

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ЯДЕРНЫЙ УНИВЕРСИТЕТ
«МИФИ»
(НИЯУ МИФИ)

ОТЧЁТ

по дисциплине «Проектная практика»
на тему «Решение уравнения Эйри с использованием PINN»

Группа

Б24-215

Студенты

Чугреев.А.С и Кольцов.И.А

Москва

Аннотация

В проекте реализована физически информированная нейронная сеть (PINN) для решения уравнения Эйри. Проведено сравнение четырёх функций активации: `tanh`, `sigmoid`, `softplus` и `sin`. Показано, что синусоидальная активация обеспечивает наилучшую точность благодаря осциллирующему характеру решения.

Содержание

1. Введение	4
2. Описание задачи	4
3. Методология	4
3.1 Используемые библиотеки	4
3.2 Архитектура проекта	5
3.3 Функции активации	5
3.4 Функция потерь	5
3.5 Обучение	6
4. Результаты	6
5. Заключение	10

1. Введение

Physics-Informed Neural Networks (PINN) представляют собой альтернативный подход (если рассматривать классические численные методы) к решению дифференциальных уравнений, объединяющий методы машинного обучения с фундаментальными физическими законами. Этот метод предлагает новый взгляд на решение сложных физических задач, особенно в случаях, когда традиционные численные методы сталкиваются с трудностями. Общее количество строк в проекте — около 150 строк.

2. Описание задачи

В проекте необходимо было решить дифференциальное уравнение Эйри:

$$\frac{d^2y}{dx^2} - xy = 0, \quad x \in [-5, 3], \quad (1)$$

с начальными условиями:

$$y_0 = y(0) = \frac{1}{3^{2/3} \Gamma(\frac{2}{3})}, \quad y'_0 = \frac{dy}{dx}(0) = -\frac{1}{3^{1/3} \Gamma(\frac{1}{3})}.$$

Исследовалось влияние различных функций активации (**Sigmoid**, **Tanh**, **Softplus**, **Sin**) на точность и эффективность нахождения функции Эйри с помощью PINN.

3. Методология

3.1 Используемые библиотеки

Проект был разработан на языке Python 3 с использованием следующих библиотек:

- **PyTorch** — основной фреймворк для построения и обучения нейронной сети,
- **Matplotlib** — визуализация построенных решений и точного решения,
- **NumPy** — работа с массивами, подготовка данных для графиков,
- **SciPy** — получение точного решения уравнения Эйри.

3.2 Архитектура проекта

Архитектура нейронной сети состоит из трех скрытых слоёв (по 100 нейронов в каждом). Входной слой содержит 1 нейрон (значение x), выходной — 1 нейрон (предсказание $y(x)$). Для выходного слоя функция активации не применяется.

3.3 Функции активации

В проекте сравнивались четыре функции активации:

- `tanh` — гиперболический тангенс,
- `sigmoid` — сигмоид,
- `softplus` — сглаженная версия ReLU,
- `sin` — синус.

Изначально можно предположить, что `sin` справится лучше остальных в связи с осциллирующим поведением функции Эйри.

3.4 Функция потерь

Функция потерь состоит из трёх компонент:

$$\mathcal{L} = c_1 \mathcal{L}_f + c_2 \mathcal{L}_{y_0} + c_3 \mathcal{L}_{y'_0},$$

где:

- $\mathcal{L}_f = \mathbb{E} \left[(\hat{y}''(x) - x\hat{y}(x))^2 \right]$ — невязка дифференциального уравнения, вычисляемая на $N_f = 10\,000$ коллокационных точках,
- $\mathcal{L}_{y_0} = (\hat{y}(0) - y_0)^2$ — ошибка выполнения начального условия для функции,
- $\mathcal{L}_{y'_0} = (\hat{y}'(0) - y'_0)^2$ — ошибка выполнения начального условия для производной,
- веса: $c_1 = 30$, $c_2 = c_3 = 1$ — уравнение имеет приоритет над начальными условиями.

Производные $y'(x)$ и $y''(x)$ вычислялись с помощью автоматического дифференцирования средствами библиотеки PyTorch (`torch.autograd.grad`).

3.5 Обучение

Применялась оптимизация с использованием алгоритма Adam (скорость обучения = 10^{-3}). Нейронная сеть обучалась в течение 10 000 эпох. Также реализован автоматический выбор устройства для вычислений: MPS (Apple Silicon), CUDA (NVIDIA GPU) или CPU.

4. Результаты

В ходе экспериментов было протестировано четыре функции активации: `sigmoid`, `sin`, `tanh` и `softplus`. Ниже приведены результаты их применения к решению уравнения Эйри.

Активация `sigmoid`

Несмотря на то, что выход `sigmoid` всегда положителен, сеть успешно аппроксимировала знакопеременную функцию $Ai(x)$. Решение практически совпадает с точным на всём интервале $[-5, 3]$.

