

# Лабораторна робота №3. Кластеризація засобами бібліотеки Scikit-Learn Python

Недашківська Н.І.

Для отримання максимальної оцінки потрібно виконати ВСІ етапи Ходу виконання роботи та оформити звіт. Звітом може бути, наприклад, файл jupyter notebook з кодом програми і текстовими поясненнями отриманих цифр відповідно до Ходу виконання роботи.

**Звіт має містити:**

- результати по кожному пункту Ходу виконання роботи, в тому числі порівняльний аналіз декількох моделей,
- опис методу кластеризації, який використовувався,
- опис метрик якості кластеризації, за якими порівнювалися моделі.

Бажаємо опрацювати матеріал <https://scikit-learn.org/stable/modules/clustering.html>. За цим посиланням є **ОПИСИ МЕТОДІВ КЛАСТЕРИЗАЦІЇ**, які не увійшли до лекцій, приклади використання методів, **ОПИС МЕТРИК ЯКОСТІ**.

**УВАГА!** Задані за варіантом набори даних розглядати **без еталонних значень цільової змінної**. Еталонні значення цільової змінної можуть бути використані при розрахунку окремих метрик якості кластеризації.

**Захист роботи:**

- Усно: демонстрація коду програми, яка реалізує завдання згідно з варіантом і Ходом виконання роботи. Відповіді на питання щодо коду програми, отриманих результатів та методу кластеризації, який використовувався.
- Відповідь на теоретичне питання, розв'язання задачі, написання коду в редакторі пайтон за темою "Методи і алгоритми кластеризації".

## 1 Хід виконання роботи:

1. Представити початкові дані графічно.
2. Побудувати модель кластеризації згідно з варіантом.
3. Виконати кластеризацію даних на основі моделі.

4. Представити розбиття на кластери графічно, наприклад, різними кольорами.
5. Розрахувати додаткові результати кластеризації (згідно з варіантом).
6. Побудувати декілька альтернативних моделей:
  - шляхом зміни значень параметрів основної моделі,
  - використати різні функції відстані в алгоритмах, де це можна зробити,
  - задати різні значення кількості кластерів, в алгоритмах де кількість кластерів - параметр.
7. Для кожної альтернативної моделі розрахувати метрики якості кластеризації, що реалізовані в модулі `sklearn.metrics` (тільки метрики згідно з варіантом):
  - Estimated Number of Clusters.
  - Adjusted Rand Index.
  - Adjusted Mutual Information.
  - Homogeneity.
  - Completeness.
  - V-measure.
  - Silhouette Coefficient.
  - Calinski-Harabasz Index.
  - Davies-Bouldin index.
  - Contingency Matrix.
8. Виконати аналіз результатів кластеризації одним з неформальних методів (тільки методом згідно з варіантом):
  - чи є розбиття стабільним на підвибірках даних,
  - чи є розбиття стабільним після видалення окремих об'єктів,
  - чи є розбиття стабільним після зміни порядку об'єктів у множині об'єктів,
  - чи існує взаємозв'язок між результатами кластеризації і змінними, які не враховувалися при кластеризації,
  - чи можна інтерпретувати результати кластеризації.
9. Зробити висновки про якість роботи моделей на досліджених даних. Дослідити різні значення параметрів основної моделі, різні функції відстані та різну кількість кластерів в алгоритмах, де кількість кластерів слугує параметром.

10. Оцінити результати кластеризації на основі метрик якості та на основі неформальних методів. У кожному варіанті задано два набори даних. Для кожного набору даних обрати найкращу модель.

## 2 Варіанти завдань

1. Агломеративний алгоритм AgglomerativeClustering. Дослідити методи розрахунку відстані між кластерами: ward, single, average, complete. Побудувати матриці відстаней між кластерами, використовуючи `metrics.pairwise_distances`.

Чи є розбиття стабільним після видалення окремих об'єктів?

Метрики якості: Estimated number of clusters, Homogeneity, Completeness, V-measure.

**Початкові дані:**

(a) `sklearn.datasets.make_moons`

(б) 

```
import numpy as np
np.random.seed(0)
X = np.random.randn(300, 2)
Y = np.logical_xor(X[:, 0] > 0, X[:, 1] > 0)
```

2. Алгоритм розділу суміші expectation-maximization (EM), GaussianMixture та BayesianGaussianMixture модулю sklearn.mixture.

Метрики якості: Homogeneity, Completeness, V-measure.

Чи є розбиття стабільним на підвибірках даних?

