**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет прикладной математики-процессов управления**

**Программа бакалавриата**

**“Большие данные и распределенная цифровая платформа”**

**ОТЧЁТ**

**По лабораторной работе № 2**

**По дисциплине “Алгоритмы и структуры данных”**

**На тему “Обезличивание данных”**

**Студент гр. 24Б16-пу**

**Музиров Б.Р.**

**Преподаватель**

**Дик А.Г.**

**Санкт-Петербург**

**2025 г.**

Оглавление

[**1.** **Цель работы** 3](#_Toc209554251)

[**2.** **Описание задачи** 3](#_Toc209554252)

[**3.** **Теоретическая часть** 3](#_Toc209554253)

[**4.** **Основные шаги программы** 5](#_Toc209554255)

[**5.** **Блок схема программы** 7](#_Toc209554256)

[**6.** **Описание программы** 8](#_Toc209554257)

[**7.** **Рекомендации пользователя** 12](#_Toc209554258)

[**8.** **Рекомендации программиста** 13](#_Toc209554259)

[**9.** **Контрольный пример** 14](#_Toc209554260)

[**Вывод** 21](#_Toc209554262)

1. **Цель работы**

Цель работы — разработать программу для обезличивания синтетического датасета посетителей платной поликлиники, подсчёта k-анонимности и оценки полезности данных

1. **Описание задачи**

Для решения поставленной задачи необходимо изучить методы обезличивания медицинских данных. Используя их, требуется разработать алгоритм обезличивания датасета посетителей платной поликлиники, с возможностью выбора пользователем квази-индефикаторов.

При этом необходимо соблюдать условия, накладываемые на степень анонимности данных в зависимости от размера датасета:

1. до 51000 записей - К>=10
2. до 105000 записей - К>=7
3. до 260000 записей - К>=5

Также, нужно написать алгоритмы, работающие с метрикой k-anonymity:

* Высчитывание К-анонимити входного датасета
* Высчитывание К-анонимити обезличненного набора
* Нахождение и вывод 5 "плохих" значений К-анонимити
* Нахождение и вывод количества уникальных строк в датасете, согласно квази-индефикаторам
* Вывод уникальных строк, если переменная К=1

Кроме того, требуется оценить полезность данных, путем сравнения обезличенного набора с исходным набором. Для этого необходимо провести исследование, а также опробовать Расстояние Кульбака-Лейблера для оценки, на сколько синтетические данных отличается от распределения исходных данных, основываясь на понятии энтропии. Нужно реализовать алгоритм, который рассчитывает KL дивергенцию по столбцам датасетов.

Таким образом, итоговая программа должна содержать все вышеперичисленные алогритмы и, посредством GUI, позволять пользователю просто и удобно работать с датасетами.

1. **Теоретическая часть**

Существует множество методов обезличивания данных, существенно отличающихся друг от друга. Все методы являются параметризуемыми, то есть содержат настраиваемые характеристики (такие как набор квази-идентификаторов), определяющие конкретный алгоритм в отношении конкретного набора данных в заданных условиях (сценариях и контексте).

В данной работе используются следующие методы обезличивания:

Микро-агрегация — формирование групп записей, для которых вместо точных значений указывается диапазон. Используется для обезличивания ФИО, карты оплаты, даты посещения врача, даты получения анализов.

Локальное обобщение — уменьшение специфичности атрибута за счет подмены точного значения атрибута его общим значением. Используется для обезличивания стоймости анализов, а также медицинских данных, таких как выбор врача, симптомы и анализы.

Удаление атрибутов — удаление чувствительного контента без добавления замен. Используется для обезличивания паспорта и СНИЛС.

Локальное подавление — удаление или перекодирование относительно редких записей данных (но не более 5% от общего количества записей). Используется для обезличивания всего датасета.

K-anonymity — свойство обезличенных данных, при котором в наборе данных должно быть не менее k лиц, обладающих общим набором атрибутов, которые могут стать идентифицирующими для каждого человека.

Дивергенция KL – это несимметричная мера относительной энтропии между двумя распределениями, отражающая информационное различие между ними. Она считается по формуле:

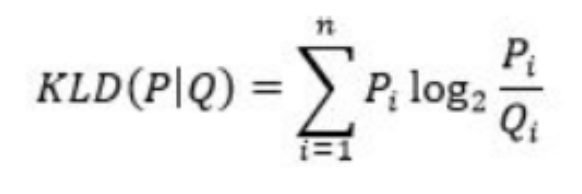


Рис 3.1 Формула расчёта KL дивергенции

1. **Основные шаги программы**
2. **Загрузка и отображение графического интерфейса.**  
   С помощью библиотеки PyQt5 создаётся окно программы, содержащее все основные элементы:

* Область с чекбоксами для выбора квази-идентификаторов;
* таблицы для вывода показателей k-анонимности и оценки полезности данных;
* кнопки управления («Загрузить файл», «Обезличить», «Рассчитать k-анонимность», «Показать уникальные строки», «Оценить полезность данных», «Сохранить»).

