

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Национальный исследовательский университет «МЭИ»

Институт: ИРЭ РТФ

Кафедра: РТС

Направление подготовки:

11.04.01 – Радиотехника

ОТЧЕТ по практике

**Наименование
практики:**

Учебная практика: Научно-исследовательская
работа (получение первичных навыков научно-
исследовательской работы)

СТУДЕНТ

(подпись)

/

Масалкова Н.В.

/

(Фамилия и инициалы)

Группа

ЭР-12М-20

(номер учебной группы)

**ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ
ПО ПРАКТИКЕ**

ОТЛИЧНО

(отлично, хорошо, удовлетворительно, неудовлетворительно,
зачтено, не зачтено)

(подпись)

/

Куликов Р.С.

/

(Фамилия и инициалы члена комиссии)

(подпись)

/

Шатилов А.Ю./

(Фамилия и инициалы члена комиссии)

**Москва
2020**

Оглавление

Введение.....	3
1. Анализ этапов обработки радиолокационной информации	7
2. Постановка задачи.....	10
3. Основные принципы нейросетевых технологий	11
3.1. Определение нейронных сетей.....	11
3.2. Нейросетевые модели	12
3.3. Кластерный анализ	13
3.3.1. Меры расстояний.....	14
3.3.2. Обзор алгоритмов кластеризации	17
3.4. Описание алгоритма DBSCAN.....	21
4. Моделирование.....	25
4.1. Исходные данные моделирования	25
4.2. Формирование выборки	26
4.3. Результаты эксперимента	27
4.4. Результаты работы алгоритма кластеризации DBSCAN	36
Заключение	37
Список литературы	38

Введение

Современные радиолокаторы должны работать в условиях, для которых характерны высокая скорость изменения внешней обстановки и большое количество объектов, подлежащих обнаружению [1]. Обработка столь большого количества информации в требуемые сжатые сроки не может быть эффективно осуществлена человеком-оператором [8]. Поэтому перед радиолокационными станциями (РЛС), являющимися основным источником информации, ставится задача проведения законченного цикла её автоматической обработки. Актуальность решения данной задачи в следующем, что в отличие от применявшихся ранее систем обработки, в которых конечное решение принимал человек-оператор, в этих образцах конечное решение принимается автоматом [16].

Для разработки подобных систем потребовалось создание теории автоматической обработки радиолокационной информации (РЛИ). Существуют методы, способные осуществлять автоматическую обработку РЛИ [4-9]. Однако большинство данных методов базируются на положениях классической теории радиолокации, которая опирается на предположения, многие из которых не выполняются на практике. Необходимость использования этих предположений обусловлена трудностью формализации и математического описания всевозможных воздействий помех и других факторов, отсутствием единой методологии оценки систем обработки в различных воздушных и помеховых условиях. И оказалось, что при создании образцов РЛС с автоматической обработкой информации, классическая теория часто оказывается неприменима [9]. До сих пор нет универсальной теории для создания автоматической обработки РЛИ.

Действительно, современные радиолокационные средства в достаточно простых условиях успешно справляются со своими задачами, а в сложных условиях (нестационарных и негауссовых помех) их эффективность может резко снижаться. Снижается достоверность выдаваемой РЛС информации за

счет появления большого числа ложных отметок и появления ложных траекторий. Их число может намного превышать число целей в зоне обзора, а использование недостоверной информации ведет к снижению эффективности, например, стрельбовых средств в несколько раз [7].

Непредсказуемость внешней обстановки и высокая динамика её изменения создают значительные трудности для формулировки алгоритмов и для обеспечения высокого качества их работы. Появляются и новые задачи, например, распознавание, кластеризация, классификация и анализ ситуаций. И классических методов решения таких задач нет [11].

На данный момент одно из актуальных и многообещающих направлений построения эффективных систем обработки информации является использование такой информационной технологии, как искусственные нейронные сети (ИНС). Аппарат нейронных сетей предлагает, там, где не существует алгоритмов или эти алгоритмы недостаточно эффективны, многообещающий подход к созданию систем обработки, обладающих столь привлекательными чертами как гибкость, способность адаптироваться к изменениям внешних условий, сохраняя устойчиво высокое качество работы.

Нейронные сети предусматривают применение принципиально нового подходу к синтезу методов обработки в алгоритмическом смысле. Данная технология предоставляет компьютерной модели нейронной сети возможность обучаться на примерах и получать решение для проблем, ранее считавшихся неразрешимыми без участия человека. При этом, как уже упоминалось ранее, достигается гибкость и активность работы, сохранение устойчиво высоких показателей работы при отличиях внешних условий, от рассматриваемых или создаваемых на этапе разработки, возможность построения эффективных систем без трудоемких, а зачастую и невыполнимых, построений аналитических описаний, способность оперирования нечеткими понятиями и так далее.

ИНС в последнее время успешно применяются в широком спектре приложений от бытовых приборов, в которых с их помощью осуществляется выбор наиболее благоприятных режимов работы, до крупномасштабных систем финансового анализа и сложнейших вычислительных комплексов управления военными действиями.

Всё это свидетельствует о необходимости внедрения нейросетевой технологии в перспективные системы обработки информации [11]. И соответственно в системы, решающие радионавигационные задачи, так как они требуют высокой скорости обработки больших массивов данных для получения достоверных результатов анализа обстановки.

Следует сказать, что ИНС – это алгоритмический аппарат построения систем принятия решений на основе сетей формальных нейронов, которые реализуют элементарные арифметические функции. Данное понятие позволяет исследовать нейронные сети с помощью традиционного математического аппарата или развивая этот аппарат, но оставаясь в рамках традиционных математических парадигм. В итоге решения, получаемые с помощью нейросетевых технологий, позволяют приблизиться к сколь угодно близко к истинно оптимальным решениям для сложных и нетривиальных условий принятия решений.

Тогда, чтобы успешно применять аппарат нейронных сетей в процессах обработки РЛИ должны быть тщательно проанализированы условия использования, этапы преобразования входных данных, обоснованы принципы оценки качества обработки, определены наиболее перспективные места применения.

В настоящее время актуальными являются исследования по повышению качества обработки информации за счет использования нейросетевой технологии в каждом из подтипов подсистем обработки в зависимости от физической структуры входных и выходных сигналов, места их применения,

характера изменения внешних условий. Объем исследований при такой постановке задачи оказывается очень широк, и охватить весь спектр возможных применений нейронных сетей в полном объеме не удастся. В этих условиях встает задача разделения подобных между собой этапов обработки информации в разнородных системах, эффективность которых могла бы быть резко повышена за счет применения нейронных сетей.

В данной работе нас будет интересовать автоматизация одного из этапов обработки радиолокационной информации.

1. Анализ этапов обработки радиолокационной информации

Обработка РЛИ представляет собой наиболее важный комплекс задач радиолокационных станций. Назначение обработки подготовить к выдаче в требуемом виде полную, достоверную и современную информацию для потребителя о состоянии воздушной обстановки, появлении и местоположении воздушных средств объектов, параметрах их движения, возможных вариантах развития динамики изменения воздушно-помеховой обстановки.

В ходе обработки РЛИ решается целый ряд задач, которые можно сгруппировать по близости объектов и применяемых методов в основные этапы.

К основным этапам обработки радиолокационной информации можно отнести [9,10].

- Формирование диаграммы направленности антенны для фазирования антенных решеток;
- Первичная (спектральная) обработка;
- Обнаружение радиолокационных объектов
- Вторичная (траекторная) обработка
- Распознавание объектов и ситуаций
- Анализ обстановки и адаптация

Решаемые на этих этапах задачи служат выполнению главной цели, для которой предназначен радиолокатор – выдаче полной, достоверной и своевременной информации потребителю о воздушных целях.

Большого внимания заслуживает такой этап обработки, как обнаружение отметок от целей и их кластеризация для дальнейшей вторичной (траекторной) обработки целей.

Обнаружение радиолокационных целей обычно включает в себя два подэтапа: обнаружение импульсов, отраженных от целей, и их последующего пакетирования в ходе которого объединяются решения, принятые по каждому отдельному импульсу. Обнаружение целей должно проводиться автоматически, сохраняя высокие характеристики при работе в различных тактических условиях, при воздействии разнотипных помех, в том числе нестационарных, т.е. в существенно неопределенных условиях.

Рассмотрим некоторые методы, которые применяются для автоматического обнаружения воздушных целей в РЛС, а именно использование адаптивного порога обнаружения [18], использование фазовых соотношений в многобазовых сигналах (знаковые методы) и ранговые методы [5].

Данным методам присущи крупные недостатки, которые определяют практическую полную непригодность для РЛС, ведущих работу в реальных условиях. Рассмотрим подробнее: в реальных условиях при воздействии на РЛС помех нестационарного типа, а также хотя и стационарных, но негауссовых помех таких, как отражения от подвижных и местных предметов, пассивные помехи, уровень ложных тревог может возрасти на 2...3 порядка, что недопустимо. Кроме того, при наличии в пределах опорной выборки сигналов, отраженных от соседних с обнаруживаемой целью, что характерно для обнаружения целей в группах, энергетические потери могут составлять 10...15 дБ и более [18]. Исключение составляют ранговые методы, обеспечивающие практически полную стабилизацию уровня ложных тревог, но эти методы при малой пачке имеют очень высокие потери даже при обнаружении одиночных целей, а при обнаружении групп целей эти потери катастрофически возрастают.

