Лабораторна робота №7

Просторова кластеризація

Мета роботи: набути практичних навичок роботи у побудові просторової кластеризації з використанням алгоритмів DBSCAN і KMeans

Література

K-Means Clustering in Python: A Practical Guide: https://realpython.com/k-means-clustering-python/#understanding-the-k-means-algorithm

The Ultimate Guide to K-Means Clustering: Definition, Methods and Applications:

sklearn.cluster.DBSCAN:

https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html

Зміст роботи

Завдання 1. Для набору даних eng-climate-summaries-All-2_2015.csv провести кластеризацію з використанням алгоритму DBSCAN.

DBSCAN — алгоритм кластеризації, який об'єднує точки, розташовані щільно одна до одної (наприклад, кілька найближчих сусідів). Викиди визначаються як окремі точки в регіонах з низькою щільністю.

Структура даних eng-climate-summaries-All-2_2015.
Stn_Name: Назва станції
Lat: широта (північ + , градуси)
Long: довгота (захід - , градуси)
Prov: Провінція
Tm: Середня температура (°C)
DwTm: Дні без середньої температури
D: Різниця середньої температури від нормальної (1981-2010) (°C)
Tx: Найвища максимальна місячна температура (°C)
DwTx: Дні без максимальної температури
Tn: Найнижча мінімальна температура за місяць (°C)
DwTn: Дні без мінімальної температури
S: Снігопад (см)
DwS: Дні без снігопаду
S%N: Відсоток від норми (1981-2010) Снігопад
P: Загальна кількість опадів (мм)
DwP: Дні без реальних опадів
P%N: Відсоток норми (1981-2010) опадів
S_G: Сніг на землі в кінці місяця (см)
Pd: Кількість днів з опадами 1,0 мм і більше
BS: Яскраве сонце (годин)
DwBS: Дні без яскравого сонячного світла
BS%: Відсоток від норми (1981-2010) Яскраве сонячне світло
HDD: Градус Дні нижче 18 °C
CDD: Градусні дні вище 18 °C
Stn_No: Ідентифікатор кліматичної станції (перші 3 цифри позначають водозбірний басейн, останні 4 символи для сортування за алфавітом).
NA: Не доступно

– Завдання рекомендується виконувати в Google Colaboratory.

– Бібліотеки, які знадобляться для роботи:

```
import numpy as np
import pandas as pd
import csv
from mpl_toolkits.basemap import Basemap
import matplotlib.pyplot as plt
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import DBSCAN, KMeans
import sklearn.utils
```

– Завантажити файл eng-climate-summaries-All-2_2015.csv і переглянути перших n рядків;

– Переглянути інформацію про набір даних. Провести чистку – вилучити всі нульові рядки;

– Провести просторову візуалізацію даних;

```
plt.figure(figsize=(14,10))

Long = [-140,-50] # Діапозон Довгота
Lat = [40,65] # Діапазон Широта
```

– Встановити діапазон Довгота/Широта;

```
df = df[(df['Long'] > Long[0]) & (df['Long'] < Long[1])
    & (df['Lat'] > Lat[0]) & (df['Lat'] < Lat[1])
```

– Створити базову карту;

```
my_map = Basemap(projection='merc', resolution='l', area_thresh = 1000.0,
    llcrnrlon = Long[0],
    urcrnrlon = Long[1],
    llcrnrlat = Lat[0],
    urcrnrlat = Lat[1]
)
```

– Вказати параметри малювання базової карти

```
my_map.drawcoastlines()
my_map.drawcountries()
my_map.drawmapboundary()
my_map.fillcontinents(color='grey',alpha=0.3)

my_map.shadedrelief()
```

– Отримати положення точок на карті за осями x і y за допомогою my_map;

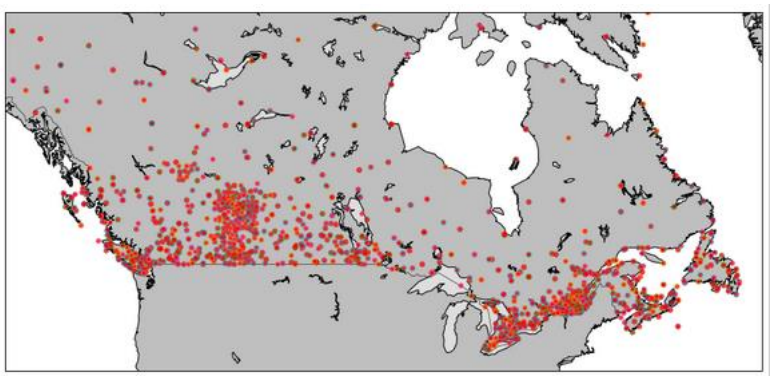
```
my_longs = df.Long.values
my_lats = df.Lat.values
```

```
X,Y = my_map(my_longs, my_lats)

df['xm'] = X
df['ym'] = Y
```

– Нанести на карту метеостанції;

```
for (x,y) in zip(X,Y):
    my_map.plot(x,y,
                markerfacecolor=([1,0,0]),
                marker = 'o',
                markersize = 5,
                alpha = 0.75)
```

