**СОДЕРЖАНИЕ**

[Введение 6](#_Toc138761961)

[1 Искусственные нейронные сети 8](#_Toc138761962)

[1.1 Полносвязные нейронные сети прямого распространения 8](#_Toc138761963)

[1.2 Физико-информированные нейронные сети 9](#_Toc138761964)

[2 Решение задач Коши для обыкновенных дробно-дифференциальных уравнений 13](#_Toc138761965)

[2.1 Постановка задачи 13](#_Toc138761966)

[2.2 Решение задач с фиксированными параметрами 15](#_Toc138761967)

[2.2.1 Выбор архитектуры нейронной сети 15](#_Toc138761968)

[2.2.2 Анализ полученных решений 20](#_Toc138761969)

[2.3 Решение задач с варьируемыми параметрами 22](#_Toc138761970)

[2.3.1 Выбор архитектуры нейронной сети 22](#_Toc138761971)

[2.3.2 Анализ полученных решений 23](#_Toc138761972)

[Заключение 26](#_Toc138761973)

[Список литературы 27](#_Toc138761974)

[Приложение А (обязательное) Список гистограмм 29](#_Toc138761975)

[Приложение Б (обязательное) Список рисунков 32](#_Toc138761976)

# Введение

Стремительное развитие высокопроизводительных вычислительных технологий, а также последние достижения в области машинного обучения, и, в частности, глубокого обучения, привели к революционным результатам в различных научных дисциплинах и областях, включая задачи классификации изображений, обработки естественного языка, распознавания жестов и др. Глубокое обучение использует искусственные нейронные сети с одним или множеством слоев для анализа и обработки данных [1]. Такая популярность и эффективность применения нейронных сетей (НС) связана, в первую очередь, с их способностью обрабатывать большие объемы информации и находить в ней сложные зависимости. Это можно объяснить способностью нейронных сетей хорошо аппроксимировать нелинейные функции.

В ходе анализа сложных физических и биологических моделей или инженерных систем исследователи сталкиваются с высокой стоимостью и сложностью получения и отбора данных. В таком случае большинство методов глубокого обучения (например, свёрточные или рекуррентные нейронные сети) становятся ненадежными и не обеспечивают достаточной сходимости и точности. Однако при моделировании подобных систем многие важные знания предметной области (физические законы, некоторые подтвержденные эмпирически правила и др.) не используются в практике современных методов машинного обучения, хотя они могут действовать как агенты регуляризации, ограничивая пространство допустимых решений.

Относительно недавно в сообществе научного машинного обучения появилось новое многообещающее направление: решение дифференциальных уравнений с использованием искусственных нейронных сетей, называемых физико-информированными нейронными сетями (physics-informed neural networks, PINN). Это новый класс нейронных сетей, который позволяет интегрировать знания о физических законах в процесс обучения, что позволяет повысить точность и эффективность решения дифференциальных уравнений. Таким образом, они представляют собой инновационный инструмент, позволяющий решать огромный спектр сложных задач в различных областях науки и техники.

Несмотря на спектр рассмотренных различными исследователями применений PINN, понимание их возможностей и ограничений все еще находится на довольно ранней стадии. Представление дифференциальных уравнений в виде задач оптимизации приводит к многочисленным проблемам, свойственным этой области, одной из которых является достаточно частое зацикливание градиентных методов на локальном минимуме. В силу разнообразности таких задач не существует единых алгоритмов обучения нейросетей, обеспечивающих эффективность и желаемую точность получаемых решений.

Целью данной работы является исследование применения физико-информированных нейронных сетей к решению задачи Коши для обыкновенного дробно-дифференциального уравнения (ДДУ). Цель работы достигается решением следующих задач:

1. построение модели физико-информированной нейронной сети для решения линейной и нелинейной задач Коши для обыкновенных ДДУ с фиксированными параметрами;
2. исследование влияния архитектуры нейросети на точность получаемых решений;
3. изучение влияния порядка дробной производной на точность решения;
4. модификация нейронной сети с учетом вариации параметров задач (начальное условие, параметры ДДУ).

