

Федеральное государственное автономное образовательное  
учреждение высшего образования  
«Южно-Уральский государственный университет  
(национальный исследовательский университет)»

На правах рукописи



**Аботалев Мостафа Салахелдин Абделсалам**

**АЛГОРИТМЫ ПРОГНОЗИРОВАНИЯ ВРЕМЕННЫХ  
РЯДОВ НА ОСНОВЕ ВЗВЕШЕННОГО МЕТОДА  
НАИМЕНЬШИХ МОДУЛЕЙ**

Специальность 1.2.3 — Теоретическая информатика, кибернетика

Диссертация на соискание ученой степени  
кандидата физико-математических наук

Научный руководитель

Т. А. Макаровских

доктор физ.-мат. наук, доцент

Челябинск–2024

# ОГЛАВЛЕНИЕ

<b>Введение</b>	<b>6</b>
<b>1 Современные методы прогнозирования временных рядов</b>	<b>30</b>
1.1 Основные понятия и определения . . . . .	30
1.2 Регрессионные модели . . . . .	36
1.2.1 Модель ARIMA . . . . .	37
1.2.2 Экспоненциальное сглаживание. . . . .	39
1.2.3 Модели BATS и TBATS. . . . .	41
1.2.4 Модель Prophet для одномерных временных рядов. . . . .	44
1.3 Гибридные модели . . . . .	46
1.3.1 Гибридная модель Auto-ARIMA и экспоненциального сглаживания для одномерных временных рядов . . . . .	46
1.3.2 Гибридная модель Auto-ARIMA и полиномиальной регрессии для одномерных временных рядов . . . . .	48
1.4 Модели машинного обучения. . . . .	50
1.4.1 Многослойный перцептрон (MLP) для одномерных временных рядов . . . . .	51
1.4.2 Метод опорных векторов для одномерных временных рядов . . . . .	52
1.4.3 Модель NNAR (Neural Network Autoregressive). . . . .	54
1.4.4 Модели LSTM, BiLSTM, GRU . . . . .	56
1.5 Сравнение рассмотренных моделей. . . . .	59

1.6	Прогнозирование с помощью систем дифференциальных или разностных уравнений . . . . .	61
1.6.1	Классическая эпидемиологическая модель . . . . .	61
1.6.2	Модели на основе разностных уравнений . . . . .	63
1.7	Выводы по главе 1 . . . . .	67
<b>2</b>	<b>Оценка параметров квазилинейного рекуррентного уравнения</b>	<b>68</b>
2.1	Оценка по WLDM . . . . .	69
2.2	Алгоритм GLDM-оценки . . . . .	76
2.3	Прогнозирование . . . . .	80
2.4	Анализ коэффициентов модели . . . . .	83
2.4.1	Выбор оптимального порядка модели для заданного временного ряда. . . . .	84
2.4.2	Алгоритм выявления значимых коэффициентов модели . . . . .	86
2.4.3	Обоснование. . . . .	88
2.4.4	Сходимость, устойчивость и согласованность . . . . .	91
2.5	Выводы по главе 2 . . . . .	94
<b>3</b>	<b>Система для прогнозирования одномерных временных рядов</b>	<b>96</b>
3.1	Epidemic.TA для прогнозирования одномерных временных рядов . . . . .	97
3.2	Система прогнозирования с использованием методов глубокого обучения. . . . .	100

3.3	Объединение моделей временных рядов с помощью оптимизатора весов для моделей временных рядов . . . . .	103
3.4	Программное обеспечение для одномерных временных рядов	104
3.5	Возможности работы алгоритма определения параметров квазилинейного рекуррентного уравнения для больших данных . . . . .	106
3.5.1	Возможности функционального параллелизма при определении параметров квазилинейного уравнения .	107
3.5.2	Возможности параллелизма по данным при определении параметров квазилинейных рекуррентных соотношений. . . . .	112
3.6	Выводы и результаты по главе 3 . . . . .	118
<b>4</b>	<b>Вычислительные эксперименты</b>	<b>120</b>
4.1	Параметры квазилинейного рекуррентного уравнения, описывающего динамику вегетационного индекса . . . . .	121
4.1.1	Описание исходных данных и способов их представления. . . . .	122
4.1.2	Моделирование динамики вегетационного индекса и анализ прогнозирования урожайности . . . . .	124
4.1.3	Анализ значимости коэффициентов модели для разных порядков уравнения . . . . .	130
4.2	Вычислительный эксперимент для анализа значимости коэффициентов модели при анализе данных о температуре . .	133
4.3	Вычислительный эксперимент для анализа значимости коэффициентов модели при анализе данных о скорости ветра	

4.4	Вычислительные эксперименты для временных рядов с данными по заражениям Covid-19 . . . . .	142
4.5	Влияние длины временного ряда на выбор порядка модели .	152
4.6	Прочие вычислительные эксперименты . . . . .	154
4.7	Выводы и результаты по главе 4 . . . . .	156
	<b>Заключение</b>	<b>157</b>
	<b>Список литературы</b>	<b>161</b>

## ВВЕДЕНИЕ

**Актуальность темы.** В настоящее время накоплен большой опыт измерения вибрационных сигналов, разработки методов вибродиагностики и прогнозирования состояния и ресурса механических систем, а также анализа больших объемов данных, описывающих различные промышленные, социальные, экономические и прочие процессы. В связи с этим одним из наиболее актуальных направлений исследований является повышение точности и скорости определения диагностических признаков. Прежде всего, это относится к уникальным высоконагруженным механическим системам, рассмотренным, например, в [99]. Решение этой задачи во многих случаях можно получить через динамические характеристики механических систем. Определению этих характеристик в значительной степени способствует правильный выбор диагностической математической модели, устанавливающей связь между пространством состояний объекта и пространством диагностических признаков. К ним относятся динамические модели, представленные в виде разностных уравнений, феноменологические, структурные, регрессионные модели, модели машинного обучения и т.п. Выбор модели зависит от определяемых характеристик и характера анализируемого процесса.

Идентификация с помощью различных статистических методов, нейронных сетей или математических моделей применяется достаточно давно в разных сферах деятельности. В настоящее время эти методы используются не только в промышленности, но и при попытках прогнозирования развития пандемии Covid-19 и прочих социально-значимых процессов. Например, в [34] приводятся сравнения качества прогнозирования процесса пандемии по разным известным моделям, описывается созданное программное обеспечение для проанализированных методов и

проводятся вычислительные эксперименты с временными рядами данных о распространении Covid-19. Авторы приходят к выводу, что их система прогнозирования может быть реализована для любых временных рядов, описывающих любые (не только эпидемиологические) процессы.

В настоящее время большинство прогнозов, особенно для больших объемов данных, проводятся с использованием различных моделей нейронных сетей. Например, в [118] рассматривается нейросетевая модель, с помощью которой можно прогнозировать изменение цены ферросилиция на внутреннем рынке РФ в краткосрочной перспективе. Предлагаемая модель отличается высокой точностью прогнозирования и может быть полезна при обосновании стратегических решений в деятельности отраслевых НИИ и металлургических предприятий. В [128] описаны эконометрические модели качественного экономического показателя металлургической отрасли, производства, применимые для оценки статистических характеристик производства черных металлов и перспектив развития черной металлургии. Отдельно можно выделить модели, используемые в области точного земледелия, например, для прогнозирования урожайности сельскохозяйственных угодий. Тем не менее, все подобные модели не позволяют описать исследуемый процесс при помощи уравнений, и полученный в результате применения этих моделей результат имеет достаточно узкую область применения. Ряд исследователей используют так называемое когнитивное моделирование для повышения качества прогнозирования с помощью нейронных сетей. В качестве одного из примеров можно рассмотреть статью [92], которая направлена на сравнение эффективности когнитивных и математических предикторов временных рядов с точки зрения точности. Авторы обнаруживают, что результаты их эксперимента показали, что когнитивные модели имеют

по крайней мере эквивалентную точность по сравнению с известными и хорошо исследованными моделями ARIMA. Большинство подобных подходов используются для прогнозирования некоторых экономических единиц, таких как объем производства, некоторые параметры логистики и т.д.

Стоит отметить, что все перечисленные выше модели работают на краткосрочное прогнозирование. Следовательно, **актуальна задача** разработки математического подхода, позволяющего в явном виде получать качественные квазилинейные разностные уравнения (адекватно описывающие рассматриваемый процесс) для среднесрочного и долгосрочного прогнозирования. Известны некоторые исследования в этой области, например, [98], где предлагаемая модель включает очистку данных, сглаживание данных и окончательные данные после предварительной обработки, которые вводятся в регрессионную модель для прогнозирования промышленного потребления электроэнергии. Но в этой статье, как и во многих других, снова рассматриваются статистические методы.

### **Степень разработанности темы исследования**

Первые исследования в области статистического анализа данных датируются 1927 г. [130], а развитие методов прогнозирования имеет еще более длинную историю. В середине 1970-х гг. была популяризирована модель авторегрессионного интегрированного скользящего среднего (ARIMA) с использованием процедуры итеративного моделирования, состоящей из идентификации, оценки и проверки модели. Если модель ARIMA построена и признана адекватной, прогнозы будущих значений являются просто условными ожиданиями модели, если в качестве критерия используется минимальная среднеквадратическая ошибка. Тем не менее, большинство современных публикаций посвящено не столько раз-

виту и улучшению прогностических свойств модели в общем случае, сколько ее использованию в ряде практических ситуаций (прогнозирование различных социальных явлений [129], динамики цен [37], природных явлений [74] и прочих зависимостей).

Для анализа сложных закономерностей в [50] рассматриваются различные модификации моделей пространства состояний для экспоненциального сглаживания для обработки более широкого спектра сезонных закономерностей. Подбор параметров модели осуществляется полностью автоматически. Это позволяет модели улавливать постепенно изменяющуюся сезонность путем введения в модель комбинаций из рядов Фурье. Одним из недостатков этой модели является низкая скорость вычислений [51, 57, 67, 112].

Важной движущей силой будущих исследований в области анализа временных рядов является прогресс в сборе больших объемов данных в различных сферах деятельности. Открытыми являются задачи сбора и хранения выборок большого объема, тяжелые хвосты, неравномерно распределенные наблюдения и комбинации многомерных дискретных и непрерывных переменных, возникновение т.н. «черного лебедя» (события с нулевой дисперсией и математическим ожиданием) могут легко сделать существующие методы неадекватными. Таким образом, актуальной задачей является разработка новых моделей, использующих другие уравнения для управления эволюцией условных моментов более низкого порядка. Решению данной задачи для практических целей (эпидемиология, финансы) посвящены, например, работы [55, 60, 123].

В современных исследованиях и при моделировании сложных закономерностей зачастую использование одной модели не является эффективным, по этой причине разрабатываются различные гибридные мо-

дели, объединяющие в себе два и более методов. Наиболее популярны гибридные модели, в которых совмещаются только регрессионные подходы (один из которых – auto-ARIMA с автоматически определяемыми параметрами), либо регрессионные модели и нейросетевые методы [47, 79, 94, 97, 107, 120, 124, 129]. Наибольшую популярность здесь получили гибриды регрессионных моделей с моделями машинного обучения [5].

Отдельно хотелось бы отметить подход к анализу временных рядов, основанный на использовании квазилинейных рекуррентных соотношений [102, 103], развитию которого посвящена данная диссертационная работа. Существуют и другие методы, связанные с использованием детерминированных алгоритмов для подбора параметров модели [15]. В отличие от нейронных сетей этот подход позволяет в явном виде получать качественные квазилинейные разностные уравнения (адекватно описывающие рассматриваемый процесс).

### **Цели и задачи исследования**

Целью диссертационного исследования является разработка методов определения коэффициентов моделей для прогнозирования процессов, представленных в виде дискретных временных рядов.

Для достижения указанной цели необходимо решить следующие задачи.

1. Разработать метод квазилинейного детерминированного анализа моделей нелинейной динамики для решения задачи прогнозирования развивающихся процессов. В качестве целевой функции для минимизации невязок использовать функцию, повышающую устойчивость модели к выбросам, что позволит получить более точные прогнозы.
2. Разработать систему прогнозирования временных рядов с возмож-

ностью подключения дополнительных модулей. Система должна давать возможность вывода значений погрешностей прогнозирования для дальнейшего выбора экспертом лучшей модели для использования.

3. Провести вычислительные эксперименты для рассмотренных в работе математических моделей с применением разработанной системы прогнозирования.

**Методология и методы исследования.** В работе развиваются методы регрессионного анализа динамики наблюдаемых процессов с применением статистических методов, методов построения нейронных сетей, глубокого обучения, линейного программирования, метода наименьших модулей, компьютерного моделирования.

#### **Научная новизна.**

- Разработан новый метод определения коэффициентов разностного квазилинейного уравнения по заданному временному ряду (по наблюдаемым отсчетам) для прогнозирования развивающихся процессов, основанный на методе наименьших модулей. Метод позволяет эффективно минимизировать значение функции потерь.
- Спроектирован и реализован подход к выявлению закономерностей в данных и прогнозирования одномерного временного ряда, позволяющая получить результаты построения прогноза с использованием статистических, нейросетевых и квазилинейных моделей, проанализировать погрешности прогнозирования и выбрать лучшую модель для работы с конкретными исходными данными.

**Теоретическая значимость** результатов обусловлена решением актуальных задач обнаружения закономерностей развивающихся процессов с применением современного математического аппарата. При

этом возможность учета нелинейности позволяет использовать полученные модели в различных областях.

**Практическая ценность** заключается в том, что разработанные Алгоритмы реализованы в виде программного обеспечения для прогнозирования временных рядов различными методами и проведения сравнительного анализа результатов работы реализованных алгоритмов и выбора лучших алгоритмов для обнаружения закономерностей в заданном наборе данных. Результаты, изложенные в диссертации, могут быть использованы в прогнозировании социальных, экономических, природных явлений, в медицине, промышленности и пр. Разработанный метод квазилинейного анализа одномерного временного ряда применен для исследования ряда прикладных задач (прогнозирование урожайности, распространение эпидемии, анализ температуры воздуха, прогнозирование скорости ветра).

### **Положения, выносимые на защиту**

1. Предложен метод обобщенных наименьших отклонений (GLDM) для анализа моделей нелинейной динамики при прогнозировании развивающихся процессов. В данном методе функция арктангенса  $\arctan(x)$  используется в качестве функции потерь, что повышает устойчивость модели к выбросам и обеспечивает более точные прогнозы.
2. Разработана система прогнозирования временных рядов с возможностью подключения дополнительных модулей. Система дает возможность вывода значений погрешностей прогнозирования для дальнейшего выбора экспертом лучшей модели для использования.
3. Проведены вычислительные эксперименты на реальных данных,

подтвердившие эффективность разработанных методов и подходов.

**Степень достоверности результатов.** Достоверность научных результатов и выводов исследования определяется корректным использованием современных математических методов, подтверждены доказательствами в соответствии с современным уровнем математической строгости, объемом апробации и представления этапов работы на внутривузовских, всероссийских и международных научных конференциях и семинарах. Результаты и выводы не противоречат ранее полученным результатам других авторов.

**Апробация результатов.** Основные положения диссертационной работы, разработанные методы, алгоритмы и результаты вычислительных экспериментов докладывались на следующих международных и всероссийских научных конференциях.

1. The 7th International conference on Time-series and Forecasting (19-21 июля 2021 г., Испания, г. Гран-Канария).
2. Computer Science Protecting Human Society Against Epidemics. ANTICOVID 2021. IFIP Advances in Information and Communication Technology (28-29 июня 2021 г., Польша, г. Варшава).
3. International Conference on Information Technology and Nanotechnology (ITNT) (20-24 сентября 2021 г., Самара).
4. International Conference on Information Technology and Nanotechnology (ITNT) (23-27 мая 2022 г., Самара).
5. Mathematical Optimization Theory and Operations Research: Recent Trends. MOTOR (05-10 июля 2021 г., Иркутск).
6. Mathematical Optimization Theory and Operations Research: Recent Trends. MOTOR (02-06 июля 2022 г., Петрозаводск).

7. Mathematical Optimization Theory and Operations Research: Recent Trends. MOTOR (02-08 июля 2023 г., Екатеринбург).
8. Mathematical Optimization Theory and Operations Research: Recent Trends. MOTOR (30 июня-06 июля 2024 г., Омск).
9. Параллельные вычислительные технологии (ПаВТ) (29-31 марта 2022 г., Дубна).
10. Параллельные вычислительные технологии (ПаВТ) (28-30 марта 2023 г., Санкт-Петербург).
11. XIV Международная конференция «Оптимизация и приложения» (ОРТИМА), (26-30 сентября 2022 г., Черногория, г. Петровац).
12. XV Международная конференция «Оптимизация и приложения» (ОРТИМА), (18-22 сентября 2023, Черногория, г. Петровац).
13. 4th International Scientific Conference of Alkafeel University (ISCKU 2022), (20-21 декабря 2022 г., Ирак, г. Эн-Наджаф).
14. Цифровая индустрия: состояние и перспективы развития 2023 (ЦИСП'2023), (21-23 ноября 2023 г., Челябинск).
15. 8th International Conference on Emerging Applications of Information Technology (EAIT 2024) (12-13 января, 2024 г., Индия, г. Калькутта).
16. The 3rd IEEE International Conference on Electronic Engineering (ICEEM2023) (07-08 октября 2023 г., Египет, г. Минуф).
17. International Conference on Trends in Quantum Computing and Emerging Business Technologies (TQCEBT 2024) (22-23 марта 2024 г., Индия, г. Пуна).

**Публикации.** Основные результаты диссертации опубликованы в 7 научных работах, в том числе 3 статьи в журналах, входящих в категорию К2 перечня ВАК [1–3], 2 статьи в журналах, входящих в квинтиль

Q1 наукометрической базы Scopus [4, 7], и 1 статья в журнале, входящем в квартиль Q2 наукометрической базы Scopus [5]. Получено 5 свидетельств Роспатента о государственной регистрации программ для ЭВМ [8–12].

#### Публикации в журналах из списка ВАК

1. Аботалев М. С. А. Установление значимости коэффициентов квазилинейного уравнения N-факторной авторегрессии // Проблемы информатики. – 2024. – № 3. – С. 5–28. DOI: 10.24412/2073-0667-2024-3-5-28 (Перечень ВАК К2).
2. Макаровских Т. А., Аботалев М. С. А. Автоматический подбор параметров модели ARIMA для прогноза количества случаев заражения и смерти от Covid-19 // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. – 2021. – Т. 10, № 2. – С. 20–37. DOI: 10.14529/cmse210202 (Перечень ВАК К2, RSCI).
3. Аботалев М. С. А., Макаровских Т. А., Панюков А. В. Возможности параллелизма при идентификации квазилинейного рекуррентного уравнения // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. – 2023. – Т. 12, № 4. – С. 94–109. DOI: 10.14529/cmse230404 (Перечень ВАК К2, RSCI).

#### Публикации, индексируемые в Web of Science и Scopus

4. Abotaleb M. Solving The Optimizing Parameters Problem For Non-linear Datasets Using The High-Order General Least Deviations Method (GLDM) Algorithm // Computational Methods for Differential Equations. – 2024. DOI: 10.22034/cmde.2024.62441.2751 (Web of Science Q2, Scopus Q1).
5. Abotaleb M. Soft Computing-Based Generalized Least Deviation Method Algorithm for Modeling and Forecasting COVID-19 using

Quasilinear Recurrence Equations // Iraqi Journal For Computer Science and Mathematics. – 2024. – Vol. 5, No. 3. – P. 441–472. DOI: 10.52866/ijcsm.2024.05.03.028 (Scopus Q2).

6. Abotaleb M. Proving Optimal Model Selection and Zero Coefficient Cases in Time Series Forecasting with the Generalized Least Deviation Algorithm // Advanced Mathematical Models & Applications. – 2024. – Vol. 9, No. 3. – P. 475–524. DOI: 10.62476/amma93475 (Scopus Q3).
7. Abotaleb M., Makarovskikh T. Advanced milk production modelling using high-order generalized least deviation method // Modeling Earth Systems and Environment. – 2024. – Vol. 10. – P. 7019–7047. DOI: 10.1007/s40808-024-02142-9 (Web of Science Q3, Scopus Q1).

#### Свидетельства о регистрации программ для ЭВМ

8. Аботалёб М. С. А., Макаровских Т. А. Свидетельство Роспатента о государственной регистрации программы для ЭВМ «Параллельная программа для идентификации параметров квазилинейных рекуррентных соотношений для массива временных рядов» № 2023661570 от 01.06.2023.
9. Аботалёб М. С. А., Макаровских Т. А. Свидетельство Роспатента о государственной регистрации программы для ЭВМ «Библиотека для квазилинейного анализа временных рядов» № 2022669812 от 25.10.2022.
10. Макаровских Т. А., Аботалёб М. С. А. Свидетельство Роспатента о государственной регистрации программы для ЭВМ «Система прогнозирования для одномерных временных рядов» № 2022660664 от 08.06.2022.
11. Макаровских Т. А., Аботалёб М. С. А. Свидетельство Роспатента о государственной регистрации программы для ЭВМ «Библиотека

epidemic.ta для прогнозирования случаев заражения Covid-19» № 2021616747 от 26.04.2021.

12. Аботалёб М. С. А., Макаровских Т. А. Свидетельство Роспатента о государственной регистрации программы для ЭВМ «Univariate time series software Version 1 (Программное обеспечение для одномерных временных рядов Версия 1)» № 2024682644 от 25.09.2024.

**Личный вклад автора.** Содержание диссертации и основные положения, выносимые на защиту, отражают личный вклад автора в опубликованные работы. Работы [1, 4, 5, 6] выполнены автором самостоятельно. В работах [2, 7–12] постановка задачи выполнена научным руководителем Макаровских Т.А.; все полученные результаты принадлежат М.С.А. Аботалёб. В работе [3] Панюкову А.В. принадлежит обзор полученных ранее результатов, Макаровских Т.А. – постановка задачи, все остальные результаты принадлежат М.С.А. Аботалёб.

**Соответствие диссертации паспорту научной специальности.** Тематика, содержание и полученные результаты диссертационного исследования М.С.А. Аботалёб соответствуют следующему направлению исследований паспорта специальности 1.2.3. «Теоретическая информатика, кибернетика»: п. 9. Математическая теория исследования операций.

**Структура и объем работы.** Диссертация состоит из введения, 4 глав, заключения и библиографии. Общий объем диссертации 182 страниц, 22 страницы библиографии.

### **Содержание работы**

**Во введении** приводится обоснование актуальности темы и степень ее разработанности; формулируются цели и задачи; раскрывается новизна, теоретическая и практическая значимость полученных результатов;

формулируется методологическая основа диссертационного исследования; дается обзор содержания диссертации.

**В первой главе** приводится анализ существующих методов прогнозирования, которые можно подразделить на статистические, нейросетевые, а также методы, использующие дифференциальные и разностные уравнения.

Показано, что большинство из известных в настоящее время методов применимы для краткосрочного прогнозирования, поэтому актуальна задача разработки математического подхода, позволяющего в явном виде получать качественные квазилинейные разностные уравнения (адекватно описывающие рассматриваемый процесс). Кроме того, в настоящее время активно исследуются нейросетевые модели и модели глубокого обучения, обладающие одним существенным недостатком, связанным с невозможностью точно описать параметры процесса.

Несмотря на обилие методов прогнозирования временных рядов, среди них не существует универсального наилучшего метода прогнозирования. В течение долгого времени остается актуальным вопрос выбора наиболее подходящего метода прогнозирования имеющегося временного ряда. Данная задача не является тривиальной, поскольку выбор метода зачастую зависит от структуры временного ряда, а также самого процесса, частоты получения новых данных, объема выборки, горизонта прогнозирования и пр.

Поэтому создание единой гибкой системы прогнозирования, осуществляющей автоматический подбор метода прогнозирования является важной задачей. Существующее программное обеспечение позволяет использовать последовательно каждый из методов и не позволяет автоматически получить сводный анализ примененных методов. Также

в систему прогнозирования представляется разумным добавить метод определения параметров квазилинейного рекуррентного уравнения. Поскольку в настоящее время не разработано универсального средства для построения прогнозов, предлагается разработать расширяемую систему прогнозирования, которая способна обрабатывать любые временные ряды произвольной длины и не зависит от характера прикладных задач.

В качестве прототипа взята модель, рассмотренная в работах Панюкова А.В. и Межаала Я.А., в которой рассматривается алгоритм определения коэффициентов  $a_1, a_2, a_3 \dots, a_m \in \mathbb{R}$  квазилинейной авторегрессионной модели  $m$ -го порядка

$$y_t = \sum_{j=1}^{n(m)} a_j g_j(\{y_{t-k}\}_{k=1}^m) + \varepsilon_t, \quad t = 1, 2, \dots, T \quad (1)$$

актуальной информацией о значениях переменных состояния  $\{y_t \in \mathbb{R}\}_{t=1-m}^T$  в моменты времени  $t$ ; где  $g_j : (\{y_{t-k}\}_{k=1}^m) \rightarrow \mathbb{R}$ ,  $j=1, 2, \dots, n(m)$ , функции  $n(m)$  заданы, и  $\{\varepsilon_t \in \mathbb{R}\}_{t=1}^T$  – неизвестные ошибки.

Традиционные модели, используемые в предыдущих исследованиях [100, 103, 104, 136], основываются на квазирекуррентных уравнениях с девятью коэффициентами для захвата нелинейных финансовых трендов. В разработанном подходе применяется обобщенный метод наименьших отклонений (GLDM) первого порядка, который может расширяться до пятого порядка, включая члены высокого порядка для обработки более сложных систем. Алгоритмические шаги в традиционных моделях включают инициализацию фиксированного числа коэффициентов и минимизацию абсолютных ошибок, ориентированных на финансовые данные. В отличие от них, предложенный метод инициализирует коэффициенты в зависимости от порядка модели GLDM и использует функции  $\arctan$  для нелинейных преобразований, что повышает гибкость модели.

В аспекте оптимизации и сложности традиционные подходы фокусируются на минимизации ошибок без применения функций  $\arctan$ , что обеспечивает высокую специализацию для финансовых данных, но ограничивает гибкость моделей при применении к другим областям. Предлагаемый подход, напротив, представляет собой универсальный метод с взвешенными ошибками и нелинейными корректировками, что делает его применимым в различных сферах, однако увеличивает вычислительные затраты.

С точки зрения сильных и слабых сторон, традиционные модели демонстрируют высокую эффективность при работе с финансовыми данными, но подвержены риску переобучения и обладают ограниченной адаптивностью. Предлагаемый подход обеспечивает большую универсальность и устойчивость к шумам, однако требует значительных вычислительных ресурсов из-за своей сложности.

Таким образом, предлагаемая модель превосходит традиционные подходы по универсальности и устойчивости, что расширяет возможности анализа и прогнозирования сложных систем, несмотря на повышенные вычислительные затраты, делая ее перспективным инструментом для применения в различных областях исследования временных рядов. Отдельно рассматривается подход, основанный на квазилинейных рекуррентных соотношениях, который позволяет явно получать качественные квазилинейные разностные уравнения, адекватно описывающие рассматриваемый процесс. В отличие от нейронных сетей, данный метод обеспечивает более прозрачное моделирование и интерпретацию результатов.

**Во второй главе** рассмотрен алгоритм определения и оценки параметров квазилинейного рекуррентного уравнения. Оригинальность дан-

ного подхода заключается в том, что он вместо метода наименьших квадратов использует обобщенный метод наименьших модулей с подбором весов. Задача (1.32), т.е. задача GLDM-оценки, является задачей многоэкстремальной оптимизации. GLDM-оценки устойчивы к наличию корреляции значений в  $\{y_t \in \mathbb{R}\}_{t=1-m}^T$  и (при соответствующих настройках) лучше всего подходят для вероятностных распределений ошибок с более тяжелыми (чем у нормального распределения) хвостами.

Оригинальность рассмотренного алгоритма определения параметров квазилинейного рекуррентного уравнения заключается в том, что он вместо метода наименьших квадратов использует обобщенный метод наименьших модулей с подбором весов. Показано, что задача GLDM-оценки, является задачей многоэкстремальной оптимизации, и GLDM-оценки устойчивы к наличию корреляции значений в  $\{y_t \in \mathbb{R}\}_{t=1-m}^T$  и (при соответствующих настройках) лучше всего подходят для вероятностных распределений ошибок с более тяжелыми (чем у нормального распределения) хвостами.

Установленные результаты позволяют свести задачу определения оценки GLDM к итерационной процедуре с оценками WLDM (см. алгоритм 1).

Алгоритм 1 **GLDM-estimator** используется для вычисления оптимального решения обобщенного метода наименьших отклонений (GLDM). Этот алгоритм работает итеративно, с пересчетом весов и остатков на каждом шаге до тех пор, пока решение не стабилизируется.

### **Шаги работы алгоритма:**

**Шаг 1: Инициализация весов** - Алгоритм инициализирует все весовые коэффициенты как равные 1 для всех  $t$ :

$$p \leftarrow \{p_t = 1 : t = 1, 2, \dots, T\}$$

- Затем вычисляется начальное приближение решения с помощью вспомогательной функции **WLDMSoluter**, которая принимает в качестве входных данных матрицу  $S$ , проекцию градиента  $\nabla$ , веса  $p_t$  и значения переменных состояния  $y_t$ . Это возвращает начальные коэффициенты  $A^{(1)}$  и остатки  $z^{(1)}$ .

**Шаг 2: Обновление весов** - После получения начальных остатков  $z^{(1)}$ , весовые коэффициенты пересчитываются для каждого временного шага по следующей формуле:

$$p_t \leftarrow \left( 1 / \left( 1 + (z_t^{(1)})^2 \right) \right)$$

для всех  $t = 1, 2, \dots, T$ . Это обновление позволяет учесть величину ошибок и адаптировать веса в зависимости от величины остатка.

**Шаг 3: Повторное решение с обновленными весами** - После пересчета весов алгоритм снова вызывает **WLDMSoluter**, чтобы вычислить новые коэффициенты  $A^{(2)}$  и остатки  $z^{(2)}$ . - Счетчик итераций  $k$  устанавливается равным 2.

**Шаг 4: Итеративный процесс** - Алгоритм выполняет итерации до тех пор, пока решения на двух последовательных шагах не станут одинаковыми, т.е.  $A^{(k)} = A^{(k-1)}$ . - На каждой итерации весовые коэффициенты обновляются на основе текущих остатков:

$$p_t^{(k)} \leftarrow \left( 1 / \left( 1 + (z_t^{(k)})^2 \right) \right)$$

- Алгоритм снова вызывает **WLDMSoluter**, чтобы вычислить новые значения  $A^{(k+1)}$  и  $z^{(k+1)}$ , и увеличивает счетчик  $k$  на единицу.

**Шаг 5: Завершение** - Алгоритм завершает работу, когда на двух последовательных шагах изменения в решении становятся незначитель-

ными (т.е.  $A^{(k)} = A^{(k-1)}$ ). - Окончательные коэффициенты  $A^*$  и остатки  $z^*$  принимаются равными значениям, вычисленным на последней итерации:  $A^* = A^{(k)}$ ,  $z^* = z^{(k)}$ .

**Шаг 6: Вывод результатов** - На последнем шаге алгоритм возвращает оптимальные коэффициенты  $A^*$  и остатки  $z^*$ , которые могут быть использованы для анализа и оценки точности модели.

Таким образом, алгоритм **GLDM-estimator** проводит итеративную процедуру оптимизации, где на каждом шаге пересчитываются веса и остатки для достижения более точного решения. Итерации продолжаются до тех пор, пока изменения в решении не стабилизируются.

---

### Algorithm 1 . GLDM-estimator

---

**Require:**

$S = \{S_t \in \mathbb{R}^N\}_{t \in T}$  ▷ Матрица линейного подпространства  $\mathcal{L}$   
 $\nabla_L$  ▷ Проекция градиента на  $\mathcal{L}$   
 $\{p_t \in \mathbb{R}^+\}_{t=1}^T$  ▷ Весовые коэффициенты  
 $\{y_t \in \mathbb{R}^+\}_{t=1-m}^T$  ▷ Значения заданных переменных состояния

**Ensure:**

$A^* \in \mathbb{R}^{n(m)}$  ▷ Оптимальное решение GLDM  
 $z^* \in \mathbb{R}^T$  ▷ Остатки

```

1:  $p \leftarrow \{p_t = 1 : t = 1, 2, \dots, T\}$ 
2:  $(A^{(1)}, z^{(1)}) \leftarrow \mathbf{WLDMSoluter}(S, \nabla_{\mathcal{L}}, \{p_t\}_{t=1}^T, \{y_t\}_{t=1-m}^T)$ ;
3: for all  $t = 1, 2, \dots, T$  do
4:    $p_t \leftarrow \left(1 / \left(1 + (z_t^{(1)})^2\right)\right)$ 
5: end for
6:  $(A^{(2)}, z^{(2)}) \leftarrow \mathbf{WLDMSoluter}(S, \nabla_{\mathcal{L}}, \{p_t\}_{t=1}^T, \{y_t\}_{t=1-m}^T)$ ;
7:  $k \leftarrow 2$ 
8: while  $(A^{(k)} \neq A^{(k-1)})$  do
9:   for all  $t = 1, 2, \dots, T$  do
10:     $p_t^{(k)} \leftarrow \left(1 / \left(1 + (z_t^{(k)})^2\right)\right)$ 
11:   end for
12:    $(A^{(k+1)}, z^{(k+1)}) \leftarrow \mathbf{WLDMSoluter}(S, \nabla_{\mathcal{L}}, \{p_t^{(k)}\}_{t=1}^T, \{y_t\}_{t=1-m}^T)$ ;
13:    $k \leftarrow (k + 1)$ 
14: end while
15:  $z^* \leftarrow z^{(k)}$ ,  $(A^*) \leftarrow A^{(k)}$  ▷ Поиск ограничений
return  $(A^*, z^*)$ 

```

---

**Теорема 1.** Последовательность  $\{(A^{(k)}, z^{(k)})\}_{k=1}^{\infty}$ , построенная по

алгоритму *GLDM-оценки* сходится к глобальному минимуму  $(a^*, z^*)$  задачи (1.32).

Изложенное в данной главе показывает возможность решения задачи определения параметров уравнения с использованием алгоритма **GLDM-оценка**. Алгоритм показывает, что его вычислительная сложность пропорциональна вычислительной сложности алгоритма решения прямой и/или двойственной задачи **WLDM**.

Наличие нулевых коэффициентов в моделях более высокого порядка свидетельствует об эффективности метода исключения факторов, не влияющих на прогнозирование, что упрощает модели и повышает их интерпретируемость. Это исследование подтверждает ключевую роль выбора оптимального порядка построения модели и интерпретации коэффициентов модели в контексте прогнозирования временных рядов.

Проведенные вычислительные эксперименты показывают, что среднее число итераций алгоритма **GLDM-оценка** равно количеству коэффициентов в идентифицированном уравнении. Вычислительная сложность решения практических задач не превышает  $O((n(m))^3 T + n(m) \cdot T^2)$ , таким образом, алгоритм позволяет получить решение задачи за сравнительно короткий срок.

Показано, что в контексте прогнозирования временных рядов с использованием моделей более высокого порядка часто наблюдается, что ряд коэффициентов может быть равен нулю. Данную ситуацию можно объяснить несколькими математическими и статистическими причинами, которые имеют решающее значение при оптимизации точности прогнозирования модели и эффективности вычислений. Обнуление коэффициента можно обосновать математически путем анализа градиента целевой функции по отношению к каждому коэффициенту в контексте

обобщенного метода наименьших отклонений (GLDM).

Выбор оптимального порядка построения модели при прогнозировании временных рядов с использованием обобщенного метода наименьших отклонений (GLDM) имеет решающее значение для обеспечения баланса между сложностью модели и ее способностью обобщаться для прогнозирования новых значений. Приводятся рассуждения, которые описывают методологию определения оптимального порядка  $m$  модели путем минимизации функции потерь. Описан алгоритм выявления значимых коэффициентов модели. Он может быть особенно полезен в тех случаях, когда решающее значение имеет баланс между сложностью и функциональной совместимостью модели и точностью, что позволяет создать оптимизированную модель, которая эффективно отражает существенную динамику данных.

**Теорема 2 [Выбор оптимальной модели].** Пусть  $\{y_t\}_{t=1}^T$  – временной ряд, и функция потерь  $F(\mathbf{a})$  определена как:

$$F(\mathbf{a}) = \sum_{t=1}^T \arctan |y_t - f(\mathbf{a}, \mathbf{x}_t)|,$$

где функция  $f$  зависит от параметров  $\mathbf{a}$ . Тогда оптимальный порядок модели  $m^*$  равен:

$$m^* = \arg \min_m \left( \min_{\mathbf{a}} F(\mathbf{a}, m) \right).$$

Условия, при которых коэффициенты  $a_j$  модели могут быть обнулены без потери точности прогнозирования, приведены в следующей теореме. Если производная  $F(\mathbf{a})$  по коэффициенту  $a_j$  равна нулю на всем интервале, это указывает на то, что изменения  $a_j$  не влияют на значение функции потерь, и коэффициент  $a_j$  может быть исключен из модели.

**Теорема 3 [Условия наличия нулевых коэффициентов модели].** Если для коэффициента  $a_j$ :

$$\frac{\partial F(\mathbf{a})}{\partial a_j} = \sum_{t=1}^T \frac{\partial}{\partial a_j} \arctan |y_t - f(\mathbf{a}, \mathbf{x}_t)| = 0,$$

то  $a_j$  не влияет на функцию потерь  $F(\mathbf{a})$  и может быть обнулен.

Приведенные теоремы описывают основные аспекты обобщенного метода наименьшего отклонения (GLDM) для анализа данных временных рядов. Они определяют метод определения эффективной структуры модели и определения избыточных коэффициентов модели.

**В третьей главе** рассматривается система прогнозирования, включающая в себя реализацию известных статистических и нейросетевых подходов, а также алгоритма определения параметров квазилинейного рекуррентного уравнения. Разработанная система прогнозирования является расширяемой, она допускает подключение дополнительных модулей прогнозирования и включение их в анализ точности работы моделей. Система позволяет анализировать временные ряды для различных прикладных задач.

Для достижения этой цели было разработано три программных модуля.

1. Система прогнозирования **Epidemic.TA**, позволяющая построить прогнозы для одномерного временного ряда с использованием регрессионных моделей, и модели SIR. В состав первой версии системы входила функция, использующая модель SIR, но в более поздних версиях данная функция была исключена, поскольку она имеет низкую прогностическую способность. Вместо SIR используется модель на основе нейронной сети NNAR.
2. Система прогнозирования **DL-system**, использующий методы глу-

бокого обучения для построения прогнозов для одномерного временного ряда;

3. Модуль GLDM, использующий методы определения параметров квазилинейного рекуррентного соотношения.

