

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное бюджетное образовательное учреждение высшего  
образования  
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий  
(наименование института полностью)

Кафедра «Прикладная математика и информатика»  
(наименование)

01.03.02 Прикладная математика и информатика  
(код и наименование направления подготовки, специальности)

Компьютерные технологии и математическое моделирование  
(направленность (профиль)/ специализация)

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
(БАКАЛАВРСКАЯ РАБОТА)**

на тему «Оптимизация планирования доставки грузов на основе алгоритма  
k-means»

Студент Д.С. Пирогов (И.О. Фамилия) \_\_\_\_\_ (личная подпись)

Руководитель д.т.н., доцент, С.В. Мкртычев  
(ученая степень, звание, И.О. Фамилия)

Консультант Дайнеко М.В.  
(ученая степень, звание, И.О. Фамилия)

## **Аннотация**

Тема выпускной квалификационной работы – «Оптимизация планирования доставки грузов на основе алгоритма k-means».

Для обеспечения бесперебойной и своевременной доставки грузов с высоким качеством обслуживания каждый маршрут доставки необходимо тщательно спланировать и проверить. Как показывает практика, для решения данной задачи необходимо использовать методы интеллектуального анализа данных – Data mining. Одним из таких методов является метод кластеризации данных на основе алгоритма k-means.

Объектом исследования бакалаврской работы является процесс планирования доставки грузов.

Предметом исследования бакалаврской работы является оптимизация планирования доставки грузов.

Цель бакалаврской работы – оптимизация планирования доставки грузов на основе алгоритма k-means.

Методы исследования – интеллектуальный анализ данных, алгоритм k-means.

Практическая значимость бакалаврской работы заключается в разработке программы оптимизации планирования доставки грузов на основе алгоритма k-means, обеспечивающей высокую эффективность процесса доставки.

Результаты бакалаврской работы представляют научно-практический интерес и могут быть рекомендованы для разработчиков программ для поддержки транспортных задач на основе методов интеллектуального анализа данных.

Выпускная квалификационная работа состоит из 44 страниц текста, 20 рисунков, 4 таблиц и 20 источников.

## **Abstract**

The topic of the given graduation work is Optimization of cargo delivery planning based on the k-means algorithm.

To ensure uninterrupted and timely delivery of goods with high quality service, each delivery route must be carefully planned and verified. As practice shows, to solve this problem, it is necessary to use data mining methods. One such method is the data clustering method based on the k-means algorithm.

The object of study of the graduation work is the process of planning the delivery of goods.

The subject of study of the graduation work is the optimization of cargo delivery planning.

The aim of the graduation work is to optimize cargo delivery planning based on the k-means algorithm.

Research methods: Data mining, k-means algorithm.

The practical significance of the graduation work lies in the development of a program for optimizing the planning of cargo delivery based on the k-means algorithm, which ensures a high efficiency of the delivery process.

The results of the graduation work are of scientific and practical interest and can be recommended for program developers to support transport problems based on data mining methods.

The graduation work consists of an explanatory note on 44 pages including 20 figures, 4 tables, the list of 20 references.

## Оглавление

Введение.....	5
Глава 1 Постановка задачи исследования и анализ алгоритма k-means .....	7
1.1 Постановка задачи исследования.....	7
1.2 Методы решения задачи кластеризации .....	8
1.3 Анализ алгоритма k-means.....	12
Глава 2 Разработка решения для оптимизации планирования доставки грузов на основе алгоритма k-means .....	18
Глава 3 Разработка программы оптимизации планирования доставки грузов на основе алгоритма k-means .....	26
3.1 Программа «1С:TMS Логистика. Управление перевозками» .....	26
3.2 Программа «Простые маршруты» .....	27
3.3 Онлайн-сервис «Калькулятор для кластеризации по алгоритму K-средних».....	29
3.4 Выбор платформы и разработка программы .....	30
Заключение .....	39
Список используемой литературы и используемых источников.....	42

## Введение

В условиях пандемии резко возросла нагрузка на службы доставки грузов по заказам клиентов.

Для обеспечения бесперебойной и своевременной доставки грузов с высоким качеством обслуживания каждый маршрут доставки необходимо тщательно спланировать и проверить.

С оперативной точки зрения задача оптимизации планирования доставки груза состоит в обеспечении эффективности процесса доставки за счет экономии топлива и сокращения времени простоя.

Как показывает практика, для решения данной задачи необходимо использовать методы интеллектуального анализа данных – Data mining.

Одним из таких методов является метод кластеризации данных на основе алгоритма k-means.

Решение задачи оптимизации планирования доставки грузов на основе алгоритма k-means представляет актуальность и научно-практический интерес.

Объектом исследования бакалаврской работы является процесс планирования доставки грузов.

Предметом исследования бакалаврской работы является оптимизация планирования доставки грузов.

Цель бакалаврской работы – оптимизация планирования доставки грузов на основе алгоритма k-means.

Для достижения данной цели необходимо выполнить следующие задачи:

- выполнить постановку задачи оптимизации планирования доставки грузов и проанализировать алгоритм кластеризации данных k-means на предмет решения данной задачи;
- разработать решение для оптимизации планирования доставки грузов на основе алгоритма k-means;
- разработать программу для оптимизации планирования доставки

грузов на основе предлагаемого решения и подтвердить ее работоспособность.

Методы исследования – интеллектуальный анализ данных, алгоритм k-means.

Практическая значимость бакалаврской работы заключается в разработке программы оптимизации планирования доставки грузов на основе алгоритма k-means, обеспечивающей высокую эффективность процесса доставки.

Данная работа состоит из введения, трех глав, заключения и списка используемой литературы.

Первая глава работы посвящена постановке задачи исследования и анализу алгоритма k-means.

Вторая глава работы посвящена разработке решения для оптимизации планирования доставки грузов на основе алгоритма k-means.

В третьей главе рассматривается процесс разработки программы оптимизации планирования доставки грузов на основе предлагаемого решения.

В заключении описываются результаты выполнения выпускной квалификационной работы.

Бакалаврская работа состоит из 44 страниц текста, 20 рисунков, 4 таблиц и 20 источников.

# Глава 1 Постановка задачи исследования и анализ алгоритма k-means

## 1.1 Постановка задачи исследования

По мнению экспертов, основными проблемами компаний, занимающихся доставкой грузов являются следующие:

- своевременная доставка;
- оптимизация затрат;
- безопасность и инфраструктура и др.

Однако безусловно ключевым показателем эффективности логистических услуг является своевременная доставка груза. Задержка запланированной (ожидаемой) доставки увеличивает недовольство клиентов.

Чтобы предотвратить задержки, поставщики логистических услуг в значительной степени полагаются на автоматизированные решения планирования доставки грузов, которые опираются на эффективные алгоритмы оптимизации планирования [16].

Рассмотрим постановку задачи оптимизации планирования доставки грузов.

«Поставлена задача нахождения для заданного парка автотранспорта маршрутов доставки грузов от поставщика потребителям, обеспечивающее снижение суммарных затрат на перевозку товаров за счет минимизации пробега.

Рассматривается сетка дорог с большим количеством узлов, через которые должны пройти маршруты движения транспорта.

Сетке дорог ставится в соответствие ориентированный граф, вершинами которого являются узлы данной сетки, а ребрами – отрезки дорог между узлами (движение по дороге может быть односторонним).

Каждому ребру приписывается длина – расстояние между соответствующими узлами» [13].

Ищется набор оптимальных маршрутов, начинающихся и заканчивающихся в заданных точках, и ограниченных некоторой функцией от длин ребер графа, которая может учитывать физическую длину маршрута (километраж), либо время движения транспорта, либо стоимостные характеристики маршрута движения.

«Таким образом, расстояния между объектами задаются квадратной матрицей расстояний:

$$A = [a(i, j)] \text{ размерности } n \times n, \quad (1)$$

где  $a(i, j)$  – расстояние от пункта  $i$  до пункта  $j$ .

Матрица расстояний не является симметричной (одностороннее движение, сложные транспортные развязки и т.д.).

Грузоподъемность транспорта не учитывается.

Необходимо решить задачу разбиения региона на компактные зоны обслуживания (группировка объектов-получателей для каждого маршрута).

Задача кластеризации реализуется набором методов (алгоритмов), каждый из которых осуществляет разбиения региона на компактные зоны обслуживания» [13].

Таким образом, поставленная задача оптимизации планирования доставки грузов может быть рассматриваться как задача кластеризации и описана с помощью метода наименьших квадратов [18]:

$$\sum_{j=1}^k \sum_{i=1}^{n_j} |x_i - s_j|^2 \rightarrow \min \quad (2)$$

для всех  $s_j$  и  $k$ .

Численная реализация этой задачи называется методом  $k$ -средних [7].

## 1.2 Методы решения задачи кластеризации

Кластеризация - один из наиболее распространенных методов исследовательского анализа данных, используемых для получения интуитивного представления о структуре данных.

Кластеризацию можно определить, как задачу идентификации групп в данных, так чтобы точки данных в одной группе (кластере) были очень похожи и при этом сильно отличались от точек данных в других кластерах [17].

Другими словами, мы пытаемся найти однородные группы в данных, чтобы точки данных в каждом кластере были как можно более похожими в соответствии с такой мерой сходства, как расстояние на основе евклидово расстояние или расстояние на основе корреляции.

Решение о том, какую меру сходства использовать, зависит от приложения.

Формально кластеризация описывается следующим образом [6].

Объект- элементарная группа данных, с которой работают алгоритмы кластеризации.

С каждым объектом связан вектор характеристик:

$$\mathbf{x} = (x_1, \dots, x_d), \quad (3)$$

где:

$x_i$  – отдельные характеристики объекта;

$d$  - размерность характеристик объекта.

Множество всех векторов характеристик:

$$\mathbf{æ} = (x_1, \dots, x_n), \quad (4)$$

где:

$$x_i = (x_{i,1}, \dots, x_{i,d})$$

Таким образом, кластер – это подмножество объектов с близкими характеристиками из множества  $\mathbf{æ}$ .

Расстояние  $d(x_i, x_j)$  между объектами  $x_i$  и  $x_j$  – результат применения выбранной метрики в пространстве характеристик.

Общий алгоритм кластеризации представлен на рисунке 1.