Для синтеза эффективных методов автоматического обнаружения необходимо привлечь статическую теорию различения гипотез. Синтезируемые методы должны эффективно работать в условиях воздействия

основных помех. При этом необходимо определить совокупность параметров помех и обеспечить устойчивость и совокупность параметров.

Пакетирование импульсов проводится для формирования отметок от целей, на основе полученных нескольких импульсов от цели. Эта задача аналогична задаче кластеризации: необходимо из множества одиночных отметок выбрать несколько центров группирования, которые соответствуют обнаруживаемым целям. В итоге проведение группирования отметок в кластеры позволит улучшить алгоритмы вторичной обработки сигналов, а именно сопровождение траекторий групп целей. Группирование позволит сократить количество ложных траекторий примерно на порядок.

Задача кластеризации эффективно решается с помощью нейронных сетей. Подробнее про аппарат ИНС будет показано далее.

2. Постановка задачи

Глобальная цель настоящей научно исследовательской работы является повышения эффективности обработки информации в системах пассивной радиолокации за счет более точного отождествления сигналов с целями путем обработки сигналов алгоритмами кластеризации.

В данной же работе будут рассмотрены начальные этапы решения глобальной цели. А именно требуется изучить какие существуют алгоритмы для решения задачи кластеризации, сравнить существующие алгоритмы, выбрать метод кластеризации для данной задачи и описать его принципы работы. Смоделировать модель, которая будет имитировать реальные измерения, чтобы оценить работу метода кластеризации.

Задача настоящей работы – поиск повторяющихся сигналов в смоделированной выборке, которая включает в себя шумовые данные. Требуется также оценить качество работы алгоритма кластеризации на данной выборке.

3. Основные принципы нейросетевых технологий

Для анализа методов нейросетевых технологий определим основные содержательные элементы технологии: информационной и нейросетевой.

3.1. Определение нейронных сетей

Воспользуемся определением нейронной сети автора нескольких классических трудов по теории нейронных сетей С. Хайкина [13,19]. Нейронная сеть – это процессор с массово-параллельной архитектурой, который имеет естественную склонность к записи эмпирических знаний и представления для их последующего использования. Этот процесс похож на мозг в двух отношениях:

1. Знания получаются сетью посредством процесса обучения.
2. Для запоминания знаний используются межнейронные веса, известные как синаптические связи.

В данной работе под нейросетевой технологией понимаются приемы, способы и методы обработки информации, основанные на использовании сетей, связанных между собой достаточно простых элементов (нейронов), реализующих методы совместной работы, близкие по содержанию к работе биологических нейронных систем. Данное определение рассматривает близость к биологическим нейронам лишь в самом широком смысле. Здесь не пытаемся осуществить полное моделирование биологической системы. Поэтому многие нейронные сети, не имеющие биологических аналогов, оказываются успешными и полезными в вычислительном отношении и в плане применения в различных приложениях [17,20]. Основные достижения нейросетевой технологии связаны с реальными успешными применениями математических алгоритмов, появившихся как развитие моделей биологических нейронных сетей.

В итоге теория нейронных сетей опирается на предположение о том, что с помощью, рассматриваемой в ее рамках методологически простой модели

удастся получить достаточно сложное поведение, необходимое, для получения правильных решений в изменяющихся условиях.

3.2. *Нейросетевые модели*

В основе нейросетевой технологии лежит концепция распределенных адаптивных нелинейных вычислений. Нейронные сети строятся из большого числа простых элементов, каждый из которых обрабатывает свою часть единой задачи. Принципы вычислений для стандартного элемента достаточно просты. Вся мощь нейронных сетей основывается на множестве взаимосвязей между этими простыми элементами, которые совместно участвуют в решении задачи, и в адаптивной настройке параметров (весов), связывающих элементы между собой.

Каждой нейросетевой архитектуре присущ собственный метод обучения, ориентированный на конкретную организацию взаимосвязей в ней. Поэтому обычно рассматриваются не просто архитектуры нейронных сетей, а комплекс из организации элементов сети, нейронов, в виде слоев и метода обучения или алгоритма достижения нейронной сетью состояния, обеспечивающего выполнению ею требуемых действий.

Рассмотрим подробнее две большие группы на которые разделены нейронные сети в соответствии с применяемыми методами обучения: обучение с учителем и обучение без учителя.

При обучении с учителем правильные результаты функционирования сети (желаемые выходы) считаются известными, и одновременно с каждым набором входных значений сети предъявляется информация о том, насколько правильно она функционирует или каким должен быть правильный отклик – референсом. Эта информация используется нейронной сетью для подстройки весов, которые изменяются таким образом, чтобы максимально приблизиться к задаваемым целевым значениям. После обучения нейронная сеть тестируется при различных входных значениях и оценивается, насколько ее

выход оказывается близко к желаемому отклику. В итоге обучение с учителем требует на вход нейронной сети представление об окружающем пространстве в виде пары вход-выход, и для измерения качества обучения сети используются прямолинейные и понятные метрики.

При обучении без учителя нейронной сети не сообщаются правильные выходные значения. Сеть работает автономно: она исследует данные, которые ей представляются, находит в них некоторые характерные закономерности и обучается отображать эти закономерности в своем отклике. Она сама подстраивает веса синаптических связей, опираясь на внутреннюю структуру данных. Какие именно признаки способно отыскивать и отображать нейронная сеть, зависит от ее конкретной архитектуры и используемого метода обучения. Такие нейронные сети обычно выполняют задачи сжатия информации, снижения размерности или кластеризации.

Итак, в данной работе нас интересует задача кластеризации, которая решается способом обучения нейронной сети без учителя. Для метода обучения без учителя есть несколько особенностей – во-первых это возможность использования несопоставимо больших объемов данные, поскольку их не нужно размечать данные для обработки, что является плюсом, т.к. РЛИ поступает на радиолокатор в больших объемах и человек-оператор не всегда в силах разобрать информацию. Во-вторых, есть и минусы – это неясность в выборе измерения качества метода и отсутствие прямолинейных и интуитивно понятных метрик, как в задачах обучения с учителем.

3.3. Кластерный анализ

Для начала дадим определение понятию кластеризация. Кластеризация или кластерный анализ — это задача разбиения множества объектов на группы, называемые кластерами. Внутри каждой группы оказываются объекты с подобными параметрами, а объекты разных группы имеют максимальные отличия друг от друга. Существенное отличие задачи кластеризации от других задач, решаемых нейронными сетями, например,

классификации, состоит в том, что перечень групп четко не заданы и требуется глубокий анализ входной базы данных и подбор параметров в процессе работы алгоритма кластеризации.

Применение кластерного анализа в общем виде сводится к следующим этапам:

1. Отбор выборки объектов для кластеризации.
2. Определение множества переменных, по которым будут оцениваться объекты в выборке. Перед запуском алгоритма чаще всего требуется либо *нормализовать* данные, либо *стандартизировать*. Нормализация предполагает замену номинальных признаков так, чтобы каждый из них находился в диапазоне от нуля до единицы. Стандартизация же подразумевает предобработку данных, после которой каждый признак имеет среднее значение равное нулю и среднее отклонение равное единице. В итоге нормализация или стандартизация дает равную важность переменным во время кластеризации.
3. Вычисление значений *меры сходства* между объектами.
4. Применение метода кластерного анализа для создания групп сходных объектов (кластеров).
5. Представление результатов анализа.

После получения и анализа результатов возможна корректировка выбранной метрики и метода кластеризации до получения оптимального результата.

3.3.1. Меры расстояний

Требуется ответить на вопрос как оценивать качественно работу алгоритма кластеризации. Как было сказано ранее задача оценивания качества кластеризации является более сложной по сравнению с оценкой качества классификации. Во-первых, такие оценки не должны зависеть от самих значений меток, а только от самого разбиения выборки. Во-вторых, не всегда

известны истинные метки объектов, поэтому также нужны оценки, позволяющие оценить качество кластеризации, используя только неразмеченную выборку.

Выделяют *внешние* и *внутренние* метрики качества. Внешние используют информацию об истинном разбиении на кластеры, в то время как внутренние метрики не используют никакой внешней информации и оценивают качество кластеризации, основываясь только на наборе данных. Оптимальное число кластеров обычно определяют с использованием внутренних метрик.

Приведем меры для измерения расстояний, наиболее часто встречающихся в кластерном анализе.

- **Евклидово расстояние**

Наиболее распространенная функция для определения расстояния. Представляет собой геометрическое расстояние в многомерном пространстве между двумя точками:

$$\rho(x, x') = \sqrt{\sum_i^N (x_i - x'_i)^2} \quad (3.1)$$

Где x и x' точки в N -мерном пространстве.

