

Slightly paramilitary Predictive Maintenance (draft)

Михаил В. Цыганов*, кафедра фотоники и физики микроволн ФФ МГУ

Алексей В. Ермаков†, ИТМФ МГУ

Артём В. Васильев‡, ИТМФ МГУ

02.04.2020

[Страница с проектом на github.](#)

Содержание

1	Постановка задачи	2
2	Общий анализ сигнала	2
2.1	Предварительный визуальный осмотр	2
2.2	Автокорреляция и кросскорреляция временных рядов температур блоков	3
2.3	Спектральный состав температурных рядов	5
2.4	Определение порогов включения и отключения для блоков	5
2.5	Восстановление логической схемы работы системы	5
2.6	Wavelet-анализ	6
2.7	Статистический анализ рядов состояний блоков	6
2.8	A Few More Things	8
3	Предсказание ряда температур	11
3.1	Модели класса Авторегрессия-скользящее среднее	11
3.2	Модели с глубокими нейронными сетями	14
3.2.1	Подготовка данных	14
3.2.2	Извлечение свойств из ряда	14
3.2.3	Результаты предсказаний и некоторые итоги	15
4	Предсказание ряда состояний блоков	17
4.1	Модель с глубокими нейронными сетями	18
4.1.1	Подготовка данных для элементарной модели	18
4.1.2	Архитектура элементарной модели и особенности подхода к обучению	19
4.1.3	Композиция элементарных моделей и итоговое предсказание	20
5	Вместо заключения	22

*tsyganov.mv15@physics.msu.ru

†ermakov.av15@physics.msu.ru

‡vasiliev.av15@physics.msu.ru

1 Постановка задачи

"Здравствуйте! На этот раз задание следующее. Данные представляют собой запись состояния системы и 10 её блоков за продолжительный период времени. Первые 2 столбца (?) – номер записи, 3-й столбец TimeCount – номер отсчета, 4-й столбец Mode - режим работы всей системы, определяющий нагрев блоков, 5-14-й столбцы T_i – температуры каждого из 10-ти блоков в градусах Цельсия, 15-24-й столбцы S_i - состояния каждого из 10 блоков (работает/не работает), 25-й столбец SystemState – состояние системы (работает/не работает). Важно отметить, что Mode в текущий отсчет определяет работу системы в следующий интервал времени (до следующего отсчета). Задача: построить алгоритм, который позволил бы спрогнозировать состояние системы на следующие 20 и 60 отсчетов, и предоставить спрогнозированные значения. Точность работы алгоритма оценивается по отклонению предсказанного ряда от истинного в расстоянии (во времени) до ближайшего отказа системы."

2 Общий анализ сигнала

2.1 Предварительный визуальный осмотр

На первом этапе визуализируем информацию связанную с работой отдельных блоков и системы в целом. Данные "чистые" и без пропусков значений. Приведём графики для первого блока.

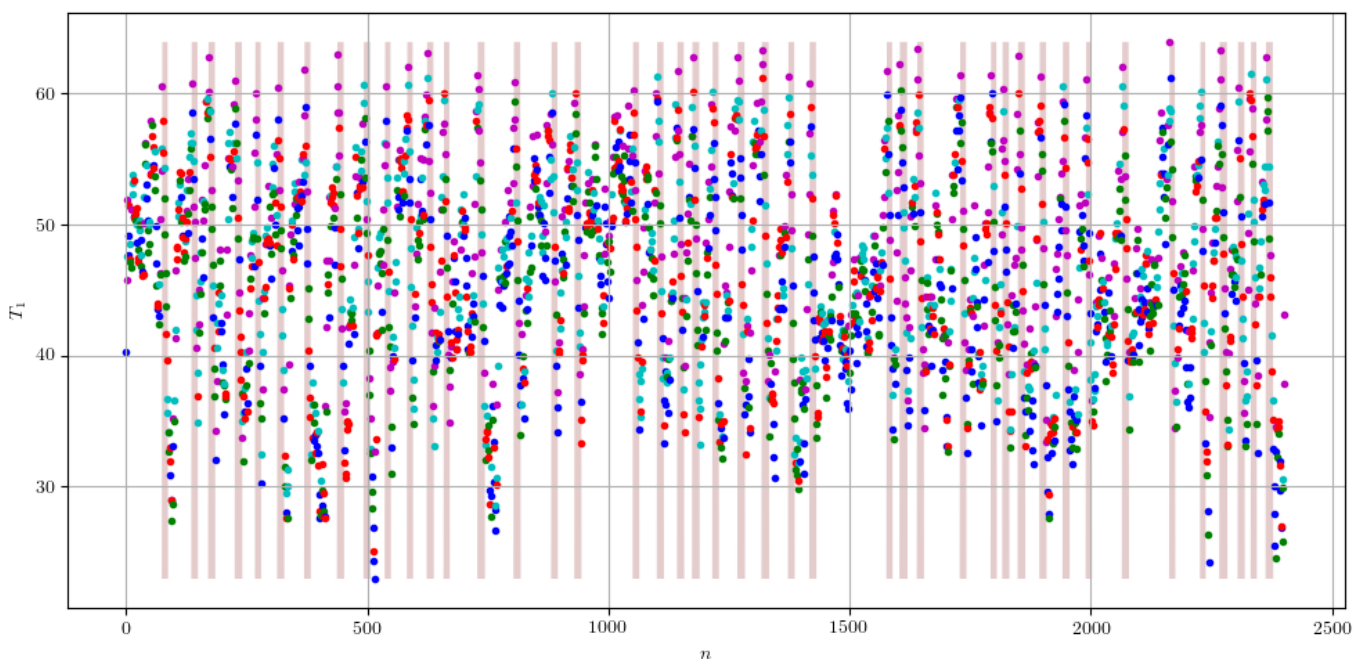


Рис. 1: График временного ряда для температуры первого блока. Цвет каждой точки показывает режим работы системы в следующий отсчет времени (0 - синий, 1 - зелёный, 2 - красный, 3 - голубой, 4 - пурпурный). Красная полупрозрачная заливка на фоне соответствует отказу блока.

Из визуального анализа данных графиков можно заключить, что все ряды для температур представляют собой циклические процессы нагрева и остывания, причем большую часть времени в рамках датасета температуры блоков совершают близкое к периодическому движение со слабо меняющимся от цикла к циклу периодом. Также стоит отметить, что существуют пороговые значения температур (вероятно одни и те же для всех блоков, что будет проанализировано ниже), при которых блоки перестают работать или вновь включаются.

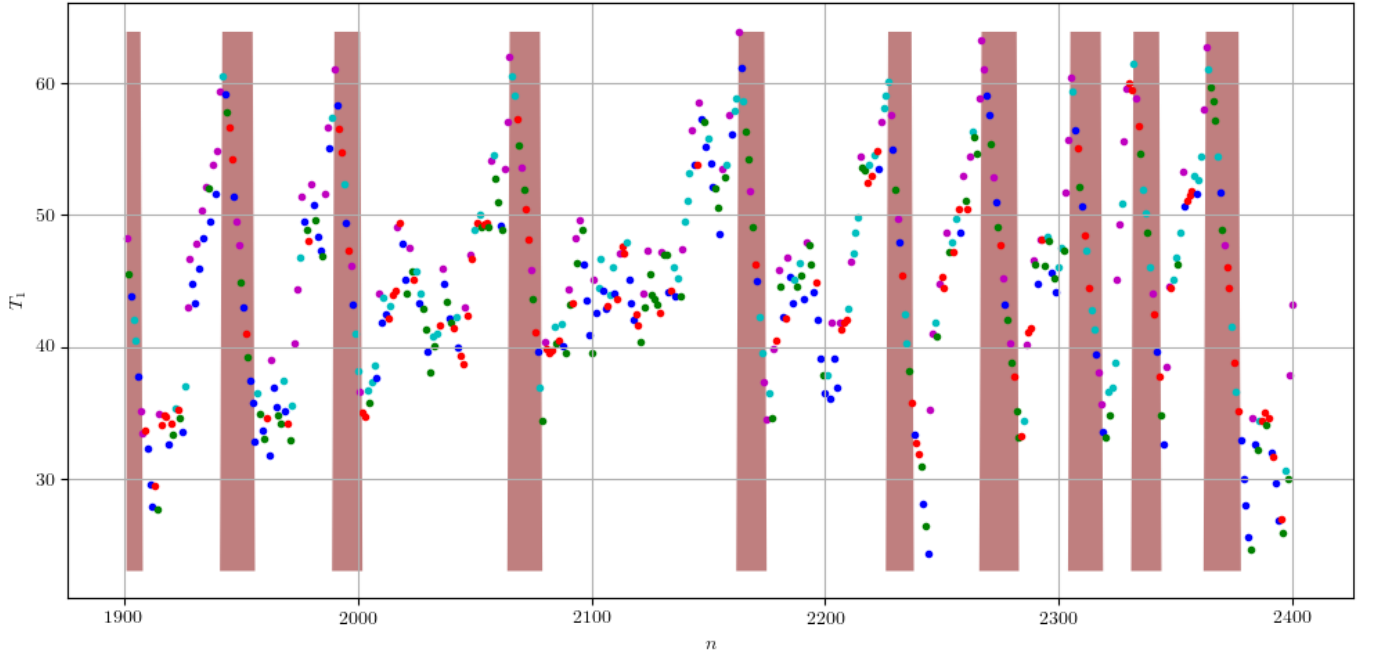


Рис. 2: График отрезка временного ряда за последние 500 отсчетов в датасете для температуры первого блока. Условные обозначения соответствуют предыдущему рисунку. Графики для остальных блоков можно посмотреть в [соответствующей директории](#) на странице проекта.

2.2 Автокорреляция и кросскорреляция временных рядов температур блоков

Начнём статистический анализ с изучения автокорреляционных функций для каждого из блоков. Используем стандартное определение для корреляционной функции для ряда температур для i -го блока ($i = \overline{1, 10}$):

$$r_{\tau}^{(i)} = \frac{\sum_{n=1}^{N-\tau} \left(T_n^{(i)} - \bar{T}^{(i)} \right) \left(T_{n+\tau}^{(i)} - \bar{T}^{(i)} \right)}{\sum_{n=1}^N \left(T_n^{(i)} - \bar{T}^{(i)} \right)^2}. \quad (1)$$

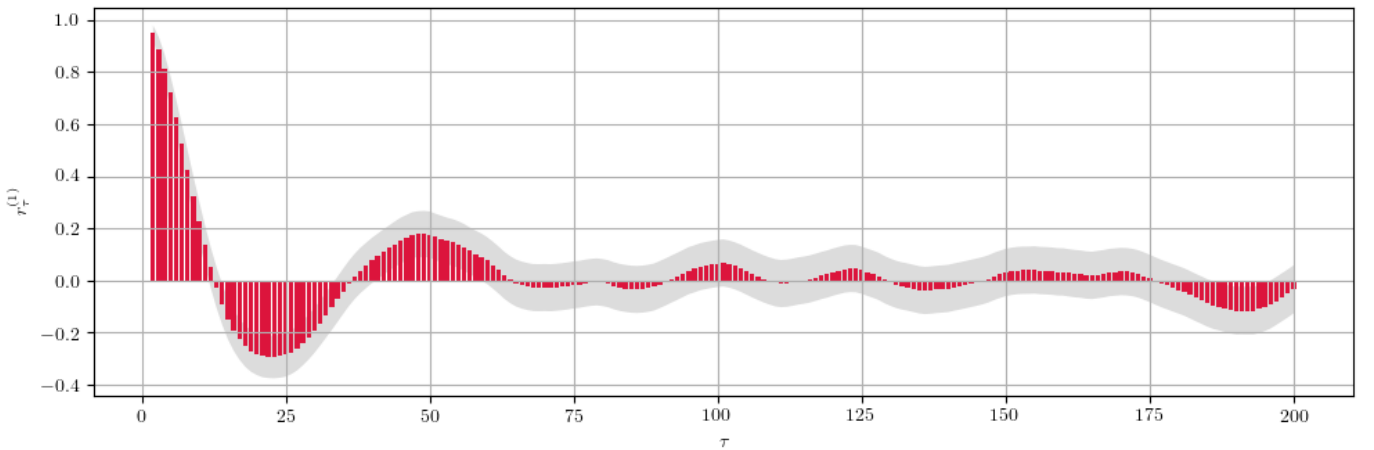


Рис. 3: График автокорреляционной функции для первого блока. τ - лаг, по оси ординат ординат величина корреляционной функции. Серым цветом обозначен доверительный интервал с уровнем значимости 0.8. Данное поведение является типичным для всех блоков. С графиками для всех блоков можно ознакомиться на [странице](#) проекта.

