Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования «Кубанский государственный университет»

Кафедра вычислительных технологий

**ОТЧЕТ**

о выполнении лабораторной работы №5

по дисциплине «Обработка больших данных»

Тема**:** Задачи классификации и кластеризации

Выполнил: ст. гр. 39/1

Козлов Э.Д.

Проверил: преподаватель

Яхонтов А.А.

Краснодар

2025

**Часть 1.**

**Цель:** Закрепить знания, об алгоритмах классификации и кластеризации данных,

ознакомиться с некоторыми функциями языка R, осуществляющими этот вид анализа,

принципами их работы. Научиться визуализировать результаты работы функций кластерного

анализа и классификаторов, интерпретировать полученные результаты.

### **Задание**

1. В начале отчета необходимо разместить формулировку задания и фрагмент

исходного датасета.

2. Выполнить дескриптивный анализ данных (здесь приветствуются дополнительные

исследования).

3. Оценить оптимальное число кластеров, для этого построить диаграмму "Метод

силуэта", “Метод локтя”, "Статистику разрыва" и Алгоритм консенсуса.

4. Выполнить иерархическую кластеризацию вашего набора данных, построив

дендрограмму. Подробно обосновать Ваш выбор числа групп.

5. Построить диаграмму со столбчатыми диаграммами (рис. 5.8) и боксплотами групп

(рис. 5.12). Провести сравнительный анализ полученных групп.

6. Выполнить кластеризацию своего датасета по k-means (рис.5.9, 5.10).

7. Выполнить построение scatterplot (рис. 5.13) с помощью функций plot или pairs.

8. Построить трехмерную кластеризацию по scatterplot3d (5.16)

9. В целом: выполнить шаги 1-3,5 анализа для своего набора данных (если какие-то из

шагов нерелевантны вашему набору данных, объяснить почему).

**Ход работы**

**Дескриптивный анализ.**

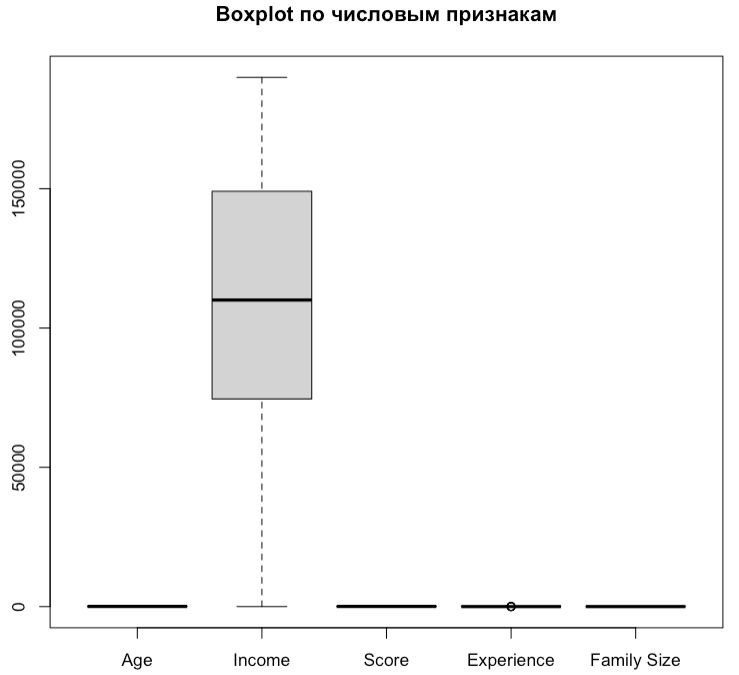


Рисунок 1 – Боксплот 1

Income (доход) достигает **190 000** — на фоне этого все остальные признаки (Age, Score, Family Size) выглядят как нули. Такой график бесполезен для сравнения — признаки не видны из-за разного масштаба. Поэтому следует нормализовать данные.

