САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет прикладной математики-процессов управления

Программа бакалавриата
“Большие данные и распределенная цифровая платформа”

ОТЧЕТ

по лабораторной работе №4

по дисциплине «Алгоритмы и структуры данных»

на тему «Восстановление данных»

Вариант – «Хот-Дек, метод заполнения средним значением и заполнение пропусков на основе линейной регрессии»

Студент гр. 23Б15-пу
Сериков К.Г.

Преподаватель
Дик А.Г.

Санкт-Петербург
2025 г.

Оглавление

Цель работы

Целью данной лабораторной работы является изучение и реализация различных методов заполнения пропусков в датасетах, а также оценка их эффективности на основе сравнительного анализа. В ходе работы будут использованы такие методы, как: Хот-Дек, метод заполнения средним значением и заполнение пропусков на основе линейной регрессии.

Описание задачи (формализация задачи)

В данной лабораторной работе необходимо исследовать эффективность различных методов восстановления датасета.

Формализация задачи включает следующие компоненты:

1. **Изучение особенностей муравьиного алгоритма:** Исследовать принципы работы различных методов заполнения.
2. **Создание датасетов**: Для начала формируются три датасета различных размеров (маленький - 1000 строк, средний — 25.000 строк и большой — 100.000 строк) при помощи первой лабораторной работы третьего семестра.
3. **Удаление значений**: Необходимо случайным образом удалить 3%, 5%, 10%, 20% и 30% значений.
4. **Применение методов заполнения**: Для восстановления пропущенных значений применяются следующие методы: Хот-Дек, метод заполнения средним значением и заполнение пропусков на основе линейной регрессии.
5. **Тестирование программы на взвешенном ориентированном графе:** Провести тестирование программы на контрольных разных датасетах и с разным процентами удалённых значений.
6. **Анализ эффективности заполнения**: Провести сравнительный анализ для каждого метода. Вычисляется суммарная погрешность предсказанных значений по сравнению с истинными значениями.

Теоретическая часть

1. Метод Хот-Дек

Метод Хот-Дек основан на заполнении пропущенных значений случайно выбранными значением из того же столбца. В представленной реализации для каждого столбца независимо извлекается подмножество существующих значений, после чего пропуски заменяются случайными элементами данного подмножества. Преимущество метода заключается в сохранении распределения данных, однако он не учитывает потенциальные взаимосвязи между переменными.

2. Метод заполнения средним значением

Данный метод представляет собой подход к заполнению пропусков, при котором отсутствующие значения в числовых столбцах заменяются средним арифметическим. Алгоритм демонстрирует вычислительную эффективность и обеспечивает сохранение общего среднего по столбцу, что упрощает последующий статистический анализ. Однако метод систематически искажает распределение данных. Применение ограничено исключительно численными переменными.

3. Метод заполнения пропусков на основе линейной регрессии

Импутация на основе линейной регрессии относится к продвинутым методам восстановления данных, использующим зависимость целевой переменной от предикторов. Метод обеспечивает более точную импутацию за счет учета зависимостей, но требует достаточного объема данных для обучения модели и вычислительно более затратен по сравнению с простыми методами.

Оценка качества восстановления данных

Для оценки качества восстановления данных рассчитываются суммарные относительные погрешности для каждого метода. Метод, для которого суммарная относительная погрешность будет минимальной, будет наиболее эффективным.

Погрешность для числовых столбцов вычисляется как относительное отклонение между восстановленным и оригинальным значением. Формула расчета погрешности для каждого значения:

$$\Delta_{M_j} = \sum_{i=1}^{n} \frac{|a_i - \overline{a}_i|}{a_i} \cdot 100\ \%,$$

Погрешность для категориальных данных измеряется как процент ошибок в восстановленных значениях, когда восстановленное значение не совпадает с оригинальным.

Основные шаги программы

1. **Запуск программы:** Инициализируется графический интерфейс с элементами управления.
2. **Выбор и загрузка файлов**: С помощью графического интерфейса позволяет выбрать исходный файл с датасетом и задать процент пропусков для удаления, а также можно выбрать файл для восстановления и файлы для расчёта ошибок.
3. **Создание пропусков в данных**: Программа случайным образом удаляет заданный процент значений из каждого столбца таблицы. Результат сохраняется в отдельный файл.
4. **Восстановление данных**: Пользователю предлагается выбор из нескольких методов восстановления: Хот-Дек, метод заполнения средним значением и заполнение пропусков на основе линейной регрессии.
5. **Расчёт погрешности восстановления**: Программа сравнивает восстановленные данные с оригинальными и выводит процентные ошибки по каждому столбцу, а также общие средние значения погрешности.

