

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ
Н.Г.ЧЕРНЫШЕВСКОГО»**

Кафедра математического и компьютерного моделирования

Разработка программного обеспечения для аппроксимации

вейвлет преобразованиями

АВТОРЕФЕРАТ МАГИСТЕРСКОЙ РАБОТЫ

Студентки 2 курса 247 группы
направления 09.03.03 — Прикладная информатика

механико-математического факультета

Волжаниной Анастасии Викторовны

Научный руководитель
профессор, д.ф.-м.н., доцент

Д.В. Кондратов

Заведующий кафедрой
зав. каф., д.ф.-м.н., доцент

Ю.А. Блинков

Саратов 2023

Введение. В настоящее время данные используются во многих областях жизни. Каждый день обрабатывается терабайты данных по всему миру. Анализ потоковых данных в реальном времени представляет собой большую проблему для людей в настоящее время. Данных оказывается больше чем мощностей, которые могут их обработать. Зачастую такие данные представляют в виде временных рядов. Временные ряды встречаются в повседневной жизни многих людей, например: информация о погоде, данные акции различных компаний и т.д. Обработка временных рядов данных и их анализ в реальном времени необходимы для функционирования различных приложений.

Различные методы сжатия могут быть использованы для сокращения времени обработки больших массивов данных и снижения нагрузки путем эффективной экономии пространства для хранения и пропускной способности сети.

Методы сжатия в основном могут быть в виде сжатия без потерь и сжатия с потерями. Для больших массивов данных временных рядов некоторая потеря менее важной информации может привести к значительному уменьшению размеру данных, в результате чего это может привести к снижению стоимости передачи и хранения.

Тем не менее, менее точные данные могут повлиять на дальнейший анализ данных. Например, это может привести к снижению точности прогноза в соответствующей модели машинного обучения по сравнению с исходными данными. Чтобы реализовать эффективную и быструю передачу данных необходимо использовать методы сжатия, в которых будет присутствовать максимально возможное сжатие с минимальной потерей данных.

В данной работе в качестве аппроксимирующей функции будет выступать аппарата вейвлетов из-за его способности хорошо работать с функциями, характеристики которых эволюционируют во времени. Вейвлеты широко применяются для исследования нестационарных сигналов и временных рядов, для распознавания образов, а также при решении различных задач в радиотехнике, ядерной физике, биологии, экономике и многих других областях науки и техники.

Вместе с тем следует отметить, что при аппроксимации массивов данных техника применения вейвлетов разработана значительно менее подроб-

но, нежели традиционные методы. Указанные обстоятельства объясняют актуальность настоящей выпускной квалификационной работы, посвящённой аппроксимации с помощью вейвлетов.

Цель настоящей работы – создание программного обеспечения для обработки и аппроксимации данных с помощью вейвлет преобразований.

Для достижения поставленной цели необходимо решить несколько задач:

1. изучить необходимые теоретические основы;
2. проанализировать технические требования;
3. разработать программный продукт.

Магистерская работа состоит из 5 основных разделов:

1. введение;
2. описание теории и предметной области;
3. анализ технических средств и требований;
4. проектирование и разработка программного обеспечения;
5. заключение.

Во **введении** формулируется цель работы и решаемые задачи.

Первый раздел посвящён описанию основных теоретических основ вейвлет преобразований и статистике.

Впервые принцип вейвлет-анализа был рассмотрен в работе Гроссмана и Морле в 1984 году. Впоследствии теория и практическая часть дополнялись в различных книгах и утнях: И.Добечи в 1992, Койфман в 1992, Фостер в 1996, Скаргл в 1997. С тех пор вейвлет-анализ стал одним из популярных разделов математики.

С точки зрения теории аппроксимации гармонического анализа, теория вейвлетов важна по многим параметрам. Недавним развитием теории аппроксимации является аппроксимация произвольной функции вейвлет-полиномами. Существует несколько видов вейвлетов, таких как вейвлет Хара, вейвлет Морде, вейвлет Шеннона, вейвлет Добечи, вейвлет Мейера и т.д.

Интегральным вейвлет–преобразованием функции $f(t) \in L^2(R)$ называют выражение:

$$W(a, b) = \frac{1}{|a|^{1/2}} \int_{-\infty}^{\infty} f(t) \psi^* \left(\frac{t-b}{a} \right) dt, \quad (1)$$

где $a, b \in R, a \neq 0$.

Входящая в выражение (1) функция $\psi(t)$ называется вейвлетом (анализирующим, базисным или материнским вейвлетом). Стоит отметить, что в формуле (1) символом $*$ обозначена процедура комплексного сопряжения.

