



# Руководство администратора

---

Программный модуль

PhoenixDS Analytics

## Аннотация

Программный модуль **PhoenixDS Analytics** (далее – ПМ PhoenixDS Analytics), расширяющий базовый функционал Программной платформы PhoenixDS (далее – ПП PhoenixDS), предназначен для анализа и синтеза функций поведения сложных динамических систем и позволяет прогнозировать поведение контролируемых параметров, осуществлять поиск аномалий.

В настоящем Руководстве администратора приведены общие сведения об архитектуре ПМ **PhoenixDS Analytics** и принципах его функционирования, описан порядок установки и удаления его компонент, технология настройки решения математических задач в web-приложении **Редакторе математических моделей**, варианты их применения в ПП PhoenixDS.

Документ рассчитан на администратора, владеющего основами ПП PhoenixDS, изложенными в базовом документе "ПП PhoenixDS. Руководство администратора".



**ДИНАМИЧЕСКИЕ  
СИСТЕМЫ**

Россия, 101066, Москва, ул. Бауманская, д.53

тел.: +7 (495) 228-1100

[www.dynasystems.ru](http://www.dynasystems.ru)

### © ООО «Динамические Системы»

Документ является собственностью ООО «Динамические Системы» и защищен законодательством о правах на результаты интеллектуальной деятельности. Никакая информация, содержащаяся в данном документе, не может быть воспроизведена, искажена, переработана, переведена на иностранный язык, записана или скопирована для любых коммерческих целей. Не допускается передача данного документа третьей стороне без письменного согласия ООО «Динамические Системы». Графические изображения и названия продуктов, упоминаемые в данном документе, могут быть зарегистрированными товарными знаками, охраняемыми законодательством о правах на результаты интеллектуальной деятельности. ООО «Динамические Системы» оставляет за собой право вносить изменения в содержание данного документа без предварительного уведомления.

---

# Типографские соглашения и условные обозначения

В документе приняты следующие шрифты и обозначения:

Примеры использования	Тип информации
<b>Дерево конфигурации</b> – это... кнопка <b>Сохранить</b> пункт меню <b>Конфигурировать</b>	Термины, названия компонент ПП PhoenixDS, наименования элементов пользовательского интерфейса (поля, кнопки, пункты меню).
... выбрать значение <b>off</b> ...	Значения полей, отображаемых в пользовательском интерфейсе.
<code>konfne --devlist</code> ... директория <code>C:\Program Files (x86)\ ...</code>	Команды, фрагменты кода программы, текст файлов.
См. <a href="#">Справочники</a> <sup>[49]</sup> ... с помощью <a href="#">дополнительной настройки</a> <sup>[12]</sup> ...	Ссылка на другой раздел документа. Значок возле ссылки содержит номер страницы, на которой расположен целевой раздел.

В абзацах с заголовком Примечание содержатся дополнительные сведения, например:

Примечание. Таким способом представлена дополнительная информация по теме раздела.

В ограниченные рамкой текстовые блоки с заголовком **ВНИМАНИЕ!** помещены требования или инструкции, невыполнение которых может привести к необратимым последствиям, а также особо важная информация, например:

**ВНИМАНИЕ!** Таким способом выделена информация, которую нужно обязательно принять к сведению.

# Содержание

<b>1. Введение.....</b>	<b>7</b>
1.1. Общие сведения о ПМ PhoenixDS Analytics.....	8
1.1.1. Архитектура ПМ PhoenixDS Analytics .....	8
1.1.2. Технические требования .....	9
<b>2. Понятия и концепции.....</b>	<b>11</b>
2.1. Математические инструменты ПМ PhoenixDS Analytics.....	12
2.2. Задача прогнозирования временного ряда на основании одного временного ряда .....	12
2.3. Задача прогнозирования временного ряда на основе множества временных рядов .....	13
<b>3. Установка и удаление ПМ PhoenixDS Analytics.....</b>	<b>15</b>
3.1. Установка компонент на сервер MADA.....	16
3.2. Установка и настройка компонент на сервере PhoenixDS .....	16
3.3. Установка Web-приложения Редактор математических моделей.....	16
3.4. Удаление компонент ПМ PhoenixDS Analytics.....	16
<b>4. Web-приложение Редактор математических моделей.....</b>	<b>17</b>
4.1. Вход и выход из приложения.....	18
4.2. Основное окно.....	19
4.2.1. Заголовок и главное меню .....	19
4.2.2. Представление Сценарии .....	20
4.2.2.1. Окно Обработки.....	21
4.2.2.2. Область редактирования.....	23
4.2.3. Представление Список задач .....	25
4.2.3.1. Окно Список задач .....	25
4.2.3.2. Таблица выполнения задачи .....	27
4.3. Настройка сценариев.....	27
4.3.1. Создание сценария .....	27
4.3.2. Сохранение сценария .....	28
4.3.3. Импорт сценария .....	29
4.3.4. Дублирование сценария .....	29
4.3.5. Настройка канвы .....	30
4.3.6. Удаление сценария .....	30
4.4. Настройка задач.....	31
4.4.1. Создание задачи .....	31
4.4.2. Изменение задачи .....	35

4.4.3.	Запуск задачи .....	35
4.4.4.	Удаление задачи .....	36
4.5.	Просмотр результата выполнения задачи.....	36
<b>5.</b>	<b>Применение ПМ PhoenixDS Analytics.....</b>	<b>39</b>
5.1.	Настройка прогнозирования поведения параметра целевого объекта.....	40
5.1.1.	Настройка сценария и задачи в Редакторе матем. моделей .....	40
5.1.2.	Настройка набора сигналов в Редакторе сигналов и оповещений .....	42
5.1.2.1.	Параметры сигнала типа MadaCalc .....	46
5.1.3.	Привязка набора сигналов к целевому объекту .....	50
<b>6.</b>	<b>Справочники.....</b>	<b>53</b>
6.1.	Термины и определения.....	54
6.2.	Системные процессы ПК PhoenixDS Monitoring.....	56
6.2.1.	Стандартные параметры системных процессов .....	56
6.2.2.	Обмен данными между ПК PhoenixDS Monitoring и ПМ PhoenixDS Analytics (mada.pl) ...	56
6.3.	Обработчики.....	57
6.3.1.	Ввод/Вывод .....	57
6.3.2.	Виджеты .....	57
6.3.3.	Модели .....	57
6.3.3.1.	Линейный тренд.....	59
6.3.3.2.	Средний тренд.....	59
6.3.3.3.	Тренд скользящего среднего .....	60
6.3.3.4.	ARIMA .....	60
6.3.3.5.	Auto-ARIMA.....	62
6.3.3.6.	ARM .....	64
6.3.3.7.	Рекуррентная нейронная сеть (RNN).....	64
6.3.3.8.	Машина опорных векторов (SVM).....	68
6.3.3.9.	Экстремальный градиентный бустинг (XGBoost).....	70
6.3.4.	Модификаторы .....	71
6.3.4.1.	Удаление интервалов.....	71
6.3.4.2.	Удаление шипов.....	72
6.3.4.3.	Регуляризация временного ряда.....	82
6.3.5.	Нормализация .....	83
6.3.5.1.	Масштабирование данных.....	83
6.3.5.2.	Демасштабирование по ядру.....	84
6.3.5.3.	Демасштабирование по минимуму и максимуму.....	84
6.3.6.	Общие .....	84
6.3.6.1.	Приведение к матрице.....	85
6.3.6.2.	Приведение к множественным временным рядам.....	85
6.3.6.3.	Приведение к регулярным временным рядам .....	85
6.3.6.4.	Вырезать данные .....	85
6.3.6.5.	Загрузить и выполнить модель.....	86
6.3.6.6.	Вставить данные .....	87
6.3.6.7.	Транспонирование .....	87

6.3.7.	Оценщики .....	87
6.3.8.	Партиционирование .....	88
6.3.9.	Постобработка .....	89
6.3.9.1.	Оценка выбросов (интегральный риск) .....	89
6.3.9.2.	Преобразование по диапазону .....	90
6.3.9.3.	Преобразование знака .....	90
6.3.9.4.	Сглаживание .....	92
6.3.9.5.	Расчёт порогов .....	93
6.3.10.	Причинно-следственный анализ .....	98
6.3.10.1.	MC .....	98
6.3.10.2.	Корреляция Пирсона .....	99
6.3.10.3.	Причинность Грейнджера .....	100
6.3.10.4.	DTW .....	100
6.3.10.5.	TCDF .....	101
6.3.11.	Фильтры .....	102
6.3.11.1.	Заполнить NaN .....	102
6.3.11.2.	Фильтр Хампеля .....	106
6.3.11.3.	Удаление выбросов по сплайнам .....	106
6.3.12.	Функции .....	107

<b>Предметный указатель.....</b>	<b>109</b>
----------------------------------	------------

---

# 1.

## Введение

## 1.1. Общие сведения о ПМ PhoenixDS Analytics

ПМ **PhoenixDS Analytics** предназначен для анализа и синтеза функций поведения сложных динамических систем и позволяет с высокой достоверностью прогнозировать поведение контролируемых параметров и осуществлять поиск аномалий. Модуль анализирует статистическую информацию посредством комплексных математических алгоритмов, основанных на теории детерминированного хаоса, используя нейронные сети обратного распространения со многими скрытыми слоями, строит функции поведения исследуемых показателей.

В качестве входных данных могут использоваться временные ряды из разных источников, в том числе и значения параметров, собранных с целевых объектов системными процессами ПК **PhoenixDS Monitoring**. По этим данным рассчитывается прогноз поведения параметров, а именно: общую тенденцию изменения значений параметров, сезонные колебания (часовые, суточные, месячные, годовые и пр.) и в зависимости от этих изменений устанавливаются доверительные интервалы возможных колебаний значений.

ПМ **PhoenixDS Analytics** содержит инструменты и готовые решения по проблемам прогнозного и математического моделирования поведения сложных нелинейных систем. Он решает математические задачи прогнозирования временных рядов.

### 1.1.1. Архитектура ПМ PhoenixDS Analytics

Компоненты ПМ **PhoenixDS Analytics**:

- ❖ web-приложение **Редактор математических моделей** – позволяет создавать расчётные сценарии для решения задач с использованием встроенных математических инструментов, настраивать на основе сценариев задачи прогнозирования поведения параметров, просматривать результаты выполнения задач в графической форме;
- ❖ математическая библиотека (включая системный сервис `phoenixds-analytics.service`), реализующая математический аппарат для решения задач. Рекомендуется устанавливать её на отдельный сервер (отличный от сервера PhoenixDS). Сервер с установленной математической библиотекой будем называть сервером MADA;
- ❖ системные процессы, запускаемые на сервере PhoenixDS и обеспечивающие взаимодействие ПК **PhoenixDS Monitoring** с сервером MADA:

[mada](#)<sup>[56]</sup> (клиент MADA) – необходим для прогнозирования временных рядов.

Системный сервис `phoenixds-analytics.service` взаимодействует с системными процессами `mada` и `auto-thresholds` посредством шины ESB. Для оптимального функционирования ПМ **PhoenixDS Analytics** может быть запущено несколько системных сервисов и системных процессов, подключённых к одной шине ESB. В этом случае нагрузка между ними распределяется по алгоритму Round-robin.



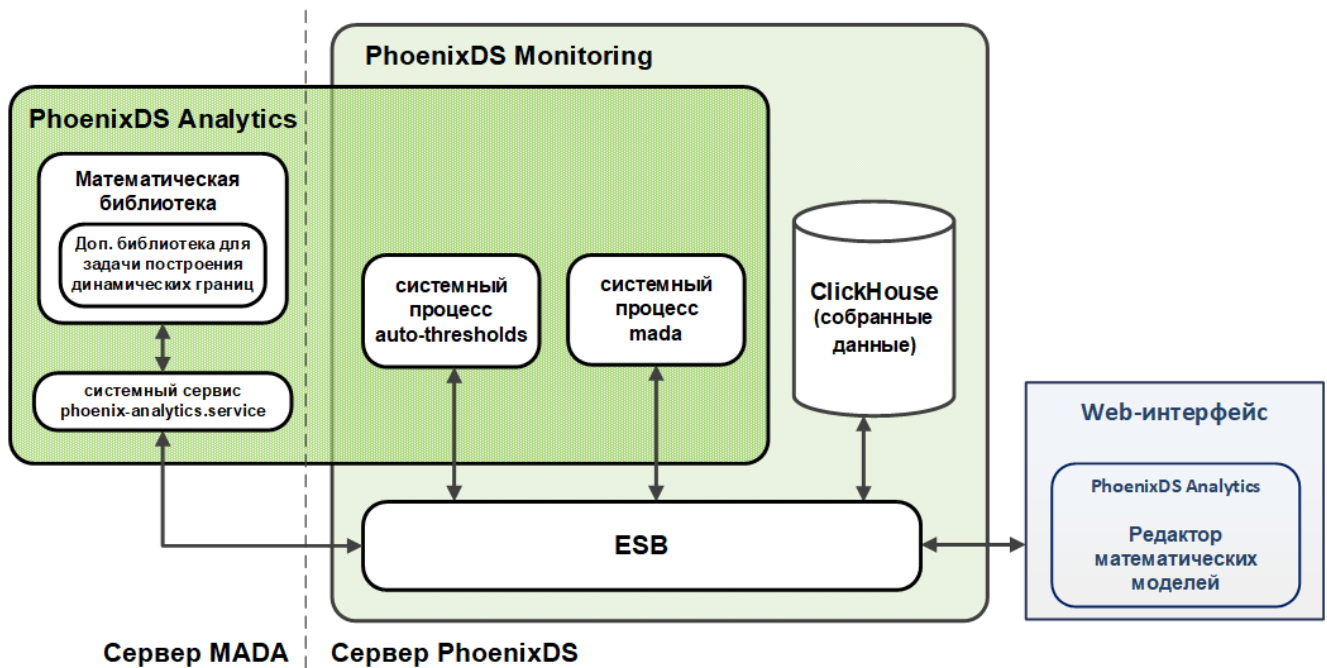


Рисунок 1. Структура ПМ PhoenixDS Analytics

### 1.1.2. Технические требования

Конфигурация серверного оборудования, необходимого для функционирования ПМ **PhoenixDS Analytics**, зависит от целей проектного решения, его планируемой мощности и т.п. Поэтому технические требования к используемым серверам согласовываются с Заказчиком.

Клиент MADA следует устанавливать на сервер PhoenixDS, т.е. на сервер с установленным ПК **PhoenixDS Monitoring** версии 8.0. Для хранения данных, собранных с объектов контроля, должна использоваться база данных производительности, реализованная средствами СУБД ClickHouse.

Сервер MADA должен отвечать следующим минимальным требованиям (для обработки массива входных данных, состоящего из 5100 точек):

- ❖ операционная система – CentOS 7.6;
- ❖ процессор – 8 core 2.6 ГГц;
- ❖ оперативная память – 16 Гб и выше;
- ❖ жесткий диск – 200 Гб;
- ❖ сетевые интерфейсы – 2\*FastEthernet 1000 Мбит/сек;
- ❖ поддержка AVX-инструкций (для реализации [нейросетевых математических моделей](#) <sup>[57]</sup>).



---

# 2.

## Понятия и концепции

## 2.1. Математические инструменты ПМ PhoenixDS Analytics

ПМ **PhoenixDS Analytics** с помощью реализованных математических моделей, алгоритмов и методов позволяет решать задачи прогнозирования временных рядов на основании:

- ❖ одного исходного временного ряда ([Time Series Prediction](#)<sup>[12]</sup>);
- ❖ множества исходных временных рядов ([Multi Time Series Prediction](#)<sup>[13]</sup>).

Настройку математического аппарата осуществляет администратор посредством web-приложения [Редактор математических моделей](#)<sup>[18]</sup>. Рассмотрим его основные сущности.

**Обработчик** – это встроенный математический инструмент для обработки данных (например, математическая модель, фильтр, метод оценки и др.). Классификация обработчиков и описание их параметров приведены в справочнике [Обработчики](#)<sup>[5]</sup>.

**Сценарий** – это совокупность обработчиков и связей между ними, определяющая алгоритм расчёта выходных значений для решения поставленных задач.

**Шаблон сценария** – предустановленный сценарий, предназначенный для решения типовой прикладной задачи (например: прогнозирование, сглаживание, вычисление корреляции и др.).

**Задача** – это применение сценария для решения прикладной задачи на конкретных входных данных из допустимого источника.

В web-приложении **Редактор математических моделей** администратор, используя обработчики, создаёт сценарии для расчёта предсказания временных рядов и настраивает (тестирует) их применение к конкретным временным рядам, т.е. настраивает выполнение задач.

## 2.2. Задача прогнозирования временного ряда на основании одного временного ряда

Решение задачи прогнозирования временного ряда состоит в определении будущих значений исследуемого параметра при помощи методов машинного обучения на основе исторических данных. Используемые модели позволяют прогнозировать изменения значений параметра (общее поведение параметра, сезонные колебания – часовые, суточные, месячные, годовые и пр.) и, в зависимости от этих изменений, устанавливать доверительные интервалы возможных колебаний значений. На основе полученного предсказания и статистического анализа делаются выводы о временном промежутке, через который с заданной вероятностью наступит событие – значение параметра достигнет заданного порога.

Исходными временными рядами могут быть:

- ❖ значения параметров, собранные с целевых объектов системными процессами ПК **PhoenixDS Monitoring**;
- ❖ значения из базы данных ClickHouse;
- ❖ значения из файла.

Решение задачи складывается из нескольких этапов, за каждый из которых отвечают математические алгоритмы (обработчики), настраиваемые администратором в сценарии в [Редакторе математических моделей](#)<sup>[18]</sup>. Рекомендуемая последовательность этапов:

1. подготовка данных – проверка на валидность и фильтрация входных данных, проверка временного ряда на равномерный шаг (и преобразование к равномерному по необходимости);
2. математическое моделирование поведения исследуемого параметра;

## 2.2. Задача прогнозирования временного ряда на основании одного временного ряда

---

3. оценка качества полученной математической модели;
4. предсказание будущих значений показателя;
5. анализ полученных данных (постобработка) – формирование выводов на основе полученного прогноза.

### Предоставление результатов пользователям Web-интерфейса ПП PhoenixDS

Результаты выполнения задачи могут быть представлены пользователям Web-интерфейса ПП PhoenixDS на странице целевого объекта, входящего в структуру объекта дерева конфигурации, в виде графика предсказания поведения параметра.

Для оценки степени критичности предсказанных значений с точки зрения правильности функционирования объекта контроля используется механизм сигналов, реализованный в ПП PhoenixDS, а именно: сигнал типа **MadaCalc**.

Сигнал типа **MadaCalc** показывает прогнозируемое значение показателя целевого объекта и позволяет определить доверительные интервалы (динамические границы) для будущих значений показателя.

Значение сигнала типа **MadaCalc** вычисляется по стандартному для ПП PhoenixDS алгоритму – с помощью агрегирующей функции, указанной при настройке сигнала. Функция применяется к значениям, полученным в результате выполнения задачи за период времени **Диапазон**. Исключение – сигнал с задачей, у которой в сценарии применяется обработчик **Расчёт порогов**<sup>[93]</sup>; в этом случае алгоритм расчёт значения сигнала зависит от указанного обработчика.

См. также:

[Настройка прогнозирования поведения параметра целевого объекта](#)<sup>[40]</sup>

Справочник [Обработчики](#)<sup>[57]</sup>

## 2.3. Задача прогнозирования временного ряда на основе множества временных рядов

Эта задача состоит в прогнозировании значений одного временного ряда на основе множества других временных рядов и позволяет:

- ❖ провести анализ данных для понимания того, как изменения во входных временных рядах могут повлиять на целевой временной ряд;  
либо
- ❖ определить единичный результат без привязки к последовательности данных на основе значений в один момент времени ("исторические данные" являются лишь обучающей выборкой).

Такая задача также называется задачей "что, если" (что будет с итоговым значением, если изменить входные значения).

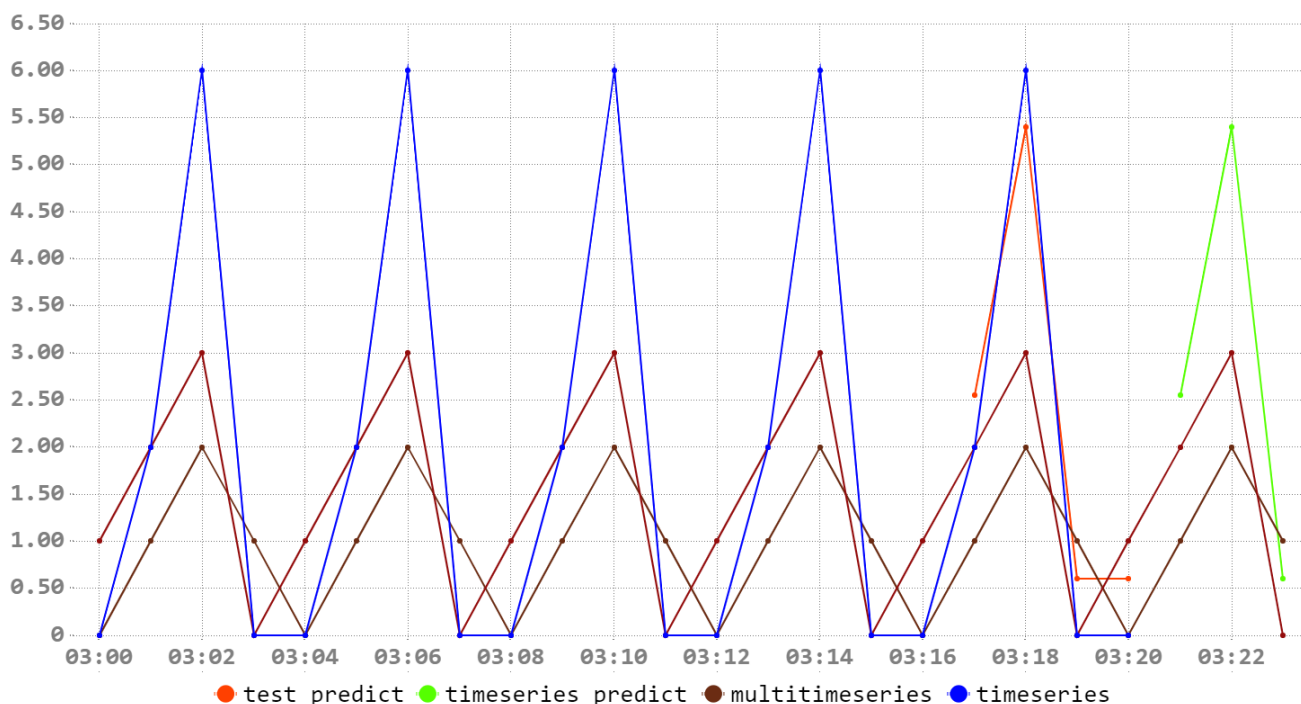


Рисунок 2. Пример: исходные временные ряды (бордового и коричневого цветов) продолжены обучающим временным рядом (синего цвета), и в результате построен график предсказания (зелёного цвета)

Решение задачи прогнозирования временного ряда на основе множества временных рядов состоит из тех же этапов, что и решение обычной задачи прогнозирования временного ряда. За каждый этап отвечают математические алгоритмы (обработчики), настраиваемые администратором в сценарии в [Редакторе математических моделей](#)<sup>18</sup>. Рекомендуемая последовательность этапов:

1. подготовка данных – проверка на валидность и фильтрация входных данных, проверка временного ряда на равномерный шаг (и преобразование к равномерному по необходимости);
2. математическое моделирование поведения исследуемого параметра – создание математической модели и её обучение на входных данных. При настройке сценария администратор выбирает количество данных (в процентах), на которых будет происходить обучение. Остальные данные будут использоваться для оценки качества модели. Временные ряды, у которых нет соответствующих значений целевой функции, используются для предсказания. Таким образом, данные делятся на 3 части: обучающие данные, тестовые (для метрик) и производственные, по которым происходит основное предсказание;
3. оценка качества полученной математической модели;
4. предсказание будущих значений показателя;
5. анализ полученных данных (постобработка) – формирование выводов на основе полученного прогноза.

См. также:

Справочник [Обработчики](#)<sup>57</sup>

---

# 3.

## Установка и удаление ПМ PhoenixDS Analytics

### 3. Установка и удаление ПМ PhoenixDS Analytics

Состав необходимых компонент ПМ **PhoenixDS Analytics** зависит от математических задач, применяемых в проектном решении.

Приведённые в настоящем разделе команды по установке и удалению компонент следует выполнять от имени администратора `root`, если не указано особо.

Перед установкой следует убедиться, что сервера удовлетворяют [техническим требованиям](#)<sup>[9]</sup>.

## 3.1. Установка компонент на сервер MADA

Математическую библиотеку, реализующую методы и алгоритмы для решения задач, рекомендуется устанавливать на отдельный сервер – сервер MADA (отличный от сервера PhoenixDS).

Команда установки основной библиотеки из инсталляционного файла на сервер MADA:

```
rpm -ivh phoenixds-analytics-8-0.<номер сборки>.el7.x86_64.rpm
```

Например:

```
rpm -ivh phoenixds-analytics-8-0.202310250835.5217775.el7.x86_64.rpm
```

Для решения [задачи прогнозирования временных рядов](#)<sup>[12]</sup> дополнительные библиотеки не требуются.

## 3.2. Установка и настройка компонент на сервере PhoenixDS

Состав компонент, устанавливаемых на сервер PhoenixDS, и настройки ПК **PhoenixDS Monitoring** зависят от математических задач, применяемых в проектном решении.

## 3.3. Установка Web-приложения Редактор математических моделей

Команды установки Web-приложения **Редактор математических моделей** из инсталляционных файлов:

```
rpm -ivh phoenixds-analytics-manager-core-8-0.<номер сборки>.el7.x86_64.rpm
```

```
rpm -ivh phoenixds-analytics-manager-embedded-8-0.<номер сборки>.el7.x86_64.rpm
```

Например:

```
rpm -ivh phoenixds-analytics-manager-core-8-0.202310300835.5217775.el7.x86_64.rpm
```

```
rpm -ivh phoenixds-analytics-manager-embedded-8-0.202310300928.67174375.el7.x86_64.rpm
```

## 3.4. Удаление компонент ПМ PhoenixDS Analytics

Последовательность команд для удаления компонент ПМ **PhoenixDS Analytics** с сервера MADA:

1. удалить компоненты Web-приложения **Редактор математических моделей**:

```
rpm -e phoenixds-analytics-manager-core-8-0.<номер сборки>.el7.x86_64
```

```
rpm -e phoenixds-analytics-manager-embedded-8-0.<номер сборки>.el7.x86_64
```

3. удалить основную библиотеку:

```
rpm -e phoenixds-analytics-8-0.<номер сборки>.el7.x86_64
```



---

# 4.

## Web-приложение Редактор математических моделей


## 4. Web-приложение Редактор математических моделей

Web-приложение **Редактор математических моделей** является компонентой ПМ **PhoenixDS Analytics** и позволяет с помощью встроенных математических инструментов создавать расчётные сценарии, настраивать на их основе выполнение задач прогнозирования поведения параметров, просматривать результаты в наглядной графической форме.

Приложение доступно, если администратор входит в группу пользователей с глобальной политикой **Запуск Редактора математических моделей**. См. документ "ПП PhoenixDS. Руководство администратора".

### 4.1. Вход и выход из приложения

#### Вход в приложение

ПМ **PhoenixDS Analytics**, согласно проектному решению, может быть интегрирован с другими компонентами ПП PhoenixDS. Войти в приложение **Редактор математических моделей** можно через Web-интерфейс, предоставляемый ПК **PhoenixDS Web**. Для этого в заголовке страницы Web-интерфейса ПП PhoenixDS следует нажать кнопку  и в меню выбрать иконку **Редактор математических моделей**.

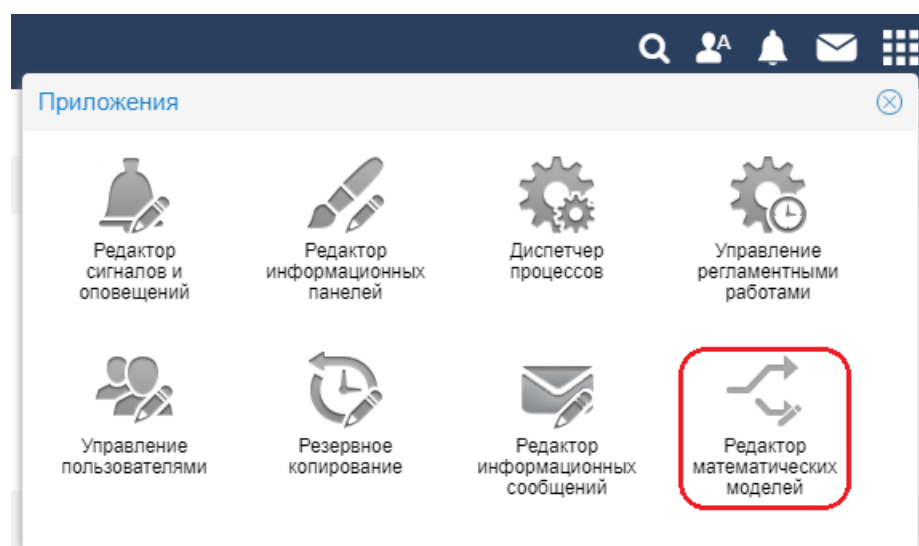



Рисунок 3. Вход в приложение Редактор математических моделей через Web-интерфейс ПП PhoenixDS

#### Выход из приложения

Чтобы выйти из приложения **Редактор математических моделей**, следует в заголовке страницы нажать кнопку  и в появившемся меню выбрать пункт **Выход**. В результате соединение с сервером будет завершено, и откроется страница с приглашением в Web-интерфейс.

## 4.2. Основное окно

Основное окно приложения **Редактор математических моделей** включает в себя:

- ❖ [заголовок](#) <sup>[19]</sup>;
- ❖ вертикальную панель с кнопками для выбора представления:



– открывает представление [Сценарии](#) <sup>[20]</sup>;



– открывает представление [Список задач](#) <sup>[25]</sup>;

- ❖ рабочую область – ее структура и содержание зависит от выбранного представления.

PhoenixDS Analytics		Запущена	Закончена	Состояние	Результат
		21.08.23 11:20:41	21.08.23 11:20:43	Завершена	trend_ens
>	test_task_memory	21.08.23 11:20:31	21.08.23 11:20:33	Завершена	trend_ens
>	test_task_cpu	21.08.23 11:20:21	21.08.23 11:20:23	Завершена	trend_ens
>	test_task_ens	21.08.23 11:20:11	21.08.23 11:20:13	Завершена	trend_ens
		21.08.23 11:20:01	21.08.23 11:20:03	Завершена	trend_ens
		21.08.23 11:19:51	21.08.23 11:19:53	Завершена	trend_ens
		21.08.23 11:19:41	21.08.23 11:19:43	Завершена	trend_ens
		21.08.23 11:19:31	21.08.23 11:19:33	Завершена	trend_ens
		21.08.23 11:19:21	21.08.23 11:19:23	Завершена	trend_ens
		21.08.23 11:19:11	21.08.23 11:19:13	Завершена	trend_ens
		21.08.23 11:19:01	21.08.23 11:19:03	Завершена	trend_ens
		21.08.23 11:18:31	21.08.23 11:18:33	Завершена	trend_ens

Рисунок 4. Основное окно приложения Редактор математических моделей. Представление Список задач

### 4.2.1. Заголовок и главное меню

Заголовок основного окна приложения содержит, как правило, в левой части логотип и название ПМ **PhoenixDS Analytics**. Логотип и наименование могут быть изменены в соответствии с проектным решением.

В правой части заголовка расположены иконки:



– меню, содержащее имя пользователя и команду для выхода из Web-интерфейса;



– главное меню.

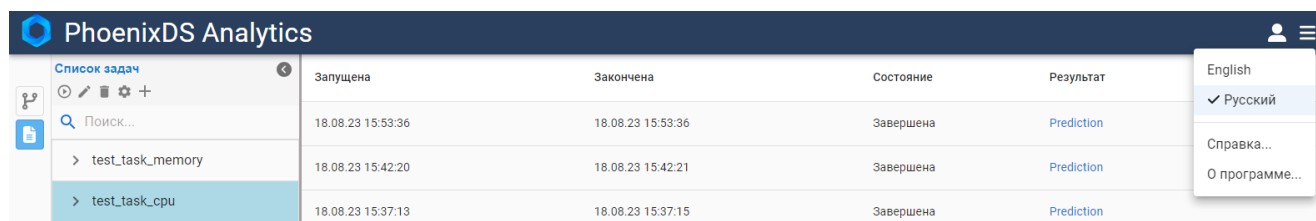


Рисунок 5. Главное меню конечного пользователя

Конечному пользователю доступны следующие пункты главного меню:

- ❖ **English / Русский** – переключение языка Web-интерфейса (выбор локализации);
- ❖ **Справка...** – вызов справки по Web-интерфейсу;
- ❖ **О программе...** – вызов окна с информацией о версии установленных компонент ПП PhoenixDS и

сроке действия лицензии. Кнопка **КОПИРОВАТЬ** в нижнем левом углу окна позволяет скопировать в буфер обмена выведенную информацию и в дальнейшем использовать ее для передачи в службу поддержки ООО «Динамические Системы».

