

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«Национальный исследовательский ядерный университет «МИФИ»
Обнинский институт атомной энергетики –
филиал федерального государственного автономного образовательного учреждения высшего
образования «Национальный исследовательский ядерный университет «МИФИ»
(ИАТЭ НИЯУ МИФИ)

Отделение Интеллектуальные кибернетические системы

**Выпускная квалификационная работа —
бакалаврская работа**

по направлению подготовки **09.03.02 Информационные системы и технологии**
Направленность (профиль) **Информационные технологии**

**«Разработка приложения для автоматизированного
статистического анализа данных, полученных при
проведении КГО стендовым методом на реакторах
типа ВВЭР»**

Выполнил:

студент гр. ИС2-Б20

_____ Костевич А.Е.

Руководитель ВКР,
старший преподаватель ОИКС

_____ Радаев А.В.

Нормоконтроль
доцент отделения ИКС, к.ф.-м.н.

_____ Качанов Б.В.

Выпускная квалификационная
работа допущена к защите

Руководитель
образовательной программы
09.03.02 Информационные системы
и технологии
канд. тех. наук

_____ Мирзеабасов О.А.

Обнинск, 2024 г

РЕФЕРАТ

Работа 41 стр., 1 табл., 12 рис., 8 ист.

РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ АВТОМАТИЗИРОВАННОГО СТАТИСТИЧЕСКОГО АНАЛИЗА ДАННЫХ, ПОЛУЧЕННЫХ ПРИ ПРОВЕДЕНИИ КГО СТЕНДОВЫМ МЕТОДОМ ДЛЯ РЕАКТОРОВ ТИПА ВВЭР.

Объектом разработки является настольное приложение с графическим интерфейсом, поддерживающее функции визуализации, экспорта, а также статистического анализа данных.

Целью данной работы является разработка программного обеспечения для повышения точности и упрощения процесса анализа данных, полученных при проведении КГО в пеналах СОДС.

Задачами ВКР являются изучение и анализ метода проведения КГО, выбор критериев для анализа данных, выбор и освоение инструментов разработки, реализация алгоритмов анализа данных, а также разработка приложения.

В работе производится подробный анализ существующего подхода к проведению процедуры и анализу данных КГО на реакторах типа ВВЭР, приводятся предложения по улучшению и автоматизации этого подхода, а также описывается процесс разработки программного обеспечения с учётом внесённых предложений. Разработанное приложение было успешно апробировано на Нововоронежской АЭС.

ОПРЕДЕЛЕНИЯ

Реперный радионуклид — радионуклид, по выходу которого из твэла в теплоноситель первого контура судят о герметичности оболочки твэла, так как он обладает ядерно-физическими и химическими характеристиками, позволяющими надежно регистрировать его в условиях эксперимента.

Негерметичный твэл - твэл, в оболочке которого имеется повреждение, приводящее к выходу продуктов деления из него.

Негерметичная ТВС — ТВС, в составе которой имеются негерметичные твэлы.

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

БВ — Бассейн выдержки.

ВВЭР — Водо-водяной энергетический реактор.

КГО — Контроль герметичности оболочек.

ЛПР — Лицо, принимающее решение.

ПД — Продукты деления.

ПК — Продукты коррозии.

СОДС — Система обнаружения дефектных сборок.

ТВС — Тепловыделяющая сборка.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Обзор существующей методики проведения процедуры КГО	4
стендовым методом	4
1.1 Основные положения	4
1.2 Процедура проведения КГО стендовым методом	4
1.3 Обработка результатов	5
1.4 Учет выгорания топлива ТВС	10
2 Предложения по улучшению методики обработки данных	12
2.1 Проблемы существующего подхода	12
2.2 Предложения по улучшению	13
2.2.1 Метод IQR	13
2.2.2 Критерий Шапиро-Уилка	15
3 Проектирование приложения с учётом внесённых предложений	16
3.1 Требования к проектируемому приложению	16
3.2 Архитектура приложения	17
3.3 Пользовательский сценарий использования	17
4 Разработка приложения	19
4.1 Обзор инструментов разработки	19
4.1.1 Python	19
4.1.2 Pandas + NumPy	19
4.1.3 SciPy	20
4.1.4 Matplotlib+Seaborn	21
4.1.5 PyQt	21
4.2 Импорт данных	22
4.3 Поиск выбросов	25
4.4 Пользовательский интерфейс	27
4.4.1 Основные окна приложения	27
4.4.2 Окно импорта файла	28

4.4.3	Интеграция Matplotlib в качестве QWidget	29
4.4.4	Главное окно приложения	31
4.4.5	Окно анализа выборок	32
4.5	Экспорт таблиц	34
ЗАКЛЮЧЕНИЕ		37
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ		38
Приложение А		39

ВВЕДЕНИЕ

Атомные электростанции играют ключевую роль в современной энергетике. Однако сопутствующие ядерной энергетике риски требуют непрерывного совершенствования методов контроля и обслуживания ядерных установок.

В частности, одним из значимых аспектов эксплуатации ядерных реакторов является контроль герметичности оболочек тепловыделяющих элементов. В настоящее время анализ данных, полученных при проведении КГО, частично осуществляется в ручном режиме, что требует значительных ресурсов времени и труда. Более того, этот подход подвержен человеческим ошибкам и может ограничивать возможности в проведении анализа данных с высокой точностью и скоростью.

Как известно, одним из недостатков реактора типа ВВЭР является невозможность перегрузки топлива без остановки реактора и ошибка, допущенная при принятии решения относительно герметичности ТВС, может повлечь за собой существенные экономические издержки.

Цель настоящей работы заключается в разработке программного обеспечения, работа которого направлена на повышение эффективности и достоверности результатов КГО, а также снижение трудовых затрат.

В данной работе будет проведен обзор существующего метода обработки результатов КГО, приведены предложения по его автоматизации, а также описан процесс создания прототипа программного обеспечения.

1 Обзор существующей методики проведения процедуры КГО стендовым методом

1.1 Основные положения

В данной работе рассматривается метод КГО в пеналах СОДС [1], который является одним из наиболее надёжных способов определения негерметичных ТВС. СОДС входит в состав обязательного оборудования всех действующих и проектируемых АЭС с реактором ВВЭР.

Метод основан на измерении утечки ПД из-под оболочек твэлов путем гамма-спектрометрического анализа изотопного состава проб воды, отбираемой из контура циркуляции СОДС, по активностям реперных радионуклидов ^{131}I , ^{134}Cs , ^{136}Cs , ^{137}Cs и ^{133}Xe . Инициирование выхода радионуклидов в воду стенда КГО осуществляется посредством изменения давления в процессе выдержки ТВС в этой воде — настаивания.

1.2 Процедура проведения КГО стендовым методом

1. Процедура проведения КГО начинается проведения испытаний для каждой ТВС в пеналах СОДС с последующим отбором проб воды.

Проверка ТВС проводится при циркуляции воды по контуру стенда КГО без ее замены и состоит из двух циклов:

- Настаивание ТВС при избыточном (верхнем) давлении в контуре от $4,5 \cdot 10^5$ Па до $6,0 \cdot 10^5$ Па продолжительностью 5 минут.
- Настаивание ТВС при избыточном (нижнем) давлении в контуре от $1,0 \cdot 10^5$ Па до $1,5 \cdot 10^5$ Па до полного перемешивания (не менее 15 минут).

С целью соблюдения одинаковых условий испытаний требуется, чтобы значения верхнего и нижнего избыточного давления были одинаковыми при проверке всех ТВС.

2. После завершения настаивания ТВС производится отбор пробы воды из контура стенда КГО.

3. В каждой j -ой пробе воды, взятой из стенда КГО при испытании j -ой ТВС, на спектрометрической установке измеряются значения удельной активности и приводятся на момент останова реактора:

- $A_{j, \text{КГО}}^i$ — реперных i -х радионуклидов продуктов деления (^{131}I , ^{134}Cs , ^{136}Cs , ^{137}Cs и ^{133}Xe)
- $A_{j, \text{КГО}}^{i'}$ — радионуклида продуктов коррозии(ПК) (^{54}Mn или ^{58}Co , ^{60}Co , ^{51}Cr , ^{59}Fe).

4. Для учета фоновой активности радионуклидов йода, цезия и продуктов коррозии периодически производится измерение их активности в воде, подаваемой в стенд КГО (с каждой вновь приготовленной порцией раствора борной кислоты на СВО), и в бассейне выдержки (один раз в сутки).

5. Проверка фоновой составляющей за счет загрязнения стенда радиоактивными продуктами (холостая проба) производится перед началом работ по КГО, а также периодически (не реже одного раза в сутки). Для этого без загрузки ТВС в пенал проводятся все операции по промывке контура и настаиванию с отбором и анализом пробы.

6. Итогом проведения спектрометрического анализа проб воды является таблица значений, в которых для каждой j -ой ТВС приводятся в соответствие значения активности $A_{j, \text{КГО}}^i$ каждого из регистрируемых реперных радионуклидов и $A_{j, \text{КГО}}^{i'}$ продуктов коррозии. Статистический анализ результатов измерения проводится для ТВС, в пробах которых значимо регистрировались ПД. Результаты измерений ТВС, при проверке которых реперные ПД не регистрировались, из статистического расчета исключаются.

1.3 Обработка результатов

1. Анализ герметичности ТВС, согласно [1], основан на выборочном поиске выбросов методом "трёх сигм".

2. Основными реперными радионуклидами, по которым устанавливается наличие(отсутствие) негерметичных твэлов в ТВС являются ^{131}I , ^{134}Cs , ^{137}Cs . Наличие в контролируемой пробе ^{136}Cs и(или) ^{133}Xe , значимо

превышающих их содержание в холостых пробах, является однозначным основанием для включения ТВС в список подозрительных, требующих, как минимум, дополнительной проверки.

3. Полученные значения представляются в графическом виде в такой хронологической последовательности, в какой ТВС проверялись в стенде КГО. Примеры графического представления результатов КГО приведены на рисунках 1 и 2. На основании визуального анализа этих данных на графике может быть сделано заключение, относятся ли они к одному статистическому распределению. Таким способом проводится оценка соблюдения одинаковых условий проверки всех ТВС. Если условия менялись с течением времени (на практике так происходит почти всегда), то производится разделение исходных данных на выборки, которые относятся к одному статистическому распределению.

