**«Санкт-Петербургский государственный электротехнический университет**

**«ЛЭТИ» им. В.И.Ульянова (Ленина)»**

**(СПбГЭТУ «ЛЭТИ»)**

|  |  |  |
| --- | --- | --- |
| **Направление** | 09.03.04 – Программная инженерия | |
| **Профиль** | Разработка программно-информационных систем | |
| **Факультет** | КТИ | |
| **Кафедра** | МО ЭВМ | |
| *К защите допустить* |  | |
| Зав. кафедрой |  | А.А. Лисс |

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

БАКАЛАВРА

Тема: Разработка метода и алгоритма генерации условно-реальных данных нейтронных мониторов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент |  |  |  | А.И. Мусаев |
|  |  | *подпись* |  |  |
| Руководитель | д.т.н, профессор каф. МО ЭВМ |  |  | А.Р. Лисс |
|  | *(Уч. степень, уч. звание)* | *подпись* |  |  |
| Консультанты | к.т.н, ассистент  каф. МО ЭВМ |  |  | Б.С. Мандрикова |
|  | *(Уч. степень, уч. звание)* | *подпись* |  |  |

Санкт-Петербург

2025

**ЗАДАНИЕ**

**на выпускную квалификационную работу**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Утверждаю | | | | | | | |
|  | | Зав. кафедрой МО ЭВМ | | | | | | | |
|  | | \_\_\_\_\_\_\_\_\_\_\_\_ А.А. Лисс | | | | | | | |
|  | | «28» февраля 2025 г. | | | | | | | |
| Студент | Мусаев А.И. | | | | |  | Группа | 1304 | |
| Тема работы: Разработка метода и алгоритма генерации условно-реальных данных нейтронных мониторов | | | | | | | | | |
| Место выполнения ВКР: СПбГЭТУ «ЛЭТИ» каф. МО ЭВМ | | | | | | | | | |
| Исходные данные (технические требования):  Необходимо разработать алгоритм генерации временных рядов для нейтронных мониторов и создать приложение с графическим интерфейсом | | | | | | | | | |
| Содержание ВКР:  Введение, Обзор предметной области, Формулировка требований к решению и постановка задачи, Алгоритм работы, Графический интерфейс, Исследование разработанного инструмента, Заключение | | | | | | | | | |
| Перечень отчетных материалов: пояснительная записка, иллюстративный материал. | | | | | | | | | |
| Дополнительные разделы: Безопасность жизнедеятельности | | | | | | | | | |
|  | | | | | | | | | |
| Дата выдачи задания | | | | Дата представления ВКР к защите | | | | | |
| «28» февраля 2025 г. | | | | «17» июня 2025 г. | | | | | |
|  | | | |  | | | | | |
| Студент | | |  | | А.И. Мусаев | | | |  |
| Руководитель д.т.н, профессор  каф. МО ЭВМ | | |  | | А.Р. Лисс | | | |  |
| *(Уч. степень, уч. звание)* | | |  | |  | | | |  |
| Консультант д.т.н, профессор  каф. МО ЭВМ | | |  | | Б.С. Мандрикова | | | |  |
| *(Уч. степень, уч. звание)* | | |  | |  | | | |  |

**календарный план выполнения**

**выпускной квалификационной работы**

|  |  |
| --- | --- |
|  | Утверждаю |
|  | Зав. кафедрой МО ЭВМ |
|  | \_\_\_\_\_\_\_\_\_\_\_\_ А.А. Лисс |
|  | «28» февраля 2025 г. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | Мусаев А.И. |  | Группа | 1304 |
| Тема работы: Разработка метода и алгоритма генерации условно-реальных данных нейтронных мониторов | | | | |

|  |  |  |
| --- | --- | --- |
| № п/п | Наименование работ | Срок выполнения |
| 1 | Обзор литературы по теме работы | 28.02 – 07.03 |
| 2 | Обзор предметной области | 08.03 – 15.03 |
| 3 | Формулировка требований к решению и постановка задачи | 16.03 –16.03 |
| 4 | Алгоритм работы | 17.03 –31.03 |
| 5 | Графический интерфейс | 01.04 –14.04 |
| 6 | Программная реализация | 14.04-14.04 |
| 7 | Исследование разработанного инструмента | 15.04 – 30.04 |
| 9 | Дополнительный раздел | 01.05-08.05 |
| 8 | Заключение | 09.05-15.05 |

|  |  |  |
| --- | --- | --- |
| Студент |  | А.И. Мусаев |
| Руководитель д.т.н, профессор  каф. МО ЭВМ |  | А.Р. Лисс |
| *(Уч. степень, уч. звание)* |  |  |
| Консультант к.т.н, профессор  каф. МО ЭВМ |  | Б.С. Мандрикова |
| *(Уч. степень, уч. звание)* |  |  |

**РЕФЕРАТ**

Пояснительная записка 65 стр., 19 рис., 16 табл., 24 ист.

УСЛОВНО-РЕАЛЬНЫЕ ДАННЫЕ, ВРЕМЕННЫЕ РЯДЫ, ГЕНЕРАЦИЯ ДАННЫХ.

**Объектом исследования** является алгоритм генерации условно-реальных данных нейтронных мониторов.

**Предметом исследования** является точность алгоритмов генерации условно-реальных временных рядов.

**Целью данной работы** является создание алгоритма генерация временных рядов нейтронных мониторов и приложения для удобной генерации данных.

Современные исследования космической погоды и солнечной активности требуют доступа к данным, предоставляемым нейтронными мониторами. Однако доступ к реальным данным может быть ограничен из-за высокой стоимости и трудности сбора. В связи с этим генерация условно-реальных данных становится важным инструментом для моделирования и прогнозирования влияния солнечной активности на земные и космические технологии.

**ABSTRACT**

Modern research on space weather and solar activity requires access to data provided by neutron monitors. However, access to real-world data may be limited due to high costs and the difficulty of data collection. In this regard, the generation of conditionally realistic data becomes an important tool for modeling and forecasting the impact of solar activity on terrestrial and space technologies.

**СОДЕРЖАНИЕ**

Оглавление

[ВВЕДЕНИЕ 9](#_Toc196173786)

[1 Обзор предметной области 11](#_Toc196173787)

[1.1 Влияние космической погоды 11](#_Toc196173788)

[1.2 Сбор космической погоды 12](#_Toc196173789)

[1.3 Формальная постановка задачи и описание решаемой проблемы 13](#_Toc196173790)

[1.4 Существующие методы генерации 14](#_Toc196173791)

[2. Формулировка требований к решению и постановка задачи 24](#_Toc196173792)

[2.1 Постановка задачи 24](#_Toc196173793)

[2.2 Требования к решению 24](#_Toc196173794)

[2.3 Обоснование требований к решению 24](#_Toc196173795)

[2.4 Алгоритм работы 25](#_Toc196173796)

[3 Алгоритм работы 26](#_Toc196173797)

[3.1 Данные для алгоритма 26](#_Toc196173798)

[3.2 Создание усредненных данных 27](#_Toc196173799)

[3.3 Выделение тренда 27](#_Toc196173800)

[3.4 Добавление аномалий в тренд 29](#_Toc196173801)

[3.5 Добавление шума 31](#_Toc196173802)

[4. Графический интерфейс 32](#_Toc196173803)

[4.1 Запуск приложения 32](#_Toc196173804)

[4.2 Загрузка данных 32](#_Toc196173805)

[4.3 Настройка генерации 34](#_Toc196173806)

[4.4 Генерация данных 37](#_Toc196173807)

[5. Программная реализация 37](#_Toc196173808)

[6. Исследование разработанного инструмента 38](#_Toc196173809)

[6.1 Сравнение с другими методами выделения тренда 38](#_Toc196173810)

[6.2 Сравнение с аналогами 48](#_Toc196173811)

[6.3 Оценка устойчивости метода 49](#_Toc196173812)

[6.4 Обучение модели на основе сгенерированных данных 50](#_Toc196173813)

[6.5 Вывод 53](#_Toc196173814)

[ЗАКЛЮЧЕНИЕ 54](#_Toc196173815)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 55](#_Toc196173816)

[ПРИЛОЖЕНИЕ А (заголовок второго уровня) 57](#_Toc196173817)

**ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ**

ЯП – язык программирования;

IDE – Integrated Development Environment;

ML – Machine Learning;

КМА – кратномасштабный анализ;

SNR - signal-to-noise ratio – отношение сигнал-шум;

GAN - Generative adversarial network –генеративно-состязательная сеть;

LSTM – Long short-term memory – длинная цепь элементов краткосрочной памяти;

Датафрейм – двумерная структура данных как таблица;

## ВВЕДЕНИЕ

Нейтронные мониторы используются для изучения космической погоды, солнечной активности, магнитных бурь и их воздействие на земные и космические технологические системы. Доступ к реальным данным нейтронных мониторов может быть ограничен из-за высокой стоимости сбора или недостаточного объема данных для конкретных исследований [1, 2]. Кроме того, данные часто содержат шумы и сбои, что снижает их количество и качество для исследований [3]. Генерация условно-реальных данных нейтронных мониторов позволяет решить проблему сбора данных.

**Целью данной работы** является создание алгоритма генерация временных рядов нейтронных мониторов и приложения для удобной генерации данных.

**Задачи данной работы:**

1. Создать требования к приложению для генерации данных.
2. Разработать алгоритм генерации данных.
3. Реализовать алгоритм в среде MATLAB.
4. Создать графический интерфейс для работы с алгоритмом.
5. Исследовать разработанное приложение.

**Объектом исследования** является алгоритм генерации условно-реальных данных нейтронных мониторов.

**Предметом исследования** является точность алгоритмов генерации условно-реальных временных рядов.

**Практическая значимость работы** состоит в разработке программного обеспечения, позволяющего создавать синтетические данные в условиях недостаточной доступности или неполноты реальных наблюдений. Такое приложение может использоваться для обучения систем машинного обучения в условиях недостаточности данных.

**Публикации по теме работы:**

* Мусаев А.И. (науч. рук. Мандрикова Б.) Разработка алгоритма генерации условно-реальных данных для адаптации моделей машинного обучения к малому набору натурных данных // Сборник тезисов докладов конгресса молодых ученых. Электронное издание. – СПб: Университет ИТМО, [2025]. URL: <https://kmu.itmo.ru/digests/article/15331>
* Мусаев А.И., Разработка метода и алгоритма генерации условно-реальных данных нейтронных мониторов // Сборник докладов студентов и аспирантов научно-технического семинара кафедры МО ЭВМ СПбГЭТУ «ЛЭТИ» – 2025.

## 1 Обзор предметной области

### 1.1 Влияние космической погоды

Жизнь на Земле неразрывно связана с Солнцем. Солнце – Земля – это устойчивая система. Поэтому отдельные явления солнечной активности очень сильно возмущают Землю и околоземное пространство[1]. Космическая погода – это все такие условия в околоземном космическом пространстве, которые в основном определяются Солнцем и его активностью.

Основными постоянными источниками космической погоды являются:

* Солнечный ветер – он дует постоянно и растекается в гелиосферу (околосолнечное пространство). Достигая магнитосферы сферы Земли, взаимодействует с ней, возмущает её. Из-за него возникают различные явления, например, магнитные бури, полярные сияния и т.д.
* Свет – излучение солнца – самое мощное излучение в видимом оптическом диапазоне. Это то, что все видят – изменения дня, ночи, жизненные циклы и так далее.

Есть эффекты космической погоды, которые возникают случайным образом, причем очень мощные – это солнечные вспышки и выбросы корональной массы.

Солнечные вспышки – это внезапный взрывной процесс выделения большого количества кинетической, световой и тепловой энергии.

Выбросы корональной массы – мощные выбросы вещества из звездной короны.

Космическая погода создает опасность для космонавтов и Международной космической станции:

* Радиационная опасность – солнечные лучи вызывают радиационную опасность, приводя к значительным радиационным дозам.
* Потоки частиц способны вызвать сбои в работе спутников – разрушать важные элементы в электронике.
* Геомагнитные бури приводят к изменению плотности верхних слоев атмосферы, что вызывает дополнительное торможение космических аппаратов.

Также космическая погода влияет на наземные технологии. Например, на энергетические системы и системы радиосвязи – мощные вспышки на солнце образуют поток частиц, которые, доходя до земли, вызывают магнитную бурю, может быть прервана радиосвязь или выведены из строя генераторы электричества[4].

По этим причинам космическую погоду необходимо контролировать и изучать.

### 1.2 Сбор космической погоды

Сбор данных о космической погоде ведется с помощью разных вещей, в том числе нейтронных мониторов. Нейтронный монитор – это наземный детектор, который измеряет количество высокоэнергетических заряженных частиц, попадающих в атмосферу земли из космоса. Нейтронные мониторы измеряют интенсивность космических лучей, попадающих на земле, и её изменение со временем.

Нейтронные мониторы собирают различные данные, например, количество солнечных пятен за время (рисунок 1). Обычно это данные вида – число за определенный период, что можно иначе назвать временными рядами.

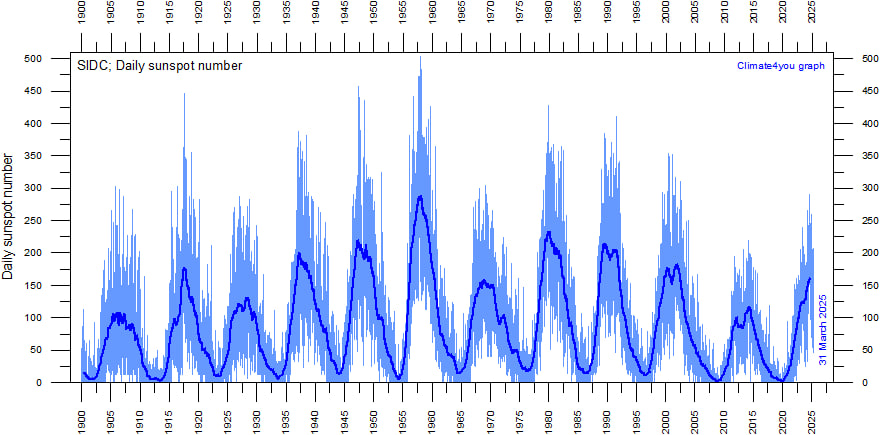


Рисунок 1 – количество солнечных пятен за определенные периоды времени[5]

Однако, исследования космической погоды требуют наличия выборок данных, что получается не всегда[1]. Доступ к натурным данным может быть ограничен из-за высокой стоимости и трудности сбора[2]. Кроме того, такие данные часто содержат шумы и пропуски, что снижает их количество и качество для исследований [3]. В связи с этим, генерация условно-реальных данных становится важным инструментом для моделирования и прогнозирования влияния солнечной активности на земные и космические технологии.