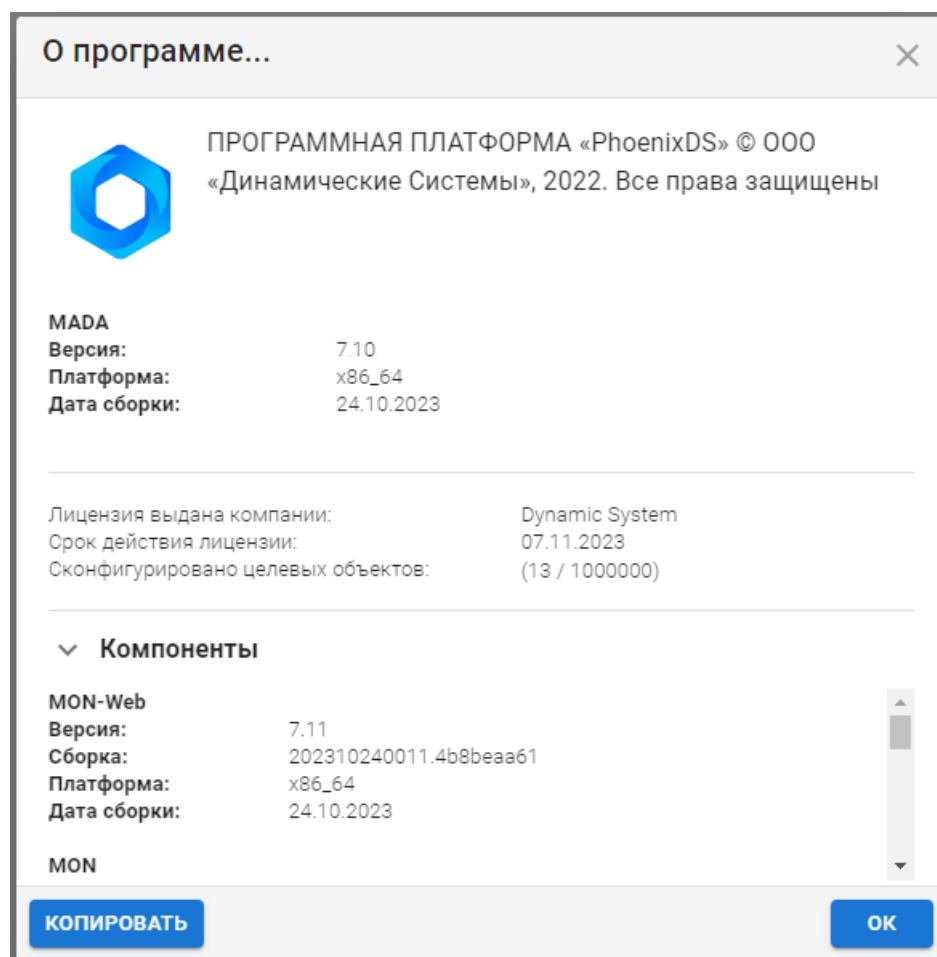


Рисунок 6. Информация об установленных компонентах ПП PhoenixDS

## 4.2.2. Представление Сценарии

Представление **Сценарии** предназначено для создания алгоритмов – шаблонов расчёта выходных значений для поставленных задач. Представление включает в себя:

- ❖ окно **Сценарии**;
- ❖ [область редактирования](#)<sup>23</sup>;

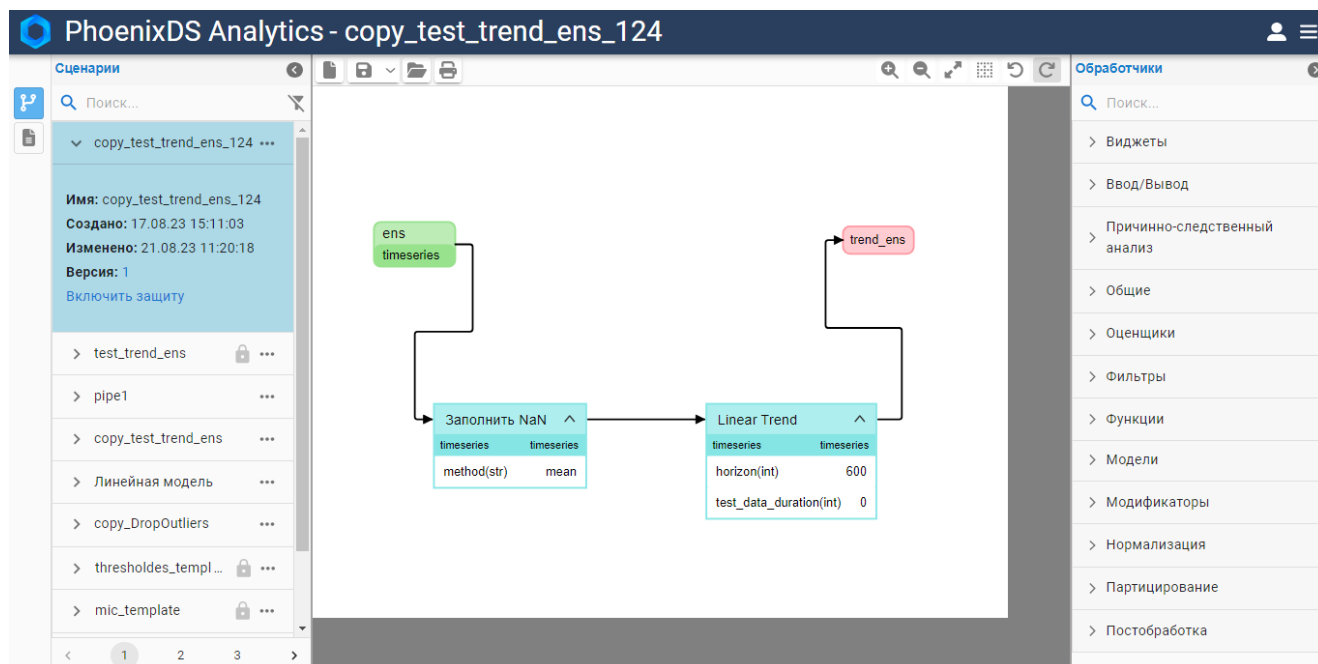

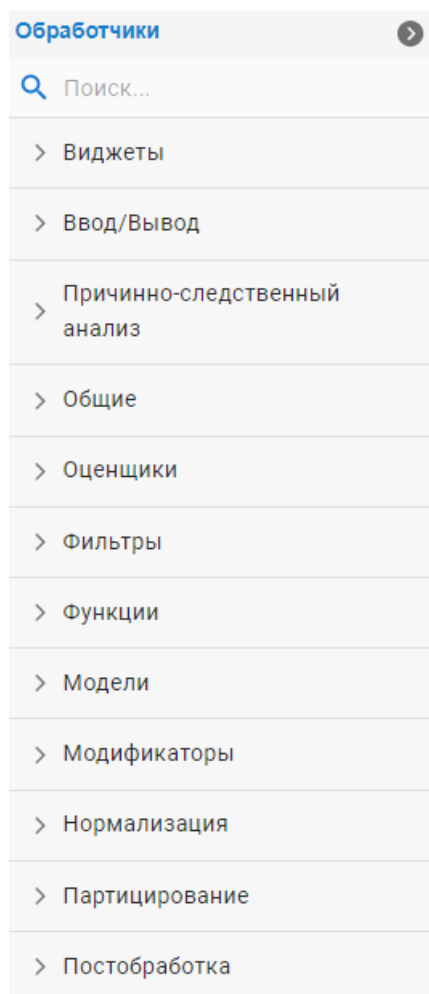
❖ [окно](#) <sup>21</sup> [Обработчики](#) <sup>21</sup>.

Рисунок 7. Представление Сценарии

### 4.2.2.1. Окно Обработчики

Окно **Обработчики** расположено в правой части основного окна и имеет следующую структуру:

- ❖ наименование окна и кнопка , позволяющая свернуть окно;
- ❖ поле для поиска по введенному текстовому фрагменту;
- ❖ список обработчиков.



**Рисунок 8. Окно Обработчики**

Каждая строка в списке представляет собой отдельную группу обработчиков. При нажатии на определенную строку, соответствующую группе обработчиков, раскрывается полный список отдельных обработчиков, входящих в выбранную группу.

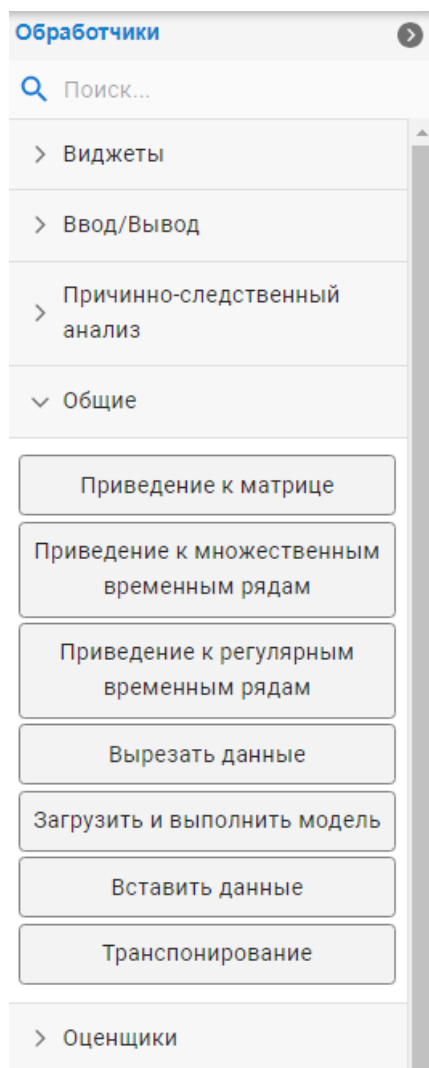


Рисунок 9. Группа обработчиков

Для построения сценария необходимо перетащить в область редактирования нужные обработчики, настроить их параметры и установить связи между ними. Подробно про обработчики см. в справочнике [Обработчики](#) <sup>57</sup>.

#### 4.2.2.2. Область редактирования

Область редактирования (канва) – это рабочая зона, в которой пользователь размещает и настраивает обработчики. Область расположена в центре основного окна приложения.

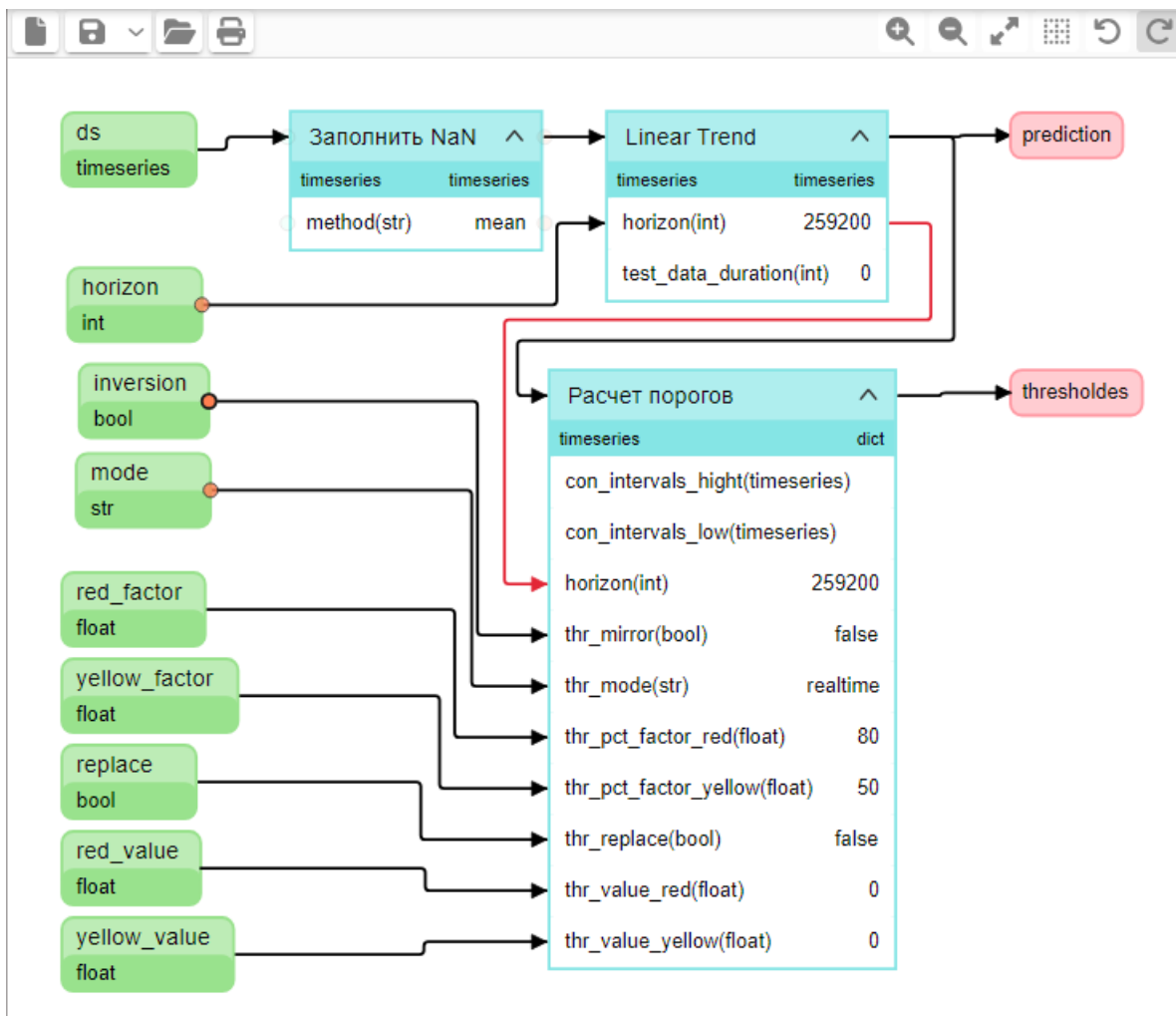


Рисунок 10. Область редактирования (канва)

Панель инструментов, расположенная в верхней части канвы, содержит следующие кнопки:



– кнопка для создания нового сценария;



| Ctrl + S

– группа кнопок для сохранения сценария: **Сохранить** (на сервер) | **Сохранить как** (**Сохранить на сервер**, **Сохранить на устройство**);



| Ctrl + O

– кнопка для импорта сценария;



| Ctrl + P

– кнопка для печати сценария;



| Ctrl + +

– кнопка для увеличения масштаба канвы;



| Ctrl + -

– кнопка для уменьшения масштаба канвы;



– кнопка для изменения размера канвы;





– кнопка для отображения сетки на канве.



| Ctrl + Z

– кнопка для отмены последнего изменения;



| Ctrl + Y

– кнопка для повтора отмененного изменения.

### 4.2.3. Представление Список задач

Представление **Список задач** позволяет настроить задачу расчёта выходных значений по созданному сценарию.

Представление включает в себя:


- ❖ [окно](#) <sup>25</sup> **Список задач** <sup>25</sup>;
- ❖ [таблицу выполнения задачи](#) <sup>27</sup>.

Запущена	Закончена	Состояние	Результат
21.08.23 11:20:41	21.08.23 11:20:43	Завершена	trend_ens
21.08.23 11:20:31	21.08.23 11:20:33	Завершена	trend_ens
21.08.23 11:20:21	21.08.23 11:20:23	Завершена	trend_ens
21.08.23 11:20:11	21.08.23 11:20:13	Завершена	trend_ens
21.08.23 11:20:01	21.08.23 11:20:03	Завершена	trend_ens
21.08.23 11:19:51	21.08.23 11:19:53	Завершена	trend_ens
21.08.23 11:19:41	21.08.23 11:19:43	Завершена	trend_ens
21.08.23 11:19:31	21.08.23 11:19:33	Завершена	trend_ens
21.08.23 11:19:21	21.08.23 11:19:23	Завершена	trend_ens
21.08.23 11:19:11	21.08.23 11:19:13	Завершена	trend_ens
21.08.23 11:19:01	21.08.23 11:19:03	Завершена	trend_ens
21.08.23 11:18:31	21.08.23 11:18:33	Завершена	trend_ens

Рисунок 11. Представление Список задач

#### 4.2.3.1. Окно Список задач

Окно **Список задач** расположено в левой части основного окна и имеет следующую структуру:

- ❖ наименование окна и кнопка , позволяющая свернуть окно;
- ❖ панель инструментов для управления задачами, содержащая следующие кнопки:






– запуск задачи;



– изменение задачи;

## 4. Web-приложение Редактор математических моделей

-  – удаление задачи;
-  – автозапуск задачи;
-  – создание новой задачи;

- ❖ поле для поиска по введенному текстовому фрагменту;
- ❖ список созданных задач;
- ❖ панель навигации, содержащая кнопки для перемещения между страницами списка.

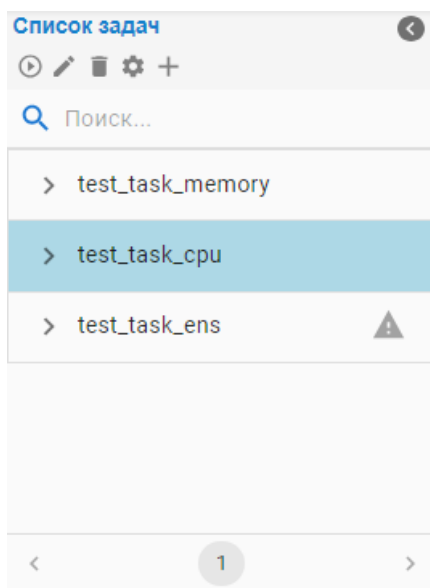



Рисунок 12. Окно Список задач

При нажатии в строке списка задач кнопки  отображается основная информация о задаче:

- ❖ **Имя** – название задачи;
- ❖ **Изменено** – дата и время последнего изменения задачи;
- ❖ **Сценарий** – название прикрепленного сценария и его версия;
- ❖ **Описание** – краткое описание задачи, комментарий.

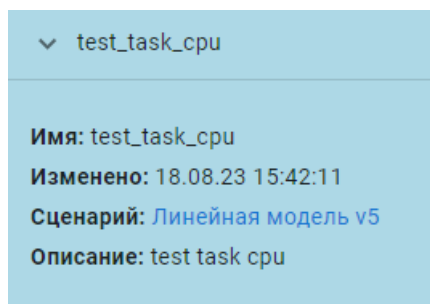



Рисунок 13. Пример отображения основной информации о задаче

Если строка списка задач содержит иконку , это означает, что соответствующая задача имеет непоследнюю версию сценария, и необходимо обновить задачу до последней.

### 4.2.3.2. Таблица выполнения задачи

В центре основного окна расположена таблица выполнения задачи, выбранной в окне **Список задач**.

Таблица содержит записи с информацией о запуске, окончании, состоянии задачи и результате ее выполнения.

Запущена	Закончена	Состояние	Результат	
18.08.23 15:53:36	18.08.23 15:53:36	Завершена	Prediction	
18.08.23 15:42:20	18.08.23 15:42:21	Завершена	Prediction	
18.08.23 15:37:13	18.08.23 15:37:15	Завершена	Prediction	
18.08.23 15:35:32	18.08.23 15:35:33	Завершена	Prediction	

Рисунок 14. Таблица выполнения задачи

По нажатию в правой части заголовка таблицы кнопки  происходит обновление результатов расчета задачи.

См. также:


[Просмотр результата выполнения задачи](#) 

## 4.3. Настройка сценариев

Все действия по настройке сценариев выполняются в представлении **Сценарии**.

### 4.3.1. Создание сценария

Чтобы создать сценарий, следует:

1. на панели инструментов области редактирования нажать кнопку ;
2. в появившемся окне **Создание нового сценария** ввести имя сценария и нажать кнопку **СОЗДАТЬ**;

Создание нового сценария

Введите имя сценария

СОЗДАТЬ    ЗАКРЫТЬ

Рисунок 15. Окно создания нового сценария

в результате пустой сценарий с заданным именем будет создан и займет первую строку в списке в окне **Сценарии**;

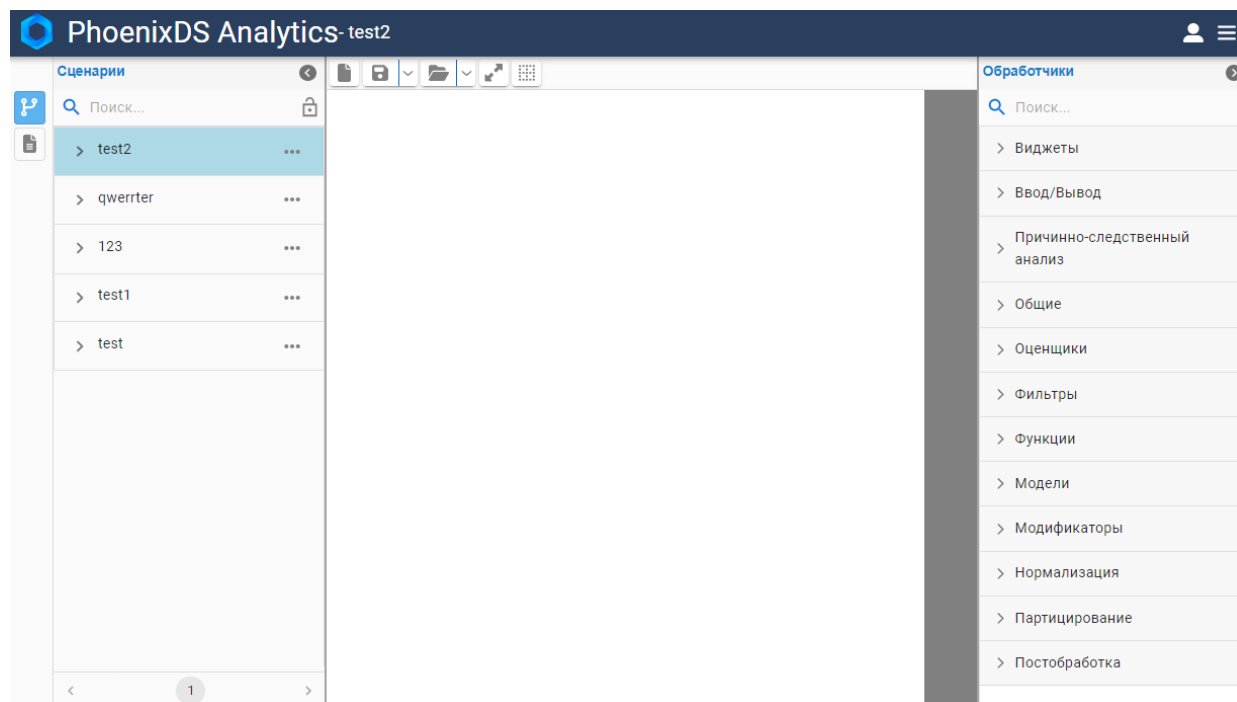



Рисунок 16. Созданный сценарий пуст

**Примечание.** При необходимости имя сценария можно изменить; для этого следует нажать кнопку  и в меню выбрать пункт **Изменить сценарий**, после чего в появившемся окне ввести новое имя и нажать кнопку **ОБНОВИТЬ**.

- перетащить из окна **Обработчики** на канву необходимые обработчики; настроить их и связи между ними (см. справочник [Обработчики](#)<sup>[57]</sup>);
- сохранить сценарий (см. п. [Сохранение сценария](#)<sup>[28]</sup>).

### 4.3.2. Сохранение сценария

Сохранить сценарий можно двумя способами: на сервер и на устройство. Для этого используется группа

кнопок  (Сохранить | Сохранить как).

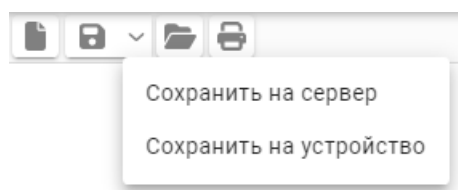


Рисунок 17. Меню выбора способа сохранения сценария

#### Сохранение на сервер

Чтобы сохранить сценарий на сервер, следует на панели инструментов области редактирования нажать кнопку **Сохранить** или кнопку **Сохранить как** и выбрать в выпадающем меню пункт **Сохранить на сервер**. Также можно использовать горячие клавиши **Ctrl + S**. В результате в правом нижнем углу основного окна web-приложения появится уведомление об успешном сохранении сценария на сервер.

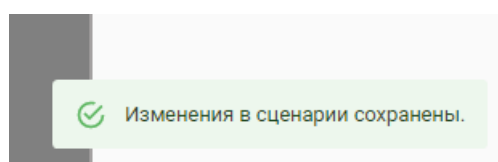


Рисунок 18. Оповещение об успешном сохранении сценария на сервер

### Сохранение на устройство


Чтобы сохранить сценарий на устройство (рабочую станцию), следует на панели инструментов области редактирования нажать кнопку **Сохранить как** и в выпадающем меню выбрать пункт **Сохранить на устройство**. В результате сценарий в форматах `pipeline.json` и `view.json` сохранится в директорию рабочей станции в соответствии с настройками браузера.



Рисунок 19. Сценарий сохранен на устройство

### 4.3.3. Импорт сценария

Чтобы импортировать сценарий в web-приложение, следует:


1. на панели инструментов области редактирования нажать кнопку  или воспользоваться горячими клавишами **Ctrl + O**;
2. в открывшемся окне проводника выбрать необходимый для загрузки файл с расширением `view.json`.

В результате выбранный файл загрузится в web-приложение.

**ВНИМАНИЕ!** При импорте сценария с локального устройства (рабочей станции), он не доступен для редактирования. Чтобы внести изменения в импортированный сценарий, необходимо сохранить его на сервер.

### 4.3.4. Дублирование сценария

Чтобы продублировать сценарий, следует:

1. в окне **Сценарии** выбрать сценарий, копию которого необходимо создать, и нажать кнопку ;
2. в появившемся меню выбрать пункт **Дублировать сценарий**;

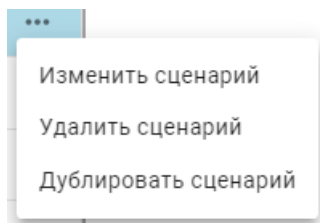


Рисунок 20. Меню доступных действий для сценария

3. в появившемся окне ввести имя сценария (по умолчанию в начале имени добавляется приписка **сору**) и нажать кнопку **СОЗДАТЬ**.

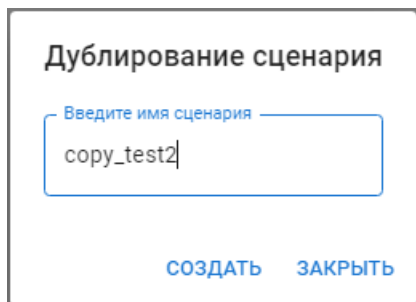


Рисунок 21. Подтверждение дублирования сценария

В результате копия сценария с заданным именем будет создана и займет первую строку в списке в окне **Сценарии**.

### 4.3.5. Настройка канвы

#### Изменение размеров канвы

При создании нового сценария размеры канвы устанавливаются по умолчанию (800 на 800 пикселей). Для изменения размеров канвы следует:



1. на панели инструментов области редактирования нажать кнопку ;
2. в появившемся окне ввести ширину и высоту в пикселях и нажать кнопку **СОХРАНИТЬ**.



Рисунок 22. Окно для ввода размеров канвы


В результате размеры канвы поменяются в соответствии с настройками.

#### Отображение сетки на канве

Для более удобной работы с канвой можно отобразить сетку. Для этого следует на панели инструментов области редактирования нажать кнопку . Повторное нажатие кнопки отключает сетку.

### 4.3.6. Удаление сценария

Чтобы удалить сценарий, следует:

1. в окне **Сценарии** выбрать сценарий, который необходимо удалить, и нажать кнопку ;
2. в появившемся меню выбрать пункт **Удалить сценарий**;

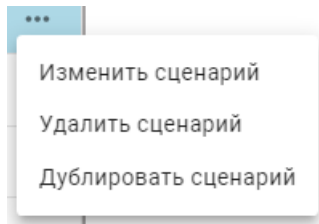


Рисунок 23. Меню доступных действий для сценария

3. в появившемся окне подтвердить удаление, нажав кнопку **УДАЛИТЬ**.

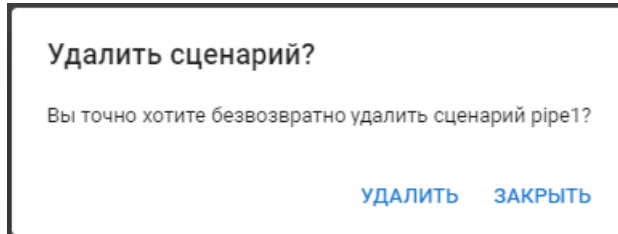


Рисунок 24. Подтверждение удаления сценария

В результате сценарий удалится из web-приложения.

## 4.4. Настройка задач

Все действия по настройке задач выполняются в представлении **Список задач**.

### 4.4.1. Создание задачи

Чтобы создать задачу, следует:

1. на панели инструментов окна **Список задач** нажать кнопку **+**;
2. в появившемся окне **Создание новой задачи** заполнить следующие поля:
  - ❖ **Имя задачи**;
  - ❖ **Сценарий** – созданный в представлении **Сценарии** шаблон расчёта;
  - ❖ **Выбор версии сценария** – версия выбранного ранее сценария;
  - ❖ **Описание** – текстовое описание задачи; поле необязательно для заполнения;
3. нажать кнопку **ДАЛЕЕ**;

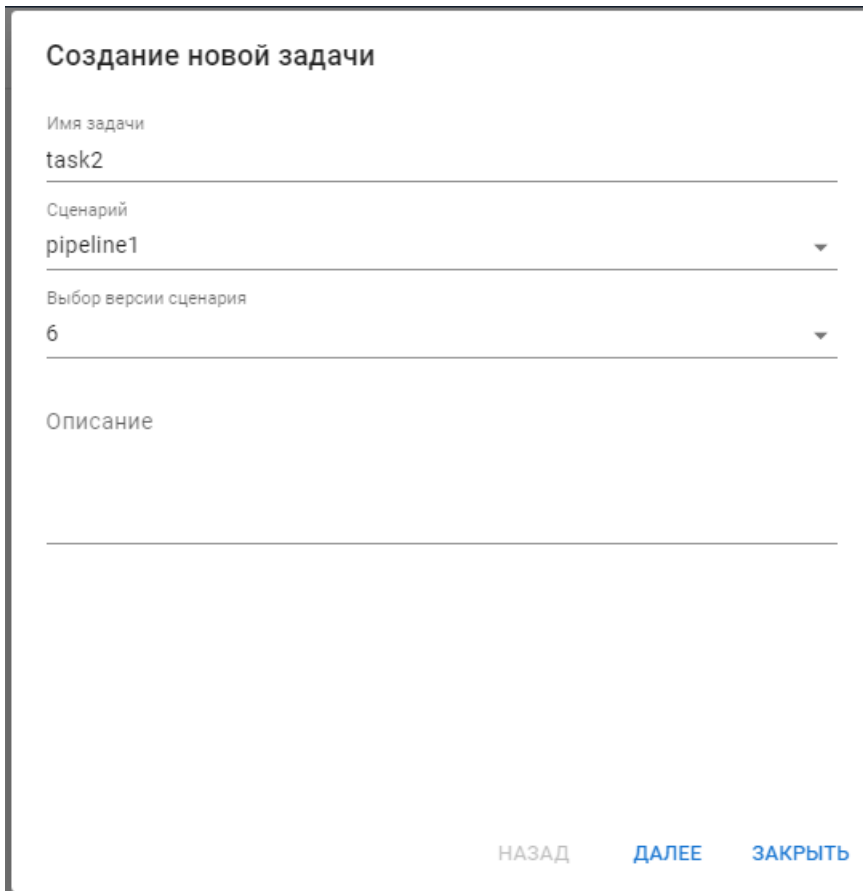


Рисунок 25. Окно создания новой задачи

4. на второй странице окна следует для каждой начальной точки, отображенной в списке слева, заполнить поле **Источник** и другие поля, наличие которых зависит от выбранного типа источника:

❖ **Файл** – исходные данные содержатся в файле; если выбран данный тип источника, необходимо дополнительно заполнить следующие поля:

- Δ **Путь к файлу;**
- Δ **Разделитель (; , / : Пробел);**
- Δ **Номер столбца в файле;**
- Δ **Тип столбца (int, float);**

нажав кнопку **+** можно добавить дополнительную строку с полями **Номер столбца в файле** и **Тип столбца**, нажав кнопку **×** – удалить строку;



**Создание новой задачи**

start point 2 (int)	Источник Файл
start point 1 (timeseries)	Путь к файлу
	Разделитель ;
Номер столбца в файле 0	Тип столбца int
	+

НАЗАД СОЗДАТЬ ЗАКРЫТЬ

Рисунок 26. Окно создания новой задачи. Тип источника Файл

- ❖ **База данных** – исходные данные содержатся в базе данных; если выбран данный тип источника, необходимо дополнительно заполнить следующие поля:

- △ Тип базы данных;
- △ Конфигурация базы данных;
- △ Запрос к базе данных – sql-запрос;

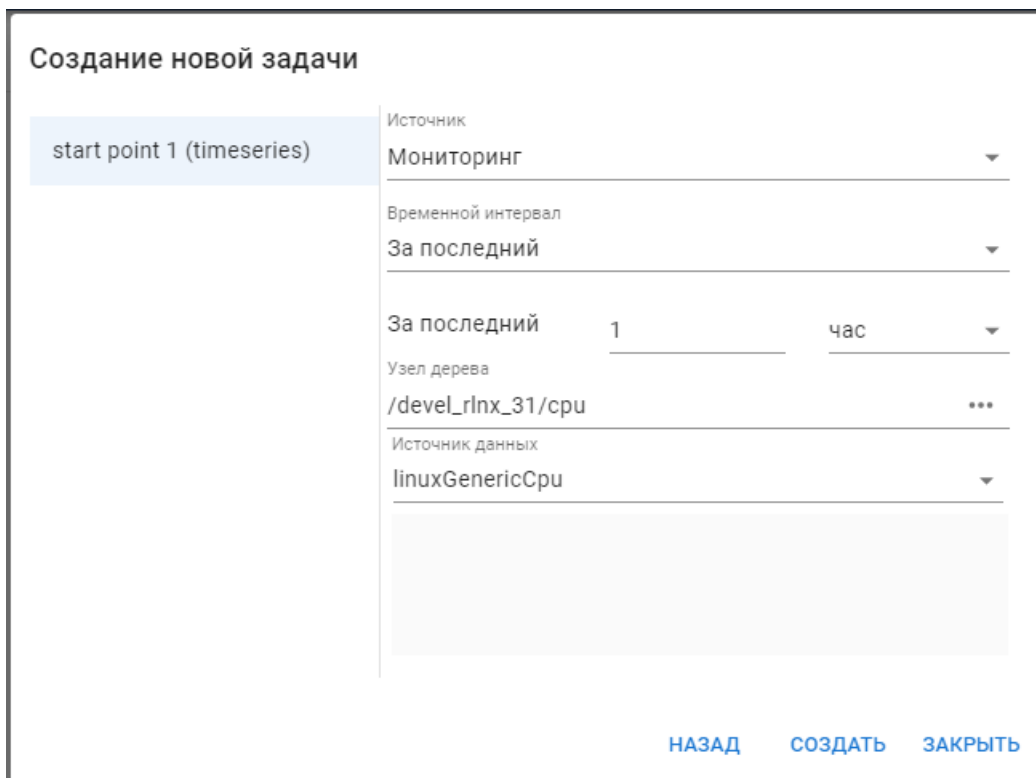
**Создание новой задачи**

start point 1 (timeseries)	Источник База данных
	Тип базы данных clickhouse
	Конфигурация базы данных lancelot
	Запрос к базе данных
	1

НАЗАД СОЗДАТЬ ЗАКРЫТЬ

Рисунок 27. Окно создания новой задачи. Тип источника База данных

- ❖ **Мониторинг** – исходные данные берутся из источника данных целевого объекта в ПК **PhoenixDS Monitoring**; если выбран данный тип источника, необходимо дополнительно заполнить следующие поля:
  - Δ **Временной интервал** – можно настроить временной диапазон ОТ и ДО в формате ДД.ММ.ГГГГ чч:мм, либо выбрать промежуток за последний(ие) час/минуту/секунду;
  - Δ **Узел дерева** – целевой объект, к которому относится исследуемый параметр;
  - Δ **Источник данных** – исследуемый параметр целевого объекта, выбираемый из дерева конфигурации;



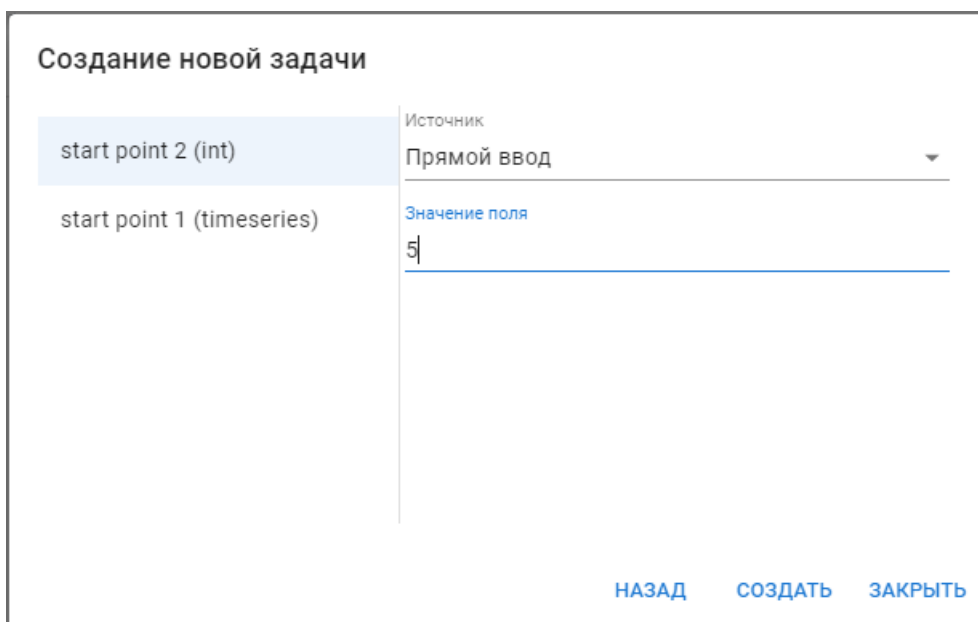
The screenshot shows a web form titled "Создание новой задачи" (Create new task). On the left, there is a list of task types, with "start point 1 (timeseries)" selected. On the right, a form is displayed for the "Мониторинг" (Monitoring) source type. The form includes the following fields:

- Источник** (Source): A dropdown menu with "Мониторинг" selected.
- Временной интервал** (Time interval): A dropdown menu with "За последний" (For the last) selected.
- За последний** (For the last): A text input field containing "1".
- час** (hour): A dropdown menu with "час" (hour) selected.
- Узел дерева** (Tree node): A text input field containing "/devel\_rlnx\_31/cpu".
- Источник данных** (Data source): A dropdown menu with "linuxGenericCpu" selected.

At the bottom right of the form, there are three buttons: "НАЗАД" (Back), "СОЗДАТЬ" (Create), and "ЗАКРЫТЬ" (Close).

Рисунок 28. Окно создания новой задачи. Тип источника Мониторинг

- ❖ **Прямой ввод** – исходные данные заполняются путем прямого ввода значения; если выбран данный тип источника, необходимо заполнить поле **Значение поля**;



The screenshot shows a web form titled "Создание новой задачи" (Create new task). On the left, there is a list of task types, with "start point 2 (int)" selected. On the right, a form is displayed for the "Прямой ввод" (Direct input) source type. The form includes the following fields:

- Источник** (Source): A dropdown menu with "Прямой ввод" selected.
- Значение поля** (Field value): A text input field containing "5".

At the bottom right of the form, there are three buttons: "НАЗАД" (Back), "СОЗДАТЬ" (Create), and "ЗАКРЫТЬ" (Close).


