САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет прикладной математики-процессов управления

Программа бакалавриата
“Большие данные и распределенная цифровая платформа”

ОТЧЕТ

по лабораторной работе №5

по дисциплине «Алгоритмы и структуры данных»

на тему «Кластеризация данных»

Студент гр. 23Б15-пу
Трофимов И. А.

Преподаватель
Дик А.Г.

Санкт-Петербург

2025 г.

Оглавление

Цель работы

Целью данной лабораторной работы является разработка и реализация программного средства для кластеризации табличных данных с использованием односвязного метода и метрики корреляции Пирсона, а также проведение отбора наиболее информативных признаков методом последовательного добавления (Add), реализация процедуры анонимизации данных путём случайного перемешивания признаков и визуализация результатов кластеризации с помощью метода главных компонент (PCA) с экспериментальным сравнением качества кластеризации по индексу Rand на различных этапах обработки данных.

Теоретическая часть

В современных задачах анализа данных важную роль играет проблема выявления и структурирования скрытых закономерностей в больших объемах информации. Одним из ключевых инструментов для решения этой задачи является кластеризация — метод группирования объектов в однородные по структуре группы (кластеры) без использования априорной информации о принадлежности к классам.

Кластеризация применяется для анализа структуры данных, поиска аномалий, сокращения размерности и предварительной обработки перед построением других моделей. Существует множество методов кластеризации, различающихся принципом объединения объектов и используемыми метриками расстояния. Одним из базовых подходов является иерархическая кластеризация, в частности, односвязный (single linkage) метод, при котором расстояние между кластерами определяется как минимальное расстояние между объектами из разных групп. В качестве меры близости объектов в данной работе применяется расстояние, основанное на корреляции Пирсона, что позволяет учитывать линейные взаимосвязи между признаками.

Для повышения качества кластеризации и уменьшения влияния нерелевантных или избыточных признаков часто используется этап отбора признаков. Метод последовательного добавления признаков (Add) предполагает жадный перебор: на каждом шаге к текущему набору добавляется такой признак, который в наибольшей степени повышает качество разбиения данных на кластеры по выбранной метрике.

Отдельное значение в современных исследованиях приобретает задача анонимизации и обезличивания данных. Одним из простых способов является случайное перемешивание порядка признаков, что затрудняет восстановление исходной структуры и защищает приватность информации.

Для визуальной интерпретации результатов кластеризации и проверки ее успешности в работе применяется метод главных компонент (PCA), позволяющий проецировать многомерные данные на двумерное пространство с сохранением максимального разброса, что облегчает анализ полученных кластеров.

Оценка качества кластеризации проводится с помощью индекса Rand, который отражает степень совпадения полученных кластеров с истинным разбиением и позволяет объективно сравнивать различные варианты обработки и отбора признаков.

Описание задачи

В данной работе рассматривается задача кластеризации большого табличного датасета, состоящего преимущественно из числовых признаков, на примере набора данных Covertype. В процессе эксперимента реализуется процедура разбиения объектов на кластеры с использованием односвязного метода и метрики корреляции Пирсона, а также выполняется отбор наиболее информативных признаков методом последовательного добавления (Add). Для анализа устойчивости результатов проводится анонимизация данных

путём случайного перемешивания признаков. Качество кластеризации на различных этапах оценивается с помощью индекса Rand, а полученные кластеры визуализируются методом главных компонент (PCA). Эксперимент проводится на выборках различной размерности и при различных наборах признаков, что позволяет оценить влияние отбора признаков и анонимизации на структуру данных и качество кластеризации.

Основные шаги

1. Выбрать исходный табличный датасет для проведения эксперимента
2. Выполнить предварительную обработку данных: стандартизацию и сокращение числа признаков
3. Провести кластеризацию исходных данных односвязным методом с использованием метрики корреляции Пирсона
4. Осуществить отбор наиболее информативных признаков методом последовательного добавления (Add)
5. Провести кластеризацию по отобранным признакам
6. Выполнить анонимизацию данных путём случайного перемешивания признаков
7. Провести кластеризацию анонимизированных данных
8. Визуализировать результаты кластеризации методом главных компонент (PCA)
9. Оценить качество кластеризации на каждом этапе по индексу Rand
10. Проанализировать и сравнить полученные результаты

