04:04,  5.69s/it]

  7%|          | 3/45 [00:17<04:05,  5.84s/it]

  9%|          | 4/45 [00:23<04:03,  5.95s/it]

 11%|          | 5/45 [00:30<04:04,  6.11s/it]

 13%|        | 6/45 [00:36<04:05,  6.29s/it]
```

```
16%|          | 7/45 [00:43<04:02,  6.39s/it]

18%|          | 8/45 [00:50<03:59,  6.48s/it]

20%|          | 9/45 [00:57<04:04,  6.80s/it]

22%|          | 10/45 [01:04<03:55,  6.73s/it]

24%|          | 11/45 [01:10<03:47,  6.70s/it]

27%|          | 12/45 [01:18<03:45,  6.83s/it]
```

```
29%|        | 13/45 [01:25<03:43,  6.99s/it]
```

```
31%|        | 14/45 [01:32<03:35,  6.94s/it]
```

```
33%|       | 15/45 [01:39<03:29,  6.99s/it]
```

```
36%|       | 16/45 [01:46<03:24,  7.07s/it]
```

```
38%|       | 17/45 [01:53<03:15,  6.98s/it]
```

```
40%|       | 18/45 [02:00<03:09,  7.02s/it]
```

```
42%|      | 19/45 [02:07<03:00,  6.94s/it]
```

```
44%|      | 20/45 [02:14<02:54,  6.98s/it]
```

```
47%|      | 21/45 [02:21<02:45,  6.90s/it]
```

```
49%|      | 22/45 [02:28<02:38,  6.90s/it]
```

```
51%|      | 23/45 [02:34<02:29,  6.79s/it]
```

```
53%|     | 24/45 [02:41<02:21,  6.76s/it]
```

```
56%|     | 25/45 [02:48<02:18,  6.92s/it]

58%|     | 26/45 [02:55<02:10,  6.85s/it]

60%|     | 27/45 [03:01<02:02,  6.83s/it]

62%|    | 28/45 [03:08<01:57,  6.88s/it]

64%|    | 29/45 [03:16<01:51,  6.97s/it]

67%|    | 30/45 [03:22<01:43,  6.89s/it]
```

```
69%|   | 31/45 [03:29<01:35,  6.85s/it]
```

```
71%|   | 32/45 [03:36<01:28,  6.84s/it]
```

```
73%|  | 33/45 [03:43<01:21,  6.77s/it]
```

```
76%|  | 34/45 [03:49<01:14,  6.81s/it]
```

```
78%|  | 35/45 [03:57<01:10,  7.03s/it]
```

```
80%|  | 36/45 [04:04<01:02,  6.95s/it]
```

```
82%| | 37/45 [04:11<00:55,  6.91s/it]

84%| | 38/45 [04:18<00:49,  7.06s/it]

87%| | 39/45 [04:25<00:41,  6.99s/it]

89%| | 40/45 [04:32<00:34,  6.91s/it]

91%| | 41/45 [04:38<00:27,  6.88s/it]

93%|| 42/45 [04:46<00:20,  6.99s/it]
```

```
 96%|| 43/45 [04:52<00:13,  6.89s/it]

 98%|| 44/45 [04:59<00:06,  6.97s/it]

100%|| 45/45 [05:06<00:00,  6.94s/it]

  0%|          | 0/45 [00:00<?, ?it/s]

  2%|         | 1/45 [00:05<04:20,  5.93s/it]
```

```
  4%|          | 2/45 [00:11<04:14,  5.91s/it]
```

```
  7%|          | 3/45 [00:17<04:08,  5.93s/it]
```

```
  9%|          | 4/45 [00:23<04:04,  5.96s/it]
```

```
 11%|          | 5/45 [00:30<04:06,  6.17s/it]
```

```
 13%|         | 6/45 [00:37<04:10,  6.43s/it]
```

```
 16%|         | 7/45 [00:45<04:17,  6.77s/it]
```

```
18%|          | 8/45 [00:51<04:07,  6.70s/it]

20%|          | 9/45 [00:58<04:03,  6.76s/it]

22%|          | 10/45 [01:05<03:58,  6.82s/it]

24%|          | 11/45 [01:12<03:57,  7.00s/it]