Все модули реализованы на языках R (модуль `updated Epidemic.TA`), Python (модуль `DL-system`). Модуль GLDM – на языках Python и C++, кроме того, на языке C++ имеется параллельная версия алгоритма, реализованная с использованием библиотеки OpenMP.

Последовательная версия алгоритма определения параметров квазилинейного рекуррентного уравнения с использованием персонального компьютера работает для временных рядов порядка  $10^3$  элементов и требует несколько минут времени. При попытке реализации функционального параллелизма выяснилось, что данный вид параллелизма для данного алгоритма не является эффективным. При задействовании более мощных вычислителей, ускорение отсутствует. С другой стороны, возможен достаточно эффективный параллелизм по данным, что позволяет работать с большим количеством временных рядов даже на персональном компьютере.

Поскольку для запуска разработанного алгоритма необходимо использовать матрицы высокого порядка, полезно распараллелить алгоритм с целью ускорения процесса получения результата. Проведенные вычислительные эксперименты показывают, что при простейших способах распараллеливания рассмотренных алгоритмов имеем ускорение алгоритма в три раза на современных персональных компьютерах при использовании половины из возможных потоков. Тестирование алгоритма на более мощных вычислителях показало не только отсутствие эффекта от распараллеливания для временных рядов длиной до 100 000 значений, но и значительное замедление вычислений за счет накладных

расходов на организацию параллельных областей в каждой из функций. Таким образом, для реализации алгоритма определения параметров одного квазилинейного рекуррентного соотношения нецелесообразно использовать возможности параллельных вычислений.

Исследования возможностей использования параллелизма для алгоритма определения коэффициентов модели с помощью обобщенного метода наименьших модулей для достаточно большого числа коротких временных рядов показали, что можно разработать параллельное персональное приложение для ноутбуков и/или мобильных телефонов, максимально используя ресурсы этих устройств и позволяя получить результат в 5–6 раз быстрее, поскольку разница во времени использования персонального компьютера для расчета больших наборов данных будет иметь важное значение. Учитывая, что новые данные для рассматриваемого в качестве примера динамического процесса получаются один раз в 24–72 часа, то время является пределом для расчетов, и этот предел достигается для временных рядов порядка  $10^7$  с использованием одного потока, и для  $(5 \cdot 10^7)$  объекты с использованием 7–8 потоков персонального компьютера. Этого количества данных достаточно для анализа с помощью персональных компьютеров. Что касается корпораций, использующих больше объектов, то целесообразно использовать ресурсы суперкомпьютеров.

**В четвертой главе** на основе анализа временных рядов с данными из различных областей с помощью обобщенного метода наименьших отклонений (GLDM) был определен оптимальный порядок построения моделей для прогнозирования. Полученные результаты вычислительных экспериментов показывают, что порядок прогнозирующей модели, в значительной степени зависит от характеристик набора данных (характер

данных, лежащая в их основе динамика и наличие нелинейных закономерностей, количество доступных данных).

Результаты проведенных вычислительных экспериментов показали, что в большинстве случаев для воссоздания уравнения, описывающего динамику временного ряда, достаточно использовать модель второго порядка. Тем не менее, для рядов, характеризующихся значительными сезонными колебаниями и автокорреляциями (например, данные о десятилетних колебаниях температуры) модели более высоких порядков дадут более точные результаты.

**В заключении** приведены результаты исследования, даны рекомендации по их применению, намечены перспективы дальнейших исследований.

# ГЛАВА 1. СОВРЕМЕННЫЕ МЕТОДЫ ПРОГНОЗИРОВАНИЯ ВРЕМЕННЫХ РЯДОВ

## 1.1 Основные понятия и определения

Под временным рядом (или рядом динамики) [66] понимают упорядоченную по времени последовательность значений некоторой произвольной переменной величины. Тем самым, временной ряд существенно отличается от простой выборки данных. Каждое отдельное значение данной переменной называется отсчетом (уровнем элементов) временного ряда.

Временные ряды состоят из двух элементов: (1) периода времени, за который или по состоянию на который приводятся числовые значения; (2) числовых значений того или иного показателя, называемых уровнями ряда.

Временные ряды как правило возникают в результате измерения некоторого показателя. Данные временных рядов описывают множество различных явлений во времени. По мере того, как наш мир становится все более технически оснащенным, датчики и системы постоянно выдают огромный поток данных. Это могут быть как показатели (характеристики) технических систем, так и показатели природных, социальных, экономических и других процессов. Типичным примером временного ряда можно назвать биржевой курс, при анализе которого пытаются определить основное направление развития (тенденцию или тренда). В настоящее время используют временные ряды, содержащие информацию о погодных данных, о развитии эпидемии (количество заражений, смертей, выздоровлений, вакцинаций), для анализа информации, поступающей с

датчиков.

Различают два типа временных рядов:

1. **Непрерывные временные ряды:** наблюдения проводятся непрерывно во времени, при этом измеряемые переменные  $Y(t)$  могут принимать только дискретный набор значений.
2. **Дискретные временные ряды:** наблюдения осуществляются только в определенные моменты времени  $t_k = k\Delta t$ , где  $k \in \mathbb{N}$  и  $\Delta t$  – равный промежуток между наблюдениями. В этом случае измеряемая переменная  $Y(t_k)$  может быть и непрерывной, тогда она будет отражать состояние системы в момент времени  $t_k$ .

Выбор типа временного ряда зависит от характера данных и целей анализа, что влияет на выбор методов моделирования и прогнозирования.

Анализ временных рядов представляет собой совокупность математико-статистических методов, предназначенных для выявления структуры временных рядов и их прогнозирования. Основная цель анализа временных рядов заключается в понимании закономерностей изменений во времени. Статистики выделяют несколько ключевых компонентов временного ряда, включая тренды ( $T(t)$ ), циклы ( $C(t)$ ) и нерегулярные колебания ( $I(t)$ ). Когда эти компоненты присутствуют во временном ряде, модели прогнозирования должны учитывать соответствующие закономерности для обеспечения точности предсказаний.

Модели временных рядов обычно предполагают, что временная переменная  $t$  монотонно возрастает ( $t \rightarrow t + 1$ ). Прошлые значения  $Y(t - 1), Y(t - 2), \dots, Y(t - k)$  могут влиять на будущие наблюдения  $Y(t)$ , но не наоборот. Кроме того, наблюдения, расположенные близко

друг к другу по времени, часто демонстрируют более сильную корреляцию, чем более отдаленные ( $\text{Corr}(Y(t), Y(t - k)) > \text{Corr}(Y(t), Y(t - l))$ ) при  $k < l$ ). Как и все данные, временные ряды содержат случайные колебания ( $\epsilon(t)$ ), которые могут маскировать основные закономерности. Для устранения влияния этих случайных факторов применяются методы сглаживания, такие как скользящие средние и экспоненциальное сглаживание, что позволяет более четко выявить тенденции и циклы.

### **Классификация методов анализа временных рядов**

Методы анализа временных рядов можно условно разделить на следующие категории: методы регрессионного анализа, методы машинного обучения, методы, использующие дифференциальные и разностные уравнения, гибридные модели. Независимо от используемого подхода, выявление структуры временного ряда необходимо для построения математической модели того явления, которое является источником анализируемого временного ряда. Построенный прогноз будущих значений временного ряда используется для эффективного принятия решений.

При оценке модели применяют метрики ошибок, позволяющие оценить точность прогнозов и сравнить модели для выбора лучшей. Меньшая ошибка указывает на более точные прогнозы. Для объективности данные делятся на обучающую  $\mathcal{D}_{\text{train}}$  и тестовую  $\mathcal{D}_{\text{test}}$ . Обучающая выборка строит модель, тестовая проверяет предсказания на новых данных. Ошибки на обеих выборках выявляют переобучение и оценивают обобщающую способность модели.

В дальнейшем при оценке качества прогнозов и сравнении моделей будут использованы следующие показатели [68, 116] (табл. 1.1).

Таблица 1.1 – Оценочные показатели при моделировании одномерных временных рядов

Метрика	Полное название	Описание	Достоинства	Недостатки
RMSE	Среднеквадратичная ошибка (Root Mean Square Error)	Равняется квадратному корню из среднеквадратичных различий между прогнозируемыми и фактическими значениями.	<ul style="list-style-type: none"> <li>- Акцентирует внимание на крупных ошибках</li> <li>- Легко интерпретируется в тех же единицах измерения, что и данные.</li> <li>- Широко используется, облегчая сравнение.</li> </ul>	<ul style="list-style-type: none"> <li>- Чрезмерно чувствительна к большим ошибкам.</li> <li>- Не нормализована, что затрудняет сравнение между наборами данных.</li> </ul>
MAE	Средняя абсолютная ошибка (Mean Absolute Error)	Среднее значение абсолютных разниц между прогнозируемыми и фактическими значениями.	<ul style="list-style-type: none"> <li>- Менее чувствительна к выбросам.</li> <li>- Легка для понимания и интерпретации.</li> </ul>	<ul style="list-style-type: none"> <li>- Не акцентирует больших ошибок.</li> <li>- Не нормализована, трудна для сравнения между наборами данных.</li> </ul>

Продолжение на следующей странице

Продолжение таблицы 1.1

Метрика	Полное название	Описание	Достоинства	Недостатки
MAPE	Средняя абсолютная процентная погрешность (Mean Absolute Percentage Error)	Среднее значение абсолютной процентной погрешности между прогнозируемыми и фактическими значениями.	<ul style="list-style-type: none"> <li>- Легко интерпретируется.</li> <li>- Не зависит от масштаба, полезна для различных наборов данных.</li> </ul>	<ul style="list-style-type: none"> <li>- Не определена или принимает большие значения для значений, близких к нулю.</li> <li>- Асимметрична, приводит к потенциальному смещению.</li> </ul>
MSE	Среднеквадратичная ошибка (Mean Squared Error)	Усредняет квадрат разницы между прогнозируемыми и фактическими значениями.	<ul style="list-style-type: none"> <li>- Исправляет более крупные ошибки, важные во многих сценариях.</li> <li>- Основа для многих статистических методов, привычная метрика.</li> </ul>	<ul style="list-style-type: none"> <li>- Квадратичные ошибки приводят к получению высоких значений с выбросами.</li> <li>- Единицы измерения не совпадают с исходными данными, менее интуитивно понятны.</li> </ul>
$R^2$	Коэффициент детерминации	Доля отклонения зависимой переменной, предсказуемая по независимой переменной (переменным).	<ul style="list-style-type: none"> <li>- Простая в интерпретации статистика для соответствия модели.</li> <li>- Не зависит от масштаба, подходит для сравнения моделей.</li> </ul>	<ul style="list-style-type: none"> <li>- Не указывает на точность прогнозирования.</li> <li>- Может быть повышена путем добавления дополнительных переменных.</li> </ul>
Продолжение на следующей странице				

Продолжение таблицы 1.1

Метрика	Полное название	Описание	Достоинства	Недостатки
МВЕ	Средняя ошибка смещения (Mean Bias Error)	Средние различия между прогнозируемыми и фактическими значениями, указывающие на смещение.	<ul style="list-style-type: none"> <li>- Непосредственно измеряет среднюю погрешность в прогнозах.</li> <li>- Проста в вычислении и интерпретации.</li> </ul>	<ul style="list-style-type: none"> <li>- Положительные и отрицательные ошибки могут компенсироваться.</li> <li>- Отражает направление смещения, а не величину ошибки.</li> </ul>

## 1.2 Регрессионные модели

Прогнозируемая переменная  $y$  иногда также называется регрессионной, зависимой или объясняемой переменной. В простейшем случае регрессионная модель допускает линейную связь между наблюдаемым значением  $y$  и его прогнозируемым значением  $\hat{y}$  (см. уравнение (1.1)):

$$y_t = \hat{y}_t + \varepsilon_t, \quad (1.1)$$

где прогнозируемое значение  $\hat{y}_t$  вычисляется как:

$$\hat{y}_t = \beta_0 + \beta_1 y_{t-1}. \quad (1.2)$$

Коэффициенты  $\beta_0$  и  $\beta_1$  обозначают точку пересечения и наклон соответственно. Точка пересечения  $\beta_0$  представляет прогнозируемое значение  $\hat{y}$ , когда  $y_{t-1} = 0$ . Наклон  $\beta_1$  представляет среднее прогнозируемое изменение  $\hat{y}$  при увеличении  $y_{t-1}$  на одну единицу. Заметим, что наблюдения не лежат на прямой, а разбросаны вокруг нее. Мы можем рассматривать каждое наблюдение  $y_t$  как состоящее из систематической или объясненной части модели  $\hat{y}_t$  и случайной ошибки  $\varepsilon_t$  (см. уравнение (1.1)). Термин «ошибка» подразумевает не ошибку, а отклонение от базовой прямолинейной модели. Он фиксирует все, что может повлиять на  $y_t$ , кроме  $y_{t-1}$ .

Наиболее распространенными подходами к прогнозированию временных рядов являются модели ARIMA (которые нацелены на описание автокорреляций в данных) и экспоненциальное сглаживание (которые основаны на описании тенденции и сезонности данных).

### 1.2.1 Модель ARIMA

Модель ARIMA состоит из трех компонентов [26, 32, 41, 62, 93]:

- **AR** (авторегрессионный термин) — относится к прошлым значениям, используемым для прогнозирования следующего значения; определяется параметром  $p$  в модели авторегрессии (см. уравнение (1.3)):

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \dots + \phi_p Y_{t-p} + \varepsilon_t, \quad (1.3)$$

где  $\varepsilon_t$  — случайная ошибка. Параметр  $p$  определяется по PACF (Partial Auto-Correlation Function) — частичной автокорреляционной функции между  $Y_t$  и  $Y_{t-k}$  при исключении влияния  $Y_{t-1}, \dots, Y_{t-k+1}$ .

- **MA** (скользящее среднее) — используется для определения количества прошлых ошибок прогноза, используемых для прогнозирования будущих значений; определяется параметром  $q$ , получаемым из ACF (Auto-Correlation Function, автокорреляционная функция). Корреляция между значениями временного ряда определяется как (см. уравнение (1.4)):

$$\rho_k = \frac{\text{cov}\{Y_t, Y_{t-k}\}}{\text{var}\{Y_t\}}, \quad (1.4)$$

а модель скользящего среднего представлена уравнением (1.5):

$$Y_t = \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q}, \quad (1.5)$$

где  $\varepsilon_t$  — белый шум<sup>1</sup>. Скользящее среднее показывает наличие ко-

---

<sup>1</sup>Белый шум — процесс с постоянным математическим ожиданием, постоянной дисперсией и нулевой автоковариационной функцией для всех лагов, кроме нулевого.

лебаний в ряду. Чем выше значение скользящего среднего, тем выше вероятность колебаний.

- **I** (интегрирующий член) — если ряд не является стационарным, то вычисляется его разность порядка  $d$ , которая становится стационарным рядом. Для проверки стационарности ряда используются расширенный тест Дики—Фуллера (ADF)<sup>2</sup> и тест Квятковского—Филлипса—Шмидта—Шина (KPSS)<sup>3</sup>. Эти тесты позволяют определить параметр  $d$  модели.

Приведем алгоритм построения модели ARIMA.

### Алгоритм ARIMA-Construct

1. **Шаг 1.** Загрузить входные данные в вектор  $TS$ .
2. **Шаг 2.** Проверить, является ли временной ряд  $TS$  стационарным. Если  $TS$  стационарен, перейти к шагу 4. Если ряд не является стационарным, вычислить его разности порядка  $d$ , чтобы получить стационарный ряд.
3. **Шаг 3.** Положить  $d = 0$ .
4. **Шаг 4.** Построить графики ACF и PACF для определения входных параметров модели ARIMA.
5. **Шаг 5.** С помощью графиков ACF и PACF определить значения  $p$  и  $q$  для модели ARIMA.
6. **Шаг 6.** Выполнить подгонку параметров  $p$ ,  $d$  и  $q$  для идентификации модели ARIMA.
7. **Шаг 7.** Спрогнозировать значения на проверочном наборе  $TS$ : будущие значения.

---

<sup>2</sup>Методика, используемая для анализа временных рядов на стационарность в прикладной статистике и эконометрике.

<sup>3</sup>Критерий для проверки стационарности наблюдаемого временного ряда.

8. **Шаг 8.** Рассчитать среднеквадратическое отклонение для сравнения прогнозов и фактических значений.

Несмотря на то, что ARIMA является очень мощной моделью для прогнозирования временных рядов, процессы подготовки данных и настройки параметров могут отнимать много времени: требуется сделать ряд стационарным, определить значения  $p$  и  $q$  с помощью графиков.

К особенностям модели ARIMA можно отнести следующие:

- Модели требуется регулярное «дообучение» на разных этапах развития процесса.
- Часто требуется изменять не только предполагаемые параметры модели, но и количество гиперпараметров.
- Этот инструмент хорошо подходит только для краткосрочного прогнозирования.

### 1.2.2 Экспоненциальное сглаживание

Экспоненциальное сглаживание — это метод прогнозирования временных рядов, который придает больший вес более недавним наблюдениям и меньший — более старым, посредством экспоненциально убывающих весов [26, 32, 59, 64]. Этот метод особенно эффективен для данных, которые не содержат значительных трендов или сезонности.

**Простое экспоненциальное сглаживание** используется для временных рядов без тренда и сезонности. Модель определяется уравнением сглаженного уровня (см. уравнение (1.6)):

$$\ell_t = \alpha y_t + (1 - \alpha)\ell_{t-1}, \quad (1.6)$$

где  $\ell_t$  — сглаженный уровень в момент времени  $t$ ;  $y_t$  — наблюдаемое значение временного ряда в момент времени  $t$ ;  $\alpha$  — параметр сглаживания,  $0 \leq \alpha \leq 1$ .

Прогноз для будущих периодов является константой и равен последнему сглаженному уровню (см. уравнение (1.7)):

$$\hat{y}_{t+h|t} = \ell_t, \quad (1.7)$$

где  $\hat{y}_{t+h|t}$  – прогноз на  $h$  шагов вперед, сделанный в момент времени  $t$ .

Для временных рядов с трендом применяется **модель Хольта**, которая расширяет простое экспоненциальное сглаживание введением компонента тренда. Модель Хольта состоит из трех уравнений: уравнения уровня

$$\ell_t = \alpha y_t + (1 - \alpha)(\ell_{t-1} + b_{t-1}), \quad (1.8)$$

уравнения тренда

$$b_t = \beta(\ell_t - \ell_{t-1}) + (1 - \beta)b_{t-1}, \quad (1.9)$$

и уравнения прогноза

$$\hat{y}_{t+h|t} = \ell_t + hb_t, \quad (1.10)$$

где  $\ell_t$  – оценка уровня временного ряда в момент времени  $t$ ;  $b_t$  – оценка тренда (наклона) временного ряда в момент времени  $t$ ;  $\alpha$  – параметр сглаживания уровня,  $0 \leq \alpha \leq 1$ ;  $\beta$  – параметр сглаживания тренда,  $0 \leq \beta \leq 1$ ;  $h$  – горизонт прогнозирования.

Уравнение уровня (1.8) показывает, что  $\ell_t$  является взвешенным средним текущего наблюдения  $y_t$  и прогноза на предыдущий период ( $\ell_{t-1} + b_{t-1}$ ). Уравнение тренда (1.9) показывает, что  $b_t$  является взвешенным средним разности уровней ( $\ell_t - \ell_{t-1}$ ) и предыдущей оценки тренда  $b_{t-1}$ . Уравнение прогноза (1.10) демонстрирует, что прогноз на  $h$  шагов вперед равен сумме последнего уровня  $\ell_t$  и произведения тренда  $b_t$  на горизонт прогнозирования  $h$ .

Таким образом, прогнозы модели Хольта являются линейной функцией от  $h$ , что позволяет учитывать тренд в данных.

### 1.2.3 Модели BATS и TBATS

Модели BATS (**B**ox-Cox transformation, **A**RMA errors, **T**rend and **S**easonal components) и их тригонометрическая модификация TBATS были разработаны Де Ливера, Хайндманом и Снайдером в 2011 году [50]. Эти модели представляют собой различные модификации моделей пространства состояний для экспоненциального сглаживания, позволяющие обрабатывать более широкий спектр сезонных закономерностей. Чтобы избежать проблем, связанных с нелинейными моделями, авторы [26, 32, 50] сосредоточились на линейных гомоскедастичных моделях, однако допускают некоторые виды нелинейности посредством использования преобразования Бокса—Кокса. Это ограничивает их подход только положительными временными рядами, но большинство практических временных рядов являются положительными.

Модель TBATS расширяет модель BATS, используя комбинацию тригонометрических рядов Фурье для моделирования сложной сезонности. Подбор параметров модели осуществляется полностью автоматически, что позволяет модели TBATS, в отличие от других моделей, моделировать постепенно изменяющуюся сезонность. Одним из недостатков этой модели является высокая вычислительная сложность [26, 32, 51, 57, 67, 112].

Рассмотрим структуру модели TBATS. Преобразование Бокса—Кокса, примененное к временному ряду  $y_t$ , определяется следующим образом:

$$y_t^{(\omega)} = \begin{cases} \frac{y_t^\omega - 1}{\omega}, & \omega \neq 0, \\ \ln(y_t), & \omega = 0, \end{cases} \quad (1.11)$$

где  $\omega$  — параметр преобразования.

Модель ТВАТС представляется в виде (см. уравнение (1.12)):

$$y_t^{(\omega)} = \ell_{t-1} + \phi b_{t-1} + \sum_{i=1}^T \sum_{k=1}^{K_i} [s_{k,t}^{(i)} + s_{k,t}^{*(i)}] + d_t, \quad (1.12)$$

где  $\ell_t$  — уровень временного ряда в момент времени  $t$ ;  $b_t$  — тренд временного ряда в момент времени  $t$ ;  $s_{k,t}^{(i)}$ ,  $s_{k,t}^{*(i)}$  — сезонные компоненты для  $i$ -й сезонности и  $k$ -й гармоники;  $d_t$  — остаточный шум, моделируемый с помощью модели ARMA;  $\phi$  — параметр демпфирования,  $0 \leq \phi \leq 1$ .

Уровень и тренд обновляются по следующим уравнениям (см. уравнения (1.13) и (1.14)):

$$\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha d_t, \quad (1.13)$$

$$b_t = \phi b_{t-1} + \beta d_t, \quad (1.14)$$

где  $\alpha$  и  $\beta$  — параметры сглаживания уровня и тренда соответственно.

Сезонные компоненты обновляются по следующим уравнениям (см. уравнения (1.15) и (1.16)):

$$s_{k,t}^{(i)} = s_{k,t-1}^{(i)} \cos \lambda_k^{(i)} + s_{k,t-1}^{*(i)} \sin \lambda_k^{(i)} + \gamma_k^{(i)} d_t, \quad (1.15)$$

$$s_{k,t}^{*(i)} = -s_{k,t-1}^{(i)} \sin \lambda_k^{(i)} + s_{k,t-1}^{*(i)} \cos \lambda_k^{(i)} + \gamma_k^{*(i)} d_t, \quad (1.16)$$

где  $\lambda_k^{(i)} = \frac{2\pi k}{m_i}$ ,  $m_i$  — период сезонности  $i$ ;  $\gamma_k^{(i)}$ ,  $\gamma_k^{*(i)}$  — параметры сглажи-

вания сезонных компонентов.

Остаточный шум  $d_t$  моделируется с помощью модели ARMA( $p, q$ ) (см. уравнение (1.17)):

$$d_t = \sum_{j=1}^p \phi_j d_{t-j} + \sum_{k=1}^q \theta_k \varepsilon_{t-k} + \varepsilon_t, \quad (1.17)$$

где  $\varepsilon_t$  – белый шум.

Параметр демпфирования  $\phi$  из уравнения (1.12) позволяет моделировать демпфированный тренд. Если  $\phi = 1$ , тренд является недемпфированным.

Модель BATS является частным случаем модели TBATS, где сезонные компоненты моделируются аддитивными сезонными эффектами с целочисленными периодами, а не тригонометрическими функциями.

К преимуществам модели TBATS можно отнести следующие:

1. Возможность моделирования сложной и нецелочисленной сезонности, включая множественные сезонные периоды.
2. Автоматическое определение параметров модели, включая количество гармоник  $K_i$  для каждой сезонности.
3. Учет автокорреляции в остатках посредством включения модели ARMA.
4. Обработка нелинейностей в данных с помощью преобразования Бокса–Кокса.
5. Эффективная процедура оценки параметров, позволяющая улучшить качество прогнозов.

Однако модель TBATS может быть вычислительно затратной для временных рядов с большим количеством наблюдений или сложной сезонностью.

Модели BATS и TBATS представляют собой расширения моделей экспоненциального сглаживания в пространстве состояний, включающие преобразование Бокса—Кокса, модель ARMA для ошибок, тренд и сезонные компоненты. Основное отличие между ними состоит в том, что TBATS использует тригонометрические функции для моделирования сезонности, что позволяет учитывать нецелочисленные и множественные сезонные периоды.

#### 1.2.4 Модель Prophet для одномерных временных рядов

Модель Prophet — это инструмент для прогнозирования временных рядов, разработанный специально для моделирования данных социальных сетей. Она особенно эффективна для учета сезонности, праздников и трендов. Модель предназначена для обработки пропущенных данных и выбросов, что делает ее надежной для практических приложений. Архитектура модели включает входной временной ряд, разбиение на тренд, сезонность и праздничные эффекты, а также генерацию окончательного прогноза [26, 96, 106, 122].

Входными данными для модели Prophet являются одномерные временные ряды, состоящие из временных меток  $t_1, t_2, \dots, t_n$  и соответствующих значений  $y(t_1), y(t_2), \dots, y(t_n)$ .

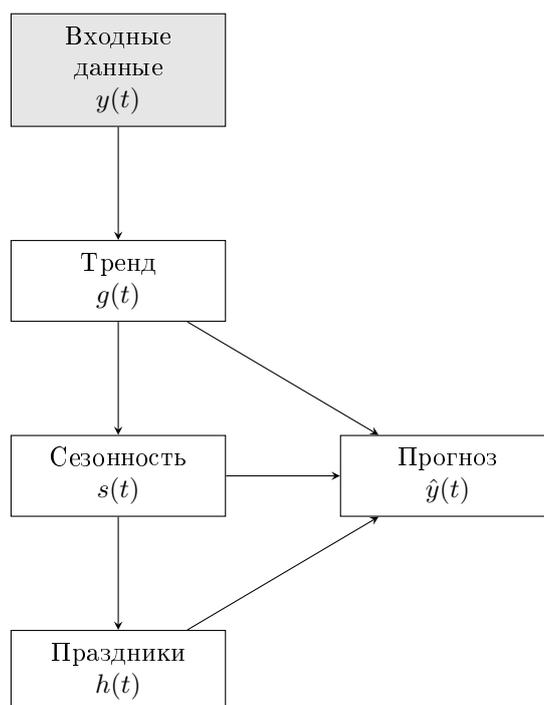
Модель разбивает временной ряд на три основные составляющие: тренд  $g(t)$ , сезонность  $s(t)$  и праздничные эффекты  $h(t)$ . Таким образом, в общем виде модель представляется суммой этих трех компонентов (см. уравнение (1.18)):

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t, \quad (1.18)$$

где  $\epsilon_t$  — ошибка модели.

Модель Prophet обучается путем сопоставления параметров тренда  $g(t)$ , сезонности  $s(t)$  и праздничных эффектов  $h(t)$  с историческими данными, используя метод максимального правдоподобия.

Архитектура модели Prophet для одномерных временных рядов представлена на рис. 1.1.



**Рисунок 1.1.** – Архитектура модели Prophet для прогнозирования одномерного временного ряда

На вход модели подается исходный временной ряд, который декомпозируется на три основных компонента: тренд  $g(t)$ , сезонность  $s(t)$  и праздничные эффекты  $h(t)$ . Каждый компонент вносит свой вклад в окончательный прогноз  $\hat{y}(t)$ . Связи на рисунке показывают, как входные данные разбиваются и объединяются для получения прогноза.

Обучаясь на исторических данных, модель Prophet выявляет основные закономерности и зависимости, что делает ее надежным инструментом для решения задач прогнозирования временных рядов.

## 1.3 Гибридные модели

В современных исследованиях и при моделировании сложных закономерностей зачастую использование одной модели не является эффективным, по этой причине разрабатываются различные гибридные модели, объединяющие в себе два и более методов. Наиболее популярны гибридные модели, в которых совмещаются только регрессионные подходы, либо регрессионные модели и нейросетевые методы. Рассмотрим некоторые из них.

### 1.3.1 Гибридная модель Auto-ARIMA и экспоненциально-го сглаживания для одномерных временных рядов

Гибридная модель, объединяющая Auto-ARIMA и экспоненциальное сглаживание (ES), использует сильные стороны обоих методов для улучшения точности прогнозирования. Модель Auto-ARIMA автоматически определяет наиболее подходящую ARIMA модель, в то время как ES помогает учесть остаточные закономерности во временных рядах [26, 94, 107].

**Автоматический выбор параметров ARIMA** осуществляется с использованием алгоритма, аналогичного алгоритму ARIMA-Construct, описанному в разделе 1.2.1. В частности, Auto-ARIMA автоматически выбирает оптимальные параметры модели ARIMA, основываясь на таких критериях, как информационный критерий Акаике (AIC) или байесовский информационный критерий (BIC). Как было описано в разделе 1.2.1, ARIMA модель может быть представлена следующим образом:

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \alpha_1 \varepsilon_{t-1} + \alpha_2 \varepsilon_{t-2} + \dots + \alpha_q \varepsilon_{t-q} + \varepsilon_t, \quad (1.19)$$

где  $\phi_i$  при  $i = 1, 2, \dots, p$  — коэффициенты авторегрессии,  $\alpha_j$  при  $j = 1, 2, \dots, q$  — коэффициенты скользящего среднего, а  $\varepsilon_t$  — ошибка модели. Модель Auto-ARIMA возвращает прогнозные значения  $\hat{y}_t^{ARIMA}$ .

**Экспоненциальное сглаживание** применяется к остаткам, полученным из модели Auto-ARIMA, которые определяются следующим образом:

$$e_t = y_t - \hat{y}_t^{ARIMA}. \quad (1.20)$$

Модель ES подгоняется к этим остаткам по следующему уравнению:

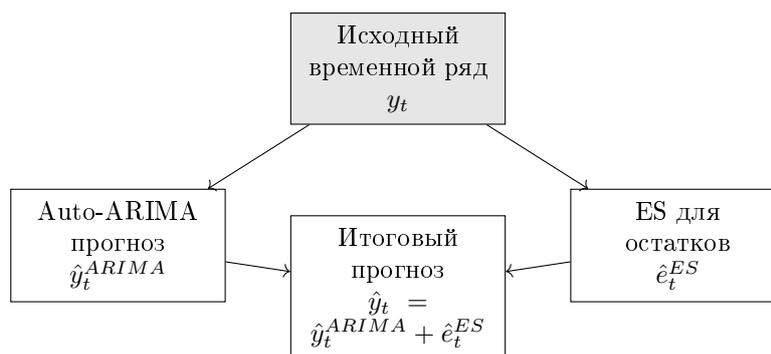
$$e_t = \ell_t + b_t + s_t + \varepsilon_t, \quad (1.21)$$

где  $\ell_t$  — уровень,  $b_t$  — тренд,  $s_t$  — сезонная составляющая, а  $\varepsilon_t$  — ошибка модели. Модель ES возвращает предсказанные остатки  $\hat{e}_t^{ES}$ .

**Гибридная модель** объединяет прогнозы Auto-ARIMA и остатки, подогнанные моделью ES, для получения окончательного прогноза:

$$\hat{y}_t = \hat{y}_t^{ARIMA} + \hat{e}_t^{ES}. \quad (1.22)$$

На вход модели подается исходный временной ряд  $y_t$  (серый прямоугольник). Вначале данные обрабатываются в блоке Auto-ARIMA, который формирует прогноз  $\hat{y}_t^{ARIMA}$ . Остатки, полученные при прогнозировании, моделируются с использованием экспоненциального сглаживания (ES), что дает  $\hat{e}_t^{ES}$ . Окончательный прогноз  $\hat{y}_t$  получается путем суммирования прогноза Auto-ARIMA и предсказанных остатков ES. Стрелки



**Рисунок 1.2.** – Гибридная модель Auto-ARIMA и ES для одномерных временных рядов

указывают на процесс обработки и объединения данных для получения окончательного прогноза.

Гибридная модель Auto-ARIMA и экспоненциального сглаживания (ES) объединяет автоматический подбор модели ARIMA и гибкость ES для учета остатков, что позволяет эффективно отражать как краткосрочные, так и долгосрочные закономерности в данных, повышая точность прогноза.

### 1.3.2 Гибридная модель Auto-ARIMA и полиномиальной регрессии для одномерных временных рядов

Гибридная модель, сочетающая Auto-ARIMA и полиномиальную регрессию, использует сильные стороны обоих методов для повышения точности прогнозирования. Auto-ARIMA автоматически определяет наиболее подходящую модель ARIMA, в то время как полиномиальная регрессия фиксирует остаточные закономерности во временных рядах [26, 47, 97].

**Автоматический подбор параметров ARIMA** осуществляется по алгоритму, аналогичному описанному в разделе 1.2.1. Модель Auto-ARIMA автоматически выбирает оптимальные параметры на основе таких критериев, как информационный критерий Акаике (AIC) или бай-

есовский информационный критерий (BIC). Уравнение модели ARIMA, как показано ранее, имеет вид:

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \alpha_1 \varepsilon_{t-1} + \alpha_2 \varepsilon_{t-2} + \dots + \alpha_q \varepsilon_{t-q} + \varepsilon_t, \quad (1.23)$$

где  $\phi_i$ ,  $i = 1, 2, \dots, p$  — коэффициенты авторегрессии,  $\alpha_j$ ,  $j = 1, 2, \dots, q$  — коэффициенты скользящего среднего, и  $\varepsilon_t$  — ошибка модели. Прогнозные значения модели ARIMA обозначаются как  $\hat{y}_t^{ARIMA}$ .

**Полиномиальная регрессия** применяется к остаткам, полученным из модели Auto-ARIMA. Остатки вычисляются по формуле:

$$e_t = y_t - \hat{y}_t^{ARIMA}. \quad (1.24)$$

Для этих остатков строится модель полиномиальной регрессии, описываемая следующим уравнением:

$$e_t = \beta_0 + \beta_1 t + \beta_2 t^2 + \dots + \beta_d t^d + \varepsilon_t, \quad (1.25)$$

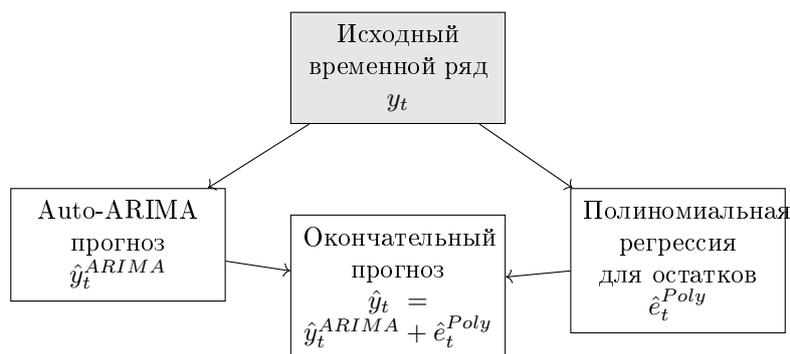
где  $t$  — время,  $\beta_i$  — коэффициенты полинома, а  $\varepsilon_t$  — ошибка модели. Предсказанные значения из полиномиальной регрессии обозначаются как  $\hat{e}_t^{Poly}$ .

Как уже было отмечено, **гибридная модель** объединяет прогнозы Auto-ARIMA и остатки, полученные с помощью полиномиальной регрессии, для получения окончательного прогноза:

$$\hat{y}_t = \hat{y}_t^{ARIMA} + \hat{e}_t^{Poly}. \quad (1.26)$$

Архитектура данной гибридной модели для прогнозирования одно-

мерных временных рядов показана на рис. 1.3.



**Рисунок 1.3. – Гибридная модель Auto-ARIMA и полиномиальная регрессия для одномерного временного ряда**

На вход модели подается одномерный временной ряд  $y_t$  (серый прямоугольник). Этот ряд передается сначала в компонент Auto-ARIMA, который генерирует прогноз  $\hat{y}_t^{ARIMA}$ . Остатки прогнозируются с использованием полиномиальной регрессии, что дает  $\hat{e}_t^{Poly}$ . Итоговый прогноз  $\hat{y}_t$  получается путем объединения прогнозов Auto-ARIMA и полиномиальной регрессии. Стрелки показывают процесс обработки и объединения данных для получения окончательного прогноза.

Гибридная модель Auto-ARIMA и полиномиальной регрессии для одномерных временных рядов использует преимущества обеих методик для повышения точности прогнозирования.

## 1.4 Модели машинного обучения

Еще одним важным компонентом науки о данных является машинное обучение, которое в настоящее время считается важным инструментом для использования технологий искусственного интеллекта. Данный подход использует алгоритмы и модели нейронных сетей. Алгоритмы машинного обучения автоматически создают математическую модель, используя выборочные данные, также известные как «данные для обу-

чения», для принятия решений без специального программирования для принятия этих решений. Благодаря использованию статистических методов алгоритмы обучаются классифицировать и прогнозировать, раскрывая ключевые идеи в рамках проектов интеллектуального анализа данных. Эти идеи впоследствии влияют на принятие решений.

#### 1.4.1 Многослойный перцептрон (MLP) для одномерных временных рядов

Многослойный перцептрон (MLP) – это тип искусственной нейронной сети прямого распространения, обычно используемой для решения задач регрессии и классификации. В прогнозировании одномерных временных рядов MLP может моделировать взаимосвязь между прошлыми наблюдениями и будущими значениями. Архитектура сети включает входной слой, один или несколько скрытых слоев и выходной слой [26, 39, 43, 127]. Модель обучается на исторических данных, извлекая скрытые закономерности, что делает ее мощным инструментом для прогнозирования временных рядов.

**Входной слой** модели MLP состоит из фиксированного числа прошлых наблюдений.

Каждый **скрытый слой** выполняет преобразование входных данных с помощью взвешенной суммы, за которой следует нелинейная функция активации. Выходные данные  $k$ -го скрытого слоя обозначаются  $\mathbf{h}^k$ .

**Выходной слой** генерирует окончательный прогноз. Для одномерного временного ряда результатом является одно значение  $\hat{y}_t$ .

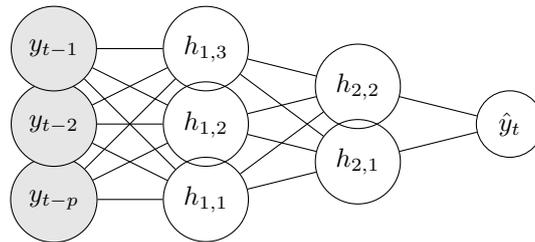
Модель MLP обучается путем минимизации функции потерь, которая обычно определяется как среднеквадратичная ошибка (MSE) между

прогнозируемыми значениями  $\hat{y}_t$  и фактическими значениями  $y_t$ :

$$MSE = \frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2, \quad (1.27)$$

где  $n$  – количество наблюдений в обучающей выборке.

На рисунке 1.4 представлена архитектура MLP для прогнозирования одномерных временных рядов.



**Рисунок 1.4. – Архитектура модели MLP для одномерных временных рядов**

Модель состоит из трех основных компонентов: входного слоя, скрытых слоев и выходного слоя. Входной слой (обозначен серыми кружками) содержит предыдущие наблюдения  $y_{t-1}, y_{t-2}, \dots, y_{t-p}$ , которые подаются в первый скрытый слой. Скрытые слои (обозначены узорами) обрабатывают данные с помощью нейронов, а выходной слой (белый кружок) генерирует прогнозируемое значение  $\hat{y}_t$ . Стрелки указывают на передачу и преобразование данных через сеть.

#### 1.4.2 Метод опорных векторов для одномерных временных рядов

Метод опорных векторов (Support Vector Machine, SVM) – это модель машинного обучения с учителем, которая может использоваться как для задач классификации, так и для регрессии. В прогнозировании одномерных временных рядов SVM моделирует взаимосвязь между про-

шлыми наблюдениями и будущими значениями, находя гиперплоскость в многомерном пространстве, которая наилучшим образом соответствует данным. Архитектура модели включает пространство ввода, преобразование пространства признаков и функцию принятия решений [26, 95, 117].

**Входные данные** для SVM представляют собой фиксированное число прошлых наблюдений.

SVM преобразует входное пространство в **пространство признаков** более высокой размерности с использованием функции ядра. Функция ядра  $K(\mathbf{y}_i, \mathbf{y}_j)$  вычисляет внутреннее произведение в этом пространстве признаков.

SVM находит оптимальную гиперплоскость, минимизирующую ошибку прогноза.

Модель SVM обучается путем решения задачи оптимизации:

$$\min_{\alpha, \alpha^*} \frac{1}{2} \sum_{i,j=1}^N (\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) K(\mathbf{y}_i, \mathbf{y}_j) - \sum_{i=1}^N (\alpha_i - \alpha_i^*) y_i \quad (1.28)$$

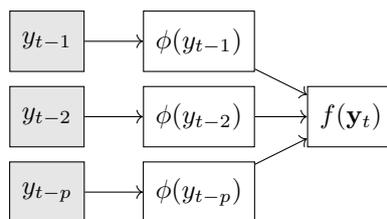
при ограничениях:

$$0 \leq \alpha_i, \alpha_i^* \leq C, \quad \sum_{i=1}^N (\alpha_i - \alpha_i^*) = 0, \quad (1.29)$$

где  $y_i$  — целевые значения, а  $C$  — параметр регуляризации.

На рисунке 1.5 представлена архитектура SVM для одномерного прогнозирования временных рядов.

Входные данные представляют собой прошлые наблюдения  $y_{t-1}, y_{t-2}, \dots, y_{t-p}$ , которые преобразуются в пространство признаков более высокой размерности с помощью функции ядра. Функция принятия решений  $f(\mathbf{y}_t)$  использует преобразованные признаки для



**Рисунок 1.5.** – Метод опорных векторов для одномерных временных рядов

генерации прогнозируемого значения. Все уровни взаимосвязаны, данные последовательно проходят через модель.

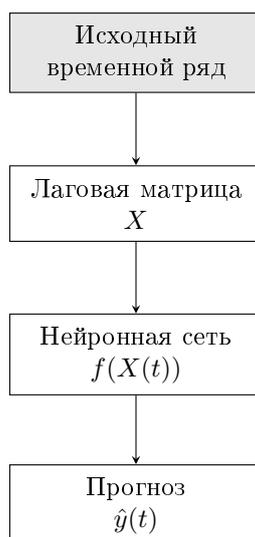
### 1.4.3 Модель NNAR (Neural Network Autoregressive)

Искусственная нейронная сеть (ИНС) представляет собой многоуровневую структуру связанных нейронов, аналогичную биологическим нейронным сетям. Она использует комбинацию различных алгоритмов для выполнения сложных операций с данными. Нейронная сеть может быть описана как сеть «нейронов», организованных в слоях. Предикторы (входные данные) формируют нижний слой, прогнозы (выходные данные) — верхний слой. В промежуточных слоях могут быть скрытые нейроны [91, 108].

Модель NNAR (Neural Network Autoregressive) — это нейронная сеть для прогнозирования временных рядов, которая расширяет стандартную авторегрессионную модель за счет использования нелинейных компонент, реализованных с помощью нейронных сетей. Эта модель особенно полезна для улавливания сложных нелинейных зависимостей и динамики временных рядов [66, 91].

Архитектура модели NNAR для одномерных временных рядов представлена на рисунке 1.6.

На вход модели подаются фактические данные временного ряда  $y(t)$ , которые преобразуются в лаговую матрицу  $X$ . Нейронная сеть обраба-



**Рисунок 1.6.** – Модель NNAR для одномерных временных рядов

тывает лаговую матрицу для получения прогноза  $\hat{y}(t)$ . На каждом этапе данные проходят через соответствующие слои обработки, начиная с лагов и заканчивая финальным прогнозом.

Модель NNAR, обученная на исторических данных, способна выявлять сложные нелинейные зависимости, что делает ее мощным инструментом для прогнозирования временных рядов. В отличие от линейных моделей, NNAR эффективно улавливает сложные нелинейные паттерны, присущие многим временным рядам.

Прогнозирование модели NNAR осуществляется итеративно. Для прогноза на один шаг вперед используются доступные исторические данные. Для прогноза на два шага вперед одношаговый прогноз  $\hat{y}(t + 1)$  используется как входное значение вместе с историческими данными, и процесс продолжается до тех пор, пока не будут получены все прогнозы.

Модель NNAR часто используется в гибридных подходах совместно с моделями ARIMA, поскольку каждая модель компенсирует слабые стороны другой. В частности, NNAR хорошо справляется с нелинейными зависимостями, тогда как ARIMA превосходна в моделировании линейных и сезонных компонентов временного ряда [45, 73, 113, 133].

#### 1.4.4 Модели LSTM, BiLSTM, GRU

Нейросетевые модели были разработаны для решения проблемы долгосрочной зависимости, с которой сталкиваются рекуррентные нейронные сети (RNN) при возникновении проблемы исчезающего градиента. В 1997 году была предложена модель LSTM (Long-Short Time Memory), которая решает проблему исчезающего градиента для данных временных рядов [63]. LSTM регулирует поток информации, передает соответствующую информацию и прогнозирует результаты на основе поступающей информации. Функция активации LSTM считается функцией идентичности из-за ее рекуррентного характера. Следовательно, градиент при обратном распространении остается постоянным. Функция активации узла определяет вывод этого конкретного узла. Функции активации, которые обычно используются в сети LSTM, представляют собой сигмоид и гиперболический тангенс.

В отличие от традиционных нейросетевых моделей с прямой связью, модель LSTM имеет обратную связь. Эта особенность позволяет LSTM обрабатывать большие последовательности данных (например, временные ряды) без независимой обработки каждой точки в последовательности путем сохранения важной информации о более ранних точках данных для поддержки обработки новых точек данных. В результате такие модели особенно эффективны при обработке последовательностей общих данных временных рядов [48, 54].

То есть, можно сказать, что LSTM и ей подобные модели – это особый вид рекуррентной нейронной сети, способной изучать долгосрочные зависимости в данных за счет того, что повторяющийся модуль модели представляет собой комбинацию четырех слоев, взаимодействующих друг с другом.

**Двунаправленная модель LSTM (BiLSTM)** поддерживает два отдельных состояния для прямых и обратных входных данных, которые генерируются двумя разными LSTM [115]. Первая LSTM представляет собой обычную последовательность, которая начинается с начала предложения, тогда как во второй LSTM входная последовательность подается в обратном порядке. Идея двунаправленной сети состоит в том, чтобы собирать информацию об окружающих входах. Обычно она обучается быстрее, чем однонаправленная модель, хотя это зависит от характера исходных данных.

Эта модель сочетает в себе преимущества моделей BiRNN и LSTM, она используется для распространения использования прямого и обратного направлений. Модель BiLSTM представляет собой двустороннюю сеть, используемую для хранения будущих и прошлых данных. Она эффективно увеличивает объем информации, доступной для сети, улучшая контекст, доступный для использования алгоритма.

Недостатком модели является то, что в ней не уделяется должного внимания взаимодействию внутри данных. При изменении состояния скрытого слоя эта модель учитывает только влияние сохраненной информации на текущий ввод, игнорируя при этом влияние на нее последующей информации в последовательности [125, 132]. Итоговым результатом модели является объединение двух обратных выходных данных LSTM [77].

**Модель GRU (Gated Recurrent Unit)** также направлена на решение проблемы исчезающего градиента, которая возникает со стандартной рекуррентной нейронной сетью.

GRU также можно рассматривать как разновидность LSTM, потому что обе эти модели имеют схожие архитектуры, и в некоторых случаях

дают одинаково отличные результаты. Вообще, GRU— это то же самое, что и RNN, но разница заключается в работе модели и вратах, связанных с каждым блоком GRU. Чтобы решить проблему, с которой сталкивается стандартная RNN, модель GRU включает два механизма: ворота обновления и ворота сброса. По сути, это два вектора, которые определяют, какая информация должна быть передана на выход. Особенность их в том, что их можно научить хранить информацию, полученную давно, не размывая ее во времени и не удаляя информацию, не имеющую отношения к прогнозированию.

### **Различия между моделями LSTM, BiLSTM и GRU:**

1. **LSTM** имеет три типа ворот: забывающие, входные и выходные, что делает ее способной сохранять долгосрочную информацию. Модель более сложна и использует больше параметров.
2. **BiLSTM** включает две LSTM: прямую и обратную, что позволяет учитывать как прошлые, так и будущие временные шаги. Это увеличивает контекст, но требует больше вычислительных ресурсов.
3. **GRU** – более простая модель по сравнению с LSTM. Она использует два типа ворот — обновления и сброса, что снижает сложность и ускоряет вычисления, но может быть менее точной на сложных задачах.

Таким образом, модель GRU использует меньше параметров обучения и, следовательно, использует меньше памяти и выполняется быстрее, нежели LSTM, в то время как LSTM более точна для большего набора данных. Следовательно, для больших последовательностей, когда важная точность, есть смысл выбрать LSTM, модель GRU используется при необходимости сокращения объема используемой памяти и с целью более быстрого получения результатов[23–25].

## 1.5 Сравнение рассмотренных моделей

Обобщим информацию о рассмотренных выше моделях в таблице 1.2.

Таблица 1.2 – Сравнение моделей прогнозирования для одномерных временных рядов

Модель	Достоинства	Недостатки
ARIMA [41, 62, 93]	Фиксирует автокорреляции и подходит для нестационарных данных.	Требуется дифференцированность функции для обеспечения нестационарности и требует сложной настройки параметров.
Линейная модель Хольта [59, 64]	Учитывает тренд и проста в реализации.	Не подходит для данных с нерегулярными циклами и требует постоянного обновления.
BATS [50]	Моделирует различные сезонные модели и подходит для создания сложных моделей.	Требуется больших вычислительных затрат и больших выборок данных.
TBATS [51, 57, 67, 112].	Расширяет возможности BATS, моделируя нецелочисленную сезонность с помощью тригонометрических функций.	Более требовательна, чем BATS, к вычислительным ресурсам.
Prophet [96, 106, 122]	Обрабатывает недостающие данные и выбросы, а также фиксирует тренды, сезонность и праздничные дни.	Требуется настройки под конкретные данные и не идеальна для краткосрочных прогнозов.
Гибрид Auto-ARIMA+ES [94, 107]	Сочетает в себе сильные стороны ARIMA и ES, отражая как автокорреляцию, так и тренд/сезонность.	Более сложна в реализации и требует комплексного анализа остатков.

Продолжение на следующей странице

Таблица 1.2 – продолжение, начало на предыдущей странице

Модель	Достоинства	Недостатки
Гибрид Auto-ARIMA и полиномиальная регрессия [47, 97]	Фиксирует нелинейные закономерности и повышает точность прогноза за счет моделирования остатков.	Требует больших вычислительных затрат и выбора степени многочлена.
MLP [39, 43, 127]	Позволяет уловить сложные закономерности.	Требует больших объемов данных и подвержена переобучению.
SVM [95, 117]	Эффективен для больших наборов данных и устойчив к переобучению.	Требует вычисления функции ядра и требует больших вычислительных затрат для больших наборов данных.
NNAR [91, 108]	Захватывает сложные нелинейные зависимости и динамику временных рядов.	Требует настройки гиперпараметров и может переобучаться при малых выборках данных.
LSTM [56, 61]	Хорошо справляется с задачами долгосрочного запоминания зависимостей во временных рядах.	Требует больших вычислительных затрат и времени на обучение.
BiLSTM [46, 70]	Использует информацию как из прошлого, так и из будущего для повышения точности прогноза.	Более сложная архитектура, увеличивает вычислительные затраты по сравнению с LSTM.
GRU [76, 134]	Более простая архитектура по сравнению с LSTM, что ускоряет обучение.	Может уступать LSTM в улавливании долгосрочных зависимостей для сложных задач.

## 1.6 Прогнозирование с помощью систем дифференциальных или разностных уравнений

Известно, что дифференциальные уравнения связывают значение функции и ее производных в определенный момент времени  $y'(t) = f(t)$ . Разностные уравнения связывают значения наблюдаемой функции в разные моменты времени. Потому численные методы для дифференциальных моделей часто сводятся к разностным моделям. Поэтому эти модели целесообразно рассмотреть в одном разделе. Кроме того, при решении дифференциальных уравнений (описывающих непрерывную модель) не возникает особенностей, в отличие от решения разностных уравнений (описывающих дискретную модель), когда возможно появление ситуаций, когда не известны частные критерии, и осуществляется выбор функции потерь, не имеющей математического ожидания и дисперсии. Такие случайные события еще называют «черным лебедем». При моделировании таких событий невозможно использовать классический подход – метод наименьших квадратов.

### 1.6.1 Классическая эпидемиологическая модель

Традиционные эпидемиологические модели типа SIR (Susceptible, Infected, Recovered) и их многочисленные варианты описывают плотность инфицированных  $I$  типичным уравнением [49]

$$\frac{dI}{dt} = \lambda IS - \nu I, \quad (1.30)$$

где  $S$  и  $I$  – плотности восприимчивых и инфицированных лиц, а  $\lambda$ ,  $\mu$  – положительные константы. Первый член в правой части этого уравне-

ния характеризует появление вновь инфицированных лиц в результате их контакта с восприимчивыми. Второй член соответствует уменьшению  $I$  за счет выздоровления или смерти инфицированных. В начале распространения инфекции число инфицированных и выздоровевших значительно меньше числа восприимчивых, так что мы можем аппроксимировать  $S$  константой  $S \approx S_0$ . Используя это приближение, получим линейное дифференциальное уравнение с постоянными коэффициентами. Его решение задается функцией  $I(t) = I_0 \exp(\alpha t)$ , где  $I_0$  – число инфицированных в начальный момент времени, а  $\alpha = \lambda S_0 - \nu$ . Если  $\alpha > 0$ , тогда  $I(t)$  растет экспоненциально. Это же условие можно записать в виде  $R_0 > 1$ , где  $R_0 = \lambda S_0 / \nu$  – базовый номер репродукции.

Таким образом, согласно уравнению (1.30), рост числа инфицированных в начале эпидемии экспоненциальный, а затем, при уменьшении  $S$ , замедляется. Однако подобное уравнение не описывает ускоренный рост во время второй и последующих волн эпидемии. Возможное объяснение такого ускорения состоит в том, что в начале эпидемии болезнь распространяется в основном среди людей со слабым иммунным ответом. На последующих этапах заражаются и люди с более сильным иммунным ответом.

Большой толчок в развитии методов моделирования с помощью этой модели произошел в 2020–2022 годах во время распространения Covid-19, когда при использовании реальных данных выяснилось, что классическая модель SIR не обеспечивает высокого качества получаемых прогнозов [40, 65, 119] из-за различий в алгоритмах выбора ее параметров. Так, в [119] выпущенное летом 2020 года расширение для R covid19.analytics [109] использует модель SIR и позволяет получить полную картину распространения Covid-19 в любой точке мира.

В самом начале пандемии появились исследования с применением усложненных моделей SIR, учитывающих как выбытие инфицированных людей по причине смертности (модель SIR(D) в Китае), так и инкубационный период болезни (модель SEIR) для краткосрочного прогнозирования эпидемии в Мексике [40]. В [49] авторы пытаются разработать модель SEIR, предполагая, что эпидемическое прогрессирование начинается с субпопуляции, характеризующейся слабым иммунным ответом, и на более позднем этапе продолжается во всей популяции. Существует и много других усложнений и модификаций классической модели, учитывающих различные факторы.

### 1.6.2 Модели на основе разностных уравнений

**Определение 1.** *Квазилинейное уравнение – тип дифференциального (разностного) уравнения, в котором функция является линейной относительно всех производных (разностей) неизвестной функции. Такое уравнение имеет вид*

$$A_1(x, u) \frac{\partial u}{\partial x_1} + \dots + A_n(x, u) \frac{\partial u}{\partial x_n} = B(x, u).$$

Модели на основе квазилинейных уравнений используются для решения задачи регрессионного анализа с взаимозависимыми наблюдаемыми переменными. Это позволяет реализовать метод наименьших отклонений (GLDM) [103]. Модель, использующая данные параметры может позволить получить долгосрочный прогноз. В отличие от нейронных сетей этот подход позволяет в явном виде получать качественные квазилинейные разностные уравнения (адекватно описывающие рассматриваемый процесс). В данном случае не используется никаких частных

критериев, поскольку они неизвестны, и осуществляется выбор функции потерь, не имеющей математического ожидания и дисперсии, что позволяет моделировать случайные события, называемые «черным лебедем».

Теория черного лебедя или теория событий черного лебедя – это метафора, описывающая событие, которое становится неожиданностью, имеет большое влияние и часто неуместно рационализируется постфактум с выгодой задним числом. Этот термин основан на древнем высказывании о том, что черных лебедей якобы не существует.

Теория была разработана Нассимом Николасом Талебом, начиная с 2001 года, чтобы объяснить:

- Несоразмерную роль громких, труднопрогнозируемых и редких событий, выходящих за рамки обычных ожиданий в истории, науке, финансах и технологиях.
- Невычислимость вероятности последовательных редких событий с использованием научных методов (из-за самой природы малых вероятностей).
- Психологические предубеждения, которые слепят людей, как индивидуально, так и коллективно, к неопределенности и огромной роли редкого события в исторических делах.

В качестве прототипа взята модель, рассмотренная в работах Панюкова А.В. и Мезаала Я.А. [103], в которой рассматривается алгоритм определения коэффициентов  $a_1, a_2, a_3 \dots, a_m \in \mathbb{R}$  квазилинейной авторегрессионной модели  $m$ -го порядка

$$y_t = \sum_{j=1}^{n(m)} a_j g_j(\{y_{t-k}\}_{k=1}^m) + \varepsilon_t, \quad t = 1, 2, \dots, T \quad (1.31)$$

актуальной информацией о значениях переменных состояния  $\{y_t \in$

$\mathbb{R}\}_{t=1-m}^T$  в моменты времени  $t$ ; где  $g_j : (\{y_{t-k}\}_{k=1}^m) \rightarrow \mathbb{R}$ ,  $j=1,2,\dots,n(m)$ , функции  $n(m)$  заданы, и  $\{\varepsilon_t \in \mathbb{R}\}_{t=1}^T$  – неизвестные ошибки.

Использованный для построения алгоритмов метод проекции градиента, подробно рассмотрен в работе [81] для задач минимаксной оптимизации с ограничениями-неравенствами. Разработанный алгоритм строит свертки частных критериев. В [38] представлен релаксированный метод проекции градиента. В этом методе используются модификации, которые показывают хорошее улучшение значений целевой функции. Рассмотренный в [38] метод не требует точной настройки параметров; поэтому он используется в практических расчетах. В [78] обсуждается новый алгоритм проецирования градиента. Этот метод использует линейный поиск в допустимом направлении и адаптивный выбор длины шага, основанный на стратегиях чередования хорошо известных правил Барзилая-Борвейна. Авторы этой работы обсуждают и сходимость своего подхода. Тем не менее, большинство рассмотренных подходов не учитывает возможности появления «черного лебедя». Следовательно, разработка нового подхода для этой ситуации является актуальной задачей.

Алгоритм GLDM [103] в качестве исходных данных принимает вектор  $\{y_t \in \mathbb{R}\}_{t=1-m}^T$  длины  $T+m \geq (1+3m+m^2)$  и определяет факторы  $a_1, a_2, a_3, \dots, a_m \in \mathbb{R}$  при решении задачи оптимизации

$$\sum_{t=1}^T \arctan \left| \sum_{j=1}^{n(m)} a_j g_j(\{y_{t-k}\}_{k=1}^m) - y_t \right| \rightarrow \min_{\{a_j\}_{j=1}^{n(m)} \subset \mathbb{R}} \quad (1.32)$$

Распределение Коши

$$F(\xi) = \frac{1}{\pi} \arctan(\xi) + \frac{1}{2}$$

имеет максимальную энтропию среди распределений случайных вели-

чин, не имеющих математического ожидания и дисперсии. Поэтому для данного исследования применяется функция  $\arctan(*)$ .

В дальнейшем рассмотрим модель  $m$ -го порядка с квадратичной нелинейностью, поэтому базовый набор функций  $g_{(*)}$  содержит

$$\begin{aligned} g_{(k)}(\{y_{t-k}\}_{k=1}^m) &= y_{t-k}, \\ g_{(kl)}(\{y_{t-k}\}_{k=1}^m) &= y_{t-k} \cdot y_{t-l}, \\ k &= 1, 2, \dots, m; \quad l = k, k+1, \dots, m. \end{aligned}$$

Очевидно, в этом случае  $n(m) = 2m + C_m^2 = m(m+3)/2$ , а нумерация функций  $g_{(*)}$  может быть произвольной.

Предиктор формирует индексированное  $t = 1, 2, \dots, T-1, T$  семейство разностных уравнений  $m$ -го порядка

$$\begin{aligned} \overline{y[t]}_{\tau} &= \sum_{j=1}^{n(m)} a_j^* g_j(\{\overline{y[t]}_{\tau-k}\}_{k=1}^m), \\ \tau &= t, t+1, t+2, t+3, \dots, T-1, T, T+1, \dots \quad (1.33) \end{aligned}$$

для решетчатых функций  $\overline{y[t]}$  со значениями  $\overline{y[t]}_{\tau}$ , которые интерпретируются как построенный в момент времени  $t$  прогноз для  $y_{\tau}$ . Для нахождения значений функции  $\overline{y[t]}$  воспользуемся решением задачи Коши для ее разностного уравнения (1.33) при начальных условиях

$$\begin{aligned} \overline{y[t]}_{t-1} &= y_{t-1}, \quad \overline{y[t]}_{t-2} = y_{t-2}, \quad \dots, \quad \overline{y[t]}_{t-m} = y_{t-m}, \\ t &= 1, 2, \dots, T-1, T \quad (1.34) \end{aligned}$$

Множество  $\overline{Y}_{\tau} = \left\{ \overline{y[t]}_{\tau} \right\}_{t=1}^T$  возможного прогнозирования значения  $y_{\tau}$  используется для оценки вероятностных характеристик величины  $y_{\tau}$ .

## 1.7 Выводы по главе 1

Большинство из известных в настоящее время методов применимы для краткосрочного прогнозирования, поэтому актуальна задача разработки математического подхода, позволяющего в явном виде получать качественные квазилинейные разностные уравнения (адекватно описывающие рассматриваемый процесс).

В течение долгого времени остается актуальным вопрос выбора наиболее подходящего метода прогнозирования имеющегося временного ряда. Данная задача не является тривиальной, поскольку выбор метода зачастую зависит от структуры временного ряда, а также самого процесса, частоты получения новых данных, объема выборки, горизонта прогнозирования и пр. Поэтому создание единой гибкой системы прогнозирования, осуществляющей автоматический подбор метода прогнозирования является важной задачей. Существующее программное обеспечение позволяет использовать последовательно каждый из методов и не позволяет автоматически получить сводный анализ примененных методов.

Таким образом, актуальна задача создания системы прогнозирования, которая способна обрабатывать любые временные ряды произвольной длины и не зависит от характера прикладных задач, автоматически настраивает параметры используемых моделей, расширяема (допускает подключение дополнительных модулей для прогнозирования), позволяет выбрать критерий отбора «лучшей» модели.

В систему прогнозирования представляется разумным добавить метод определения параметров квазилинейного рекуррентного уравнения — подробно описанный в следующей главе. Для метода приведены доказательства результативности и сходимости итерационного процесса.

## ГЛАВА 2. ОЦЕНКА ПАРАМЕТРОВ КВАЗИЛИНЕЙНОГО РЕКУРРЕНТНОГО УРАВНЕНИЯ

В данной главе рассмотрим способ оценки параметров квазилинейного рекуррентного уравнения. Оригинальность данного подхода заключается в том, что он вместо метода наименьших квадратов использует обобщенный метод наименьших модулей с подбором весов.

Блок-схема рассматриваемого алгоритма приведена на рис. 2.1.



Рисунок 2.1. – Схема реализации модели

## 2.1 Оценка по WLDM

Алгоритм оценки по WLDM [103] в качестве исходных данных получает временной ряд  $\{y_t \in \mathbb{R}\}_{t=1-m}^T$  и весовые коэффициенты  $\{p_t \in \mathbb{R}^+\}_{t=1}^T$  и вычисляет коэффициенты

$$a_1, a_2, a_3 \dots, a_{n(m)} \in \mathbb{R}$$

с помощью решения задачи оптимизации

$$\{a_j^*\}_{j=1}^{n(m)} = \arg \min_{\{a_j\}_{j=1}^{n(m)} \in \mathbb{R}^{n(m)}} \left[ \sum_{t=1}^T p_t \cdot \left| \sum_{j=1}^{n(m)} a_j g_j(\{y_{t-k}\}_{k=1}^m) - y_t \right| \right]. \quad (2.1)$$

Эта задача представляет собой задачу выпуклой кусочно-линейной оптимизации, а введение дополнительных переменных сводит ее к задаче линейного программирования

$$\sum_{t=1}^T p_t z_t \rightarrow \min_{\substack{(a_1, a_2, \dots, a_{n(m)}) \in \mathbb{R}^m, \\ (z_1, z_2, \dots, z_T) \in \mathbb{R}^T}} \quad (2.2)$$

$$-z_t \leq \sum_{j=1}^{n(m)} [a_j g_j(\{y_{t-k}\}_{k=1}^m)] - y_t \leq z_t, \quad t = 1, 2, \dots, T, \quad (2.3)$$

$$z_t \geq 0, \quad t = 1, 2, \dots, T \quad (2.4)$$

Задача (2.2)-(2.4) имеет канонический вид с  $n(m) + T$  переменными и  $3n$  ограничениями-неравенствами, включая условия неотрицательности переменных  $z_j$ ,  $j = 1, 2, \dots, T$ .

Задача, двойственная к (2.2) –

$$\sum_{t=1}^T (u_t - v_t) y_t \rightarrow \max_{u, v \in \mathbb{R}^T}, \quad (2.5)$$

$$\sum_{t=1}^T a_j g_j(\{y_{t-k}\}_{k=1}^m) (u_t - v_t) = 0, \quad j = 1, 2, \dots, n(m), \quad (2.6)$$

$$u_t + v_t = p_t, \quad u_t, v_t \geq 0, \quad t = 1, 2, \dots, T. \quad (2.7)$$

Введем переменные  $w_t = u_t - v_t$ ,  $t = 1, 2, \dots, T$ . Из (2.7) следует

$$u_t = \frac{p_t + w_t}{2}, \quad v_t = \frac{p_t - w_t}{2}, \quad -p_t \leq w_t \leq p_t, \quad t = 1, 2, \dots, T.$$

Поэтому оптимальное значение задачи (2.5)-(2.7) равно оптимальному значению задачи

$$\sum_{t=1}^T w_t \cdot y_t \rightarrow \max_{w \in \mathbb{R}^T}, \quad (2.8)$$

$$\sum_{t=1}^T g_j(\{y_{t-k}\}_{k=1}^m) \cdot w_t = 0, \quad j = 1, 2, \dots, n(m), \quad (2.9)$$

$$-p_t \leq w_t \leq p_t, \quad t = 1, 2, \dots, T. \quad (2.10)$$

Ограничения (2.9) определяют  $(T - n(m))$ -мерное линейное подпространство  $\mathcal{L}$  с  $(n(m) \times T)$ -матрицей

$$S = \begin{bmatrix} g_1(\{y_{1-k}\}_{k=1}^m) & g_1(\{y_{2-k}\}_{k=1}^m) & \cdots & g_1(\{y_{T+1-k}\}_{k=1}^m) \\ g_2(\{y_{1-k}\}_{k=1}^m) & g_2(\{y_{2-k}\}_{k=1}^m) & \cdots & g_2(\{y_{T+1-k}\}_{k=1}^m) \\ \vdots & \vdots & \ddots & \vdots \\ g_{n(m)}(\{y_{1-k}\}_{k=1}^m) & g_{n(m)}(\{y_{2-k}\}_{k=1}^m) & \cdots & g_{n(m)}(\{y_{T+1-k}\}_{k=1}^m) \end{bmatrix}$$

Ограничения (2.10) определяют  $T$ -мерный параллелепипед  $\mathcal{T}$ .

Простая структура допустимого множества задачи (2.8)–(2.10): пересечение  $(T - n(m))$ -мерного линейного подпространства  $\mathcal{L}$  (2.9) и  $T$ -мерный параллелепипед  $\mathcal{T}$  (2.10), – позволяют найти его решение с помощью алгоритма, использующего проекцию градиента целевой функции (2.8) (т.е. вектор  $\nabla = \{y_t\}_{t=1}^T$ ) на допустимую область  $\mathcal{L} \cap \mathcal{T}$ , определяемой ограничениями (2.9)–(2.10). Матрица оператора проектирования на  $\mathcal{L}$  имеет вид

$$S_{\mathcal{L}} = E - S^T \cdot (S \cdot S^T)^{-1} \cdot S,$$

а проекция градиента на  $\mathcal{L}$  равна  $\nabla_{\mathcal{L}} = S_{\mathcal{L}} \cdot \nabla$ . Если внешняя нормаль на какой-либо грани параллелепипеда образует острый угол с проекцией градиента  $\nabla_{\mathcal{L}}$ , то перемещение по этой грани равно нулю.

---

### Algorithm 2 . DualWLDMSolver

---

**Require:** :

$\nabla_L$  ▷ Проекция градиента  
 $\{p_t \in \mathbb{R}^+\}_{t=1}^T$  ▷ Весовые коэффициенты

**Ensure:** :

$w^* = \arg \max_{w \in \mathbb{R}^T} \sum_{i=1}^T w_i \cdot y_i$  ▷ Оптимальное решение двойственной задачи

$R^* = \{t \in T : |w_t^*| = p_t\}$  ▷ Несколько активных ограничений

1:  $w \leftarrow \{w_i = 0 : i = 1, 2, \dots, T\}; R \leftarrow \emptyset; g = \nabla_L$  ▷ Начальная точка

2: **while**  $(\alpha_* \neq 0)$  **do**

3:  $\{(\alpha_*, t_*) \leftarrow \arg \max_{\substack{\alpha \in \mathbb{R}^+ \\ t \notin R}} \{\alpha \geq 0 : -p_t \leq w_t + \alpha g_t \leq p_t\}\}$  ▷ Параметры шага

4:  $w \leftarrow w + \alpha_* g; g_{t_*} \leftarrow 0; R := R \cup \{t_*\};$  ▷ Переход к следующей точке

5: **end while**

6:  $w^* = w, R^* = R$

**return**  $(w^*, R^*)$

---

Алгоритм 2 **DualWLDMSolver** используется для решения задачи двойственной оптимизации, описанной в уравнениях (2.8)–(2.10). Этот алгоритм применяет метод проекционного градиента для нахождения оптимального решения на пересечении линейного подпространства и параллелепипеда, что делает его подходящим для выпуклых задач опти-

мизации.

Вычислительная сложность такого алгоритма не превышает  $O(T^2)$  из-за простой структуры допустимого множества: пересечение  $T$ -мерного кубоида (2.10) и  $(T - n(m))$ -мерное линейное подпространство (2.9). Если  $(w^*, R^*)$  – результат выполнения алгоритма 2, то  $w^*$  – оптимальное решение задачи (2.8)–(2.10), а оптимальное решение задачи (2.5)–(2.7) равно

$$u_t^* = \frac{p_t + w_t^*}{2}, \quad v_t^* = \frac{p_t - w_t^*}{2}, \quad t = 1, 2, \dots, T.$$

Это следует из условия дополненности пары взаимно двойственных задач [33] (2.2)–(2.4) и (2.5)–(2.7) такие, что

$$y_t = \sum_{j=1}^{n(m)} [a_j g_j(\{y_{t-k}\}_{k=1}^m)] \quad \forall t \notin R^*, \quad (2.11)$$

$$y_t = \sum_{j=1}^{n(m)} [a_j g_j(\{y_{t-k}\}_{k=1}^m)] + z_t^*, \quad \forall t \in R^* : w_t^* = p_t, \quad (2.12)$$

$$y_t = \sum_{j=1}^{n(m)} [a_j g_j(\{y_{t-k}\}_{k=1}^m)] - z_t^*, \quad \forall t \in R^* : w_t^* = -p_t. \quad (2.13)$$

Фактически решение  $(\{a_j^*\}_{j=1}^{n(m)}, z^*)$  системы линейных алгебраических уравнений (2.11)–(2.13) представляет собой двойственное оптимальное решение задачи (2.8)–(2.10) и оптимальное решение задачи (2.1), что доказывает справедливость следующей теоремы.

**Теорема 1.** Пусть  $w^*$  – оптимальное решение задачи (2.8)–(2.10),  $(\{a_j^*\}_{j=1}^{n(m)}, z^*)$  – решение системы линейных алгебраических уравнений (2.11)–(2.13), то  $(\{a_j^*\}_{j=1}^{n(m)})$  – оптимальное решение задачи (2.1).

*Доказательство.* Из выпуклости вверх и дифференцируемости функции  $\arctan$  следует, что

$$\arctan |z_t| > \arctan |z| - \frac{|z|}{1 + (z_t)^2} + \frac{|z_t|}{1 + (z_t)^2}, \quad z \neq z_t. \quad (2.14)$$

Рассмотрим два последовательно решаемых варианта  $\mathcal{P}_k$  и  $\mathcal{P}_{k+1}$  задачи (2.2)-(2.4) (см. строку 13 алгоритма (4)). Варианты  $\mathcal{P}_k$  и  $\mathcal{P}_{k+1}$  отличаются только наборами весовых коэффициентов. Оптимальное решение  $\mathcal{P}_k$  является правильным, но не оптимальным решением  $\mathcal{P}_{k+1}$ , и мы имеем

$$\sum_{t=1}^T \frac{|z_t^{(k)}|}{1 + (z_t^{(k)})^2} \geq \sum_{t=1}^T \frac{|z_t^{(k+1)}|}{1 + (z_t^{(k+1)})^2} \quad (2.15)$$

поскольку  $\mathcal{P}_{k+1}$  веса являются

$$\left\{ p_t^{(k)} = \frac{1}{1 + (z_t^{(k)})^2} \right\}$$

(см. строки 9–12 алгоритма 4).

Из (2.17) и (2.18) следует, что

$$\sum_{t=1}^T \arctan |z_t^{(k)}| \geq \sum_{t=1}^T \arctan |z_t^{(k+1)}| \quad (2.16)$$

Таким образом, последовательность

$$\left\{ \sum_{t=1}^T \arctan \left| y_t - \sum_{j=1}^{n(m)} a_j^{(k)} g_j \{y_{t-k}\}_{k=1}^m \right| \right\}_{k=0,1,\dots}$$

монотонно убывает и ограничена снизу нулем, поэтому можно заключить, что эта последовательность имеет единственную предель-

ную точку. Существование предельной точки последовательности  $a_1^{(k)}, a_2^{(k)}, a_3^{(k)} \dots, a_{n(m)}^{(k)}$ ,  $k = 1, 2, \dots$  следует из непрерывности и монотонности функции  $\arctan(*)$ .

Докажем, что предельная точка  $(a_1^*, a_2^*, a_3^*, \dots, a_{n(m)}^*)$  построенная по алгоритму 4, является точкой глобального минимума.

Предположим противное:  $\tilde{a} = (\tilde{a}_1, \tilde{a}_2, \tilde{a}_3, \dots, \tilde{a}_{n(m)})$  – глобальный минимум функции потерь (1.32), а

$$\sum_{t=1}^T \arctan |\tilde{z}_t| < \sum_{t=1}^T \arctan |z_t^*|,$$

Очевидно, точка  $(\tilde{a}, \tilde{z})$  является некоторой предельной точкой последовательности, построенной по алгоритму 4, то из (2.17) следует, что для  $z^* \neq \tilde{z}$

$$\arctan |\tilde{z}_t| - \arctan |z_t^*| > \frac{|\tilde{z}_t|}{1 + \tilde{z}_t^2} - \frac{|z_t^*|}{1 + z_t^{*2}}$$

С другой стороны

$$\arctan |\tilde{z}_t| - \arctan |z_t^*| = \frac{|\tilde{z}_t| - |z_t^*|}{1 + |\tilde{z}_t| \cdot |z_t^*|}, \quad t = 1, 2, \dots, T,$$

поэтому

$$\frac{|\tilde{z}_t| - |z_t^*|}{1 + |\tilde{z}_t| \cdot |z_t^*|} > \frac{|\tilde{z}_t| - |z_t^*|}{1 + \tilde{z}_t^2}$$

следовательно,

$$0 < \frac{|\tilde{z}_t| - |z_t^*|}{1 + |\tilde{z}_t| \cdot |z_t^*|} - \frac{|\tilde{z}_t| - |z_t^*|}{1 + \tilde{z}_t^2} = (|\tilde{z}_t| - |z_t^*|) \cdot \left( \frac{1}{1 + |\tilde{z}_t| \cdot |z_t^*|} - \frac{1}{1 + \tilde{z}_t^2} \right) = \frac{(|\tilde{z}_t| - |z_t^*|)^2 \cdot |\tilde{z}_t|}{(1 + \tilde{z}_t^2) \cdot (1 + z_t^{*2})} \quad t = 1, 2, \dots, T.$$

По этой причине  $|\tilde{z}_t| > 0$  для всех  $t = 1, 2, \dots, T$ . Но это противоречит

предположению о том, что  $(a_1^*, a_2^*, a_3^*, \dots, a_{n(m)}^*)$  не является глобальным минимумом.

