Яблонцева А.Д.
Хакасский государственный университет им. Н.Ф. Катанова, Россия, г. Абакан

ИНДЕКС ДЭВИСА-БОЛДИНА ДЛЯ ОЦЕНКИ КЛАСТЕРИЗАЦИИ МЕТОДОМ K-СРЕДНИХ В PYTHON

Аннотация

В статье рассмотрены формулы для оценки кластеризации. Приведён пример, как рассчета индекса Дэвиса-Болдина для оценки кластеризации в Python на примера набора данных Ирисы Фишера и библиотеки Sklearn [1].

Ключевые слова: внутрикластерная дисперсия, кластеризация методом k-средних.
Keywords: intra-cluster variance, DBI, K-means clustering.

Научный руководитель Голубничий А.А.
старший преподаватель кафедры ПОВТиАС,
ХГУ им. Н.Ф. Катанова, Россия, г. Абакан

Индекс Дэвиса-Болдина (DBI) является одним из критериев оценки алгоритмов кластеризации. Чаще всего он используется для оценки качества разделения алгоритмом кластеризации методом k-средних [2] для заданного числа кластеров. То есть, оценка (DBI) [3] рассчитывается как среднее сходство каждого кластера с наиболее похожим на него кластером. Чем меньше среднее сходство, тем лучше разделены кластеры и тем лучше результат выполненной кластеризации. Исследование меры разделения кластеров было опубликовано еще в 1979 году. Представленный материал описывает измерение сходства между кластерами как функцию внутрикластерной дисперсии и разделения между кластерами. Дисперсия кластера вычисляется по формуле:

$$S_i = \left\{ \frac{1}{T_i} \sum_{j=1}^{T_i} |X_j - A_j|^q \right\}^{\frac{1}{q}}$$

где i – номер кластера, T_i – количество векторов (значений) в кластере i, X_j – j-й вектор в кластере i, A_i – центроид кластер i.

Чтобы получить внутрикластерную дисперсию, мы вычисляем среднее расстояние между каждым наблюдением внутри кластера и его центроидом. Предположим, что кластеризация K-средних, которая была выполнена для некоторого набора данных, сгенерировала три кластера (рисунок 1).

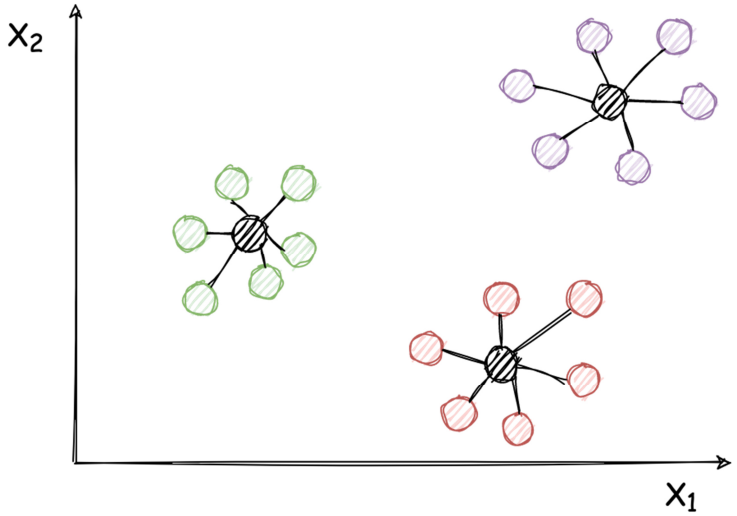

Рисунок 1 – Пример разбиения данных на три кластера

Используя приведенную формулу, внутрикластерная дисперсия будет рассчитана для каждого из кластеров, и у нас будут значения для S_1, S_2, и S_3. Следующее уравнение вычисляет расстояние между кластерами i и j (рисунок 2):

$$M_{ij} = \left\{ \sum_{k=1}^{N} |a_{ki} - a_{kj}|^p \right\}^{\frac{1}{p}}$$

где a_{ki} – k-й компонент n-размерного центроида A_i, a_{kj} – k-й компонент n-размерного центроида A_j, N – общее количество кластеров.

Используя приведенную выше формулу, мы вычислим меру разделения для каждой возможной комбинации двух кластеров: M_{11}, M_{12}, M_{13}, M_{21}, M_{22}, M_{23}, M_{31}, M_{32} и M_{33}. Сходство между кластерами рассчитывается с помощью формулы Дэвиса-Болдина:

$$R_{ij} = \frac{S_i + S_j}{M_{ij}}$$

В данном случае мы вычисляем сходство между кластерами как сумму двух внутрикластерных дисперсий, разделенных на меру разделения. Чем больше R_{ij}, тем более схожи кластеры i и j. Так же можем сказать, что лучший случай – когда значение R будет меньше. Далее рассчитываем наиболее похожий кластер, для каждого кластера i находим самое высокое соотношение из всех расчётов R_{ij}. R_{12} указывает на сходство между кластерами 1 и 2, а R_{13} на сходство между кластерами 1 и 3, соответственно. Из этих двух мер мы выберем самую большую и назовем максимальную меру как R_1. Следуя той же логике, найдем R_2 и R_3.