1. **Выбор и загрузка входного файла.**  
   Пользователь выбирает CSV-файл (итоговый набор данных из первой лабораторной работы).  
   Файл считывается с помощью библиотеки pandas.
2. **Выбор квази-идентификаторов.**  
   В интерфейсе пользователь отмечает поля, по которым будет выполняться анонимизация.  
   Также предусмотрена кнопка «Выбрать всё» для автоматического выбора всех атрибутов.
3. **Обезличивание данных.**  
   После нажатия кнопки «Обезличить» выполняется анонимизация выбранных полей:

* Группировка симптомов, анализов, врачей по категориям;
* Обобщение ФИО посетителей по полам.
* Преобразование банковских карт в зависимости от того, российская или международная у карты система.
* Удаление паспортных данных и СНИЛС;
* Разделение стоимости анализов на интервалы;
* Даты преобразуются в полугодия;
* Для повышения уровня приватности применяется **локальное подавление**, удаляющее до 5% строк с наименьшими значениями *k*.

1. **Расчёт k-анонимности.**  
   После анонимизации программа вычисляет показатель *k* для каждой записи, определяя размер группы одинаковых квази-идентификаторов.  
   Отдельно выводятся пять наименее анонимных групп (с минимальным *k*) и их процент от общего объёма данных.
2. **Вывод количества уникальных строк.**  
   Программа вычисляет количество уникальных групп по выбранным квази-идентификаторам, а при K = 1 — выводит соответствующие уникальные строки.
3. **Оценка полезности данных.**  
   Сравниваются распределения признаков исходного и анонимизированного датасетов с использованием дивергенции Кульбака–Лейблера (KL-дивергенция).  
   Для числовых данных строятся гистограммы, для категориальных — частотные распределения.  
   Чем меньше значение KL-дивергенции, тем ближе распределения и тем выше полезность данных.
4. **Сохранение обезличенного набора.**  
   После завершения обработки пользователь может сохранить обезличенный датасет в новый CSV-файл с сохранением кодировки UTF-8.
5. **Блок схема программы**

A diagram of a company

AI-generated content may be incorrect.

Рис. 5.1. Блок-схема основной программы

1. **Описание программы**

Программная реализация задачи написана на языке Python с использованием модулей sys, numpy, pandas. Для вычисление дивергенции используется Scipy. Для GUI используется графическая библиотека PyQt5. Программа представляет собой графический интерфейс, который позволяет загружать данные и выполнять определенные команды

Таблица 1. Описание методов класса MainWindow файла anonymization\_interface

|  |  |  |
| --- | --- | --- |
| Метод | Тип возвращаемого значения | Описание |
| \_\_init\_\_(self,) | None | Конструктор класса. Создаёт главное окно программы, интерфейс, элементы управления (кнопки, таблицы, чекбоксы) и задаёт их компоновку. |
| set\_selected(self) | list[str] | Возвращает список выбранных пользователем квази-идентификаторов из набора чекбоксов. Если включён флажок «Выбрать всё», возвращаются все возможные идентификаторы. |
| load\_file(self) | None | Открывает диалог выбора файла и загружает CSV-датасет в pandas.DataFrame. При успешной загрузке копирует данные в anonymized\_df. Выводит сообщение об успехе или ошибке. |
| run\_anonymization(self) | None | Выполняет анонимизацию по выбранным квази-идентификаторам, вызывая соответствующие функции из модуля anonymization\_functions. После этого применяет локальное подавление (local\_suppression). |
| calculate\_k(self) | None | Вычисляет распределение K-анонимности по выбранным квази-идентификаторам. Использует функции calculate\_k\_anonymity и get\_bad\_k. Отображает таблицу с «плохими» значениями K. |
| recommend\_k(self) | None | Определяет рекомендуемое значение параметра K-анонимности на основе размера загруженного набора данных. Результат отображается на панели интерфейса. |
| show\_unique\_rows(self) | None | Показывает строки с K = 1 (уникальные комбинации квази-идентификаторов). Выводит их количество и отображает в отдельной таблице k1\_table. |
| evaluate\_utility(self) | None | Оценивает полезность обезличенных данных путём расчёта KL-дивергенции между исходным и анонимизированным распределением данных. Результаты выводятся в таблице kl\_table. |
| save\_file(self) | None | Сохраняет обезличенные данные (anonymized\_df) в CSV-файл, выбранный пользователем. При успешном сохранении выводит уведомление. |