Рисунок 29. Окно создания новой задачи. Тип источника Прямой ввод

5. нажать кнопку **СОЗДАТЬ**.

В результате задача появится в окне **Список задач**.

## 4.4.2. Изменение задачи

Чтобы изменить задачу, следует:

1. в окне **Список задач** выбрать задачу, которую необходимо изменить, и на панели инструментов окна нажать кнопку  ;
2. в появившемся окне **Обновление задачи** изменить необходимые параметры (при этом заменить сценарий на другой невозможно);
3. нажать кнопку **ОБНОВИТЬ**.

Обновление задачи

start point 1 (timeseries)

Источник

Мониторинг

Временной интервал

За последний

За последние 3 часа

Узел дерева

/devel\_rlnx\_31/cpu

Источник данных

ssCpuSystem

НАЗАД

ОБНОВИТЬ


ЗАКРЫТЬ

Рисунок 30. Окно обновления задачи

## 4.4.3. Запуск задачи

Существует два режима запуска задачи: **ручной** и **автозапуск**.


### Ручной режим запуска задачи

Чтобы запустить задачу в ручном режиме, следует в окне **Список задач** выбрать задачу, которую необходимо запустить, и на панели инструментов окна нажать кнопку .

В результате в таблице состояния задачи появится запись о запуске выбранной задачи с информацией о времени запуска и окончании, состоянии выполнения задачи (**завершена** / **ошибка**) и результатом ее выполнения в виде графика, который можно посмотреть, нажав на ссылку в столбце **Результат**.


## Автозапуск задачи

Чтобы настроить автозапуск задачи, следует:

1. в окне **Список задач** выбрать задачу, для которой необходимо настроить автоматический запуск, и на панели инструментов окна нажать кнопку ;
2. в открывшемся окне **Расписание** нажать кнопку в правой части окна под заголовком таблицы; в результате откроется окно для добавления триггера, по которому будет происходить запуск задачи;
3. в окне **Добавление триггера** выбрать тип триггера:
  - ❖ **По расписанию** – триггер, представляющий собой настроенное расписание; при выборе данного типа триггера необходимо заполнить дополнительные поля:
    - Δ **Название** – название создаваемого триггера;
    - Δ **Период** – период, в течение которого будет запускаться задача (каждый **час** / **день** / **неделя** / **месяц** / **год**);
    - Δ группы полей **В/Каждый(ую)** и **Час(ы) / Минута(ы)** – с помощью данных полей можно настроить конкретное время запуска задачи;
  - ❖ **Интервал** – триггер, представляющий собой фиксированный временной интервал; при выборе данного типа триггера необходимо заполнить дополнительные поля:
    - Δ **Название** – название создаваемого триггера;
    - Δ **Значение** – значение интервала в секундах;
4. нажать кнопку **ДОБАВИТЬ**.

## 4.4.4. Удаление задачи

Чтобы удалить задачу, следует:

1. в окне **Список задач** выбрать задачу, которую необходимо удалить, и на панели инструментов окна нажать кнопку ;
2. в появившемся окне подтвердить удаление, нажав кнопку **УДАЛИТЬ**.

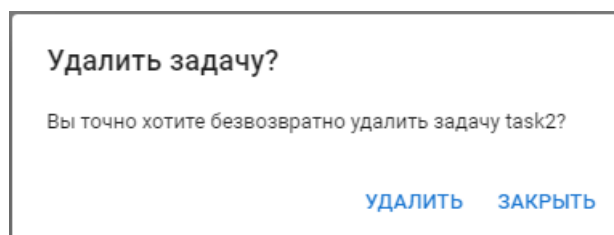


Рисунок 31. Подтверждение на удаление задачи

## 4.5. Просмотр результата выполнения задачи

Чтобы просмотреть результат выполнения задачи, следует в таблице в столбце **Результат** нажать на ссылку в ячейке необходимой строки (количество ссылок в столбце соответствует количеству обработчиков **Вывод** в сценарии). В результате откроется окно с графиком, содержащим входные и выходные значения.

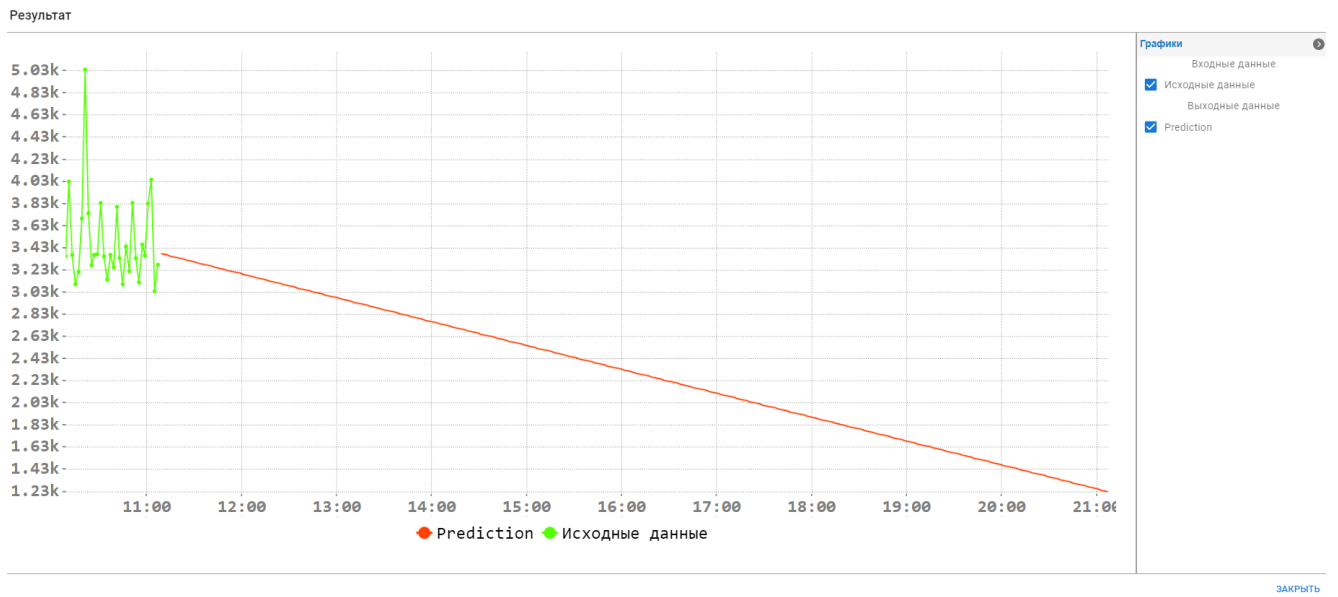


Рисунок 32. Пример графика с результатом выполнения задачи



---

# 5.

## Применение ПМ PhoenixDS Analytics

В web-приложении [Редактор математических моделей](#)<sup>[18]</sup> администратор может настраивать выполнение задач прогнозирования, а также просматривать их результаты.

Задачи прогнозирования применяются для предсказания поведения параметров целевых объектов. Чтобы настроить отображение результатов прогноза на графике исследуемого параметра для предоставления пользователям Web-интерфейса ПП PhoenixDS, следует использовать сигнал типа **MadaCalc**.

### 5.1. Настройка прогнозирования поведения параметра целевого объекта

Для объектов контроля, размещенных в дереве конфигурации, можно настроить предсказание поведения параметров целевых объектов с отображением на графиках.

Рассмотрим общий порядок настройки прогноза для параметра целевого объекта. Процедуру настройки можно разделить на 3 этапа:

1. [настройка сценария и задачи в Редакторе матем. моделей](#)<sup>[40]</sup>;
2. [настройка набора сигналов](#)<sup>[42]</sup>;
3. [привязка набора сигналов к целевому объекту](#)<sup>[50]</sup>.

#### 5.1.1. Настройка сценария и задачи в Редакторе матем. моделей

Детально перечисленные ниже действия описаны в разделе [Web-приложение Редактор математических моделей](#)<sup>[18]</sup>.

Последовательность действий следующая:

1. в представлении **Сценарии** настроить [сценарий](#)<sup>[27]</sup> вычисления прогноза;

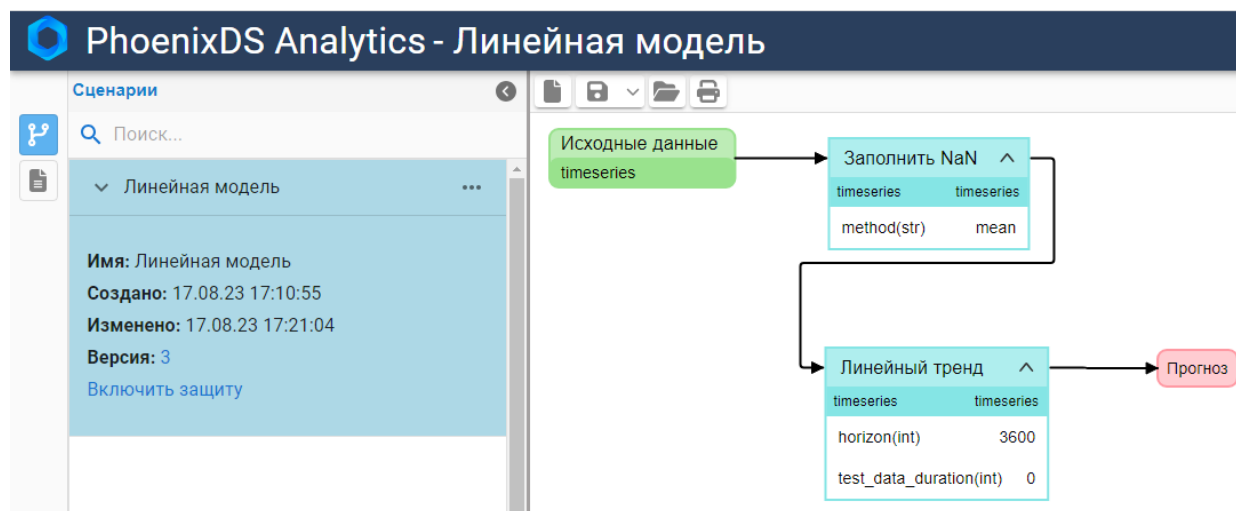


Рисунок 33. Пример сценария вычисления прогноза на основе линейной модели

2. в представлении **Задачи** создать [задачу](#)<sup>[31]</sup> выполнения настроенного сценария. В окне настройки задачи на первой странице задать следующие параметры:
  - ❖ **Имя задачи** – наименование задачи;
  - ❖ **Сценарий** – следует выбрать сценарий, настроенный на шаге 1 и его версию;
  - ❖ **Выбор версии сценария** – версия сценария, которая будет использоваться для расчёта.



### Создание новой задачи

Имя задачи

Тренд для трафика

Сценарий

Линейная модель

Выбор версии сценария

3

Описание

Прогноз входящего трафика

НАЗАД

ДАЛЕЕ

ЗАКРЫТЬ

Рисунок 34. Настройка задачи по сценарию (первая страница)

На второй странице:

- ❖ **Источник** – выбрать вариант **Мониторинг**, т.к. в качестве исходного временного ряда будут использоваться данные, собранные с целевого объекта в ПК **PhoenixDS Monitoring**;
- ❖ **Временной интервал** – период, за который будут отбираться исходные данные при каждом запуске задачи. Следует выбрать вариант **За последний** и указать значение периода, например: **За последние 2 часа**;
- ❖ **Узел дерева** – целевой объект, к которому относится исследуемый параметр;
- ❖ **Источник данных** – исследуемый параметр целевого объекта следует выбрать из дерева конфигурации;

Создание новой задачи

Исходные данные (timeseries)

Источник

Мониторинг

Временной интервал

За последний

За последние 2 часа

Узел дерева

/devel\_rlnx\_mada\_2/ens192

Источник данных

ifHCInOctets

НАЗАД

СОЗДАТЬ

ЗАКРЫТЬ

Рисунок 35. Настройка задачи по сценарию (вторая страница)

Примечание. Создать и отредактировать задачу можно также при настройке сигнала типа **MadaCalc**<sup>[46]</sup> (см. поле **Задача из MADA-Manager**).

- для настроенной задачи создать расписание **автозапуска**<sup>[35]</sup>, в соответствии с которым будет выполняться расчет прогноза.

## 5.1.2. Настройка набора сигналов в Редакторе сигналов и оповещений

В **Редакторе сигналов и оповещений** для исследуемого типа целевого объекта создать набор сигналов, включающий сигнал типа **MadaCalc** со ссылкой на настроенную задачу.

Чтобы создать новый набор сигналов, следует:

- открыть web-приложение **Редактор сигналов и оповещений** через меню приложений;

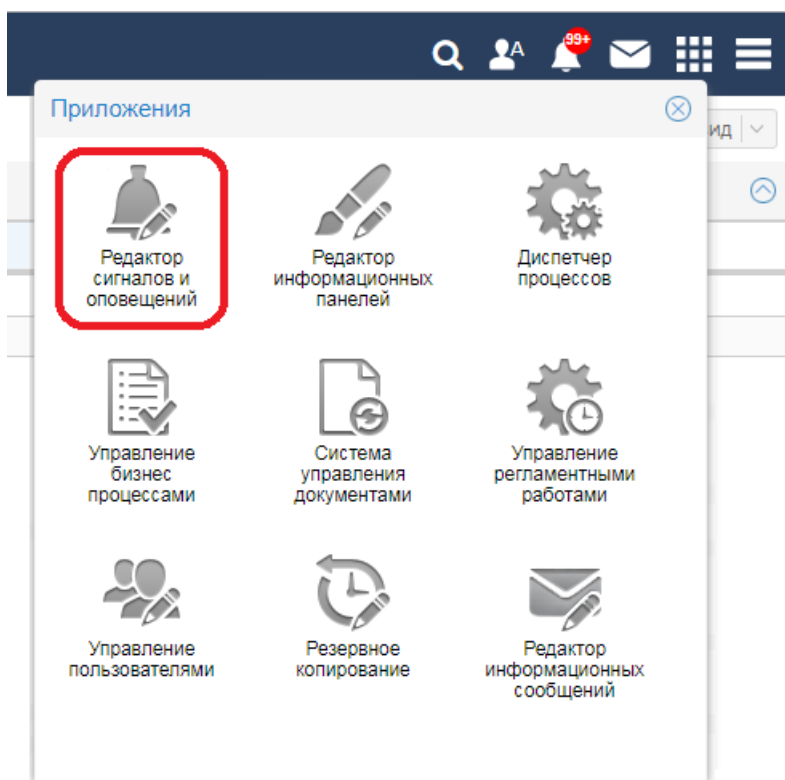




Рисунок 36. Кнопка для входа в web-приложение Редактор шаблонов сигналов

2. в окне **Шаблоны** открыть вкладку **Сигналы** и в контекстном меню выбрать пункт **Новый шаблон**, либо нажать клавишу **Ins** или кнопку  на панели инструментов;
3. в появившемся окне заполнить поля:
  - ❖ **Имя** – ввести имя нового набора сигналов;
  - ❖ **Выберите** – указать тип целевого объекта, для которого настраивается прогноз поведения параметра. Для этого нажать кнопку  и в дереве конфигурации выбрать тот же целевой объект, что и при настройке задачи в поле [Узел дерева](#)<sup>41</sup>.

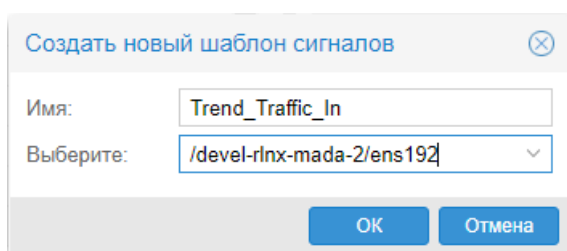


Рисунок 37.

По нажатию кнопки **ОК** созданный набор сигналов будет добавлен в список вкладки **Сигналы**;

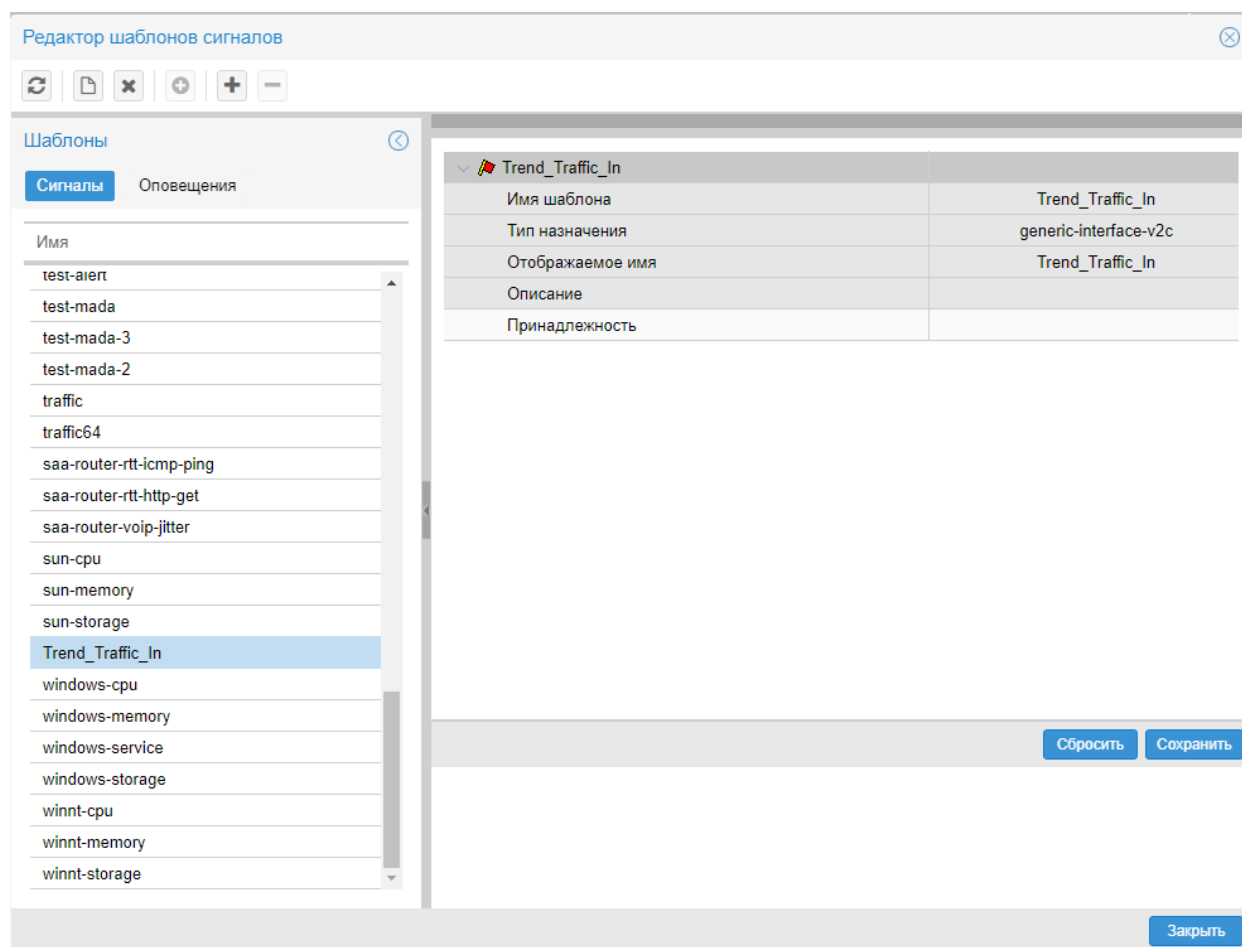


Рисунок 38. Создан набор сигналов

- чтобы в созданный набор добавить сигнал типа **MadaCalc**, в окне **Сигналы** поместить курсор на созданный пустой набор сигналов и в контекстном меню выбрать пункт **Добавить подсигнал** и далее – подпункт **MadaCalc**;

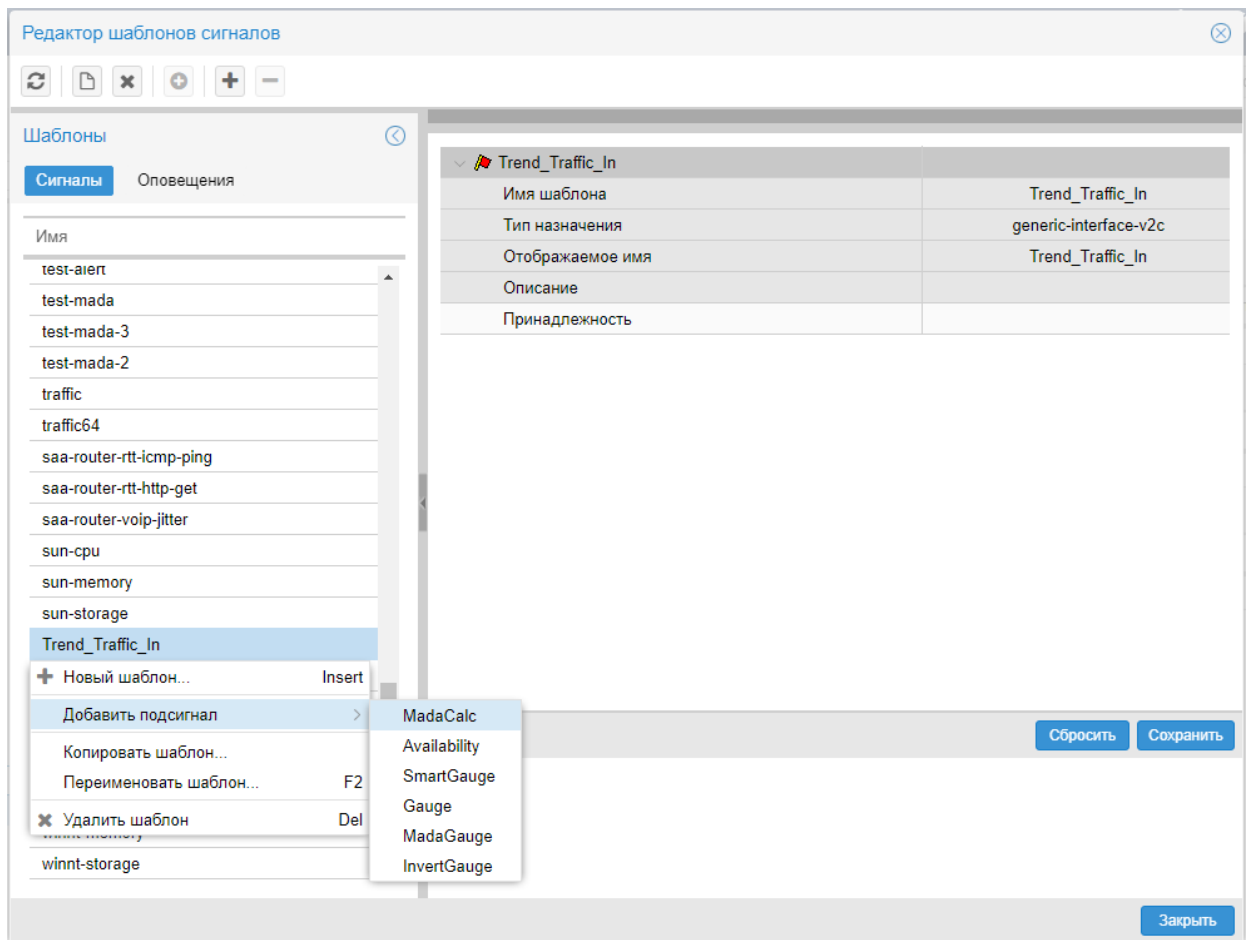


Рисунок 39. Меню для добавления сигнала типа MadaCalc в набор

7. в появившемся окне ввести:
  - ❖ **Имя подсигнала** – ввести имя сигнала;
  - ❖ **Источник данных** – исследуемый параметр целевого объекта; выбрать тот же, что указан при настройке задачи в поле [Источник данных](#) <sup>[41]</sup>;

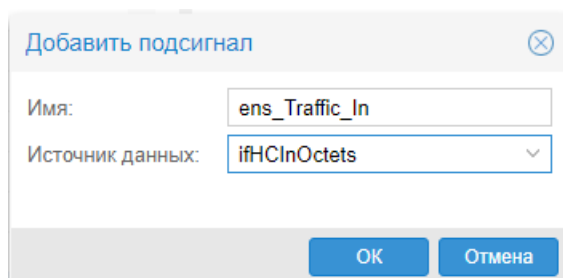


Рисунок 40. Ввод имени сигнала

По нажатию кнопки **ОК** в рабочую область справа будет добавлен блок с параметрами сигнала;

9. в рабочей области ввести глобальные значения [параметров сигнала типа MadaCalc](#) <sup>[46]</sup>;
10. после заполнения параметров нажать кнопку **Сохранить**.

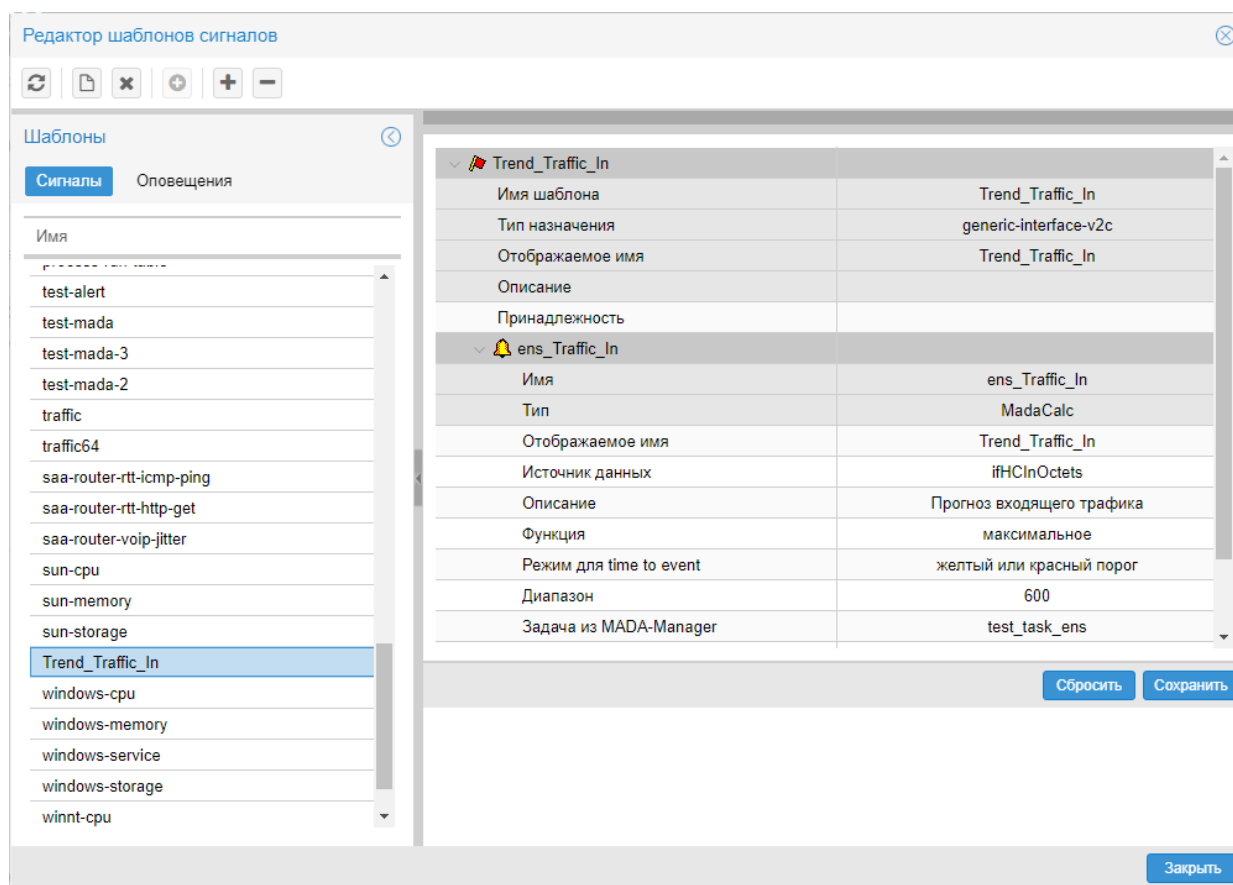


Рисунок 41. Сигнал типа MadaCalc добавлен в набор сигналов

### 5.1.2.1. Параметры сигнала типа MadaCalc

Сигнал типа **MadaCalc** показывает прогнозируемую тенденцию изменения значения показателя, получаемого от целевого объекта.

#### Имя

Имя сигнала. Вводится при создании сигнала и в дальнейшем изменению не подлежит.

#### Тип

Тип сигнала. Выбирается при создании сигнала и в дальнейшем изменению не подлежит.

#### Отображаемое имя

Имя сигнала, отображаемое в дереве конфигурации.

#### Описание

Краткое описание проблемы. Введенный текст отображается в представлении **Консоль событий** в столбце **Описание**.

#### Функция

Функция для расчёта значения сигнала, если **Режим** = **Диапазон**. Доступны следующие функции:

- ❖ **среднее** (avg) – среднее значение;
- ❖ **минимальное** (min) – минимальное значение;
- ❖ **максимальное** (max) – максимальное значение;
- ❖ **последнее** (last) – последнее известное значение. Это значение может быть только числом;

- ❖ **текущее** (*cur*) – последнее значение. Это значение не обязательно будет числом;  
Например, если целевой объект стал недоступным, то функция *last* возвращает значение, полученное в момент, когда целевой объект был еще доступен, а функция *cur* возвращает значение **NAN**;
- ❖ **последнее известное** (*ldef* – от last defined) – выбирает из базы данных последнее известное значение. Если значения отсутствуют или не определены, то функция возвращает значение **NAN**;
- ❖ **не определено** – эквивалент незаданному значению; допустимо только как локальное значение параметра **Функция** для указания на использование глобального значения.

### Режим для time to event

Параметр применяется, только если в сценарии задачи используется обработчик **Расчёт порогов** в режиме **time to event**<sup>97</sup>. Параметр задаёт предполагаемое событие для алгоритма расчёта значения и статуса сигнала типа **MadaCalc**.

Алгоритм оперирует следующими величинами:

- x* – реальное значение исследуемого параметра, полученное от целевого объекта;
- p* – агрегированное предсказанное значение исследуемого параметра; рассчитывается следующим образом: значения, вычисленные по задаче, указанной в поле **Задачи из MADA-менеджера**, и ограниченные периодом, заданным в поле **Диапазон**, агрегируются функцией, заданной в поле **Функция**;

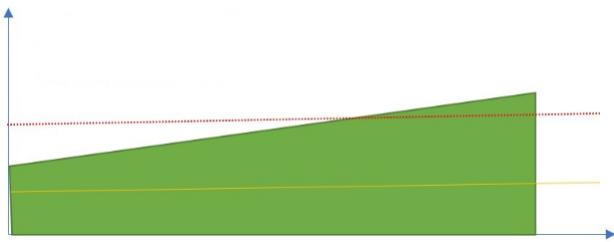
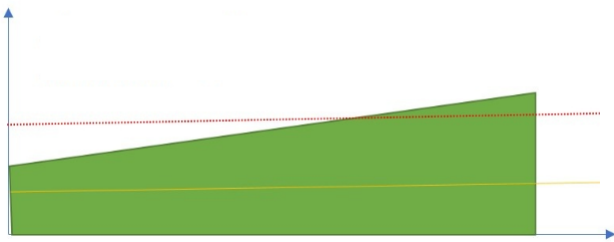
Алгоритм рассчитывает значение сигнала (величину *t*) – прогнозируемый интервал времени (в секундах) до наступления предполагаемого события. Если предполагаемое событие уже наступило, либо оно не предвидится (согласно предсказанию), то сигналу присваивается значение **0**. На основании значения сигнала и в соответствии с логикой алгоритма сигналу типа **MadaCalc** присваивается статус. Возможны 3 варианта алгоритма – в зависимости от предполагаемого события. Далее описан каждый вариант.

**жёлтый порог** – в качестве предполагаемого события рассматривается пересечение исследуемым показателем жёлтого порога;

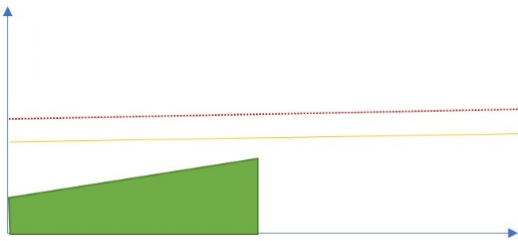
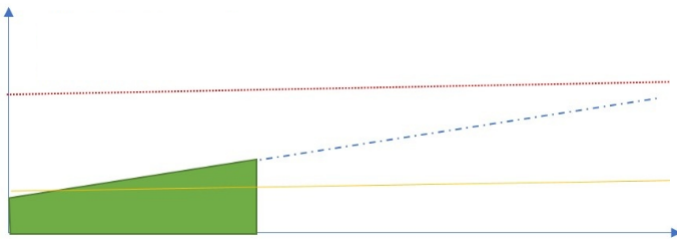
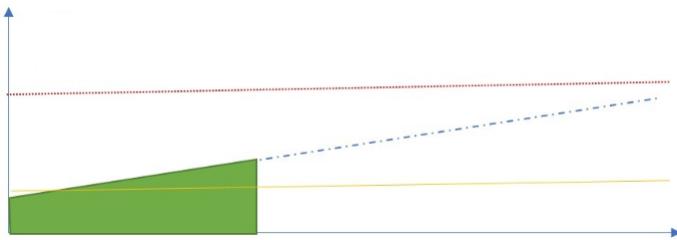
Условие	Результат (сигнал типа MadaCalc)	
	Значение сигнала	Статус сигнала
$x > \text{жёлтый порог}$  (значение исследуемого параметра превысило жёлтый порог)	<b>0</b>	<b>красный</b>
$x < \text{жёлтый порог} < p$  (значение исследуемого параметра ниже жёлтого порога, однако предсказанное значение превышает его)	<b>t</b>	<b>красный</b>
в остальных случаях	<b>0</b>	<b>зелёный</b>

**красный порог** – в качестве предполагаемого события рассматривается пересечение исследуемым показателем красного порога;

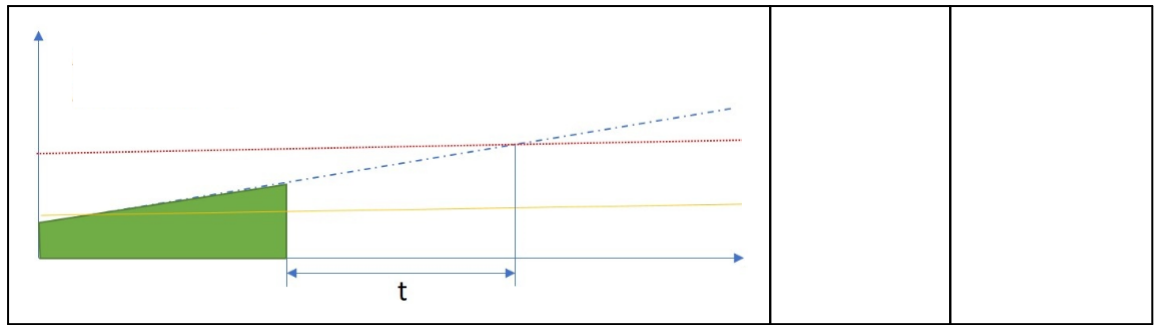
Условие	Результат	
	Значение сигнала	Статус сигнала

$x > \text{красный порог}$ (т.е. значение исследуемого параметра превысило желтый порог) 	0	красный
$x < \text{красный порог} < p$ (т.е. значение исследуемого параметра ниже жёлтого порога, однако предсказанное значение превышает его) 	t	красный
в остальных случаях	0	зелёный

**желтый и красный порог** – в качестве предполагаемого события рассматривается пересечение исследуемым параметром жёлтого или красного порога;

Условие	Результат	
	Значение сигнала	Статус сигнала
$x < \text{жёлтый порог}$ 	0	зелёный
$\text{жёлтый порог} < x < \text{красный порог}$ и $p < \text{красный порог}$ 	0	жёлтый
$\text{жёлтый порог} < x < \text{красный порог}$ и $p > \text{красный порог}$ 	t	красный





### Диапазон (range)

Период (в секундах), используемый функцией, выбранной в поле **Функция**, для определения значения сигнала. По умолчанию **600** секунд

### Задачи из MADA-Manager

Задача для прогнозирования значений показателя. По нажатию кнопки ☐ открывается окно со списком задач, созданных в **Редакторе математических моделей**. Чтобы выбрать задачу, следует поместить курсор на соответствующую строку и нажать кнопку **Сохранить**.