**Початкові дані:**

(a) 

```
import numpy as np
np.random.seed(0)
X = np.random.randn(30000, 2)
Y = np.logical_xor(X[:, 0] > 0, X[:, 1] > 0)
```

(б) 

```
from sklearn.datasets import make_blobs
centers = [[1, 1], [-1, -1], [1, -1]]
X, labels_true = make_blobs(n_samples=30000, centers=centers,
                             cluster_std=0.5, random_state=0)
```

3. Алгоритм Mean Shift. Відобразити графічно центри кластерів.

Метрики якості: Estimated number of clusters, Adjusted Rand Index, Adjusted Mutual Information, Silhouette Coefficient.

Чи є розбиття стабільним після зміни порядку об'єктів у множині об'єктів?

**Початкові дані:**

```
(a) from sklearn.datasets.samples_generator import make_blobs
X, y_true = make_blobs(n_samples=400, centers=4,
                        cluster_std=0.60, random_state=0)
rng = np.random.RandomState(13)
X_stretched = np.dot(X, rng.randn(2, 2))

(б) from sklearn.datasets import make_blobs
X, y = make_blobs(n_samples=500,
                  n_features=2,
                  centers=4,
                  cluster_std=1,
                  center_box=(-10.0, 10.0),
                  shuffle=True,
                  random_state=1)
```

#### 4. Алгоритм Spectral clustering.

Метрики якості: Homogeneity, Completeness, V-measure.

Чи є розбиття стабільним на підвибірках даних?

**Початкові дані:**

```
(a) import numpy as np
np.random.seed(0)
n_points_per_cluster = 300
C1 = [-6, -2] + 0.7 * np.random.randn(n_points_per_cluster, 2)
C2 = [-2, 2] + 0.3 * np.random.randn(n_points_per_cluster, 2)
C3 = [1, -2] + 0.2 * np.random.randn(n_points_per_cluster, 2)
C4 = [4, -4] + 0.1 * np.random.randn(n_points_per_cluster, 2)
C5 = [5, 0] + 1.4 * np.random.randn(n_points_per_cluster, 2)
C6 = [5, 6] + 2.0 * np.random.randn(n_points_per_cluster, 2)
X = np.vstack((C1, C2, C3, C4, C5, C6))

(б) from sklearn.datasets.samples_generator import make_blobs
X, y_true = make_blobs(n_samples=400, centers=4,
                        cluster_std=0.60, random_state=0)
rng = np.random.RandomState(13)
X_stretched = np.dot(X, rng.randn(2, 2))
```

#### 5. Алгоритм DBSCAN. Розрахувати додатковий результат кластеризації: estimated number of noise points.

Метрики якості: Estimated number of clusters, Adjusted Rand Index, Adjusted Mutual Information, Silhouette Coefficient.

Чи є розбиття стабільним після зміни порядку об'єктів у множині об'єктів,

**Початкові дані:**

```
(a) import numpy as np
    np.random.seed(0)
    X = np.random.randn(300, 2)
    Y = np.logical_xor(X[:, 0] > 0, X[:, 1] > 0)

(б) from sklearn.datasets import make_blobs
    n_samples_1 = 1500
    n_samples_2 = 100
    n_samples_3 = 300
    centers = [[0.0, 0.0], [2.0, 2.0], [-2.0, -2.0]]
    clusters_std = [1.5, 0.5, 1.0]
    X, y = make_blobs(n_samples=
                      [n_samples_1, n_samples_2, n_samples_3],
                      centers=centers,
                      cluster_std=clusters_std,
                      random_state=0, shuffle=False)
```

6. Алгоритм Affinity propagation. Відобразити графічно центри кластерів.  
 Метрики якості: Estimated number of clusters, Calinski-Harabasz Index, Davies-Bouldin index, Contingency Matrix.

Чи є розбиття стабільним після вилучення окремих об'єктів?

**Початкові дані:**

```
(a) sklearn.datasets.make_moons

(б) from sklearn.datasets import make_blobs
    X, y = make_blobs(n_samples=500,
                      n_features=2,
                      centers=4,
                      cluster_std=1,
                      center_box=(-10.0, 10.0),
                      shuffle=True,
                      random_state=1)
```

7. Алгоритм DBSCAN. Розрахувати додатковий результат кластеризації:  
 estimated number of noise points.

Метрики якості: Estimated number of clusters, Homogeneity, Completeness, V-measure.

Чи є розбиття стабільним на підвибірках даних?