Коррелограммы для рядов температур различных блоков демонстрируют схожее поведение: наибольшее значение корреляции наблюдается на лагах до 12-15 отсчетов, после чего корреляционная функция меняет знак, достигает следующего максимума около лага в 48-50 отсчетов и далее осциллирует около нуля, будучи значимо уже не отличимой от нуля. Данные наблюдения позволяют однозначно заключить что временные ряды для температур не представляют собой гармонический сигнал с постоянным во времени главным периодом, пусть и обладают главным циклом, со схожей для всех блоков продолжительностью (48-50 отсчетов).

Если из постановки задачи и визуального осмотра рядов уже было очевидно, что ряды существенно не представляют собой белый шум, то степень величины влияния температуры одного блока на температуры других совсем не

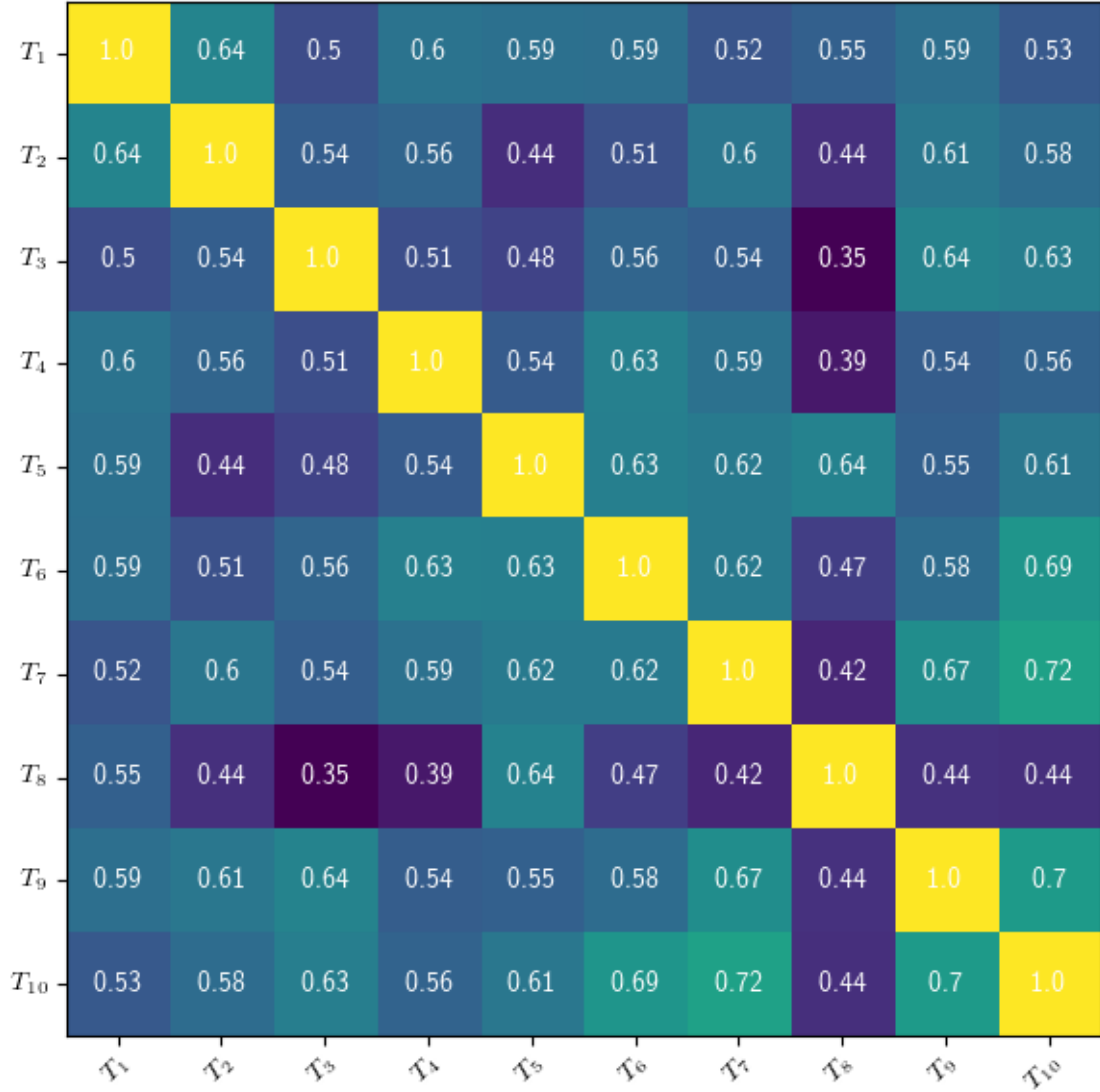


Рис. 4: Таблица со значениями коэффициента корреляции Пирсона для блоков "каждый с каждым".

ясна. Для определения кросскорреляции временных рядов температур блоков вычислим между ними коэффициент корреляции Пирсона:

$$\rho_{T^{(i)}, T^{(j)}} = \frac{\mathbb{E}[(T^{(i)} - \mu_{T^{(i)}})(T^{(j)} - \mu_{T^{(j)}})]}{\sigma_{T^{(i)}} \sigma_{T^{(j)}}}. \quad (2)$$

Изучая матрицу кросскорреляций легко заметить, что почти все ряды для температур существенно зависимы друг от друга. Более того, для каждого выбранного блока можно сказать какие блоки на него влияют "сильнее", а какие "слабее". Можно попробовать интерпретировать это в качестве факта физического соседства блоков на схеме. (На основе этой информации можно [попробовать](#) восстановить возможное относительное положение блоков, считая схему планарной).

Общую высокую взаимную корреляцию частей системы следует принять как факт и строить последующую модель анализа и предсказания исходя из того, что блоки системы обладают существенными коллективными эффектами. В этом смысле представляется идеологически неправильным строить модель предсказания для различных рядов температур независимо – разумным решением мы предлагаем считать одновременный анализ и построение предсказаний для всех 10 температурных рядов.

2.3 Спектральный состав температурных рядов

Для анализа спектров временных рядов была произведена оценка спектральной плотности мощности сигнала путём вычисления быстрого преобразования Фурье с оконной функций Хэмминга.

Спектры сигнала для всех блоков представлены на [соответствующей](#) странице проекта.

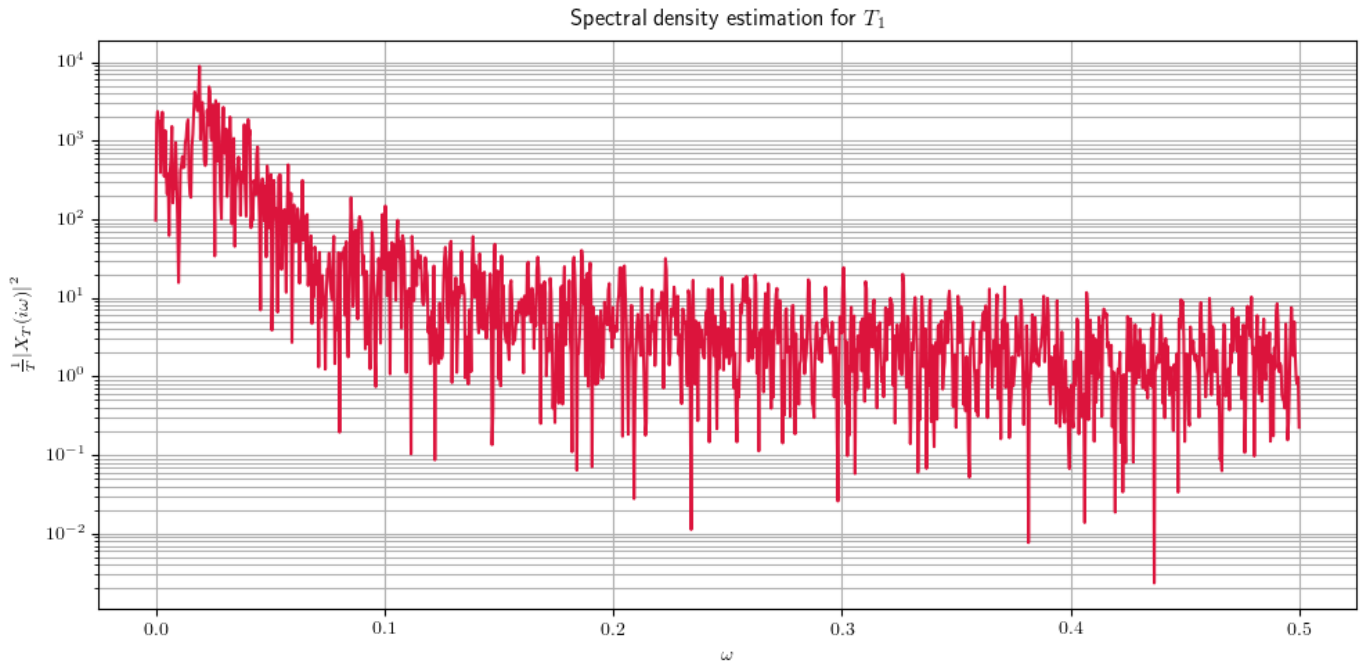


Рис. 5: Периодограмма температуры первого блока.

Спектр сигнала можно оценить как непрерывный, с очевидным насыщением в области низких частот (с частотой около 0.02, т.е. периодом 50 отсчетов). Никаких очевидных пиков в спектре для всех десяти блоков не наблюдается. В этом смысле, методами FFT, никакой новой информации получить не удалось. Возможно, в силу очевидной негладкой "пилообразной" структуры сигнала, более результативным было бы вейвлет-исследование ряда.

2.4 Определение порогов включения и отключения для блоков

Рассмотрим значения температур блоков во время тех отсчетов, когда логических сигнал их статуса имеет восходящий или нисходящий фронт. Легко заметить, что обоим фронтам соответствует практически одинаковые значения пар температур. Более того, такие значения оказываются общими для всех блоков. Это даёт основания полагать, что статус блока определяется исключительно расположением значения его текущей температуры относительно двух пороговых значений, которые мы далее будем называть порогом отключения блока и порогом включения блока. Оценку для них легко получить, находя среднее арифметическое по датасету от рассматриваемых величин температур.

	Temperature	Error
Sleep	61.55	0.06
Wake	33.90	0.03

Таблица 1: Средние значения порогов включения и выключения блоков по температуре и оценка ошибки среднего арифметического. Средние значения для каждого отдельного блока [есть](#) на странице проекта.

2.5 Восстановление логической схемы работы системы

Рассмотрим последние 11 колонок датасета, которые заполнены бинарными признаками статусов 10 отдельных блоков и статуса системы в целом. Данные колонки задают, как мы предполагаем, фрагмент таблицы истинности состояния системы, как функции 10 логических переменных. Используем стандартные средства работы с булевыми выражениями в WolframMathematica12 и упростим логическую функцию, заданную данным фрагментом таблицы истинности ([boolean.nb](#)). Полученное логическое выражение указывает на тривиальную схему резервирования – система теряет работоспособность, если более одного блока теряют работоспособность.

Проверить правильность такого предположения об архитектуре системы резервирования оказалось просто. Возьмём последние 11 колонок датасета и удалим все повторяющиеся строки, после чего отсортируем их по статусу работоспособности системы. Полученный результат полностью совпадает с полученным в WolframMathematica12

логическим выражением: статус системы "true", если все 10 блоков "true" или любой один из блоков "false", в любом другом случае на всем протяжении датасета система имеет статус "false". ([boolean.csv](#))

2.6 Wavelet-анализ

Как было замечено выше, анализ периодичности негармонического сигнала целесообразно проводить с помощью непрерывного (но не аналитического, а численного, это историческая путаница с терминологией) wavelet-преобразования. В силу тестового характера работы в большинстве случаев были выбраны параметры по умолчанию, однако несмотря на это удалось получить достаточный результат.