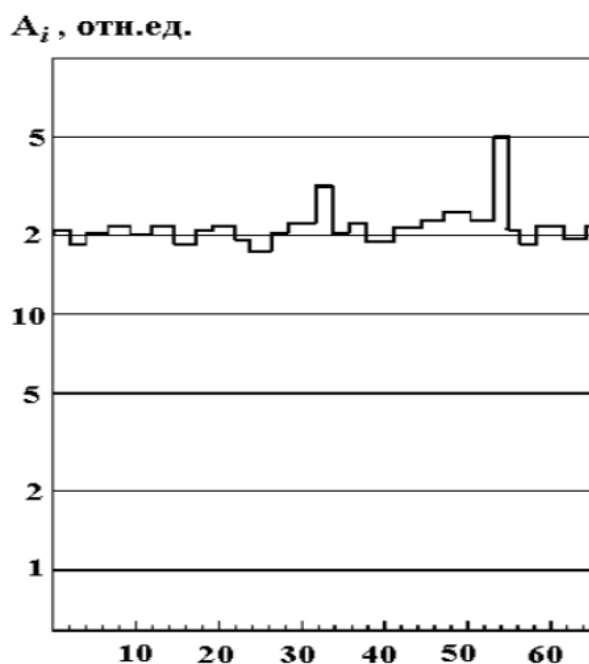


Рисунок 1 – Графическое хронологическое представление данных, принадлежащих к одному распределению. [1]

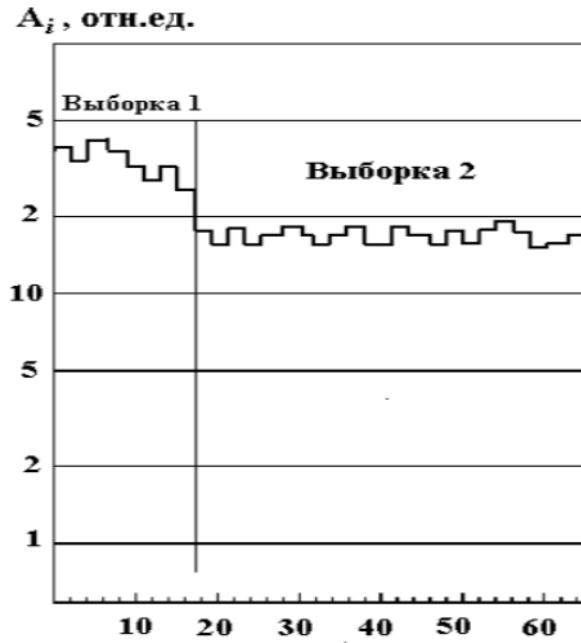


Рисунок 2 – Графическое хронологическое представление данных, принадлежащих к различным распределениям. [1]

4. Для каждой полученной совокупности данных, относящихся к одному и тому же статистическому распределению, вычисляются $\bar{A}_{\text{кго}}^i$ — среднеарифметические значения удельной активности радионуклидов ^{131}I (и ^{134}Cs , ^{136}Cs , ^{137}Cs , ^{133}Xe) и $\bar{A}_{\text{кго}}^{i'}$ — среднеарифметическое значение удельной активности ^{54}Mn (или ^{58}Co , ^{60}Co , ^{51}Cr , ^{59}Fe), по формулам 1 и 2:

$$\bar{A}_{\text{кго}}^i = \frac{1}{N} \sum_{j=1}^N A_{j,\text{кго}}^i. \quad (1)$$

$$\bar{A}_{\text{кго}}^{i'} = \frac{1}{N} \sum_{j=1}^N A_{j,\text{кго}}^{i'}. \quad (2)$$

Кроме того, рассчитывают соответствующие им среднеквадратичные отклонения (стандартные статистические неопределенности) по формулам 3 и 4:

$$\sigma_{A_{\text{кго}}^i} = \sqrt{\frac{1}{N-1} \sum_{j=1}^N (A_{j,\text{кго}}^i - \bar{A}_{\text{кго}}^i)^2}, \quad (3)$$

$$\sigma_{A_{\text{кго}}^{i'}} = \sqrt{\frac{1}{N-1} \sum_{j=1}^N (A_{j,\text{кго}}^{i'} - \bar{A}_{\text{кго}}^{i'})^2}, \quad (4)$$

где N — количество проверенных ТВС.

5. Если $N > 10$, то ТВС, для которых выполняется условие 5 являются герметичными.

$$A_{j,\text{кго}}^i \leq \bar{A}_{\text{кго}}^i + 3 * \sigma_{A_{\text{кго}}^i}. \quad (5)$$

ТВС, для которых одновременно выполняются условия 6 и 7 являются негерметичными.

$$A_{j,\text{кго}}^i > \bar{A}_{\text{кго}}^i + 3 * \sigma_{A_{\text{кго}}^i}. \quad (6)$$

$$A_{j,\text{кго}}^{i'} \leq \bar{A}_{\text{кго}}^{i'} + 3 * \sigma_{A_{\text{кго}}^{i'}}. \quad (7)$$

Важно отметить, что активности радионуклидов ПК измеряются с целью учёта при анализе данных. ПК, образующиеся в конструкционных материалах реактора по мере эксплуатации, переносятся по теплоносителю и могут откладываться на ТВС, что влечёт за собой повышение активности в том числе и реперных ПД [2]. Именно поэтому повышение активности реперных ПД совместно с активностями ПК может являться признаком некачественной отмывки ТВС при подготовке к проведению испытаний.

6. Если количество ТВС в выборке $N < 10$, то в формулах 5-7 в качестве коэффициента при σ вместо значения 3 используются коэффициенты Стьюдента, приведенные в таблице 1, для доверительной вероятности 0,95.

Таблица 1 – Значения коэффициента Стьюдента в зависимости от количества проверенных ТВС и вероятности, с которой ТВС могут быть отнесены к разряду имеющих негерметичные твэлы

Кол-во ТВС	0,95	0,99	0,999
2	12,7	66,7	637
3	4,30	9,93	31,6
4	3,18	5,84	12,9
5	2,78	4,60	8,61
6	2,57	4,03	6,86
7	2,45	3,71	5,96
8	2,36	3,50	5,41
9	2,31	3,36	5,04
10	2,26	3,25	4,78

7. Заключение о герметичности ТВС, для которых при выполнении условия 7 условие 6 выполняется не для всех основных реперных ПД(^{131}I , ^{134}Cs , ^{137}Cs), производится с учетом дополнительной информации: наличие (величина удельной активности) в пробах КГО других реперных ПД(^{136}Cs , ^{133}Xe), а также соотношения активности ^{134}Cs и ^{137}Cs с учётом выгорания.

8. Если в результате вычислений выявлены ТВС, содержащие твэлы с негерметичными оболочками, то проводится повторный расчет величин по формулам 1-4 и проверка по условиям 6 и 7 для остальных ТВС.

9. Повторение расчетов и проверок производится до тех пор, пока все ТВС, включаемые в повторную проверку, не будут удовлетворять условию 5.

10. После завершения последовательно проведенных расчетов и проверок повторный КГО твэлов проводится для следующих ТВС:

- для которых выполняется условие 6 для основных реперных ПД и одновременно не выполняется условие 7;
- для которых выполняется условие 6, но проверенных сразу после ТВС, для которых также выполняется условие 6.
- для которых выполняются условие 6 и условие 7, но удельная активность ^{134}Cs и ^{137}Cs в пробе не превышала $7,4 \cdot 10^4$ Бк/кг ($2 \cdot 10^{-6}$ Ки/кг).

11. Результат повторной проверки включается в анализируемую совокупность вместо первичного, если он качественно отличается от него.

1.4 Учет выгорания топлива ТВС

Радионуклид ^{137}Cs ($t_{1/2}=30$ лет) является конечным продуктом радиоактивного бета-распада предшественников в цепочке радиоактивного распада. Влияние других мод образования ^{137}Cs несущественно, так что количество атомов этого радионуклида в топливе и под оболочкой ТВЭЛ оказывается прямо пропорционально выгоранию.

Радионуклид ^{134}Cs ($t_{1/2}=2,06$ лет) образуется преимущественно за счет реакции (n, γ) на ядрах стабильного изотопа ^{133}Cs , который является последним членом цепочки радиоактивного распада и накапливается в топливе практически линейно с выгоранием.

Поскольку образование ^{134}Cs из ^{133}Cs в первом приближении линейно с выгоранием, общее количество ^{134}Cs оказывается близким к квадратичной функции от выгорания, естественно, с учетом радиоактивного распада. Таким образом, соотношение активностей ^{134}Cs и ^{137}Cs в топливе и под оболочкой оказывается значимо зависящим от выгорания. При проведении КГО на остановленном реакторе в пеналах СОДС определение соотношения удельных активностей радионуклидов ^{134}Cs и ^{137}Cs в пробах КГО может быть использовано в качестве индикатора для дополнительной проверки факта обнаружения ТВС с негерметичными ТВЭЛами.

1. Для ТВС, в которых при выполнении условия (7) условие (6) выполняется не для всех контролируемых реперных ПД, строится коэффициент, равный отношению отношения удельных активностей ^{134}Cs и ^{137}Cs в j-ой пробе стенда КГО:

$$K_{\text{кго}}^j = \frac{A_{j,\text{кго}}^{134\text{Cs}}}{A_{j,\text{кго}}^{137\text{Cs}}}. \quad (8)$$

На основе сопоставления определенных по формуле (8) коэффициентов $K_{\text{кго}}^j$ - соотношений измеренных удельных активностей ^{134}Cs и ^{137}Cs в пробах от ТВС, определенных как негерметичные, с кривой, приведенной на рисунке 3 определяются расчетные выгорания топлива ТВС с негерметичными ТВЭЛами.

