# Аппроксимация Паде ядра свертки решения волнового уравнения в канале с помощью нейронной сети

Родионов Данила Олегович

Механико-математический факультет МГУ, кафедра вычислительной механики

/04/2023

# Содержание

# Введение

Среди большого количества методов построения искусственных граничных условий на открытых границах особое место занимают так называемые прозрачные граничные условия (ПГУ), основанные на преобразовании Лапласа по времени и аппроксимация ядра свертки полученного точного граничного условия суммой экспонент. В пространстве изображений преобразования Лапласа ей соответствует сумма полюсов, которая после простейших алгебраических преобразований обращается в рациональную функцию. Одним из алгоритмов подбора коэффициентов в данной задаче может быть решение системы линейных уравнений, однако по ряду причин этот метод не всегда является оптимальным. Альтернативой является создание архитектуры нейронной сети, реализующей заданную функцию, суть обучения которой - поиск параметров для минимизации некоторого функционала

# Аналитическая постановка задачи

Рассмотрим задачу о распространении волн в бесконечном канале кругового сечения радиуса $a$. Будем считать, что ось канала совпадает с осью $z$ цилиндрической системы координат. Будем искать численное решение задачи в конечной подобласти $\Omega$ этого канала, которая ограничена искусственными границами - плоскостями $z = z_L$ и $z = z_R$

# Волновое уравнение

Движение описывается волновым уравнением:

$$\frac{1}{c^2}\frac{\partial^2 w}{\partial t^2} = \Delta w$$

где c - скорость звука среды, t - время, $\Delta$ - оператор Лапласа
Граничные условия при $r = a$:

$$\alpha\frac{\partial w}{\partial r} + \beta w = 0, |\alpha| + |\beta| \neq 0$$

На границах $z = z_L$ и $z = z_R$ требуется поставить такие граничные условия, чтобы решение задачи в $\Omega$ совпадало с решением во всей неограниченной области.

# Получение точных граничных условий

Рассмотрим разложение решения $w$ по собственным функциям оператора Лапласа в круге радиуса $a$:

$$w(r,\varphi,z,t) = \sum_{n=0}^{\infty} \sum_{k=1}^{\infty} \sum_{m=1}^{2} \psi_{k,n,m}(r,\varphi) w_{k,n,m}(z,t)$$

При этом выполняется соотношение: $\Delta\psi_{k,n,m} + \lambda_{k,n}\psi_{k,n,m} = 0$
Подставим разложение по собственным функциям Лапласа функции $w$ в исходное волновое уравнение. Тогда получим следующее соотношение для коэффициентов Фурье:

$$\frac{1}{c^2}\frac{\partial^2 w_{k,m,n}}{\partial t^2} = \frac{\partial^2 w_{k,m,n}}{\partial z^2} - \lambda_{k,n} w_{k,m,n} \text{ (a)}$$

Введем замену координат:

$$\bar{t} = c\lambda_{k,n} t$$
$$\bar{z} = \lambda_{k,n} z$$

Тогда уравнение (a) перейдет в уравнение вида:

$$\ddot{w}_{k,m,n} = w''_{k,m,n} - w_{k,m,n}$$

Исходя из этого будем строить ПГУ для уравнения: $\ddot{u} = u'' - u$
Буквой u здесь обозначена амплитуда гармоники $\psi_{k,n,m}$
После применения преобразования Лапласа это уравнение перейдет в уравнение:

$$U'' = (s^2 + 1)U$$

Его можно переписать в виде $U' = PU$, так как это и есть искомое граничное условие. Теперь необходимо определить $P$.
Подставим полученный вид функции U в уравнение второго порядка, полученное после преобразования Лапласа.

Для ограниченности решения на правой и левой границах получим $P = \pm\sqrt{s^2 + 1}$.