- **Квадрат евклидова расстояния**

Применяется для придания большего веса более отдаленным друг от друга объектам. Это расстояние вычисляется следующим образом:

$$\rho(x, x') = \sum_i^N (x_i - x'_i)^2 \quad (3.2)$$

- **Расстояние городских кварталов (манхэттенское расстояние)**

Это расстояние является средним разностей по координатам. В большинстве случаев эта мера расстояния приводит к таким же результатам, как и для обычного расстояния Евклида. Однако для этой меры влияние отдельных больших разностей (выбросов) уменьшается (т.к. они не возводятся в квадрат). Формула для расчета манхэттенского расстояния:

$$\rho(x, x') = \sum_i^N |x_i - x'_i| \quad (3.3)$$

- **Расстояние Чебышева**

Это расстояние может оказаться полезным, когда нужно определить два объекта как «различные», если они различаются по какой-либо одной координате. Расстояние Чебышева вычисляется по формуле:

$$\rho(x, x') = \max (|x_i - x'_i|) \quad (3.4)$$

- **Степенное расстояние**

Применяется в случае, когда необходимо увеличить или уменьшить вес, относящийся к размерности, для которой соответствующие объекты сильно отличаются. Степенное расстояние вычисляется по следующей формуле:

$$\rho(x, x') = \sqrt[r]{\sum_i^N (x_i - x'_i)^p} \quad (3.5)$$

где r и p – параметры, определяемые пользователем. Параметр p ответственен за постепенное взвешивание разностей по отдельным координатам, параметр r ответственен за прогрессивное взвешивание больших расстояний между объектами. Если оба параметра – r и p — равны двум, то это расстояние совпадает с расстоянием Евклида.

В итоге выбор метрики полностью лежит на исследователе, поскольку результаты кластеризации могут существенно отличаться при использовании различных мер.

3.3.2. Обзор алгоритмов кластеризации

Приведем обзор различных алгоритмов кластеризации, чтобы решить, какой из алгоритмов лучше всего подойдет для решения поставленной задачи.

- **Иерархическая кластеризация**

Иерархическая кластеризация группирует данные в различных масштабах, создавая дерево кластеров или дендрограмму. Дерево – это не единственный набор кластеров, а многоуровневая иерархия, в которой кластеры на одном уровне объединяются в кластеры на следующем. Это позволяет решить какой уровень кластеризации подходит для решения поставленной задачи.

Среди алгоритмов иерархической кластеризации выделяют два основных типа: восходящие и нисходящие алгоритмы. Нисходящие алгоритмы работают по принципу «сверху-вниз»: в начале все объекты помещаются в один кластер, который затем разбивается на все более мелкие кластеры. Восходящие же алгоритмы работают по обратному алгоритму «снизу-вверх»: в начале работы каждый объект помещается в отдельный кластер, а затем объединяются во все более крупные кластеры, пока все объекты выборки не будут содержаться в одном кластере. В итоге строится система вложенных разбиений.

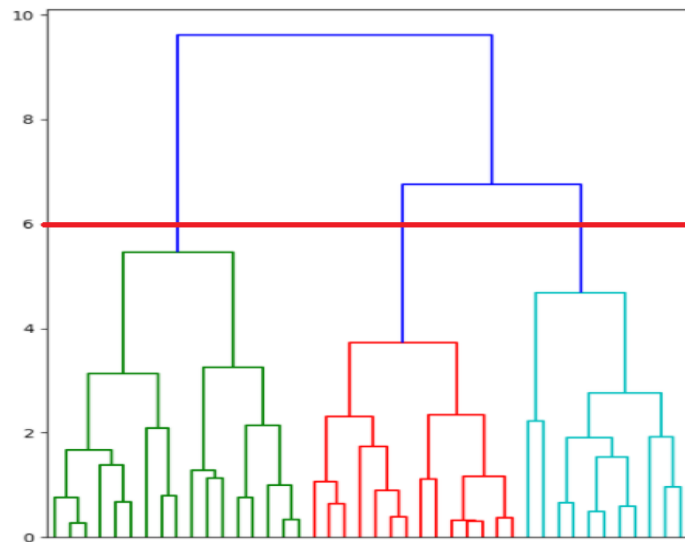


Рис. 3-1. Пример иерархической кластеризации – дендрограмма. Уровнями можно задать количество кластеров разбиения.

- **Метод k-средних**

Данный метод кластеризации считается наиболее простым, но в то же время недостаточно точным. Суть данного алгоритма в следующем: метод разбивает множество элементов векторного пространства на заранее известное число кластеров k . Алгоритм стремится минимизировать среднеквадратическое отклонения на точках каждого кластера. На каждой итерации перевычисляется центр масс для каждого кластера, полученного на предыдущей итерации, затем векторы разбиваются на кластеры вновь в соответствии с тем, какой из новых центров оказался ближе по выбранной метрике. Алгоритм завершается, когда на какой-то итерации не происходит изменения внутрикластерного расстояния. Это происходит за конечное число итераций, так как количество возможных разбиений конечного множества конечно, а на каждом шаге суммарное квадратичное отклонение уменьшается, поэтому заикливание невозможно.

- **Смешанная гауссовская модель**

Смешанная гауссовская модель формирует кластеры по принципу суперпозиции многомерных гауссовских законов распределения плотности вероятности. Для каждого наблюдения данный метод определяет

апостериорные плотности вероятности, которые указывают, что наблюдение имеет некоторую вероятность принадлежности к каждому кластеру.

По смешанной гауссовской модели можно выполнить *hard-кластеризацию* (жесткая кластеризация), которая гарантирует высокую точность группировки, где все запросы привязываются друг к другу и в один кластер попадают только на сто процентов совместимые запросы, путем выбора компонента, который максимизирует апостериорную вероятность. Hard-кластеризация определяет точку из данных только в один кластер. И в итоге кластер образуется там, где наибольшая апостериорная вероятность.

Кроме hard-кластеризации можно использовать смешанную гауссовскую модель для так называемой *soft-кластеризации* (нечетной кластеризации). При данном методе присваивается вероятность попадания точки данных для каждого кластера, и эта вероятность указывает на силу связи точки данных с кластером. В отличие от методов hard-кластеризации, методы soft-кластеризации являются гибкими, поскольку они могут назначать точку данных нескольким кластерам.

- **Спектральная кластеризация**

Спектральная кластеризация использует спектр (собственных значений) матрицы сходства данных для осуществления снижения размерности перед кластеризацией в пространство меньших размерностей. Матрица сходства подается в качестве входа и состоит из количественных оценок относительно схожести каждой пары точек в данных. В низкой размерности данные в кластерах разделяются более широко, позволяя использовать далее алгоритмы кластеризации, например, k-средних. Низкая размерность данных основана на значимых собственных векторах матрицы Кирхгофа. Данная матрица является одним из способов представить график подобия, который отображает локальные отношения между точками данных, как неориентированный граф.

- **DBSCAN**

DBSCAN (Density-based spatial clustering of applications with noise, плотностной алгоритм пространственной кластеризации с присутствием шума) является основанным на плотности алгоритмом, который идентифицирует кластеры произвольной формы и выбросы (шум) в данных. Во время кластеризации DBSCAN определяет точки, которые не принадлежат ни одному из кластеров и относит к шуму. Для этого метода не требуется предварительное знание количества кластеров.

Рассмотрев данные методы кластеризации составим таблицу основных характеристик этих методов (таблица 1).

Таблица 3.3-1. Характеристики основных методов кластеризации

Метод кластеризации	Описание алгоритма	Входные данные	Требование конкретного количества кластеров
Иерархическая кластеризация	Расстояние между объектами	Попарные расстояния между наблюдениями	Нет
К-средних	Расстояние между объектами и центроидами	Фактические наблюдения	Да
Смешанные гауссовские модель	Смесь нормальных распределений	Фактические наблюдения	Да
Спектральная кластеризация	График, представляющий связи между точками данных	Фактические наблюдения или матрица подобия	Да, но алгоритм также обеспечивает способ оценить количество кластеров
DBSCAN	Плотность областей в данных	Физические наблюдения или попарные расстояния между наблюдениями	Нет

Требуется определить какой метод кластеризации использовать для решения поставленной задачи. Напомним, что решается задача проведения группирования отметок в кластеры [6]. В свою очередь количество кластеров

заранее не известно, поэтому подойдут алгоритмы, которые не учитывают количество кластеров, как входной параметр. Также радиолокационные отметки имеют зашумленные данные, поэтому нужно найти такой алгоритм, который бы учитывал случайные выбросы. Тогда учитывая все высказывания подходит алгоритм DBSCAN. Этим алгоритмом в данной работе будет реализована кластеризация отметок.

3.4. Описание алгоритма DBSCAN

Имея дело с пространственными кластерами различной плотности, размера и формы, может быть сложно обнаружить группу точек. Задача может быть еще более сложной, если данные содержат шум и выбросы. Для работы с большими пространственными базами данных обычно применяется алгоритм DBSCAN [15]. Основных причины использования алгоритма:

1. Требуется минимальных знаний предметной области.
2. Он может обнаружить кластеры произвольной формы.
3. Эффективен для большой базы данных, то есть размер выборки превышает несколько тысяч.

Рассматривая данные причины убеждаемся, что с помощью данного алгоритма возможно решить поставленную задачу.

Основная концепция алгоритма DBSCAN состоит в том, чтобы найти области высокой плотности, которые отделены друг от друга областями низкой плотности. И чтобы правильно применить этот алгоритм кластеризации для решения поставленной задачи, требуется подробнее рассмотреть его параметры и характеристики. На вход метод просит матрицу близости и два параметра – радиус *epsilon*-окрестности и *minpts*-минимальное количество соседей.

