Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО
ITMO University

ОТЧЁТ ПО ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

«Разработка веб-приложения на Python, предназначенное для анализа кластеризации с использованием алгоритмов k-means и иерархической кластеризации»

Обучающиеся:
Опалев Владимир Константинович
(редактор, ответственный за оформление отчёта, поток 23.5)
Арройо Ариас Николас Рафаэль
(редактор, ответственный за разработку кода, поток 23.5)
Руководитель:
Яворук Татьяна Олеговна

Санкт-Петербург
2025

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Данный проект представляет собой интерактивное веб-приложение на Python, предназначенное для анализа кластеризации с использованием алгоритмов k-means и иерархической кластеризации. Пользователь может выбрать один из предустановленных наборов данных или загрузить свой собственный .csv файл для анализа

Цель: Целью данного проекта является проведение кластерного анализа на основе реальных многомерных данных с использованием алгоритмов **k-means** и **иерархической кластеризации**. Предполагается, что ни структура, ни количество классов заранее не известны, поэтому задача относится к классу *обучения без учителя*.

Задачи:

Разработать модульную систему анализа данных, позволяющую:

- Выполнять предобработку данных (очистка, нормализация, понижение размерности);
- Строить кластеры с помощью методов k-means и иерархической кластеризации;
- Визуализировать распределение объектов и полученные кластеры;
- Оценивать качество кластеризации с использованием метрик (силуэт, инерция);
- Предоставить пользователю удобный интерфейс для запуска анализа.

Выбор методов

Алгоритм **k-means** позволяет разбить данные на заранее заданное количество кластеров, минимизируя внутрикластерное расстояние. Он широко применяется благодаря своей простоте и эффективности.

Алгоритм **иерархической кластеризации** формирует древовидную структуру объединения объектов на основе меры расстояния между ними. В отличие от k-means, он не требует указания числа кластеров на этапе построения дерева. Однако, для выделения финальных кластеров

необходимо "разрезать"дендрограмму на определённом уровне, что в данной реализации осуществляется путём задания желаемого количества кластеров пользователем.

Использование обоих методов даёт возможность сравнить поведение кластеризаторов на разных типах данных и оценить стабильность кластеров.

Описание используемых наборов данных

В ходе работы используются следующие открытые датасеты:

- *Wine dataset* — содержит 13 химических признаков винных образцов, принадлежащих к трём различным сортам (класс `target`). Это хорошо сбалансированный набор для демонстрации кластеризации с возможностью валидации по меткам.
- *Wholesale Customers dataset* — включает данные о закупках (в единицах объёма) различных категорий продуктов (молоко, бакалея, заморозка и др.) для клиентов оптовой торговли. Классы не заданы, задача — выделение клиентских сегментов.
- *Mall Customers dataset* — содержит демографические и поведенческие данные клиентов (возраст, доход, индекс расходов). Используется для сегментации клиентов на основе потребительского поведения.

Каждый из наборов данных подвергается масштабированию, а при необходимости — понижению размерности методом **PCA** (главных компонент) до двух признаков для визуализации кластеров.

Кроме предустановленных наборов, пользователь также может загрузить собственный CSV-файл для кластерного анализа. В этом случае система автоматически применит те же этапы обработки и визуализации к загруженному датасету, при условии, что структура данных позволяет это (числовые признаки, отсутствие пропусков и т.д.).

1 Теоретическая часть

1.1 Алгоритм k-means

`k-means` — это один из самых популярных алгоритмов кластеризации, относящийся к методам обучения без учителя. Его цель — разделить данные на k непересекающихся кластеров таким образом, чтобы внутрикластерная вариация (разброс) была минимальна.

Основная идея

Пусть задан набор данных $X = \{x_1, x_2, \ldots, x_n\}$, где каждый объект $x_i \in \mathbb{R}^d$ — это точка в d-мерном пространстве. Алгоритм k-means разбивает эти точки на k кластеров $C_1, C_2, \ldots, C_k$ путём минимизации следующей функции потерь (суммы квадратов расстояний до центроидов):

$$J = \sum_{j=1}^{k} \sum_{x_i \in C_j} \|x_i - \mu_j\|^2$$

где:

- μ_j — центр (центроид) кластера C_j;
- $\|x_i - \mu_j\|^2$ — квадрат евклидова расстояния между точкой и центром кластера.

Пошаговый алгоритм

Алгоритм работает итеративно и состоит из следующих этапов:

1. Инициализация: случайным образом выбираются k центров кластеров $\mu_1, \mu_2, \ldots, \mu_k$.
2. Шаг присваивания: каждому объекту x_i присваивается кластер C_j, чей центроид ближайший:

$$C_j = \left\{ x_i \mid \|x_i - \mu_j\|^2 \leq \|x_i - \mu_l\|^2 \text{ для всех } l = 1, \ldots, k \right\}$$

3. Шаг пересчёта центров: пересчитываются центры кластеров как среднее всех точек в каждом кластере:

$$\mu_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i$$

4. Повторять шаги 2–3 до сходимости (например, пока центры не перестанут существенно меняться).

Метрика расстояния

Наиболее часто используется евклидово расстояние между точками:

$$d(x_i, x_j) = \sqrt{\sum_{p=1}^{d} (x_i^{(p)} - x_j^{(p)})^2}$$

Однако могут применяться и другие метрики (манхэттенское, косинусное и т.д.), в зависимости от природы данных.

Преимущества и недостатки

`Преимущества`:

- Простота реализации и высокая скорость работы на больших данных;
- Хорошо работает при чётко выраженных, компактных и равномерных кластерах.

`Недостатки`:

- Требует заранее указать число кластеров k;
- Чувствителен к инициализации (может попасть в локальный минимум);
- Плохо работает при наличии выбросов и кластеров неправильной формы.

Выбор оптимального числа кластеров

Алгоритм k-means требует предварительно указать количество кластеров k, что может быть нетривиальной задачей. Для его выбора применяются следующие методы:

- *Метод локтя (Elbow method)* — основан на анализе *инерции* (внутрикластерной суммы квадратов расстояний до центроидов). Вычисляется инерция $J(k)$ для разных значений k и строится график зависимости $J(k)$ от k. С увеличением k инерция всегда уменьшается, но начиная с некоторого значения k^* темп уменьшения резко замедляется. Оптимальное значение k определяется как «излом» (локоть) на графике, изображённом на рисунке 1.1:

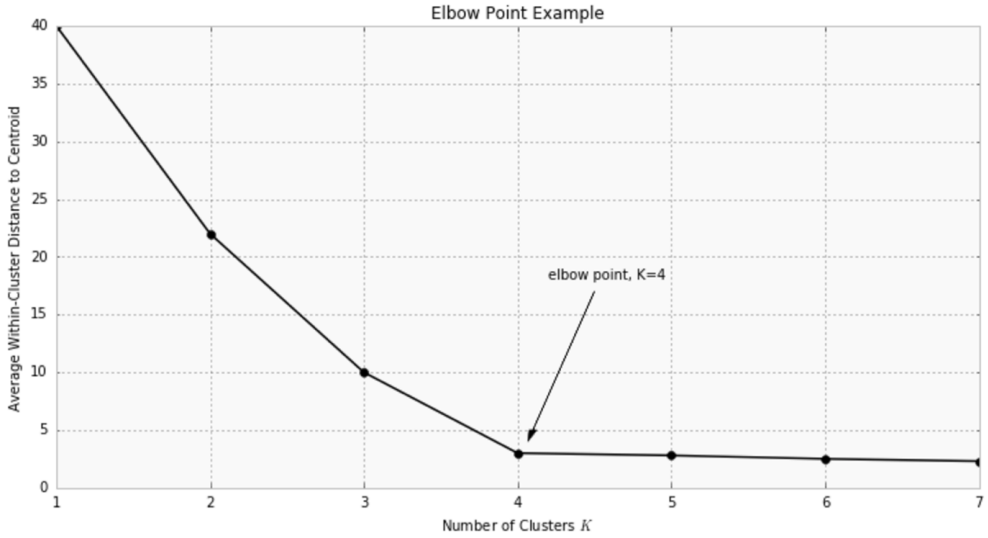

Рисунок 1.1 — Оптимальное значение k

В точке излома добавление новых кластеров перестаёт существенно улучшать качество разбиения.

- *Силуэт-анализ (Silhouette Score)* — измеряет качество кластеризации, оценивая, насколько хорошо каждый объект x_i соответствует своему кластеру по сравнению с ближайшим другим кластером. Для каждого объекта вычисляется коэффициент силуэта $s(i)$:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

где:

  - $a(i)$ — среднее расстояние от x_i до всех других точек в том же кластере;
  - $b(i)$ — минимальное среднее расстояние от x_i до точек в другом (наиболее близком) кластере.

Значения $s(i) \in [-1, 1]$, где:

- $s(i) \approx 1$ означает хорошо кластеризованный объект;
- $s(i) \approx 0$ — объект находится на границе кластеров;
- $s(i) < 0$ — вероятно, объект отнесён к неправильному кластеру.

Затем вычисляется среднее значение силуэта для всех объектов. Оптимальное значение k соответствует максимуму средней silhouette-оценки. Это значение можно визуализировать на *силуэт-графике*, который показывает распределение $s(i)$ по кластерам.

1.2 Алгоритм иерархической кластеризации

`Иерархическая кластеризация` — это метод группировки объектов, при котором создаётся иерархическая (древовидная) структура кластеров. В отличие от алгоритма k-means, иерархическая кластеризация не требует указания числа кластеров на этапе инициализации и является методом *обучения без учителя*.

Виды иерархической кластеризации

Существует два базовых подхода:

- *Агломеративная кластеризация* (снизу вверх): каждый объект сначала рассматривается как отдельный кластер, после чего на каждом шаге объединяются два наиболее близких кластера. Объединения продолжаются до тех пор, пока все объекты не окажутся в одном кластере.
- *Дивизивная кластеризация* (сверху вниз): начинается с одного общего кластера, который рекурсивно делится на подгруппы. Используется реже из-за вычислительной сложности.

В рамках данного проекта используется агломеративная стратегия.

Расстояние между кластерами (методы связи)

Выбор метода расчёта расстояния между кластерами напрямую влияет на структуру формируемого дерева и итоговые группы. Пусть A и B — два кластера, содержащие объекты $x_i \in A$ и $x_j \in B$. Тогда расстояние $D(A, B)$ между ними может определяться следующим образом:

- *Single linkage* (ближайший сосед):

$$D(A, B) = \min_{x_i \in A, x_j \in B} \|x_i - x_j\|$$

  Этот подход может приводить к «цепочкам», объединяющим разрозненные объекты.
- *Complete linkage* (самый дальний сосед):

$$D(A, B) = \max_{x_i \in A, x_j \in B} \|x_i - x_j\|$$

  Обеспечивает компактные, плотно сгруппированные кластеры.
- *Average linkage* (среднее расстояние):

$$D(A, B) = \frac{1}{|A||B|} \sum_{x_i \in A} \sum_{x_j \in B} \|x_i - x_j\|$$

  Представляет собой компромисс между предыдущими методами.
- *Метод Уорда (Ward's method)* — минимизирует прирост внутрикластерной дисперсии при объединении:

$$D(A, B) = \frac{|A||B|}{|A| + |B|} \cdot \|\mu_A - \mu_B\|^2$$

  где μ_A, μ_B — центры кластеров A и B соответственно.

