****

**Московский авиационный институт**

**(Национальный исследовательский университет)**

**Институт № 3**

**Кафедра 311**

**Интеллектуальные системы**

**Лабораторная работа № 6**

**«Кластерный анализ»**

**Выполнил студент  
Веденеев Максим Кириллович**

**Группа М3З-501-БК**

**Дата 06.10.2023 г.**

**Принял преподаватель  
Кос Оксана Игоревна**

Оглавление

[Цель лабораторной работы 3](#__RefHeading___1)

[Глава 1. Кластеризация методом k-средних 4](#__RefHeading___2)

[Глава 2. Реализация 6](#__RefHeading___3)

[Итоги лабораторной работы 8](#__RefHeading___4)

[Список литературы 9](#__RefHeading___5)

# Цель лабораторной работы

Изучить:

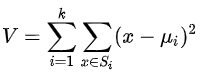
1. Кластерный анализ.
2. Реализовать его.

# Глава 1. Кластеризация методом k-средних

Кластерный анализ - многомерная статистическая процедура, выполняющая сбор данных, содержащих информацию о выборке объектов, и затем упорядочивающая объекты в сравнительно однородные группы. Задача кластеризации относится к статистической обработке, а также к широкому классу задач обучения без учителя.

Метод k-средних— наиболее популярный метод кластеризации.

Действие алгоритма таково, что он стремится минимизировать суммарное квадратичное отклонение точек кластеров от центров этих кластеров:

(1)

где k — число кластеров, Si — полученные кластеры, i=1,2,k,а mu — центры масс всех векторов x из кластера S.

Алгоритм представляет собой версию EM-алгоритма, применяемого также для разделения смеси гауссиан. Он разбивает множество элементов векторного пространства на заранее известное число кластеров k.

Основная идея заключается в том, что на каждой итерации перевычисляется центр масс для каждого кластера, полученного на предыдущем шаге, 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 V уменьшается, поэтому зацикливание невозможно.

На Рисунке 1 демонстрируется алгоритм разбития на кластеры:

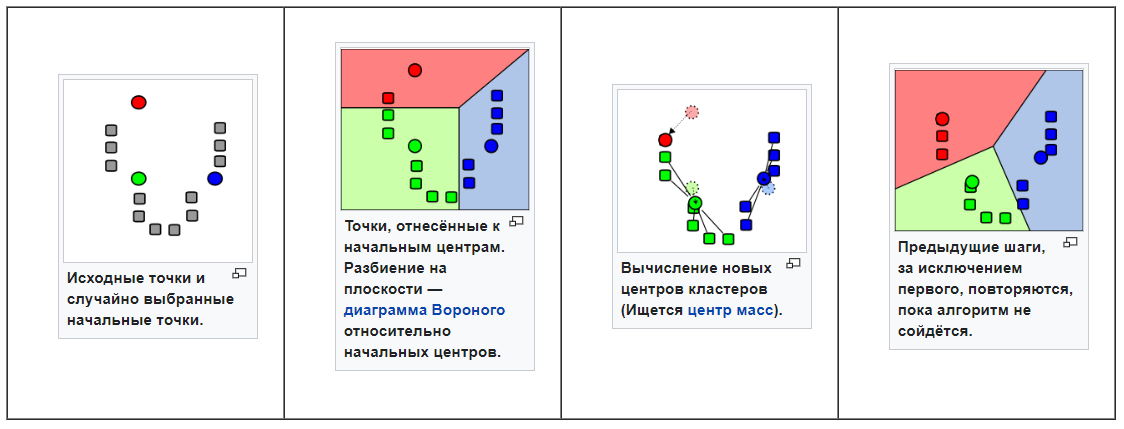


Рисунок 1 – схема работы алгоритма

# Глава 2. Реализация

public class Cluster {

public static void main(String[] args) {

int[] arr = {2, 4, 10, 12, 3, 20, 30, 11, 25};

System.out.println("Input: " + Arrays.toString(arr));

int i, m1, m2, m = 0; //значение центроидов, m - число иттераций

boolean flag; // флаг, который даст знать, что центры масс m1, m2 не меняются

int sum1, sum2; // суммы одномерных массивов

int a = arr[0];

int b = arr[1]; //a, b сохраняют предыдущее состояние центра масс, чтобы сравнить с текущим значением

m1 = a;

m2 = b;

int[] cluster1, cluster2;

do {

sum1 = 0;

sum2 = 0;

cluster1 = new int[arr.length];

cluster2 = new int[arr.length];

m++; //подсчет числа итераций

int k = 0, j = 0; //элементы, которые будут записывать значения в первый и второй массив

for (i = 0; i < arr.length; i++) {

if (Math.abs(arr[i] - m1) <= Math.abs(arr[i] - m2)) { //принимаем за первую центроиду первое значение массива, за вторую - второе и сравниваем

cluster1[k] = arr[i]; //если проходит, то записываем в первый кластер

k++;

} else {

cluster2[j] = arr[i];

j++;

}

}

for (i = 0; i < k; i++) { //вычисляем сумму для 1го кластера

sum1 += cluster1[i];

}

for (i = 0; i < j; i++) { //вычисляем сумму для 2го кластера

sum2 += cluster2[i];

}

a = m1; //записываем старые центроиды

b = m2;

m1 = sum1 / k; //высчитываем новые центроиды

m2 = sum2 / j;

flag = (m1 == a && m2 == b); //если оба значения центроида = старому => выйти из цикла

System.out.printf("%n\_\_\_\_\_\_\_\_\_\_\_\_ITERATION %d\_\_\_\_\_\_\_\_\_\_\_\_%n", m);

System.out.println("m1+" + m1 + " m2+" + m2);

System.out.println("Cluster 1:\n" + Arrays.toString(cluster1));

System.out.println("Cluster 2:\n" + Arrays.toString(cluster2));

} while (!flag);

System.out.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

System.out.println("Final value of centorid: m1=" + m1 + " m2=" + m2);

System.out.println();

System.out.println("Final cluster 1 is:\n" + Arrays.toString(cluster1));

System.out.println("Final cluster 2 is:\n" + Arrays.toString(cluster2));

}

}

# Итоги лабораторной работы

Мы освоили метод k-средних кластерного анализа.

# Список литературы

1. [[wikipedia.org](https://ru.wikipedia.org/wiki/Метод_k-средних)] – Метод k-средних