# 1 Искусственные нейронные сети

## 1.1 Полносвязные нейронные сети прямого распространения

Искусственные нейронные сети – это математические модели, которые имитируют работу человеческого мозга. Они состоят из множества связанных между собой нейронов, каждый из которых получает некоторый объем входных данных, обрабатывает его и передает дальше в другие нейроны [1]. Нейронные сети объединены в слои, каждый из которых выполняет свою функцию: входной слой получает данные, которые далее обрабатываются и передаются через скрытые слои на выходной слой, выдающий ответ. Несмотря на довольно простую функцию одного нейрона, их множество, соединенное в большую сеть с настроенным взаимодействием между единицами, способно решать сложные задачи, которые невозможно решить с помощью классических алгоритмов.

Нейронные сети, состоящие из нескольких слоев, каждый из которых содержит нейроны, соединенные со всеми нейронами предыдущего и следующего слоев, называются полносвязными нейронными сетями [2]. Если соединения устанавливаются только от нижних слоев к верхним и отсутствует связь между нейронами одного слоя, то подобные нейросети называются сетями прямого распространения сигнала.

Пусть

|  |  |
| --- | --- |
|  | (1) |

– один слой нейронной сети, где – входные данные,

– функция активации,

– обучаемые веса,

– смещение.

Тогда нейронная сеть прямого распространения выражается как композиция 𝑛 слоев

|  |  |
| --- | --- |
|  | (2) |

где – совокупность всех обучаемых параметров .

Процесс обучения нейронной сети состоит в настройке весов нейронов сети таким образом, чтобы минимизировать целевую функцию, называемую функцией потерь. Она показывает разницу между значениями, предсказанными нейронной сетью, и фактическими значениями. Математически такая задача может быть записана следующим образом:

|  |  |
| --- | --- |
|  | (3) |

где 𝐿(𝜃) – функция потерь. В различных задачах могут использоваться различные функции потерь, например, кросс-энтропия для задач классификации или среднеквадратичная ошибка для задач регрессии.

Алгоритм обучения нейронной сети состоит из нескольких шагов:

1. начальная инициализация весов;
2. реализация алгоритма прямого распространения ошибки;
3. расчет функции потерь;
4. реализация алгоритма обратного распространения ошибки;
5. обновление весов с использованием градиентных методов;
6. возврат к шагу 2 и повторение процесса обучения, пока критерий останова не будет выполнен.

## 1.2 Физико-информированные нейронные сети

Впервые термин PINN был представлен в работе [3] за авторством M. Raissi, P. Perdikaris и G.E. Karniadakis в 2019 году. В зависимости от характера доступных данных они разработали два различных типа алгоритмов, а именно модели непрерывного и дискретного времени. Эффективность предложенных решений демонстрируется на классических задачах течения жидкости, квантовой механики, а также распространения волн на мелкой воде.

Pan Zhang с группой авторов в представленной ими в феврале 2021 года работе [4] предлагают применить данный метод глубокого обучения без учителя для моделирования электромагнитных процессов на основе уравнений Максвелла, который превращает задачу электромагнитного моделирования в процесс оптимизации. В работе изучается несколько численных примеров, чтобы продемонстрировать эффективность этого метода для моделирования электромагнитных процессов.

Teeratorn Kadeethum, Thomas M. Jørgensen и Hamidreza M. Nick в работе [5] представляют возможности применения физико-информированных нейронных сетей для решения нелинейных мультифизических проблем, которые являются важными для многих областей, таких как биомедицинская инженерия, прогнозирование землетрясений и др. В частности, они исследуют, как расширить методологию физико-информированных нейронных сетей для решения как прямых, так и обратных задач для уравнений нелинейной диффузии и Биота. Также исследуется точность физико-информированных нейронных сетей с различными размерами обучающих примеров и выбором гиперпараметров.

В работе [6] Hamidreza Eivazi, Mojtaba Tahani, Philipp Schlatter и Ricardo Vibuesa используют PINN для решения осредненных по Рейнольдсу уравнений Навье-Стокса. Их результаты показывают превосходную применимость физико-информированных нейронных сетей для моделирования ламинарного и турбулентного потоков течения жидкости.

Работа [7], написанная Guofei Pang, Lu Lu и G.E. Karniadakis представляет гибридный подход к построению функции потерь с использованием автоматического дифференцирования для операторов целого порядка и числовой дискретизации для дробных операторов. Эффективность данного подхода демонстрируется на примере одно-, двух- и трехмерных задач для дробных уравнений адвекции-диффузии. На вход PINN подается множество точек пространственно-временной области, распределение которых оказывает влияние на «гибкость» обученной сети.