### 1.3 Формальная постановка задачи и описание решаемой проблемы

Исследования в области космической погоды требуют большого количества данных, но в многих случаях:

* Объем данных недостаточен
* Данные содержат шумы
* Невозможно получить данные, охватывающие экстремальные события

Чтобы решить данную проблему предлагается разработать генератор условно-реальных временных данных.

В данной работе будет создаваться генератор временных рядов. Эти ряды должны быть максимально похожи на изначальные по структуре, содержать те же закономерности, показатели.

Кроме того, помимо повторения свойств данных должна иметься возможность редактировать их, например, позволять добавлять аномальные события или шум.

Формально задачу можно записать следующим образом. Имеется матрица временных рядов – формула 1.

где каждая строка – это наблюдение за определенный период (например, за день).

И требуется построить множество синтетических временных рядов – формула 2.

где – изначальные закономерности данных, – шум или аномалия.

Ниже приведен обзор существующих методов генерации временных и сравнение их по введенным критериям.

### 1.4 Существующие методы генерации

#### 1.4.1 Классификация методов

Для анализа были отобраны приложения, реализующие методы генерации синтетических временных рядов.

*A. Visualising forecasting algorithm performance using time series* [4] – генерация основанная на генетическом алгоритме. Для поданных рядов происходит подсчет метрик и создание датафрейма, где каждому ряду соответствуют столбцы его метрик. С помощью PCA (алгоритм понижения размерности) происходит понижение размерности до 2 этого датафрейма с метриками. С помощью генетического алгоритма происходит создание новых рядов, похожих на выбранные. В качестве метрики для алгоритма берется евклидово расстояние до точки в новом датафрейме.

*B. Correlated synthetic time series generation for energy system simulations using Fourier and ARMA signal processing* [5] – в данной статье предлагается сначала выделить из данных все сезонные составляющие с помощью преобразования Фурье, затем для оставшегося шума построить линейную модель, которая будет генерировать данные. После генерации предлагается вернуть все сезонные составляющие выделенные ранее.

*C. GAN* [6] - метод основан на взаимодействии двух нейросетей: генератора и дискриминатора. Генератор получает на вход случайный шум и создает временной ряд, а дискриминатор пытается отличить поддельные данные от настоящих. В ходе обучения обе сети соревнуются: генератор учится создавать всё более правдоподобные ряды, а дискриминатор — точнее различать ненастоящие и реальные данные.

*D. LSTM* [7] – метод генерации временных рядов на основе LSTM использует рекуррентную нейросеть с памятью долгосрочных зависимостей для создания синтетических данных. Она может использоваться для создания новых последовательностей на основе обучающих данных.

*E. ARIMA* [8] - метод генерации временных рядов с использованием модели ARIMA (AutoRegressive Integrated Moving Average). Он основывается на предсказании следующего значения по предыдущим.

#### 1.4.2 Visualising forecasting algorithm performance using time series[4]

В данном алгоритме сначала для временного ряда подсчитывается ряд метрик. В статье предлагаются следующий метрики:

1. Special entropy F1 – формула 3

где – спектральная плотность – показывает, насколько сильно каждая частота участвует во временном ряде. Интеграл суммирует все частоты по диапазону от до .

Низкий означает, что временной ряд имеет ярко выраженные частоты (например, регулярные волны, сезонность). Высокий , что частоты вносят вклад равномерно, ряд выглядит случайным или шумным.

1. Strength of trend – формула 4

где – исходный временной ряд, – сезонная компонента, - временной ряд без сезонности, – остатки, шум, var – дисперсия.

Если тренд выражен очень сильно, изменения в данных в основном объясняются трендом. Тренд практически отсутствует, изменения больше связаны с шумом или другими факторами, .

1. Strength of seasonality – формула 5

Где - тренд.

Если ряд сильно сезонный. Временной ряд практически не имеет сезонности, если

1. Seasonal period

Этот параметр, который показывает длину одного полного цикла сезонности во временном ряде. Проще говоря, это интервал времени, через который повторяются закономерности в данных. В работе рассматривается, что такое число только одно для всего ряда.

1. First order autocorrelation – формула 6

Показывает корреляцию между рядом и им же со сдвигом на 1 элемент.

1. Optimal Box-Cox transformation parameter

– оптимальный параметр для преобразования Бокса-Кокса. Это метод преобразования данных, используемый для стабилизации дисперсии (убирания зависимости между средним значением и разбросом) и приближения распределения данных к нормальному.

После подсчитывания всех метрик создается датафрейм, где каждому временному ряду соответствуют его метрики. Далее размерность данного датафрейма понижается с помощью PCA до 2.

Далее, чтобы создать временные ряды, похожие на существующие, случайно генерируется много временных рядов, которые модернизируются с помощью генетического алгоритма. В качестве метрики для каждого поколения берется евклидово расстояние до желаемого временного ряда.

Недостаток данного алгоритма заключается в том, что он может генерировать лишь короткие временные ряды, так как при генерации больших временных рядов будет затратно считать для каждого метрики и, если генерировать их с одинаковым распределением, при увеличении рядов они будут становиться по метрикам всё более похожими друг на друга по ЗБЧ.

#### 1.4.3 Correlated synthetic time series generation for energy system simulations using Fourier and ARMA signal processing[5]

В данной статье предлагается сначала выделить из данных все сезонные составляющие с помощью преобразования Фурье, затем для оставшегося шума построить линейную модель, которая будет генерировать данные. После генерации предлагается вернуть все сезонные составляющие выделенные ранее.

Формула 7 разлагает разлагает исходный сигнал на сумму синусоидальных и косинусоидальных волн с разными частотами. Это позволяет выделить повторяющиеся (периодические) компоненты сигнала, такие как годовые или дневные циклы. Сначала выбираются соответствующие периоды, на которые нужно разложить данные – , для каждого периода определяются частоты и соответствующие амплитуды , , которые лучше всего описывают данные. Итоговый сигнал собирается как сумма всех волн.

После этого берется остаточный сигнал (исходный сигнал , из которых вычтены сезонные составляющие ) и преобразует его в данные, которые распределены как стандартное нормальное распределение.

После чего создается модель авторегрессии – формула 8.

где – сдвиг для шума, а - вес скользящего среднего.

Проблема данной модели в том, что она совсем не учитывает тренд. Кроме того, при генерации данных она будет рекурсивно возвращаться к созданными ею же данными. Ещё будет проблема в том, что шум – это плохие данные для предсказания с помощью регрессии, так как в них маловероятно будут присутствовать закономерности.

#### 1.4.4 GAN

GAN – это нейронные сети, которые генерируют реалистичные образцы данных, на которых они обучались. Обычно GAN используется для генерации реалистичных изображений, но может использоваться и для задачи генерации временных рядов.

GAN работают на основе двух нейронных сетей, которые соревнуются друг с другом в процессе обучения. GAN состоит из 2 компонентов: генератора и дискриминатора. Генератор создает новые данные, его задача создать настолько похожие, чтобы дискриминатор не мог отличить их от настоящих. Дискриминатор пытается отличить сгенерированные данные от реальных.

Обучение GAN происходит следующим образом:

1. Генератор создает данные на основе случайного шума.
2. Дискриминатор оценивает реальные данные и сгенерированные.
3. Обе сети получают обратную связь:

* Дискриминатор старается уменьшить ошибку классификации
* Генератор старается увеличить ошибку предсказания дискриминатора

1. Алгоритм повторяется

Недостатки данного метода в том, что мы никак не можем добавить изменения в генерацию, например, аномалии, потому что тогда дискриминатор не сможет сравнивать их с реальными данными. Кроме того, под каждые новые данные модель нужно обучать заново, что может быть не очень удобно и затратно по времени.

#### 1.4.5 LSTM

LSTM – нейронная сеть, которая работает с обработкой последовательностей. Она работает таким образом, что выбирает, какую информацию сохранить в долгосрочную память и на основе этой информации предсказывает следующие данные.

В случае предсказания временных рядов, нейросеть будет каждый раз предсказывать новые данные на основе предыдущих. Эта нейросеть больше используется для других задач, но можно использовать и в случае временных рядов.

Недостаток сразу виден, он заключается в том, что ошибка предсказания будет накапливаться с каждым новым предсказанием

#### 1.4.6 ARIMA

Модель ARIMA используется для временных рядов. Она работает следующим образом:

* AR (autoregressive)– модель авторегрессии. Она использует просто предыдущие значения для предсказания.
* MA (moving average) – модель скользящего среднего. Она создает окно, с которым идёт по временному ряду и считает некоторые показатели, затем эти показатели добавляются к изначальным данным ещё одним столбцом.
* I (integrated) – в данные добавляются разности между соседними значениями, можно добавить разности между разности, что будет соответствовать производным 1 и 2 порядка.

Модель можно использовать без производных, просто ARMA.

Недостаток у этой модели такой же, как и у предыдущей, она будет накапливать ошибку при каждом новом предсказании из-за того, что основывается на предыдущем.

#### 1.4.7 Сравнительный анализ методов

Основные критерии для сравнения методов генерации были выбраны следующие:

*Гибкость настройки генерации* – критерий показывает, как можно настраивать параметры при генерации временных рядов.

*Поддержка трендов и аномалий* - этот критерий отвечает, учитывает ли данный метод генерации тренд и можно, ли с помощью него добавить аномалии в генерируемый ряд.

*Масштабируемость -* масштабируемость определяет способность

алгоритма эффективно работать с большими объемами данных. С помощью этого критерия можно понять, насколько большие временные ряды можно генерировать и как будет падать качество генерации при увеличении длины ряда.

Сравнение методов выполнено в табл. 1.

Таблица 1 – Сравнение аналогов

|  |  |  |  |
| --- | --- | --- | --- |
| Метод | Критерии сравнения | | |
| Гибкость настройки генерации | Поддержка трендов и аномалий | Масштабируемость |
| A | Этот метод может генерировать временные ряды на основе на существующего ряда, но только учитывая его метрики. При генерации множества рядов можно менять параметры их генерации, меняя случайное распределение, из которого будет происходить генерация. Кроме того, при понижении размерности можно выбирать какая метрика важнее, чтобы не терять корреляцию с ней. Плюс к этому можно выбрать сами метрики, на основе которых будет работать генетический алгоритм. | Данный метод не поддерживает задание тренда, так как он генерирует случайные ряды. Кроме того, он генерирует лишь короткие ряды, на которых плохо будет прослеживаться тренд. Аномалии же он генерирует лучше, потому что при случайной генерации аномалии в любом случае создадутся, но это не контролируемое добавление. | Данный метод не приспособлен к масштабируемости, потому что генерирует лишь короткие временные ряды в силу своего алгоритма. |
| B | Данный метод поддерживает генерацию временного ряда по имеющемуся ряду. Для ряда Фурье можно задать частоту и количество гармоник (еженедельные, ежемесячные и т.д.). Также самостоятельно можно выбрать модель регрессии, которая будет использоваться для предсказания шума. | Данная модель никак не обрабатывает наличие трендов и добавление аномалий. Тренды обработаются только, если ряд стационарный. | Данная модель может генерировать большие временные ряды, но при их генерации она будет рекурсивно возвращаться к созданным ею же данным, что может повлечь проблемы. |
| C | Можно варьировать архитектуру нейросети. | GAN справляется с моделированием трендов в обучающих данных. Однако аномалии он добавлять не может. | GAN масштабируется и может генерировать много данных. |
| D | Можно варьировать архитектуру нейросети. | Модель способна улавливать тренды, так как имеет память. Однако аномалии добавлять не может. | Подходит для генерации как коротких, так и длинных последовательностей. |
| E | Гибкость ограничена – можно настраивать порядок авторегрессии, дифференцирования и скользящего среднего. Однако архитектурно метод ориентирован на линейные зависимости. | Данная модель никак не обрабатывает наличие трендов и добавление аномалий. | ARIMA масштабируется ограниченно – может генерировать длинные ряды, но при этом результат становится зависимым от предыдущих значений, что снижает устойчивость. |

Исходя из таблицы, можно сделать вывод, что нет решения, которое подошло бы в полной мере. У каждого решения есть свои плюсы и свои минусы, но ни одно из них полностью не отвечает критериям.

*Гибкость настройки генерации*: Наибольшую гибкость показывают методы C, D (GAN и LSTM), так как в нейронных сетях можно варьировать архитектуру модели. В методе A (генетические алгоритмы можно задать метрики, на основе которых ряды будут обрабатываться. В методе B можно вручную задать частоты разложения, чтобы выделить нужные гармоники, кроме того, так как это Фурье разложение можно задать до какой гармоники будет разложение. В модели E гибкость ограничена, можно только выбрать модель, на основе которой будет предсказываться следующее значение.

*Поддержка трендов и аномалий:* Метод C справляется с созданием трендов, так как в GAN генератору придется создать данные с похожим трендом, чтобы дискриминатор увидел его. Метод D может сохранить тренд в долгосрочной памяти. Остальные модели плохо справятся с трендом, так как они не рассчитаны на его воспроизведение.

Однако, ни одна из моделей не способна добавить аномалии в тренд. GAN не может из-за архитектуры, так как данные с аномалией будут помечаться дискриминатором как ненастоящие, а в LSTM нельзя самостоятельно загрузить в долгосрочную память нужные данные. Только модель A сможет случайно сгенерировать аномалию, что тоже не подходит.

*Масштабируемость:* Лучше всего масштабируется модель GAN, так как она не будет наращивать ошибку с каждым новым предсказанием. Модель A совсем не подходит для длинных рядов, так как, чем больше случайных данных она будет генерировать, тем ближе они будут друг к другу по метрикам. Модели B и E будут накапливать ошибку предсказания с каждый новым предсказанием, поэтому они не подходят для масштабирования, они больше подходят для предсказания следующего значения, нежели для генерации данных. Метод D тоже будут накапливать ошибку, так как он генерирует на основе предыдущих данных.