Выделим наибольшие степени по переменной $s$ функции $P$ перед тем как перейти к оригиналу:

$$P(s) = p_1 s + p_0 + F(s)$$

Для задачи с постоянными коэффициентами получим:

$$F(s) = \sqrt{s^2 + 1} - s$$

Теперь наша задача - построить аппроксимацию $F(s)$ с помощью нейронной сети. Аналитически это выражение будет являть собой сумму экспонент, которая в пространстве изображений даст сумму полюсов, которую, очевидно, можно привести к виду:

$$\tilde{F}(s) = \frac{Q_{M-1}(s)}{P_M(s)}$$

где $Q_{M-1}(s), P_M(s)$ - полиномы степеней соответствующих индексов.

# Вопрос существования решения задачи аппроксимации

Одним из методов построения аппроксимации типа Паде является следующий алгоритм, основанный на равенстве производных самой функции и её аппроксиманта:
Представим функцию $F(s)$ в виде своего ряда Тейлора $\sum_{n=0}^{\infty} c_n x^n$
Тогда

$$\frac{Q_{M-1}(x)}{P_M(x)} = \sum_{n=0}^{\infty} c_n x^n$$

Приравнивая коэффициенты при $x^k$ получаем систему линейных алгебраических уравнений на коэффициенты полиномов $P$ и $Q$ соответственно.

Альтернативным методом является поиск минимума некоторого функционала относительно набора параметров.
Пусть $\mathbf{a}, \mathbf{b}$ - векторы коэффициентов полиномов. Тогда задачу можно сформулировать следующим образом:

$$\sum_{j=0}^{M} b_j x^j > 0$$
$$\inf \max_{x \in I} \left| F(x) - \frac{\sum_{i=0}^{M-1} a_i x^i}{\sum_{j=0}^{M} b_j x^j} \right|$$
$$\forall x \in I \subset \mathbf{R},\ \mathbf{a} \in \mathbf{R}^{M-1},\ \mathbf{b} \in \mathbf{R}^{M}$$

# Решение задачи оптимизации

Таким образом, для решения задачи оптимизации нам понадобится следующее утверждение из выпуклого анализа

**Лемма([7], лемма 1.4)**
Пусть $A$ - некоторое множество, и $f(x, y)$ - выпуклая по $x$ функция $\forall y \in A$. Тогда функция

$$g(x) = \sup_{y \in A} f(x, y)$$

является выпуклой по $x$ с областью определения

$$\text{dom } g = \{x | (x, y) \in \text{dom } (f) \ \forall y \in A \text{ и } \sup_{y \in A} f(x, y) < \infty\}$$

# Аналитическое обоснование существования нейронной сети, реализующей рациональную аппроксимацию

Существование аппроксимации с помощью параметрических семейств функций, реализуемых нейронной сетью заданной архитектурой обусловлено теоремой, приведенной ниже

**Универсальная теорема об аппроксимации([5])**

*Пусть $\varphi$ - некоторая непрерывная сигмоидальная функция, $f(x)$ - произвольная непрерывная функция, заданная на некотором компактном в $\boldsymbol{R}^n$ множестве $K$. Тогда $\forall \epsilon > 0\ \exists$ набор векторов $w_1, ..., w_N, \theta, \alpha$ и функция $G = \sum_{i=1}^{N} \alpha_i \varphi(w_i x + \theta_i)$, такая, что $\forall x \in R^n, |G - f(x)| < \epsilon$.*

Однако в условии задано ограничение на класс функций, выступающих в роли функции активации нейронной сети. Для обобщения формулировки теоремы введем следующее

**Определение**

Функцию $g : A \to \mathbf{R}$ будем называть функцией Таубера-Виннера (**TW**), если линейные комбинации вида

$$\sum_{j=0}^{M} c_j g(\lambda_j x + \theta_j)$$

плотны в пространстве $\mathbf{C}(A)$.

содержательное описание этого класса функций даёт следующая теорема

**Теорема**([8])

Пусть $g(x) \in \mathbf{C}(\mathrm{A}) \bigcap \mathbf{S'}(\mathrm{A})$. Тогда $g(x) \in \mathbf{TW}$ если и только если $g(x)$ - не является полиномом.