Блок схема программы

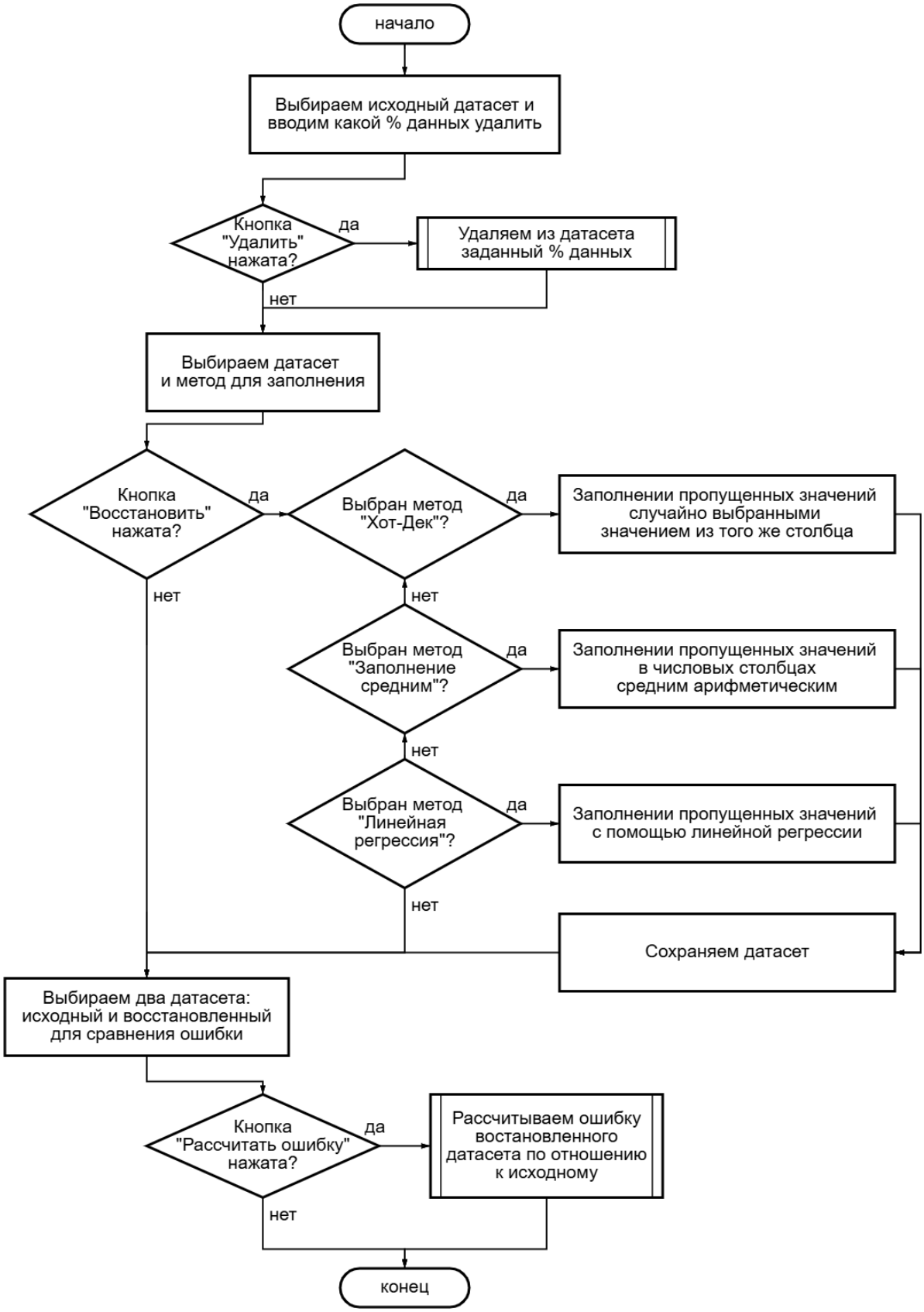

Рис 1. Блок-схема основной программы

Описание программы

Программная реализация написана на языке Python 3.13.0 с использованием следующих библиотек: tkinter [1], numpy [2], pandas [3], os [4] и sklearn.linear_model [5]. Программа организована в виде графического интерфейса для восстановления данных. В программе реализован только один класс – DataRecoveryApp, который отвечает за работу всей программы. В процессе разработки программы использовался main.py, включающий 13 функций, каждая из которых имеет чётко определённое назначение:

Таблица 1. main.py

Функция	Описание	Возвращаемое значение
__init__	Инициализация интерфейса.	None
create_widgets	Создает и размещает все элементы интерфейса	None
select_source_file	Открывает диалог выбора исходного CSV-файла	None
select_restore_file	Открывает диалог выбора файла для восстановления	None
select_input_file	Открывает диалог выбора исходного файла для расчета ошибки	None
select_output_file	Открывает диалог выбора восстановленного файла для расчета ошибки	None
delete_data	Удаляет указанный процент данных из файла	None
hot_deck_imputation	Заполняет пропуски методом Hot-deck	DataFrame

mean_imputation	Заполняет пропуски средними значениями	DataFrame
regression_imputation	Заполняет пропуски через линейную регрессию	DataFrame
recover_data	Выбирает метод восстановления данных	DataFrame
restore_data	Применяет выбранный метод восстановления к файлу	None
calculate_error	Сравнивает исходные и восстановленные данные, вычисляя ошибки	None

