МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**
Кафедра математической кибернетики и компьютерных наук

# РАЗРАБОТКА СРЕДСТВ ДЛЯ КЛАСТЕРИЗАЦИИ ДАННЫХ В МНОГОМЕРНОМ ПРОСТРАНСТВЕ

АВТОРЕФЕРАТ МАГИСТЕРСКОЙ РАБОТЫ

Студентки 2 курса 273 группы
направления 02.04.03 — Математическое обеспечение и администрирование информационных систем
факультета КНиИТ
Степановой Анастасии Александровны

Научный руководитель
доцент, к. ф.-м. н. _____ Ю. Н. Кондратова

Заведующий кафедрой
к. ф.-м. н., доцент _____ С. В. Миронов

Саратов 2021

# ВВЕДЕНИЕ

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. Любой алгоритм кластеризации в результате работы присваивает каждому объекту данных номер кластера, которому он принадлежит. Для кластеризации не требуются предварительные знания о наборе данных. Такое понятие кластеризации вводят авторы Andreas C. Mueller и Sarah Guido в книге «An Introduction to Machine Learning with Python».

В настоящее время существует большое количество информации, которую можно обработать с помощью алгоритмов кластеризации. Кластеризация данных используется, для выделения групп общения по данным из социальных сетей, выделения групп товаров или групп объектов по геотегам, для группировки событий и т. д.

В работе [1] была проведена пространственная кластеризация гео-данных, полученных с сайта Flickr, для выделения популярных объектов Лондона. Геоданными являются данные, которые содержат информацию о долготе и широте объектах данных.

В статье [2] была представлена модификация алгоритма кластеризации $k$-MXT [1]. Модифицированный алгоритм был назван алгоритм $k$-MXT-W. С помощью алгоритмов кластеризации $k$-MXT, $k$-MXT-W были выделены популярные места города Санкт-Петербург.

Во всех вышеперечисленных работах кластеризация данных применялась к данных в многомерном пространстве. Данными в многомерном пространстве являются данные, объекты которых расположены в двухмерном или более пространстве.

Разные алгоритмы кластеризации, по разному выполняют кластеризацию на разных типах данных. В задаче кластеризации данных очень важно выбрать алгоритм кластеризации, который соответствует типу данных.

Помимо исследования структуры данных для кластеризации, чтобы легко проверить качество работы какого-либо алгоритма кластеризации на этом типе данных, можно применить выбранный алгоритм кластеризации на данных и

изучить результат кластеризации. Для этого необходимы средства для кластеризации данных, которые позволяют быстро выполнить кластеризацию. Многие алгоритмы кластеризации, например DBSCAN [3], $k$-Means [4], Mean-shift [5] и прочие, реализованы в библиотеке scikit-learn для языка Python. Алгоритмы $k$-MXT, $k$-MXT-W являются новыми алгоритмами и не имеют реализации в известных пакетах языков программирования. Так же наличие пакетов, которые содержат реализации алгоритмов кластеризации, позволяет не реализовывать алгоритмы самостоятельно, а использовать проверенные реализации этих алгоритмов, что сокращает число ошибок при кластеризации.

Целью данной работы является создание средств для кластеризации данных в многомерном пространстве алгоритмами кластеризации $k$-MXT, $k$-MXT-W. Для достижения данной цели необходимо решить следующие задачи:

- реализовать алгоритмы кластеризации $k$-MXT и $k$-MXT-W;
- провести сравнительный анализ алгоритмов $k$-MXT и $k$-MXT-W, а так же сравнить эти алгоритмы с другими алгоритмами кластеризации;
- создать пакет для языка программирования Python, содержащий алгоритмы кластеризации $k$-MXT и $k$-MXT-W, и разместить его в Python Package Index (возможна установка с помощью команды PIP);
- протестировать пакет для языка Python, содержащий алгоритмы кластеризации
- реализовать web-приложение, которое при загрузке файла с многомерными гео-данными выдает результат кластеризации алгоритмами $k$-MXT или $k$-MXT-W по выбору и отображает объекты данных на карте.
- протестировать web-приложение для кластеризации многомерных гео-данных.

Данная работа состоит из 5 разделов, введение, заключения, списка используемых источников и 10 приложений. Общий объем работы 98 страниц, из них 55 страниц — основное содержание, список используемых источников 38 наименований.

**Основное содержание работы**

Первый раздел «Описание алгоритмов кластеризации». В нем описываются популярный алгоритм кластеризации $k$-Means, алгоритм кластеризации $k$-MXT, который был описан в 2018 в работе [1] и модификация алгоритма $k$-MXT — алгоритм кластеризации $k$-MXT-W [2].