Таблица 2. Описание функций файла anonymization\_functions

|  |  |  |
| --- | --- | --- |
| **Функция** | Тип возвращаемого значения | Описание функции |
| get\_season(date: datetime) | str | Определяет, к какому полугодию относится переданная дата (1 или 2 полугодие). |
| get\_bank(card\_number: int) | str | По первым шести цифрам банковской карты (BIN) определяет платёжную систему и возвращает её тип — «Российская» или «Международная». |
| change\_to\_gender(SNP: str) | str | Преобразует ФИО в пол человека, определяя его по фамилии на основе словаря names. |
| categorize\_tests(value: str) | str | Классифицирует наименование анализа (теста) по категориям из словаря test\_local\_aggregation. |
| anonymize\_SNP(df: pd.DataFrame) | None | Обезличивает столбец ФИО, заменяя значения на пол пациента. |
| anonymize\_passport(df: pd.DataFrame) | None | Удаляет паспортные данные, заменяя значения на «Удалено». |
| anonymize\_snils(df: pd.DataFrame) | None | Удаляет СНИЛС, заменяя значения на «Удалено». |
| anonymize\_bankcard(df: pd.DataFrame) | None | Определяет платёжную систему и страну по номеру карты и заменяет исходные значения на результат функции get\_bank(). |
| anonymize\_price(df: pd.DataFrame) | None | Делит значения стоимости анализов на интервалы (категории) по диапазонам. |
| anonymize\_dates\_visit(df: pd.DataFrame) | None | Преобразует даты посещения врача в категории «1 Полугодие» / «2 Полугодие». |
| anonymize\_dates\_takeaway(df: pd.DataFrame) | None | Преобразует даты получения анализов в полугодия. Поддерживает согласованность с датой визита. |
| anonymize\_symptoms(df: pd.DataFrame) | None | Обобщает симптомы, заменяя конкретные жалобы на категорию. |
| anonymize\_tests(df: pd.DataFrame) | None | Обобщает названия анализов с помощью функции categorize\_tests(). |
| anonymize\_doctors(df: pd.DataFrame) | None | Группирует врачей по специализациям. |
| calculate\_k\_anonymity(df: pd.DataFrame, quasi\_identifiers: list) | tuple[list[int], int, pd.DataFrame] | Рассчитывает значение K-анонимности для каждой строки по выбранным квази-идентификаторам. Возвращает список K для каждой записи, количество уникальных комбинаций и DataFrame со строками, где K = 1. |
| local\_suppression(df: pd.DataFrame, quasi\_identifiers: list, max\_fraction=0.05) | tuple[pd.DataFrame, int, int] | Реализует метод локального подавления: удаляет до 5% строк с наименьшими значениями K для повышения анонимности. Возвращает обновлённый DataFrame, общее количество строк и число удалённых записей. |
| get\_bad\_k(k\_list: list, total\_length: int, top\_n=5) | list[tuple[int, str]] | Находит и возвращает топ-5 наименьших значений K с указанием их доли (%) в наборе данных. |
| unique\_rows\_info(df: pd.DataFrame, quasi\_identifiers: list) | tuple[int, pd.DataFrame] | Подсчитывает количество уникальных комбинаций квази-идентификаторов и возвращает DataFrame со строками, где K = 1. |

# **Рекомендации пользователя**

1. Запуск программы

* Введите в командную строку python anonymization\_interface.py в директории программы:



Рис. 7.1 Запуск

1. Загрузка файла с данными
2. Нажмите кнопку "Загрузить файл".
3. Выберите файл в формате CSV на своём компьютере и нажмите Открыть.
4. После успешной загрузки появится сообщение: "Файл успешно загружен".
5. Программа автоматически предложит рекомендуемое значение K-анонимности.
6. Выбор квази-идентификаторов
7. В верхней части окна вы увидите список с названиями столбцов (ФИО, Паспортные данные, СНИЛС и т.д.).
8. Отметьте столбцы, которые хотите обезличить. Если хотите выбрать все столбцы сразу — отметьте "Выбрать всё".
9. Обезличивание данных
10. Нажмите кнопку "Обезличить".
11. Программа автоматически анонимизирует личные
12. В конце появится сообщение о том, сколько строк было удалено при обезличивании.
13. Проверка K-анонимности
14. Нажмите кнопку "Рассчитать k-анонимность".
15. В таблице появятся "плохие K" — это группы данных, которые легче всего деанонимизировать.
16. Просмотр уникальных строк
17. Нажмите кнопку "Уникальные строки (K=1)".
18. Программа покажет, сколько строк уникальны и могут раскрыть личность человека.
19. Эти строки будут отображены в отдельной таблице для проверки.
20. Оценка полезности данных
21. Нажмите кнопку "Оценить полезность данных".
22. Программа сравнит обезличенные данные с исходными и покажет, насколько они похожи.
23. В таблице вы увидите столбцы и значения KL-дивергенции — чем меньше, тем данные полезнее для анализа.
24. Сохранение обезличенных данных
25. Нажмите кнопку "Сохранить".
26. Выберите место на компьютере и введите название файла.
27. Программа сохранит обезличенные данные в CSV-файл.
28. Закрытие программы

Когда закончите работу, просто закройте окно программы.

# **Рекомендации программиста**

Перед запуском программы убедитесь, что на вашем компьютере установлены следующие модули и библиотеки Python: PyQt5, pandas, numpy, scipy, collections, datetime, random.

В случае отсутствия установите их через pip:



Рис. 8.1 Установка

Рекомендации по работе с данными:

* CSV-файл должен быть разделён точкой с запятой (;), кодировка — utf-8-sig.
* Обязательные квази-идентификаторы: ФИО, Паспортные данные, СНИЛС, Симптомы, Выбор врача, Дата посещения врача, Анализы, Дата получения анализов, Стоимость анализов, Карта оплаты.

