Практикум 3. Кластеризація засобами бібліотеки Scikit-Learn Python

Недашківська Н.І.

Варіанти завдань та початкові дані вибирати відповідно до номеру в списку групи.

УВАГА! При навчанні моделей кластеризації вибірки даних розглядати **без еталонних значень цільової змінної**.

Для отримання максимальної оцінки потрібно виконати ВСІ етапи ходу виконання роботи та оформити звіт.

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.

Бажано 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** кластеризації.

Захист роботи:

- Демонстрація програми, яка реалізує завдання згідно з варіантом.
- Письмово теоретичне питання, задача та/ або виконання іншого варіанту завдання на комп'ютері.

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,
    - задати різні значення кількості кластерів, де кількість кластерів - параметр алгоритму.
7. Для кожної альтернативної моделі розрахувати метрики якості кластеризації, що реалізовані в класі metrics, 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 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. Для кожного набору даних вибрати найкращу модель.

2 Варіанти завдань для групи КА-75

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

   `metrics.pairwise_distances`.

   Метрики якості: Homogeneity, Completeness, V-measure.

   Чи є розбиття стабільним після вилучення окремих об'єктів?

   Початкові дані:

   (а)
   ```
   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)
   ```

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

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

2. Алгоритм Spectral clustering.

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

   Чи є розбиття стабільним на підвибірках даних?

   Початкові дані:

   (а)
   ```
   from sklearn.datasets import make_blobs
   centers = [[1, 1], [-1, -1], [1, -1]]
   X, labels_true = make_blobs(n_samples=300, centers=centers,
                               cluster_std=0.5, random_state=0)
   ```

   (б) `sklearn.datasets.load_iris`

3. Алгоритм k-середніх, методи cluster.KMeans і cluster.MiniBatchKMeans. 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:

(а)
```
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: Estimated number of clusters, Adjusted Rand Index, Adjusted Mutual Information, Silhouette Coefficient.

   Чи є розбиття стабільним після вилучення окремих об'єктів?

   Початкові дані:

   (а)
   ```
   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=0.90, random_state=0)
   rng = np.random.RandomState(13)
   X_stretched = np.dot(X, rng.randn(2, 2))
   ```

5. Алгоритм розділу суміші expectation-maximization (EM), методи GaussianMixture та BayesianGaussianMixture класу mixture.

   Метрики якості: Homogeneity, Completeness, V-measure.

   Чи є розбиття стабільним на підвибірках даних?

   Початкові дані:

   (а)
   ```
   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 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, Homogeneity, Completeness, V-measure.

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

   Початкові дані:

   (а) `sklearn.datasets.samples_generator.make_circles`
   ```
   X, y = make_circles(200, factor=.1, noise=.1)
   ```

   (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 Birch.

   Метрики якості: Estimated number of clusters, Calinski-Harabasz Index, Davies-Bouldin index, Contingency Matrix.

   Чи є розбиття стабільним на підвибірках даних? Дослідити виконання алгоритму на даних XOR різного розміру, в тому числі на даних дуже великого розміру.

   Початкові дані:

(а)
```
import numpy as np
np.random.seed(0)
X = np.random.randn(5300, 2)
Y = np.logical_xor(X[:, 0] > 0, X[:, 1] > 0)
```

(б) `sklearn.datasets.load_digits`

8. 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:

   (а)
   ```
   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) `sklearn.datasets.load_iris`

9. Алгоритм k-середніх, методи 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? Дослідити виконання алгоритму на даних XOR та `make_circles` різного розміру, в тому числі на даних дуже великого розміру.

   Початкові дані:

   (а)
   ```
   import numpy as np
   np.random.seed(0)
   X = np.random.randn(5300, 2)
   Y = np.logical_xor(X[:, 0] > 0, X[:, 1] > 0)
   ```

   (б)
   ```
   from sklearn.datasets.samples_generator import make_circles
   X, y = make_circles(5500, factor=.1, noise=0.1)
   ```

10. Алгоритм розділу суміші expectation-maximization (EM), методи GaussianMixture та BayesianGaussianMixture класу mixture.

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

    Чи є розбиття стабільним після вилучення окремих об'єктів?

    Початкові дані:

    (а) `sklearn.datasets.make_moons`

    (б) `sklearn.datasets.load_digits`

11. Алгоритм OPTICS (Ordering Points To Identify the Clustering Structure), використати cluster.OPTICS і `cluster.cluster_optics_dbscan`.

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

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

    Початкові дані:

    (а)
    ```
    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)
    ```
    sklearn.datasets.samples_generator.make_circles
    X, y = make_circles(400, factor=.1, noise=.1)
    ```

12. Алгоритм k-середніх, методи cluster.KMeans і cluster.MiniBatchKMeans. Відобразити графічно центри кластерів.

    Початкові дані:

    (а)
    ```
    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) `sklearn.datasets.load_iris`

13. Алгоритм OPTICS (Ordering Points To Identify the Clustering Structure), використати cluster.OPTICS і `cluster.cluster_optics_dbscan`.

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

    Чи є розбиття стабільним на підвибірках даних?

    Початкові дані:

(а)
```
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) `sklearn.datasets.make_moons`

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 внутрішньокласових відстаней, використовуючи `metrics.pairwise_distances`.

    Чи є розбиття стабільним після вилучення окремих об'єктів?

    Початкові дані:

    (а)
    ```
    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)
    ```

    (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))
    ```

15. Алгоритм Birch.

    Метрики якості: Homogeneity, Completeness, V-measure.

    Чи є розбиття стабільним після зміни порядку об'єктів у множині об'єктів? Дослідити виконання алгоритму на даних blobs різного розміру, в тому числі на даних дуже великого розміру.

Початкові дані:

(а) `sklearn.datasets.make_moons`

(б)
```
from sklearn.datasets.samples_generator import make_blobs
X, y_true = make_blobs(n_samples=5400, centers=5,
                       cluster_std=0.90, random_state=0.1)
rng = np.random.RandomState(13)
X_stretched = np.dot(X, rng.randn(2, 2))
```

16. Алгоритм розділу суміші expectation-maximization (EM). Використати методи GaussianMixture та BayesianGaussianMixture класу mixture.

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

    Чи є розбиття стабільним на підвибірках даних?

    Початкові дані:

    (а)
    ```
    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) `sklearn.datasets.load_iris`

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

    `metrics.pairwise_distances`.

    Чи є розбиття стабільним після вилучення окремих об'єктів?

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

    Початкові дані:

    (а)
    ```
    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) `sklearn.datasets.load_iris`

18. Алгоритм k-середніх, використати методи 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.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))
    ```

19. Алгоритм DBSCAN (Density-Based Spatial Clustering of Applications with Noise). 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:

    (а)
    ```
    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) `sklearn.datasets.load_iris`

20. Алгоритм Affinity propagation. Відобразити графічно центри кластерів.

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

    Чи є розбиття стабільним після вилучення окремих об'єктів?

    Початкові дані:

    (а)
    ```
    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 внутрішньокласових відстаней, 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:

    (а) `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)
    ```