Метод Уорда чаще всего применяется в практических задачах благодаря склонности формировать кластеры равномерной плотности и формы.

Результат кластеризации: дендрограмма

В результате работы агломеративного алгоритма строится *дендрограмма* — дерево, на каждом уровне которого отображается слияние двух ближайших кластеров. По оси Y указывается расстояние между объединяемыми кластерами. Такой способ представления позволяет пользователю интерпретировать структуру данных и выявлять естественные разбиения (Рисунок 1.2).

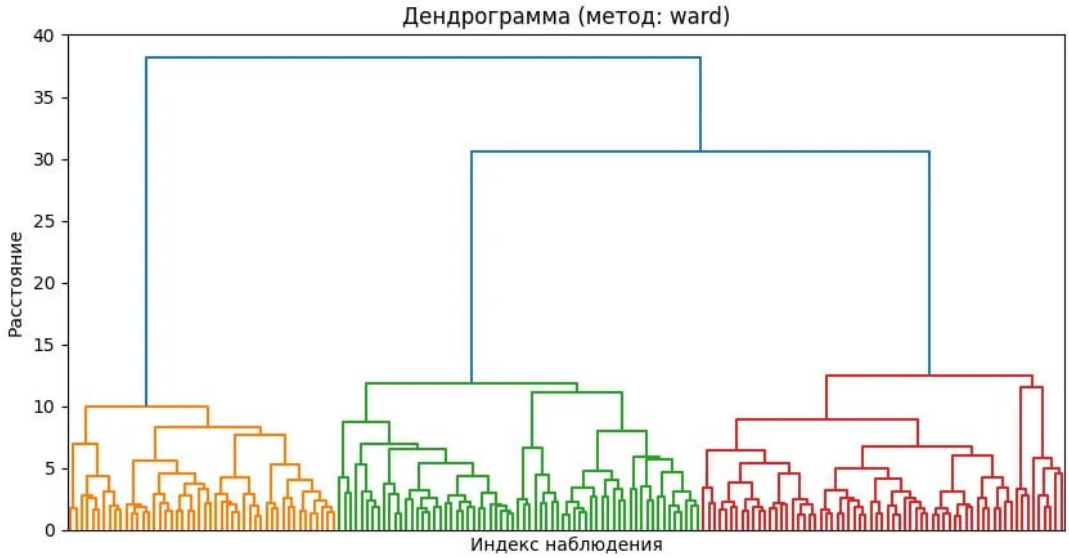

Рисунок 1.2 — Дендрограмма

Выбор количества кластеров

Хотя алгоритм сам по себе не требует указания числа кластеров k, на практике часто возникает необходимость получить фиксированное разбиение на k групп. Это реализуется путём «обрезания» дендрограммы на определённой высоте, соответствующей k кластерам. Такой подход позволяет контролировать уровень агрегации без изменения алгоритма.

В рамках нашей веб-интерфейсной системы, реализованной с использованием библиотеки Gradio, пользователю предоставляется возможность вручную задать желаемое количество кластеров. Это приводит к автоматическому усечению иерархической структуры на нужном уровне, после чего визуализируются полученные группы. Такой подход остаётся полностью валидным в контексте иерархической кластеризации и широко

применяется в реальных аналитических задачах, поскольку сохраняет все преимущества метода и гибкость интерпретации.

1.3 Метод главных компонент (PCA)

`Метод главных компонент` (Principal Component Analysis, PCA) — это классический метод понижения размерности данных, широко используемый в анализе многомерных выборок и визуализации. Его цель — уменьшить число признаков, сохранив при этом как можно больше информации о дисперсии данных.

Математическая постановка задачи

Пусть задана выборка $X \in \mathbb{R}^{n \times d}$, содержащая n наблюдений и d признаков. Метод PCA ищет новую ортонормированную систему координат (главные компоненты), в которой: - первая компонента объясняет максимально возможную дисперсию данных; - каждая следующая компонента ортогональна предыдущим и объясняет максимально возможную оставшуюся дисперсию.

Построение PCA основано на спектральном разложении ковариационной матрицы Σ:

$$\Sigma = \frac{1}{n} X^\top X = Q \Lambda Q^\top$$

где:

- Q — матрица собственных векторов (направления главных компонент),
- Λ — диагональная матрица собственных значений (дисперсий вдоль компонент).

Результатом применения PCA является проекция исходных данных X на пространство меньшей размерности:

$$Z = XQ_k$$

где Q_k — первые k столбцов из Q, соответствующие наибольшим k собственным значениям.

Зачем применять PCA в кластеризации

Кластеризация работает хуже, если признаки:

- имеют сильно разную шкалу или коррелированы;
- содержат «шумовые» измерения (низкоинформативные);
- превышают 2–3 измерения, затрудняя визуализацию.

Метод PCA позволяет:

- устранить коррелированные признаки;
- визуализировать многомерные данные в двумерном пространстве (для графиков кластеров);
- ускорить работу алгоритмов кластеризации за счёт уменьшения размерности;
- устранить выбросы и уменьшить влияние шума.

Когда PCA применялся в проекте

В нашей системе PCA применяется по выбору пользователя перед кластеризацией, если исходное пространство имеет более двух признаков ($d > 2$). Это особенно полезно для:

- *Wine dataset* — содержит 13 признаков: без PCA визуализация невозможна;
- *Wholesale dataset* — 6 признаков: PCA позволяет компактно отразить клиентские сегменты;

Для *Mall Customers dataset*, где уже есть только два числовых признака (ежемесячные расходы и доход), PCA не применяется, поскольку дальнейшее понижение размерности нецелесообразно.

Объяснённая дисперсия

Одним из важных выходов метода является *доля объяснённой дисперсии* (explained variance ratio), т.е. какая часть общей изменчивости данных сохраняется при проекции. В нашей системе она рассчитывается как:

$$\text{Explained variance ratio} = \frac{\sum_{i=1}^{k} \lambda_i}{\sum_{j=1}^{d} \lambda_j}$$

Это значение отображается пользователю после применения PCA, позволяя судить, насколько адекватно сниженная размерность сохраняет структуру данных.

1.4 Метрики оценки качества кластеризации

Поскольку кластеризация является методом обучения без учителя и заранее заданных меток классов не существует, оценка качества разбиения на кластеры требует специальных метрик. В данной работе используются две ключевые меры:

- *Инерция (inertia)* — мера компактности кластеров;
- *Силуэт-оценка (silhouette score)* — мера согласованности объекта с кластером.

Обе метрики автоматически рассчитываются в веб-приложении для каждого результата кластеризации методом k-means, и их значения отображаются пользователю после выполнения анализа.

Инерция (Inertia)

Инерция — это сумма квадратов расстояний всех точек до центров своих кластеров. В терминах функции потерь алгоритма k-means она выражается как:

$$\text{Inertia} = \sum_{j=1}^{k} \sum_{x_i \in C_j} \|x_i - \mu_j\|^2$$

где:

- C_j — кластер j;
- μ_j — центр (центроид) кластера C_j;
- $\|\cdot\|$ — евклидова норма.

Инерция всегда убывает с увеличением числа кластеров k, так как каждый объект находится ближе к более локальному центру. Однако чрезмерное увеличение k приводит к переобучению, поэтому инерция полезна в сочетании с *методом локтя* (Elbow method) для выбора оптимального k.

В нашем проекте график зависимости инерции от k автоматически строится и предоставляется пользователю для анализа.

Силуэт-оценка (Silhouette Score)

Силуэт-оценка позволяет оценить, насколько хорошо каждый объект соответствует своему кластеру по сравнению с другими кластерами. Для каждого объекта x_i вычисляется коэффициент силуэта $s(i)$:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}, \quad s(i) \in [-1, 1]$$

где:

- $a(i)$ — среднее расстояние от x_i до всех других точек в том же кластере (внутрикластерная плотность);
- $b(i)$ — минимальное среднее расстояние от x_i до точек в другом (наиболее близком) кластере (межкластерная близость).

Интерпретация значений:

- $s(i) \approx 1$ — объект хорошо соответствует своему кластеру;
- $s(i) \approx 0$ — объект находится на границе между кластерами;
- $s(i) < 0$ — объект, вероятно, попал не в свой кластер.

Среднее значение $s(i)$ по всей выборке используется как глобальная метрика качества кластеризации. В нашем приложении оно рассчитывается для результата k-means и визуализируется в виде силуэт-графика, где показано распределение $s(i)$ по кластерам.

Зачем нужны обе метрики

Инерция — это внутренняя характеристика плотности кластеров и удобна для построения графика «локтя», но она не учитывает расстояние до других кластеров.

Силуэт-оценка, напротив, сравнивает каждый объект как внутри, так и вне своего кластера, предоставляя более сбалансированную оценку качества.

Использование обеих метрик вместе позволяет:

- выбирать число кластеров, которое не только минимизирует инерцию, но и максимизирует силуэт;
- оценивать, есть ли «плохие» кластеры с низким качеством;
- определять, насколько разбиение отражает естественную структуру данных.

2 Используемые программные средства

2.1 Технологический стек проекта

Для реализации данного проекта была использована современная экосистема Python, подходящая для задач анализа данных, машинного обучения и визуализации. Ниже перечислены основные компоненты, использованные в процессе разработки, а также их назначение и роль в проекте.

Язык программирования

Проект реализован на языке `Python 3.8` — одном из самых распространённых языков в области анализа данных и машинного обучения. Python обеспечивает высокую читаемость кода, широкую поддержку научных библиотек и активное сообщество. Его гибкость позволила удобно реализовать как математическую логику кластеризации, так и визуальную часть.

Среда разработки

Для написания и отладки кода использовалась среда `Visual Studio Code (VSCode)`, обладающая встроенной поддержкой Python, автодополнением, управлением виртуальными окружениями и интеграцией с GitHub.

Изоляция зависимостей обеспечивалась с помощью `виртуального окружения (venv)`, созданного локально. Это позволяет избежать конфликтов библиотек и обеспечить воспроизводимость среды.

Используемые библиотеки

Для выполнения задач обработки данных, кластеризации, визуализации и построения веб-интерфейса были использованы следующие Python-библиотеки:

- *pandas* — обработка табличных данных, загрузка CSV-файлов, фильтрация и очистка. Используется в модуле `preprocessing.py` для предварительной подготовки данных.
- *numpy* — работа с массивами, линейная алгебра, передача данных между модулями. Является фундаментом для большинства других библиотек.
- *scikit-learn* — реализация основных алгоритмов машинного обучения: `KMeans`, `AgglomerativeClustering`, `PCA`, `StandardScaler`. В проекте эта библиотека лежит в основе всей логики кластеризации и снижения размерности.
- *scipy* — поддержка алгоритма иерархической кластеризации и построения дендрограмм через `scipy.cluster.hierarchy` и `linkage`. Используется в модуле визуализации.
- *matplotlib* и *seaborn* — построение графиков распределения, кластеров, силуэт-оценок и метода локтя. Визуальные компоненты проекта реализованы через эти библиотеки в `visualization.py`.
- *gradio* — библиотека для быстрого создания веб-интерфейсов к Python-программам. Используется в `app.py` для организации интерактивного интерфейса, позволяющего пользователю выбрать датасет, метод кластеризации, количество кластеров и параметры PCA, а также сразу видеть результат в виде графиков.