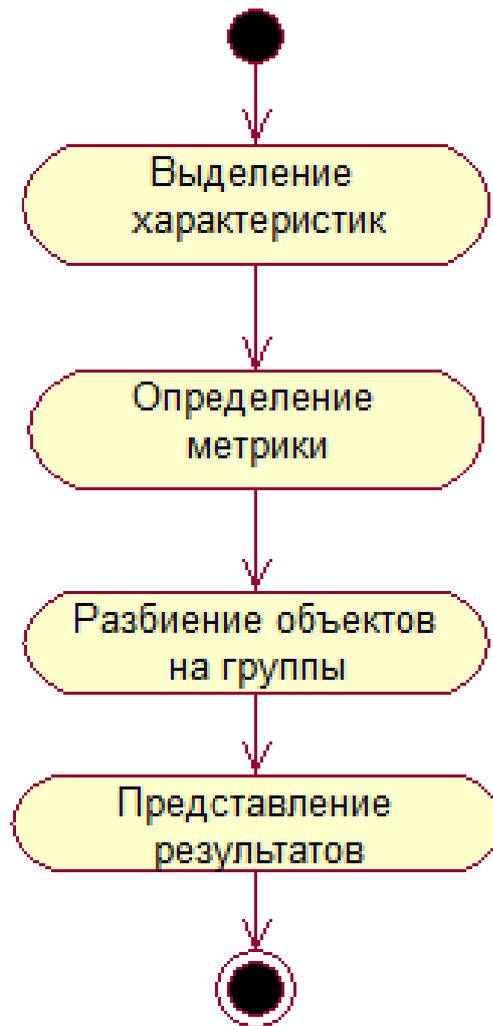


Рисунок 1 - Общий алгоритм кластеризации

Кластерный анализ относится к методам Data mining или машинного обучения и может быть выполнен на основе функций или на основе образцов.

Кластеризация выполняется с помощью специальных алгоритмов, которые можно разделить на следующие классы [19]:

- алгоритмы на основе плотности. При кластеризации на основе плотности данные группируются по областям с высокой концентрацией точек данных, окруженным областями с низкой концентрацией точек данных. В основном алгоритм находит места, в которых много точек данных, и вызывает эти кластеры. При этом кластеры могут иметь любую форму;
- алгоритмы на основе распределения. При кластеризации на основе

распределения все точки данных считаются частями кластера на основе вероятности того, что они принадлежат данному кластеру. Это работает так: есть центральная точка, и по мере увеличения расстояния точки данных от центра вероятность того, что она является частью этого кластера, уменьшается.

- алгоритмы на основе центроидов. Это наиболее распространенный тип алгоритмов. Он немного чувствителен к исходным параметрам, но быстрый и эффективный. Эти типы алгоритмов разделяют точки данных на основе нескольких центроидов в данных. Каждая точка данных назначается кластеру на основе ее квадрата расстояния от центроида. Это наиболее часто используемый тип кластеризации.
- алгоритмы на основе иерархии. Иерархическая кластеризация обычно используется для иерархических данных, например, из базы данных компании или таксономий. Алгоритм строит дерево кластеров, поэтому все организовано сверху вниз. Это более строгий, чем другие типы кластеризации, но он идеально подходит для определенных типов наборов данных.

Правило, по которому производится вычисление расстояния, называется метрикой.

Наиболее часто в алгоритмах кластеризации в качестве метрики используется евклидово расстояние, которое определяется по формуле [3]:

$$d_E(x, y) = \sqrt{\sum_i (x_i - y_i)^2}, \quad (5)$$

где:

$x = (x_1, x_2, \dots, x_m)$ ,  $y = (y_1, y_2, \dots, y_m)$  – векторы значений двух записей.

Следует отметить, что в отличие от обучения с учителем, кластеризация считается методом обучения без учителя, поскольку у нас нет достоверных данных для сравнения результатов алгоритма кластеризации с истинными метками для оценки его производительности.

Можно только попытаться исследовать структуру данных, сгруппировав

точки данных в отдельные группы.

Рассмотрим кластеризацию на основе функций.

### 1.3 Анализ алгоритма k-means

Одним из широко применяемых на практике алгоритмов кластеризации является алгоритм k-means [20].

«Алгоритм k-means - это итеративный алгоритм, который пытается разделить набор данных на заранее определенные  $k$ -отдельных неперекрывающихся групп-кластеров, где каждая точка данных принадлежит только одной группе.

Алгоритм k-means пытается сделать точки данных внутри кластера как можно более похожими, но при этом сохраняя кластеры как можно более разными.

Он назначает точки данных кластеру таким образом, чтобы сумма квадратов расстояния между точками данных и центроидом кластера (среднем арифметическим всех точек данных, принадлежащих этому кластеру) была минимальной.

Чем меньше вариаций внутри кластеров, тем более однородные (похожие) точки данных находятся в одном кластере.

Подход, который использует k-means для решения проблемы, называется EM (Expectation-Maximization)-алгоритмом.

EM-алгоритм состоит из итерационного повторения двух шагов. На E-шаге вычисляется ожидаемое значение (expectation) вектора скрытых переменных  $G$  по текущему приближению вектора параметров  $\Theta$ . На M-шаге решается задача максимизации правдоподобия (maximization) и находится следующее приближение вектора  $\Theta$  по текущим значениям векторов  $G$  и  $\Theta$  [6].

Общее представление EM-алгоритма имеет вид:

1: Вычислить начальное приближение вектора параметров  $\Theta$ ;

- 2: повторять
- 3:  $G := EStep(\Theta)$ ;
- 4:  $\Theta := MStep(\Theta, G)$ ;
- 5: пока  $\Theta$  и  $G$  не стабилизируются [6].

Целевая функция кластерной оптимизации будет иметь вид:

$$J = \sum_{i=1}^m \sum_{k=1}^K w_{ik} \|x^i - \mu_k\|^2, \quad (6)$$

где:

$w_{ik} = 1$  для точек данных, принадлежащих  $k$ -му кластеру и  $w_{ik} = 0$  – в противном случае;

$\mu_k$  – центрoид  $k$ -го кластера.

Задача минимизации данной функции по EM-алгоритму состоит из двух шагов.

Сначала дифференцируем  $J$  по  $w_{ik}$  и обновляем назначения кластера (E-шаг). Затем дифференцируем  $J$  по  $\mu_k$  и пересчитываем центрoиды после присвоения кластеров из предыдущего E-шага (M-шаг).

Формально, E-шаг - это:

$$\begin{aligned} \frac{\partial J}{\partial w_{ik}} &= \sum_{i=1}^m \sum_{k=1}^K w_{ik} \|x^i - \mu_k\|^2 \\ \Rightarrow w_{ik} &= \begin{cases} 1, & \text{если } k = \operatorname{argmin}_j \|x^i - \mu_j\|^2 \\ 0, & \text{в противном случае} \end{cases} \end{aligned} \quad (7)$$

Другими словами, определяем точку данных  $x_i$  в ближайший кластер, исходя из суммы квадратов расстояния от центрoида кластера.

Соответственно, M-шаг - это:

$$\begin{aligned} \frac{\partial J}{\partial \mu_k} &= 2 \sum_{i=1}^m w_{ik} (x^i - \mu_k) = 0 \\ \Rightarrow \mu_k &= \frac{\sum_{i=1}^m w_{ik} x^i}{\sum_{i=1}^m w_{ik}} \end{aligned} \quad (8)$$

Это означает пересчет центроида каждого кластера для отражения новых определений.

Здесь следует отметить несколько моментов:

- поскольку алгоритмы кластеризации, включая *k-means*, используют измерения на основе расстояния для определения сходства между точками данных, рекомендуется стандартизировать данные, чтобы иметь среднее значение, равное нулю, и стандартное отклонение, равное единице, поскольку почти всегда функции в любом наборе данных будут иметь разные единицы измерения. например, возраст и доход;
- учитывая итеративный характер *k-means* и случайную инициализацию центроидов в начале алгоритма, разные инициализации могут привести к разным кластерам, поскольку алгоритм *k-means* может застрять в локальном оптимуме и может не сходиться к глобальному оптимуму. Поэтому рекомендуется запускать алгоритм, используя разные инициализации центроидов, и выбирать результаты прогона, которые дали меньшую сумму квадратов расстояния [2].

«Алгоритм *k-means* состоит из следующих шагов:

Шаг 1. Случайным образом выбирается  $k$  объектов обучающей выборки, которые будут служить начальными центрами кластеров.

Шаг 2. Для каждого объектов обучающей выборки определяется ближайший к ней центр кластера. Для этого вычисляется расстояние между объектами и центрами кластеров. Считается, что объект принадлежит тому кластеру, к которому он ближе. В качестве формулы для оценки близости объектов в многомерном пространстве признаков используется одна из известных метрик» [20].

Шаг 3. Как только состав кластеров на данной итерации известен, производится расчёт новых центров кластеров. Это делается путем расчета

средних значений для каждого числового признака по всем объектам рассматриваемого кластера. Например, в двухмерном пространстве координаты центр кластера на основе вошедших в него  $t$  объектов рассчитывается следующим образом:

$$(P_{2ц}, P_{1ц}) = \left( \frac{\sum_1^t P_1(t)}{t}, \frac{\sum_1^t P_2(t)}{t} \right) \quad (9)$$

«Шаг 4. Шаги 2 и 3 повторяются до тех пор, пока не выполнятся один из двух критериев остановки:

- границы кластеров и расположения центров кластеров не перестанет изменяться от итерации к итерации, т.е. на каждой итерации в каждом кластере будет оставаться один и тот же набор записей. На практике алгоритм k-means обычно находит набор стабильных кластеров за несколько десятков итераций;
- достигнут критерий сходимости. Чаще всего используется критерий суммы квадратов ошибок между центром кластера и всеми вошедшими в него объектами:

$$E = \sum_{i=1}^k \sum_{p \in C_i} (p - m_i)^2 \quad (10)$$

где  $p \in C_i$  - произвольная точка данных, принадлежащая кластеру  $C_i$ ,  $m_i$  – центр данного кластера. Иными словами, алгоритм остановится тогда, когда ошибка  $E$  достигнет достаточно малого значения» [20].

Блок-схема алгоритма k-means представлена на рисунке 2.

