МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Тульский государственный университет»
Институт прикладной математики и компьютерных наук
Кафедра Информационной безопасности

УТВЕРЖДАЮ
Зав. кафедрой ИБ
_____ А.А. Сычугов
«___» _____ 20___ г.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе по дисциплине

ТЕОРИЯ ВЕРОЯТНОСТЕЙ И МАТЕМАТИЧЕСКАЯ СТАТИСТИКА

на тему

«Кластеризация и её применение»

Автор работы _____ студент гр. 230711 Павлова В.С.
(дата, подпись) (фамилия и инициалы)

Руководитель работы _____ доц. каф. ПМиИ Родионова Г.А.
(дата, подпись) (должность) (фамилия и инициалы)

Работа защищена _____ с оценкой _____
(дата)

Члены комиссии _____ _____ _____
(дата, подпись) (должность) (фамилия и инициалы)

_____ _____ _____
(дата, подпись) (должность) (фамилия и инициалы)

_____ _____ _____
(дата, подпись) (должность) (фамилия и инициалы)

Тула 2023

ЗАДАНИЕ

на курсовую работу по дисциплине

«Теория вероятностей и математическая статистика»

студента гр. №230711 Павловой Виктории Сергеевны

Тема курсовой работы:

«Кластеризация и её применение».

Исходные данные:

Классификация и кластер. Под ред. Дж. Вэн Райзина. М.: Мир, 1980. 390 с.

Задание получил _____ _____
(ФИО) (подпись)

Задание выдал _____ _____
(ФИО) (подпись)

Дата выдачи задания 18 октября 2023 г.

График выполнения КР _____

в соответствии с методическими указаниями.

Рекомендации и особые отметки _____

«__» _____20__г

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

С развитием информационных технологий во всех областях науки и техники стали появляться всё большие объёмы данных, вследствие чего приоритетной стала задача их анализа и систематизации.

В математической статистике применяются различные методы решения данной задачи: регрессионный анализ, факторный анализ, методы описательной статистики, кластерный анализ. Последний метод, представляющий собой технику разделения объектов на группы (кластеры) в зависимости от их сходства, является одним из эффективно используемых в современности подходов.

Истоки кластерного анализа уходят в глубину истории развития статистики и психологии. Ранние попытки систематизации данных были предприняты Френсисом Гальтоном в XIX веке при исследованиях в области наследственности и психологии, однако полноценное развитие методов кластерного анализа произошло только в середине XX века в контексте многомерного анализа данных. С появлением компьютеров стало возможным проведение более сложных вычислений, что способствовало активному развитию методов кластерного анализа. С течением времени они становились всё более сложными и адаптированными к разнообразным видам данных: иерархическая кластеризация, метод k-средних, DBSCAN и другие методы были разработаны и усовершенствованы для более эффективной обработки больших объёмов данных и учёта различий в структуре кластеров.

С развитием машинного обучения и технологий обработки BigData, кластерный анализ стал неотъемлемой частью инструментария аналитиков и исследователей. Новые подходы, такие как кластерный анализ графов, позволяют расширить область применения метода, внося инновации в анализ социальных и производственных сетей.

Эволюция кластерного анализа продолжается, а его методы становятся все более точными, быстрыми и универсальными, что позволяет применять его в

различных областях, в частности, в сфере информационной безопасности, где он является мощным инструментом для выявления паттернов, обнаружения аномалий и классификации данных об инцидентах.

В настоящей курсовой работе в рамках задачи теоретического исследования рассматриваются такие методы и применения кластерного анализа, как иерархическая кластеризация, метод k-средних и DBSCAN. В рамках исследования практического применения методов кластеризации рассмотрены их прикладные применения в контексте решения задач обеспечения и предотвращения инцидентов информационной безопасности автоматизированных систем. Актуальность выбранной тематики обоснована возможностью эффективно анализировать и интерпретировать большие объемы данных с помощью кластеризации, повышая общий уровень безопасности автоматизированной системы и обеспечивая быстрый отклик на потенциальные угрозы.

1 ИССЛЕДОВАНИЕ ТЕОРЕТИЧЕСКОЙ ОСНОВЫ КЛАСТЕРНОГО АНАЛИЗА

1.1 Основные понятия и алгоритмы

Пусть дано исходное множество объектов $X = \{X_1, \dots, X_n\}$ в некотором пространстве признаков, размерность которого равна количеству признаков m и набор характеристик $y = \{y_1, \dots, y_m\}$. Тогда информация о признаках представима в виде матрицы вида (1):

$$\begin{array}{c} \\ y_1 \\ \vdots \\ y_j \\ \vdots \\ y_m \end{array} \begin{array}{c} \begin{array}{ccc} X_1 & \dots & X_i & \dots & X_n \end{array} \\ \begin{pmatrix} x_{11} & \cdots & x_{1m} \\ \\ \vdots & x_{ji} & \vdots \\ \\ x_{n1} & \cdots & x_{nm} \end{pmatrix} \end{array} \qquad (1)$$

Элемент x_{ij} представляет собой значение j-го признака, измеренное на i-ом объекте. Получено m-мерное пространство, где оси координат соответствуют отдельным признакам матрицы данных [1], и каждую строку матрицы можно представить как вектор в этом пространстве. Таким образом, каждый из объектов наблюдения представлен своей изображающей точкой в m-мерном пространстве признаков. А поскольку с помощью нормировки всегда можно расположить значения признака в пределах (0; 1), тогда все объекты представимы точками внутри m-мерного единичного гиперкуба в пространстве признаков, как показано на рисунке 1.

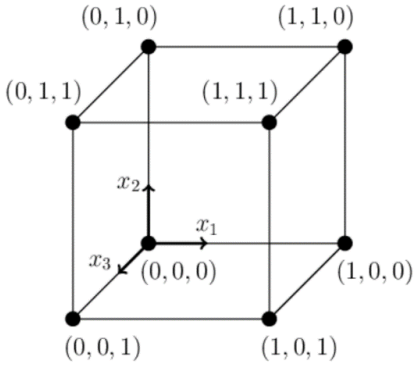

Рисунок 1 – Трёхмерное нормированное признаковое пространство

Как правило, объекты в выборке «похожи» друг на друга в различной степени, то есть у некоторых из них можно выделить общий признак. Под «похожестью» понимается близость объектов в многомерном пространстве признаков, и тогда задача анализа полученных наблюдений сводится к выделению в этом пространстве естественных скоплений объектов, которые и считаются однородными группами [2].

В основе различных методов анализа матрицы данных лежит неформальное предположение, условно названное «гипотезой компактности», согласно которой каждый класс должен занимать относительно компактную область в m-мерном пространстве, таким образом чтобы и различные классы оказывались отделимыми друг от друга сравнительно простыми поверхностями, в идеале – гиперплоскостями. Под **гиперплоскостью** в m-мерном пространстве понимают n-мерную плоскость, где $n = (m - 1)$ [3].

Предполагается, что всё множество объектов X представимо в виде небольшого числа подмножеств, внутри которых объекты наблюдения объединены этим общим признаком. Таким образом, ставится задача разбиения исходного множества на конечное число классов. Этим и занимается **кластерный анализ** — многомерная статистическая процедура [4], включающая в себя сбор данных о выборке объектов и упорядочивание их в сравнительно однородные группы.

Основным понятием, которым оперирует кластерный анализ, является **кластер** — группа или класс элементов, характеризуемых общим свойством, а главной целью кластерного анализа является нахождение групп схожих, то есть однородных объектов в выборке.

