

Санкт-Петербургский государственный университет

Кафедра системного программирования

Группа 23.Б10-мм

Развитие проектов PySATL-criterion и PySATL-experiment

Похабов Иван Ильич

Отчёт по учебной практике
в форме «Производственное задание»

Научный руководитель:
доцент кафедры системного программирования, к. ф.-м. н., Гориховский В. И.

Санкт-Петербург
2025

Оглавление

Введение	3
1. Постановка задачи	5
2. Реализация	6
2.1. Имплементация графовых критериев согласия	6
2.2. Развитие библиотеки PySATL-criterion	6
2.3. Развитие фреймворка PySATL-experiment	8
2.4. Рефакторинг инфраструктуры проектов	9
3. Графовые критерии согласия	12
3.1. Алгоритм	12
3.2. Нормализация данных и выбор параметра дистанции . .	13
3.3. Статистическая мощность и уровень значимости критерия	14
3.4. Критические значения и методы их вычисления	16
4. Эксперимент	19
4.1. Условия эксперимента	19
4.2. Исследовательские вопросы	20
4.3. Метрика	20
4.4. Результаты	21
4.5. Обсуждение результатов	22
Заключение	23
Список литературы	24

Введение

Методы математической статистики находят применение в самых разных сферах, включая проверку гипотез в биологии, медицине и других научных направлениях [2][3]. Многие статистические критерии доступны на языке R, однако его интеграция в программные комплексы затруднена. Альтернативы вроде библиотек SciPy и Statsmodels предлагают базовые методы анализа, но их функциональность часто ограничена для узкоспециализированных задач из-за недостаточной точности или отсутствия редких критериев. Это подчеркивает необходимость развития адаптируемых и гибких инструментов, способных покрывать разнообразные исследовательские потребности.

Для решения этих задач существует проект PySATL, включающий в себя множество библиотек для различных целей: `pysatl-cpd`¹ (обнаружение разладки), `pysatl-nmvm`² (оценка параметров смеси непрерывных распределений), `pysatl-criterion`³ (проверка статистических гипотез), `pysatl-mpest`⁴ и другие.

Одной из ключевых задач этой дисциплины выступает тестирование статистических гипотез, особенно связанных с соответствием эмпирических данных теоретическим распределениям. Под критериями согласия понимают инструменты, которые на основе анализа выборки и заданного уровня значимости дают возможность опровергнуть предположение о соответствии данных определённому распределению. Хотя существует множество критериев — как универсальные (например, критерии Колмогорова–Смирнова или хи-квадрат), так и узкоспециализированные (критерий Шапиро–Уилка), — ни один из них не является абсолютно надёжным для безошибочного принятия гипотез. Создание новых методов проверки способно повысить точность выводов при принятии или отклонении статистических гипотез.

Графовый подход к проверке гипотез уже применяется, например,

¹<https://github.com/PySATL/pysatl-cpd>

²<https://github.com/PySATL/pysatl-nmvm>

³<https://github.com/PySATL/pysatl-criterion>

⁴<https://github.com/PySATL/pysatl-mpest>

для решения задачи обнаружения разладки. Например, в работе Нао Chen[1] предложен комбинаторный критерий обнаружения разладки, основанный на анализе дистанционных графов. Этот подход может найти применение и в задаче проверки гипотез согласия, так как дистанционные графы помогают искать структурные различия в выборках, чего другие критерии явно не делают.

В данной работе исследуются и реализуются некоторые графовые критерии согласия, а также ставятся задачи развития фреймворка `pysatl-experiment`⁵ и `pysatl-criterion`.

⁵<https://github.com/PySATL/pysatl-experiment>

1. Постановка задачи

Целью работы является развитие библиотеки PySATL-criterion и фреймворка PySATL-experiment

1. Продолжение исследований и имплементация графовых критериев проверки гипотез согласия
2. Развитие библиотеки PySATL-criterion
 - Интеграция графовых критериев согласия
 - Рефакторинг уже существующих критериев
 - Рефакторинг инфраструктуры проекта
3. Развитие фреймворка PySATL-experiment
 - Добавление функциональности для автоматической генерации табличных отчётов экспериментов по вычислению мощностей критериев согласия
 - Добавление возможности использования двусторонних критериев
 - Добавление проверки типов для строгой типизации кода
 - Рефакторинг инфраструктуры проекта

2. Реализация

2.1. Имплементация графовых критериев согласия

Были реализованы графовые критерии согласия для нормального и экспоненциального семейств распределений, базирующиеся на различных графовых численных характеристиках:

- Количество рёбер
- Максимальная степень вершины
- Количество компонент связности
- Среднее значение степеней вершин

Также был проведен эксперимент вычисления мощностей для критериев нормальности, основанных на статистиках количества рёбер и максимальной степени вершины, который будет описан в дальнейшем.