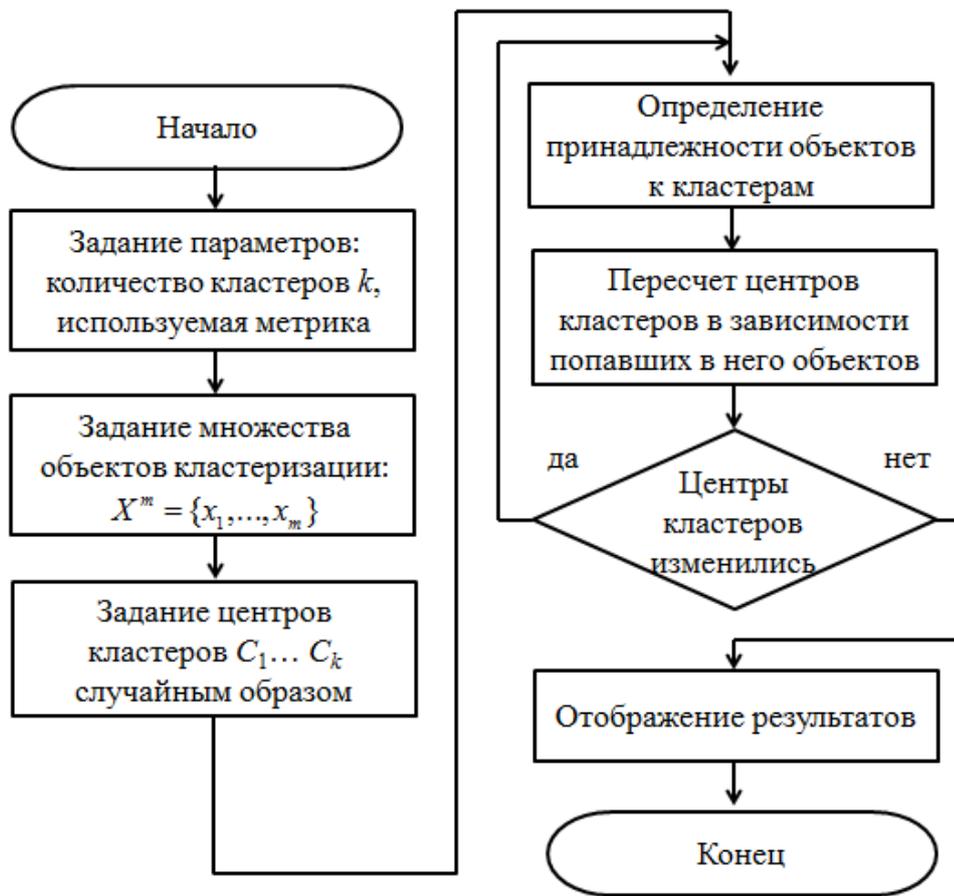


Рисунок 2 – Блок-схема алгоритма k-means

Графическое представление алгоритма k-means показано на рисунке 3.

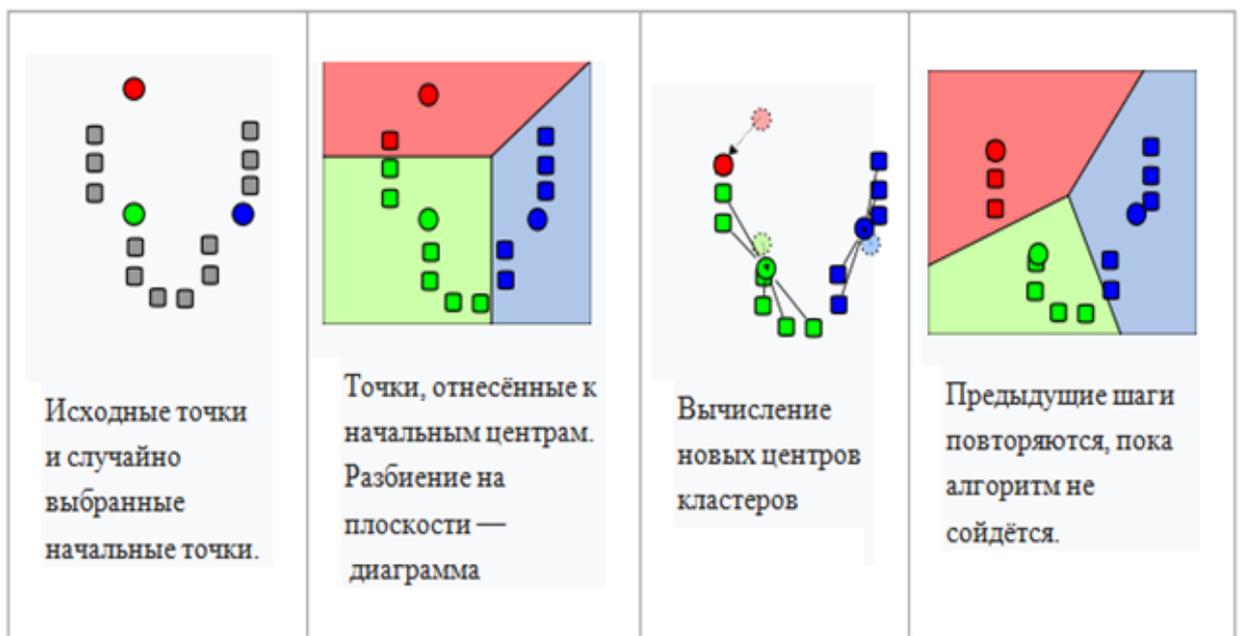


Рисунок 3 - Графическое представление алгоритма k-means

Преимуществом алгоритма k-means является высокая производительность, поскольку все, что мы на самом деле делаем, это вычисляем расстояния между точками и центроидами групп.

Таким образом, он имеет линейную сложность  $O(n)$ .

С другой стороны, у k-means есть недостатки:

- необходимо выбрать количество групп/классов. Это не всегда тривиально, и в идеале с алгоритмом кластеризации мы хотели бы, чтобы он выяснил это за нас, потому что его цель - получить некоторое представление о данных;
- алгоритм начинается со случайного выбора центроидов кластеров и, следовательно, может давать разные результаты кластеризации при разных прогонах алгоритма.

Таким образом, результаты могут быть неповторимыми и противоречивыми.

## **Выводы к главе 1**

Первая глава посвящена постановке задачи на исследование, обзору и анализу методов кластеризации данных и алгоритма k-means.

Результаты проделанной работы позволили сделать выводы, представленные ниже.

Для предотвращения задержек поставщики логистических услуг используют автоматизированные решения планирования доставки грузов, которые опираются на эффективные алгоритмы оптимизации планирования;

Задача оптимизации планирования доставки грузов может быть рассматриваться как задача кластеризации.

Одним из широко применяемых на практике алгоритмов кластеризации является алгоритм k-means. Главным преимуществом данного алгоритма является его высокая производительность.

## Глава 2 Разработка решения для оптимизации планирования доставки грузов на основе алгоритма k-means

Рассмотрим решение задачи оптимизации планирования доставки грузов на следующем примере.

Пусть имеется набор из 8 точек данных в двумерном пространстве (региона), из которого требуется получить два кластера-зоны компактной доставки грузов.

Для решения задачи используем табличный процессор Excel [15].

Наборы точек данных для кластеризации приведены соответственно в таблице 1 и рисунке 4.

Таблица 1 – Набор точек данных для кластеризации

#	X	Y
A	1	3
B	3	3
C	4	3
D	5	3
E	1	2
F	4	2
G	1	1
H	2	1

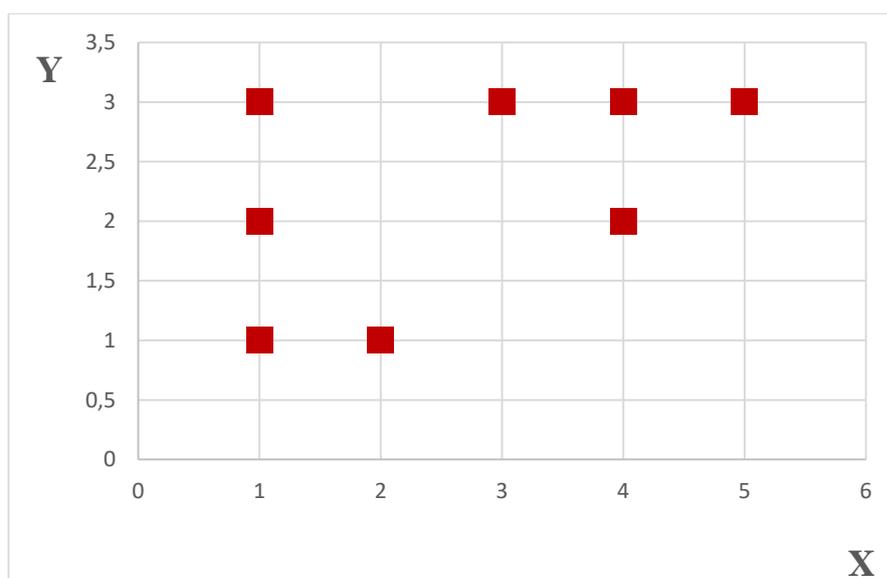


Рисунок 4 – Начальное состояние области кластеризации

Шаг 1.

Задаем число кластеров-зон, на которое требуется разбить исходное множество регионов,  $k=2$ .

Шаг 2.

Выбираем две точки, которые будут центрами кластеров.

Пусть это будут точки  $m_1=(1;1)$  и  $m_2=(2;1)$ .

На рисунке 5 они отмечены знаком ромбика.

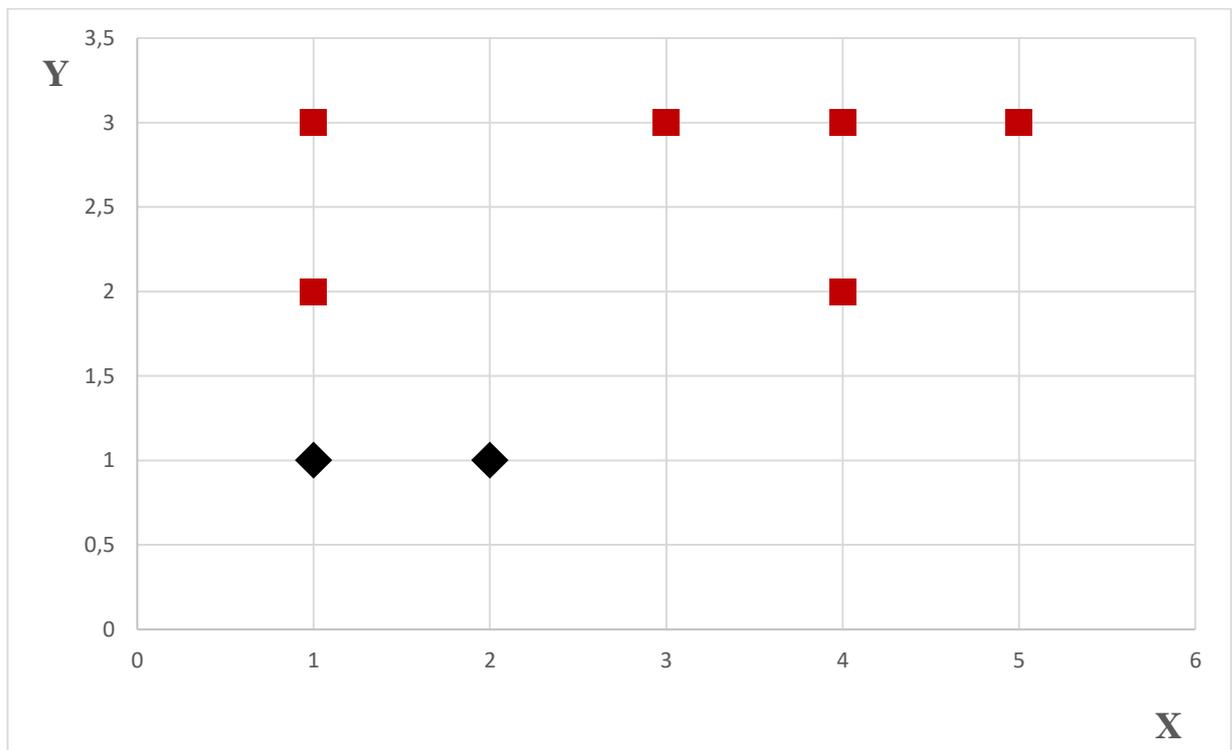


Рисунок 5 – Состояние данных после инициализации

Шаг 3.