Стоит различать задачи **кластеризации** и **классификации**. Главной целью кластерного анализа является выявление структуры и отношения между данными путём выделения групп схожих элементов в данных. В кластерном анализе это происходит без использования заранее заданных классов или признаков групп.

Задача классификации же заключается в присвоении объекту одной из заранее определенных категорий или классов.

При решении задач кластеризации используют иерархические и итеративные алгоритмы [5]. **Иерархическая кластеризация**, построенная на вероятностно-статистическом подходе, предполагает, что каждый кластер есть реализация некоторой случайной величины, и к ней относят следующие методы:

- **Агломеративные или восходящие алгоритмы.**

Данный подход исходит из того, что изначально каждый объект является кластером, а вероятностные модели используются для описания структуры данных и связей между объектами. На каждой итерации алгоритма происходит объединение двух ближайших кластеров, как показано на рисунке 2, что позволяет образовывать все более крупные группы.

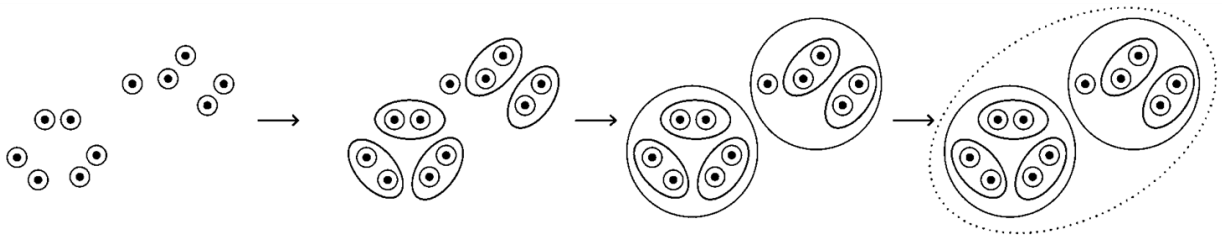

Рисунок 2 – Шаги агломеративного алгоритма кластеризации

Графическое изображение процесса объединения кластеров [6] также может быть представлено с помощью дендрограммы — дерева объединения кластеров, пример которого представлен на рисунке 3.

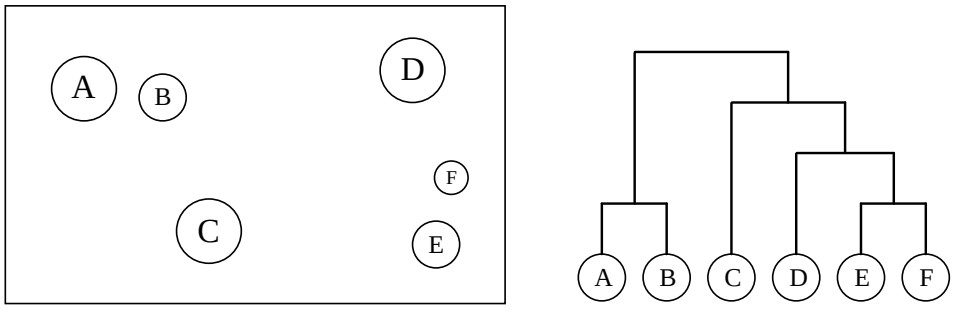

Рисунок 3 – Графическое изображение процесса иерархической кластеризации данных (дендрограмма)

К агломеративным иерархическим алгоритмам относятся [7]:

- Кластеризация с одной связью или метод одиночной связи (англ. single-linkage clustering), в рамках которой близость между кластерами определяется как минимальное расстояние между всеми парами точек, где одна точка принадлежит первому кластеру, а другая – второму;
- Метод полной связи (англ. complete-linkage clustering), где близость кластеров устанавливается как максимальное расстояние между точками кластера;
- Метод средней связи (англ. average-linkage clustering), в свою очередь, использует среднее расстояние между всеми парами точек в разных кластерах для определения близости.
- Метод Уорда (Варда, англ. Ward's method), который стремится минимизировать увеличение дисперсии внутри кластеров при объединении. На каждом шаге метода Уорда выбираются два кластера, которые минимизируют увеличение дисперсии после объединения (для вычисления дисперсии используются методы дисперсионного анализа или ANOVA).

- **Дивизивные (дивизионные) или нисходящие алгоритмы.**

При таком подходе объекты сначала помещают в один кластер, как показано на рисунке 4, а потом постепенно разделяют на кластеры всё меньшего и меньшего размера.

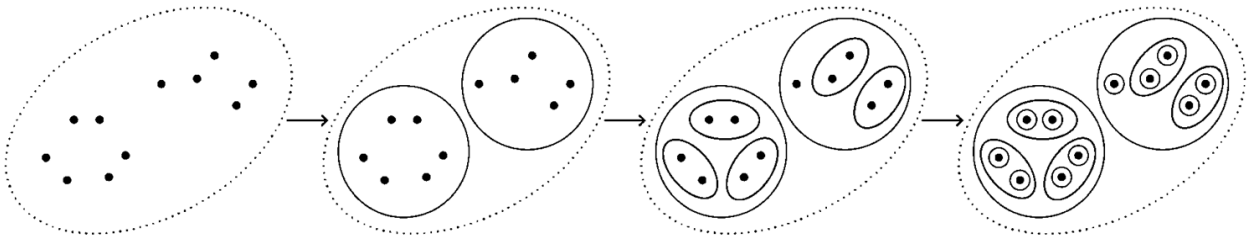

Рисунок 4 – Шаги работы дивизионного алгоритма кластеризации

Примерами дивизивных алгоритмов являются [7]:

- DIANA (от англ. divisive analysis): Начинает с одного кластера и разделяет его на более мелкие части, выбирая объекты, которые наименее похожи на остальные.
- BIRCH (англ. Balanced Iterative Reducing and Clustering using Hierarchies):

Итеративные алгоритмы [7] кластеризации основаны на повторяющихся итерациях, в ходе которых объекты присваиваются кластерам, а параметры кластеров обновляются в соответствии с выбранной мерой близости или расстояния. Примерами таких алгоритмов являются:

- Метод k-средних (англ. K-means), где кластеры строятся на основе математической формулы среднеквадратичной ошибки. Данный метод более подробно рассматривается в пункте 3 главы 1 настоящей курсовой работы.
- EM-алгоритм (англ expectation-maximization), применяемый в задачах, где данные имеют скрытую структуру.

Помимо вышеперечисленных методов, выделяют также системы искусственного интеллекта, которые позволяют разделить объекты с помощью нейронных сетей (чаще всего они применяются, когда число кластеров неизвестно), и логический подход, при котором данные делят по кластерам с помощью дерева решений. К нейросетевым подходам [4] относится самоорганизующаяся карта (англ. SOM) – нейронная сеть, которая может использоваться для кластеризации данных, обучаясь адаптировать свою топологию к структуре данных. Деревья решений [8] используются в алгоритме C4.5, который может использоваться для разделения данных на кластеры на основе логических правил, а также CHAID (англ. chi-square Automatic Interaction Detection), который может применяться для кластеризации данных на основе статистических тестов.

Существует также такой алгоритм, как DBSCAN [9], который не является ни иерархическим (агломеративным или дивизивным), ни итеративным алгоритмом. Он рассматривается в 5 пункте главы 1 настоящей курсовой работы. Здесь же отметим, что вместо того, чтобы разделять или объединять кластеры, DBSCAN

формирует кластеры на основе плотности данных внутри некоторой эпсилон-окрестности точек, что делает его более гибким для обнаружения кластеров произвольной формы и способным обрабатывать шум в данных.