Описание программы

Программная реализация выполнена на языке Python версии 3.13.0 с использованием библиотек pandas[1], numpy[2], matplotlib[3], scikit-learn[4] и

PyQt5[5] для построения графического интерфейса пользователя. Приложение представляет собой GUI-программу для проведения кластерного анализа и отбора признаков на табличных данных. Пользователь может выбрать число признаков для анализа, запускать кластеризацию исходных и анонимизированных данных односвязным методом с использованием корреляции Пирсона, а также выполнять автоматический отбор информативных признаков методом последовательного добавления (Add). В программе реализована функция анонимизации путём случайного перемешивания признаков. Интерфейс приложения отображает историю выполненных действий, результаты кластеризации (значения индекса Rand), а также обеспечивает визуализацию структуры данных с помощью метода главных компонент (PCA). Программа предназначена для учебных целей, позволяет проводить экспериментальное сравнение качества кластеризации на разных этапах обработки данных и способствует развитию практических навыков работы с методами анализа и визуализации многомерных данных. В процессе разработки программы использовался следующий модуль:

Таблица1 main.py

Функция	Описание	Возвращаемое значение
single_linkage_clustering()	Выполняет иерархическую кластеризацию односвязным методом по матрице расстояний	Вектор меток кластеров для каждого объекта

pearson_distance()	Вычисляет расстояние между двумя объектами на основе корреляции Пирсона	Число от 0 до 2, отражающее степень различия двух объектов
evaluate_rand()	Оценивает качество кластеризации по индексу Rand	Значение индекса Rand (от 0 до 1)
select_features_additive()	Выполняет последовательный жадный отбор наиболее информативных признаков (метод Add)	Список индексов выбранных признаков
shuffle_features()	Перемешивает порядок признаков (столбцов) в матрице данных для анонимизации	Новая матрица признаков с перемешанными столбцами
visualize()	Выполняет понижение размерности данных методом главных компонент (PCA) и строит график	График (scatter plot) кластеров на двух главных компонентах

Рекомендации пользователя

Перед запуском программы убедитесь, что на вашем компьютере установлены Python и необходимые библиотеки: pandas, numpy, matplotlib, scikit-learn и PyQt5. Запустите основной файл программы (например, main.py). В графическом интерфейсе выберите количество признаков для

анализа с помощью соответствующего поля, затем используйте кнопки для запуска кластеризации, отбора признаков и анонимизации данных. Все выполненные действия и результаты отображаются в области истории событий, а визуализация кластеров доступна по нажатию соответствующей кнопки. Рекомендуется выполнять отбор признаков на небольших подвыборках для ускорения работы программы.

Рекомендации программиста

1. Используйте актуальные версии библиотек PyQt5, pandas, numpy, matplotlib и scikit-learn.
2. Тестируйте работу программы на выборках разного объёма и с различным количеством признаков для оценки производительности и корректности работы алгоритмов.
3. При необходимости расширяйте функциональность приложения, добавляя новые методы кластеризации или варианты отбора признаков в отдельные функции.
4. Для быстрой проверки работоспособности интерфейса и основных алгоритмов рекомендуется использовать небольшие подвыборки данных.
5. Обеспечьте информативные сообщения для пользователя и корректную обработку возможных ошибок во всех сценариях использования GUI.

Исходный код программы

Гитхаб

Контрольный пример

1. **Запуск программы**

   Для запуска программы используйте файл main.py. Программа загружает GUI. (Рис. 1)