Масштаб (scale) вейвлета – параметр определяющий размер вейвлета.

Его аналогом в Фурье – является период (частота) гармонического колебания. Следует сказать, что понятие масштаба – более широкое (хотя и менее наглядное), чем понятие периода. Связано это с тем, что в Фурье–преобразовании функциональный вид ядра преобразования зафиксирован раз и навсегда, в то время как вейвлет–преобразование одной и той же функции можно получить с помощью различных базисных вейвлетов (т. е. в разных системах масштабов).

Сдвиг (shift) вейвлета – параметр b задающий временную локализацию вейвлета. Этот параметр не имеет аналога в Фурье–преобразовании.

Обратное интегральное вейвлет–преобразование задается выражением

$$f(t) = C_{\psi}^{-1} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} W(a, b) \psi \left(\frac{t-b}{a} \right) \frac{1}{a^{1/2}} \frac{dad b}{a^2}, \quad (2)$$

где C_{ψ} – нормирующий коэффициент:

$$C_{\psi} = \int_{-\infty}^{\infty} |\hat{\psi}|^2 |w|^{-1} dw < \infty. \quad (3)$$

Первое свойство вейвлетов это частотно-временная локализация. Это свойство означает, что что вейвлеты $\psi(t)$ и их преобразования Фурье $\hat{\psi}(w)$ существенно отличаются от нуля лишь на малых интервалах времени и ча-

стоты и очень мало отличаются от нуля (или просто равны нулю) вне этих интервалов.

Количественной мерой локализации функции $z(t) \in L^2(R)$ могут служить ее центр $\langle t \rangle$ и радиус Δ_t :

$$\langle t \rangle = \frac{1}{\|z\|^2} \int_{-\infty}^{\infty} t |z(t)|^2 dt, \quad (4)$$

$$\Delta_t^2 = \frac{1}{\|z\|^2} \int_{-\infty}^{\infty} [t - \langle t \rangle]^2 |z(t)|^2 dt. \quad (5)$$

При этом эффективная ширина вейвлета принимается равной $2\Delta_t$

Второе свойство вейвлетов это выполнение условия для нулевого момента. Чтобы обеспечить обратимость вейвлет-преобразования, вейвлет должен удовлетворять следующему соотношению (3). Это обеспечивается условием

$$\hat{\psi}_{(0)} = 0, \quad (6)$$

откуда следует

$$\int_{-\infty}^{\infty} \psi(t) dt = 0. \quad (7)$$

Для приложений бывает важно, чтобы не только нулевой момент, но и m старших моментов были равны нулю:

$$\int_{-\infty}^{\infty} \psi(t) t^m dt = 0. \quad (8)$$

Вейвлеты, обладающие таким свойством, оказываются полезными при анализе временных рядов.

Второй раздел посвящён анализу технических средств и требований. В ней рассмотрены требования к функциональным характеристикам программы.

Программа должна обеспечивать выполнение следующих функций:

1. считывание данных из файла;
2. производить статистическую обработку данных;
3. производить аппроксимацию с помощью вейвлет преобразований;
4. визуализировать результаты;
5. сохранять полученный результат в файл.

Для реализации программного продукта был выбран высокоуровневый язык Python.

Python – это интерпретируемый, объектно-ориентированный, высокоуровневый язык программирования с динамической семантикой. Его высокоуровневые встроенные структуры данных в сочетании с динамической типизацией и динамическим связыванием делают его очень привлекательным для быстрой разработки приложений, а также для использования в качестве языка сценариев или для соединения существующих компонентов.

В работе используются следующие библиотеки:

1. qt5 – это кроссплатформенное программное обеспечение для создания графических пользовательских интерфейсов, а также кроссплатформенных приложений, которые работают на различных программных и аппаратных платформах, таких как Linux, Windows, macOS, Android; библиотека необходима для написания интерфейса программы;
2. numpy – это библиотека для языка программирования Python, добавляющая поддержку больших многомерных массивов и матриц, а также большой набор высокоуровневых математических функций для работы с этими массивами;
3. pandas – программная библиотека для обработки и анализа данных; работа pandas с данными строится поверх библиотеки NumPy, являющейся инструментом более низкого уровня; библиотека необходима для нахождения статистических данных;
4. matplotlib – библиотека для визуализации данных двумерной и трёхмерной графикой; библиотека необходима для построения графиков;
5. pywt – это бесплатная библиотека с открытым исходным кодом для вейвлет-преобразований в Python.