Рекомендации пользователя

1. **Запуск программы**: Откройте программу с помощью Python 3.13.0, чтобы инициализировать интерфейс.
2. **Удаление данных (создание пропусков):** Нажмите кнопку «Выбрать» напротив поля «Исходный файл» и выберите csv-файл с данными. Введите в поле «Удалить (%):» процент ячеек, которые нужно удалить. Нажмите кнопку «Удалить» — программа создаст копию файла с пропущенными данными.
3. **Восстановление данных:** Нажмите на вкладку «Восстановление данных», затем на кнопку «Выбрать» напротив поля «Файл для восстановления» и выберите файл с пропущенными значениями. В выпадающем списке «Метод восстановления» выберите один из трёх методов: Хот-Дек, метод заполнения средним значением и заполнение пропусков на основе линейной регрессии. Нажмите кнопку «Восстановить» — будет создан файл с восстановленными значениями.
4. **Расчёт ошибки:** Нажмите на вкладку «Расчёт ошибки», затем исходный файл— нажмите «Выбрать» напротив «Исходный файл». Выберите восстановленный файл — нажмите «Выбрать» напротив «Получившийся файл».

Нажмите кнопку «Рассчитать ошибку». В правой части окна появится подробная оценка ошибок восстановления по столбцам и в целом.

Рекомендации программиста

Актуальность версии Python: Используйте обновленную версию Python, tkinter [1], numpy [2], pandas [3], os [4] и sklearn.linear_model [5]. Уделяйте внимание четкому именованию переменных и функций. Тщательно тестируйте на различных графах, чтобы удостовериться в корректной работе алгоритма и визуализации. Убедитесь, что интерфейс программы в Tkinter легко понимается пользователями. Разделите входные данные, управление алгоритмом и отображение результатов по разным секциям.

Исходный код программы

https://github.com/romplle/spbu-algorithms-and-data-structures/

Контрольный пример

1. Запуск программы и ввод параметров: Для запуска программы откройте main.py. Программа откроет графический интерфейс (Рис. 2).

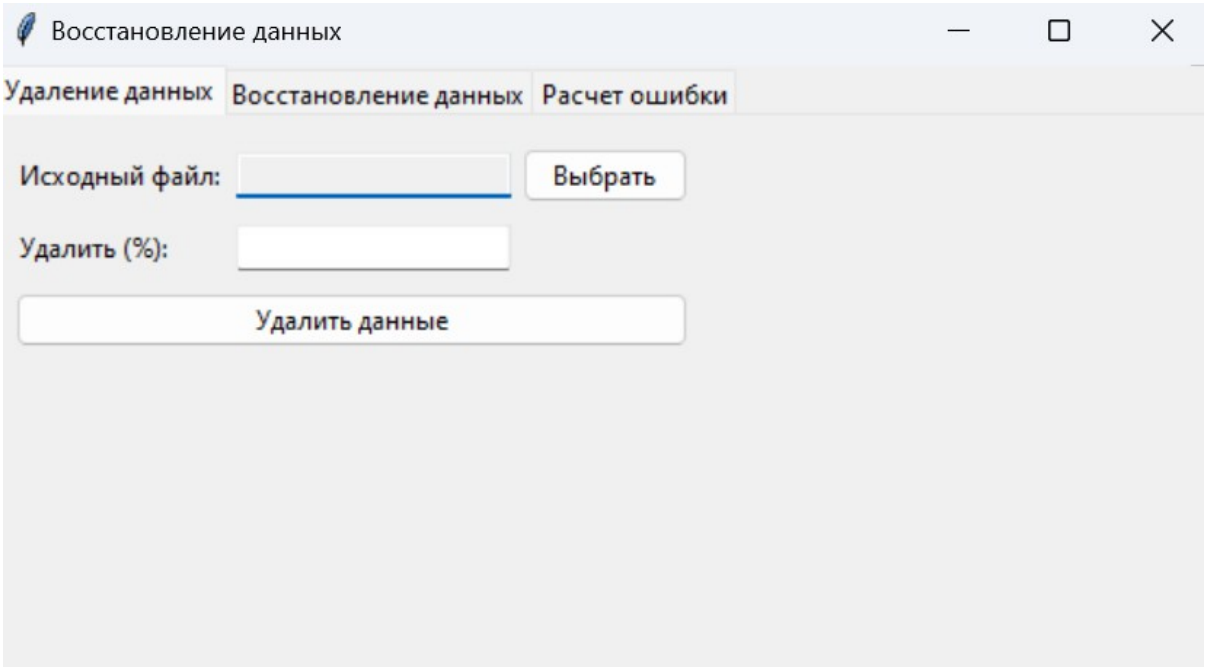

Рис. 2. Вкладка «Удаление данных

2. Удаление данных: Введите в поле «Удалить (%):» процент ячеек, которые нужно удалить. Нажмите кнопку «Удалить» — программа создаст копию файла с пропущенными данными. (Рис. 3).