Изображение выглядит как текст, диаграмма, снимок экрана, Прямоугольник

Автоматически созданное описание

Рисунок 2 – Нормализованные данные

На графике видно, что:

У признака Family Size заметны выбросы, то есть есть клиенты с сильно отличающимся размером семьи.

Work Experience имеет медиану ниже среднего, что говорит о преобладании клиентов с малым стажем.

Изображение выглядит как текст, Шрифт, белый, чек

Автоматически созданное описание

Нормализация происходила с использованием метода scale() – он вычитает среднее значение по каждому признаку и делит результат на стандартное отклонение

Оценка оптимального числа кластеров.

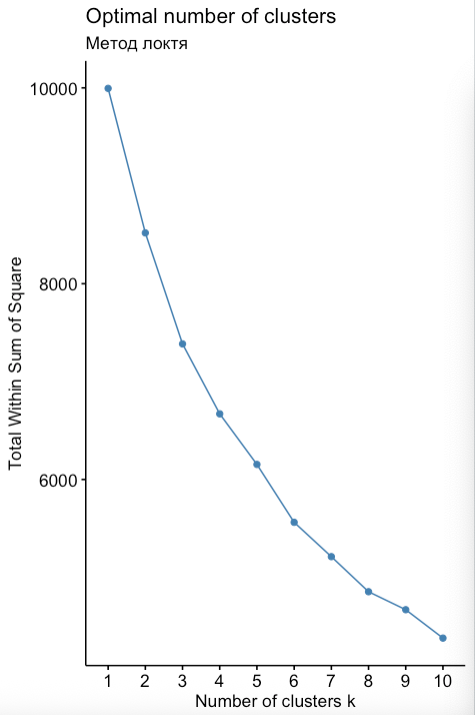


Рисунок 3 – Метод локтя

Видно, что от k=1 до k=2 и до k=3 WSS (сумма квадратов отклонений объектов от центроидов своих кластеров) падает сильно, а дальше снижение становится более плавным. Значит, кластеры k=3 объясняют большую часть разброса, а добавление 4‑го даёт уже небольшой выигрыш.

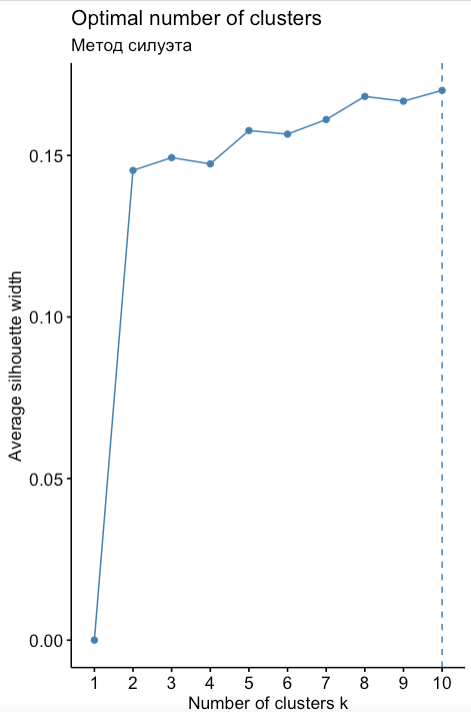


Рисунок 4 – Метод Силуэта

Сильный рост от k=1 к k=2 и к k=3 (с 0 до ~0.15), а дальше рост идёт очень вяло. Следовательно лучше брать k=3 как компромисс между чистотой и числом кластеров.

Изображение выглядит как текст, диаграмма, линия, График

Автоматически созданное описание

Рисунок 5 – Статистика разрыва

Чем больше gap, тем сильнее структура. Выбираем 7.

|  |  |
| --- | --- |
|  |  |

Таблица 1 – Алгоритм консенсуса

В обоих случаях (Hubert и Dindex) максимум второй разности приходится на k = 2. Значит, по этим двум метрикам наиболее существенное улучшение достигается при разбиении на два кластера.