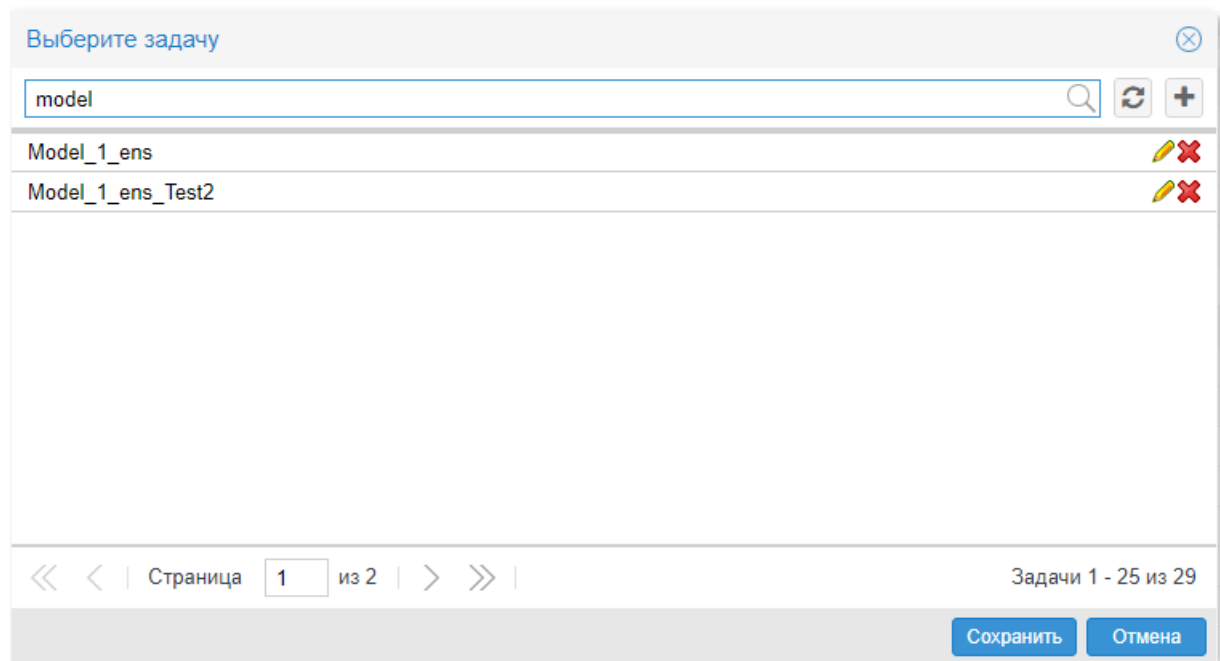





Рисунок 42. Поиск задачи, созданной в web-приложении Редактор математических моделей

В окне выбора задачи можно воспользоваться строкой поиска.

Если нужная задача в списке отсутствует, то ее можно создать: нажать кнопку  и в появившемся окне **Создать задачу** (из web-приложения **Редактор математических моделей**) заполнить поля на нескольких страницах, см. п. [Настройка сценария и задачи в Редакторе матем. моделей](#)<sup>[40]</sup> или [Создание задачи](#)<sup>[31]</sup>. Существующую задачу можно просмотреть или отредактировать (кнопка ) или удалить (кнопка ).

### Результат задачи для предсказания

Конечная точка задачи. Выбирается из списка.

### Результат задачи для границ

Конечная точка задачи. Выбирается из списка.

### Показать предсказание

Необходимость отображать прогноза на графике исследуемого параметра.

### 5.1.3. Привязка набора сигналов к целевому объекту

К целевому объекту необходимо привязать настроенный набор сигналов.

Примечание. Если к целевому объекту уже привязан набор из сигналов типа **Availability**, **Gauge** и т.п., то рекомендуется в дереве конфигурации создать ссылку на целевой объект и к ней привязать набор сигналов, включающий в себя один сигнал типа **MadaCalc**.

Чтобы привязать настроенный набор сигналов к целевому объекту, следует:

1. в окне **Дерево конфигурации** поместить курсор на нужный целевой объект;
2. с помощью правой кнопки мыши в контекстном меню выбрать пункт **Привязать сигнал**;

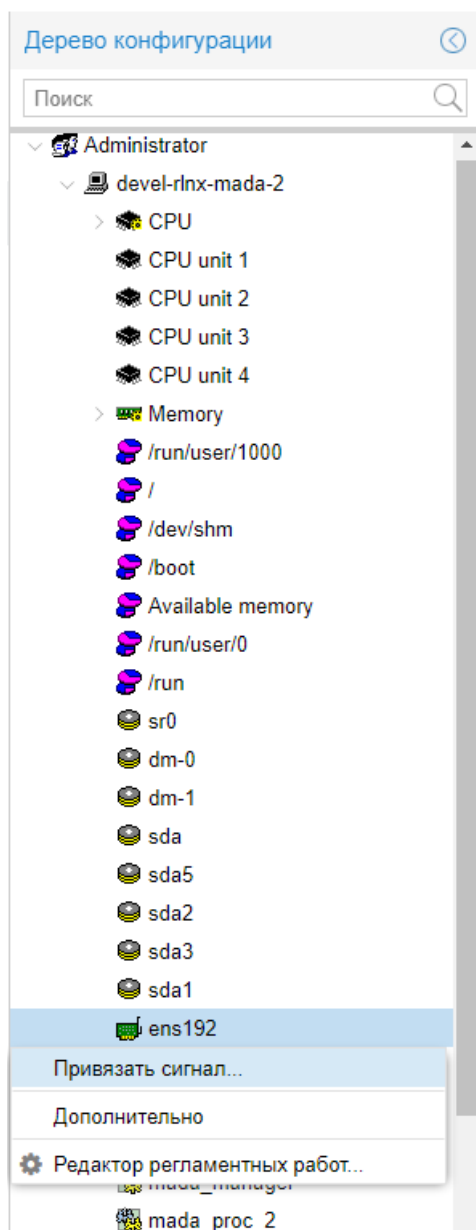


Рисунок 43. Пункт меню для вызова окна для привязки набора сигналов

в результате открывается окно **Привязать сигнал** со списком наборов сигналов, у которых тип назначения совпадает с типом целевого объекта;

3. отметить набор сигналов, созданный на предыдущем [этапе](#)<sup>42</sup>;

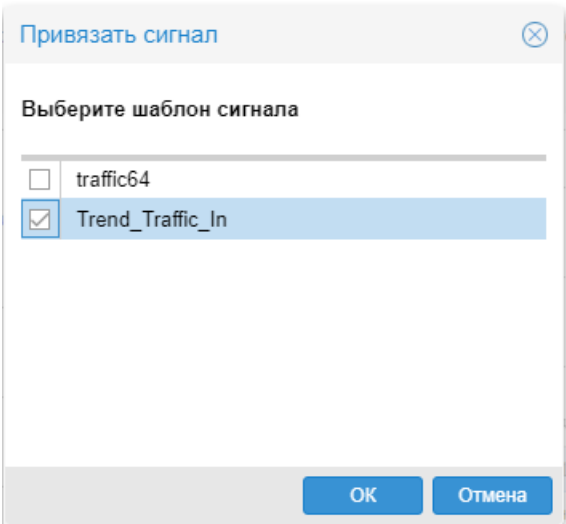


Рисунок 44. Выбор набора сигналов

4. нажать кнопку **ОК**.

В результате в окне **Дерево конфигурации** привязанный набор сигналов отображается как дочерний узел целевого объекта, а на пиктограмме целевого объекта, к которому был привязан набор сигналов, появляется желтый значок.

Результат прогнозирования выводится на графике на web-странице целевого объекта. Прогноз отображается в виде точек, соединённых линией.

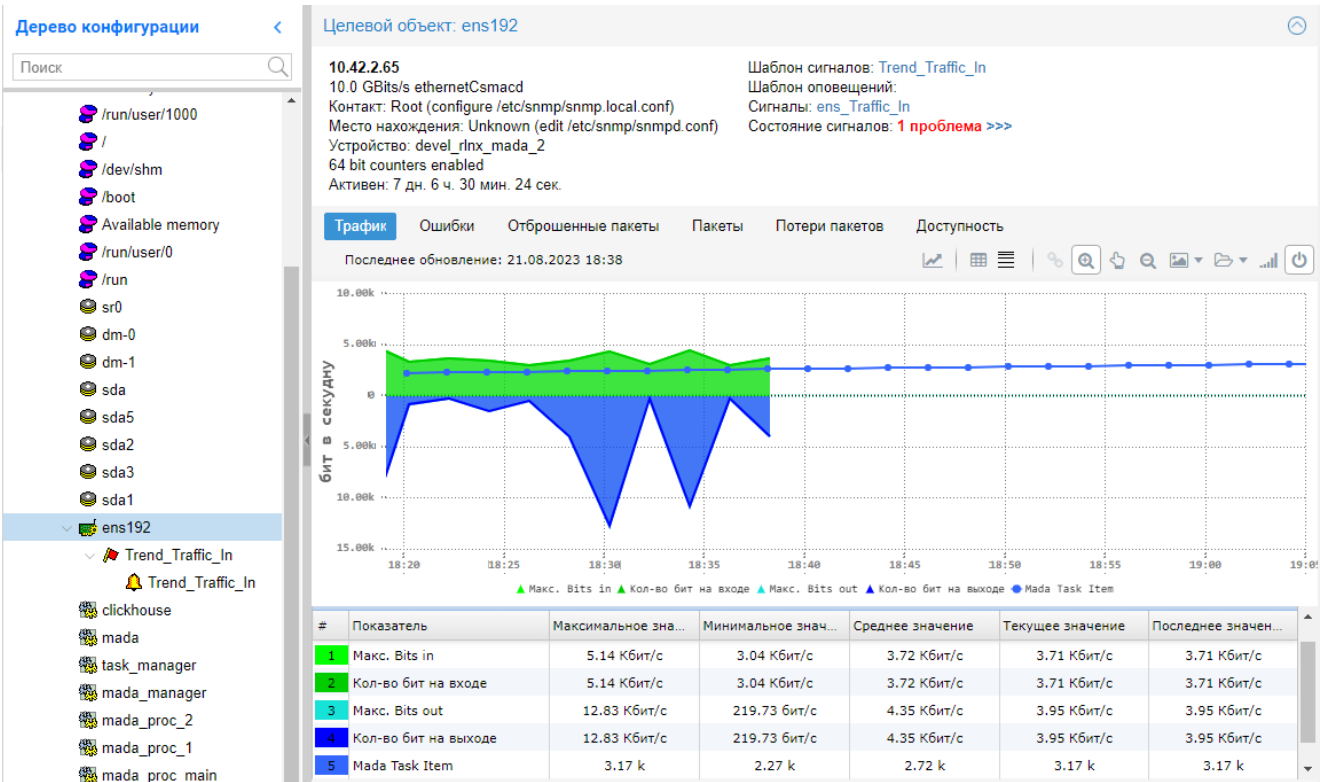


Рисунок 45. Для параметра "Количество бит на входе" построен график прогноза "Mada Task Item"



---

# 6.

## Справочники

## 6.1. Термины и определения

В данном разделе в алфавитном порядке приведены используемые в документе термины, определения и сокращения.

**ClickHouse** – колоночная аналитическая СУБД с открытым кодом, позволяющая выполнять аналитические запросы в режиме реального времени на структурированных больших данных.

**База данных производительности** – база данных, обеспечивающая хранение и отображение динамических параметров, полученных от объектов контроля.

**Дерево конфигурации** – иерархическая структура папок, построенная по принципу вложенности, содержащая информацию об объектах контроля и их модели представления.

Приложение **Диспетчер процессов** – компонента Web-интерфейса. Позволяет администратору посредством графического пользовательского интерфейса управлять системными процессами, функционирующими на сервере PhoenixDS и обеспечивающими работу ПК **PhoenixDS Monitoring** и других компонент ПП PhoenixDS.

**Задача** – это применение сценария для решения прикладной задачи на конкретных входных данных из доступного источника.

**Значение сигнала** – числовое значение, полученное в результате обработки значений параметров целевых объектов за определенный интервал времени (например: среднее значение, максимальное или минимальное значение). Значение сигнала используется для вычисления статуса сигнала.

**Набор сигналов** – совокупность сигналов, которые применимы к источникам данных, принадлежащим целевым объектам определенного типа. Количество сигналов в наборе ограничивается количеством источников данных, поддерживаемых ПП PhoenixDS для выбранного типа целевого объекта.

**Обработчик** – это встроенный математический инструмент для обработки данных (например, математическая модель, фильтр, метод оценки и др.).

**Объект дерева конфигурации** – компонента дерева конфигурации, созданная администратором или локальным администратором на основе *шаблона объекта* и выполняющая конкретную задачу в реализации функционального назначения ПП PhoenixDS.

**Объект контроля** – актив, который необходимо контролировать при осуществлении деятельности предприятия; например: технологический или информационно-телекоммуникационный процесс, производственный ресурс, состояние которого влияет на принятие решений в процессах управления предприятием.

**Период регламентных работ** (период РР) – запланированный интервал времени, в течение которого целевой объект может подвергаться внешним воздействиям (например, может быть недоступен по причине отключения электроэнергии).

**Проектное решение** – совокупность программных компонентов, созданная на базе ПП PhoenixDS с учетом особенностей информационной среды заказчика и настроенная для решения согласованного круга задач в рамках определенного договора (проекта).

Web-приложение **Редактор математических моделей** является компонентой ПМ **PhoenixDS Analytics** и позволяет с помощью встроенных математических инструментов создавать расчётные сценарии, настраивать на их основе выполнение задач прогнозирования поведения параметров, просматривать результаты в наглядной графической форме.

Приложение **Редактор сигналов и оповещений** – компонента Web-интерфейса. Позволяет администратору настраивать наборы сигналов и наборы оповещений.

**Сервер PhoenixDS** – сервер, на котором установлен ПК **PhoenixDS Monitoring**. На этот сервер (в соответствии с согласованными техническими условиями проектного решения) также могут быть установлены другие серверные компоненты ПП PhoenixDS и модули.

**Сервер MADA** – сервер с установленной основной библиотекой, реализующей математический аппарат ПМ **PhoenixDS Analytics**.

**Системный процесс** – это процесс (daemon) на сервере PhoenixDS, запускаемый с определенными значениями параметров. **Процесс** (daemon) является встроенным объектом ПК **PhoenixDS Monitoring**. Системные процессы создаются и настраиваются администратором в зависимости от задач, решаемых с помощью ПП PhoenixDS.

**Сигнал** – правило обработки значений параметров целевых объектов и определения статуса сигнала.

Сигнал типа **MadaCalc** показывает прогнозируемое значение показателя целевого объекта и позволяет определить доверительные интервалы (динамические границы) для будущих значений показателя.

**Событие** – существенное изменение состояния целевого объекта, например, изменение статуса сигнала.

**Статус сигнала** – характеристика сигнала, отражающая наличие проблемы на целевом объекте и уровень ее критичности. Статус сигнала идентифицируется цветом и числовой константой.

**Сценарий** – это совокупность обработчиков и связей между ними, определяющая алгоритм расчёта выходных значений для решения поставленных задач.

**Тип сигнала** – алгоритм вычисления статуса сигнала, ограничивающий множество возможных статусов сигнала.

**Целевой объект** – минимальная неделимая компонента (логическая или физическая) объекта контроля, которая является источником данных, для последующего формирования обобщенного состояния объекта контроля.

**Шаблон сценария** – предустановленный сценарий, предназначенный для решения типовой прикладной задачи (например: прогнозирование, сглаживание, вычисление корреляции и др.).

## 6.2. Системные процессы ПК PhoenixDS Monitoring

В настоящем справочнике приведено формализованное описание системных процессов ПК **PhoenixDS Monitoring**, необходимых для функционирования ПМ **PhoenixDS Analytics**. Пункты отсортированы по именам процессов в алфавитном порядке.

Для управления системными процессами (если не указано особо) можно использовать утилиту `overlord.pl` или приложение **Диспетчер процессов** в Web-интерфейсе.

Параметры системных процессов можно условно разделить на две группы:

- ❖ стандартные – имеются у большинства системных процессов и описаны в п. [Стандартные параметры системных процессов](#)<sup>56</sup>;
- ❖ специфические – описаны в пунктах, посвященных конкретным системным процессам.

### 6.2.1. Стандартные параметры системных процессов

В настоящем разделе приведены параметры, которые имеются у большинства системных процессов. Особенности ввода значений некоторых стандартных параметров содержатся в пунктах с описанием системных процессов.

`name`

Имя системного процесса.

`loglevel`

Уровень логирования – степень детализации информации в лог-файле. Возможные значения:

**Info** – базовая информация (наиболее эффективный уровень);

**Debug** – детальная информация (применяется на этапе разработки/внедрения);

**Warn** – информация о предупреждениях;

**Error** – информация об ошибках.

`logfile`

Директория и имя лог-файла для записи отладочной информации. Не рекомендуется изменять директорию хранения лог-файла по умолчанию `/usr/local/lancelot/var/lancelot-logs/`. Системный процесс создает отдельный файл для каждого потока программы с указанием в имени файла номера PID программы, порядкового номера потока и номера файла в данном потоке.

`interval`

Периодичность работы системного процесса, в секундах.

`esb`

IP-адрес хоста, на котором работает шина `esb` (по умолчанию **127.0.0.1**). Чтобы отключить системный процесс от шины ESB, необходимо ввести значение **false**.

`fastcache`

Отключение от сервера конфигурации, использование отдельного кеша.

`list`

Ветви дерева конфигурации (через запятую).

### 6.2.2. Обмен данными между ПК PhoenixDS Monitoring и ПМ PhoenixDS Analytics (mada.pl)

Системный процесс `mada`, основанный на процессе `mada.pl`, отвечает за отправку и получение данных от сервера **PhoenixDS Analytics**, а также за сравнение новых данных с прогнозируемым значением.



## 6.3. Обработчики

**Обработчики** – это составные элементы сценария, определяющие его поведение. Они включают в себя различные группы, в том числе [Виджеты](#)<sup>[57]</sup>, [Ввод/Вывод](#)<sup>[57]</sup>, [Причинно-следственный анализ](#)<sup>[98]</sup>, [Общие](#)<sup>[84]</sup>, [Оценщики](#)<sup>[87]</sup>, [Фильтры](#)<sup>[102]</sup>, [Функции](#)<sup>[107]</sup>, [Модели](#)<sup>[57]</sup>, [Модификаторы](#)<sup>[71]</sup>, [Нормализация](#)<sup>[83]</sup>, [Партиционирование](#)<sup>[88]</sup>, [Постобработка](#)<sup>[89]</sup>. Эти разнообразные типы обработчиков могут быть легко интегрированы в **Редактор математических моделей** с помощью перетаскивания их из окна **Обработчики** на канву. Каждый обработчик требует настройки, в ходе которой задаются основные параметры и устанавливаются необходимые связи, причем тип входных данных следующего обработчика должен совпадать с типом выходных данных предыдущего обработчика.

### 6.3.1. Ввод/Вывод

Группа обработчиков Ввод/Вывод играет роль в структурировании и определении потока сценария. Эта группа включает в себя два основных обработчика: **Ввод** и **Вывод**, которые служат маркерами начала и конца процесса создания сценария.

Обработчик **Ввод** обозначает начальную точку сценария и позволяет определить тип входных данных; это важно для точного представления характера исходных данных, будь то временные ряды, изображения, матрицы или другие типы данных.

Обработчик **Вывод** обозначает конечную точку сценария, свидетельствующую о завершении его работы.

#### Параметры обработчика Ввод

##### Тип начальной точки

Тип входных данных: **timeseries**, **image**, **multitimeseries**, **matrix**, **vector**, **bool**, **int**, **str**, **float**, **dict**.

##### Имя точки

Название начальной точки.

#### Параметры обработчика Вывод

##### Имя точки

Название конечной точки.

### 6.3.2. Виджеты

Виджеты – это вспомогательные элементы, которые дополняют сценарий. Группа Виджеты включает в себя обработчик **Комментарий** – фундаментальный виджет, позволяющий вставлять пользовательские комментарии непосредственно на канву.

### 6.3.3. Модели

Модель – это математический аппарат для моделирования временного ряда на основе входных данных (массива значений исследуемого параметра) и получения прогноза.

Реализованы следующие классы моделей:

линейные (**Линейный тренд**, **Средний тренд**, **Тренд скользящего среднего**);

Линейная модель служит для описания поведения зависимой переменной (отклика) как непрерывной линейной функции нескольких независимых переменных (предикторов). Модель предусматривает только линейные зависимости, причем нелинейность взаимосвязей может быть приведена к линейности путем математических преобразований переменных.

Линейные модели позволяют описать тенденцию поведения рассматриваемого временного ряда, но, в силу специфики функции, не подходят для прогнозирования колебаний.

авторегрессионные (**ARIMA, Auto-ARIMA, ARM**);

Авторегрессионная (autoregressive model, AR) модель – модель временного ряда, в которой его текущее значение линейно зависит от предыдущих (ретроспективных) значений этого же ряда. От того, сколько ретроспективных значений включено в модель, зависит ее порядок. Основное назначение авторегрессионной модели – прогнозирование. Кроме этого, с её помощью можно производить анализ временных рядов – выявлять тенденции, сезонность и другие особенности.

нейросетевые (**Рекуррентная нейронная сеть, Машина опорных векторов**);

Нейронные сети – обширная группа математических моделей, позволяющая решать различные задачи, начиная от аппроксимации и заканчивая, но не ограничиваясь, генераторами текста и распознаваниями образов, в зависимости от выбранной архитектуры. Применительно к задаче прогнозирования временных рядов, нейронные сети используются для моделирования поведения рассматриваемого процесса и поиска скрытых зависимостей между его текущими и историческими значениями.

Нейронная сеть состоит из отдельных модулей – нейронов, являющихся частью аппроксимирующей функции, параметры для которой подбираются в процессе обучения модели на большом наборе данных. Такой подход позволяет моделировать сложные процессы, описываемые суперпозициями различных функций.

Общие параметры нейросетевых моделей:

**Функция активации** (activation function) – функция, определяющая выходное значение нейрона на каждом слое в зависимости от результата взвешенной суммы, пришедшей на нейрон.

**Функция потерь** (loss function) используется для расчета ошибки между реальными данными и полученными с помощью нейронной сети значениями. Глобальная цель обучения нейронной сети – минимизировать эту ошибку.

**Функция оптимизации** (optimization function) – алгоритм поиска минимума функции потерь.

ансамблевые (**Экстремальный градиентный бустинг**).

Парадигма ансамблевых моделей использует несколько обучающих алгоритмов с целью получения лучшей эффективности прогнозирования, чем могли бы получить от каждого обучающего алгоритма по отдельности. Главная гипотеза ансамблевых моделей состоит в том, что при правильном сочетании моделей можно получить более точные и/или надежные модели.

Каждая модель имеет параметр **доля** – количество данных (в процентах), используемых для моделирования. Остальные данные используются для расчёта оценки качества модели. Если **доля** = **100**, то оценка качества модели не вычисляется.

## Стандартные параметры моделей

В результате выбора модели формируются поля для ввода ее параметров. Многие модели имеют специфические параметры. Однако каждая модель имеет два стандартных параметра:

**горизонт (horizon)** – интервал времени (в секундах), на который строится предсказание;

**доля (test\_data\_duration)** – доля данных для обучения модели.

### 6.3.3.1. Линейный тренд

**Линейный тренд** – интуитивно понятная модель, описывающая равномерное изменение параметра во времени. Относится к классу [линейных моделей](#)<sup>[57]</sup>.



Рисунок 46. График линейного тренда

### 6.3.3.2. Средний тренд

**Средний тренд** – линейная модель, предназначенная для построения среднего значения для исходных данных. Относится к классу [линейных моделей](#)<sup>[57]</sup>.

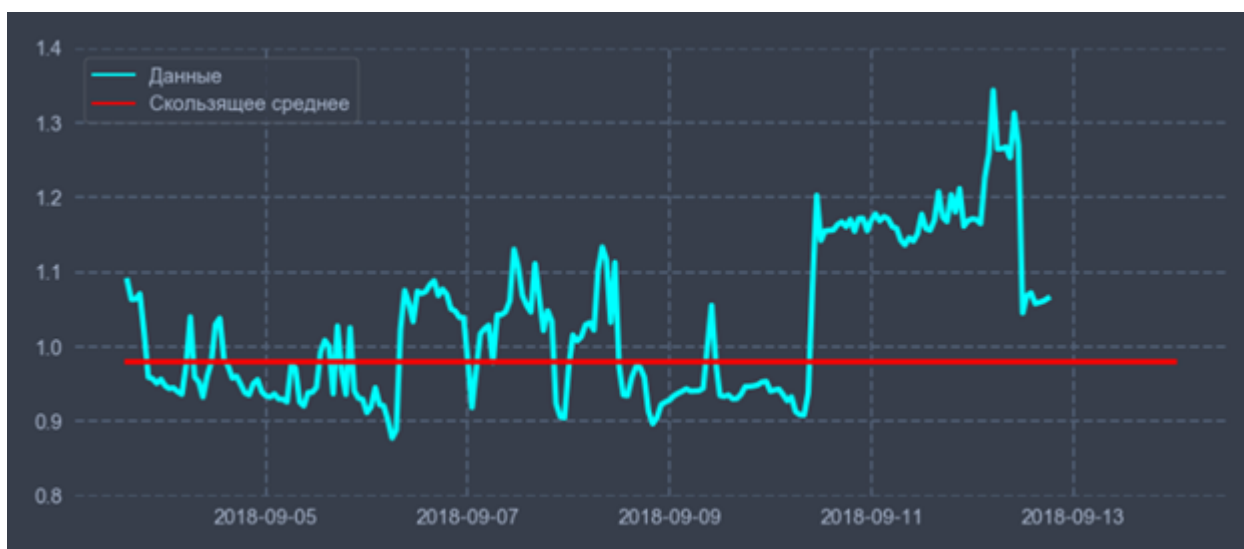


Рисунок 47. График с рассчитанным средним

### 6.3.3.3. Тренд скользящего среднего

**Тренд скользящего среднего** – линейная модель построения скользящего среднего, то есть для каждой точки значение равно среднему значению N данных за предыдущий период. Относится к классу [линейных моделей](#) <sup>[57]</sup>.

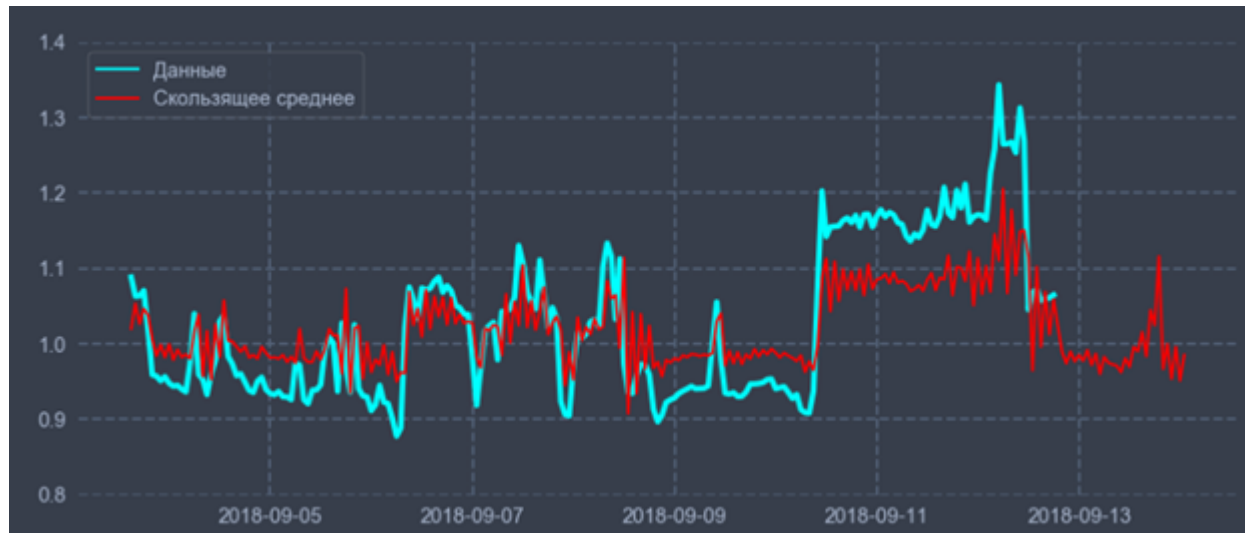


Рисунок 48. График с рассчитанным скользящим средним

### Специфические параметры модели

ma

Параметр, связанный со скользящим средним, который позволяет установить погрешность модели как линейную комбинацию наблюдавшихся ранее значений ошибок. Возможные значения: **1** (по умолчанию), **2**, **3**.

### 6.3.3.4. ARIMA

**ARIMA** – авторегрессионное интегрированное скользящее среднее (autoregressive integrated moving average, ARIMA) является расширением моделей ARMA для нестационарных временных рядов, которые можно сделать стационарными взятием разностей некоторого порядка от исходного временного ряда. Относится к классу [авторегрессионных моделей](#) <sup>[57]</sup>.



Рисунок 49. График с рассчитанным и предсказанным ARIMA смоделированного процесса

Примечание. Обычно модель упоминается как ARIMA (**ar**, **diff**, **ma**), где **ar**, **diff** и **ma** — целые неотрицательные числа, характеризующие порядок для частей модели (соответственно авторегрессионной, интегрированной и скользящего среднего). Следует отметить, что модель ARIMA в основном используется для построения краткосрочных прогнозов.

## Специфические параметры модели

### diff

Параметр, связанный с интегрированной частью модели, который влияет на величину разности, применяемой к временному ряду. Возможные значения:

- 0** — временной ряд не нужно дифференцировать, он обладает свойством стационарности (т.е. свойства этого ряда не изменяются во времени, в частности, он имеет постоянное математическое ожидание, постоянную дисперсию и др.). Временной ряд с трендом и сезонностью не является стационарным.
- 1** — временной ряд необходимо привести к стационарному путем дифференцирования. Рекомендуется использовать, когда временной ряд имеет трендовую или сезонную составляющую (по виду сигнала можно определить возрастает ли график, убывает или колеблется вокруг горизонтальной прямой).
- 2** — модель с двумя порядками разности необходима, если предполагается, что исходный ряд имеет тенденцию, изменяющуюся во времени.

**ВНИМАНИЕ!** Лучше избегать излишней дифференциации, поскольку передифференцированный ряд может исказить данные, а значит, — и предсказание.

### ma

Параметр, связанный со скользящим средним, показывающий сколько ошибок прошлых значений включает в себя модель. Возможные значения: **0** (по умолчанию), **1**, **2**.

### ar

Параметр, связанный с авторегрессионной частью модели, показывающий, сколько прошлых значений включает в себя модель. Возможные значения: **0** (по умолчанию), **1**, **2**.

## Рекомендации по выбору параметров $ar$ и $ma$

- ❖  $ar + ma > 1$ , т.е. хотя бы один параметр должен быть больше нуля.
- ❖ Без дополнительного анализа можно подбирать значения параметров так, чтобы они не были равны (например ARMA(1, 2)). В анализе часто выбираются  $ar$  и  $ma$  с различием в единицу.
- ❖ На основе различных оценок модели (estimators) можно выбрать наиболее подходящую для определенного сигнала.

### 6.3.3.5. Auto-ARIMA

**Auto-ARIMA** – позволяет подобрать лучшую ARIMA модель на основании информационного критерия Акайки и Байесовском информационном критерии. Выполняется пошаговый поиск по всем возможным модельным и сезонным порядкам (в пределах предоставленных ограничений) и выбираются параметры, минимизирующие данную метрику. Относится к классу [авторегрессионных моделей](#)<sup>[57]</sup>.



Рисунок 50. График с рассчитанным и предсказанным SARIMA смоделированного процесса

Модель может учитывать также сезонные составляющие данных. Тогда в общем виде получается модель: SARIMA(p, d, q)(P, D, Q)S, где

p, d, q – параметры модели ARIMA;

P, D, Q – порядки сезонной составляющей;

S – размерность сезонности (день, месяц, квартал, год и т.д.).

## Специфические параметры модели

### diff

Начальное значение параметра, связанного с интегрированной частью модели. Влияет на величину разности, применяемой к временному ряду. Если параметр **diff** = **-1** (по умолчанию), то значение будет автоматически выбрано на основе результатов теста (KPSS, ADF или PP), а время построения прогноза может значительно возрасти. Возможные значения: **-1, 0, 1, 2**.

### diff1

Порядок сезонной дифференциации; если **diff1** = **-1** (по умолчанию), то значение будет автоматически выбрано по результатам специального теста. Возможные значения: **-1, 0, 1**.

### max\_ar

Максимальное значение **ar**, причём **max\_ar** >= **start\_ar**. Возможные значения: **0, 1, 2, 3, 4, 5** (по умолчанию).

#### **max\_ar1**

Максимальное значение **ar1**, причём **max\_ar1** >= **start\_ar1**. Возможные значения: **0, 1, 2** (по умолчанию).

#### **max\_diff**

Максимальное значение **diff**, причём **max\_diff** >= **diff**. Возможные значения: **0, 1, 2** (по умолчанию).

#### **max\_diff1**

Максимальное значение **diff1**, причём **max\_diff1** >= **diff1**. Возможные значения: **0** и **1** (по умолчанию).

#### **max\_ma**

Максимальное значение **ma**, причём **max\_ma** >= **start\_ma**. Возможные значения: **0, 1, 2, 3, 4, 5** (по умолчанию).

#### **max\_ma1**

Максимальное значение **ma1**, причём **max\_ma1** >= **start\_ma1**. Возможные значения: **0, 1, 2** (по умолчанию).

#### **период сезонности (season\_period)**

Период для сезонной разности в секундах. Данный параметр относится к числу периодов в каждом сезоне. Если **период сезонности** = **1**, то сезонность отсутствует (**diff1**, **max\_ar1**, **max\_ma1** будут установлены в **0**).

#### **start\_ar**

Начальное значение параметра, связанного с авторегрессионной частью модели. Показывает, сколько прошлых значений включает в себя модель. Возможные значения: **0, 1, 2** (по умолчанию).

#### **start\_ar1**

Начальное значение **ar1**, порядок авторегрессионной части сезонной модели. Возможные значения: **0, 1** (по умолчанию).

#### **start\_ma**

Начальное значение параметра, связанного с частью модели – скользящим средним. Возможные значения: **0, 1, 2** (по умолчанию).

#### **start\_ma1**

Начальное значение **ma1**, порядок скользящего среднего для сезонной модели. Возможные значения: **0, 1** (по умолчанию).

### **Рекомендации по выбору параметров**

- ❖ Любые начальные и максимальные значения могут быть равны, например, **start\_ar** = **max\_ar**. Тогда итерационного процесса по этому параметру (в приведенном примере по **ar1**) не будет. Если знать все параметры, можно задать таким образом модель ARIMA или SARIMA, не запуская итерационный процесс, сократив тем самым время работы.
- ❖ Если временной ряд имеет сезонную составляющую, то необходимо использовать порядок сезонных различий **diff1**. Ярко выраженную сезонность можно определить по графику временного ряда и заметить повторяющиеся элементы с определенной периодичностью. Можно также опираться на знания о природе поведения исследуемого параметра: существует ли его активность в определенное время, повторяющаяся с каким-либо периодом.

- ❖ Не следует использовать более одного порядка сезонных различий или более двух порядков общего дифференцирования (**diff + diff1**).
- ❖ Следует стараться избегать использования более одного или двух сезонных параметров (**ar1 + ma1**) в одной и той же модели, так как это, скорее всего, приведет к переобучению данных и/или к проблемам в ее оценках.

### 6.3.3.6. ARM

**ARM** – аддитивная модель, в которой каждый уровень временного ряда может быть представлен как сумма трендовой, сезонной и случайной компонент. При моделировании происходит подгонка нескольких линейных и нелинейных функций от времени, при этом регрессором в данном случае является временной показатель. Моделирование сезонности, как составляющей аддитивной модели, осуществляется методом, аналогичным методу Хольта-Винтерса (создаётся трёхпараметрическая модель прогноза, учитывающая сглаженный экспоненциальный ряд, тренд и сезонность). Отличительной особенностью настоящей модели является то, что она рассматривает задачу прогнозирования как подгонку под некоторую кривую, а не смотрит явно на зависимость каждого наблюдения от времени во временном ряду. Относится к классу [авторегрессионных моделей](#)<sup>[57]</sup>.