Проход 1.

Для каждой точки определяем по формуле (4) ближайший к ней центр кластера с помощью расстояния Евклида.

Формулы и результаты расчета представлены на рисунке 6.

	A	B	C	D	E	F	G
1	#	X	Y	m1	m2	Кластер	
2	A	1	3	2,00	2,24	1	ЕСЛИ(D2<E2;1;2)
3	B	3	3	2,83	2,24	2	
4	C	4	3	3,61	2,83	2	
5	D	5	3	4,47	3,61	2	
6	E	1	2	1,00	1,41	1	
7	F	4	2	3,16	2,24	2	
8	G	1	1	0,00	1,00	1	
9	H	2	1	1,00	0,00	2	

Рисунок 6 – Таблица определения центров для точек данных

Таким образом, кластер 1 содержит точки A, E, G, а кластер 2 – точки B, C, D, F, H.

После определения членов кластеров, рассчитываем сумму квадратичных ошибок:

$$E = \sum_{i=1}^k \sum_{p \in C_i} (p - m_i)^2 = 2^2 + 2,24^2 + 2,83^2 + 3,61^2 + 1 + 2,24^2 + 0 + 0 = 36 \quad (11)$$

Шаг 4.

Проход 1.

Для каждого кластера вычисляется его центроид, и центр кластера перемещаем в него.

Центроид для кластера 1:  $[(1+1+1)/3, (3+2+1)/3] = (1; 2)$ .

Центроид для кластера 2:  $[(3 + 4 + 5 + 4 + 2)/5, (3 + 3 + 3 + 2 + 1)/5] = (3,6; 2,4)$ .

Расположение кластеров и центроидов (обозначены знаком «звездочка») после первого прохода алгоритма представлено на рисунке 7.

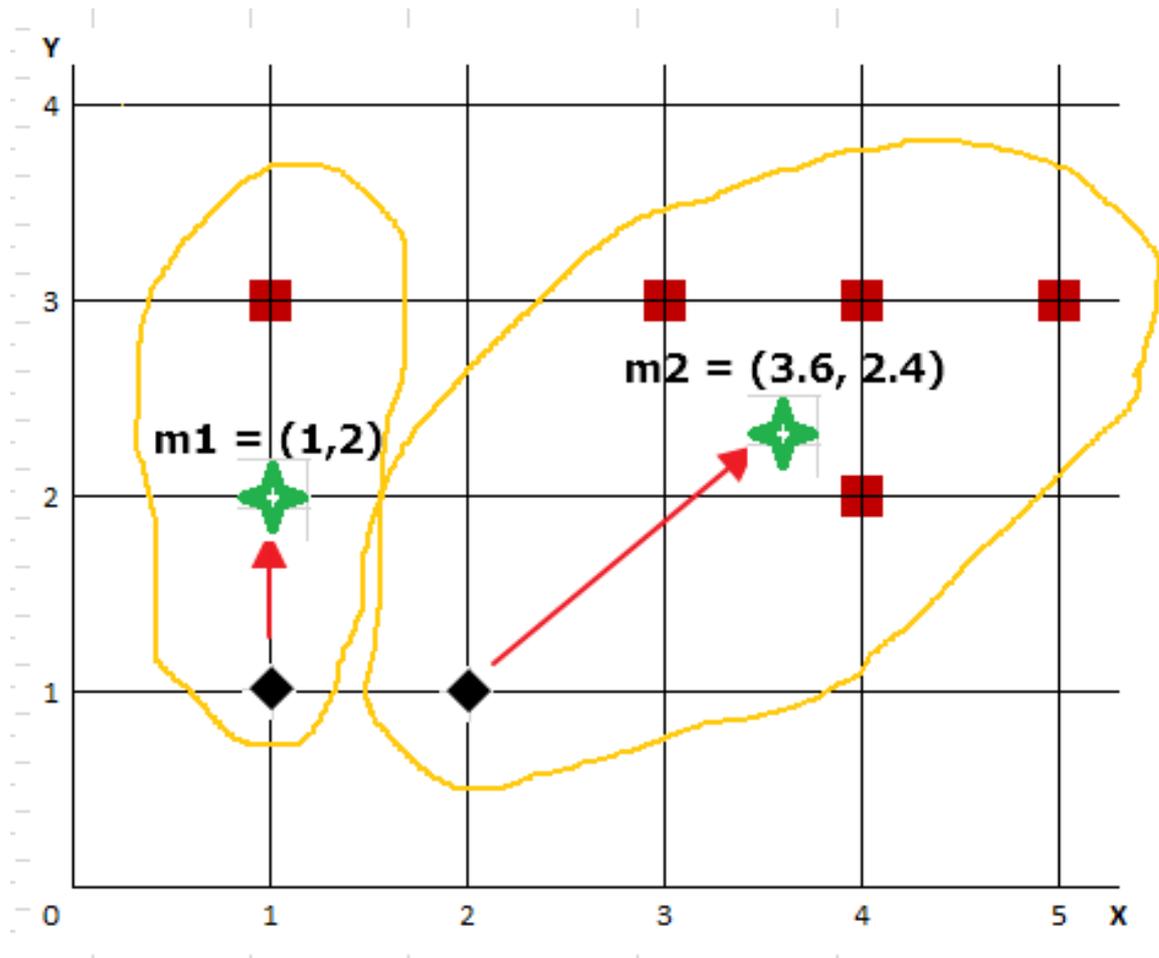


Рисунок 7 – Расположение кластеров и центроидов после 1-го прохода

Шаг 3.

Проход 2.

После того, как найдены новые центры кластеров, для каждой точки снова определяется ближайший к ней центр и ее отношение к соответствующему кластеру.

Для это еще раз вычисляются евклидовы расстояния между точками и центрами кластеров.

Результаты вычислений представлены на рисунке 8.

	A	B	C	D	E	F	G
1	#	X	Y	m1	m2	Кластер	
2	A	1	3	1,00	2,67	1	ЕСЛИ(D2<E2;1;2)
3	B	3	3	2,24	0,85	2	
4	C	4	3	3,16	0,72	2	
5	D	5	3	4,12	1,52	2	
6	E	1	2	0,00	2,63	1	
7	F	4	2	3,00	0,57	2	
8	G	1	1	1,00	2,95	1	
9	H	2	1	1,41	2,13	1	

Рисунок 8 – Таблица определения центров для точек данных (проход 2)

«Относительно большое изменение m2 привело к тому, что запись H оказалась ближе к центру m1, что автоматически сделало ее членом кластера 1. Все остальные записи остались в тех же кластерах, что и на предыдущем проходе алгоритма. Таким образом, кластер 1 будет A, E, G, H, а кластер 2 – B, C, D, F.

Новая сумма квадратов ошибок составит:

$$E = 7,88$$

Таким образом, имеет место уменьшение ошибки относительно начального состояния центров кластеров.

Это подтверждает улучшение качества кластеризации, т.е. более высокую «кучность» объектов относительно центра кластера

Шаг 4.

Проход 2.

Для каждого кластера вновь вычисляется его центроид, и центр кластера перемещается в него.

Новый центроид для 1-го кластера:  $[(1+1+1+2)/4, (3+2+1+1)/4] = (1.25, 1.75)$  » [11]..

Центроид для кластера 2:  $[(3 + 4 + 5 + 4)/4, (3 + 3 + 2 + 4)/4] = (4, 2.75)$ .

Расположение кластеров и центроидов после 2-го прохода алгоритма представлено на рисунке 9.

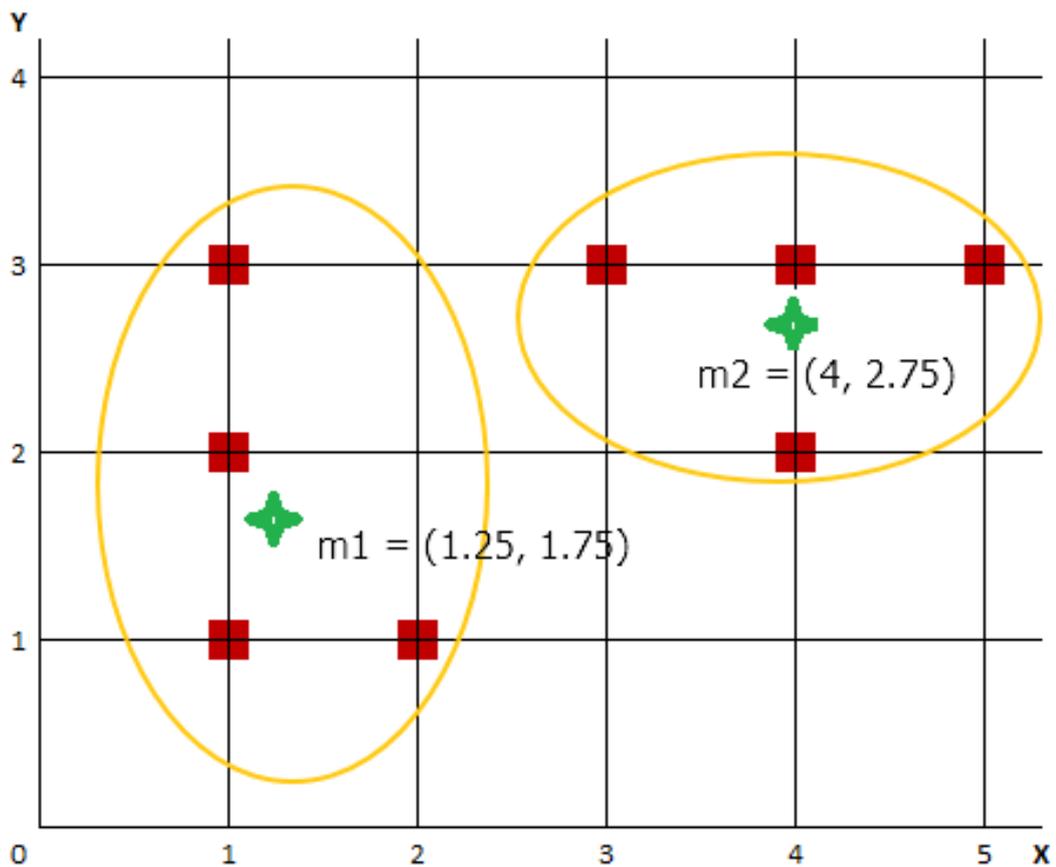


Рисунок 9 – Расположение кластеров и центроидов после 2-го прохода

По сравнению предыдущим проходом центры кластеров изменились незначительно.