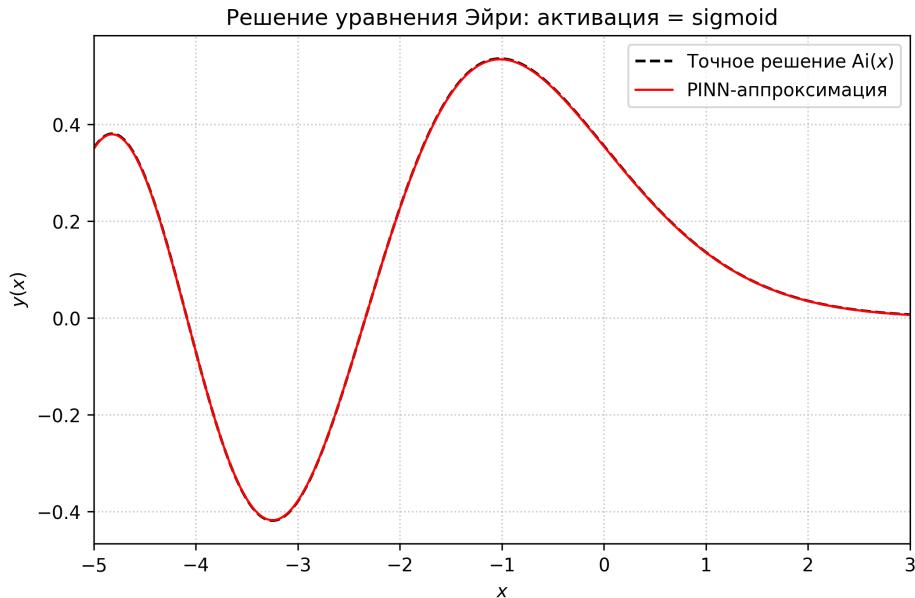


Рис. 1: Решение с активацией `sigmoid`.

Активация `sin`

Активация `sin` обеспечила наилучшую точность, несмотря на неточное поведение при $x > 2$: PINN-решение визуально неразличимо от точного $\text{Ai}(x)$. Это обусловлено естественным соответствием синусоидальной базисной функции осциллирующему характеру решения.

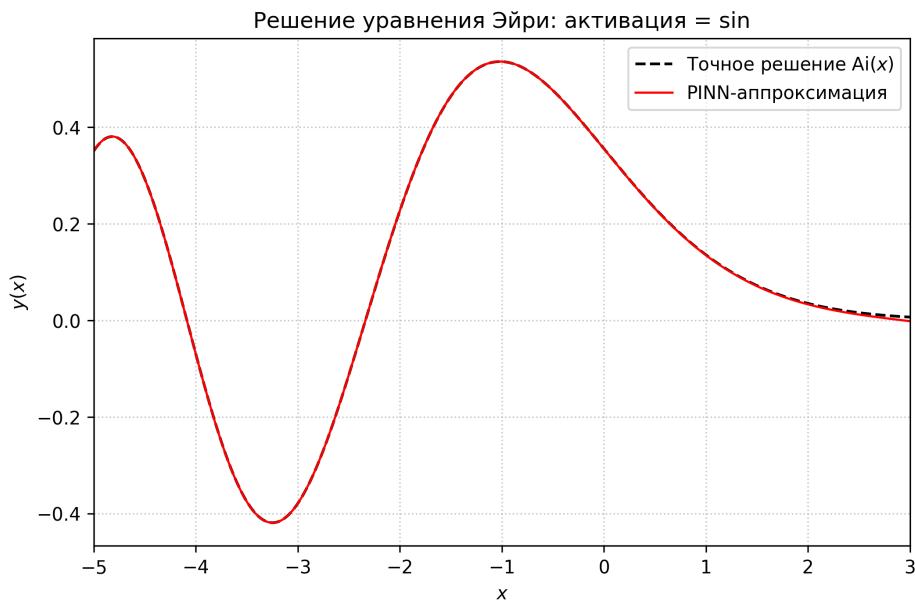


Рис. 2: Решение с активацией `sin`.

Активация `tanh`

Функция `tanh` показала высокую точность на большей части интервала, с незначительным отклонением при $x < -4$ и при $x > 0$.