Таким образом, физико-информированные нейронные сети представляют собой тип универсальных аппроксиматоров функций, которые могут встраивать знания о любых физических законах в процесс обучения и могут быть описаны уравнениями в частных производных. Они преодолевают ограниченную доступность данных некоторых биологических и инженерных систем, из-за которой большинству современных методов машинного обучения не хватает надежности, что делает их неэффективными в этих сценариях. Предварительное знание общих физических законов действует при обучении нейронных сетей в качестве агента регуляризации, который ограничивает пространство допустимых решений, повышая точность аппроксимации функции. Таким образом, встраивание этой априорной информации в нейронную сеть приводит к увеличению информационного содержания доступных данных, облегчая алгоритм обучения для захвата правильного решения и хорошего обобщения даже с небольшим количеством обучающих примеров.

Пусть дана задача:

|  |  |
| --- | --- |
|  | (4) |

где – дифференциальный оператор, – вектор параметров задачи и – неизвестное решение. Тогда PINN аппроксимирует решение для набора параметров функцией , встраивая уравнения в функцию потерь следующим образом:

|  |  |
| --- | --- |
|  | (5) |

где и – вектора значений аргумента из и , на которых ведется обучение,

и – длины и ,

– вектор обучающих значений параметра ,

– длина

Одной из особенностей PINN является использования алгоритма автоматического дифференцирования, который позволяет вычислять значения целочисленных дифференциальных операторов, не прибегая к дискретизации. Однако в случае дифференциального оператора дробного порядка возникает необходимость использовать численную аппроксимацию.

Для оценки качества работы НС в данной работе используется метрика MAE (mean absolute error) – средняя абсолютная ошибка:

где – значение, предсказанное НС,

– фактическое значение.

# 2 Решение задач Коши для обыкновенных дробно-дифференциальных уравнений

## 2.1 Постановка задачи

Дифференциальными уравнениями дробного порядка или, кратко, дробно-дифференциальными уравнениями (ДДУ), называются уравнения, в которых неизвестная функция входит под знаком производной дробного порядка. Как и в теории классических дифференциальных уравнений целого порядка, дробно-дифференциальное уравнение называется обыкновенным, если входящая в уравнение неизвестная функция является функцией одной независимой переменной [8].

Рассматриваются 2 тестовые задачи для обыкновенных ДДУ, имеющие аналитическое решение:

1. Задача типа Коши для линейного обыкновенного ДДУ:

|  |  |
| --- | --- |
|  | (6) |

где – дробная производная Капуто,

– порядок дробной производной.

Решение имеет вид:

|  |  |
| --- | --- |
|  | (7) |

где – функция Миттаг-Леффлера.

1. Задача типа Коши для нелинейного обыкновенного ДДУ:

|  |  |
| --- | --- |
|  | (8) |

Решение имеет вид:

|  |  |
| --- | --- |
|  | (9) |

где

С учетом уравнения (5) задачи (6) и (8) могут быть переписаны в следующем виде:

где N – длина вектора значений нейронной сети. Данные функции потерь являются актуальными в случае решения задач с фиксированными параметрами, рассматриваемыми подробно в подразделе 2.2.

Если на вход нейронной сети подаются дополнительные параметры, то функции потерь для задачи (6) могут принимать следующие виды:

|  |  |
| --- | --- |
|  | (10) |

где – вектор входных значений параметра .

|  |  |
| --- | --- |
|  | (11) |

где – вектор входных значений параметра .

|  |  |
| --- | --- |
|  | (12) |

где – вектор входных значений параметра .

Аналогично функция может быть определена для задачи (8). Результаты, полученные с использованием при обучении функций (10), (11) и (12), приведены в подразделе 2.3.