Шаг 3.

Проход 3.

Для каждой записи вновь ищется ближайший к ней центр кластера. Полученные на данном проходе расстояния представлены на рисунке 10.

1	A	B	C	D	E	F	G
1	#	X	Y	m1	m2	Кластер	
2	A	1	3	1,27	3,01	1	ЕСЛИ(D2<E2;1;2)
3	B	3	3	2,15	1,03	2	
4	C	4	3	3,02	0,25	2	
5	D	5	3	3,95	1,03	2	
6	E	1	2	0,35	3,09	1	
7	F	4	2	2,76	0,75	2	
8	G	1	1	0,79	3,47	1	
9	H	2	1	1,06	2,66	1	

Рисунок 10 – Таблица определения центров для точек данных (проход 3)

«Следует отметить, что записей, сменивших кластер на данном проходе алгоритма, не было.

Новая сумма квадратов ошибок составит:

$$E = 6,25$$

Таким образом, изменение суммы квадратов ошибок является незначительным по сравнению с предыдущим проходом.

Шаг 4.

Проход 3.

Для каждого кластера вновь вычисляется его центроид, и центр кластера перемещается в него.

Ввиду того, что на данном проходе ни одна запись не изменила своего членства в кластерах, то положение центроида не поменялось, и алгоритм завершает свою работу» [11].

Таким образом, полученное решение позволяет решить задачу оптимизации планирования доставки грузов, путем разбиения региона доставки на ближние зоны-кластеры.

## Выводы к главе 2

Во второй главе рассматривается решение задачи оптимизации планирования доставки грузов с помощью алгоритма k-means.

Результаты проделанной работы позволили сделать выводы, представленные ниже.

Рассмотрено решение задачи оптимизации планирования доставки грузов на примере разбиения набора из восьми точек данных в двумерном пространстве-региона, из которого требуется получить два кластера-зоны компактной доставки грузов.

Для решения использован алгоритм k-means.

Как показали результаты, алгоритм k-means позволяет решить задачу оптимизации планирования доставки грузов путем разбиения региона доставки на ближние зоны-кластеры.

Таким образом, подход к оптимизации планирования доставки грузов на основе алгоритма k-means может быть рекомендован для решения задач формирования оптимальных маршрутов.

### Глава 3 Разработка программы оптимизации планирования доставки грузов на основе алгоритма k-means

Рассмотрим существующие программы для управления планированием доставки грузов на предмет использования в качестве платформы для разработки.

#### 3.1 Программа «1С: TMS Логистика. Управление перевозками»

ИТ-решение «1С:TMS Логистика. Управление перевозками» – это программный продукт для решения задач транспортной логистики и создания цепочек перевозок, которые могут состоять из звеньев, обслуживаемых различными видами транспорта.

С возможностью автоматического планирования доставок по городу и области с привязкой к картографии. Система управления заказами. TMS – имеет модульную структуру, все модули функционируют в едином информационном пространстве (единая СУБД). Программный продукт реализован на платформе 1С: Предприятие 8.

Схема работы 1С:TMS представлена на рисунке 11.



Рисунок 11– Схема работы программы 1С: TMS

Для планирования маршрутов используется модуль «Автоматическое планирование».

«Модуль предназначен для планирования работы водителей транспортных средств по заданным параметрам. Планирование осуществляется как в ручном, так и в автоматическом режиме. Планирование заявок осуществляется в обработке «Рабочее место логиста». В диалоге отображаются заявки и маршруты, как в виде списков, так и на карте. Заявки могут быть сгруппированы по произвольным группировкам (например по географическим зонам, дате доставки, и любым другим связанным полям).

При планировании автоматически рассчитываются плановые показатели маршрута: пробег (проезд с учетом ПДД), время маршрута, время посещения каждого из адресов» [1].

Как отмечено в описании программы, алгоритм автопланирования разработан специалистами компании-разработчика ИТ-решения.

Однако, в материалах к программе данный алгоритм не описан.

### **3.2 Программа «Простые маршруты»**

Программа «Простые маршруты» предназначена для автоматизированной разработки оптимальных маршрутов с учетом временных окон доставки на основе полученных данных из Вашей учетной системы [12].

В программе используется онлайн-сервис Google Maps [10].

Программа имеет простой интерфейс, позволяющий без проблем, для пользователя любого уровня, обработать строку с адресом доставки, получив координаты и местоположение пункта доставки на карте и сохранить эту информацию в пользовательской системе для будущего использования

Таким образом, исключается возможность доставки по «ошибочному» адресу.

Эта программа решает следующие задачи:

- быстрый и простой способ получения набора данных с адресами из любых документов базы данных с любой конфигурацией;
- наглядное представление точек доставки и складов на карте, не выходя из 1С;
- разные интерактивные методы работы с точками на карте для формирования маршрутов вручную;
- автоматический расчёт оптимальных маршрутов доставки с учётом ряда заданных параметров;
- определение длины маршрута и времени его выполнения;
- автоматическая сортировка следования документов в маршрутном листе, согласно обхода точек маршрута.

Программа имеет возможность «ручного» составления маршрутов, а также может менять состав и порядок обхода для уже сформированных автоматически маршрутов (рисунок 12).

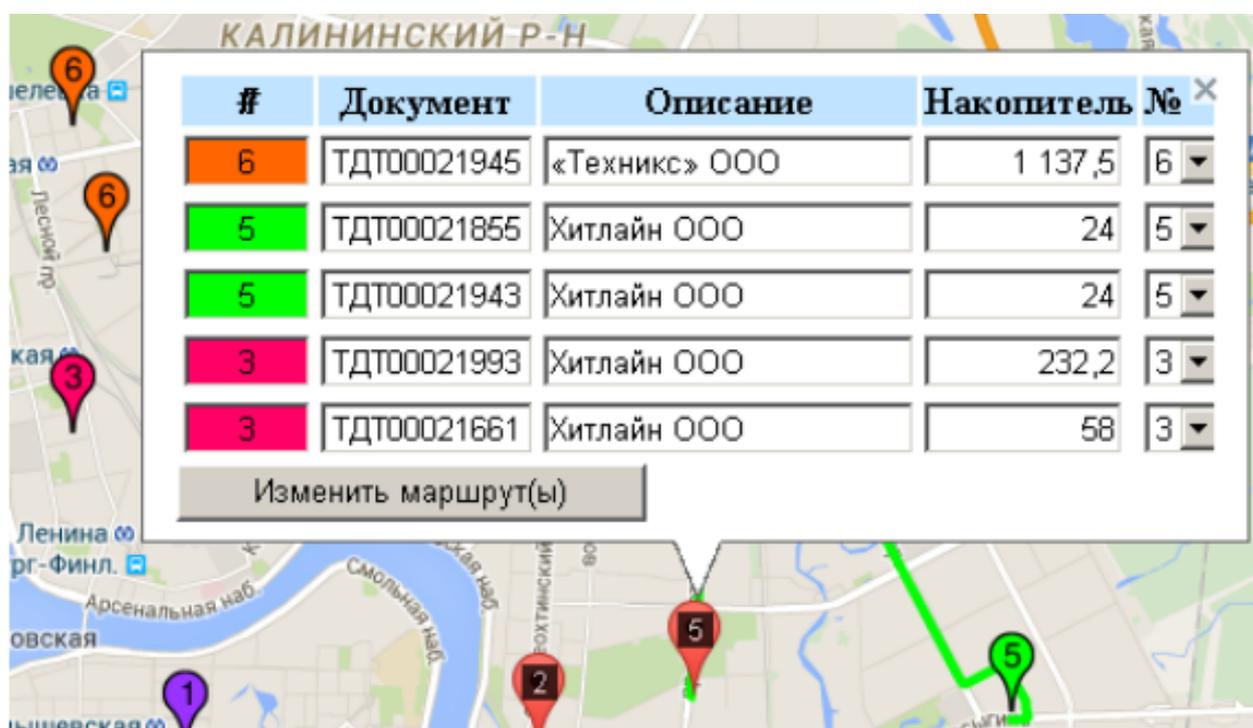


Рисунок 12 - Список маршрутов

В программе реализованы математические методы обработки данных, для получения маршрутов исходя из двух вариантов:

Вариант 1. Формирование количества маршрутов для заданного количества /транспорта.

Вариант 2. Определение оптимального количества маршрутов а/транспорта для заданной максимальной грузоподъемности автомашины.

Для решения данных задач в программе используются алгоритмы К-средних (k-means) и Кларка-Райта [8].

### 3.3 Онлайн-сервис «Калькулятор для кластеризации по алгоритму К-средних»

Данный сервис содержит ряд онлайн-калькуляторов по экономическим направлениям. Результат работы каждого калькулятора - это отчет в формате Word и Excel, в котором приведены все расчетные формулы и выводы [9].

С помощью онлайн-калькулятора можно проводить кластеризацию объектов методом К-средних (рисунок 13).

Методы принятия оптимальных решений▼		Статистика онлайн		ЗАКАЗАТЬ ▼	Поиск
Метод k-средних	Показатели динамики	Индекс сезонности	Аддитивная модель ряда	Мультипликативная модель	
Цепные показатели	Общий индекс цен	Группировка данных		Аналитическое выравнивание	

#### Алгоритм К-средних

**Метод К-средних** – это метод кластерного анализа, целью которого является разделение  $m$  наблюдений на  $k$  кластеров, при этом каждое наблюдение относится к тому кластеру, к центру (центроиду) которого оно ближе всего.

Шаг №1	Шаг №2	Видеоинструкция	Оформление Word
--------	--------	-----------------	-----------------

**ИНСТРУКЦИЯ.** Укажите количество данных, нажмите **Далее**. Полученное решение сохраняется в файле **Word**.

Размерность матрицы разбиения  x

**Далее** **Из Excel**

Рисунок 13 – Окно решения задач кластеризации методом k-средних онлайн-калькулятора

Результат тестового решения задачи, описанной в главе 2, совпал с результатом, полученным при решении в Excel (таблица 2).

