Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Пермский государственный аграрно-технологический университет

имени академика Д.Н. Прянишникова»

Кафедра информационных технологий и программной инженерии

**ЛАБОРАТОРНАЯ РАБОТА № 3**

Кластеризация.

|  |  |
| --- | --- |
| Выполнил: | группа ПИНб-4  П.С. Плотников |
| Проверил: | доцент каф. ИТиПИ,  Т.А. Казаченко |

Пермь-2024 г

# Ответы на вопросы

Таблица 1

|  |  |  |
| --- | --- | --- |
| **№** | **Вопрос** | **Ответ** |
|  | Как называются методы, изучаемые в лабораторной 3? |  |
|  | К какой категории методов ML они относятся? |  |
|  | Что может прогнозировать алгоритм, обученный данным методом? |  |
|  | В чем принципиальное отличие задачи классификации от задачи кластеризации? |  |
|  | Что значит «метод сошелся» для метода K-means? |  |
|  | Чем могут отличаться задачи кластеризации? |  |
|  | Приведите примеры «жесткой» и «мягкой» кластеризаций (не из указанных в лекции). |  |

**Формулировка задачи**

Разбейте множество объектов из набора данных pluton на 3 кластера методом центров тяжести (К-means). Сравните качество разбиения в зависимости от макисмального числа итераций алгоритма.

**Описание признаков используемого датасета**

Исходный датасет pluton содержит в себе четыре количественных признака, каждый из которых означает процент содержания конкретного изотопа плутония в партии замеров. Признаки:

1. Pu238: всегда меньше 2% ;
2. Pu239: находится в промежутке 60-80%;
3. Pu240: находится в промежутке 18-26%;
4. Pu241: всегда меньше 10%.

Всего в файле 45 объектов, то есть 45 партий замеров изотопов.

**Программный код**

# Подключение библиотеки визуализации

library(factoextra)

# Чтение данных

pluton = read.table("./pluton.txt", header = TRUE)

# Кол-во кластеров

k = 3

# Кол-во итераций

# При iters < 6 – нечеткое разделение

# При iters >= 6 – ровное разделение на 3 кластера

iters = 6

# Кластеризация методом k-средних

clusters = kmeans(pluton, k, nstart = iters)

# Отображение результата кластеризации на исходном датасете

fviz\_cluster(clusters, pluton, main = "Кластеризация плутона",

xlab = FALSE, ylab = FALSE)

**Результат вычислений**

На рисунке 1 показан результат кластеризации при трех итерациях, на рисунке 2 – при шести.