По совокупности результатов можно сделать так:

«Локоть» (WSS) даёт локоть примерно при k=3

Силуэт достигает первого максимума тоже при k=3

Gap‑статистика даёт максимум чуть дальше (k≈7), но у k=3 разрыв тоже значимый

Вторая разность индексов из NbClust показывает резкий прирост при переходе от 1 к 2, а более гладкое, но всё ещё заметное улучшение – при 3

Из этого сделаю вывод, что будем делать 3 кластера.

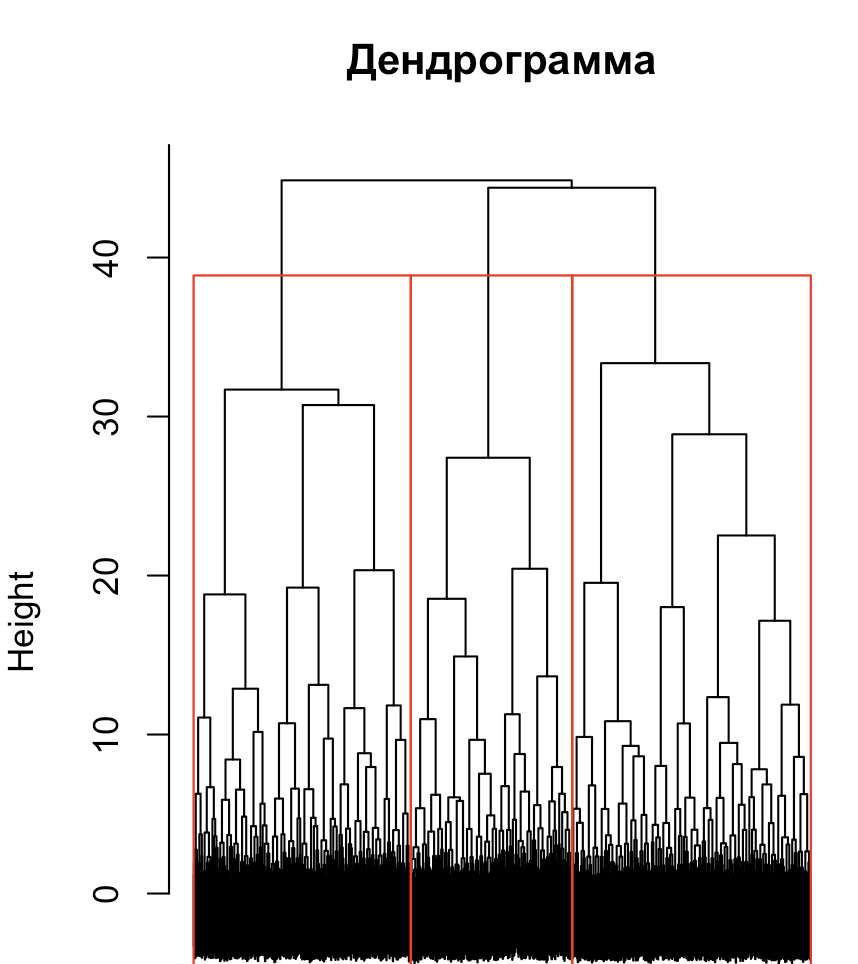


Рисунок 6 – Дендограмма

По дендрограмме видно, что наиболее похожие между собой клиенты объединяются в кластеры на низких уровнях, а более разнородные группы сливаются лишь на больших высотах. Чем выше точка слияния, тем сильнее отличаются объединяемые группы — именно поэтому после первых множества «тонких» слияний внутри кластеров на высоте около 30 формируются три хорошо разделённые группы.