Теорема доказана.  $\square$

Вышеизложенное позволяет нам предложить алгоритм 3 оценки WLDM.

Алгоритм **WLDM-estimator**, основанный на взвешенном методе наименьших отклонений, предоставляет точные и оптимальные решения для прогнозирования временных рядов, основываясь на решении двойственной задачи и минимизации невязок в системе линейных уравнений. Этот подход гарантирует высокую точность результатов и эффективность при работе с данными.

---

### Algorithm 3 . WLDM-estimator

---

**Require: :**

$$S = \{S_t \in \mathbb{R}^N\}_{t \in T}$$

▷ Матрица линейного подпространства  $\mathcal{L}$

$$\nabla_{\mathcal{L}}$$

▷ Проекция градиента на  $\mathcal{L}$

$$\{p_t \in \mathbb{R}^+\}_{t=1}^T$$

▷ Веса

$$\{y_t \in \mathbb{R}^+\}_{t=1-m}^T$$

▷ Значения переменных

**Ensure: :**

$$A^* \in \mathbb{R}^{n(m)}$$

▷ Оптимальное решение прямой задачи

$$z^* \in \mathbb{R}^T$$

▷ Ограничения

$$1: (w^*, R^*) \leftarrow \text{DualWLDMSolver} \left( \nabla, \{p_t \in \mathbb{R}^+\}_{t=1}^T \right)$$

$$2: S^* \leftarrow \{S_t : t \notin R^*\}; y^* \leftarrow \{y_t : t \notin R^*\}$$

▷ Матрица системы (2.11)

$$3: (A^*)^T \leftarrow y^* \cdot (S^*)^{-1}$$

▷ Решение системы (2.11)

$$4: z^* \leftarrow (A^*)^T S - y$$

▷ Невязки

5: **Output:**  $(A^*, z^*)$

---

Основной проблемой при использовании алгоритма оценивания параметров WLDM является отсутствие общих формальных правил выбора весовых коэффициентов. Следовательно, этот подход требует дополнительных исследований.

Установленные в [104], [102] результаты позволяют свести задачу определения GLDM-оценки к итерационной процедуре с WLDM-

оценками.

## 2.2 Алгоритм GLDM-оценки

Задача (1.32) GLDM-оценки является задачей вогнутой оптимизации. GLDM-оценки устойчивы к наличию корреляции значений в  $\{y_t \in \mathbb{R}\}_{t=1-m}^T$ , и (с соответствующими настройками) как лучшая для вероятностных распределений ошибок с более тяжелыми (чем у нормального распределения) хвостами [99]. Вышеизложенное показывает возможность решения задачи идентификации (1.31) с помощью алгоритма (3).

---

### Algorithm 4 . GLDM-estimator

---

**Require:** :

$$S = \{S_t \in \mathbb{R}^N\}_{t \in T}$$

▷ Матрица линейного подпространства  $\mathcal{L}$

$$\nabla_L$$

▷ Проекция градиента на  $\mathcal{L}$

$$\{p_t \in \mathbb{R}^+\}_{t=1}^T$$

▷ Весовые коэффициенты

$$\{y_t \in \mathbb{R}^+\}_{t=1-m}^T$$

▷ Значения заданных переменных состояния

**Ensure:** :

$$A^* \in \mathbb{R}^{n(m)}$$

▷ Оптимальное решение GLDM

$$z^* \in \mathbb{R}^T$$

▷ Остатки

```

1:  $p \leftarrow \{p_t = 1 : t = 1, 2, \dots, T\}$ 
2:  $(A^{(1)}, z^{(1)}) \leftarrow \mathbf{WLDMSolver}(S, \nabla_{\mathcal{L}}, \{p_t\}_{t=1}^T, \{y_t\}_{t=1-m}^T);$ 
3: for all  $(t = 1, 2, \dots, T)$  do
4:    $p_t \leftarrow \left(1 / \left(1 + (z_t^{(1)})^2\right)\right)$ 
5: end for
6:  $(A^{(2)}, z^{(2)}) \leftarrow \mathbf{WLDMSolver}(S, \nabla_{\mathcal{L}}, \{p_t\}_{t=1}^T, \{y_t\}_{t=1-m}^T);$ 
7:  $k \leftarrow 2$ 
8: while  $(A^{(k)} \neq A^{(k-1)})$  do
9:   for all  $(t = 1, 2, \dots, T)$  do
10:     $p_t^{(k)} \leftarrow \left(1 / \left(1 + (z_t^{(k)})^2\right)\right)$ 
11:   end for
12:    $(A^{(k+1)}, z^{(k+1)}) \leftarrow \mathbf{WLDMSolver}(S, \nabla_{\mathcal{L}}, \{p_t^{(k)}\}_{t=1}^T, \{y_t\}_{t=1-m}^T);$ 
13:    $k \leftarrow (k + 1)$ 
14: end while
15:  $z^* \leftarrow z^{(k)}, (A^*) \leftarrow A^{(k)}$ 
return  $(A^*, z^*)$ 

```

▷ Поиск ограничений

---

Установленные результаты позволяют свести задачу определения

оценки GLDM к итерационной процедуре с оценками WLDM (см. алг. 4).

Алгоритм **GLDM-estimator** выполняет итеративную процедуру пересчета весов и оптимизации коэффициентов, основываясь на остатках каждой итерации. Каждый шаг приближает решение к оптимальному результату, используя обобщенный метод наименьших отклонений (GLDM).

**Теорема 2.** [33] Последовательность  $\{(A^{(k)}, z^{(k)})\}_{k=1}^{\infty}$ , построенная по алгоритму GLDM-оценки (см. Теорему 2.1) сходится к глобальному минимуму  $(a^*, z^*)$  задачи (1.32).

*Доказательство.* Из выпуклости вверх и дифференцируемости функции  $\arctan$  следует, что

$$\arctan |z_t| > \arctan |z| - \frac{|z|}{1 + (z_t)^2} + \frac{|z_t|}{1 + (z_t)^2}, \quad z \neq z_t. \quad (2.17)$$

Рассмотрим два последовательно решаемых варианта  $\mathcal{P}_k$  и  $\mathcal{P}_{k+1}$  задачи (2.2)-(2.4) (см. строку 13 алгоритма (4)). Варианты  $\mathcal{P}_k$  и  $\mathcal{P}_{k+1}$  отличаются только наборами весовых коэффициентов. Оптимальное решение  $\mathcal{P}_k$  является правильным, но не оптимальным решением  $\mathcal{P}_{k+1}$ , и мы имеем

$$\sum_{t=1}^T \frac{|z_t^{(k)}|}{1 + (z_t^{(k)})^2} \geq \sum_{t=1}^T \frac{|z_t^{(k+1)}|}{1 + (z_t^{(k)})^2} \quad (2.18)$$

поскольку  $\mathcal{P}_{k+1}$  веса являются

$$\left\{ p_t^{(k)} = \frac{1}{1 + (z_t^{(k)})^2} \right\}$$

(см. строки 9–12 алгоритма 4).

Из (2.17) и (2.18) следует, что

$$\sum_{t=1}^T \arctan |z_t^{(k)}| \geq \sum_{t=1}^T \arctan |z_t^{(k+1)}| \quad (2.19)$$

Таким образом, последовательность

$$\left\{ \sum_{t=1}^T \arctan \left| y_t - \sum_{j=1}^{n(m)} a_j^{(k)} g_j \{y_{t-k}\}_{k=1}^m \right| \right\}_{k=0,1,\dots}$$

монотонно убывает и ограничена снизу нулем, поэтому можно заключить, что эта последовательность имеет единственную предельную точку. Существование предельной точки последовательности  $a_1^{(k)}, a_2^{(k)}, a_3^{(k)}, \dots, a_{n(m)}^{(k)}$ ,  $k = 1, 2, \dots$  следует из непрерывности и монотонности функции  $\arctan(*)$ .

Докажем, что предельная точка  $(a_1^*, a_2^*, a_3^*, \dots, a_{n(m)}^*)$  построенная по алгоритму 4, является точкой глобального минимума.

Предположим противное:  $\tilde{a} = (\tilde{a}_1, \tilde{a}_2, \tilde{a}_3, \dots, \tilde{a}_{n(m)})$  – глобальный минимум функции потерь (1.32), а

$$\sum_{t=1}^T \arctan |\tilde{z}_t| < \sum_{t=1}^T \arctan |z_t^*|,$$

Очевидно, точка  $(\tilde{a}, \tilde{z})$  является некоторой предельной точкой последовательности, построенной по алгоритму 4, то из (2.17) следует, что для  $z^* \neq \tilde{z}$

$$\arctan |\tilde{z}_t| - \arctan |z_t^*| > \frac{|\tilde{z}_t|}{1 + \tilde{z}_t^2} - \frac{|z_t^*|}{1 + z_t^{*2}}$$

С другой стороны

$$\arctan |\tilde{z}_t| - \arctan |z_t^*| = \frac{|\tilde{z}_t| - |z_t^*|}{1 + |\tilde{z}_t| \cdot |z_t^*|}, \quad t = 1, 2, \dots, T,$$

поэтому

$$\frac{|\tilde{z}_t| - |z_t^*|}{1 + |\tilde{z}_t| \cdot |z_t^*|} > \frac{|\tilde{z}_t| - |z_t^*|}{1 + \tilde{z}_t^2}$$

следовательно,

$$0 < \frac{|\tilde{z}_t| - |z_t^*|}{1 + |\tilde{z}_t| \cdot |z_t^*|} - \frac{|\tilde{z}_t| - |z_t^*|}{1 + \tilde{z}_t^2} = (|\tilde{z}_t| - |z_t^*|) \cdot \left( \frac{1}{1 + |\tilde{z}_t| \cdot |z_t^*|} - \frac{1}{1 + \tilde{z}_t^2} \right) = \frac{(|\tilde{z}_t| - |z_t^*|)^2 \cdot |\tilde{z}_t|}{(1 + \tilde{z}_t^2) \cdot (1 + |z_t^{*2}|)} \quad t = 1, 2, \dots, T.$$

По этой причине  $|\tilde{z}_t| > 0$  для всех  $t = 1, 2, \dots, T$ . Но это противоречит предположению о том, что  $(a_1^*, a_2^*, a_3^*, \dots, a_{n(m)}^*)$  является глобальным минимумом.

Теорема доказана. □

Описание алгоритма **GLDM-оценка** показывает, что его вычислительная сложность пропорциональна вычислительной сложности алгоритма решения прямой и/или двойственной задачи WLDM (2.1). Многократные вычислительные эксперименты показывают, что среднее число итераций алгоритма **GLDM-оценка** равно количеству коэффициентов в идентифицированном уравнении. Если эта гипотеза верна, то вычислительная сложность решения практических задач не превышает  $O((n(m))^3 T + n(m) \cdot T^2)$ . Необходимо учитывать, что поиск и нахождение уравнения авторегрессии высокого порядка имеют свои специфические условия. Одним из таких условий, в частности, является высокая чувствительность алгоритма к ошибкам округления. Чтобы исключить возможность ошибки в вычислениях, необходимо точно выполнять основные арифметические действия над полем рациональных чисел [105] и дополнить их распараллеливанием.

## 2.3 Прогнозирование

Алгоритм 5 **PredictorEstimator** предназначен для прогнозирования значений временного ряда на основе модельных коэффициентов, а также для оценки точности прогноза с помощью метрик средней систематической ошибки (МВЕ), средней абсолютной ошибки (МАН) и минимального горизонта прогнозирования (minFH).

**Шаг 1: Инициализация.** Алгоритм инициализирует индекс старта (*startIndex*) и начальные прогнозы для первых значений временного ряда. Этот шаг необходим для подготовки начальных условий для следующего прогноза.

**Шаг 2: Прогнозирование значений.** Для каждого временного шага алгоритм использует модельные коэффициенты  $A = \{a_i\}_{i=1}^{n(m)}$  для вычисления прогнозов на основе предыдущих значений. Прогноз формируется с использованием модельных функций  $G_j$ , которые зависят от предыдущих прогнозов. Если ошибка между прогнозируемым и фактическим значением превышает допустимый порог, процесс прогнозирования для данного временного ряда прекращается.

**Шаг 3: Обновление горизонта прогнозирования.** Алгоритм обновляет горизонт прогнозирования (*ForecastHorizon*), который показывает, на сколько шагов вперед можно достоверно прогнозировать значения.

**Шаг 4: Вычисление метрик ошибок.** После завершения прогнозирования для всех временных шагов вычисляются *средняя систематическая ошибка (МВЕ)* – среднее отклонение прогнозируемых значений от фактических; *средняя абсолютная ошибка (МАН)* – среднее абсолютное отклонение прогнозируемых значений от фактических. Алгоритм также вычисляет минимальный разумный горизонт прогнози-

рования ( $\min FH$ ), который показывает, сколько шагов вперед можно с уверенностью делать прогноз.

**Шаг 5: Возврат результатов** Алгоритм предоставляет значения МВЕ, МАЕ и  $\min FH$  для оценки качества модели и точности прогнозов.

Таким образом, алгоритм прогнозирует временные ряды с динамическим обновлением весов и метрик, оценивая качество и глубину прогноза.

Алгоритм формирует индексированное  $t = 1, 2, \dots, T - 1, T$  семейство разностных уравнений  $m$ -го порядка (1.33) для решетчатых функций  $\overline{y[t]}$  со значениями  $\overline{y[t]}_\tau$ , что интерпретируется как построенный в момент времени  $t$  прогноз для  $y_\tau$ . Воспользуемся решением задачи Коши для ее разностного уравнения (1.33) при начальных условиях (1.34), для нахождения значений функции  $\overline{y[t]}$ . Итак, имеем множество  $\overline{Y}_\tau = \left\{ \overline{y[t]}_\tau \right\}_{t=1}^T$  возможного предсказания значения  $y_\tau$ . В дальнейшем используем этот набор для оценки вероятностных характеристик величины  $y_\tau$ .

---

**Algorithm 5 PredictorEstimator Algorithm**


---

**Require:** Time series data  $Y = \{y_t \in \mathbb{R}^+\}_{t=1-m}^T$

**Require:** Model coefficients  $A = \{a_i\}_{i=1}^{n(m)}$

**Ensure:** Predicted values Predictions[1 : T][1 : T], mean bias error MBE, mean absolute error MAE, minimum forecast horizon minFH

```

1: Initialization:
2: Set starting index: startIndex  $\leftarrow$  0
3: while ForecastHorizon[startIndex] <  $m$  do
4:   startIndex  $\leftarrow$  startIndex + 1 ▷ Увеличить начальный индекс
5:   Predictions[startIndex][0]  $\leftarrow$  Y[startIndex]
6:   Predictions[startIndex][1]  $\leftarrow$  Y[startIndex + 1] ▷ Установить начальные прогнозы
7:   for  $t \leftarrow$  startIndex + 2 to  $m$  do
8:     prediction  $\leftarrow$  0 ▷ Инициализировать значение прогноза
9:     for  $j \leftarrow$  0 to  $n$  do ▷ Вычислить модельные значения
10:       $A1 \leftarrow G_j$ (Predictions[startIndex][ $t - 1$ ], Predictions[startIndex][ $t - 2$ ])
11:      prediction  $\leftarrow$  prediction +  $a_j \times A1$  ▷ Накопить прогноз
12:    end for
13:    Predictions[startIndex][ $t$ ]  $\leftarrow$  prediction ▷ Сохранить результат прогноза
14:    if |Predictions[startIndex][ $t$ ] - Y[startIndex +  $t$ ]| > Threshold then
15:      break ▷ Прервать, если ошибка превышает порог
16:    end if
17:  end for
18:  ForecastHorizon[startIndex]  $\leftarrow$   $t$  ▷ Обновить горизонт прогнозирования
19: end while

20: Compute Error Metrics:
21: LastStartIndex  $\leftarrow$   $t$ 
22: minFH  $\leftarrow$  ForecastHorizon[startIndex]
23: tempMinFH  $\leftarrow$  minFH
24: for  $t \leftarrow$  3 to startIndex do
25:   if minFH > ForecastHorizon[ $t$ ] then
26:     tempMinFH  $\leftarrow$  ForecastHorizon[ $t$ ]
27:   end if
28: end for
29: minFH  $\leftarrow$  min(tempMinFH, minFH) ▷ Определить минимальный горизонт прогнозирования
30: MBE  $\leftarrow$  0 ▷ Инициализировать среднюю систематическую ошибку
31: MAE  $\leftarrow$  0 ▷ Инициализировать среднюю абсолютную ошибку
32: for  $t \leftarrow$  3 to minFH do
33:   MAE  $\leftarrow$  MAE + |Y[ $t +$  startIndex] - Predictions[startIndex][ $t$ ]|
34:   MBE  $\leftarrow$  MBE + (Y[ $t +$  startIndex] - Predictions[startIndex][ $t$ ])
35: end for
36: MBE  $\leftarrow$  MBE/minFH ▷ Вычислить среднюю систематическую ошибку
37: MAE  $\leftarrow$  MAE/minFH ▷ Вычислить среднюю абсолютную ошибку

38: Return (MBE, MAE, minFH)

```

---

## 2.4 Анализ коэффициентов модели

В контексте прогнозирования временных рядов с использованием моделей более высокого порядка часто наблюдается, что ряд коэффициентов может быть равен нулю. Данную ситуацию можно объяснить несколькими математическими и статистическими причинами, которые имеют решающее значение при оптимизации точности прогнозирования модели и эффективности вычислений.

Существуют временные ряды определенной структуры, при которой модель будет иметь нулевые коэффициенты. Такое происходит, если ряд значений не вносит существенного вклада в прогнозирование будущих значений.

Обнуление коэффициента можно обосновать математически путем анализа градиента целевой функции по отношению к каждому коэффициенту в контексте обобщенного метода наименьших отклонений (GLDM) [21].

Рассмотрим целевую функцию

$$F(\mathbf{a}) = \sum_{t=1}^T \arctan |y_t - f(\mathbf{x}_t, \mathbf{a})|, \quad (2.20)$$

где  $y_t$  – реальные значения,  $\mathbf{x}_t$  – вектор временных отсчетов  $t$ , а  $f(\mathbf{x}_t, \mathbf{a})$  – модельные значения, описанные выражением (1.31).

Чтобы определить, в каких случаях коэффициент  $a_j$  обращается в

ноль, исследуем частную производную функции  $F(\mathbf{a})$  по  $a_j$ :

$$\frac{\partial F}{\partial a_j} = \sum_{t=1}^T \frac{\partial}{\partial a_j} \arctan |y_t - f(\mathbf{x}_t, \mathbf{a})|. \quad (2.21)$$

Данную производную можно записать как

$$\frac{\partial F}{\partial a_j} = \sum_{t=1}^T \frac{1}{1 + (y_t - f(\mathbf{x}_t, \mathbf{a}))^2} \cdot \frac{\partial}{\partial a_j} (y_t - f(\mathbf{x}_t, \mathbf{a})). \quad (2.22)$$

Если она равна нулю, это означает, что

$$\sum_{t=1}^T \frac{1}{1 + (y_t - f(\mathbf{x}_t, \mathbf{a}))^2} \cdot \frac{\partial}{\partial a_j} (y_t - f(\mathbf{x}_t, \mathbf{a})) = 0. \quad (2.23)$$

Условие (2.23) указывает на то, что вклад коэффициента  $a_j$  в значение ошибки пренебрежимо мал, следовательно,  $a_j$  может быть установлен равным нулю без существенного влияния на качество модели. Данный подход не только упрощает модель за счет сокращения числа активных параметров, но и повышает эффективность вычислений и интерпретируемость модели.

Такое заключение имеет решающее значение для понимания структуры и поведения моделей временных рядов высокого порядка, что определяет правило выбора моделей и интерпретацию их результатов [21].

#### **2.4.1 Выбор оптимального порядка модели для заданного временного ряда**

Выбор оптимального порядка построения модели при прогнозировании временных рядов с использованием обобщенного метода наименьших отклонений (GLDM) имеет решающее значение для обеспечения ба-

ланса между сложностью модели и ее способностью обобщаться для прогнозирования новых значений. Приведенные ниже рассуждения описывают методологию определения оптимального порядка  $m$  модели путем минимизации функции потерь, которая учитывает компромисс между смещением, дисперсией и вычислительной сложностью. Требуется минимизировать целевую функцию (2.20) [21].

Очевидно, что сложность модели возрастает с увеличением порядка  $m$ , что позволяет ей отображать временные ряды более сложной структуры. Однако повышенная сложность может привести к возникновению переобучения, особенно когда не достаточно данных для поддержания большего числа параметров. Ожидаемая ошибка прогнозирования (Expected Prediction Error, EPE) теоретически выражается следующим образом:

$$\text{EPE} = E_B^2(m) + E_V(m) + E_{IE}, \quad (2.24)$$

где  $E_B$  – ошибка смещения,  $E_V(m)$  – ошибка отклонения,  $E_{IE}$  – неснижаемая ошибка. Известно, что смещение уменьшается с ростом  $m$ , а дисперсия увеличивается, что иллюстрирует критический компромисс.

Оптимальный порядок  $m^*$  модели определяется как порядок, который минимизирует функцию потерь по всем возможным значениям  $\mathbf{a}$ :

$$m^* = \arg \min_m \min_{\mathbf{a}} \{\mathcal{L}(m, \mathbf{a})\}. \quad (2.25)$$

Процесс оптимизации порядка модели включает в себя вычисление значения  $\mathcal{L}(m, \mathbf{a})$  для диапазона значений  $m$ , с целью определить порядок построения модели, который обеспечивает наилучший баланс между точностью и сложностью. Оптимальный порядок модели  $m^*$  определяется путем систематической оценки, которая уравнивает слож-

ность модели и ее соответствие данным, руководствуясь функцией потерь. Этот метод гарантирует, что выбранная модель является статистически надежной и эффективной, способной давать надежные прогнозы, оставаясь при этом устойчивой к изменениям в невидимых данных [21].

### 2.4.2 Алгоритм выявления значимых коэффициентов модели

Алгоритм 6 [21] определяет и выделяет незначимые коэффициенты.

---

#### Algorithm 6 Поиск коэффициентов модели

---

**Require:** Временной ряд  $\{y_t\}$  длины  $T$ , порядок модели  $m$ , функции  $g_i$  — базисные функции (1.6.2)

**Ensure:** Вектор коэффициентов  $\mathbf{a}$

$n(m) \leftarrow \frac{m(m+3)}{2}$  ▷ Число коэффициентов

$\mathbf{a}_i \leftarrow 0, i = 1, n(m)$  ▷ Инициализация

$\mathbf{a} \leftarrow \text{GLDM-estimator}(y)$

$F(\mathbf{a}) \leftarrow \sum_{t=1}^T \arctan \left| y_t - \sum_{j=1}^{n(m)} a_j g_j(\{y_{t-k}\}_{k=1}^m) \right|$  ▷ Значение целевой функции

**while**  $|F(\mathbf{a})| > \epsilon$  **do** ▷ Проверка сходимости

**for**  $j = 1$  **to**  $n(m)$  **do**

$$\nabla F_{a_j} = \frac{\partial}{\partial a_j} \sum_{t=1}^T \arctan \left| y_t - \sum_{k=1}^{n(m)} a_k g_k(\{y_{t-i}\}_{i=1}^m) \right|$$

▷ Вычисление градиента

**if**  $|\nabla F_{a_j}| < \epsilon$  **then** ▷  $\epsilon$  — пороговое значение градиента

$a_j \leftarrow 0$  ▷ Обнуление коэффициента

**end if**

**end for**

**end while**

**return**  $\mathbf{a}$  ▷ Вывод результата

---

Алгоритм состоит из следующих этапов.

1. **Инициализация:** Алгоритм вычисляет  $n(m)$  коэффициентов. Он обнуляет эти коэффициенты и определяет целевую функцию  $F(\mathbf{a})$ , которая минимизирует суммарный арктангенс модулей разностей

между фактическими данными временных рядов  $\{y_t\}$  и модельными значениями.

2. **Вычисление градиента:** Для каждого коэффициента  $a_j$  рассчитывается значение градиента целевой функции по  $a_j$ . Этот градиент показывает, насколько мало приращения значения  $a_j$  влияют на приращение значения  $F(\mathbf{a})$ .
3. **Обнуление коэффициентов:** Если абсолютное значение градиента коэффициентов ниже небольшого порогового значения  $\epsilon$ , этот коэффициент полагается равным нулю, поскольку влияние коэффициента с минимальным значением градиента на модель незначительно, и его обнуление упростит модель, не оказывая существенного влияния на ее эффективность.
4. **Проверка сходимости:** Алгоритм повторяется, обновляя коэффициенты, до тех пор, пока изменения в целевой функции  $|F(\mathbf{a})|$  не станут незначительными, т.е. меньше  $\epsilon$ , что указывает на сходимость.
5. **Вывод результата:** После завершения цикла алгоритм выводит оптимизированный вектор коэффициентов  $\mathbf{a}$ , где некоторые коэффициенты могут быть равны нулю, что соответствует потенциально более простой и поддающейся интерпретации модели.

Приведенный алгоритм может быть особенно полезен в тех случаях, когда решающее значение имеет баланс между сложностью и функциональной совместимостью модели и точностью, что позволяет создать оптимизированную модель, которая эффективно отражает существенную динамику данных [21].

### 2.4.3 Обоснование

В этом разделе рассматриваются теоретические обоснования выбранных порядков модели для различных наборов данных, обсуждается роль нулевых коэффициентов в повышении эффективности модели и рассматриваются потенциальные направления будущих исследований.

Как было показано ранее, выбор порядка построения модели существенно влияет на способность модели точно отражать динамику данных.

**Теорема 3.** [Выбор оптимальной модели][21] Пусть  $\{y_t\}_{t=1}^T$  – временной ряд, и функция потерь  $F(\mathbf{a})$  определена как:

$$F(\mathbf{a}) = \sum_{t=1}^T \arctan |y_t - f(\mathbf{a}, \mathbf{x}_t)|,$$

где функция  $f$  зависит от параметров  $\mathbf{a}$ . Тогда оптимальный порядок модели  $m^*$  равен:

$$m^* = \arg \min_m \left( \min_{\mathbf{a}} F(\mathbf{a}, m) \right).$$

*Доказательство.* [21] Рассмотрим дифференцируемую функцию  $F(\mathbf{a})$  и ее первые производные:

$$\frac{\partial F}{\partial a_i} = \sum_{t=1}^T \frac{-1}{1 + (y_t - f(\mathbf{a}, \mathbf{x}_t))^2} \cdot \frac{\partial f}{\partial a_i}(\mathbf{a}, \mathbf{x}_t),$$

которые равны нулю в критических точках  $\mathbf{a}_0$ , где  $\nabla F(\mathbf{a}) = \mathbf{0}$ .

Гессиан функции  $F(\mathbf{a})$ , матрица вторых производных, задается как:

$$H(\mathbf{a}) = \left[ \frac{\partial^2 F}{\partial a_i \partial a_j} \right] = \sum_{t=1}^T \left[ \frac{-2 \frac{\partial f}{\partial a_i} \frac{\partial f}{\partial a_j}}{(1 + (y_t - f(\mathbf{a}, \mathbf{x}_t))^2)^2} + \frac{\frac{\partial^2 f}{\partial a_i \partial a_j}}{1 + (y_t - f(\mathbf{a}, \mathbf{x}_t))^2} \right]. \quad (2.26)$$

Рассмотрим дифференцируемую функцию  $F(\mathbf{a})$ , определенную на множестве параметров  $\mathbf{a}$ , и пусть  $\mathbf{a}_0$  — критическая точка этой функции, где градиент  $\nabla F(\mathbf{a}_0) = \mathbf{0}$ .

Гессиан функции  $F(\mathbf{a})$ , который представляет собой матрицу вторых производных, в точке  $\mathbf{a}_0$  определяется следующим образом:

$$H(\mathbf{a}_0) = \left[ \frac{\partial^2 F}{\partial a_i \partial a_j}(\mathbf{a}_0) \right] = \sum_{t=1}^T \left[ \frac{-2 \frac{\partial f}{\partial a_i} \frac{\partial f}{\partial a_j}}{(1 + (y_t - f(\mathbf{a}_0, \mathbf{x}_t))^2)^2} + \frac{\frac{\partial^2 f}{\partial a_i \partial a_j}}{1 + (y_t - f(\mathbf{a}_0, \mathbf{x}_t))^2} \right].$$

Матрица Гессе функции  $F(\mathbf{a})$ , определяемая как матрица вторых производных в точке  $\mathbf{a}_0$ , имеет вид:

$$H(\mathbf{a}_0) = \begin{bmatrix} H_{11} & H_{12} & \dots & H_{1n} \\ H_{21} & H_{22} & \dots & H_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ H_{n1} & H_{n2} & \dots & H_{nn} \end{bmatrix},$$

где каждый элемент  $H_{ij}$  определяется следующим образом:

$$H_{ij} = \sum_{t=1}^T \left[ \frac{-2 \frac{\partial f}{\partial a_i} \frac{\partial f}{\partial a_j}}{(1 + (y_t - f(\mathbf{a}_0, \mathbf{x}_t))^2)^2} + \frac{\frac{\partial^2 f}{\partial a_i \partial a_j}}{1 + (y_t - f(\mathbf{a}_0, \mathbf{x}_t))^2} \right].$$

Для анализа положительной определенности матрицы  $H(\mathbf{a}_0)$ , рассмотрим квадратичную форму:

$$\mathbf{v}^T H(\mathbf{a}_0) \mathbf{v} = \sum_{i,j=1}^n v_i v_j \left( \sum_{t=1}^T \left[ \frac{-2 \frac{\partial f}{\partial a_i} \frac{\partial f}{\partial a_j}}{(1 + (y_t - f(\mathbf{a}_0, \mathbf{x}_t))^2)^2} + \frac{\frac{\partial^2 f}{\partial a_i \partial a_j}}{1 + (y_t - f(\mathbf{a}_0, \mathbf{x}_t))^2} \right] \right),$$

где  $\mathbf{v} = (v_1, \dots, v_n)$  — произвольный ненулевой вектор.

Положительная определенность  $H(\mathbf{a}_0)$  означает, что для всех ненулевых  $\mathbf{v}$  выражение  $\mathbf{v}^T H(\mathbf{a}_0) \mathbf{v}$  строго больше нуля. Это достигается, если сумма положительных слагаемых, связанных со вторыми производными  $\frac{\partial^2 f}{\partial a_i \partial a_j}$ , превышает сумму отрицательных слагаемых, возникающих из-за члена с произведением первых производных  $\frac{\partial f}{\partial a_i}$  и  $\frac{\partial f}{\partial a_j}$ .

Таким образом, можно заключить, что если  $\mathbf{v}^T H(\mathbf{a}_0) \mathbf{v} > 0$  для всех ненулевых векторов  $\mathbf{v}$ , то матрица  $H(\mathbf{a}_0)$  положительно определена, и, следовательно,  $\mathbf{a}_0$  является точкой локального минимума функции  $F(\mathbf{a})$ .

□

В этой теореме сформулированы условия определения оптимального порядка модели  $m^*$  для временного ряда  $\{y_t\}_{t=1}^T$  с использованием функции потерь  $F(\mathbf{a})$ . Функция  $F(\mathbf{a})$  вычисляется как сумма арктангенсов абсолютных разностей между фактическими значениями  $y_t$  и значениями, предсказанными моделью  $f(\mathbf{a}, \mathbf{x}_t)$ .

Следующая теорема устанавливает условия, при которых коэффициенты  $a_j$  модели могут быть обнулены без потери точности прогнозирования. Если производная  $F(\mathbf{a})$  по коэффициенту  $a_j$  равна нулю на всем интервале, это указывает на то, что изменения  $a_j$  не влияют на значение функции потерь, и коэффициент  $a_j$  может быть исключен из модели.

**Теорема 4.** *[Условия наличия нулевых коэффициентов модели] Если*

для коэффициента  $a_j$ :

$$\frac{\partial F(\mathbf{a})}{\partial a_j} = \sum_{t=1}^T \frac{\partial}{\partial a_j} \arctan |y_t - f(\mathbf{a}, \mathbf{x}_t)| = 0,$$

то  $a_j$  не влияет на функцию потерь  $F(\mathbf{a})$  и может быть обнулен.

*Доказательство.* Доказательство фокусируется на анализе чувствительности  $F(\mathbf{a})$  к изменениям в  $a_j$ . Рассмотрим чувствительность  $F(\mathbf{a})$  к изменениям  $a_j$ . Для этого используем разложение Тейлора до второго порядка:

$$\frac{\partial^2 F}{\partial a_j^2} = \sum_{t=1}^T \frac{-2 \left( \frac{\partial f}{\partial a_j} \right)^2}{(1 + (y_t - f(\mathbf{a}, \mathbf{x}_t))^2)^2}.$$

Поскольку  $\frac{\partial F}{\partial a_j} = 0$  и  $\frac{\partial^2 F}{\partial a_j^2} \approx 0$ , то влияние  $a_j$  на  $F$  минимально. Таким образом, значение коэффициента  $a_j$  можно считать нулевым без потери точности модели.  $\square$

Приведенные теоремы 3 и 4 описывают основные аспекты обобщенного метода наименьшего отклонения (GLDM) для анализа данных временных рядов. Они определяют метод определения эффективной структуры модели и определения избыточных коэффициентов модели.

#### 2.4.4 Сходимость, устойчивость и согласованность

Рассмотрим динамическое поведение модели квазилинейной регрессии, применяемой к данным временных рядов, с акцентом на адаптацию параметров модели с помощью итеративного алгоритма оптимизации. В частности, параметры  $\mathbf{a} \in \mathbb{R}^n$  обновляются итеративно таким образом, что величина их ошибок, обозначаемая как  $\epsilon$ , обеспечивает наличие всех трех критических аспектов [21]: (1) сходимость оценок параметров к их

оптимальным значениям, (2) согласованность параметров оценки по мере неограниченного расширения ряда данных и (3) устойчивость полученных прогнозов модели на последующих итерациях.

**Утверждение 4.** Пусть  $\{Y_t\}_{t=1}^T$  – временной ряд, смоделированный с помощью квазилинейной регрессии с параметрами  $\mathbf{a} \in \mathbb{R}^n$ . Пусть алгоритм итеративной оптимизации обновляет параметры с  $\mathbf{a}$  до  $\mathbf{a}'$  на каждой итерации таким образом, что  $\|\mathbf{a}' - \mathbf{a}\| \leq \epsilon$ , где  $\epsilon$  – это малый положительный скаляр. Эти обновления предназначены для обеспечения сходимости оценок параметров, согласованности параметров модели от  $T \rightarrow \infty$  и устойчивости выходных прогнозов на каждой итерации.

1. **Сходимость** демонстрируется путем систематического уменьшения целевой функции  $F(\mathbf{a})$ , которая количественно определяет соответствие модели исходным данным. В доказательстве описываются условия, при которых функция убывает, тем самым подтверждая эффективность обновления параметров. Если изменения параметров не уменьшаются, как ожидалось, рассматриваются корректировки параметров оптимизации или структуры алгоритмов.

**Сведение целевой функции.** Определим целевую функцию  $F(\mathbf{a})$  для измерения расхождения между прогнозами модели и фактическими значениями данных. Если алгоритм гарантирует, что:

$$F(\mathbf{a}^{(k+1)}) \leq F(\mathbf{a}^{(k)}) - \gamma \|\mathbf{a}^{(k+1)} - \mathbf{a}^{(k)}\|^2,$$

где  $\gamma$  – положительная константа, тогда целевая функция монотонно уменьшается, что гарантирует эффективное обновление параметров. В противном случае требуется повторно оценить значение  $\gamma$  или изменить

механизм обновления.

**Сходимость оценок параметров.** Если  $\|\mathbf{a}^{(k+1)} - \mathbf{a}^{(k)}\| \leq \epsilon$ , положим  $\epsilon$  уменьшается по мере того, как возрастает  $k$ , приводящий к сходимости  $\mathbf{a}^{(k)}$  к  $\mathbf{a}^*$ . В противном случае, если сходимость не наблюдается, рассмотреть возможность корректировки критерия сходимости или изучения других факторов, влияющих на сходимость, таких как скорость обучения или шум в данных.

**2. Согласованность** проверяется при условии корректной спецификации модели и независимых, одинаково распределенных ошибок. Она подтверждается приближением вероятности оценок параметров к истинным значениям по мере увеличения набора данных. Также обсуждаются альтернативные сценарии, связанные с неправильной спецификацией модели или ошибками, не связанными с идентификацией, которые предполагают дополнительные аналитические или методологические корректировки для поддержания согласованности.

**Спецификация модели и условия возникновения ошибок.** Если модель задана правильно и ошибки являются допустимыми, стандартные оценки подтверждают, что:

$$\mathbf{a}^{(k)} \xrightarrow{p} \mathbf{a}_0 \quad \text{as } T \rightarrow \infty,$$

гарантируя, что  $\mathbf{a}^{(k)}$  согласованы. В противном случае, если модель не задана должным образом или ошибки не учтены, могут потребоваться дополнительные исправления или применение более надежных методов для обеспечения согласованности оценок.

**3. Анализ устойчивости** основан на липшицевой непрерывности модельной функции, гарантирующей, что небольшие изменения параметров приводят к пропорционально малым изменениям выходных дан-

ных. В этой части доказательства рассматриваются условия, при которых модель устойчива относительно выходных данных, и обсуждаются возможные корректировки, если модельная функция не удовлетворяет условию Липшица.