Меры расстояния в кластерном анализе

В кластерном анализе для измерения степени близости или удаленности между объектами данных используются различные меры подобия. Выбор конкретной меры (**метрики**) будет зависеть от природы данных и задачи кластеризации. Пусть в нашем m-мерном пространстве заданы две точки $X_p = \{x_{p1}, \ldots, x_{pm}\}$, и $X_q = \{x_{q1}, \ldots, x_{qm}\}$. Расстояние между ними можно обозначить как $\rho(X_p, X_q)$ или $d(X_p, X_q)$. Тогда к числу метрик, применяемых при кластеризации данных, измеренных в количественной шкале, можно отнести [6]:

1) Евклидово расстояние (2), обозначаемое L^2 и являющееся одной из самых распространённых мер расстояний. Евклидово расстояние выступает геометрическим расстоянием в многомерном пространстве.

$$d(X_p, X_q) = \sqrt{\sum_{i=1}^{m} (x_{pi} - x_{qi})^2} \quad (2)$$

2) Квадрат евклидова расстояния (3), который используется в некоторых методах агломерации, в частности, в методе k-средних и в методе Варда (Уорда) [10].

$$d(X_p, X_q) = \sum_{i=1}^{m} (x_{pi} - x_{qi})^2 \quad (3)$$

3) Манхэттенское расстояние (4) или так называемое блочное расстояние – средняя разность по координатам.

$$d(X_p, X_q) = \sum_{i=1}^{m} |x_{pi} - x_{qi}| \quad (4)$$

4) Расстояние Чебышёва (5), которое определяется как максимальное абсолютное различие между соответствующими координатами точек:

$$d(X_p, X_q) = max(|x_{pi} - x_{qi}|) \quad (5)$$

5) Степенное расстояние (6) позволяет придавать различным измерениям различный вес в вычислении расстояния. Соответствует евклидовой метрике при степени $R = 2$. Если $R > 2$, то измерение чувствительно к малым расстояниям, а если $R < 2$, то значимыми являются большие расстояния.

$$d(X_p, X_q) = \sqrt[R]{\sum_{i=1}^{m} (x_{pi} - x_{qi})^r} \quad (6)$$

В зависимости от степени важности переменных им могут задаваться некоторые **веса** w_{X_i}, которые повлияют на состав и количество формируемых кластеров, а также степень сходства объектов внутри кластеров. Веса вводятся в качестве коэффициентов, поэтому, например, формула для вычисления евклидова расстояния в преобразованном виде будет выглядеть следующим образом (2.1):

$$d(X_p, X_q) = \sqrt{\sum_{i=1}^{m} w_{X_i} * (x_{pi} - x_{qi})^2} \quad (2.1)$$

Существуют и другие меры сходства, такие как, например, косинусное подобие, корреляция Пирсона или коэффициент Дайса. Они часто применяются в машинном обучении, обеспечивая основу для многих методов анализа данных и принятия решений, включая кластерный анализ.

1.2 Иерархическая агломеративная кластеризация

Как было упомянуто ранее, **агломеративная кластеризация** – это метод кластеризации, который строит иерархию кластеров путем последовательного объединения ближайших кластеров. В качестве меры расстояния, как правило,

принимается евклидова метрика. Алгоритм начинается с того, что каждый объект считается отдельным кластером, а затем на каждом шаге объединяются два ближайших кластера. Процесс продолжается до тех пор, пока все объекты не объединятся в один кластер или пока не достигнуто заранее заданное их число.

Кластерная **иерархия** [11] на множестве объектов X – это совокупность H вложенных подмножеств S, называемых кластерами, и удовлетворяющая следующему условию: для любых $(S_i; S_j) \in H$ их пересечение $S_i \cap S_j$ либо равно пустому множеству $\emptyset$, либо совпадает с одним из кластеров. Такие иерархии получаются путем агломерации дихотомий, то есть они являются бинарными.

Основные шаги алгоритма:

1. На начальном этапе каждый объект рассматривается как отдельный кластер.
2. Вычисляется матрица расстояний M между всеми парами кластеров или объектов в соответствии с выбранной метрикой.
3. Выбираются два ближайших кластера на основе матрицы, полученной в шаге 2.
4. Выбранные кластеры объединяются в новый кластер.
5. Матрица расстояний обновляется с учётом нового кластера.
6. Шаги 3-5 повторяются до тех пор, пока все объекты не объединятся в один кластер или до достижения заданного числа кластеров.

На рисунке 5 приведен результат выполнения данного алгоритма для случая, где в качестве множества объектов взяты различные города Тульской области. Код программы, реализующей приведённый алгоритм на языке программирования Python в программной среде Jupyter Notebook приведён в листинге 1 в приложении к данной курсовой работе. В качестве метрики было использовано евклидово расстояние, а объекты объединялись до тех пор, пока не остался лишь один кластер.

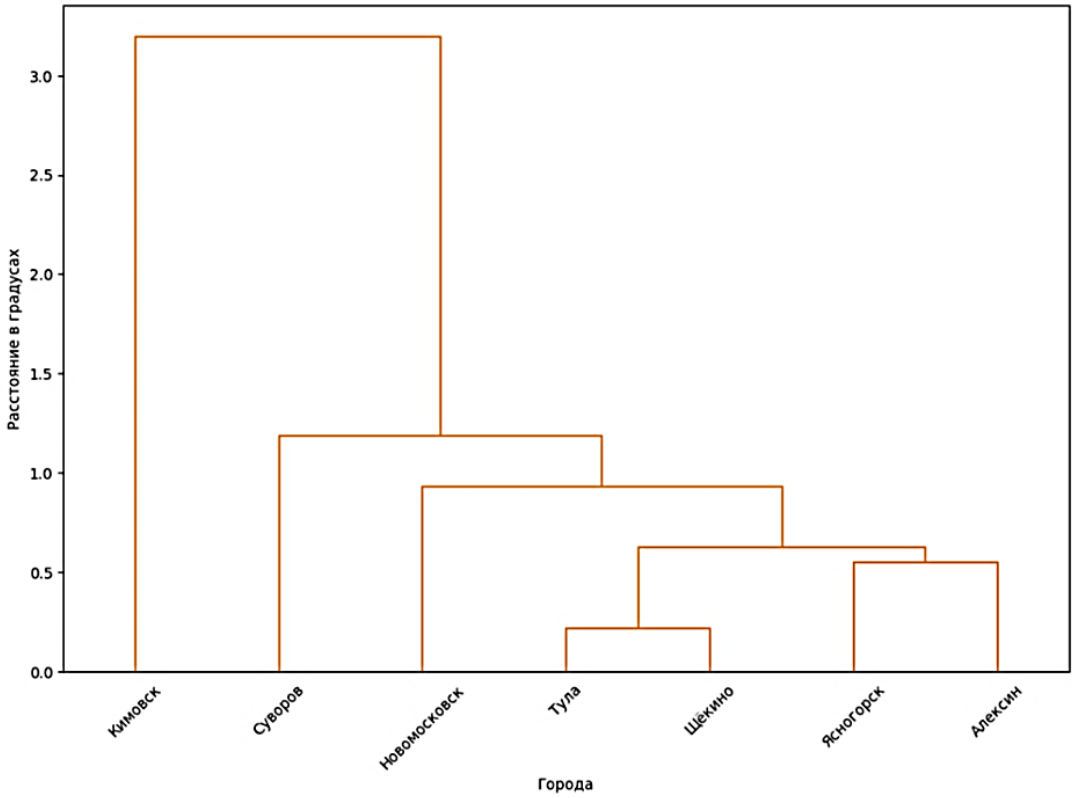

Рисунок 5 – Дендрограмма кластеризации городов Тульской области

Характеристика иерархической кластеризации представлена в таблице 1 путём сравнения достоинств и недостатков данного метода.

