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

Вяхирев Иван

Академическая группа №J3112

Принято

Практик, Максим Дунаев

Санкт-Петербург

2024

**Структура отчёта:**

1. Введение

Цель: написать алгоритм разбивки на 8 кластеров, строго за O(2^N) или больше.

Задачи:

1. Написать алгоритм
2. Записать измерения времени работы
3. Теоретическая подготовка

Для решения за >O(2^N) я решил использовать рекурсию. Она позволяет просто и без нагромождения кода выполнять полный перебор значений. Также я использовал типы данных вектора, дабл для плавающих значений итд.

1. Реализация

Для начала я написал тестирующую систему, дописав функцию проверки равенства классов и первоначальные тесты. После этого я начал писать алгоритм рекурсии. Он перебирает абсолютно все возможные кластеры и сохраняет лучший (по подсчитанной метрике среднего отклонения) кластер в отдельную переменную.

1. Экспериментальная часть

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

* vector<double>& arr – 64n бит (double – 64bit, n – количество точек)
* vector<vector<double>>& clusters и vector<vector<double>>& top\_clusters – 2^k \* 64n каждый – все возможные разбиения \* размер одного кластер вектора

Подсчёт асимптотики (только для циклов и сложных структур):

* metric\_func – O(N) линейный проход с подсчетом метрики, зависит от размера кластера
* get\_clusters – O(8^N)

График зависимости времени от числа элементов. Пример выполнения:

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 | 8 | 10 | 12 |
| O(k^N), с | 0,01211 | 0,3034 | 10,2212 |

1. Заключение

В ходе выполнения работы был реализован рекурсивный алгоритм для полного перебора возможных кластеров. Для дальнейшей оптимизации и улучшения можно исследовать другие методы разбивки на k кластеров (к примеру KMeans).

1. Приложения

ПРИЛОЖЕНИЕ А

Листинг кода файла test.cpp Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание