МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ  
Національний технічний університет   
«Харківський політехнічний інститут»  
Кафедра ГМКГ

**Лабораторна работа №3**

З дисципліни «Інтелектуальний аналіз даних»

Виконав:

Студент групи ІКМ-220 г.

Ульянов Кирило Юрійович

Перевірив:

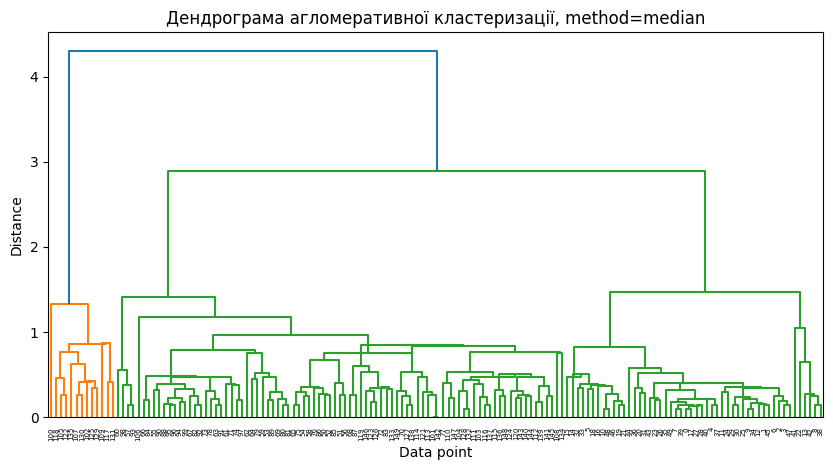
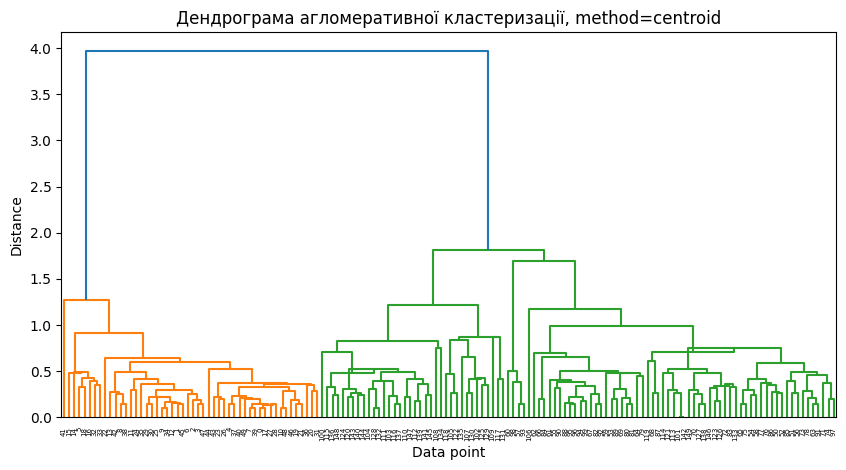
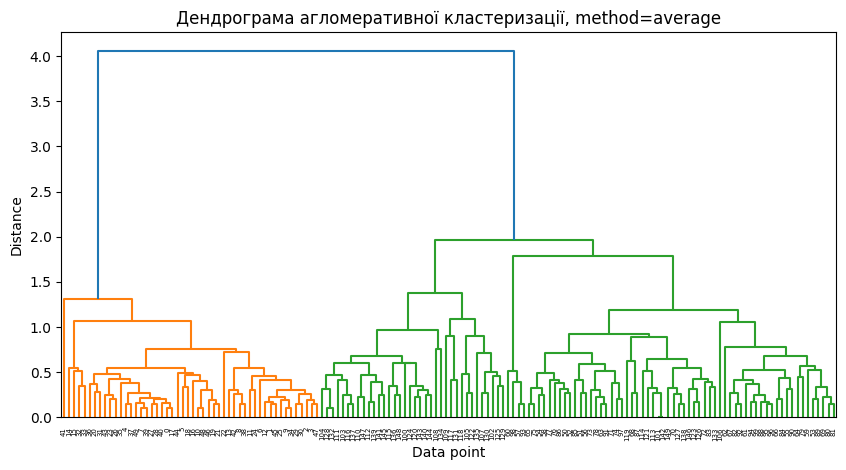
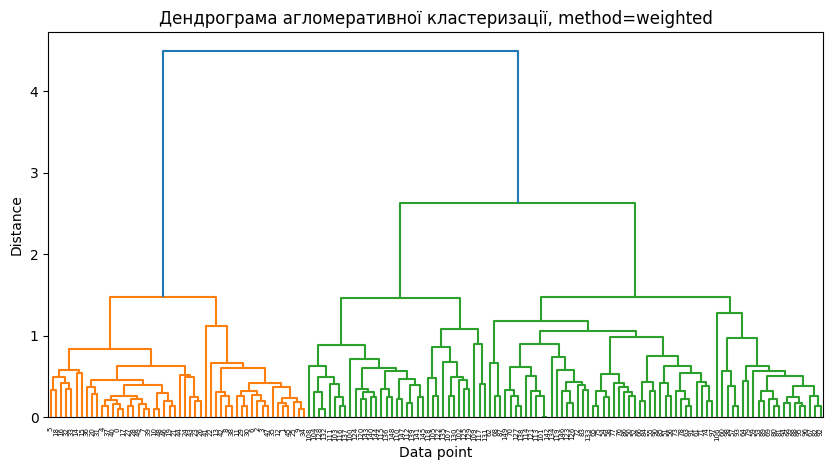
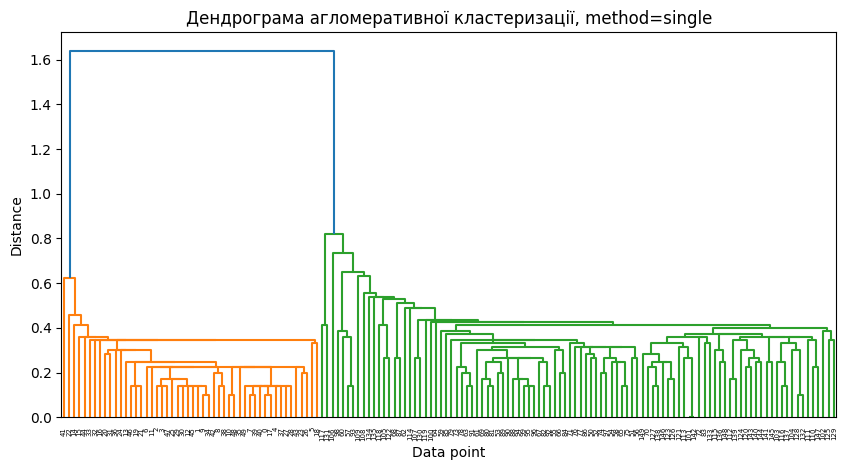
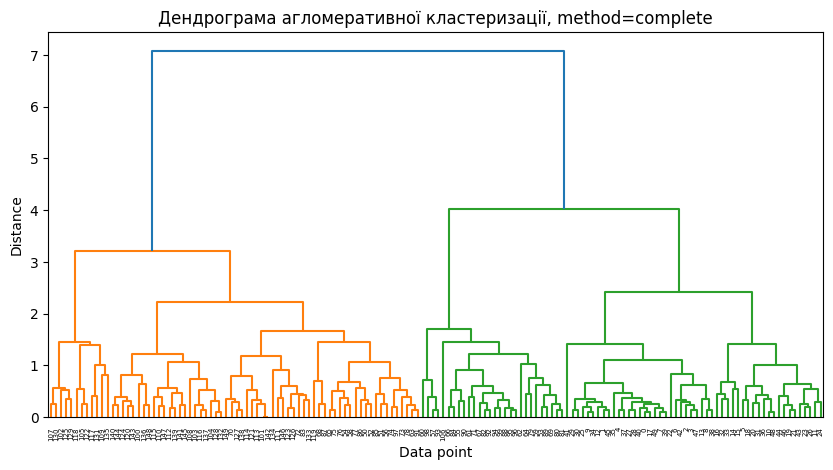
Доц. Дашкевич А.О.