Таблица 1 – Характеристика иерархического метода кластеризации

Достоинства: **Наглядность** – иерархия может быть представлена наглядно с помощью дендрограммы; **Нет необходимости задавать параметр числа кластеров заранее,** поскольку алгоритм сам определяет необходимое количество кластеров.	**Недостатки**: **Вычислительная сложность** – при больших объемах данных вычисления ресурсозатратны. **Чувствительность к метрике –** результаты могут меняться в зависимости от выбора метрики расстояния. **Невозможность разделения** – обратный процесс разделения кластеров не всегда поддерживается.

1.3 Метод k-средних

В алгоритме k-средних, также известном как метод Ллойда, данные разделяются на k кластеров путем минимизации среднеквадратичного отклонения

между точками внутри кластеров и их центроидами [9]. **Центроид** – это ключевое понятие метода k-средних, он представляет собой центральную точку кластера, то есть среднее значение координат всех точек в кластере. Центроид можно назвать представительным «центром» кластера. **Мерой расстояния** в данном алгоритме обычно принимается сумма квадратов расстояний между точками и центроидами внутри каждого кластера, вычисляемое по формуле (3).

Когда речь шла об иерархическом кластерном анализе, заранее знать количество кластеров, которое нужно получить, было необязательно, однако при кластеризации методом k-средних количество искомых кластеров является необходимым параметром [12].

Основные шаги алгоритма:

1. Выбор начальных центроидов в пространстве признаков случайным образом. Выгодно инициализировать в качестве центров какие-то из объектов выборки.
2. Каждый объект выборки относят к тому кластеру, к центру которого объект оказался ближе.
3. Пересчитываются центроиды для каждого кластера как среднее значение точек в этом кластере.
4. Шаги 2 и 3 повторяются до тех пор, пока центроиды не стабилизируются или пока не будет выполнено определенное условие завершения.

На рисунке 6 показан результат выполнения данного алгоритма. В качестве выборки используется датафрейм с городами Тульской области из пункта 3. Код программы, реализующей приведённый алгоритм на языке программирования Python в программной среде Jupyter Notebook приведён в листинге 2 в приложении к данной курсовой работе. В качестве метрики было использовано евклидово расстояние, а в качестве параметра (числа кластеров) выбрано k = 5.

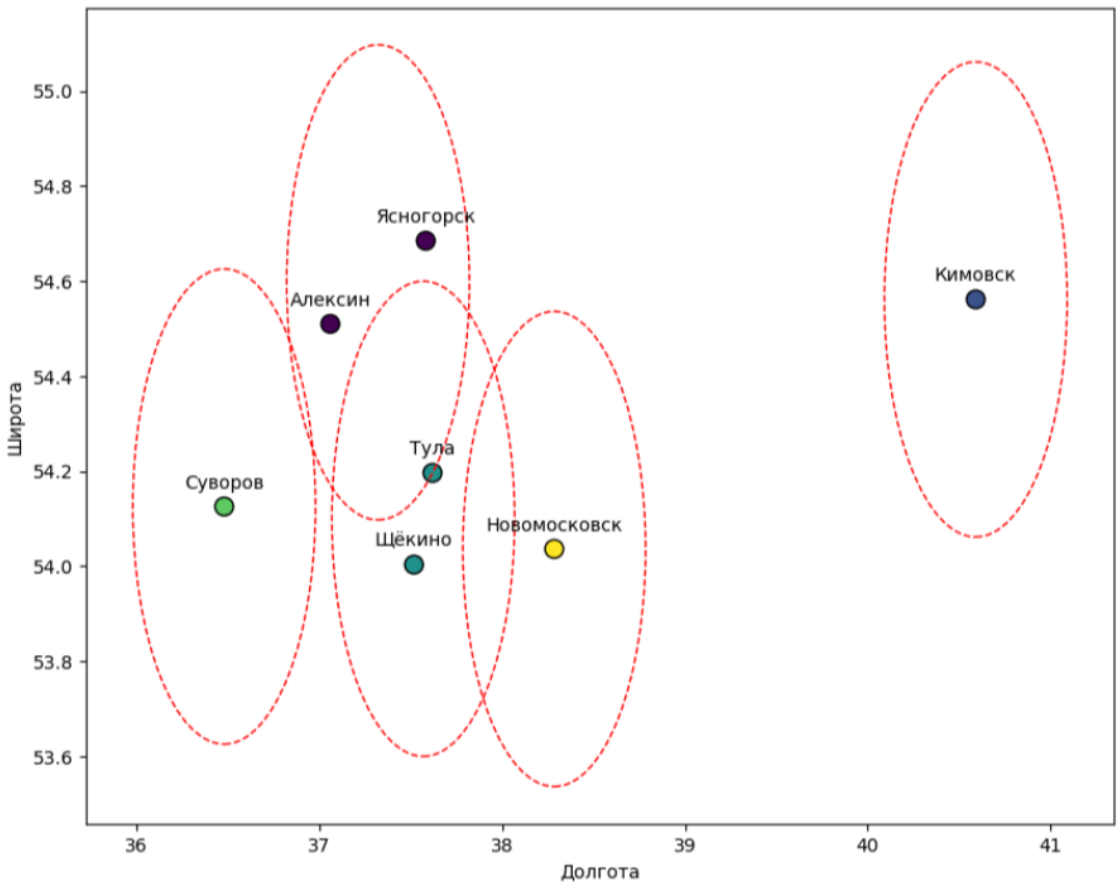

Рисунок 6 – Результат кластеризации методом k-средних

Характеристика кластеризации методом k-средних представлена в таблице 2 путём сравнения достоинств и недостатков метода.

Таблица 2 – Характеристика кластеризации методом k-средних

Достоинства: **Простота** реализации и понимании; **Эффективность** на больших наборах данных; Хорошо работает в случае кластеров примерно **одинакового размера и плотности**.	**Недостатки**: **Чувствителен** к выбросам. Требует **заранее заданное** количество кластеров (k), что не всегда известно заранее. Результаты могут зависеть от **начального выбора центроидов**. **Не гарантирует** глобальный оптимум.

Кластеризация методом k-средних является весьма распространённым способом решения задач кластерного анализа, но у него наблюдается ряд проблем, в частности, кучное размещение центров, когда их начальное положение с большой вероятностью окажется далёким от итогового положения центров кластеров [12].

Для устранения недостатков метода k-средних существуют различные эвристические усовершенствования, например, понижение размерности пространства признаков или выбор центра из случайного распределения на объектах выборки, в котором вероятность выбрать объект пропорциональна квадрату расстояния от него до ближайшего к нему центра кластера (k-means++).

1.4 DBSCAN

DBSCAN или «*density-based spatial clustering of applications with noise*» — это алгоритм кластеризации данных, который основан на понятии плотности, что позволяет ему успешно обнаруживать кластеры произвольной формы и обрабатывать шум в данных [13]. Для определения местоположения точек данных в пространстве DBSCAN использует евклидово расстояние (2), хотя возможно использование и других метрик [9].