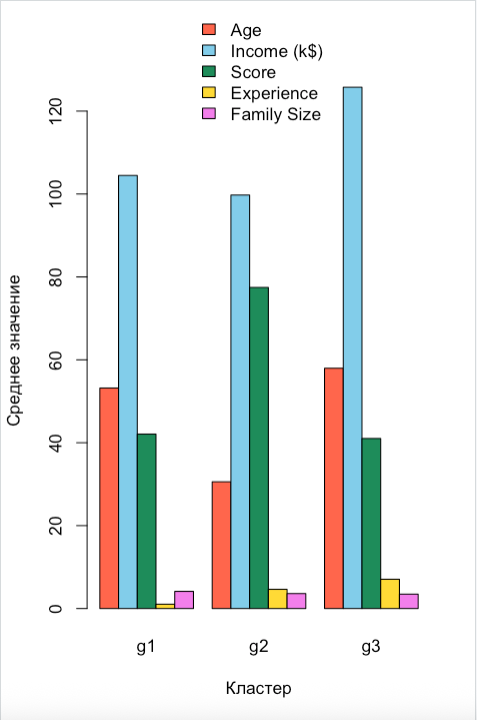


Рисунок 7 – Диаграмма средних значений

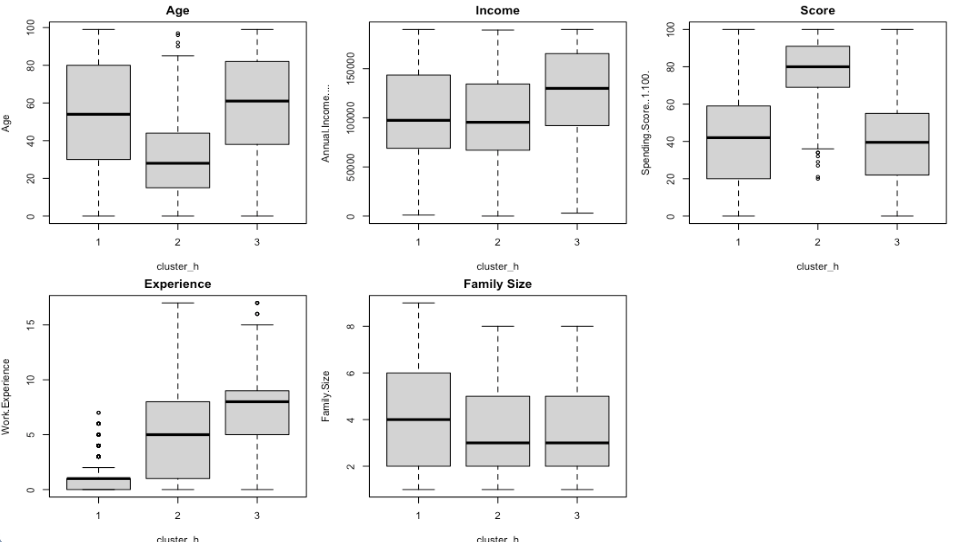


Рисунок 8 – Боксплоты групп

Выводы

– g2 (самые молодые, со средним доходом и самой высокой активностью расходов) можно назвать «энергичными» покупателями

– g3 (старшие, с самым высоким доходом и большим стажем, но низкой тратностью) представляют собой «обеспеченных» клиентов, тратящих сравнительно мало

– g1 (зрелые, со средним доходом, умеренной тратой, небольшим стажем и крупными семьями) — это «семейные» покупатели с ограниченным рабочим опытом

