СЕРВИС ДЛЯ КЛАСТЕРИЗАЦИИ ДАННЫХ С ГЕОТЕГАМИ АЛГОРИТМОМ k-MXT-W

А. А. Степанова, С. П. Сидоров, В. А. Балаш

Саратовский национальный исследовательский государственный университет им. Н. Г. Чернышевского, Россия
E-mail: stacey.stepanova@gmail.com, sidorovsp@yahoo.com, vladimirbalash@yandex.ru

Кластеризация - задача разбиения набора данных на группы, называемые кластерами. В задаче кластеризации необходимо разделить данные таким образом, чтобы объекты (точки данных), находящие в одном и том же кластере, были схожи между собой по определенному множеству их свойств, а объекты, находящиеся в разных кластерах, отличались друг от друга на этом множестве значимых свойств. В данной работе будет анонсирован пакет для языка Python, который выполняет кластеризацию данных с помощью алгоритмов k-MXT и k-MXT-W.

SERVICE FOR DATA CLUSTERING WITH GEOTAGS USING THE K-MXT-W ALGORITHM

A. A. Stepanova, S. P. Sidorov, V. A. Balash

Clustering is the task of breaking a dataset into groups called clusters. In the clustering task, it is necessary to divide data in such a way that objects (data points) located in the same cluster are similar to each other in a certain set of their properties, and objects located in different clusters differ from each other on this set of significant properties. In this paper, we will describe a package for the Python language that performs data clustering using the k-MXT and k-MXT-W algorithms.

Алгоритм кластеризации k-MXT.

Впервые алгоритм k-MXT был описан в работе [1].

Для применения алгоритма k-MXT на данных с геотегами необходимо построить граф G = {V, E} (V - множество вершин, E - множество ребер графа), вершинами которого являются объекты наборов данных. Ребрами графа G являются связи между вершинами, расположенные на расстоянии не превосходящим ε. Расстояние считается на основе признаков, которыми обладают точки, по выбранной метрике. Например, при кластеризации геоданных признаками объектов данных являются значения долготы и широты точек, в качестве метрики может быть выбрано евклидово расстояние. Граф G является неориентированным, не содержит кратных ребер и петель.

Далее на основе графа G необходимо построить граф G' = {V', E'}, где V' - множество вершин, E' - множество дуг графа G'. Изначально граф G' содержит все вершины графа G, но граф G' не содержит ребер, т.е. V' = V, E' = ⊕.

Пусть множество E(q) содержит все дуги, исходящие из вершины q графа G. Для каждой вершины q графа G считается вес всех дуг, исходящих из этой вершины. Вес дуги (q, r) равен количеству общих соседей вершины q и верши-

ны r. Тогда дуги в ориентированный граф G' необходимо добавлять следующим образом:

– Если в множестве E(q) меньше, чем k дуг или ровно k дуг, то в граф добавляются все дуги из множества E(q).

– Если в множестве E(q) больше, чем k дуг, то в граф добавляются k дуг с максимальным весом. Если необходимо выбрать из нескольких дуг с одинаковым весом, то эти дуги выбираются случайным образом. Качество кластеризации сильно ухудшится, если дуги с одинаковым весом не будут выбираться случайным образом.

Таким образом параметр k показывает, какое максимальное количество исходящих дуг может быть у каждой вершины графа G'.

Кластерами будут являться компоненты сильной связности полученного ориентированного графа G'.

Компонентой сильной связности некоторого ориентированного графа G называется подмножество вершин графа G такое, что любые две вершины этого подмножества достижимы (есть путь) друг из друга по ориентированным дугам этого графа [2].

Алгоритм кластеризации k-MXT-W.

В работе [3] была представлена модификация алгоритма кластеризации k-MXT. Модификация алгоритма была названа алгоритм k-MXT-W.

Алгоритм k-MXT-W отличается от алгоритма k-MXT способом подсчета весов дуг. Вес дуги (q, r) равен количеству общих соседей вершины q и вершины r умноженному на значение функции плотности распределения Гаусса в точке x, равной расстоянию между вершинами q и r; $\sigma = \frac{\varepsilon}{3}$; $\mu = 0$.

$$N(x,\sigma,\mu) = \frac{1}{\sigma\sqrt{2\pi}} \times e^{\frac{-(x-\mu)^2}{2\sigma^2}}, \qquad (1)$$

где N – функция плотности распределения Гаусса

Поскольку была использована плотность распределения Гаусса, по правилу трех сигма для полного покрытия всех точек, расположенных на расстоянии ε от заданной вершины, достаточно взять $\sigma = \frac{\varepsilon}{3}$

Метрики для кластеризации данных.

Для построения графа G, необходимого для алгоритмов кластеризации k-MXT и k-MXT-W, следует определить каким способом наиболее оптимально считать расстояние (степень похожести) между объектами в зависимости от характера данных. Расстояние (степень схожести) между объектами данных подсчитывается с помощью выбранной метрики.

Метрика - это функция для подсчета расстояния между объектами некоторого множества X, которое считается с помощью функции ρ. Функция ρ определена на декартовом произведении $X \times X$ и для любых $x, y \in X$ выполняются следующие аксиомы:

– $\rho(x,y) = 0$, если $x = y$ - аксиома тождества,

– $\rho(x,y)+\rho(y,z)=\rho(x,z)$ - аксиома треугольника,

– $\rho(x,y)=\rho(y,x)$- аксиома симметрии.

При решение задач необходимо выбирать метрику таким образом, чтобы объекты данных, которые меньше похожи между собой (расположены на большем расстоянии) имели более высокое значение метрики, а объекты данных, которые больше похожи между собой (расположены на меньшем расстоянии) имели меньшее расстояние (значение метрики).