Для определения *epsilon* и *minpts* введем несколько определений. Пусть задана некоторая симметричная функция расстояния $\rho(x, x')$ и константы ε и m . Тогда:

1. Назовем область $E(x)$, для которой $\forall x: \rho(x, x') \leq \varepsilon$, где ε – окрестность объекта x
2. Центральным объектом или ядерным объектом степени m называется объект, ε -окрестность которого содержит не менее m объектов:
 $|E(x)| \geq m$
3. Объект p непосредственно плотно-достижим из объекта q , если $p \in E(q)$ и q – корневой объект
4. Объект p плотно-достижим из объекта q , если $\exists p_1, p_2 \dots p_n, p_1 = q, p_n = p$, такие что $\forall i \in 1 \dots n - 1: p_i + 1$ непосредственно плотно-достижим из p_i

Следуя определению плотной области, точка может быть классифицирована как *основная точка* если $|E(x)| \geq m$. Центральные точки, как следует из названия, обычно находятся внутри кластера. *Пограничная точка (достижимая по плотности точка)* имеет меньше, чем m в своей $E(x)$ области, но лежит в окрестности другой центральной точки. *Шум (выпадающая точка)* – это любая точка данных, которая не является ни основной, ни пограничной.

Достижимость не является симметричным отношением, поскольку, по определению, никакая точка не может быть достигнута из неосновной точки, независимо от расстояния (так что неосновная точка может быть достижимой, но ничто не может быть достигнуто из неё). Поэтому дальнейшее понятие *связности* необходимо для формального определения области кластеров, найденных алгоритмом DBSCAN. Две точки p и q *связаны по плотности*, если имеется точка o , такая что и p , и q достижимы из o . Связность по плотности является симметричным отношением.

Тогда кластер удовлетворяет двум свойствам:

1. Все точки в кластере попарно связны по плотности.
2. Если точка достижима по плотности из какой-то точки кластера, она также принадлежит кластеру.

Этапы работы алгоритма DBSCAN.

Выше приведены определения и теперь можно описать шаги алгоритма DBSCAN:

1. Алгоритм начинается с произвольной точки, которая ещё не была просмотрена.
2. Выбирается ϵ -окрестность выбранной точки и, если она содержит m точек, то начинается формирование кластера, в противном случае точка помечается как шум. Эта точка может быть позже найдена в ϵ -окрестности другой точки и, таким образом, может стать частью кластера.
3. Если точка найдена как основная точка, то точки в ϵ -окрестности также являются частью этого кластера. Таким образом, все точки найденные в ϵ -окрестность, добавляются вместе с их собственной ϵ -окрестностью, если они также являются основными точками.
4. Вышеописанный процесс продолжается, пока не будет найден связный по плотности кластер.
5. Процесс возобновляется. Выбирается и обрабатывается новая непосещённая точка, что ведёт к обнаружению следующего кластера или шума.

Введя определения и описание работы алгоритма DBSCAN перейдем к недостаткам. Приведем два самых больших недостатков алгоритма:

1. Если в базе данных есть точки, которые образуют кластеры различной плотности, то с помощью данного алгоритма не удастся хорошо

раскластеризовать точки, поскольку кластеризация зависит от параметров ε и m , они не могут быть выбраны отдельно для всех кластеров.

2. Если данные и функции не так хорошо понятны специалисту в области, то может быть настроить параметры ε и m . Тогда требуется сравнение нескольких итераций с различными значениями ε и m .

В итоге сложность применения алгоритма DBSCAN – это выбор ε и m .

Существуют эвристики для подбора ε и m . Чаще всего применяется такой метод и его вариации:

1. Выбирается m . Используются значения больше или равные порядку размерности входных данных. Чем более неоднородный датасет, тем соответственно больше уровень шума и тогда следует взять m большим.
2. Для оценки значения ε требуется вычислить среднее расстояние по m для каждой точки входных данных. Далее сортируются полученные значения по возрастанию и строится график.
3. Построится резко возрастающий график. Значение ε выбирается в полосе, где происходит сильный перегиб кривой, это область, где точки начинают затухать в область выбросов [15].

Итог. DBSCAN отлично работает на плотных, хорошо отделенных друг от друга кластерах. При этом их форма совершенно не важна. Алгоритм отлично обнаруживает кластеры малой размерности. Успешно применяется для большого датасета $N = 10^6 \dots 8$, причем сложность элементов датасета значения не имеет. Количество элементов в кластере может варьироваться, количество выбросов значения не имеет, если они рассеяны по большому объему, а также количество кластеров значения не имеет.

4. Моделирование

С целью качественной оценки эффективности алгоритма кластеризации DBSCAN на данных, которые представляют собой элементарные радиоимпульсы, требуется разработать имитационную модель сигналов.

4.1. Исходные данные моделирования

Для того, чтобы описать, что из себя представляют элементарные импульсы, был проведен анализ реальных записей обнаруживаемых элементарных импульсов с многопозиционного пассивного радиолокационного комплекса. Одиночные импульсы имеют следующие параметры: время прихода импульса $T_{пр}$, длительность импульса $\tau_{и}$, период импульса $T_{и}$, это разность времен между текущим импульсом и предыдущим импульсом, несущая частота F . Анализируя реальные записи было выяснено, что импульсы с некоторыми параметрами встречаются чаще всего. А именно: минимальное значение периода $T_{и.мин} = 2$ мкс, наиболее часто встречающиеся несущие частоты сигналов представляют собой следующий дискретный набор $F = [1,09; 1,5; 5,48; 9,8; 16]$ ГГц, наиболее часто встречающиеся длительности импульса представляют собой дискретный набор $\tau_{и} = [50 \ 100 \ 500 \ 20000 \ 65000]$ нс. Параметры этих импульсов взяты за основу сигналов в имитационной модели.

Реальные записи всегда содержат шумы наблюдения, поэтому это следует учесть и для имитационной модели. В итоге подобрались следующие шумы наблюдения: для времени прихода импульса $\Delta T_{пр} = 50$ нс, по несущей частоте $\Delta F = 1$ кГц, для длительности импульса $\Delta \tau_{и} = 10$ нс.

Также для модели следует учесть, что радиолокационные импульсы могут быть не приняты радиолокатором, а значит пропущены. Учтем это для модели введя число импульсов, которые могут быть пропущены $N_{и.пр} = 10$.

Кроме шума наблюдения существуют также отклонения от мгновенных значений. Положим следующее: девиация частоты в долях от несущей

принимает равновероятные значения в следующем диапазоне целых чисел $\Delta F_d = 0.001 \dots 0.005$ (0.1 ... 0.5%), девиация длительности импульса в долях $\Delta \tau_{и.д.} = 0.001 \dots 0.005$ (0.1 ... 0.5%).

4.2. Формирование выборки

Задача следующая. Требуется сформировать датасет в котором будут присутствовать повторяющиеся несколько раз сигналы с одинаковыми параметрами – паттерны. Именно эти паттерны должен выделить алгоритм DBSCAN при кластеризации.

Формирование датасета происходит следующим образом. Задаются параметры импульса:

1. Формируется случайным образом период между импульсами:

$$\Delta = T_{и.мин} \cdot f([1, 10]) + g(0, \Delta T_{пр}) \quad (4.1)$$

Где $f([1,10])$ – непрерывное равномерное распределение целых чисел, $g(0, \Delta T_{пр})$ – нормальный закон распределения с нулевым средним значением и дисперсией $T_{пр}$.

2. Формируется несущая частота импульса:

$$F_0 = f(F) \cdot \left(1 + (-1) \cdot f([0, 1]) \cdot f(\Delta F_d) \right) \quad (4.2)$$

3. Формируется случайным образом длительность импульса:

$$\tau_0 = f(\tau) \cdot \left(1 + (-1) \cdot f([0, 1]) \cdot f(\Delta \tau_{и.д.}) \right) \quad (4.3)$$

4. Далее задается длина паттерна L

5. Формируется датасет из $N = 10000$ элементов, где случайным образом распределены паттерны. К параметрам импульсов добавляется шум наблюдения, распределенный по нормальному закону.

4.3. Результаты эксперимента

В данном разделе описано применение алгоритма кластеризации DBSCAN ко входным данным, смоделированным на имитационной модели.

Задача проанализировать работу алгоритма кластеризации. Посмотреть как влияют параметры DBSCAN *epsilon* и *minpts* на результат кластеризации.

Выберем длину паттерна $L = 10$, который хотим выделить в выборке длиной $N = 10000$ импульсов. Данный паттерн встречается 89 раз в сформированном датасете. Положение первого импульса паттерна в датасете можно отслеживать.

Требуется аналитически подобрать число *minpts*- минимальное количество соседей. Пусть $m = 10$ – число равное числу паттернов в выборке.

Чтобы оценить значение ε -окрестности вычисляется среднее расстояние по m до точки входных данных, как метрика используется евклидовое расстояние (формула 3.1). Затем сортируются полученные значения по возрастанию и строим график k -расстояний.

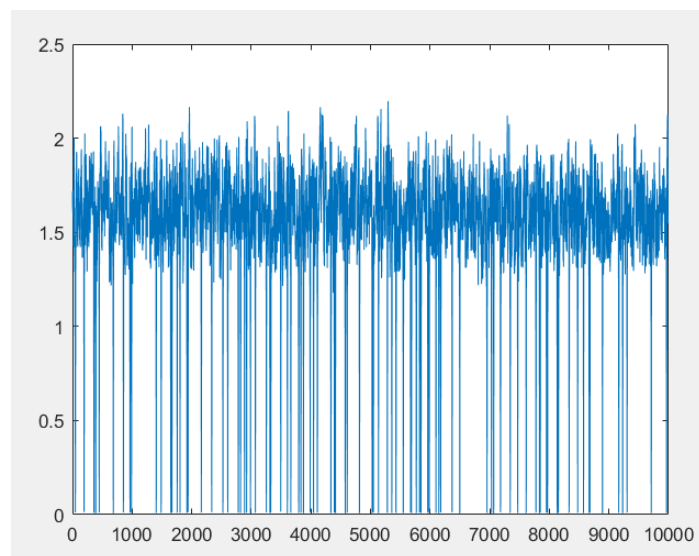


Рисунок 4-1 Евклидовое расстояние для каждой пары наблюдений, для последней точки $m = 10$.