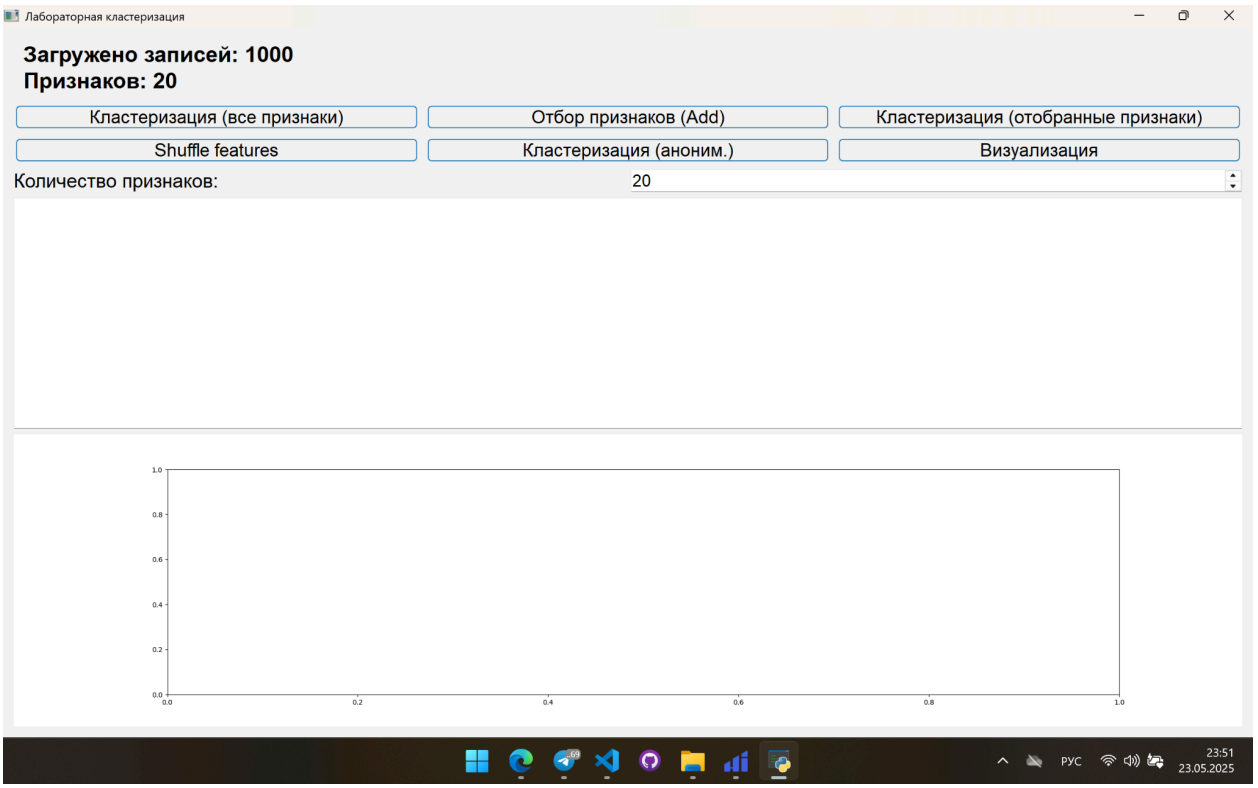

Рис 1. Окно GUI

2. Выбор параметров

Датасет загружается автоматически, можно лишь поменять количество параметров. (Рис. 2)