27%|          | 12/45 [01:19<03:51,  7.02s/it]

29%|          | 13/45 [01:27<03:49,  7.18s/it]
```

```
31%|        | 14/45 [01:34<03:39,  7.09s/it]
```

```
33%|       | 15/45 [01:41<03:30,  7.03s/it]
```

```
36%|       | 16/45 [01:47<03:20,  6.92s/it]
```

```
38%|       | 17/45 [01:54<03:14,  6.94s/it]
```

```
40%|       | 18/45 [02:01<03:07,  6.93s/it]
```

```
42%|     | 19/45 [02:08<02:57,  6.82s/it]
```

```
44%|      | 20/45 [02:15<02:53,  6.94s/it]
```

```
47%|      | 21/45 [02:22<02:43,  6.80s/it]
```

```
49%|      | 22/45 [02:29<02:40,  7.00s/it]
```

```
51%|      | 23/45 [02:36<02:32,  6.91s/it]
```

```
53%|    | 24/45 [02:42<02:21,  6.76s/it]
```

```
56%|    | 25/45 [02:49<02:15,  6.76s/it]
```

```
58%|     | 26/45 [02:56<02:08,  6.77s/it]

60%|     | 27/45 [03:03<02:02,  6.83s/it]

62%|    | 28/45 [03:10<01:56,  6.88s/it]

64%|    | 29/45 [03:17<01:50,  6.93s/it]

67%|    | 30/45 [03:24<01:44,  6.96s/it]

69%|    | 31/45 [03:32<01:42,  7.29s/it]
```

```
71%|   | 32/45 [03:38<01:32,  7.09s/it]
```

```
73%|  | 33/45 [03:45<01:23,  6.96s/it]
```

```
76%|  | 34/45 [03:52<01:15,  6.85s/it]
```

```
78%|  | 35/45 [03:58<01:07,  6.78s/it]
```

```
80%|  | 36/45 [04:05<01:00,  6.76s/it]
```

```
82%| | 37/45 [04:12<00:53,  6.67s/it]
```

```
84%| | 38/45 [04:18<00:46,  6.61s/it]
```

```
87%| | 39/45 [04:25<00:39,  6.62s/it]
```

```
89%| | 40/45 [04:31<00:33,  6.65s/it]
```

```
91%| | 41/45 [04:38<00:26,  6.60s/it]
```

```
93%|| 42/45 [04:44<00:19,  6.57s/it]
```

```
96%|| 43/45 [04:51<00:13,  6.55s/it]
```

```
 98%|| 44/45 [04:57<00:06,  6.52s/it]
```

```
100%|| 45/45 [05:04<00:00,  6.55s/it]
```

```
In [109]: # plot the results
          plt.figure()
          plt.plot(my_range, one, two)
          plt.plot(my_range, three, four)
          plt.plot(my_range, five, six)
          plt.plot(my_range, seven, eight)
          plt.plot(my_range, nine, ten)
          plt.xlabel('Number of Epsilon Increase')
          plt.ylabel('Silhouette Coefficient')
          plt.grid(True)
          plt.show()
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.HTML object>
```

1.2 2. Clustering your own data

Using your own data, find relevant clusters/groups within your data. If your data is labeled already, with a class that you are attempting to predict, be sure to not use it in fitting/training/predicting.

You may use the labels to compare with predictions to show how well the clustering performed using one of the clustering metrics (http://scikit-learn.org/stable/modules/clustering.html#clustering-performance-evaluation).

If you don't have labels, use the silhouette coefficient to show performance. Find the optimal fit for your data but you don't need to be as exhaustive as above.

Additionally, show the clusters in 2D and 3D plots.

For bonus, try using PCA first to condense your data from N columns to less than N. Two items are expected: - Metric Evaluation Plot - Plots of the clustered data

In [142]:
```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib notebook
from mpl_toolkits.mplot3d import Axes3D
from sklearn import decomposition
from sklearn import datasets
import pandas as pd
import seaborn
plt.rcParams['font.size'] = 8
plt.rcParams['figure.figsize'] = (8.0, 7.0)

centers = [[1, 1], [-1, -1], [1, -1]]
wine = datasets.load_wine()
X = wine.data
y = wine.target