Все библиотеки указаны в файле *requirements.txt* и могут быть установлены одной командой, обеспечивая воспроизводимость проекта.

2.1.1 Ссылка на репозиторий

Полный исходный код проекта размещён в открытом репозитории на GitHub по ссылке: `https://github.com/NicolasRAA/clustering-webapp`

2.2 Описание структуры проекта

Проект структурирован в виде модульного репозитория, обеспечивающего раздельную ответственность каждого компонента. Все модули разделены по функциональности, что упрощает поддержку и расширение проекта. Ниже приведено описание структуры:

- *app.py* — основной исполняемый файл. Запускает веб-интерфейс Gradio, обрабатывает действия пользователя, вызывает предобработку, кластеризацию и визуализацию. Здесь объединяются все логические блоки проекта.
- *generate_datasets.py* — вспомогательный скрипт, позволяющий загрузить и сохранить три предустановленных набора данных (Wine, Wholesale, Mall Customers) в виде CSV-файлов в папке *datasets/*.
- *requirements.txt* — список зависимостей проекта, необходимых для установки. Позволяет создать воспроизводимую среду.
- *README.md* — документация по установке, запуску и использованию проекта. Также содержит описание рекомендованных параметров кластеризации.
- *.gitignore* — файл, исключающий временные и служебные файлы из индексации Git (например, *venv/*, *__pycache__*).
- Папка *analysis/* — основной логический блок, содержащий все модули, отвечающие за обработку и анализ данных:
  - *__init__.py* — файл инициализации, обозначает папку как модуль Python.
  - *preprocessing.py* — содержит функции для загрузки данных, удаления нечисловых колонок, масштабирования признаков с помощью *StandardScaler*, а также применения PCA для снижения размерности. Использует формулу дисперсионного объяснения:

$$\text{Explained Variance Ratio} = \frac{\lambda_i}{\sum \lambda_j}, \quad \lambda_i = \text{собственное значение}$$

  - *clustering.py* — реализует методы кластеризации:

    * *KMeans* — с фиксированным числом кластеров k, использует метод минимизации инерции.
    * *Agglomerative Clustering* — иерархическая кластеризация с различными метриками связи: *ward*, *average*, *complete*, *single*.
  - *evaluation.py* — рассчитывает количественные метрики качества кластеризации:
    * *Инерция (Inertia)*: сумма квадратов расстояний до ближайшего центра.
    * *Silhouette Score*:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}},$$

      где $a(i)$ — внутрикластерное расстояние, $b(i)$ — расстояние до ближайшего кластера.
  - *visualization.py* — отвечает за построение всех графиков:
    * Графики кластеров после KMeans и иерархии
    * Дендрограмма
    * График локтя
    * Силуэт-график
    * График с истинными метками, если доступны (например, колонка *target* в Wine)
- Папка *datasets/* — содержит три предустановленных CSV-файла с наборами данных:
  - *wine.csv* — химические свойства и целевая переменная (`target`) для классификации 3 видов вина.
  - *wholesale.csv* — данные о закупках различных категорий товаров.
  - *mall.csv* — данные о клиентах торгового центра: возраст, пол, доход и индекс трат.
- Папка *assets/* — содержит вспомогательные ресурсы, включая изображение логотипа *logo.png*, отображаемое в верхней части интерфейса Gradio.

Важно отметить, что в рамках отчёта представлено только текстовое и визуальное описание основных компонентов проекта. Для более глубокого

изучения реализации, подробностей по архитектуре и комментариев к коду рекомендуется ознакомиться с исходным кодом на GitHub.

3 Результаты (анализ по каждому датасету)

В данном разделе проводится подробный анализ результатов кластеризации для каждого из трёх наборов данных. Мы исследуем, как предварительная обработка данных (в частности, метод главных компонент — PCA), а также выбор количества кластеров k и метода связи влияют на итоговую сегментацию. Для оценки качества кластеризации используются метрики инерции и силуэта.

3.1 Wine

Выбор параметров

Для анализа набора данных Wine были выбраны следующие параметры:

- *Количество кластеров:* $k = 3$, так как известно, что в наборе содержится три типа вина (метка `target`: 0, 1, 2).
- *Метод кластеризации:* как k-means, так и иерархическая кластеризация.
- *Метод связи:* `ward`, так как он минимизирует внутрикластерную дисперсию и наиболее подходит для числовых признаков.
- *Применение PCA:* Да, так как изначально в данных 13 признаков, и визуализация возможна только после снижения размерности.

Применение PCA и объяснённая дисперсия

Для визуализации кластеров и анализа структуры данных был применён метод главных компонент (PCA). После снижения размерности до двух компонент, совокупная объяснённая дисперсия составила `57.38%`, что позволяет сделать вывод о том, что две главные компоненты улавливают большую часть информации о вариации в данных.

Визуализация распределения и истинных меток

На рисунке 3.1 представлено распределение объектов в пространстве первых двух главных компонент с окраской по реальным меткам (целевой переменной). Видно, что три класса достаточно чётко разделяются, особенно класс 2 (зелёный), который формирует компактную группу справа.

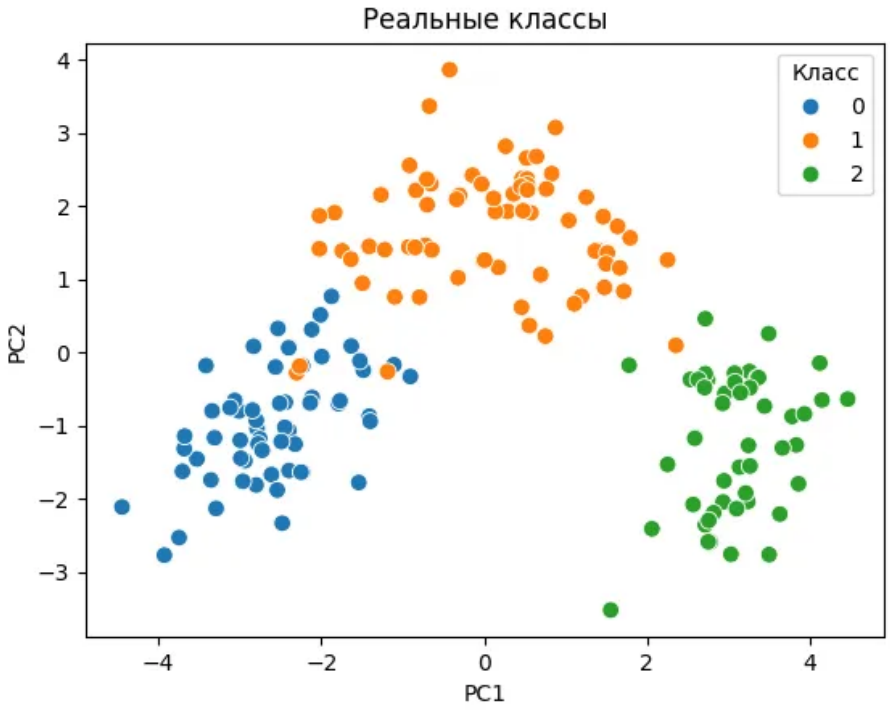

Рисунок 3.1 — Распределение по PCA с истинными метками классов

Кластеры, полученные методом K-means

На рисунке 3.2 показаны результаты кластеризации методом k-means. Центроиды отмечены чёрными крестами. По визуальному совпадению с рисунком 3.1 можно заметить хорошее соответствие между реальными метками и предсказанными кластерами, хотя некоторые элементы (особенно между классами 0 и 1) были перепутаны.

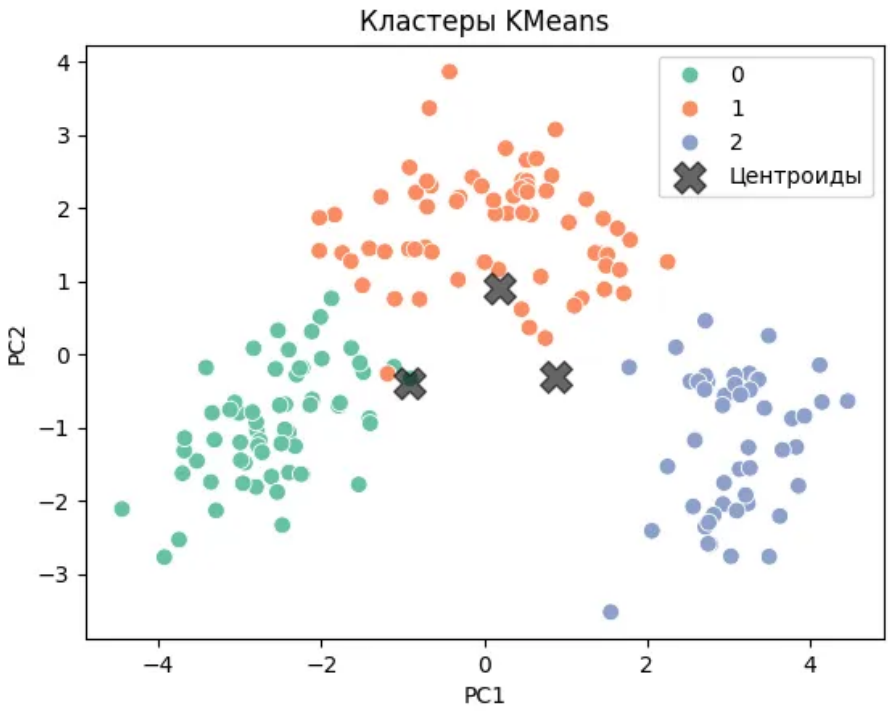

Рисунок 3.2 — Результаты кластеризации методом K-means ($k = 3$)

Результаты иерархической кластеризации

На рисунке 3.3 показан результат агломеративной кластеризации с методом связи `ward`. Видно, что структура кластеров также повторяет разделение на три группы, хотя наблюдаются отличия в границах между сегментами.

Рисунок 3.3 — Иерархическая кластеризация ($k = 3$, метод `ward`)

Дендрограмма

На рисунке 3.4 показана дендрограмма, отражающая иерархическую структуру слияния объектов. Видно, что выбор трёх кластеров (по трём крупным ветвям) является обоснованным.

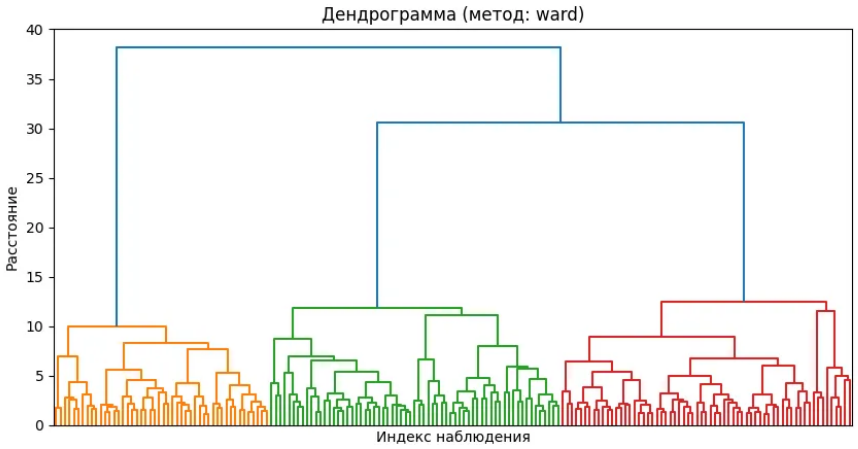

Рисунок 3.4 — Дендрограмма для Wine (метод `ward`)

Оценка качества кластеризации

Для количественной оценки качества кластеризации использовались две ключевые метрики: `инерция` и `силуэт-оценка (Silhouette Score)`.