# **Контрольный пример**

Для проверки корректности работы программы проведём запуск и используем весь функционал

Шаг 1. Запуск программы



Рис. 9.1 Запуск

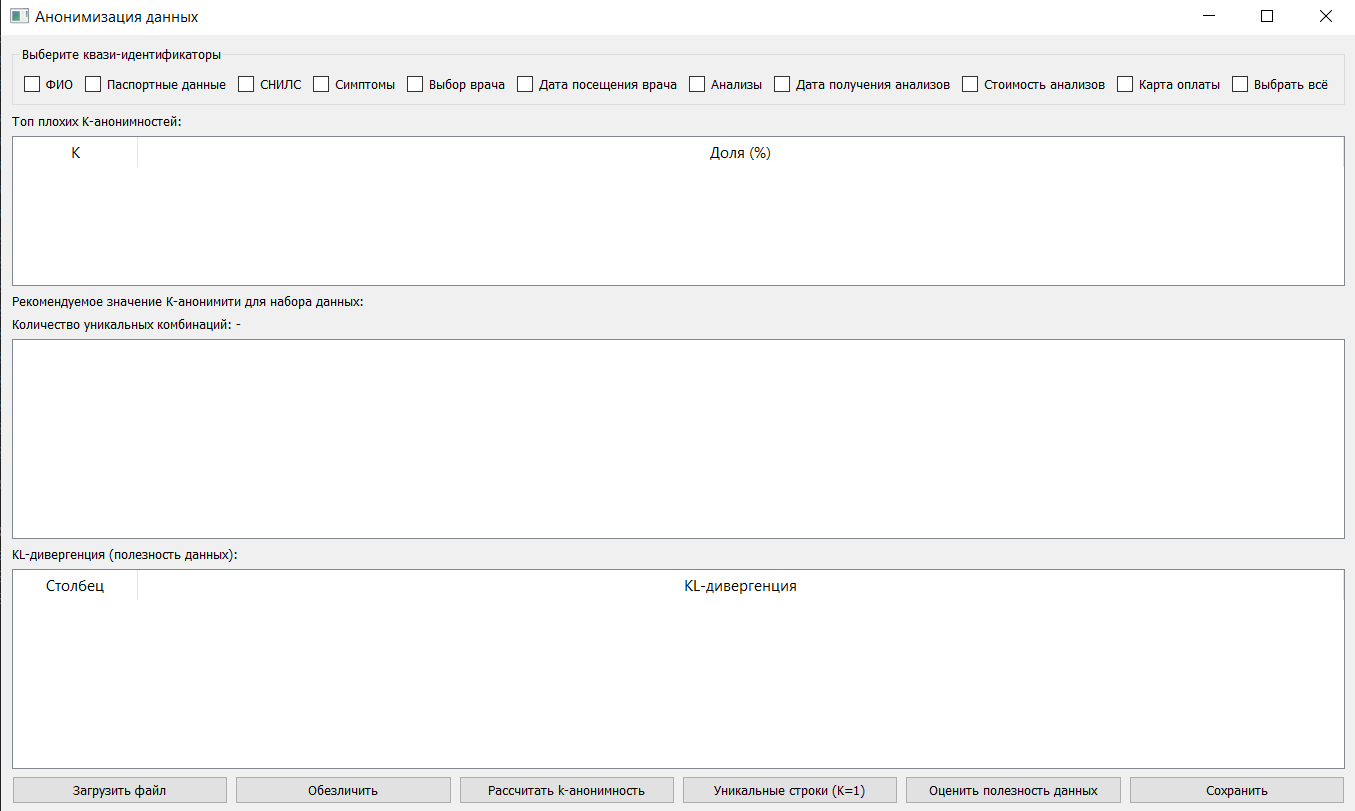
Шаг 2. Окно программы

Рис. 9.2 Окно программы

Перед нами — окно программы (см. Рис 9.2). Перед тем, обрабатывать данные, их нужно загрузить. Нажмём на кнопку «Загрузить файл».