fig = plt.figure(0)
plt.clf()
ax = Axes3D(fig, rect=[0, 0, .95, 1], elev=48, azim=100)

plt.cla()

pca = decomposition.PCA(n_components=3)
pca.fit_transform
pca.fit(X)
X = pca.transform(X)

for name, label in [('class_0', 0), ('class_1', 1), ('class_2', 2)]:
    ax.text3D(X[y == label, 0].mean(),
              X[y == label, 1].mean() + 1.5,
              X[y == label, 2].mean(), name,
              horizontalalignment='center',
              bbox=dict(alpha=.5, edgecolor='w', facecolor='w'))
# Reorder the labels to have colors matching the cluster results
y = np.choose(y, [1, 2, 0]).astype(np.float)
ax.scatter(X[:, 0], X[:, 1], X[:, 2], c=y, cmap="Paired")

ax.w_xaxis.set_ticklabels([])
ax.w_yaxis.set_ticklabels([])
ax.w_zaxis.set_ticklabels([])
ax.set_xlabel('Magnesium')
ax.set_ylabel('Flavinoids')
ax.set_zlabel('Alcohol')

plt.show()
```

<IPython.core.display.Javascript object>

<IPython.core.display.HTML object>

In [143]:
```
centers = [[1, 1], [-1, -1], [1, -1]]
wine = datasets.load_iris()
X = wine.data
y = wine.target

fig = plt.figure(1)
plt.clf()
ax = Axes3D(fig, rect=[0, 0, .95, 1], elev=48, azim=140)

plt.cla()

for name, label in [('class_0', 0), ('class_1', 1), ('class_2', 2)]:
    ax.text3D(X[y == label, 0].mean(),
              X[y == label, 3].mean() + 1.5,
              X[y == label, 2].mean(), name,
              horizontalalignment='center',
              bbox=dict(alpha=.5, edgecolor='w', facecolor='w'))
# Reorder the labels to have colors matching the cluster results
y = np.choose(y, [1, 2, 0]).astype(np.float)
ax.scatter(X[:, 0], X[:, 3], X[:, 2], c=y, cmap="Paired")

ax.w_xaxis.set_ticklabels([])
ax.w_yaxis.set_ticklabels([])
ax.w_zaxis.set_ticklabels([])
ax.set_xlabel('Magnesium')
ax.set_ylabel('Flavinoids')
ax.set_zlabel('Alcohol')

plt.show()
```

<IPython.core.display.Javascript object>

<IPython.core.display.HTML object>

In [171]:
```
plt.rcParams['font.size'] = 12
plt.rcParams['figure.figsize'] = (8.0, 5.0)

X = pd.read_csv('../data/Credit.csv')
X = X.drop(["Unnamed: 0", "Gender", "Student", "Married", "Ethnicity"], axis = 1)
X.head()
```

```
#K-means with N clusters
N = 7
from sklearn.cluster import KMeans
km = KMeans(n_clusters=N, random_state=1)
km.fit(X)

# review the cluster labels
set(km.labels_)

X['cluster'] = km.predict(X)
X.cluster.value_counts()

fig = plt.figure()
plt.clf()
ax = Axes3D(fig, rect=[0, 0, .95, 1], elev=90, azim=270)

plt.cla()

ax.scatter(X['Income'], X['Rating'], X['Age'], c=X.cluster, s=5)

ax.set_xlabel('Income')
ax.set_ylabel('Rating')
ax.set_zlabel('Age')
plt.show()

# Scores are grouped in bands and by the increment of 100 until it reaches 600.
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.HTML object>
```

```
In [148]: plt.rcParams['figure.figsize'] = (8.0, 20.0)

          fig = plt.figure()
          plt.scatter(X.Income, X.Rating, c=X.cluster, s=5, cmap='Paired')

          plt.xlabel('Income')
          plt.ylabel('Rating')
          plt.show()

          # Expanded 2D view
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.HTML object>
```

In [172]:
```
plt.rcParams['figure.figsize'] = (8.0, 5.0)

fig = plt.figure()
plt.scatter(X.Income, X.Age, c=X.cluster, s=5, cmap='Paired')

plt.xlabel('Income')
plt.ylabel('Age')
plt.show()