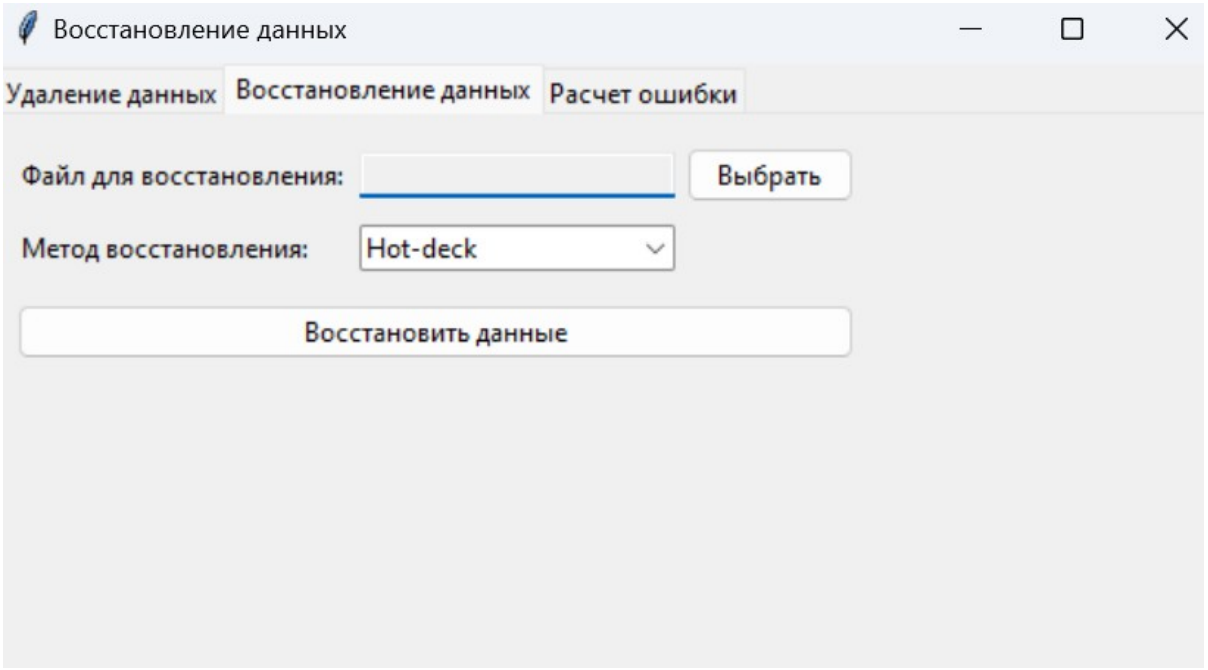

Рис. 3. Вкладка «Восстановление данных

3. **Восстановление пропущенных значений**: В списке «Метод восстановления» выберите один из трёх методов: Хот-Дек, метод заполнения средним значением и заполнение пропусков на основе линейной регрессии. Нажмите кнопку «Восстановить» — будет создан файл с восстановленными значениями) (Рис. 4).