На рисунке 4-1 показано евклидовое расстояние для каждой пары наблюдений. Видно, как выделяются отдельные расстояния, которые расположены ниже среднего значения 1.7. Это в явном виде те самые паттерны, которые требуется найти во входном датасете. Все последующие результаты проводились для одного и того же сформированного датасета.

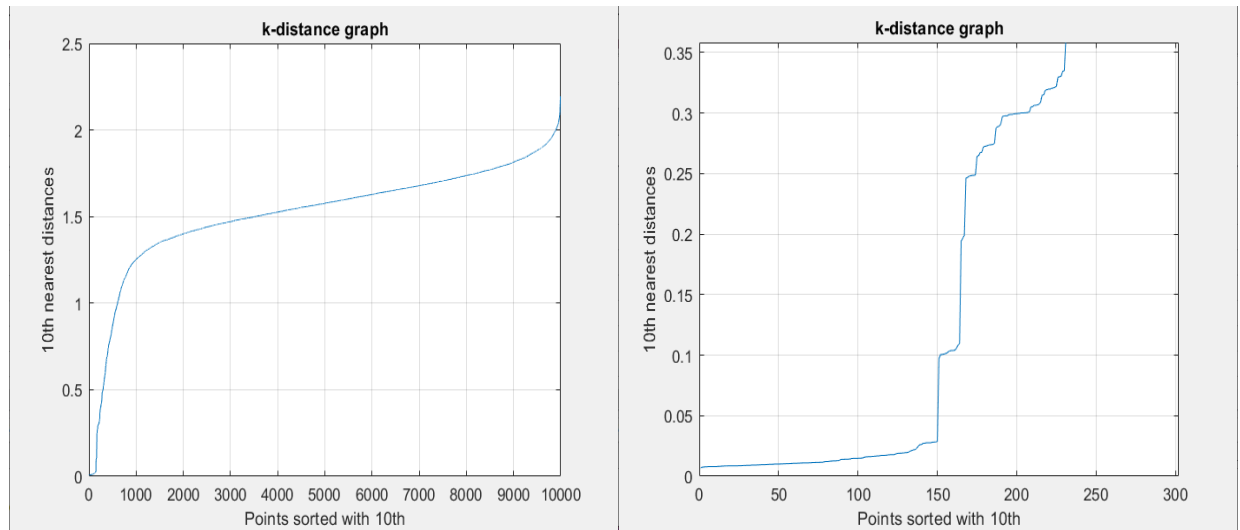


Рисунок 4-2 График k -расстояний для $m = 10$. График справа – первый резкий перегиб график.

Таблица 4-1 Результаты кластеризации для $m = 10$.

$minpts = 10$ минимальное количество соседей	$\varepsilon = 0.01$	$\varepsilon = 0.011$	$\varepsilon = 0.012$	$\varepsilon = 0.013$	$\varepsilon = 0.014$	$\varepsilon = 0.015$
Число найденных кластеров и количество элементов в нём	1x81	1x84	1x85 1x10	1x89 1x13 1x11	1x89 1x17 1x11	1x89 1x17 1x13

По рисунку 4-2 виден первый перегиб графика k -расстояний, в этой области и выбирается значение ε -окрестности. Результаты подбора ε приведены в таблице 4-1. Получилось, что чем больше значение ε , тем большее количество кластеров выделяет алгоритм. Причем, анализируя полученные кластеры – это паттерны, которые отличаются от истинного, положением прихода первого импульса, а именно сдвинуты вправо. Также по

таблице 4-1 видно, что алгоритм не нашел исключительно один кластер во всей выборке, поэтому требуется проверить алгоритм, при других значениях *minpts*- минимальных количества соседей.

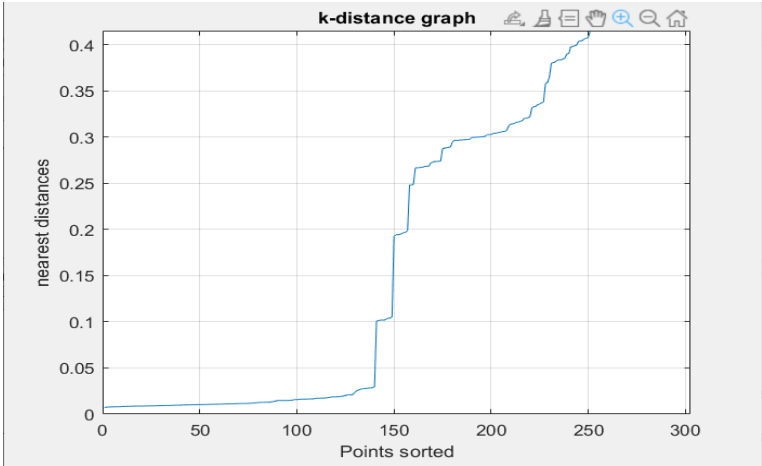


Рисунок 4-1 График *k*-расстояний для *m* = 11.

Таблица 4-2 Результаты кластеризации для *m* = 11.

<i>minpts</i> = 11 минимальное количество соседей	$\varepsilon = 0.0125$	$\varepsilon = 0.0126$
Число найденных кластеров и количество элементов в нём	1x85	1x88 1x11

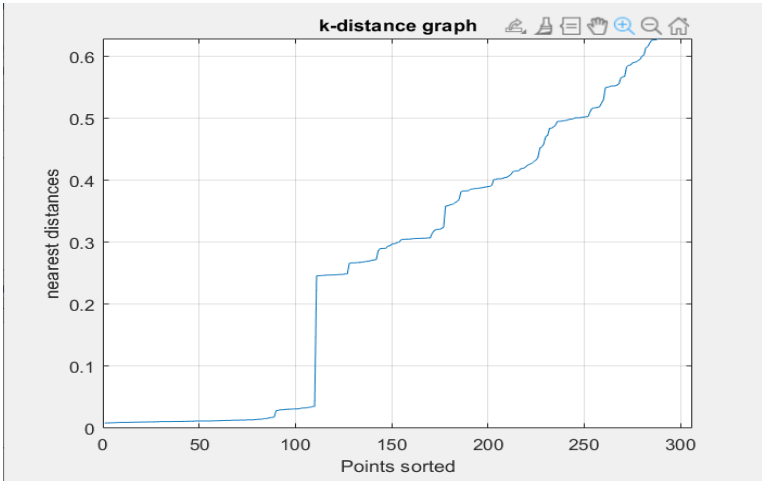


Рисунок 4-2 График *k*-расстояний для *m* = 20.

Таблица 4-3 Результаты кластеризации для *m* = 20.

<i>minpts</i> = 20 минимальное количество соседей	$\varepsilon = 0.015$	$\varepsilon = 0.02$	$\varepsilon = 0.025$	$\varepsilon = 0.03$
---	-----------------------	----------------------	-----------------------	----------------------

Число найденных кластеров и количество элементов в нём	1x89	1x89	1x89	1x89 1x21
--	------	------	------	--------------

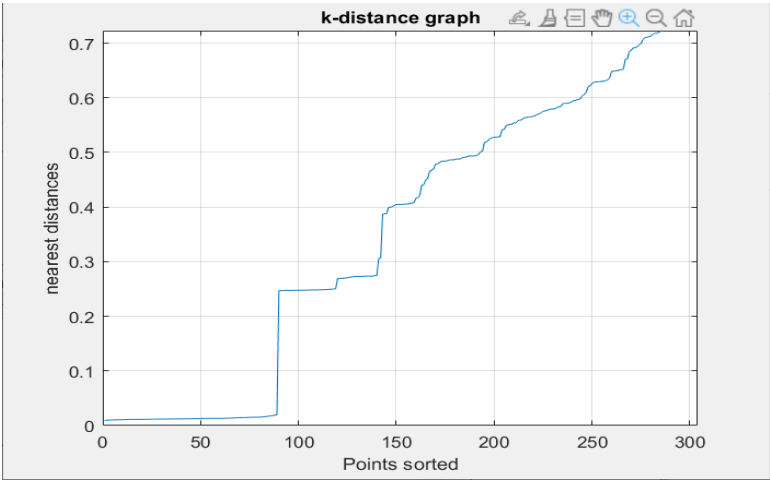


Рисунок 4-3 График k -расстояний для $m = 30$.

Таблица 4-4 Результаты кластеризации для $m = 30$.