Проведённый сравнительный анализ показывает, что наилучшие результаты по совокупности критериев демонстрирует метод на основе нейросетей — GAN. Он обеспечивают высокую гибкость, способность учитывать тренды и масштабируемость, что делает его полезным инструментом для генерации временных рядов в различных задачах, включая моделирование процессов космической погоды.

Метод ARIMA остаётся надёжным решением для простых линейных зависимостей. Метод Фурье + ARMA показывает хорошие результаты при наличии выраженной сезонности. Метод на основе генетического алгоритма (A) эффективен в задачах генерации краткосрочных рядов, однако ограничен в моделировании длительных и сложных структур временных рядов. Метод LSTM не совсем подходит, так как тоже предсказывает рекурсивно, что может быть опасно.

Тем не менее, ни один из рассмотренных методов не предоставляет возможности контролируемого добавления аномалий в структуру тренда.

Кроме того, во всех моделях невозможно явно задать или редактировать тип шума, используемого в итоговых данных. Пользователь не может контролировать тип шума, что ограничивает гибкость данных в сценариях, где шумовая составляющая играет важную роль.

Исходя из вышеперечисленного, было принято решение разработать алгоритм генерации временных рядов, который будет учитывать тренд входных данных, предоставлять возможность явного добавления аномалий с заданными характеристиками, а также позволять редактировать параметры шумовой составляющей. Дополнительно нужно разработать приложение с графическим интерфейсом, которое обеспечит удобную настройку параметров генерации и визуализацию полученных данных.

## 2. Формулировка требований к решению и постановка задачи

### 2.1 Постановка задачи

Необходимо разработать алгоритм генерации условно-реальных временных рядов и создать приложение, через которое можно будет им воспользоваться.

### 2.2 Требования к решению

Алгоритм должен обладать следующими свойствами:

1. Поддержка генерации длинных временных рядов (свыше 10 000 измерений).
2. Возможность генерации ряда на основе существующих данных.
3. Сохранение исходного тренда.
4. Возможность добавления аномалий.
5. Возможность редактирования вида шума в данных.
6. Удобное приложение для работы с алгоритмом.

### 2.3 Обоснование требований к решению

1. Нужно, чтобы приложение было масштабируемым без потери качества данных, так как для обучения моделей может понадобиться большое количество данных, поэтому нужно генерировать большие ряды (свыше 10 000 измерений).
2. Должна быть возможность подать на вход приложению имеющиеся данные и генерировать на их основе новые, а не задавать характеристики вручную.
3. Сохранение тренда при генерации синтетических данных важно, так как именно он задаёт общее поведение данных и определяет долгосрочные зависимости.
4. Возможность вручную добавлять аномалии в синтетические ряды позволяет использовать их для тестирования алгоритмов обнаружения отклонений, оценки устойчивости систем и моделирования редких сценариев.
5. Тип и характеристики шума могут существенно влиять на поведение ряда. Возможность настраивать параметры шумовой составляющей позволяет адаптировать данные под конкретные условия применения.
6. Наличие графического интерфейса существенно упрощает работу с инструментом, позволяя пользователям, не желающим разбираться в коде, работать с приложением.

### 2.4 Алгоритм работы

Генерация будет происходить по следующему алгоритму:

1. Предобработка данных

На первом этапе выполняется агрегирование исходных временных рядов с использованием медианы. Медианные значения рассчитываются по каждому временному шагу на основе набора имеющихся наблюдений (например, за разные дни). Это позволяет снизить влияние выбросов и локальных шумов.

1. Выделение тренда из медианного ряда

Из медианного временного ряда извлекается тренд. Выделенный тренд представляет собой «каркас» будущего синтетического ряда.

1. Добавление аномалий в тренд

На данном этапе пользователю предоставляется возможность добавить в тренд аномалии — резкие отклонения, нехарактерные для выделенного тренда.

1. Добавление шума и генерация новых данных

После формирования базового ряда (тренд + аномалии) в него добавляется шумовая составляющая. С разными поведениями шума формируются новые временные ряды.

Схема работы алгоритма представлена на рисунке 2.

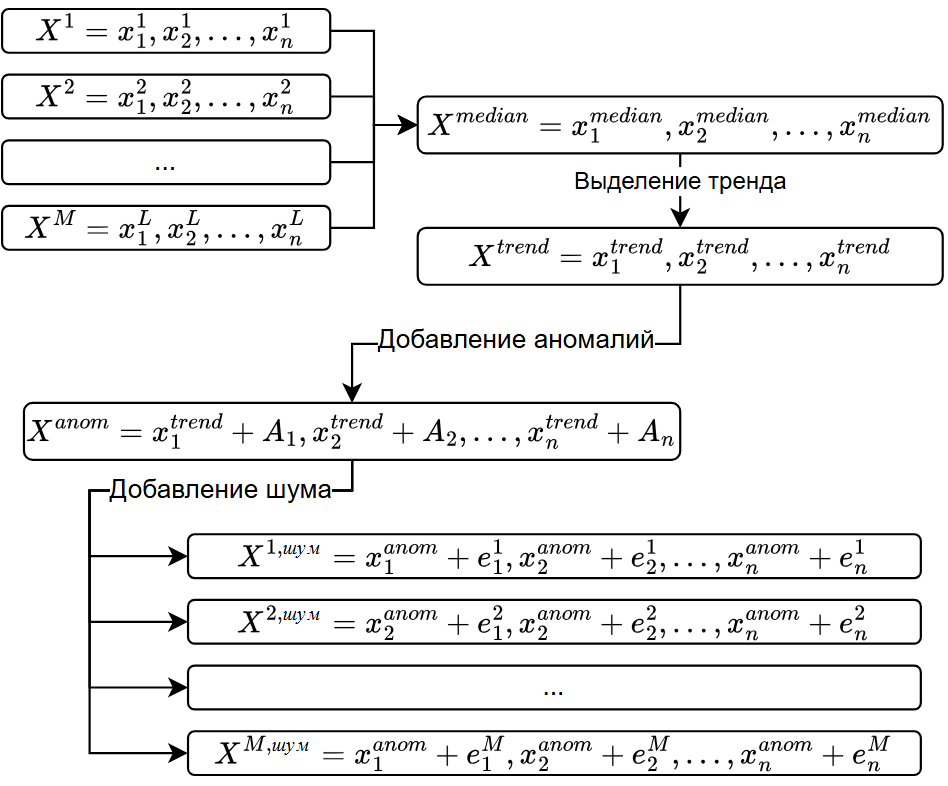


Рисунок 2 – Схема работы алгоритма

## 3 Алгоритм работы

### 3.1 Данные для алгоритма

Входные данные алгоритма представляют собой матрицу, в которой каждая строка соответствует одному полному периоду наблюдений (например, одному дню), а каждый столбец — конкретному моменту времени внутри этого периода.

Иными словами, если есть данные за N дней, и в каждом дне сделано M измерений (например, каждые 5 минут, каждый час и т.д.), то входная матрица будет иметь размер N × M:

* N строк — это количество периодов (дней)
* M столбцов — это количество измерений внутри одного периода

### 3.2 Создание усредненных данных

На первом этапе из входной матрицы вычисляется медианное значение по каждому столбцу. Это означает, что для каждого момента времени внутри периода (например, для каждого часа в течение дня) берется медианное значение по всем доступным дням. Таким образом, формируется один усреднённый период — вектор из M значений, где M — количество измерений в одном периоде (дне).

Использование медианы, а не среднего арифметического, позволяет минимизировать влияние выбросов и локальных аномалий в данных. Полученный медианный ряд представляет собой сглаженную версию типичного дня без резких колебаний и шумов. Этот ряд выступает в качестве подспорья для выделения основного тренда, на который затем будут накладываться дополнительные компоненты: аномалии и шум.

### 3.3 Выделение тренда

После построения медианной строки из входной матрицы данных (то есть усреднённого по всем периодам сигнала), идёт выделение тренда. Это делается с помощью вейвлет-преобразования, которое извлекает медленно изменяющуюся составляющую сигнала, то есть его тренд.

Функция выполняет два ключевых действия:

1. Применяется дискретное вейвлет-преобразование (wavedec) к медианному сигналу, разлагая его на несколько уровней с различной степенью детализации.
2. Восстанавливает только низкочастотную компоненту на заданном уровне (wrcoef) — это и есть выделенный тренд.

Выделение тренда происходит с помощью кратномасштабного анализа (КМА) и вейвлет-преобразования.

Кратномасштабный анализ позволяет получить хорошее разрешение по времени (плохое по частоте) на высоких частотах и хорошее разрешение по частоте (плохое по времени) на низких частотах. Тренд медленно меняющийся компонент, поэтому такие характеристики хорошо подходят для выделения тренда.

Кроме того, использование вейвлетов обеспечивает детектирование локальных особенностей разной формы и длительности, в том числе резких всплесков, особенностей пикообразной формы и др. (подобных природным аномалиям), благодаря широкому набору вейвлет-функций.

Как работают вейвлет-преобразования?

1. Выбирается материнский вейвлет. Он сдвигается по времени и сжимается/растягивается по масштабу – формула 9

где – масштаб, – сдвиг по времени.

1. Вычисляется вейвлет-преобразование – формула 10

Это скалярное произведение между сигналом и масштабирующей вейвлет-функцией. Чем больше значение, тем сильнее присутствует соответствующая частота на данном участке времени.

Почему не преобразования Фурье как в одном из аналогов?

Преобразование Фурье показывает, какие частоты есть в сигнале, но не когда они появляются. Это делает его неэффективным для анализа сигналов и выделения тренда. Оно больше подходит для стационарных сигналов - в которых частотное содержимое не меняется во времени.

Оконное преобразование Фурье частично решает эту проблему, но страдает от фиксированного окна: при высоких частотах оно даёт плохое временное разрешение, а при низких - плохое частотное.

Вейвлеты лишены этого недостатка - они автоматически адаптируют окно: маленькое для быстрых изменений, большое для медленных. Это даёт высокую локализацию во времени и частоте.

На рисунке 3 представлен пример выделенного тренда. На рисунке синим цветом изображены оригинальные данные, красным – выделенным тренд.

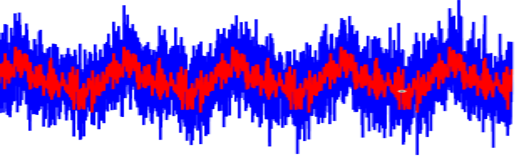


Рисунок 3 – выделенный тренд

### 3.4 Добавление аномалий в тренд

После того как из исходных данных был выделен сглаженный тренд, следующим шагом является добавление аномалий — то есть резких, нестандартных отклонений от общего поведения сигнала. Это необходимо для создания реалистичных сценариев, имитирующих сбои, всплески, пиковые нагрузки и другие нетипичные события, которые могут возникнуть в реальных условиях.

Функция модифицирует тренд, добавляя в указанное место аномалию выбранной формы. Это может быть:

* Импульсный скачок, имитированный по Гауссу – рисунок 4.

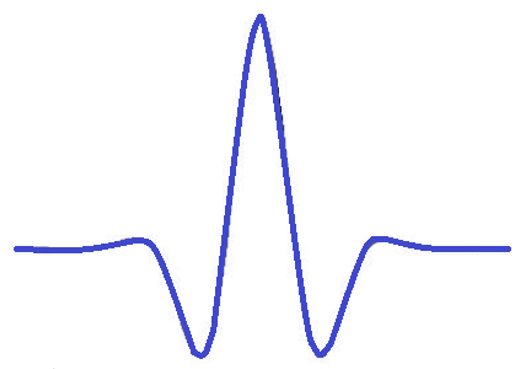


Рисунок 4 – импульс, имитированный по Гауссу

* Треугольная волна, имитирующая постепенное нарастание и спад – рисунок 5.

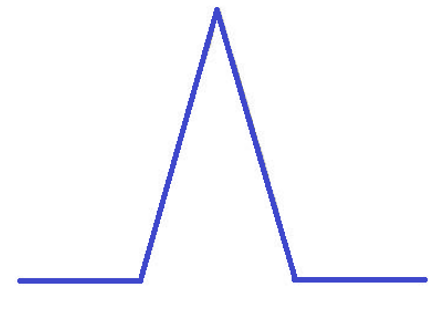


Рисунок 5 – треугольный импульс

* Пользовательская форма, что позволяет воспроизводить специально смоделированные события.

Это позволяет симулировать редкие события и даёт возможность моделировать стресс-сценарии и проверять устойчивость моделей анализа. Кроме того, позволяет избежать переобучения моделей, создавая аномальные данные.

На рисунке 6 – тренд с добавленными аномалиями. Красным – старый тренд, зеленым – новый.

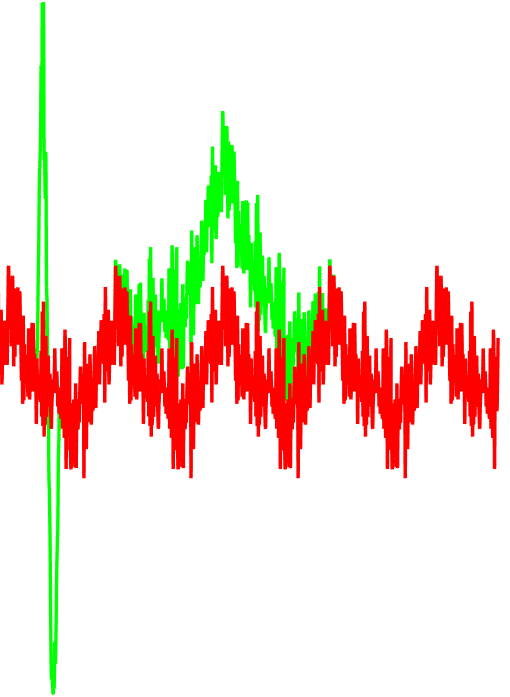


Рисунок 6 – тренд с аномалиями

### 3.5 Добавление шума

После формирования основного тренда и вставки аномалий наступает добавление шумов.

Функция генерирует случайный шум заданного типа и масштабирует его таким образом, чтобы полученное значение соответствовало заданному уровню SNR. Затем этот шум суммируется с трендом, формируя временной ряд. SNR считается по формуле 11. Он нужен, чтобы отношение сигнал – шум было контролируемым.