Рисунок 51. График поведения функции, рассчитанный и предсказанный аддитивной моделью

### 6.3.3.7. Рекуррентная нейронная сеть (RNN)

**Рекуррентная нейронная сеть (RNN)** – это нейронная сеть, в которой обратный сигнал с внутреннего слоя поступает не на «главные» входные нейроны, а на дополнительный слой – нейроны в таком слое хранят информацию о предыдущем входном векторе. Получается, что выходной слой (реакция сети) зависит не только от текущего входа, но и от предыдущего. Относится к классу [нейросетевых моделей](#)<sup>[57]</sup>.

На представленной ниже упрощенной схеме рекуррентной нейронной сети:

$X_1 \dots X_{\text{input\_dim}}$  – значения на входных нейронах;

$Z_1 \dots Z_{\text{hidden\_dim}}$  – значения на скрытом (рекуррентном) слое нейронов;

$Y$  – значение на выходном нейроне;

$w$  – весовые коэффициенты модели.



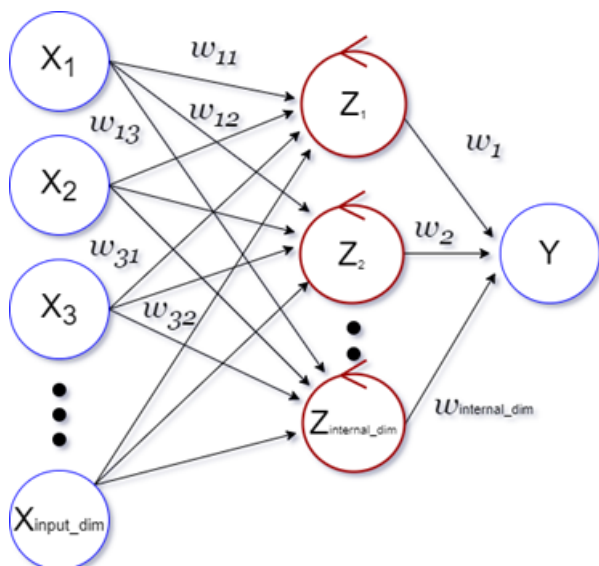


Рисунок 52. Упрощённая схема рекуррентной нейронной сети

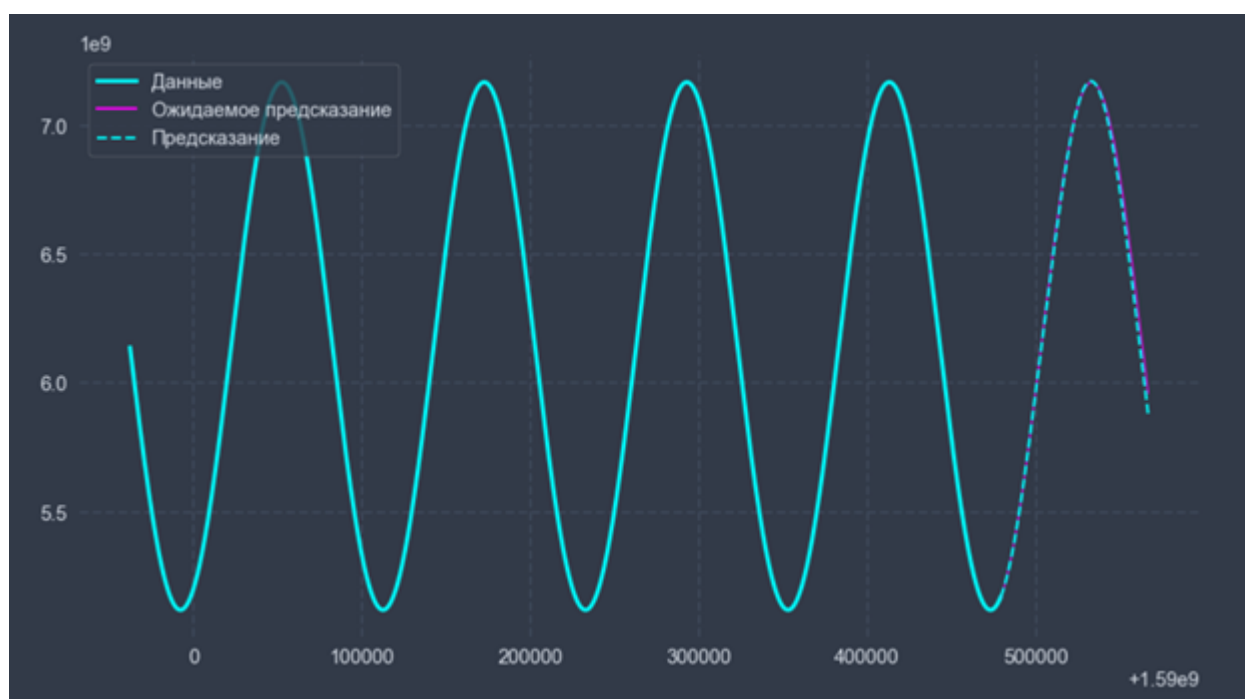
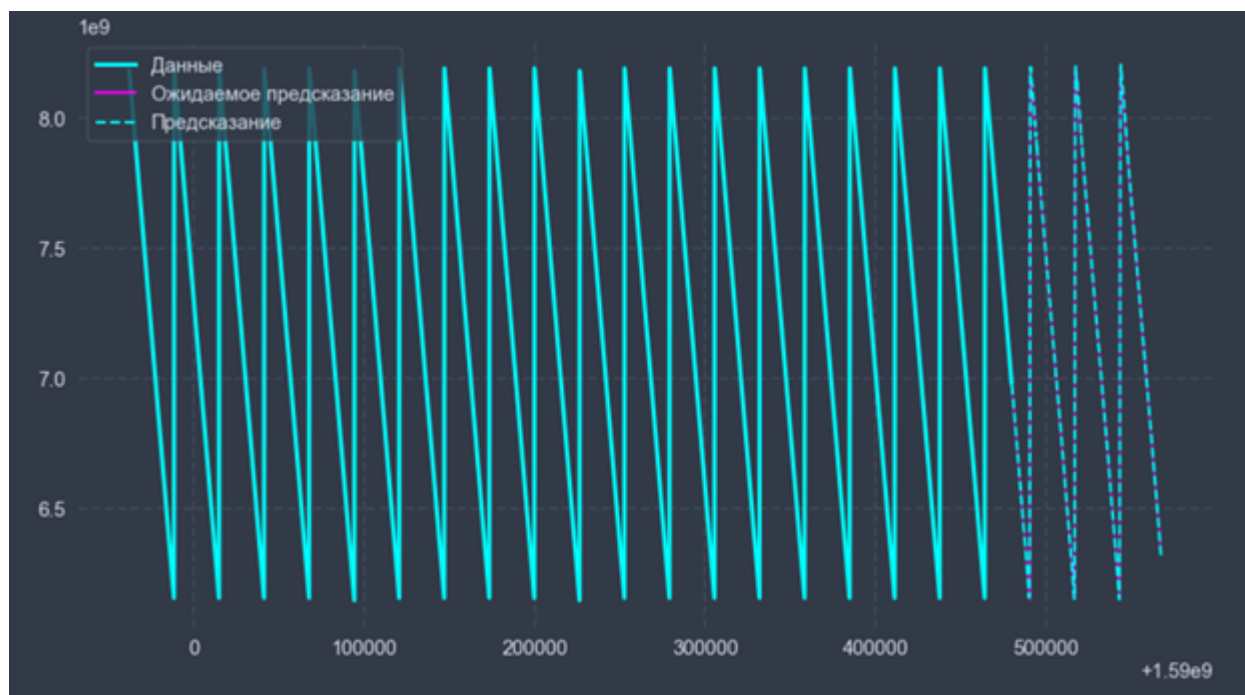
Тип рекуррентного слоя определяет особенности нейронной сети.

Большой сложностью сетей с **простым рекуррентным слоем** является проблема исчезающего (или взрывного) градиента, которая заключается в быстрой потере информации с течением времени. Конечно, это влияет лишь на веса, а не состояния нейронов, но ведь именно в них накапливается информация.

Сети со **слоем долгой краткосрочной памяти** стараются решить вышеупомянутую проблему потери информации, используя фильтры и явно заданную клетку памяти. У каждого нейрона имеется клетка памяти и три фильтра: входной, выходной и забывающий. Целью этих фильтров является защита информации. Входной фильтр определяет, сколько информации из предыдущего слоя будет храниться в клетке. Выходной фильтр определяет, сколько информации получают следующие слои. Забывающий фильтр забывает устаревшую информацию: например, если сеть изучает какой-либо процесс объекта, который переходит на другой режим эксплуатации, то предыдущие особенности можно забыть.

Сети, содержащие **управляемый рекуррентный блок** является вариацией предыдущего типа сети. У такого слоя на один фильтр меньше, и связи реализованы иначе. Фильтр обновления определяет, сколько информации останется от прошлого состояния и сколько будет взято из предыдущего слоя. Фильтр сброса работает как забывающий фильтр.

Примеры графиков с рассчитанными и предсказанными многослойной нейронной сетью значениями на стандартных функциях:





## Специфические параметры модели

### рекуррентный слой (recurrent\_layer)

Тип рекуррентного слоя нейронной сети. Возможные варианты:

- ❖ **простой рекуррентный слой (SimpleRNN)**;
- ❖ **слой долгой краткосрочной памяти (LSTM)** – по умолчанию;
- ❖ **управляемый рекуррентный блок (GRU)**.

### функция оптимизации (optimizer)

Название функции оптимизации, используемой на этапе обучения сети. Возможные варианты:

- ❖ **метод стохастического градиентного спуска (SGD)**;
- ❖ **алгоритм адаптивной оптимизация скорости обучения по методу Джеффа Хинтона (RMSprop)**;
- ❖ **адаптивный градиентный алгоритм (Adagrad)** – по умолчанию;
- ❖ **Adadelata**;
- ❖ **метод адаптивной оценки моментов (Adam)**;
- ❖ **Adamax**;
- ❖ **Нестеров-ускоренный адаптивный алгоритм (Nadam)**.

### функция потерь (loss)

Используется для расчета ошибки между реальными данными и полученными с помощью нейронной сети значениями. Глобальная цель обучения нейронной сети – минимизировать эту ошибку. Возможные варианты:

- ❖ **среднеквадратическая ошибка (mse)** – по умолчанию;
- ❖ **средняя абсолютная ошибка (mae)**;
- ❖ **средняя абсолютная ошибка в процентах (mape)**;
- ❖ **среднеквадратичная логарифмическая ошибка (msle)**;

- ❖ **квадрат верхней границы (squared hinge)**;
- ❖ **верхняя граница (hinge)**;
- ❖ **категориальная верхняя граница (categorical hinge)**;
- ❖ **логарифм гиперболического косинуса (logcosh)**;
- ❖ **расстояние Кульбака-Лейблера (kullback leibler divergence)**.

**размер порции данных (batch\_size)**

Размер выборки (точек данных) для каждого обновления градиента (используется при обучении). Минимально допустимое значение – **1**; значение по умолчанию – **20**.

**количество эпох (epochs)**

Количество эпох для обучения модели. Эпоха – это итерация обучения по всем предоставленным данным. Диапазон допустимых значений – от **1** до **150**; по умолчанию – **100**.

**количество итераций ожидания изменения (patience)**

Количество эпох, в течение которых контролируемая величина не улучшается, и после которых обучение останавливается. Минимально допустимое значение – **1**; по умолчанию – **4**.

**метод преобразования данных (split\_method)**

Способ преобразования данных. Возможные значения: **simple\_split** и **delay\_split**.

**Рекомендации по выбору параметров**

- ❖ В случае возникновения сложностей при подборе параметров сети, необходимо вернуться к значениям по умолчанию и, в первую очередь, варьировать параметр **размера порции данных (batch\_size)** и рекуррентный слой, а затем остальные нечисловые параметры **функция потерь (loss)** и **функция оптимизации (optimizer)**.
- ❖ Эффективность обучения нейронной сети при большом объеме входных данных будет выше, если использовать достаточно большие пакеты обучения (параметр **размер порции данных (batch\_size)**). При сравнительно небольшой длине входной выборки можно начать настройку сети с больших пакетов (тем самым сократить время обучения при подборе) и при достижении адекватного предсказания постепенно уменьшать размер пакета для достижения большей точности.  
Примечание. Параметр **размер порции данных (batch\_size)** должен нацело разбивать обучающую выборку, поэтому если заданное значение не удовлетворяет этому условию, алгоритм будет использовать ближайший делитель выборки. (В связи с этим, мало отличающиеся значения данного параметра могут приводить к одному результату обучения и предсказания сети).
- ❖ Параметр **количество эпох (epochs)** можно оставить достаточно большим, алгоритм отрегулирует значение, если для обучения потребуется меньшее количество.
- ❖ Значение по умолчанию параметра **количество итераций ожидания изменения (patience)**, определяющего остановку обучения, оптимально для большинства задач. Изменять его необходимо только для более тонкой настройки обучения.

**6.3.3.8. Машина опорных векторов (SVM)**

**Машина опорных векторов** – модель характеризуется отсутствием локальных минимумов при обучении; для решения задачи регрессии используются ядра (kernel trick). Относится к классу [нейросетевых моделей](#)<sup>57</sup>.

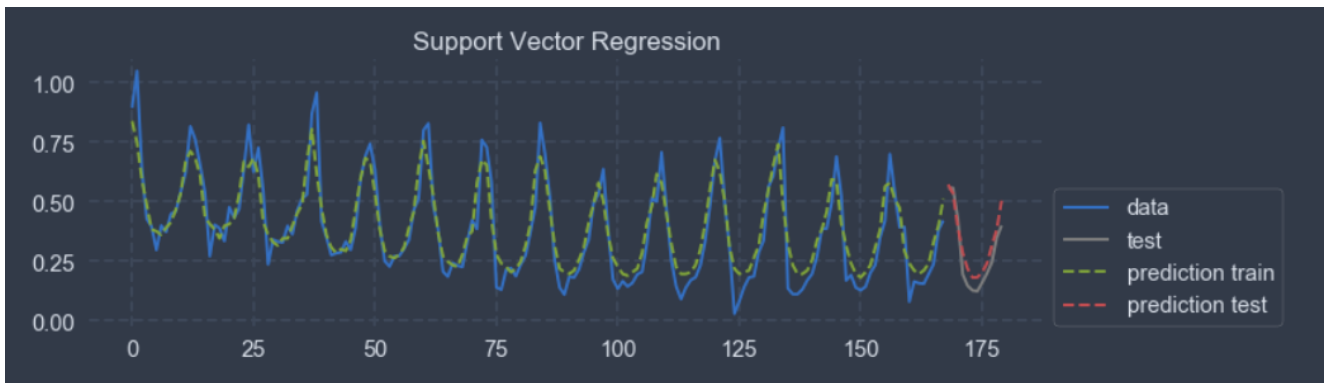


Рисунок 53. Пример прогноза, построенного машиной опорных векторов

## Специфические параметры модели

### ядро (kernel)

Определяет тип ядра, который будет использоваться в алгоритме. Возможные значения:

- ❖ **poly**;
- ❖ **rbf** (по умолчанию);
- ❖ **linear**;
- ❖ **sigmoid**.

### показатель степени (degree)

Степень функции ядра полинома (poly). Игнорируется всеми другими ядрами. Диапазон допустимых значений: от **2** до **100**, по умолчанию – **3**.

### коэффициент сходимости (learning\_rate)

Параметр скорости сходимости метода оптимизации. Диапазон допустимых значений: от **0.001** до **0.1**, по умолчанию – **0.001**.

### гамма (gamma)

Коэффициент ядра для **rbf**, **poly** и **sigmoid**. Возможные значения: **scale** (по умолчанию), **auto**.

### эпсилон (epsilon)

Параметр определяет эпсилон-окрестность, при попадании в которую разница между предсказанным и реальным значениями не учитывается при подсчете функции потерь. Диапазон допустимых значений: от **0.1** до **0.0001**, по умолчанию – **0.1**.

### параметр регуляризации (c)

Параметр регуляризации. Диапазон допустимых значений: от **1.0** до **100.0**, по умолчанию – **1.0**.

### независимый коэффициент (coef)

Независимый член в функции ядра. Используется для ядер **poly** и **sigmoid**. Диапазон допустимых значений: от **0.0** до **1.0**, по умолчанию: **0.0**.

### автоматический выбор c (auto\_select\_c)

Автоматический выбор параметра **c** (параметра регуляризации). Если **auto\_select\_c** принимает значение **True**, оптимальное значение параметра **c** будет определяться автоматически, если **auto\_select\_c** принимает значение **False** (по умолчанию), будет использоваться переданный параметр **c**.

### метод преобразования данных (split\_method)

Метод преобразования данных. Возможные значения: **simple\_split** (по умолчанию) и **delay\_split**.

## Рекомендации по выбору параметров

- ❖ **Параметр регуляризации (c)** является одним из самых важных параметров при настройке модели. Как правило, его увеличение ведет к улучшению прогноза модели. Однако, если установить его слишком большим, то велика вероятность не учесть главные свойства обучающей выборки. Увеличение параметра **c** ведет за собой увеличение времени работы модели.
- ❖ Увеличение параметра **показатель степени (degree)** для ядра **poly** увеличивает время вычисления модели.
- ❖ Уменьшение параметра **коэффициент сходимости (learning\_rate)** обычно ведет к более точной настройке модели, но увеличивается время обучения.
- ❖ Выбор параметра **автоматический выбор c (auto\_select\_c)** позволяет избежать ручной подборки параметра **c**, однако время, которое тратится на автоматическую подборку, увеличивает общее время обучения модели.

### 6.3.3.9. Экстремальный градиентный бустинг (XGBoost)

В основе модели **экстремального градиентного бустинга (XGBoost)** лежит алгоритм градиентного бустинга деревьев решений. Градиентный бустинг – это техника машинного обучения для задач регрессии, которая строит модель предсказания в форме ансамбля слабых предсказывающих моделей, обычно деревьев решений. Обучение ансамбля проводится последовательно. На каждой итерации вычисляются отклонения предсказаний уже обученного ансамбля на обучающей выборке. Следующая модель, которая будет добавлена в ансамбль будет предсказывать эти отклонения. Таким образом, добавив предсказания нового дерева к предсказаниям обученного ансамбля, можно уменьшить среднее отклонение модели. Данная модель относится к классу [ансамблевых моделей](#) <sup>57</sup>.

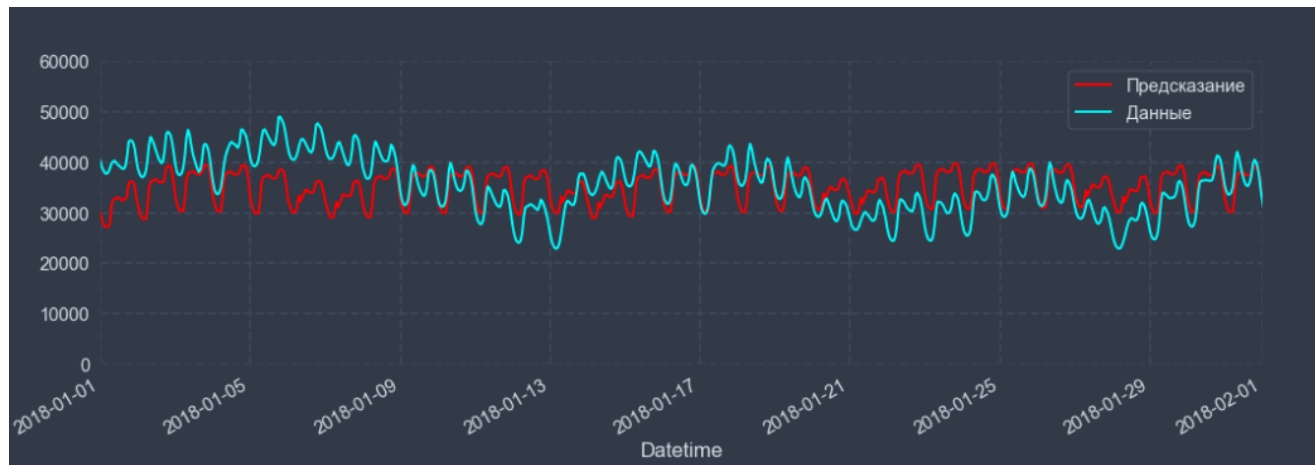


Рисунок 54. Предсказание экстремального градиентного бустинга и тестовые данные для проверки предсказания

## Параметры модели

### количество деревьев (n\_estimators)

Количество градиентных деревьев. Диапазон значений от **1** до **10000**, по умолчанию **1**.

### глубина дерева (max\_depth)

Параметр максимальной глубины дерева. Увеличение этого значения сделает модель более сложной и более склонной к переобучению. Диапазон значений от **1** до **100**, по умолчанию **6**.

**коэффициент сходимости (learning\_rate)**

Параметр скорости сходимости метода оптимизации. Диапазон значений от **0.001** до **0.1**, по умолчанию **0.03**.

**минимальная сумма веса (min\_child\_weight)**

Минимальная сумма веса экземпляра (гессииана), необходимая ветке дерева. Диапазон значений от **1.0** до **100.0**, по умолчанию **2.5**.

**метод преобразования данных (split\_method)**

Способ преобразования данных. Возможное значение **simple\_split**.

**Рекомендации по выбору параметров**

- ❖ В случае возникновения сложностей при подборе параметров, необходимо вернуться к значениям по умолчанию. Наиболее сильное влияние на предсказание оказывают следующие параметры: **количество деревьев (n\_estimators)**, **глубина дерева (max\_depth)** и **коэффициент сходимости (learning\_rate)**.
- ❖ Значения параметров **количество деревьев (n\_estimators)** и **глубина дерева (max\_depth)** можно выбирать любые из допустимого диапазона. Однако, увеличение разряда значений увеличивает время работы модели.
- ❖ Рекомендуемое значение параметра **коэффициент сходимости (learning\_rate)** – **0.1**. Уменьшение этого параметра ведет к увеличению времени сходимости метода оптимизации.
- ❖ Рекомендуемые значения параметра **минимальная сумма веса (min\_child\_weight)**: от **1** до **10**.
- ❖ Параметр **метод преобразования данных (split\_method)** относится к предварительной обработке данных. В настоящий момент может быть использован только метод **simple\_split**.

**6.3.4. Модификаторы**

Модификаторы преобразовывают временной ряд по предусмотренным правилам: согласно выбранным параметрам и заданному порядку.

Доступны следующие модификаторы:

- ❖ [Удаление интервалов](#)<sup>[71]</sup>;
- ❖ [Удаление шипов](#)<sup>[72]</sup>;
- ❖ [Регуляризация временного ряда](#)<sup>[82]</sup>.

**6.3.4.1. Удаление интервалов**

Модификатор **Удаление интервалов** преобразовывает временной ряд, удаляя значения в заданных временных интервалах. После преобразования значения будут считаться пропущенными (т.е. значениями `nan`).

**Пример использования модификатора**

На рисунке ниже приведена демонстрация режима учета регламентных работ. В рассматриваемом случае до 17.30 (заштрихованная зона) проводились регламентные работы. В качестве способа заполнения пропусков использовался метод фильтрации **ближайший**<sup>[104]</sup>. На этапе модификации данных для периода с начала временного ряда до окончания регламентных работ данным были присвоены значения `nan`, на этапе заполнения пропущенных данных в этих точках было выставлено ближайшее непустое значение (первое значение после 17.30). Таким образом, на графике для заштрихованной зоны регламентных работ можно наблюдать различия между реальными данными (зеленая линия) и данными, использованными для обучения модели (синяя линия).

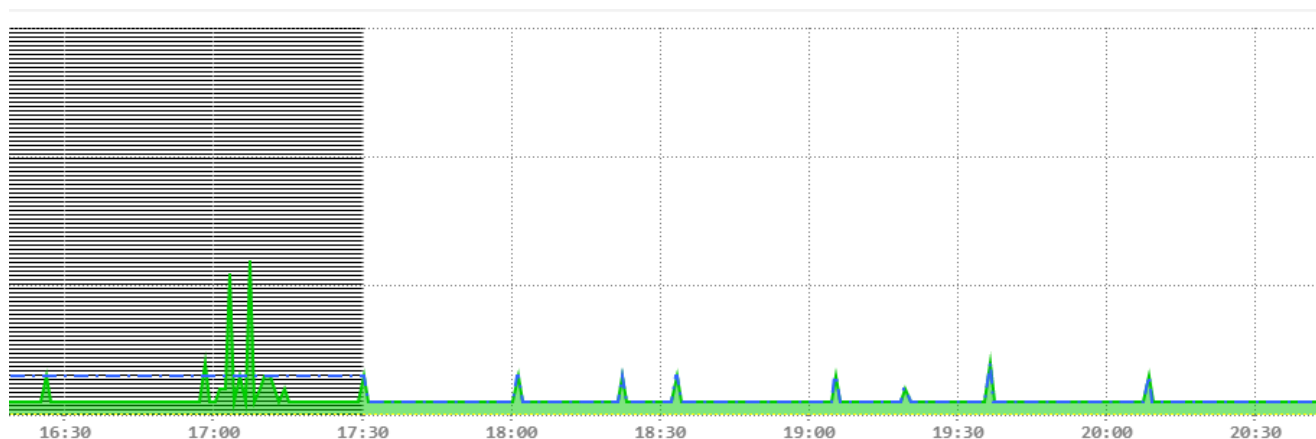


Рисунок 55. Пример преобразования временного ряда с модификатором Удаление интервалов

## Параметры модификатора

### временные интервалы (time\_intervals)

Временные интервалы, которые требуется удалить из временного ряда. Возможные значения: матрица размерностью  $n \times 2$ .

### 6.3.4.2. Удаление шипов

Шипы (выбросы) – это значения, резко отличающиеся от остальных значений ряда. Выброс находится либо на границе интервалов допустимых значений показателя, либо за их пределами. Обычно выбросы являются результатом ошибки измерений или необычной природы входных данных.

Модификатор **Удаление шипов** находит во временном ряду аномальные значения (выбросы) и заменяет их значениями `nan`. Алгоритм распознавания шипов определяется режимом работы модификатора:

#### **стандартизованная оценка (z-score);**

Стандартизованная оценка – это мера относительного разброса наблюдаемого или измеренного значения, которая показывает сколько стандартных отклонений составляет его разброс относительно среднего значения. В данном режиме значения, находящиеся вне трёх стандартных отклонений вокруг среднего значения, считаются выбросами.

### Пример

Применение модификатора в режиме **стандартизованная оценка** и последующим заполнением `nan`-значений "средним" (метод фильтрации – **среднее**):



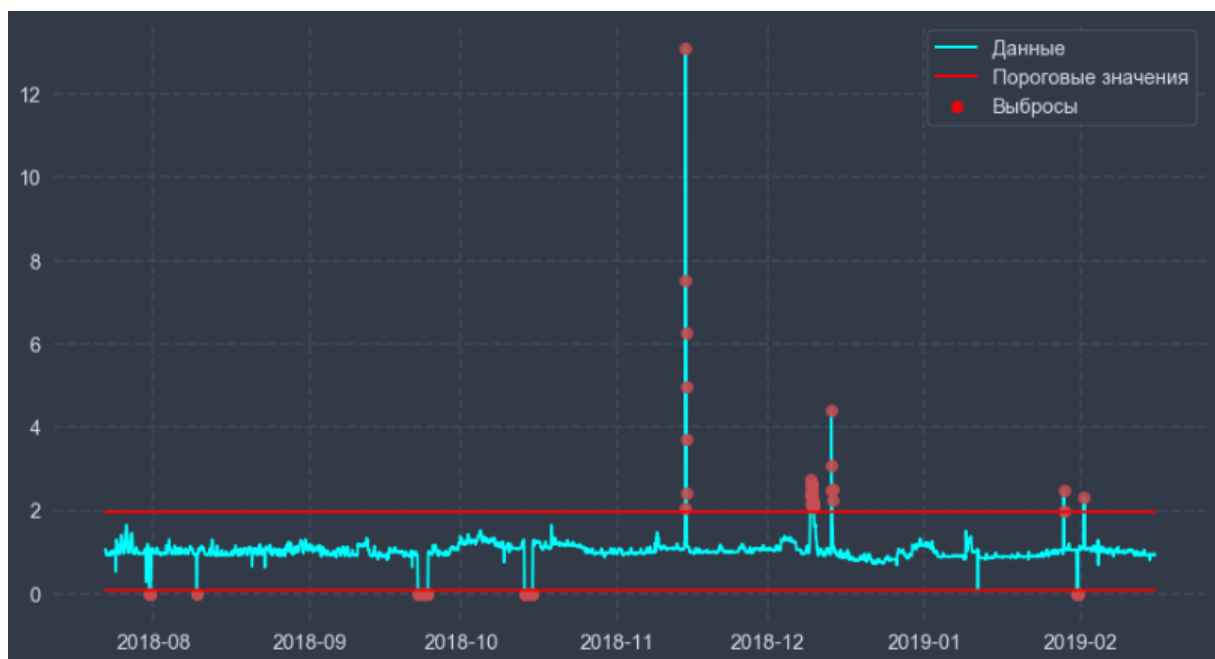


Рисунок 56. Исходный временной ряд

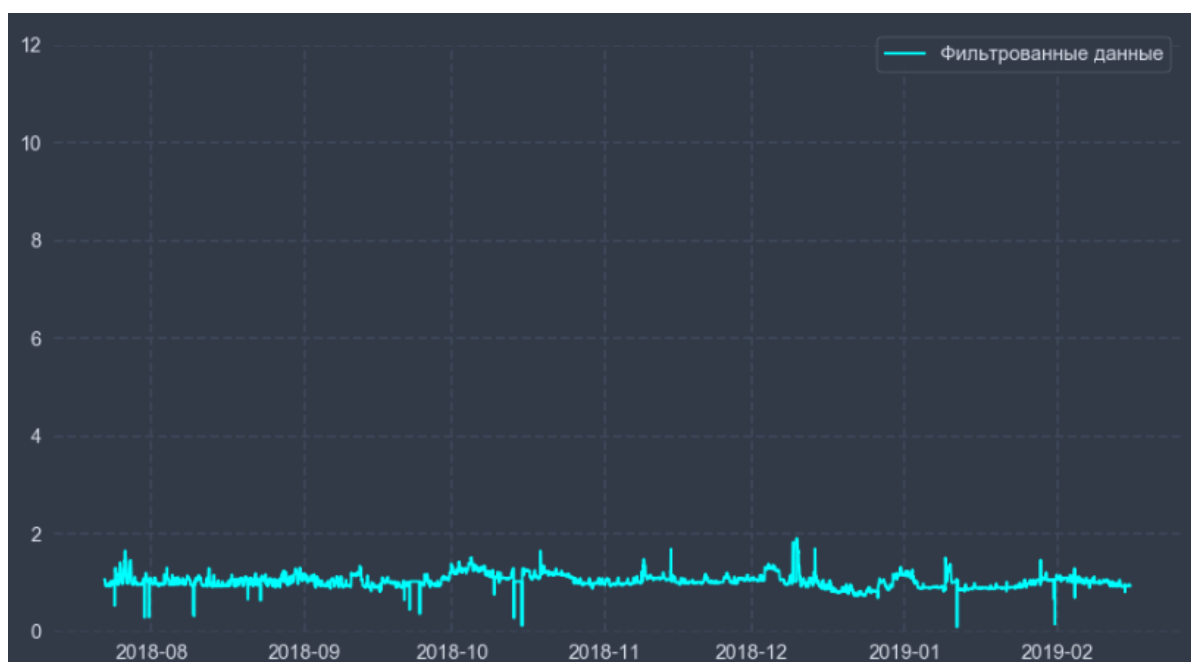


Рисунок 57. Временной ряд после преобразования

**межквартильный диапазон (iqr);**

В режиме **межквартильный диапазон** выбросами считаются значения, которые меньше ( $Q1 - 1.5 * IQR$ ) или больше ( $Q3 + 1.5 * IQR$ ), где:

$Q1$  – нижний (первый) квартиль, находящийся в 1/4 от начала распределения;

$Q3$  – верхний (третий) квартиль, находящийся в 3/4 от начала распределения;

$IQR = Q3 - Q1$  – расстояние между третьей и первой квартилями.

## Пример

Применение модификатора в режиме **межквартильный диапазон** и последующим заполнением пап-значений "средним" (метод фильтрации – **среднее**):

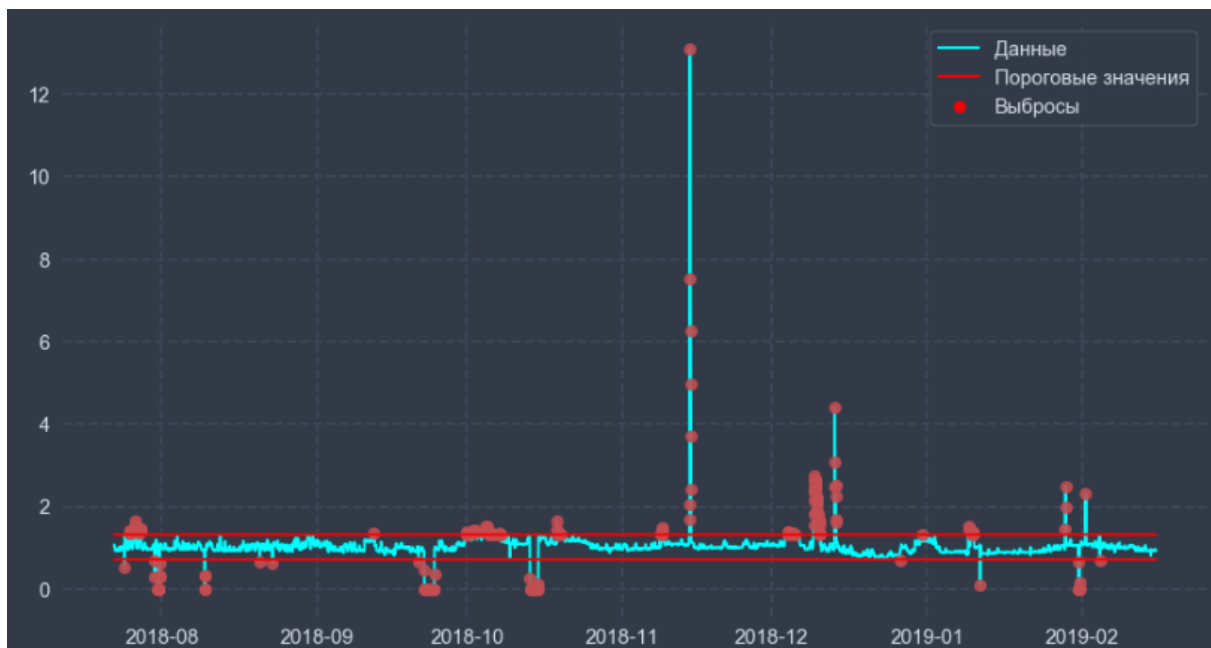


Рисунок 58. Исходных временной ряд

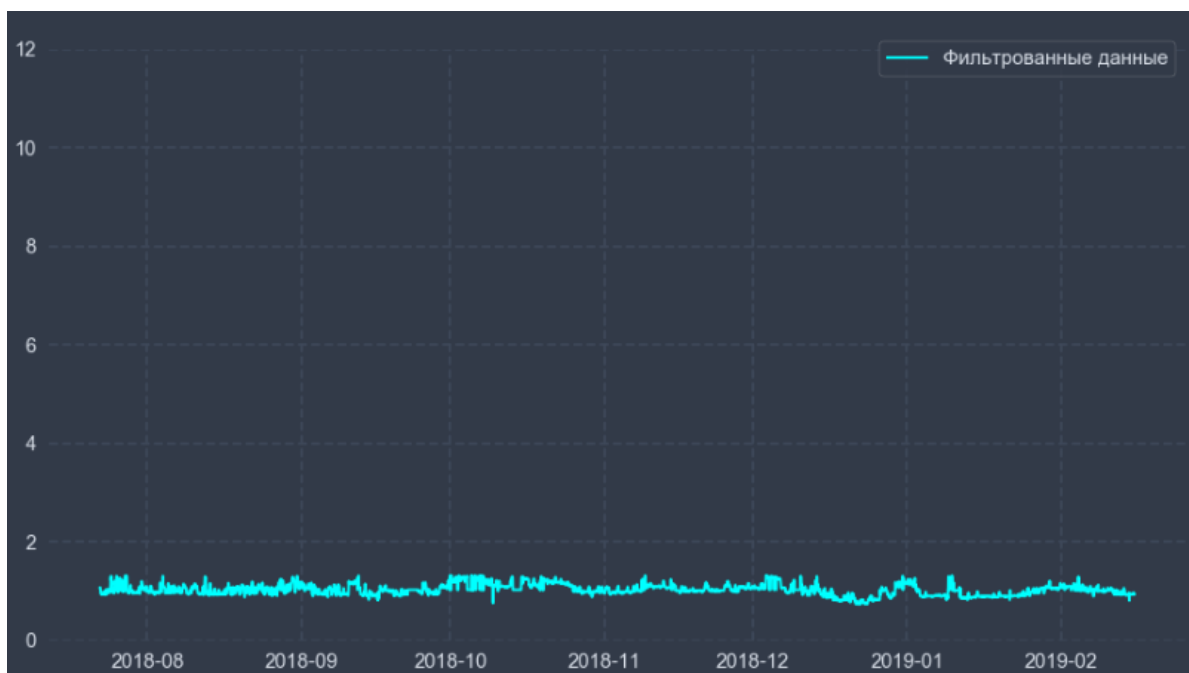


Рисунок 59. Временной ряд после преобразования

### скользящая медиана (rolling-median);

В данном режиме рассчитывается скользящая медиана (median) и скользящее стандартное отклонение (std). Значения, которые меньше ( $\text{median} - 3 * \text{std}$ ) или больше ( $\text{median} + 3 * \text{std}$ ), считаются выбросами.