2.2. Развитие библиотеки PySATL-criterion

Интеграция графовых критериев

Интегрированы графовые критерии в библиотеку: Был введён базовый класс *AbstractGraphTestStatistic*, который наследуется от родительского класса *AbstractGoodnessOfFitStatistic* и переопределяет метод *execute_statistic* вычисляющий статистику следующим образом:

- Вычисляется параметр дистанции (метод *_compute_dist*)
- Строится графовое представление в виде списка смежности (метод *_make_adjacency_list*)
- Вычисляется статистика графа (метод *get_graph_stat*) – в данном классе этот метод является абстрактным

Это сделано для удобной расширяемости — для того, чтобы добавить новую графовую статистику, нужно лишь в потомке класса *AbstractGoodnessOfFitStatistic* переопределить метод *get_graph_stat*. Для того, чтобы сделать новую статистику для определённого распределения, достаточно сделать класс-смесь, наследующийся от класса статистики распределения и класса конкретной графовой статистики, и переопределить в нём метод *_compute_dist*. С UML-диаграммой классов можно ознакомиться на Рис. 1.

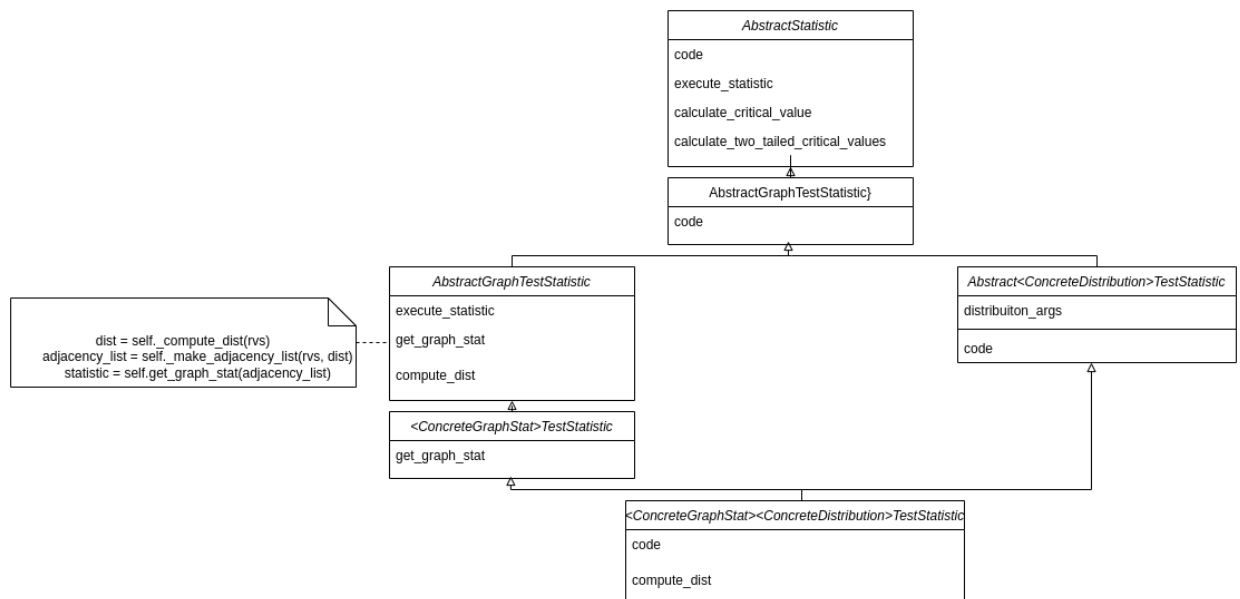


Рис. 1: Диаграмма классов для графовых статистик

Рефакторинг критериев

- Исправлены ошибки в вычислениях статистики для критериев Андерсона-Дарлинга
 - При реализации статистики Андерсона-Дарлинга у автора возникла опечатка, и вместо вычисления значения функции, в переменную присваивалась просто функция
- При реализации статистики Чжан-Ву при вычислении выборочной дисперсии был неверно выставлен параметр степени свободы:

- **Суть ошибки:** Использование `ddof=0` вместо `ddof=1` при вычислении выборочной дисперсии:

$$\text{Неверно: } \hat{\sigma}^2 = \frac{1}{n} \sum (x_i - \bar{x})^2 \quad (\text{ddof}=0)$$

$$\text{Верно: } s^2 = \frac{1}{n-1} \sum (x_i - \bar{x})^2 \quad (\text{ddof}=1)$$

2.3. Развитие фреймворка PySATL-experiment

В ходе работы были выполнены следующие задачи:

- Реализован табличный генератор отчётов для экспериментов по вычислению мощностей критериев согласия. Создан класс *PdfPowerReportBuilder*, который спроектирован согласно паттерну «Строитель»:
 - Метод *process* накапливает информацию о результате вычисления мощности одного из критериев на определенной альтернативе с фиксированным размером выборки
 - Метод *build* строит отчёт в формате pdf табличный отчёт – для каждой альтернативы строится табличка, в которой строка определяет критерий, а столбец размер выборки
- Добавлена возможность вычисления двусторонних критических значений статистики с помощью Монте-Карло моделирования. Для критериев, у которых нет вычисления критических значений с помощью анализа предельного распределения, перед оценкой мощности вычисляются критические значения методом Монте-Карло моделирования. До этого рассматривались только критерии, для которых нужно было лишь одностороннее вычисление критического значения:
 - Для этого был добавлен абстрактный метод `calculate_two_tailed_critical_value` класс

AbstractTestStatistic, и функция с одноимённым названием, которая используется как фабричный метод в функции *get_or_calculate_critical_value*

- Добавлена проверка типов с помощью библиотеки *myru* и проведён рефакторинг кода с целью повышения надёжности и поддерживаемости проекта. Использование строгой типизации позволило:
 - Упростить рефакторинг за счёт явного описания контрактов между компонентами (например, между генератором отчётов и расчётными модулями)
 - Улучшить читаемость кода за счёт явной аннотации типов параметров и возвращаемых значений
 - Избежать в дальнейшем ошибок связанных с использованием переменных с невалидными типами

Интеграция *myru* в CI/CD-процесс обеспечила автоматический контроль типизации при каждом коммите, что критически важно для проекта с активно развивающейся архитектурой.

2.4. Рефакторинг инфраструктуры проектов

Для обоих проектов был проведён рефакторинг инфраструктуры. Основными задачами которого являлись:

- **Настройка Poetry как менеджера пакетов:**
 - **Управление зависимостями:** Poetry упрощает установку, обновление и удаление пакетов через файл `pyproject.toml`. Он автоматически генерирует `poetry.lock`, фиксируя точные версии зависимостей, что гарантирует воспроизводимость окружения на всех машинах

- **Изоляция окружений:** Инструмент создаёт виртуальные окружения «из коробки», избегая конфликтов между версиями пакетов в разных проектах
- **Интеграция с pre-commit:** Через Poetry легко добавляются хуки git (например, форматирование кода с помощью black или проверка типов mypy). Например:

```
poetry add --group dev pre-commit
poetry run pre-commit install
```
- **Публикация пакетов:** Поддержка сборки и публикации в PyPI через команду `poetry publish`
- **Добавление Dependabot для автоматического обновления зависимостей:**
 - **Автоматизация обновлений:** Dependabot сканирует зависимости в `pyproject.toml` и `poetry.lock`, создавая пул-реквесты при обнаружении новых версий пакетов. Это избавляет разработчиков от ручного мониторинга обновлений
 - **Повышение безопасности:** Устаревшие пакеты часто содержат уязвимости. Dependabot минимизирует риски, предлагая своевременные обновления
 - **Интеграция с CI/CD:** Каждый пул-реквест от Dependabot автоматически запускает тесты, проверяя совместимость обновлений с кодом
 - **Гибкая настройка:** Можно исключать пакеты из проверки или задавать диапазоны версий (например, только минорные обновления)