Ключевым понятием метода является **эпсилон-окрестность**, которая определяется как множество точек U_ε, отстающих от точки X_i на расстояние, не превышающее параметр ε [13]:

$$U_\varepsilon(X_i) = \{u \in U : \rho(X_i, u) \leq \varepsilon\} \quad (7)$$

Здесь $\rho(X_i, u)$ – произвольная метрика пространства признаков. Для определённости примем её за евклидову. Параметр $\varepsilon > 0$ также задаётся исследователем. Объекты (точки) в рамках данного метода классифицируется на три группы:

- **Базовая точка –** точка, которая имеет по меньшей мере minPts соседей в пределах своей эпсилон-окрестности.
- **Граничная точка –** точка, которая находится в эпсилон-окрестности базовой, но сама не является базовой точкой (имеет менее minPts соседей).
- **Шум (noise) –** точка, которая не является ни базовой, ни граничной и не имеет достаточного числа соседей.

Теперь можно выделить **основные шаги алгоритма:**

1. Выбор начальной точки и проверка, удовлетворяет ли она условию базовой точки.
2. Распространение кластера путем добавления кластеру всех достижимых точек (базовых или граничных) в их эпсилон-окрестности.
3. Поиск новой, еще не посещенной точки, которая удовлетворяет условию базовой точки, и повторение распространения кластера.

Пример результатов работы данного алгоритма показан на рисунке 7, где в качестве датафрейма используется специфическая форма набора данных – концентрические окружности.

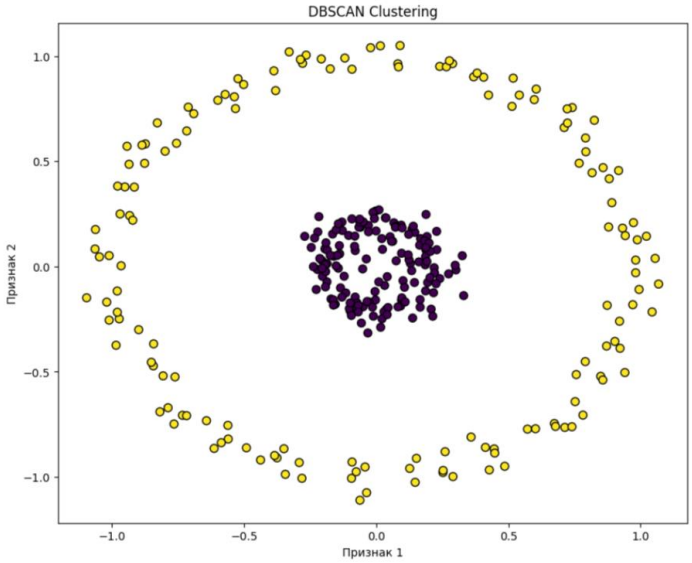

Рисунок 7 – Результат кластеризации методом DBSCAN

Код программы, реализующей данный алгоритм на языке программирования Python в программной среде Jupyter Notebook приведён в листинге 3 в приложении к данной курсовой работе. В качестве метрики использовано евклидово расстояние, в качестве параметров $\varepsilon = 0.3$, $minPts = 5$.

Несмотря на то, что впервые DBSCAN был предложен в 1996 году, он широко применяется и сейчас. Этот подход алгоритмический, напрямую не связанный с теорией вероятностей и плотностями распределений данных. Он базируется на эвристиках, предложенных авторами этого алгоритма.

Характеристика кластеризации методом DBSCAN представлена в таблице 3 путём сравнения достоинств и недостатков метода.

Таблица 3 – Характеристика кластеризации методом DBSCAN

Достоинства:	**Недостатки**:
Способен обнаруживать **кластеры произвольной формы**. Эффективен для данных с **переменной плотностью**.	**Чувствителен** к параметрам эпсилон и minPts. Может иметь проблемы с кластеризацией данных с **различной плотностью**. Не всегда хорошо обрабатывает кластеры **различных размеров**.

Стоит отметить, что в случаях, когда набор данных имеет специфическую форму, как на рисунке 7, метод DBSCAN показывает более целесообразный результат кластеризации, чем, например, метод Ллойда, как показано на рисунке 8.1. В качестве параметра метода k-средних взято k = 2.

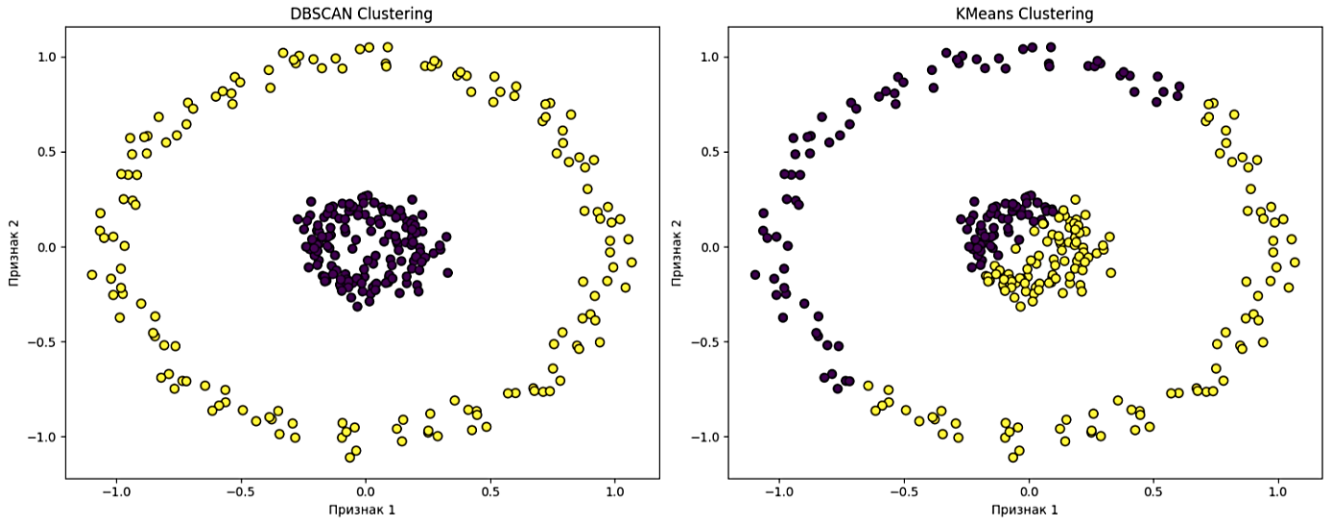

Рисунок 8 – Преимущество DBSCAN (слева) перед методом k-средних (справа) в случае необычной формы кластера

Однако, когда данные являются скоплениями объектов, именно метод k-средних показывает более приемлемые результаты (рисунок 8.2). Здесь в качестве параметра метода k-средних взято k = 3. В качестве метрики DBSCAN было

использовано евклидово расстояние, а в качестве параметров выбрано $\varepsilon = 0.5$, $minPts = 5$.