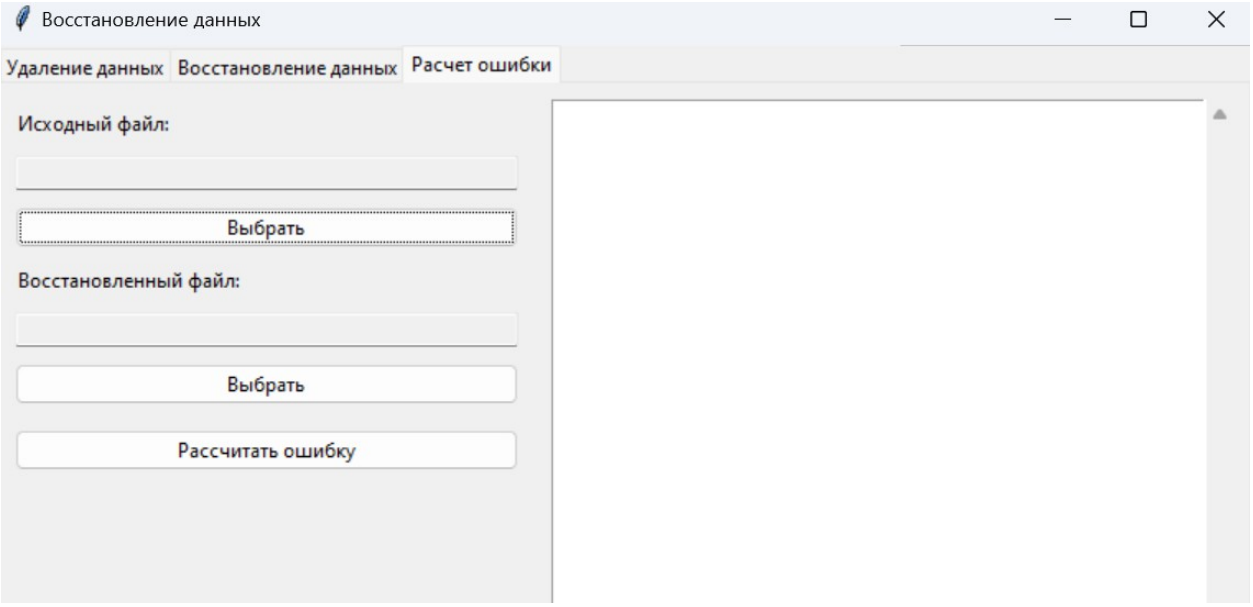

Рис. 4. Вкладка «Расчёт ошибки» до расчёта

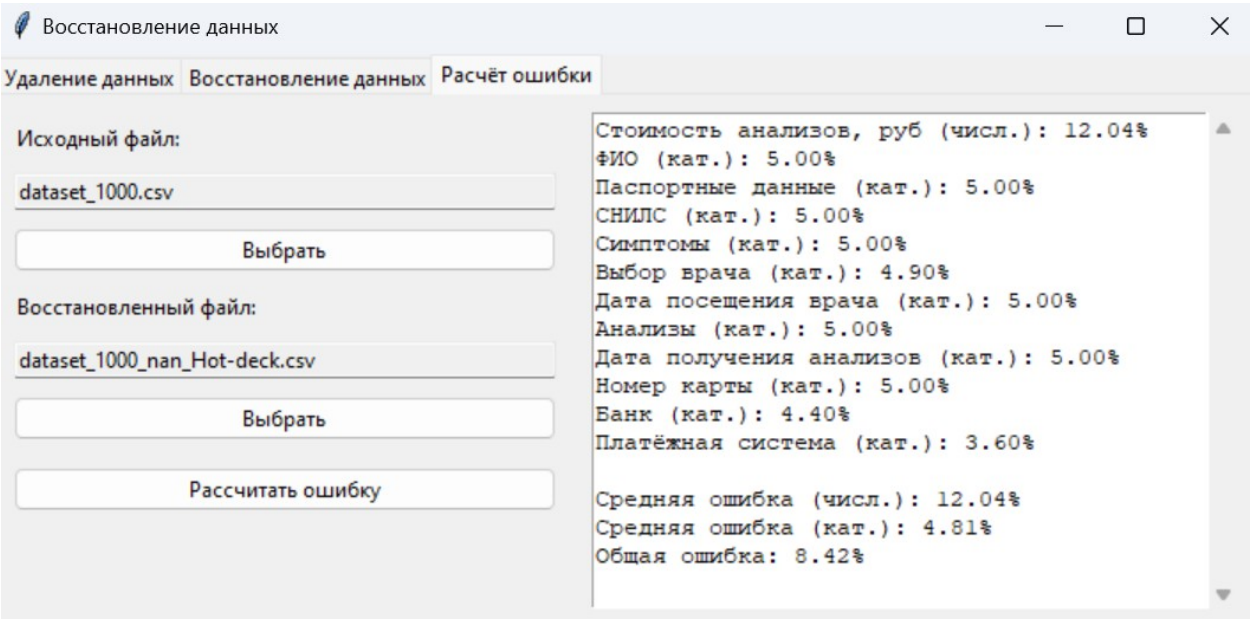

Рис. 5. Вкладка «Расчёт ошибки» после расчёта

Анализ

Таблица 2. Результаты для маленького датасета (1000 строк)

<table>
<tr><th>Название столбцов</th><th>Процент удаления</th><th>Хот-Дек</th><th>Заполнения средним значением</th><th>Заполнение пропусков на основе линейной регрессии.</th></tr>
<tr><td>ФИО (кат.)</td><td rowspan="15">3%</td><td colspan="3" rowspan="4">3.00%</td></tr>
<tr><td>Паспортные данные (кат.)</td></tr>
<tr><td>СНИЛС (кат.)</td></tr>
<tr><td>Симптомы (кат.)</td></tr>
<tr><td>Выбор врача (кат.)</td><td>2.90%</td><td colspan="2">3.00%</td></tr>
<tr><td>Дата посещения врача (кат.)</td><td colspan="3" rowspan="4">3.00%</td></tr>
<tr><td>Анализы (кат.)</td></tr>
<tr><td>Дата получения анализов (кат.)</td></tr>
<tr><td>Номер карты (кат.)</td></tr>
<tr><td>Банк (кат.)</td><td>2.30%</td><td colspan="2">3.00%</td></tr>
<tr><td>Платёжная система (кат.)</td><td>2.10%</td><td colspan="2">3.00%</td></tr>
<tr><td>Стоимость анализов, руб (числ.)</td><td>5.39%</td><td>4.58%</td><td>3.96%</td></tr>
<tr><td>Средняя ошибка (кат.)</td><td>2.85%</td><td colspan="2">3.00%</td></tr>
<tr><td>Общая ошибка</td><td>3.06%</td><td>3.13%</td><td>3.08%</td></tr>
<tr><td colspan="5"></td></tr>
</table>