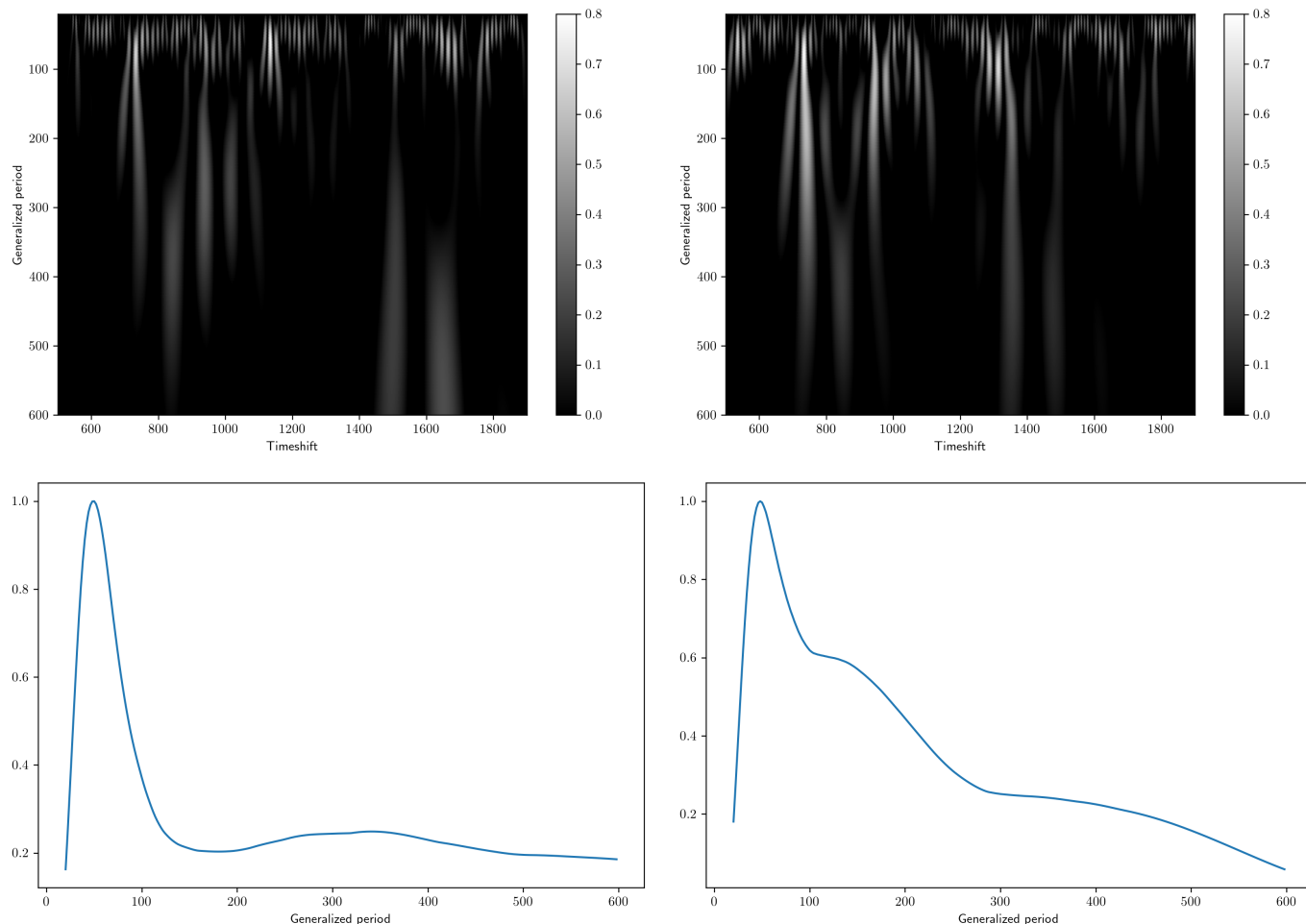


Рис. 6: Верхний ряд: абсолютное значение wavelet-отображения для рядов температур 3 и 4 блоков, нормированное на максимум, трешхолд снизу — 0.2. Нижний ряд: "спектры" температур в пространстве периодов (ширин) вейвлета.

Для исключения краевых искажений были исключены области с 0 по 499 отсчет и с 1900 по 2401. Такой отступ был выбран случайно из-за того, что максимальное значение в диапазоне параметра ширины вейвлета составляет порядка 600 отсчетов. Можно видеть характерные линии в области обобщенного периода порядка 50 отсчетов. Кроме того, есть некоторые низкочастотные явления, но их структура значительно менее регулярна.

По жертвуем временной разверткой и посмотрим на статистику временных масштабов явлений во временных рядах. На полученных спектрах видим явное преобладание процессов с периодом около 50 отсчетов. Такая ситуация характерна для всех блоков, средний период — 49.6 отсчетов. Кроме того, для некоторых блоков спектр не имеет выраженной низкочастотной составляющей (например 3), а для некоторых она существенна (4). С остальными графиками можно ознакомиться [здесь](#).

В заключение можно отметить, что используемый метод достаточно чувствителен к выбору функции вейвлета, это значит, что при более аккуратном выборе вейвлета (а может, и создании собственной функции) можно получить значительно более точные результаты. Кроме того, предсказать значения ряда можно, работая в пространстве вейвлет образа датасета.

2.7 Статистический анализ рядов состояний блоков

Кроме Wavelet-анализа рядов температур проведем анализ рядов состояний блоков. Мы будем искать распределения периодов между соседними отключениями блока (переход из $S = 1$ в $S = 0$). Получить массив периодов можно

так: численно продифференцировать ряд состояний, найти индексы всех элементов, равных -1 , полученный массив еще раз продифференцировать. Для этого массива можно построить функцию распределения, оценить время следующего отказа и доверительный интервал для него. Для 10 блоков приведем такие результаты (доверительный интервал на уровне 80%):

Таблица 2:

Номер блока	Средний период	Стандартное отклонение	Доверительный интервал (80%)
1	54	26	31, 75
2	57	25	34, 75
3	56	23	34, 71
4	56	27	37, 76
5	53	26	33, 80
6	58	26	36, 85
7	55	24	33, 78
8	55	27	33, 90
9	55	25	34, 75
10	55	25	33, 79

Исходя из данных, приведенных в таблице, можно построить наивное предсказание для времени отказа блока, например, на 2 отказа вперед. В таком случае наиболее вероятные "времена" отказа для блоков приведем в следующей таблице:

Таблица 3:

Номер блока	Время 1-го отказа	Время 2-го отказа
1	2416 ± 26	2470 ± 52
2	2408 ± 25	2464 ± 50
3	2406 ± 23	2461 ± 46
4	2401 ± 27	2457 ± 54
5	2428 ± 26	2481 ± 52
6	2401 ± 26	2458 ± 52
7	2417 ± 24	2472 ± 48
8	2401 ± 27	2455 ± 54
9	2407 ± 25	2462 ± 50
10	2413 ± 25	2467 ± 50

В дальнейшем можно усовершенствовать этот анализ, усложнив модель (учитывая, например, значение температуры, производной температуры, проводя сглаживание с помощью скользящего среднего или фильтра Калмана) но для тестового задания достаточно. Приведу также картинку функции распределения для 1 блока. Информацию о расчетах и данные для всех блоков можно найти на [гитхабе](#).

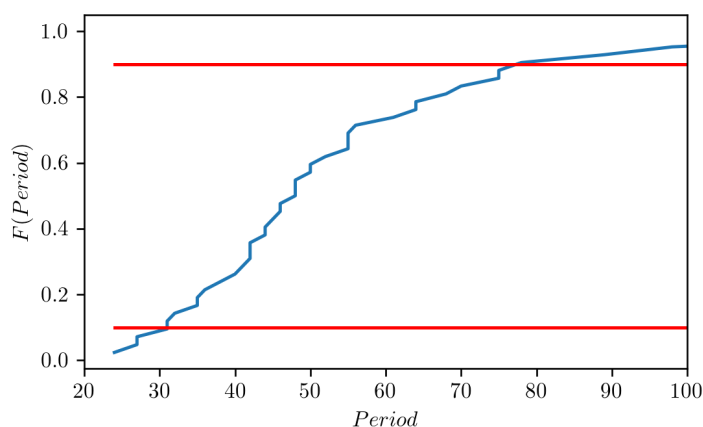


Рис. 7: Функция распределения периодов между отключениями первого блока. Красные линии соответствуют уровням вероятности 0.1 и 0.9.

2.8 A Few More Things

Перед началом работы над построением модели предсказания, на наш взгляд, было бы полезно узнать как можно больше информации о системе в целом.

Одним из важных вопросов остался связан с категориальным признаком Mode: является ли он "внешним параметром", или же определяется функционально от состояния системы. В сопроводительном письме к задаче утверждалось, что Mode в текущий отсчет связан с работой (нагревом?) системы в следующий момент времени. Для того что бы выявить влияния данного категориального признака на работу системы мы построили зависимость для средней температуры по всем блокам, максимальную температуру среди блоков, оценку первой производной от средней температуры по времени, а также оценку второй производной от средней температуры по времени, как функцию времени, и отметили на этих графиках соответствующие значения параметра Mode.

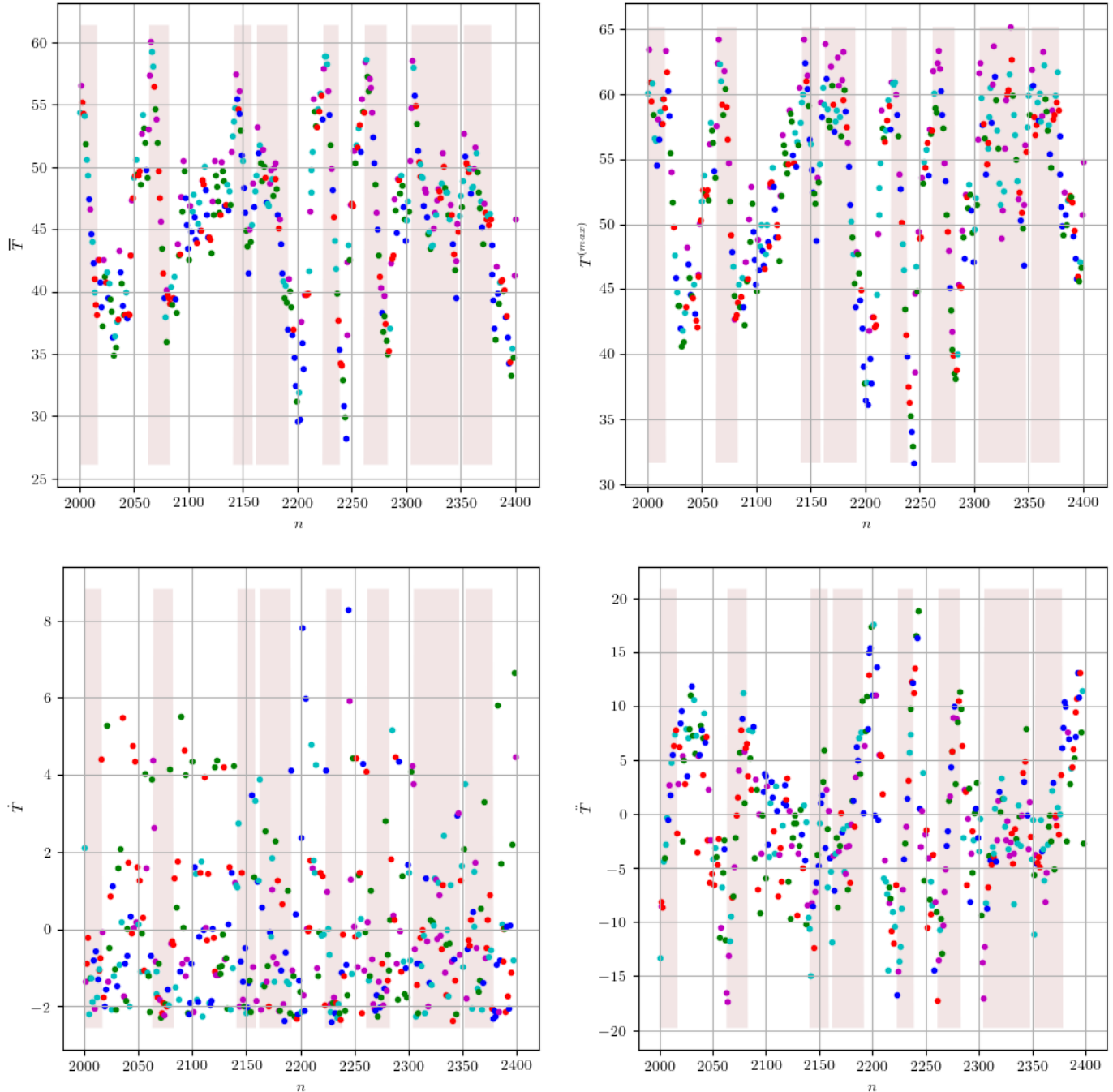


Рис. 8: Различные зависимости температур "системы в целом" от времени на некотором временном отрезке. Цвет каждой точки показывает режим работы системы в следующий отсчет времени (0 - синий, 1 - зелёный, 2 - красный, 3 - голубой, 4 - пурпурный). Красная полупрозрачная заливка на фоне соответствует отказу системы. Множество различных графиков зависимости температуры системы от времени можно найти [на странице проекта](#).

Даже после анализа таких временных зависимостей не представляется возможным однозначно ответить на то, каково влияние Mode на состояние системы. Зависимость от данного категориального параметра выглядит совершенно случайной. Возможно сама эволюция во времени параметра Mode окажется осмысленной. Сравним

график временного ряда, значение которого в каждом отсчете равняется числу перегревшихся блоков, с временным рядом значений параметра Mode.