где – среднеквадратичное значение амплитуды

Поддерживаемые типы шума:

* Белый шум
* Розовый шум

На рисунке 7 – данные после добавления шума, синие данные – итоговые, красные – изначальный тренд, зеленые тренд с аномалиями.

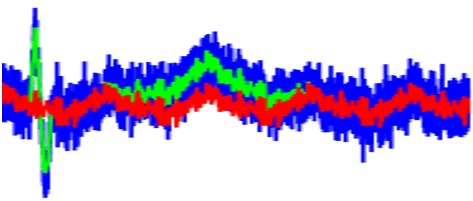


Рисунок 7 – Данные с шумом

## 4. Графический интерфейс

### 4.1 Запуск приложения

В среде MATLAB было также разработано приложение для работы с алгоритмом. Чтобы запустить приложение нужно либо скопировать код из репозитория: <https://github.com/arturmusaev1/TimeSeriesGeneration> и запустить файл setup\_workspace, либо из того же репозитория из папки starting/TimeSeriesGenerator/for\_redistribution\_files\_only скачать и запустить TimeSeriesGenerator.exe, но для этого обязательно должен быть установлен компилятор MATLAB R2024a.

### 4.2 Загрузка данных

На рисунке 8 показан графический интерфейс, открывающийся после запуска приложения.

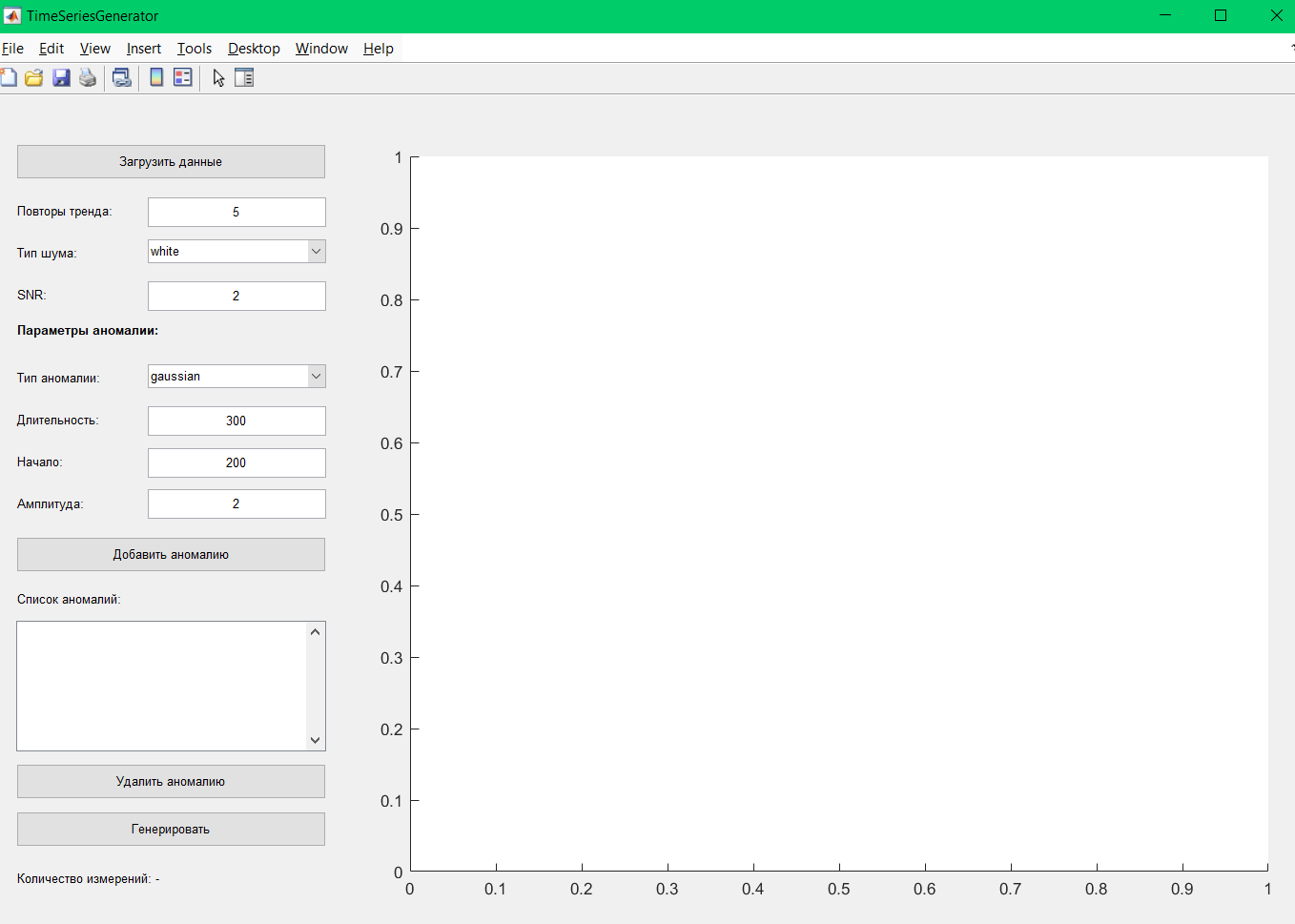


Рисунок 8 – Стартовый экран приложения

Чтобы загрузить данные, нужно нажать на кнопку «Загрузить данные», откроется в файловая система, в которой нужно выбрать файл для загрузки. Можно загрузить либо .mat файл, в которых хранятся MATLAB переменные, либо csv.

Если загружаются csv файлы, как уже писалось выше, это должны быть файлы в виде матрицы. В первой строке файла должны быть номера наблюдений, начиная с 0, ниже строчки с наблюдениями - пример файла представлен на рисунке 6.

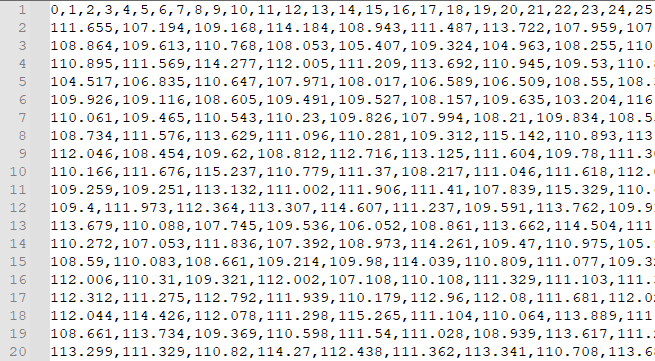


Рисунок 9 – пример csv данных

В примере на рисунке 9 есть 19 строк по 26 измерений в каждой.

Если загружаются .mat файлы, то в файле должна быть переменная Data, в которой должна храниться матрица, из которой будут создаваться данные.

После загрузки данных отобразиться их график и выведется количество измерений в левом нижнем углу – рисунок 10.

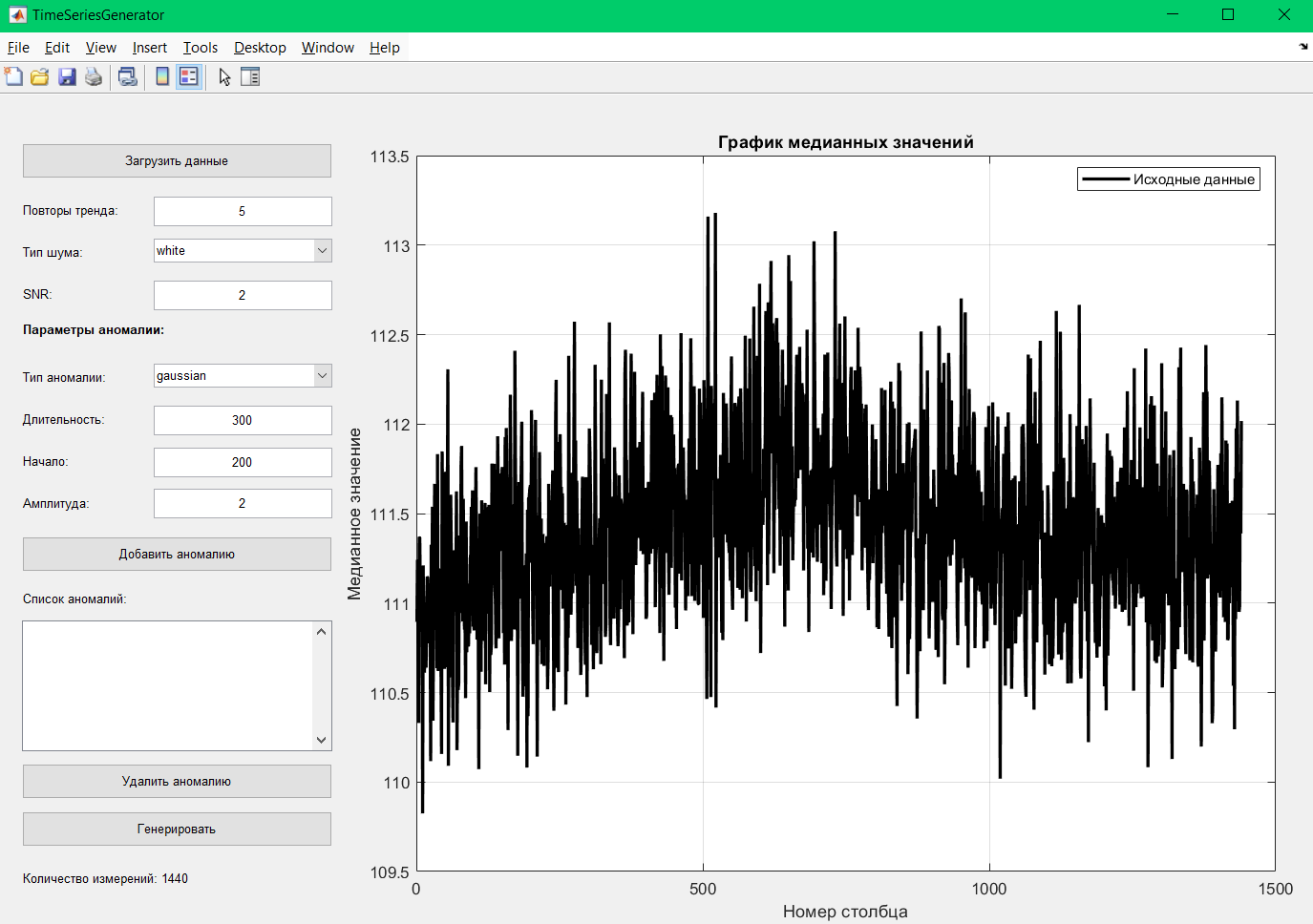


Рисунок 10 – приложение после загрузки данных

### 4.3 Настройка генерации

В окне «Повторы тренда» можно задать количество повторений тренда. Это окно нужно, чтобы указать, сколько новых строк данных генерировать. Допустим, если в оригинальной матрице в каждой строке были дни, то «Повторы тренда» указывают, сколько новых дней нужно сгенерировать. На рисунке 10 – тренд повторен 5 раз.

В окне тип шума можно выбрать white или pink – это белый и розовый шум, соответственно. Это шум, который добавится к тренду при генерации новых данных. В окне SNR можно задать отношение сигнал - шум (считается по формуле 11), рекомендуется задавать от 1 до 2, на рисунке 11 – введено значение 1. При введении значения 0 – будет выведен просто тренд без добавления шума – рисунок 12. На рисунках синим отмечены новые данные, красным – повторенный тренд, черным – оригинальные данные, розовым – оригинальный тренд.