В следующем разделе «Метрики для кластеризации данных» описываются использованные метрики для подсчета расстояний между объектами и метрики оценки качества кластеризации. В качестве метрик для подсчета расстояний между объектами данных были описаны метрика Евклида и Манхэттенское расстояние. Выбор метрики сильно влияет на результат кластеризации. Для того, чтобы оценить качество, полученной кластеризации, используются метрики для оценки качества кластеризации. Эти метрики делятся на две группы. К первому типу относятся метрики оценки качества кластеризации, предполагающие знание истинной кластеризации. Примером таких оценок качества кластеризации является метрика ARI (Adjusted Rand index) [6]. Вторым типом являются метрики оценки качества кластеризации, не требующие знания истинной кластеризации. Примером таких оценок являются метрика Modularity, [7]. Так же в этом разделе описан метод нормализации данных. При кластеризации данных с различными единицами измерений между признаками необходимо устранить влияние этих единиц измерения. Например, если один из признаков данных представлен в метрах, а другой в километрах, то признак данных, указанный в метрах будет оказывать большее влияние при подсчете расстояний между объектами данных, что может привести к неправильному результату кластеризации.

В разделе «Сравнение алгоритмов кластеризации» проводится сравнение алгоритмов $k$-Means, $k$-MXT, $k$-MXT-W на смоделированных данных, полученных из библиотеки scikit-learn. Алгоритм $k$-Means плохо справляется с кластеризацией некоторых типов данных, а алгоритмы $k$-MXT, $k$-MXT-W хорошо справляются с кластеризацией рассмотренных типов данных. В тоже самое время алгоритм $k$-MXT-W лучше работает на рассмотренных типах данных, чем алгоритм $k$-MXT.

В разделе «Описание функционала пакета `k-mxt-w3`» описан, созданные в данной работе, пакет для языка Python. Данный пакет содержит реализацию алгоритмов k-MXT и k-MXT-W и вспомогательные модули для подготовки дан-

ных для кластеризации. Алгоритмы могут выполнять кластеризацию данных в двумерном и более пространстве, предварительно данные будут нормализованы по формуле нормализации данных с различными единицами измерений. В качестве функции подсчета расстояний между объектами можно выбрать метрику Евклида или можно еще использовать Манхэттенское расстояние. Пакет `k-mxt-w3` опубликован в Python Package Index, что упрощает установку пакета. Пакет может быть легко установлен с помощью команды в терминале `pip install k-mxt-w3`. Библиотека работает корректно, для версии Python 3.8.* и выше. Было выпущено две рабочии версии пакета 0.0.2 и 1.0.7, которые не являются обратно совместимыми.

Пакет состоит из следующих модулей:

— `data`, содержит классы для получения признаков данных, по которым будет производиться кластеризация данных. В модуле реализован функционал для загрузки данных из файла, содержащий эти данные;

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

— `graph` ищет компоненты сильной связности графа с помощью обхода в глубину (DFS), данный модуль вызывается из модуля `clustering_algorithms`. Поиск компонент сильной связности необходим для работы алгоритмов $k$-MXT, $k$-MXT-W.;

— `clustering_algorithms`, который содержит алгоритмы кластеризации данных $k$-MXT, $k$-MXT-W.

Также в пакете `k-mxt-w3` функция поиска в глубину (DFS), которая используется при нахождении кластеров алгоритмами $k$-MXT, $k$-MXT-W, реализована не рекурсивным способом, что позволяет использовать эти алгоритмы кластеризации независимо от характера и объема 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 `k-mxt-w3` протестирован. Процент покрытия кода тестами составляет $85\%$, что свидетельствует о стабильной работе пакета `k-mxt-w3`.

В разделе «Web-приложение для кластеризации данных в многомерном пространстве» описано созданное в данной работе web-приложение для класте-

ризации многомерных геоданных (данные, содержащие информацию о широте и долготе объектов данных). Web-приложение было написано на языке Python с использованием фреймворка Django для кластеризации данных в многомерном пространстве. Django — бесплатный, с открытым исходным кодом, высоуровневый фреймворк для разработки web-приложений, который использует паттерн Model-View-Controller (MVC). MVC определяет способ разработки программного обеспечения, в котором код разделен на отдельные части [8]:

— Файл `models.py` содержит описание таблиц базы данных с помощью Python классов, которые называются моделями. Используя модели, можно создавать, извлекать, обновлять и удалять записи в базе данных, используя простой Python код, вместо использования SQL шаблонов;
— Файл `views.py` содержит бизнес-логику страниц, функции описывающие бизнес-логику называются `view`;
— Файл `urls.py` определяет какая view-функция будет вызвана для заданной URL шаблона;
— Так же существуют html-шаблоны, которые описывают дизайн страницы.

Web-приложение позволяет загружать файл, содержащий числовые признаки объектов данных, по которым будет выполняться кластеризация. Пользователю необходимо ввести параметры для алгоритмов кластеризации, выбрать один из алгоритмов кластеризации (или $k$-MXT, или $k$-MXT-W) и выбрать признаки, по которым будет выполняться кластеризация. После подсчета результатов кластеризации, будет подсчитано значение метрики Modularity для полученных результатов кластеризации. Также будет доступна ссылка для загрузки результатов кластеризации в виде json-файла. Помимо возможности загрузки результатов кластеризации и вывода значения метрики Modularity для полученных результатов, приложение выводит две карты. На первой карте отображен разброс точек данных в двумерном пространстве. На второй карте показан результат кластеризации данных по выбранным признакам. Цвет точек в данном случае будет зависеть от номера кластера каждой точки. Для упрощения процесса разворачивания web-приложения, был создан Dockerfile. Docker — платформа для разработки, распространения и запуска приложений. Docker предоставляет возможность упаковывать и запускать приложения в изолированном окружении, которое называется container. Объекты container имеют небольшой объем и содержат все, что необходимо для запуска приложения, независимо от того,

что установлено на компьютере или сервере. В работе указана инструкция по разворачиванию web-приложения для кластеризации данных. Web-приложение для кластеризации данных было протестировано. Покрытие тестами приложения составляет $70\%$, что свидетельствует о стабильной работе web-приложения.