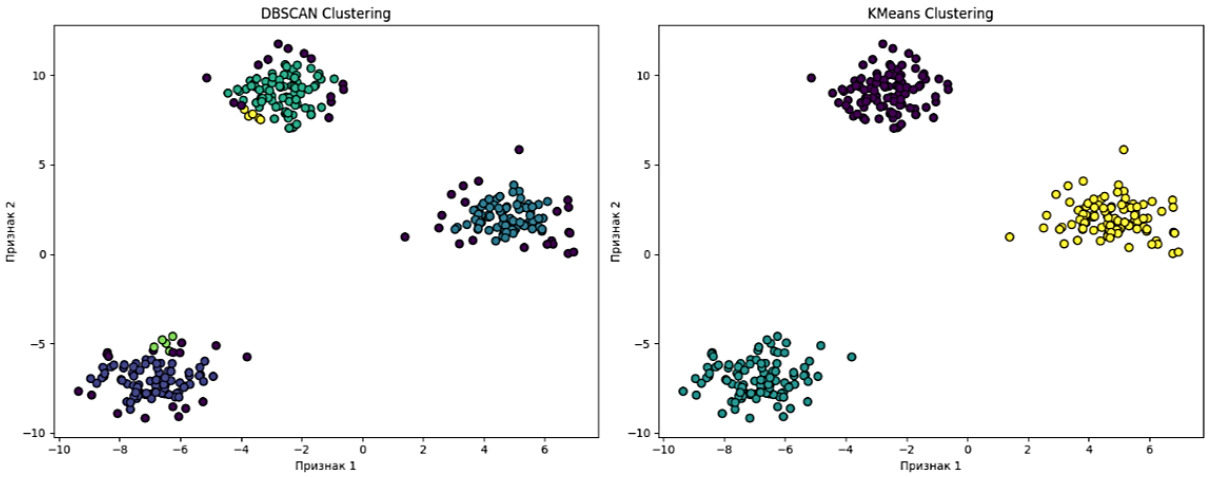

Рисунок 8 – Преимущество k-средних (справа) перед методом DBSCAN в случае скоплений объектов

2 ПРАКТИЧЕСКИЕ ПРИМЕНЕНИЯ КЛАСТЕРИЗАЦИИ

2.1 Цели и задачи кластерного анализа

Как было отмечено в первой главе, для проведения кластерного анализа необходимо иметь многомерный массив данных (или датафрейм), состоящий из n наблюдений и m переменных. Кластеризация данных часто представляет собой предварительный этап анализа, который упрощает последующее применение других методов анализа. В контексте задачи кластеризации выделяются четыре основные цели: [5]

1. **Понимание:** разделение выборки на конкретные группы схожих объектов способствует пониманию особенностей исследуемых данных. Это упрощает дальнейшую обработку данных, так как к каждому кластеру можно применить соответствующий метод анализа. Такой подход широко используется в статистике и области Big Data.
2. **Выявление аномалий:** после проведения кластеризации могут появиться отдельные данные, не входящие ни в один из кластеров. Их изучение необходимо для определения, являются ли эти данные ошибочными или представляют собой интересный феномен.
3. **Расширение:** некоторые данные обладают большим количеством признаков, в то время как у других признаков меньше. Кластеризация позволяет предположить отсутствующие признаки у других элементов в кластере. Например, если известно, что клиенты в кластере «М» проводят в среднем 15 минут на сайте, то при появлении нового клиента с неизвестным временем пребывания можно предположить, что это время также равно 15 минутам.
4. **Сжатие:** в случае избытка данных их можно разделить на кластеры, усреднить и оставить по одному объекту на каждый кластер. Это позволяет использовать меньше вычислительных ресурсов при последующем анализе.

Так, на каждом этапе кластеризации ставится своя задача, это можно выразить следующей схемой (рисунок 9): [12]

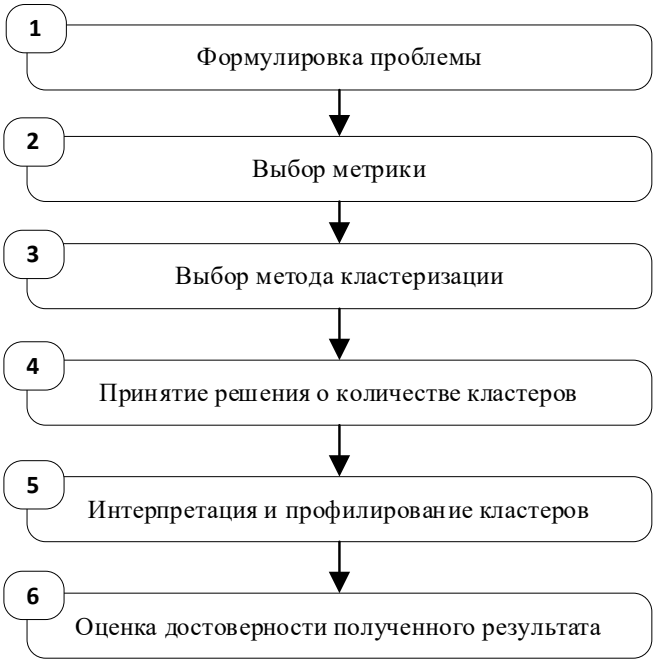

Рисунок 9 – Этапы кластеризации

Классический кластерный анализ решает задачу распределения имеющихся в массиве данных наблюдения на группы [6]. В машинном обучении существуют такие методы кластеризации, которые позволяют ещё и предсказывать, к какому кластеру отнести новое наблюдение – подобные методы применяются при разработке беспилотных автомобилей в рамках задач классификации объектов искусственным интеллектом [14].

Один из важных этапов предварительной обработки данных в контексте кластерного анализа – стандартизация данных [15]. Этот процесс направлен на приведение переменных к одному масштабу, что позволяет более эффективно проводить кластеризацию и улучшает качество получаемых результатов. Выражается это в следующем:

1. Различные переменные в наборе данных могут иметь разные единицы измерения и различные диапазоны значений. Это может привести к тому, что переменные с более высокими значениями и большим разбросом будут оказывать большее влияние на процесс кластеризации. Стандартизация

позволяет уравнять масштабы переменных, что важно для корректного определения схожести объектов.

2. Кластерный анализ может быть чувствителен к выбросам, которые могут значительно исказить результаты. Стандартизация помогает уменьшить влияние выбросов, поскольку значения переменных приводятся к одному стандартному масштабу.
3. Стандартизация данных облегчает интерпретацию результатов кластерного анализа. Единый масштаб переменных делает кластеры более интерпретируемыми, поскольку объекты внутри кластеров оцениваются по одним и тем же критериям.

Стандартизация данных может быть осуществлена путем преобразования каждой переменной z к виду z' по формуле (7):

$$z' = \frac{(z-\mu)}{\sigma}, \tag{7}$$

где z'– стандартизированное значение, z – исходное значение переменной, μ – среднее значение переменной, σ – стандартное отклонение переменной.

Стандартизация помогает алгоритмам сходиться быстрее, так как расстояния становятся более устойчивыми к различиям в масштабах переменных. Так же предварительная обработка данных помогает уменьшить влияние шума в данных, что особенно важно при работе с реальными, зашумленными данными. Это способствует выделению более четких и структурированных кластеров. В случае многомерных данных стандартизация должна быть проведена по каждой переменной отдельно, чтобы сохранить важные относительные отношения между переменными [15].

2.2 Сегментация и идентификация инцидентов кибербезопасности

В современном информационном обществе безопасность данных становится все более приоритетным вопросом, и кибератаки становятся все более сложными и изощренными. Для борьбы с угрозами кибербезопасности эффективные методы

сегментации и идентификации инцидентов становятся ключевыми [16]. В данном контексте, кластерный анализ играет важную роль в обнаружении угроз, а методы кластеризации применяются для анализа сетевого трафика и логов безопасности.

Кластерный анализ в области кибербезопасности является мощным инструментом для выявления паттернов, аномалий и группировки схожих событий. Разделяя данные на кластеры, кластерный анализ помогает выделить аномалии, которые могут свидетельствовать о потенциальных атаках. Он способен автоматически обнаруживать изменения в сетевом трафике или в логах безопасности, что делает его неотъемлемой частью системы обнаружения инцидентов [16].

