Chapter 12 - Exercise 1: Shopping Data

Cho dữ liệu shopping_data.csv, thực hiện việc phân nhóm dữ liệu theo Hierarchical Clustering theo 2 thuộc tính là Annual Income (k$)và Spending Score (1-100)

- Đọc dữ liệu, chuẩn hóa dữ liệu nếu cần
- Dùng dendrogram để xác định số nhóm/cụm
- Áp dụng thuật toán
- Trực quan hóa kết quả, nhận xét
- *(Theo: http://stackabuse.com/hierarchical-clustering-with-python-and-scikit-learn/ (http://stackabuse.com/hierarchical-clustering-with-python-and-scikit-learn/))*

In [ ]:
```
import matplotlib.pyplot as plt
import pandas as pd
%matplotlib inline
import numpy as np
```

In [ ]:
```
customer_data = pd.read_csv('shopping_data.csv')
customer_data.shape
```

Out[2]:
```
(200, 5)
```

In [ ]:
```
customer_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 5 columns):
CustomerID                200 non-null int64
Genre                     200 non-null object
Age                       200 non-null int64
Annual Income (k$)        200 non-null int64
Spending Score (1-100)    200 non-null int64
dtypes: int64(4), object(1)
memory usage: 7.9+ KB
```

In [ ]:
```
customer_data.head()
```

Out[4]:

	CustomerID	Genre	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40

In [ ]:
```
data = customer_data.iloc[:, 3:5].values
```

In [ ]:
```
data[:5]
```

Out[6]:
```
array([[15, 39],
       [15, 81],
       [16,  6],
       [16, 77],
       [17, 40]], dtype=int64)
```

In [ ]:
```
from scipy.cluster import hierarchy
plt.figure(figsize=(10, 5))
plt.title("Customer Dendograms")
dend = hierarchy.dendrogram(hierarchy.linkage(data, method='ward'))
```

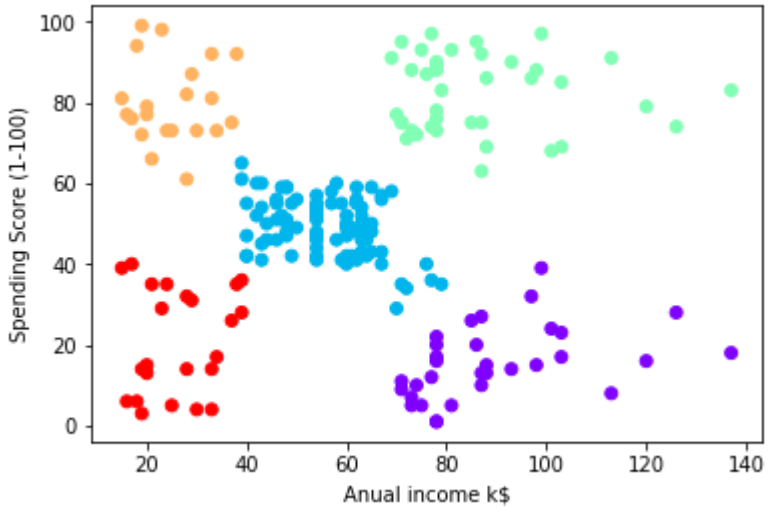
In [ ]:
```
# cluster = 5
from sklearn.cluster import AgglomerativeClustering
cluster = AgglomerativeClustering(n_clusters=5,
                                  affinity='euclidean', linkage='ward')
cluster.fit(data)
```

Out[8]:
```
AgglomerativeClustering(affinity='euclidean', compute_full_tree='auto',
            connectivity=None, linkage='ward', memory=None, n_clusters=5,
            pooling_func='deprecated')
```

In [ ]:
```
cluster.labels_
```

Out[9]:
```
array([4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3,
       4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 3, 4, 1,
       4, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 2, 1, 2, 0, 2, 0, 2,
       1, 2, 0, 2, 0, 2, 0, 2, 0, 2, 1, 2, 0, 2, 1, 2, 0, 2, 0, 2, 0, 2,
       0, 2, 0, 2, 0, 2, 1, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2,
       0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2, 0, 2,
       0, 2], dtype=int64)
```

In [ ]:
```
plt.scatter(data[:,0], data[:,1], c=cluster.labels_, cmap='rainbow')
plt.xlabel("Anual income k$")
plt.ylabel("Spending Score (1-100)")
plt.show()
```



Nhận xét: Ta có thể thấy các điểm dữ liệu tập trung vào 5 cụm.

- Các điểm dữ liệu ở góc dưới bên phải thuộc về khách hàng với mức lương cao nhưng chi tiêu thấp. Đây là những khách hàng chi tiêu tiền của họ một cách cẩn thận.
- Khách hàng ở trên cùng bên phải (dữ liệu màu xanh), đây là những khách hàng có mức lương cao và chi tiêu cao. Đây là loại khách hàng mà công ty nhắm mục tiêu.
- Các khách hàng ở giữa (dữ liệu xanh nước biển) là những khách hàng có mức lương trung bình và chi tiêu trung bình. Số lượng khách hàng nhiều nhất thuộc về nhóm này. Các công ty cũng có thể nhắm mục tiêu các khách hàng này với thực tế là họ đang có số lượng lớn, v.v.