Наиболее распространенной метрикой для подсчета расстояний между объектами является Евклидова метрика $\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}$.

Нормализация данных с различными единицами измерений.

При выборе метрики для определения расстояния между объектами (степени похожести между объектами) необходимо определить какие признаки объектов данных имеют наибольшее влияние. При кластеризации данных с различными единицами измерений между признаками необходимо устранить влияние этих единиц измерения. Все признаки данных должны быть нормированы по формуле $X'=\frac{X-E}{\sigma}$, где X' - нормированное значение признака объекта, X - исходное значение признака, E - среднее значение всех объектов данных по заданному признаку, σ- стандартное отклонение всех объектов по заданному признаку.

Этот подход был применен в статье [4] в задаче выделения климатических зон для нормирования объектов данных содержащие признаки: осадки, максимальная и минимальная температура и т.д.

Описание функционала пакета k-MXT-W3.

Пакет k-mxt-w3 [5] был опубликована на сайте [6] и может быть установлен с помощью команды в терминале *pip install k-mxt-w3*.

Данный пакет содержит реализацию алгоритмов k-MXT и k-MXT-W и вспомогательные модули для подготовки данных для кластеризации. В последней версии 0.0.2 алгоритмы могут выполнить кластеризацию данных любой размерности, предварительно данные будут нормализованы по формуле нормализации данных с различными единицами измерений. В качестве функции подсчета расстояний между объектами используется метрика Евклида. В дальнейших версиях пакета планируется, добавить другие метрики для подсчета расстояний между объектами, а также добавить возможность использования пользователем собственных метрик для подсчета расстояний.

В пакете k-mxt-w3 реализация функции поиска в глубину (DFS) реализована не рекурсивным способом, что позволяет использовать алгоритмы кластеризации k-MXT и k-MXT-W не зависимо от характера кластеризуемых данных. В предыдущих реализациях алгоритмов для данных с большим количеством вершин в одном кластере, программа прекращала работу из-за превышения лимита по памяти.

Пакет состоит из следующих модулей:

– clustering_algorithms, который содержит алгоритмы кластеризации данных k-MXT, k-MXT-W;

– clusters_data, содержит класс, в котором задается метрика для подсчета расстояний между объектами данных, нормализуются признаки данных, находится массив признаков данных и для каждого объекта данных записывается номер кластера;

– data, содержит классы для получения признаков данных, по которым будет производиться кластеризация данных;

– graph ищет компоненты сильной связности графа, данный модуль вызывается из модуля clustering_algorithms.

Пример использования пакета k-mxt-w3 представлен в листинге.

```
import k_mxt_w3.clustering_algorithms
import k_mxt_w3.clusters_data
import k_mxt_w3.data
import pandas as pd

filename = 'dataset.csv'
df = pd.read_csv(filename)                                  # reading csv-file data
dataset = k_mxt_w3.data.DataPropertyImportSpace(df=df)
features_list = ['column_x']                                # list of the dataset columns
x, y, features = dataset.get_data(name_latitude_cols='latitude',    # name of the column containing the
latitude values
                                  name_longitude_cols='longitude',  # name of the column containing the longi-
tude values
                                  features_list=features_list)      # list of others
clusters = k_mxt_w3.clusters_data.ClustersDataSpaceFeaturesEuclidean(x_init=x,
                                                                     y_init=y,
                                                                     features_init=features)  # creating an object
                                                                                              # containing clusters
alg = k_mxt_w3.clustering_algorithms.K_MXT_gauss(k=5, eps=0.05, clusters_data=clusters)      #
creating an object
                                                                                              # with k=5, eps=0.05
alg()                                  # run the clustering algorithm
print(clusters.cluster_numbers)     # print cluster number for each vertex
```

Пример использования пакета k-mxt-w3

Заключение.

В данной работе был создан пакет k-mxt-w3 для языка Python, содержащий реализацию алгоритмов кластеризации k-MXT и k-MXT-W, а также вспомогательные модули для подготовки данных для кластеризации. С помощью пакета может быть выполнена кластеризация данных любой размерности, предварительно данные будут нормализованы по формуле нормализации данных с различными единицами измерений. Так же данный пакет можно применять к данных, которые будут содержать большое количество вершин в одном кластере, так как был рекурсивная реализация обхода в глубину был заменена

не рекурсивной реализацией. Что позволяет использовать данный пакет для любого набора данных.

СПИСОК ЛИТЕРАТУРЫ

1. *Cooper, C.* An experimental study of the k-mxt algorithm with applications to clustering geo-tagged data / C. Cooper, N. Vu. 2018. Vol. 10836. Pp. 145–169. doi: https://doi.org/10.1007/978-3-319-92871-5_10

2. *Cormen, T. H.* Introduction to Algorithms, Third Edition / T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein. 3rd. edition. The MIT Press, 2009. Pp. 587–623.

3. Stepanova A., Mironov S. V., Sidorov S., Faizliev A. Modification of the k-mxt algorithm and its application to the geotagged data clustering // Springer International Publishing. 2019. Pp. 296–307.

4. *Sathiaraj D. Huang X., Chen J.* Predicting climate types for the continental united states using unsupervised clustering techniques // Environmetrics. 2019. Vol. 30, no. 4. Pp. 1–12.

5. k-mxt-w3 PyPI. [Электронный ресурс]. URL: https://pypi.org/project/k-mxt-w3/ (дата обращения: 08.06.2020).

6. PyPI The Python Package Index. [Электронный ресурс]. URL: https://pypi.org/ (дата обращения 20.04.2020).