Третий раздел посвящён проектированию и разработке программного обеспечения.

В нем рассматриваются основы моделирования интерфейса. Рассмотрено понятие юзабилити. И то как оно влияет на создание интерфейса.

Юзабилити — это степень эффективности, продуктивности и удовлетворенности, с которыми продукт может быть использован определенными пользователями в определенном контексте использования для достижения определенных целей. Данное определение юзабилити соответствует стандарту ISO 9241-11. Но это также эргономический подход и группа принципов и методов, направленных на разработку удобных и доступных продуктов, основанных на дизайне, ориентированном на пользователя.

Основные принципы юзабилити:

1. простота обучения — система должна быть интуитивно понятной, т.е. простой в использовании, позволяющей даже неопытному пользователю удовлетворительно работать с ней;
2. эффективность использования — система должна иметь эффективное исполнение, позволяющее обеспечить высокую производительность, т.е. ресурсы, затрачиваемые на достижение целей должны быть минимальными;
3. последовательность — сходные задачи выполняются одинаково;
4. визуальная ясность — отображаемая информация должна быть прочитана быстро и легко, не вызывая путаницы;
5. приоритизация функциональности — наиболее важные функциональные возможности легко доступны для пользователей;
6. приоритизация информации — наиболее важная информация легко доступна для пользователей и т.д.

Для создания графического интерфейса программы используется библиотека PyQt5 и её дополнительные инструменты — в частности, утилита QtDesigner, позволяющая быстро создать интерфейс, и pyuic5 для конвертирования ui файлов, генерируемых QtDesigner, в формат py для работы с его кодом в дальнейшем.

В качестве экспериментальных данных (временного ряда) выступают данные об акциях компании «Сбербанк» за период с декабря 2022 по май 2023 года.

Временной ряд выглядит в соответствии с рисунком 1.



Рисунок 1 — Временной ряд

Пример аппроксимации с помощью вейвлета Морле созданным алгоритмом представлено в соответствии с рисунком 2.

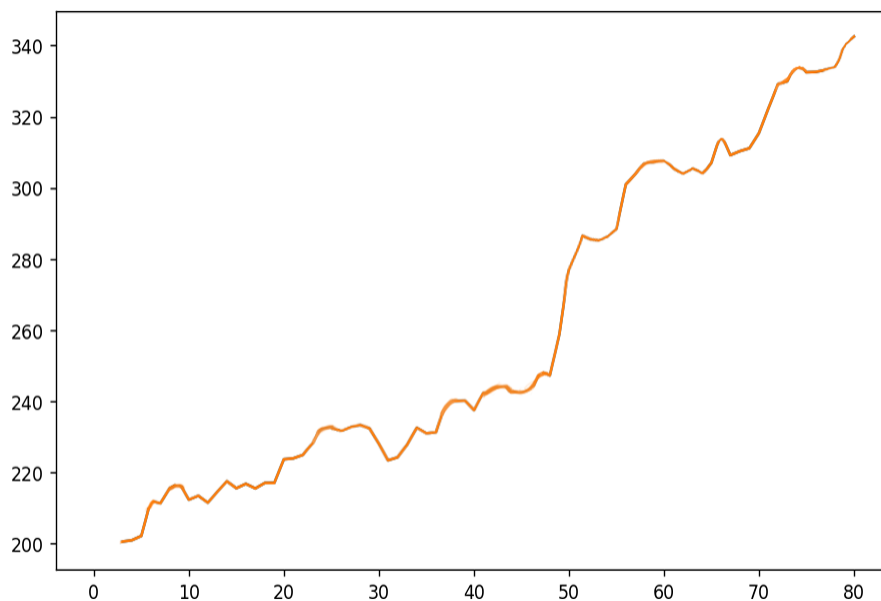


Рисунок 2 — График аппроксимации с помощью вейвлета Морле

Пример аппроксимации с помощью вейвлета Хаара созданным алгоритмом представлено в соответствии с рисунком 3.