В каждом разделе доказательства используются условия «если-иначе» для изучения различных сценариев, что обеспечивает изучение поведения модели в различных теоретических и практических условиях. Такой подход не только проясняет условия, необходимые для каждого свойства модели, но и повышает надежность применения модели к реальным данным.

**Условие Липшица для модельной функции.** Положим, что  $f(\mathbf{X}; \mathbf{a})$  – Липшиц непрерывная функция с постоянной величиной  $L$ , тогда небольшие изменения параметров приведут к небольшим, ограниченным изменениям выходных данных:

$$|Y_t(\mathbf{a}') - Y_t(\mathbf{a})| \leq L \|\mathbf{a}' - \mathbf{a}\|.$$

В противном случае, если  $f$  не является липшицевой, следует изучить, можно ли применить другие условия ограниченности или меры устойчивости для понимания и контроля чувствительности выходных данных.

## 2.5 Выводы по главе 2

Оригинальность рассмотренного алгоритма определения параметров квазилинейного рекуррентного уравнения заключается в том, что он вместо метода наименьших квадратов использует обобщенный метод наименьших модулей с подбором весов. Показано, что задача GLDM-оценки является задачей многоэкстремальной оптимизации, а GLDM-

оценки устойчивы к наличию корреляции значений в  $\{y_t \in \mathbb{R}\}_{t=1-m}^T$  и (при соответствующих настройках) лучше всего подходят для вероятностных распределений ошибок с более тяжелыми (чем у нормального распределения) хвостами.

Изложенное в данной главе показывает возможность решения задачи определения параметров уравнения (1.31) с использованием решения (1.32), а описание алгоритма **GLDM-оценка** показывает, что его вычислительная сложность пропорциональна вычислительной сложности алгоритма решения прямой и/или двойственной задачи WLDM (2.1).

Наличие нулевых коэффициентов в моделях более высокого порядка свидетельствует об эффективности метода исключения факторов, не влияющих на прогнозирование, что упрощает модели и повышает их интерпретируемость. Это исследование подтверждает ключевую роль выбора оптимального порядка построения модели и интерпретации коэффициентов модели в контексте прогнозирования временных рядов.

Проведенные вычислительные эксперименты показывают, что среднее число итераций алгоритма **GLDM-оценка** равно количеству коэффициентов в уравнении. Вычислительная сложность решения практических задач не превышает  $O((n(m))^3 T + n(m) \cdot T^2)$ , т.е. алгоритм позволяет получить решение задачи за сравнительно короткий срок.

Результаты данной главы опубликованы в [1, 21, 22, 33]. Программа для ЭВМ зарегистрирована Роспатентом [2, 3, 11–13].

Исходный код пакета `abotaleb1` для моделирования временных рядов методом обобщённых наименьших отклонений (GLDM) доступен на сайте <https://gitverse.ru/mostafa/GLDM>. Инструкции по установке и дополнительная информация представлены на странице PyPI: <https://pypi.org/project/abotaleb1/> [135].

## ГЛАВА 3. СИСТЕМА ДЛЯ ПРОГНОЗИРОВАНИЯ ОДНОМЕРНЫХ ВРЕМЕННЫХ РЯДОВ

Рассмотрим созданное программное обеспечение, позволяющее построить прогноз для выбранного пользователем периода с использованием любой из рассмотренных выше моделей. Для достижения этой цели было разработано пять программных модуля.

1. Система прогнозирования `Epidemic.TA`, позволяющая построить прогнозы для одномерного временного ряда с использованием регрессионных моделей, и модели SIR [29] [11]. В состав первой версии системы входила функция, использующая модель SIR, но в более поздних версиях данная функция была исключена, поскольку она имеет низкую прогностическую способность. Вместо SIR используется модель на основе нейронной сети NNAR [27, 28].
2. Система прогнозирования `DL-system`, использующий методы глубокого обучения для построения прогнозов для одномерного временного ряда [13, 84, 86];
3. Модуль `GLDM`, использующий методы определения параметров квазилинейного рекуррентного соотношения [1, 3, 10, 22, 30, 31, 33, 83, 89, 101, 114, 121].
4. Для работы с длинными временными рядами, требующими матриц большого порядка, в программном обеспечении реализована возможность использования двух видов параллелизма: функционального и параллелизма по данным. Реализация алгоритма прогнозирования [4, 12, 82, 88] поддерживает многопоточные вычисления и была протестирована на различных вычислительных платформах для повышения производительности.

**5. Программное обеспечение для одномерных временных рядов**, предназначенное для анализа и прогнозирования временных рядов с использованием различных моделей, таких как классические статистические методы (Auto ARIMA, экспоненциальное сглаживание, Prophet), гибридные модели (Auto ARIMA + экспоненциальное сглаживание и полиномиальная регрессия), методы машинного обучения (SVM, многослойный перцептрон) и обобщенный метод наименьшего отклонения (GLDM) с различными порядками зависимостей [2, 26].

Все модули реализованы на языках R (модуль `updated Epidemic.TA`), Python (модуль `DL-system`). Модуль GLDM – на языках Python и C++, кроме того, на языке C++ имеется параллельная версия алгоритма, реализованная с использованием библиотеки OpenMP. Программное обеспечение для одномерных временных рядов Версия 1– на языках Python и R.

### **3.1 Epidemic.TA для прогнозирования одномерных временных рядов**

Своим названием модуль обязан пандемии Covid-19, на данных которой было проведено его тестирование. Целью при разработке данной системы было создание алгоритма, позволяющего по имеющимся исходным данным о распространении коронавирусной инфекции в определенном регионе за заданный период времени определить наилучшую модель для составления прогноза на заданный пользователем период.

Алгоритм, заложенный в основу финальной версии системы, анализирует прогнозы на основе моделей временных рядов (ARIMA, линейная модель Хольта, BATS и TBATS) и модели нейронных сетей (NNAR)



Рисунок 3.1. – Глобальные переменные для запуска системы updated Epidemic.TA, использующей модели временных рядов и нейронные модели

и выбирает модель, которая дает прогноз с минимальной средней абсолютной ошибкой в процентах (MAPE). В статье [27–29, 34] описана программа на языке R, позволяющая получать прогноз с использованием описанных выше моделей, либо с последовательным использованием нескольких из них с оптимизатором весов для временных рядов. Критерием для выбора лучшей модели является минимальная ошибка MAPE.

На рисунке 3.1 приведен перечень глобальных переменных для запуска системы прогнозирования.

На рисунке 3.2 приводится схема разработанной updated системы Epidemic.TA, позволяющей выбрать лучшую модель для имеющихся исходных данных [29].

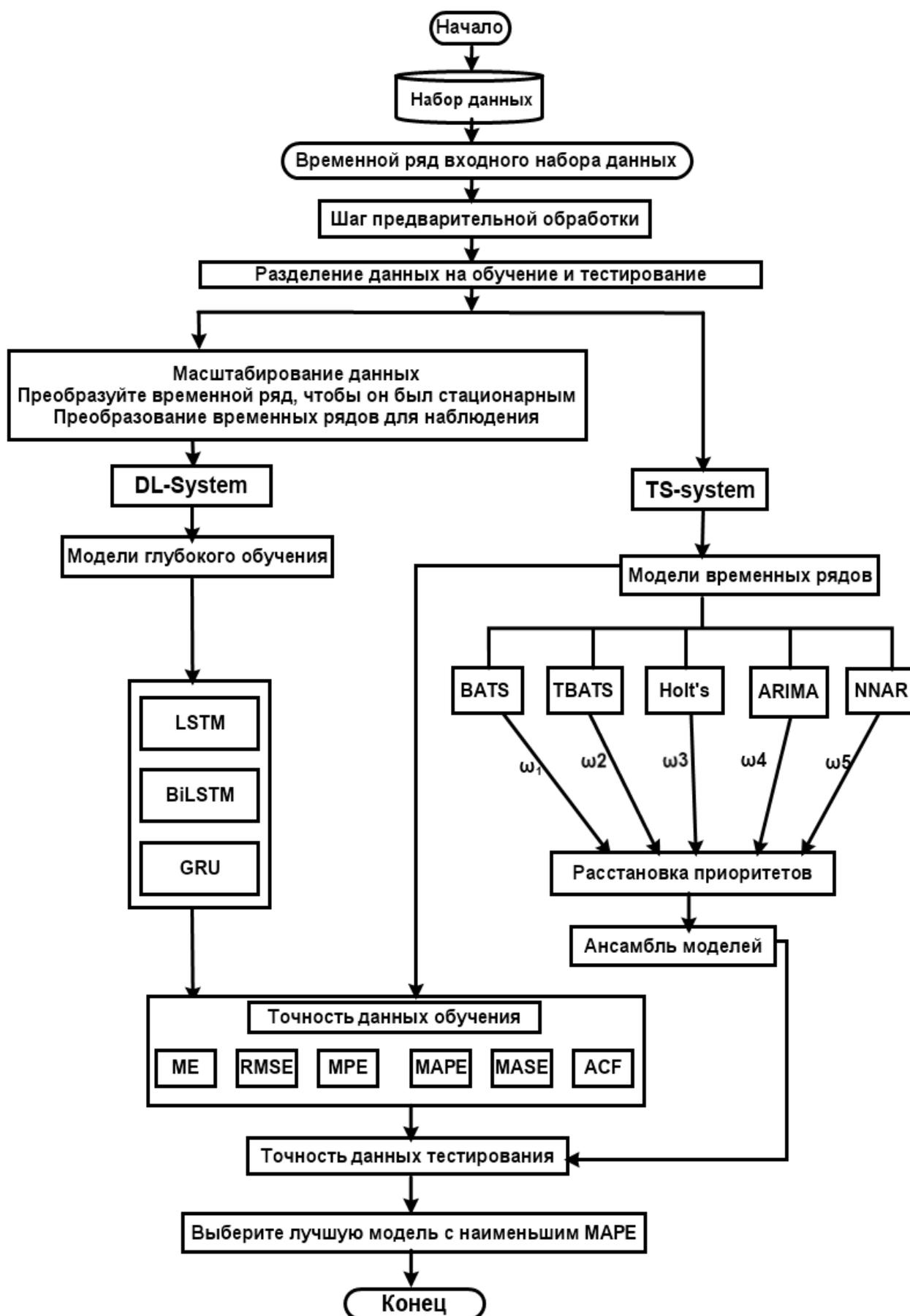


Рисунок 3.2. – Структурная схема системы прогнозирования updated Epidemic.TA With deeplearning models

## 3.2 Система прогнозирования с использованием методов глубокого обучения

Рассмотрим системы глубокого обучения DL-System для моделирования и прогнозирования данных одномерных временных рядов. Лучшая модель выбирается после сравнения средней абсолютной ошибки (MAE), RMSE и R-квадрат ( $R^2$ ).

При моделировании возник ряд проблем с длинными последовательностями в базе данных [72].

1. обучение требует много времени и больших объемов памяти;
2. получение плохо обученной модели из-за обратного распространения длинных последовательностей.

Таким образом, важно подготовить и предварительно обработать данные перед их импортом в нейронные сети. Для этого на этапе подготовки данных могут использоваться методы нормализации и стандартизации данных. В рассматриваемом случае использована стандартизация данных, которая реализована как метод масштабирования. Этот метод позволяет установить среднее значение и стандартное отклонение равными 0 и 1 соответственно.

Набор данных, как и в системе Epidemic.TA разбивается на три набора: для обучения, проверки и тестирования. Набор данных дополнительно подготавливается следующим образом:

- Первые 95% значений из набора данных используются для обучения и валидации: 80% от этих данных выделяются для обучения, а оставшиеся 20% – для валидации. Эти значения являются стандартными, но пользователь может изменить их в зависимости от своих потребностей.

- последние 5% значений из набора данных используются для тестирования.

Учебный набор используется для обучения и улучшения моделей, поэтому 20% обучающих данных используются для проверки, диагностики переобучения и недостаточного обучения, а тестовый набор используется для оценки производительности модели. Полный текст программы и набор данных, на котором проводился эксперимент, представлены в [86].

После использования обработанных данных обучающего набора для обучения LSTM, BiLSTM, и GRU соответственно, модель, завершенная обучением, используется для прогнозирования данных тестового набора, а реальное значение сравнивается с прогнозным. Лучшая модель должна иметь наименьшие значения RMSE, MAE и MSE и наибольшее значение  $R^2$ .

В рассматриваемой системе прогнозирования архитектура LSTM реализована с помощью сигмовидной функции для ворот забывания и входных ворот и с функцией  $\tanh$  для вектора-кандидата, который обновляет вектор состояния ячейки. Эти функции активации LSTM рассчитываются в следующих случаях.

### 1. Для входных ворот

$$I_t = \text{Sigmoid}(\omega_i[h_{t-1}, x_t] + b_i), \quad (3.1)$$

где Sigmoid сигмоидная функция,  $[h_{t-1}, x_t]$  – вектор исходных данных,  $\omega_i$  – весовая функция во входных вратах,  $b_i$  – вектор смещения входного затвора.

Учитывая вес, скрытое состояние и текущий ввод, сигмоидная функция вычисляет значение функции в интервале  $[0; 1]$ .

## 2. Для выходных ворот

$$O_t = \text{Sigmoid}(\omega_\Omega[h_{t-1}, x_t] + b_\Omega), \quad (3.2)$$

где  $\text{Sigmoid}$  сигмоидная функция,  $[h_{t-1}, x_t]$  – вектор исходных данных,  $\omega_o$  – весовая функция ввыходных вратах,  $b_o$  – вектор смещения выходного затвора.

Функция  $O_t$  принимает значение из интервала  $[0; 1]$  и умножает это значение на функцию активации  $\tanh$ , чтобы определить, какие данные следует удалить из долговременной памяти  $C_t$  в момент  $t + 1$  для дальнейшего использования.

## 3. Для забывающих ворот

$$F_t = \text{Sigmoid}(\omega_f[h_{t-1}, x_t] + b_f), \quad (3.3)$$

где  $\text{Sigmoid}$  сигмоидная функция,  $[h_{t-1}, x_t]$  – вектор исходных данных,  $\omega_f$  – вес вентиля забывания,  $b_f$  – вектор затвора забывания.

На основе весов и текущих входных данных сигмоидная функция вычисляет значение  $F_t$  в диапазоне от 0 до 1, определяя, какие данные будут сохранены, а какие забыты. Забвение данных означает, что они были навсегда удалены из долговременной памяти.

## 4. Для вектора-кандидата

$$\tilde{C}_t = \tanh(\omega_\Omega[h_{t-1}, x_t] + b_\Omega) \quad (3.4)$$

## 5. Для состояния ячейки

$$C_t = F_t \cdot C_{t-1} + I_t \cdot \tilde{C}_t \quad (3.5)$$

## 6. Для скрытого состояния

$$h_t = O_t \cdot \tanh(C_t). \quad (3.6)$$

На рис. 3.3 приведен механизм предлагаемой системы прогнозирования DL-System.

### 3.3 Объединение моделей временных рядов с помощью оптимизатора весов для моделей временных рядов

На рисунке 3.3 приводится идея работы оптимизатора, где после получения прогнозных значений от моделей временных рядов и нейронных сетей они объединяются и им присваиваются веса.

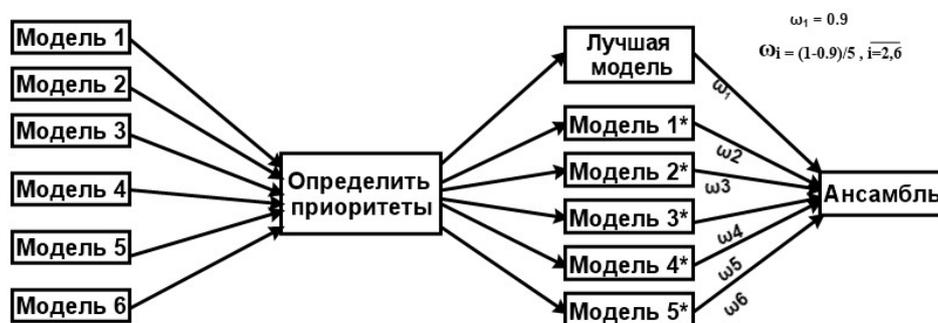


Рисунок 3.3. – Объединение моделей временных рядов с помощью оптимизатора весов для моделей временных рядов

Вычислительный эксперимент показал, что задание веса, равного 0.9, для лучшей модели временного ряда и нейронных сетей и равномерное распределение (0.0–0.9) по остальным моделям позволяет получить точные результаты с меньшими значениями ошибок. Исходный код для двух рассмотренных систем прогнозирования с использованием этого

алгоритма опубликован на GitHub [87], имеются свидетельства о регистрации программ для ЭВМ [13] и опубликовано в [27].

### 3.4 Программное обеспечение для одномерных временных рядов

Данное программное обеспечение разработано для решения задач анализа и прогнозирования одномерных временных рядов, включая применение различных моделей: от классических статистических методов до современных подходов машинного обучения и обобщенного метода наименьшего отклонения (GLDM). На рисунке 3.4 представлена схема, описывающая процесс обработки данных, выбор моделей и оценку точности прогнозов [2, 26].

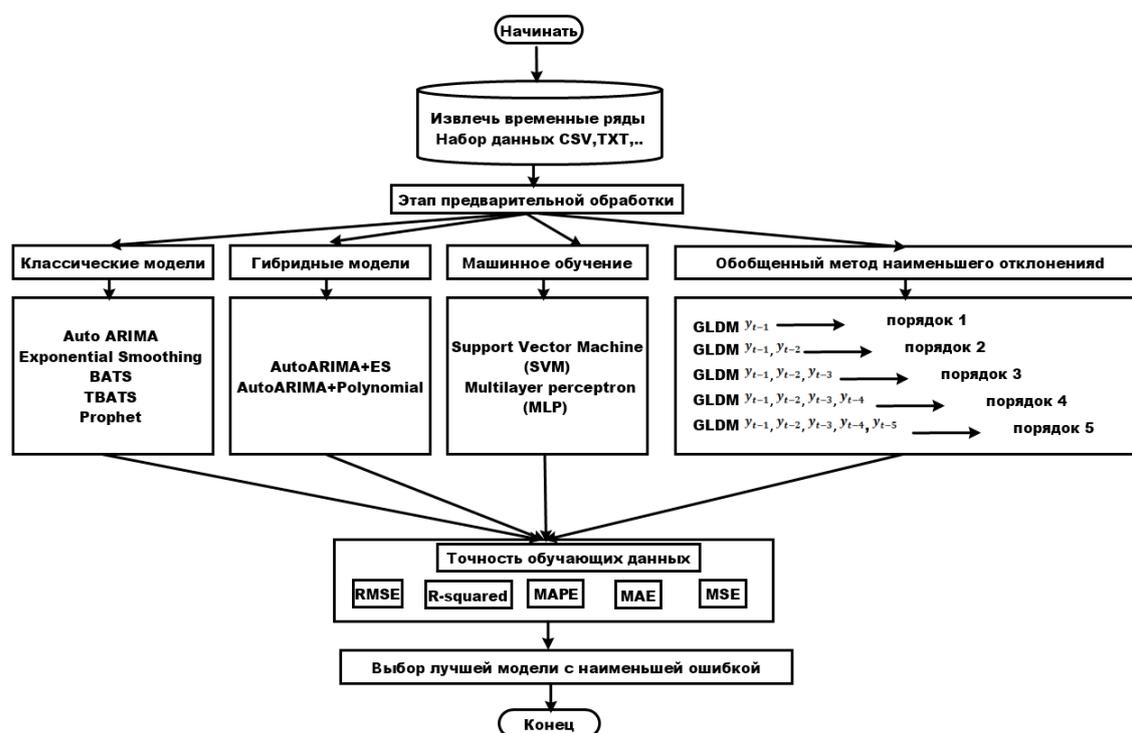


Рисунок 3.4. – Схема программного обеспечения для одномерных временных рядов.

**Шаг 1: Извлечение данных.** Процесс начинается с загрузки вре-

менных рядов из различных источников, таких как файлы CSV, TXT и другие текстовые форматы. Как показано в верхней части схемы на рисунке 3.4, каждый временной ряд представляет собой последовательность наблюдений, где каждая точка содержит информацию, относящуюся к конкретному временному интервалу (например, день, месяц или год).

**Шаг 2: Предварительная обработка данных** (см. «Этап предварительной обработки» на рис. 3.4) включает очистку и подготовку данных для последующего анализа. В зависимости от типа модели этапы обработки данных могут различаться.

Для классических, гибридных моделей и моделей машинного обучения основными шагами являются:

- Обработка пропусков данных (например, интерполяция или удаление неполных записей);
- Масштабирование данных (например, нормализация или стандартизация временных рядов);
- Выделение сезонных и трендовых компонентов для лучшего понимания структуры данных.

Для метода GLDM предварительная обработка упрощена и не требует таких шагов, как:

- Масштабирование данных;
- Выделение сезонных и трендовых компонентов.

**Шаг 3: Применение моделей прогнозирования** использует различные модели (см. «Выбор моделей» на рис. 3.4). Программное обеспечение поддерживает несколько категорий моделей: Auto ARIMA, экспоненциальное сглаживание, BATS и TBATS, Prophet, гибридные модели AutoARIMA + Экспоненциальное сглаживание и AutoARIMA + Полиномиальная регрессия, метод опорных векторов (SVM), многослойный

перцептрон (MLP), обобщенный метод наименьшего отклонения, представленный в диссертации, представленный моделями 1–3 порядков.

**Шаг 4: Оценка точности модели** После построения моделей необходимо оценить их точность с использованием следующих метрик: RMSE (Корень средней квадратичной ошибки); R-squared (Коэффициент детерминации); MAPE (Средняя абсолютная процентная ошибка); MAE (Средняя абсолютная ошибка); MSE (Средняя квадратичная ошибка).

**Шаг 5: Выбор наилучшей модели** Программное обеспечение автоматически выбирает модель с наименьшей ошибкой прогнозирования на основе вышеуказанных метрик (см. «Выбор модели» на рис. 3.4).

**Шаг 6: Завершение процесса** После выбора наилучшей модели система готова к использованию выбранной модели для прогнозирования будущих значений временного ряда на основе исторических данных.

Данное программное обеспечение предоставляет мощные и гибкие инструменты для анализа временных рядов, от этапа обработки данных до выбора модели и получения прогнозов.

### **3.5 Возможности работы алгоритма определения параметров квазилинейного рекуррентного уравнения для больших данных**

Поскольку при работе с длинными временными рядами требуются матрицы высокого порядка, для чего необходимы большие затраты памяти и времени, в работе исследована реализация [82] алгоритма прогнозирования, использующая два вида параллелизма: функциональный и параллелизм по данным. Для исследования возможности многопо-

точного программирования разработанного алгоритма прогнозирования проведены вычислительные эксперименты на разных вычислителях.

### 3.5.1 Возможности функционального параллелизма при определении параметров квазилинейного уравнения

В разработанной программе используется распараллеливание средствами OpenMP, что предполагает модель параллельного программирования SPMD. В этом случае для всех параллельных потоков используется один и тот же код, а в программу вставляются специальные директивы. Программа имеет последовательные области, когда выполняется один процесс (поток), а при входе в параллельную область порождается определенное количество процессов, между которыми в дальнейшем распределяются части кода. Максимально возможное количество потоков в рассматриваемой программе определяется с помощью функции `omp_get_max_threads()` в зависимости от ресурсов компьютера, на котором запущена программа. При проведении вычислительного эксперимента программа запускается с количеством потоков от 1 до максимально возможного значения для конкретного вычислителя.

На рис. 2.1 приведена последовательность вызова разработанных процедур из функции `void main()`, используемая для запуска алгоритма. Листинги всех функций, вызываемых из `main()`, приведены в [82]. В разработанной программе распараллелено выполнение всех циклов, для которых это целесообразно, один за другим. Циклы, которые не могут быть адекватно распараллелены и остаются последовательными областями: (1) первый цикл функции решения прямой задачи WLDM; (2) базовый цикл `do... while` алгоритма оценки GLDM (вложенные циклы распараллелены); (3) циклы функции прогнозирования, поскольку

ку в ней вычисляется рекуррентная последовательность. Все остальные циклы могут выполняться как параллельные участки.

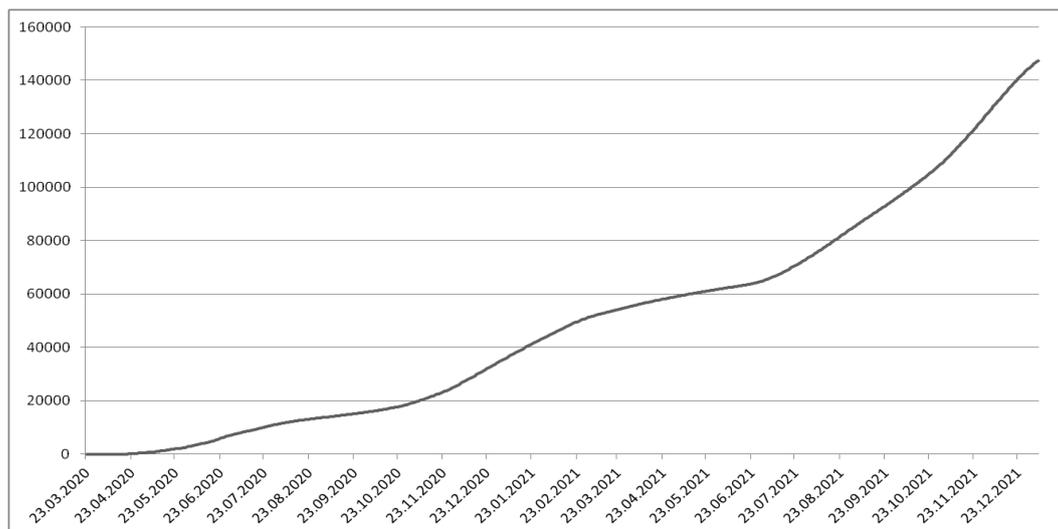
Функция прогнозирования `ForecastingEst()` (см. алгоритм 5) не может быть адекватно распараллелена, поскольку в ней осуществляется расчет рекуррентной последовательности значений. Возможности параллельных вычислений могут быть использованы только во вложенном цикле, где осуществляется вычисление значения в заданной точке временного ряда. Тем не менее, такое использование не представляется разумным, поскольку число коэффициентов модели не велико. Поэтому единственный цикл, в котором использованы директивы `OpenMP`, – это цикл, отвечающий за поиск надежного горизонта прогнозирования, который заключается в поиске минимального значения в массиве. В результате работы этой функции формируется массив `YU[] []` каждый элемент которого равен значениям решеточных функций  $\overline{y[t]}_\tau$  семейства разностных уравнений  $m$ -го порядка (1.33). Используя этот массив, можно получить значения ошибок  $MBE$  (равна сумме ошибок предсказания  $Y[i]$  по значениям  $Y[i-T-1]$  и  $Y[i-T]$ ) и  $MAE$  (равна сумме абсолютных ошибок предсказания  $Y[i]$  по значениям  $Y[i-T-1]$  и  $Y[i-T]$ ):

$$MBE = \sum_{t=3}^{T_R} \left( \overline{y[t]} - \overline{y[t]}_\tau \right), \quad MAE = \sum_{t=3}^{T_R} \left| \overline{y[t]} - \overline{y[t]}_\tau \right|,$$

где  $T_R$  – надежный горизонт прогнозирования, т.е.

$$T_R = \min \left\{ \tau : z > \left| \overline{y[t]} - \overline{y[t]}_\tau \right| \right\}.$$

В качестве примера рассмотрим временной ряд, содержащий кумулятивные данные по случаям заражения вирусом Covid-19 в Челябинской области. В эксперименте рассматриваются векторы разной длины, са-



**Рисунок 3.5.** – Прирост числа кумулятивных случаев Covid-19 по Челябинской области (данные с 23.03.2020 по 06.01.2022)

мый длинный из которых имеет 655 значений. График процесса показан на рис. 3.5. Полученные средние ошибки прогноза разработанной модели составляют  $MBE = 1,615210 \cdot 10^{-1}$  и  $MAE = 9,852680 \cdot 10^{-1}$  при доверительном горизонте реализации 645 дней. Этот результат лучше результатов, полученных с помощью классической статистической модели и модели нейронных сетей, поскольку модель, полученная с помощью GLDM-оценки, позволяет получить долгосрочный прогноз (тем не менее, ARIMA, линейная модель Хольта, NNAR, LSTM позволяют получать хорошие прогнозы только на 30–60 дней, т.к. это было показано в вычислительных экспериментах, описанных в [34]).

Исследуем время выполнения и ускорение при распараллеливании для векторов длиной 150, 300, 500 и 655 элементов. Эксперименты проводились на двух персональных компьютерах. Для компьютера Intel Core i7-4770, 3.4GHz, RAM 16Gb программа определила максимальное количество потоков равное 8, а для компьютера Intel Core i5-2430M, 2.4 GHz, RAM 4 Gb — 4 потока. Для нашего эксперимента зафиксируем количество потоков, равное 8 для обоих компьютеров. Время выполнения

Таблица 3.1. – Время вычислений для разного количества потоков и длин векторов

$N$	Количество нитей							
	1	2	3	4	5	6	7	8
Intel Core i7-4770, 8 Gb ОЗУ, 8 ядер								
150	0.0970	0.0690	0.0490	0.0420	0.0320	0.0300	0.0420	0.0300
300	0.236	0.164	0.131	0.102	0.114	0.099	0.091	0.107
500	0.624	0.411	0.353	0.347	0.317	0.249	0.232	0.229
655	1.05	0.76	0.63	0.57	0.45	0.41	0.38	0.38
Intel Core i5-2430M, 4 Gb ОЗУ, 4 ядра								
150	0.642	0.461	0.449	0.421	0.424	0.421	0.419	0.417
300	2.46	1.77	1.73	1.64	1.64	1.63	1.63	1.62
500	6.67	4.85	4.72	4.49	4.51	4.49	4.46	4.50
655	11.47	8.33	8.15	7.71	8.30	7.76	7.68	7.62

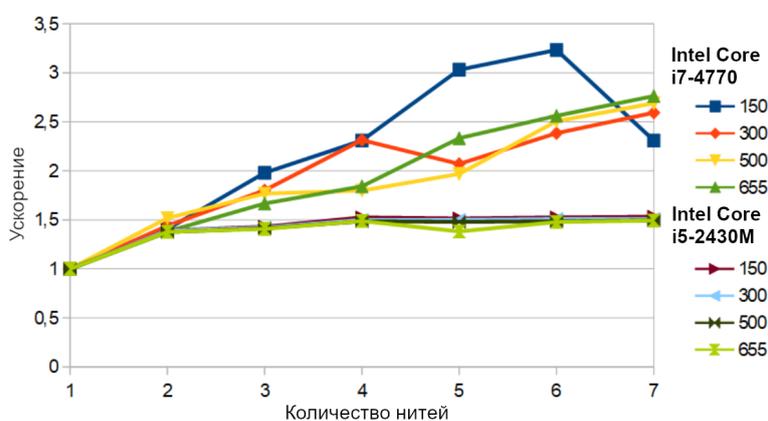
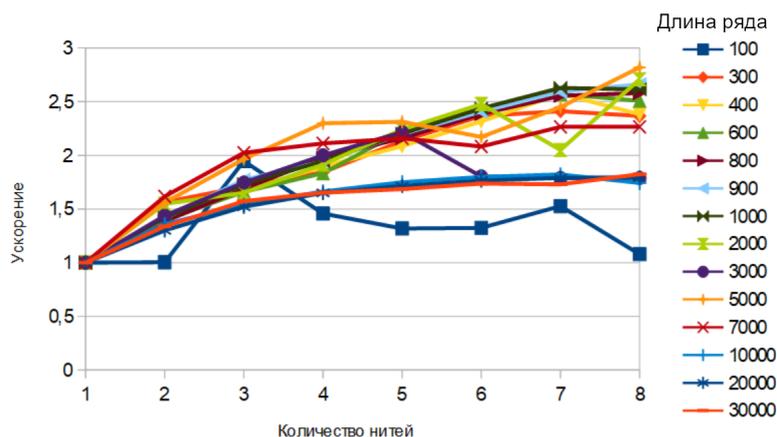


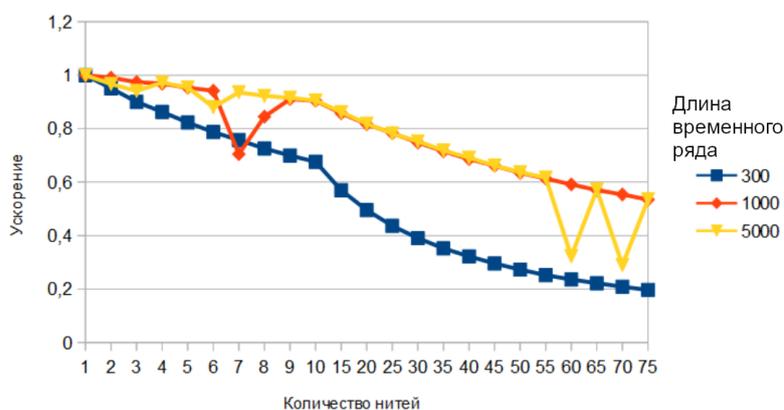
Рисунок 3.6. – График ускорения работы параллельной версии программы для временных рядов длиной 150, 300, 500, 655 точек на исследуемых вычислительных ресурсах

разработанной программы для разного количества  $N$  потоков показано в таблице 3.1. На рис. 3.6 приводится полученное ускорение при вычислении рассматриваемых временных рядов для прогнозирования заболеваемости Covid-19 на двух исследуемых персональных компьютерах.

При рассмотрении более длинных временных рядов, например, рядов лаборатории OikoLab (<https://oikolab.com>), которые содержат почасовые климатические данные рядом с университетом Монаш, Клейтон, Виктория, Австралия, с 01 января 2010 г. по 31 мая 2021 г., получим график, приведенный на рис. 3.7.



**Рисунок 3.7.** – График ускорения работы параллельной версии программы для временных рядов длиной до 30 000 значений на компьютере с процессором Intel Core i7-4770



**Рисунок 3.8.** – Графики ускорения вычислений при использовании вычислительного комплекса «Нейрокомпьютер ЮУрГУ» для анализа временных рядов длиной 300, 1000 и 5000 элементов

Эксперименты показывают, что в лучшем случае имеем ускорение в 3.2 раза на персональном компьютере, позволяющем получить 8 нитей. В среднем ускорение составляет величину 2.7 раз. Проведение экспериментов на более мощных вычислителях не оправдано, поскольку для временных рядов даже размерности порядка  $10^3 - 10^5$  элементов возникают значительные накладные расходы (см. рис. 3.8).

Таким образом, для реализации рассматриваемого подхода функциональный параллелизм не является эффективным.

### 3.5.2 Возможности параллелизма по данным при определении параметров квазилинейных рекуррентных соотношений

Еще одним из возможных применений алгоритма определения параметров квазилинейных рекуррентных соотношений является, например, анализ урожайности сельскохозяйственных угодий на основании снимков, полученных с помощью аэрофотосъемки. В данном случае будем за один посевной сезон иметь относительно короткие временные ряды (порядка 50–60 точек), тем не менее, количество таких рядов будет равно числу участков, обладающих определенными свойствами, описывающих отдельно взятое поле. Тогда параллелизм можно использовать не для самого алгоритма получения коэффициентов модели и построения прогнозов, а для разделения по потокам различных временных рядов. Такое распараллеливание, очевидно, будет более эффективным при использовании не только персональных компьютеров, но и более мощных вычислительных ресурсов. Данный вопрос является темой отдельного исследования.

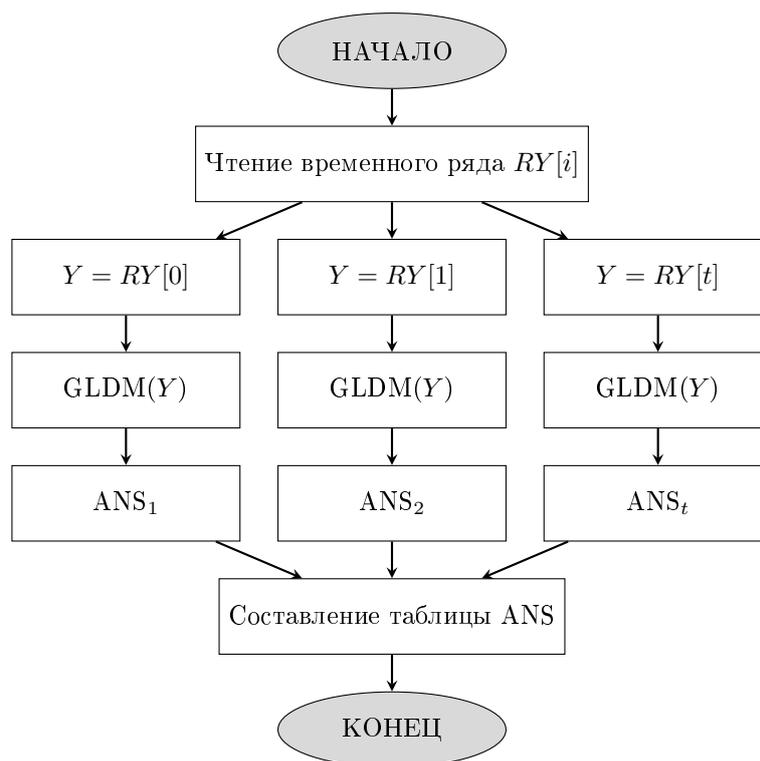
Различные подходы к распараллеливанию различных методов наименьших квадратов с использованием многоядерных архитектур обсуждались в [80]. Авторы этой статьи представили эффективную процедуру решения подзадач наименьших квадратов в алгоритме активного множества. Авторы [110] обсуждают, как использовать специальные функции моделей ARIMA и многослойных перцептронов (MLP) для одновременного моделирования линейных и нелинейных закономерностей в данных и добиться наилучшей производительности при разумных вычислительных затратах. В большинстве работ, посвященных распараллеливанию, например, [58] обсуждается задача с длинными временными рядами. Эта

же задача была рассмотрена и в разделе 3.5.1 (см. работы [88]), где были исследовали процесс распараллеливания для GLDM-алгоритма, работающего с одним длинным временным рядом (в качестве примера рассмотрены временные ряды распространения эпидемий и данные о погоде). Напомним, что эта задача плохо распараллеливается при использовании OpenMP.

В данном разделе рассмотрим множество временных рядов, то есть случай, когда исходные данные представляют собой список векторов длиной менее 100 элементов каждый, количество векторов является произвольным, зависит от размера изучаемых областей и может варьироваться от 1 до десятков тысяч. В этом случае технология распараллеливания отличается, поскольку используется параллелизм по данным. В этом случае распараллеливаются расчеты для разных рядов, поэтому нет необходимости распараллеливания самого GLDM-алгоритма для коротких временных рядов. Для этой цели могут использоваться обе технологии, OpenMP или MPI. Ниже рассмотрим вычислительные эксперименты для OpenMP-программы.