- *Инерция:* 1285.56 — это сумма квадратов расстояний между каждым объектом и центроидом своего кластера. Чем меньше инерция, тем плотнее объекты расположены внутри кластера, а значит, сегментация считается более «собранной». Полученное значение 1285.56 при $k = 3$ указывает на приемлемую плотность кластеров, особенно учитывая изначальную размерность данных (13 признаков) и её снижение до 2D через PCA.
- *Silhouette Score:* 0.308 — данная метрика измеряет, насколько хорошо каждый объект согласуется со своим кластером по сравнению с соседними. Значения выше 0.5 считаются хорошими, в диапазоне 0.3–0.5 — умеренными, а ниже 0.2 — слабыми. Таким образом, полученное значение говорит о среднем качестве кластеризации, что ожидаемо при наличии некоторого перекрытия между классами.

На рисунке 3.5 изображён график метода локтя. Видно, что при $k = 3$ наблюдается заметный излом (локоть), после которого темп уменьшения инерции замедляется. Это подтверждает, что выбор $k = 3$ является обоснованным, так как добавление большего числа кластеров не даёт существенного прироста в уплотнении данных.

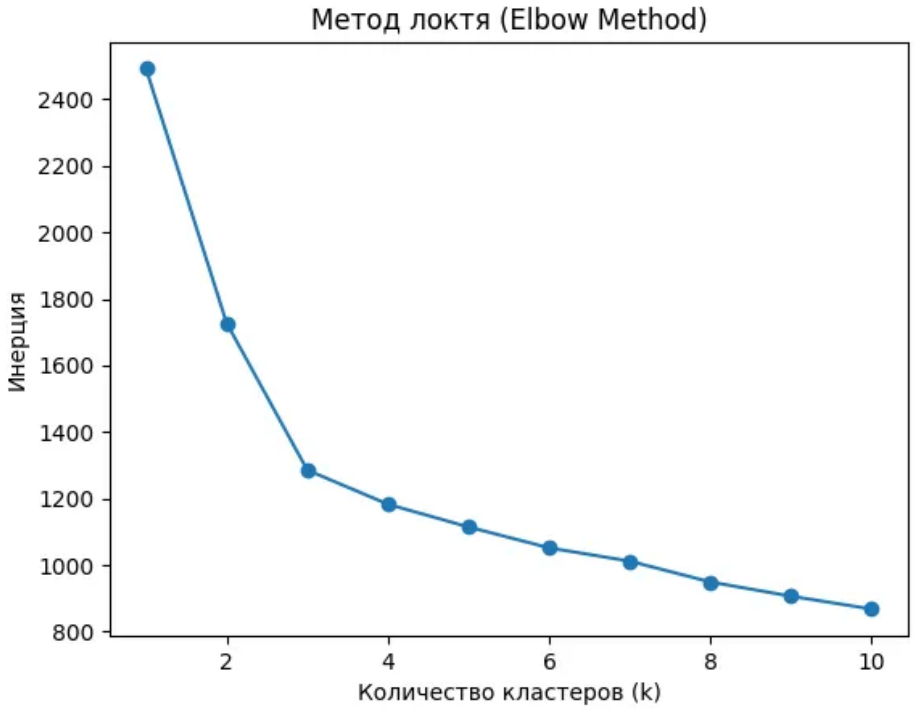

Рисунок 3.5 — График метода локтя (Elbow method)

На рисунке 3.6 представлен силуэт-анализ для $k = 3$. Каждому кластеру соответствует горизонтальный блок, ширина которого пропорциональна количеству объектов, а ширина в горизонтальном направлении — значение силуэт-коэффициента. Красная пунктирная линия показывает среднее значение — около 0.308. Видно, что большинство объектов имеют положительное значение силуэта, что указывает на удовлетворительное разделение кластеров. Однако наличие части объектов с отрицательным значением говорит о некотором наложении классов и граничных случаях, когда объект может относиться к соседнему кластеру.

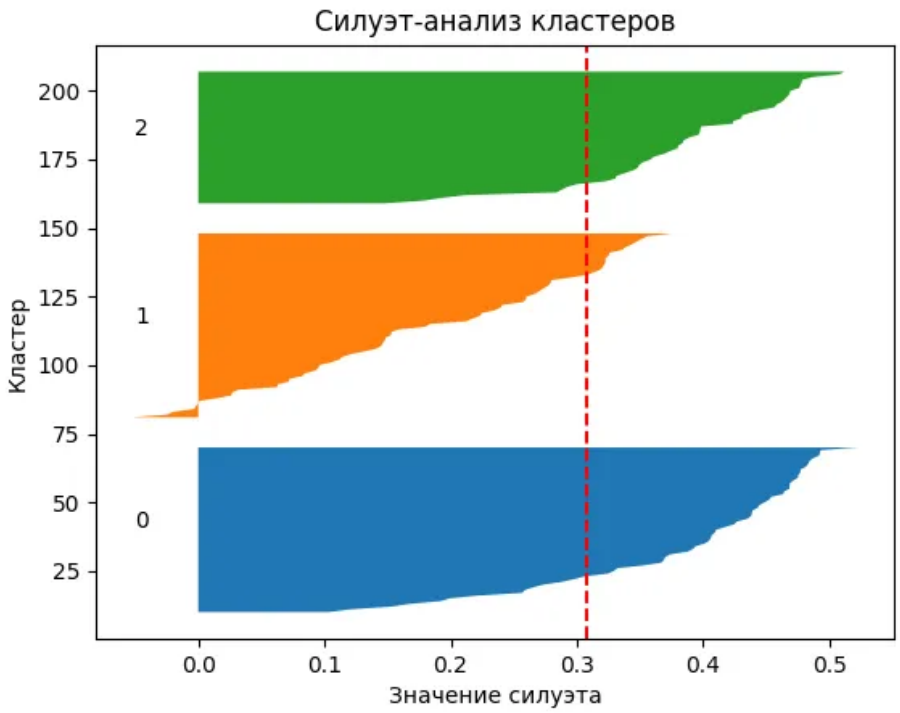

Рисунок 3.6 — Силуэт-анализ кластеров

Выводы по Wine

- Алгоритмы кластеризации успешно воспроизводят структуру данных: как k-means, так и иерархическая кластеризация корректно сегментируют три вида вина.
- Использование PCA позволило визуализировать и интерпретировать результаты, сохранив 57% дисперсии.
- Выбранное значение $k = 3$ подтверждается как визуально, так и количественно (метод локтя, silhouette).
- Метки кластеров в значительной степени совпадают с реальными метками `target`, что доказывает эффективность методов.

3.2 Wholesale

Выбор параметров

Для анализа набора данных `Wholesale` были выбраны следующие параметры:

- *Количество кластеров:* $k = 5$. Это значение было выбрано на основе визуального анализа дендрограммы и метода локтя (см. рисунки 3.9 и 3.11), где при $k = 5$ наблюдается заметное «плечо» в графике инерции.
- *Метод кластеризации:* k-means и агломеративная кластеризация.
- *Метод связи:* `average` — часто используется для бизнес-данных, таких как покупки и потребительские расходы, поскольку учитывает среднее расстояние между всеми парами объектов из разных кластеров, обеспечивая более стабильную структуру кластеров в условиях высокой дисперсии.
- *Применение PCA:* Да, так как набор содержит 6 признаков, и для качественной визуализации было выполнено понижение размерности до двух компонент.

Wholesale (метод average)

Применение PCA и объяснённая дисперсия.

После применения метода главных компонент (PCA), две первые компоненты объясняют 61.12% общей дисперсии признаков. Это означает, что основная структура данных хорошо сохраняется в двумерном пространстве, что подтверждает допустимость визуализации.

Визуализация кластеров.

На рисунке 3.7 показаны кластеры, полученные методом k-means. Можно выделить компактные сегменты (например, кластер 3) и более рассеянные группы, особенно кластеры 0 и 2. Наличие точек вдали от

основной массы данных указывает на потенциальные выбросы — клиентов с нестандартным профилем потребления.

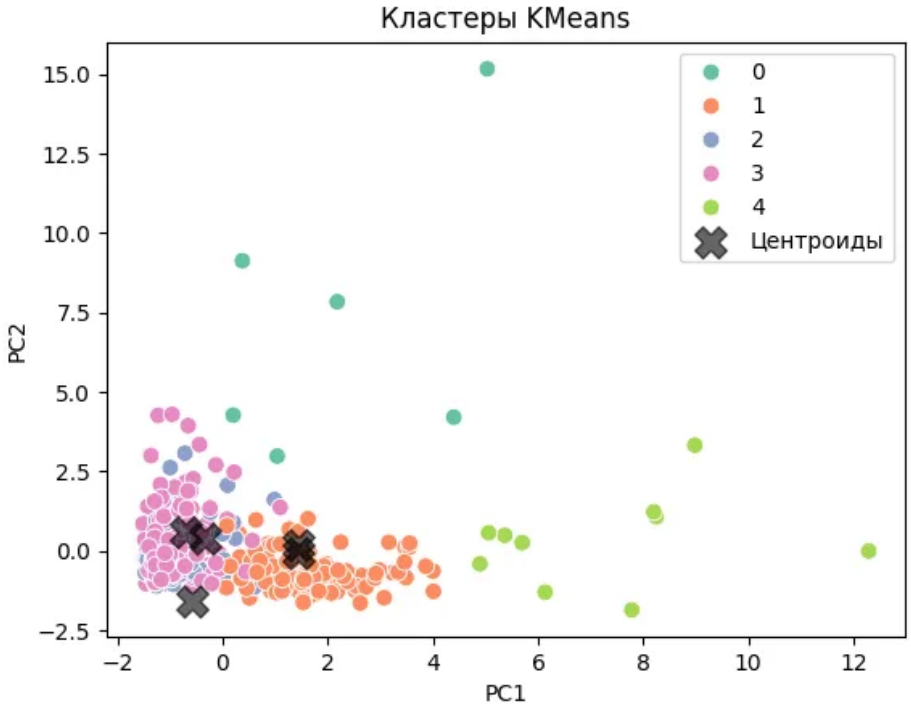

Рисунок 3.7 — Кластеры KMeans ($k = 5$) для набора Wholesale (PCA)

Агломеративная кластеризация с методом `average` дала результат, при котором большинство точек оказалось в одном кластере (кластер 1), а остальные распределены между малыми группами. Это может указывать на слабую дифференциацию клиентов при использовании среднего расстояния.