Результат:



– Провести кластеризацію розташування. Згрупувати станції лише на основі їхнього розташування, тобто довготи та широти;

```
sklearn.utils.check_random_state(1000)
loc = [list(a) for a in zip(X,Y)]
loc = np.nan_to_num(loc)
scaler = StandardScaler()
loc = scaler.fit_transform(loc)
```

– Створити об’єкт DBSCAN;

```
db = DBSCAN(eps=0.15, min_samples=10)

db.fit(loc)
mask = np.zeros_like(db.labels_, dtype=bool)
mask[db.core_sample_indices_] = True
labels = db.labels_ + 1
```

– Додати мітки до фрейму даних і переглянути вибірку кластерів;

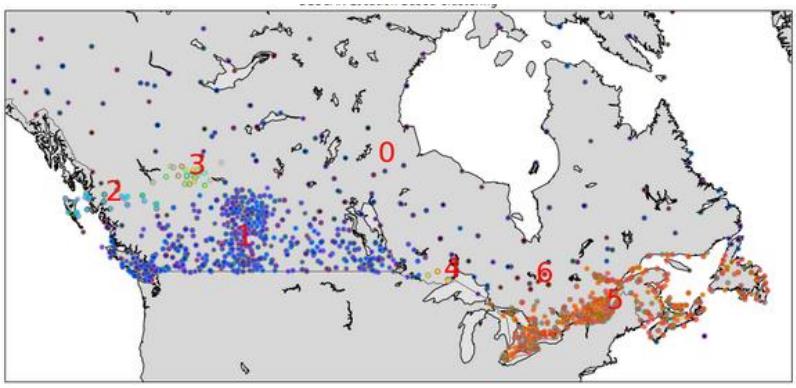
```
df['DBSCAN_L_Clusters'] = labels
df[["Stn_Name","Tx","Tm","DBSCAN_L_Clusters"]].head()
```

– Провести візуалізацію кластерів на карті. Для відображення кластерів на карті проведіть етапи створення базової карти з параметрами.

Далі потрібно додати точки, які будуть розфарбовані у різні кольори і поставити мітки кластерів;

```
colors = plt.get_cmap('jet')(np.linspace(0.0, 1.0, len(set(labels))))

for index,row in df.iterrows():
    my_map.plot(row.xm, row.ym,
                markerfacecolor=colors[row.DBSCAN_L_Clusters],
                marker = 'o',
                markersize = 5,
                alpha = 0.75
               )
# Мітки кластерів
for i in range(len(set(labels))):
    cluster = df[df.DBSCAN_L_Clusters == i][["Stn_Name","Tm","xm","ym","DB
SCAN_L_Clusters"]]
    xc = np.mean(cluster.xm)
    yc = np.mean(cluster.ym)

    #Отримайте середню температуру кластера
    Tavg = np.mean(cluster.Tm)

    #мітка кластера на карті
    plt.text(xc,yc,str(i),fontsize=30,color='red')

    # Вивести середні температури
    print ("Cluster "+str(i)+', Avg Temp: '+ str(np.mean(cluster.Tm)))
```

Результат:



– Провести дослідження роботи алгоритму DBSCAN.

Завдання 2. Для заданого набору реалізувати алгоритм кластеризації KMeans і одну з варіацій цього методу. Результати відобразити на карті. Порівняти результати.

Варіант	Метод

1, 7, 13, 19	K-Medoids
2, 8, 14, 20	K-Medians
3, 9, 15, 21	K-means++
4, 10, 16, 22	K-Modes
5, 11, 17, 23	Intelligent K-Means
6, 12, 18, 24	Genetic K-Means

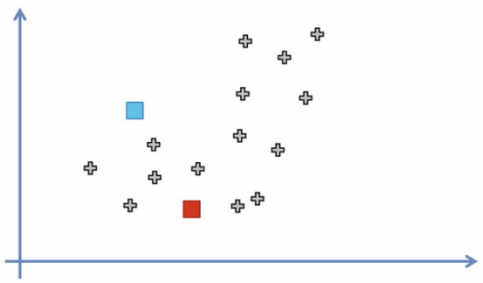
Методичні рекомендації

Кластеризація (або кластерний аналіз) - це задача розбиття множини об'єктів на групи, які називаються кластерами. Усередині кожної групи повинні виявитися «схожі» об'єкти, а об'єкти різних груп повинні бути якомога більш відмінні. Головна відмінність кластеризації від класифікації полягає в тому, що перелік груп чітко не заданий і визначається в процесі роботи алгоритму.

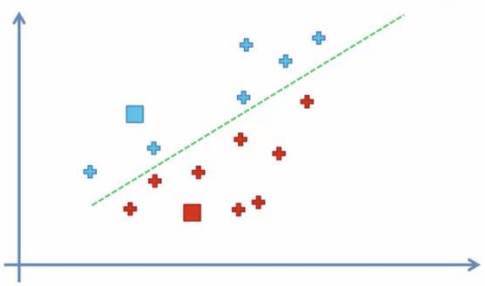
K- Means (MacQueen, 1967) - один з найпростіших алгоритмів навчання без контролю, який вирішує добре відому проблему кластеризації. Це простий спосіб класифікації заданого набору даних через певну кількість кластерів (наприклад, k). Основна ідея полягає в тому, щоб визначити k центроїдів, по одному для кожного кластера. Кращий вибір центроїда - розмістити їх якомога далі один від одного.