# Age and Income, not so much related among the credit data population
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.HTML object>
```

In [150]:
```
fig = plt.figure()
plt.scatter(X.Rating, X.Age, c=X.cluster, s=5, cmap='Paired')

plt.xlabel('Rating')
plt.ylabel('Age')
plt.show()

# The same goes for Rating and Income
# What's more interesting in this plot is that there are hollow areas between the colo
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.HTML object>
```

In [173]:
```
XX = X.copy()
XX['Income'] = (X.Income - X.Income.mean())/X.Income.std()
XX['Rating'] = (X.Rating - X.Rating.mean())/X.Rating.std()
XX['Age'] = (X.Age - X.Age.mean())/X.Age.std()

km = KMeans(n_clusters=N, random_state=1)
XX['cluster'] = km.fit_predict(XX[['Age', 'Rating', 'Income']])

fig = plt.figure()
plt.scatter(XX.Age, XX.Income, c=XX.cluster, s=5, cmap='Paired')

plt.xlabel('Age')
plt.ylabel('Income')
plt.show()
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.HTML object>
```

```
In [174]: XX = X.copy()
          XX['Income'] = (X.Income - X.Income.mean())/X.Income.std()
          XX['Rating'] = (X.Rating - X.Rating.mean())/X.Rating.std()
          XX['Age'] = (X.Age - X.Age.mean())/X.Age.std()

          km = KMeans(n_clusters=N, random_state=1)
          XX['cluster'] = km.fit_predict(XX[['Age', 'Rating', 'Income']])

          fig = plt.figure()
          plt.scatter(XX.Rating, XX.Income, c=XX.cluster, s=5, cmap='Paired')

          plt.xlabel('Rating')
          plt.ylabel('Income')
          plt.show()
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.HTML object>
```

```
In [175]: XX = X.copy()
          XX['Income'] = (X.Income - X.Income.mean())/X.Income.std()
          XX['Rating'] = (X.Rating - X.Rating.mean())/X.Rating.std()
          XX['Age'] = (X.Age - X.Age.mean())/X.Age.std()

          km = KMeans(n_clusters=N, random_state=1)
          XX['cluster'] = km.fit_predict(XX[['Age', 'Rating', 'Income']])

          fig = plt.figure()
          plt.scatter(XX.Age, XX.Rating, c=XX.cluster, s=5, cmap='Paired')

          plt.xlabel('Age')
          plt.ylabel('Rating')
          plt.show()
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.HTML object>
```

```
In [176]: fig = plt.figure()
          plt.clf()
          ax = Axes3D(fig, rect=[0, 0, .95, 1], elev=48, azim=140)
```

```
plt.cla()

ax.scatter(XX['Age'], XX['Rating'], XX['Income'], c=XX.cluster, s=5)

ax.set_xlabel('Age')
ax.set_ylabel('Rating')
ax.set_zlabel('Income')
plt.show()
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.HTML object>
```

```
In [155]: fig = plt.figure()
          XX.Rating.hist(bins=1000)
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.HTML object>
```

```
Out[155]: <matplotlib.axes._subplots.AxesSubplot at 0x117d8e240>
```

```
In [156]: fig = plt.figure()
          plt.scatter(XX.Rating, XX.Income, alpha=.1, s=5, )
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.HTML object>
```

```
Out[156]: <matplotlib.collections.PathCollection at 0x1a27527048>
```

```
In [168]: from sklearn.cluster import DBSCAN
          dbscan = DBSCAN(eps=.12)
          XX.cluster = dbscan.fit_predict(XX[['Income','Rating', 'Age']])
          XX.cluster.value_counts()
```

```
Out[168]: -1    400
          Name: cluster, dtype: int64
```

```
In [158]: fig = plt.figure()
          plt.clf()
          ax = Axes3D(fig, rect=[0, 0, .95, 1], elev=48, azim=140)

          plt.cla()
```

```
        ax.scatter(XX['Income'], XX['Rating'], XX['Age'], c=XX.cluster, s=5, cmap='Paired')

        ax.set_xlabel('Income')
        ax.set_ylabel('Rating')
        ax.set_zlabel('Age')
        plt.show()
```