## ЗАКЛЮЧЕНИЕ

При сравнении алгоритмов кластеризации $k$-MXT и $k$-MXT-W видно, что алгоритм $k$-MXT-W лучше справляется с кластеризацией рассмотренных типов данных, чем алгоритм $k$-MXT. Алгоритмы $k$-MXT, $k$-MXT-W лучше справляются с кластеризацией данных типов moons и circles, чем алгоритм $k$-Means.

Алгоритмы $k$-MXT, $k$-MXT-W, в отличие от других популярных алгоритмов кластеризации, не реализованы в популярных пакетах языка Python, например в пакете scikit-learn. Для упрощения использования алгоритмов кластеризации $k$-MXT, $k$-MXT-W в данной работе был создан пакет `k-mxt-w3`, который содержит реализации алгоритмов $k$-MXT, $k$-MXT-W. Пакет был опубликован в Python Package Index и может быть легко установлен. Пакет `k-mxt-w3` был протестирован, процент покрытия тестами кода составляет 85%, что является высоким процентом и означает, что пакет `k-mxt-w3` работает стабильно.

Для получения результатов кластеризации алгоритмами $k$-MXT, $k$-MXT-W, не тратя времени для изучения кода пакета `k-mxt-w3`, в данной работе было создано приложения для кластеризации гео-данных в многомерном пространстве, которое принимает csv-файл с данными и параметры кластеризации, и выдает результат кластеризации, значение метрики оценки качества кластеризации и отображает объекты данных и результаты кластеризации на карте. Приложение было протестировано. Процент покрытия тестами кода составляет 70%, что свидетельствует о стабильной работе приложения для кластеризации гео-данных в многомерном пространстве.

Данная работа была представлена на следующих конференциях:

— IX Международная научно-практическая конференция «Математическое и компьютерное моделирование в экономике, страховании и управлении рисками», 2020 г.

— Студенческая научная конференция факультета КНиИТ, 24.04.2020, доклад «Модификация алгоритма k-MXT с помощью оконных функций»

А также по теме данной работы были опубликованы следующие статьи:

— Stepanova A., Mironov S.V., Sidorov S., Faizliev A. Modification of the k-MXT Algorithm and Its Application to the Geotagged Data Clustering. Lecture Notes in Computer Science, 2019, Vol. 11943. P. 296-307. SCOPUS, WoS [2]

— Степанова А. А., Сидоров С. П., Балаш В. А. Сервис для кластеризации данных с геотегами алгоритмом k-MXT-w // Математическое и компью-

терное моделирование в экономике, страховании и управлении рисками. 2020. № 5. С. 130-134 [9].

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ


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 *Stepanova, A.* Modification of the k-mxt algorithm and its application to the geotagged data clustering / A. Stepanova, S. V. Mironov, S. Sidorov, A. Faizliev // *Springer International Publishing*. — 2019. — Pp. 296–307. — doi: https://doi.org/10.1007/978-3-030-37599-7_25.

3 *Ester, M.* A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise // Proceedings of the Second International Conference on Knowledge Discovery and Data Mining KDD'96. — AAAI Press, 1996. — Pp. 226–231. http://dl.acm.org/citation.cfm?id=3001460.3001507.

4 *Lloyd, S.* Least squares quantization in pcm's. / S. Lloyd // *Bell Telephone Laboratories Paper*. — Vol. 28. — Pp. 129–137.

5 *Fukunaga, K.* The estimation of the gradient of a density function, with applications in pattern recognition / K. Fukunaga, L. Hostetler // *IEEE Transactions on Information Theory*. — 1975. — Vol. 21, no. 1. — Pp. 32–40.

6 *Rand, W.* Objective criteria for the evaluation of clustering methods / W. Rand // *Journal of the American Statistical Association*. — 1971. — Vol. 66. — Pp. 846–850.

7 *Brandes, U.* On modularity clustering / U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, D. Wagner // *IEEE Transactions on Knowledge and Data Engineering*. — 2008. — Vol. 20, no. 2. — Pp. 172–188.

8 *Holovaty, A.* The Definitive Guide to Django: Web Development Done Right / A. Holovaty, J. K. Moss. — Apress, 2008.

9 *Степанова, А. А.* Сервис для кластеризации данных с геотегами алгоритмом k-mxt-w / А. А. Степанова, С. П. Сидоров, В. А. Балаш // *«Математическое и компьютерное моделирование в экономике, страховании и управлении рисками»: материалы IX Международной научно-практической конференции*. — 2020. — Pp. 130–134.