## 2.2 Решение задач с фиксированными параметрами

### 2.2.1 Выбор архитектуры нейронной сети

Правильный выбор архитектуры нейронной сети является одним из самых важных факторов, определяющих ее эффективность. Она имеет серьезное влияние на обучаемость сети: слишком маленькая сеть может не иметь достаточной вычислительной способности для нахождения сложных зависимостей в данных, а слишком большая может привести к явлению, называемому переобучением, а также низкой способности к обобщению. Кроме того, каждой конкретной задаче требуется отдельный подбор оптимальной архитектуры. В условиях отсутствия правил или каких-либо законов по выбору архитектуры, этот процесс превращается в итеративный перебор с различными параметрами.

На процесс обучения НС также серьезное влияние оказывает начальная инициализация весов. Неправильная инициализация может привести к проблемам с градиентными методами оптимизации, а именно затуханию или «взрыву» градиентов, а также к проблемам с недо- и переобучением. Правильная инициализация, наоборот, может серьезно ускорить сходимость сети, а также повысить окончательную точность модели. Существует множество различных алгоритмов начальной инициализации, однако в данной работе используется метод инициализации по Ксавье [9], который является одним из самых распространенных в области глубокого обучения нейронных сетей.

В таблице 1 приведены гиперпараметры нейронной сети, используемые в дальнейших расчетах данного раздела.

Таблица 1 – Гиперпараметры НС (одно входное значение)

|  |  |
| --- | --- |
| Гиперпараметр | Значение |
| Функция активации | Tanh |
| Скорость обучения (learning rate) | 0.01 |
| Коэффициент регуляризации | 1e-5 |
| Количество эпох обучения | 20000 |
| Оптимизатор | Adam [10] |

В таблице 2 приведены параметры задач, зафиксированные при проведении расчетов в данном подразделе.

Таблица 2 – Параметры задач (одно входное значение НС)

|  |  |  |
| --- | --- | --- |
| Параметр | Линейная задача (6) | Нелинейная задача (8) |
|  | [0, 1] | [0, 1] |
|  | 0.1 | 0.9 |
|  | 1 | – |
|  | 0.5 | 0.5 |
|  | – | 0.5 |

В силу стохастического поведения НС (в том числе, вследствие случайной инициализации весов), выбор количества скрытых слоев, а также их ширины производится по средней ошибке за несколько испытаний. На рисунках 1 и 2 приведены гистограммы, соответствующие 1 и 5 испытаниям на разных архитектурах для задачи (6).

Рисунок 1 – Зависимость ошибки от архитектуры сети   
(1 испытание для каждого варианта)

Рисунок 2 – Зависимость ошибки от архитектуры сети   
(5 испытаний для каждого варианта)

Таким образом, можно заметить, что наименьшие ошибки по результатам 5 испытаний показывают НС с 1 или 2 скрытыми слоями. Дополнительно были проведены 10 испытаний с увеличенной шириной данных слоев (рис. 3).

Рисунок 3 – Зависимость ошибки от архитектуры сети   
(10 испытаний для каждого варианта)

Таким образом, в соответствии с гистограммами, представленными выше, можно сделать вывод, что наименьшая ошибка при решении задачи (6) достигается на архитектуре НС, состоящей из 2 скрытых слоев по 5 нейронов в каждом.

Аналогичные гистограммы для задачи (8) представлены в приложении А. По ним можно сделать вывод, что для нелинейной задачи оптимальной является иная архитектура, а именно 1 слой шириной в 80 нейронов.

Важным критерием при обучении НС является объем обучающей выборки. Чем больше данных используется при обучении, тем эффективнее и точнее становится сеть. Использование небольшой выборки может привести к проблеме переобучения, когда НС будет запоминать определенные примеры, вместо их обобщения, что приведет к неудовлетворительным результатам на новых входных данных. С другой стороны, использование слишком большого объема обучающих данных может привести к серьезному росту вычислительной сложности, а также времени обучения. Поэтому важной задачей является определение оптимального размера обучающей выборки. На рисунках 4 и 5 представлены графики зависимости ошибки от длины входного вектора для линейной и нелинейной задач соответственно.

Рисунок 4 – Зависимость ошибки от объема обучающей выборки   
(линейная задача)

Рисунок 5 – Зависимость ошибки от объема обучающей выборки (нелинейная задача)

Из рисунков 4 и 5 видно, что изменение ошибки для задачи (6) является гладким, тогда как для задачи (8) гладкости не наблюдается. Однако общим фактом для обеих задач является достижение наименьшей ошибки при длине входного вектора в 40 элементов.