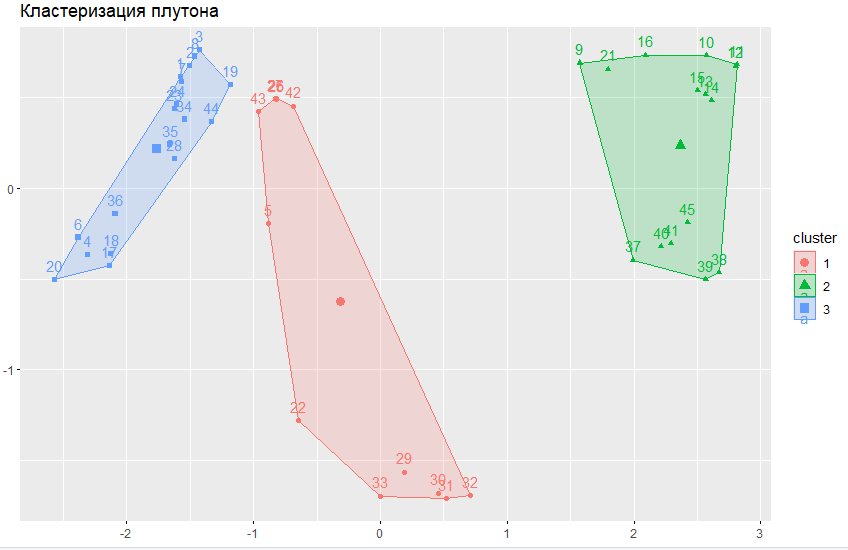


Рисунок 1 – Результат кластеризации при трех итерациях

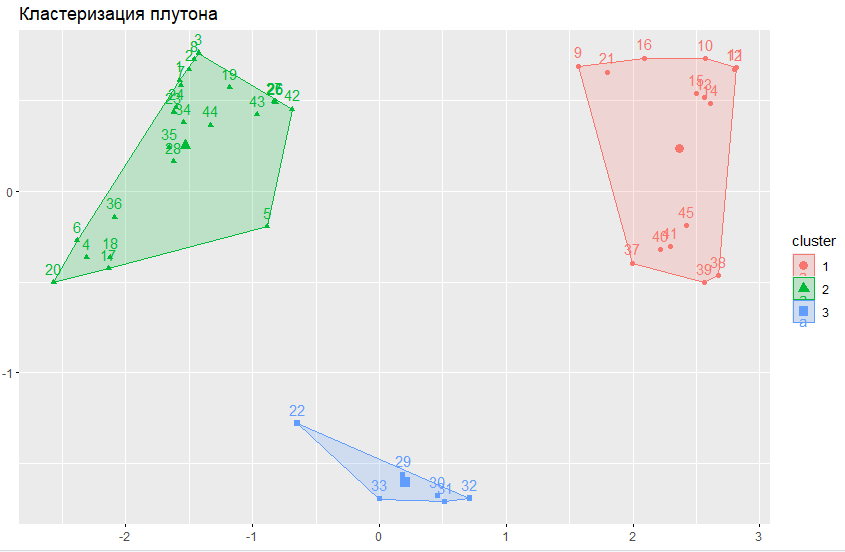


Рисунок 2 – Результат кластеризации при шести итерациях

**Пояснение результатов**

Как можно заметить на рисунке 1, некоторые объекты очевидно находятся не в своем кластере, а именно объекты под номерами: 5, 26, 27, 42 и 43. Связано это с тем, что центроиды кластеров выбираются случайным образом. Более того, количества итераций не достаточно, чтобы четко разбить облака данных на кластеры.

На рисунке 2, при шести итерациях, объекты четко разделяются на три кластера.

В кластер 1 попали изотопы, у которых количество Pu239 минимально. В верхней части кластера находятся объекты, у которых, в добавок, максимально Pu240. В нижней – максимально Pu241.

В кластер 2 попали объекты, у которых максимальны и Pu239, и Pu240.

В кластере 3 значение Pu240 минимально, остальные изотопы имеют приблизительно средние значений от своих возвожны

**Формулировка задачи**

Постройте дендрограмму для набора данных votes.repub (число голосов, поданных за республиканцев на выборах с 1856 по 1976 год). Проинтерпретируйте полученный результат.

**Описание признаков используемого датасета**

В используемом датасете votes.repub имеется два вида признаков: номинальный (штат) и количественные (процент проголосовавших). Для построения дендрограммы требуются только количественные признаки.

Количественные признаки делятся на года, в каждом годе имеется процент проголовавших за республиканцев.

**Прграммный код**

# Чтение данных

votes = read.table("./votes.repub.txt", header = TRUE)

# Избавляемся от NA

votes[is.na(votes)] = 0

# Заменим номера строк названиями шататов

rownames(votes) = votes$States

# Делаем срез, т.к. для дендрограммы важны голоса, а не штаты

votes = votes[,2:32]

# Получаем расстояния между точками

votes\_d = dist(votes)^2

# Получаем кластеризацию

votes\_h = hclust(votes\_d, method = "ward.D")

plot(votes\_h, main = "Дендрограма votes.repub")

# Выделим на графике 3 кластера

rect.hclust(votes\_h, k = 3, border="red")

**Результаты вычислений**

На рисунке 3 представлена дендрограмма иерархической кластеризации.

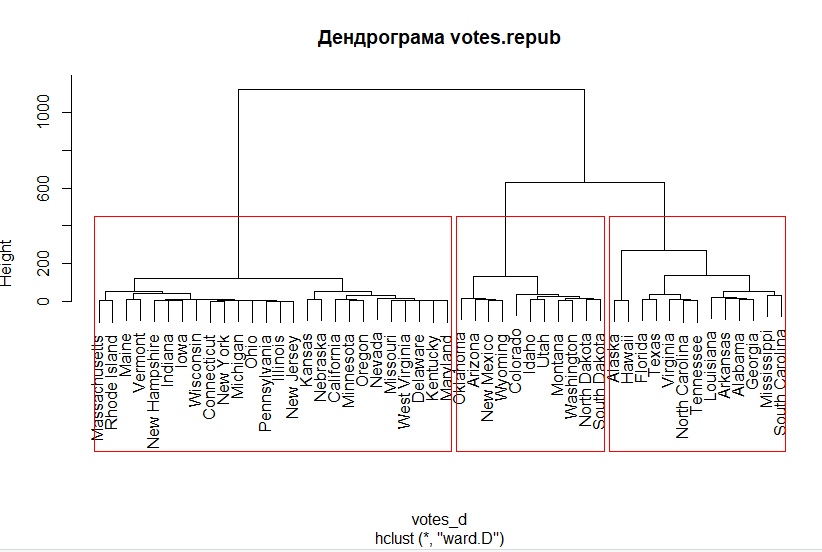


Рисунок 3 – Дендрограмма решения

**Пояснение результатов**

а