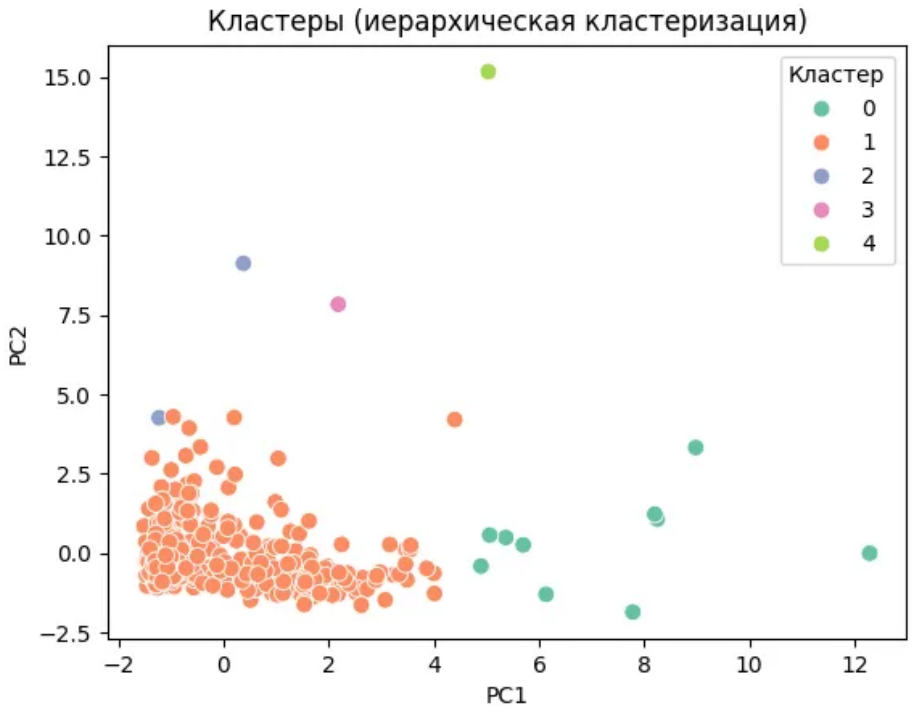

Рисунок 3.8 — Кластеры иерархической кластеризации (average)

Структура данных и дендрограмма.

На рисунке 3.9 показана дендрограмма, построенная методом `average`. При высоте отсечения около 10 формируются примерно пять основных ветвей, что подтверждает выбор $k = 5$. Однако наблюдается слабая иерархическая структура, где большая часть наблюдений сгруппирована близко, а удалённые точки формируют отдельные кластеры.

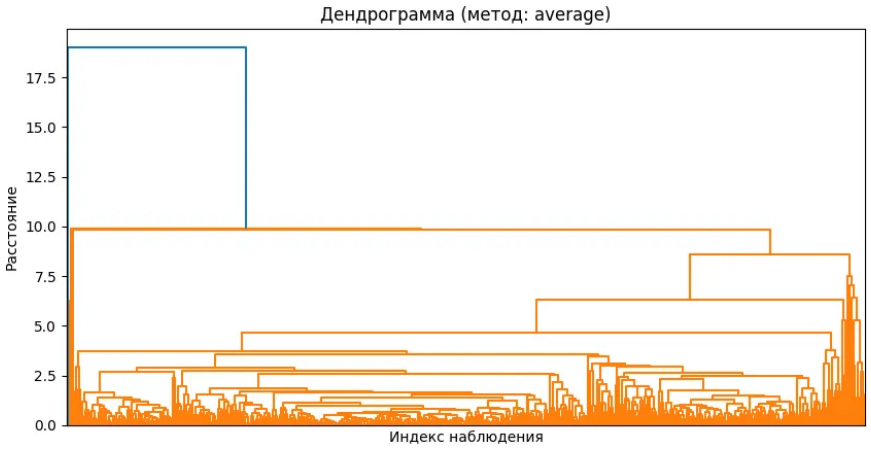

Рисунок 3.9 — Дендрограмма (метод average)

Метрики кластеризации.

- *Инерция:* 1551.67. Это отражает внутрикластерную дисперсию. Учитывая высокий разброс расходов у клиентов, значение является приемлемым, но не минимальным.
- *Silhouette Score:* 0.353 — умеренное качество кластеризации. Как видно из рисунка 3.10, кластер 0 имеет отрицательные значения силуэта, что указывает на возможную ошибку в его определении. Наоборот, кластер 3 демонстрирует хорошую степень согласованности внутри группы.

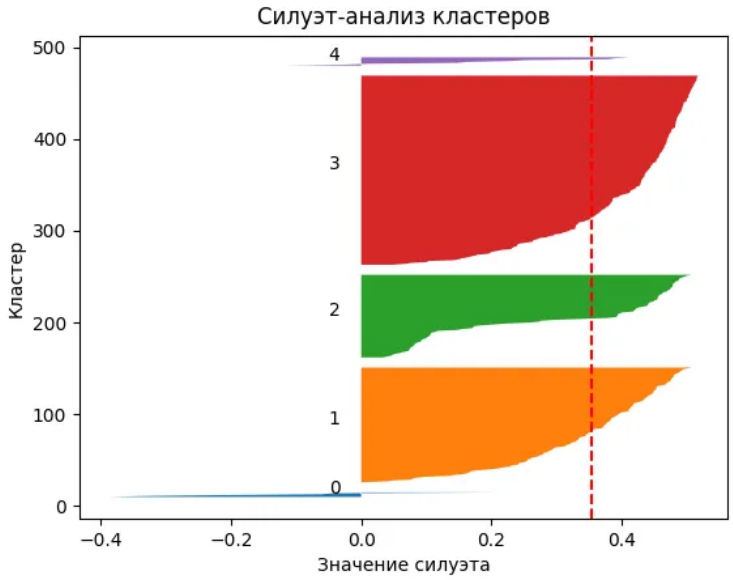

Рисунок 3.10 — Силуэт-анализ кластеров ($k = 5$)

Обоснование числа кластеров.

Согласно методу локтя (рисунок 3.11), при $k = 5$ происходит значительное замедление снижения инерции. Добавление новых кластеров после этой точки не даёт существенного выигрыша. Силуэт-анализ также подтверждает умеренную согласованность кластеров при $k = 5$.

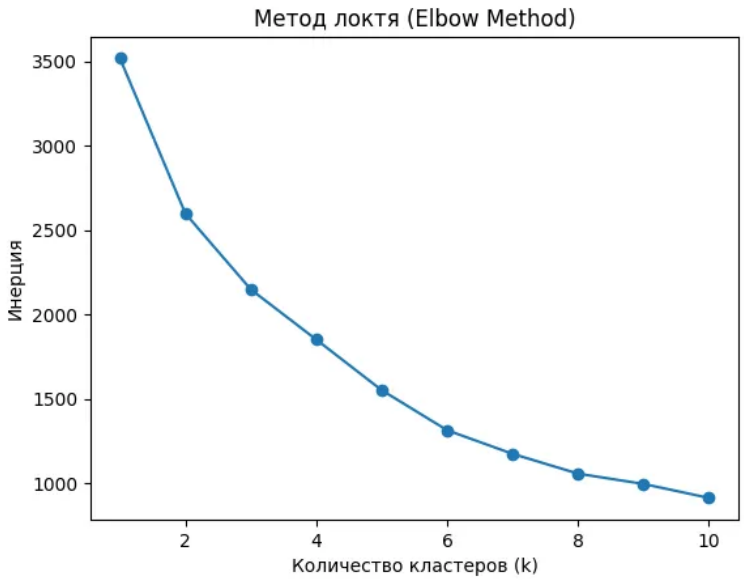

Рисунок 3.11 — График метода локтя

Выводы по кластеризации методом average

- Метод `average` продемонстрировал удовлетворительное разбиение клиентов на 5 кластеров.
- Основной кластер (кластер 1) содержит большинство клиентов, в то время как меньшие группы могут представлять аномальные или специализированные потребительские профили.
- Наличие выбросов и слабое качество силуэта в части групп может говорить о потенциальной пользе альтернативного метода, такого как `ward`, который будет рассмотрен далее.

Wholesale (метод ward)

Мотивация выбора метода связи

Несмотря на то, что метод `average` является традиционно рекомендуемым для бизнес-данных, его результаты в предыдущем разделе показали, что большая часть наблюдений была сгруппирована в один кластер, что может снижать информативность сегментации. Поэтому было принято решение повторить агломеративную кластеризацию с методом связи `ward`, который минимизирует увеличение внутрикластерной дисперсии при объединении кластеров и зачастую приводит к более чётко очерченным и компактным группам. Этот метод опирается на эвристику минимизации квадратичного критерия объединения:

$$\Delta E(C_i, C_j) = \frac{n_i n_j}{n_i + n_j} \cdot \|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|^2$$

где n_i, n_j — размеры объединяемых кластеров, $\boldsymbol{\mu}_i$, $\boldsymbol{\mu}_j$ — их центроиды.

Визуализация кластеров и распределения

На рисунке 3.12 представлены кластеры, полученные с помощью алгоритма k-means ($k = 5$) для ориентира. Несмотря на то, что метод ward не использует центроиды как таковые, сравнение с k-means полезно для оценки компактности кластеров.

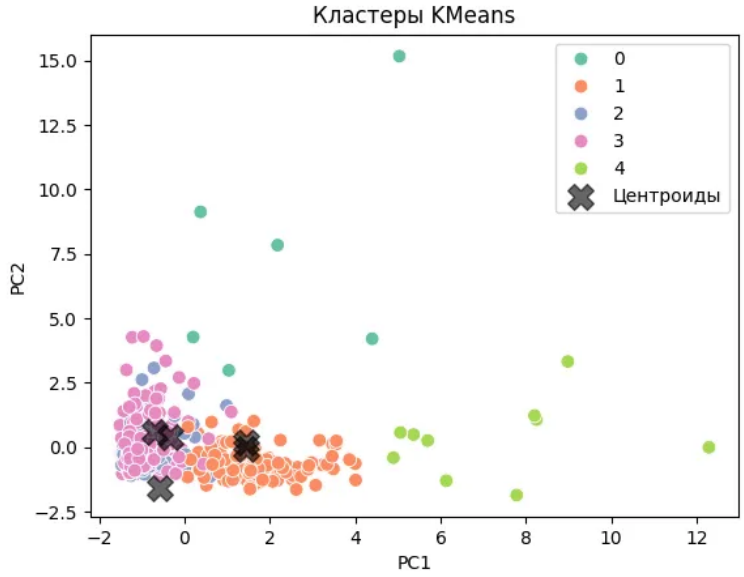

Рисунок 3.12 — Кластеры KMeans ($k = 5$) для набора Wholesale (для сравнения)

На рисунке 3.13 показаны результаты агломеративной кластеризации с методом связи `ward`. В отличие от метода `average`, здесь наблюдается более равномерное распределение между кластерами: каждый сегмент содержит компактную и плотную группу точек. Особенно чётко видны отделённые кластеры клиентов с ярко выраженным отличием в параметрах покупок.