Итоговые преимущества:

- **Снижение ошибок:** Воспроизводимое окружение и автоматические проверки уменьшают риски «работает на моей машине»

- **Экономия времени:** Dependabot и pre-commit автоматизируют рутину, позволяя сосредоточиться на разработке
- **Стандартизация репозитория:** В ближайшем будущем все репозитории проекта должны будут использовать данные инструменты

3. Графовые критерии согласия

3.1. Алгоритм

Данная работа является продолжением предыдущей⁶. В прошлой работе исследовалась корреляция между графовыми численными характеристиками выборки и семейством распределения. В этой работе рассматриваются уже сами графовые критерии, устроенные следующим образом: критерий определяет пороговое расстояние для выборки и формирует графовую структуру, где вершины соответствуют элементам выборки. Ребра между вершинами проводятся только в том случае, если разница по модулю между значениями элементов не превышает заданного порога. На следующем этапе рассчитываются числовые характеристики графа (например, количество компонент связности, степень вершин), которые сравниваются с референтными показателями для предполагаемого распределения. Альтернативно, для оценки значимости может применяться метод Монте-Карло, генерирующий искусственные выборки и определяющий вероятность соответствия исходных данных гипотезе.

Важной особенностью таких критериев является их адаптивность: они могут быть параметрическими (с предварительной нормализацией данных или учётом параметров распределения при расчёте порога расстояния) или непараметрическими, что расширяет их применимость к различным типам данных. На основе полученных результатов статистическая гипотеза либо отклоняется, либо принимается.

Алгоритм критерия:

1. **Нормализация данных.** Преобразуем выборку S к нормализованной форме S' , например:

$$S' = \begin{bmatrix} 0.50 & -0.14 & 0.65 & 1.52 & -0.23 & -0.23 & 1.58 & 0.77 \\ -0.47 & 0.54 & -0.46 & -0.47 & 0.24 & -1.91 & -1.72 & -0.56 \\ -1.01 & 0.31 & -0.91 & -1.41 & & & & \end{bmatrix}$$

⁶https://se.math.spbu.ru/thesis/texts/Pohabov_Ivan_Il'ich_Autumn_practice_2nd_year_2024_text.pdf

2. **Построение дистанционного графа.** Соединяем вершины рёбрами, если расстояние между ними меньше заданного порога ($dist = 0.95$)

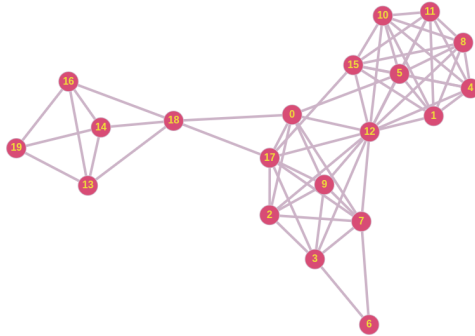


Рис. 2: Построенный граф

3. **Вычисление характеристик графа:**

- Количество рёбер: 51
- Максимальная степень вершины: 13
- Количество компонент связности: 2
- Средняя степень вершины: 5.10

4. **Сравнение значения статистики с критическими значениями**

3.2. Нормализация данных и выбор параметра дистанции

- **Стандартный параметр:**

$$dist = \frac{\max(S) - \min(S)}{\alpha},$$

где S — выборка, α — коэффициент чувствительности

– Инвариантен к сдвигу:

$$\max(S + c) - \min(S + c) = \max(S) - \min(S)$$

– Не учитывает масштаб данных (дисперсию)

- **Параметр для нормального распределения:**

$$\text{dist} = \frac{\max(S) - \min(S)}{\alpha \cdot \sigma^2},$$

где σ^2 — выборочная дисперсия

– Учитывает как сдвиг, так и масштаб данных

– Устойчив к линейным преобразованиям $S \rightarrow aS + b$

- **Нормирование для экспоненциального распределения:**

$$\text{dist} = \frac{\max(S) - \min(S)}{\alpha \cdot \sigma},$$

где σ — выборочное стандартное отклонение

– Для экспоненциального распределения $\sigma = \frac{1}{\lambda}$, что позволяет учесть параметр масштаба λ

Итог:

- Для нормального распределения — нормировка на дисперсию (σ^2)
- Для экспоненциального — на стандартное отклонение (σ)
- Стандартный параметр подходит для задач с инвариантностью к сдвигу

3.3. Статистическая мощность и уровень значимости критерия

Пусть задан статистический критерий для проверки гипотезы H_0 (например, выборка соответствует распределению $F_0(x)$) против альтернативы H_1 (выборка не соответствует $F_0(x)$).