Среди методов, описанных в главе 1, своё применение в информационной безопасности нашли следующие алгоритмы:

1. **Метод k-средних –** этот метод позволяет разделить сетевой трафик на ранее определенное число кластеров, что позволяет выделить группы схожих сетевых активностей и может помочь выявить атаки или аномальное поведение.
2. **Иерархическая кластеризация –** построение дерева кластеров, иерархия которых может предоставить более глубокий анализ структуры сетевого трафика, полезно для выявления необычных сочетаний сетевых событий.
3. **DBSCAN** основан на плотности данных и может выделять кластеры переменной формы. Этот метод особенно эффективен при обнаружении аномалий, так как позволяет выявлять области низкой плотности.

Кластерный анализ может помочь в выделении групп событий в логах безопасности, что облегчает их интерпретацию и анализ. Анализ логов с использованием методов кластеризации помогает выявлять аномальные события или последовательности событий, которые могут свидетельствовать о кибератаках. Также кластерный анализ может использоваться для группировки логов по типам инцидентов, что облегчает классификацию и понимание их характеристик.

Сегментация и идентификация инцидентов кибербезопасности с использованием методов кластерного анализа представляют собой важный инструмент в борьбе с киберугрозами, позволяют выявлять паттерны, группировать события и обнаруживать аномалии, что делает системы безопасности более эффективными в предотвращении и реагировании на угрозы. Однако необходимо учитывать, что эффективность этих методов зависит от правильной подготовки данных и настройки параметров алгоритмов под конкретные условия и требования организации.

Так, например, если предприятие подверглось N угрозам в течение некоторого времени, а информация о каждой i-ой угрозе описывается вектором $\vartheta_i = (U_i, \tau_i)$, где U_i – ущерб, τ_i – длительность устранения последствий, тогда, если объектом кластеризации принять ϑ_i, можно получить следующий результат (рисунок 10): [16]

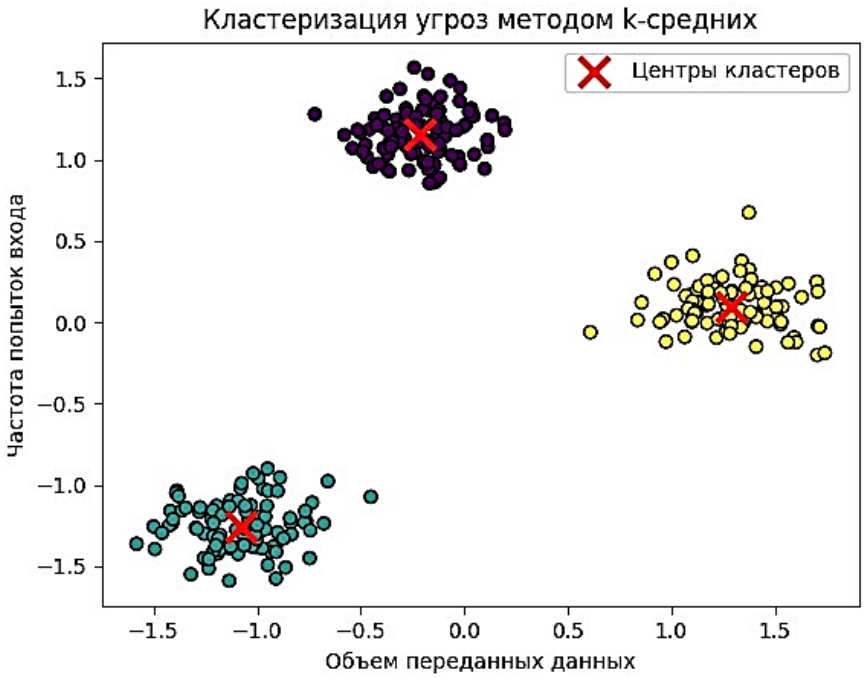

Рисунок 10 – Пример кластеризации угроз методом k-средних

2.3 Методы оценки качества кластеризации

В контексте кибербезопасности, где важна точность обнаружения аномалий и выделение подозрительных паттернов, оценка результатов кластеризации – существенный этап анализа. К основным методам оценки можно отнести [7]:

1. Выделение кластеров – это первичный шаг оценки. Эффективность кластеризации определяется способностью алгоритма правильно группировать схожие объекты в один кластер. Существуют различные метрики, оценивающие качество разделения данных на кластеры:

- Индекс силуэта – данный индекс измеряет, насколько объект хорошо согласуется с кластером, к которому он относится, по сравнению с ближайшим соседним кластером. Значения находится в интервале (-1;1), где 1 указывает на хорошее разделение кластеров.
- Коэффициент Дэвиса-Болдуина оценивает «разделимость» кластеров, основываясь на среднем расстоянии между кластерами и их внутренней компактности.

2. Оценка значимости кластеров. После выделения кластеров, важно оценить их значимость с точки зрения конкретной задачи, например, в области кибербезопасности:

- Размер кластеров – большие кластеры могут указывать на обычное поведение, тогда как маленькие кластеры могут быть интересными с точки зрения аномалий.
- Степень отклонения кластеров – оценка степени отклонения (аномальности) кластеров от общего распределения может быть выполнена с использованием статистических методов.
- Кластеры перекрывающихся границ – перекрывающиеся кластеры могут указывать на сложные, многомерные атаки, которые не поддаются четкому выделению в один кластер.

3. Интерпретация выделенных кластеров в контексте ИБ:

- Семантическая интерпретация – присвоение семантического значения каждому кластеру важно для понимания, что представляет собой каждый кластер. Например, кластер с повышенной активностью может быть ассоциирован с обычной бизнес-деятельностью, в то время как необычно активные кластеры могут указывать на потенциальные атаки.
- Визуализация данных – использование визуализации помогает представить многомерные данные в двух или трех измерениях, что облегчает визуальное исследование выделенных кластеров.

4. Оценка эффективности применения кластерного анализа в ИБ:

- Уровень обнаружения аномалий – оценка того, насколько эффективно кластеризация выявляет аномалии и подозрительные паттерны в данных. Это может быть измерено сравнением с известными инцидентами.
- Ложные срабатывания – оценка того, насколько часто алгоритм создает ложные срабатывания, то есть ошибочно выделяет нормальные паттерны как аномалии.
- Время обнаружения – оценка того, как быстро система кластеризации обнаруживает аномалии и реагирует на них. Сокращение времени обнаружения важно для предотвращения ущерба от кибератак.
- Способность к адаптации – оценка того, насколько хорошо алгоритм кластеризации адаптируется к новым и появляющимся угрозам, не требуя постоянного переобучения.

Методы оценки качества кластеризации в области кибербезопасности не только помогают понять эффективность применения алгоритмов, но и предоставляют важные инсайты в выделенные кластеры. Эти оценки позволяют аналитикам информационной безопасности принимать обоснованные решения, опираясь на результаты кластерного анализа, улучшая тем самым проактивность и эффективность системы безопасности.

ЗАКЛЮЧЕНИЕ

В ходе выполнения данной курсовой работы была рассмотрена такая задача математической статистики, как кластерный анализ. Развитие информационных технологий и увеличение объемов данных сделали кластеризацию важным инструментом для анализа и систематизации информации.

Введение в тему позволило рассмотреть исторический контекст развития кластерного анализа, его роль в современных технологиях, исследовать истоки метода и его эволюцию. Различные методы, такие как иерархическая кластеризация, метод k-средних и DBSCAN, были рассмотрены в контексте их теоретических аспектов, что позволило сформировать базовое понимание принципов кластерного анализа.

Следующий этап работы был посвящен практическим применениям кластеризации. Рассмотрены цели и задачи, которые можно достичь с помощью данного метода, а также методы оценки качества кластеризации, необходимые для правильной интерпретации результатов. Особое внимание уделено сегментации и идентификации инцидентов кибербезопасности, где кластерный анализ проявляет себя как мощный инструмент для обнаружения угроз, выявления аномалий и классификации данных о инцидентах.