Шаг 3. Загрузка

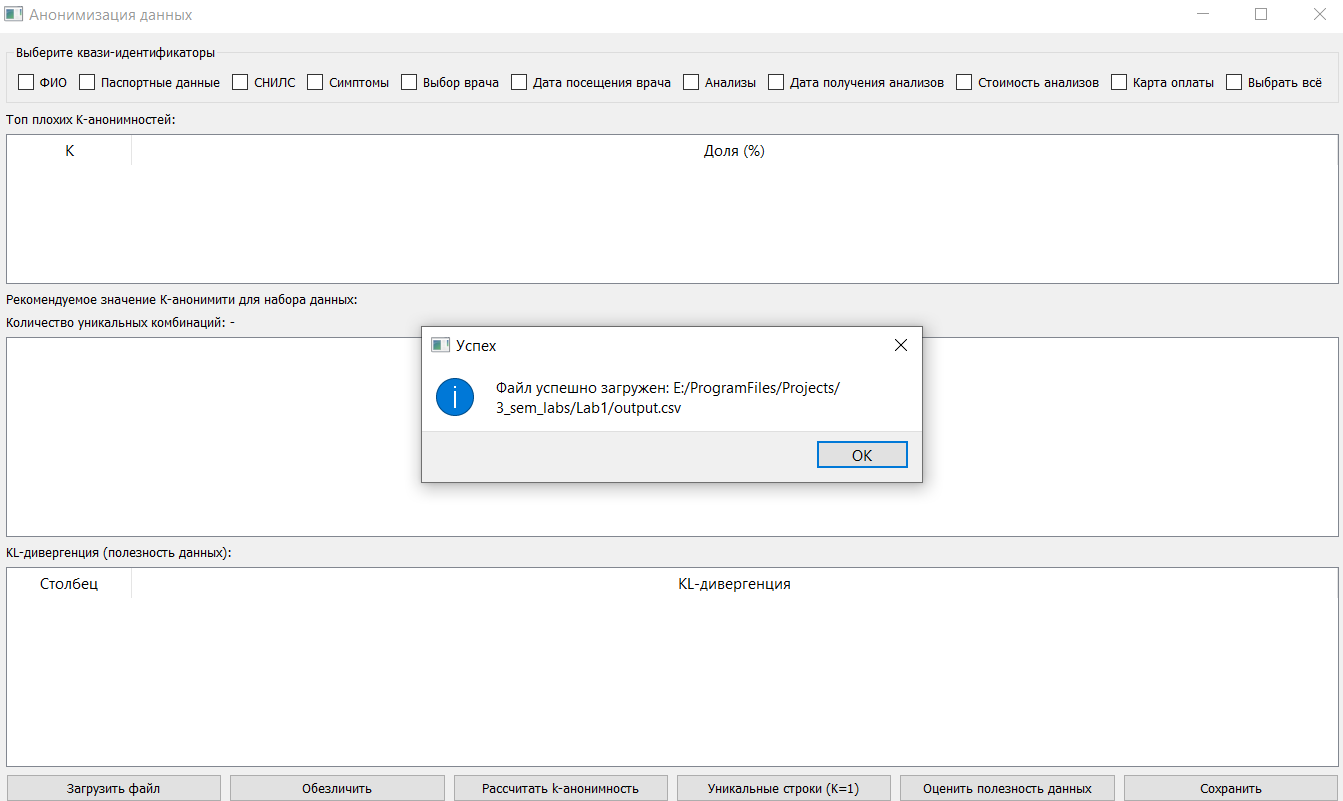


Рис. 9.3 Загрузка

При успешной загрузке программа оповестит об этом сообщением (см. Рис. 9.3).

Шаг 4. Обезличивание

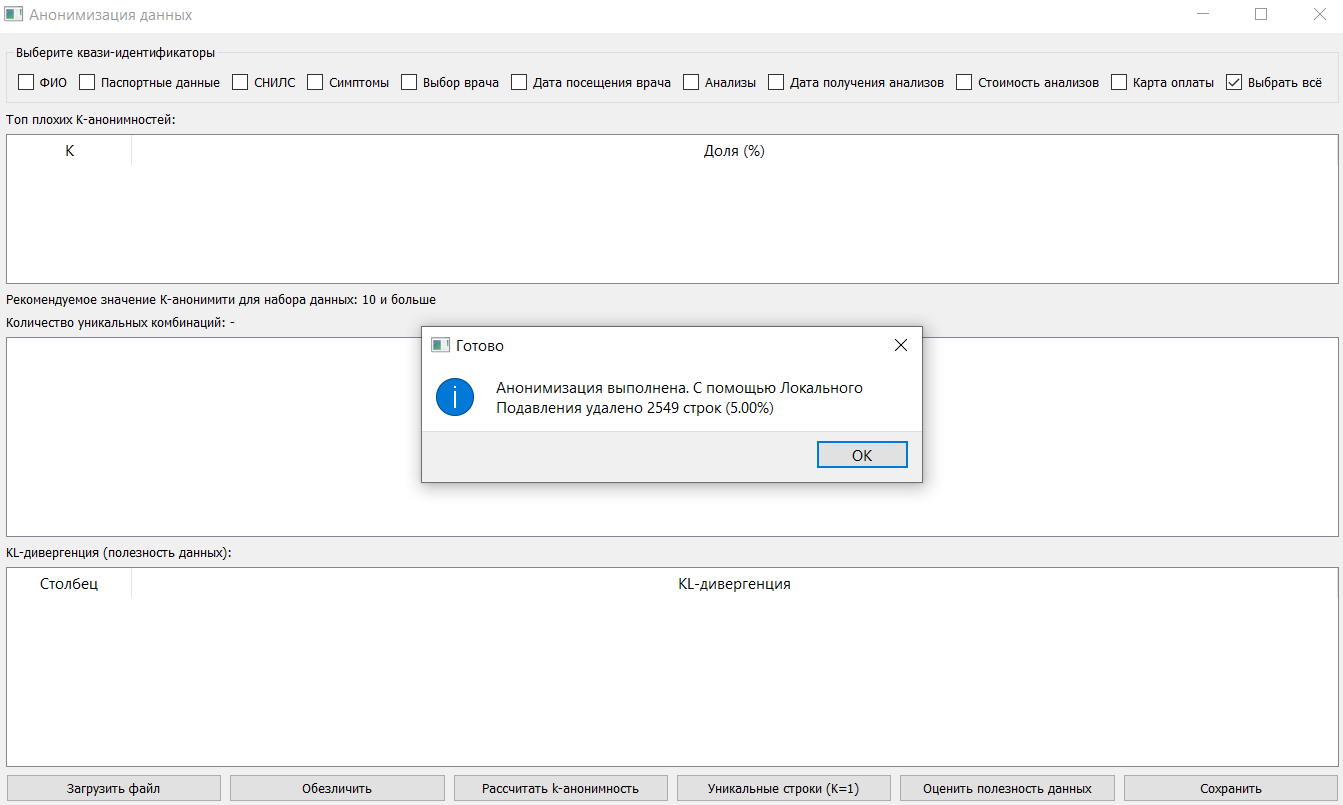


Рис. 9.4 Обезличивание

Мы можем выбрать квази-идентефикаторы, по которым будем проводить обезличивание. Выберем все, отметив соответствующий чекбокс в верхней части окна. После этого нажмём «Обезличить». В случае успеха получим сообщение об этом (см. Рис 9.4).