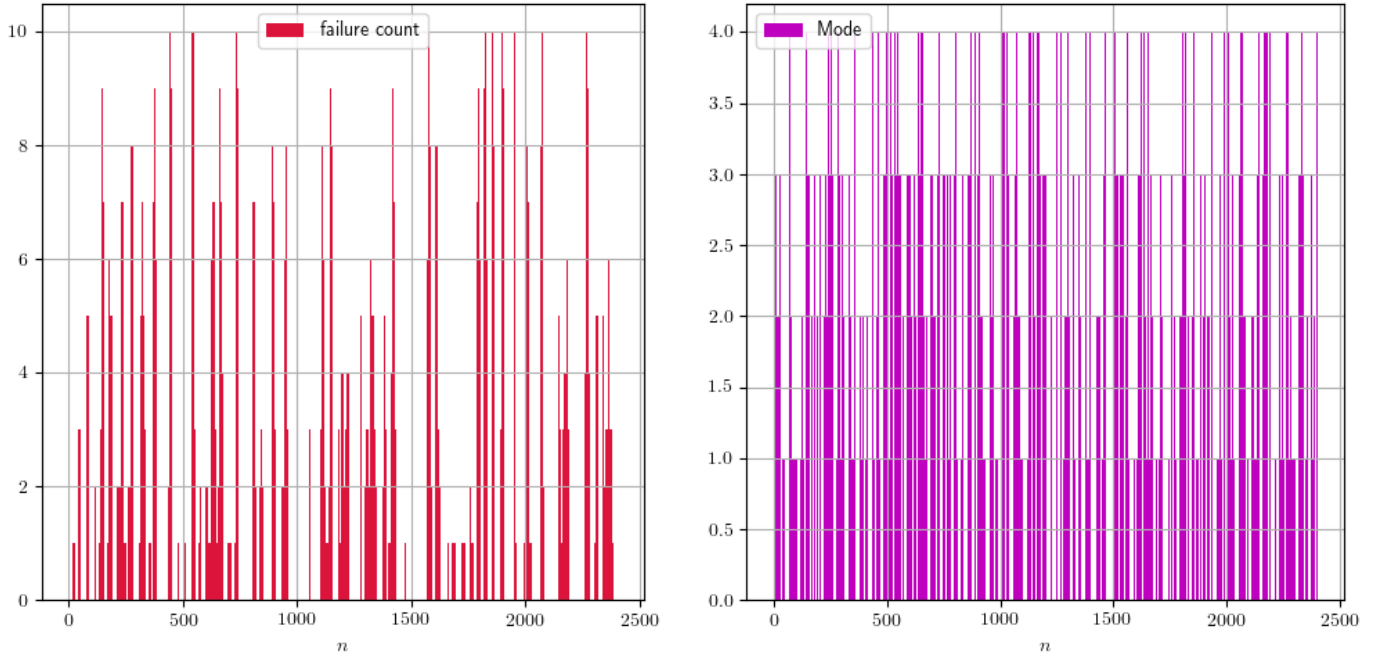


Рис. 9: К определению роли параметра Mode.

Как выяснилось, система провела примерно равное количество времени во всех пяти значениях параметра Mode (486, 470, 508, 478 и 459 отсчета в каждом из значений Mode от 0 до 4 соответственно), причём все возможные значения равномерно распределены во времени. Никакой структурой сигнал для параметра Mode не обладает. В этой связи возникает вопрос о необходимости вовсе принимать во внимание данный категориальный параметр при дальнейшем анализе и построении модели предсказаний.

Рассмотрим ещё одну возможность влияния параметра Mode на систему: построим для различных значений Mode зависимости изменений температуры блоков к следующему отсчету времени, как функцию текущей температуры.

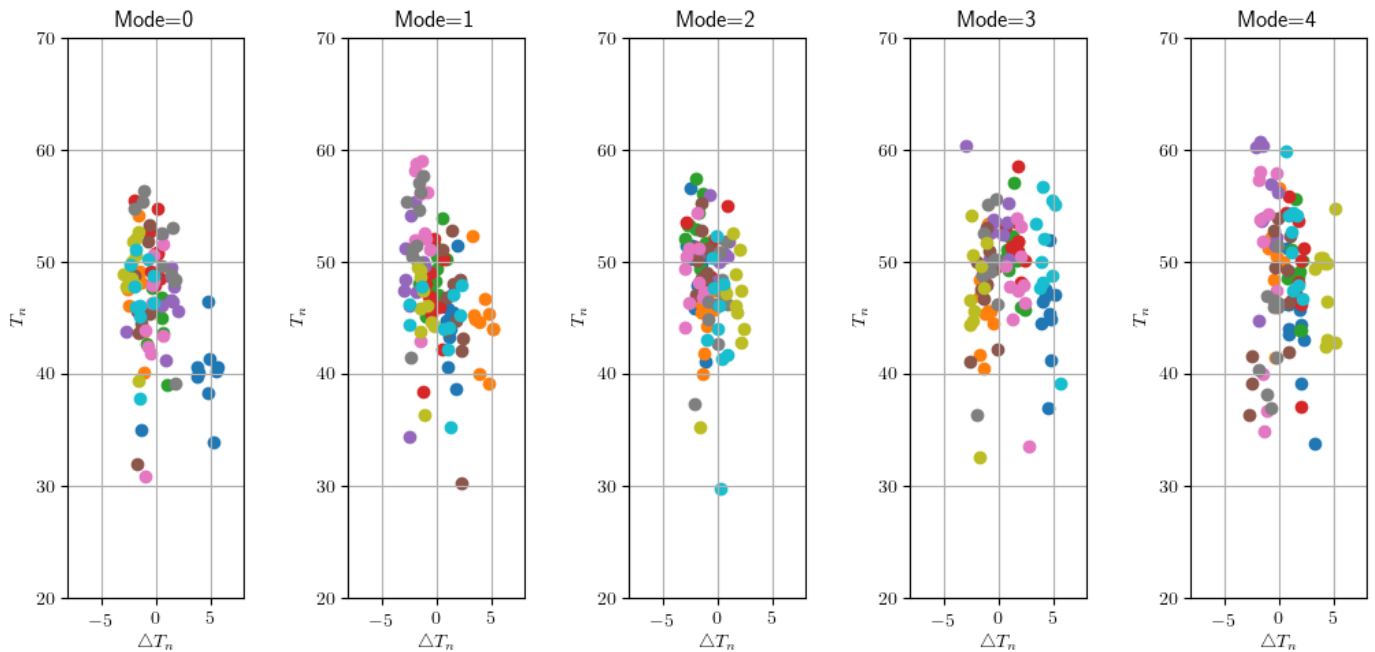


Рис. 10: Зависимости изменения температуры блоком на следующем отсчете, как функция текущей температуры (на графиках транспонировано) для различных значений Mode. Цвета точек на графиках кодируют номера блоков.

Можно заметить, что все блоки демонстрируют схожий характер указанной зависимости при всех значениях параметра Mode. Также видно, что экспериментальные точки с хорошей точностью объединены в один кластер, на

положение которого на графиках оказывает влияние значение Mode. Для проверки гипотезы о том, что параметр Mode всё же нетривиально влияет на систему, рассмотрим положение центра масс кластера точек, характеризующих систему в пространстве параметров "средняя температура системы в текущий отсчет времени", "среднее изменение температуры системы за время от текущего до следующего отсчета".

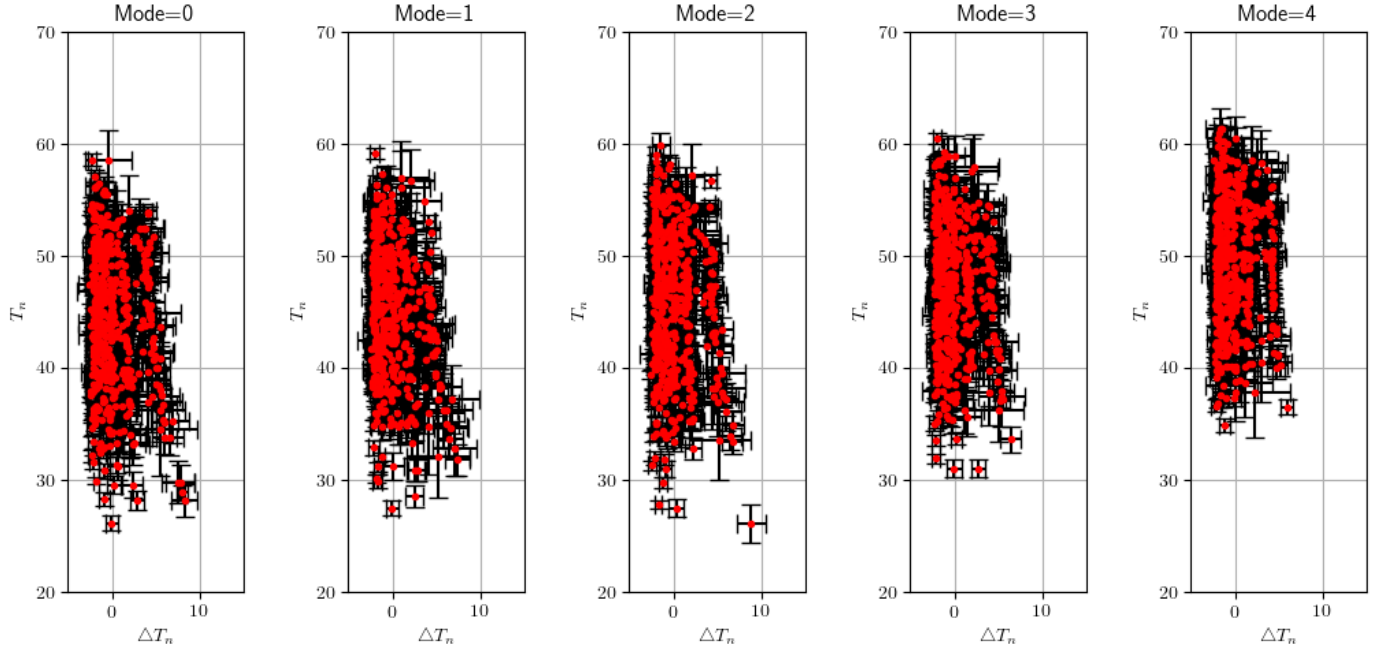


Рис. 11: Зависимости изменения средней температуры системы на следующем отсчете, как функция текущей температуры (на графиках транспонировано) для различных значений Mode.

Заметно, что положение центра масс данного кластера сдвигается в сторону более высоких температур с ростом значения параметра Mode. Данная зависимость системы от параметра Mode весьма не straightforward, потому не вполне понятно как следует использовать это знание для разумного фича-инжиниринга.

Завершая наиболее очевидный предварительный анализ также следует отметить, что все блоки системы "загружены" равномерно и они находились в выключенном состоянии после перегрева примерно равное количество времени.

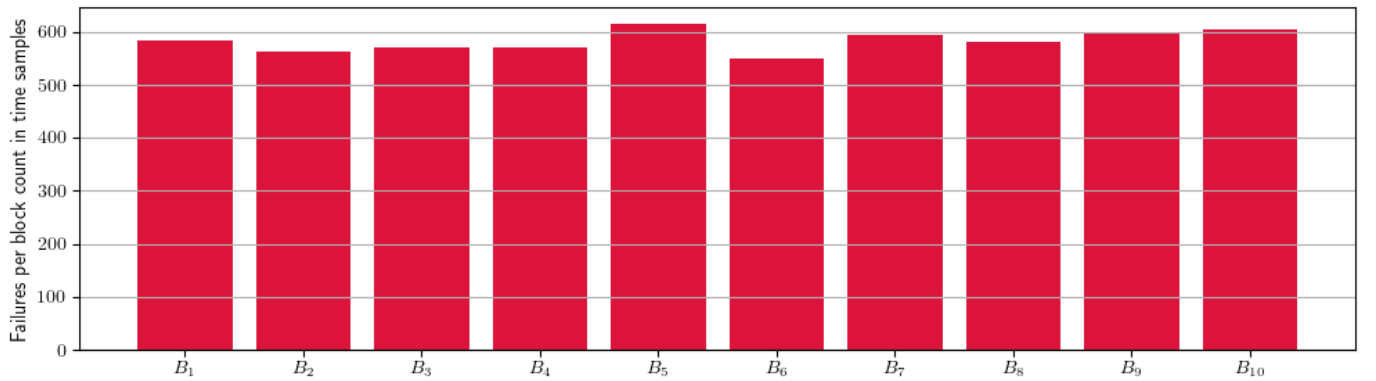


Рис. 12: Число отсчётов, которые блоки провели в режиме отказа.

3 Предсказание ряда температур

После беглого ознакомления со статьями о создании моделей предсказаний для многомерных временных рядов было выявлено, что доказали свою работоспособность на практике следующие подходы: использование классических статистических моделей авторегрессия—скользящее среднее и их расширений; различные модификации рекуррентных нейронных сетей; градиентный бустинг на деревьях решений. Воспользуемся этим при построении нашего собственного решения.