[Уровень значимости] Уровень значимости α — вероятность отвергнуть H_0 , когда она верна (ошибка первого рода):

$$\alpha = P(\text{отклонить } H_0 \mid H_0 \text{ верна}) \quad (1)$$

Для критериев согласия α задаёт порог, при котором гипотеза о распределении признаётся несовместимой с данными. *Пример:* При $\alpha = 0.05$ критерий Колмогорова–Смирнова (КС) ошибочно отвергает нормальность данных в 5% случаев, даже если выборка взята из нормального распределения.

[Мощность критерия] Мощность $1 - \beta$ — вероятность отвергнуть H_0 , когда верна H_1 :

$$1 - \beta = P(\text{отклонить } H_0 \mid H_1 \text{ верна}), \quad (2)$$

где β — вероятность ошибки второго рода (не обнаружить отклонение от $F_0(x)$). Для критериев согласия мощность отражает способность обнаруживать *конкретные* отклонения от гипотезы. *Пример:* Критерий Шапиро–Уилка имеет высокую мощность против асимметричных распределений (например, экспоненциального), но низкую против мульти-модальных.

Факторы, влияющие на мощность критериев согласия

- **Объём выборки (n):** Мощность растёт с увеличением n . *Пример:* Для обнаружения смещения $\delta = 0.5$ от $N(0, 1)$ с помощью КС-критерия при $\alpha = 0.05$:
 - При $n = 30$ мощность ≈ 0.4
 - При $n = 100$ мощность ≈ 0.9
- **Характер отклонения от H_0 :** Критерии чувствительны к разным типам отклонений. *Пример:* Критерий хи-квадрат эффективен для дискретных распределений, но теряет мощность при непрерывных альтернативах.
- **Уровень значимости (α):** Увеличение α повышает мощность, но усиливает риск ложных отвержений. *Пример:* Для $H_0 : F(x) = \text{Exp}(1)$ и $H_1 : F(x) = \text{Weibull}(1, 0.8)$:
 - При $\alpha = 0.05$ мощность = 0.75

– При $\alpha = 0.1$ мощность = 0.85

Расчёт мощности для критериев согласия

Мощность критерия согласия можно оценить двумя основными методами:

1. Метод предельного распределения (для параметрических критериев):

- Используется, когда известно асимптотическое распределение статистики критерия
- *Пример:* Для критерия хи-квадрат мощность вычисляется через нормальное приближение:

$$1 - \beta \approx 1 - \Phi(z_{1-\alpha} - \sqrt{n} \cdot D),$$

где D — мера различия между F_0 и F_1 , Φ — функция стандартного нормального распределения

2. Метод Монте-Карло (для любых критериев):

- Шаги:
 - (а) Генерируется M выборок из альтернативного распределения F_1
 - (b) Для каждой выборки вычисляется p -значение критерия
 - (с) Мощность = доля выборок, где $p < \alpha$
- *Пример:* Для критерия Колмогорова–Смирнова (КС) при проверке нормальности (H_0) против t -распределения (H_1):
 - При $n = 100$, $\alpha = 0.05$, мощность ≈ 0.89

3.4. Критические значения и методы их вычисления

Критическое значение c_α — это порог, при превышении которого статистика критерия T приводит к отклонению нулевой гипотезы H_0 .