Схема организации параллельных вычислений приведена на рисунке 3.9.

В общем случае количество векторов  $n$  значительно больше количества потоков  $t$  (или параллельных процессов). Следующий для обработки вектор передается первому освобожденному процессу. У рассматриваемом случае имеется один уровень распараллеливания с  $t$  процессами. Для проведения расчетов будем использовать асинхронный вариант расчетной схемы. Существенным недостатком синхронного варианта схемы алгоритма является возможность простоя всех узлов процесса, кроме корневого. Задержка может возникнуть, если какой-то из процессов за-



**Рисунок 3.9.** – Блок-схема параллельного алгоритма определения параметров моделей для  $n$  временных рядов

кончил решение своих подзадач и отправил данные родителю раньше остальных, поскольку родитель создаст новые подзадачи для этих процессов только после получения решений от всех оставшихся процессов.

### Алгоритм для корневого процесса

**Шаг 1.** Получить массив векторов для набора временных рядов  $RY$ .

**Шаг 2.** Отправить один вектор  $RY[i]$  для обработки свободным подчиненным процессом.

**Шаг 3.** Дождаться решения каждой из подзадач, сгенерированных подчиненными процессами, получить данные от каждой из них (массив  $ANS$ , состоящий из коэффициентов модели  $a_i$ , значений функции потерь  $L$ , ошибок  $MAE$  и  $MBE$ ). Принять решение подзадачи  $ANS$  от любого процесса, отправившего данные.

**Шаг 4.** Проверить, все ли подзадачи решены. Если да, то завершить выполнение алгоритма. В противном случае перейти к шагу 2.

## Алгоритм для подчиненного процесса

**Шаг 0.** Взять от родительского процесса точку с вектором временного ряда  $Y$ .

**Шаг 1.** Выполнить алгоритм GLDM-оценки для временного ряда  $Y$

1.1. Сгенерировать базисный набор функций  $g_{(*)}$ .

1.2. Сформировать проекционную матрицу SST.

1.3. Выполнить для SST преобразование Жордана-Гаусса.

1.4. Запустить процедуру оценки с использованием обобщенного метода наименьших отклонений. Сохранить коэффициенты модели  $a_i$  и значение функции потерь  $L$  в структуре ANS.

1.5. Вычислить средние ошибки прогнозирования MAE и MBA и сохранить их значения в структуре ANS.

**Шаг 2.** Передать структуру ANS, содержащую результаты вычислений, родительскому процессу.

Рассмотренный подход к распараллеливанию при использовании OpenMP потребляет много памяти из-за наличия большого числа локальных переменных, хранящих векторы и квадратные матрицы, размерность которых сравнима с длиной исследуемого временного ряда. Эти переменные создаются в общей памяти для каждого потока параллельно, поэтому ограничение используемой приложением памяти расходуется очень быстро. Таким образом, запуск разработанного алгоритма на локальном компьютере (Intel(R) Core(TM) i5-1135G7 11-го поколения, 2.40ГГц, 2.42ГГц, 16Гб ОЗУ) с предоставлением компилятором Visual C++ 2Гб памяти, доступной для разрабатываемых решений, позволил оценить ускорение для временных рядов из 100, 300, 500 и 1000 элементов. Количество временных рядов, пригодных для расчета, составило 1000, 300, 200 и 80 соответственно. Для запуска разработанного GLDM-

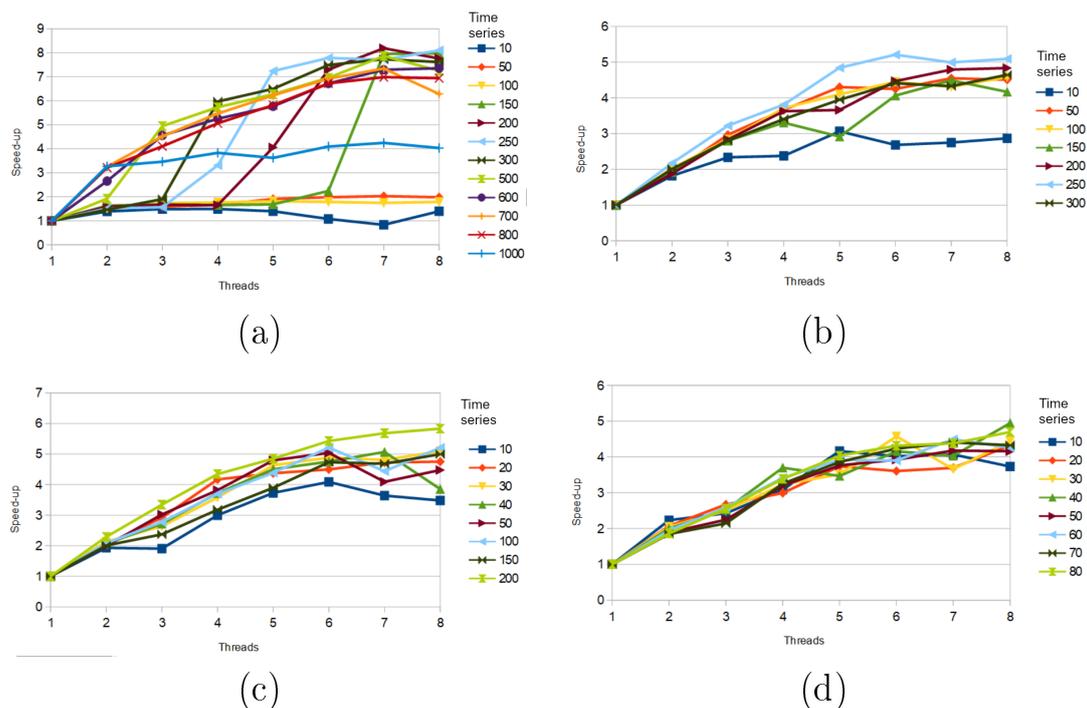


Рисунок 3.10. – Графики ускорения для временных рядов: (а) длины 100; (б) длины 300; (с) длины 500; (д) длины 1000.

алгоритма необходимо выделить память для дополнительных векторов и квадратных матриц для каждого потока отдельно. Размер этих матриц равен квадрату размера исследуемого вектора.

Исследуем ускорение с помощью персонального компьютера для исследуемых наборов временных рядов. Результаты эксперимента приведены на рис. 3.10. Посчитаем время, необходимое нам только для запуска алгоритма 4 для набора векторов данных без учета времени на выполнение функций ввода/вывода данных. Величина ускорения в этом случае получается по следующей формуле:

$$S = \frac{T_1}{T_n},$$

, где  $S$  – ускорение;  $T_1$  – время, необходимое для вычислений с использованием одного потока (последовательная программа);  $T_n$  – время, необходимое для вычислений с использованием  $n$  нитей,  $n = \overline{2, 8}$ .

Легко видеть, что с ростом длины временного ряда значения ускорения становятся практически постоянными для любого числа векторов временного ряда исходных данных. С другой стороны, для коротких временных рядов (длина 100 для рассматриваемого случая) наилучшие результаты получаются для 500–800 векторов временного ряда, а затем при увеличении числа векторов ускорение значительно падает (поскольку память используется на его пределе для этого случая). Эти колебания можно объяснить размером одной задачи. Он очень короткий, на решение уходит очень мало времени, но решение требует временных накладных затрат, что также учитывается при расчете производительности. Это предположение может быть подтверждено рисунками 3.10(b) и (c) с более длинными временными рядами. Отметим, что колебания значений ускорения становятся меньше с ростом длины временного ряда.

Таким образом, проведенный эксперимент показывает, что подход с использованием технологии OpenMP подходит для наборов коротких временных рядов. Для более эффективного использования памяти приложения следует исследовать возможности подхода с использованием MPI. Задача эффективного использования кэш-памяти является открытым вопросом и предметом дальнейших исследований. Поскольку приложение, работающее на персональном компьютере, справляется с 50–150 временными рядами за один запуск, то необходимо разработать дополнительные подходы к использованию памяти для анализа большого числа исследуемых временных рядов и оптимизации вычислительных ресурсов.

### 3.6 Выводы и результаты по главе 3

Разработанная система прогнозирования является расширяемой, она допускает подключение дополнительных модулей прогнозирования и включение их в анализ точности работы моделей. Система позволяет анализировать временные ряды для различных прикладных задач.

Алгоритм определения параметров квазилинейного рекуррентного уравнения реализован средствами языков C++ и Python. Последовательная версия алгоритма с использованием персонального компьютера работает для временных рядов порядка  $10^3$  элементов и требует несколько минут времени. При попытке реализации функционального параллелизма выяснилось, что данный вид параллелизма для данного алгоритма не является эффективным. При задействовании более мощных вычислителей, ускорение отсутствует. С другой стороны, возможен достаточно эффективный параллелизм по данным, что позволяет работать с большим количеством временных рядов даже на персональном компьютере.

Поскольку для запуска разработанного алгоритма необходимо использовать матрицы высокого порядка, полезно распараллелить алгоритм с целью ускорения процесса получения результата. Проведенные вычислительные эксперименты показывают, что при простейших способах распараллеливания рассмотренных алгоритмов имеем ускорение алгоритма в три раза на современных персональных компьютерах при использовании половины из возможных потоков. Тестирование алгоритма на более мощных вычислителях показало не только отсутствие эффекта от распараллеливания для временных рядов длиной до 100 000 значений, но и значительное замедление вычислений за счет накладных расходов на организацию параллельных областей в каждой из функций.

Таким образом, для реализации алгоритма определения параметров одного квазилинейного рекуррентного соотношения нецелесообразно использовать возможности параллельных вычислений.

Исследования возможностей использования параллелизма для алгоритма определения коэффициентов модели с помощью обобщенного метода наименьших модулей для достаточно большого числа коротких временных рядов показали, что можно разработать параллельное персональное приложение для ноутбуков и/или мобильных телефонов, максимально используя ресурсы этих устройств и позволяя получить результат в 5–6 раз быстрее, поскольку разница во времени использования персонального компьютера для расчета больших наборов данных будет иметь важное значение. Учитывая, что новые данные для рассматриваемого в качестве примера динамического процесса получаются один раз в 24–72 часа, то время является пределом для расчетов, и этот предел достигается для временных рядов порядка  $10^7$  с использованием одного потока, и для  $(5 \cdot 10^7)$  объектов с использованием 7–8 потоков персонального компьютера. Этого количества данных достаточно для анализа с помощью персональных компьютеров. Что касается корпораций, использующих больше объектов, то целесообразно использовать ресурсы суперкомпьютеров, например, [53].

Результаты, приведенные в данной главе, опубликованы в работах [4, 14, 26]. Получено свидетельство Роспатента о государственной регистрации программы для ЭВМ [2, 3, 11–13].

## ГЛАВА 4. ВЫЧИСЛИТЕЛЬНЫЕ ЭКСПЕРИМЕНТЫ

Рассмотрим вычислительные эксперименты для различных наборов данных из различных областей. В таблице 4.1 приводится краткий перечень используемых наборов данных и количество измерений в каждом из них  $n$ . Отметим, что наборы 2 и 3 имеют достаточно большой объем для проведения достоверного статистического анализа и проверки модели.

Таблица 4.1. – Перечень используемых наборов данных и их длина  $n$

№	Набор данных	Длина ряда $n$	Публикация
1	NDVI	15	[33]
2	Температура	9 999	[33]
3	Скорость ветра для SCADA	50 530	[33]
4	Случаи смерти COVID-19 в РФ	882	[22]
5	случаи заражения covid 19 в самарской области	1 003	[22]

Приведем краткое описание каждого из рассмотренных наборов данных.

Приведем краткое описание каждого набора данных.

1. **NDVI.** Данные этого набора получены по графикам из работы [16] и отражают значение вегетационного индекса для полей с пшеницей в Ставропольском крае (**ежемесячный набор данных**).
2. **Температура.** Один из рядов лаборатории OikoLab (<https://oikolab.com>), который содержит почасовые климатические данные рядом с университетом Монаш, Клейтон, Виктория, Австралия, с 01 января 2010 г. по 31 мая 2021 г.
3. **Скорость ветра для SCADA.** Набор данных из комплексного набора данных системы SCADA (диспетчерского управления и

сбора данных) для ветряной турбины, размещенного в открытом доступе на Kaggle по адресу [69]. Этот набор данных содержит рабочие данные высокой точности, полученные от ветряной турбины, активно вырабатывающей электроэнергию в Турции. Набор данных содержит записи с десятиминутными интервалами, что помогает установить динамический характер и колебания скорости ветра.

4. **Случаи смерти COVID-19 в РФ.**<sup>1</sup> Ежедневные случаи смерти от Covid-19 в Российской Федерации с 25 марта 2020 года по 23 августа 2022 года. Данные получены из открытых источников [18–20].
5. **Ежедневные случаи заражения COVID-19 в Самарской области**<sup>1</sup> с 16 апреля 2020 года по 13 января 2023 года. Данные получены из открытых источников [18–20].

#### **4.1 Параметры квазилинейного рекуррентного уравнения, описывающего динамику вегетационного индекса**

Как в России, так и в других странах ученые отмечают необходимость использования методов искусственного интеллекта для точного земледелия. Однако программное обеспечение, используемое для него в настоящее время, часто находится на уровне разработки и не выпускается в продажу, методы, заявленные в отдельных статьях, используют стандартные подходы.

---

<sup>1</sup>В работе использовалась длина доступных данных  $n$ , так как это была единственная информация, доступная в открытых источниках. Данные о COVID-19 для регионов России после 13 января 2023 года недоступны, и не все регионы имели полные данные до этой даты, поэтому для анализа были использованы только доступные данные. Также стоит отметить, что данные о случаях смерти от COVID-19 в Российской Федерации доступны только до 23 августа 2022 года.

Для проведения вычислительного эксперимента рассмотрим систему прогнозирования урожайности на основе среднего значения вегетационного индекса за сезон, которую в дальнейшем планируется внедрить в «Геопортальную систему мониторинга сельского хозяйства» [90] с целью предоставления пользователям дополнительных сервисов для мониторинга сельскохозяйственных угодий. Предлагаемая система включает в себя несколько моделей для описания динамического процесса и прогнозирования.

#### **4.1.1 Описание исходных данных и способов их представления**

Актуальной является задача анализа изображений, полученных со спутниковых снимков, с целью оценки ожидаемой урожайности на один гектар. В качестве исходных данных имеется изображение в формате GeoTIFF. Это может быть как спутниковый снимок, так и снимок, сделанный мультиспектральной камерой БПЛА. В обоих случаях GeoTIFF представляет собой многослойное растровое изображение, состоящее из миллионов пикселей. Каждый слой этого изображения соответствует одному из каналов.

Для дальнейшего анализа необходимо сформировать векторы пикселей каждого канала изучаемого объекта. Каждый объект хранится в базе данных системы [90] в виде shp-файла. Для получения информации только для интересующего объекта, можно воспользоваться библиотеками языка Python, позволяющими сохранить в отдельном графическом файле только пиксели, относящиеся к интересующему объекту (данный процесс рассмотрен ниже как часть процесса расчета вегетационного индекса). Для дальнейшей обработки пиксели из сохраненного

графического изображения сохраняются в виде вектора. Каждый сельскохозяйственный объект должен быть исследован отдельно, поскольку он имеет свою динамику развития вегетационных процессов и, возможно, требует индивидуального подхода при выборе агротехники. Причем, каждый объект исследуется попиксельно. Информация для других моментов времени вегетационного периода предоставляется в том же виде.

Используя векторы значений по всем исследуемым диапазонам, можно вычислить значение различных индексов, характеризующих развитие объекта. Одним из наиболее важных и значимых индексов при исследовании качества вегетации является NDVI. NDVI (нормализованный разностный индекс растительности) – это показатель, описывающий плотность растительности на территории на основе спутниковых либо мультиспектральных снимков. Для расчета этого используется формула:

$$NDVI = \frac{NIR - RED}{NIR + RED},$$

где  $NDVI$  – нормированный разностный вегетационный индекс;  $NIR$  – отражение в ближнем инфракрасном спектре (полоса 5 снимка Landsat);  $RED$  – отражение в красной области спектра (4 полоса снимка Landsat).

В результате предварительной обработки данных имеем набор временных рядов, соответствующих динамическому процессу для каждого рассматриваемого сегмента (или объекта). Эти временные ряды (векторы) используются в качестве исходных данных для получения модели рассматриваемого процесса. Полученную модель можно будет использовать в дальнейших вегетационных периодах для прогнозирования урожайности как сегмента, так и поля в целом. Очевидно, модель должна учитывать не только данные предыдущих периодов, но и различные внешние факторы: погоду в определенные периоды, лунный цикл, влаж-

ность, температуру почвы и т. д. Но изучение зависимости динамики вегетационного индекса от этих явлений – предмет отдельного большого исследования и не рассматривается в данном исследовании.

Для прогнозирования динамики вегетационного индекса можно использовать любую математическую модель, анализирующую временные ряды. Выбор модели может зависеть от различных факторов, начиная от простоты использования, скорости вычислений, наличия программных модулей для проведения расчетов, заканчивая высокой точностью модели и ее универсальностью. Ниже приведем исследование динамики вегетационного индекса при помощи разработанной в диссертации модели и получим числовые коэффициенты, описывающие процесс.

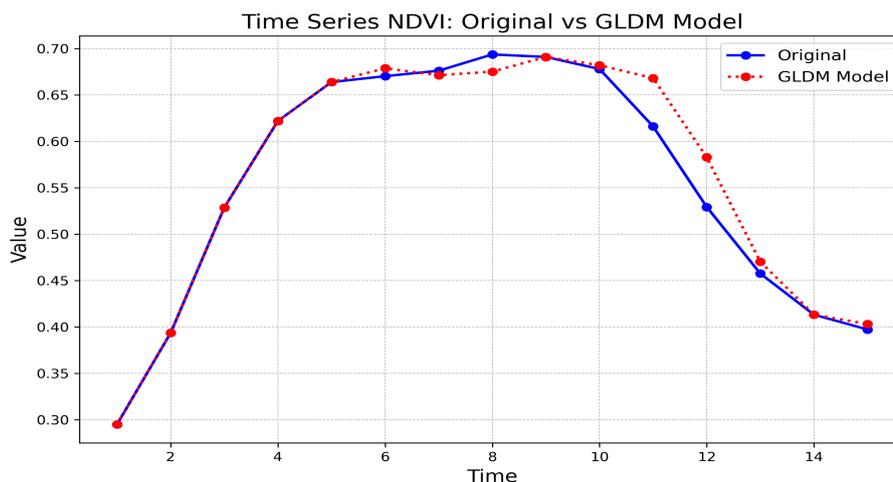
#### 4.1.2 Моделирование динамики вегетационного индекса и анализ прогнозирования урожайности

Пример реализации этой модели показан на рис. 4.1, где приведены результаты моделирования динамики вегетационного индекса для озимой пшеницы в Ставропольском крае в 2014 г. Данные получены по графикам, приведенным в работе [16]. Этот набор данных содержит средние данные для одного поля, но если поле разбито на несколько сегментов, то аналогичным образом можно взять средние значения для одного сегмента.

В результате использования алгоритма для имеющихся данных была получена модель [33]

$$y_t = (3.46935y_{t-1} - 2.18641y_{t-2}) + (-5.59237y_{t-1}^2 - 2.5635y_{t-1}y_{t-2} + 7.72991y_{t-2}^2). \quad (4.1)$$

Модель, представленная в уравнении (4.1), описывает зависимость



**Рисунок 4.1.** – Моделирование динамики NDVI для озимой пшеницы в Ставропольском крае

переменной  $y_t$  от предыдущих значений  $y_{t-1}$  и  $y_{t-2}$ .

В данной модели коэффициенты  $a_1 = 3.46935$  и  $a_2 = -2.18641$  представляют влияние предыдущих значений  $y_{t-1}$  и  $y_{t-2}$  на текущее значение  $y_t$ . Эти члены описывают линейную зависимость в модели.

Коэффициенты  $a_3 = -5.59237$ ,  $a_4 = -2.5635$  и  $a_5 = 7.72991$  относятся к квадратичным членам и взаимодействиям между  $y_{t-1}$  и  $y_{t-2}$ . В частности, слагаемое  $-5.59237y_{t-1}^2$  отражает нелинейное влияние  $y_{t-1}$  на  $y_t$ , слагаемое  $-2.5635y_{t-1}y_{t-2}$  описывает взаимодействие между  $y_{t-1}$  и  $y_{t-2}$ , а слагаемое  $7.72991y_{t-2}^2$  характеризует нелинейное влияние  $y_{t-2}$  на  $y_t$ .

Таким образом, данная модель учитывает как линейные, так и нелинейные зависимости, что позволяет более точно прогнозировать динамику NDVI на основе предыдущих значений.

Значение функции потерь в данном случае равно 0.9172588, а полученные значения ошибок  $MAE = 0.01$ .

Заметим, что полученная модель устойчива и может использоваться при описании динамики вегетационного индекса для отдельно взятых

объектов интереса как в текущем вегетационном периоде, так и в последующих. Отметим, что при анализе динамики индекса в последующих периодах модель дает более низкую точность, поскольку на динамику индекса влияет множество немалооажных факторов: температура воздуха, влажность и пр. Примеры вычислительных экспериментов с использованием полученной модели, приведены в [85].

Достоинством рассмотренной квазилинейной модели является то, что, однажды определив ее коэффициенты за сравнительно небольшое время, модель можно использовать для прогнозирования урожайности того же объекта в следующем вегетационном периоде сразу же после получения первых двух значений вегетационного индекса после всходов. Допускается дальнейшая корректировка коэффициентов модели для исследуемого вегетационного периода, на основе новых фактических значений NDVI. Это позволяет оценить урожай в самом начале вегетационного периода и корректировать полученный прогноз.

Для прогнозирования урожайности поля в текущем сезоне можно использовать различные подходы, например, [36] или [44]. Так, подход [44] использовался для анализа вегетационных индексов кукурузы и оливок в 2013–2017 годах в пригородах Мадрида. Для анализа вегетационного индекса данных культур авторы применяют аппроксимацию с двойным сглаживанием, подходящую для временных рядов с линейным трендом. Авторы исследуют зависимость урожайности от среднесуточной температуры, количества осадков и влажности. К сожалению, как и многие другие подходы, использующие эконометрические модели, данный подход позволяет получить качественные модели только для исследуемых объектов, либо объектов, находящихся в географической близости от них. Подход [6] был выработан в Воронежской области для твердых

сортов пшеницы. Авторы используют модель, требующую меньшее количество различных параметров. Результаты построения регрессионной зависимости урожайности  $U$  от значений интегрального показателя  $S$  (представляет собой среднее значение вегетационного периода индекса NDVI на интервале) выглядят следующим образом:

$$U = -37.06 + 129.83S. \quad (4.2)$$

Авторы статьи [6] утверждают, что это уравнение можно использовать для прогнозирования урожайности сельскохозяйственных культур и приводят таблицу значений урожайности поля в зависимости от среднего значения NDVI за сезон. В диссертации [16] для рассматриваемого нами примера приведена своя модель линейной регрессии

$$U = -22.50 + 100.34S. \quad (4.3)$$

На основе данных, полученных с использованием результатов моделирования с использованием квазилинейного рекуррентного уравнения, удалось построить еще одну регрессионную модель:

$$U = 14.0277 + 37.8439S. \quad (4.4)$$

Расчет урожайности с помощью приведенных выше уравнений линейной регрессии приведен в таблице 4.2.

Авторы работы [6] утверждают, что полученная ими формула демонстрирует более тесную связь между интегральным показателем  $S$  (средним NDVI) и урожайностью  $U$ , нежели с характеристиками используемой ими модели. Тем не менее, следует понимать, что, несмотря на

Таблица 4.2. – Сравнение фактической урожайности [16] и значений, полученных с использованием моделей, описанных уравнениями (4.2)–(4.4)

Поле	Средний NDVI	Урожайность [16]	Модель (4.2)	Модель (4.3)	Модель (4.4)
1	0,536	29,41	32,58	31,32	34,33
2	0,538	39,14	32,73	31,44	34,37
3	0,541	33,75	33,18	31,78	34,50
4	0,548	34,76	34,03	32,44	34,75
5	0,560	37,02	35,62	33,67	35,21
6	0,576	35,65	37,67	35,26	35,81
7	0,587	35,47	39,10	36,36	36,23

удовлетворительные, на первый взгляд, результаты для данных другого региона, формула из [6] применима для сельскохозяйственных угодий Воронежской области, и в других регионах с другим климатом приводит к увеличению погрешности прогнозирования урожайности. Аналогичное утверждение справедливо и для разных сортов и культур. Следовательно, для получения более точного прогнозного значения урожайности  $U$  для другого региона РФ или другой сельскохозяйственной культуры, предлагается после сбора данных о фактических значениях вегетационных индексов и урожайности угодий, выработать подход для разработки программного модуля определения коэффициентов регрессионной модели. Такой подход рассмотрен в [6] для отдельно взятого региона на выборке из 105 полей. Следовательно, предметом дальнейших исследований является верификация и корректировка этой модели на данных регионов других климатических зон, поиск гибкого общего подхода, автоматизация получения коэффициентов модели, а также изучение влияния различных внешних факторов, как в подходе [9], и добавление экономических расчетов для оценки и прогнозирования себестоимости единицы сельскохозяйственных культур.

Авторы еще одной работы, посвященной анализу урожайности [17] орошаемых озимых пшеницы и ячменя иракской провинции Дивания,

описывают двухкомпонентную эвристическую модель урожайности на основе спутниковых данных NDVI. Однако эти методы имеют несколько значительных недостатков. Модель содержит не только тренд урожайности, обусловленный сравнительно медленным изменением технологии возделывания культур, но и климатическую компоненту, связанную с колебаниями биологической продуктивности из-за погодных условий вегетации. Используемые регрессионные модели опираются на предположение о линейной зависимости между предикторами и результативными показателями. Рассмотренные авторами модели имеют достаточно высокую прогностическую способность для рассмотренных исходных данных, однако они требуют настройки под каждый конкретный набор данных, что снижает их универсальность и адаптивность.

Рассмотренная в работе модель GLDM второго порядка не имеет таких ограничений, она учитывает как линейные, так и нелинейные зависимости между переменными и позволяет более точно моделировать сложные процессы. Для рассмотренного набора данных уравнение, полученное с помощью модели, приведено выше (4.1). GLDM модель снижает зависимость от интервалов обучения и обладает большей устойчивостью к изменениям в данных, обеспечивая стабильные и точные прогнозы даже при изменении климатических условий и агротехнологий. Данная модель демонстрирует высокую гибкость и адаптивность. Она легко модифицируется для учета специфики различных регионов и культур, что делает ее более универсальной по сравнению с традиционными регрессионными методами. Это уменьшает необходимость в значительной калибровке под каждый новый набор данных, повышая общую эффективность и точность прогнозов.

### 4.1.3 Анализ значимости коэффициентов модели для разных порядков уравнения

В таблице 4.3 представлены результаты вычисления ошибок моделирования для моделей 1–5 порядков для временного ряда 1.

Таблица 4.3. – Таблица ошибок для количественного анализа динамики вегетационного индекса (временной ряд 1) для моделей 1–5 порядков

Порядок модели	RMSE	$R^2$	MAPE	MAE	MSE
1	0.08	0.68	10.80	0.05	0.01
2	0.02	0.96	2.16	0.01	0.00
3	0.88	0.04	82.09	0.50	0.78
4	0	1.00	0	0	0
5	0	1.00	0	0	0

Анализ таблицы показывает, что лучшие результаты получены при помощи моделей 4–5 порядка, что позволяет предположить идеальную подгонку и учесть всю вариабельность данных. Тем не менее, модель второго порядка также позволяет получить хорошие показатели. Что касается моделей 1 и 3 порядка, для данного набора они дают менее качественную подгонку.

Несмотря на идеальные результаты, полученные от моделей четвертого и пятого порядка, мы выбрали модель GLDM второго порядка для оценки NDVI. Этот выбор связан с тщательным балансом между сложностью модели и риском переобучения, особенно учитывая ограниченный набор данных всего из 15 точек. Хотя модели более высокого порядка могут захватывать сложные нелинейные зависимости, они вводят больше коэффициентов, что приводит к избыточной параметризации и повышенному риску переобучения на обучающих данных. Идеальное соответствие моделей четвертого и пятого порядка может плохо обобщаться на новые, невиданные данные, снижая их практическую ценность.

В таблице 4.4 представлены коэффициенты полученных моделей для временного ряда 1 (динамика вегетационного индекса). В каждом столбце перечислены коэффициенты, полученные в результате подгонки модели GLDM к имеющемуся набору данных, для модели порядков 1–5 соответственно.

Таблица 4.4. – Результаты расчета коэффициентов модели для временного ряда 1 (динамика вегетационного индекса)

Коэффициент	Значение для моделей разного порядка				
	1	2	3	4	5
$a_1$	1.7073	<b>3.4694</b>	-9.6495	-21.8416	0.0000
$a_2$	-1.0511	<b>-2.1864</b>	-16.2326	52.4809	-29.0004
$a_3$	–	<b>-5.5924</b>	29.1697	30.7212	64.0513
$a_4$	–	<b>-2.5635</b>	76.3993	-48.3575	-44.3069
$a_5$	–	<b>7.7299</b>	122.9467	-132.0576	10.2588
$a_6$	–	–	-71.5312	-177.0664	1.7283
$a_7$	–	–	-229.9915	4.7422	-23.5562
$a_8$	–	–	98.9790	-2.5712	-90.7331
$a_9$	–	–	0.0000	273.2420	30.3171
$a_{10}$	–	–	–	-66.8228	-6.6379
$a_{11}$	–	–	–	83.1160	-2.0578
$a_{12}$	–	–	–	0.0000	90.0347
$a_{13}$	–	–	–	0.0000	0.0000
$a_{14}$	–	–	–	0.0000	0.0000
$a_{15}$	–	–	–	–	0.0000
$a_{16}$	–	–	–	–	0.0000
$a_{17}$	–	–	–	–	0.0000
$a_{18}$	–	–	–	–	0.0000
$a_{19}$	–	–	–	–	0.0000
$a_{20}$	–	–	–	–	0.0000

Показатели производительности для различных моделей, примененных к данным NDVI, приведены в таблице 4.5.

Худший результат демонстрируют модель машинного обучения MLP и Prophet модель, значительно более высокими прогностическими характеристиками обладает модель SVM. Модели на основе ARIMA и экс-

Таблица 4.5. – Показатели производительности различных моделей для NDVI

Модель	RMSE	MSE	MAE	$R^2$	MAPE
MLP	0.34	0.12	0.25	0.47	51.71%
SVM	0.08	0.01	0.08	0.59	14.74%
Auto-ARIMA	0.07	0.00	0.03	0.74	8.50%
Эксп.сглаживание	0.03	0.00	0.02	0.95	4.17%
BATS	0.10	0.01	0.07	0.36	16.53%
TBATS	0.04	0.01	0.03	0.90	6.99%
Prophet	0.12	0.01	0.10	0.17	20.37%
Гибрид Auto-ARIMA+ES	0.06	0.00	0.04	0.76	9.45%
Гибрид Auto-ARIMA+Polynomial	0.06	0.00	0.03	0.80	8.42%
<b>GLDM (порядок 2)</b>	<b>0.02</b>	<b>0.00</b>	<b>0.01</b>	<b>0.96</b>	<b>2.16%</b>

пониженного сглаживания превосходят по качеству модели на основе машинного обучения. То же можно сказать и про модели BATS и TBATS. Что касается модели GLDM второго порядка, то она превосходит все другие модели: она имеет наименьшие значения RMSE, MSE, MAE, MAPE, наибольшее значение  $R^2$ , что свидетельствует о высокой точности и надежности модели.

В целом, анализ производительности показывает, что разработанные модели на основе квазилинейных уравнений высокого порядка с использованием обобщенного метода наименьшего отклонения (GLDM) превосходят известные модели на коротком наборе данных о вегетационном индексе. Ниже будет показано, что для набора данных о температуре высокую производительность дает модель GMDM пятого порядка, а модель SVM превосходит все другие модели, достигая наилучших показателей по всем оцениваемым критериям.

Результаты проверки значимости коэффициентов были опубликованы в [33].

## 4.2 Вычислительный эксперимент для анализа значимости коэффициентов модели при анализе данных о температуре

В данном разделе рассмотрим определение коэффициентов модели для временного ряда, состоящего из более, чем  $10^5$  значений. В качестве исходных данных возьмем один из рядов лаборатории OikoLab (<https://oikolab.com>), который содержит почасовые климатические данные рядом с университетом Монаш, Клейтон, Виктория, Австралия, с 01 января 2010 г. по 31 мая 2021 г. В таблице 4.6 представлены результаты вычисления ошибок моделирования для моделей 1–5 порядков для временного ряда 2.

Таблица 4.6. – Таблица ошибок для количественного анализа динамики температуры в Австралии (временной ряд 2) для моделей 1–5 порядков

Порядок модели	RMSE	$R^2$	MAPE	MAE	MSE
1	1.10	0.96	4.63	0.65	1.21
2	1.05	0.97	4.43	0.62	1.10
3	2.82	0.77	14.31	2.05	7.98
4	0.95	0.97	4.20	0.59	0.91
5	0.94	0.97	4.12	0.58	0.88

Для данного временного ряда лучшей является модель пятого порядка. Эта модель позволяет получить для данного ряда наименьшую среднеквадратичную ошибку (RMSE), что свидетельствует о минимальном отклонении от наблюдаемых значений. Кроме того, значение  $R^2 = 0,97$ , означает, что модель отражает 97% вариабельности данных, что свидетельствует о ее надежности. Средняя абсолютная процентная погрешность (MAPE) и средняя абсолютная ошибка (MAE) также находятся на самом низком уровне в этом порядке, что свидетельствует о высокой

точности модели. Среднеквадратичная ошибка (MSE) подтверждает это наблюдение, подтверждая что оптимальной из рассмотренных является модель 5 порядка.

Таблица 4.7. – Результаты расчета коэффициентов модели для временного ряда 2 (температура в Австралии)

Коэффициент	Значение для моделей разного порядка				
	1	2	3	4	5
$a_1$	1.0159	1.0498	-0.1658	1.1661	<b>0.0000</b>
$a_2$	-0.0009	-0.0302	0.0395	-0.3931	<b>1.0667</b>
$a_3$	–	0.0229	1.1547	1.6191	<b>-0.4329</b>
$a_4$	–	0.0098	0.0362	-1.2894	<b>1.4878</b>
$a_5$	–	-0.0340	0.0298	-0.0031	<b>-0.7536</b>
$a_6$	–	–	0.0175	0.1141	<b>-0.2734</b>
$a_7$	–	–	-0.0489	-0.1237	<b>0.0154</b>
$a_8$	–	–	-0.0365	-0.0320	<b>0.0804</b>
$a_9$	–	–	0.0000	-0.0594	<b>0.0679</b>
$a_{10}$	–	–	–	0.1424	<b>0.0827</b>
$a_{11}$	–	–	–	-0.0881	<b>0.0083</b>
$a_{12}$	–	–	–	-0.2193	<b>-0.0745</b>
$a_{13}$	–	–	–	0.1068	<b>-0.0029</b>
$a_{14}$	–	–	–	0.1502	<b>0.0609</b>
$a_{15}$	–	–	–	–	<b>-0.0092</b>
$a_{16}$	–	–	–	–	<b>-0.1386</b>
$a_{17}$	–	–	–	–	<b>0.0510</b>
$a_{18}$	–	–	–	–	<b>0.0560</b>
$a_{19}$	–	–	–	–	<b>-0.1933</b>
$a_{20}$	–	–	–	–	<b>0.0442</b>

Заметим, что в таблице 4.7 большинство полученных коэффициентов в моделях высших порядков отличны от нуля, в то время как в таблице 4.4 для короткого временного ряда имеется большое число нулевых коэффициентов, что позволяет сделать выбор в пользу моделей более низкого порядка.

Проанализируем точность прогнозирования с использованием различных моделей для временного ряда 3. Полученные результаты приведены в таблице 4.8. Жирным шрифтом в таблице выделены модели, де-

Таблица 4.8. – Показатели производительности различных моделей для временного ряда 2 (температура в Австралии)

Модель	RMSE	MSE	MAE	$R^2$	MAPE
MLP	4.60	21.17	3.62	0.37	29.17%
<b>SVM</b>	<b>0.40</b>	<b>0.16</b>	<b>0.24</b>	<b>0.99</b>	<b>1.80%</b>
AutoARIMA	0.92	0.85	0.62	0.97	4.54%
Экспоненциальное сглаживание	1.05	1.10	0.70	0.97	4.98%
BATS	0.96	0.91	0.59	0.97	4.25%
<b>TBATS</b>	<b>0.71</b>	<b>0.50</b>	<b>0.45</b>	<b>0.99</b>	<b>3.23%</b>
Пророческая модель	4.03	16.22	3.12	0.52	23.49%
Гибридная AutoARIMA-ES	0.93	0.86	0.62	0.97	4.54%
Гибридная AutoARIMA-полиномиальная	0.92	0.85	0.62	0.97	4.54%
<b>GLDM (5 порядок)</b>	<b>0.94</b>	<b>0.88</b>	<b>0.58</b>	<b>0.97</b>	<b>4.12%</b>

монстрирующие высокую производительность. Обратим внимание, что рассматриваемая модель пятого порядка – в их числе.

Таким образом, анализ производительности модели показывает, что полученные квазилинейные уравнения высокого порядка с использованием обобщенного метода наименьшего отклонения (GLDM) превосходят современные модели. Полученные модели демонстрируют высокую точность и надежность при более низких значениях RMSE, MSE и MAPE и более высоких значениях  $R^2$ . Основными конкурентами рассматриваемой модели можно считать модели SVM и TBATS, способные улавливать сезонные закономерности и сложные тренды. Высокая производительность GLDM в обоих наборах данных демонстрирует его способность учитывать выбросы и нестандартные распределения ошибок, что делает данный метод надежным и эффективным подходом к обработке сложных нелинейных данных.