Мы планировали реализовать все эти подходы к предсказанию 10 временных рядов температур блоков, после чего объединить предсказания моделей—предсказателей температур в качестве входа в модель второго уровня, которая должна иметь архитектуру 10-выходного бинарного классификатора предсказывающего уже работоспособность отдельных блоков в заданных отсчетах времени в будущем. На этом пути мы успели подробно исследовать ARIMA-класс модели и рекуррентные нейронные сети. "Деревянные" модели ограничились только стандартными имплементациями из пакета WM – уже на этом этапе стало понятно, что данный подход в целом "тупиковый" и нет никакого смысла тратить время на обучение и стакание XGBoost-ов. Подпункты данной главы оставлены в отчёте с целью продемонстрировать нашу работу по исключению гипотез о возможном виде решения. Для несколько более успешной части см. главу "Предсказание ряда состояний блоков".

3.1 Модели класса Авторегрессия-скользящее среднее

Так как на протяжении всего датасета среднее значение почти не менялось, то можно было бы попробовать построить глубокую авторегрессионную модель. Такие попытки не увенчались успехом, так как графики даже примерно не отражали пилообразной структуры данных. Мы разделили данные на первые 2300 отсчётов, по которым и выявлялся фиттирующий случайный процесс и на последние 100 отсчётов, которые считали тестовыми. Выбор гиперпараметров вычислялся автоматически:

	Candidate	AIC
1	ARProcess(7)	3748.3
2	ARProcess(6)	3758.58

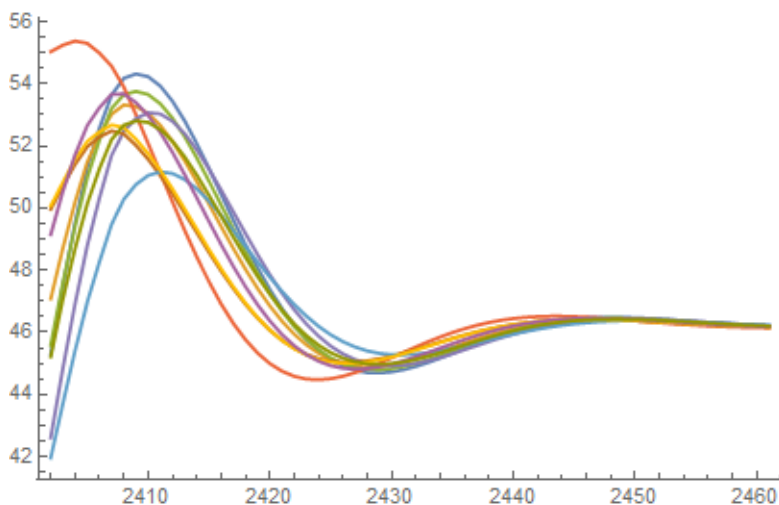


Рис. 13: Прогноз на 60 отсчётов для всех блоков с использованием авторегрессии 8 порядка.

Далее было решено попробовать учесть эффекты движения среднего, но ARIMA модели давали результаты ещё хуже — сваливание в среднее значение происходило быстрее. Тогда появилась идея, что данные методы плохо улавливают какие-то общие тренды датасета. Ограничение данных на последние 200-500 отсчётов не улучшило ситуацию. Последней надеждой были "сезонные" колебания, для учёта которых была использована самая ресурсоёмкая модель SARIMA. Выбор гиперпараметров также происходил автоматически. Несмотря на казалось бы меньшие значения параметров AIC для авторегрессионных алгоритмов, можно явно видеть, что на тестовом участке (последние 100 отсчётов) AR-модель даёт худшие предсказания, которые не отражают ничего, кроме среднего значения температуры блока. Было решено проверить значения для автокорреляционных функций и провести Q-тест Льюнга-Бокса для остатка полученной модели (а не для начальных данных). И если в случае корреляционных функций, действительно, результаты лучше выглядят для AR-процесса (меньше высоких спайков), то Q-тест явно демонстрирует предпочтительность использования SARIMA.

Предсказания температуры на 60 отсчётов для 1 блока и для всех блоков сразу

Можно подумать, что максимумы, хоть они и не доходят до threshold'a в 60 градусов, будут соответствовать моменту отключения блока, в то время как минимум - наоборот. Однако анализ погрешностей и прочие результаты

	Candidate	AIC
1	SARIMAProcess($\{4, 0, 6\}, \{0, 1, 3\}_{42}$)	4309.12
2	SARIMAProcess($\{4, 0, 5\}, \{0, 1, 3\}_{42}$)	4327.96
3	SARIMAProcess($\{4, 0, 4\}, \{0, 1, 3\}_{42}$)	4328.36
4	SARIMAProcess($\{3, 0, 6\}, \{0, 1, 3\}_{42}$)	4332.96
5	SARIMAProcess($\{4, 0, 6\}, \{0, 1, 2\}_{42}$)	4334.03
6	SARIMAProcess($\{5, 0, 6\}, \{0, 1, 3\}_{42}$)	4334.6
7	SARIMAProcess($\{4, 0, 5\}, \{0, 1, 2\}_{42}$)	4352.46
8	SARIMAProcess($\{5, 0, 5\}, \{0, 1, 3\}_{42}$)	4356.28
9	SARIMAProcess($\{3, 0, 5\}, \{0, 1, 3\}_{42}$)	4360.33
10	SARIMAProcess($\{3, 0, 4\}, \{0, 1, 3\}_{42}$)	4361.45

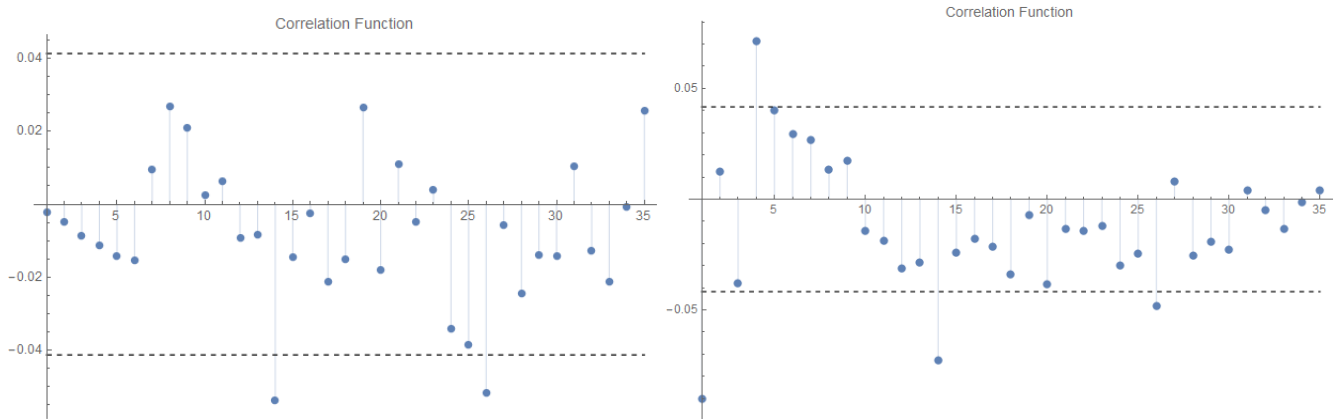


Рис. 14: Корреляционные функции для AR- и SARIMA-процессов.

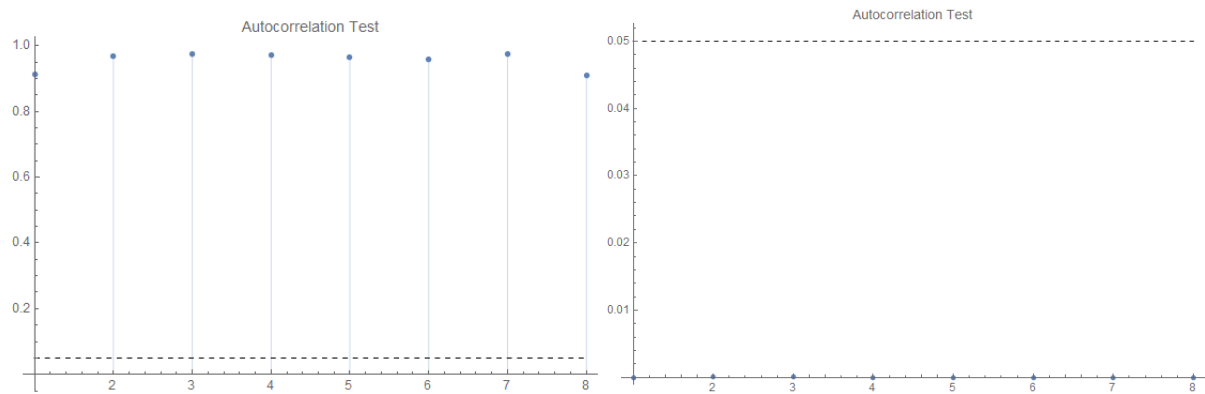


Рис. 15: Q-тест Льюнга-Бакса для AR- и SARIMA-процессов.

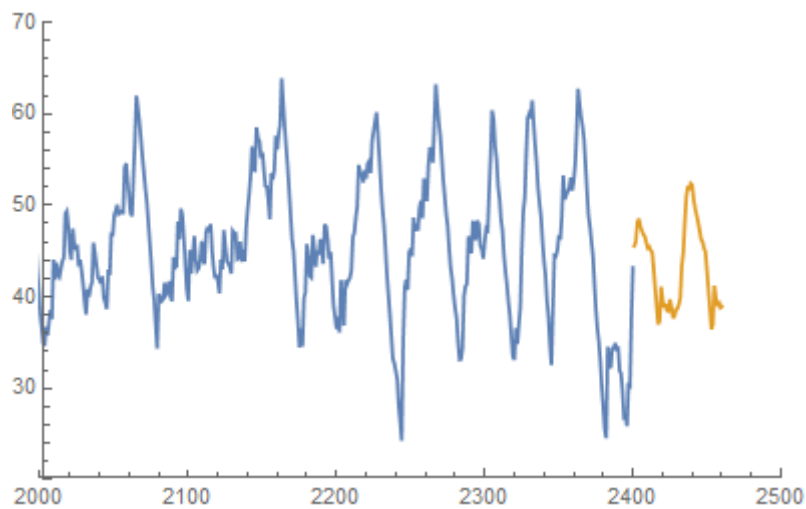


Рис. 16: Прогноз на 60 отсчётов для первого блока с использованием SARIMA.

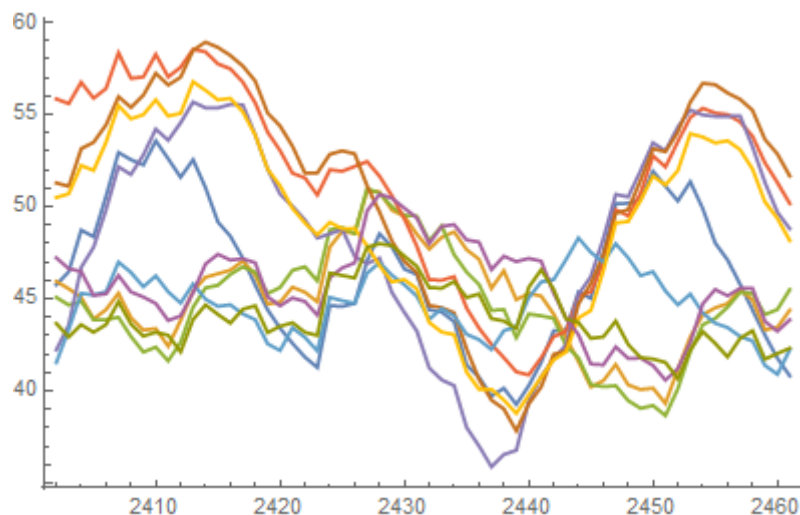


Рис. 17: Прогноз на 60 отсчётов для всех блоков с использованием SARIMA.

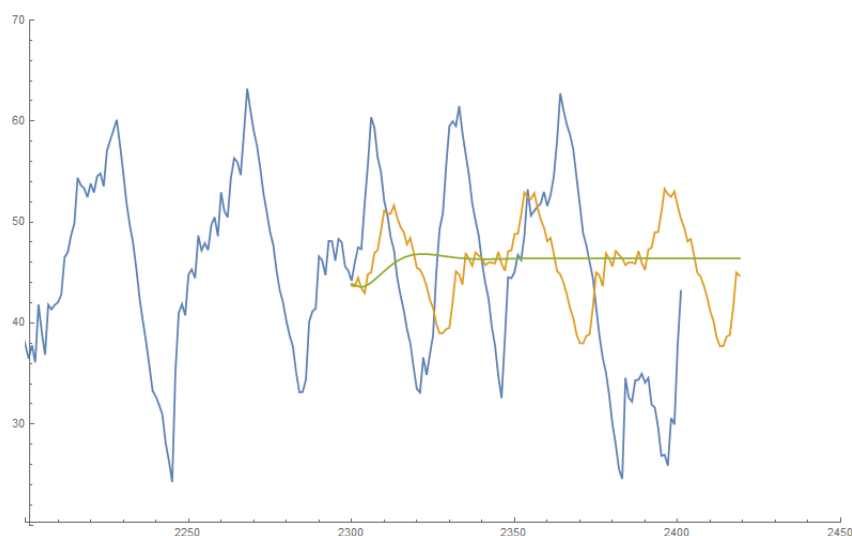


Рис. 18: Предсказания для тестового участка и прогноз на 20 отсчётов для первого блока с использованием AR- и SARIMA-процессов.