В заключении стоит отметить, что кластерный анализ продолжает эволюционировать, становясь более точным и универсальным инструментом. С появлением новых методов и технологий, таких как кластерный анализ графов, методы этой области находят применение в самых разнообразных сферах, включая область информационной безопасности. Эффективное применение кластерного анализа в данной области подтверждено практическими примерами исследований, где методы кластеризации успешно применяются для выявления угроз и обеспечения безопасности автоматизированных систем.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Двоенко С. Д. Методы анализа больших массивов данных / Конспект лекций. - Тула: ТулГУ, 2001. - С. 1-93.
2. Мандель И. Д. Кластерный анализ. - Университет Вирджинии: Финансы и статистика, 1988. - 176 с.
3. Многомерные пространства / QuData. URL: https://qudata.com/ml/ru/ML_Feature_Space.html (дата обращения: 27.11.2023).
4. Тюрин А.Г, Зуев И.О. КЛАСТЕРНЫЙ АНАЛИЗ, МЕТОДЫ И АЛГОРИТМЫ КЛАСТЕРИЗАЦИИ / HERALD of MSTU MIREA. - 2014. - №2. - С. 86-97.
5. Анализ данных / Яндекс.Практикум URL: https://practicum.yandex.ru/blog/chto-takoe-klasterizaciya-i-klasternyi-analiz/ (дата обращения: 27.11.2023).
6. Кластерный анализ: введение / RPubs URL: https://api.rpubs.com/AllaT/clust1 (дата обращения: 23.11.2023).
7. Бантикова, О.И. Методы кластерного анализа. Классификация без обучения (непараметрический случай): методические указания к лабораторному практикуму, курсовой работе, дипломному проектированию и самостоятельной работе студентов / О.И. Бантикова, Е.Н. Седова, О.С. Чудинова; под ред. А.Г. Реннера; Оренбургский гос. ун-т. – Оренбург: ГОУ ОГУ, 2011.– 93 с.
8. Классификация и кластер. Под ред. Дж. Вэн Райзина. М.: Мир, 1980. 390 с.
9. Кластеризация // Учебник по машинному обучению URL: https://education.yandex.ru/handbook/ml/article/klasterizaciya (дата обращения: 23.11.2023).
10. ЛИНЕЙНАЯ АЛГЕБРА И ГЕОМЕТРИЯ. Конспект лекций // teach-in URL: https://teach-in.ru/file/synopsis/pdf/linear-algebra-timashev-M2.pdf (дата обращения 23.11.2023).
11. Миркин, Б. Г. Методы кластер-анализа для поддержки принятия решений: обзор : препринт WP7/2011/03 [Текст] / Б. Г. Миркин ; Национальный

исследовательский университет «Высшая школа экономики». – М. : Изд. дом Национального исследовательского университета «Высшая школа экономики», 2011. – 88 с. – 150 экз.

12. Бююль А., Цёфель П.: искусство обработки информации. Анализ статистических данных и восстановление скрытых закономерностей : Пер. с нем. / Ахим Бююль, Петер Цёфель - СПб. : ООО «ДиаСофтЮП», 2005 - 608 с.
13. Машинное обучение. Метрические методы. Алгоритм кластеризации DBSCAN // proproprogs URL: https://proproprogs.ru/ml/ml-algoritm-klasterizacii-dbscan (дата обращения: 23.11.2023).
14. Метод K-Nearest Neighbors. Разбор без использования бибилотек и с использованием бибилиотек // Хабр URL: https://habr.com/ru/articles/680004/ (дата обращения: 23.11.2023).
15. How to Normalize or Standardize a Dataset in Python / Christian V. [Электронный ресурс] // Github : [сайт]. — URL: https://github.com/christianversloot/machine-learning-articles/blob/main/how-to-normalize-or-standardize-a-dataset-in-python.md (дата обращения: 02.12.2023).
16. Куринных, Д. Ю., Айдинян, А. Р., Цветкова, О. Л. Подход к кластеризации угроз информационной безопасности предприятий [Текст] / Д. Ю. Куринных, А. Р. Айдинян, О. Л. Цветкова // Инженерный вестник Дона. — 2018. — № 1. — С. 1-7.

ПРИЛОЖЕНИЕ

Листинг 1 – Код программы на языке программирования Python для реализации иерархической кластеризации

```
import pandas as pd
from scipy.cluster.hierarchy import linkage, dendrogram
import matplotlib.pyplot as plt
data = {
    'City': ['Тула', 'Ясногорск', 'Новомосковск', 'Алексин', 'Щёкино', 'Суворов', 'Кимовск'],
    'Latitude': [54.1960, 54.6846, 54.0359, 54.5096, 54.0029, 54.1252, 54.5612],
    'Longitude': [37.6188, 37.5820, 38.2849, 37.0596, 37.5174, 36.4787, 40.5918]
}
df = pd.DataFrame(data)
# Агломеративная иерархическая кластеризация методом средней связи
linkage_matrix = linkage(df[['Latitude', 'Longitude']], method='average', metric='euclidean')
# Дендрограмма
plt.figure(figsize=(12, 8))
dendrogram(linkage_matrix, labels=df['City'].values, color_threshold=5.0, leaf_rotation=45, leaf_font_size=10, above_threshold_color='pink')
plt.xlabel('Города')
plt.ylabel('Расстояние в градусах')
plt.show()
```

Листинг 2 – Код программы на языке программирования Python для реализации кластеризации методом k-средних

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
data = {
    'City': ['Тула', 'Ясногорск', 'Новомосковск', 'Алексин', 'Щёкино', 'Суворов', 'Кимовск'],
    'Latitude': [54.1960, 54.6846, 54.0359, 54.5096, 54.0029, 54.1252, 54.5612],
    'Longitude': [37.6188, 37.5820, 38.2849, 37.0596, 37.5174, 36.4787, 40.5918]
}
```

Листинг 2 – Код программы на языке программирования Python для реализации кластеризации методом k-средних (продолжение)

```
df = pd.DataFrame(data)
kmeans = KMeans(n_clusters=5, random_state=42, n_init=10)
df['Cluster'] = kmeans.fit_predict(df[['Latitude', 'Longitude']])
plt.figure(figsize=(10, 8))
plt.scatter(df['Longitude'], df['Latitude'], c=df['Cluster'], cmap='viridis', edgecolors='black', s=100)
for cluster_center in kmeans.cluster_centers_:
    plt.gca().add_patch(plt.Circle((cluster_center[1], cluster_center[0]), 0.5, fill=False, edgecolor='red', linestyle='--'))
for i, city in enumerate(df['City']):
    plt.annotate(city, (df['Longitude'][i], df['Latitude'][i]), textcoords="offset points", xytext=(0,10), ha='center')
plt.xlabel('Долгота')
plt.ylabel('Широта')
plt.show()
```

Листинг 3 – Код программы на языке программирования Python для реализации кластеризации методом DBSCAN

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.cluster import DBSCAN
from sklearn.datasets import make_circles
X, y = make_circles(n_samples=300, factor=0.5, noise=0.05, random_state=42)
dbscan = DBSCAN(eps=0.3, min_samples=5)
clusters = dbscan.fit_predict(X)
plt.figure(figsize=(10, 8))
plt.scatter(X[:, 0], X[:, 1], c=clusters, cmap='viridis', edgecolors='black', s=50)
plt.title('DBSCAN Clustering - Concentric Circles')
plt.xlabel('Признак 1')
plt.ylabel('Признак 2')
plt.show()
```