### 2.2.2 Анализ полученных решений

В предыдущем пункте было показано, что для решения задач (6) и (8) оптимальными являются различные архитектуры НС. Зафиксируем определенные ранее оптимальные параметры нейросети для исследования зависимости ошибки от порядка дробной производной . Графики данных зависимостей представлены на рисунках 6 и 7. Примеры решения данных задач приведены в приложении Б.

Рисунок 6 – Зависимость ошибки от порядка дробной производной (линейная задача)

Рисунок 7 – Зависимость ошибки от порядка дробной производной (нелинейная задача)

В соответствии с рисунками 6 и 7 можно сделать вывод об около монотонной зависимости ошибки при решении линейной задачи (6) и отсутствия таковой при решении нелинейной задачи (8). Также следует отметить, что для линейной задачи минимум достигается на правом конце интервала, в то время как для нелинейной задачи это значение .

## 2.3 Решение задач с варьируемыми параметрами

### 2.3.1 Выбор архитектуры нейронной сети

Отличие данного подраздела от предыдущего заключается в передаче дополнительного параметра задачи () на вход нейронной сети. Это приводит к необходимости внесения изменений в архитектуру НС. Обучение происходит с учетом новых функций потерь , представленных формулами (10)-(12). Увеличение числа входных параметров также требует изменения размерности входного слоя. Кроме того, появляется необходимость пересмотра глубины и ширины нейронной сети для обеспечения достаточной способности НС к обработке большего числа данных.

Как и в пункте 2.2.1, были проведены испытания с эмпирически подобранными значениями количества скрытых слоев и числа нейронов. Гистограммы с результатами приведены в приложении А. Можно сделать вывод о необходимости увеличения глубины нейронной сети с учетом новых входных данных. Для линейной задачи (6) оптимальной становится архитектура, состоящая из 4 скрытых слоев по 5 нейронов в каждом, а для нелинейной задачи (8) – 2 скрытых слоя по 20 нейронов. Увеличение числа входных данных требует пересмотра и других гиперпараметров нейронной сети, влияющих на процесс обучения. Список параметров, использованных при обучении моделей НС в данном подразделе, приведен в таблице 3.

Таблица 3 – Список гиперпараметров НС (два входных значения)

|  |  |
| --- | --- |
| Гиперпараметр | Значение |
| Функция активации | Tanh |
| Скорость обучения (learning rate) | 0.01 |
| Коэффициент регуляризации | 1e-5 |
| Количество эпох обучения | 40000 |
| Оптимизатор | Adam |

### 2.3.2 Анализ полученных решений

Проведем обучение моделей нейронных сетей с различными вариациями входных параметров. В таблице 4 представлены варианты входных значений НС для обучения различных моделей решения задач (6) и (8).

Таблица 4 – Параметры задач (два входных значения НС)

|  |  |  |
| --- | --- | --- |
| Входные параметры | Линейная задача (6) | Нелинейная задача (8) |
|  |  | |
|  |  | – |
|  |  | – |
|  | – |  |

Значения остальных параметров задач (кроме варьируемых) сохранены в соответствии с таблицей 2.

На рисунке 8 представлена обученная модель PINN для решения линейной задачи (6) с различными значениями порядка дробной производной .

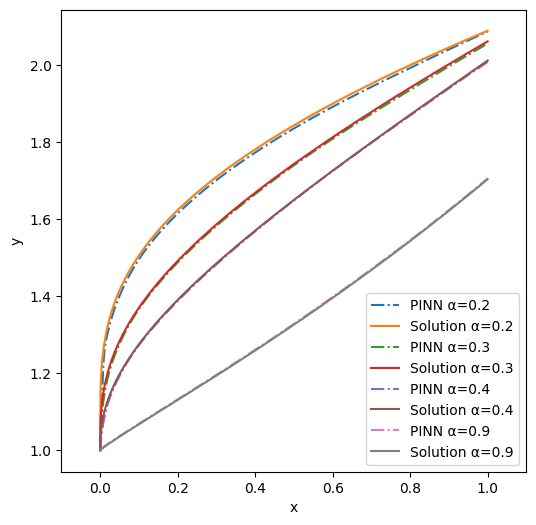


Рисунок 8 – Решения линейной задачи с различными входными значениями порядка дробной производной

Аналогичная модель для решения нелинейной задачи (8) представлена в приложении Б. Таким образом, в отличие от НС, представленных в пункте 2.2.2, обученная один раз модель способна решать поставленные задачи Коши для значений порядка дробной производной из интервала .