Таблица 2 - Состояние кластеров после завершения процесса кластеризации

e <sub>1</sub>	e <sub>2</sub>
AEGH	BCDF

По умолчанию все вычисления оформляются в файле формата MS Word.

Однако следует учесть, что такая возможность имеется только для браузеров, поддерживающих Javascript.

Если поддержка Javascript отключена, необходимо включить ее в настройках браузера.

В остальных случаях всегда доступен просмотр решения непосредственно в браузере.

К достоинствам данного онлайн-сервиса следует отнести то, что помимо калькулятора пользователю по каждой дисциплине для ознакомления предлагается соответствующий теоретический материал и примеры решения задач.

Кроме того, онлайн-калькулятор имеет простой в использовании и многоязычный интерфейс.

Недостатками решения являются отсутствие средств для построения и визуализации маршрутов и некоторая избыточность информации в рабочих окнах, что усложняет поиск конкретной дисциплины или определенного метода решения задачи.

### **3.4 Выбор платформы и разработка программы**

Сравним характеристики рассмотренных программ на предмет возможности использования в качестве платформы для разработки программы

оптимизации планирования доставки грузов на основе алгоритма k-means.

Результаты сравнительного анализа сведем в таблицу 4.

Критерии оценивания:

0 – полное несоответствие требованиям;

1 – значительное несоответствие требованиям;

2 – незначительное несоответствие требованиям;

3 – полное соответствие требованиям.

Таблица 4 – Сравнительный анализ программ для оптимизации планирования доставки грузов

Характеристика/балл (макс. -3)	1С:TMS	Простые маршруты	Онлайн- калькулятор
возможность использования алгоритма k-means	Нет данных	3	3
простота адаптации	2	3	0
уровень автоматизации решения	3	2	1
визуализация маршрутов	3	3	0
Итого	8	11	4

На основании результатов анализа в качестве платформы для разработки выбираем программу «1С: Простые маршруты».

Реализован алгоритм k-means в рамках конфигурации данной программы.

Ниже приведен листинг функции, реализующей алгоритм k-means на языке 1С8 [14].

Листинг 1 - Функция, реализующая алгоритм k-means

```
//Функция поиска расстояния между двумя точками
```

```
&НаСервере
```

```
Функция ПолучитьРасстояниеМеждуТочками(Точка1, Точка2)
```

```
Если Точка1.Широта = Точка2.Широта И Точка1.Долгота =
```

```
Точка2.Долгота Тогда
```

```
Возврат 0
```

```

Иначе
«Возврат  A Cos(Sin(Точка1.Широта) * Sin(Точка2.Широта) +
Cos(Точка1.Широта) * Cos(Точка2.Широта) * Cos(Точка1.Долгота -
Точка2.Долгота));
КонецЕсли;
КонецФункции
&НаСервере
Процедура РассчитатьСервер()
Если Объект.РаспределениеЗаказов.Количество() <
Объект.КоличествоКластеров Тогда
Сообщить("Расчёт невозможен. Количество заказов меньше количества
кластеров");
Возврат;
КонецЕсли;
ТаблицаРасстоянийДоЦентроидов = Новый ТаблицаЗначений;
ТаблицаРасстоянийДоЦентроидов.Колонки.Добавить("Кластер");
ТаблицаРасстоянийДоЦентроидов.Колонки.Добавить("Заказ");
ТаблицаРасстоянийДоЦентроидов.Колонки.Добавить("Расстояние");
ТаблицаЦентроидов = Новый ТаблицаЗначений;
ТаблицаЦентроидов.Колонки.Добавить("Кластер"); »[5]
ТаблицаЦентроидов.Колонки.Добавить("Широта");
ТаблицаЦентроидов.Колонки.Добавить("Долгота");
ТаблицаЦентроидов.Колонки.Добавить("ШиротаБыло");
ТаблицаЦентроидов.Колонки.Добавить("ДолготаБыло");
Для Сч = 1 По Объект.КоличествоКластеров Цикл
НоваяСтрока = ТаблицаЦентроидов.Добавить();
НоваяСтрока.Кластер = Сч;
НоваяСтрока.Широта = Объект.РаспределениеЗаказов[Сч-1].Широта;
НоваяСтрока.Долгота = Объект.РаспределениеЗаказов[Сч-1].Долгота;
НоваяСтрока.ШиротаБыло = 0;

```

НоваяСтрока.ДолготаБыло = 0;  
 КонецЦикла;  
 «ПродолжаемКластеризацию = Истина;  
 КоличествоИтераций = 0;  
 Пока ПродолжаемКластеризацию И КоличествоИтераций < 50 Цикл  
 ТаблицаРасстоянийДоЦентроидов.Очистить();  
 Для Каждого ТекЦентроид Из ТаблицаЦентроидов Цикл  
 Для Каждого ТекЗаказ Из Объект.РаспределениеЗаказов Цикл  
 НоваяСтрока = ТаблицаРасстоянийДоЦентроидов.Добавить();  
 НоваяСтрока.Кластер = ТекЦентроид.Кластер;  
 НоваяСтрока.Заказ = ТекЗаказ.Заказ;  
 НоваяСтрока.Расстояние =  
 ПолучитьРасстояниеМеждуТочками(ТекЦентроид,ТекЗаказ);» [5]  
 КонецЦикла;  
 КонецЦикла;  
 Для Каждого ТекЗаказ Из Объект.РаспределениеЗаказов Цикл  
 СтрокиРасстояний =  
 ТаблицаРасстоянийДоЦентроидов.НайтиСтроки(Новый  
 Структура("Заказ",ТекЗаказ.Заказ));  
 МинимальноеРасстояние = Неопределено;  
 КластерМинимальноеРасстояние = Неопределено;  
 Для Каждого ТекСтрока Из СтрокиРасстояний Цикл  
 Если МинимальноеРасстояние = Неопределено Тогда  
 МинимальноеРасстояние = ТекСтрока.Расстояние;  
 КластерМинимальноеРасстояние = ТекСтрока.Кластер;  
 КонецЕсли;  
 Если МинимальноеРасстояние > ТекСтрока.Расстояние Тогда  
 МинимальноеРасстояние = ТекСтрока.Расстояние;  
 КластерМинимальноеРасстояние = ТекСтрока.Кластер;  
 КонецЕсли;

КонецЦикла;  
 «ТекЗаказ.Кластер = КластерМинимальноеРасстояние;  
 КонецЦикла;  
 ПродолжаемКластеризацию = Ложь;  
 Для Каждого ТекЦентроид Из ТаблицаЦентроидов Цикл  
 СтрокиЗаказовКластера =  
 Объект.РаспределениеЗаказов.НайтиСтроки(Новый  
 Структура("Кластер",ТекЦентроид.Кластер));  
 СуммаШирота = 0;  
 СуммаДолгота = 0;  
 Для Каждого ТекСтрока Из СтрокиЗаказовКластера Цикл  
 СуммаШирота = СуммаШирота + ТекСтрока.Широта;  
 СуммаДолгота = СуммаДолгота + ТекСтрока.Долгота;  
 КонецЦикла;  
 ТекЦентроид.ШиротаБыло = ТекЦентроид.Широта;  
 ТекЦентроид.ДолготаБыло = ТекЦентроид.Долгота;  
 Если СтрокиЗаказовКластера.Количество() = 0 Тогда  
 ТекЦентроид.Широта = СуммаШирота;  
 ТекЦентроид.Долгота = СуммаДолгота;  
 Иначе  
 ТекЦентроид.Широта =  
 СуммаШирота/СтрокиЗаказовКластера.Количество();  
 ТекЦентроид.Долгота =  
 СуммаДолгота/СтрокиЗаказовКластера.Количество();  
 КонецЕсли; » [5]  
 Если ТекЦентроид.Широта <> ТекЦентроид.ШиротаБыло ИЛИ  
 ТекЦентроид.Долгота <> ТекЦентроид.ДолготаБыло Тогда  
 ПродолжаемКластеризацию = Истина;  
 КонецЕсли;  
 КонецЦикла;

КоличествоИтераций = КоличествоИтераций + 1;

КонецЦикла;

Конец листинга 1

Для проверки программы используем метод функционального тестирования.

Функциональное тестирование - это тестирование программного обеспечения в целях проверки реализуемости функциональных требований, то есть способности программного обеспечения в определённых условиях решать задачи, нужные пользователям [4].

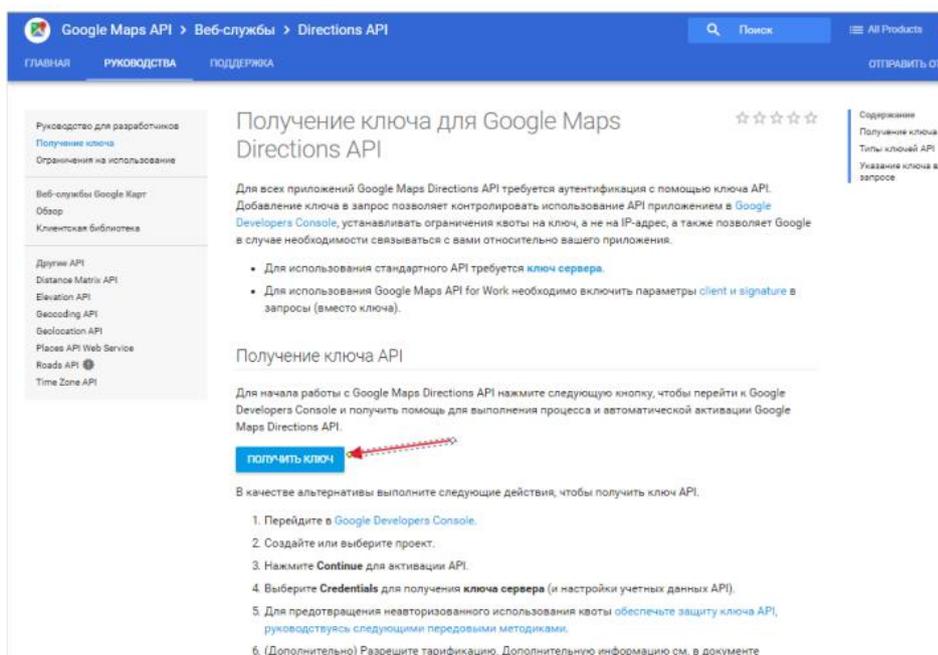
Произведем функциональное тестирование разработанной программы, используя тестовую базу данных, поставляемую вместе с демоверсией программы.

Для подключения к сервису Google Maps необходимо предварительно получить ключ в соответствии с инструкцией, показанной на рисунке 14.