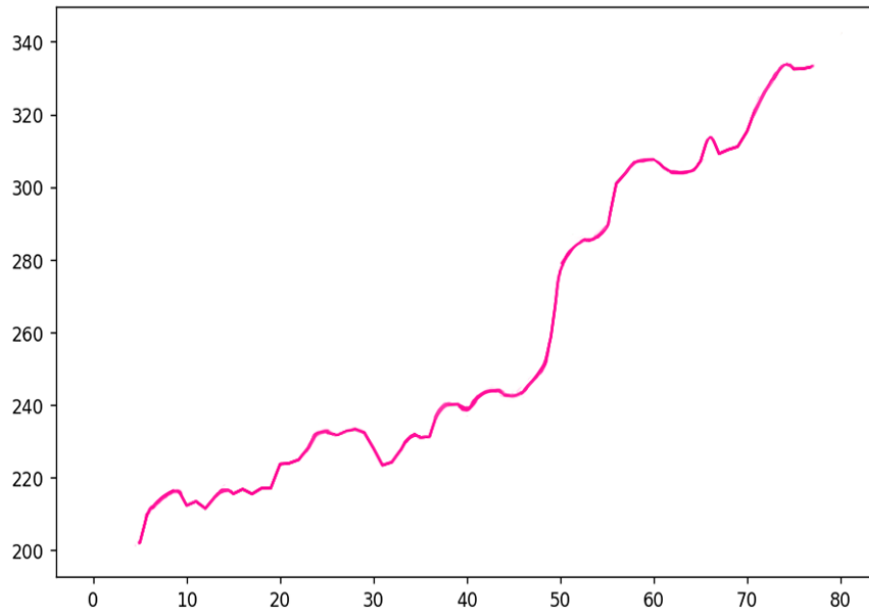


Рисунок 3 — График аппроксимации с помощью вейвлета Морле

Конечный результат работы программы представлен в соответствии с рисунком 4.

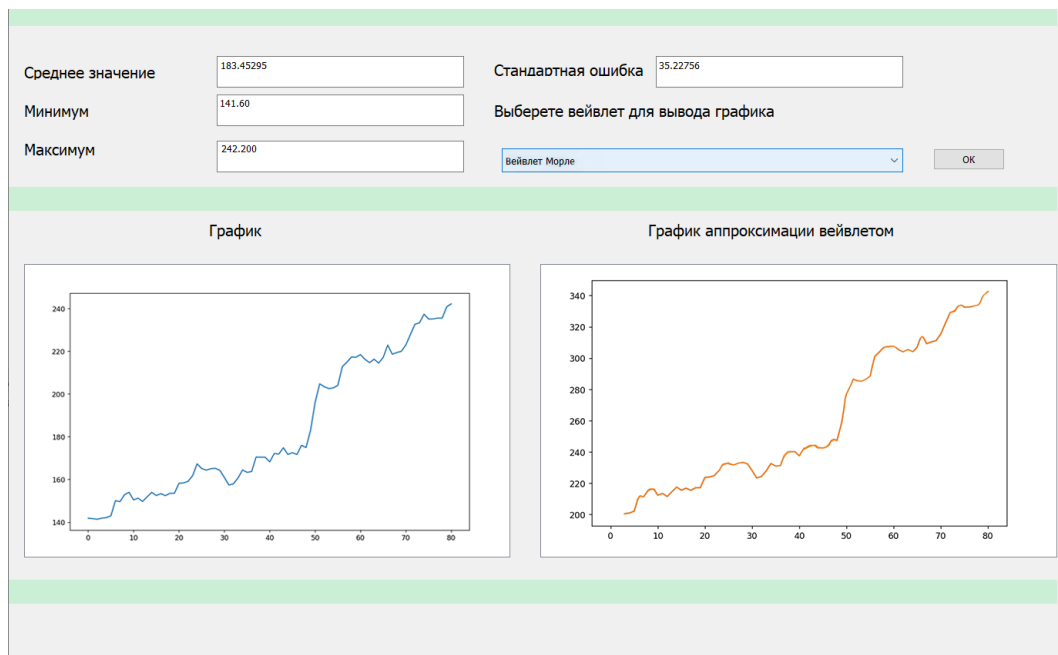


Рисунок 4 — Результат работы программы

Все полученные данные для дальнейшего использования сохраняются в файл.

Для того что бы оценить результаты работы программы по аппроксимации вейвлетами воспользуемся оценкой средней ошибкой аппроксимации (MAPE), средней абсолютной ошибки (RMSE), а так же визуальной оценкой.

Визуализация аппроксимации данных с помощью вейвлета Морле в сравнении с исходным временным рядом выглядит в соответствии с рисунком 5.