## Пример

Применение модификатора в режиме **скользящая медиана** с параметром **размер окна = 10.0** (%) и последующим заполнением nan-значений "средним" (метод фильтрации – **среднее**):

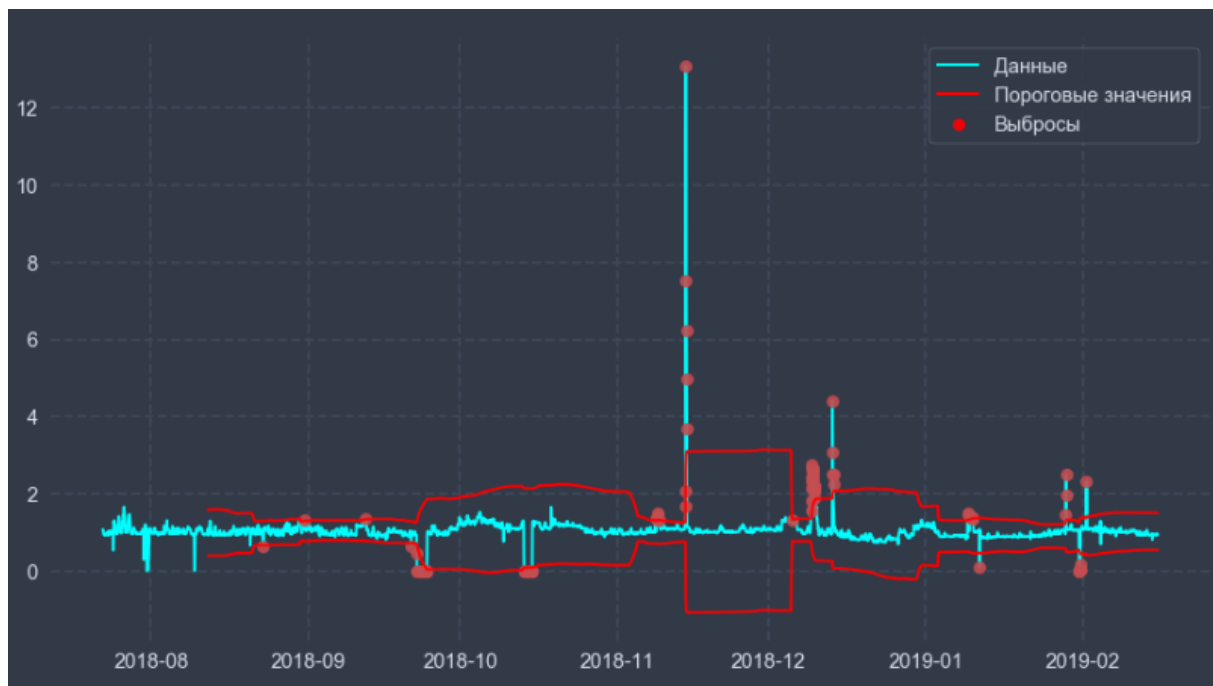


Рисунок 60. Исходный временной ряд

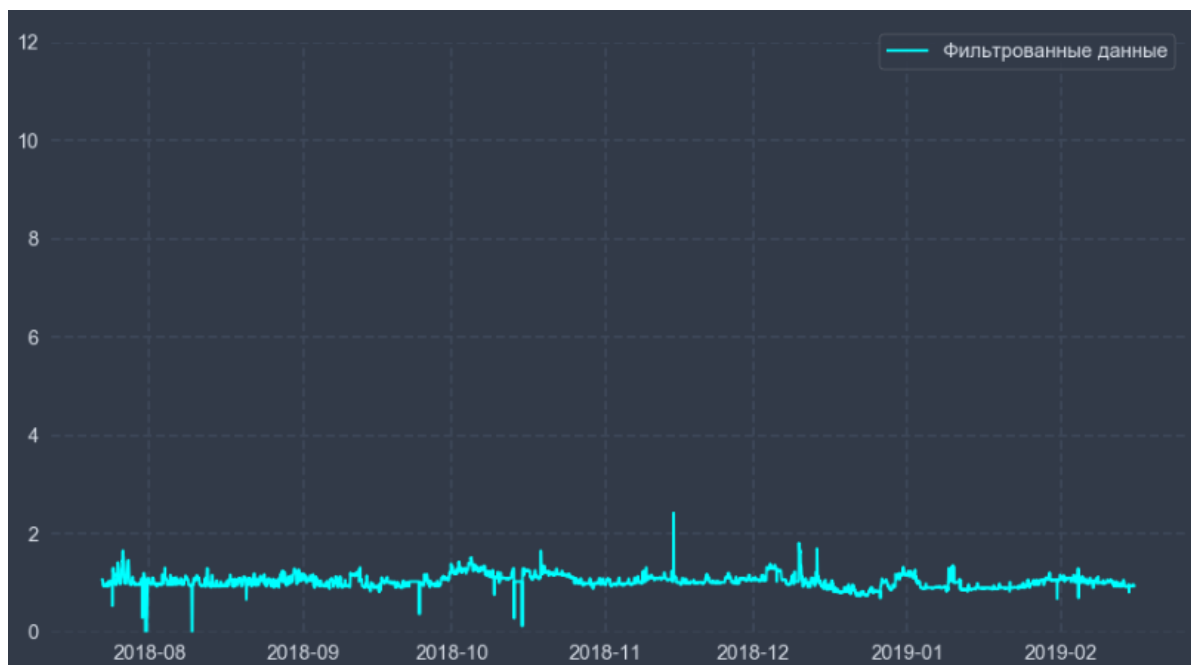


Рисунок 61. Временной ряд после преобразования

**фильтр Хампеля (hampel);**

В данном режиме используется абсолютное отклонение от медианы (Median Absolute Deviation, MAD), причем медиана рассчитывается в скользящем окне. Выбросом является значение, удовлетворяющее неравенству:

$$|y_i - median_i| > 3 * L * MAD_i, \text{ где}$$

$y$  – значения ряда;

$L$  – поправочный коэффициент.

Если размер окна (параметр **окно**) в этом режиме больше 50, то он устанавливается равным 50, поскольку при большем значении фильтрация не может быть выполнена.

### Пример

Преобразование временного ряда в режиме **фильтр Хампеля** с параметром **размер окна = 7.0** (%) и последующим заполнением nan-значений "средним" (метод фильтрации – **среднее**):

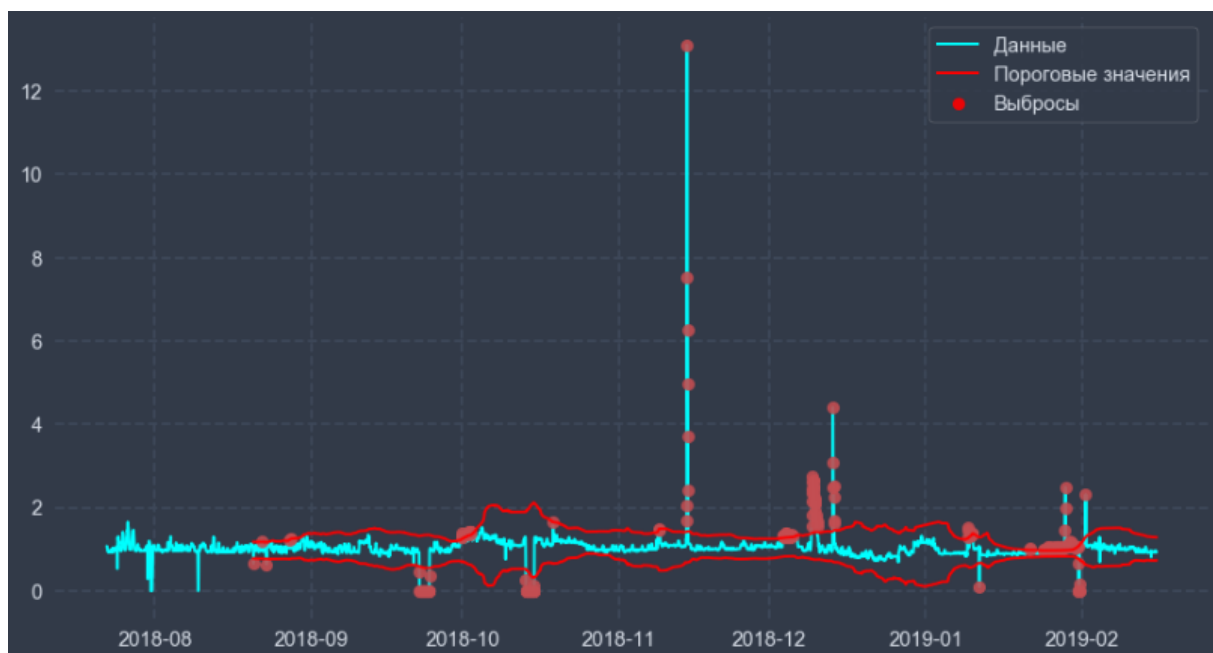


Рисунок 62. Исходный временной ряд

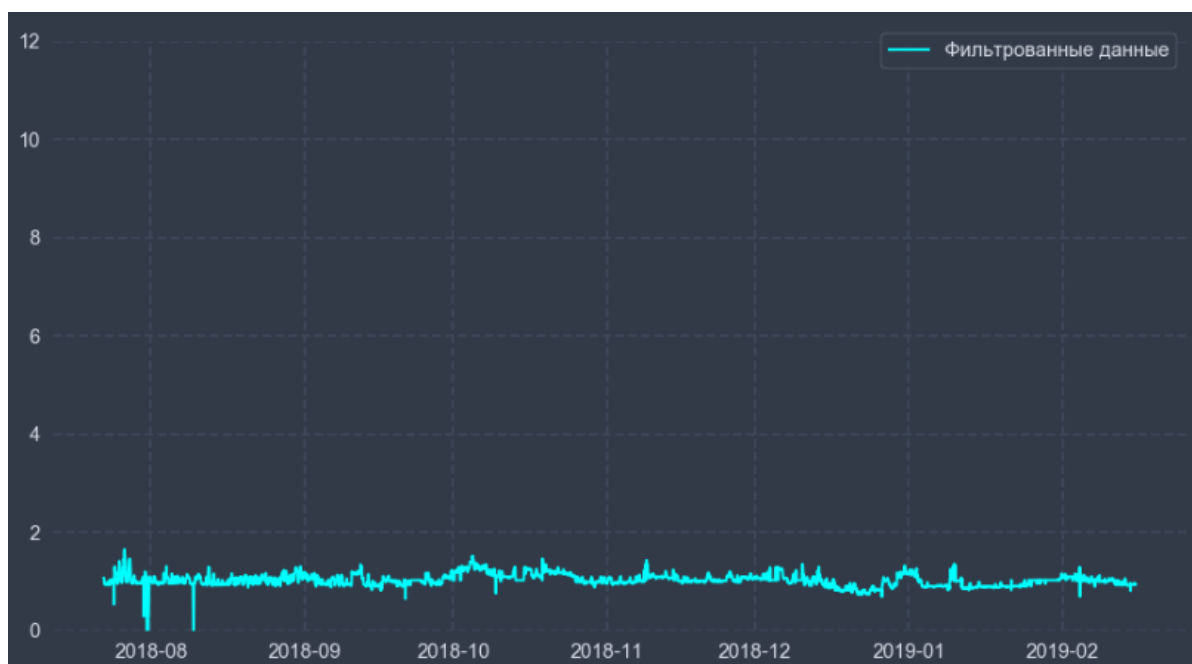


Рисунок 63. Временной ряд после преобразования

**фильтр БПФ (fft);**

В данном режиме обнаружение выбросов производится с помощью быстрого преобразования Фурье (БПФ).

В данном методе сначала определяется пороговое значение амплитуды частоты, с помощью которого можно будет определить наличие выброса в текущем окне. Для обнаружения выбросов в скользящем окне с помощью БПФ находятся значения сигнала в частотной области. Затем определяется максимальное по модулю значение из текущего окна сигнала. Если какое-либо из значений сигнала в частотной области больше порогового значения амплитуды частоты, значит, найденный максимальный по модулю элемент является выбросом. В отличие от режимов **Скользящая медиана** и **Фильтр Хампеля**, фильтрация применяется ко всему временному ряду.

Необходимо задать размер окна (параметр **окно**), рекомендуемое значение – не более **10%**. Если размер окна больше **50%**, выбросов будет найдено значительно меньше, чем в обратном случае.

**Пример**

Преобразование временного ряда для анализа выбросов в режиме **БПФ фильтр** с параметром **окно = 2.0 (%)** и последующим заполнением пап-значений "средним" (метод фильтрации – **среднее**):

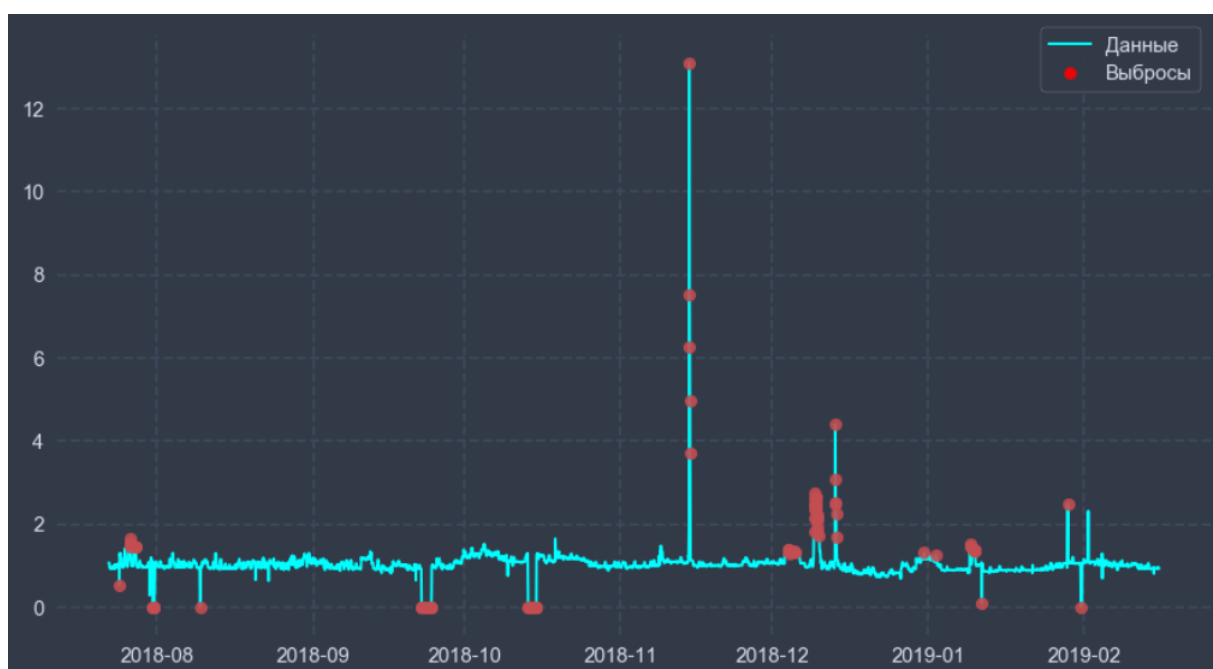


Рисунок 64. Исходный временной ряд

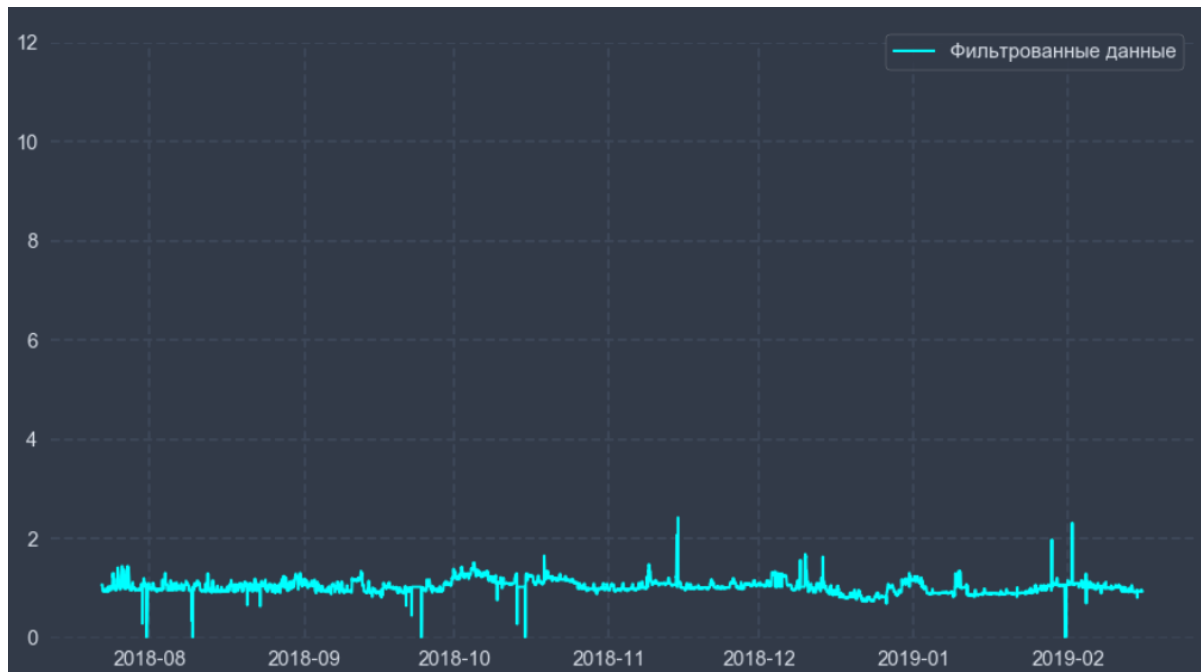


Рисунок 65. Временной ряд после преобразования

#### изолирующий лес (isolation-forest);

В данном режиме обнаружение выбросов производится с помощью изолирующего леса, основанного на принципе Монте-Карло. В данном методе изолированные точки отсекаются от нормальных, строится случайное бинарное решающее дерево. Если точка быстро изолировалась от других точек, то она будет находится в близких к корню листьях. Такие точки и будут считаться выбросами.

Рекомендуется указать предполагаемую **загрязненность** данных, т.е. долю выбросов в данных в процентах, по умолчанию **загрязненность** = **10.0** (%).

#### Пример

Преобразование временного ряда для анализа выбросов в режиме **Изолирующий лес** с параметром **загрязненность** = **5.0** (%) и последующим заполнением nan-значений "средним" (метод фильтрации – **среднее**):

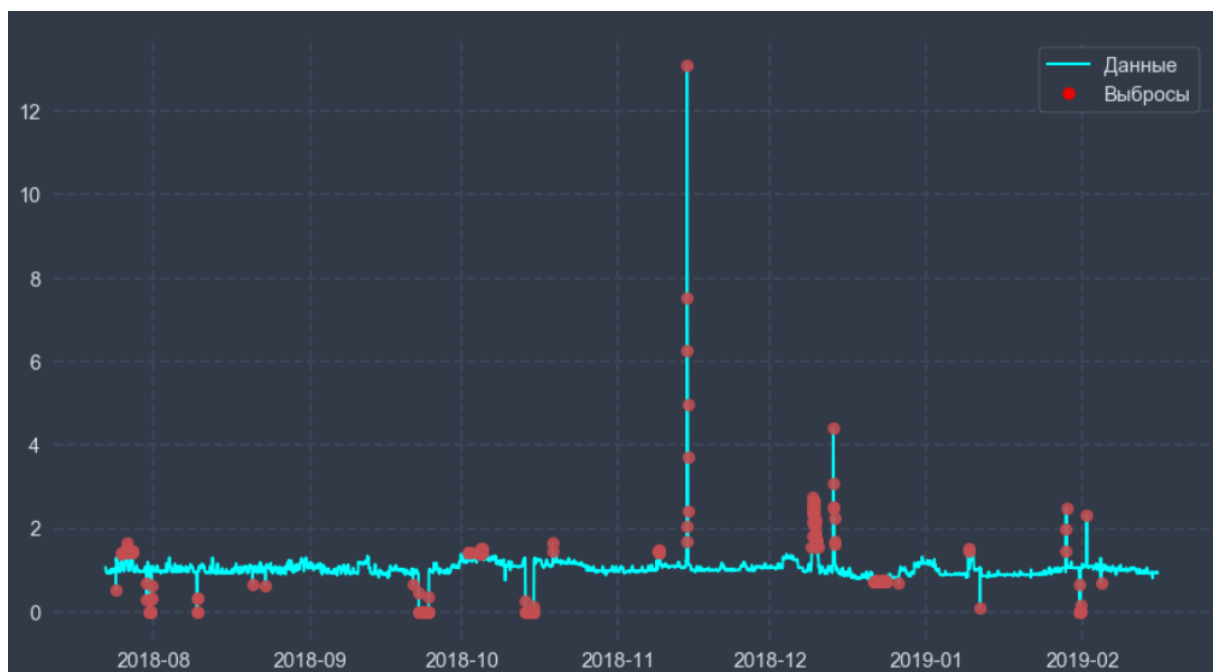


Рисунок 66. Исходный временной ряд

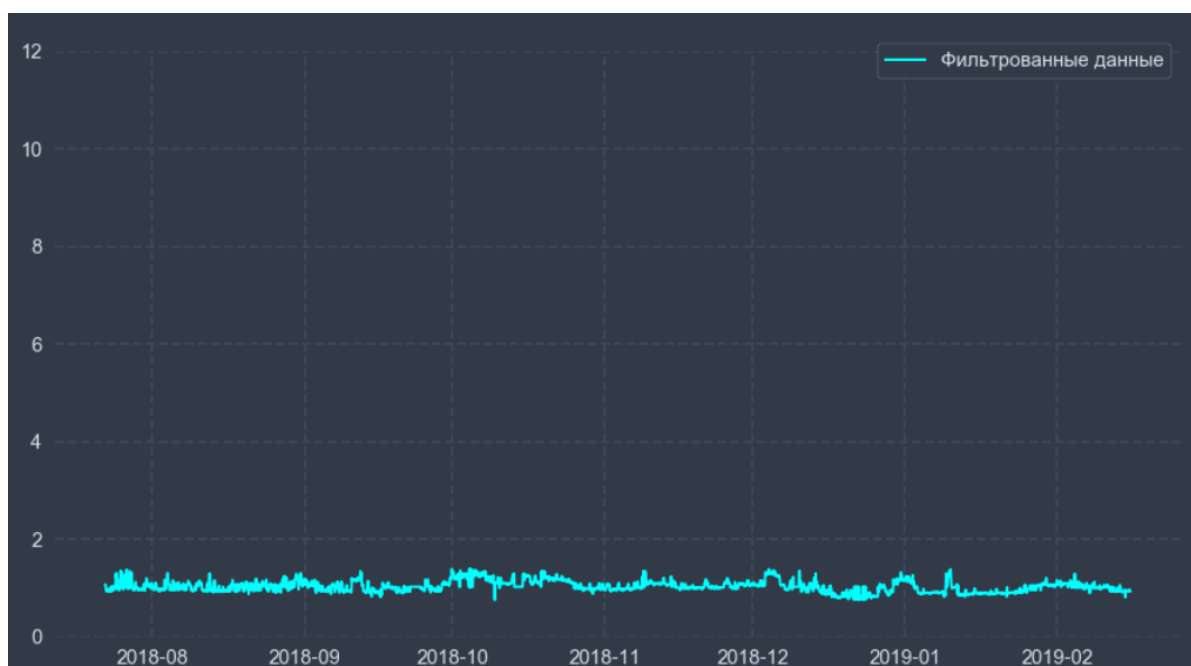


Рисунок 67. Временной ряд после преобразования

#### к-ближайших соседей (k-Nearest Neighbors).

В данном режиме для увеличения производительности временной ряд с количеством данных больше 2000 разбивается на подряды, для которых и выполняется алгоритм "к-ближайших соседей" с заданным числом  $k$ . Для оценки, является ли значение выбросом, используется расстояние до  $k$ -ых ближайших соседей. Расстояние определяется с помощью евклидовой метрики. Выброс – это точка, удаленная от соседних точек, поэтому наиболее изолированные значения будут считаться выбросами.

Рекомендуется указать предполагаемую **загрязненность** данных, т.е. долю выбросов в данных в процентах; по умолчанию **загрязненность** = **10.0** (%).

## Пример

Преобразование временного ряда для анализа выбросов в режиме **k ближайших соседей** с параметром **загрязненность** = **5.0** (%) и последующим заполнением nan-значений "средним" (метод фильтрации – **среднее**):

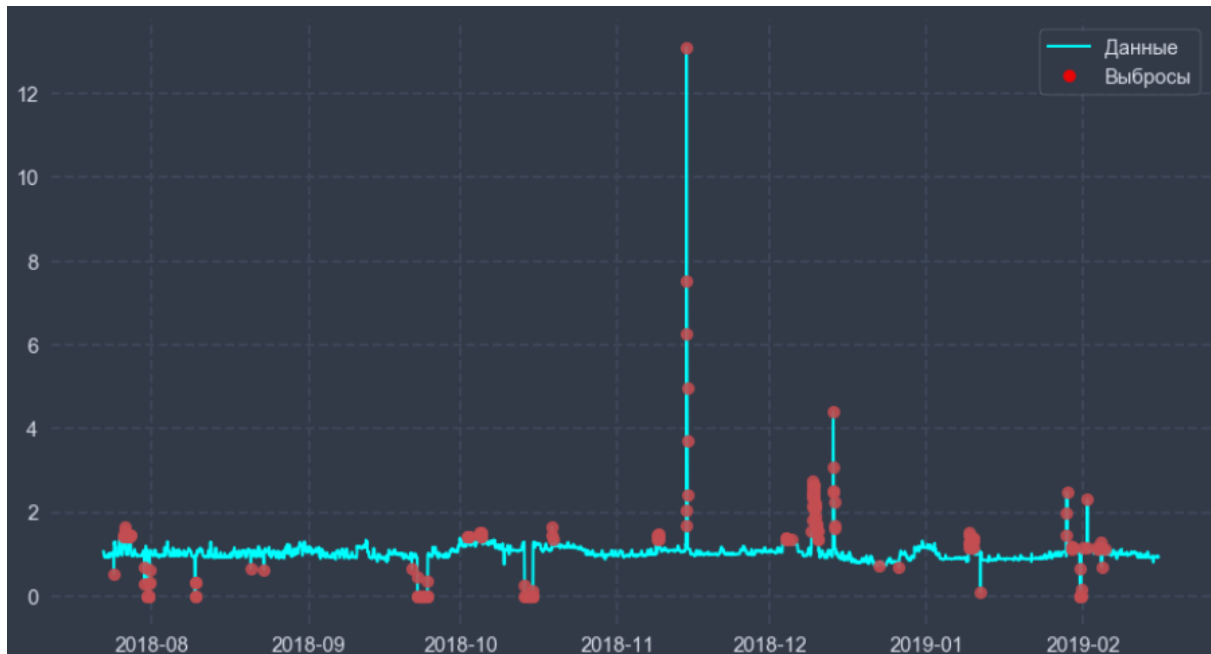


Рисунок 68. Исходный временной ряд

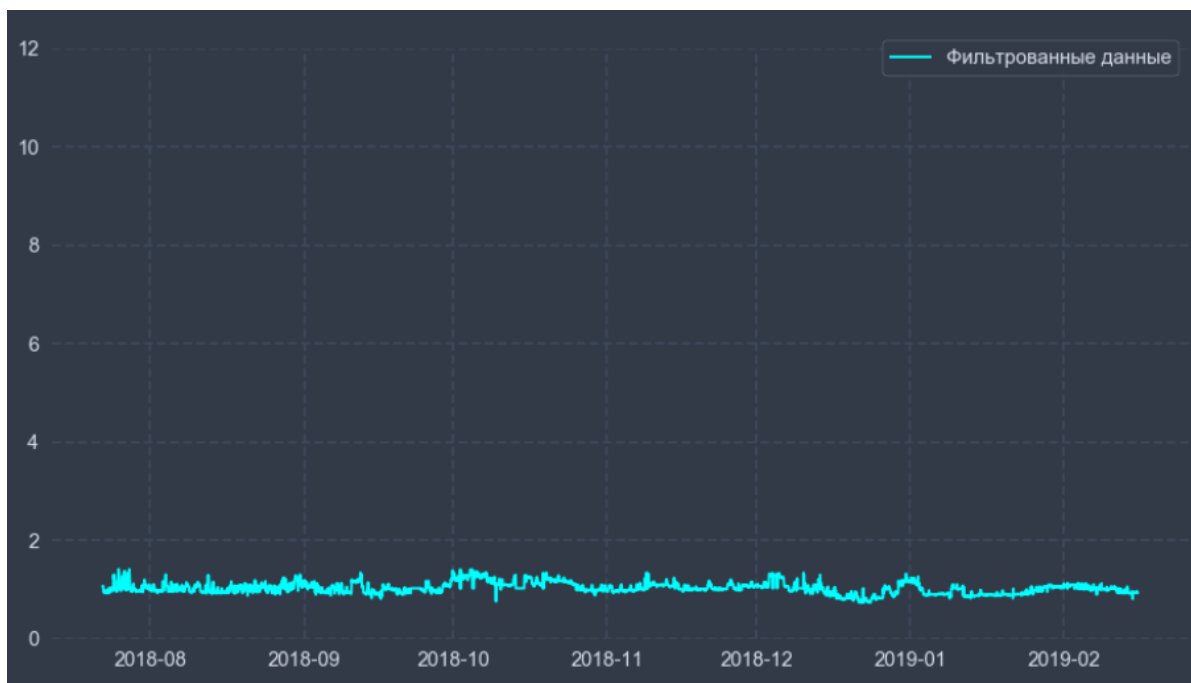


Рисунок 69. Временной ряд после преобразования

Для каждого режима администратор может указать минимальное и максимальное допустимое значение показателя. Значение показателя вне этого интервала считается аномальным (выбросом), а график "обрезается" ниже минимума и выше максимума.



## Параметры модификатора

### режим (mode)

Режим анализа выбросов. Доступные варианты:

[стандартизованная оценка](#)<sup>[72]</sup> (**z-score**) – по умолчанию;

[межквартильный диапазон](#)<sup>[72]</sup> (**iqr**);

[скользящая медиана](#)<sup>[72]</sup> (**rolling-median**);

[фильтр Хампеля](#)<sup>[72]</sup> (**hampel**);

[фильтр БПФ](#)<sup>[72]</sup> (**fft**);

[изолирующий лес](#)<sup>[72]</sup> (**isolation-forest**);

[k-ближайших соседей](#)<sup>[72]</sup> (**knn**).

### минимум (min)

Минимальное допустимое значение фильтрации сигнала; значения, которые меньше минимума, считаются выбросами.

### максимум (max)

Максимальное допустимое значение фильтрации сигнала; значения, превышающие максимум, считаются выбросами.

### окно (window)

Доля данных (в процентах) для установки размера окна. Задаётся для режимов **скользящая медиана**, **фильтр Хампеля** и **фильтр БПФ** (в других режимах параметр игнорируется). Допустимый диапазон: от **0.1** до **100.0**; если для режима **фильтр Хампеля** значение параметра **окно** > 50, то размер окна устанавливается равным **50**; для режима **фильтр БПФ** рекомендуется устанавливать значение не более **10**.

### загрязненность (contamination)

Доля выбросов в данных в процентах. Задаётся для режимов **изолирующий лес** и **k-ближайших соседей** (в других режимах параметр игнорируется). Допустимый диапазон: от **0.0** до **50.0**, по умолчанию – **10.0**.

## Пример применения модификатора

Применение модификатора **Удаление шипов** (**минимум** = **0.2**, **максимум** = **2.0**) и последующим заполнением nan-значений "средним" (метод фильтрации – **среднее**):

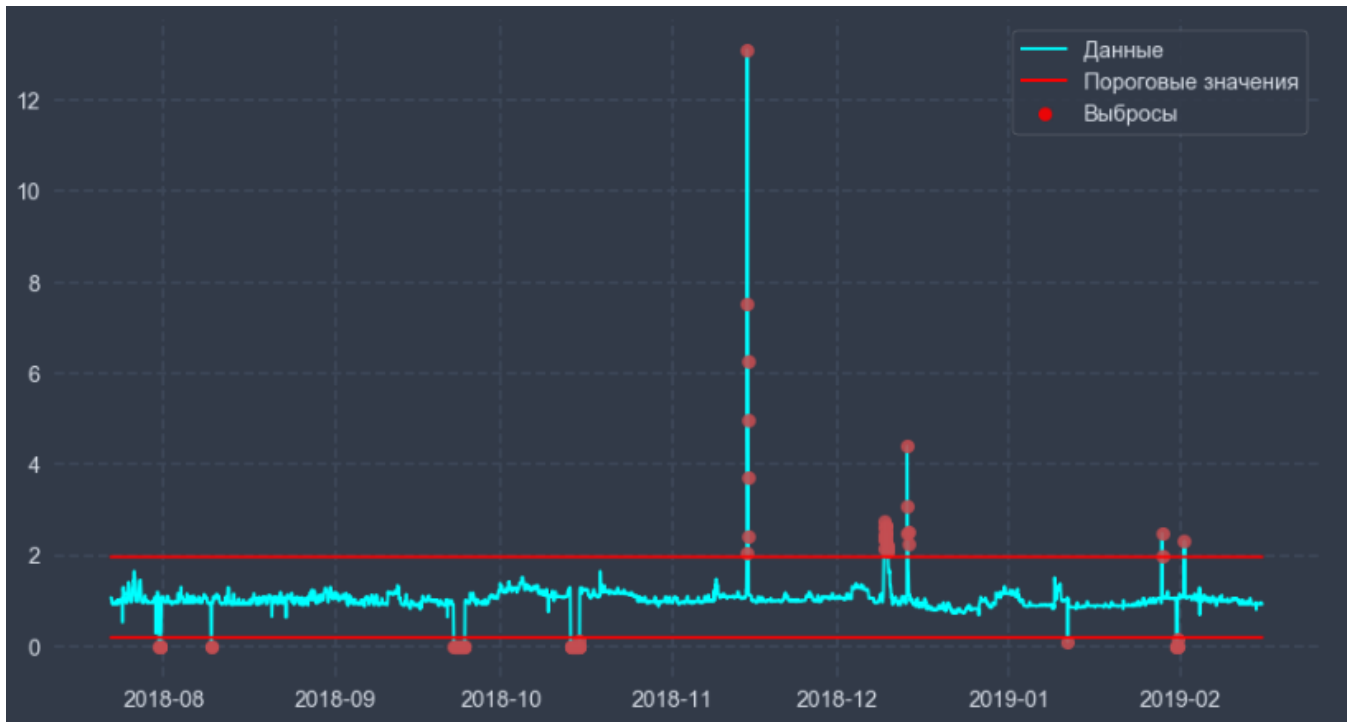


Рисунок 70. Исходный временной ряд

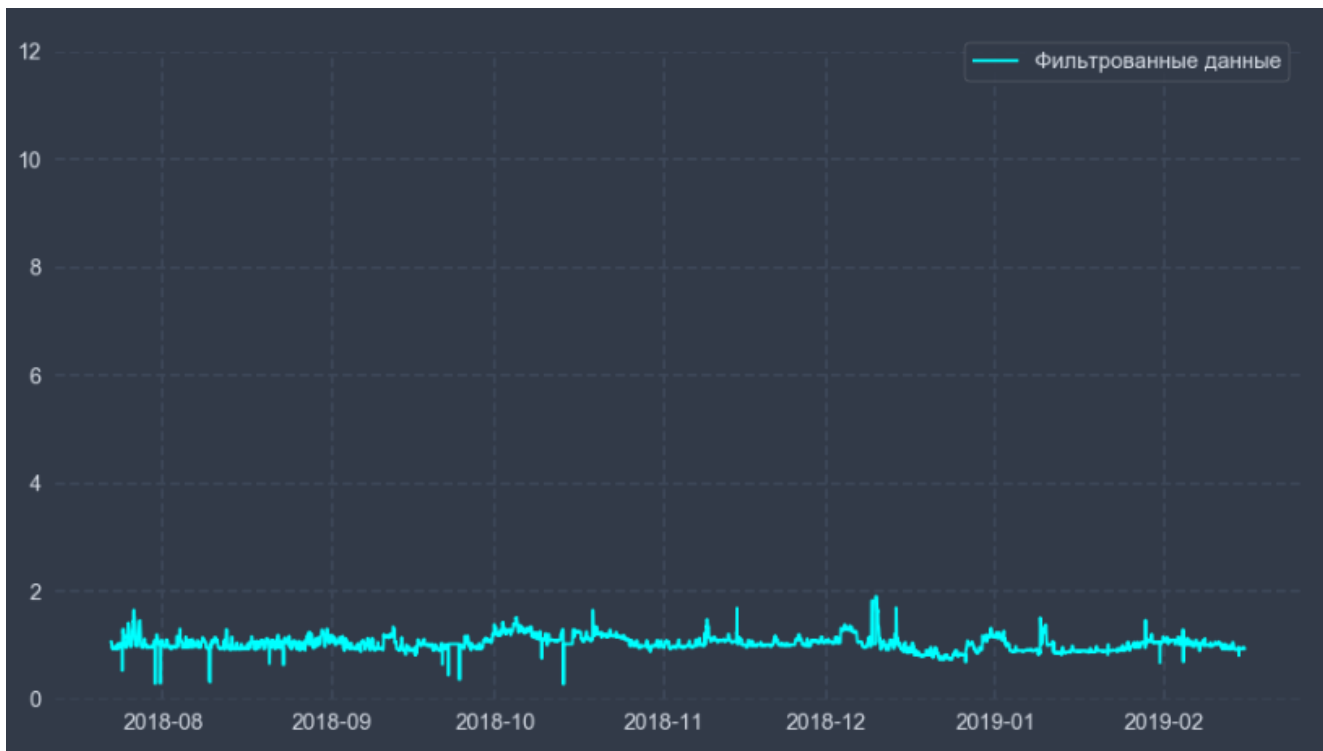


Рисунок 71. Временной ряд после преобразования

### 6.3.4.3. Регуляризация временного ряда

Модификатор **Регуляризация временного ряда** преобразовывает нерегулярный временной ряд (т.е. ряд с пропущенными значениями и моментами времени), в регулярный ряд, где каждому значению соответствует момент времени. При настройке администратор задаёт временной шаг.