Таким же образом могут варьироваться и другие параметры в соответствии с таблицей 4. На рисунке 9 приведен пример решения линейной задачи (6) с различными начальными условиями для .

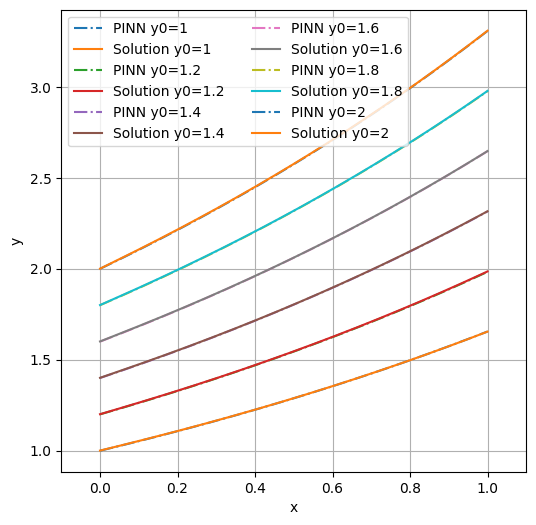


Рисунок 9 – Решения линейной задачи с различными начальными условиями

Примеры решения этой же задачи для других порядков производной приведены в приложении Б. В соответствии с рисунком 9 полученная модель показывает хорошую точность независимо от начального условия задачи (6).

Таким же образом обучаются модели с другими входными параметрами задач. Модель для решения линейной задачи (6) с параметром представлена в приложении Б. На рисунке 10 представлена модель PINN для решения нелинейной задачи с различными значениями параметра при .

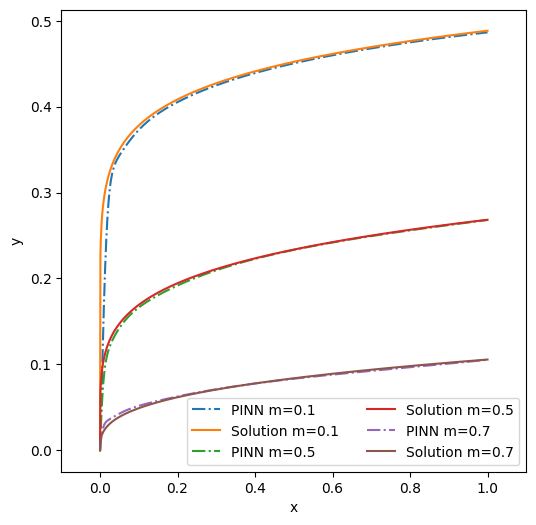


Рисунок 10 – Решения нелинейной задачи с различными входными значениями

Таким образом, в данном пункте была продемонстрирована возможность обучения PINN с учетом вариации входных параметров, что является серьезным преимуществом перед используемыми на практике в данный момент классическими численными методами.

# Заключение

В ходе выполнения данной выпускной квалификационной работы было исследовано применение физико-информированных нейронных сетей к решению задачи Коши для обыкновенного дробно-дифференциального уравнения.

Показано влияние архитектуры нейронной сети на точность получаемых решений. Установлено, что для решения линейной и нелинейной задач оптимальными становятся различные архитектуры, как по количеству скрытых слоев, так и по их ширине. Однако наименьшая ошибка достигается при одинаковой длине вектора входных данных при решении обеих задач.

Показаны возможности физико-информированных нейронных сетей к решению задач с вариацией параметров. А именно, для линейной задачи обучены модели решения с различным начальным условием , порядком дробной производной , а также параметром уравнения ; для нелинейной задачи – с порядком дробной производной и параметром уравнения . Данная особенность таких нейронных сетей является выигрышной по отношению к традиционным численным методам решения подобных задач, так как обученная однажды нейронная сеть способна давать решения независимо от значения параметра, использованного при ее обучении.