Обозначенный класс функций позволяет обобщить универсальную теорему об аппроксимации в следующем виде

**Теорема([8])**

Пусть $K \subset \mathbf{R}^n$, $U \subset \mathbf{C}(K)$ - некоторые компактные множества, $g \in \mathbf{TW}$. Тогда $\forall \epsilon > 0$ найдутся такие $N, \theta_i \in \mathbf{R}, w_i \in \mathbf{R}^n$, $i = 1, ..., N$, которые не зависят от $f \in U$ и константы $c_i(f)$, зависящие от $f$, такие, что

$$|f(x) - \sum_{i=1}^{N} c_i(f) g(w_i x + \theta_i)| < \epsilon$$

для всех $x \in K, f \in U$. При этом $c_i(f)$ являются непрерывными функционалами на $U$.

# Построение модели

Для решения этой задачи воспользуемся API Keras , которая предоставляет возможность реализации полносвязной архитектуры и её дальнейшего алгоритма обучения.

Для сравнения результатов воспользуемся библиотекой scipy, которая имеет в своем арсенале метод scipy.interpolate.Pade, реализующий разложение Паде для заданной функции. Отметим при этом, что точность разложения, вообще говоря, может очень сильно меняться, так как на вход метод получает коэффициенты ряда Тейлора разложения исходной функции. Поэтому очевидно, что если использовать, например, разложение в ряд Тейлора с остаточным членом в форме Пеано с точностью до $\bar{o}(x^4)$ и $\bar{o}(x^5)$, то точность полученного разложения будет соответственно ниже и выше.

Далее спроектируем сеть для аппроксимации:
1) архитектура нейронной сети - полносвязная сеть прямого распространения
2) количество слоёв равно трем(в среднем, для задач аппроксимации достаточно двух-трех)
3) функция потерь - среднеквадратичная ошибка

$$L(y, \bar{y}) = \sqrt{\frac{\sum_{i=1}^{m} (y_i - \bar{y}_i)^2}{m}}$$

где $\bar{y}_i$ - выходное значение $i-$го нейрона
4) Норма оценки $|| \; ||_1 = \sup\limits_{i=1:N} |y_i - \bar{y}_i|$

5) Количество итераций обучения - $10^4$, размер множества для усреднения градиента batch size - 64

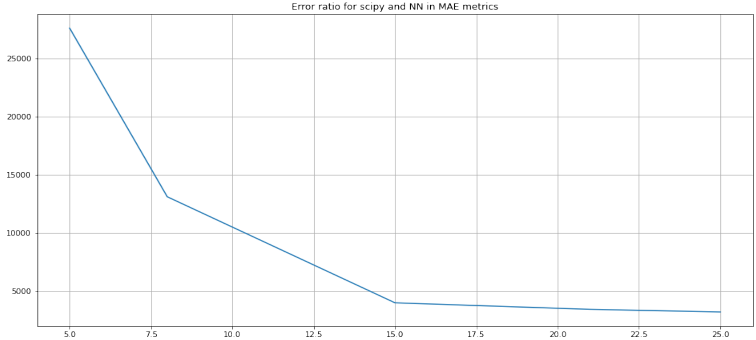
6) Данные для обучения: рассмотрим равномерную сетку на отрезке $I \in \mathbf{R}$ и значение функции $F|_{x \in I}$. В качестве входных данных будем подавать два вектора $P$ и $Q$, $k$-ая компонента которых - значение монома $x^k$ в данной точке. Также к входным данным применяется процесс нормализации и централизации.

7) Начальное распределение весов - суть коэффициентов рациональной функции - положим в виде инициализации Ксавье с нормальным распределением $w_i \sim \mathcal{N}(0, \sqrt{\frac{2}{n_{in}}})$. Однако вопрос исходного распределения коэффициентов играет центральную роль и требует дальнейшего детального анализа.