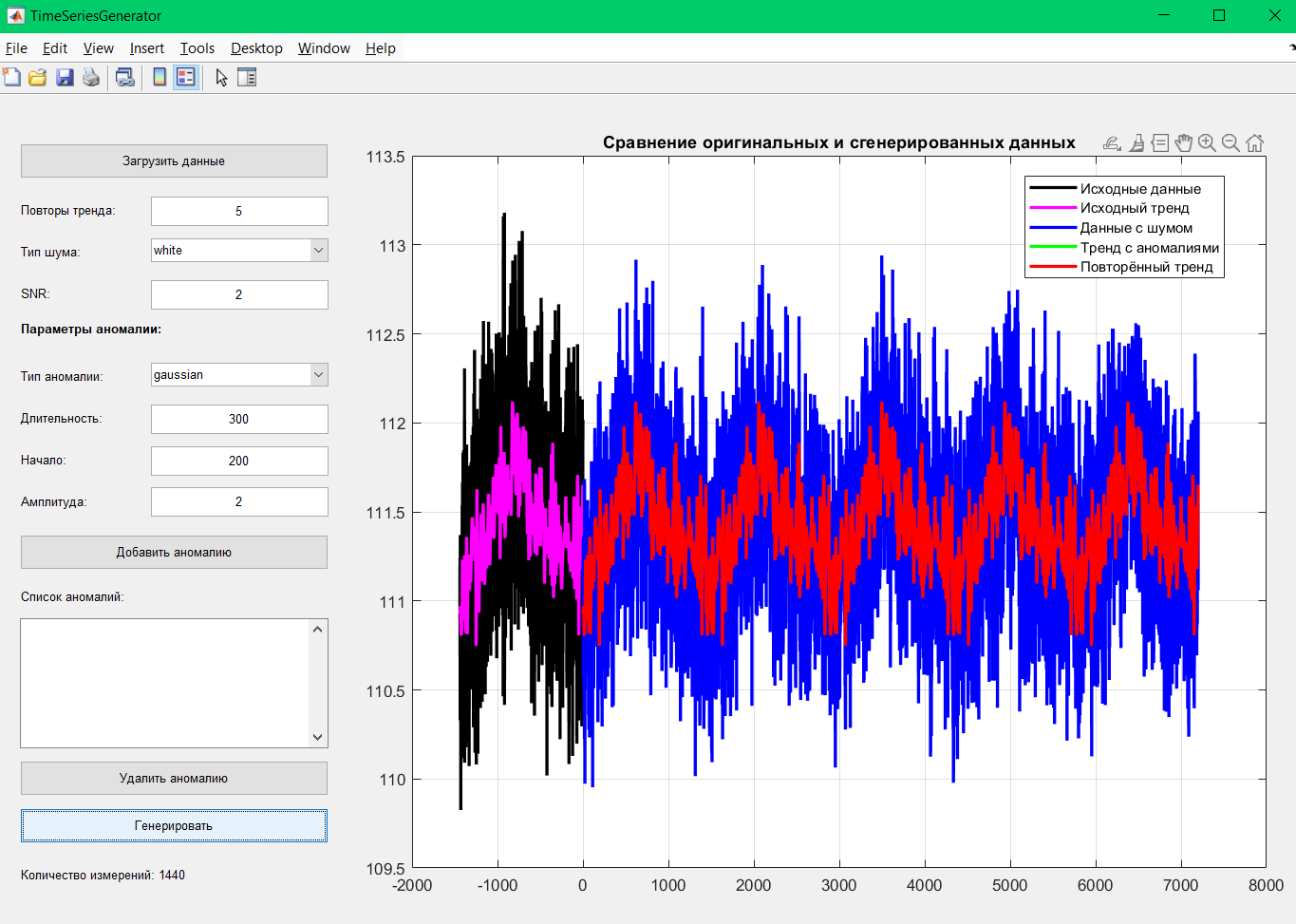


Рисунок 11 – сгенерированные 5 трендов с белым шумом и SNR = 1

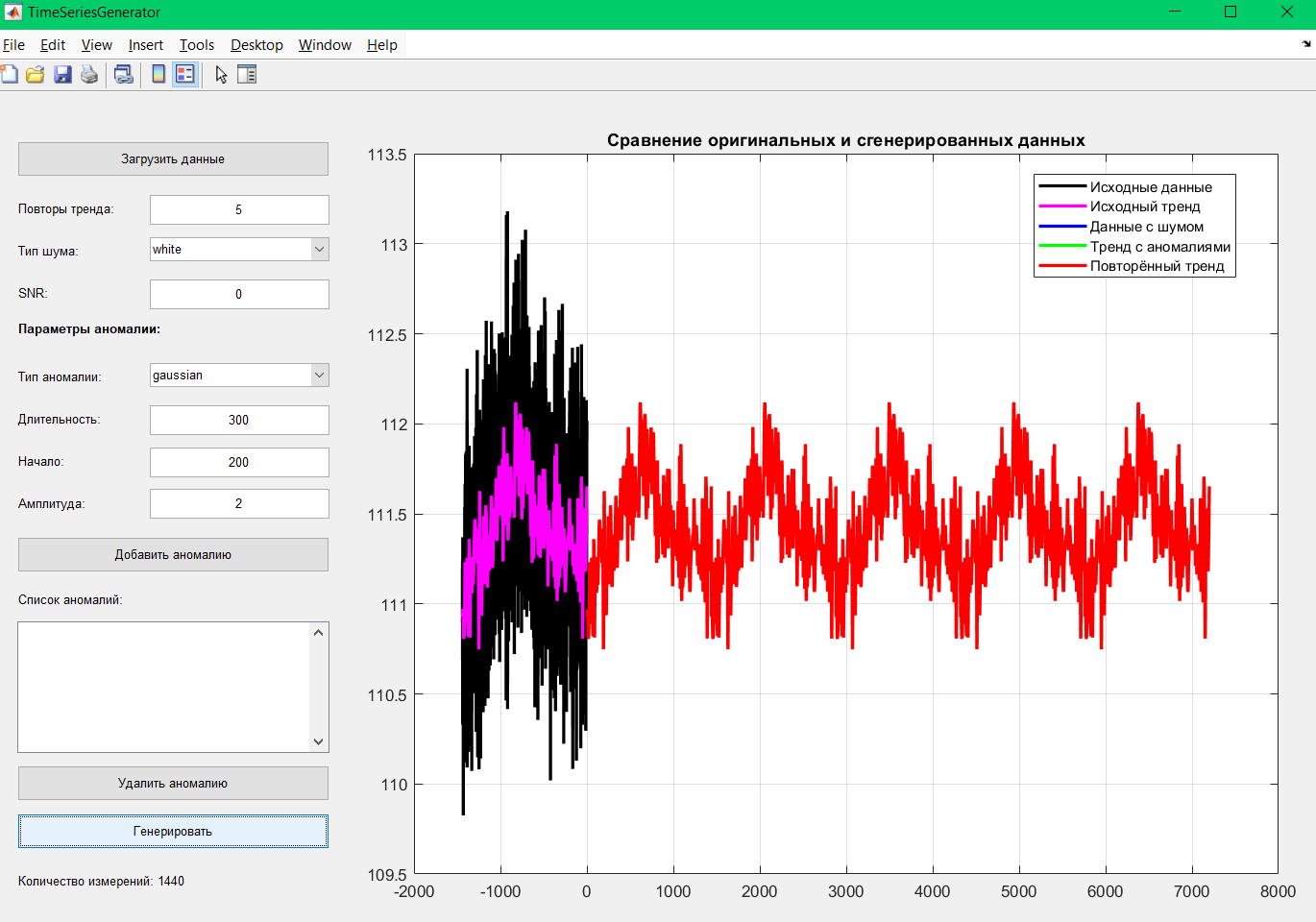


Рисунок 12 – сгенерированные 5 трендов с SNR = 0

Добавление аномалий делается с помощью параметров аномалий. Тип аномалии отвечает за то, какой тип аномалии будет выбран: gaussian, triangle и user, что соответствует имитированной по Гауссу – рисунок 4, треугольной – рисунок 4 и пользовательской.

Для Гауссовой и треугольной можно указать начало, длительность и амплитуду. Начало отвечает за то, с какого момента будет создана аномалия, начало считается в повторенном тренде, то есть можно указать начало аномалии в одной строке, а конец – в другой, это показано на рисунке 13. Зеленым отмечен тренд с аномалией.

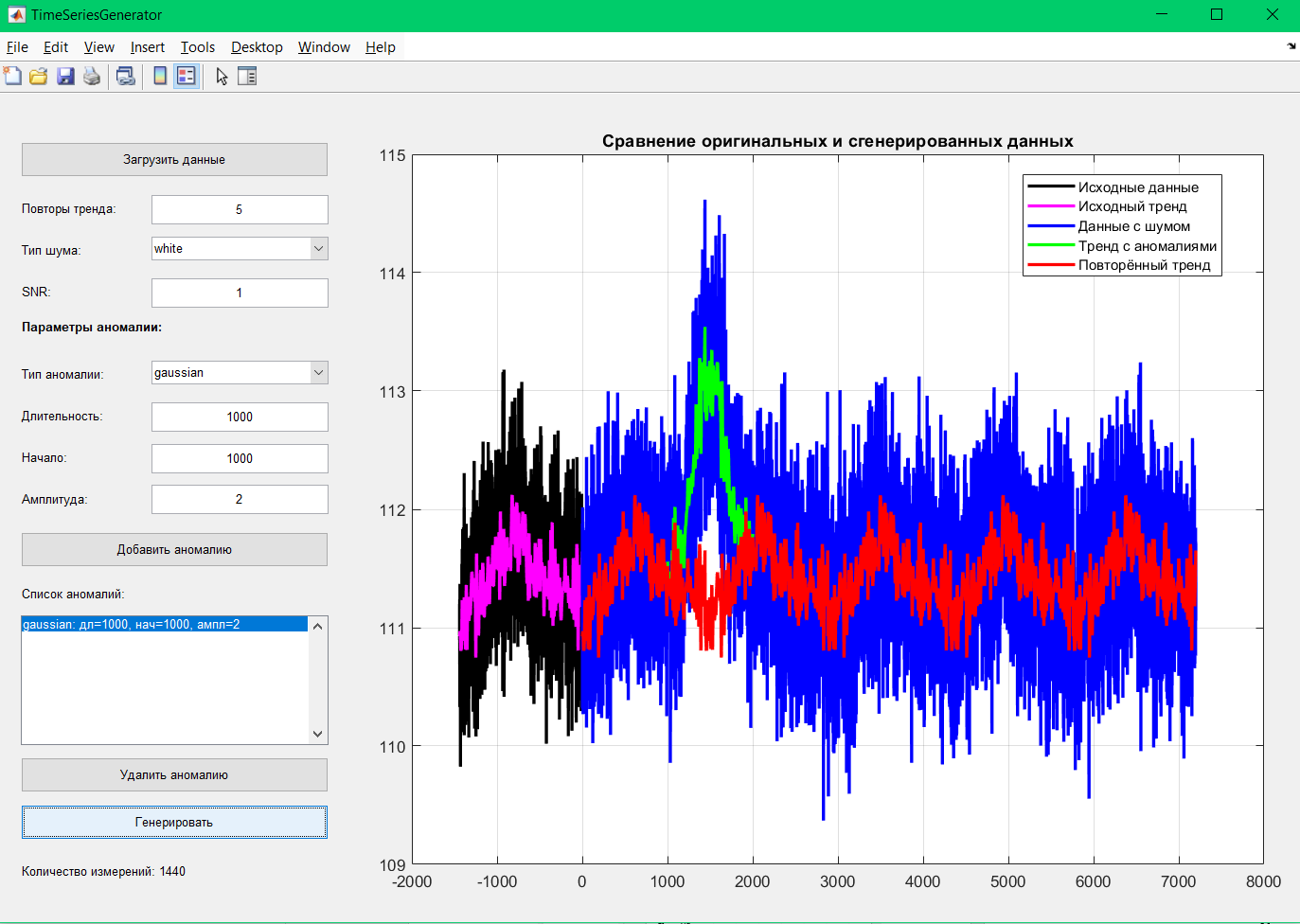


Рисунок 13 – Гауссовый импульс, переходящий из одного повтора в другой

Длительность отвечает за продолжительность аномалии. Амплитуда за коэффициент, на который умножается аномалию при добавлении.

При добавлении пользовательской аномалии нельзя выбрать амплитуду и длительность, можно только загрузить файл и указать начало – рисунок 14. Значения из файла сложатся с трендом.

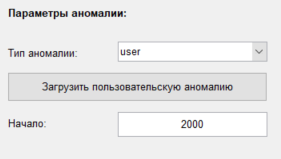


Рисунок 14 – добавление пользовательской аномалии

Чтобы добавить аномалию нужно нажать «Добавить аномалию» и она добавится в список. Выбрав аномалию в списке и нажав «Удалить аномалию», можно удалить аномалию из списка.

### 4.4 Генерация данных

После выставления желаемых настроек и нажатия кнопки «Генерировать» на экране отобразятся новые данные и в папку с программой добавится файл generated\_data.csv, либо generated\_data.mat в зависимости от того, в каком формате данные были загружены.

## 5. Программная реализация

Программа реализована с помощью функционального программирования. Каждое действие вынесено в функции.

Графический интерфейс сделан через элемент uicontrol в MATLAB. Для каждого элемента на экране заданы его координаты. К некоторым кнопкам привязана callback функция, чтобы обновить графический интерфейс сразу, например, кнопка добавления/удаления аномалии, после обновления который графический интерфейс сразу должен обновиться, добавив в список аномалию. Для вывода графика в окне задан элемент axes.

В функции trend\_analysis\_gui задаются глобальные переменны:

* Data – сами данные
* anomalies – список желаемых аномалий
* user\_anomaly\_data – загруженные пользователем аномалии
* file\_ext – расширение файла, чтобы возвращать .mat, либо .csv файл

Сама генерация происходит в функции generate\_data. Сначала происходит агрегирование данных с помощью функции median. Затем для этих медианных вызывается функция trend\_highlighting, которая выделяет тренд.

trend\_highlighting извлекает данные с помощью Вейвлет-преобразования. Она ждет на вход медианные данные, уровень разложения и название вейвлета. Она использует встроенные в MATLAB функции по разложению, затем берет из них трендовые данные.

Далее выделенный тренд копируется столько раз, сколько раз указано в окошке «Повторы тренда», чтобы создать каркас для новых данных.

После запускается цикл по имеющимся аномалиям и они по очереди с помощью функции add\_impulse добавляются в тренд. Функция add\_impulse ждет на вход тренд, тип аномалии, индекс начала, длительность, амплитуду и, если есть, пользовательская аномалия. Гауссовая (рис. 5) и треугольная аномалия (рис. 6) добавляются построением готовых в MATLAB распределение triang и Гауссовое. Если выбрана пользовательская аномалия, то сначала проверяется, что она загружена. Затем полученная аномалия добавляется на нужный участок тренда сложением.

После добавления аномалии идёт добавление шума. Это делается функцией adding\_noise. Она получает на вход тренд, тип шума, SNR. Шум генерируется той же длины, что и полученный тренд и добавляется к тренду.

После всего новый тренд делится на длины изначального тренда, чтобы получить строки новых данных. Наглядно программная реализация видна рисунке 2.