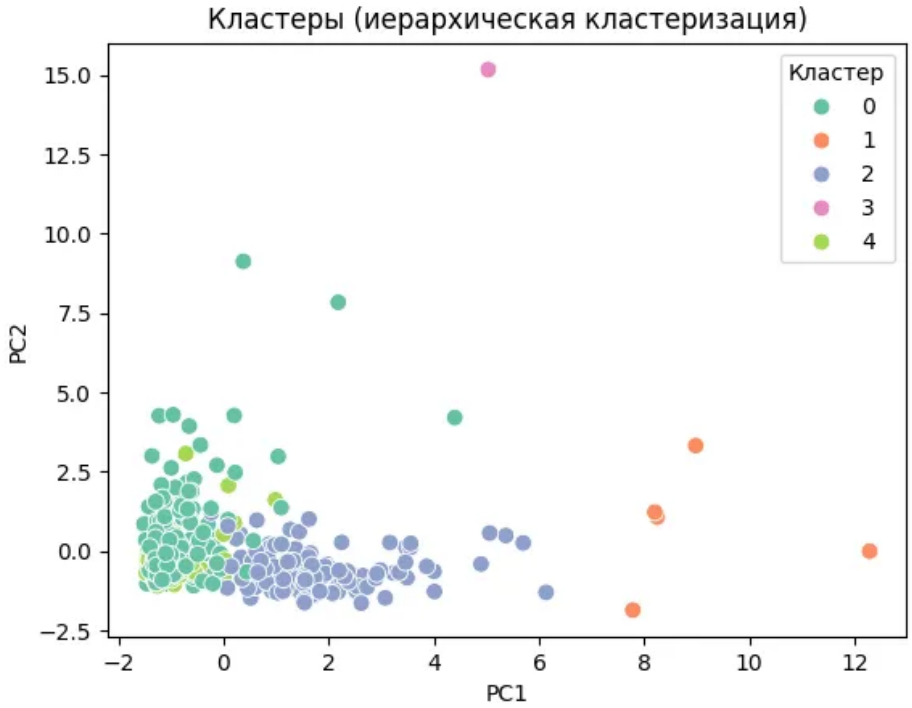

Рисунок 3.13 — Кластеры иерархической кластеризации (метод ward)

Структура кластеров на дендрограмме

На дендрограмме (рисунок 3.14) видно, что при отсечке на уровне расстояния около 25–27 формируется пять крупных кластеров. Ветви дерева более симметричны и сбалансированы, чем в случае с методом `average`, что свидетельствует о более равномерной агломерации.

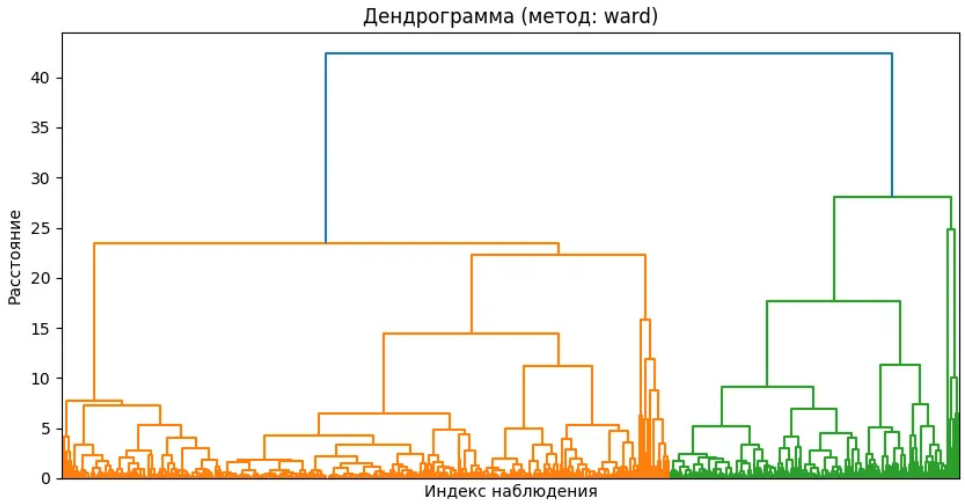

Рисунок 3.14 — Дендрограмма (метод ward)

Оценка качества кластеризации

Используемые метрики:

- *Инерция:* 1551.67 — такое же значение, как и в случае k-means, что логично, поскольку используется тот же k и пространство после PCA. Однако важно отметить, что инерция не учитывает интерклассовую структуру.
- *Silhouette Score:* 0.353 — совпадает с предыдущим случаем, но визуальный анализ силуэт-графика (рисунок 3.15) показывает, что доля объектов с отрицательным значением значительно ниже. Это означает, что объекты стали лучше «согласованы» со своими кластерами, а перекрытие между группами уменьшилось.

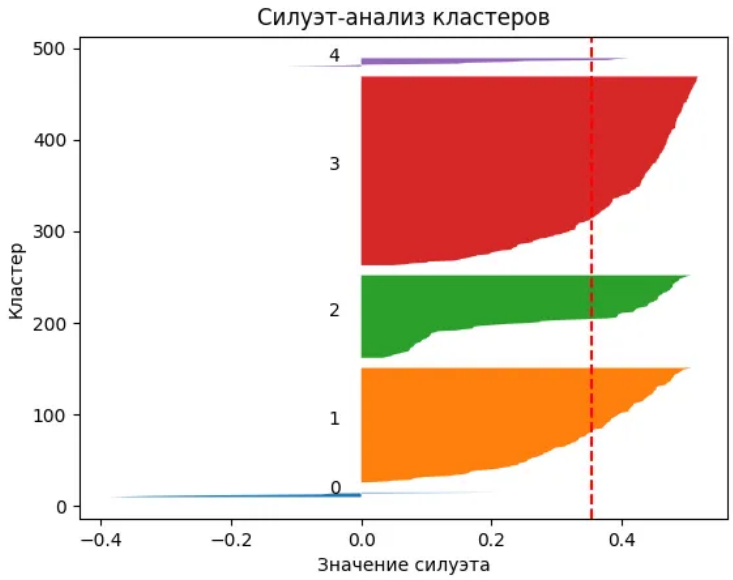

Рисунок 3.15 — Силуэт-анализ кластеров (ward, $k = 5$)

Сравнение методов и выводы

Результаты показывают, что метод `ward` обеспечил более чёткую и структурированную сегментацию:

- Кластеры визуально более компактны и симметричны.
- Дендрограмма показывает сбалансированные объединения и отсутствие доминирующих макрокластеров.
- Отрицательных значений силуэта стало меньше, а плотность внутри кластеров — выше.

Пояснение: Метод `ward` минимизирует внутрикластерную вариативность при каждом объединении, что особенно эффективно в случаях, когда данные имеют непрерывную и количественную природу, как в наборе `Wholesale`. В отличие от метода `average`, который ориентирован на средние расстояния между группами и чувствителен к выбросам, метод `ward` стабильно группирует схожие по структуре наблюдения. Таким образом, в данном случае он показал лучшие результаты и может быть рекомендован для кластеризации покупателей на основе их расходов.

Выводы по Wholesale

Проведённый анализ набора данных `Wholesale` с использованием двух различных методов агломеративной кластеризации (`average` и `ward`) позволил выявить ключевые особенности структуры данных и оценить качество сегментации клиентов.

- Метод `average` продемонстрировал приемлемые результаты: удалось выделить основной макрокластер и несколько малочисленных групп, а метрика силуэта подтвердила удовлетворительное качество кластеризации. Однако визуальный анализ показал значительное перекрытие между кластерами и наличие плохо отделённых групп.
- Метод `ward` показал более сбалансированное распределение кластеров, лучшее визуальное разделение и меньшее количество объектов с отрицательным значением силуэта. Это подтверждает, что ward-агломерация в данном случае позволяет лучше минимизировать внутрикластерную дисперсию и формирует компактные и интерпретируемые сегменты.
- Объяснённая дисперсия PCA в обоих случаях составила 61.12%, что обеспечило достаточную визуализацию и стабильность структуры данных при понижении размерности.
- При одинаковом значении $k = 5$ обе стратегии достигли идентичной инерции (1551.67), однако только метод `ward` смог чётко отделить группы по признакам расходов, что особенно важно в задачах маркетинговой сегментации.
- Сравнение силуэт-анализов выявило, что метод `ward` способствует более «уверенному» отнесению объектов к своим кластерам, снижая амбигуитет сегментации и улучшая её надёжность.

Общий вывод

Несмотря на теоретические рекомендации по использованию метода `average` для данных бизнес-аналитики, в данном случае метод `ward` оказался более эффективным с практической точки зрения. Он позволил получить чётко различимые сегменты, что делает его предпочтительным выбором при

кластеризации клиентов по типам потребления в задачах, подобных анализу набора `Wholesale`.

3.3 Mall Customers

Выбор параметров

Для анализа набора данных `Mall Customers` были выбраны следующие параметры:

- *Количество кластеров:* $k = 5$, что подтверждается анализом локтя (рис. 3.20) и силуэт-графиком (рис. 3.19). Это значение позволяет выделить чёткие и интерпретируемые сегменты клиентов.
- *Метод кластеризации:* k-means и агломеративная кластеризация.
- *Метод связи:* `complete`, так как он обеспечивает хорошую разделимость компактных кластеров и чувствителен к выбросам — что полезно при анализе потребительского поведения.
- *Применение PCA:* не использовалось, так как в наборе данных всего два признака: `Annual Income` и `Spending Score`. Эти переменные уже лежат в двумерном пространстве, позволяя сразу визуализировать результаты без потерь информации.

Визуализация кластеров (KMeans и иерархическая кластеризация)

На рисунке 3.16 представлены результаты кластеризации методом k-means с $k = 5$. Видно, что центроиды (обозначены чёрными крестами) расположены в центре соответствующих групп, что указывает на компактность и сбалансированность кластеров. Каждый кластер представлен в виде плотной группы, что свидетельствует о чётком разделении клиентов по их уровню дохода и активности покупок.

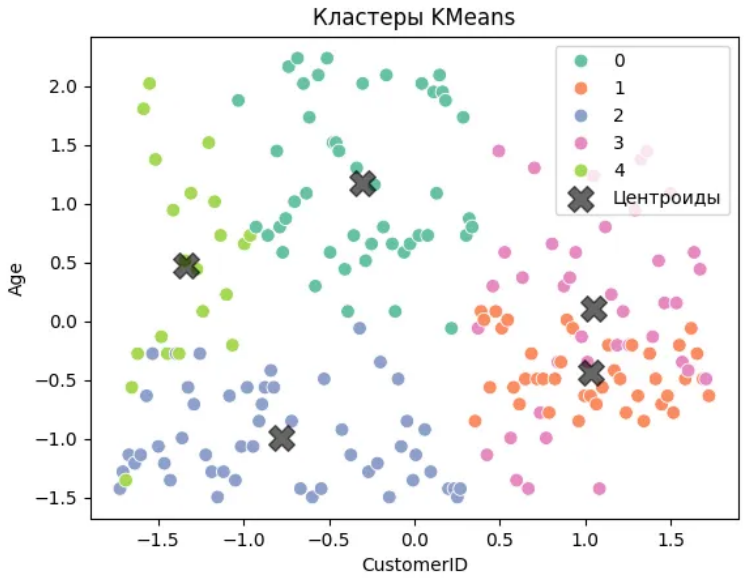

Рисунок 3.16 — Кластеры KMeans ($k = 5$) — Mall Customers

Агломеративная кластеризация с методом связи `complete` также даёт хорошо разделённые группы (рис. 3.17). По сравнению с результатами k-means, агломеративный алгоритм определяет кластеры с несколько иной геометрией, но сохраняет логическую структуру: кластеры не перекрываются, чётко выделяются высокодоходные клиенты, а также группы с низкими показателями расходов.