Алгоритм складається з наступних кроків:

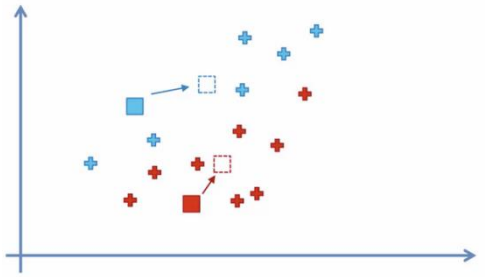
1. Виберіть K (на прикладі 2) випадкових точок як центри кластерів, що називаються *центроїдами*:



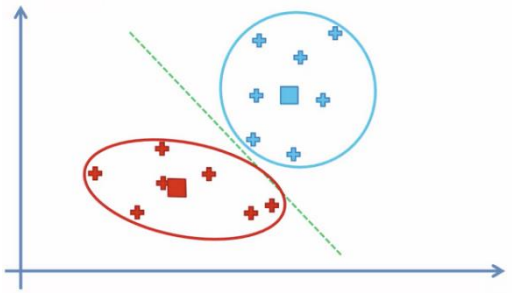
2. Згрупуйте точки даних до найближчого кластеру, обчисливши їх відстань відносно кожного центроїда:



3. Визначте новий центр кластера, обчисливши середнє значення точок:



4. Повторюйте кроки 2 і 3 до того моменту, поки не досягнете критерію збіжності:



Цей алгоритм спрямований на мінімізацію цільової функції, в даному випадку квадрат функції помилок. Цільова функція:

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^j - \mu_j \right\|^2,$$

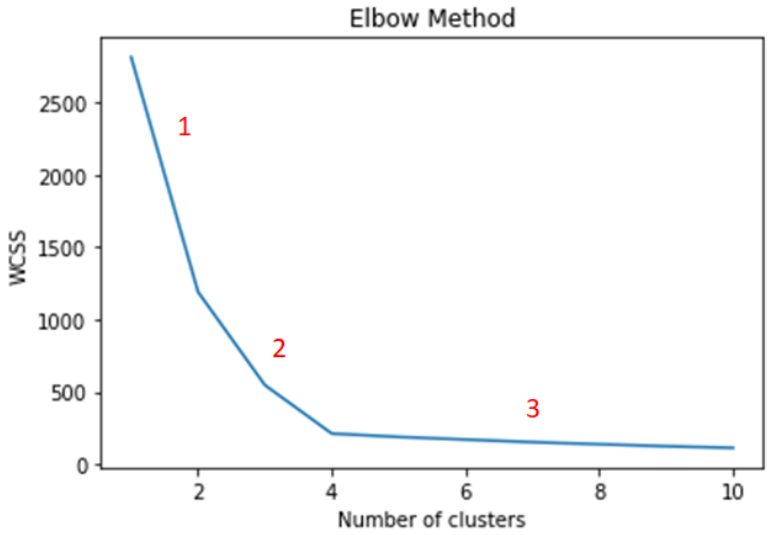
де $\left\| x_i^j - \mu_j \right\|^2$ обрана міра відстані між точкою даних x_i^j і центром кластера μ_j, є показником відстані n точок даних від їх відповідних центрів кластера.

Алгоритм k-Means не завжди знаходить найбільш оптимальну конфігурацію, відповідну мінімуму глобальної цільової функції. Алгоритм також істотно чутливий до початкового випадково вибраного кластерного центра.

Як вибрати потрібну кількість кластерів?

Часто дані матимуть декілька вимірів, що ускладнює їх візуалізацію і як наслідок, кількість кластерів вже не очевидна. Є спосіб визначити це математично.

Необхідно побудувати графік залежності між кількістю кластерів і сумою квадратів в кластері (Within Cluster Sum of Squares,WCSS), а потім вибрати кількість кластерів, в яких зміна рівня починає вирівнюватися (*метод ліктя*).



WCSS визначається як сума квадратів відстані між кожним членом кластера і його центроїдом.

$$WCSS = \sum_{i=1}^{n}\left(x_i^j - \mu_j\right)^2,$$

де μ_j - центр кластера, j – номер кластера

Контрольні запитання

1. В чому відмінність задач кластеризації від задач класифікації
2. У чому полягає суть кластеризації?
3. Дайте визначення поняття «кластер».
4. Опишіть роботу алгоритму DBSCAN.
5. В яких випадках слід використовувати алгоритм DBSCAN?
6. Опишіть роботу алгоритму K-means.
7. Алгоритм K-means є точним або наближеним? Чому?
8. Наведіть приклади функцій відстані між кластерами.
9. Як залежить результат кластеризації від початкових центрів? Як можна обійти цей недолік?
10. Як вибрати кількість кластерів?
11. Прокоментуйте переваги і недоліки методу кластеризації K-means.