

Міністерство освіти та науки України  
Львівський національний університет імені Івана Франка

# Звіт

про виконання лабораторної роботи №5

## **“Вивчення застосування кластеризації k-means для сенсорних даних ”**

Виконав:  
студент групи Фес-31  
Криницький Юрій  
Перевірив:

Сінкевич О.О.

Львів 2019

**Мета :** Вивчити та застосувати кластеризацію k-means для сенсорних даних.

## Теоретичні відомості :

Кластеризація методом k-середніх (англ. *k-means clustering*) — популярний метод кластеризації, — впорядкування множини об'єктів в порівняно однорідні групи. Винайдений в 1950-х роках математиком [Гуґо Штайнгаузом](#)<sup>[1]</sup> і майже одночасно Стюартом Ллойдом<sup>[2]</sup>. Особливу популярність отримав після виходу роботи МакКвіна<sup>[3]</sup>.

Мета методу — розділити  $n$  спостережень на  $k$  кластерів, так щоб кожне спостереження належало до кластера з найближчим до нього **середнім значенням**. Метод базується на мінімізації суми квадратів відстаней між кожним спостереженням та центром його кластера, тобто функції

$$\sum_{i=1}^N d(x_i, m_j(x_i))^2$$

де  $d$  — метрика,  $x_i$  —  $i$ -ий об'єкт даних, а  $m_j(x_i)$  — центр кластера, якому на  $j$ -ій ітерації приписаний елемент  $x_i$ .

## Опис алгоритму [\[ред. | ред. код\]](#)

Маємо масив спостережень (об'єктів), кожен з яких має певні значення по ряду ознак. Відповідно до цих значень об'єкт розташовується у багатовимірному просторі.

1. Дослідник визначає кількість кластерів, що необхідно утворити
2. Випадковим чином обирається  $k$  спостережень, які на цьому кроці вважаються центрами кластерів
3. Кожне спостереження «приписується» до одного з  $n$  кластерів — того, відстань до якого найкоротша
4. Розраховується новий центр кожного кластера як елемент, ознаки якого розраховуються як середнє арифметичне ознак об'єктів, що входять у цей кластер
5. Відбувається така кількість ітерацій (повторюються кроки 3-4), поки кластерні центри стануть стійкими (тобто при кожній ітерації в кожному кластері опинятимуться одні й ті

самі об'єкти), дисперсія всередині кластера буде мінімізована, а між кластерами — максимізована

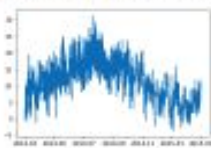
Вибір кількості кластерів відбувається на основі дослідницької гіпотези. Якщо її немає, то рекомендують створити 2 кластери, далі 3,4,5, порівнюючи отримані результати.

```
In [16]: from sklearn.cluster import KMeans, dbSCAN
import numpy as np
import matplotlib inline
import matplotlib.pyplot as plt
import pandas as pd

In [68]: # import temperature time series
df = pd.read_csv('out_temp.csv', usecols=['dateTime', 'data'])
display(df.columns)
# convert to datetime object
df['dateTime'] = pd.to_datetime(df['dateTime'])
# convert to pandas time series
ts = df.set_index('dateTime')
# plot results
ts = ts.loc['2014-12-01': '2014-12-31'] # розкomentуйте
plt.plot(ts)
```

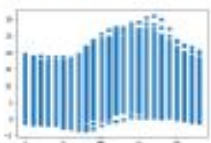
Index(['dateTime', 'data'], dtype='object')

Out[68]: <matplotlib.lines.Line2D at 0x7f170b057d10>



```
In [69]: # get hours from time series
ts = ts.resample('H').mean() # resampling by hour
X = np.array(ts.index.hour)
y = ts.values.flatten()
x = np.array(list(zip(X, y)))
```

Out[69]: <matplotlib.collections.PathCollection at 0x7f170b7eae0b>



```
In [70]: range n clusters = [2, 3, 4, 5]
for n_clusters in range(n_clusters):
    clust = KMeans(n_clusters=n_clusters, random_state=10)
    cluster_labels = clust.fit_predict(x) # запустити
    silhouette_avg = silhouette_score(x, cluster_labels)
    print(silhouette_avg)
```

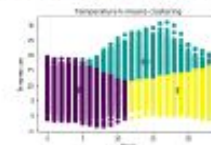
0.4214629864560592  
0.42173682214531133  
0.379264769104106  
0.36290977344039826

```
In [73]: # select best clusters: тут у нас 2 або 3
clust = KMeans(n_clusters=3, random_state=10)
cluster_labels = clust.fit_predict(x)
cluster_centers = clust.cluster_centers_
```

array([[ 4.7060274 , 8.64468950],  
 [13.92398012, 17.82982181],  
 [18.72168535, 8.42077513]])

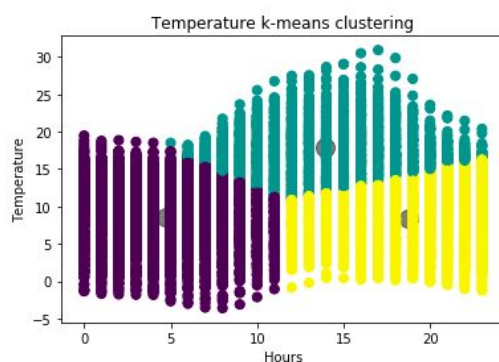
```
In [79]: fig = plt.figure()
ax = fig.add_subplot(111)
ax.set_xlabel('Hours')
ax.set_ylabel('Temperature')
ax.set_title('Temperature k-means clustering')
ax.scatter(cluster_centers[:, 0], cluster_centers[:, 1],
           ax.scatter(x[:, 0], x[:, 1], c=cluster_labels, s=50, cma
```

Out[79]: <matplotlib.collections.PathCollection at 0x7f170b584e50>



## Результат роботи :

Out[79]: <matplotlib.collections.PathCollection at 0x7f170b584e50>



## Висновок :

На цій лабораторній роботі я вивчив та застосував кластеризацію k-means для сенсорних даних.