**Початкові дані:**

```
(a) sklearn.datasets.make_moons
(б) sklearn.datasets.load_digits
```

8. Алгоритм Birch. Метрики якості: Estimated number of clusters, Homogeneity, Completeness, V-measure.

Чи є розбиття стабільним на підвибірках даних?

**Початкові дані:**

```
(a) import numpy as np
    np.random.seed(0)
    X = np.random.randn(300, 2)
    Y = np.logical_xor(X[:, 0] > 0, X[:, 1] > 0)

(б) from sklearn.datasets.samples_generator import make_blobs
    X, y_true = make_blobs(n_samples=400, centers=4,
                           cluster_std=0.60, random_state=0)
    rng = np.random.RandomState(13)
    X_stretched = np.dot(X, rng.randn(2, 2))
```

9. Алгоритм Mean Shift. Відобразити графічно центри кластерів.

Метрики якості: Estimated number of clusters, Adjusted Rand Index, Adjusted Mutual Information, Silhouette Coefficient.

Чи є розбиття стабільним після видалення окремих об'єктів?

**Початкові дані:**

```
(a) sklearn.datasets.make_moons

(б) from sklearn.datasets import make_blobs
    from sklearn.preprocessing import StandardScaler
    centers = [[1, 1], [-1, -1], [1, -1]]
    X, labels_true = make_blobs(n_samples=750, centers=centers,
                                cluster_std=0.4, random_state=0)

    X = StandardScaler().fit_transform(X)
```

10. Алгоритм OPTICS.

Метрики якості: Estimated number of clusters, Adjusted Rand Index, Adjusted Mutual Information, Silhouette Coefficient.

Чи є розбиття стабільним після зміни порядку об'єктів у множині об'єктів?

**Початкові дані:**

```
(a) sklearn.datasets.samples_generator.make_circles

    X, y = make_circles(500, factor=.1, noise=.1)
```

```
(6) from sklearn.datasets import make_blobs
    n_samples_1 = 1000
    n_samples_2 = 100
    centers = [[0.0, 0.0], [2.0, 2.0]]
    clusters_std = [1.5, 0.5]
    X, y = make_blobs(n_samples=[n_samples_1, n_samples_2],
                      centers=centers,
                      cluster_std=clusters_std,
                      random_state=0, shuffle=False)
```

#### 11. Алгоритм Spectral clustering.

Метрики якості: Estimated number of clusters, Adjusted Rand Index, Silhouette Coefficient, Davies-Bouldin index.

Чи є розбиття стабільним після вилучення окремих об'єктів?

**Початкові дані:**

- (a) `sklearn.datasets.load_digits`
- (б) `sklearn.datasets.make_moons`

#### 12. Алгоритм Birch.

Метрики якості: Estimated number of clusters, Adjusted Rand Index, Adjusted Mutual Information, Silhouette Coefficient.

Чи є розбиття стабільним після зміни порядку об'єктів у множині об'єктів?

**Початкові дані:**

- (a) `sklearn.datasets.samples_generator.make_circles`
- (б) `sklearn.datasets.load_iris`

#### 13. Алгоритм OPTICS.

Метрики якості: Estimated number of clusters, Homogeneity, Completeness, V-measure.

Чи є розбиття стабільним на підвибірках даних?

**Початкові дані:**

- (a) `sklearn.datasets.make_moons`
- (б) 

```
from sklearn.datasets.samples_generator import make_blobs
X, y_true = make_blobs(n_samples=400, centers=4,
                      cluster_std=0.60, random_state=0)
rng = np.random.RandomState(13)
X_stretched = np.dot(X, rng.randn(2, 2))
```

14. Агломеративний алгоритм AgglomerativeClustering. Дослідити методи розрахунку відстані між кластерами: ward, single, average, complete.

Метрики якості: Estimated number of clusters, Adjusted Rand Index, V-measure. Побудувати матриці відстаней між кластерами, використовуючи `metrics.pairwise_distances`.

Чи є розбиття стабільним після видалення окремих об'єктів?

**Початкові дані:**

(a) `sklearn.datasets.load_iris`

(б) `from sklearn.datasets.samples_generator import make_blobs`  
`X, y_true = make_blobs(n_samples=400, centers=4,`  
`cluster_std=0.60, random_state=0)`  
`rng = np.random.RandomState(13)`  
`X_stretched = np.dot(X, rng.randn(2, 2))`

15. Алгоритм DBSCAN. Розрахувати додатковий результат кластеризації: estimated number of noise points.