## 6. Исследование разработанного инструмента

### 6.1 Сравнение с другими методами выделения тренда

Сначала нужно провести сравнение с другими методами выделения тренда. Сравнение проводится следующим образом:

Сгенерированы данные, с заранее заготовленным трендом, в них добавлен шум. Было создано 3 вида данных – с линейным трендом, с квадратичным трендом, с экспоненциальным трендом и добавлен шум, сгенерированный по нормальному распределению (от 0 до 10 для линейного и квадратичного и от 0 до 100 для экспоненциального) – рисунок 15

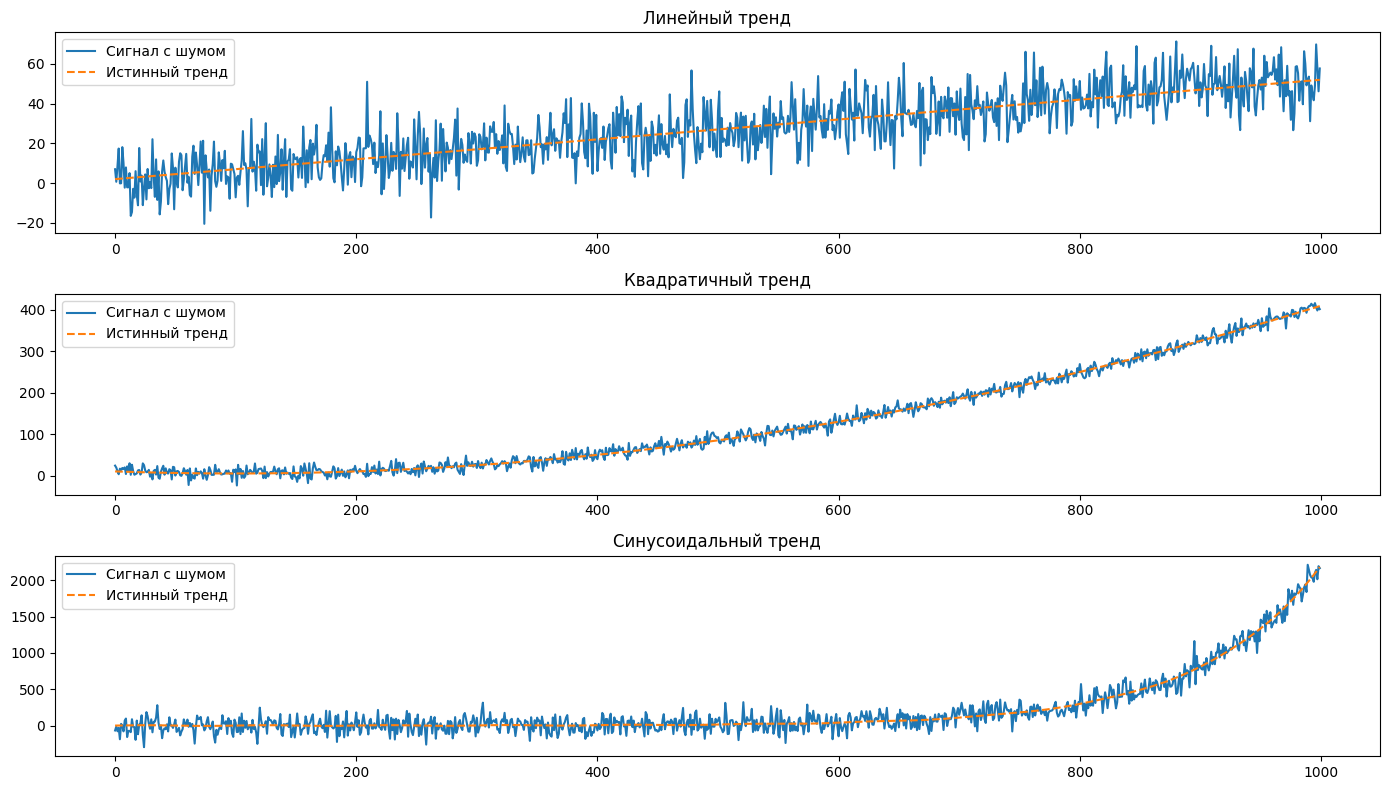


Рисунок 15 – сгенерированные с трендом данные

Сравнение будут проводиться по следующим метрикам:

1. Среднеквадратичная ошибка (MSE) – формула 12
2. Средняя абсолютная ошибка (MAE) – формула 13
3. Средняя абсолютная процентов ошибка (MAPE) – формула 14

где – истинный тренд, – тренд, выделенный оцениваемым методом, – количество точек в ряде.

Сравнение будет проводиться со следующими методами: STL-декомпозиция, метод скользящего среднего, разные Вейвлеты и уровни разложения, полиномиальное разложение.

Одновременно с этим получится определить, какие лучшие параметры для Вейвлет разложения для разных данных.

#### 6.1.1 STL декомпозиция

STL – Seasonal-Trend decomposition using LOESS. Один из самых популярных методов для декомпозиции временного ряда. LOESS – модель линейной регрессии, где вес объекта определяется его расстоянием до точек обучающей выборки. STL разбивает ряд следующим образом – формула 15.

где – тренд, долгосрочная составляющая, – сезонность, – остаток.

#### 6.1.2 Метод скользящего среднего

Простой метод выделения тренда.

Работа:

1. Выбирается окно (например, 15 точек)
2. Для каждой точки временного ряда тренд считается как среднее арифметическое соседних точек внутри окна – формула 16.

где – размер окна.

#### 6.1.3 Полиномиальное разложение

Временной ряд аппроксимируется полиномом заданной степени – формула 17.

где – степень полинома, коэффициенты подбираются методом наименьших квадратов.

#### Поиск лучших параметров вейвлет-разложения

В таблицах 2-4 сравнение метрик для разных вейвлетов.

Таблица 2 – Сравнение вейвлетов для линейных данных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Вейвлет | Уровень разложения | MSE | MAE | MAPE |
| db4 | 3 | 11.959308 | 2.788465 | 18.70% |
| 5 | 2.653848 | 1.1264946 | 8.09% |
| 8 | 1.076281 | 0.830130 | 6.08% |
| beyl | 3 | 11.544864 | 2.695420 | 18.44% |
| 5 | 2.742957 | 1.270492 | 8.13% |
| **8** | **0.706230** | **0.705834** | **3.46%** |
| vaid | 3 | 11.407107 | 2.662839 | 18.51% |
| 5 | 2.829846 | 1.273042 | 8.48% |
| 8 | 1.005602 | 0.820621 | 6.41% |
| haar | 3 | 12.75704 | 2.862852 | 20.23% |
| 5 | 2.406624 | 1.2278172 | 8.28% |
| 8 | 13.784841 | 3.190111 | 22.85% |

Из таблицы 2 видно, что лучше всего с повторением линейного тренда справляется вейвлет beyl – рисунок 16 с 8 уровнем разложения, он показывает лучшие метрики относительно остальных.

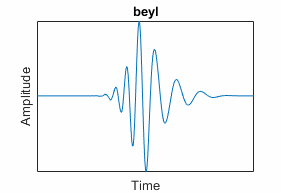


Рисунок 16 – вейвлет beyl

На рисунке 17 представлен график выделенного тренда.

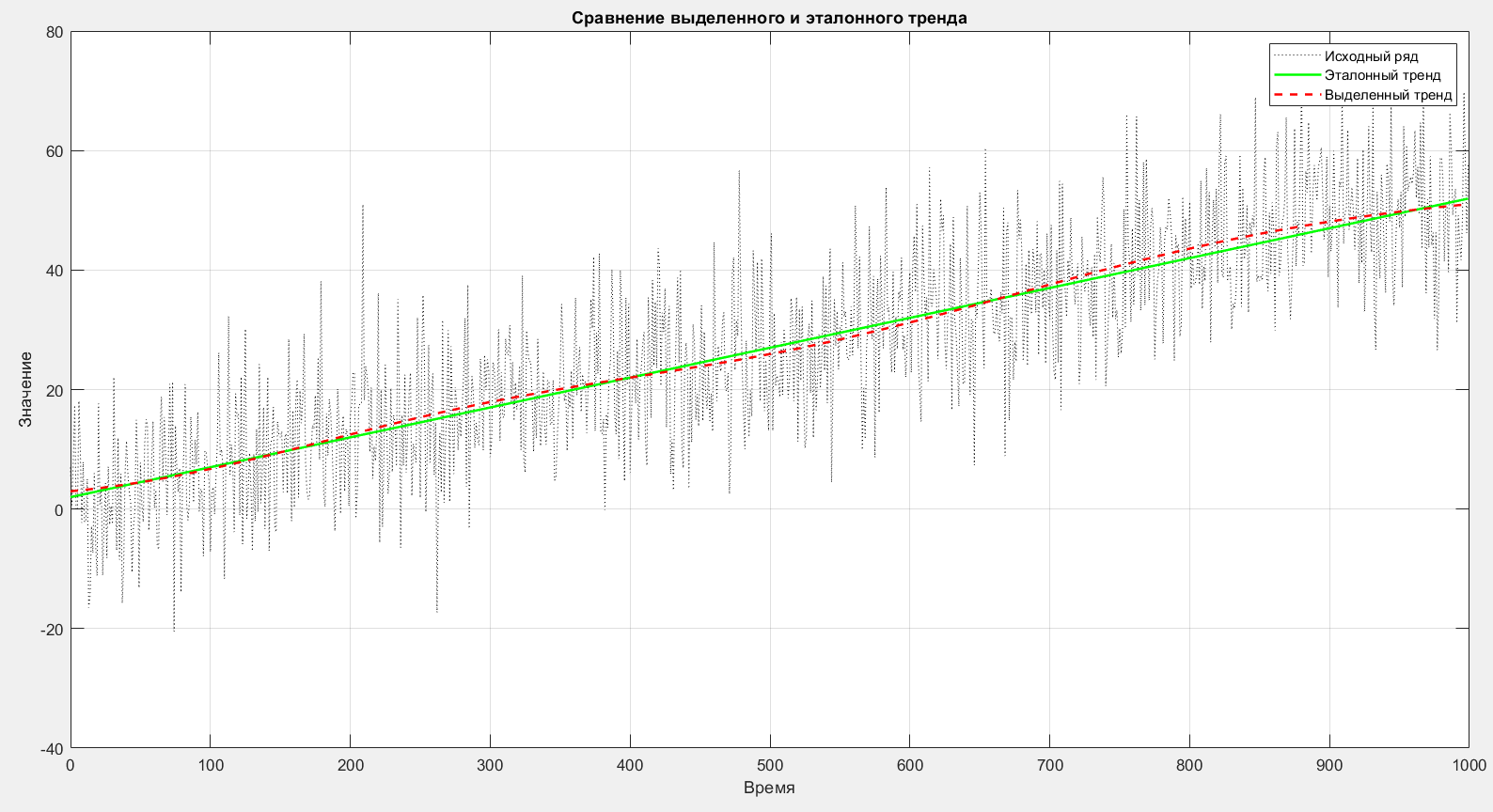


Рисунок 17 – выделенный линейный тренд

Таблица 3 - Сравнение вейвлетов для квадратичного тренда

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Вейвлет | Уровень разложения | MSE | MAE | MAPE |
| db4 | 3 | 13.732338 | 2.923066 | 14.22% |
| **5** | **3.357401** | **1.500893** | **6.39%** |
| 8 | 21.275364 | 3.116204 | 11.69% |
| beyl | 3 | 13.056110 | 2.899685 | 13.90% |
| 5 | 3.383600 | 1.504217 | 6.95% |
| 8 | 22.851595 | 3.038254 | 12.79% |
| vaid | 3 | 12.241787 | 2.830031 | 13.28% |
| 5 | 3.740718 | 1.548390 | 7.04% |
| 8 | 43.280588 | 4.54439 | 7.61% |
| haar | 3 | 11.425183 | 2.696597 | 12.13% |
| 5 | 22.969906 | 3.617626 | 8.06% |
| 8 | 1182. | 24.994929 | 33.58% |

С квадратичными данными лучше всего справился Вейвлет db4 на 5 уроне разложения – рисунок 18.

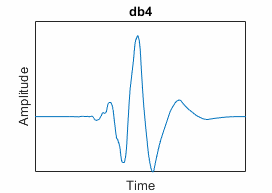


Рисунок 18 – вейвлет db4

Рисунок 19 представляет график выделенного тренда.

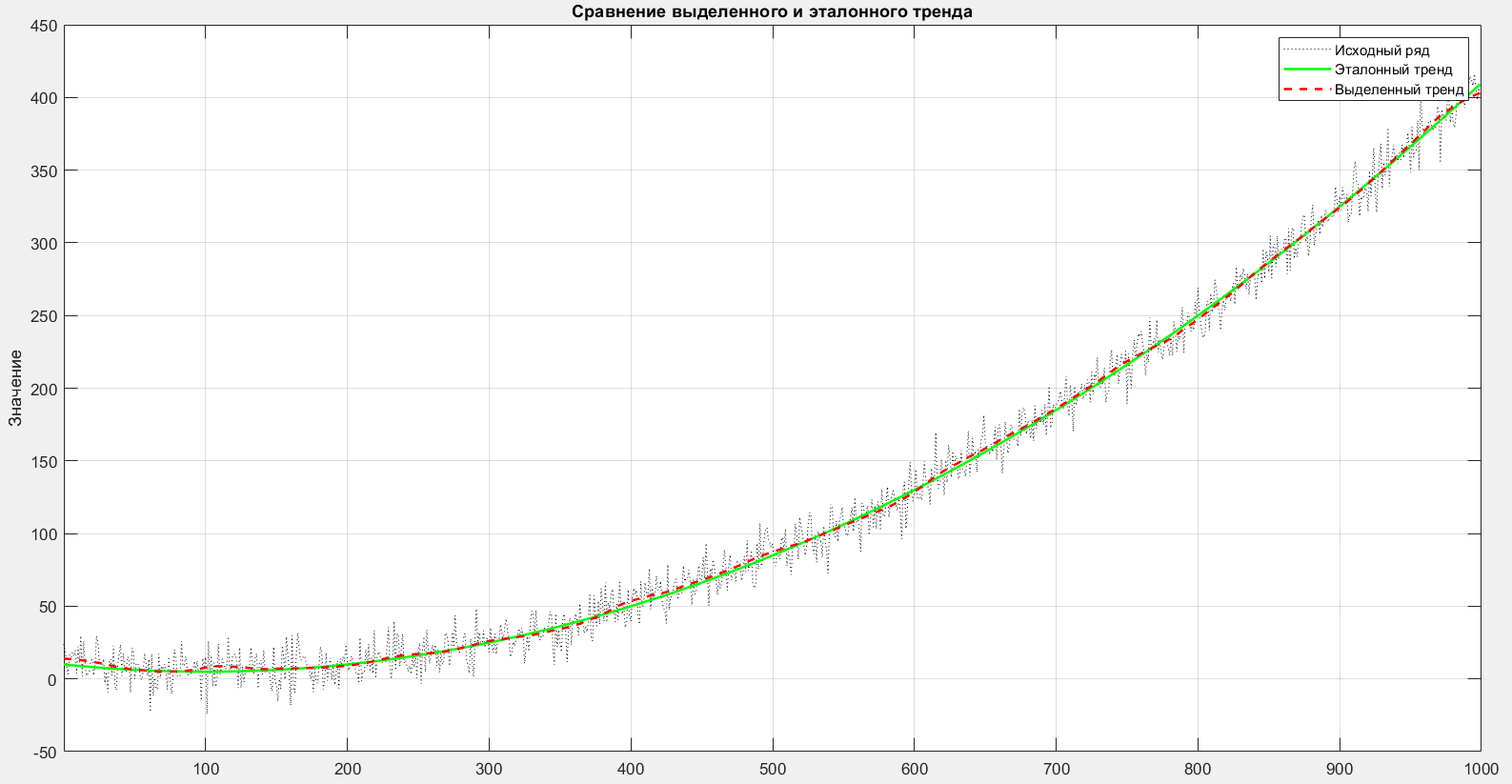


Рисунок 19 – график выделенного квадратичного тренда

Таблица 4 - Сравнение вейвлетов для экспоненциального тренда

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Вейвлет | Уровень разложения | MSE | MAE | MAPE |
| db4 | 3 | 1101.475261 | 26.797283 | 1053.33% |
| **5** | **351.803** | **13.873292** | **670.75%** |
| 8 | 1674.992934 | 20.555419 | 346.82% |
| beyl | 3 | 1106.425773 | 26.871255 | 1075.91% |
| 5 | 375.761856 | 14.498270 | 664.21% |
| 8 | 6557.768889 | 56.207687 | 1069.64% |
| vaid | 3 | 1181.652169 | 27.026819 | 1313.91% |
| 5 | 526.141154 | 15.964850 | 685.78% |
| 8 | 10863.243518 | 66.525762 | 1569.62% |
| haar | 3 | 1442.056770 | 29.778120 | 1182.29% |
| 5 | 2042.430248 | 25.321457 | 574.90% |
| 8 | 72966.553720 | 127.900232 | 293.02% |

Лучшее MAPE показал 8 уровень разложения вейвлета haar, но не всегда можно доверять метрикам. Если посмотреть на график – рисунок 20.