<table>
<tr><td>ФИО (кат.)</td><td rowspan="14">5%</td><td colspan="3" rowspan="3">5.00%</td></tr>
<tr><td>Паспортные данные (кат.)</td></tr>
<tr><td>СНИЛС (кат.)</td></tr>
<tr><td>Симптомы (кат.)</td><td colspan="3" rowspan="2">5.00%</td></tr>
<tr><td>Выбор врача (кат.)</td><td>4.90%</td><td colspan="2">5.00%</td></tr>
<tr><td>Дата посещения врача (кат.)</td><td colspan="3" rowspan="4">5.00%</td></tr>
<tr><td>Анализы (кат.)</td></tr>
<tr><td>Дата получения анализов (кат.)</td></tr>
<tr><td>Номер карты (кат.)</td></tr>
<tr><td>Банк (кат.)</td><td>4.00%</td><td colspan="2" rowspan="2">5.00%</td></tr>
<tr><td>Платёжная система (кат.)</td><td>3.20%</td></tr>
<tr><td>Стоимость анализов, руб (числ.)</td><td>6.72%</td><td>5.79%</td><td>4.12%</td></tr>
<tr><td>Средняя ошибка (кат.)</td><td>4.74%</td><td colspan="2">5.00%</td></tr>
<tr><td>Общая ошибка</td><td>4.90%</td><td>5.07%</td><td>4.93%</td></tr>
<tr><td colspan="5"></td></tr>
<tr><td>ФИО (кат.)</td><td rowspan="4">10%</td><td colspan="3" rowspan="4">10.00%</td></tr>
<tr><td>Паспортные данные (кат.)</td></tr>
<tr><td>СНИЛС (кат.)</td></tr>
<tr><td>Симптомы (кат.)</td></tr>
<tr><td>Выбор врача (кат.)</td><td colspan="3"></td></tr>
</table>

Дата посещения врача (кат.)				
Анализы (кат.)				
Дата получения анализов (кат.)				
Номер карты (кат.)				
Банк (кат.)		7.40%	10.00%	
Платёжная система (кат.)		7.00%		
Стоимость анализов, руб (числ.)		11.70%	12.99%	9.89%
Средняя ошибка (кат.)		9.49%	10.00%	
Общая ошибка		9.68%	10.25%	9.99%
ФИО (кат.)	20%	20.00%		
Паспортные данные (кат.)		19.90%	20.00%	
СНИЛС (кат.)		20.00%		
Симптомы (кат.)				
Выбор врача (кат.)		19.40%	20.00%	
Дата посещения врача (кат.)		20.00%		
Анализы (кат.)				
Дата получения анализов (кат.)				

Номер карты (кат.)				
Банк (кат.)		15.60%	20.00%	
Платёжная система (кат.)		13.00%		
Стоимость анализов, руб (числ.)		31.73%	24.86%	21.17%
Средняя ошибка (кат.)		18.90%	20.00%	
Общая ошибка		19.97%	20.41%	20.10%
ФИО (кат.)	30%	30.00%		
Паспортные данные (кат.)				
СНИЛС (кат.)				
Симптомы (кат.)				
Выбор врача (кат.)		29.10%	30.00%	
Дата посещения врача (кат.)		30.00%		
Анализы (кат.)				
Дата получения анализов (кат.)				
Номер карты (кат.)				
Банк (кат.)		22.50%	30.00%	
Платёжная система (кат.)		18.80%	30.00%	
Стоимость анализов, руб		55.17%	40.62%	36.10%

(числ.)				
Средняя ошибка (кат.)		28.22%	30.00%	
Общая ошибка		30.46%	30.89%	30.51%

Таблица 3. Результаты для среднего датасета (25.000 строк)

Название столбцов	Процент удаления	Хот-Дек	Заполнения средним значением	Заполнение пропусков на основе линейной регрессии.
ФИО (кат.)	3%	3.00%		
Паспортные данные (кат.)				
СНИЛС (кат.)				
Симптомы (кат.)				
Выбор врача (кат.)		2.92%	3.00%	
Дата посещения врача (кат.)		3.00%		
Анализы (кат.)				
Дата получения анализов (кат.)				
Номер карты (кат.)				
Банк (кат.)		2.20%	3.00%	
Платёжная система (кат.)		1.97%		
Стоимость анализов, руб (числ.)		5.62%	4.44%	3.61%
Средняя ошибка		2.83%	3.00%	

(кат.)				
Общая ошибка		3.06%	3.12%	3.05%
ФИО (кат.)	5%	5.00%		
Паспортные данные (кат.)				
СНИЛС (кат.)				
Симптомы (кат.)				
Выбор врача (кат.)		4.92%	5.00%	
Дата посещения врача (кат.)		5.00%		
Анализы (кат.)				
Дата получения анализов (кат.)				
Номер карты (кат.)				
Банк (кат.)		3.74%	5.00%	
Платёжная система (кат.)		3.34%		
Стоимость анализов, руб (числ.)		8.77%	7.18%	5.87%
Средняя ошибка (кат.)		4.73%	5.00%	
Общая ошибка		5.06%	5.18%	5.07%
ФИО (кат.)	10%	10.00%		
Паспортные данные				