Метрики якості: Adjusted Mutual Information, Silhouette Coefficient, Calinski-Harabasz Index.

Чи є розбиття стабільним після зміни порядку об'єктів у множині об'єктів?

**Початкові дані:**

(a) `from sklearn.datasets import make_blobs`  
`n_samples_1 = 1000`  
`n_samples_2 = 100`  
`centers = [[0.0, 0.0], [-2.0, -2.0]]`  
`clusters_std = [2.0, 1.0]`  
`X, y = make_blobs(n_samples=[n_samples_1, n_samples_2],`  
`centers=centers,`  
`cluster_std=clusters_std,`  
`random_state=0, shuffle=False)`

(б) `import numpy as np`  
`np.random.seed(0)`  
`n_points_per_cluster = 300`  
`C1 = [-6, -2] + 0.7 * np.random.randn(n_points_per_cluster, 2)`  
`C2 = [-2, 2] + 0.3 * np.random.randn(n_points_per_cluster, 2)`  
`C3 = [1, -2] + 0.2 * np.random.randn(n_points_per_cluster, 2)`  
`C4 = [4, -4] + 0.1 * np.random.randn(n_points_per_cluster, 2)`  
`C5 = [5, 0] + 1.4 * np.random.randn(n_points_per_cluster, 2)`  
`C6 = [5, 6] + 2.0 * np.random.randn(n_points_per_cluster, 2)`  
`X = np.vstack((C1, C2, C3, C4, C5, C6))`



16. Алгоритм розділу суміші expectation-maximization (EM), GaussianMixture та BayesianGaussianMixture модулю sklearn.mixture.

Метрики якості: Homogeneity, Completeness, V-measure.

Чи є розбиття стабільним на підвибірках даних?

**Початкові дані:**

```
(a) sklearn.datasets.make_moons

(б) from sklearn.datasets.samples_generator import make_blobs
X, y_true = make_blobs(n_samples=400, centers=4,
                        cluster_std=0.60, random_state=0)
rng = np.random.RandomState(13)
X_stretched = np.dot(X, rng.randn(2, 2))
```

17. Алгоритм OPTICS.

Метрики якості: Estimated number of clusters, Adjusted Rand Index, Adjusted Mutual Information, Silhouette Coefficient.

Чи є розбиття стабільним після зміни порядку об'єктів у множині об'єктів?

**Початкові дані:**

```
(a) import numpy as np
np.random.seed(0)
X = np.random.randn(300, 2)
Y = np.logical_xor(X[:, 0] > 0, X[:, 1] > 0)

(б) import numpy as np
np.random.seed(0)
n_points_per_cluster = 300
C1 = [-6, -2] + 0.7 * np.random.randn(n_points_per_cluster, 2)
C2 = [-2, 2] + 0.3 * np.random.randn(n_points_per_cluster, 2)
C3 = [1, -2] + 0.2 * np.random.randn(n_points_per_cluster, 2)
C4 = [4, -4] + 0.1 * np.random.randn(n_points_per_cluster, 2)
C5 = [5, 0] + 1.4 * np.random.randn(n_points_per_cluster, 2)
C6 = [5, 6] + 2.0 * np.random.randn(n_points_per_cluster, 2)
X = np.vstack((C1, C2, C3, C4, C5, C6))
```

18. Алгоритм  $k$ -середніх, використати cluster.KMeans і cluster.MinibatchKMeans. Відобразити графічно центри кластерів.

Метрики якості: Adjusted Rand Index, Calinski-Harabasz Index, Davies-Bouldin index.

Чи є розбиття стабільним на підвибірках даних?

**Початкові дані:**

```

(a) from sklearn.datasets import make_blobs
    n_samples_1 = 1000
    n_samples_2 = 100
    centers = [[0.0, 0.0], [-2.0, -2.0]]
    clusters_std = [2.0, 1.0]
    X, y = make_blobs(n_samples=[n_samples_1, n_samples_2],
                      centers=centers,
                      cluster_std=clusters_std,
                      random_state=0, shuffle=False)

(б) import numpy as np
    np.random.seed(0)
    n_points_per_cluster = 30000
    C1 = [-6, -2] + 0.7 * np.random.randn(n_points_per_cluster, 2)
    C2 = [-2, 2] + 0.3 * np.random.randn(n_points_per_cluster, 2)
    C3 = [1, -2] + 0.2 * np.random.randn(n_points_per_cluster, 2)
    C4 = [4, -4] + 0.1 * np.random.randn(n_points_per_cluster, 2)
    C5 = [5, 0] + 1.4 * np.random.randn(n_points_per_cluster, 2)
    C6 = [5, 6] + 2.0 * np.random.randn(n_points_per_cluster, 2)
    X = np.vstack((C1, C2, C3, C4, C5, C6))

```

19. Алгоритм розділу суміші expectation-maximization (EM), GaussianMixture та BayesianGaussianMixture модулю sklearn.mixture.