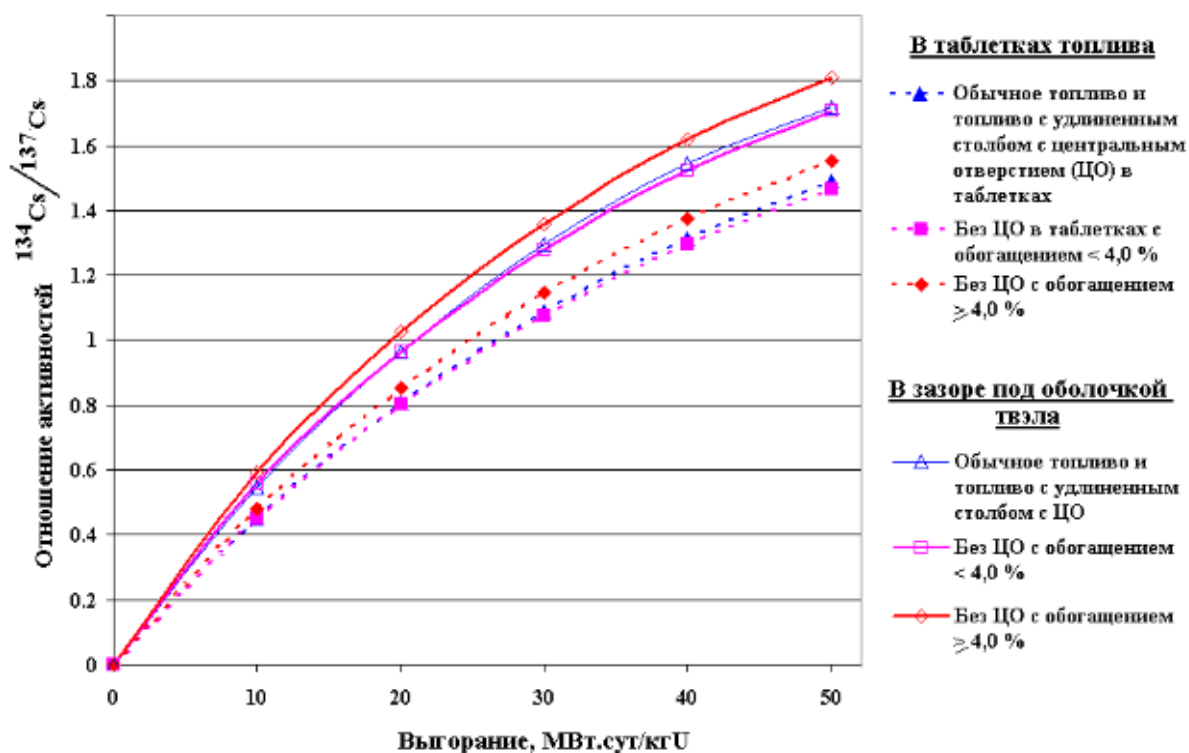


Рисунок 3 – Расчетные соотношения активностей радионуклидов ^{134}Cs и ^{137}Cs в топливе и в зазоре под оболочкой твэлов в функции выгорания для типичных историй облучения топлива реакторов ВВЭР-1000. [1]

2. Совпадение (с учетом неопределенностей) полученных таким образом выгораний топлива с выгораниями топлива этих же ТВС по данным физических расчетов (с учетом неравномерности энерговыделения и выгорания по высоте и радиусу кассеты) является дополнительным фактом, подтверждающим наличие негерметичных твэлов в составе контролируемых ТВС.

2 Предложения по улучшению методики обработки данных

2.1 Проблемы существующего подхода

Рассмотрев методику проведения процедуры КГО согласно[1] можно выделить несколько замечаний, которые можно пересмотреть:

1. Метод анализа данных, приведённый в параграфе 1.3, основывается на поиске выбросов по правилу "3 сигм". Данное правило утверждает, что абсолютная величина отклонения нормально распределённой случайной величины от её математического ожидания не превосходит трёх среднеквадратичных отклонений с вероятностью[3]:

$$P(|X - m| < 3\sigma) = 0,9973. \quad (9)$$

Проблема этого метода заключается в том, что он применим для выборок, значения которых извлечены из нормально распределённых генеральных совокупностей. Но в случае проведения КГО(согласно разделу 1) нет достаточных оснований утверждать, что все значения активностей будут распределены по нормальному закону. Следовательно, требуется проверить характер распределения для каждой выборки с целью установления корректности применения метода "3 сигм".

Кроме того, среднее и среднеквадратическое отклонение, рассчитываемые в данном методе, также изменяются под воздействием аномальных значений, что приводит к маскировке выбросов[4]. Следовательно, имеет смысл рассмотреть альтернативы, например метод межквартильного размаха(IQR).

2. Процедура КГО с учётом времени и объёма испытаний может проходить до нескольких недель. С течением времени в БВ, а также в воде, подаваемой на стенд КГО, может изменяться концентрация борной кислоты с целью борного регулирования, что негативно сказывается на однородности условий проведения испытаний. В связи с этим значения активностей ПД, полученные в разное время, могут принадлежать разным статистическим распределениям, следовательно, анализироваться должны отдельно.

Согласно пункту 1.3.3, разделение на выборки происходит "На основании визуального анализа" графических данных. Хочу отметить, что в изучаемой методике существует способ анализа полученных выборок на принадлежность к одному статистическому распределению с целью объединения нескольких выборок, но он не учитывает анализ на корректность разбиения исходных данных. В связи с вышеперечисленным возникает необходимость проверки данных в выборках, полученных на основании визуального анализа.

3. Описанный процесс анализа данных производится в ручном режиме с использованием программного комплекса Excel. Автоматизация этого процесса позволит снизить вероятность ошибок, ведущих к преждевременной остановке реактора, а также снизить потребность во временных и трудовых затратах.

2.2 Предложения по улучшению

2.2.1 Метод IQR

Учитывая изложенное в 2.1.1, в качестве альтернативы методу "3 сигм" имеет смысл рассмотреть метод поиска выбросов с помощью межквартильного размаха (Interquartile range/IQR).

Межквартильный размах (IQR) — это статистическая мера, равная разности между первым и третьим квартилями распределения (25-м и 75-м процентилями). Первый квартиль (Q_1) — это значение, ниже которого находится 25% данных, а третий квартиль (Q_3) — это значение, ниже которого находится 75% данных. Можно так же сказать, что интерквартильный размах это половина выборки, центрированная относительно медианы.

Этот показатель полезен для оценки изменчивости признака в асимметричных распределениях или наборах данных с выбросами. Интерквартильный размах является устойчивым (робастным) аналогом дисперсии, поскольку не подвержен влиянию аномальных значений. Именно поэтому он является наиболее перспективным аналогом методу "3 сигм". Таким образом, интерквартильный размах позволяет обнаруживать аномальные значения, являясь альтернативой среднеквадратическому отклонению, ко-

торое эффективно только для нормально распределенных данных. IQR — это непараметрическая оценка, что делает его применимым для анализа данных любого распределения.

В современных практиках поиска выбросов с использованием межквартильного размаха применяется следующий метод расчёта критических значений:

$$X^{min} = Q1 - 1.5 * IQR; \quad (10)$$

$$X^{max} = Q3 + 1.5 * IQR; \quad (11)$$

На рисунке 4 приведена иллюстрация коэффициента IQR для нормального распределения.

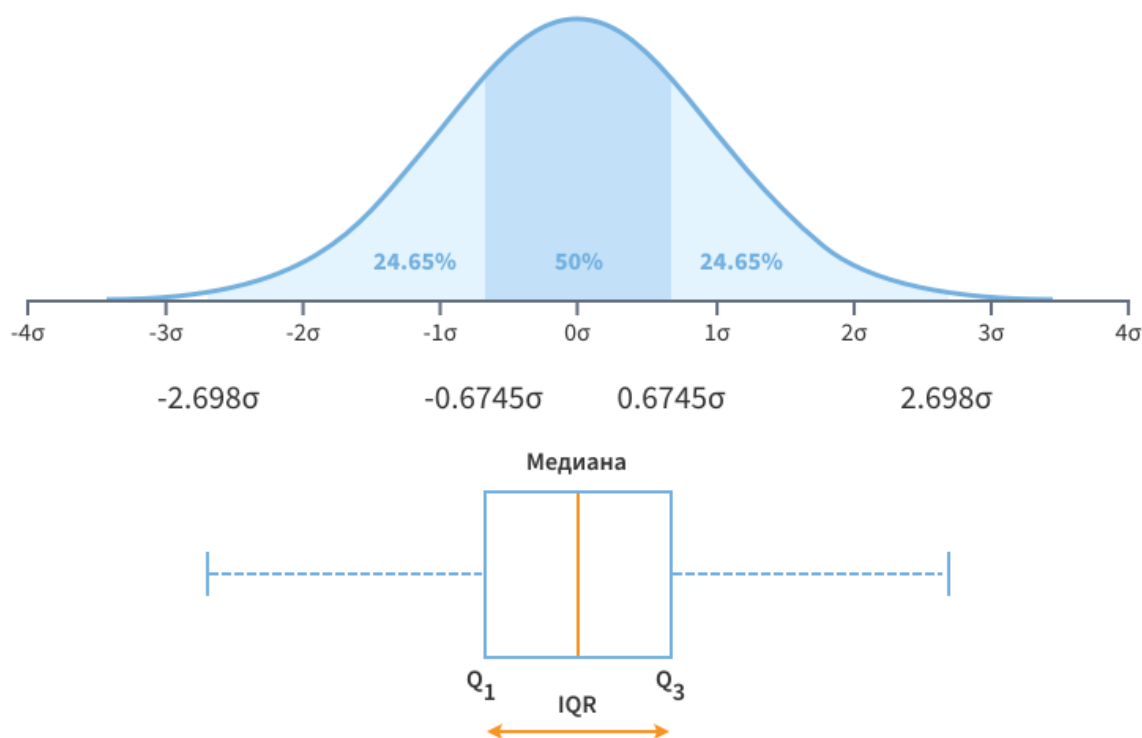


Рисунок 4 – Графическое представление межквартильного размаха на примере нормального распределения.

Из рисунка 4 видно, что данный подход в случае нормального рас-

пределения охватывает интервал:

$$P(|X^{min/max} - X| < 2.698\sigma) = 0,9545. \quad (12)$$

2.2.2 Критерий Шапиро-Уилка

Как было описано выше, применение метода "3 сигм" обосновано в случаях, когда распределение выборки близко к нормальному. Следовательно, требуется установить насколько сильно анализируемое распределение отклоняется от нормального закона. На основании результата данной проверки принимается решение относительно выбора метода поиска выбросов.

В качестве критерия проверки распределения на нормальность был выбран критерий Шапиро-Уилка. Данный критерий был выбран на основании его мощности для выборок низкого объёма ($8 < n < 50$) [5].

Критерий Шапиро-Уилка, базируется на анализе линейной комбинации разностей ранговых статистик. При построении статистики для вариационного ряда $X_{(1)} \leq X_{(2)} \leq X_{(3)} \leq X_{(4)} \leq \dots \leq X_{(n)}$, полученного по наблюдаемой выборке X_1, X_2, \dots, X_n , вычисляют величину [6]:

$$S = \sum_k \alpha_k [X_{(n+1-k)} - X_{(k)}], \quad (13)$$

где индекс k изменяется от 1 до $n/2$ или от 1 до $(n-1)/2$ при четном и нечетном n соответственно. Коэффициенты α_k приведены в [5]. Статистика критерия вычисляется по формуле:

$$W = \frac{S^2}{\sum_{i=1}^n (X_i - \bar{X})^2}. \quad (14)$$

Таблица критических значений p -квантилей для статистики W приводятся в [7].