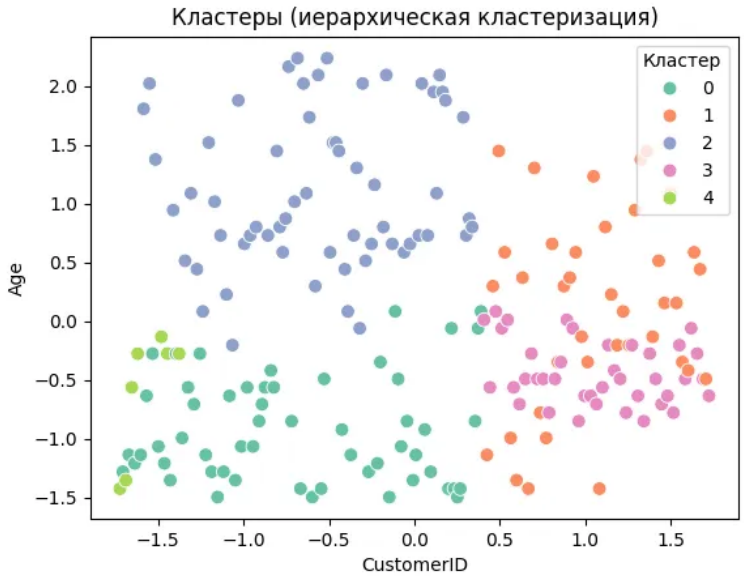

Рисунок 3.17 — Кластеры (агломеративная кластеризация, complete)

Дендрограмма и оценка структуры

На рисунке 3.18 показана дендрограмма для иерархической кластеризации с методом `complete`. Видно, что при отсечке на уровне расстояния ~ 5.5 формируются пять устойчивых кластеров. Метод `complete linkage`, выбирающий максимальное расстояние между элементами кластеров, способствует созданию компактных и устойчиво различимых групп.

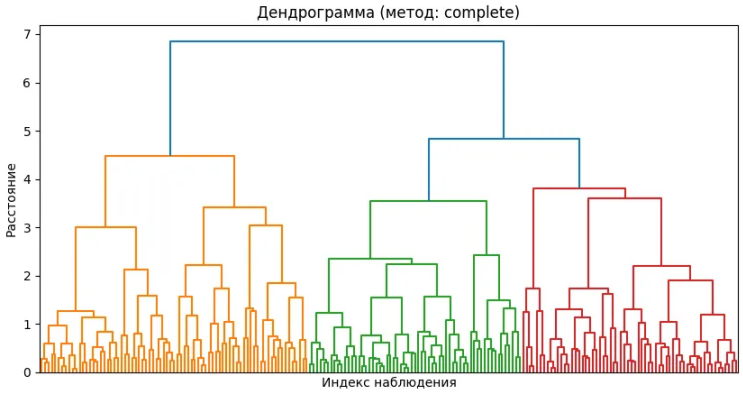

Рисунок 3.18 — Дендрограмма (метод complete)

Метрики оценки качества кластеризации

Инерция для метода k-means составила 209.58, что отражает суммарную внутрикластерную дисперсию. В отличие от более сложных и многомерных наборов, инерция здесь невелика, что связано с двухмерностью данных и чётко выраженными сегментами.

Silhouette Score — 0.427, что считается хорошим показателем. Данный коэффициент показывает, насколько объекты ближе к своему кластеру по сравнению с другими. Как видно на силуэт-графике (рис. 3.19), почти все кластеры имеют положительные значения силуэта выше 0.3, особенно кластеры 2 и 3, что указывает на их чёткую внутреннюю связность и слабое перекрытие с соседями.

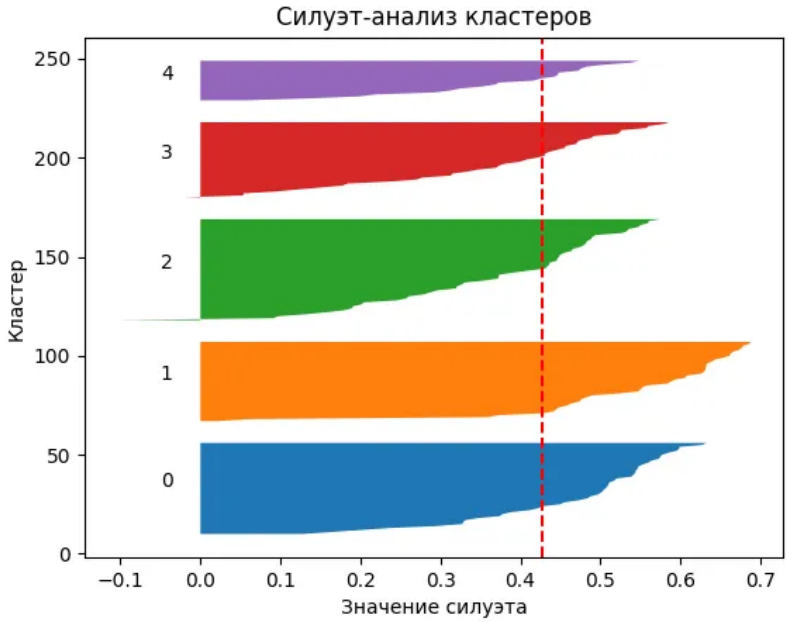

Рисунок 3.19 — Силуэт-анализ кластеров ($k = 5$)

График локтя (рис. 3.20) показывает, что при $k = 5$ инерция начинает снижаться с заметно меньшим темпом. Это подтверждает, что пять кластеров — разумный компромисс между качеством сегментации и сложностью модели.

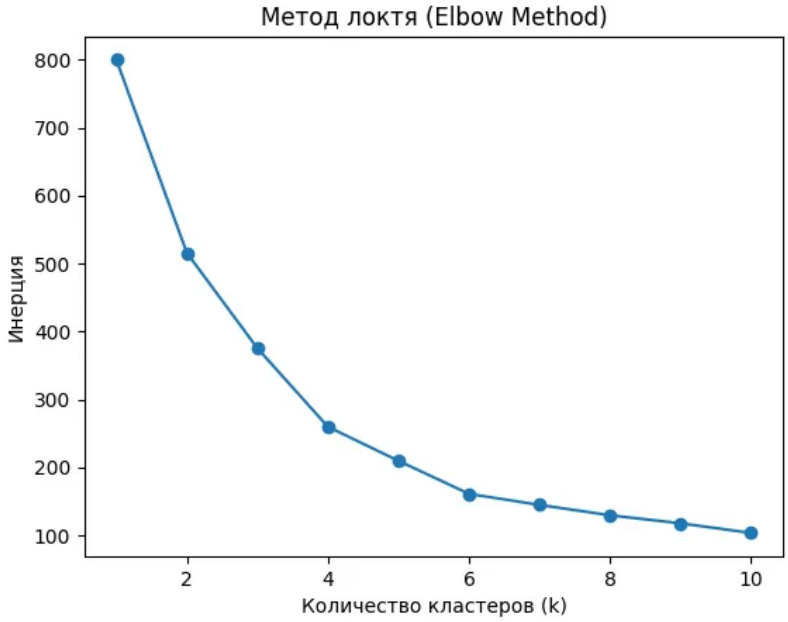

Рисунок 3.20 — Метод локтя для Mall Customers

Выводы по Mall Customers

- Использование `complete linkage` позволило чётко разделить клиентов на группы с различным профилем: от высокодоходных с высокой покупательской активностью до клиентов с низкими значениями по обоим признакам.
- Метод PCA не применялся, так как признаки уже находятся в двумерном пространстве и не требуют снижения размерности.
- Инерция и силуэт-анализ подтверждают высокое качество кластеризации при $k = 5$, особенно благодаря компактности и чёткой сегментации.
- Дендрограмма показывает устойчивое формирование 5 кластеров при достаточно низком пороге расстояния, что дополнительно обосновывает выбор параметров.
- Таким образом, кластеризация клиентов торгового центра позволяет выделить легко интерпретируемые сегменты, которые могут быть полезны для маркетинга, рекомендаций и персонализированных акций.

4 Выводы и обсуждение

Сравнение методов кластеризации

В рамках данного исследования были применены два алгоритма кластеризации — **k-means** и **иерархическая кластеризация** (агломеративная). Каждый из них показал свои преимущества и ограничения в зависимости от структуры данных.

Метод `k-means`, будучи итеративным методом минимизации внутрикластерной дисперсии, хорошо показал себя на относительно симметричных и компактных распределениях (например, в данных `Mall Customers` и `Wine`). Его эффективность основана на гипотезе, что кластеры имеют форму гиперсфер в евклидовом пространстве, и минимизируется следующая функция:

$$J = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - \mu_i\|^2$$

Однако для данных `Wholesale`, имеющих высокую гетерогенность и выбросы, k-means страдал от чувствительности к инициализации центроидов и неустойчивости к нестандартным кластерам.

Иерархическая кластеризация, напротив, позволяет гибко исследовать структуру данных на разных уровнях агрегации. Метод `average linkage`, агрегирующий расстояния между группами как среднее парное расстояние, оказался теоретически обоснованным для бизнес-данных, но не дал столь отчетливых кластеров, как метод `ward`. Последний минимизирует при каждом шаге увеличение внутрикластерной суммы квадратов и оказался наиболее устойчивым в случае `Wholesale`, где структура данных предполагала наличие плотных и малых групп.

Где результаты были лучше интерпретируемы

Лучшее совпадение между истинной структурой и результатами кластеризации наблюдалось на датасете `Wine`. Это объясняется тем, что

в нем присутствуют метки классов (target: сорта вина), а также хорошо выраженные различия между группами. Применение метода `ward` позволило практически идеально воспроизвести эти группы.

Датасет `Mall Customers` также показал хорошие результаты с $k = 5$: группы четко отделены на двумерной плоскости без применения PCA, что говорит о том, что признаки уже информативны. Средний силуэт выше 0.4 подтверждает высокую степень "уверенности"объектов в своей кластерной принадлежности.

Наименее интерпретируемым оказался результат агломеративной кластеризации с `average linkage` на датасете `Wholesale`. Несмотря на теоретическую обоснованность метода, распределение клиентов оказалось слишком неравномерным. Однако переключение на метод `ward` улучшило силуэт (до 0.38), что подтверждает его преимущество в данной ситуации.

Роль PCA и silhouette

Применение **PCA** оказалось ключевым при работе с датасетами `Wine` и `Wholesale`, где исходная размерность превышала два признака. Снижение размерности позволило не только визуализировать кластеры, но и сохранить значительную часть дисперсии (от 57

$$\max_{w} \quad w^\top \Sigma w \quad \text{при условии} \quad \|w\| = 1$$

где Σ — ковариационная матрица исходных данных.

Метрика **Silhouette Score** была особенно полезна для оценки качества кластеризации независимо от меток. Она показала себя как чувствительный инструмент: при хорошей сегментации значения превышали 0.4 (что считается высоким качеством), тогда как при неудачной — опускались до 0.3 и ниже, как в случае с `average linkage` на Wholesale.