Метрики якості: Homogeneity, Completeness, V-measure.

Чи є розбиття стабільним на підвибірках даних?

**Початкові дані:**

```

(a) sklearn.datasets.samples_generator.make_circles

(б) from sklearn.datasets import make_blobs
    X, y = make_blobs(n_samples=500,
                      n_features=2,
                      centers=4,
                      cluster_std=1,
                      center_box=(-10.0, 10.0),
                      shuffle=True,
                      random_state=1)

```

20. Алгоритм Affinity propagation. Відобразити графічно центри кластерів.

Метрики якості: Adjusted Rand Index, Adjusted Mutual Information, Silhouette Coefficient.

Чи є розбиття стабільним після видалення окремих об'єктів?

**Початкові дані:**

```
(a) import numpy as np
    np.random.seed(0)
    X = np.random.randn(300, 2)
    Y = np.logical_xor(X[:, 0] > 0, X[:, 1] > 0)

(б) from sklearn.datasets import make_blobs
    n_samples_1 = 1000
    n_samples_2 = 100
    centers = [[0.0, 0.0], [2.0, 2.0]]
    clusters_std = [2.0, 1.0]
    X, y = make_blobs(n_samples=[n_samples_1, n_samples_2],
                      centers=centers,
                      cluster_std=clusters_std,
                      random_state=0, shuffle=False)
```

21. Агломеративний алгоритм AgglomerativeClustering. Дослідити методи розрахунку відстані між кластерами: ward, single, average, complete. Побудувати матриці відстаней між кластерами, використовуючи

`metrics.pairwise_distances`.

Метрики якості: Estimated number of clusters, Homogeneity, Completeness, V-measure.

Чи є розбиття стабільним після зміни порядку об'єктів у множині об'єктів?

**Початкові дані:**

```
(a) sklearn.datasets.samples_generator.make_circles

(б) from sklearn.datasets import make_blobs
    X, y = make_blobs(n_samples=500,
                      n_features=2,
                      centers=4,
                      cluster_std=1,
                      center_box=(-10.0, 10.0),
                      shuffle=True,
                      random_state=1)
```

22. Алгоритм  $k$ -середніх, використати `cluster.KMeans` і `cluster.MinibatchKMeans`. Відобразити графічно центри кластерів.

Метрики якості: Adjusted Rand Index, Calinski-Harabasz Index, Davies-Bouldin index.

Чи є розбиття стабільним після видалення окремих об'єктів?

**Початкові дані:**

```
(a) sklearn.datasets.load_iris
```

(б) `sklearn.datasets.samples_generator.make_circles`  
`X, y = make_circles(500, factor=.1, noise=.1)`

23. Алгоритм Spectral clustering.

Метрики якості: Estimated number of clusters, Adjusted Rand Index, Silhouette Coefficient, Davies-Bouldin index.

Чи є розбиття стабільним після вилучення окремих об'єктів?

**Початкові дані:**

(а) `sklearn.datasets.load_digits`  
(б) `sklearn.datasets.samples_generator.make_circles`  
`X, y = make_circles(500, factor=.1, noise=.1)`

24. Алгоритм  $k$ -середніх, використати `cluster.KMeans` і `cluster.MinibatchKMeans`.  
Відобразити графічно центри кластерів.

Метрики якості: Adjusted Rand Index, Calinski-Harabasz Index, Davies-Bouldin index.

Чи є розбиття стабільним на підвибірках даних?

**Початкові дані:**

(а) `sklearn.datasets.make_moons`  
(б) **from** `sklearn.datasets` **import** `make_blobs`  
`X, y = make_blobs(n_samples=500,`  
`n_features=2,`  
`centers=4,`  
`cluster_std=1,`  
`center_box=(-10.0, 10.0),`  
`shuffle=True,`  
`random_state=1)`

25. Агломеративний алгоритм AgglomerativeClustering. Дослідити методи розрахунку відстані між кластерами: `ward`, `single`, `average`, `complete`.

Метрики якості: Estimated number of clusters, Adjusted Rand Index, V-measure. Побудувати матриці відстаней між кластерами, використовуючи `metrics.pairwise_distances`.