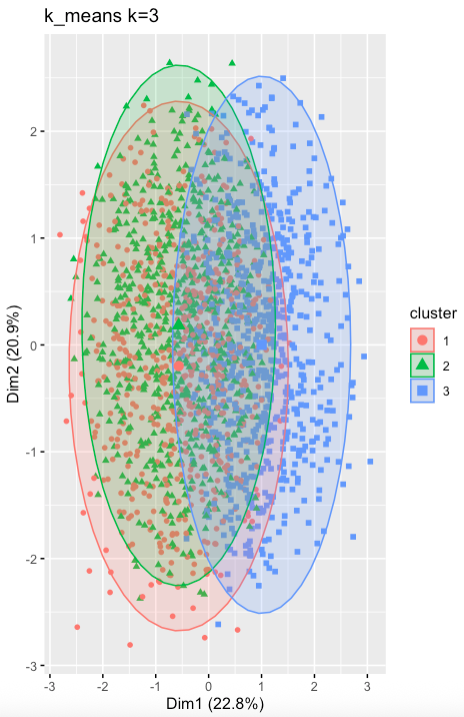


Рисунок 9 – Кластеризация по k-means

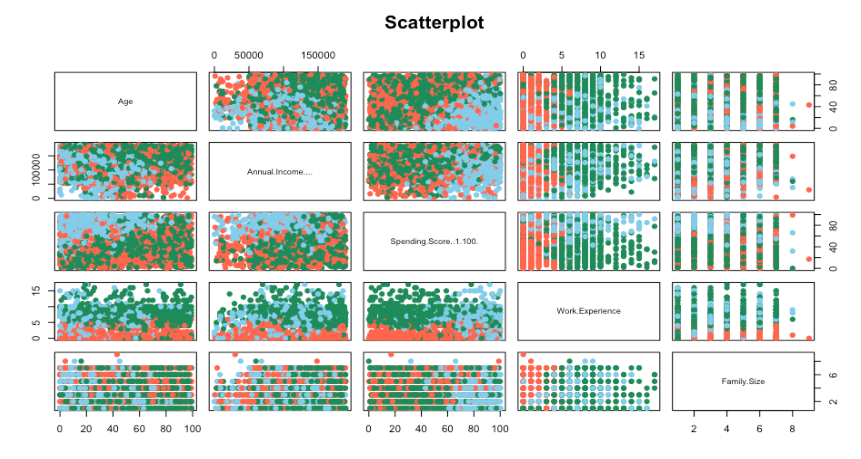


Рисунок 10 – Scatterplot

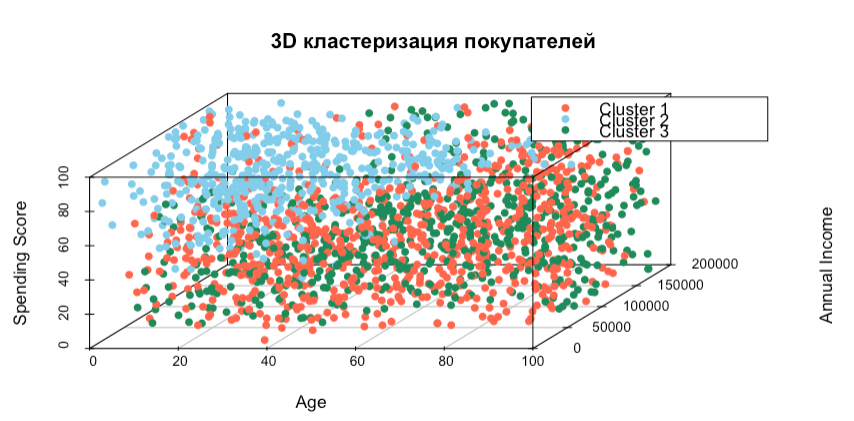


Рисунок 11 - 3D кластеризация покупателей

**Вывод:** В ходе работы мы закрепили принципы классификации и кластеризации, освоили ключевые функции R для этих задач, научились визуализировать результаты (дендрограммы, barplot, boxplot, scatterplot, 3D‑график) и дали интерпретацию полученным группам, полностью выполнив поставленную цель.

**Часть 2.**

**Цель**: Научиться выполнять классификацию на основе формулы Байеса и деревьев решений.

**Подготовка данных для классификации**

Добавили к датафрейму столбец кластеров cluster\_km.

Преобразовали cluster\_km в фактор (as.factor()).

Разделили данные на: Обучающую выборку (70% данных) и Тестовую выборку (30% данных).

**Наивный Байес.**

Построили модель с помощью функции naiveBayes()

Результатом точности оказалось 0.9057592 ~ 90%

**Дерево решений.**

Построили дерево решений с помощью функции ctree() и отрисовали его.