в этом подходе ещё только предстоит получить. При этом на тестовом участке в последние 100 отсчётов результаты выглядят не слишком обнадеживающе.

Пытаясь окончательно "похоронить" модели класса авторегрессия—скользящее среднее, мы организовали алгоритмический "полный перебор руками" гиперпараметров модели, предположив что WM давала не точные значения локального минимума функционала ошибки в силу некоторых скрытых особенностей жадного алгоритма перебора параметров модели. Также мы попробовали применить модели класса SARIMA для ряда температур, прошедшего через рекурсивный фильтр Калмана или для ряда, представляющего собой значения скользящего среднего от исходного ряда температур. Результаты работы можно посмотреть в таблице со значениями AIC для различных моделей на странице проекта. [Исходный ряд](#), [фильтрованный по Калману](#), [ряд значений скользящих средних за 13 отсчетов](#). К сожалению, после этой работы также пришлось заключить, что все исследуемые модели дают качество предсказаний значимо не лучше, чем просто предсказание рядов средним значением температур.

3.2 Модели с глубокими нейронными сетями

Как количество известных возможных подходов ([arXiv:1807.00425](#), [arXiv:2001.08317](#), [arXiv:1802.06338](#), [arXiv:1811.05852](#), [arXiv:1711.00073](#), ...) к применению нейронных сетей в задачах прогнозирования многомерных временных рядов само по себе, так и число наших подходов к собственному решению данной проблемы (соответствующие файлы и веса моделей в случае необходимости можно обнаружить на странице проекта в github: [\[1\]](#), [\[2\]](#), [\[3\]](#), [\[4\]](#), [\[5\]](#), [\[6\]](#), [\[7\]](#), [\[8\]](#), [\[9\]](#), [\[10\]](#), [\[11\]](#), [\[12\]](#), [\[13\]](#), [\[14\]](#), [\[15\]](#), ...) взрывным образом стремиться к бесконечности. В этой связи не представляется уместным описывать подробно все наши попытки построить нейростоевое решение. Ограничимся лишь общим описанием.

Все опробованные решения выполнены в архитектуре последовательность—в—последовательность (seq2seq). Опробованы решения как с двусоставными сетями (encoder-decoder), так с единственным рекуррентным блоком. В качестве рекуррентных элементов сети использовались ячейки GRU (не только ради каламбура), LSTM, а также их комбинации.

Наилучшие модели получены для следующей архитектуры: входной полносвязный слой \rightarrow стек из 2-4 слоёв с LSTM ячейками \rightarrow 2-3 "обычных" полносвязных слоя с постепенно уменьшающимся до 1-цы числом нейронов. На вход модель принимает отрезок из записей свойств за несколько сотен предыдущих отсчетов и на выходе получает отрезок такой же длины значений ряда температур для одного выбранного блока с заданным сдвигом во времени "в будущее".

Начиная с данной главы и далее будем считать известными следующие обозначения:

- Нумерация всех последовательностей начинается с нуля.
- Заглавные латинские буквы N , M , T , S идентифицированы с рядами номеров отсчетов по времени, значений категориального параметра Mode, температурами блоков и состоянием блоков соответственно;
- Заглавная латинская D оставлена для обозначения тренировочного датасета.
- Нижний правый индекс соответствует номеру временного отсчета. То есть T_n — значения температур блоков в момент n -го временного отсчета;
- Правый верхний индекс в круглых скобках нумерует свойства (фичи). То есть $T^{(i)}$ — временной ряд температур для i -го блока. $D^{(k)}$ — временной ряд свойств фичи под номером k из тренировочного датасета.
- Левый нижний индекс в квадратных скобках нумерует временной лаг, на который записи фич для данной сущности сдвинуты во времени в будущее. Таким образом, $_{[k]}T_n = T_{n+k}$
- Левый верхний индекс соответствует номеру аугментированной копии ряда (см. определение ниже), где нулевому номеру соответствует исходный ряд. То есть $^0D = D$, а lD — l -ая копия датасета.

3.2.1 Подготовка данных

Как уже было отмечено выше, M принимает значения равномерно в множестве $\{0, 1, 2, 3, 4\}$, а T колеблется гомоскедастично в отрезке $[18.40, 65.14]$ градусов. В силу очевидных технических особенностей NN лучше обучается, если диапазон значений входных данных соответствует множеству значений функции активации нейронов на выходном слое. В этой связи оказалась разумным совершить сдвиг и масштабное преобразование данных: на первом этапе из M и всех $T^{(k)}$ были вычтены соответствующие средние значения; далее все положительные и отрицательные значения из M были "растянуты" до 1.0 или -1.0 соответственно, ряды $T^{(k)}$ подверглись такому же преобразованию, однако масштабное преобразование было проведено в отрезок $95\% \times [-1.0, 1.0]$, что позволило оставить 5%-ный запас для пороговых значений ряда температур. Такие преобразование может показаться несколько контринтуитивным, однако оно показало себя лучше чем линейный, логарифмический и экспоненциальный мапы исходного множества значений в указанные выше целевые отрезки.

Отдельно пришлось решать проблему сравнительно короткого тренировочного датасета. В рамках подхода к решению задачи предсказания непосредственно рядов температур мы использовали следующую стратегию аугментации данных: рассматриваем некоторую временную последовательность значений выбранного свойства $D^{(n)}$ из датасета после масштабного преобразования, вычисляем наибольшую разность двух последовательных значений данного свойства $\sigma = \max(D_k^{(n)} - D_{k+1}^{(n)})$, на основе данной последовательности $D^{(n)}$ строим $10^0 - 10^4$ последовательностей $^lD^{(n)}$, которые получены добавлением к каждому её члену случайной величины, распределение которой представляет собой белый шум с нулевым средним и дисперсией $\frac{\sigma}{2}$. Во время обучения нейросетевых моделей мы последовательно увеличивали число аугментированных копий датасета контролируя растущую степень переобучения на валидационных данных.

3.2.2 Извлечение свойств из ряда

Наибольший интерес для нас представляет достижение рядом температур пороговых значений. Исходные временные ряды имели выраженную "нежелательную" высокочастотную составляющую на фоне целевых циклов нагрева и остывания. Все наши попытки извлечь свойства из рядов температур так или иначе были связаны с попытками

подавить высокочастотную часть сигнала. На этом этапе мы пропускали временной ряд температур через [рекурсивный фильтр Калмана](#), и вычисляли скользящее среднее с различными длинами окон от 2 до 144 отсчетов.

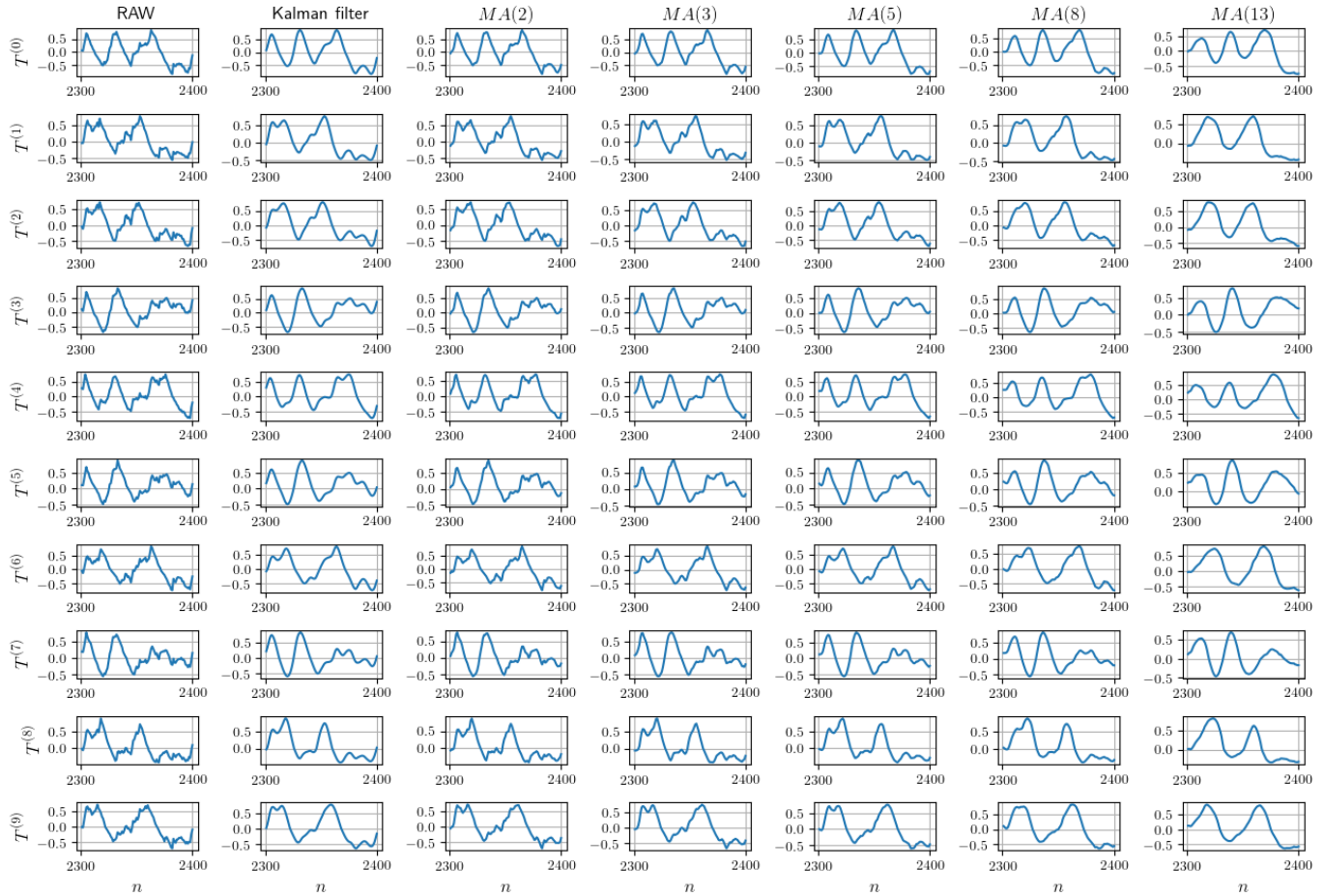


Рис. 19: Временные ряды температур блоков и некоторые производные ряды свойств

3.2.3 Результаты предсказаний и некоторые итоги

Наилучшее качество моделей удалось получить при следующем подходе к процессу обучения:

- Отрезок временных меток датасета разделяется на две области: с начала и до %80 длины для тренировки и оставшиеся %20 для валидации. Последние 120 отсчетов игнорируются полностью для последующей проверки качества работы на данных "в которых гарантированно не может быть никаких прямых или косвенных утечек в датасет, на котором учатся модели".
- Случайно выбирается временная метка и номер аугментированной копии датасета. Формируется последовательность последовательностей из отрезка несколько предыдущих временных отчетов рядов свойств (не менее 120 и не 512 предшествующих данной временной метке значений) из данной копии датасета и определяется температура для заданного блока через заданное количество отсчетов во времени в будущее относительно выбранной временной метки. Такая процедура повторяется batch size раз и формируется оптимальный для обучения на нескольких имеющихся в нашем распоряжении GPU входной и выходной тензор для тренировки модели. "Нарезка" валидационной части датасета проходила аналогичным образом.
- Модель обучается предсказывать целевой ряд температур для заданного блока на один отсчет вперед. Полученные веса используются в качестве инициализации при обучении на сдвинутом на два отсчета целевой переменной. Такой процесс повторяется необходимое количество раз и модель последовательно учится предсказывать целевой ряд температур со все большим временным лагом.
- Целевым функционалом ошибки является незначительно модифицированный логарифм гиперболического косинуса от сумм почленных разностей целевой и предсказанной последовательностей: нескольким первым (от 1 до 15) отсчетом выходной последовательности температур ассоциирован нулевой вес, а отсчетам с конца последовательности (которые являются содержательной частью предсказания) поставлен в соответствие вес в 10.0 больший, чем промежуточным.

К сожалению, следует признать недостаточную для дальнейшего использования точность предсказания рядов температур. Все десятки обученных моделей оказались не способны уверенно предсказывать даже крупномасштабное поведение рядов. В этой связи, подход с многоступенчатым решением с промежуточным предсказанием рядов температур предлагается объявить неудачным и работу в этой области прекратить.