Данный критерий проверяет следующие гипотезы:

- Гипотеза H_0 : "Случайная величина X распределена нормально".
- Гипотеза H_1 : "Распределение случайной величины X не соответствует нормальному закону".

3 Проектирование приложения с учётом внесённых предложений

3.1 Требования к проектируемому приложению

С целью обеспечения наибольшей точности и минимизации рисков, связанных с возможностью некорректного принятия решений относительно герметичности ТВС, предлагаю разработать приложение для обработки и анализа данных, полученных при КГО стендовым водным методом на реакторе типа ВВЭР, которое призвано помочь лицу, принимающему решения относительно герметичности ТВС. Основной задачей данного приложения является автоматизированная (под контролем ЛПР) обработка и анализ данных. Исходные данные хранятся в табличном виде в формате ODS. Под обработкой и анализом данных понимается разделение исходных данных на выборки, принадлежащие к одному статистическому распределению, произведение всех необходимых расчётов, а также принятие решения относительно герметичности ТВС. После предварительного анализа методики, описанной в главе 1, а также учитывая изложенное в главе 2, могу выдвинуть следующие требования к приложению:

1. Приложение должно наглядно демонстрировать ЛПР основания принятия решения относительно герметичности ТВС, т.е. иметь обширный функционал для визуализации и экспорта графиков распределений, негерметичных ТВС и т.д. Исходя из этого был выбран формат настольного приложения с графическим интерфейсом.
2. Приложение должно иметь функционал для проверки выборок на нормальность с целью определения метода поиска выбросов.
3. Приложение должно поддерживать поиск выбросов методами "IQR" и "3 сигма". По усмотрению ЛПР может применяться любой метод в зависимости от ситуации.
4. Приложение должно иметь возможности для экспорта всех преобразованных данных, на основании которых принимались решения о герметичности, в исходный ODS формат.

3.2 Архитектура приложения

На данном этапе проектирования выделяются независимые модули, из которых будет состоять проектируемое приложение:

1. Модуль пользовательского интерфейса, который отвечает за создание графического интерфейса, позволяющего пользователям взаимодействовать с другими модулями системы. Данный модуль поддерживает функции работы с графиками, в том числе редактирование, создание легенды, а также сохранение в формате картинки для последующего использования в отчётах о проведении КГО.

2. Модуль предварительной обработки(подготовки), который отвечает за импорт и очистку исходных данных.

3. Модуль обработки и анализа данных, который отвечает за проведение расчётов и статистического анализа, полученных после предварительной обработки. Включает в себя выполнение статистических тестов, поиск выбросов, а также экспорт полученных данных в .ods формат.

3.3 Пользовательский сценарий использования

Ниже приведены шаги, которые будет выполнять пользователь при работе с проектируемым приложением:

1. Импорт данных: пользователь выбирает файл в формате .ods с входными данными для анализа.

2. Анализ входных данных: перед пользователем открывается окно, в котором имеется возможность построения и гибкого редактирования графиков по входным данным. Согласно, 1.3.3 на основании визуального анализа графиков, пользователь разделяет входные данные на выборки.

3. Анализ выборок: после разбиения входных данных возникает возможность выполнения статистических тестов для каждой выборки. Пользователь выбирает выборку и анализируемую величину, после чего запускает статистические тесты. Результаты тестов выводятся в отдельном окне для каждой выборки и анализируемой величины. На основании результатов тестов, пользователь делает заключение о корректности разбиения входных данных и возможности дальнейшего анализа.

В случае, если результаты статистических тестов не позволяют проводить дальнейший поиск выбросов в выборках, окно с выборками закрывается и шаги 2-3 повторяются до тех пор, пока все выборки не будут удовлетворять условиям, позволяющим проводить поиск выбросов.

4. После проведения статистических тестов для выборок пользователь запускает поиск выбросов в выборках согласно выбранному методу. Результатом поиска является 3 таблицы:

- Таблица негерметичных ТВС
- Таблица ТВС, для которых необходимо провести повторную проверку
- Таблица параметров, на основании которых принималось заключение о герметичности

5. После проведения анализа пользователь экспортирует полученные таблицы в .ods для составления отчёта о проведении КГО.

4 Разработка приложения

4.1 Обзор инструментов разработки

Для разработки приложения на основании требований, описанных в 3.1, были выбраны следующие инструменты:

4.1.1 Python

Язык программирования Python — высокоуровневый, интерпретируемый, объектно-ориентированный язык программирования, который широко используется для разработки веб-приложений, научных вычислений, анализа данных, искусственного интеллекта, автоматизации задач и многих других областей. Он отличается простотой синтаксиса и легкостью в обучении, что делает его отличным выбором как для начинающих, так и для опытных программистов. В совокупности с большим набором пользовательских библиотек, python предоставляет мощные инструменты для обработки, анализа и визуализации данных, что делает его идеальным выбором для научных и инженерных проектов.

В качестве альтернативы Python рассматривался язык R. R предоставляет более широкий спектр функций по обработке и анализу данных, особенно в статистических и биостатистических исследованиях. Однако, набор инструментов для визуализации и создания интерфейса приложений у R более ограничен по сравнению с Python. Именно этот фактор стал решающим в пользу Python, так как визуализация данных и создание удобного пользовательского интерфейса играют важную роль в данном проекте.

4.1.2 Pandas + NumPy

NumPy (Numerical Python) — это библиотека для работы с массивами и матрицами, а также для выполнения математических операций над ними. Преимущество NumPy заключается в эффективности и скорости выполнения операций, что делает её неотъемлемым инструментом для анализа данных, машинного обучения, статистики и других областей, где требуется

работа с числовыми данными. NumPy поддерживает различные линейные алгебраические операции, генерацию случайных чисел и другие функции, которые часто необходимы в научных исследованиях и разработке алгоритмов.

Pandas — это библиотека для анализа данных, построенная на основе NumPy. Она предоставляет высокоуровневые структуры данных и инструменты для работы с табличными и временными рядами данных, что делает её идеальной для обработки, анализа и визуализации данных. С Pandas можно легко выполнять такие операции, как фильтрация, агрегация и преобразование данных, что значительно упрощает процесс анализа.

Совместное использование Pandas и NumPy позволяет эффективно решать широкий спектр задач в области научных исследований, финансового анализа, обработки больших данных и машинного обучения. NumPy обеспечивает высокую производительность при выполнении числовых вычислений, тогда как Pandas предоставляет удобные инструменты для обработки и анализа данных. Вместе эти библиотеки образуют мощный инструмент для решения задач, связанных с анализом и обработкой данных в Python. Это делает их неотъемлемой частью проектируемого приложения, обеспечивая эффективность и точность анализа данных.

4.1.3 SciPy

SciPy — библиотека, основанная на расширении NumPy, которая применяется для более сложных научных и инженерных вычислений. SciPy включает модули для оптимизации, интеграции, интерполяции, эвристических алгоритмов, статистики и других задач. SciPy в основном написана на Python и частично на языках C, C++ и Fortran, благодаря чему отличается высокой производительностью и скоростью работы. В рамках разработки приложения в большей степени использовался модуль `scipy.stats`, который предоставляет обширный функционал для проведения статистических вычислений, таких как тесты гипотез, распределения вероятностей и корреляционный анализ.

Использование SciPy в проектируемом приложении позволило значительно расширить возможности анализа данных, обеспечив точность и

надёжность результатов. Высокая производительность библиотеки обеспечивает быстрое выполнение сложных вычислений, что особенно важно при работе с большими объёмами данных.

4.1.4 Matplotlib+Seaborn

Библиотеки Matplotlib и Seaborn предоставляют возможности для визуализации данных в Python. Matplotlib является основной библиотекой для создания различных типов графиков, таких как линейные графики, гистограммы, графики рассеяния и другие виды визуализаций. Она позволяет создавать статические, анимированные и интерактивные визуализации, что делает её крайне гибкой и мощной.

Seaborn, построенная на основе Matplotlib, предоставляет более высокоуровневый интерфейс для создания статистических графиков. Она упрощает создание сложных визуализаций и обеспечивает реализацию информативных графиков по умолчанию, что значительно сокращает время на настройку и оформление визуализаций.

В проектируемом приложении использование Matplotlib и Seaborn позволяет создавать наглядные и информативные графики, которые помогают лучше понять данные и представить результаты анализа в удобной форме. Это особенно важно для представления данных пользователям и принятия решений на основе анализа.

4.1.5 PyQt

PyQt — набор расширений кроссплатформенного графического фреймворка Qt, выполненный в виде библиотеки Python. Qt — фреймворк для разработки кроссплатформенного программного обеспечения с графическим интерфейсом, написанный на языке программирования C++.

Базовыми элементами графического интерфейса PyQt являются виджеты (Widgets) — основные строительные блоки для создания графических интерфейсов пользователя. PyQt предоставляет широкий ассортимент виджетов, которые можно использовать для построения сложных и функциональных приложений. Виджеты в PyQt охватывают разнообразные элементы, такие как кнопки, текстовые поля, метки, списки и многое

другое.

PyQt использует систему сигналов и слотов для реализации взаимодействия с пользователем. Сигналы используются для уведомления об изменениях состояния, а слоты — для обработки этих изменений. Эта система обеспечивает гибкость и мощность при разработке интерфейсов, позволяя легко управлять событиями и действиями пользователя.

Использование PyQt в проектируемом приложении позволяет создать кроссплатформенный графический интерфейс, который будет работать на различных операционных системах, таких как Windows, macOS и Linux. Это обеспечивает широкую доступность моего приложения и удобство для пользователей.

4.2 Импорт данных

На первом этапе разработки приложения необходимо реализовать функционал для импорта входных данных из формата .ods (OpenDocument Spreadsheet). Этот формат используется для хранения данных в виде электронных таблиц и широко поддерживается различными офисными пакетами, такими как LibreOffice и OpenOffice. Пример данных, используемых для демонстрации функционала, представлен на рисунках 5-6.