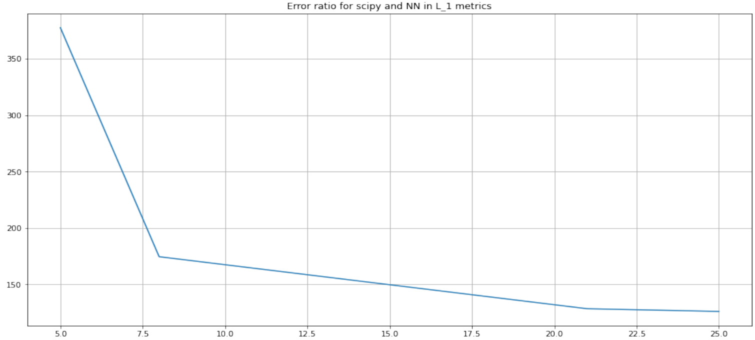
# Результаты

Ниже представлены графики зависимости отношения ошибки алгоритма scipy.Pade и результата нейронной сети от размерности входных данных:

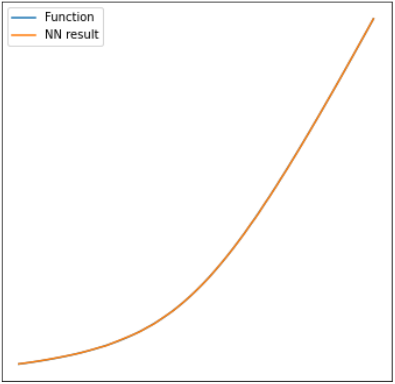
1) Метрика MAE $\sqrt{\frac{\sum_{i=1}^{m}(y_i-\bar{y}_i)^2}{m}}$

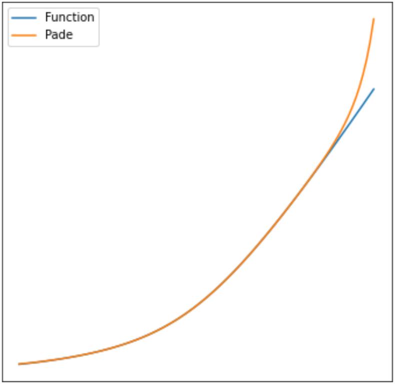


$$2)\ \text{норма}\ \|\ \|_1 = \sup_{i=1:N} |y_i - \bar{y}_i|$$



# Визуальная интерпретация результатов нейронной сети и алгоритма scipy.Pade



# Заключение

В работе продемонстрирован метод построения рациональной аппроксимации ядра свертки решения волнового уравнения с помощью нейронной сети.
Получены оценки относительной ошибки аппроксимации для нескольких порядков знаменателя, проведен анализ с методом scipy.interpolate.Pade библиотеки scipy языка программирования Python. Построение рациональной функции активации нейронной сети-суть аппроксимации некоторой рациональной функции - задача, имеющая множественные применения не только в области численных методов решения задач математической физики, но и в современных областях анализа данных.

В частности, последние статьи демонстрируют превосходство этого класса функций над другими в области компьютерного зрения (задача классификации изображений), моделирования языковых моделей-трансформеров([10, 11]).

Дальнейшее изучение этой проблемы также абсолютно оправдано с точки зрения изучения вопроса оптимальности начального распределения весов, а также обобщения на более сложные структуры (в частности, в работах [8] и [12] речь идёт о локально абелевых группах)

# Список литературы


1) Н. А. Зайцев "Прозрачные граничные условия для волнового уравнения в канале кругового сечения"

2) S. Haykin "Neural networks and machine learning"

3) Charu C. Aggarwal "Neural networks and deep learning"

4) C. Zhou "The Pade approximant based network for variational problems"

5) Cybenko, G. V. Approximation by Superpositions of a Sigmoidal function

6) Keras API documentation

7) Выпуклая оптимизация : учебное пособие / Е.А. Воронцова, Р.Ф. Хильдебранд, А.В. Гасников, Ф.С. Стонякин. Москва, МФТИ

8) T.P. Chen and H. Chen, Approximations of continuous functionals by neural networks with application to dynamic systems

9)V. Peiris, N. Sharonc, N. Sukhorukovaa, J. Ugonb Generalised rational approximation and its application to improve deep learning classifiers

10) V. Peiris. Rational activation functions in neural networks with uniform based loss functions and its application in classification.

11) Nicolas Boullé, Yuji Nakatsukasa, Alex Townsend. Rational neural networks.
12) Stella Rose Biderman. Neural Networks on Groups

Спасибо за внимание!