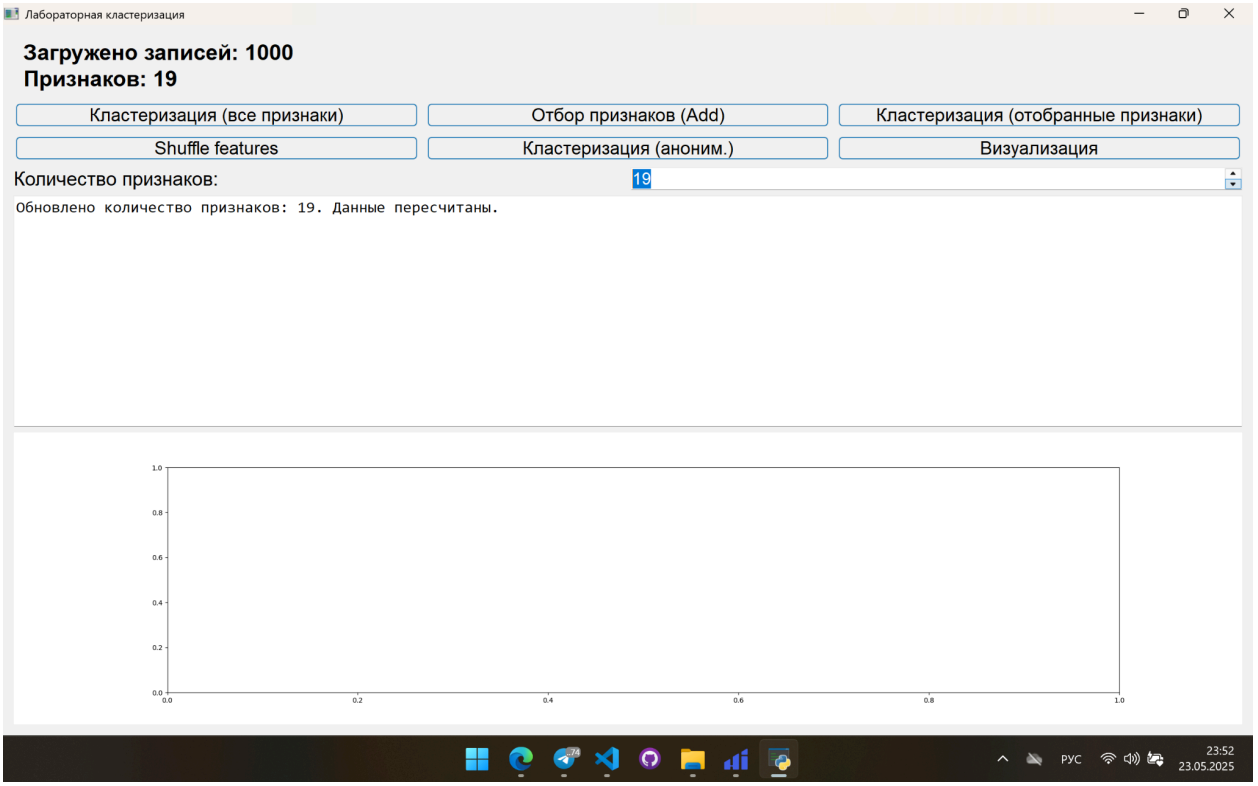

Рис 2. Настройки признаков

3. **Обработка графа и вывод результатов**

Запускаем по очереди все методы. (Рис. 3)