Харків 2023

**Мета роботи:** вивчення базових алгоритмiв кластеризацiї агломеративного та центроїдного типу.

**Завдання на роботу:** завантаження набору даних, формування вхiдної вибiрки даних, кластеризацiя iз застосуванням алгоритмiв агломеративної та центроїдної (алгоритми k-means та Mean Shift) кластеризацiї, вiзуалiзацiя дендрограми за результатами агломеративної кластеризацiї.

Побудованi дендрограми для набору «Iриси Фiшера»



**Отримані точності:**

Точність на ненормалізованих даних: **0.77629**

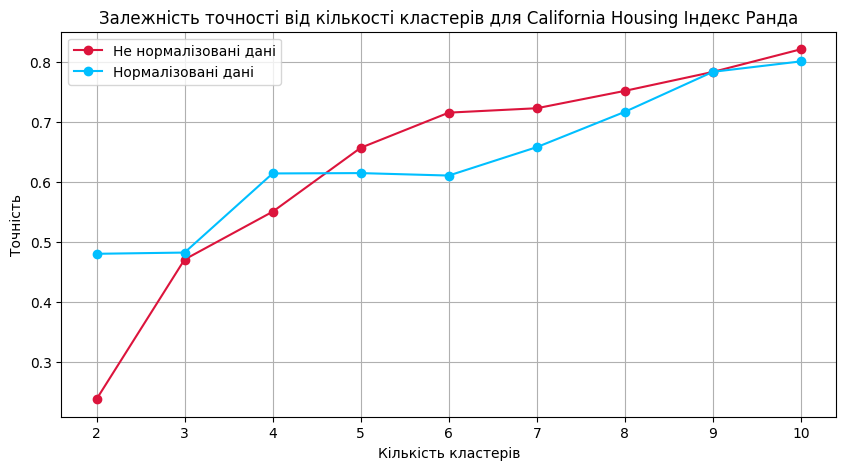
Точність на стандартизованих даних: **0.76295**

Точність на нормалізованих даних: **0.77629**

*Залежність точності від кількості кластерів в методі KMeans.*

*Метрика – Індекс Ранда*

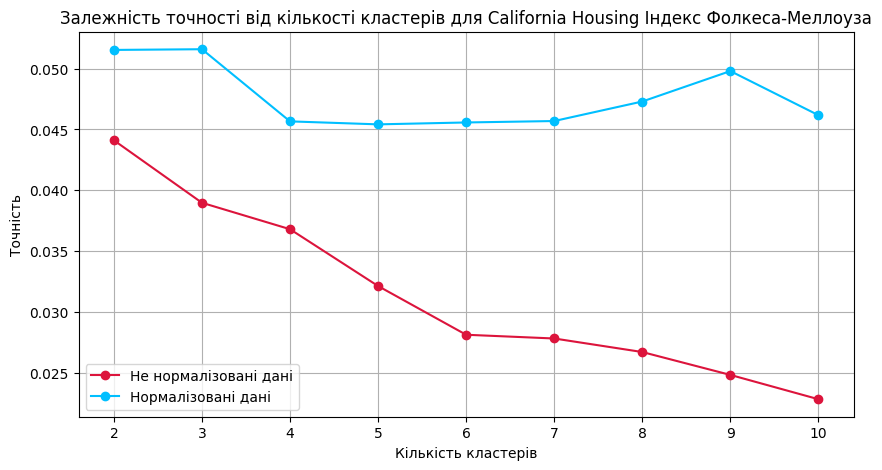
|  |  |  |
| --- | --- | --- |
| Кількість кластерів | Точність на ненормалізованих даних | Точність на нормалізованих даних |
| 2 | 0.23788 | 0.48034 |
| 3 | 0.47083 | 0.48239 |
| 4 | 0.55051 | 0.61431 |
| 5 | 0.65709 | 0.61488 |
| 6 | 0.71569 | 0.61079 |
| 7 | 0.72302 | 0.65820 |
| 8 | 0.75200 | 0.71707 |
| 9 | 0.78359 | 0.78374 |
| 10 | 0.82141 | 0.80117 |