Шаг 5. Расчёт k-anonymity

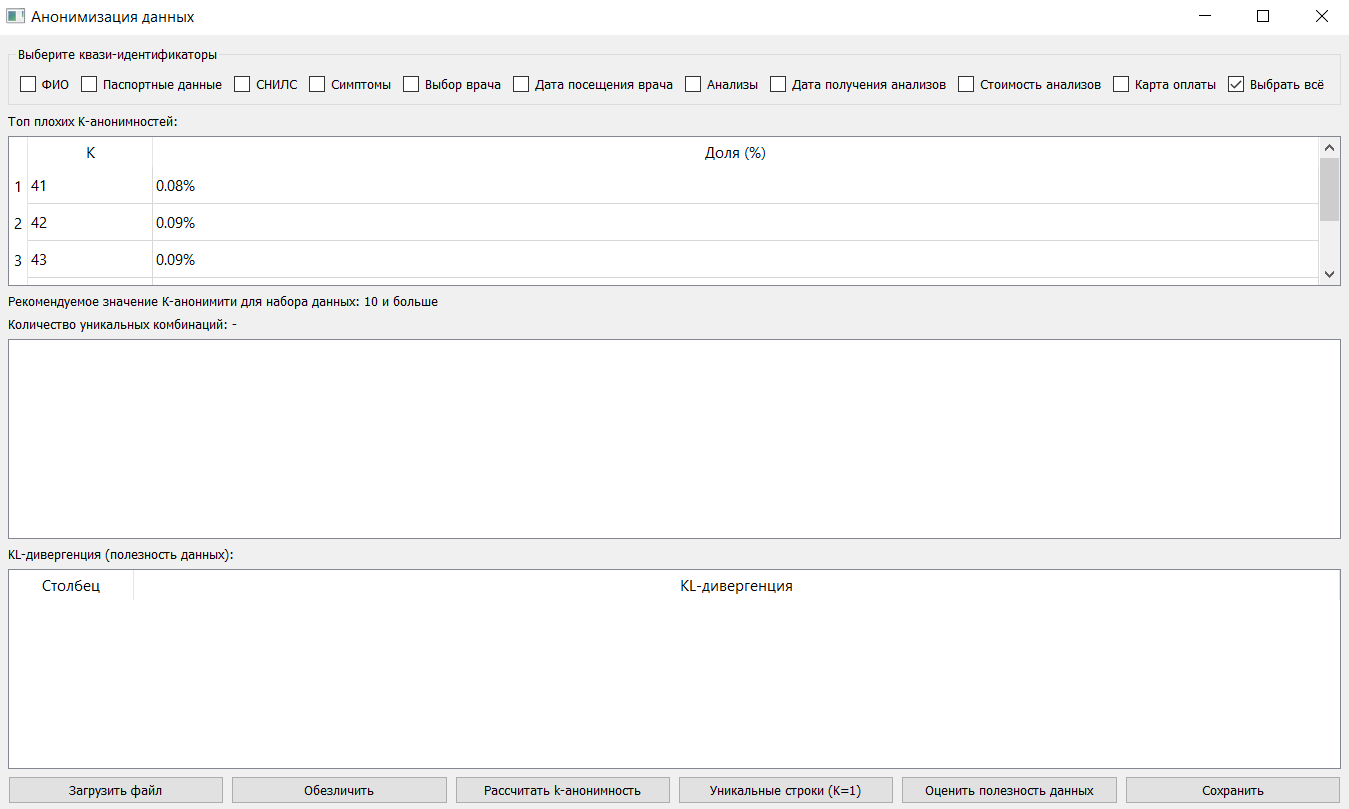


Рис. 9.5 Расчёт k-anonymity

После анонимизации можем рассчитать k-anonymity для обезличненного датасета. Для этого нажмём соответствующую кнопку. После расчёта в верхнем поле появятся 5 самых малых значений k-anonymity (см. Рис 9.5).