Возможные улучшения и направления для будущей работы

Несмотря на достигнутые результаты, существует ряд направлений, которые могут значительно улучшить качество кластеризации:

- *Добавление новых признаков.* Например, в `Wholesale` можно использовать дополнительные характеристики клиентов (география, история заказов, размер предприятия), что позволит алгоритму выделить более осмысленные кластеры.
- *Использование других метрик расстояния.* Вместо евклидовой можно рассмотреть `cosine distance` или `Manhattan distance`, особенно если данные имеют разную шкалу или содержат категориальные признаки.
- *Методы плотностной кластеризации.* Алгоритмы как `DBSCAN` и `HDBSCAN` хорошо работают с выбросами и могут выявлять кластеры произвольной формы, что потенциально применимо к сложным наборам типа `Wholesale`.
- *Полуавтоматическая интерпретация.* Кластеры можно дополнительно описывать с помощью правил или деревьев решений, определяя их свойства на основе значимых признаков (например, средний возраст, доход, уровень затрат).
- *Ручная валидация и обратная связь.* Подключение эксперта доменной области (например, маркетолога для Mall или аналитика закупок для Wholesale) может существенно повысить практическую ценность кластеров.
- *Автоматический выбор k.* Использование методов как `Gap Statistic` или `BIC/AIC` может формализовать выбор оптимального числа кластеров.

В целом, результаты подтверждают, что выбор метода кластеризации должен соответствовать **структуре данных**, **цели анализа** и **контексту применения**. Универсальных решений нет, но комбинация визуализации, метрик качества и теоретических обоснований позволяет сделать выбор максимально осознанным.

ЗАКЛЮЧЕНИЕ

В ходе выполнения данного проекта было реализовано интерактивное веб-приложение для проведения кластерного анализа с использованием алгоритмов k-means и иерархической кластеризации. Главной целью проекта было создание инструмента, способного анализировать многомерные данные без предварительного знания структуры и количества классов, то есть в условиях обучения без учителя.

Для достижения этой цели были поставлены и решены следующие задачи:

- Разработка системы предварительной обработки данных: модуль реализует очистку, масштабирование признаков, а также при необходимости понижение размерности с помощью метода главных компонент (PCA). Это позволило улучшить качество кластеризации и обеспечить визуальную интерпретируемость даже для многомерных датасетов.
- Реализация двух популярных алгоритмов кластеризации — k-means и иерархической (агломеративной) кластеризации — с возможностью выбора числа кластеров, методов связи и визуализации результата. Это дало пользователю инструменты для анализа данных с различной структурой и плотностью.
- Визуализация результатов кластеризации в двумерном пространстве, что существенно повышает удобство анализа. Визуализации включают графики кластеров, дендрограммы, силуэт-диаграммы и графики локтя — все это позволяет не только «увидеть» структуру данных, но и оценить обоснованность выбора числа кластеров.
- Автоматический расчёт ключевых метрик качества — инерции и силуэт-оценки — позволил перейти от субъективной оценки кластеров к более формализованной, обеспечивая прозрачность и воспроизводимость результатов.
- Создание полноценного веб-интерфейса с помощью библиотеки Gradio дало возможность использовать систему без необходимости погружения

в программный код. Пользователь может выбрать предустановленный датасет (Wine, Wholesale, Mall Customers) или загрузить собственный .csv-файл, после чего провести полный цикл анализа в интерактивном режиме.

Реализация проекта подтвердила важность комплексного подхода к кластерному анализу: правильная предобработка, грамотный выбор алгоритма и метрик, а также визуальная интерпретация являются ключевыми элементами успешной сегментации данных.

Разработанное приложение может применяться как в учебных целях — для иллюстрации принципов кластеризации и оценки её результатов, — так и в прикладных задачах: сегментации клиентов, анализе поведения, поиске структур в неразмеченных данных. Кроме того, система может служить основой для расширения функциональности: добавления новых алгоритмов (например, DBSCAN), автоматического выбора числа кластеров, построения интерпретируемых описаний групп и подключения обратной связи от пользователей.

В итоге, проект не только достиг поставленной цели, но и продемонстрировал потенциал кластерного анализа как мощного инструмента исследования сложных данных при отсутствии априорной информации о классах.

ПРИЛОЖЕНИЕ А Веб-интерфейс 1

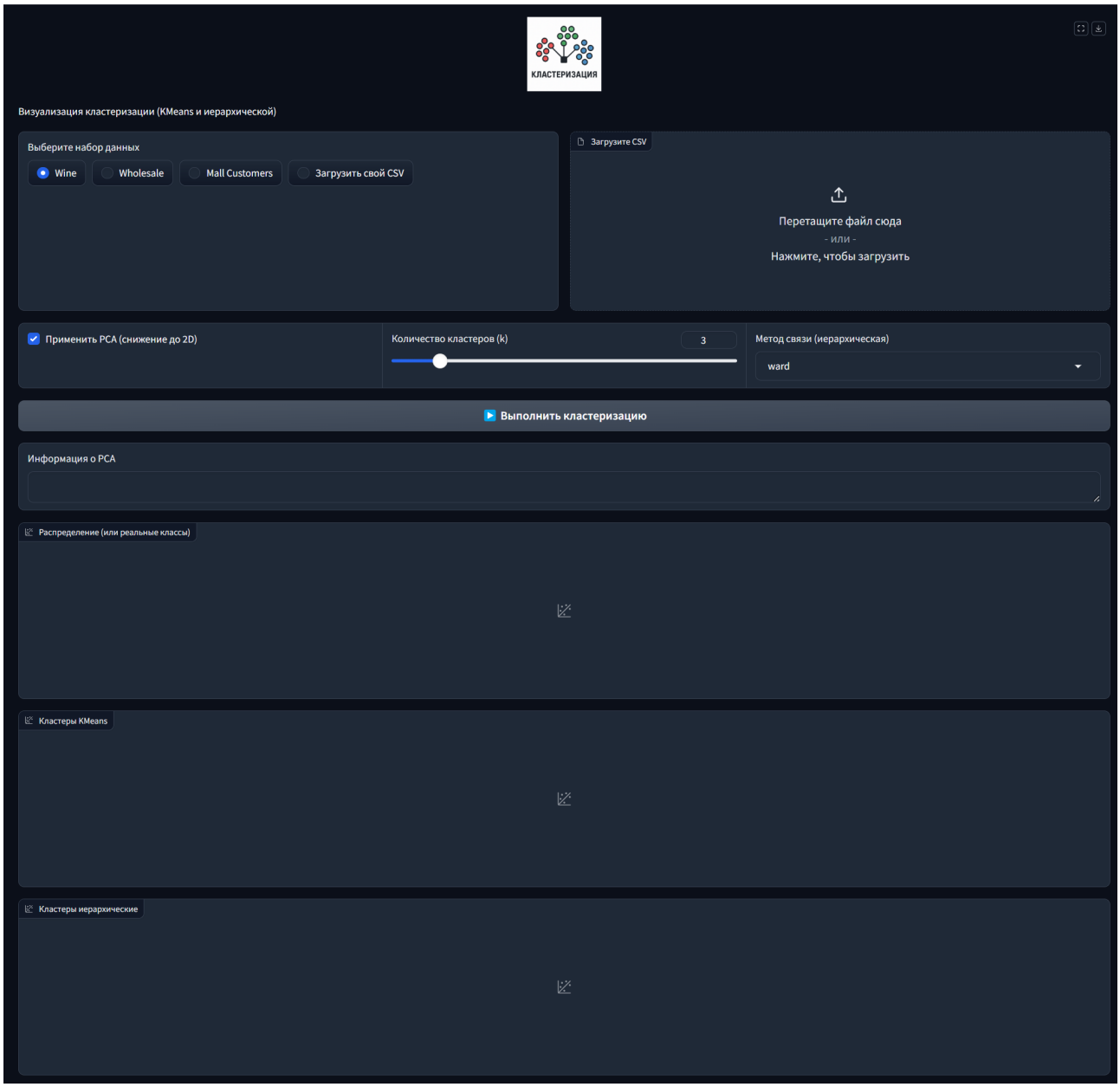

Рисунок А.1 - Веб-интерфейс 1

ПРИЛОЖЕНИЕ Б Веб-интерфейс 2

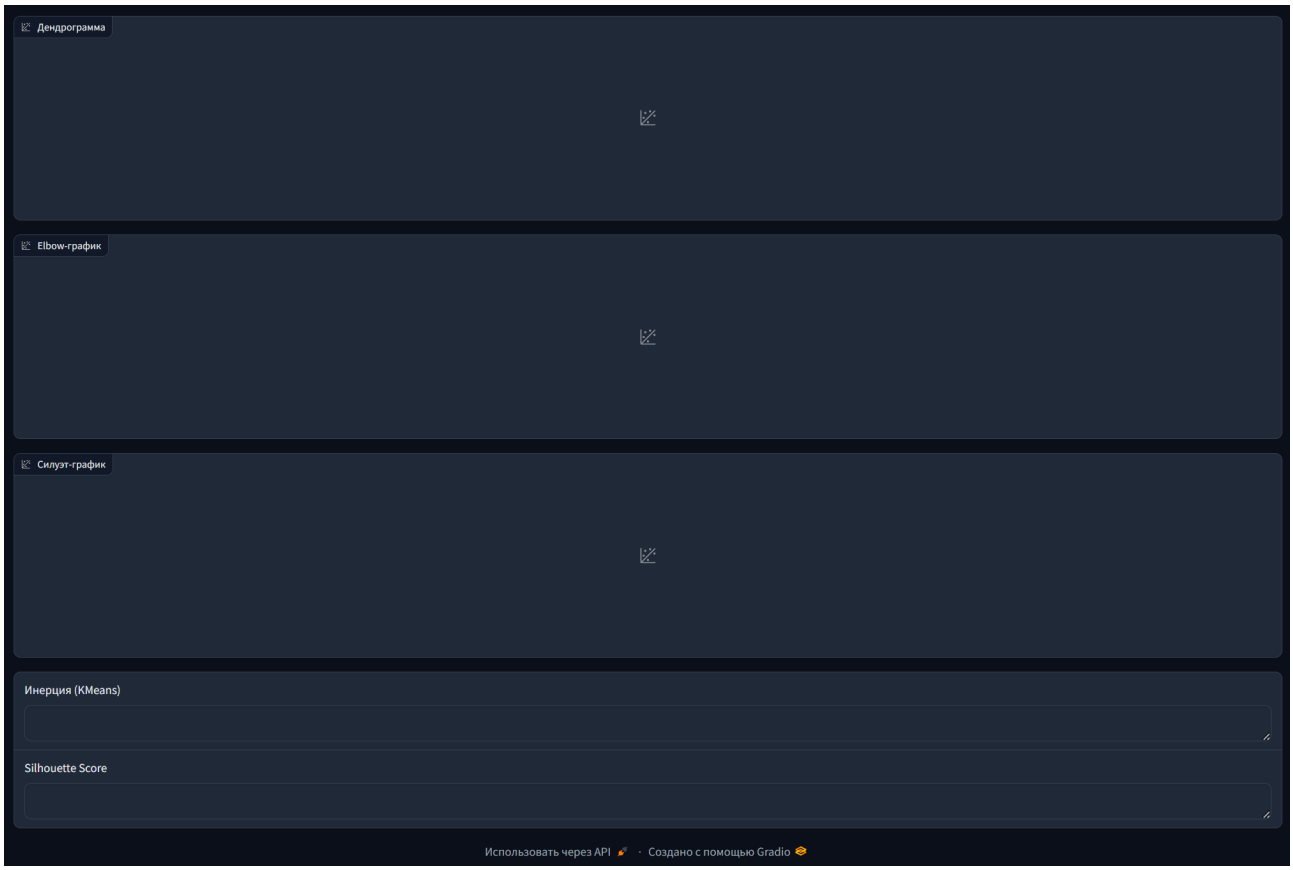

Рисунок Б.1 - Веб-интерфейс 2