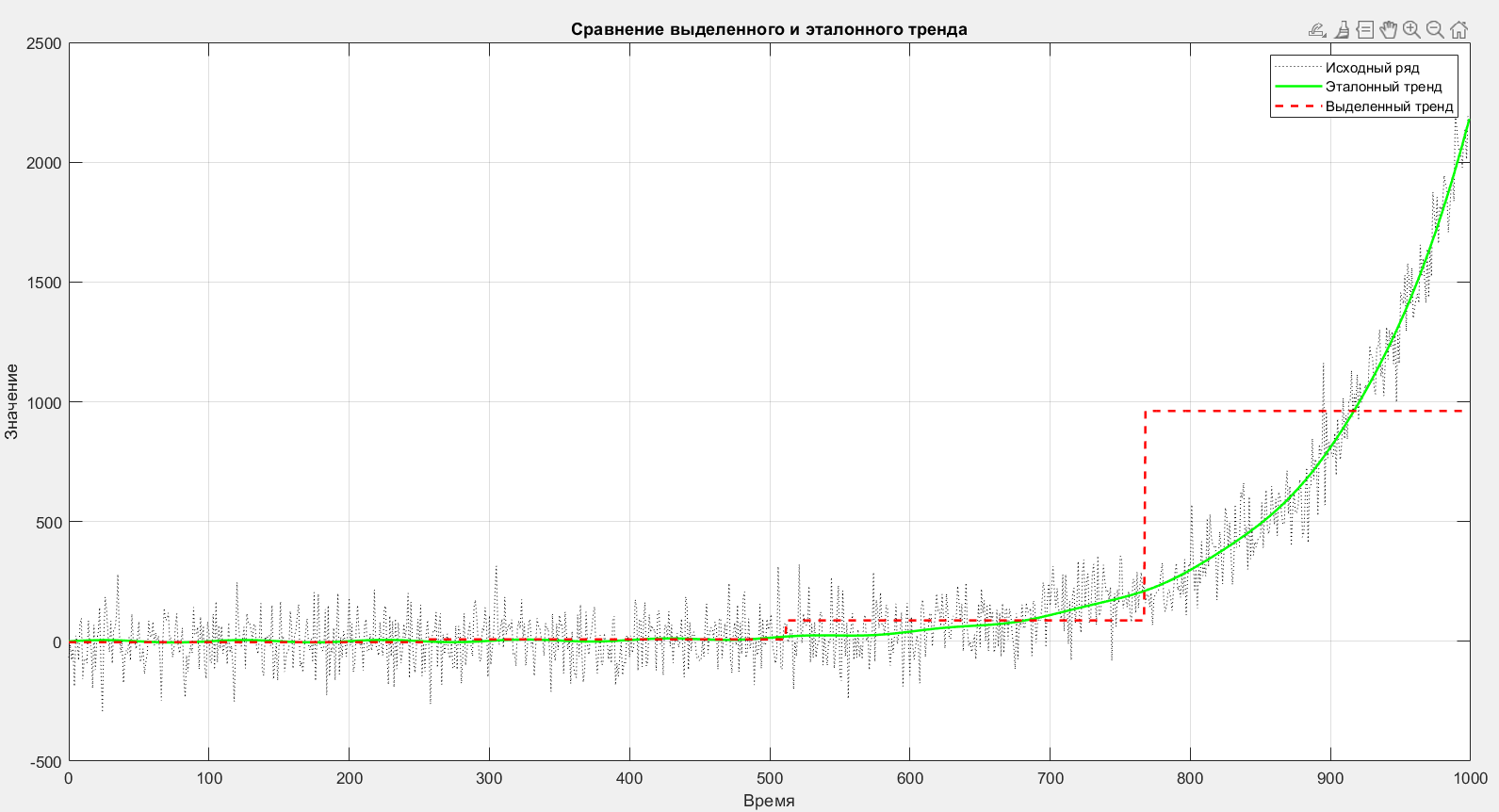


Рисунок 20 - выделенный тренд с помощью haar

На самом деле, тренд выделен плохо, поэтому посмотрим лучший MAE и MSE db4 с 5 уровнем – рисунок 21.

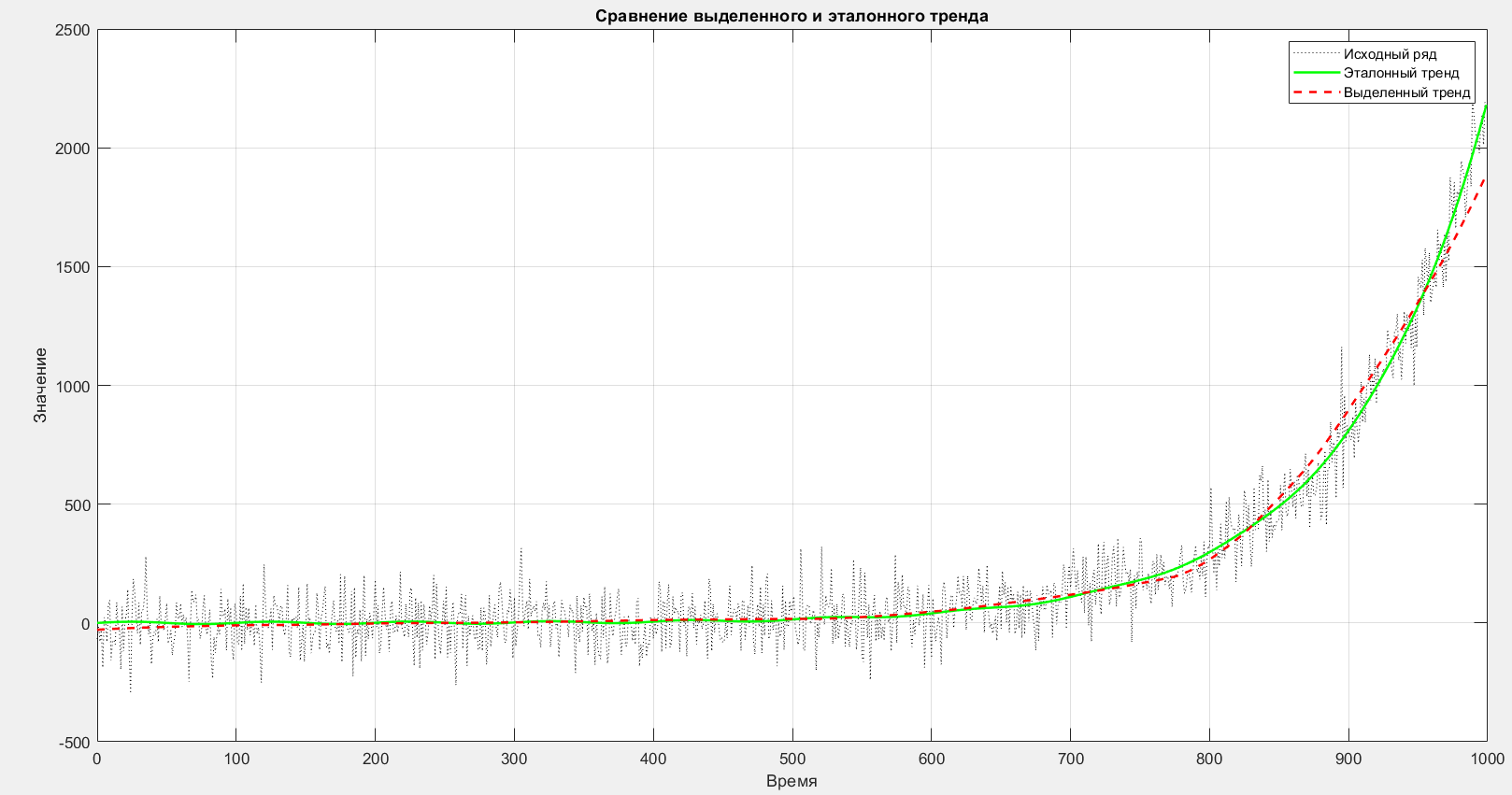


Рисунок 21 – выделенный экспоненциальный тренд

Исходя из вышесказанного, для каждой задачи есть свой подходящий вейвлет и уровень, в каждой задаче разные разложения показывали лучшие показатели. Это один из плюсов вейвлет преобразований, что можно выбирать разные вейвлеты для разложения, чтобы как можно лучше обработать данные.

Для анализа будем использовать 5 уровень разложения вейвлета – db4, так как он показывал хорошие метрики на всех задачах.

#### Сравнение с другими методами выделения тренда

Для сравнения посмотрим, как существующие методы выделения тренда отрабатывают на этих же данных. Эти методы были реализованы на python, так как в нем есть готовые реализации. Для скользящего среднего взять размер окна = 15, для STL период = длина ряда/2, для полинома степень = 15.

В таблицах 5-7 метрики разных методов.

Таблица 5 – Сравнение методов разложения для линейного тренда

|  |  |  |  |
| --- | --- | --- | --- |
| Аналоги | MSE | MAE | MAPE |
| Вейвлет db4 – 5 уровень | 2.653848 | 1.1264946 | 8.09% |
| STL | 0.774651 | 0.756649 | 4.82% |
| Скользящее среднее | 6.376100 | 2.007191 | 12.76% |
| Полиномиальное разложение | 0.248936 | 0.322438 | 2.47% |

В данном случае лучше всех оказалось полиномиальное разложение, так как оно работает на основе линейной регрессии, оно просто «нашло» прямую очень близкую к оригинальной – рисунок 22.

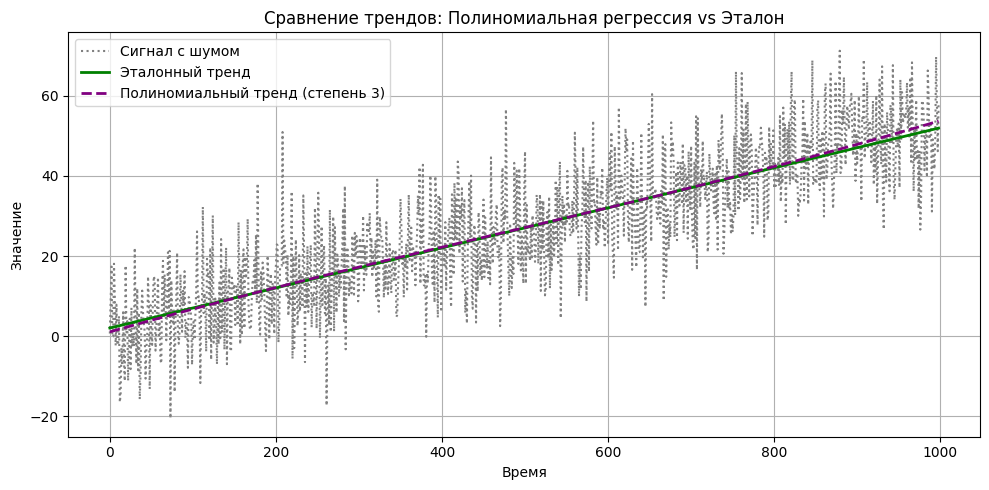


Рисунок 22 – Линейный тренд с полиномиальным разложением

Таблица 6 – Сравнение методов разложения для квадратичного тренда

|  |  |  |  |
| --- | --- | --- | --- |
| Аналоги | MSE | MAE | MAPE |
| Вейвлет db4 – 5 уровень | 3.357401 | 1.500893 | 6.39% |
| STL | 3.327386 | 1.549273 | 8.24% |
| Скользящее сред-нее | 6.718477 | 2.101691 | 10.52% |
| Полиномиальное разложение | 0.653859 | 0.708362 | 5.18% |

В данном случае лучше всего опять проявило себя полиномиальное разложение, так как опять «подобрало» нужную квадратичную функцию – рисунок 23.

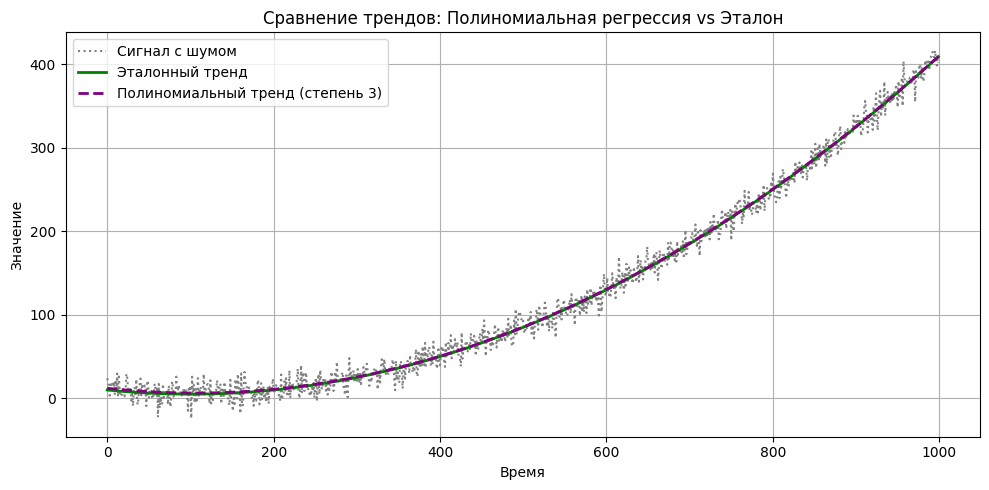


Рисунок 23 – квадратичный тренд с полиномиальным разложением.