Чи є розбиття стабільним після вилучення окремих об'єктів?

**Початкові дані:**

(а) `sklearn.datasets.make_moons`  
(б) `sklearn.datasets.fetch_covtype`, використати для побудови моделі лише частину даних

26. Алгоритм  $k$ -середніх, `sklearn.cluster.KMeans` і `sklearn.cluster.MinibatchKMeans`.  
Відобразити графічно центри кластерів.

Метрики якості: Adjusted Rand Index, Calinski-Harabasz Index, Davies-Bouldin index.

Чи є розбиття стабільним після зміни порядку об'єктів у множині об'єктів?

**Початкові дані:**

(a) `sklearn.datasets.load_digits`

(б) `sklearn.datasets.samples_generator.make_circles`

```
X, y = make_circles(50000, factor=.1, noise=.1)
```

27. Алгоритм  $k$ -середніх, `sklearn.cluster.KMeans` і `sklearn.cluster.MinibatchKMeans`.  
Відобразити графічно центри кластерів.

Метрики якості: Estimated number of clusters, Adjusted Rand Index, V-measure.

Чи є розбиття стабільним після видалення окремих об'єктів?

**Початкові дані:**

(a) `sklearn.datasets.load_iris`

(б) `import numpy as np`

```
np.random.seed(0)
```

```
X = np.random.randn(300, 2)
```

```
Y = np.logical_xor(X[:, 0] > 0, X[:, 1] > 0)
```

### 3 Контрольні питання для захисту роботи

1. Загальна постановка задачі кластеризації. Функції відстані в задачах кластеризації.
2. Агломеративний ієрархічний алгоритм найближчого сусіда. Як можна визначати відстань між кластерами?
3. Клас `sklearn.cluster.AgglomerativeClustering`.
4. Розділяючий ієрархічний алгоритм DIANA.
5. Етапи алгоритму  $k$ -середніх. Переваги і недоліки алгоритму  $k$ -середніх.
6. Клас `sklearn.cluster.KMeans`.
7. Клас `sklearn.cluster.MinibatchKMeans`.
8. Етапи алгоритму нечітких  $k$ -середніх. Переваги і недоліки.

9. Алгоритм G-середніх.
10. Постановка задачі розділу суміші. Базовий алгоритм Expectation-Maximization (ЕМ): ідея та етапи.
11. Алгоритм ЕМ з фіксованою кількістю компонент. Недоліки алгоритму.
12. Алгоритм ЕМ з послідовним додаванням компонент.
13. Класи `sklearn.mixture.GaussianMixture` та `sklearn.mixture.BayesianGaussianMixture`.
14. Алгоритм знаходження зв'язних компонент. Переваги і недоліки.
15. Поняття мінімального покриваючого дерева (МПД). «Жадібний» алгоритм побудови МПД.
16. Алгоритми Крускала і Прима побудови мінімального покриваючого дерева.
17. Алгоритм Борувки побудови мінімального покриваючого дерева.
18. Алгоритм Mean Shift.
19. Клас `sklearn.cluster.MeanShift`.
20. Алгоритми DBSCAN та OPTICS.
21. Клас `sklearn.cluster.DBSCAN`.
22. Клас `sklearn.cluster.OPTICS`.
23. Алгоритм Spectral clustering.
24. Клас `sklearn.cluster.SpectralClustering`.
25. Алгоритми Affinity propagation та Birch.
26. Клас `sklearn.cluster.AffinityPropagation`.
27. Клас `sklearn.cluster.Birch`.
28. Ідея базового алгоритму FOREL. Етапи базового алгоритму FOREL. Переваги і недоліки.
29. FOREL-3 і FOREL-4.
30. Постановка задачі кластеризації на основі мережі Кохонена. Використання методу стохастичного градієнта.
31. Опис мережі Кохонена.

32. Алгоритм стохастичного градієнта для розрахунку центрів кластерів в мережі Кохонена.
33. Поняття карти Кохонена. Метрики на карті.
34. Алгоритм навчання карти Кохонена.
35. Візуалізація карт Кохонена, типи карт. Переваги і недоліки методу карт Кохонена.
36. Метрики якості кластеризації: Adjusted Rand Index, Adjusted Mutual Information.
37. Метрики якості кластеризації: Homogeneity, Completeness, V-measure.
38. Метрика якості кластеризації - коефіцієнт силуету Silhouette Coefficient.
39. Метрики якості кластеризації: Calinski-Harabasz Index, Davies-Bouldin index.