(кат.)				
СНИЛС (кат.)				
Симптомы (кат.)				
Выбор врача (кат.)		9.82%	10.00%	
Дата посещения врача (кат.)		10.00%		
Анализы (кат.)				
Дата получения анализов (кат.)				
Номер карты (кат.)				
Банк (кат.)		7.49%	10.00%	
Платёжная система (кат.)		6.66%		
Стоимость анализов, руб (числ.)		16.49%	13.79%	11.16%
Средняя ошибка (кат.)		9.45%	10.00%	
Общая ошибка		10.04%	10.32%	10.10%
ФИО (кат.)	20%	20.00%		
Паспортные данные (кат.)				
СНИЛС (кат.)				
Симптомы (кат.)				
Выбор врача (кат.)		19.61%	20.00%	

Дата посещения врача (кат.)		20.00%		
Анализы (кат.)		19.99%	20.00%	
Дата получения анализов (кат.)		20.00%		
Номер карты (кат.)				
Банк (кат.)		15.05%	20.00%	
Платёжная система (кат.)		13.42%		
Стоимость анализов, руб (числ.)		30.99%	27.30%	22.85%
Средняя ошибка (кат.)		18.92%	20.00%	
Общая ошибка		19.92%	20.61%	20.24%
ФИО (кат.)	30%	30.00%		
Паспортные данные (кат.)				
СНИЛС (кат.)		29.99%	30.00%	
Симптомы (кат.)		30.00%		
Выбор врача (кат.)		29.33%	30.00%	
Дата посещения врача (кат.)		30.00%		
Анализы (кат.)		29.98%	30.00%	
Дата получения анализов (кат.)		30.00%		
Номер карты (кат.)				

Банк (кат.)		22.32%	30.00%	
Платёжная система (кат.)		19.95%		
Стоимость анализов, руб (числ.)		48.28%	41.80%	35.67%
Средняя ошибка (кат.)		28.32%	30.00%	
Общая ошибка		29.99%	30.98%	30.47%

Таблица 4. Результаты для большого датасета (100.000 строк)

Название столбцов	Процент удаления	Хот-Дек	Заполнения средним значением	Заполнение пропусков на основе линейной регрессии.
ФИО (кат.)	3%	3.00%		
Паспортные данные (кат.)				
СНИЛС (кат.)				
Симптомы (кат.)				
Выбор врача (кат.)		2.93%	3.00%	
Дата посещения врача (кат.)		3.00%		
Анализы (кат.)				
Дата получения анализов (кат.)				
Номер карты (кат.)				

Банк (кат.)		2.29%	3.00%	
Платёжная система (кат.)		2.00%		
Стоимость анализов, руб (числ.)		4.69%	4.15%	3.10%
Средняя ошибка (кат.)		2.84%	3.00%	
Общая ошибка		2.99%	3.10%	3.01%
ФИО (кат.)	5%	5.00%		
Паспортные данные (кат.)				
СНИЛС (кат.)				
Симптомы (кат.)				
Выбор врача (кат.)		4.89%	5.00%	
Дата посещения врача (кат.)		5.00%		
Анализы (кат.)				
Дата получения анализов (кат.)				
Номер карты (кат.)				
Банк (кат.)		3.71%	5.00%	
Платёжная система (кат.)		3.43%		
Стоимость анализов, руб		8.72%	7.16%	5.84%

(числ.)				
Средняя ошибка (кат.)		4.73%	5.00%	
Общая ошибка		5.06%	5.18%	5.07%
ФИО (кат.)	10%	10.00%		
Паспортные данные (кат.)				
СНИЛС (кат.)				
Симптомы (кат.)				
Выбор врача (кат.)		9.82%	10.00%	
Дата посещения врача (кат.)		10.00%		
Анализы (кат.)				
Дата получения анализов (кат.)		9.99%	10.00%	
Номер карты (кат.)		10.00%		
Банк (кат.)		7.55%	10.00%	
Платёжная система (кат.)		6.64%		
Стоимость анализов, руб (числ.)		15.51%	13.45%	10.94%
Средняя ошибка (кат.)		9.46%	10.00%	
Общая ошибка		9.96%	10.29%	10.08%

ФИО (кат.)	20%	20.00%		
Паспортные данные (кат.)				
СНИЛС (кат.)				
Симптомы (кат.)				
Выбор врача (кат.)		19.61%	20.00%	
Дата посещения врача (кат.)		20.00%		
Анализы (кат.)		19.99%	20.00%	
Дата получения анализов (кат.)		20.00%		
Номер карты (кат.)				
Банк (кат.)		14.95%	20.00%	
Платёжная система (кат.)		13.19%		
Стоимость анализов, руб (числ.)		31.95%	27.03%	22.40%
Средняя ошибка (кат.)		18.89%	20.00%	
Общая ошибка		19.97%	20.59%	20.20%
ФИО (кат.)	30%	30.00%		
Паспортные данные (кат.)				