Формально:

$$P(T \geq c_\alpha \mid H_0) = \alpha,$$

где α — уровень значимости. Вычисление c_α зависит от выбранного подхода:

- **Предельные распределения:** Если статистика критерия имеет известное асимптотическое распределение при H_0 , критические значения берутся из таблиц этого распределения *Пример:* Для критерия хи-квадрат с k степенями свободы:

$$c_\alpha = \chi^2_{1-\alpha}(k),$$

где $\chi^2_{1-\alpha}(k)$ — квантиль распределения хи-квадрат. *Применимость:* Метод работает для больших выборок, когда асимптотика «сходится»

- **Метод Монте-Карло:** Если предельное распределение неизвестно или сложно вычислимо, критические значения оцениваются эмпирически:

1. Генерируется M выборок из распределения F_0 , соответствующего H_0
2. Для каждой выборки вычисляется статистика T_i
3. c_α — $(1 - \alpha)$ -квантиль полученных значений $\{T_i\}$

Пример: Для критерия Андерсона-Дарлинга критическое значение при $\alpha = 0.05$ оценивается как 2.492 (для $n > 30$) *Применимость:* Метод универсален, но требует вычислительных ресурсов

Замечание:

- Для малых выборок асимптотические критические значения могут быть неточными
- Метод Монте-Карло позволяет учесть специфику данных, но требует большого M (обычно $M \geq 10^4$)

Выбор метода зависит от объёма данных и сложности распределения H_0 . Для стандартных критериев (хи-квадрат, t-тест) используют табличные значения, для нестандартных или малых выборок — Монте-Карло.

4. Эксперимент

В рамках этого семестра был поставлен эксперимент измерения мощностей графовых критериев нормальности, основанных на вычислении максимальной степени вершины и количества рёбер в графе, и сравнения их с уже известными критериями.

4.1. Условия эксперимента

В рамках исследования асимптотические распределения статистик критериев не были установлены, поэтому для определения критических значений использовалось моделирование по методу Монте-Карло. Параметр дистанции рассчитывался по формуле:

$$\text{dist} = \frac{\max(S) - \min(S)}{10 \cdot \sigma^2},$$

где S — анализируемая выборка, σ^2 — её выборочная дисперсия.

Оценка мощности критериев выполнялась с использованием инструментов библиотеки проекта⁷. Важно отметить, что на данный момент в проекте не реализована генерация выборок для различных распределений, поэтому для этого используется библиотека *NumPy*, в которой не все распределения реализованы корректно. В будущем планируется написание собственных генераторов в ядре *PySATL*. Параметры эксперимента:

- **Нулевые гипотезы:**
 - Нормальное распределение $\mathcal{N}(0, 1)$
 - Нормальное распределение $\mathcal{N}(10, 20)$
- **Уровень значимости:** $\alpha = 0.05$
- **Критические значения:**

⁷https://github.com/Ivan-Pokhabov/pysatl-experiment/blob/main/experiments/graph_norm_experiment.py

- Двусторонний расчёт методом Монте-Карло ($M = 10^6$ итераций)

- **Альтернативные распределения:**

- Гамма-распределение с параметрами ($\alpha = 1, \beta = 2$)
- Гамма-распределение с параметрами ($\alpha = 0.5, \beta = 1$)
- Экспоненциальное распределение ($\lambda = 2$)
- Распределение хи-квадрат ($df = 5$)
- Распределение Лапласа ($\mu = 0, b = 1$)

- **Характеристики выборок:**

- Количество альтернативных выборок: 100
- Объём выборок: $n = 10, 20, 30, 40, 50$

- **Сравнительный критерий:** Критерий Колмогорова–Смирнова

Для обеспечения статистической достоверности критические значения оценивались на большом числе итераций ($M = 10^6$), что минимизирует ошибку аппроксимации.

4.2. Исследовательские вопросы

RQ1 : Каковы мощности новых критериев на данных альтернативах?

RQ2 : Могут ли новые критерии соперничать с уже существующими критериями?

4.3. Метрика

- Мощность критерия на определенной альтернативе и размере выборки.