№	№ п/п по программе	Код и номер кассеты	Индекс и номер ПС СУЗ	Коорд. выпр. яч. акт. зоны	Дата	к-во кампаний	Номер пенала	I-131
		БВ			06.07.2019 7:43			7,10E+04
		Исходная			06.07.2019 8:30			3,13E+01
		Холостая			06.07.2019 8:50		3	1,52E+02
		Холостая			06.07.2019 9:10		4	н/н
		Бак			06.07.2019 10:00			н/н
1	1	N43012 11957	N033472	08-30	06.07.2019 13:15	3	3	2,84E+03
2	2	N43012 11954	N033474	09-27	06.07.2019 13:55	3	4	2,67E+03
3	3	N43012 11965	N033481	07-27	06.07.2019 14:55	3	3	2,82E+03
		Холостая			06.07.2019 21:49		4	7,48E+02
		Исходная			07.07.2019 0:47			8,20E+01
4	16	N43012 13845	N033476	05-33	07.07.2019 1:29	2	3	3,08E+03

Рисунок 5 – Пример входных данных

Cs-134	Cs-137	Mn-54	Cs-136	Xe-133	Ba-140	Ru-103	Ru-106	Ce-141	Ce-144	Ag-110m
4,15E+04	4,92E+04	4,49E+03	5,22E+03	3,33E+04	н/н	н/н	н/н	н/н	н/н	8,22E+02
3,89E+02	5,20E+02	н/н	н/н	н/н	н/н	н/н	н/н	н/н	н/н	н/н
3,21E+03	6,60E+03	3,56E+02	н/н	н/н	н/н	н/н	н/н	н/н	н/н	н/н
1,43E+04	1,67E+04	8,68E+02	н/н	н/н	н/н	н/н	н/н	н/н	н/н	н/н
4,27E+01	9,36E+01	1,56E+00	н/н	н/н	н/н	н/н	н/н	н/н	н/н	н/н
3,32E+02	4,59E+02	3,76E+02	н/н	н/н	н/н	н/н	н/н	н/н	н/н	1,49E+01
1,73E+03	2,12E+03	8,48E+02	н/н	н/н	н/н	н/н	н/н	н/н	н/н	2,24E+01
2,16E+02	2,84E+02	6,40E+02	н/н	н/н	н/н	н/н	н/н	н/н	н/н	2,93E+01
1,19E+03	1,37E+03	1,00E+03	н/н	н/н	н/н	н/н	н/н	н/н	н/н	н/н
6,25E+02	7,66E+02	6,40E+00	н/н	н/н	н/н	н/н	н/н	н/н	н/н	н/н
5,93E+02	6,88E+02	5,38E+02	н/н	н/н	н/н	н/н	н/н	н/н	н/н	3,43E+01

Рисунок 6 – Пример входных данных

Как видно из примера, входные данные содержат строки и столбцы с различной информацией, которые затем используются в последующих расчетах и анализе данных. Однако, не все данные, содержащиеся в исходном файле, могут быть непосредственно полезны или необходимы для проектируемого приложения. В связи с этим, для использования в приложении эти данные необходимо предварительно очистить. Очистка данных включает удаление лишних или нерелевантных строк и столбцов, обработку пропущенных значений и приведение данных к стандартному виду, который соответствует требованиям приложения.

Чтение файла из табличных форматов (таких как .xls, .xlsx, .ods) реализовано с помощью метода `read_excel()` из библиотеки `pandas`. Этот метод позволяет легко и быстро загружать данные из электронных таблиц в формате `DataFrame`, который является основной структурой данных в `pandas`. `DataFrame` предоставляет удобные инструменты для манипуляции и анализа данных, что делает его идеальным для нашей задачи. Метод `read_excel()` возвращает объект `DataFrame`, который затем передается в специальный метод очистки данных. Этот метод выполняет различные операции по очистке и преобразованию данных в соответствии с определённым шаблоном входных данных, чтобы подготовить их для последующего использования в расчетах и анализах.

Фрагмент реализации метода, отвечающего за очистку DataFrame представлена в листинге 1. Этот метод включает в себя несколько этапов, таких как удаление ненужных столбцов, заполнение пропущенных значений, преобразование типов данных и другие необходимые преобразования. Очистка данных является важным шагом, так как от её качества зависит точность и надежность последующих анализов и расчетов.

Листинг 1 – Метод очистки данных

```
1 def _clear_data(self, df):
2     # На текущий момент данные очищаются под конкретный шаблон пробных данных.
3     # Делаем срез DataFrame, используем первые 14 столбцов
4     df = df.iloc[:,0:14]
5     # Переименовываем столбцы для удобного анализа
6     df.rename(columns = {
7         "№":"Id1",
8         "№ п/п по программе":"Id2",
9         "Код и номер кассеты":"Name",
10        "Индекс и номер ПС СУЗ":"Index",
11        "Коорд. выгр. яч. акт. зоны ":"Coordinates",
12        "Дата":"DateTime",
13        "к-во кампаний":"Age",
14        "Номер пенала":"IdPenal"}, inplace = True)
15
16    for i in range(0, len(df.columns.to_list())):
17        column = df.columns.to_list()[i]
18        # Заменяем спец. символы
19        df.replace({column:"-"}, np.NaN, inplace=True)
20        df.replace({column:"н/н"}, np.NaN, inplace=True)
21
22    # Заполняем пропуски и приводим типы
23    df["Age"]=df["Age"].fillna(0.0).astype(int)
24    df["IdPenal"]=df["IdPenal"].fillna(0.0).astype(int)
25    df["Id2"]=df["Id2"].fillna(0.0).astype(int)
26
27    ...
28
29    return penals, penals_id, cassetes
```

4.3 Поиск выбросов

Поиск выбросов в разрабатываемом приложении реализован в качестве метода у сущности, представляющей выборку. Метод основан на вычислении критического значения для выборки по заданному критерию. После вычисления критического значения объект DataFrame агрегируется по условию. Полученные строки удаляются из исходного DataFrame и производится повторный расчёт критического значения согласно параграфу 1.3.

Фрагмент реализации метода поиска выбросов, связанный с расчётом критических значений представлен в листинге 2. Полная реализация данного метода представлена в приложении А.1.

Листинг 2 – Фрагмент метода поиска выбросов

```
1  ...

2  # method_type - аргумент функции, указывающих метод поиска выбросов
3  if method_type=="IQR":
4      # Расчёт параметров для метода "IQR"
5      a_q1 = df[criteriums[i]].quantile(q=.25)
6      a_q3 = df[criteriums[i]].quantile(q=.75)
7      a_IQR = a_q3-a_q1
8
9      a_corosion_q1 = df[criteriums[i]].quantile(q=.25)
10     a_corosion_q3 = df[criteriums[i]].quantile(q=.75)
11     a_corosion_IQR = a_corosion_q3-a_corosion_q1
12
13     # Критическое значение для реперных ПД
14     a_crit = (a_q3+1.7*a_IQR)
15     # Критическое значение ПК
16     a_corosion_crit = (a_corosion_q3+1.7*a_corosion_IQR)
17 else:
18     # Расчёт параметров для метода "3 сигм"
19     a_mean = df[criteriums[i]].mean()
20     a_corosion_mean = df["Mn-54"].mean()
21     a_std = df[criteriums[i]].std()
22     # Критическое значение для реперных ПД
23     a_corosion_std = df["Mn-54"].std()
24     a_crit = self.crit_value(a_mean, a_std, len(df))
```

```

25         # Критическое значение для ПК
26         a_corosion_crit = self.crit_value(a_corosion_mean,
27             a_corosion_std, len(df))

28     # Агрегируем DataFrame по критическому значению
29     # В результате получаем таблицу ТВС,
30     # у которых аномально высокие активности реперных ПД
31     crit_df = df[(df[criteriums[i]]>a_crit)].reset_index(drop=True)

32     # Для ТВС, полученных в crit_df
33     # проверяется выброс по продуктам коррозии
34     non_hermetic = crit_df[crit_df["Mn-54"]<a_corosion_crit]
35     recheck = crit_df[crit_df["Mn-54"]>=a_corosion_crit]

36     ...

```

Кроме того, в приведённом листинге для вычисления критического значения по методу "3 сигм" используется метод `crit_value`, который имеет довольно простую реализацию:

Листинг 3 – Функция вычисления критического значения, для

```

1
2 def crit_value(self, mean, std, n):
3     student = {2:12.7, 3:4.3, 4:3.18, 5:2.78, 6:2.57, 7:2.45,
4         8:2.36, 9:2.31, 10:2.26}
5
6     # Исключаем mean==np.NaN и std==np.NaN
7     if (not pd.isna(mean)) or (not pd.isna(std)): return 0
8
9     # Возвращаем пороговое значение, выше которого считать выбросом
10    return mean+student.get(n,3)*std
11

```

В данной реализации используются значения критерия Стьюдента из таблицы 1.

Результатом выполнения метода, представленного в листинге 2, являются 3 таблицы:

- Таблица негерметичных ТВС, содержит исходные данные о ТВС, а также информацию о радионуклидах, на основании которых предполагается их негерметичность. В таблице представлены данные, которые позволяют быстро идентифицировать проблемные ТВС и радионуклиды, вызывающие подозрения. Таблица включает данные о радионуклидах, на основании которых предполагается негерметичность, что позволяет организовать дополнительные проверки и анализ.
- Таблицы ТВС для повторной проверки, В этой таблице содержатся исходные данные о ТВС, которые требуют повторной проверки. Таблица включает данные о радионуклидах, на основании которых предполагается негерметичность, что позволяет организовать дополнительные проверки и анализ.
- Таблица расчётных параметров, содержащая информацию о математическом ожидании, среднеквадратическом отклонении и квартилях для каждой выборки данных.

4.4 Пользовательский интерфейс

Данный раздел содержит описание инструментов и трудностей, с которыми я столкнулся во время разработки приложения, а также примеры реализаций графических элементов, используемых в моём приложении.

4.4.1 Основные окна приложения

На основании анализа пользовательского сценария использования, представленного в разделе 3.3 был сделан вывод, что проектируемое приложение будет состоять из четырех окон: двух основных и двух диалоговых:

- Диалоговое окно импорта файла
- Окно анализа пеналов
- Окно анализа выборок

– Диалоговое окно с результатами статистических тестов

Это разделение позволяет организовать работу приложения наиболее эффективно и удобно для пользователя. Основные и диалоговые окна предназначены для выполнения различных функций, обеспечивая гибкость и простоту использования.

Разделение функционала между окнами позволяет организовать работу приложения наиболее рационально и удобно для пользователя. Окно анализа выборок было решено сделать самостоятельным окном, а не интегрировать его внутрь другого, чтобы обеспечить возможность параллельной работы и гибкость в анализе данных. При переходе в окно анализа выборок сохраняется возможность выбрать выборки заново без изменения логики работы приложения. Это позволяет пользователю легко переключаться между различными режимами работы и эффективно выполнять необходимые аналитические операции.