Модель GLDM (Обобщенного метода наименьшего отклонения) пятого порядка для моделирования температуры используется для прогнозирования будущих значений температуры на основе последних пяти наблюдений. Эта модель захватывает сложные взаимосвязи и взаимодействия в данных, включая элементы, которые учитывают как линейные,

так и нелинейные зависимости до пятого порядка.

$$\begin{aligned}
\hat{y}_t = & (0.0000 y_{t-1}) + (1.0667 y_{t-2}) - (0.4329 y_{t-3}) + (1.4878 y_{t-4}) \\
& - (0.7536 y_{t-5}) - (0.2734 y_{t-1}^2) + (0.0154 y_{t-1} y_{t-2}) \\
& + (0.0804 y_{t-1} y_{t-3}) + (0.0679 y_{t-1} y_{t-4}) + (0.0827 y_{t-1} y_{t-5}) \\
& + (0.0083 y_{t-2}^2) - (0.0745 y_{t-2} y_{t-3}) - (0.0029 y_{t-2} y_{t-4}) \\
& + (0.0609 y_{t-2} y_{t-5}) - (0.0092 y_{t-3}^2) - (0.1386 y_{t-3} y_{t-4}) \\
& + (0.0510 y_{t-3} y_{t-5}) + (0.0560 y_{t-4}^2) - (0.1933 y_{t-4} y_{t-5}) \\
& + (0.0442 y_{t-5}^2)
\end{aligned} \tag{4.5}$$

Линейные члены представляют собой линейное влияние последних пяти значений температуры на текущее прогнозное значение. Положительные коэффициенты указывают на прямую зависимость, в то время как отрицательные коэффициенты свидетельствуют об обратной зависимости.

Квадратичные члены учитывают нелинейные квадратичные эффекты последних пяти значений температуры. Квадратичные члены помогают моделировать кривизну в данных, что обеспечивает более точное соответствие для нелинейных закономерностей.

Перекрестные произведения позволяют моделировать комбинированное влияние пар последних пяти значений температуры. Взаимодействующие члены необходимы для захвата совместного влияния нескольких факторов на текущее прогнозное значение температуры.

Результаты эксперимента опубликованы в [33]

### 4.3 Вычислительный эксперимент для анализа значимости коэффициентов модели при анализе данных о скорости ветра

Постоянный выброс парниковых газов традиционными энергетическими ресурсами, получаемыми из ископаемого топлива, вносит значительный вклад в глобальное потепление и его пагубное воздействие на атмосферу Земли. Следовательно, существует настоятельная необходимость в расширении мощностей электростанций, работающих на возобновляемых источниках энергии, для удовлетворения постоянно растущего мирового спроса на энергию. Примечательно, что береговые ветряные и фотоэлектрические солнечные электростанции, которые являются одними из наиболее технологически продвинутых возобновляемых источников энергии, обладают преимуществом самой низкой стоимости энергии в сравнении с другими энергоустановками. Таким образом, ожидается, что в будущем эти ресурсы будут составлять большую долю в общем объеме возобновляемых источников энергии. Отметим, что производство электроэнергии из этих двух источников часто имеет обратную зависимость, особенно в месячном масштабе [126]. Кроме того, энергия ветра известна своей значительной изменчивостью на почасовой основе, что делает ее непостоянным источником энергии. Кроме того, по мере расширения интеграции ветроэнергетики в систему электросетей многих стран не только растут эксплуатационные расходы, но и снижается надежность системы [111]. Таким образом, точное прогнозирование ветра имеет важное значение для обеспечения надежности энергоснабжения и устранения перебоев в работе, связанных с использованием энергии ветра.

Способность прогнозировать скорость ветра с высокой точностью играет важную роль в оперативном планировании, оптимизации выработки энергии и минимизации разрыва между генерируемой и потребляемой мощностью. Протестируем разработанный алгоритм для прогнозирования скорости ветра. Используем набор данных, предоставляемый системой диспетчерского управления и сбора данных (SCADA) системы от действующих ветряных турбин в Турции. Данные SCADA отражают фактическую производительность турбины с течением времени. Анализируя имеющиеся данные о скорости ветра, оценим способность алгоритма GLDM точно прогнозировать скорость ветра.

*Этот временной ряд содержит порядка  $5 \times 10^4$  значений. Мы обнаружили, что модели третьего–пятого порядков обладают более высокой ошибкой прогноза, поэтому сосредоточили внимание на моделях первого и второго порядков. Для рассмотренного временного ряда наилучшие результаты показала модель второго порядка (см. табл. 4.9), которая незначительно уступает линейной модели. В данном случае  $R^2 = 0,97$  для моделей обоих порядков, что свидетельствует об их высокой объяснительной способности.*

Таблица 4.9. – Таблица ошибок для количественного анализа скорости ветра в Турции (временной ряд 3) для моделей 1–2 порядков

Порядок модели	RMSE	$R^2$	MAPE	MAE	MSE
1	0.75	0.97	9.98	0.52	0.56
2	0.74	0.97	9.50	0.52	0.55

Как и в описанных в предыдущих разделах тестовых примерах при моделировании скорости ветра подход GLDM использует набор коэффициентов для описания динамики системы на различных уровнях сложности (см. табл. 4.10). Модель первого порядка (линейная) использует

простую формулировку с коэффициентами  $a_1$  и  $a_2$ , что обеспечивает баланс между простотой и возможностями прогнозирования. Модель второго порядка расширяет эту базу, добавляя в общей сложности пять коэффициентов (от  $a_1$  до  $a_5$ ), тем самым повышая способность модели фиксировать более сложные закономерности в данных о скорости ветра.

Таблица 4.10. – Результаты расчета коэффициентов модели для временного ряда 3 (скорость ветра для SCADA)

Коэффициент	Значение для моделей разного порядка	
	1	2
$a_1$	1.0092	<b>0.9300</b>
$a_2$	-0.0011	<b>0.0764</b>
$a_3$	–	<b>0.0248</b>
$a_4$	–	<b>0.0241</b>
$a_5$	–	<b>-0.0499</b>

Данные этого временного ряда 3 распределены равномерно, поэтому использование моделей 3–5 порядка оказалось нецелесообразным. Кроме того, для получения коэффициентов моделей более высоких порядков требуется использование более мощных вычислителей, что является темой отдельного исследования.

В таблице 4.11 представлены показатели производительности для различных моделей, используемых при прогнозировании скорости ветра. Среди рассматриваемых моделей метод обобщенных наименьших отклонений второго порядка (GLDM) демонстрирует высокую производительность по всем ключевым показателям.

Что касается других моделей, то многослойный перцептрон (MLP), несмотря на свою сложность, позволяет получить относительно высокие значения RMSE и MAPE, что указывает на менее точную работу. Метод опорных векторов (SVM) работает достаточно хорошо (см. высокое значение  $R^2$ ), однако его значения RMSE и MAPE выше, чем у GLDM.

Таблица 4.11. – Показатели производительности различных моделей для прогнозирования скорости ветра

Модель	RMSE	MSE	MAE	$R^2$	MAPE
MLP	5.070	25.70	4.05	0.04969	102.61%
SVM	0.80	0.64	0.638	0.96	12.13%
Auto-ARIMA	0.7453	0.5555	0.5209	0.9689	10.00%
Эксп.сглаж.	0.7493	0.5614	0.5226	0.9686	9.99%
BATS	2.4	5.76	1.915	0.3	19.15%
TBATS	4.6	21.16	3.671	0.45	24.36%
Prophet	3.8472	14.8009	3.0759	0.1717	78.03%
Гибрид autoArima+ES	0.90	0.81	0.718	0.19	71.80%
Гибрид autoArima+Polynomial	0.88	0.7744	0.702	0.2254	70.20%
<b>GLDM (порядок 2)</b>	<b>0.74</b>	<b>0.55</b>	<b>0.52</b>	<b>0.97</b>	<b>9.50%</b>

Модель auto-ARIMA также имеет высокую производительность при низких показателях ошибок и высоком значении  $R^2$ , что делает ее одним из лучших конкурентов рассматриваемого подхода для данного случая. Модель экспоненциального сглаживания в данном случае работает сравнимо с auto-ARIMA, но и она немного уступает точности, достигнутой моделью GLDM. Менее точные результаты дают модели BATS и TBATS и модель Prophet. Гибридные модели, такие как autoArima+ES и autoArima+Polynomial, обеспечивают более высокую точность прогнозирования по сравнению с отдельными моделями, но все же не превосходят производительность GLDM.

Таким образом, GLDM второго порядка является наиболее эффективной моделью для прогнозирования данных о скорости ветра в рамках проведенного сравнительного анализа. Ее способность минимизировать ошибки прогнозирования и устойчивость к выбросам делают ее важным инструментом для точного и надежного прогнозирования временных рядов схожей с рассмотренными набором данных структуры.

$$\begin{aligned}
\hat{y}_t = & (0.9300 \times y_{t-1}) + (0.0764 \times y_{t-2}) \\
& + (0.0248 \times y_{t-1}^2) + (0.0241 \times y_{t-1} \cdot y_{t-2}) \\
& + (-0.0499 \times y_{t-2}^2)
\end{aligned} \tag{4.6}$$

Модель для прогнозирования скорости ветра в момент времени  $t$ ,  $\hat{y}_t$ , включает в себя слагаемые для учета как линейных, так и нелинейных зависимостей от прошлых значений скорости ветра. Коэффициент  $a_1 = 0,9300$ , связанный с  $y_{t-1}$ , указывает на сильное положительное влияние скорости ветра на предыдущем временном шаге, предполагая, что недавние скорости ветра в значительной степени влияют на текущий прогноз. Коэффициент, равный  $a_2 = 0,0764$  при  $y_{t-2}$  имеет меньший коэффициент, означающий более слабое влияние скорости ветра на два временных шага ранее. Включение таких слагаемых как  $a_3 = 0,0248$  при  $y_{t-1}^2$ , вводит нелинейную составляющую, означающую, что более высокие значения  $y_{t-1}$  экспоненциально увеличивают прогнозируемую скорость ветра. Это показывает значительное влияние высоких скоростей ветра в предыдущий момент времени. Коэффициент взаимодействия  $a_4 = 0,0241$  при  $y_{t-1} \cdot y_{t-2}$  отражает совокупное влияние скоростей ветра в предыдущие два момента времени.

Наконец, коэффициент  $a_5 = -0,0499$  при  $y_{t-2}^2$  отрицателен и действует как демпфирующий механизм. Это слагаемое уменьшает прогнозируемую скорость ветра, когда  $y_{t-2}$  высока. Оно обеспечивает стабилизирующее влияние для противодействия влиянию других членов, особенно в периоды исключительно высоких скоростей ветра двумя моментами времени ранее. В совокупности эти коэффициенты и их знаки образуют комплексную модель. (Результаты были опубликованы в [33]).

## 4.4 Вычислительные эксперименты для временных рядов с данными по заражениям Covid-19

С началом пандемии Covid-19 появились тысячи исследований о прогнозировании ситуации с распространением вируса. Несмотря на то, что спустя несколько лет после начала пандемии прогнозирование развития пандемии было признано невозможным, за 2020–2022 годы было проведено немало ценных исследований, которые послужили развитием методов анализа данных.

Большинство работ было посвящено построению прогнозов с горизонтом 3–7 дней [7, 42, 71]. Все эти работы можно подразделить на три категории относительно используемой модели: модели SIR (классические эпидемиологические модели переходов состояний) [8, 40, 119], модели ARIMA (интегрированные модели авторегрессии и скользящего среднего) [109, 119] и модель Хольта—Уинтерса (адаптивного сглаживания) [8, 119]. Одной из достаточно удачных работ можно считать статью [8], опубликованную в России, в которой авторы рассматривают указанные три модели на временном ряде с марта 2020 г. (начало эпидемии в России) по 10 мая 2020 г.

Классическая модель SIR не позволила обеспечить высокое качество получаемых прогнозов [40, 119]. В работе [119] используется вышедшее не так давно расширение для `R covid19.analytics`, позволяющее получить полную картину по распространению вируса в любой точке земного шара.

Еще в начале пандемии появились статьи, использующие усложненные модели SIR, учитывающие как выбывание инфицированных из-за смертности (модель  $SIR(D)$  в КНР) [40], так и инкубационный период

заболевания (модель SEIR) для краткосрочного прогнозирования развития эпидемии в различных странах. Предпринимались попытки разработать модель SEIR в предположении, что развитие эпидемии начинается с подгруппы населения, характеризующейся слабым иммунным ответом, и на более поздней стадии продолжается во всем населении.

Модели, основанные на анализе временных рядов, в частности ARIMA-модели, являются сложно настраиваемыми при проведении полного анализа, однако дают практически всегда хороший результат там, где требуется качественный прогноз на среднесрочный и краткосрочный период [8]. В работах [75, 109], как и во многих других, авторы вручную подбирают параметры для имеющихся на момент публикации временных рядов и проводят расчеты в широко используемых для статистического анализа программах (R, Gretl) для отдельных государств и регионов.

Достаточно популярным инструментом прогнозирования развития распространения коронавирусной инфекции являются адаптивные модели экспоненциального сглаживания. Данные модели также послужили достаточно распространенным инструментом для построения прогнозов для временных рядов, соответствующих развитию эпидемии в различных странах [35, 119]. Как и в случае с моделью ARIMA, основным недостатком большинства представленных работ является отсутствие объяснения выбора соответствующей спецификации моделей, а также отсутствие «объяснения» подбора гиперпараметров моделей прогнозирования [8].

В 2021–2022 годах, когда уже было накоплено достаточно данных для применения методов машинного обучения, начали появляться исследования и доработки этих методов для анализа распространения панде-

мии. Например, для прогнозирования числа случаев заражения Covid-19 в мире [131] была разработана новая гибридная модель прогнозирования CNN-LSTM. Хотя предложенную модель сравнивали с 17 базовыми моделями, CNN-LSTM превзошла их все с наименьшей среднеквадратичной ошибкой (RMSE). Существует исследование по точному прогнозированию случаев заражения в Индии и Ченнаи с использованием моделей глубокого обучения. Авторы пришли к выводу, что сложенные LSTM превосходят ARIMA, LSTM и Prophet [52].

Рассмотрим вычислительный эксперимент по построению решения задачи Коши для одного квазилинейного разностного уравнения, и покажем, что полученное решение показывает высокое качество рассматриваемого алгоритма для неизвестного рекуррентного уравнения временного ряда.

Применение моделей обобщенного метода наименьших модулей (GLDM) к анализу динамики смертности от Covid-19 в России позволяет провести анализ последствий пандемии. Полученные модели первого и второго порядков показали близкие значения среднеквадратичной ошибки (RMSE), равные 33,78 и 33,31 соответственно (см. табл.

Таблица 4.12. – Результаты расчета коэффициентов модели для временного ряда 4 (смертность от Covid-19 в РФ)

Коэффициент	Значение для моделей разного порядка		
	1	2	3
$a_1$	1.0000	<b>0.7265</b>	0.5970
$a_2$	0.0000	<b>0.2610</b>	-0.3694
$a_3$	–	<b>0.0020</b>	0.7396
$a_4$	–	<b>0.0016</b>	0.0083
$a_5$	–	<b>-0.0036</b>	0.0101
$a_6$	–	–	-0.0009
$a_7$	–	–	-0.0185
$a_8$	–	–	0.0010
$a_9$	–	–	0.0000

Применение моделей обобщенного метода наименьших отклонений (GLDM) возможно не только для анализа смертности, но и для исследования динамики заражений. Ниже приведем анализ случаев заражения в Самарской области. Выбор региона обусловлен наличием наиболее полного набора данных.

В таблице 4.13 приведены коэффициенты моделей 1 и 2 порядков, описывающих динамику обнаружения новых случаев заражения.

Таблица 4.13. – Результаты расчета коэффициентов модели для анализа случаев заражения Covid-19 в Самарской области

Коэффициент	Значение для моделей разного порядка	
	1	2
$a_1$	1.0071	<b>1.2573</b>
$a_2$	$-1.2486 \times 10^{-5}$	<b>-0.2455</b>
$a_3$	–	<b>0.0002</b>
$a_4$	–	<b>0.0001</b>
$a_5$	–	<b>-0.0003</b>

На рисунках 4.2 и 4.3 приведено сравнение реальных данных о случаях заражения COVID-19 с данными моделей первого и второго порядка соответственно.

На рисунках 4.4, 4.5 и 4.6 сравниваются исходные данные о случаях смерти от COVID-19 с модельными данными GLDM первого, второго и третьего порядков для России.

Time Series: COVID-19 infection cases in the Samara region: Original vs GLDM Model

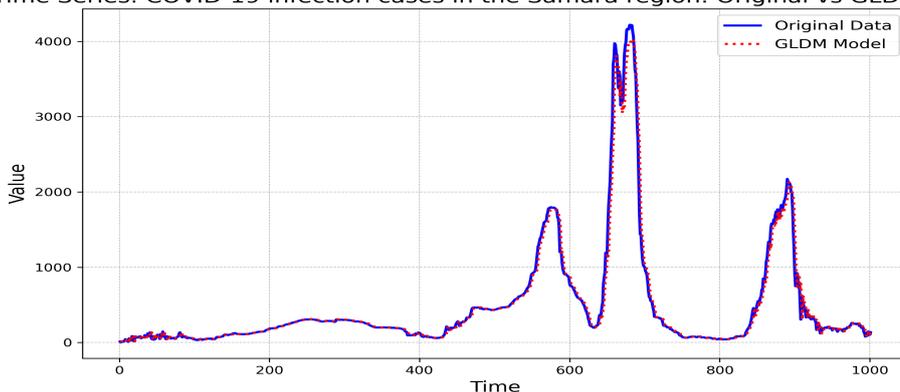


Рисунок 4.2. – Сравнение реальных данных и данных, полученных с помощью модели первого порядка для случаев заражения COVID-19 в Самарской области

Time Series COVID-19 infection cases in the Samara region: Original vs GLDM Model

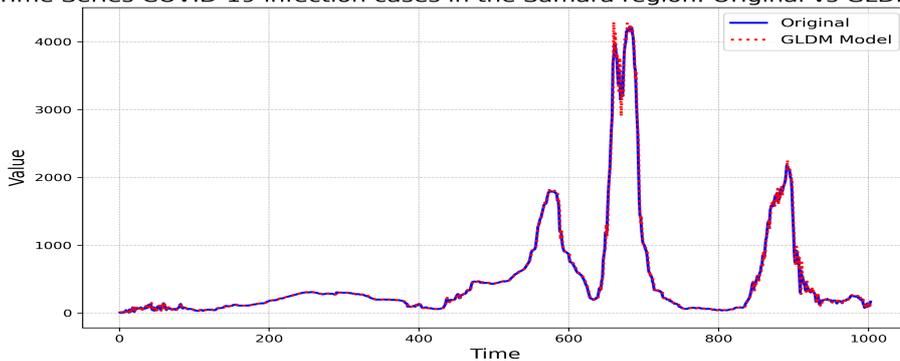


Рисунок 4.3. – Сравнение реальных данных и данных, полученных с помощью модели второго порядка для случаев заражения COVID-19 в Самарской области

Time Series: Covid 19 deaths cases in Russia: Original vs GLDM Model

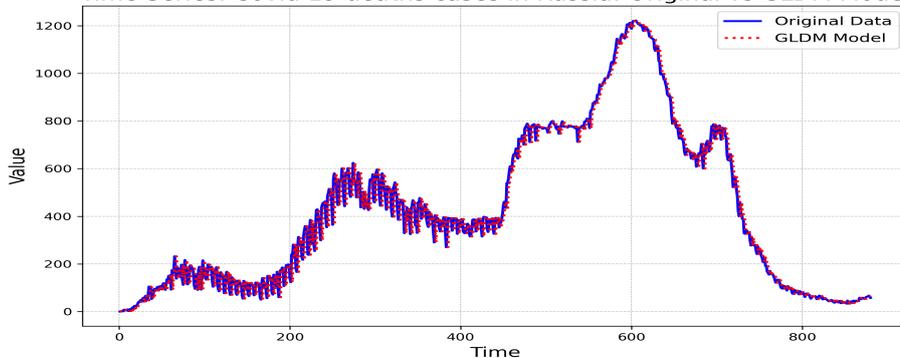


Рисунок 4.4. – Временной ряд: случаи смерти от COVID-19 в России, описанные моделью GLDM первого порядка

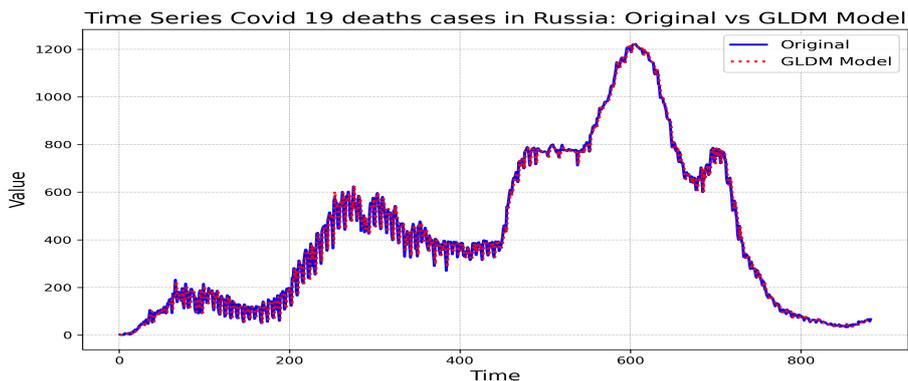


Рисунок 4.5. – Временной ряд: случаи смерти от COVID-19 в России, описанные моделью GLDM второго порядка

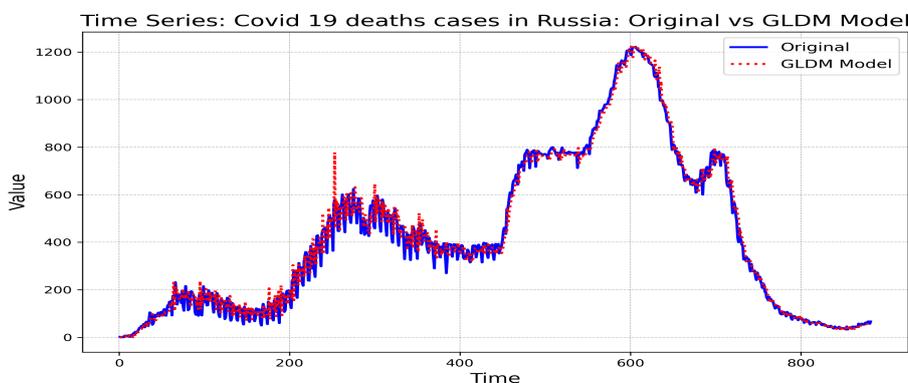


Рисунок 4.6. – Временной ряд: случаи смерти от COVID-19 в России, описанные моделью GLDM третьего порядка

В таблице 4.14 приводятся значения ошибок, полученные для исследованных моделей смертности в РФ и заражения в Самарской области. Здесь подробно описаны показатели эффективности моделей, дающие подробное представление об их эффективности с помощью различных статистических показателей.

Таблица 4.14. – Таблица значений ошибок при моделировании смертности от COVID-19 в России и заражений в отдельном регионе

Регион	Order	RMSE	$R^2$	MAPE	MAE	MSE	ME
Самарская обл.	1	69.81	0.9927	9.18	27.99	4872.80	6.99
	2	58.20	0.9943	9.36	23.68	3387.41	-3.48
Россия	1	33.78	0.9896	11.12	22.24	1141.25	0.074
	2	33.31	0.9898	10.96	22.22	1109.29	0.43
	3	41.43	0.9843	13.47	29.37	1716.66	-2.79

Так, для данных о смертности в России, модели GLDM первого, так и второго порядка имеют близкие значения RMSE – 33,78 и 33,31 соответственно, что свидетельствует об их эффективности для рассматриваемого временного ряда. Однако более высокое значение RMSE модели третьего порядка, равное 41,43, может указывать на переобучение, следовательно, повышенная сложность модели не обязательно повышает точность прогноза.

Сравнительные показатели прогнозов моделей с фактическими данными для Самарской области и России иллюстрируют высокую точность моделей GLDM в описании динамики эпидемиологических событий.

Сводный анализ суммы абсолютных разностей (SAD, Sum of Absolute Differences) для исследуемых данных для полученных моделей представлен в таблице 4.15.

Таблица 4.15. – Сумма абсолютных различий для исследуемых моделей

Регион	Порядок модели	Сумма абсолютных разностей
Самара	1	28044.96
	2	23704.61
Россия	1	19591.00
	2	19554.24
	3	23238.35

Этот анализ дает представление об эффективности различных моделей GLDM. В Самарской области модель второго порядка показывает более низкую эффективность по сравнению с моделью первого порядка, что свидетельствует о более высокой точности моделирования тенденции заражения.

Таким образом, модель динамики заражений COVID-19 в Самарской

области имеет вид:

$$\begin{aligned}\hat{y}_t &= (1.2573 \times y_{t-1}) + (-0.2455 \times y_{t-2}) \\ &+ (0.0002 \times y_{t-1}^2) + (0.0001 \times y_{t-1} \cdot y_{t-2}) \\ &+ (-0.0003 \times y_{t-2}^2)\end{aligned}\quad (4.7)$$

Коэффициент  $a_1 = 1,2573 > 0$ , что указывает на прямую зависимость между случаями заражения за предыдущий день  $y_{t-1}$  и прогнозируемым значением  $\hat{y}_t$ . Это говорит о том, что увеличение числа случаев по сравнению с предыдущим днем положительно влияет на прогноз на текущий день.

И наоборот, коэффициент  $a_2 = -0,2455 < 0$ , что подразумевает наличие обратной связи для случаев инфицирования за два дня до этого  $y_{t-2}$ . Более высокое число случаев два дня назад связано с более низким прогнозом на сегодняшний день, если учесть влияние предыдущего дня. На полученное значение могут, например, влиять такие факторы, как принятые после всплеска заболеваемости меры или естественные колебания в распространении вируса.

Дополнительные коэффициенты от  $a_3$  до  $a_5$  дополнительно уточняют модель, отражая незначительные изменения в динамике заражения на основе взаимовлияния данных предыдущего дня и дня, ему предшествующего. Это отражает сложность динамики передачи заболевания и позволяет модели учитывать незначительные изменения в данных с течением времени.

Для данных по смертности в России модель второго порядка демонстрирует лишь незначительное улучшение показателя SAD по сравнению с моделью первого порядка, а модель третьего порядка демонстрирует более высокий показатель SAD, что снова указывает на то, что

более сложные модели не всегда дают лучшие результаты прогнозирования. В таблице приведена количественная оценка точности моделей GLDM, позволяющая легко сравнить их эффективность в различных контекстах.

Лучшая из полученных моделей для анализа динамики смертности в России имеет вид:

$$\begin{aligned} \hat{y}_t = & (0.7265 \times y_{t-1}) + (0.2610 \times y_{t-2}) \\ & + (0.0020 \times y_{t-1}^2) + (0.0016 \times y_{t-1} \cdot y_{t-2}) \\ & + (-0.0036 \times y_{t-2}^2) \end{aligned} \quad (4.8)$$

где  $\hat{y}_t$  – прогнозируемое число смертей в момент времени  $t$ . Здесь оба коэффициента положительны, что указывает на то, что более высокое число смертей за последние два дня ( $y_{t-1}$  и  $y_{t-2}$ ) связано с более высоким прогнозируемым числом данного события. Коэффициент  $a_1 = 0,7265$  указывает на более сильное влияние количества смертей за предыдущий день на текущий прогноз, в то время как  $a_2 = 0,2610$  указывает на несколько меньшее, но все же положительное влияние по сравнению с двумя предыдущими днями.

Эти положительные коэффициенты отражают длительность события; в данном случае это может свидетельствовать о продолжающейся передаче вируса или других факторах, способствующих постоянному числу смертей. Отсутствие отрицательного коэффициента означает, что в течение периода времени, рассматриваемого для этой модели, не было обнаружено обратной зависимости.

Эффективность различных моделей для прогнозирования случаев заражения COVID-19 в Самарской области и случаев смерти от COVID-19 в Российской Федерации сравнивается в таблицах 4.16 и 4.17. В этих

таблицах приводятся только ключевые показатели ошибок:  $R^2$  и среднему значению абсолютной процентной погрешности (MAPE).

Таблица 4.16. – Показатели ошибок ( $R^2$  и MAPE) для различных моделей случаев заражения COVID-19 в Самарской области

Модель	$R^2$	MAPE (%)
MLP	0.0814	197.1749
SVM	0.4098	12.5884
Auto ARIMA	0.9956	9.7693
Эксп.сглаж.	0.9946	9.5721
BATS	0.9940	10.1551
TBATS	0.9940	10.1551
Prophet	0.6345	179.7204
Гибрид autoARIMA+ES	0.9955	10.8418
Гибрид autoARIMA+Polynomial	0.9956	9.7832
<b>GLDM (порядок 2)</b>	<b>0.9943</b>	<b>9.3600</b>

Таблица 4.17. – Показатели ошибок ( $R^2$  и MAPE) для различных моделей случаев смерти от COVID-19 в России

Модель	$R^2$	MAPE (%)
MLP	0.0446	167.1630
SVM	0.9742	17.3852
Auto-ARIMA	0.9917	11.0454
Эксп.сглаж.	0.9898	11.1647
BATS	0.9921	11.0584
TBATS	0.9921	11.0584
Prophet	0.9746	53.7971
Гибрид autoARIMA+ES	0.9915	11.2055
Гибрид autoARIMA+Polynomial	0.9917	11.0749
<b>GLDM (порядок 2)</b>	<b>0.9898</b>	<b>10.9600</b>

Из таблиц 4.16 и 4.17 видно, что модель GLDM второго порядка имеет наивысшую производительность из рассмотренных.

Результаты были опубликованы в [22]

## 4.5 Влияние длины временного ряда на выбор порядка модели

Длина набора данных является определяющим фактором при обучении модели. Достаточная длина данных гарантирует, что модель будет отражать важные тенденции и сезонность без переобучения. Данные о длине рассматриваемых при проведении вычислительных экспериментов наборов приведены в табл. 4.1. Обратим внимание, что длина временного ряда напрямую влияет на возможность построения моделей более высокого порядка. Например, данные о температуре и скорости ветра (содержащие порядка  $10^4 - 10^5$  значений) поддерживают модели более высоких порядков. Напротив, небольшая длина временного ряда (данные о вегетационном индексе) ограничивает возможную сложность модели. В данном случае оптимальным является второй порядок модели.

Заметим также, что несмотря на большой объем доступных данных о скорости ветра (50 530 точек) по сравнению с температурой (9 999), оптимальный порядок модели, выбранный для данных о температуре, – пятый, а для скорости ветра – второй. На это решение влияют несколько ключевых факторов, включая характер набора данных, лежащую в его основе динамику и степень присутствующей нелинейности. На выбор порядка построения модели влияет не только размер исходного временного ряда, но и его структура. Решающее значение при выборе подходящего порядка модели для эффективного прогнозирования процесса играет сложность динамики, которую отражает модель, а не количество данных.

Таким образом, анализ, представленный в предыдущих разделах, ил-

люстрирует важность выбора подходящего порядка построения моделей на основе конкретных характеристик и характера динамики каждого набора данных. В данном разделе описан подбор оптимальных порядков моделей для определения вегетационного индекса, температуры, скорости ветра и анализа эпидемических процессов. При выборе порядка модели учитывалась как сложность получаемой модели, так и возможность фиксировать соответствующие закономерности без переобучения. Полученные результаты показывают: несмотря на то, что более длинные наборы данных могут поддерживать более высокие порядки модели, присущая имеющимся данным структура и закономерности в первую очередь определяют наиболее подходящую сложность. Наличие нулевых коэффициентов в этих моделях также говорит об эффективности обобщенного метода наименьших отклонений (GLDM) в устранении факторов, не влияющих на результаты прогнозирования, тем самым оптимизируя модели для повышения производительности и интерпретируемости. Такой подход гарантирует, что каждая модель является не только статистически обоснованной, но и практически актуальной, предоставляя надежные инструменты для прогнозирования и дальнейшего анализа.

Коэффициенты модели для временного ряда 2 (температура) пятого порядка, в частности, за исключением  $a_1 = 0$ , отражают существенные взаимодействия между запаздывающими переменными исходного ряда. Нулевое значение параметра  $a_1$  может указывать на его незначительный вклад в модель. Разреженность коэффициентов является преимуществом, поскольку она может повысить интерпретируемость модели и потенциально повысить эффективность обобщения за счет уменьшения чрезмерной сложности.

## 4.6 Прочие вычислительные эксперименты

При разработке и анализе работы модели были проведены многочисленные вычислительные эксперименты, результаты которых опубликованы в различных научных изданиях. Приведем перечень и результаты наиболее значимых из них.

1. **Определение параметров квазилинейного уравнения для количества кумулятивных случаев заражения Covid-19 [101].** Данные: количество кумулятивных случаев заражения Covid-19 по Челябинской области за первые 500 дней пандемии. Объект исследования: временные ряды длиной 150, 300, 500 элементов для модели второго порядка. Результат эксперимента: для временного ряда из 300 значений имеем наименьшее значение функции потерь, таким образом, наиболее точная модель для отражения динамики эпидемиологического процесса должна содержать данные примерно за один год; модель второго порядка адекватно описывает монотонный процесс.
2. **Определение параметров квазилинейного уравнения для количества инфицированных Covid-19 [30].** Данные: количество инфицированных Covid-19 на июль 2022 года по четырем регионам РФ (Москва, Санкт-Петербург, Челябинская и Свердловская области). Объект исследования: временной ряд, содержащий 700 значений. Результат эксперимента: модель второго порядка адекватно описывает колебательный процесс.
3. **Улучшение прогнозирования COVID-19 в Дагестане с использованием квазилинейных рекуррентных уравнений [31].** В данной работе представлен усовершенствованный подход

к прогнозированию распространения COVID-19 в Республике Дагестан на основе метода обобщенного наименьшего отклонения (GLDM) и квазилинейных рекуррентных уравнений. Анализ проводился на основе одномерных временных рядов, включающих данные о ежедневных случаях заражения за период в 1005 дней. Исследуемыми моделями выступили модели первого и второго порядка. Результаты экспериментов показали, что модель второго порядка значительно улучшает точность прогнозов по сравнению с моделью первого порядка. Это подтверждается снижением значения корневой средней квадратической ошибки (RMSE) и увеличением коэффициента детерминации  $R^2$ , что свидетельствует о высоком уровне соответствия прогнозов реальным данным и эффективности предложенного метода для предсказания будущих эпидемиологических тенденций.

4. **Продвинутое моделирование производства молока с использованием высокопорядкового метода обобщенного наименьшего отклонения** [26]. В статье представлен инновационный подход к прогнозированию объемов производства молока в США с использованием метода GLDM, учитывающего нелинейные зависимости и сезонные колебания. На основе данных за 168 месяцев разработаны модели различных порядков, где модель пятого порядка показала наибольшую точность ( $R^2 = 0.9714$ , RMSE = 17.2027). Модель превосходит традиционные методы временных рядов, такие как Prophet, AutoARIMA и BATS, по точности прогнозирования, что делает её эффективным инструментом для молочной промышленности и управления цепочками поставок.

## 4.7 Выводы и результаты по главе 4

На основе анализа временных рядов с данными из различных областей с помощью обобщенного метода наименьших отклонений (GLDM) был определен оптимальный порядок построения моделей для прогнозирования. Полученные результаты показывают, что порядок прогнозирующей модели, в значительной степени зависит от характеристик набора данных (характер данных, лежащая в их основе динамика и наличие нелинейных закономерностей, количество доступных данных).

Результаты проведенных вычислительных экспериментов показали, что в большинстве случаев для воссоздания уравнения, описывающего динамику временного ряда, достаточно использовать модель второго порядка. Тем не менее, для рядов, характеризующихся значительными сезонными колебаниями и автокорреляциями (например, данные о десятилетних колебаниях температуры) модели более высоких порядков дадут более точные результаты.

Результаты прочих вычислительных экспериментов приведены в [22, 33]. Поскольку для сравнения были использованы простейшие нейросетевые подходы, в качестве темы отдельного исследования можно было бы рассмотреть построение модели машинного обучения, превосходящей описанный детерминированный подход, но данное исследование представляется авторам нецелесообразным, поскольку использование высококачественных детерминированных подходов является наиболее приоритетным.

Результаты, приведенные в данной главе, опубликованы в работах [4, 14, 21, 22, 26, 33].

## ЗАКЛЮЧЕНИЕ

В диссертации разработаны и исследованы новые методы и алгоритмы определения коэффициентов квазилинейного рекуррентного уравнения. В настоящее время эффективные детерминированные методы для анализа динамических процессов, представленных временными рядами, является актуальной проблемой, решение которой востребовано при проведении прикладных исследований в различных областях: от промышленности, до медицины и социально-экономических приложений. Большинство из известных в настоящее время методов применимы для краткосрочного прогнозирования, поэтому актуальна задача разработки математического подхода, позволяющего в явном виде получать качественные квазилинейные разностные уравнения (адекватно описывающие рассматриваемый процесс). Кроме того, в настоящее время активно исследуются нейросетевые модели и модели глубокого обучения, обладающие одним существенным недостатком, связанным с невозможностью точно описать параметры процесса. Детерминированные подходы имеют перед данными методами преимущество, заключающееся в возможности описания процесса уравнением.

Оригинальность алгоритма определения параметров квазилинейного рекуррентного уравнения заключается в том, что он вместо метода наименьших квадратов использует обобщенный метод наименьших модулей с подбором весов. Показано, что задача GLDM-оценки, является задачей многоэкстремальной оптимизации, и GLDM-оценки устойчивы к наличию корреляции значений в  $\{y_t \in \mathbb{R}\}_{t=1-m}^T$  и (при соответствующих настройках) лучше всего подходят для вероятностных распределений ошибок с более тяжелыми (чем у нормального распределения) хвостами. Таким образом, имеется возможность решения задачи иденти-

фикации (1.31) с использованием решения (1.32), а описание алгоритма **GLDM-оценка** показывает, что его вычислительная сложность пропорциональна вычислительной сложности алгоритма решения прямой и/или двойственной задачи WLDM (2.1).

В результате исследования разработана система прогнозирования, которая обладает следующими свойствами: (1) способна обрабатывать любые временные ряды произвольной длины и не зависит от характера прикладных задач; (2) автоматически настраивает параметры используемых моделей; (3) расширяема, т.е. допускает подключение дополнительных модулей для прогнозирования; (4) позволяет выбрать критерий отбора «лучшей» модели. В данную систему включен модуль, реализующий метод определения параметров квазилинейного рекуррентного уравнения.

