ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчёт по лабораторной работе № 4

«Кластеризация массива посредством полного перебора всех комбинаций значений этого массива»

Выполнил работу

Шайкин Андрей

Академическая группа №J3114

Принято

Практик, Максим Дунаев

Санкт-Петербург

2024

**Структура отчёта:**

1. Введение

Цель работы: написать алгоритм разбития массива на 9 кластеров,

больше O(2^N)

Задачи:

1. Понять смысл метрики
2. Написать алгоритм
3. Измерить время работы алгоритма
4. Теоретическая подготовка

Использовал рекурсию, лимиты, векторы(1,2,3 мерные) , дабл и т.д.

1. Реализация

Увидев сложность O(2^N) сразу решил использовать рекурсию, но столкнулся с проблемой заданной метрики, т.к. мы пытаемся минимизировать расстояние, то по сути сумма метрик кластеров должна быть минимальной, но одной рекурсии для этого не хватит, поэтому решил добавить алгоритм нахождения оптимального решения, который реализовал минимизацию расстояния. Далее я столкнулся с проблемой, что если массив не сортировать перед входом в рекурсию, то он будет неправильно разделять на кластеры, поэтому добавил этот шаг

1. Экспериментальная часть

Подсчёт по памяти (только для циклов и сложных структур) :

- vector<double>& arr - 64\*N бит

- vector<vector<vector<double>>>& result – 2^N \* 64\*N бит(все возможные разбиения)

- vector<vector<double>>& current – 64\*N бит ( примерно)

Подсчёт асимптотики (только для циклов и сложных структур):

- calculateMean() – O(M) – M – размер кластера

- calculateMetric() – O(2\*M) – M – размер кластера

- partition() – O(P\*N) – P – количество разбиений, N – размер массива

- findOptimalClusters() – O(P\*K\*M + P\*N)

График зависимости времени от числа элементов.

Согласно требованиям моего варианта, на вход к моему алгоритму подаётся до 25 элементов. Теоретически заданная сложность задачи составляет O(2^N) и более. Для тестирования алгоритма была собрана статистика, приведенная в таблице №\*.

Таблица №\* - Подсчёт сложности реализованного алгоритма

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Размер входного набора | 10 | 15 | 20 | 25 |
| O(N(2^(N-1) + log(N))) | 0.000219 | 0.045816 | 1.11063 | 10.4138 |

1. Заключение

В ходе выполнения работы мною был реализован рекурсивный алгоритм для полного перебора возможных кластеров. Для улучшения можно использовать другие методы разбивки на k кластеров (например, K-средних или иерархической кластеризации).

1. Приложения

В этом разделе вам необходимо указать всю дополнительную информацию, которая не вошла в основной текст, например полный исходный код программы. Пример:

ПРИЛОЖЕНИЕ А

Листинг кода файла lab-4.cpp

  