Результатом точности оказалось 0.938918 ~ 93%

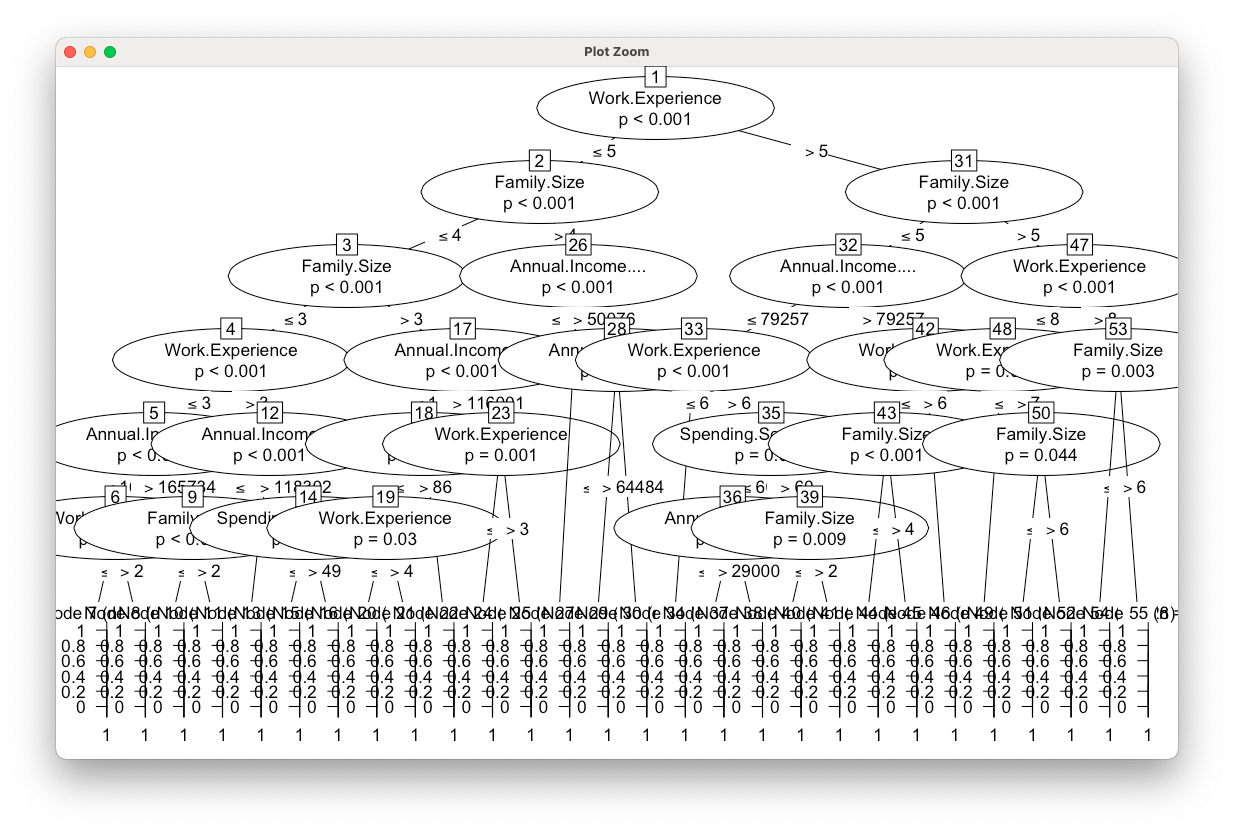


Рисунок 12 – Дерево решений.

**Случайный лес.**

Построили ансамбль деревьев решений с помощью функции randomForest()

Результат 0.9685864 ~ 96%

|  |  |
| --- | --- |
| Наивный Байес | ~ 90% |
| Дерево решений | ~ 93% |
| Случайный лес | ~ 96% |

При классификации наилучший результат показал метод Random Forest. Дерево решений тоже показало высокую точность, не сильно уступая Random Forest. Наивный Байес показал менее хороший результат, так как его предположение о независимости признаков не выполняется: возраст, доход и расходы между собой связаны.