```
<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>
```

```
In [159]: fig = plt.figure()
          plt.scatter(XX.Income, XX.Rating, s=5, c=XX.cluster, cmap='Paired')
```

```
<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>
```

```
Out[159]: <matplotlib.collections.PathCollection at 0x1a2792c198>
```

```
In [160]: fig = plt.figure()
          plt.scatter(XX.Income, XX.Age, s=5, c=XX.cluster, cmap='Paired')
```

```
<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>
```

```
Out[160]: <matplotlib.collections.PathCollection at 0x1a28998780>
```

```
In [177]: # calculate SC for K=7
          from sklearn import metrics
          metrics.silhouette_score(XX[['Income', 'Rating', 'Age']], X.cluster)
```

```
Out[177]: 0.0736170561074945
```

```
In [178]: metrics.silhouette_score(XX[['Income', 'Rating', 'Age']], XX.cluster)
```

```
Out[178]: 0.3183053475041708
```

```
In [179]: from tqdm import tqdm

          # calculate SC for K=2 through K=19
          k_range = range(2, 40)
          scores = []
          for k in tqdm(k_range):
              km = KMeans(n_clusters=k, random_state=1)
              labels = km.fit_predict(XX[['Income', 'Rating', 'Age']])
              scores.append(metrics.silhouette_score(XX[['Income', 'Rating', 'Age']], labels))
```

```
  0%|          | 0/38 [00:00<?, ?it/s]
```

```
  8%|          | 3/38 [00:00<00:01, 28.77it/s]
```

```
 13%|        | 5/38 [00:00<00:01, 23.87it/s]
```

```
 18%|        | 7/38 [00:00<00:01, 20.89it/s]
```

```
 24%|       | 9/38 [00:00<00:01, 19.02it/s]
```

```
29%|         | 11/38 [00:00<00:01, 16.90it/s]
```

```
34%|        | 13/38 [00:00<00:01, 14.91it/s]
```

```
39%|        | 15/38 [00:00<00:01, 14.17it/s]
```

```
45%|      | 17/38 [00:01<00:01, 12.22it/s]
```

```
50%|      | 19/38 [00:01<00:01, 11.08it/s]
```

```
55%|    | 21/38 [00:01<00:01, 10.25it/s]
```

```
61%|    | 23/38 [00:01<00:01,  9.77it/s]
```

```
63%|   | 24/38 [00:01<00:01,  9.32it/s]
```

```
66%|   | 25/38 [00:02<00:01,  8.92it/s]
```

```
68%|   | 26/38 [00:02<00:01,  8.41it/s]
```

```
71%|   | 27/38 [00:02<00:01,  7.72it/s]
```

```
74%|  | 28/38 [00:02<00:01,  7.15it/s]
```

```
76%|  | 29/38 [00:02<00:01,  6.70it/s]

79%|  | 30/38 [00:02<00:01,  6.25it/s]

82%| | 31/38 [00:03<00:01,  6.10it/s]

84%| | 32/38 [00:03<00:00,  6.10it/s]

87%| | 33/38 [00:03<00:00,  6.04it/s]

89%| | 34/38 [00:03<00:00,  5.72it/s]
```

```
 92%|| 35/38 [00:03<00:00,  5.30it/s]
```

```
 95%|| 36/38 [00:04<00:00,  5.29it/s]
```

```
 97%|| 37/38 [00:04<00:00,  5.26it/s]
```

```
100%|| 38/38 [00:04<00:00,  5.24it/s]
```

In [180]:
```
# plot the results
plt.figure()
plt.plot(k_range, scores)
plt.xlabel('Number of clusters')
plt.ylabel('Silhouette Coefficient')
plt.grid(True)
plt.show()
```

```
<IPython.core.display.Javascript object>
```

```
<IPython.core.display.HTML object>
```

1.3 Note

You may use any for both parts 1 and 2, I only recommend using the data I used in the Lesson for part 1. I've included several new datasets in the `data/` folder, such as `beers.csv`, `snow_tweets.csv`, `data/USCensus1990.data.txt.gz`. You do not need to unzip or ungzip any data files. Pandas can open these files on its own.