4.4. Результаты

- **Критерий EdgesNumber** (число рёбер в графе): Чувствителен к различным альтернативным распределениям, особенно при увеличении объёма выборки ($n \geq 30$)
- **Критерий MaxDegree** (максимальная степень вершин): Наиболее мощен для обнаружения отклонений в форме гамма-распределений ($\alpha = 1, \beta = 2$) и экспоненциального распределения ($\lambda = 2$) на малых выборках ($n = 10 - 20$)
- **Динамика мощности:**
 - При $n = 10 - 20$ новые критерии демонстрируют умеренную эффективность (мощность $0.4 - 0.7$)
 - При $n = 50$ мощность критериев приближается к 1.000 для большинства альтернатив
- **Сравнение с критерием Колмогорова-Смирнова:**
 - Для распределения Лапласа ($\mu = 0, b = 1$) при нулевой гипотезе $\mathcal{N}(10, 20)$ новые критерии превосходят классический метод на 15 – 20%
 - На малых выборках ($n < 30$) традиционный критерий сохраняет преимущество для 80% альтернатив

4.4.1. RQ1

Был проведён эксперимент с вычислением мощностей критериев, с результатами которого можно ознакомиться⁸.

4.4.2. RQ2

В некоторых случаях, например, при нулевой гипотезе $\mathcal{N}(10, 20)$ и альтернативе – распределение Лапласа ($\mu = 0, b = 1$) новые крите-

⁸https://github.com/Ivan-Pokhabov/pysatl-experiment/blob/main/graph_norm_results.pdf

рии показывают себя значительно лучше, чем критерий Колмогорова-Смирнова, но всё же на большей части альтернатив критерий сохраняет значительное преимущество.

4.5. Обсуждение результатов

Эксперименты показали, что графовые критерии:

- Эффективны для специфических альтернатив (тяжёлые хвосты, асимметрия)
- Требуют оптимизации для малых выборок
- Перспективны для задач, где традиционные методы теряют мощность

Дальнейшие исследования планируется направить на:

- Расширение набора графовых характеристик (класеризация, количество треугольников)
- Сравнение с другими непараметрическими критериями (Андерсона-Дарлинга, Крамера-фон Мизеса)

Заключение

В ходе данной работы были выполнены следующие задачи:

- Исследованы и реализованы графовые критерии проверки гипотез согласия
 - Результаты были представлены на международной научной конференции «Процессы управления и устойчивость»
- Развитие библиотеки PySATL-criterion
 - Интегрированы графовые критерии согласия, можно ознакомиться по ссылке в репозитории: <https://github.com/PySATL/pysatl-criterion>
 - Проведён рефакторинг уже существующих критериев⁹
 - Проведён рефакторинг инфраструктуры проекта¹⁰¹¹
- Развитие фреймворка PySATL-experiment
 - Добавлена функциональность для построения табличных отчётов экспериментов по вычислению мощностей критериев согласия¹²
 - Добавлена возможность использования двусторонних критериев¹³
 - Добавлена проверка типов для строгой типизации¹⁴
 - Проведен рефакторинг инфраструктуры проекта¹⁵

⁹<https://github.com/PySATL/pysatl-criterion/pull/15>

¹⁰<https://github.com/PySATL/pysatl-criterion/pull/10>

¹¹<https://github.com/PySATL/pysatl-criterion/pull/11>

¹²<https://github.com/PySATL/pysatl-experiment/pull/28>

¹³<https://github.com/PySATL/pysatl-experiment/pull/16>

¹⁴<https://github.com/PySATL/pysatl-experiment/pull/51>

¹⁵<https://github.com/PySATL/pysatl-experiment/pull/20>

Список литературы

- [1] Chen Hao, Zhang Nancy. Graph-based change-point detection. — URL: <https://www.projecteuclid.org/journals/annals-of-statistics/volume-43/issue-1/Graph-based-change-point-detection/10.1214/14-AOS1269.full> (дата обращения: 4 мая 2025 г.).
- [2] Steele Michael Craig, Neil A Smart Cameron Paul Hurst, Chaseling Janet. Evaluating the statistical power of goodness-of-fit tests for health and medicine survey data. — URL: https://www.researchgate.net/publication/44929281_Evaluating_the_statistical_power_of_goodness-of-fit_tests_for_health_and_medicine_survey_data (дата обращения: 4 мая 2025 г.).
- [3] Wasserstein R. L. Schirm A. L. Lazar N. A. Moving to a world beyond " $p < 0.05$ ". — URL: https://www.researchgate.net/publication/332173354_Moving_to_a_World_Beyond_p_005 (дата обращения: 4 мая 2025 г.).