1. Переходим по адресу:

<https://developers.google.com/maps/documentation/directions/get-api-key?hl=ru>

2. Появится такое окно. Нажимаем кнопку «ПОЛУЧИТЬ КЛЮЧ»



3. Далее переходим в такое окно. Принимаем условия использования. Нажимаем кнопку «Принять и продолжить»

Рисунок 14 – Инструкция для подключения к сервису Google Maps

Интерфейс программы состоит из двух частей (рисунок 15).

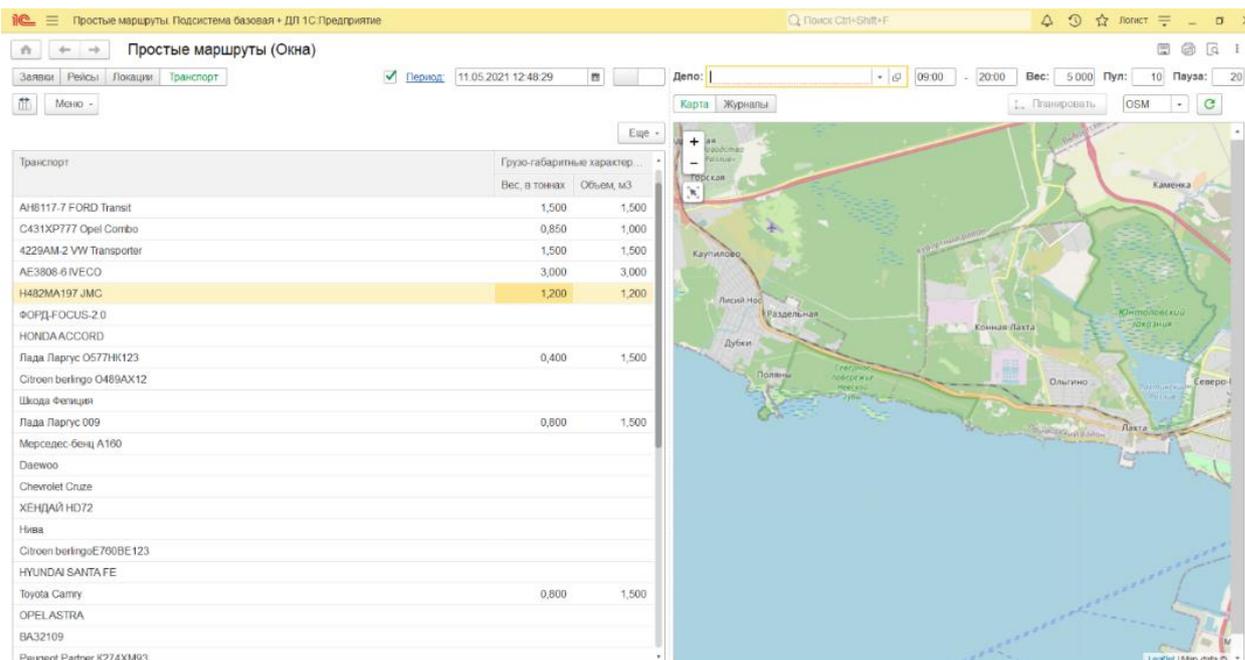


Рисунок 15- Интерфейс программы

В левой части выводится список задействованного транспорта. В правой части – карта и журнал маршрутов.

По каждому транспортному средству создается карточка (рисунок 16).

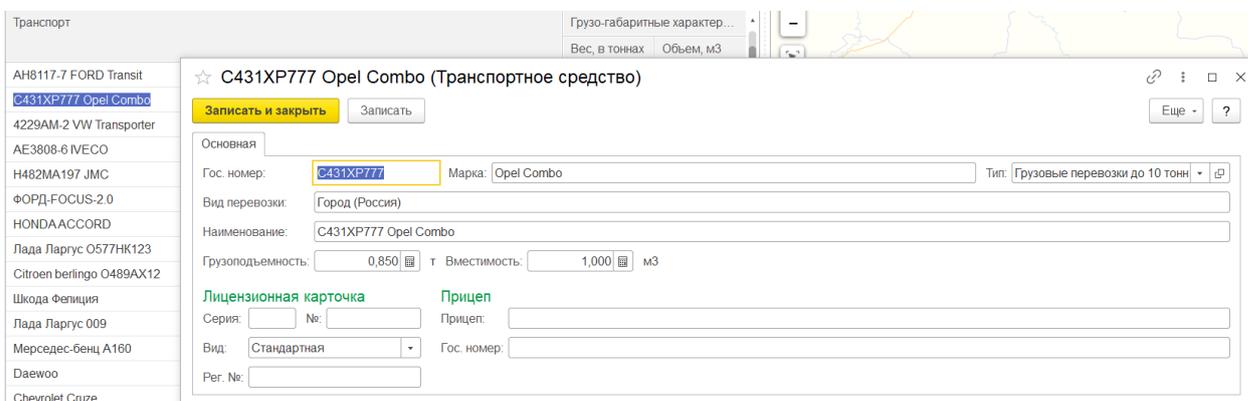


Рисунок 16 – Карточка транспортного средства

Журнал маршрутов (рисунок 17) содержит документы с адресами клиентов. Для отображения на карте в качестве точки доставки, может быть использован любой документ базы данных с реквизитом «АдресДоставки».

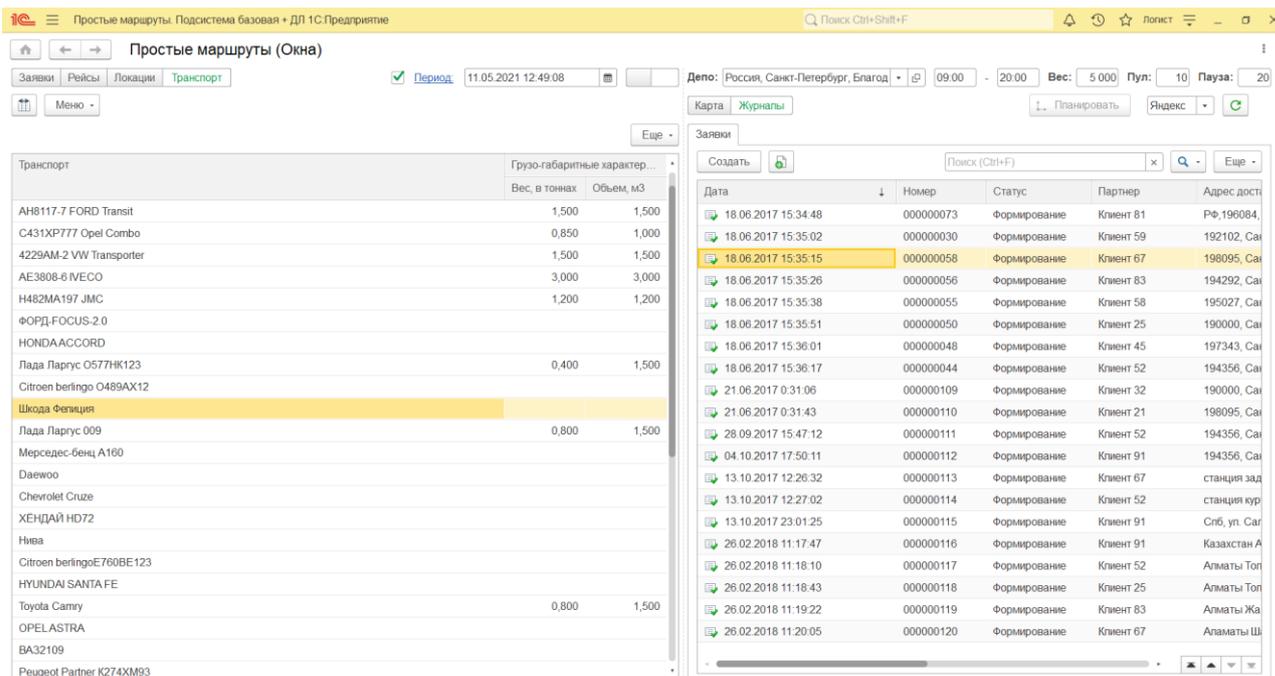


Рисунок 17 - Журнал маршрутов

Базовым документом в программе является заявка клиента на доставку груза по заданному адресу (рисунок 18).

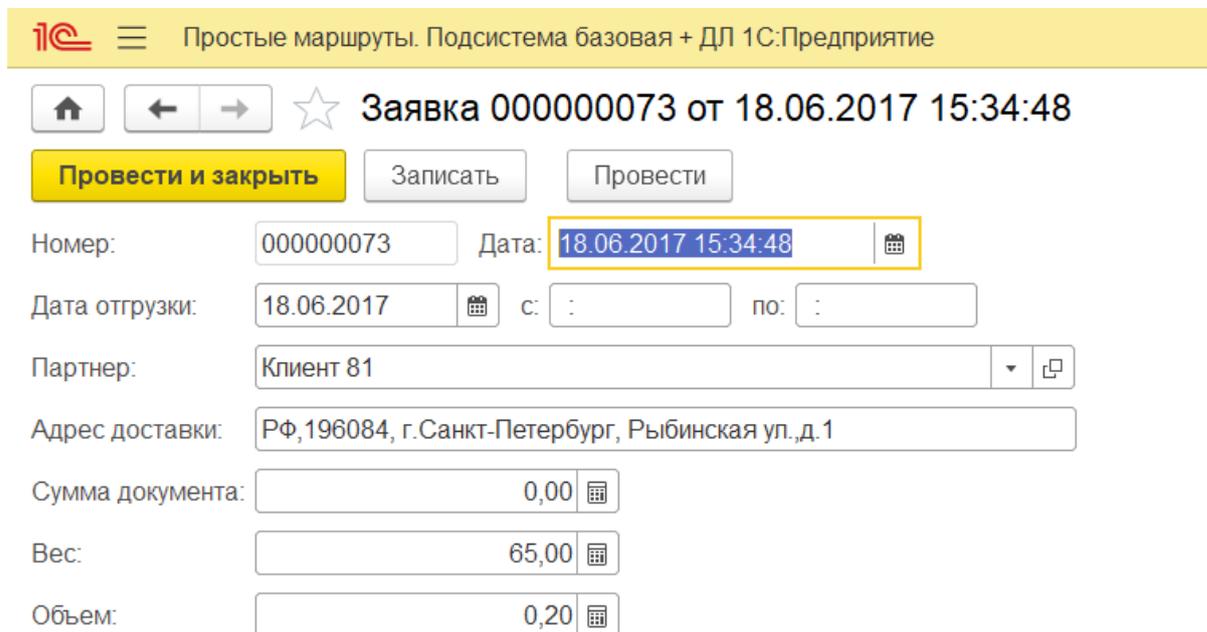


Рисунок 18 - Экран ввода заявки

После ввода заявки с помощью алгоритма k-means программа автоматически формирует оптимальный маршрут (рисунок 19).