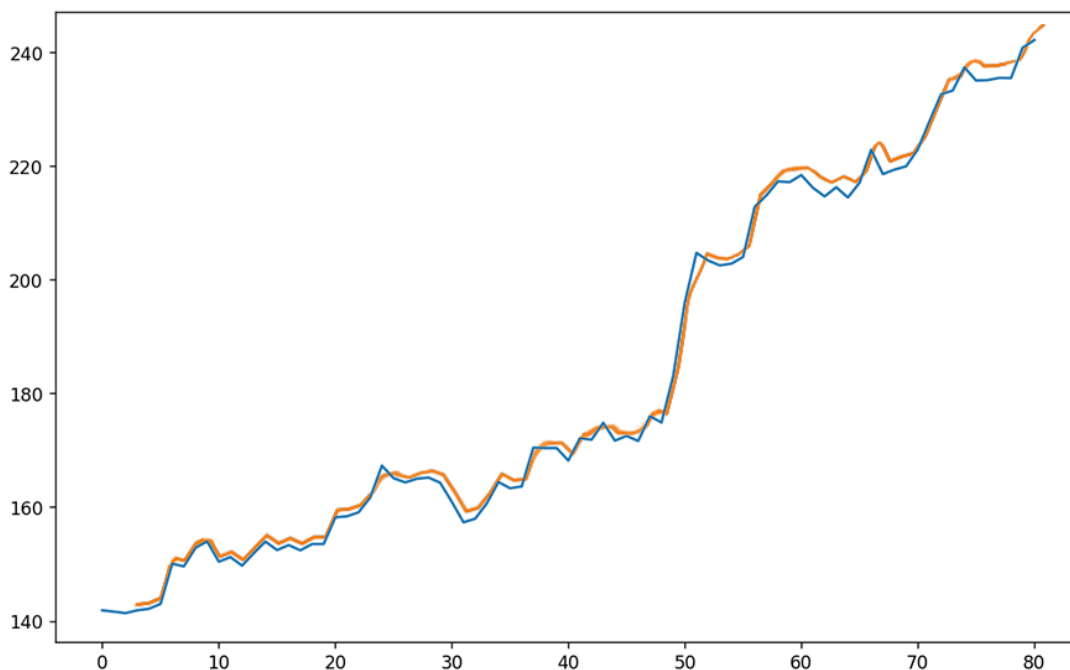


Рисунок 5 — График аппроксимации исходных данных (синяя линия) вейвлетом Морле (оранжевая линия)

Средняя ошибка аппроксимации (MAPE) - среднее отклонение аппроксимированных данных от значений фактических.

Находится по следующей формуле:

$$MAPE = \frac{1}{n} \sum \left| \frac{y - \hat{y}}{y} \right|. \quad (9)$$

Допустимый предел значений средней ошибки аппроксимации не более 8-10%.

В соответствии с таблицей 1 представленные результаты проверки с помощью MAPE.

Таблица 1 — Оценка средней ошибкой аппроксимации

Вейвлет:	MAPE:
Вейвлет Морле	0.00024461
Вейвлет МНАТ	0.00731932
Вейвлет Хаара	0.00146722
Вейвлет Добеши	0.01929399

В соответствии с таблицей 2 представленные результаты проверки с помощью RMSE.

Таблица 2 — Оценка RMSE

Вейвлет:	RMSE:
Вейвлет Морле	18.19388
Вейвлет МНАТ	25.72788
Вейвлет Хаара	21.56332
Вейвлет Добеши	30.55321

Как видно все вейвлеты аппроксимируют данные довольно хорошо, т.к отклонения довольно не значительные и соответствуют допустимым значениям. Соответственно можно сделать вывод, что программа работает корректно.

Заключение. Целью данной работы было создание программного обеспечения для обработки и анализа данных, которая позволяет аппроксимировать данные с помощью вейвлет преобразований.

В теоретической части работы были рассмотрены основы теоретические основы вейвлет-анализа. Рассмотрены отличия непрерывных и дискретных вейвлетов, их свойства и т.д.

В практической части были рассмотрены технические требования и средства необходимые для написания программы. Была спроектирована и разработана программа на языке программирование Python. С помощью данной программы можно с лёгкостью производить аппроксимацию данных с помощью вейвлет функций. Все полученные данные сохраняются в отдельный файл для дальнейшей работы с ними.

По данной теме существует очень мало исследований в данный момент и аналоги программы либо отсутствуют либо не распространяются в свободном доступе. Необходимо развивать программное обеспечение т.к. вейвлет функции являются удобным и быстрым способом аппроксимации данных.

Разработанная программа устойчиво выполняет свои функции, работая с различными данными. Но также есть возможность дальнейшего улучшения работоспособности и расширение функционала. Например, разработка подгрузки данных в реальном времени, что возможно сделать благодаря возможности языка к модульности и расширяемости.