Таблица 7 - Сравнение методов разложения для показательного тренда

|  |  |  |  |
| --- | --- | --- | --- |
| Аналоги | MSE | MAE | MAPE |
| Вейвлет db4 – 5 уровень | 351.803 | 13.873292 | 670.75% |
| STL | 4295.258734 | 28.908042 | 118.88% |
| Скользящее сред-нее | 66.481467 | 2.556187 | 79.85% |
| Полиномиальное разложение | 12.384935 | 3.147836 | 98.77% |

В данном случае лучше всех отработала модель скользящего среднего, так как из-за большого шума в этих данных, она убрала шум и остался только тренд.

Исходя из вышесказанного, можно сказать, что вейвлеты показывают хорошее разложение относительно аналогов, конкретно вейвлет db4 и 5 уровень разложения не показывают лучшие метрики относительно остальных, полиномиальное разложение работало лучше, но у полиномиального разложения минус в том, что нужно создавать модель линейной регрессии, которая будет подбирать коэффициенты, кроме того, нужно создавать степени ряда, в данном случае создавались степени до 15, но со сложными трендами может понадобиться степени сильно выше, что будет затратно по времени и по памяти, вейвлеты лишены этого недостатка, поэтому вейвлеты хороший выбор для выделения тренда из ряда.

### 6.2 Сравнение с аналогами

Чтобы сравнить с другими методами генерации временных рядов, каждым методом сгенерированы данные, оценены их характеристики и сравнены с начальными. Генерация данных производилась с помощью GAN, ARIMA и LSTM, так как у них есть готовая реализация в MATLAB. Сравнение проводилось по метрикам: среднее, стандартное отклонение, медиана, минимум, максимум.

В качестве данных использованы: <https://cosmicrays.oulu.fi>. Сайт принадлежит Станции космических лучей Оулу, которая является частью Геофизической обсерватории Соданкюля при Университете Оулу в Финляндии. С 1964 станция непрерывно регистрирует данные о космических лучах с помощью нейтронного монитора. Станция собирает почасовые данные с 1964 года, пятиминутные данные с 1985, минутные данные с 1995. Эта станция измеряет число нейтронов, зафиксированных за час (или другую частоту), которые возникают в результате взаимодействия космических лучей с атмосферой.

Сначала сгенерирован короткий ряд – 24 измерения – 1 сутки.

Были использованы данные 01.01.2011 – 31.12.2012 с поминутными измерениями.

В таблице 8 метрики разных методов генерации для короткой генерации – одного дня.

Таблица 8 – сравнение методов генерации для генерации 1 суток

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Данные | Среднее | Стандартное отклонение | Медиана | Минимум | Максимум |
| Оригинал | 6342.7 | 195.79 | 6349 | 0 | 9137 |
| Сгенерированные новым методом (SNR = 2) | 6349.3 | 15.844 | 6350 | 6307.9 | 6391.4 |
| GAN (30 эпох) | 6493 | 497.38 | 6510.3 | 4731.6 | 7833.4 |
| ARIMA | 6309.5 | 6.5432 | 6309.5 | 6288.9 | 6329.2 |
| LSTM (10 эпох | 4608.8 | 205.66 | 4610 | 3850 | 5256.4 |

Лучше всего повторил данные предложенный метод, он ближе всего повторяет изначальные характеристики. Он плохо повторил минимум и максимум, так как работал на основе медианных данных, в которых такие показатели не сохранились, но это единичные значения, которые не влияют на общий вид.

Далее был сгенерирован длинный ряд – 30 дней. Для сравнения показателей стандартного отклонения были взяты медианы сгенерированных дней – таблица 9.

Таблица 9 - сравнение методов генерации для генерации 30 суток

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Данные | Среднее | Стандартное отклонение | Медиана | Минимум | Максимум |
| Оригинал | 6342.7 | 195.79 | 6349 | 0 | 9137 |
| Сгенерированные новым методом | 6348.82 | 15.488 | 6348.61 | 6290.48 | 6410.20 |
| GAN | 6447.1 | 529.97 | 6457 | 4835.1 | 7770.3 |
| ARIMA | 6304.1 | 13.752 | 6304.4 | 6297.2 | 6313.5 |
| LSTM | 4566.9 | 187.46 | 4566.8 | 4353.5 | 4765.5 |

С увеличением генерации разработанный метод не теряет устойчивости, что хорошо.

Предложенный метод генерации хорошо генерирует данные, он показал хорошие показатели, как на короткой, так и на длинной генерации.

### 6.3 Оценка устойчивости метода

Нужно оценить, насколько хорошо разработанный метод сохраняет свойства при одних и тех параметрах генерации – запустить много генераций с одними и теми же параметрами и оценить, насколько похожи будут характеристики у всех. Будет генерироваться 1 день 12 раз, чтобы оценить разницу между полученными показателями – SNR=2. Ещё проведена оценка для разных SNR. В таблице 10 представлены результаты.

Таблица 10 – сравнение при разных генерациях

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Данные | SNR | Среднее | Стандартное отклонение | Медиана | Минимум | Максимум |
| 1 | 2.5 | 6349.2 | 14.924 | 6349.3 | 6311.1 | 6395.1 |
| 2 | 2.5 | 6348.8 | 15.095 | 6348.2 | 6306.7 | 6390.5 |
| 3 | 2.5 | 6348.5 | 14.696 | 6348.8 | 6307.5 | 6392.4 |
| 4 | 2 | 6348.9 | 15.869 | 6348.9 | 6310.3 | 6394.6 |
| 5 | 2 | 6349.3 | 15.002 | 6349.7 | 6302.3 | 6395 |
| 6 | 2 | 6349 | 15.457 | 6348.6 | 6307.4 | 6393.3 |
| 7 | 1 | 6349.1 | 18.784 | 6349.1 | 6292.5 | 6409.8 |
| 8 | 1 | 6349.4 | 17.951 | 6349 | 6290.6 | 6400.5 |
| 9 | 1 | 6348.7 | 18.095 | 6348.2 | 6293.6 | 6403.1 |
| 10 | 0.5 | 6348.1 | 23.023 | 6348.3 | 6278.9 | 6417.3 |
| 11 | 0.5 | 6348.6 | 22.693 | 6348.3 | 6278.5 | 6427.9 |
| 12 | 0.5 | 6349 | 23.221 | 6349.9 | 6268.3 | 6411.9 |

Видно, что показатели не разнятся между разными генерациями. При уменьшении SNR растёт стандартное отклонение, что логично.

Исходя из вышесказанного можно сказать, что предложенный метод генерации показывает хорошую устойчивость и хорошее повторение показателей по сравнению с аналогами, что хорошо для генерации данных. Кроме того, по сравнению с аналогами можно добавить аномалии и отредактировать шум.

### 6.4 Обучение модели на основе сгенерированных данных

Нужно проверить насколько хорошо сгенерированные данные заменяют реальные, поэтому**,** используялинейную регрессию из пакет sklearn в python будет обучена модель на тренировочных данных, с помощью которой будут предсказываться реальные, затем оценены метрики.

Предсказание проводилось с использованием тех же данных, но они были преобразованы в одну строку для удобства работы. Данные были генерированы с белым шумом и SNR =2.

К уже известным метрикам (формулы 12, 13) добавилась метрика для регрессии - формула 18.

где числитель – сумма квадратов ошибок предсказания, сумма квадратов отклонений от среднего.

Если обучить стандартную модель линейной регрессии без манипуляции с данными, получаем – таблица 11.

Таблица 11 – метрики стандартной модели

|  |  |  |  |
| --- | --- | --- | --- |
|  | MSE | MAE |  |
| Модель обученная на сгенерированных данных | 25014.98 | 123.82 | 0.347 |
| Модель обученная на реальных данных | 24999.63 | 123.70 | 0.348 |

Метрики двух моделей получились очень похожими это говорит о том, что модель очень хорошо повторила данные, что говорит в пользу алгоритма. Проведено сравнение для моделей в данных, которых добавили скользящее среднее с окном 10, производную как разность соседних значений и лаг размером 10. Таблица 12 – новые метрики.

Таблица 12 – метрики улучшенной модели

|  |  |  |  |
| --- | --- | --- | --- |
|  | MSE | MAE |  |
| Модель обученная на сгенерированных данных | 18587.89 | 104.04 | 0.515 |
| Модель обученная на реальных данных | 18900.27 | 103.36 | 0.507 |

В этом случае модель на сгенерированных данных показала метрику немого хуже, чем реальная, но все равно показатели очень похожи.

Для точности эксперимент стоит повторить, но использовать другие данные. Были взяты данные те же данные, но с 2022 по 2024. Опишем эти данные – таблица 13.

Таблица 13 – характеристика данных с 2022 по 2024 и сгенерированных по ним

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Среднее значение | Медиана | Минимум | Максимум | Стандартное отклонение |
| Реальные данные | 6090.29 | 6055.0 | 4155.0 | 7213.0 | 323.16 |
| Сгенерированные данные | 6054.35 | 6053.98 | 5991.51 | 6117.98 | 15.52 |

В таблице 14 характеристики модели без дополнительного преобразования данных.

Таблица 14 - метрики стандартной модели 2022-2024

|  |  |  |  |
| --- | --- | --- | --- |
|  | MSE | MAE |  |
| Модель обученная на сгенерированных данных | 34638.71 | 148.79 | 0.668 |
| Модель обученная на реальных данных | 27392.35 | 131.98 | 0.738 |

В данном случае метрика на сгенерированных данных вышла хуже, чем на реальных, но все равно близкая. В таблице 15 представлена модель с улучшением данных.

Таблица 15 - характеристика данных с 2022 по 2024 и сгенерированных по ним

|  |  |  |  |
| --- | --- | --- | --- |
|  | MSE | MAE |  |
| Модель обученная на сгенерированных данных | 16254.49 | 101.59 | 0.844 |
| Модель обученная на реальных данных | 16075.44 | 101.04 | 0.846 |

В таблице наблюдается приближение метрик и они обе сильно выросли, что хорошо

Можно сделать вывод, что метрика не всегда одинаковая у двух моделей, но ни похожи. В таблице 15 при улучшении данных метрики становятся лучше.

### 6.5 Вывод

В ходе сравнения были рассмотрены разные параметры генерации для алгоритма, можно сказать, что под каждую задачу стоит искать свои параметры вейвлетов и SNR, но в ходе сравнения оптимальными были выбраны SNR = 2 и вейвлет db4 с 5 уровнем разложения. Кроме того, было проведено сравнения с аналогами, которое показало хорошие показатели для предложенного алгоритма. Последним были проведены тесты на предсказании данных, которые показали достаточно хорошие метрики и было выведено, что модели хорошо обучаются как на реальных данных, так и на сгенерированных.

## ЗАКЛЮЧЕНИЕ

В результате выполнения выпускной квалификационной работы была достигнута поставленная цель: разработан алгоритм генерации условно-реальных данных нейтронных мониторов и разработано приложение для работы с этим алгоритмом. Кроме того, приложение было оценено и сравнено с аналогами.

В процессе работы были успешно выполнены поставленные задачи:

1. Созданы и обоснованы требования к приложению для генерации данных.
2. Разработан алгоритм генерации данных. Алгоритм работает на основе выделения тренда с помощью вейвлет преобразований и добавления шума в полученный тренд. Кроме того, есть возможность аномалии и редактировать тип шума при генерации.
3. Алгоритм был реализован в среде MATLAB.
4. Был разработан графический интерфейс для работы с алгоритмом всеми пользователями. Приложение представлено на рисунках 8 – 14.
5. Исследовано разработанное приложение посредством сравнения с аналогами и получением некоторых метрик. В таблицах 8-9 можно увидеть сравнение с аналогами. В таблице 10 можно рассмотреть устойчивость приложения при большом количестве одинаковых генераций. В таблицах 11-14 можно увидеть работоспособность приложения при генерации данных.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Кузнецов В. Д. Космическая погода и риски космической деятельности // Космическая техника и технологии. 2014; 3(6)
2. Бухарицин А.П. Состояние и перспективы развития рынка услуг по сбору и обработке спутниковых данных дистанционного зондирования земли // Международный журнал прикладных и фундаментальных исследований. – 2021. – № 3. – С. 85-91;
3. Балабин Юрий Васильевич, Вашенюк Эдуард Владимирович, Гвоздевский Борис Борисович Шал и множественность на нейтронных мониторах // Вестник Кольского научного центра РАН. 2010. №2.
4. Lakhina, G. & A, Suresh & Tsurutani, B. & Gonzalez, Walter. (2005). Research on Historical Records of Geomagnetic Storms. Proceedings of The International Astronomical Union. 226. 3-15. 10.1017/S1743921305000074.
5. [Электронный ресурс] climate4you : [сайт]. — URL: https://climate4you.com/Sun.htm (дата обращения: 17.04.2025).
6. Kang, Yanfei \& Hyndman, Rob \& Smith-Miles, Kate. (2017). Visualising forecasting algorithm performance using time series instance spaces. International Journal of Forecasting. 33. 345-358. 10.1016/j.ijforecast.2016.09.004.
7. Talbot, Paul \& Rabiti, Cristian \& Alfonsi, Andrea \& Krome, Cameron \& Kunz, M. \& Epiney, Aaron \& Wang, Congjian \& Mandelli, Diego. (2020). Correlated synthetic time series generation for energy system simulations using Fourier and ARMA signal processing. International Journal of Energy Research. 44. 10.1002/er.5115.
8. Train Generative Adversarial Network (GAN) [Электронный ресурс] MATLAB Help Center: [сайт]. — URL: https://uk.mathworks.com/help/deeplearning/ug/train-generative-adversarial-network.html (дата обращения: 15.04.2025).
9. Long Short-Term Memory Neural Networks (LSTM) [Электронный ресурс] MATLAB Help Center: [сайт]. — URL: https://uk.mathworks.com/help/deeplearning/ug/long-short-term-memory-networks.html (дата обращения: 15.04.2025).
10. ARIMA [Электронный ресурс] MATLAB Help Center: [сайт]. — URL: https://uk.mathworks.com/help/econ/arima.html (дата обращения: 15.04.2025).

## ПРИЛОЖЕНИЕ А (заголовок второго уровня)