## Параметры модификатора

### временной показатель (time\_ratio)

Единицы измерения временного шага: **second** (секунды), **minute** (минуты), **hour** (часы), **Day** (дни), **Week** (недели), **Month** (месяца), **Year** (годы).

### временной коэффициент (time\_coeff)

Длина временного шага. Диапазон допустимых значений: целые числа от **1**.

## 6.3.5. Нормализация

Нормализация представляет собой процесс предобработки данных с целью приведения их к одному виду и наоборот.

Реализованы следующие методы нормализации:

- ❖ [Масштабирование данных](#)<sup>[83]</sup>;
- ❖ [Демасштабирование по ядру](#)<sup>[84]</sup>;
- ❖ [Демасштабирование по минимуму и максимуму](#)<sup>[84]</sup>.

### 6.3.5.1. Масштабирование данных

Обработчик **Масштабирование данных** представляет собой метод, который переводит данные из интервала `[true_min, true_max]` в интервал `[left_bound, right_bound]`. Параметры **true\_min** и **true\_max** полезно задавать, если известны точные верхняя и нижняя границы данных наблюдения, тогда как на конкретном наборе данных минимальное и максимальное значения могут не встречаться. Если **true\_min** и **true\_max** не заданы, они вычисляются как максимальное и минимальное значения текущего набора данных. Параметры **variable\_for\_min** и **variable\_for\_scale** полезны для обратного масштабирования данных в интервал `[true_min, true_max]` после работы с ними. Сохраненные параметры **variable\_for\_min** и **variable\_for\_scale** также называются ядром масштабирования.

Новое значение данных  $x_{scaled}$  получается преобразованием старого  $x$  с помощью формул:

$$x_{std} = (x - true\_min) / (true\_max - true\_min)$$

$$x_{scaled} = x_{std} * (right\_bound - left\_bound) + left\_bound$$

## Параметры обработчика

### левая граница (left\_bound)

Левая граница интервала масштабирования. Формат данных – **float**. Значение по умолчанию – **0.0**.

### правая граница (right\_bound)

Правая граница интервала масштабирования. Формат данных – **float**. Значение по умолчанию – **1.0**.

### реальный минимум (true\_min)

Минимальное значение во всем наборе данных. Формат данных – **float**. Если значение не указано, оно вычисляется автоматически.

### реальный максимум (true\_max)

Максимальное значение во всем наборе данных. Формат данных – **float**. Если значение не указано, оно вычисляется автоматически.

### переменная для минимума (**variable\_for\_min**)

Переменная для сохранения минимума. Формат данных – **float**.

### переменная для масштаба (**variable\_for\_scale**)

Переменная для сохранения параметра масштаба. Формат данных – **float**.

## 6.3.5.2. Демасштабирование по ядру

Обработчик **Демасштабирование по ядру** представляет собой метод, который выполняет обратное масштабирование данных из интервала `[left_bound, right_bound]` в интервал `[true_min, true_max]` (см. п. [Масштабирование в интервал](#)<sup>[83]</sup>). Для обратного масштабирования используются параметры **variable\_for\_min** и **variable\_for\_scale**, сохраненные при прямом масштабировании.

### Параметры обработчика

#### минимум (**min**)

Параметр масштабирования "минимум". Формат данных – **float**.

#### масштаб (**scale**)

Параметр масштабирования "масштаб". Формат данных – **float**.

## 6.3.5.3. Демасштабирование по минимуму и максимуму

Обработчик **Демасштабирование по минимуму и максимуму** представляет собой метод, который выполняет обратное масштабирование данных из интервала `[left_bound, right_bound]` в интервал `[true_min, true_max]` (см. п. [Масштабирование в интервал](#)<sup>[83]</sup>). Для обратного масштабирования используются параметры **true\_min** и **true\_max**, показывающие, в какой интервал нужно обратно масштабировать данные, и параметры **left\_bound** и **right\_bound**, показывающие, в каком интервале находятся данные после прямого масштабирования.

### Параметры обработчика

#### реальный минимум (**true\_min**)

Минимальное реальное значение данных. Формат данных – **float**. Значение по умолчанию – **0.0**.

#### реальный максимум (**true\_max**)

Максимальное реальное значение данных. Формат данных – **float**. Значение по умолчанию – **255.0**.

#### левая граница (**left\_bound**)

Левая граница интервала масштабирования. Формат данных – **float**. Значение по умолчанию – **0.0**.

#### правая граница (**right\_bound**)

Правая граница интервала масштабирования. Формат данных – **float**. Значение по умолчанию – **1.0**.

## 6.3.6. Общие

Данный раздел включает описание следующих обработчиков:

- ❖ [Приведение к матрице](#)<sup>[85]</sup>;

- ❖ [Приведение к множественным временным рядам](#)<sup>[85]</sup>;
- ❖ [Приведение к регулярным временным рядам](#)<sup>[85]</sup>;
- ❖ [Вырезать данные](#)<sup>[85]</sup>;
- ❖ [Загрузить и выполнить модель](#)<sup>[86]</sup>;
- ❖ [Вставить данные](#)<sup>[87]</sup>;
- ❖ [Транспонирование](#)<sup>[87]</sup>.

### 6.3.6.1. Приведение к матрице

Обработчик **Приведение к матрице** представляет собой метод, который приводит данные типа `MultyTimeSeries` к типу `Matrix`. Необходим для работы сторонних моделей, требующих на вход данные типа `np.array`.

### 6.3.6.2. Приведение к множественным временным рядам

Обработчик **Приведение к множественным временным рядам** представляет собой метод, который приводит данные типа `Matrix` к типу `MultiTimeSeries`. Необходим для работы сторонних моделей для приведения выходных значений к типу, требуемому в **Редакторе математических моделей**.

#### Параметры обработчика

##### начальная временная метка (`start_timestamp`)

Временная метка, с которой будет начинаться временной ряд. Минимальное значение – **0**. Значение по умолчанию – **0**.

##### шаг (`step`)

Временной шаг между соседними точками данных. Минимальное значение – **0**. Значение по умолчанию – **60**.

### 6.3.6.3. Приведение к регулярным временным рядам

Обработчик **Приведение к регулярным временным рядам** представляет собой метод, который приводит данные типа `Vector` к типу `TimeSeries`. Необходим для работы сторонних моделей для приведения выходных значений к типу, требуемому в **Редакторе математических моделей**.

#### Параметры обработчика

##### начальная временная метка (`start_timestamp`)

Временная метка, с которой будет начинаться временной ряд. Минимальное значение – **0**.

##### шаг (`step`)

Временной шаг между соседними точками данных.

### 6.3.6.4. Вырезать данные

Обработчик **Вырезать данные** представляет собой метод, который вырезает часть данных из временного ряда.

## Параметры обработчика

### начальная временная метка (value)

Количество данных, которые нужно удалить. Минимальное значение – 0.

### край (edge)

Край, с которого нужно вырезать данные: **begin** (начало), **end** (конец).

### метод (method)

Вырезать по времени или по количеству: **duration**, **index**.

### вырезанные данные (cut\_data)

Переменная, в которую сохраняется вырезанная часть.

## 6.3.6.5. Загрузить и выполнить модель

Обработчик **Загрузить и выполнить модель** позволяет выгрузить из файла модель и записать ее в определенный тип данных, после чего произвести предсказание с помощью модели и данных, вошедших в обработчик.

Схема работы обработчика представлена на рисунке ниже:

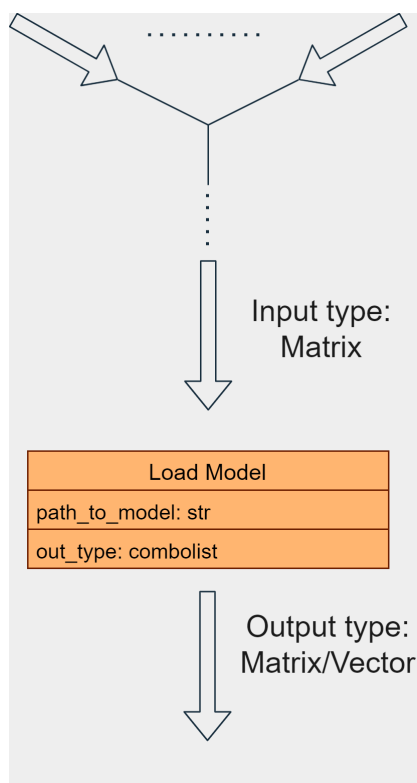


Рисунок 72. Схема работы обработчика **Загрузить и выполнить модель**

В результате работы обработчика модель, находящаяся в файле, путь к которому указывается в поле `path_to_file`, выгружается и записывается в определённый тип данных (возможные варианты: **tensorflow** и **torch**). После загрузки происходит предсказание с помощью данной модели и данных, вошедших в обработчик.

## Параметры обработчика

### выходной тип (out\_type)

Тип возвращаемого значения:

- ❖ **matrix** (по умолчанию);
- ❖ **vector**.

### путь до модели (path\_to\_model)

Путь до файла с моделью.

## 6.3.6.6. Вставить данные

Обработчик **Вставить данные** представляет собой метод, который вставляет данные во временной ряд.

## Параметры обработчика

### вставляемое (pasted)

Временной ряд, который требуется вставить в текущий.

### метод (method)

Край, с которого нужно вставить данные: **begin** (начало), **end** (конец).

## 6.3.6.7. Транспонирование

Обработчик **Транспонирование** представляет собой метод, который транспонирует входную матрицу данных (т.е. меняет местами столбцы и строки).

## 6.3.7. Оценщики

Оценщики – методы оценки качества математической модели, которые применяются при решении задачи прогнозирования временных рядов. Реализованы следующие методы:

- ❖ **Расстояние Кульбака-Лейблера (KL)** – является мерой расстояния между двумя вероятностными распределениями. Оценка обладает следующими свойствами:
  - KL>0, т.е. эта оценка является неотрицательной величиной;
  - KL=0 указывает на то, что оба рассматриваемых распределения идентичны.
- ❖ **Среднеквадратическая ошибка (MSE)** – вычисляет среднее из квадратов ошибок, т.е. средний квадрат разности значений, полученных моделью, и фактических значений. Среднеквадратическая ошибка помогает выбрать модель, которая дает меньше грубых ошибок прогноза. Оценка обладает следующими свойствами:
  - всегда неотрицательна;
  - чем ближе значение к нулю, тем модель лучше.
- ❖ **Среднеквадратическая ошибка для нормализованных данных (MSE NORM)** – работает аналогично методу **Среднеквадратическая ошибка (MSE)**, но перед проведением оценки происходит нормализация данных (более наглядна при сравнении в тех случаях, когда размерность данных сильно различается). Оценка обладает следующими свойствами:
  - всегда неотрицательна;
  - чем меньше, тем модель лучше.

- ❖ **Коэффициент детерминации ( $R^2$ )** – характеризует, насколько хорошо наблюдаемые результаты воспроизводятся моделью, на основе доли общего отклонения результатов, объясненной моделью. Оценка обладает следующими свойствами:
  - коэффициент детерминации для модели с константой принимает значения от 0 до 1. Чем ближе значение коэффициента к 1, тем сильнее зависимость;
  - в общем случае коэффициент детерминации может быть и отрицательным, это говорит о крайней неадекватности модели.
- ❖ **Средняя абсолютная процентная ошибка (MAPE)** – обычно используется как функция потерь для задач регрессии. Оценка показывает, какой процент составляет ошибка от фактических значений ряда. Оценка обладает следующими свойствами:
  - оценка всегда неотрицательна и не ограничена сверху;
  - из-за деления на фактические значения оценка чувствительна к масштабу; так, в случае если фактические значения близки к нулю, то значение оценки в разы увеличится;
  - оценка жёстче относится к случаям завышенных прогнозов, чем заниженных.
- ❖ **Средняя абсолютная ошибка (MAE)** – это среднее абсолютных ошибок. Показывает оценку расстояния от фактических до расчётных значений. Оценка обладает следующими свойствами:
  - равна нулю, только если все ошибки нулевые;
  - чем значение меньше, тем модель лучше.
- ❖ **Средняя абсолютная ошибка для нормализованных данных (MAE NORM)** – работает аналогично методу **Средняя абсолютная ошибка**, но перед проведением оценки происходит нормализация данных (более наглядна при сравнении в тех случаях, когда размерность данных сильно различается). Оценка обладает следующими свойствами:
  - равна нулю, только если все ошибки нулевые;
  - чем значение меньше, тем модель лучше.
- ❖ **Средняя процентная ошибка (MPE)** показывает процент смещения прогноза. Оценка обладает следующими свойствами:
  - положительные значения указывают на систематическое занижение прогноза, а отрицательные – на завышение;
  - оценка чувствительна к масштабу; так, в случае если фактические значения близки к нулю, то значение оценки в разы увеличится.
- ❖ **Максимальная корреляция** – определяет временной сдвиг сигналов, а также максимальное значение взаимной корреляции.

## Параметры оценщиков

Каждый из оценщиков имеет следующий параметр:

### **true\_data**

Реальные данные для сравнения с предсказанными данными.

## 6.3.8. Партицирование

Партицирование – это процесс разделения данных на части.

**Fit Part** – метод партицирования, который позволяет разделить (в процентах) данные, на которых будет происходить моделирование, остальные данные будут использоваться для оценки качества модели.

Метод разбивает набор данных на тренировочные и тестовые в отношении `value : (100 - value)`. Например, 40% данных будет использовано для обучения модели, 60% - для ее тестирования.



Тестовые данные всегда берутся с конца временного ряда и сохраняются в переменную **test\_data**, их продолжительность в секундах рассчитывается и записывается в переменную **test\_data\_duration**.

## Параметры метода Fit Part

### значение (value)

Процент от длины данных, который будет использоваться для обучения. Диапазон допустимых значений: от **0.0** до **100.0**, по умолчанию – **80.0**.

### тестовые данные (test\_data)

Переменная, в которую сохраняются тестовые данные

### продолжительность тестовых данных (test\_data\_duration)

Переменная, в которую сохранится продолжительность тестовых данных в секундах.

## 6.3.9. Постобработка

Постобработка данных – это процесс, направленный на дополнительный анализ данных, полученных при помощи математической модели.

Реализованы следующие способы постобработки:

- ❖ [Оценка выбросов \(интегральный риск\)](#)<sup>[89]</sup>;
- ❖ [Преобразование по диапазону](#)<sup>[90]</sup>;
- ❖ [Преобразование знака](#)<sup>[90]</sup>;
- ❖ [Сглаживание](#)<sup>[92]</sup>;
- ❖ [Расчет порогов](#)<sup>[93]</sup>.

### 6.3.9.1. Оценка выбросов (интегральный риск)

Способ постобработки **Оценка выбросов (интегральный риск)** вычисляет параметры, необходимые для построения границ интегрального риска. Параметры вычисляются путем построения суперпозиции гауссова распределения. Желтая (менее строгая) граница строится на распределении данных, предварительно очищенных от выбросов. Красная (критическая) граница строится на данных, выявленных как выбросы на этапе очистки данных для построения желтой границы.

## Параметры обработчика

### параметр для желтой границы (param\_y)

Приблизительное значение желтой границы в долях от общего набора данных (после очищения данных). Диапазон допустимых значений: от **0.85** до **0.99**. Значение по умолчанию – **0.93**.

### параметр для красной границы (param\_r)

Приблизительное значение красной границы в долях от общего набора данных. Диапазон допустимых значений: от **0.85** до **0.99**. Значение по умолчанию – **0.93**.

### предельное значение для желтой границы (limit\_y)

Максимальная доля данных, входящих в желтую границу (после очищения данных). Диапазон допустимых значений: от **0.85** до **0.99**. Значение по умолчанию – **0.98**.

### предельное значение для красной границы (limit\_r)

Максимальная доля данных, входящих в красную границу. Диапазон допустимых значений: от **0.85** до **0.99**. Значение по умолчанию – **0.97**.

### 6.3.9.2. Преобразование по диапазону

Способ постобработки **Преобразование по диапазону** заменяет все значения данных, которые ниже параметра **min** и выше параметра **max**, на значения параметров **min** и **max** соответственно.

#### Параметры обработчика

##### нижняя граница (min)

Минимальное значение для ранжирования сигнала. Тип данных – **float**.

##### верхняя граница (max)

Максимальное значение для ранжирования сигнала. Тип данных – **float**.

### 6.3.9.3. Преобразование знака

Способ постобработки **Преобразование знака** используется, если заведомо известно, что значение исследуемого параметра не может быть отрицательным или положительным.

После применения знакового преобразования линия предсказания и пороги имеют неотрицательные или неположительные значения, в зависимости от выбранного администратором режима (параметр **mode**):

- ❖ **неотрицательный** – отрицательные значения заменяются нулевыми;
- ❖ **неположительный** – положительные значения заменяются нулевыми.

Например, график предсказанных значений (голубая линия) и рассчитанных границ в режиме реального времени выглядит следующим образом:

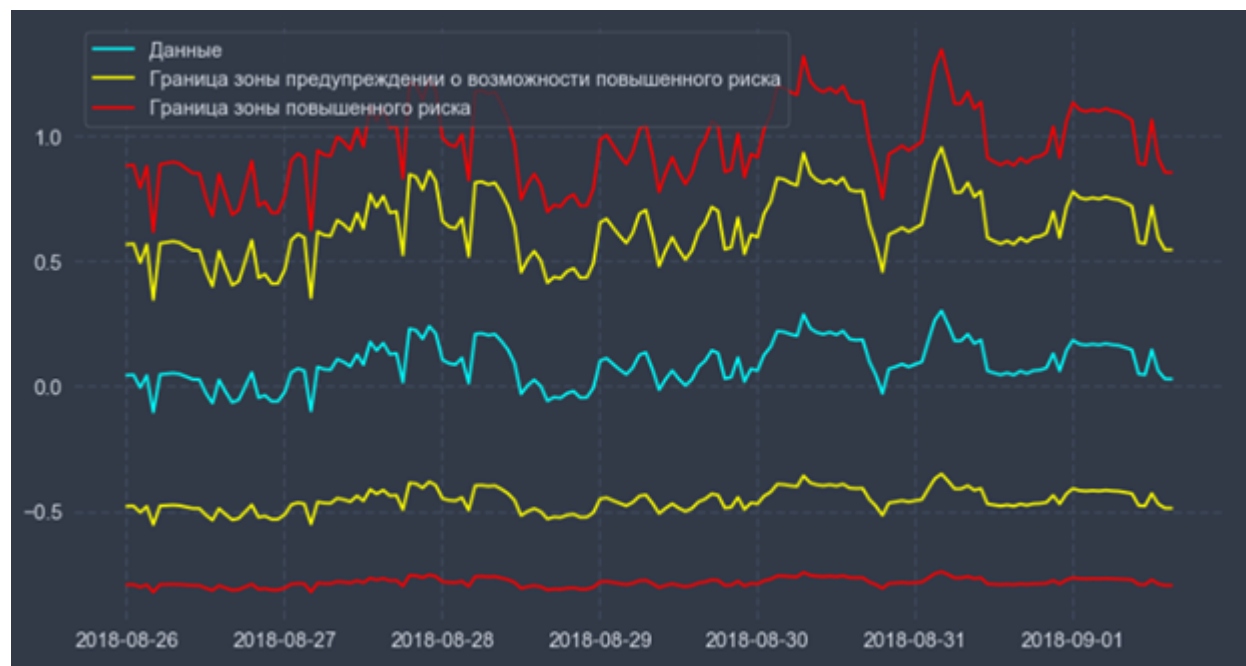


Рисунок 73. График без применения знакового преобразования

График после применения знакового преобразования в режиме неотрицательных данных:

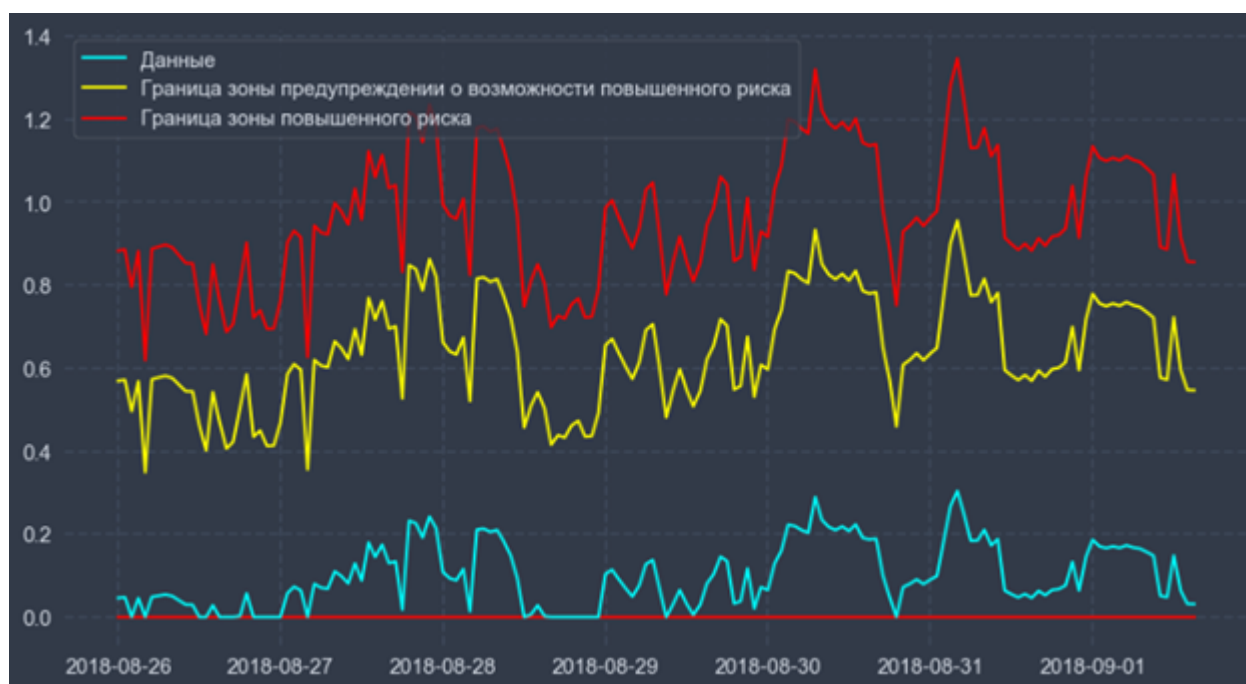


Рисунок 74. Результат применения знакового преобразования в режиме неотрицательных данных

График после применения знакового преобразования в режиме неположительных данных:

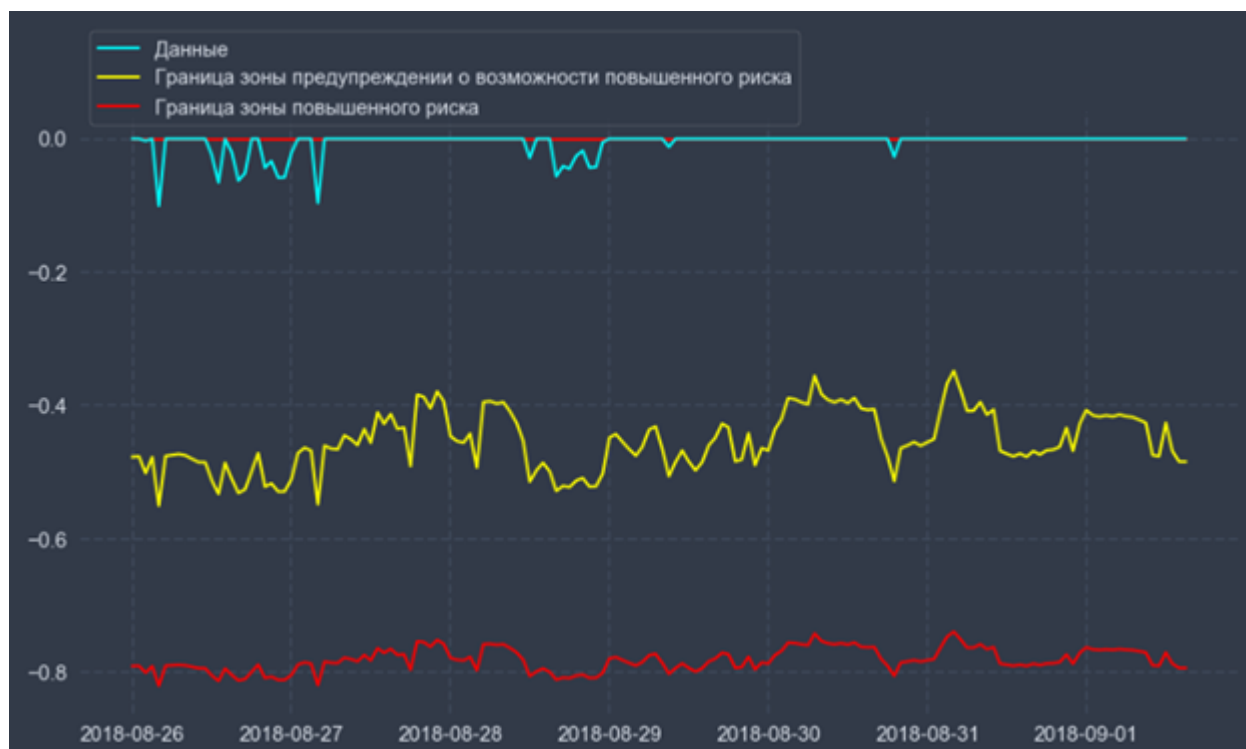


Рисунок 75. Результат применения знакового преобразования в режиме неположительных данных

## Параметры обработчика

режим (mode)

Режим для знакового преобразования: **non-negative** (неотрицательный) — по умолчанию, **non-positive** (неположительный).

### 6.3.9.4. Сглаживание

Способ постобработки **Сглаживание** основан на методе главных компонент и позволяет сделать менее резким поведение предсказанных значений и рассчитанных границ. Его следует применять до выполнения других способов постобработки предсказанных данных.

Например, график предсказанных значений и рассчитанных границ выглядит следующим образом:

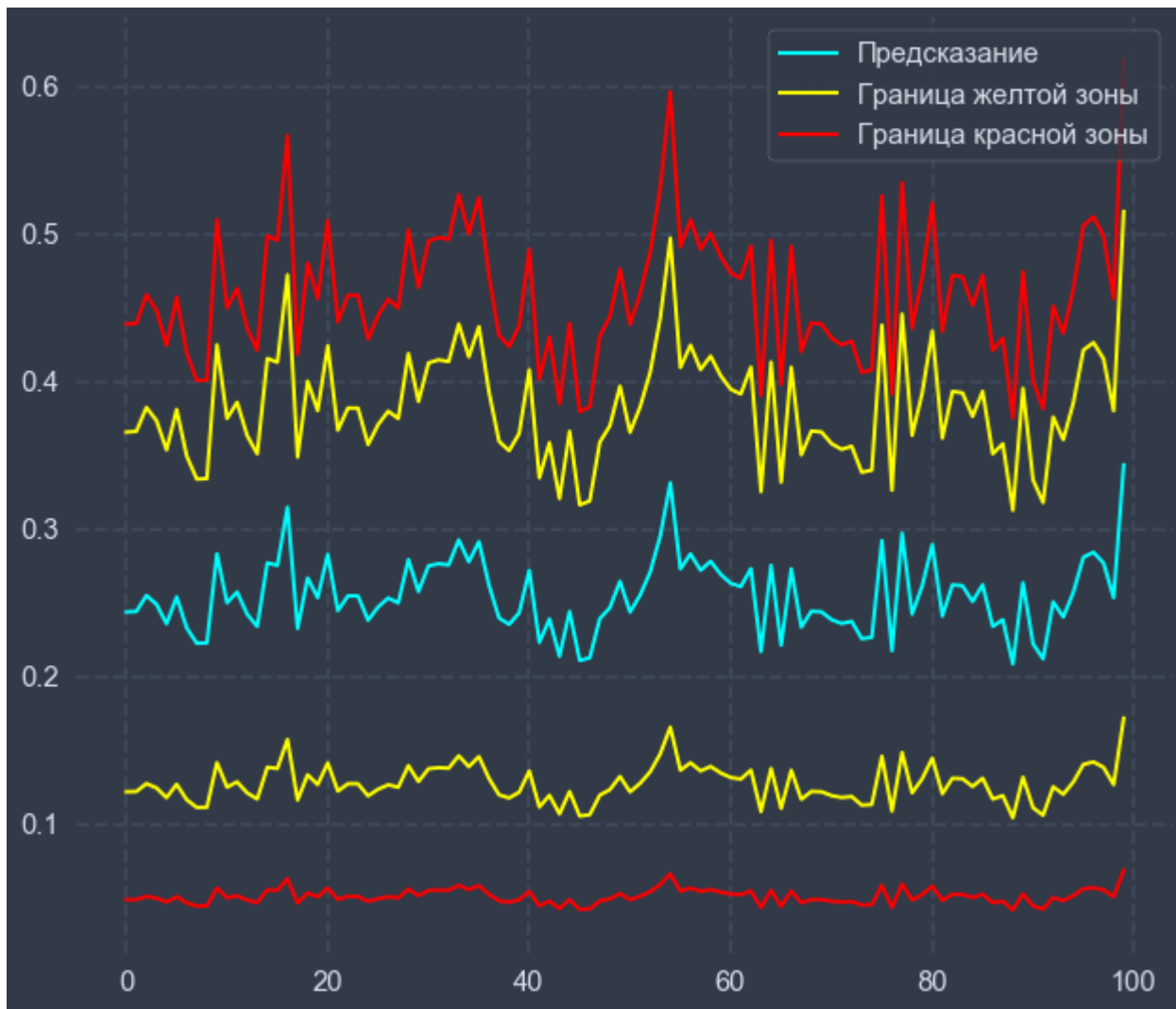


Рисунок 76. График без применения сглаживания

График после применения сглаживания:

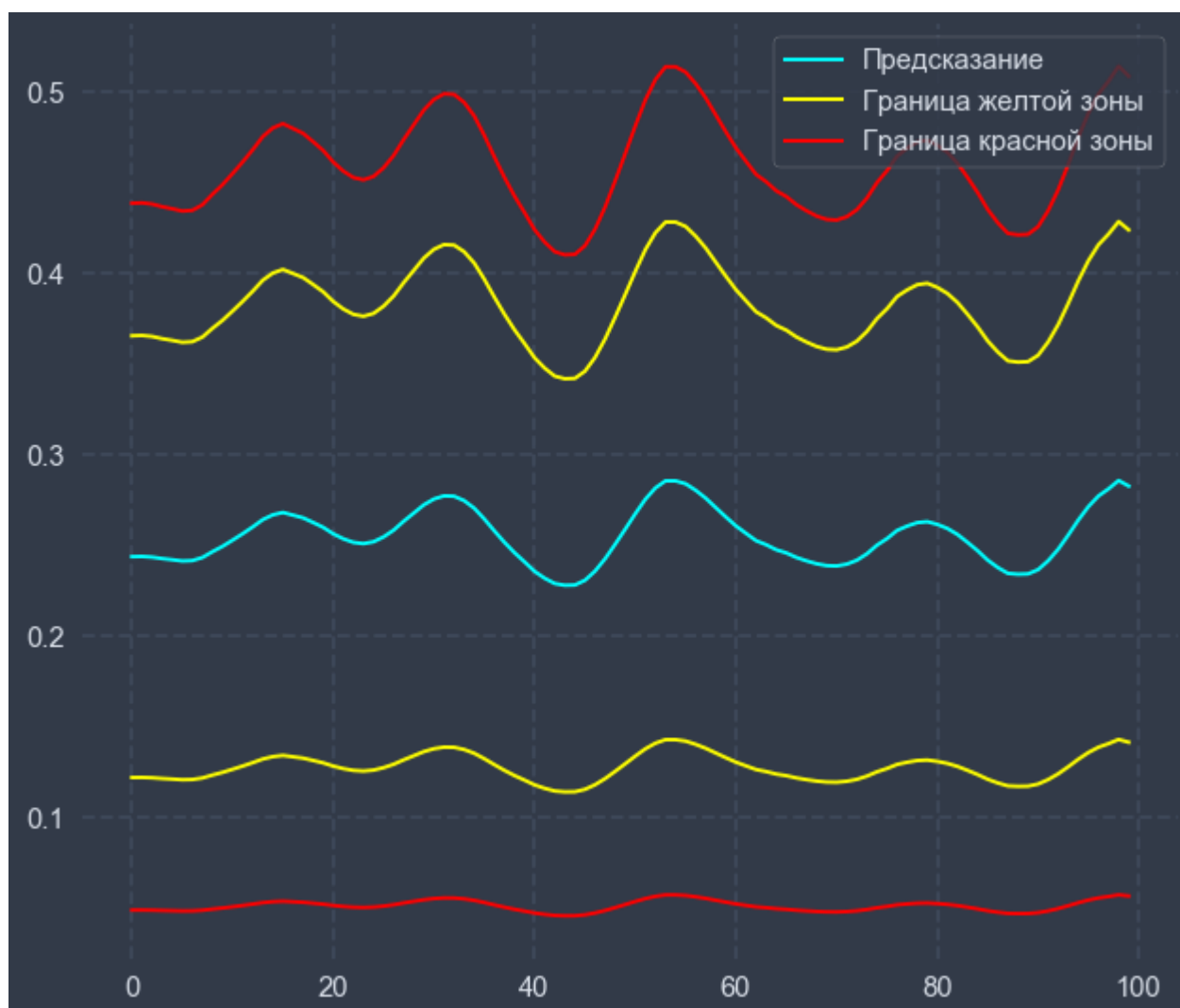


Рисунок 77. График после применения сглаживания

## Параметры обработчика

### горизонт (horizon)

Горизонт предсказания в секундах. Допустимые значения: целые числа от **1**. Значение по умолчанию – **259200**.

### 6.3.9.5. Расчёт порогов

Способ постобработки **Расчёт порогов** предусматривает вычисление вероятностных границ (порогов) возникновения событий.