4.4.2 Окно импорта файла

При открытии файла используется класс `QFileDialog`, который предоставляет окно выбора файла. Удобство этого класса заключается в том, что имеется возможность установить фильтры на расширения файлов, чтобы не вызывать исключений при попытке открыть файл с расширением, которое не поддерживается. Пример окна для выбора файла представлен на рисунке 7.

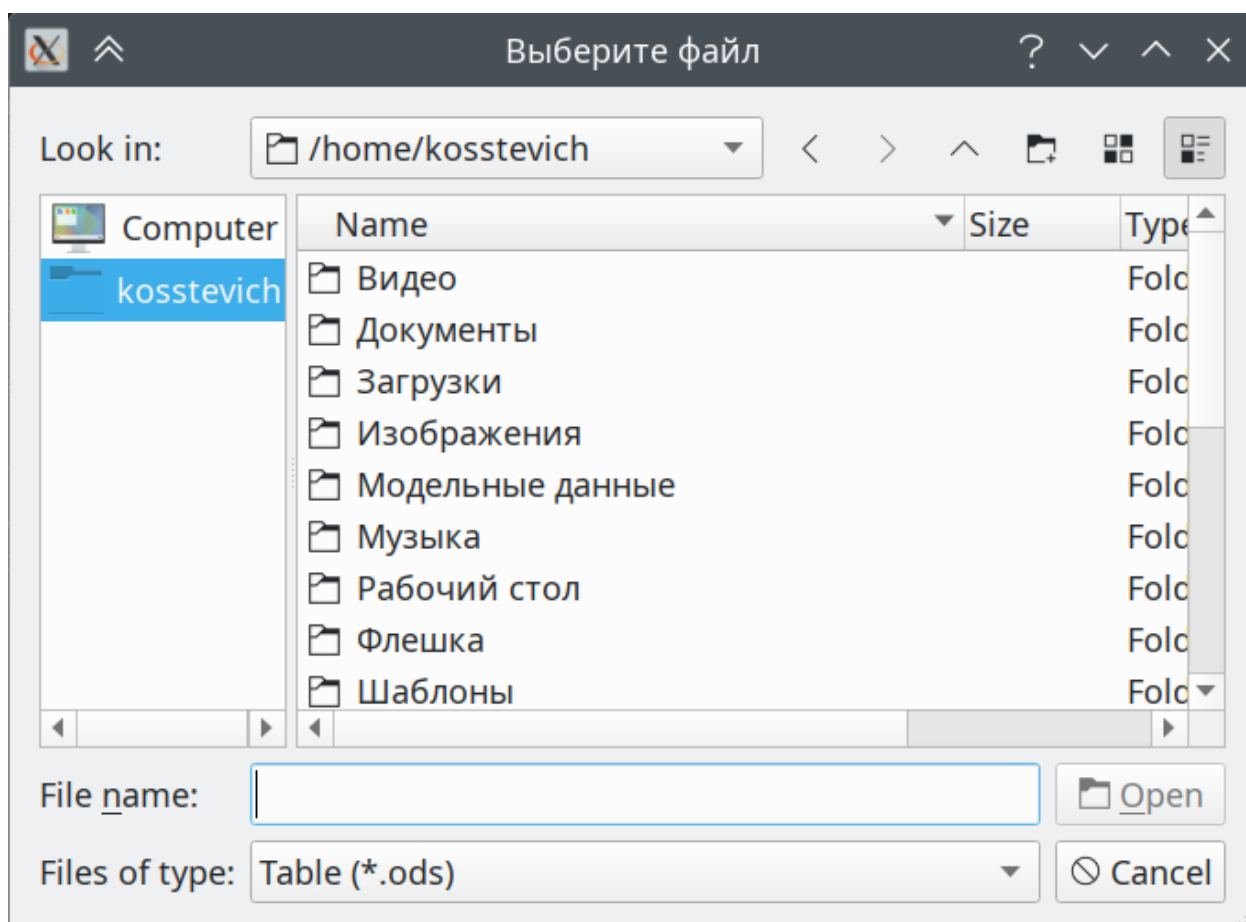


Рисунок 7 – Окно выбора файла

4.4.3 Интеграция Matplotlib в качестве QWidget

Во время разработки приложения я столкнулся с трудностью интеграции графиков Matplotlib в QWidget. Проблема заключалась в том, что базовый класс Matplotlib создает для графиков независимые окна. Эта реализация мне не подходила, так как разрабатываемое приложение должно иметь целостный и единый интерфейс, где все элементы интегрированы в одно главное окно.

Изучив документацию библиотеки Matplotlib, был найден класс FigureCanvasQTAgg, который предоставляет backend-реализацию холста Canvas для PyQt. Этот класс позволяет встраивать графики Matplotlib непосредственно в виджеты PyQt, что решает проблему независимых окон и обеспечивает интеграцию графиков в основной интерфейс приложения.

На основе класса FigureCanvasQTAgg я разработал собственный пользовательский виджет для использования в моем приложении. Этот виджет включает все необходимые функции для отображения графиков Matplotlib

внутри PyQt-приложения. Реализация пользовательского виджета представлена в листинге 4

Листинг 4 – Пользовательский класс, интегрирующий Canvas в QWidget

```
1 class PlotData(FigureCanvasQTAff):
2     # Виджет для отрисовки графиков, использующий matplotlib
3     def __init__(self, parent=None):
4         sns.set(style="whitegrid", context="paper")
5         self.fig = plt.figure(figsize=(15, 10))
6         self.axes = self.fig.add_subplot(111)
7         super(PlotData, self).__init__(self.fig)
8
9     def draw_plot(self, df, axe_x="Id1",
10                  axe_y="I-131", type = "barplot", name=None):
11         self.axes.cla()
12         # Метод позволяет рисовать разные виды графиков seaborn
13         # Выбор типа графика происходит по аргументу type
14         getattr(sns,type)(data=df, x=axe_x, y=axe_y, ax=self.axes)
15         plt.title(label= name, fontsize=16)
```

Кроме реализации холста я использовал класс `NavigationToolbar2QT`, который предоставляет дополнительные возможности для работы с графиками. Эта панель инструментов включает функции навигации по графику, редактирования легенды, а также экспорт графика в формате изображения. Это значительно улучшает удобство работы с графиками, предоставляя пользователю интуитивно понятные инструменты для взаимодействия с визуализациями.

Использование `FigureCanvasQTAff` и `NavigationToolbar2QT` позволило мне создать мощный и удобный инструмент для отображения и взаимодействия с графиками Matplotlib в моем PyQt-приложении. На рисунке 8 красной рамкой выделены реализованные мною классы.

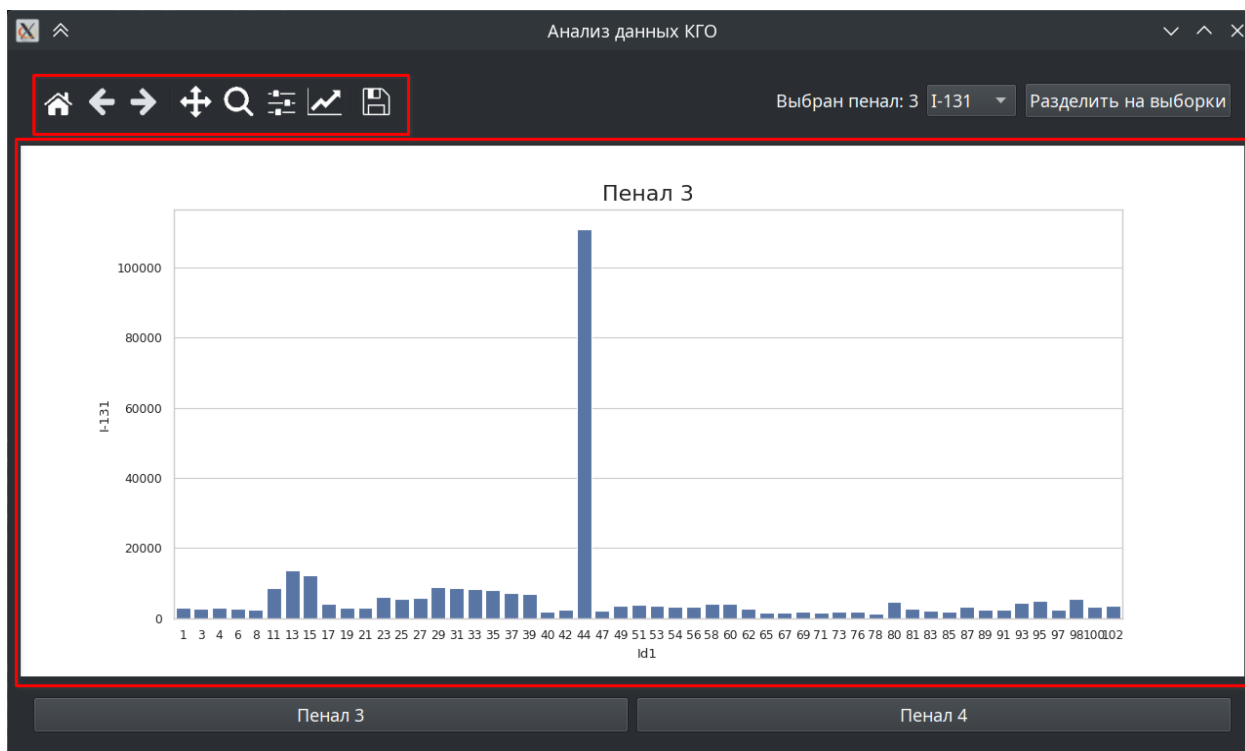


Рисунок 8 – Главное окно приложения

4.4.4 Главное окно приложения

Пример внешнего вида главного окна приложения приведён на рисунке 8. Это окно реализовано с использованием различных элементов библиотеки Qt, которые позволяют создавать удобный и функциональный интерфейс для пользователя. В главном окне представлены основные элементы управления и отображения данных, обеспечивающие легкость навигации и взаимодействия с приложением. Вот некоторые из них:

- QPushButton — класс, который используется для создания кнопок. Эти кнопки выполняют различные действия, такие как запуск анализа, импорт данных, переход между разными секциями приложения и другие операции. Каждое действие, выполняемое при нажатии кнопки, реализуется через привязку функции действия (слота) к функции изменения состояния (сигнала). Кнопки могут быть снабжены текстовыми метками или иконками для лучшего восприятия.
- QLabel — класс, который позволяет выводить текстовую информацию. Эти метки используются для отображения статических тек-

стов, таких как заголовки секций, описания и инструкции для пользователя. QLabel также может отображать изображения и другие графические элементы.