Шаг 6. Подсчёт уникальных строк

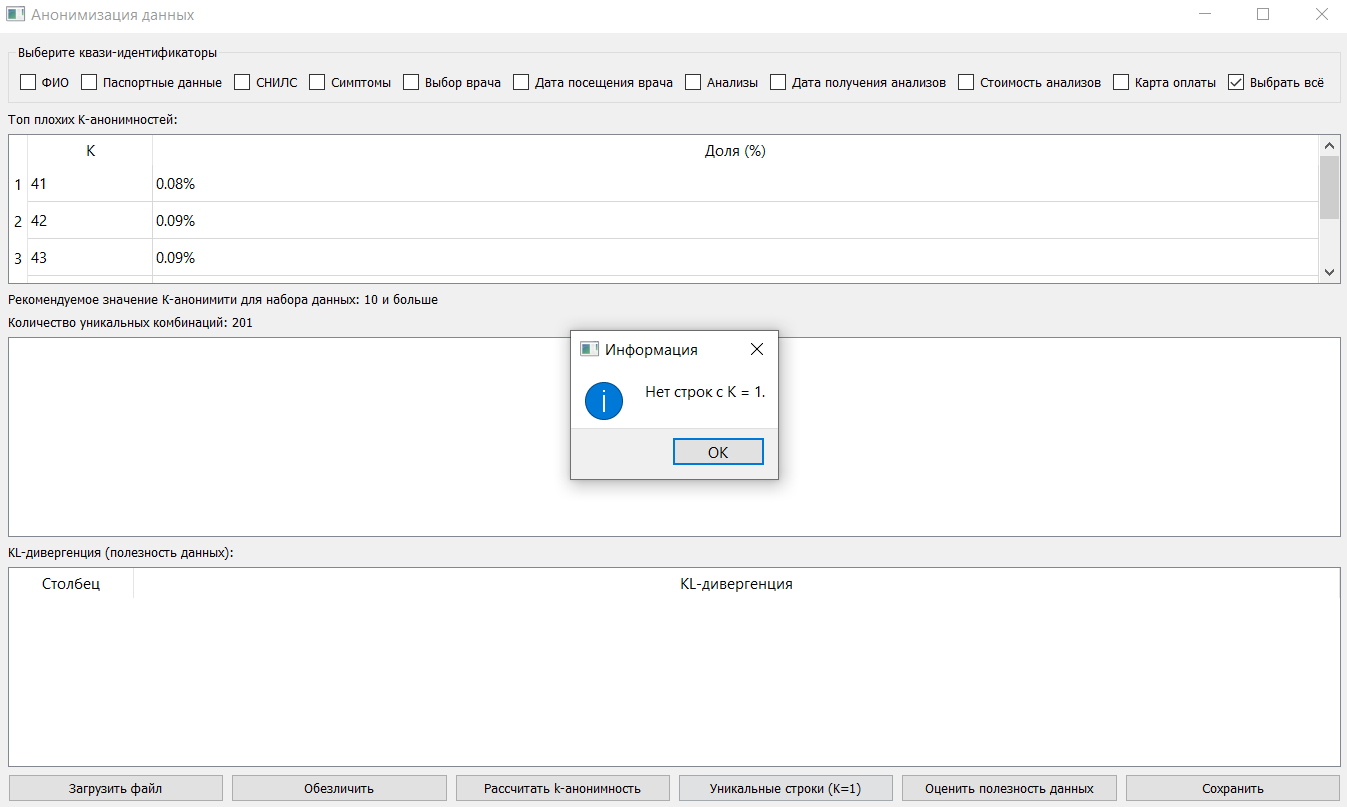


Рис. 9.6 Посчёт уникальных строк

Можем подсчитать количество уникальных строк. Для этого нажмём соответствующую кнопку. Если k-anonymity = 1, т.е. есть уникальные строки, они появятся в среднем поле. Иначе выведется сообщение, что таких строк нет. Также вне зависимости от результата отобразится количество уникальных комбинациий в соответствующей строке. (см. Рис 9.6).