Пороги ограничивают диапазоны значений исследуемого параметра. Диапазоны соответствуют различным уровням критичности:

- ❖ зелёный коридор – диапазон нормальных значений;
- ❖ жёлтый коридор – диапазон отклонения от нормы (зона повышенного риска);
- ❖ красный коридор – диапазон критических значений (зона высокого риска).

Обработчик **Расчёт порогов** применяется в сценариях для [прогнозирования](#)<sup>[46]</sup> поведения параметра целевого объекта. У таких сценариев выходными данными, как правило, являются предсказанное значение исследуемого параметра (тренд) и предсказанные значения порогов.

Пример сценария с обработчиком **Расчёт порогов**:

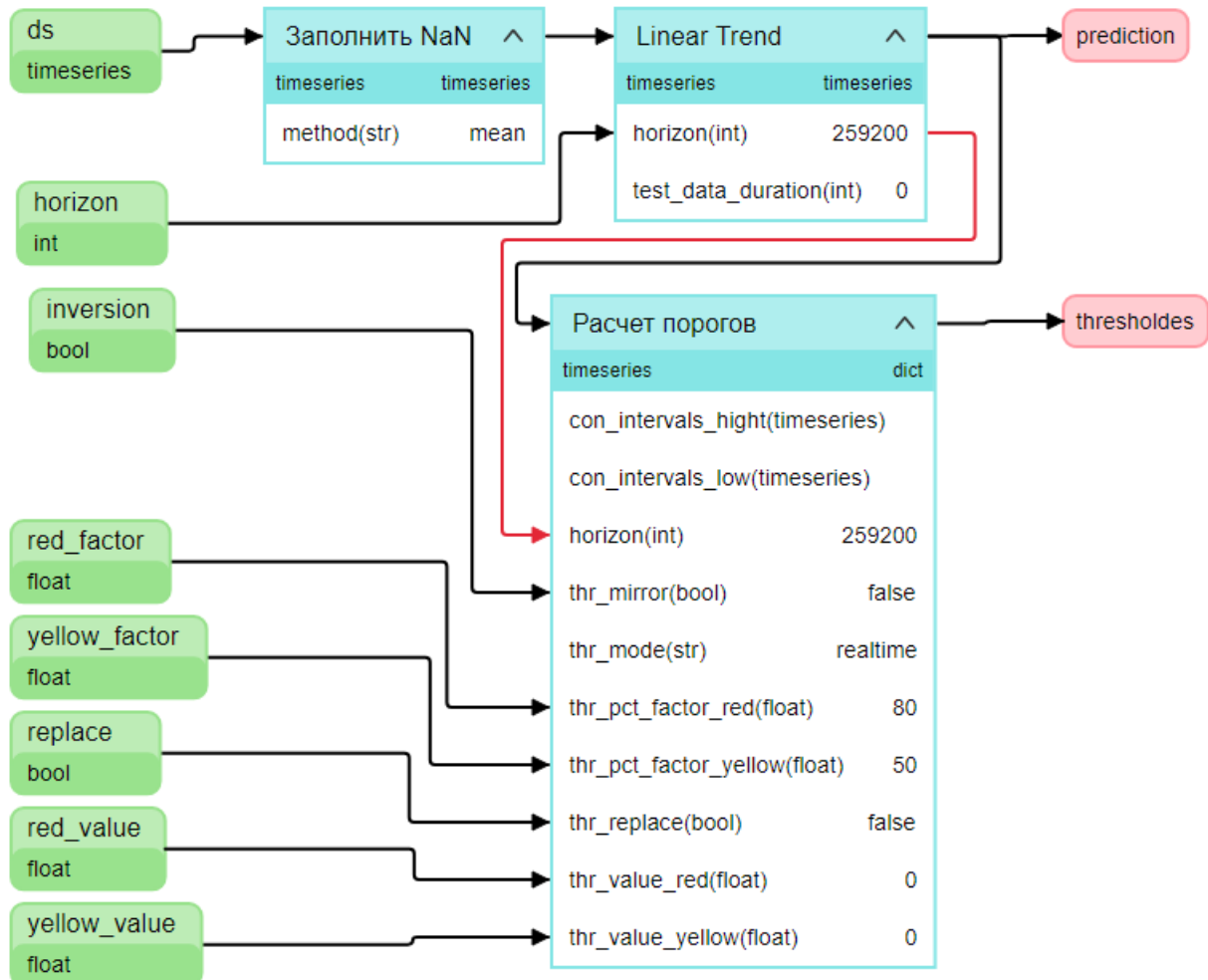


Рисунок 78. Пример сценария прогнозирования значения параметра и порогов

Если при настройке сигнала типа **MadaCalc**<sup>[46]</sup> администратор выбрал задачу на основе сценария с таким обработчиком, то алгоритм расчёта значения и статуса сигнала типа **MadaCalc** зависит от настроек обработчика.

## Параметры

### режим (thr\_mode)

Данный параметр определяет алгоритм вычисления порогов, а также алгоритм расчёта значения и статуса сигнала **MadaCalc**. Возможные режимы:

**realtime**<sup>[93]</sup> – режим реального времени;

**capacity planning**<sup>[96]</sup> – режим статических границ;

**time to event**<sup>[97]</sup> – режим расчета оставшегося времени до наступления события;

**deviation**<sup>[97]</sup> – режим отклонения от сигнала в процентах.

### процент смещения порога жёлтой зоны (thr\_pct\_factor\_yellow)

Для линейных моделей (линейный тренд, среднее, скользящее среднее) опорной точкой для расчета порогов является само предсказание, а для остальных моделей (ARIMA, Auto-ARIMA, ARM) – их доверительный интервал. Интервал допустимых значений: от **0.0** до **100.0**. Значение по умолчанию – **50.0**.

#### процент смещения порога красной зоны (thr\_pct\_factor\_red)

Для линейных моделей (линейный тренд, среднее, скользящее среднее) опорной точкой для расчета порогов является само предсказание, а для остальных моделей (ARIMA, Auto-ARIMA, ARM) – их доверительный интервал. Интервал допустимых значений: от **0.0** до **100.0**. Значение по умолчанию – **80.0**.

#### абсолютное значение сдвига порога желтой зоны (thr\_value\_yellow)

Для режима **realtime** – абсолютное значение сдвига порога жёлтого коридора **относительно предсказания** для линейных моделей (линейный тренд, среднее, скользящее среднее) или **относительно доверительного интервала** для остальных моделей.

Для остальных режимов – абсолютное значение порога жёлтого коридора.

#### абсолютное значение сдвига порога красной зоны (thr\_value\_red)

Для режима **realtime** – абсолютное значение сдвига порога красного коридора **относительно предсказания** для линейных моделей (линейный тренд, среднее, скользящее среднее) или **относительно доверительного интервала** для остальных моделей.

Для остальных режимов – абсолютное значение порога красного коридора.

#### отзеркалить (thr\_mirror)

Для режимов **статических границ** и **расчета оставшегося времени до наступления события**:

значение **false** – пересечение порога сверху будет вызывать срабатывание сигнала;

значение **true** – пересечение порога снизу будет вызывать срабатывание сигнала.

#### заменить (thr\_replace)

Если необходимо заменить предсказание значением доверительного интервала (верхнего, если **thr\_mirror = false**, нижнего, если **thr\_mirror = true**), следует установить для параметра **thr\_replace** значение **true**. По умолчанию предсказание не заменяется (параметр **thr\_replace** принимает значение **false**).

#### con\_intervals\_low

Ряд нижних значений доверительного интервала.

#### con\_intervals\_hight

Ряд верхних значений доверительного интервала.

### 6.3.9.5.1. Режим реального времени

В режиме реального времени значение сигнала рассчитывается на основе значений, собранных с целевого объекта, а статус сигнала определяется сравнением значения сигнала с порогом, предсказанным математической моделью и скорректированными настройками администратора.

При настройке данного режима администратор задает коэффициенты для модификации порогов, рассчитанных математической моделью. Причём, в случае линейных моделей<sup>57</sup> опорной точкой для расчета границ является предсказанное значение, то есть показатели моделирования, а для остальных моделей – их доверительный интервал.

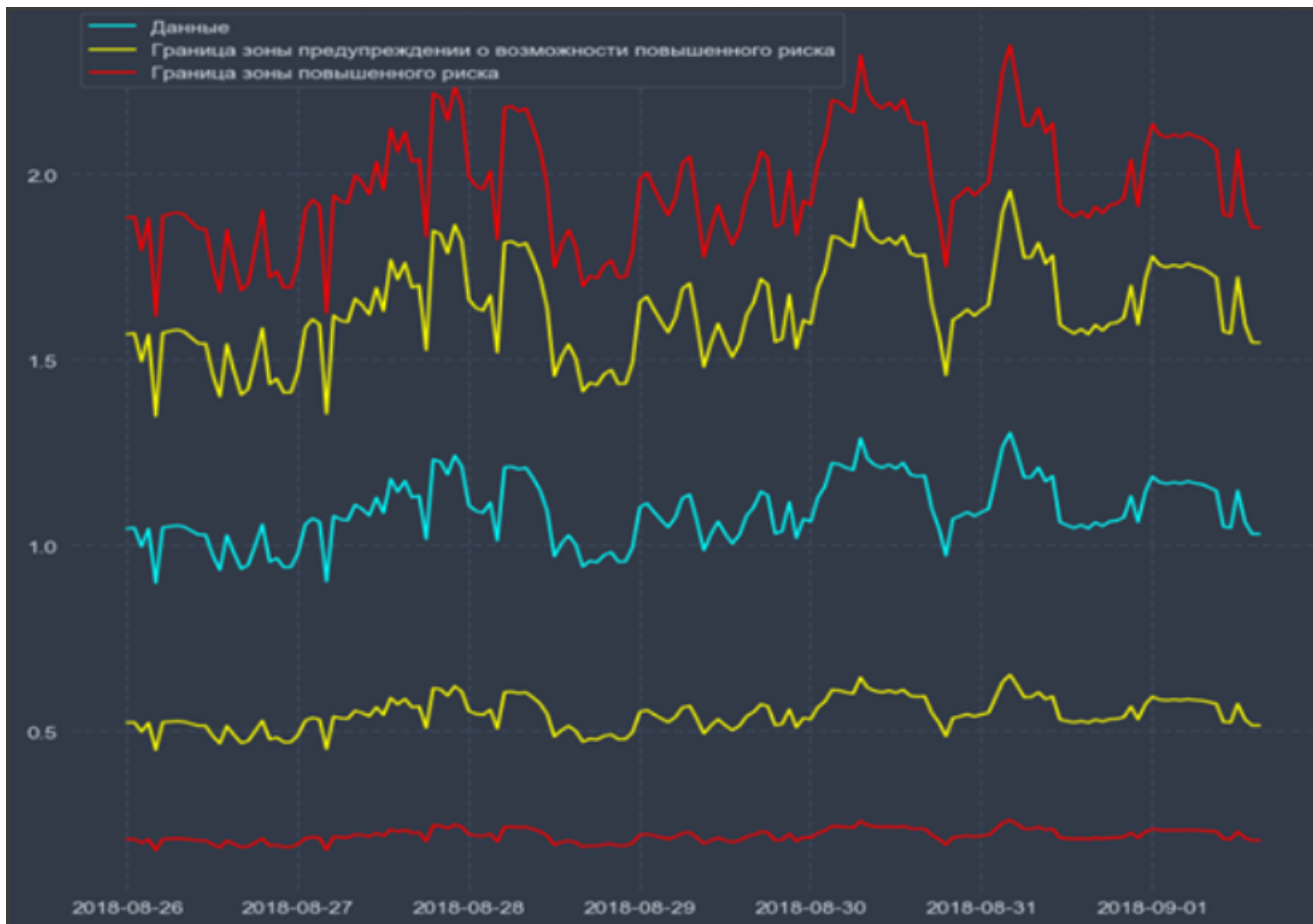


Рисунок 79. График предсказанных значений (голубая линия) и порогов, рассчитанных в режиме реального времени

Пороги рассчитываются по следующему алгоритму:

1. выбираются предыдущие значения целевого объекта в количестве, равном заданному параметру **Диапазон** (как правило, **600** секунд);
2. к выбранным значениям применяется агрегирующая функция, заданная параметром **Функция**. Полученное *значение тренда* используется на следующем шаге;
3. Пороговые значения рассчитываются по следующим формулам:

Верхний жёлтый порог = значение тренда + значение тренда \* Процент смещения порога жёлтой зоны / 100 + Абсолютное значение сдвига порога жёлтой зоны;

Нижний жёлтый порог = значение тренда - значение тренда \* Процент смещения порога жёлтой зоны / 100 + Абсолютное значение сдвига порога жёлтой зоны;

Верхний красный порог = значение тренда + значение тренда \* Процент смещения порога красной зоны / 100 + Абсолютное значение сдвига порога красной зоны;

Нижний красный порог = значение тренда - значение тренда \* Процент смещения порога красной зоны / 100 + Абсолютное значение сдвига порога красной зоны.

### 6.3.9.5.2. Режим статических границ

При настройке данного режима администратор, исходя из своего опыта, задаёт константные абсолютные значения порогов, ограничивающих зелёный, желтый и красный коридоры.

Значение сигнала устанавливается равным значению сигнала в момент времени наступления события.



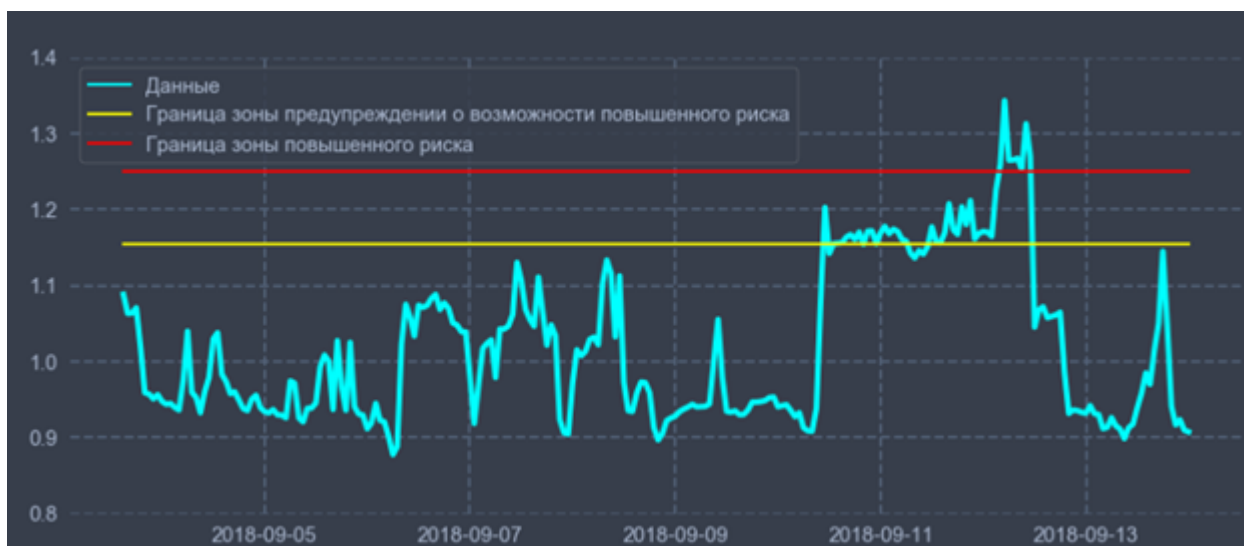


Рисунок 80. Режим статических границ. График предсказанного поведения исследуемого параметра и заданные администратором границы

### 6.3.9.5.3. Режим расчета оставшегося времени до наступления события

В данном режиме пороговые значения устанавливает администратор (аналогично режиму статических границ).

Если при настройке сигнала типа [MadaCalc](#)<sup>[46]</sup> администратор выбрал задачу на основе сценария с обработчиком **Расчёт порогов** в режиме **time to event**, то значение сигнала устанавливается равным временному интервалу, оставшемуся до наступления события, т.е. до момента времени, когда значение исследуемого параметра достигнет жёлтого или красного порога (тип события также задаёт администратор при настройке сигнала типа [MadaCalc](#)).

### 6.3.9.5.4. Режим отклонения от сигнала в процентах

Пороги рассчитываются так же, как и в режиме [реального времени](#)<sup>[95]</sup>.

Заданные администратором проценты смещения порогов используются для определения статуса сигнала типа [MadaCalc](#)<sup>[46]</sup>.

## Алгоритм расчёта значения и статуса сигнала типа MadaCalc

Алгоритм оперирует следующими величинами:

- реальное значение исследуемого параметра, полученное от целевого объекта;
- тренд – предсказанное значение исследуемого параметра.

Значение сигнала вычисляется по следующим образом:

1. выбираются предыдущие реальные значения целевого объекта в количестве, равном заданному параметру **Диапазон** (как правило 600 секунд);
2. для этих значений применяется формула:  

$$(\text{тренд} - \text{реальное значение}) / \text{тренд} * 100;$$
3. к результату применяется агрегирующая **Функция** (например, максимум). В итоге получается значение сигнала.

Для вычисления статуса сигнала его значение сравнивается с установленными администратором процентами смещения порогов:

- ❖ если значение сигнала меньше чем **Процент смещения порога жёлтой зоны**, то сигналу присваивается **зелёный** статус;
- ❖ если значение сигнала больше, чем **Процент смещения порога жёлтой зоны**, но меньше, чем **Процент смещения порога красной зоны**, то – **жёлтый** статус
- ❖ если значение сигнала превышает **Процент смещения порога жёлтой зоны**, то – **красный** статус.

### 6.3.10. Причинно-следственный анализ

Причинно-следственный анализ включает в себя алгоритмы использования причинно-следственных связей для повышения достоверности решения задачи. Реализованы следующие алгоритмы:

- ❖ [DTW](#) <sup>[100]</sup>;
- ❖ [Причинность Грейнджера](#) <sup>[100]</sup>;
- ❖ [MIC](#) <sup>[98]</sup>;
- ❖ [Корреляция Пирсона](#) <sup>[99]</sup>;
- ❖ [TCDF](#) <sup>[101]</sup>.

#### 6.3.10.1. MIC

Обработчик **MIC (Maximal information coefficient)** представляет собой алгоритм, в результате работы которого получается матрица в соответствии с выбранной оценкой (параметром **est**):

Если значения временного ряда не изменяются во времени, то в соответствии с теорией информации значение коэффициента MIC равно 0.

Формулы для вычисления коэффициента:

$$\text{MIC}(X, Y) = \max(I(X, Y) / \log(\min(n_x, n_y)))$$

$$I(X, Y) = D_{KL}(P_{(x, y)} \parallel P_x \oplus P_y), \text{ где:}$$

$I(X, Y)$  – внутренняя информация;

$D_{KL}$  – дивергенция Кульбака–Лейблера;

$P_x$  – предельное распределение  $X$ ;

$P_y$  – предельное распределение  $Y$ ;

$P_{(x, y)}$  – совместное распределение.

В данной реализации формула применяется попарно между всеми временными рядами. На основе посчитанных коэффициентов строится симметричная матрица размерности  $n \times n$  (где  $n$  - количество временных рядов).

На основе полученной матрицы (матрицы коэффициентов взаимосвязи рассматриваемых временных рядов) можно сделать вывод о зависимости временных рядов: каждый коэффициент – это число от **0** до **1**; при значении **1** – полная зависимость, при значении **0** – полная независимость.

#### Параметры обработчика

альфа (alpha)

Параметр для MIC. Если **alpha** в интервале (0, 1], тогда  $B$  (величина размера выборки, зависящая от  $\alpha$ ) будет выбран как  $\max(n^{\alpha}, 4)$ , где  $n$  – это число точек каждого временного ряда. Допустимые значения: от **0** (не включая) до **1**. Значение по умолчанию – **0.6**.

**c**

Параметр определяет, на сколько больше групп будет использоваться в сравнении с числом столбцов на каждую часть. Допустимые значения: больше **0**. Значение по умолчанию – **15**.

#### оценка (est)

Оценка. Возможные значения: **mic\_approx** (по умолчанию) и **mic\_e**. При значении **est = mic\_approx** вычисляется оригинальная статистика MINE, при **est = mic\_e** вычисляется равнохарактерная матрица.

### 6.3.10.2. Корреляция Пирсона

Обработчик **Корреляция Пирсона** представляет собой алгоритм, измеряющий линейную корреляцию между двумя наборами данных.

Формула для расчета коэффициента корреляции Пирсона:

$$\rho_{X,Y} = \text{cov}(X, Y) / (\sigma_X * \sigma_Y), \text{ где}$$

$\text{cov}$  – ковариация;

$\sigma_X, \sigma_Y$  – стандартные отклонения  $X$  и  $Y$  соответственно.

В данной реализации коэффициенты выходной матрицы (матрицы коэффициентов взаимосвязи рассматриваемых временных рядов) высчитываются попарно с каждым временным рядом. Коэффициент может принимать значения в отрезке **[-1; 1]**, где **0** – полная независимость рядов, **1** – полная зависимость, **-1** – полная обратная зависимость.

Примеры диаграмм рассеяния с различными значениями коэффициента корреляции ( $\rho$ ):

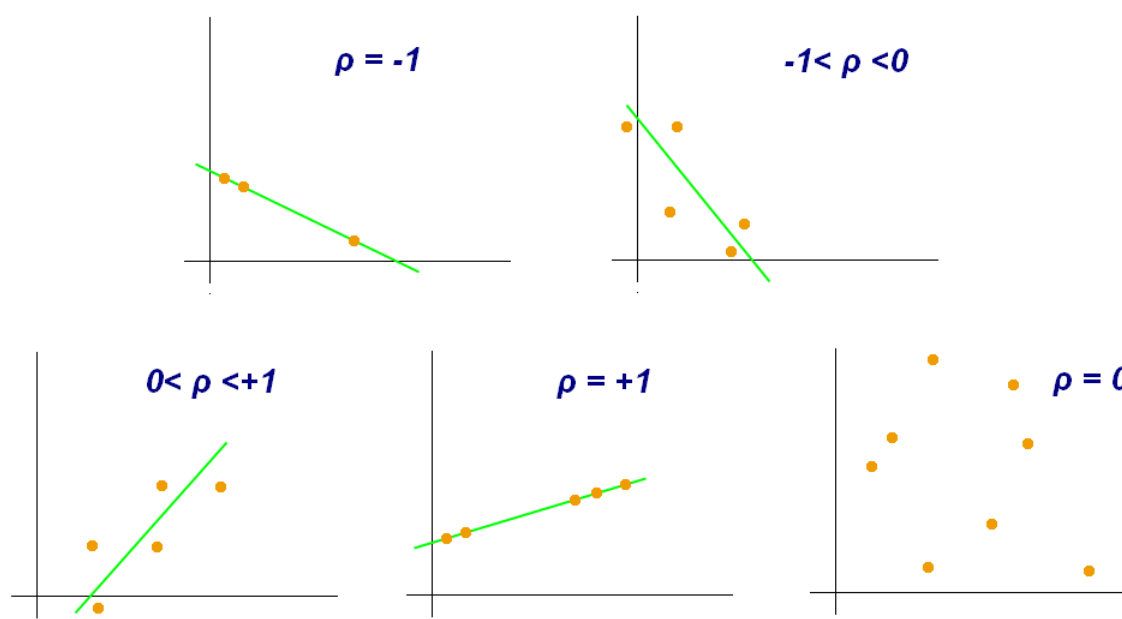


Рисунок 81. Диаграммы рассеяния со значениями коэффициента корреляции Пирсона

## Параметры обработчика

### по строкам (rowvar)

Если **rowvar** = **true**, каждая строка представляет собой переменную с набором ее значений, расположенным по столбцам. Если **rowvar** = **false** (по умолчанию) – ситуация транспонирована: каждая колонка представляет собой переменную с набором значений, расположенным по столбцам.

### 6.3.10.3. Причинность Грейнджера

Обработчик **Причинность Грейнджера** представляет собой алгоритм, определяющий причинно-следственную связь на основе двух принципов:

1. Причина предшествует ее следствию.
2. Причина обладает уникальной информацией о будущих значениях своего следствия.

Учитывая эти два предположения о причинности, выводится следующая гипотеза для выявления причинного эффекта  $X$  на  $Y$ :

$$\mathbb{P}[Y(t+1) \in A \mid \mathcal{I}(t)] \neq \mathbb{P}[Y(t+1) \in A \mid \mathcal{I}_{-X}(t)],$$

где  $\mathbb{P}$  относится к вероятности,  $A$  – произвольное непустое множество,  $\mathcal{I}(t)$  и  $\mathcal{I}_{-X}(t)$  соответственно обозначают информацию, доступную на момент времени  $t$  во всём пространстве и в модифицированном пространстве, в котором  $X$  исключено. Если вышеизложенная гипотеза принята, то считается, что  $X$  – причина по Грейнджеру для  $Y$ .

Результатом выполнения алгоритма является матрица размера  $n \times n$  (где  $n$  – количество временных рядов). Если значение в выходной матрице равно **1**, то временные ряды полностью зависимы, если значение равно **0** – полностью независимы.

### 6.3.10.4. DTW

Обработчик **DTW (Dynamic Time Warping)** представляет собой метод, который вычисляет оптимальное соответствие между двумя заданными последовательностями (например, временными рядами) с определенными ограничениями и правилами:

- ❖ Каждому индексу из первой последовательности должен соответствовать один или несколько индексов из другой последовательности, и наоборот.
- ❖ Первый индекс из первой последовательности должен совпадать с первым индексом из другой последовательности (но это не обязательно должно быть ее единственное совпадение).
- ❖ Последний индекс из первой последовательности должен совпадать с последним индексом из другой последовательности (но это не обязательно должно быть ее единственное совпадение).

Отображение индексов из первой последовательности в индексы из другой последовательности должно быть монотонно возрастающим, и наоборот, т. е. если  $j > i$  являются индексами из первой последовательности, то не может быть двух индексов  $l > k$  в другой последовательности, такой, что индекс  $i$  соответствует индексу  $l$  и индекс  $j$  соответствует индексу  $k$ , наоборот.

Оптимальное совпадение обозначается совпадением, которое удовлетворяет всем ограничениям и правилам и имеет минимальную стоимость, где стоимость вычисляется как сумма абсолютных разностей для каждой совпадающей пары индексов между их значениями.

Чем меньше значение соответствующей ячейки в таблице (выходной матрице), тем более зависимы последовательности (временные ряды).

### 6.3.10.5. TCDF

Обработчик **Temporal Causal Discovery Framework (TCDF)**, имея в качестве входных данных несколько временных рядов, обнаруживает причинно-следственные связи между этими временными рядами и выводит причинно-следственный график. Он также может прогнозировать один временной ряд на основе другого временного ряда. **TCDF** использует свёрточные нейронные сети на основе внимания в сочетании с этапом проверки причин. Интерпретируя внутренние параметры свёрточных сетей, **TCDF** также может обнаружить временную задержку между причиной и ее следствием.

Обученные временные причинно-следственные графики могут включать вмешивающиеся факторы и мгновенные эффекты. Эта широко применимая структура может использоваться для получения новых сведений о причинно-следственных зависимостях в сложной системе, что важно для надежных прогнозов, обнаружения знаний и принятия решений на основе данных.

#### Параметры обработчика

##### имя метрики (estimator\_name)

Метрика для оценки сходимости нейросети. Возможные метрики: **R2** (по умолчанию), **MSE**, **max**.

##### порог для оценки (estimator\_threshold)

Пороговое значение для проверки сходимости нейронной сети. Формат значений – **float**. Значение по умолчанию – **0.6** (для метрики **R2**)

##### максимальное число эпох (max\_epochs)

Максимальное число эпох обучения. Возможные значения: целые числа от **1**. Значение по умолчанию – **1000**.

##### размер ядра (kernel\_size)

Размер ядра свертки нейронной сети. Возможные значения: целые числа от **1**. Значение по умолчанию – **4**.

##### количество скрытых слоев (layers)

Скрытые узлы сверточной нейросети. Возможные значения: целые числа от **0**. Значение по умолчанию – **0**.

##### скорость сходимости (learning\_rate)

Скорость сходимости при обучении сети. Возможные значения: числа формата **float** от **0**. Значение по умолчанию – **0.01**.

##### имя оптимизатора (optimizer\_name)

Имя оптимизатора, используемого при обучении сети. Возможные оптимизаторы: **Adadelta**, **Adagrad**, **Adam** (по умолчанию), **AdamW**, **SparseAdam**, **Adamax**, **ASGD**, **LBFSG**, **NAdam**, **RAdam**, **RMSprop**, **Rprop**, **SGD**.

##### коэффициент расширения (dilation\_c)

Коэффициент расширения. Рекомендуется выбирать равным размеру ядра. Возможные значения: целые числа от **1**. Значение по умолчанию – **4**.

##### значимость (significance)

Уровень значимости, при котором изменение показателей является достаточным, чтобы обозначить потенциальную причину как подтвержденную причину. Возможные значения: числа формата **float** от **0**. Значение по умолчанию – **0.05**.

**ядро генератора случайных чисел (seed)**

Ядро генератора случайных чисел. Возможные значения: целые числа от **0**. Значение по умолчанию – **47**.

**вывод информации (verbose)**

Вывод информации во время расчета. Возможные значения: **true** (по умолчанию), **false**.

## 6.3.11. Фильтры

Фильтры – это методы отбора значений по определенным условиям.

Доступны следующие фильтры:

- ❖ [Заполнить NaN](#) <sup>[102]</sup>;
- ❖ [Фильтр Хампеля](#) <sup>[106]</sup>;
- ❖ [Удаление выбросов по сплайнам](#) <sup>[106]</sup>.

### 6.3.11.1. Заполнить NaN

Фильтр **Заполнить NaN** предназначен для удаления нулей и заполнения недостающих данных (т.е. очистки данных от значений nan). Реализованы следующие методы фильтрации:

- ☒ **среднее** – заполняет значения nan средним арифметическим данных;



Рисунок 82. Заполнение значений nan средним арифметическим данных

- **медиана** – заполняет значения nап медианой (серединное значение) данных, используя общий метод "среднее из двух средних";



Рисунок 83. Заполнение значений nап медианой данных

- **мода** – заполняет значения nап наиболее распространенной точкой данных (мера центрального положения);



Рисунок 84. Заполнение значений nап наиболее распространенной точкой данных

■ **кубический сплайн** – кубическая сплайн-интерполяция для заполнения значений nap;



Рисунок 85. Кубическая сплайн-интерполяция для заполнения значений nap

■ **линейный** – интерполирует данные линейной функцией для заполнения значений nap;



Рисунок 86. Заполнение значений nap с помощью линейной интерполяции

■ **ближайший** – заполняет значения nap ближайшим значением данных;



Рисунок 87. Заполнение значений nap ближайшими значениями



■ **квадратичный** – интерполирует данные квадратичной функцией для заполнения значений nan;



Рисунок 88. Заполнение значений nan с помощью квадратичной интерполяции

■ **кубический** – интерполирует данные кубической функцией для заполнения значений nan.



Рисунок 89. Заполнение значений nan с помощью кубической интерполяции

## Параметры фильтра

### метод (method)

Метод, описывающий правила для заполнения недостающих или некорректных данных. Возможные методы: **среднее** (**mean**), **медиана** (**median**), **мода** (**mode**), **кубический сплайн** (**cubic-spline**), **линейный** (**linear**), **ближайший** (**nearest**), **квадратичный** (**quadratic**), **кубический** (**cubic**). Метод по умолчанию – **среднее** (**mean**).

### Рекомендации по выбору метода фильтрации

- ❖ Способ заполнения значений nan одним и тем же значением (методы **среднее** (**mean**), **мода** (**mode**) и **медиана** (**median**)) дает грубые приближения, но вполне приемлем, если данные колеблются в небольшом диапазоне вокруг среднего, и сильных подъемов и спадов в данных нет.

- ❖ Для фильтрации данных чаще используются численные методы – интерполяция. Интерполяцией называется приближенное восстановление функции, где нам известен дискретный набор  $n$ -значений этой функции во временных точках, которые называют узлами интерполяции. Временной ряд является типичным примером такой функции. Задача состоит в том, чтобы заполнить все его пропущенные значения, т.е. найти значения в точках, не являющихся узлами интерполяции.
- ❖ Базовым из всех алгоритмов интерполяции, требующим наименьшего времени обработки является метод **ближайший (nearest)**. Такой метод лучше будет отображать динамику данных, чем средние. Для наилучшего приближения к реальным данным используется локальная интерполяция. Наиболее часто используемыми методами локальной интерполяции являются: **линейный (linear)**, **квадратичный (quadratic)**, **кубический (cubic)** и **кубический сплайн (cubic-spline)**.
- ❖ С увеличением количества пропущенных значений и временного интервала, в который попадает пропущенное значение, линейная интерполяция уступает интерполяционным полиномам большей степени. Метод **кубический сплайн (cubic-spline)** более точный по сравнению с остальными методами. Но в целях экономии вычислительной мощности, когда временной интервал, в который попадает пропущенное значение, минимальный, лучше использовать простую линейную интерполяцию – метод **линейный (linear)**.

### 6.3.11.2. Фильтр Хампеля

Фильтр **Фильтр Хампеля** (Outliers Hampel) предназначен для удаления или замены выбросов на скользящее среднее.

Работа фильтра заключается в последовательном прохождении окном фиксированного размера по всем данным. Каждая точка данных  $x_k$  последовательно является центром  $k$ -ого окна. Для обработки крайних значений данных в начало и конец ряда ставятся нули. Для каждого окна вычисляется медиана  $\tilde{X}_k$  и стандартное отклонение  $\sigma_k$  для текущей части данных. Затем центральное значение окна сравнивается с медианой. Если отклонение центральной точки  $x_k$  от медианы превышает  $n$  стандартных отклонений на этом окне  $\sigma_k$  (условие в виде формулы  $|x_k - \tilde{X}_k| > n * \sigma_k$ ), то значение этой точки заменяется на медианное  $\tilde{X}_k$ , если параметр **impulation = true**, или удаляется, если **impulation = false**.

#### Параметры фильтра

##### размер окна (window\_size)

Размер окна, в котором применяется фильтрация. Диапазон допустимых значений: целые числа от **0**; значение по умолчанию – **50**.

##### порог (n)

Число стандартных отклонений для обнаружения выбросов. Диапазон допустимых значений: целые числа от **0**; значение по умолчанию – **3**.

##### замещение (impulation)

Флаг. Если значение **true** – выбросы заполняются скользящим средним, если **false** – выбросы удаляются из набора данных. Значение флага по умолчанию – **true**.

### 6.3.11.3. Удаление выбросов по сплайнам

Фильтр **Удаление выбросов по сплайнам** (Outliers Splines) предназначен для замены выбросов с помощью интерполяции сплайнами.

В первую очередь в текущем окне ищутся выбросы. Определяется среднее  $\tilde{x}_i$  и стандартное отклонение  $\sigma_i$  по данным  $i$ -ого окна. Выбросами в текущем окне определяются точки, значение которых отличается от среднего более чем на  $\text{std\_coef} * \sigma_i$ , то есть такие  $x_i$ , для которых  $|x_i - \tilde{x}_i| > \text{std\_coef} * \sigma_i$ . Такие  $x_i$  удаляются из набора данных. По оставшимся точкам строится интерполяция сплайном  $k$ -ого порядка. Далее по построенной интерполяции восстанавливаются значения точек, определенных на предыдущем этапе как выбросы.

## Параметры фильтра

### размер окна (window\_size)

Размер окна, в котором применяется фильтрация. Диапазон допустимых значений: целые числа от **0**; значение по умолчанию – **50**.

### порог (std\_coef)

Число стандартных отклонений для обнаружения выбросов. Диапазон допустимых значений: числа типа **float** от **0.0**; значение по умолчанию – **2**.

### степень сплайна (k)

Степень интерполирующего сплайна. Диапазон допустимых значений: целые числа от **0** до **5**; значение по умолчанию – **3**.

## 6.3.12. Функции

Группа обработчиков Функции включает в себя обработчик **МинМакс**, который предназначен для получения максимального и минимального значения из временного ряда / множества временных рядов.

## Параметры метода МинМакс

### variable\_for\_max

Переменная для сохранения максимального значения. Тип данных – **float**.

### variable\_for\_min

Переменная для сохранения минимального значения. Тип данных – **float**.



---

# Предметный указатель

## С

ClickHouse 9

## М

MadaCalc 42

## А

Архитектура 8

## Д

Диапазон 46

## З

Задача 12, 13, 31

## О

Обработчик 12

## П

Период регламентных работ 54

Пороги 42

## С

Сервер 8, 16

Сигнал 42, 54

Системный процесс 8, 16, 56

Системный сервис 8

Сценарий 12, 20

## У

Удаление 16

Установка 16

## Ш

Шаблон сценария 12