СНИЛС (кат.)				
Симптомы (кат.)				
Выбор врача (кат.)		29.32%	30.00%	
Дата посещения врача (кат.)		30.00%		
Анализы (кат.)		29.98%	30.00%	
Дата получения анализов (кат.)		30.00%		
Номер карты (кат.)				
Банк (кат.)		22.70%	30.00%	
Платёжная система (кат.)		19.99%	30.00%	
Стоимость анализов, руб (числ.)		50.27%	42.11%	36.14%
Средняя ошибка (кат.)		28.36%	30.00%	
Общая ошибка		30.19%	31.01%	30.51%

Общая характеристика методов

В исследовании рассматривались три метода обработки пропущенных данных: Хот-Дек, заполнение средним значением и восстановление с помощью линейной регрессии. Анализ проводился на датасетах различного объема (1.000, 25.000 и 100.000 записей) при разных уровнях пропусков (3-30%). Особое внимание уделялось точности восстановления как категориальных, так и числовых данных, а также временным характеристикам методов.

Анализ категориальных данных

Для категориальных переменных наблюдались следующие закономерности:

1. Метод Hot-Deck:

- Продемонстрировал высокую точность для столбцов с ограниченным набором значений (Банк, Платёжная система), где ошибка составляла 2.2-2.3% при 3% пропусков против 3% у других методов.
- Для остальных категориальных переменных точность соответствовала проценту удаленных данных.
- Эффективность снижалась при увеличении доли пропусков (до 13-15% ошибки при 20% пропусков).

2. Заполнение модой и линейная регрессия не работают для категориальных данных.

Анализ числовых данных

Для числового столбца («Стоимость анализов») результаты существенно различались:

1. Метод Hot-Deck:

- Показал наихудшие результаты с ошибкой до 55.17% (30% пропусков, малый датасет)
- Время выполнения: менее 1 секунды для всех объемов данных

2. Заполнение средним:

- Обеспечил среднюю точность с ошибкой 40.62% (30% пропусков, малый датасет)
- Время выполнения: менее 1 секунды для всех объемов данных

3. Линейная регрессия:

- Продемонстрировала наилучшие результаты (36.10% ошибки при 30% пропусков)
- Время выполнения значительно возрастало с увеличением объема данных.

Влияние объема данных и процента пропусков

Анализ показал следующие зависимости:

1. Для малого датасета (1 000 строк):

При 3% пропусков:

- Hot-Deck: 3.06% общей ошибки
- Mean: 3.13%
- Regression: 3.08%

При 30% пропусков:

- Hot-Deck: 30.46%
- Mean: 30.89%
- Regression: 30.51%

2. Для среднего датасета (25 000 строк):

Линейная регрессия показала улучшение точности (35.67% против 36.10% для малого датасета при 30% пропусков).

3. Для большого датасета (100 000 строк):

Точность линейной регрессии осталась стабильной (36.14% ошибки).

Выводы и рекомендации

1. Для категориальных данных:
   - Рекомендуется использовать метод Hot-Deck для столбцов с ограниченным набором значений
   - Для остальных категориальных данных заполнение не имеет никакого осмысла.
2. Для числовых данных:
   - Линейная регрессия демонстрирует наилучшую точность, но требует значительных вычислительных ресурсов для больших датасетов.
   - Заполнение средним значением может служить компромиссным вариантом при ограниченных ресурсах.
3. Общие рекомендации:
   - Для комплексной обработки данных оптимальным является комбинированный подход:
     - Hot-Deck для категориальных данных с повторяющимися значениями
     - Линейная регрессия для числовых переменных

- При работе с большими объемами данных (100 000+ строк) необходимо учитывать временные затраты на восстановление

Результаты исследования подтверждают, что выбор метода восстановления пропущенных данных должен осуществляться с учетом типа переменных, объема данных и допустимого уровня временных затрат.

Вывод

В рамках работы был реализована программа для восстановления данных тремя методами: Хот-Дек, метод заполнения средним значением и заполнение пропусков на основе линейной регрессии. Также был разработан удобный графический интерфейс. В ходе сравнительного анализа этих методов на датасетах различного размера 5000, 25.000 и 100.000 строк с разным процентом удалённых данных 3%, 5%, 10%, 20% и 30%, было выявлено, что заполнение Хот-Дек является предпочтительным методом для большинства случаев благодаря своей простоте, скорости и стабильной точности, но самым оптимальным является комбинированный подход: Хот-Дек для категориальных данных с повторяющимися значениями и линейная регрессия для числовых переменных.

Источники

1. tkinter — Python interface to Tcl/Tk // URL: https://docs.python.org/3/library/tkinter.html (дата обращения: 14.03.2025).
2. NumPy documentation // URL: https://numpy.org/doc/stable/index.html (дата обращения: 14.05.2025).
3. pandas documentation // URL: https://pandas.pydata.org/docs/index.html (дата обращения: 14.05.2025).
4. os — Miscellaneous operating system interfaces // URL: https://docs.python.org/3/library/os.html (дата обращения: 15.05.2025).
5. Linear Models — scikit-learn documentation// URL: https://docs.python.org/3/library/random.html (дата обращения: 16.04.2025).