Шаг 7. Полезность данных

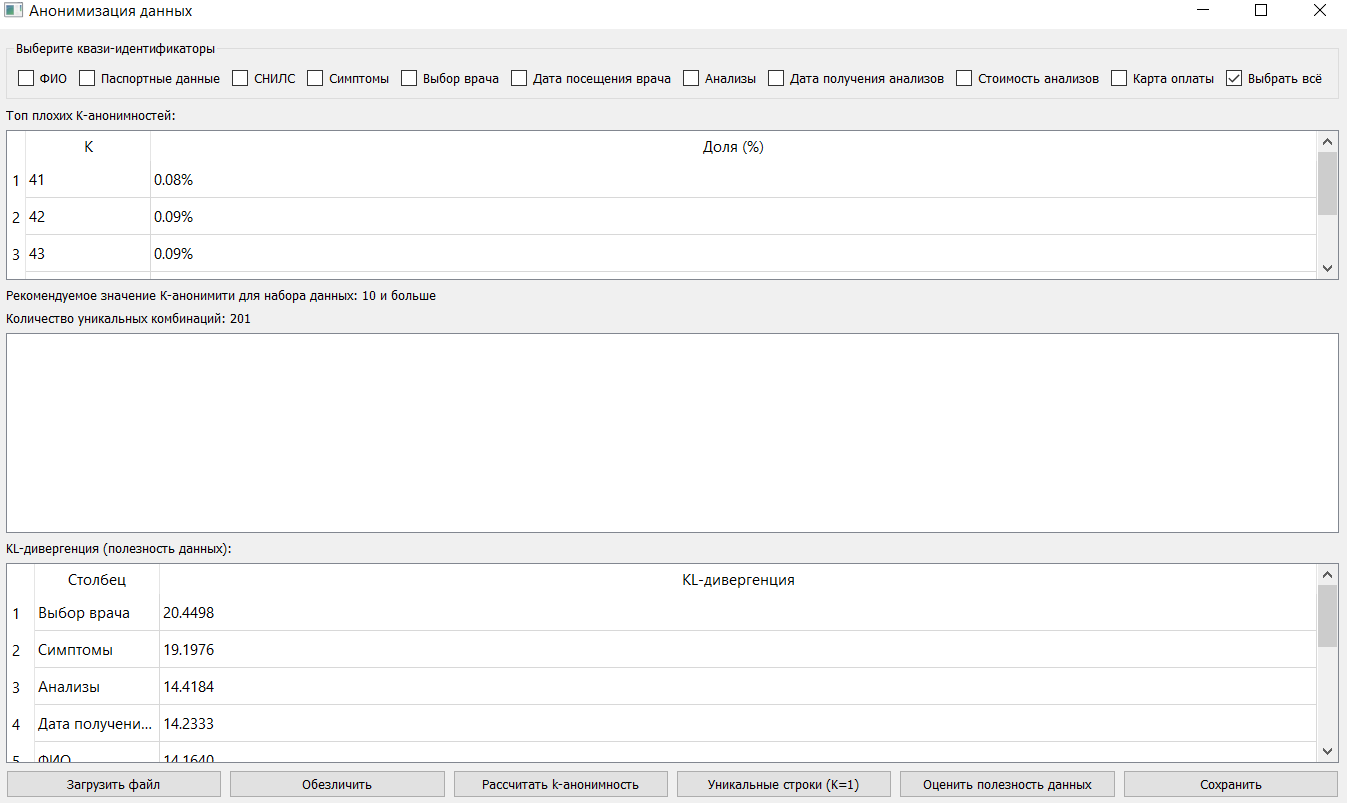


Рис. 9.7 Полезность данных

Для оценки полезности нажмём «Оценить полезность данных». Программа посчитает KL-дивергенцию для каждого идентефикатора и отобразит значение в нижнем поле (см. Рис 9.7).

Шаг 8. Сохранение

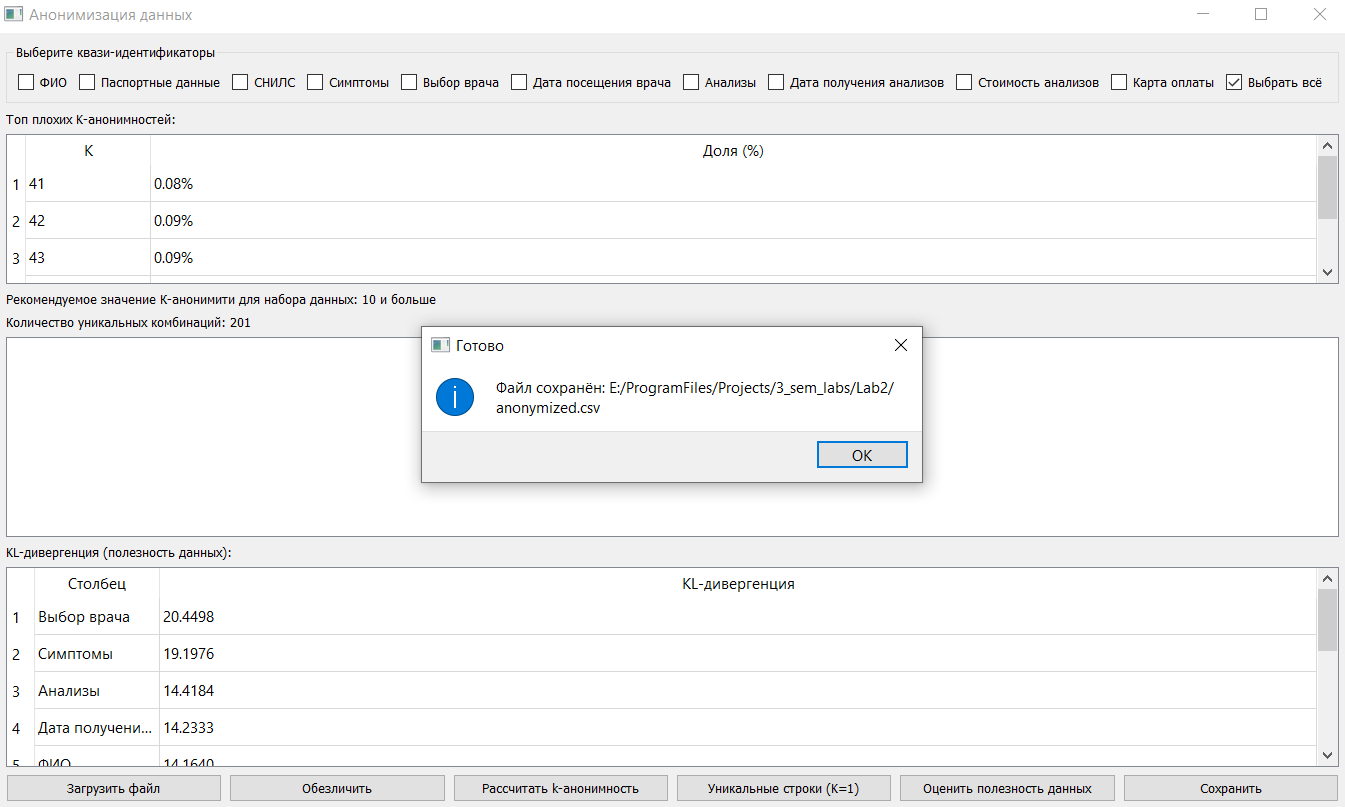


Рис. 9.8 Полезность данных

Мы можем сохранить обезличненный датасет, нажав на соответствующую кнопку. Далее выберем место, куда сохранить. При успехе получим сообщение об этом (см. Рис 9.8).

# **Вывод**

Разработана программа для обезличивания синтетического датасета посетителей платной поликлиники, подсчёта k-анонимности и оценки полезности данных. Изучены и применены на практике методы обезличивания.