Для определения индекса Дэвиса-Болдина будем использовать формулу:

$$\bar{R} = \frac{1}{N} \sum_{i=1}^{N} R_i$$

Расчёт представляет собой просто среднее значение показателей сходства каждого кластера с наиболее похожим на него кластером. Чтобы в перспективе представить, как выглядят кластеры, необходимо визуализировать их. Пример такой визуализации и исходный код для реализации алгоритмы приведены на рисунках 2 и 3 для набора данных Ирисы Фишера.

```
from sklearn.datasets import load_iris
from sklearn.cluster import KMeans
from sklearn.metrics import davies_bouldin_score
import matplotlib.pyplot as plt
iris = load_iris()
X = iris.data[:, :2]
kmeans = KMeans(n_clusters=3, random_state=30)
labels = kmeans.fit_predict(X)
unique_labels = list(set(labels))
colors = ['red', 'orange', 'grey']
for i in unique_labels:
    filtered_label = X[labels == i]
    plt.scatter(filtered_label[:,0],
                filtered_label[:,1],
                color=colors[i],
                edgecolor='k')
plt.xlabel('Sepal length')
plt.ylabel('Sepal width')
plt.show()
```

Рисунок 2 – Исходный код для разбиения на кластеры набора данных Ирисы Фишера

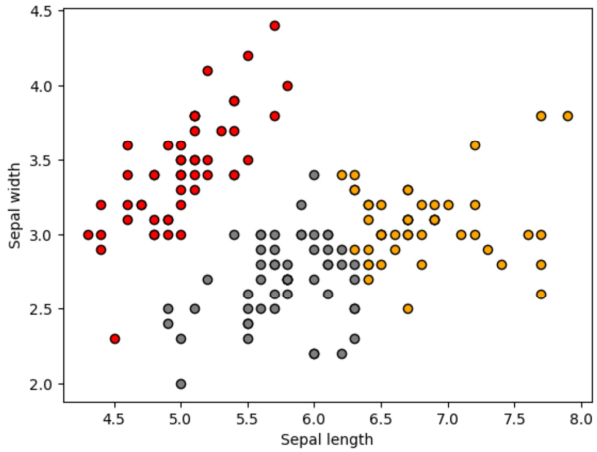

Рисунок 3 – Пример разбиения данных Ирисы Фишера на три кластера

Для сравнения расчетов и определения оптимального количества кластеров логично повторить расчет для множества вариаций разбиения. Результаты разбиения будем записывать в соответствующий словарь result для его последующей визуализации. Пример расчета и визуализации приведен на рисунках 4 и 5.

```
results = {}

for i in range(2, 11):
    kmeans = KMeans(n_clusters=i, random_state=30)
    labels = kmeans.fit_predict(X)
    db_index = davies_bouldin_score(X, labels)
    results.update({i: db_index})

plt.plot(list(results.keys()), list(results.values()))
plt.xlabel("Number of clusters")
plt.ylabel("Davies-Boulding Index")
plt.show()
```

Рисунок 4 – Исходный код для расчета индекса Дэвиса-Болдина для набора данных Ирисы Фишера (от 2 до 10 кластеров)

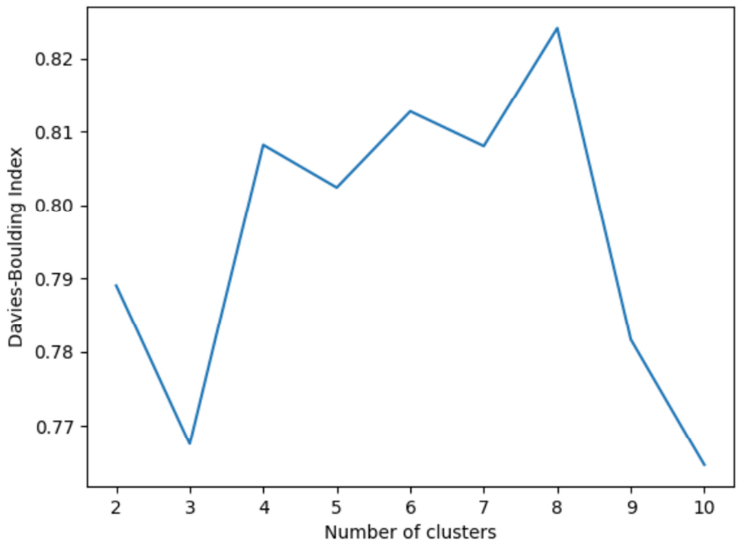

Рисунок 5 – Визуализация значений индекса Дэвиса-Болдина для набора данных Ирисы Фишера (от 2 до 10 кластеров)

Набор данных Ирисы Фишера содержит сведения о трех сортах данного растения (Виргинский, Щетинистый и Разноцветный), таким образом, итоговый расчет, изображенный на рисунке 5 подтверждает факт хорошей кластеризации на три кластера. Большее разделение (10 кластеров) также дает хорошие результаты, однако искусственное увеличение категорий, как показывает опыт, является не самой лучшей практикой и может найти признаки и свойства даже там, где их нет.

Литература

1. Installing scikit-learn – scikit-learn 0.24.0 documentation[Электронный ресурс] URL: https://clck.ru/Wu85A (дата обращения: 13.08.2021).
2. Кластеризация: метод k-средних [Электронный ресурс] URL: https://clck.ru/Wu85C (дата обращения: 13.08.2021).
3. Стратегии кластерной оценки [Электронный ресурс] https://clck.ru/Wu85J (дата обращения: 13.08.2021).