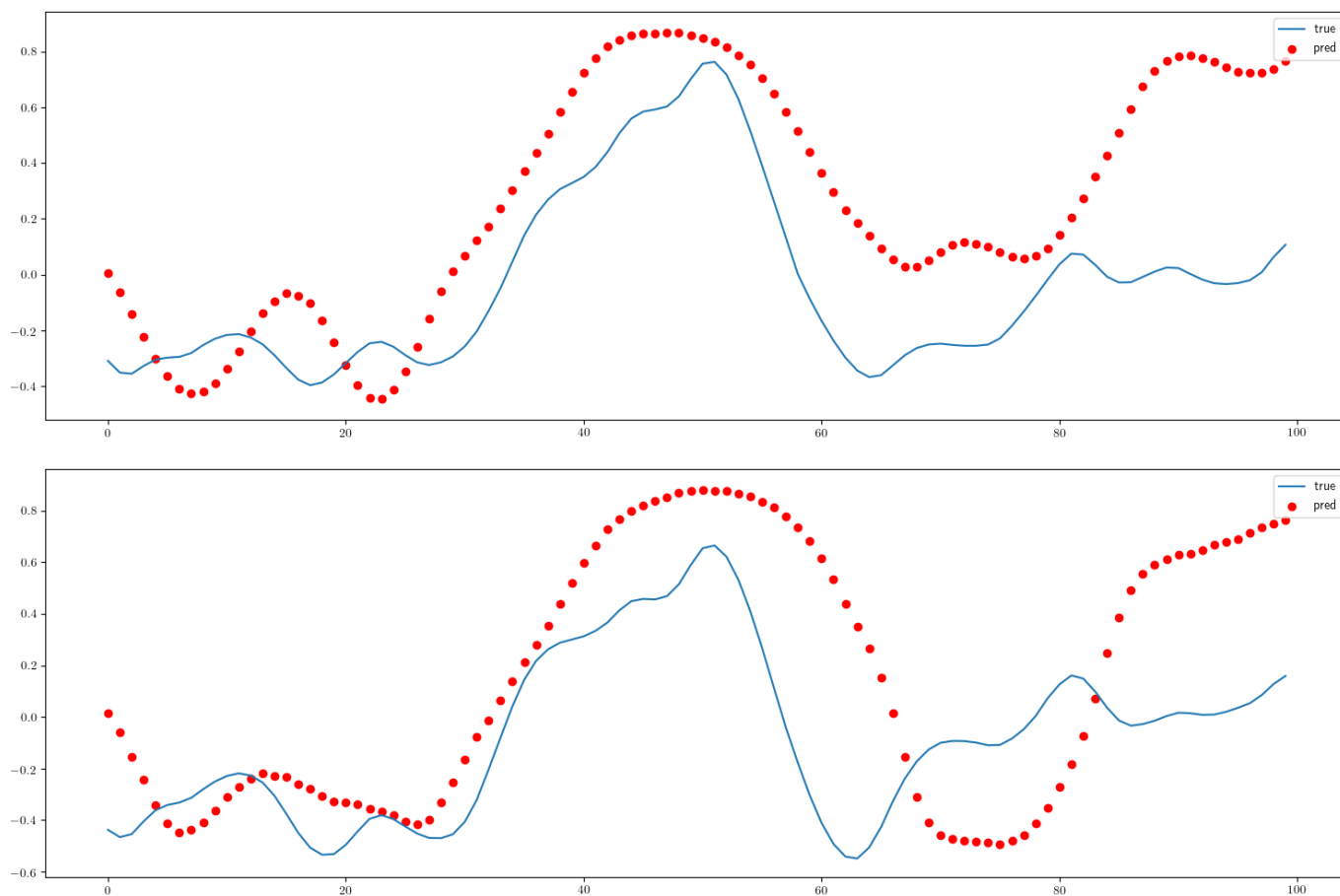


Рис. 20: Пример: Результаты предсказаний для 5-го (верхнее изображение) и 6-го (нижнее изображение) блоков. Сравнение результатов работы моделей, которые обучались предсказывать последовательность температур блоков на 20 отсчетов вперед (красные точки) и реальные значения из валидационной области, которые отмечены непрерывной синей кривой.

4 Предсказание ряда состояний блоков

После очевидного провала подхода с предсказаниями рядов температур и последующего извлечение из них предполагаемого состояний блоков и системы в целом, мы решили радикально изменить идею решения. Будем строить алгоритм, который на основе извлеченных из датасета данных стоит предсказание сразу для бинарной последовательности ряда состояний блоков.

Более того, сведём проблему предсказания бинарного ряда, фактически, к задаче бинарной классификации: построим модель, которая при фиксированных номере блока и значении временного отсчета "в будущее" на основе известного отрезка последовательности свойств из датасета предсказывает состояние блока. Будем называть такую модель элементарной. Таким образом, для осуществления предсказания для 10 блоков на 20 отсчетов "в будущее" потребуется составная модель из 200 элементарных.

4.1 Модель с глубокими нейронными сетями

В рамках нейросетевого подхода все элементарные модели будут иметь схожую архитектуру и для них будут доступны всего три очевидных гиперпараметра: номер блока, номер отсчета на который нужно предсказывать состояние в будущем, длина последовательности временных отсчетов, которую модель принимает на вход.

4.1.1 Подготовка данных для элементарной модели

Попробуем извлечь наиболее "важные" свойства из датасета, которые могут быть связаны именно с текущей элементарной моделью. Методом проб и ошибок было выявлено, что наилучшим набором рядов свойств для элементарной модели является: преобразованный фильтром Калмана ряд значений температур для текущего блока; оценка фильтром Калмана для скорости изменения температуры блока; среднее арифметическое взвешенное температур блоков взятое с весами, равными значению матрицы кросс-корреляции между рядом температур для ряда температур текущей элементарной модели и всех остальных блоков; несколько компонент скользящего среднего (в окне длиной 13, 55, 144 отсчета) для значений фильтрованного Калманом ряда температур для текущего блока; время в отсчетах с момента прошлого отключения блока; бинарный ряд состояний блока; one-hot encoded значение категориальной переменной Mode (5 бинарных рядов).

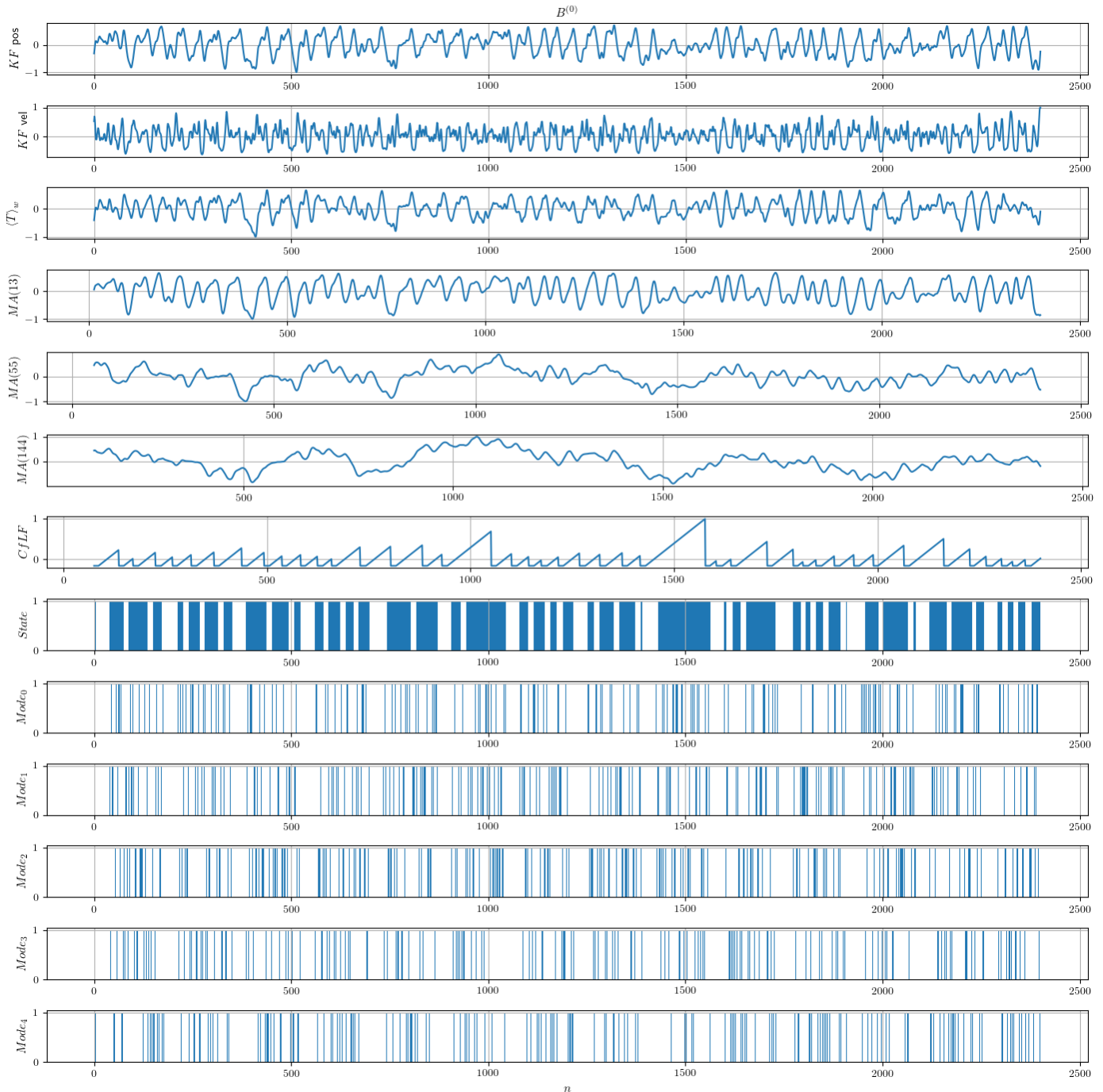


Рис. 21: Набор данных для моделей для первого блока

Над всеми числовыми свойствами было произведено масштабное преобразование и сдвиг: из каждого ряда свойств вычиталось соответствующее среднее значение и полученный ряд с нулевым средним "растягивался" таким образом, чтобы его значения по модулю не превосходили 1.0.

Аугментация данных проводилась аналогичным образом, как в случае построения моделей для предсказания температур: ко всем не бинарным свойствам в добавлялись значения случайной нормально распределенной величины с нулевым средним и дисперсией не превосходящей половину разности двух подряд идущих значений целевого ряда. Всего использовалось до 50 полученных таким способом копий исходного датасета для обслуживания каждой из элементарных моделей. Вопрос переобучения на вырожденных данных контролировался отдельно.

4.1.2 Архитектура элементарной модели и особенности подхода к обучению

В качестве универсальной элементарной модели в рамках нашего построения удовлетворительно себя показала относительно сложная нейросетевая модель. Используемая модель имеет пару входных слоёв, пару выходных и две различные целевые функции — её можно условно разделить на рекуррентную и последовательную часть.

Рекуррентная часть принимает на вход последовательность последовательностей отрезков временных рядов отсчетов заданной длины (в нашем случае 300 или 400 прошлых отсчетов по отношению к текущему обрабатываемому моменту времени) и на выходе имеет значение температуры заданного блока в текущий момент времени, который является "базой" предсказания. Данная рекуррентная подсеть представляет собой входной полносвязный слой; первый слой из LSTM ячеек, который возвращает последовательность всех cell state LSTM ячеек для входной временной последовательности; второй слой из LSTM ячеек, который возвращает только верхнее состояние со стека cell state LSTM-ов; выход со второго LSTM слоя соединён с оставшейся частью основной сети и с выходом рекуррентной подсети.