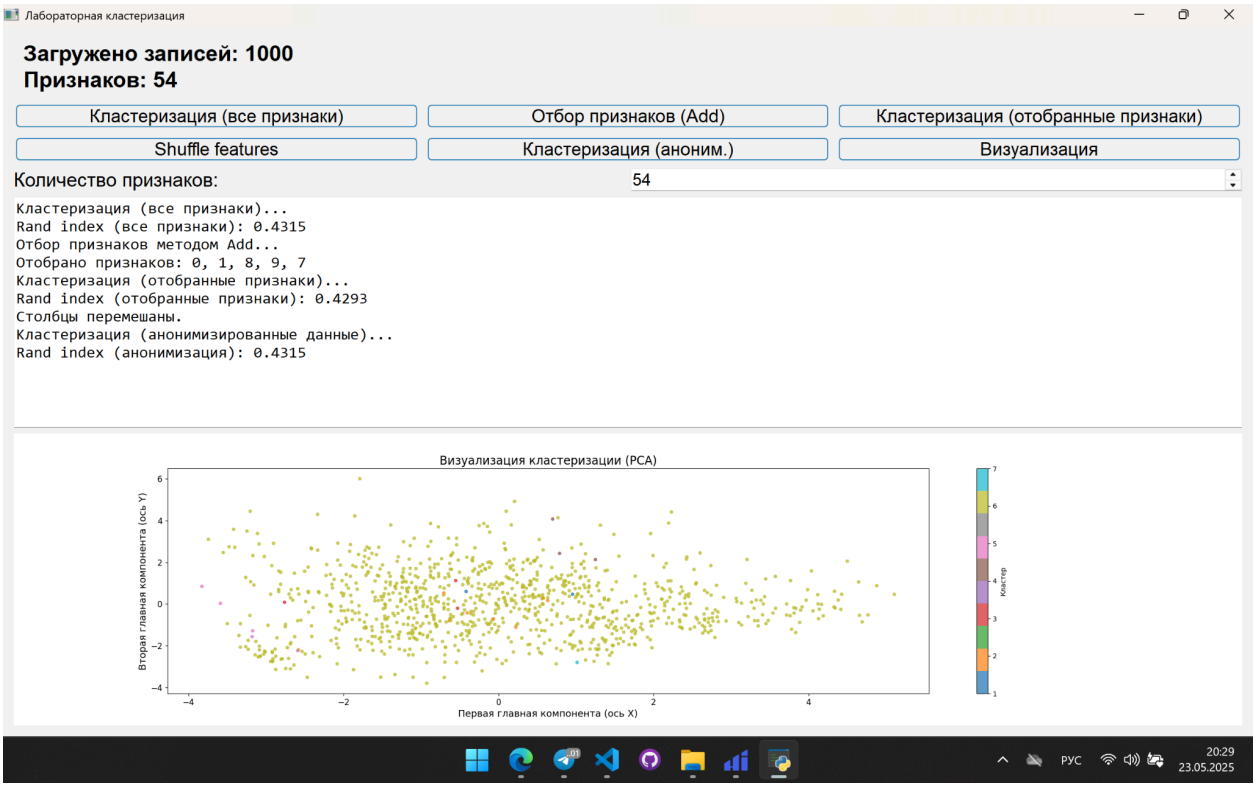

Рис 3. Результат работы

Анализ

В ходе эксперимента было проведено сравнение результатов кластеризации на разных этапах обработки данных и при различных размерах выборки и количестве признаков. При использовании полного набора признаков и большого числа объектов (например, 1000 строк и 54 признака) наблюдается сравнительно низкое качество кластеризации, выражающееся в невысоком значении индекса Rand (около 0.43) и визуально однородной структурой кластеров на графике PCA. При этом перемешивание столбцов (анонимизация) практически не влияет на итоговый результат, что свидетельствует о слабой информативности части исходных признаков и высокой степени их коррелированности.

С уменьшением объёма выборки и сокращением числа признаков эффективность кластеризации с помощью отобранных методом Add признаков значительно возрастает: индекс Rand может достигать значений выше 0.7, а на графике PCA формируются более чёткие и отделённые кластеры. На малых выборках (например, 100 строк) разница между исходной кластеризацией и кластеризацией по отобранным признакам становится особенно заметной, что подтверждает важность этапа отбора признаков для повышения качества разбиения данных.

Анализ времени работы программы показывает, что основной вклад в замедление вносит процедура отбора признаков методом Add. На каждом шаге этот метод выполняет перебор всех оставшихся признаков с последующей кластеризацией на каждом новом подмножестве, что приводит к многократному пересчёту матрицы попарных расстояний между объектами. Особенно ресурсоёмкими оказываются операции иерархической кластеризации и вычисления расстояний по метрике корреляции Пирсона, поскольку вычисления выполняются на каждой итерации для всего поднабора данных. В результате время работы программы экспоненциально возрастает с увеличением размера выборки и числа признаков.

Вывод

В ходе лабораторной работы были реализованы и экспериментально протестированы основные этапы кластерного анализа: кластеризация исходных и анонимизированных данных односвязным методом с метрикой корреляции Пирсона, а также автоматический отбор наиболее информативных признаков методом последовательного добавления (Add). Анализ результатов показал, что качество кластеризации на исходных данных при большом количестве признаков и объектов невысоко, а структура кластеров выражена слабо, что отражается как в низких значениях индекса

Rand, так и в отсутствии чётких кластеров на графиках PCA (метод главных компонент).

Источники

1. **Pandas** – библиотека для анализа и обработки данных:
   McKinney, W. (2010). Data Structures for Statistical Computing in Python. In *Proceedings of the 9th Python in Science Conference*, 51-56.
   URL: https://pandas.pydata.org/

2. **NumPy** – библиотека для работы с массивами и числовыми данными:
   Harris, C. R., Millman, K. J., van der Walt, S. J., et al. (2020). Array programming with NumPy. *Nature*, 585, 357-362.
   URL: https://numpy.org/

3. **Matplotlib** – библиотека для построения графиков:
   Hunter, J. D. (2007). Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering*, 9(3), 90-95.
   URL: https://matplotlib.org/

4. **scikit-learn** – библиотека для машинного обучения, используется для линейной регрессии:
   Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
   URL: https://scikit-learn.org/

5. **PyQt5** – библиотека для создания графического интерфейса пользователя (GUI):
   Riverbank Computing Limited. PyQt5 Reference Guide.
   URL: https://www.riverbankcomputing.com/software/pyqt/intro