Говоря о качестве самой модели на основе метода наименьших отклонений, можно отметить, что она работает не хуже нейросетевых моделей или классических статистических моделей, что подтверждается и проведенными экспериментами, в которых были сравнены результаты работы рассмотренной модели и результаты, полученные с помощью известных моделей. У модели, использующей обобщенный метод наименьших отклонений (GLDM) есть одно существенное преимущество по сравнению с этими моделями, заключающееся в возможности интерпретировать коэффициенты модели с точки зрения задачи исследования.

Результаты исследования могут быть применены для моделирования динамически развивающихся процессов в широком спектре областей, поскольку разработанный алгоритм определения коэффициентов квазилинейного уравнения и система прогнозирования, его включающая, не зависят от специфики предметной области.

Основные результаты, полученные в ходе выполнения диссертационного исследования, являются новыми и не покрываются работами других авторов, обзор которых был приведен в главе 1. Основные отличия заключаются в следующем.

1. Алгоритм **GLDM-оценки**, идеи которого были рассмотрены, позволяет получить коэффициенты квазилинейного рекуррентного соотношения, с помощью которых можно не только воссоздать заданный значениями временного ряда динамический процесс, но и построить качественный прогноз. В работе доказано, что коэффициенты, полученные при помощи указанного алгоритма, дают минимальную невязку, а последовательность  $\{(A^{(k)}, z^{(k)})\}_{k=1}^{\infty}$ , построенная по алгоритму GLDM-оценки сходится к глобальному минимуму  $(a^*, z^*)$  задачи (1.32). Приведены теоретические обоснования выбора порядков модели для наборов данных различного объема и структуры, обсуждается роль нулевых коэффициентов в повышении эффективности модели. Приведены условия наличия нулевых коэффициентов модели, сформулированы условия, при которых ряд коэффициентов модели может быть обнулен.
2. Разработанная система прогнозирования получает в качестве входных данных только временной ряд, пользователь не должен обладать дополнительной информацией о структуре временного ряда, специфике предметной области, настраивать параметры моделей. Система является расширяемой, она допускает подключение дополнительных модулей прогнозирования и включение их в анализ точности работы моделей. Система позволяет анализировать временные ряды для различных прикладных задач. Исследования возможностей использования параллелизма для алгоритма идентификации коэффициентов модели с помощью обобщенного мето-

да наименьших модулей для достаточно большого числа коротких временных рядов показали, что можно разработать параллельное персональное приложение для ноутбуков и/или мобильных телефонов, максимально используя ресурсы этих устройств и позволяя получить результат в 5–6 раз быстрее, поскольку разница во времени использования персонального компьютера для расчета больших наборов данных будет иметь важное значение.

### **Будущие направления исследований**

Направления будущих исследований включают разработку двусторонних GLDM моделей высокого порядка. Оптимизация модели позволяет найти оптимальные коэффициенты, минимизирующие отклонение модели от реальных данных. Это расширение позволяет значительно улучшить качество прогнозирования и адаптацию модели к реальным данным.

Дальнейшие исследования будут включать адаптацию GLDM для многомерных временных рядов, исследование интеграции лаговых членов порядка выше пятого, а также исследование возможности включения дополнительных объясняющих переменных для дальнейшего улучшения предсказательных возможностей модели и адаптивности к сложным реальным данным.

## СПИСОК ЛИТЕРАТУРЫ

1. Аботалёб М. С. А. Установление значимости коэффициентов квазилинейного уравнения N-факторной авторегрессии // Проблемы информатики. — 2024. — № 3. — С. 5—28. — DOI: 10.24412/2073-0667-2024-3-5-28.
2. Аботалёб М. С. А., Макаровских Т. А. Univariate time series software Version 1 (Программное обеспечение для одномерных временных рядов Версия 1) // Свидетельство Роспатента о государственной регистрации программы для ЭВМ № 2024682644. от 25.09.2024. — URL: <https://elibrary.ru/item.asp?id=73234288>.
3. Аботалёб М. С. А., Макаровских Т. А. Библиотека для квазилинейного анализа временных рядов // Свидетельство Роспатента о государственной регистрации программы для ЭВМ № 2022669812. от 25.10.2022. — URL: <https://elibrary.ru/item.asp?id=49777252>.
4. Аботалёб М. С. А., Макаровских Т. А., Панюков А. В. Возможности параллелизма при идентификации квазилинейного рекуррентного уравнения // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. — 2023. — Т. 12, № 4. — С. 94—109. — DOI: 10.14529/cmse230404.
5. Брейдо И. В. [и др.]. Создание модели прогнозирования энергопотребления на основе адаптивной нейро-нечеткой системы вывода в MATLAB // Вестник Казахской академии транспорта и коммуникаций им. М. Тынышпаева. — 2021. — 1(116). — С. 331—338. — DOI: 10.52167/1609-1817-2020-116-1-331-338.

6. Буховец А. Г. [и др.]. Моделирование динамики вегетационного индекса NDVI озимой пшеницы в условиях ЦФО // Вестник Воронежского государственного аграрного университета. — 2018. — № 2. — С. 186—199.
7. Зимовец А. В., Сорокина Ю. В., Ханина А. В. Анализ влияния пандемии COVID-19 на развитие предприятий в Российской Федерации // Экономика, предпринимательство и право. — 2020. — Т. 10, № 5. — С. 1337—1350. — DOI: 10.18334/err.10.5.110126.
8. Лакман И. [и др.]. Возможности математического прогнозирования коронавирусной инфекции в Российской Федерации // Артериальная гипертензия. — 2020. — Т. 26, № 3. — С. 288—294. — DOI: 10.18705/1607-419X-2020-26-3-288-294.
9. Лысенко С. А. Прогнозирование урожайности сельскохозяйственных культур на основе спутникового мониторинга динамики углерода в наземных экосистемах // Исследование Земли из космоса. — 2019. — № 4. — С. 48—59. — DOI: 10.31857/S0205-96142019448-59.
10. Макаровских Т. А., Аботалёв М. С. А. GLDMPredictor. — 2022. — URL: <https://github.com/tmakarovskikh/GLDMPredict> (дата обр. 01.02.2021).
11. Макаровских Т. А., Аботалёв М. С. А. Библиотека epidemic.ta для прогнозирования случаев заражения Covid-19 // Свидетельство Роспатента о государственной регистрации программы для ЭВМ № 2021616747. от 26.04.2021. — URL: <https://elibrary.ru/item.asp?id=45822962>.
12. Макаровских Т. А., Аботалёв М. С. А. Параллельная программа для идентификации параметров квазилинейных рекуррентных

- соотношений для массива временных рядов // Свидетельство Роспатента о государственной регистрации программы для ЭВМ № 2023661570. от 01.06.2023. — URL: <https://elibrary.ru/item.asp?id=54048362>.
13. Макаровских Т. А., Аботалев М. С. А. Система прогнозирования для одномерных временных рядов // Свидетельство Роспатента о государственной регистрации программы для ЭВМ № 2022660664. от 08.06.2022. — URL: <https://elibrary.ru/item.asp?id=49196026>.
  14. Макаровских Т. А., Абделсалам А. М. С. Автоматический подбор параметров модели ARIMA для прогноза количества случаев заражения и смерти от COVID-19 // Вестник Южно-Уральского государственного университета. Серия: Вычислительная математика и информатика. — 2021. — Т. 10, № 2. — С. 20—37. — DOI: 10.14529/cmse210202.
  15. Петин Г. А., Зыкина А. В. Подбор параметров для решения непрерывных оптимизационных задач методом внутренних штрафных функций // Прикладная математика и фундаментальная информатика. — 2022. — Т. 9, № 4. — С. 4—16. — DOI: 10.25206/2311-4908-2021-9-4-4-16.
  16. Сторчак И. Г. Прогноз урожайности озимой пшеницы с использованием вегетационного индекса NDVI для условий Ставропольского края : дис. ... канд. сельскохозяйственных наук : 06.01.01 / Сторчак Ирина Геннадьевна. — Ставрополь, 2016. — 160 с. — URL: <https://elibrary.ru/item.asp?id=30445017>.
  17. Халил З. Х., Абдуллаев С. М. Регрессионные прогнозы урожайности орошаемых озимых культур с использованием спутниковых

- вегетационных индексов: модели, предикторы и эксперименты // Вестник ЮУрГУ. Серия «Компьютерные технологии, управление, радиоэлектроника». — 2021. — Т. 21, № 2. — С. 23—35. — DOI: 10.14529/ctcr210203.
18. COVID-19 Data Repository / Center for Systems Science, Engineering (CSSE) at Johns Hopkins University. — 2021. — URL: <https://arcg.is/0fHmTX> (visited on 04/07/2021).
  19. COVID-19 Death Cases in the Russian Federation / Yandex DataLens. — 2024. — URL: <https://datalens.yandex.cloud/wizard/k7rsfswptkk2f-world-stat-table> (visited on 05/25/2024).
  20. COVID-19 Infection Cases in Samara / Yandex DataLens. — 2024. — URL: <https://datalens.yandex.cloud/wizard/pa8c25tdkuy0k-russia-stat-table> (visited on 05/25/2024).
  21. Abotaleb M. Proving optimal model selection and zero coefficient cases in time series forecasting with the generalized least deviation algorithm // Advanced Mathematical Models & Applications. — 2024. — Dec. — Vol. 9, no. 3. — DOI: 10.62476/amma93475.
  22. Abotaleb M. Soft Computing-Based Generalized Least Deviation Method Algorithm for Modeling and Forecasting COVID-19 using Quasilinear Recurrence Equations // Iraqi Journal For Computer Science and Mathematics. — 2024. — Vol. 5, no. 3. — P. 441–472. — DOI: 10.52866/ijcsm.2024.05.03.028.
  23. Abotaleb M., Dutta P. K. Optimizing bidirectional long short-term memory networks for univariate time series forecasting: a comprehensive guide. — Walter de Gruyter GmbH & Co KG, 2024. — P. 443. — DOI: 10.1515/9783111331133-021.

24. Abotaleb M., Dutta P. K. Optimizing convolutional neural networks for univariate time series forecasting: a comprehensive guide. — Walter de Gruyter GmbH & Co KG, 2024. — P. 459. — DOI: 10.1515/9783111331133-022.
25. Abotaleb M., Dutta P. K. Optimizing long short-term memory networks for univariate time series forecasting: a comprehensive guide. — 2024. — P. 427–442. — DOI: 10.1515/9783111331133-020.
26. Abotaleb M., Makarovskikh T. Advanced Milk Production Modelling Using High-Order Generalized Least Deviation Method // Modeling Earth Systems and Environment. — 2024. — P. 1–29. — DOI: 10.1007/s40808-024-02142-9.
27. Abotaleb M., Makarovskikh T. Development of Algorithm for Forecasting System Software // International Conference on Time Series and Forecasting. — Springer. 2021. — P. 213–225. — DOI: 10.1007/978-3-031-14197-3\_14.
28. Abotaleb M., Makarovskikh T. Development of Algorithms for Choosing the Best Time Series Models and Neural Networks to Predict COVID-19 Cases // Bulletin of the South Ural State University Ser Computer Technologies Automatic Control & Radioelectronics. — 2021. — Aug. — Vol. 21. — P. 26–35. — DOI: 10.14529/ctcr210303.
29. Abotaleb M., Makarovskikh T. System for forecasting COVID-19 cases using time-series and neural networks models // Engineering proceedings. — 2021. — Vol. 5, no. 1. — P. 46. — DOI: 10.3390/engproc2021005046.

30. Abotaleb M., Makarovskikh T., Ramadhan A. J. Exploring the Identification of Autoregression Model by General Least Deviation Method // Al-Bahir Journal for Engineering and Pure Sciences. — 2024. — Vol. 5, no. 2. — Article 4. — DOI: 10.55810/2313-0083.1074. — URL: 10.55810/2313-0083.1074.
31. Abotaleb M. [et al.]. Enhancing COVID-19 Forecasting in Dagestan with Quasi-linear Recurrence Equations by using GLDM Algorithm // Al-Bahir Journal for Engineering and Pure Sciences. — 2024. — Vol. 5, no. 2. — DOI: 10.55810/2313-0083.1075.
32. Abotaleb M. [et al.]. Modeling Covid-19 infection cases and vaccine in 5 countries highly vaccinations // Turkish Journal of Mathematics and Computer Science. — 2021. — Vol. 13, no. 2. — P. 403–417. — DOI: 10.47000/tjmcs.905508.
33. Abotaleb M. O. Solving The Optimizing Parameters Problem For Non-linear Datasets Using The High-Order General Least Deviations Method (GLDM) Algorithm // Computational Methods for Differential Equations. — 2024. — DOI: 10.22034/cmde.2024.62441.2751.
34. Abotaleb M. S., Makarovskikh T. Analysis of neural network and statistical models used for forecasting of a disease infection cases // 2021 International Conference on Information Technology and Nanotechnology (ITNT). — IEEE. 2021. — P. 1–7. — DOI: 10.1109/ITNT52450.2021.9649126.
35. Abotaleb M. S. A. Predicting COVID-19 cases using some statistical models: an application to the cases reported in China Italy and USA // Academic Journal of Applied Mathematical Sciences. —

2020. — Vol. 6, no. 4. — P. 32–40. — DOI: 10.32861/ajams.64.32.40.
36. Ahmad R. [et al.]. A machine-learning based ConvLSTM architecture for NDVI forecasting // International Transactions in Operational Research. — 2020. — Oct. — Vol. 30, no. 4. — P. 2025–2048. — DOI: 10.1111/itor.12887.
37. Alam M. S. [et al.]. Forecasting oil, coal, and natural gas prices in the pre-and post-COVID scenarios: Contextual evidence from India using time series forecasting tools // Resources Policy. — 2023. — Vol. 81. — P. 103342. — DOI: 10.1016/j.resourpol.2023.103342.
38. Antonau I., Hojjat M., Bletzinger K.-U. Relaxed gradient projection algorithm for constrained node-based shape optimization // Structural and Multidisciplinary Optimization. — 2021. — Vol. 63, no. 4. — P. 1633–1651. — DOI: 10.1007/s00158-020-02821-y.
39. Bakar M. A. A. [et al.]. Forecasting the air quality time series data using univariate and multivariate convolutional neural network models // AIP Conference Proceedings. Vol. 2895. — AIP Publishing. 2024. — DOI: 10.1063/5.0192193.
40. Banerjee M., Tokarev A., Volpert V. Immuno-epidemiological model of two-stage epidemic growth // Mathematical Modelling of Natural Phenomena. — 2020. — Vol. 15. — P. 27. — DOI: 10.1051/mmnp/2020012.
41. Benvenuto D. [et al.]. Application of the ARIMA model on the COVID-2019 epidemic dataset // Data in brief. — 2020. — Vol. 29. — P. 105340. — DOI: 10.1016/j.dib.2020.105340.

42. Biglieri S., De Vidovich L., Keil R. City as the core of contagion? Repositioning COVID-19 at the social and spatial periphery of urban society // *Cities & Health*. — 2020. — P. 1–3. — DOI: 10.1080/23748834.2020.1788320.
43. Borghi P. H., Zakordonets O., Teixeira J. P. A COVID-19 time series forecasting model based on MLP ANN // *Procedia Computer Science*. — 2021. — Vol. 181. — P. 940–947. — DOI: 10.1016/j.procs.2021.01.250.
44. Carreño-Conde F. [et al.]. A Forecast Model Applied to Monitor Crops Dynamics Using Vegetation Indices (NDVI) // *Applied Sciences*. — 2021. — Feb. — Vol. 11, no. 4. — P. 1859. — DOI: 10.3390/app11041859.
45. Castán-Lascorz M. [et al.]. A new hybrid method for predicting univariate and multivariate time series based on pattern forecasting // *Information Sciences*. — 2022. — Vol. 586. — P. 611–627. — DOI: 10.1016/j.ins.2021.12.001.
46. Chen Z. [et al.]. Long sequence time-series forecasting with deep learning: A survey // *Information Fusion*. — 2023. — Vol. 97. — P. 101819. — DOI: 10.1016/j.inffus.2023.101819.
47. Chhabra A. [et al.]. Sustainable and intelligent time-series models for epidemic disease forecasting and analysis // *Sustainable Technology and Entrepreneurship*. — 2024. — Vol. 3, no. 2. — P. 100064. — DOI: 10.1016/j.stae.2023.100064.
48. Chung J. [et al.]. Empirical evaluation of gated recurrent neural networks on sequence modeling // arXiv preprint arXiv:1412.3555. — 2014. — DOI: 10.48550/arXiv.1412.3555.

49. Daughton A. R. [et al.]. An approach to and web-based tool for infectious disease outbreak intervention analysis // Scientific reports. — 2017. — Vol. 7, no. 1. — P. 1–11. — DOI: 10.1038/srep46076.
50. De Livera A. M., Hyndman R. J., Snyder R. D. Forecasting time series with complex seasonal patterns using exponential smoothing // Journal of the American statistical association. — 2011. — Vol. 106, no. 496. — P. 1513–1527. — DOI: 10.1198/jasa.2011.tm09771.
51. De Livera A. M., Hyndman R. J., Snyder R. D. Forecasting time series with complex seasonal patterns using exponential smoothing // Journal of the American statistical association. — 2011. — Vol. 106, no. 496. — P. 1513–1527. — DOI: 10.1198/jasa.2011.tm09771.
52. Devaraj J. [et al.]. Forecasting of COVID-19 cases using deep learning models: Is it reliable and practically significant? // Results in Physics. — 2021. — Vol. 21, no. 103817. — DOI: 10.1016/j.rinp.2021.103817.
53. Dolganina N. [et al.]. HPC Resources of South Ural State University // Parallel Computational Technologies. PCT 2022. Communications in Computer and Information Science. Vol. 1618 / ed. by L. Sokolinsky, M. Zymbler. — Cham, 2022. — P. 43–55. — DOI: 10.1007/978-3-031-11623-0\_4.
54. Donahue J. [et al.]. Long-term recurrent convolutional networks for visual recognition and description // Proceedings of the IEEE conference on computer vision and pattern recognition. — 2015. — P. 2625–2634. — DOI: 10.1109/tpami.2016.2599174.

55. Eliazar I. Black swans and dragon kings: A unified model // EPL (Europhysics Letters). — 2017. — Vol. 119(6). — DOI: 10.1209/0295-5075/119/60007.
56. Essien A., Giannetti C. A deep learning framework for univariate time series prediction using convolutional LSTM stacked autoencoders // 2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA). — IEEE. 2019. — P. 1–6. — DOI: 10.1109/INISTA.2019.8778417.
57. Fan S., Hyndman R. J. Short-term load forecasting based on a semi-parametric additive model // IEEE transactions on power systems. — 2011. — Vol. 27, no. 1. — P. 134–141. — DOI: 10.1109/TPWRS.2011.2162082.
58. Galicia A. [et al.]. Multi-step forecasting for big data time series based on ensemble learning // Knowledge-Based Systems. — 2019. — Jan. — Vol. 163. — P. 830–841. — DOI: 10.1016/j.knosys.2018.10.009.
59. Gardner Jr E. S., McKenzie E. Forecasting trends in time series // Management science. — 1985. — Vol. 31, no. 10. — P. 1237–1246. — DOI: 10.1287/mnsc.31.10.1237.
60. Gonzales Martinez R. Apocalypse unleashed: a critical perspective on complexity science, catastrophes, and black swan events in international business // Critical Perspectives on International Business. — 2023. — Vol. 20, no. 1. — P. 94–120. — DOI: 10.1108/cpoib-03-2022-0014.
61. Hamami F., Dahlan I. A. Univariate time series data forecasting of air pollution using LSTM neural network // 2020 International Conference on Advancement in Data Science, E-learning and Infor-

- mation Systems (ICADEIS). — IEEE. 2020. — P. 1–5. — DOI: 10.1109/ICADEIS49811.2020.9277393.
62. Hillmer S. C., Tiao G. C. An ARIMA-model-based approach to seasonal adjustment // Journal of the American Statistical Association. — 1982. — Vol. 77, no. 377. — P. 63–70. — DOI: 10.2307/2287770.
63. Hochreiter S., Schmidhuber J. Long short-term memory // Neural computation. — 1997. — Vol. 9, no. 8. — P. 1735–1780. — DOI: 10.1162/neco.1997.9.8.1735.
64. Holt C. C. Forecasting seasonals and trends by exponentially weighted moving averages // International journal of forecasting. — 2004. — Vol. 20, no. 1. — P. 5–10. — DOI: 10.1016/j.ijforecast.2003.09.015.
65. Hussain N., Li B. Using R-studio to examine the COVID-19 Patients in Pakistan Implementation of SIR Model on Cases // Int. J. Sci. Res. in Multidisciplinary Studies Vol. — 2020. — Vol. 6, no. 8. — DOI: 10.13140/RG.2.2.32580.04482.
66. Hyndman R. J., Athanasopoulos G. Forecasting: principles and practice. — OTexts, 2018. — URL: <https://otexts.org/fpp2/>.
67. Hyndman R. J., Fan S. Density forecasting for long-term peak electricity demand // IEEE Transactions on Power Systems. — 2009. — Vol. 25, no. 2. — P. 1142–1153. — DOI: 10.1109/TPWRS.2009.2036017.
68. Hyndman R. J., Koehler A. B. Another look at measures of forecast accuracy // International journal of forecasting. — 2006. — Vol. 22, no. 4. — P. 679–688. — DOI: 10.1016/j.ijforecast.2006.03.001.

69. İŞEN B. Wind Turbine SCADA Dataset. — 2020. — URL: <https://www.kaggle.com/datasets/berkerisen/wind-turbine-scada-dataset> (visited on 05/25/2024).
70. Jovanovic A. [et al.]. Metaheuristic Optimized BiLSTM Univariate Time Series Forecasting of Gold Prices // International Conference on Data Science and Applications. — Springer. 2023. — P. 221–235. — DOI: 10.1007/978-981-99-7862-5\_17.
71. Kamel Boulos M. N., Geraghty E. M. Geographical tracking and mapping of coronavirus disease COVID-19/severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) epidemic and associated events around the world: how 21st century GIS technologies are supporting the global fight against outbreaks and epidemics. — 2020. — DOI: 10.1186/s12942-020-00202-8.
72. Khan S. D., Alarabi L., Basalamah S. Toward smart lockdown: a novel approach for COVID-19 hotspots prediction using a deep hybrid neural network // Computers. — 2020. — Vol. 9, no. 4. — P. 99. — DOI: 10.3390/computers9040099.
73. Khashei M., Bijari M. A novel hybridization of artificial neural networks and ARIMA models for time series forecasting // Applied soft computing. — 2011. — Vol. 11, no. 2. — P. 2664–2675. — DOI: 10.1016/j.asoc.2010.10.015.
74. Kim H., Park S., Kim S. Time-series clustering and forecasting household electricity demand using smart meter data // Energy Reports. — 2023. — Vol. 9. — P. 4111–4121. — DOI: 10.1016/j.egy.2023.03.042.
75. Kumar M. [et al.]. Spreading of COVID-19 in India, Italy, Japan, Spain, UK, US: a prediction using ARIMA and LSTM model //

- Digital Government: Research and Practice. — 2020. — Vol. 1, no. 4. — P. 1–9. — DOI: 10.1145/3411760.
76. Liu J. [et al.]. Novel production prediction model of gasoline production processes for energy saving and economic increasing based on AM-GRU integrating the UMAP algorithm // Energy. — 2023. — Vol. 262. — P. 125536. — DOI: 10.1016/j.energy.2022.125536.
77. Liu Y. [et al.]. Learning natural language inference using bidirectional LSTM model and inner-attention // arXiv preprint arXiv:1605.09090. — 2016. — DOI: 10.48550/arXiv.1605.09090.
78. Loris I. [et al.]. Accelerating gradient projection methods for  $\ell_1$ -constrained signal recovery by steplength selection rules // Applied and Computational Harmonic Analysis. — 2009. — Vol. 27, no. 2. — P. 247–254. — DOI: 10.1016/j.acha.2009.02.003.
79. Luo J., Gong Y. Air pollutant prediction based on ARIMA-WOA-LSTM model // Atmospheric Pollution Research. — 2023. — Vol. 14, no. 6. — P. 101761. — DOI: 10.1016/j.apr.2023.101761.
80. Luo Y., Duraiswami R. Efficient Parallel Nonnegative Least Squares on Multicore Architectures // SIAM Journal on Scientific Computing. — 2011. — Vol. 33, no. 5. — P. 2848–2863. — DOI: 10.1137/100799083.
81. Ma G., Zhang Y., Liu M. A generalized gradient projection method based on a new working set for minimax optimization problems with inequality constraints // Journal of inequalities and applications. — 2017. — Vol. 2017, no. 1. — P. 1–14. — DOI: 10.1186/s13660-017-1321-3.
82. Makarovskikh T., Panyukov A., Abotaleb M. Generalized Least Deviation Method for Identification of Quasi-linear Autoregressive

- Model. — 2022. — URL: <https://github.com/tmakarovskikh/GLDMPredictor.git> (visited on 07/27/2022).
83. Makarovskikh T. [et al.]. Forecasting of crop yields using the quasi-linear equation of n-factor autoregression // Prikladnaya informatika = Journal of Applied Informatics. — 2023. — Vol. 18, no. 6. — P. 5–19. — (in Russian).
  84. Makarovskikh T., Abotaleb M. Comparison between two systems for forecasting Covid-19 infected cases // Computer Science Protecting Human Society Against Epidemics: First IFIP TC 5 International Conference, ANTICOVID 2021, Virtual Event, June 28–29, 2021, Revised Selected Papers 1. — Springer. 2021. — P. 107–114. — DOI: 10.1007/978-3-030-86582-5\_10.
  85. Makarovskikh T., Abotaleb M. Computational experiments for winter wheat. — 2023. — URL: [https://github.com/tmakarovskikh/GLDMPredictor/blob/main/Stavropol\\_winter-wheat-2012.xls](https://github.com/tmakarovskikh/GLDMPredictor/blob/main/Stavropol_winter-wheat-2012.xls) (visited on 06/22/2023).
  86. Makarovskikh T., Abotaleb M. DL-System. — 04/2022. — URL: <https://github.com/abotalebmostafa11/DL-System> ; Accessed on 2022-April.
  87. Makarovskikh T., Abotaleb M. Two Systems for Forecasting COVID-19. — 05/2021. — URL: <https://github.com/abotalebmostafa11/2-systems-for-forecasting-covid-19> ; Accessed on 2021-May-22.
  88. Makarovskikh T., Abotaleb M., Panyukov A. Investigation of Parallelism Possibilities for Forecasting Using Quasilinear Recurrence Equation // Communications in Computer and Information Science. — 2022. — Apr. — P. 49–62. — DOI: 10.14529/pct2022.

89. Makarovskikh T., Panyukov A., Abotaleb M. Using General Least Deviations Method for Forecasting of Crops Yields // International Conference on Mathematical Optimization Theory and Operations Research. — Springer. 2023. — P. 376–390. — DOI: 10.1007/978-3-031-43257-6\_28.
90. Maksimova V. N. The use of geoinformation technologies for the rational use of agricultural land (practical aspect) // Investment Management and Innovations. — 2018. — No. 2. — DOI: 10.14529/iimj180210.
91. Maleki A. [et al.]. Comparison of ARIMA and NNAR models for forecasting water treatment plant's influent characteristics // KSCE Journal of Civil Engineering. — 2018. — Vol. 22. — P. 3233–3245. — DOI: 10.1007/s12205-018-1195-z.
92. Neto A. B. S. [et al.]. Studying the Performance of Cognitive Models in Time Series Forecasting // Revista de Informática Teórica e Aplicada. — 2020. — Vol. 27, no. 1. — P. 83–91. — DOI: 10.22456/2175-2745.96181.
93. Newbold P. ARIMA model building and the time series analysis approach to forecasting // Journal of forecasting. — 1983. — Vol. 2, no. 1. — P. 23–35. — DOI: 10.1002/for.3980020104.
94. Oh J., Seong B. Forecasting with a combined model of ETS and ARIMA // Communications for Statistical Applications and Methods. — 2024. — Vol. 31, no. 1. — P. 143–154. — DOI: 10.29220/CSAM.2024.31.1.143.
95. Okasha M. K. Using support vector machines in financial time series forecasting // International Journal of Statistics and Applica-

- tions. — 2014. — Vol. 4, no. 1. — P. 28–39. — DOI: 10.5923/j.statistics.20140401.03.
96. Oo Z. Z., Phyu S. Time series prediction based on Facebook Prophet: a case study, temperature forecasting in Myintkyina // International Journal of Applied Mathematics Electronics and Computers. — 2020. — Vol. 8, no. 4. — P. 263–267. — DOI: 10.18100/ijamec.816894.
97. Ozdemir O., Yozgatligil C. Forecasting performance of machine learning, time series, and hybrid methods for low-and high-frequency time series // Statistica Neerlandica. — 2024. — Vol. 78, no. 2. — P. 441–474. — DOI: 10.1111/stan.12326.
98. Panchal R., Kumar B. Forecasting industrial electric power consumption using regression based predictive model // Recent Trends in Communication and Electronics. — CRC Press, 2021. — P. 135–139. — DOI: 10.1201/9781003193838.
99. Panyukov A. V., Tyrsin A. N. Stable parametric identification of vibratory diagnostics objects. // Journal of Vibroengineering. — 2008. — Vol. 10, no. 2. — URL: <https://www.extrica.com/article/10181>.
100. Panyukov A. V., Mezaal Y. A. Parametric identification of quasilinear difference equation // Vestnik Yuzhno-Ural'skogo Gosudarstvennogo Universiteta. Seriya " Matematika. Mekhanika. Fizika". — 2019. — Vol. 11, no. 4. — P. 32–38.
101. Panyukov A., Makarovskikh T., Abotaleb M. Forecasting with Using Quasilinear Recurrence Equation // International Conference on Optimization and Applications. — Springer. 2022. — P. 183–195. — DOI: 10.1007/978-3-031-22990-9\_13.

102. Panyukov A., Mezaal Ya. A. Stable Identification of Linear Autoregressive Model with Exogenous Variables on the Basis of the Generalized Least Absolute Deviation Method // Bulletin of the South Ural State University, Series: Mathematical Modelling, Programming and Computer Software. — 2018. — Feb. — Vol. 11. — P. 35–43. — DOI: 10.14529/mmp180104.
103. Panyukov A. V., Mezaal Y. A. Improving of the Identification Algorithm for a Quasilinear Recurrence Equation // International Conference on Optimization and Applications. — Springer. 2020. — P. 15–26. — DOI: 10.1007/978-3-030-65739-0\_2.
104. Panyukov A. V., Mezaal Y. A. Stable estimation of autoregressive model parameters with exogenous variables on the basis of the generalized least absolute deviation method // IFAC-PapersOnLine. — 2018. — Vol. 51, no. 11. — P. 1666–1669. — DOI: 10.1016/j.ifacol.2018.08.217.
105. Panyukov A. V. Scalability of Algorithms for Arithmetic Operations in Radix Notation // Reliable Computing. — 2015. — Vol. 19, no. 4. — P. 417–434. — URL: <https://api.semanticscholar.org/CorpusID:17944869>.
106. Papacharalampous G., Tyrallis H., Koutsoyiannis D. Predictability of monthly temperature and precipitation using automatic time series forecasting methods // Acta Geophysica. — 2018. — Vol. 66. — P. 807–831. — DOI: 10.1007/s11600-018-0120-7.
107. Pełka P. Analysis and forecasting of monthly electricity demand time series using pattern-based statistical methods // Energies. — 2023. — Vol. 16, no. 2. — P. 827. — DOI: 10.3390/en16020827.

108. Perone G. Comparison of ARIMA, ETS, NNAR, TBATS and hybrid models to forecast the second wave of COVID-19 hospitalizations in Italy // *The European Journal of Health Economics*. — 2021. — P. 1–24. — DOI: 10.1007/s10198-021-01347-4.
109. Ponce M., Sandhel A. covid19. analytics: An R Package to Obtain, Analyze and Visualize Data from the Coronavirus Disease Pandemic // *arXiv preprint arXiv:2009.01091*. — 2020. — DOI: 10.48550/arXiv.2009.01091.
110. Rahimi Z., Khashei M. A least squares-based parallel hybridization of statistical and intelligent models for time series forecasting // *Computers & Industrial Engineering*. — 2018. — Apr. — Vol. 118. — P. 44–53. — DOI: 10.1016/j.cie.2018.02.023.
111. Ren G. [et al.]. Overview of wind power intermittency: Impacts, measurements, and mitigation solutions // *Applied Energy*. — 2017. — Vol. 204. — P. 47–65. — DOI: 10.1016/j.apenergy.2017.06.098.
112. Saeed W. Frequency-based ensemble forecasting model for time series forecasting // *Computational and Applied Mathematics*. — 2022. — Vol. 41, no. 2. — P. 66. — DOI: 10.1007/s40314-022-01765-x.
113. Safi S. K., Sanusi O. I. A hybrid of artificial neural network, exponential smoothing, and ARIMA models for COVID-19 time series forecasting // *Model Assisted Statistics and Applications*. — 2021. — Vol. 16, no. 1. — P. 25–35. — DOI: 10.3233/MAS-210512.
114. Sameh B. [et al.]. IoT-Based System for Crop Forecasting: Design and Implementation // *2023 3rd International Conference on Elec-*

- tronic Engineering (ICEEM). — IEEE. 2023. — P. 1–7. — DOI: 10.1109/ICEEM58740.2023.10319527.
115. Schuster M., Paliwal K. Networks bidirectional recurrent neural // IEEE Trans Signal Proces. — 1997. — Vol. 45. — P. 2673–2681. — DOI: 10.1109/78.650093.
116. Shcherbakov M. V. [et al.]. A survey of forecast error measures // World applied sciences journal. — 2013. — Vol. 24, no. 24. — P. 171–176. — DOI: 10.5829/idosi.wasj.2013.24.itmies.80032.
117. Shin K.-S., Lee T. S., Kim H.-j. An application of support vector machines in bankruptcy prediction model // Expert systems with applications. — 2005. — Vol. 28, no. 1. — P. 127–135. — DOI: 10.1016/j.eswa.2004.08.009.
118. Sirotin D. Neural network approach to forecasting the cost of ferroalloy products // Izvestiya. Ferrous Metallurgy. — 2020. — Vol. 63, no. 1. — P. 78–83. — DOI: 10.17073/0368-0797-2020-1-78-83. — (In Russ.)
119. Sun D. [et al.]. Modeling and forecasting the spread tendency of the COVID-19 in China // Advances in Difference Equations. — 2020. — Vol. 2020, no. 1. — P. 1–16. — DOI: 10.1186/s13662-020-02940-2.
120. Tarmanini Ch. [et al.]. Short term load forecasting based on ARIMA and ANN approaches // Energy Reports. — 2023. — Vol. 9. — P. 550–557. — DOI: 10.1016/j.egyr.2023.01.060. — 2022 The 3rd International Conference on Power, Energy and Electrical Engineering.

121. Tatiana Makarovskikh, Anatoly Panyukov, Mostafa Abotaleb. Monitoring and Forecasting Crop Yields // International Conference on Parallel Computational Technologies. — Springer. 2023. — P. 78–92. — DOI: 10.1007/978-3-031-38864-4\_6.
122. Taylor S. J., Letham B. Forecasting at scale // The American Statistician. — 2018. — Vol. 72, no. 1. — P. 37–45. — DOI: 10.1080/00031305.2017.1380080.
123. Valeras A. S. COVID-19: Complexity and the Black Swan // Families Systems & Health. — 2020. — Vol. 38(2). — P. 221–223. — DOI: 10.1037/fsh0000486.
124. Wang H. [et al.]. Predicting daily streamflow with a novel multi-regime switching ARIMA-MS-GARCH model // Journal of Hydrology: Regional Studies. — 2023. — Vol. 47. — P. 101374. — DOI: 10.1016/j.ejrh.2023.101374.
125. Wang Y. [et al.]. Attention-based LSTM for aspect-level sentiment classification // Proceedings of the 2016 conference on empirical methods in natural language processing. — 2016. — P. 606–615. — DOI: 10.18653/v1/D16-1058.
126. Widén J. Correlations between large-scale solar and wind power in a future scenario for Sweden // IEEE transactions on sustainable energy. — 2011. — Vol. 2, no. 2. — P. 177–184. — DOI: 10.1109/TSTE.2010.2101620.
127. Wood D. A. Weeks-ahead epidemiological predictions of varicella cases from univariate time series data applying artificial intelligence // Infectious Diseases & Immunity. — 2024. — Vol. 4, no. 01. — P. 25–34. — DOI: 10.1097/ID9.0000000000000096.

128. Yakubova D. Econometric models of development and forecasting of black metallurgy of Uzbekistan // Asian Journal of Multidimensional Research (AJMR). — 2019. — Vol. 8, no. 5. — P. 310–314. — DOI: 10.5958/2278-4853.2019.00205.2.
129. Yamacli D. S., Yamacli S. Estimation of the unemployment rate in Turkey: A comparison of the ARIMA and machine learning models including Covid-19 pandemic periods // Heliyon. — 2023. — Vol. 9, no. 1. — e12796. — DOI: 10.1016/j.heliyon.2023.e12796.
130. Yule G. U. VII. On a method of investigating periodicities disturbed series, with special reference to Wolfer's sunspot numbers // Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character. — 1927. — Vol. 226, no. 636–646. — P. 267–298. — DOI: 10.1098/rsta.1927.0007.
131. Zain Z. M., Alturki N. M. COVID-19 pandemic forecasting using CNN-LSTM: a hybrid approach // Journal of Control Science and Engineering. — 2021. — Vol. 2021. — P. 1–23. — DOI: 10.1155/2021/8785636.
132. Zeng J., Ma X., Zhou K. Enhancing attention-based LSTM with position context for aspect-level sentiment classification // IEEE Access. — 2019. — Vol. 7. — P. 20462–20471. — DOI: 10.1109/ACCESS.2019.2893806.
133. Zhang G. P. Time series forecasting using a hybrid ARIMA and neural network model // Neurocomputing. — 2003. — Vol. 50. — P. 159–175. — DOI: 10.1016/S0925-2312(01)00702-0.

134. Zhang Y. [et al.]. A novel extreme adaptive GRU for multivariate time series forecasting // Scientific Reports. — 2024. — Vol. 14, no. 1. — P. 2991. — DOI: 10.1038/s41598-024-53460-y.
135. Abotaleb M. GLDM. — 11.2024. — URL: <https://gitverse.ru/mostafa/GLDM> ; Accessed: December 2024.
136. Panyukov A. V. [и др.]. Stable identification of linear autoregressive model with exogenous variables on the basis of the generalized least absolute deviation method // Вестник Южно-Уральского государственного университета. Серия: Математическое моделирование и программирование. — 2018. — Т. 11, № 1. — С. 35—43.