*Залежність точності від кількості кластерів в методі KMeans.*

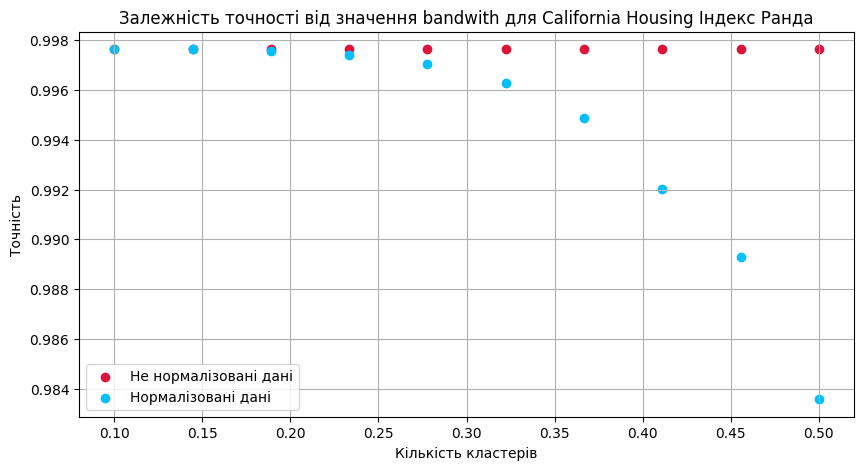
*Метрика – Індекс Фолкеса-Меллоуза*

|  |  |  |
| --- | --- | --- |
| Кількість кластерів | Точність на ненормалізованих даних | Точність на нормалізованих даних |
| 2 | 0.04410 | 0.05153 |
| 3 | 0.03897 | 0.05159 |
| 4 | 0.03679 | 0.04566 |
| 5 | 0.03212 | 0.04541 |
| 6 | 0.02812 | 0.04557 |
| 7 | 0.02781 | 0.04569 |
| 8 | 0.02670 | 0.04729 |
| 9 | 0.02481 | 0.04980 |
| 10 | 0.02282 | 0.04617 |



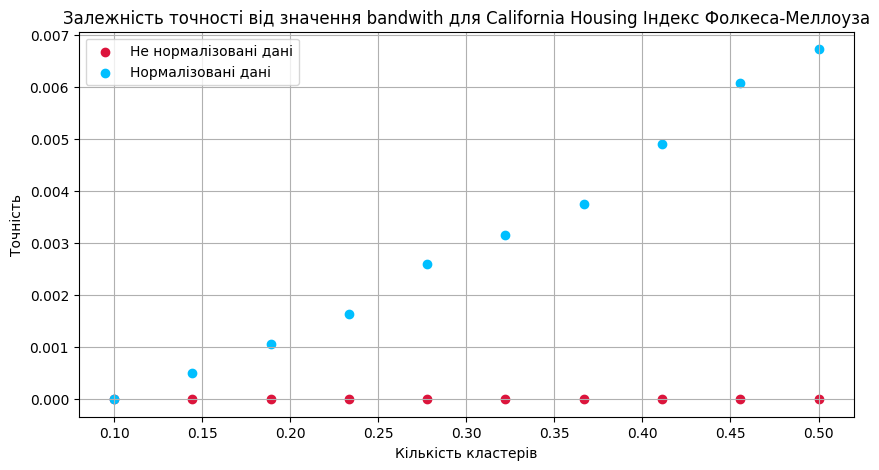
*Залежність точності від значення bandwith для алгоритму Mean Shift. Метрика - Індекс Ранда*

|  |  |  |
| --- | --- | --- |
| Значення bandwidth | Точність на ненормалізованих даних | Точність на нормалізованих даних |
| 0.10000 | 0.99763 | 0.99763 |
| 0.14444 | 0.99763 | 0.99762 |
| 0.18889 | 0.99763 | 0.99757 |
| 0.23333 | 0.99763 | 0.99741 |
| 0.27778 | 0.99763 | 0.99705 |
| 0.32222 | 0.99763 | 0.99628 |
| 0.36667 | 0.99763 | 0.99486 |
| 0.41111 | 0.99763 | 0.99202 |
| 0.45556 | 0.99763 | 0.98929 |
| 0.50000 | 0.99763 | 0.98359 |



*Залежність точності від значення bandwith для алгоритму Mean Shift. Метрика - Індекс Фолкеса-Меллоуза*

|  |  |  |
| --- | --- | --- |
| Значення bandwidth | Точність на ненормалізованих даних | Точність на нормалізованих даних |
| 0.10000 | 0.00000 | 0.00000 |
| 0.14444 | 0.00000 | 0.00051 |
| 0.18889 | 0.00000 | 0.00107 |
| 0.23333 | 0.00000 | 0.00164 |
| 0.27778 | 0.00000 | 0.00260 |
| 0.32222 | 0.00000 | 0.00316 |
| 0.36667 | 0.00000 | 0.00375 |
| 0.41111 | 0.00000 | 0.00490 |
| 0.45556 | 0.00000 | 0.00608 |
| 0.50000 | 0.00000 | 0.00672 |