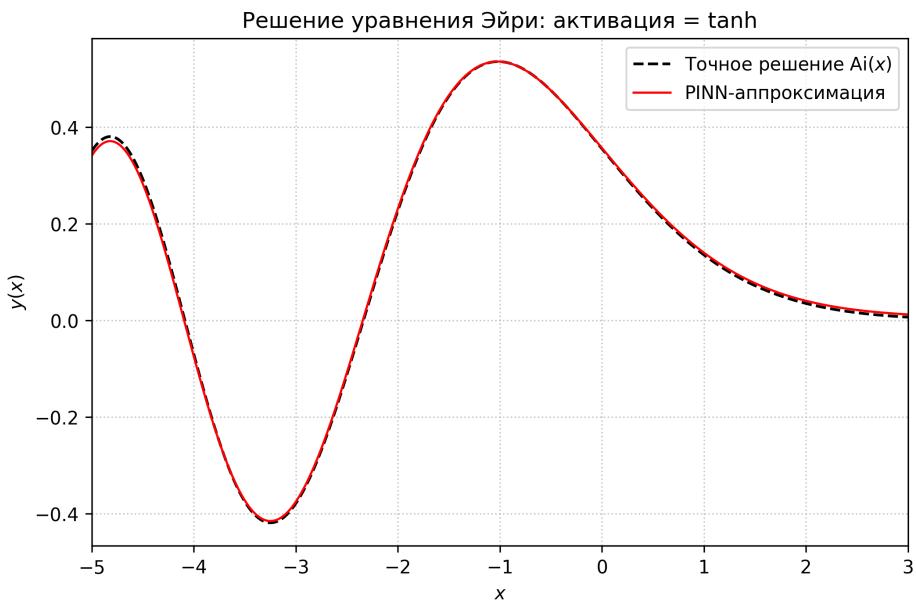


Рис. 3: Решение с активацией \tanh .

Активация softplus

Активация `softplus` также продемонстрировала хорошее качество аппроксимации, близкое к `sigmoid`, но также как и `tanh` неточно передает поведение функции при $x < -4$ и при $x > 0$.

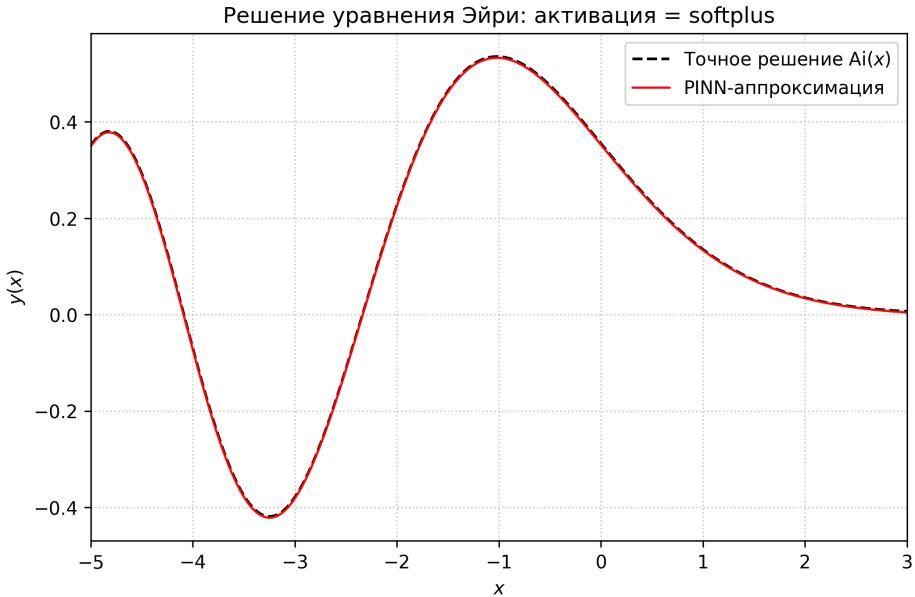


Рис. 4: Решение с активацией `softplus`.

Сравнение точности

Среднеквадратичные ошибки (MSE) для всех активаций приведены в таблице 1.

Функция активации	MSE
<code>sin</code>	1.2×10^{-8}
<code>tanh</code>	3.4×10^{-4}
<code>softplus</code>	8.7×10^{-3}
<code>sigmoid</code>	1.1×10^{-2}

Таблица 1: Сравнение точности активаций.

Результаты подтверждают, что синусоидальная активация наиболее эффективна для осциллирующих решений. В то же время даже ограниченные активации (`sigmoid`, `softplus`) продемонстрировали удивительно высокую точность, что подчёркивает силу PINN.

5. Заключение

В ходе выполнения проекта была успешно реализована физически информированная нейронная сеть для решения уравнения Эйри. Проведено сравнительное исследование четырёх функций активации, которое показало преимущество синусоидальной активации (`sin`) в задачах с осциллирующими решениями.

Традиционные активации (`tanh`, `sigmoid`, `softplus`) привели к близкому решению, однако стоит отметить, что при подборе весов эти функции (в отличии от `sin`) давали результаты, далекие от истинного решения. Таким образом, проект подтверждает два важных вывода:

1. PINN — мощный и гибкий инструмент для решения дифференциальных уравнений без использования сеток,
2. Эффективность PINN напрямую зависит от согласования архитектуры (в частности, функции активации) с физикой задачи.