$minpts = 30$ минимальное количество соседей	$\varepsilon = 0.015$	$\varepsilon = 0.02$	$\varepsilon = 0.025$	$\varepsilon = 0.03$	$\varepsilon = 0.2$	$\varepsilon = 0.3$
Число найденных кластеров и количество элементов в нём	1x89	1x89	1x89	1x89	1x89	1x89 1x60

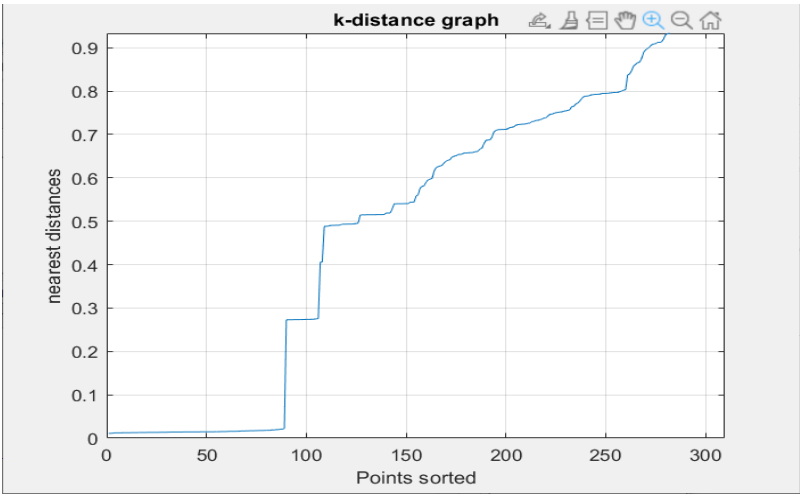


Рисунок 4-4 График k -расстояний для $m = 50$.

Таблица 4-5 Результаты кластеризации для $m = 50$.

$minpts = 50$ минимальное количество соседей	$\varepsilon = 0.2$	$\varepsilon = 0.27$	$\varepsilon = 0.29$	$\varepsilon = 0.3$	$\varepsilon = 0.4$	$\varepsilon = 0.5$
Число найденных кластеров и количество элементов в нём	1x89	1x89	1x89 1x51	1x89 1x53	1x89 1x57	1x79 1x89

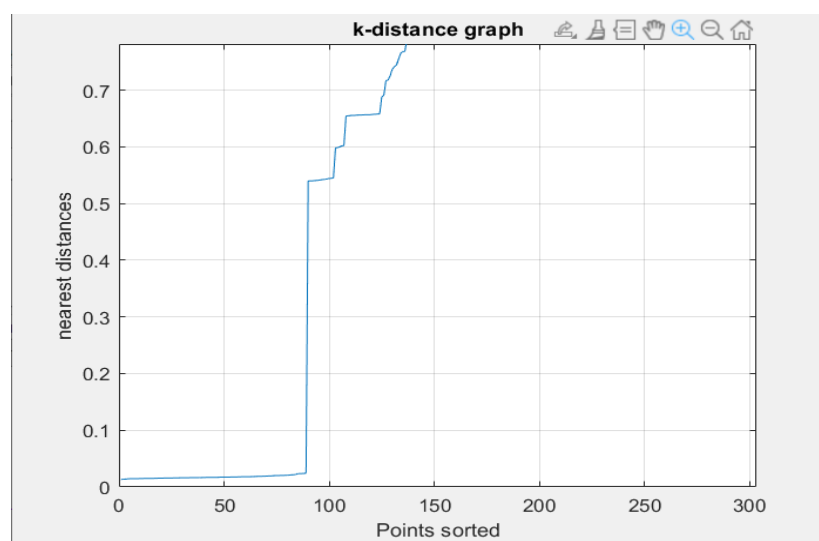


Рисунок 4-5 График k -расстояний для $m = 70$

Таблица 4-6 Результаты кластеризации для $m = 70$.

$minpts = 70$ минимальное количество соседей	$\varepsilon = 0.3$	$\varepsilon = 0.5$	$\varepsilon = 0.59$	$\varepsilon = 0.6$	$\varepsilon = 0.7$
Число найденных кластеров и количество элементов в нём	1x89	1x89	1x89	1x89 1x70	1x77 1x89 1x71

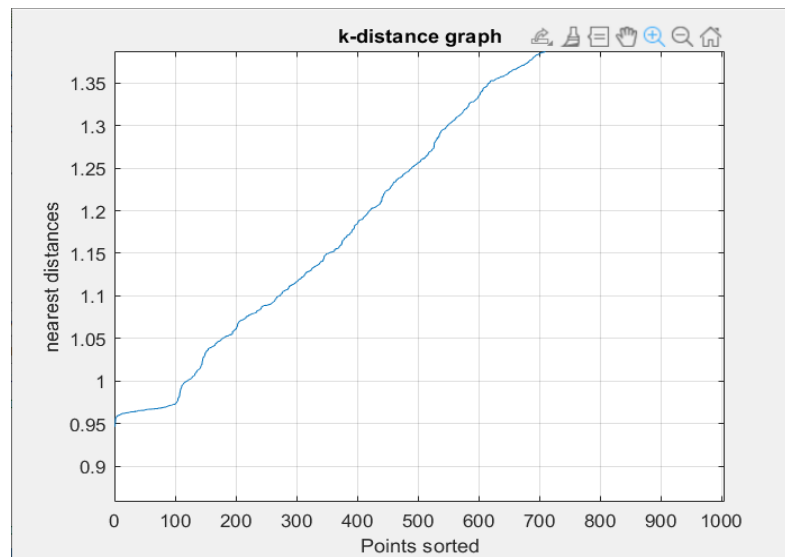


Рисунок 4-6 График k -расстояний для $m = 100$.

Таблица 4-7 Результаты кластеризации для $m = 100$.

$minpts = 100$ минимальное количество соседей	$\varepsilon = 0.95$	$\varepsilon = 0.96$	$\varepsilon = 1$
Число найденных кластеров и количество элементов в нём	1x135	1x240	1x289

Выводы по полученным результатам. Увеличение параметра m приводит к тому, что алгоритм не способен верно определить количество объектов в кластере, увеличивается значение ε -окрестности. Анализируя результаты оптимальное число точек m для решения данной задачи равно 50.

Следующая задача. Проанализировать, как влияют метрики расстояния на результаты кластеризации. Метрики для анализа: евклидовое расстояние, квадрат евклидового расстояния, расстояние городских кварталов, расстояние Чебышева. Сравним результаты для $m = 20$ и для $m = 50$.

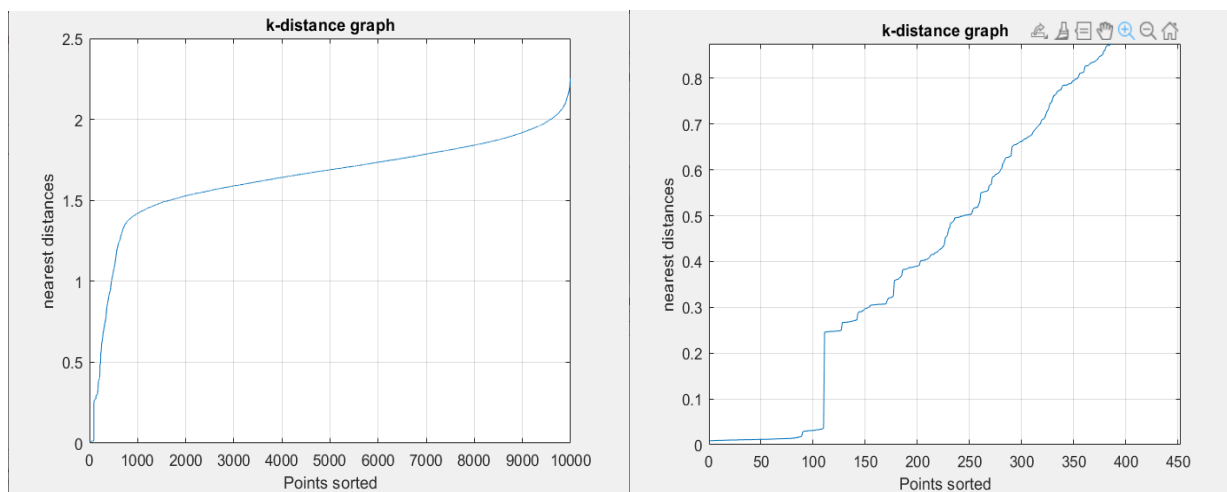


Рисунок 4-7 График k -расстояний для $m = 20$ и евклидова метрика расстояния

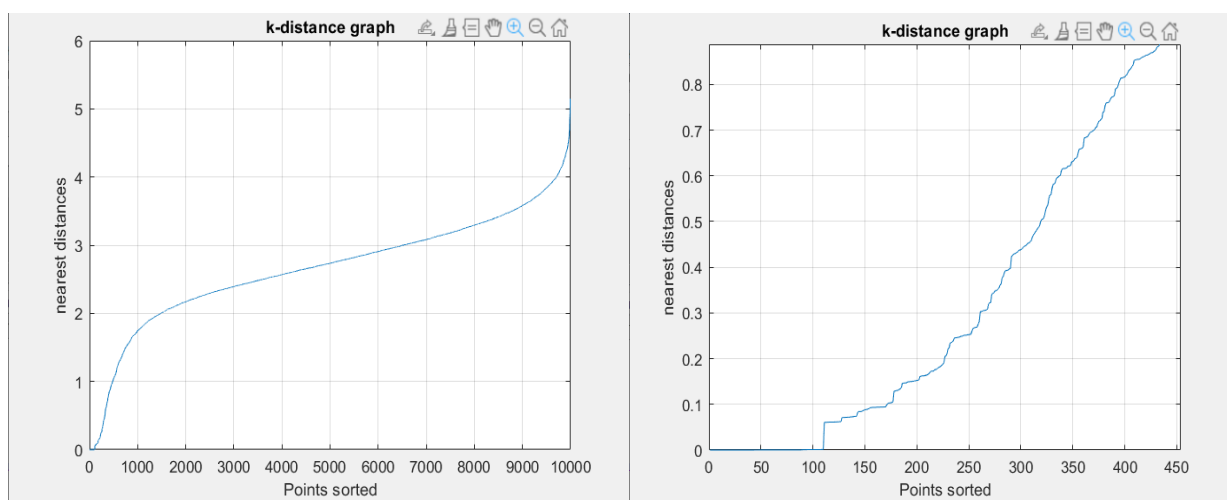


Рисунок 4-8 График k -расстояний для $m = 20$ и квадратичная евклидова метрика расстояния