- QComboBox — класс, который реализует выпадающий список, который позволяет пользователю выбирать один из множества вариантов. Этот элемент интерфейса особенно полезен для настройки параметров приложения, выбора данных для анализа и других действий, требующих выбора из заранее определённого набора значений.
- QHBoxLayout — класс, который используется для горизонтальной разметки виджетов. С его помощью виджеты располагаются в одну строку, что позволяет организовать элементы интерфейса логически и удобно для пользователя. Горизонтальная разметка часто применяется для группировки кнопок или других управляющих элементов.
- QVBoxLayout — класс, который позволяет располагать виджеты в вертикальной разметке. Элементы располагаются друг под другом, что удобно для создания списков, форм и других структурированных интерфейсов. Вертикальная разметка помогает упорядочить элементы и сделать интерфейс более читаемым.

Кнопки, отвечающие за переход по пеналам, реализованы динамически в зависимости от количества пеналов в исходных данных. Это означает, что при запуске приложения на АЭС с разным количеством пеналов СОДС, количество кнопок автоматически адаптируется к числу пеналов. Такая динамическая генерация кнопок обеспечивает гибкость интерфейса и позволяет легко масштабировать приложение под различные объёмы данных.

4.4.5 Окно анализа выборок

Пример реализации окна анализа выборок представлен на рисунке 9.

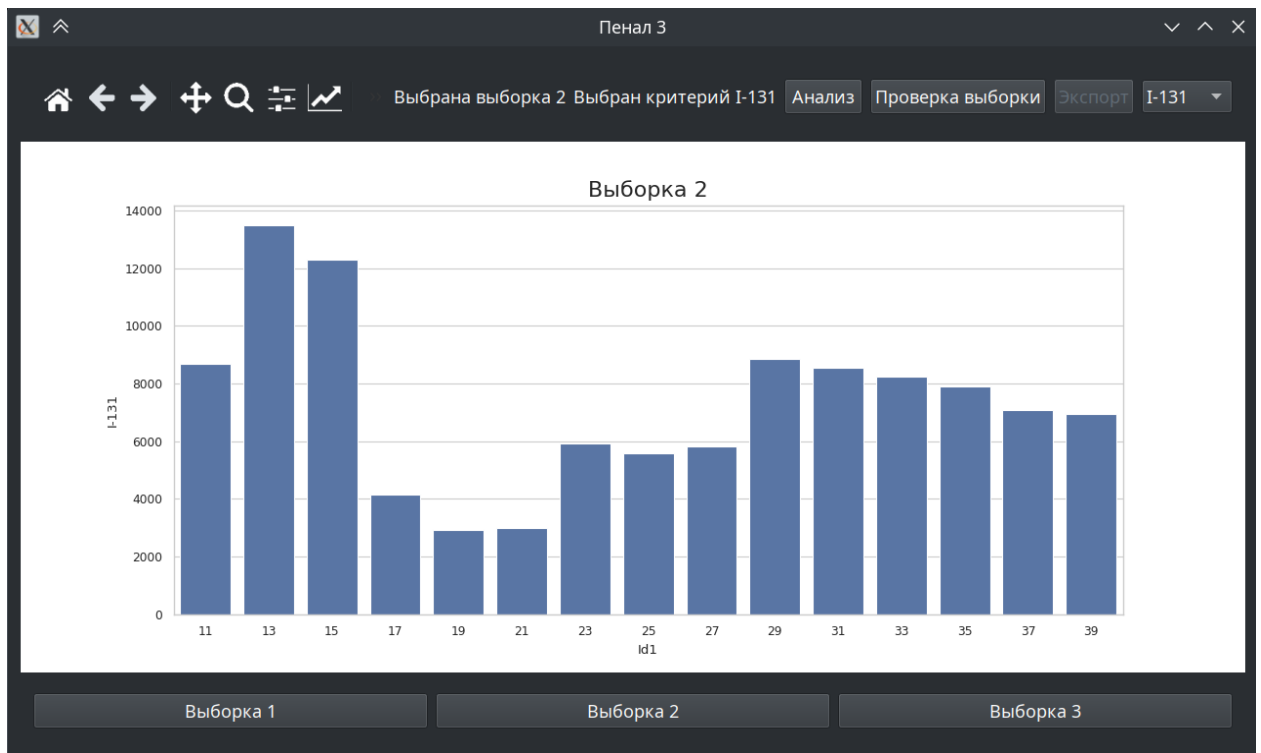


Рисунок 9 – Окно анализа выборок

Несмотря на схожесть с окном анализа выборок, данное окно имеет более сложную структуру. В частности, класс `PlotData`, описанный в 4.4.3, после проведения поиска выбросов заменяется на класс `QTableWidget`, который содержит выходную информацию в табличном виде. Данный функционал реализован с применением класса `QStackedWidget`, который позволяет отображать только один из нескольких дочерних объектов. Пример табличного вывода представлен на рисунке 10.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	Выборка 3														
2	Расширенная информация														
3		I-131	Cs-134	Cs-137	Cs-136	Xe-133	Mn-54								
4	Activity_mean	6279.0625	4589.975	4632.61875	37800.0	41100.0	313.84375								
5	Activity_std	19140.5624617...	25074.5875960...	25249.3039562...	nan	nan	362.912...								
6	Activity_critical	63700.7498851...	79813.7377880...	80380.5306188...	0.0	0.0	1402.58...								
7	TBC для повторной проверки:														
8	Id1	Id2	Name	Index	Coordinates	DateTime	Age	IdPenal	I-131	Cs-134	Cs-137	Mn-54	Cs-136	Xe-133	Criterium
9	44	56	N43012 13863	N033489	05-23	2019-07-09 ...	2	3	111000.0	142000.0	143000.0	2260.0	37800.0	41100.0	I-131
10	44	56	N43012 13863	N033489	05-23	2019-07-09 ...	2	3	111000.0	142000.0	143000.0	2260.0	37800.0	41100.0	Cs-134
11	44	56	N43012 13863	N033489	05-23	2019-07-09 ...	2	3	111000.0	142000.0	143000.0	2260.0	37800.0	41100.0	Cs-137
12	44	56	N43012 13863	N033489	05-23	2019-07-09 ...	2	3	111000.0	142000.0	143000.0	2260.0	37800.0	41100.0	Cs-136
13	44	56	N43012 13863	N033489	05-23	2019-07-09 ...	2	3	111000.0	142000.0	143000.0	2260.0	37800.0	41100.0	Xe-133

Рисунок 10 – Представление табличной информации через QTableWidget

4.5 Экспорт таблиц

После проведения необходимого анализа, имеется возможность экспорта таблицы в .ods формат. Необходимость данного функционала обосновывается тем, что полученные данные используются в дальнейшем в рабочем процессе, например при составлении отчётов о проведении КГО.

Следует заметить, что при анализе данных в ручном режиме устоялась практика структурирования данных по различным листам файла excel. Следовательно, чтобы не нарушать рабочий процесс и обеспечить наиболее простое внедрение моего приложения, требуется их структурировать.

Ниже описана структура файла, которая представлена в формате сопоставления имени листа и информации, которая хранится в таблице на данном листе:

- Исходные данные — Таблица, содержащая входные данные без изменений.
- МРi — Таблица, содержащая данные для i -го пенала.
- МРi-j — Таблица, содержащая результаты анализа j -ой выборки для i -го пенала. Данная таблица содержит данные для каждой

ТВС, расчётные параметры выборки, а также негерметичные ТВС и ТВС, для которых требуется повторная проверка.

Пример структуры файла представлен на рисунке 11.

Исходные данные	МПЗ	МПЗ-1	МПЗ-2	МПЗ-3	МП4	МП4-1	МП4-2	МП4-3	МП4-4	Базовый
-----------------	-----	-------	-------	-------	-----	-------	-------	-------	-------	---------

Рисунок 11 – Структура выходного файла

Пример экспортированного файла приведён на рисунке 12.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1		I-131	Cs-134	Cs-137	Cs-136	Xe-133	Mn-54								
2	Activity mean	6279,0625	4589,975	4632,6188	37800	41100	313,84375								
3	Activity std	19140,562	25074,588	25249,304			362,91269								
4	Activity critical	63700,75	79813,738	80380,531	0	0	1402,5818								
5	Id1	Id2	Name	Index	Coordinate	DateTime	Age	IdPenal	I-131	Cs-134	Cs-137	Mn-54	Cs-136	Xe-133	
6	40	52	N43012 13IN033492	07-35	43655,419		2	3	1890	102	121	283			
7	42	54	N43012 13IN033487	12-26	43655,492		2	3	2480	175	186	233			
8	44	56	N43012 13IN033489	05-23	43655,563		2	3	111000	142000	143000	2260	37800	41100	
9	47	71	N43012 11IN033473	08-26	43655,868		3	3	2100	214	217	320			
10	49	73	N43012 14IN033833	06-38	43655,929		1	3	3530	151	170	165			
11	51	75	N43012 14IN033826	14-26	43656,01		1	3	3710	159	168	158			
12	53	77	N43012 14IN033817	04-20	43656,11		1	3	3410	113	129	228			
13	54	78	N43012 14IN033819	02-30	43656,167		1	3	3310	149	168	216			
14	56	80	N43012 14IN033829	10-38	43656,229		1	3	3220	149	173	207			
15	58	82	N43012 14IN033831	12-20	43656,299		1	3	3970	187	198	253			
16	60	84	N43012 14IN033818	02-26	43656,358		1	3	4210	185	202	230			
17	62	86	N43012 13IN033478	11-37	43656,419		2	3	2650	251	264	305			
18	65	89	N43012 13IN033484	05-19	43656,906		2	3	1670	86,1	88,9	348			
19	67	91	N43012 11IN033479	08-20	43656,969		3	3	1690	84,1	96,7	337			
20	69	104	N43012 13IN030975	12-34	43657,219		2	3	1810	85,5	101	286			
21	71	106	N43012 13IN030977	09-19	43657,295		2	3	1570	56,4	65,1	165			
22	73	108	N43012 13IN033480	03-31	43657,389		2	3	1710	106	110	195			
23	76	111	N43012 13IN030980	13-31	43657,608		2	3	1830	144	149	138			
24	78	113	N43012 13IN031911	07-19	43657,667		2	3	1390	93,5	90,3	197			
25	80	115	N43012 13IN030982	06-34	43658,75		2	3	4760	388	424	290			
26	81	116	N43012 13IN030971	10-34	43658,816		2	3	2750	159	165	295			
27	83	118	N43012 13IN030978	10-22	43658,875		2	3	2270	127	134	218			
28	85	120	N43012 13IN031913	04-28	43658,938		2	3	1800	76,4	77,8	250			
29	87	122	N43012 13IN033457	11-31	43659,024		2	3	3130	259	288	272			
30	89	124	N43012 13IN033460	08-22	43659,081		2	3	2310	177	178	305			
31	91	126	N43012 13IN031670	05-31	43659,144		2	3	2470	154	166	312			
32	93	128	N43012 14IN033825	09-31	43659,21		1	3	4450	252	282	446			
33	95	130	N43012 14IN033823	09-25	43659,271		1	3	4980	290	297	377			
34	97	132	N43012 14IN033822	06-28	43659,34		1	3	2550	67,3	65	157			
35	98	135	N43012 11IN033479	08-40	43659,435		3	3	5500	285	307	305			
36	100	137	N43012 11IN033479	14-22	43659,494		3	3	3250	74,7	75	132			
37	102	139	N43012 11IN033479	02-22	43659,559		3	3	3560	79,2	88	160			
38	0														
39	ТВС для повторной проверки:														
40	Id1	Id2	Name	Index	Coordinate	DateTime	Age	IdPenal	I-131	Cs-134	Cs-137	Mn-54	Cs-136	Xe-133	Criterium
41	44	56	N43012 13IN033489	05-23	43655,563		2	3	111000	142000	143000	2260	37800	41100	I-131
42	44	56	N43012 13IN033489	05-23	43655,563		2	3	111000	142000	143000	2260	37800	41100	Cs-134
43	44	56	N43012 13IN033489	05-23	43655,563		2	3	111000	142000	143000	2260	37800	41100	Cs-137
44	44	56	N43012 13IN033489	05-23	43655,563		2	3	111000	142000	143000	2260	37800	41100	Cs-136
45	44	56	N43012 13IN033489	05-23	43655,563		2	3	111000	142000	143000	2260	37800	41100	Xe-133