**Код програми:**

**from sklearn.cluster import AgglomerativeClustering, KMeans, MeanShift**

**import numpy as np**

**from scipy.cluster.hierarchy import dendrogram, linkage**

**import matplotlib.pyplot as plt**

**from sklearn.preprocessing import StandardScaler, Normalizer**

**from sklearn.datasets import load\_iris, fetch\_california\_housing**

**from sklearn.metrics import accuracy\_score, mean\_squared\_error, rand\_score, fowlkes\_mallows\_score**

**iris = load\_iris()**

**X = iris.data**

**method\_list = ['single', 'complete', 'average', 'weighted', 'centroid', 'median', 'ward']**

**def plot\_metrics(cluster\_range, metric\_list\_unnormalized, metric\_list\_normalized, title=None, scatter=False):**

**"""**

**Визуализирует зависимость метрик от числа кластеров для необработанных и нормализованных данных.**

**Параметры:**

**- `cluster\_range (range)`: Диапазон числа кластеров.**

**- `metric\_list\_unnormalized (list)`: Список метрик для необработанных данных.**

**- `metric\_list\_normalized (list)`: Список метрик для нормализованных данных.**

**- `title (str)`: Заголовок графика.**

**- `scatter (boolean)`: Вивести точковий графік.**

**Возвращает:**

**- `None`**

**"""**

**if scatter:**

**plt.figure(figsize=(10, 5))**

**plt.scatter(cluster\_range, metric\_list\_unnormalized, marker='o', label='Не нормалізовані дані', color='crimson')**

**plt.scatter(cluster\_range, metric\_list\_normalized, marker='o', label='Нормалізовані дані', color='deepskyblue')**

**plt.xlabel('Кількість кластерів')**

**plt.ylabel('Точність')**

**plt.title(title)**

**plt.legend()**

**plt.grid(True)**

**plt.show()**

**else:**

**plt.figure(figsize=(10, 5))**

**plt.plot(cluster\_range, metric\_list\_unnormalized, marker='o', label='Не нормалізовані дані', color='crimson')**

**plt.plot(cluster\_range, metric\_list\_normalized, marker='o', label='Нормалізовані дані', color='deepskyblue')**

**plt.xlabel('Кількість кластерів')**

**plt.ylabel('Точність')**

**plt.title(title)**

**plt.legend()**

**plt.grid(True)**

**plt.show()**

**def fit\_and\_predict(X):**

**"""**

**Кластеризует данные методом агломерации, используя AgglomerativeClustering.**

**Параметры:**

**- `X (numpy.ndarray)`: Матрица признаков для кластеризации.**

**Возвращает:**

**- `float`: Точность предсказаний, измеренная с использованием индекса Рэнда.**

**"""**

**aglom\_cluster = AgglomerativeClustering(n\_clusters=2)**

**predictions = aglom\_cluster.fit\_predict(X)**

**accuracy = rand\_score(iris.target, predictions)**

**print(aglom\_cluster.labels\_)**

**return accuracy**

**for method in method\_list:**

**A = linkage(X, method=method)**

**plt.figure(figsize=(10, 5))**

**dendrogram(A)**

**plt.title(f'Дендрограма агломеративної кластеризації, method=**{method}**')**

**plt.xlabel('Data point')**

**plt.ylabel('Distance')**

**plt.show()**

**accuracy = fit\_and\_predict(X)**

**print(f'features are not normalized:** {accuracy:.5f}**')**

**X\_scaled = StandardScaler().fit\_transform(X)**

**accuracy\_scaled = fit\_and\_predict(X\_scaled)**

**print(f'features are standardized:** {accuracy\_scaled:.5f}**')**

**X\_normalized = Normalizer().fit\_transform(X)**

**accuracy\_normalized = fit\_and\_predict(X\_normalized)**

**print(f'features are normalized:** {accuracy\_normalized:.5f}**')**

**california = fetch\_california\_housing()**

**X = california.data[:7000]**

**X\_scaled = StandardScaler().fit\_transform(X)**

**cluster\_range = range(2, 11)**

**kmeans\_accuracy\_list\_unnormalized\_rand = []**

**kmeans\_accuracy\_list\_normalized\_rand = []**

**kmeans\_accuracy\_list\_unnormalized\_fowlkes = []**

**kmeans\_accuracy\_list\_normalized\_fowlkes = []**

**for cluster in cluster\_range:**

**kmeans = KMeans(n\_clusters=cluster)**

**predictions = kmeans.fit\_predict(X)**

**accuracy\_rand = rand\_score(california.target[:7000], predictions)**

**kmeans\_accuracy\_list\_unnormalized\_rand.append(accuracy\_rand)**

**accuracy\_fowlkes = fowlkes\_mallows\_score(california.target[:7000], predictions)**

**kmeans\_accuracy\_list\_unnormalized\_fowlkes.append(accuracy\_fowlkes)**

**kmeans = KMeans(n\_clusters=cluster)**

**predictions = kmeans.fit\_predict(X\_scaled)**

**accuracy\_rand = rand\_score(california.target[:7000], predictions)**

**kmeans\_accuracy\_list\_normalized\_rand.append(accuracy\_rand)**

**accuracy\_fowlkes = fowlkes\_mallows\_score(california.target[:7000], predictions)**

**kmeans\_accuracy\_list\_normalized\_fowlkes.append(accuracy\_fowlkes)**

**plot\_metrics(cluster\_range, kmeans\_accuracy\_list\_unnormalized\_rand, kmeans\_accuracy\_list\_normalized\_rand, 'Залежність точності від кількості кластерів для California Housing Індекс Ранда')**

**plot\_metrics(cluster\_range, kmeans\_accuracy\_list\_unnormalized\_fowlkes, kmeans\_accuracy\_list\_normalized\_fowlkes, 'Залежність точності від кількості кластерів для California Housing Індекс Фолкеса-Меллоуза')**

**bandwidth\_range = np.linspace(0.1, 0.5, num=10)**

**meanshift\_accuracy\_unnormalized\_rand = []**

**meanshift\_accuracy\_normalized\_rand = []**

**meanshift\_accuracy\_unnormalized\_fowlkes = []**

**meanshift\_accuracy\_normalized\_fowlkes = []**

**for bandwidth in bandwidth\_range:**

**mean\_shift = MeanShift(bandwidth=bandwidth)**

**predictions = mean\_shift.fit\_predict(X)**

**accuracy = rand\_score(california.target[:7000], predictions)**

**meanshift\_accuracy\_unnormalized\_rand.append(accuracy)**

**meanshift\_accuracy\_unnormalized\_fowlkes.append(fowlkes\_mallows\_score(california.target[:7000], predictions))**

**mean\_shift = MeanShift(bandwidth=bandwidth)**

**predictions = mean\_shift.fit\_predict(X\_scaled)**

**accuracy = rand\_score(california.target[:7000], predictions)**

**meanshift\_accuracy\_normalized\_rand.append(accuracy)**

**meanshift\_accuracy\_normalized\_fowlkes.append(fowlkes\_mallows\_score(california.target[:7000], predictions))**

**plot\_metrics(bandwidth\_range, meanshift\_accuracy\_unnormalized\_rand, meanshift\_accuracy\_normalized\_rand,'Залежність точності від значення bandwith для California Housing Індекс Ранда', True)**

**plot\_metrics(bandwidth\_range, meanshift\_accuracy\_unnormalized\_fowlkes, meanshift\_accuracy\_normalized\_fowlkes, 'Залежність точності від значення bandwith для California Housing Індекс Фолкеса-Меллоуза', True)**

**for i in cluster\_range:**

**print(f'**{i} **|** {kmeans\_accuracy\_list\_unnormalized\_rand[i-2]:.5f} **|** {kmeans\_accuracy\_list\_normalized\_rand[i-2]:.5f}**')**

**for i in cluster\_range:**

**print(f'**{i} **|** {kmeans\_accuracy\_list\_unnormalized\_fowlkes[i-2]:.5f} **|** {kmeans\_accuracy\_list\_normalized\_fowlkes[i-2]:.5f}**')**

**for i, j in zip(bandwidth\_range, range(len(meanshift\_accuracy\_unnormalized\_rand))):**

**print(f'**{i:.5f} **|** {meanshift\_accuracy\_unnormalized\_rand[j]:.5f} **|** {meanshift\_accuracy\_normalized\_rand[j]:.5f}**')**

**for i, j in zip(bandwidth\_range, range(len(meanshift\_accuracy\_unnormalized\_rand))):**

**print(f'**{i:.5f} **|** {meanshift\_accuracy\_unnormalized\_fowlkes[j]:.5f} **|** {meanshift\_accuracy\_normalized\_fowlkes[j]:.5f}**')**