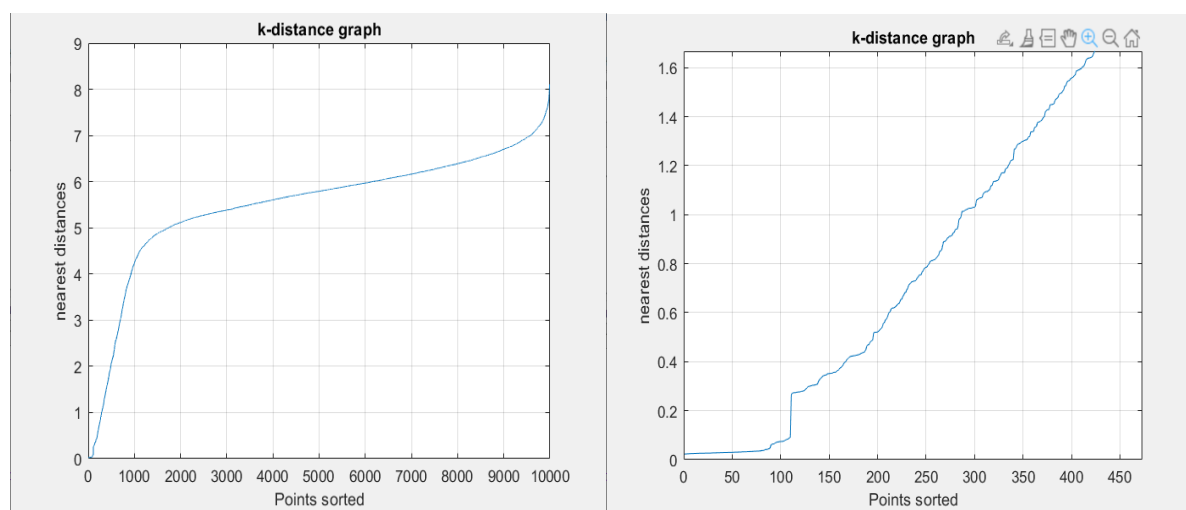


Рисунок 4-9 График k -расстояний для $m = 20$ и метрика расстояния городских кварталов

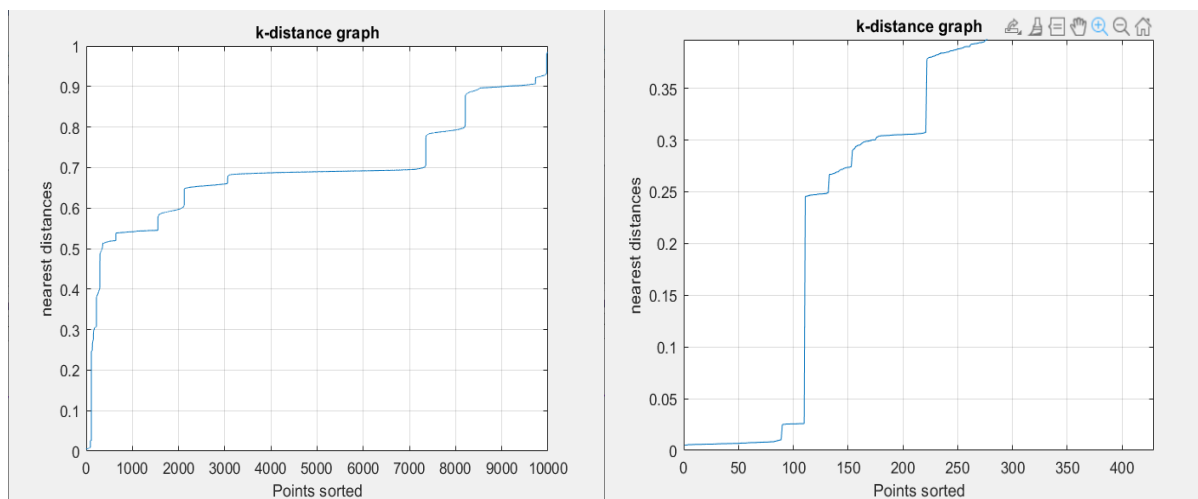


Рисунок 4-10 График k -расстояний для $t = 20$ и метрика расстояния Чебышева

Таблица 4-5 Результаты кластеризации для $t = 20$ и различных метриках расстояний

$minpts = 20$ минимальное количество соседей	$\varepsilon = 0.015$	$\varepsilon = 0.02$	$\varepsilon = 0.025$	$\varepsilon = 0.03$
Евклидово расстояние	1x89	1x89	1x89	1x89 1x21
Квадрат евклидова расстояния	1x89	1x89	1x89	1x89 1x21
Расстояние городских кварталов	1x89	1x89	1x89	1x89 1x21
Расстояние чебышева	1x89	1x89	1x89	1x89 1x21

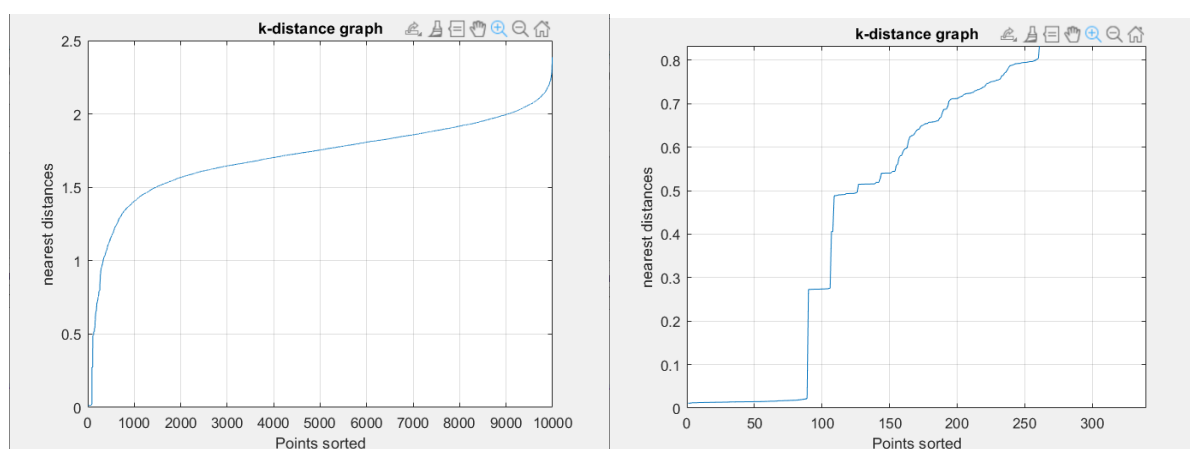


Рисунок 4-11 График k -расстояний для $t = 50$ и евклидова метрика расстояния

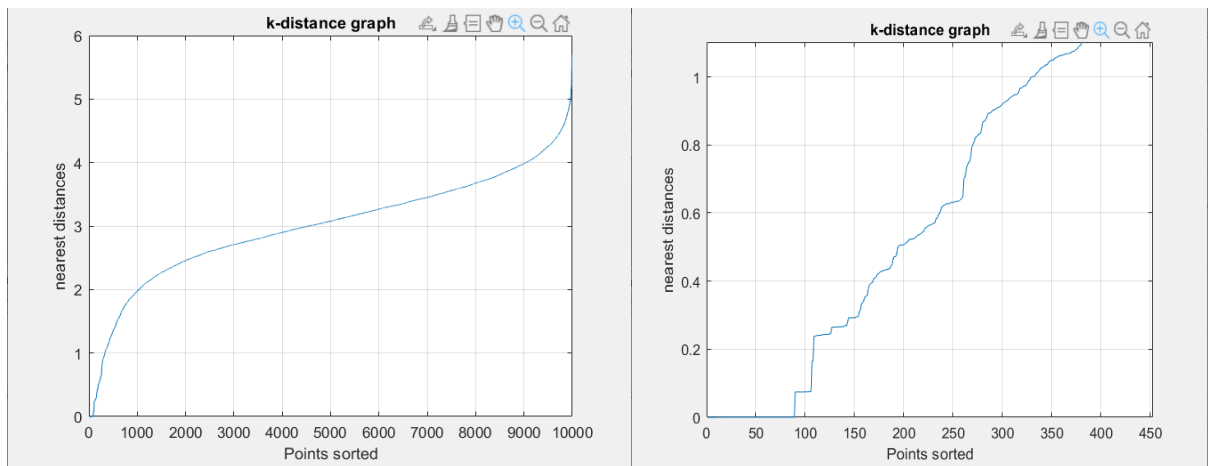


Рисунок 4-12 График k -расстояний для $m = 50$ и квадратичная евклидова метрика расстояния