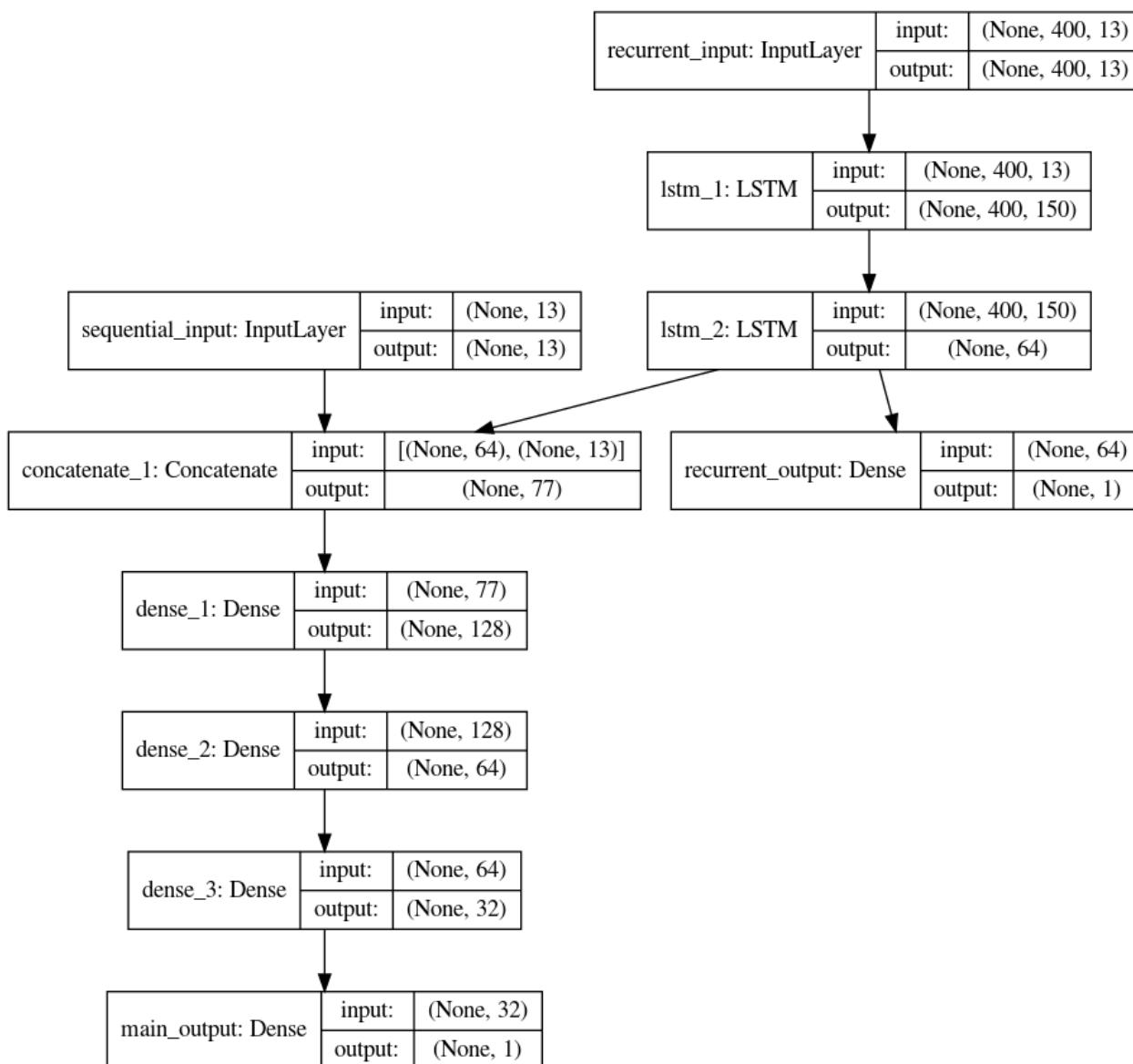


Рис. 22: Архитектура элементарной сети. "None" — предварительно не специфицированный размер batch size, "13" — число различных временных рядов свойств для заданной модели, "400" — длина входного ряда последовательности для рекуррентной подсети.

Вектор выхода со второго LSTM слоя конкатенирован с "обычным" полносвязным слоем входа "последовательной" части нейросети, который принимает в себя последнее значение из рядов свойств ряда в момент времени, который считается "базой" предсказания. Таким образом, рекуррентная подсеть просто ставит в соответствие входному тензору отрезка временного ряда "локальный" во времени многомерный вектор свойств, который характеризует в целом весь отрезок временного ряда свойств за выбранное количество отсчетов, предшествующих "базовому". Данная часть архитектуры срисована с одного из блоков нейросетей, традиционно применяющихся при машинном переводе.

Полученный композитный вектор свойств погружается на стек обычных полносвязных слоёв, число нейронов в которых "пирамидально" уменьшается. Выходной слой последовательной части нейросети, он же основной выход, имеет один единственный нейрон, на выходе которого ожидается иметь численное вероятности — выхода данной элементарной модели.

Итого: последовательный вход принимает тензор вида $D_{n_b}^{(l)}$ — индекс "фичей" l пробирает 13 значений: температура из фильтра Калмана, скорость изменения температуры из фильтра Калмана, среднее арифметическое взвешенное температур оставшихся 9-ти блоков, и т.д., — временной индекс n_b фиксирован и отвечает отсчету времени, являющимся базой предсказания; рекуррентный вход принимает тензор вида $D_{n_b-400:n_b}^{(l)}$ — отрезок временного ряда последовательных значений тензора $D_n^{(l)}$ при $n = \overline{n_b - 400, n_b}$; на рекуррентном выходе формируется скалярная величина — температура соответствующего элементарной модели блока в момент времени, соответствующий отсчету n_b ; на главном выходе формируется скалярная величина $S_{n_b+p}^{(\cdot)}$ — состояние соответствующего элементарной модели блока через p отсчетов времени в будущее относительно n_b .

Целевой функцией для основного выхода была выбрана бинарная кросс-энтропия, а целевой функцией для рекуррентной подсети был выбран функционал логарифма от гиперболического косинуса, как и для моделей для предсказаний температур блоков из прошлой главы. Общим функционалом ошибки для элементарной модели являлась сумма ошибки главного выхода сети с весом 1.0 и ошибка рекуррентной части с весом 0.5.

В качестве значимой особенности следует указать, что аугментация данных в первую очередь использовалась для балансировки классов "0" и "1" в задаче бинарной классификации, а только уже потом увеличения общей длины тренировочного датасета.

Больше технических подробностей о процессе обучения можно найти в [исходном коде](#) скрипта для обучения элементарных моделей.

Полное обучение 210 моделей для построения предсказания на 21 отсчет вперед во времени для 10 блоков на весьма среднем по производительности ПК с двумя GPU gtx 770, даже при очевидно неоптимальном подходе к балансировке нагрузки между задачами обучения элементарных моделей, занимает около 4 астрономических часов.

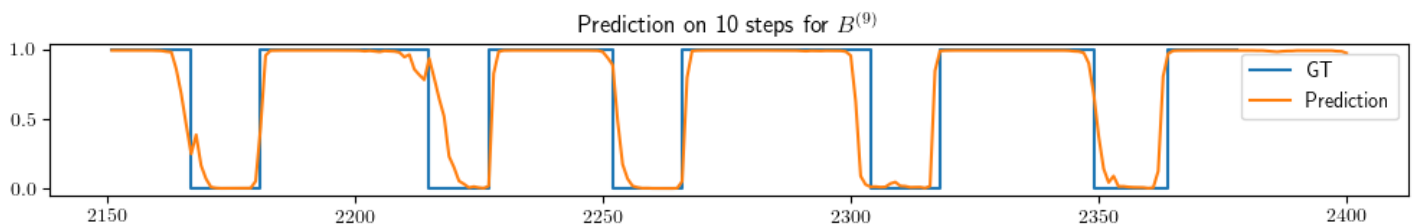


Рис. 23: Сравнение предсказания одной из элементарных моделей с истинными значениями целевой переменной.

Весь исходный код скриптов для подготовки данных, обучения моделей, построения предсказания и наивной постобработки можно найти по [ссылке](#).

4.1.3 Композиция элементарных моделей и итоговое предсказание

Имея обученные модели для всех необходимых пар номер блока—номер отсчета в будущем, совсем тривиально построить итоговое предсказание. Мы бы хотели отметить, что на этом этапе мы применили подход с аугментацией на тестовых данных (Test-Time Augmentation): как и во время обучения, мы передавали на вход моделям не только исходный ряд, но и его копии с добавленным по описанной выше методике шумом — финальным предсказанием модели считалось среднее арифметическое от 50 запусков на эквивалентных входных данных.

Всего было обучено 210 элементарных моделей. Перед запуском на оставленной для окончательной валидации работоспособности модели 60 последних отсчетах из исходного датасета, было вычислено значение предсказаний для всех возможных точек во времени из тестовой области. На основе накопленной статистики были найдены пороговые значения для всех 210 бинарных элементарных классификаторов, которые минимизируют число ошибок, считая теперь выход моделей бинарным.

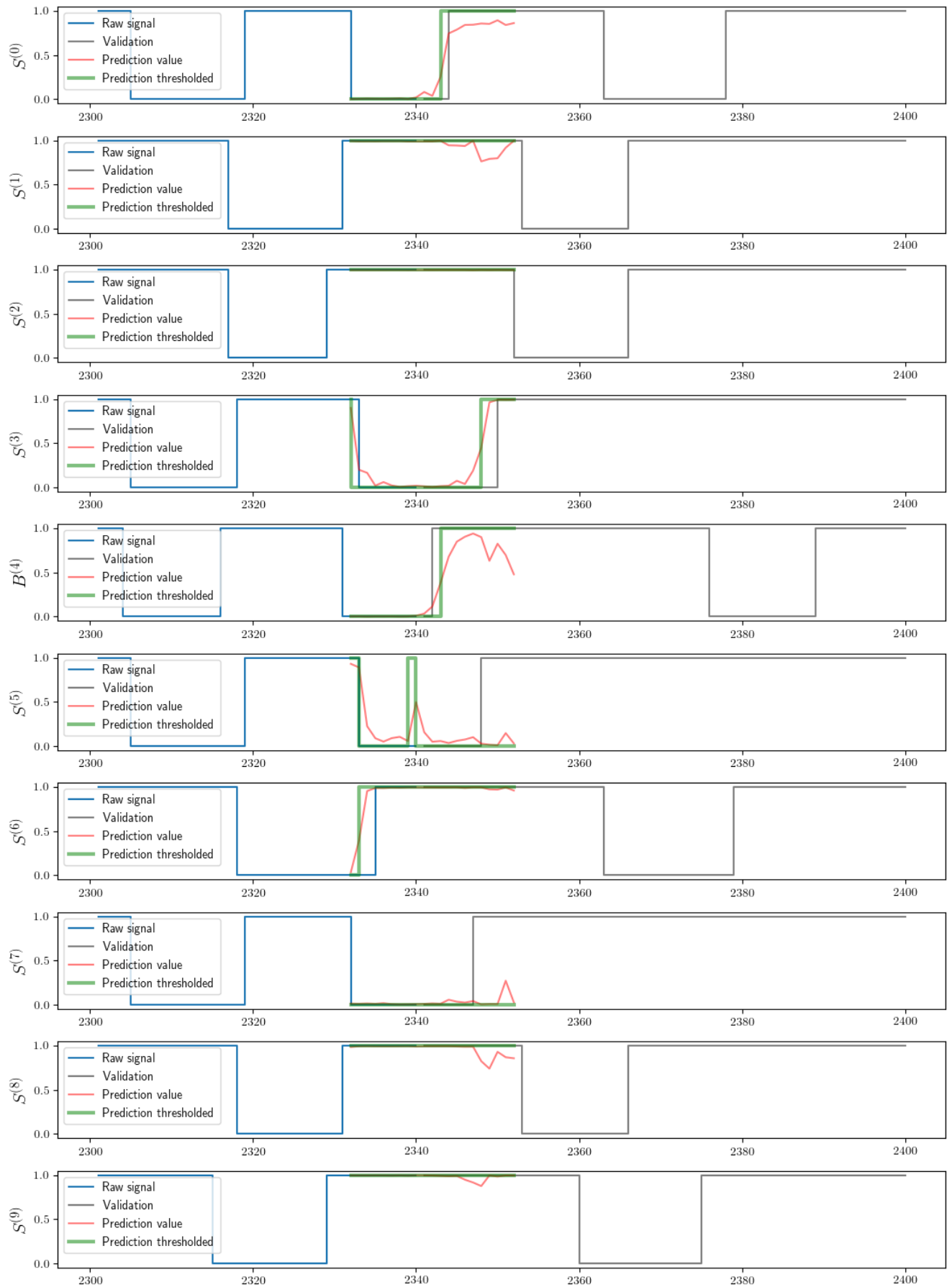


Рис. 24: Сравнение итоговых предсказаний композитной модели с истинными значениями целевых переменных в валидационной области. Различные графики предсказаний, где в качестве базового отсчета используются значения в области с 2330-го по 2400-й можно посмотреть ([тут](#)). Пронаблюдать как именно модель ошибается удобнее всего на анимации ([тут](#)).

5 Вместо заключения

Нам ещё много чего есть сказать по поводу данной задачи и как не реализованных идей, так и банально не вошедших в текст этого короткого отчета, ещё более чем хватает. Тем не менее, тут мы хотим поставить точку. Всё прошедшие с начала "соревнования" дни, мы уделяли время данной задачи по остаточному принципу и в силу самых очевидных обстоятельств ставили её ниже приоритетом, чем учёбу, научную и прочую работу.

Задача в исходной формулировке — со сложным существенно многомерным временным рядом и плохо размеченными данными с реальной установки — действительно представляет интерес. Если предоставится возможность поработать над решением "настоящей" задачи, мы готовы уделить этому своё время и подойти "серьезно".