☆ пр. Большевиков, 11к2, Санкт-Петербург, Россия, 193168 (Пункты назначения) 🔗 📄 ✕

Записать и закрыть

**Пункт назначения**

Наименование:  Код:

Тип пункта:  Зона доставки:

Адрес пункта:

**Координаты адреса пункта**

Геокодирован Широта:  Долгота:

**Карточка получателя**

Получатель:

Адрес доставки:

Вид пункта:

Контактное лицо:

Телефон основной:

Телефон (запасной):

Электронная почта:

**Временные окна для доставки:**

Понедельник:  -

Вторник:  -

Среда:  -

Четверг:  -

Пятница:  -

Суббота:  -

Воскресенье:  -

Праздники:  -

Штраф за опоздание:

**Временные задержки (в минутах)**

Время разгрузки:

Дополнительное время, затрачиваемое на подъезд к клиенту:

**Описание работы**

Рисунок 19 – Окно формирования маршрута

Для формирования количества маршрутов (вариант 1) необходимо:

- указать депо (склад отгрузки);
- указать количество машин;
- выполнить команду «Расчитать» - «Метод k-means (к-средних).

На рисунке 20 представлены маршруты, сформированные для Депо-склад 1 и двух машин по тестовой базе данных.

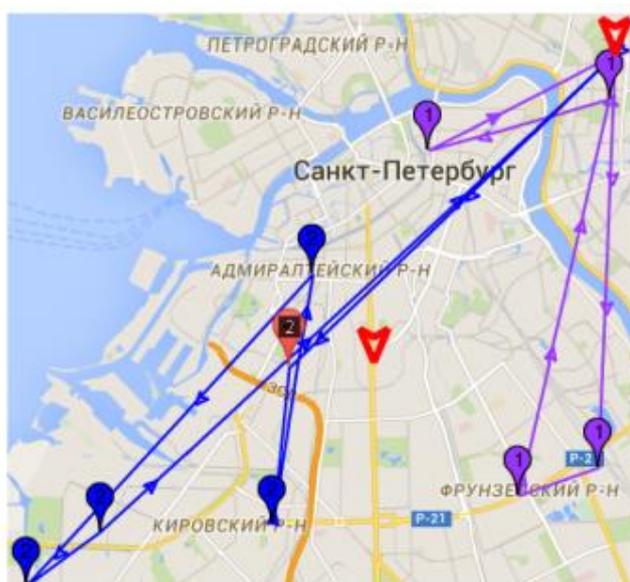


Рисунок 20 – Пример формирования количества маршрутов

Таким образом, функциональное тестирование подтвердило возможность использования программы для формирования оптимальных маршрутов доставки грузов на основе алгоритма k-means.

### **Выводы к главе 3**

Третья глава посвящена разработке программы оптимизации планирования доставки грузов на основе алгоритма k-means.

Результаты проделанной работы позволили сделать выводы, представленные ниже.

На основании результатов анализа в качестве платформы для разработки выбрана программа «1С: Простые маршруты».

Для подключения к сервису Google Maps необходимо предварительно получить ключ доступа.

Функциональное тестирование подтвердило возможность использования программы для формирования оптимальных маршрутов доставки грузов на основе алгоритма k-means.

## Заключение

Выпускная квалификационная работа посвящена актуальной проблеме оптимизация планирования доставки грузов на основе алгоритма k-means.

С оперативной точки зрения задача оптимизации планирования доставки груза состоит в обеспечении эффективности процесса доставки за счет экономии топлива и сокращения времени простоя. Как показывает практика, для решения данной задачи необходимо использовать методы интеллектуального анализа данных – Data mining.

Одним из таких методов является метод кластеризации данных на основе алгоритма k-means.

Для достижения данной цели в процессе работы над бакалаврской работой решены следующие задачи:

- выполнена постановка задачи оптимизации планирования доставки грузов. Задача оптимизации планирования доставки грузов рассматривается как задача кластеризации. Кластеризацию можно определить, как задачу идентификации групп в данных, так чтобы точки данных в одном кластере были очень похожи и при этом сильно отличались от точек данных в других кластерах. Одним из широко применяемых на практике алгоритмов кластеризации является алгоритм k-means. Главным преимуществом данного алгоритма является его высокая производительность и простота реализации;
- рассмотрено решение задачи оптимизации планирования доставки грузов на представленном в работе примере. Для решения задачи использован алгоритм k-means и табличный процессор Excel. Как показали результаты, алгоритм k-means позволяет решить задачу оптимизации планирования доставки грузов путем разбиения региона доставки на ближние зоны-кластеры. Таким образом, подход к оптимизации планирования доставки грузов на основе алгоритма k-means может быть рекомендован для решения задач формирования оптимальных маршрутов;

– разработана программа формирования оптимальных маршрутов доставки грузов на основе алгоритма k-means. Проанализированы функциональные и архитектурные особенности программ «1С:TMS», «1С: Простые маршруты» и онлайн-сервиса «Калькулятор для кластеризации по алгоритму K-средних». В результате сравнительного анализа в качестве платформы для разработки использовано типовое ИТ-решение «1С: Простые маршруты». Функциональное тестирование подтвердило возможность использования программы для формирования оптимальных маршрутов доставки грузов на основе алгоритма k-means и их визуализации.

Результаты бакалаврской работы представляют научно-практический интерес и могут быть рекомендованы для разработчиков программ поддержки задач формирования оптимальных маршрутов на основе методов интеллектуального анализа данных.

## Список используемой литературы и используемых источников

1. 1С:TMS Логистика. Управление перевозками [Электронный ресурс]. URL: <https://itob.ru/products/1c-tms/> (дата обращения: 07.05.2021).
2. Воронцов К.В. Математические методы обучения по прецедентам (теория обучения машин) [Электронный ресурс]. URL: <http://www.machinelearning.ru/wiki/images/6/6d/Voron-ML-1.pdf> (дата обращения: 07.05.2021).
3. Евклидова метрика [Электронный ресурс]. URL: [https://ru.wikipedia.org/wiki/%D0%95%D0%B2%D0%BA%D0%BB%D0%B8%D0%B4%D0%BE%D0%B2%D0%B0\\_%D0%BC%D0%B5%D1%82%D1%80%D0%B8%D0%BA%D0%B0](https://ru.wikipedia.org/wiki/%D0%95%D0%B2%D0%BA%D0%BB%D0%B8%D0%B4%D0%BE%D0%B2%D0%B0_%D0%BC%D0%B5%D1%82%D1%80%D0%B8%D0%BA%D0%B0) (дата обращения: 07.05.2021).
4. Карпович Е. Е. Методы тестирования и отладки программного обеспечения : учебник. Москва : Издательский Дом МИСиС, 2020. 136 с. [Электронный ресурс]. URL: <https://www.iprbookshop.ru/106722.html> (дата обращения: 18.05.2021).
5. Кластеризация заказов по географическим координатам [Электронный ресурс]. URL: <http://danila.org.ua/category/%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%B0/1%D1%81/page/3/> (дата обращения: 07.05.2021).
6. Котов К., Красильников Н. Кластеризация данных [Электронный ресурс]. URL: <https://logic.pdmi.ras.ru/~yura/internet/02ia-seminar-note.pdf> (дата обращения: 07.05.2021).
7. Метод k-средних. [Электронный ресурс]. URL: [https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4\\_k-%D1%81%D1%80%D0%B5%D0%B4%D0%BD%D0%B8%D1%85](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4_k-%D1%81%D1%80%D0%B5%D0%B4%D0%BD%D0%B8%D1%85) (дата обращения: 07.05.2021).
8. Метод Кларка-Райта. Оптимальное планирование маршрутов грузоперевозок [Электронный ресурс]. URL: <https://infostart.ru/1c/articles/443585/> (дата обращения: 07.05.2021).
9. Онлайн-сервис «Калькулятор кластерного анализа» [Электронный

ресурс]. URL: <https://axd.semestr.ru/upr/average.php> (дата обращения 07.05.2021).

10. Онлайн-сервис Google Maps [Электронный ресурс]. URL: <https://www.google.ru/maps> (дата обращения 07.05.2021).

11. Оптимизация планирования доставки грузов. Алгоритм кластеризации k-means (метод K-средних) [Электронный ресурс]. URL: <https://infostart.ru/1c/articles/444787/> (дата обращения: 07.05.2021).

12. Программа «Простые маршруты» [Электронный ресурс]. URL: <https://infostart.ru/public/635798/> (дата обращения: 07.05.2021).

13. Смирнов А.В., Андрианов И.А., Суконщиков А.А., Бахтенко Е.А. Математическая модель оптимизации доставки товаров автотранспортом на разветвленной сети дорог для решения задачи кластеризации // Тенденции науки и образования. Сборник научных трудов по материалам XXII международной научной конференции, Самара, 2017. С 35-37.

14. Фирма 1С [Электронный ресурс]. URL: <https://1c.ru/> (дата обращения 07.05.2021).

15. Форман Д. Много цифр: Анализ больших данных при помощи Excel [Электронный ресурс]. URL: <https://baguzin.ru/wp/dzhon-forman-mnogo-tsifr-analiz-bolshih/#%D0%93%D0%BB2> (дата обращения: 07.05.2021).

16. Alshaer M. An Efficient Framework for Processing and Analyzing Unstructured Text to Discover Delivery Delay and Optimization of Route Planning in Realtime. Data Structures and Algorithms. Université de Lyon; École Doctorale des Sciences et de Technologie (Beyrouth), 2019.

17. An Introduction to Clustering and different methods of clustering [Электронный ресурс]. URL: <https://www.analyticsvidhya.com/blog/2016/11/an-introduction-to-clustering-and-different-methods-of-clustering/> (дата обращения: 07.05.2021).

18. An optimized version of the K-Means clustering algorithm [Электронный ресурс]. URL: <https://ieeexplore.ieee.org/document/6933081> (дата обращения: 07.05.2021).

19. Eight Clustering Algorithms in Machine Learning that All Data Scientists Should Know [Электронный ресурс]. URL: <https://www.freecodecamp.org/news/8-clustering-algorithms-in-machine-learning-that-all-data-scientists-should-know/> (дата обращения: 07.05.2021).

20. K-means Clustering: Algorithm, Applications, Evaluation Methods, and Drawbacks [Электронный ресурс]. URL: <https://towardsdatascience.com/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-aa03e644b48a> (дата обращения: 07.05.2021).