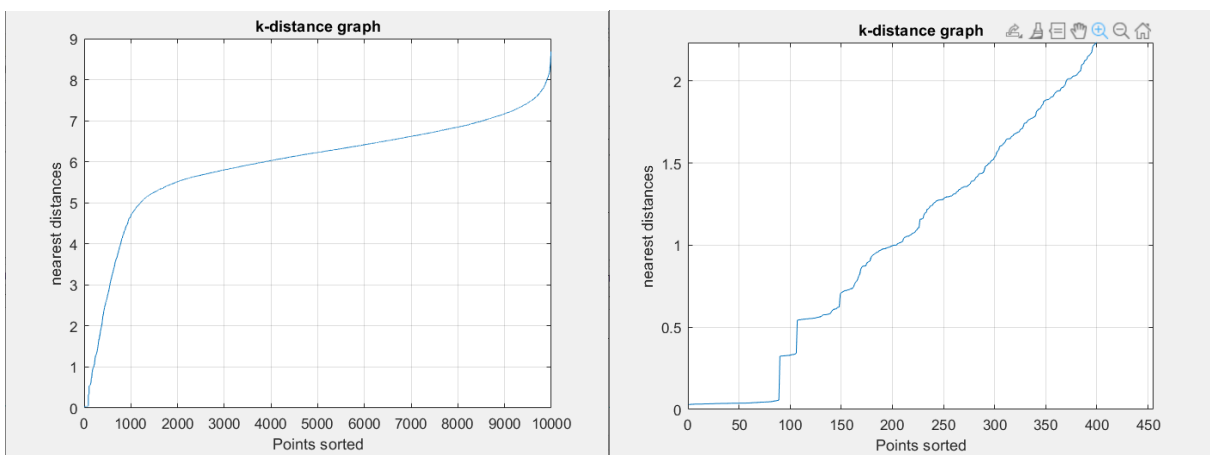


Рисунок 4-13 График k -расстояний для $m = 50$ и метрика расстояние городских кварталов

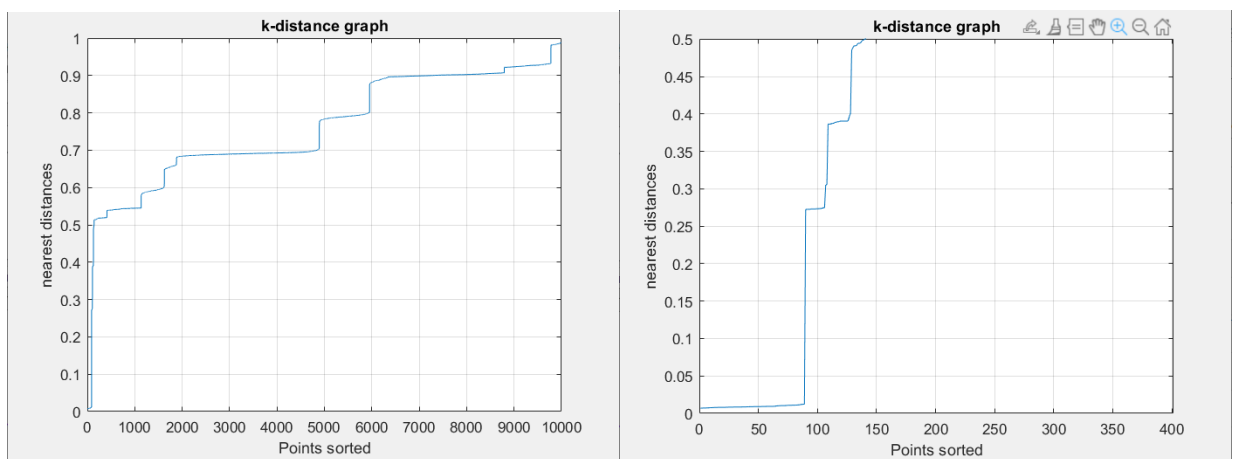


Рисунок 4-14 График k -расстояний для $m = 50$ и метрика расстояния Чебышева

Таблица 4-6 Результаты кластеризации для $m = 50$

$minpts = 50$ минимальное количество соседей	$\varepsilon = 0.2$	$\varepsilon = 0.27$	$\varepsilon = 0.29$	$\varepsilon = 0.3$	$\varepsilon = 0.4$	$\varepsilon = 0.5$
Евклидово расстояние	1x89	1x89	1x89 1x51	1x89 1x53	1x89 1x57	1x79 1x89
Квадрат евклидова расстояния	1x89	1x89	1x89 1x51	1x89 1x53	1x89 1x57	1x79 1x89
Расстояние городских кварталов	1x89	1x89	1x89 1x51	1x89 1x53	1x89 1x57	1x79 1x89
Расстояние чебышева	1x89	1x89	1x89 1x51	1x89 1x53	1x89 1x57	1x79 1x89

Делаем вывод, что выбор метрики для данной задачи не влияет на результаты кластеризации.

4.4. Результаты работы алгоритма кластеризации DBSCAN

Подводя итоги результатов моделирования и работы алгоритма кластеризации DBSCAN можно подчеркнуть следующее:

1. На фоне шума алгоритм правильно выделяет положение паттернов в моделируемой выборке.
2. Оптимальное число точек m -ближайших соседей для данной задачи является 50, с размером ε -окрестности порядка 0.2.
3. Метрика расстояния никак не влияет на результат кластеризации для данной задачи
4. Увеличение числа точек m , ведет к увеличению значения параметра ε -окрестности.
5. При увеличении параметра ε -окрестности алгоритм находит большее количество кластеров.

Заключение

В ходе выполнения научно-исследовательской работы обзор анализов обработки радиолокационной информации. Было рассмотрено, что применение аппарата искусственных нейронных сетей к системам обработки радиолокационной информации считается актуальным и многообещающим направлением. В данной работе было предложено применить искусственную нейронную сеть на этапе обнаружения отметок от целей и их кластеризации. Решение такой задачи позволит в будущем подбирать более эффективные алгоритмы вторичной (траекторной) обработки целей.

Для решения задачи кластеризации отметок от цели был проведен обзор основных принципов нейросетевых технологий. Большое внимание было уделено алгоритмам кластеризации, в особенности алгоритму DBSCAN – плотностному алгоритму пространственной кластеризации с присутствием шума.

В данной работе была смоделирована имитационная модель элементарных радиоимпульсов, основанная на анализе реальных записей обнаруживаемых элементарных импульсов с многопозиционного пассивного радиолокационного комплекса. Применение алгоритма кластеризации к данным, сформированным на модели, показало, что алгоритм кластеризации DBSCAN работает, и верно определяются паттерны в выборке. Был произведен анализ работы алгоритма DBSCAN от его настраиваемых параметров m -ближайших соседей и от ε -окрестности.

Дальнейшая работа в данной направлении – это кластеризовать реальные записи с многопозиционного пассивного радиолокационного комплекса.

Список литературы

1. Галушкин А.И. Нейрокомпьютеры (Нейрокомпьютеры и их применение. Кн. 3). М.: ИПРЖР, 2000. 528 с.
2. Кузьмин С.З. Основы проектирования систем цифровой обработки радиолокационной информации. М.: Сов. радио, 1986. 346 с.
3. Кузьмин С.З. Основы цифровой обработки радиолокационной информации. М.: Сов. радио, 1974. 432 с.
4. Кузьмин С.З. Цифровая обработка радиолокационной информации. М.: Сов. радио, 1967. 400 с.
5. Левин Б.Р. Теоретические основы статистической радиотехники. 3-е изд. перераб. и доп. М.: Радио и связь, 1989. 656 с.
6. Перов А.И. Оптимальный алгоритм дискретного сопровождения многих целей идентификацией измерений // Радиотехника, No1, 2003.
7. Петухов СИ., Степанов А.Н. Эффективность ракетных средств ПВО. М.: Воениздат, 1976. 104 с.
8. Попов Г.П. Инженерная психология в радиолокации. М.: Сов. Радио, 1971. 143 с.
9. Радиолокационные системы: Основы построения и теория. Справочник / Под ред. Я.Д. Ширмана. М.: ЗАО "МАКВИС", 1998. 828 с.
10. Татузов А.Л., Чухлеб Ф.С. Использование нейросетевой технологии при обработке радиолокационной информации // Информационные Технологии. 1999, No 1, С. 25-33.
11. Татузов А.Л. Нейронные сети в задачах радиолокации. Кн. 28. – М.: Радиотехника, 2009. – 432 с.: ил. (Научная серия «Нейрокомпьютеры и их применение»)
12. Тихонов В.И., Харисов В.Н. Статистический анализ и синтез радиотехнических устройств и систем. М.: Радио и связь, 1991. 608 с.
13. Хайкин С. Нейронные сети: полный курс - Neural Networks: A Comprehensive Foundation. 2-е. М.: «Вильямс», 2006. 1104 с.
14. Ширман Я.Д., Манжос В.Н. Теория и техника обработки радиолокационной информации на фоне помех. М.: Радио и связь, 1981. 416 с.
15. Anant Ram, Sunita J alal, Anand S. J alal, Manoj Kumar. 2010. A Density Based Algorithm for Discovering Density Varied Clusters in Large Spatial Databases. International Journal of Computer Application
16. Barton D.K. Radar system analysis and modeling. – Artech House, Boston, MA, 2005.

17. Cheng B. and Titterington, D.M. Neural Networks: A Review from a Statistical Perspective // Statistical Science, 1994, v.9, pp. 2-54.
18. Dillard G.M. Mean-level detection of nonfluctuating signals // IEEE Trans., 1974, v.AES-10, no.6 (Nov. 1974), pp.795-799.
19. Haykin S, Neural Networks: A Comprehensive Foundation, NY: Macmillan, 1994.
20. Michie D., Spiegelhalter D.J., and Taylor C.C. Machine Learning, Neural and Statistical Classification, Ellis Horwood, 1994.