Рисунок 12 – Пример выходного файла

Реализация метода, отвечающего за экспорт данных, представлена на листинге 5. Данный метод относится к сущности, представляющей пенал.

Листинг 5 – Реализация метода экспорта данных в пенале

```

1 def export_data(self, filename="output.ods",
2     writer = pd.ExcelWriter(path="output.ods",
3     datetime_format='DD-MM-YYYY HH:MM:SS')):
4     """
5     Метод, отвечающий за экспорт данных в пенале
6     """

```

```

7         sheet = "МП%s" % self.id # Генерация имени пенала
8         self.data.to_excel(excel_writer = writer, sheet_name = sheet,
9                             index=False) # Запись всех данных в выборке
10
11     # Запись данных для каждой выборки
12     for i in range(0, len(self.fragments)):
13         sheet = "МП%s-%s" % (self.id, (i+1))
14
15         # Запись расчётных параметров
16         parameters = self.fragments[i].get_parameters_df()
17         parameters.to_excel(excel_writer = writer, sheet_name = sheet)
18
19         #Запись выборочных данных
20         self.fragments[i].df.to_excel(excel_writer = writer,
21                                       sheet_name = sheet, index=False)
22
23         if not self.fragments[i].non_hermetic_df.empty:
24             df = pd.DataFrame(data = ["Негерметичные ТБС:"])
25             df.to_excel(excel_writer=writer,sheet_name=sheet,
26                       index=False)
27             self.fragments[i].non_hermetic_df.to_excel(
28                 excel_writer=writer, sheet_name=sheet, index=False)
29         if not self.fragments[i].recheck_df.empty:
30             df = pd.DataFrame(data = ["ТБС для повторной проверки:"])
31             df.to_excel(excel_writer=writer,sheet_name=sheet,
32                       index=False)
33             self.fragments[i].recheck_df.to_excel(
34                 excel_writer=writer, sheet_name=sheet, index=False)

```

ЗАКЛЮЧЕНИЕ

В ходе проделанной работы был проведён анализ существующего метода анализа данных КГО в пеналах СОДС на реакторах ВВЭР типа. Были приведены предложения по улучшению и автоматизации данного процесса. Было реализовано программное обеспечение с учётом внесённых предложений, в которой реализованы следующие функции:

- Поиск выбросов методом "3 сигм".
- Поиск выбросов методом "IQR".
- Анализ распределений выборок .
- Экспорт в табличный формат для составления отчётов КГО.
- Визуализация и экспорт графиков для составления отчётов КГО.

Разработанное приложение было апробировано на реальных данных проведения КГО за последние 5 лет на Нововоронежской АЭС. Во время проверки сравнивались реально установленные негерметичные ТВС и ТВС, определяемые приложением. В результате проверки было установлено, что разработанное приложение безошибочно определяет ТВС по методу проверки с помощью 3 среднеквадратических отклонений. Кроме того, отмечено, что данное приложение обладает всеми необходимыми пользовательскими функциями и принято в рабочий процесс.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Сборки тепловыделяющие ядерных реакторов типа ВВЭР-1000 ТИПОВАЯ МЕТОДИКА КОНТРОЛЯ ГЕРМЕТИЧНОСТИ ОБОЛОЧЕК ТЕПЛОВЫДЕЛЯЮЩИХ ЭЛЕМЕНТОВ / АО «Концерн Росэнергоатом». — 2009. — РД ЭО 1.1.2.10.0521-2009.
2. Поведение продуктов коррозии в первом контуре ЯЭУ с водным теплоносителем / Б. А. Гусев [и др.] // Вестник СПбГУ. Серия 4. Физика. Химия. — 2012. — № 4. — URL: <https://cyberleninka.ru/article/n/povedenie-produktov-korrozii-v-pervom-konture-yaeu-s-vodnym-teplonositelem>.
3. *Кремер. Н. Ш.* Теория вероятностей и математическая статистика — 5-е изд. — Москва : Издательство Юрайт, 2024. — С. 538.
4. *Дорофеев В. С., Волосатова Т. М.* Ансамблирование методов обнаружения выбросов при подготовке обучающей выборки данных. // Моделирование, оптимизация и информационные технологии. — 2022. — 10(3). — URL: <https://moitvivi.ru/ru/journal/pdf?id=1210>.
5. *Лемешко Б. Ю.* Критерии проверки отклонения распределения от нормального закона. Руководство по применению / Новосибирский государственный технический университет. — 2014.
6. *Александровская Л. Н., Кириллин А. В.* Рекомендации по применению ряда критериев проверки отклонения распределения вероятностей от нормального закона в практике инженерного статистического анализа // Известия Самарского научного центра РАН. — 2017. — № 1—1. — URL: <https://cyberleninka.ru/article/n/rekomendatsii-po-primeneniyu-ryada-kriteriev-proverki-otkloneniya-raspredeleniya-veroyatnostey-ot-normalnogo-zakona-v-praktike>.
7. ГОСТ Р ИСО 5479-2002 / Проверка отклонения распределения вероятностей от нормального распределения. — 2002.
8. *Орлов А. И.* Эконометрика. — Москва "Экзамен", 2002. — С. 437.

Приложение А

Листинг А.1 – Реализация метода поиска выбросов в выборке

```
1 def check(self, method_type="3_sigma"):
2     '''
3     Поиск выбросов в выборке согласно RD_6.7.1.5
4     В дополнение к RD_6.7.1.5 реализован поиск с помощью метода IQR
5     '''
6     # Получаем копию DataFrame, который является полем сущности
7     df = self.df
8     # Создаём экземпляры DataFrame, которые будут хранить информацию
9     # о герметичных ТБС, а также о ТБС, подлежащих повторной проверке
10    self.non_hermetic_df = pd.DataFrame()
11    self.recheck_df=pd.DataFrame()
12
13    while True:
14        remove = pd.DataFrame()
15
16        for i in range(0,len(criteriums)-1):
17            # Вычисляем параметры для поиска критических значений
18            if method_type=="IQR":
19                # Расчёт параметров для метода "IQR"
20                a_q1 = df[criteriums[i]].quantile(q=.25)
21                a_q3 = df[criteriums[i]].quantile(q=.75)
22                a_IQR = a_q3-a_q1
23
24                a_corosion_q1 = df[criteriums[i]].quantile(q=.25)
25                a_corosion_q3 = df[criteriums[i]].quantile(q=.75)
26                a_corosion_IQR = a_corosion_q3-a_corosion_q1
27
28                a_crit = (a_q3+1.7*a_IQR)
29                a_corosion_crit = (a_corosion_q3+1.7*a_corosion_IQR)
30            else:
31                # Расчёт параметров для метода "3 сигм"
32                a_mean = df[criteriums[i]].mean()
33                a_corosion_mean = df["Mn-54"].mean()
34                a_std = df[criteriums[i]].std()
35                a_corosion_std = df["Mn-54"].std()
36                a_crit = self.crit_value(a_mean, a_std, len(df))
37                a_corosion_crit = self.crit_value(a_corosion_mean,
38                                                    a_corosion_std, len(df))
```

```

39
40     # Агрегируем DataFrame по критическому значению
41     # В результате получаем таблицу ТВС,
42     # у которых аномально высокие активности реперных ПД
43     crit_df = df[(df[criteriums[i]]>a_crit)].reset_index(drop=True)
44
45     # Для ТВС, полученных в crit_df
46     # проверяется выброс по продуктам коррозии
47     non_hermetic = crit_df[crit_df["Mn-54"]<a_corosion_crit]
48     recheck = crit_df[crit_df["Mn-54"]>=a_corosion_crit]
49
50     if not non_hermetic.empty:
51         non_hermetic["Criterium"] = criteriums[i]
52         self.non_hermetic_df = pd.concat(
53             [self.non_hermetic_df,
54              non_hermetic], ignore_index=True)
55
56     if not recheck.empty:
57         recheck["Criterium"] = criteriums[i]
58         self.recheck_df = pd.concat(
59             [self.recheck_df, recheck],
60             ignore_index=True)
61
62     remove = pd.concat([remove, crit_df["Id1"]], ignore_index=True)
63
64     if remove.empty:
65         break
66
67     for i in range(0, len(remove)):
68         df = df.loc[df["Id1"]!=remove[0].iloc[i]]
69         df = df.reset_index(drop=True)
70
71     return self.non_hermetic_df, self.recheck_df

```