Таким образом, в работе показаны возможности применения физико-информированных нейронных сетей к решению задач Коши, как с фиксированными, так и с варьируемыми параметрами. Улучшение точности и эффективности представленных моделей является обширной темой для дальнейших исследований, в том числе по поиску оптимальных архитектур и правил их выбора для конкретных задач.

Я подтверждаю, что настоящая работа написана мною лично, не нарушает интеллектуальные права третьих лиц и не содержит сведения, составляющие государственную тайну.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/ Миянов М.Р.

# Список литературы

1. Николенко, С. Глубокое обучение [Текст] / С. Николенко, А. Кадурин, Е. Архангельская. – СПб : Питер, 2018. – 480 с.
2. Будума, Н. Основы глубокого обучения. Создание алгоритмов для искусственного интеллекта следующего поколения [Текст] / Н. Будума, Н. Локашо; пер. с англ. Коробейникова А. – М : Манн, Иванов и Фербер, 2020. – 304 с.
3. Raissi, M. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations [Текст] / M. Raissi, P. Perdikaris, G.E. Karniadakis // Journal of Computational Physics – 2019. – Режим доступа: https://doi.org/10.1016/j.jcp.2018.10.045.
4. A Maxwell’s Equations Based Deep Learning Method for Time Domain Electromagnetic Simulations [Текст] / P. Zhang [и др.] // IEEE Journal on Multiscale and Multiphysics Computational Techniques. – 2021. – Режим доступа: https://www.researchgate.net/publication/349158001.
5. Kadeethum, T. Physics-Informed Neural Networks for Solving Nonlinear Diffusivity and Biot’s Equations [Текст] / T. Kadeethum, T. M. Jørgensen, H. M. Nick // PLoS ONE 15(5): e0232683 – 2020. – Режим доступа: https://doi.org/10.1371/journal.pone.0232683.
6. Physics-informed neural networks for solving Reynolds-averaged Navier-Stocks equations [Текст] / H. Elvazi [и др.] // Physics of Fluids 34 – 2022. – Режим доступа: [https://www.researchgate.net/publication/353399217](https://www.researchgate.net/publication/353399217/).
7. Pang, G. fPINNs: Fractional Physics-Informed Neural Networks [Текст] / G. Pang, L. Lu, G. E. Karniadakis // Society for Industrial and Applied Mathematics – 2019. – Режим доступа: <https://doi.org/10.1137/18M1229845>.
8. Лукащук, В.О. Обыкновенные дифференциальные уравнения дробного порядка: основы классической теории и группового анализа [Текст] : учебное пособие / В.О. Лукащук, С.Ю. Лукащук. – Уфа: УГАТУ, 2022. – 170 с.
9. Xavier, G. Understanding the difficulty of training deep feedforward neural networks [Текст] / G. Xavier, Y. Bengio // Journal of Machine Learning Research. – 2010. – Режим доступа: https://www.researchgate.net/publication/215616968.
10. Kingma, D. P. Adam: A Method for Stochastic Optimization [Текст] / D. P. Kingma, J. L. Ba // 3rd International Conference for Learning Representations. – 2015. – Режим доступа: <https://doi.org/10.48550/arXiv.1412.6980>.
11. Wang, S. When and why PINNs fail to train: A neural tangent kernel perspective [Текст] / S. Wang, X. Yu, P. Perdikaris. – 2020. – Режим доступа: https://doi.org/10.48550/arXiv.2007.14527.
12. Самко С.Г. Интегралы и производные дробного порядка и некоторые их приложения [Текст] / С.Г. Самко, А.А. Килбас, О.И. Маричев. – Минск : Наука и техника, 1987. – 688 с.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/ Миянов М.Р.

# Приложение А (обязательное) Список гистограмм

Рисунок А.1 – Зависимость ошибки от архитектуры сети

(1 испытание для каждого варианта)

Рисунок А.2 – Зависимость ошибки от архитектуры сети

(5 испытаний для каждого варианта)

Рисунок А.3 – Зависимость ошибки от архитектуры сети

(10 испытаний для каждого варианта)

Рисунок А.4 – Зависимость ошибки от архитектуры сети

(линейная задача с параметром, 1 испытание для каждого варианта)

Рисунок А.5 – Зависимость ошибки от архитектуры сети

(нелинейная задача с параметром, 1 испытание для каждого варианта)

# Приложение Б (обязательное) Список рисунков

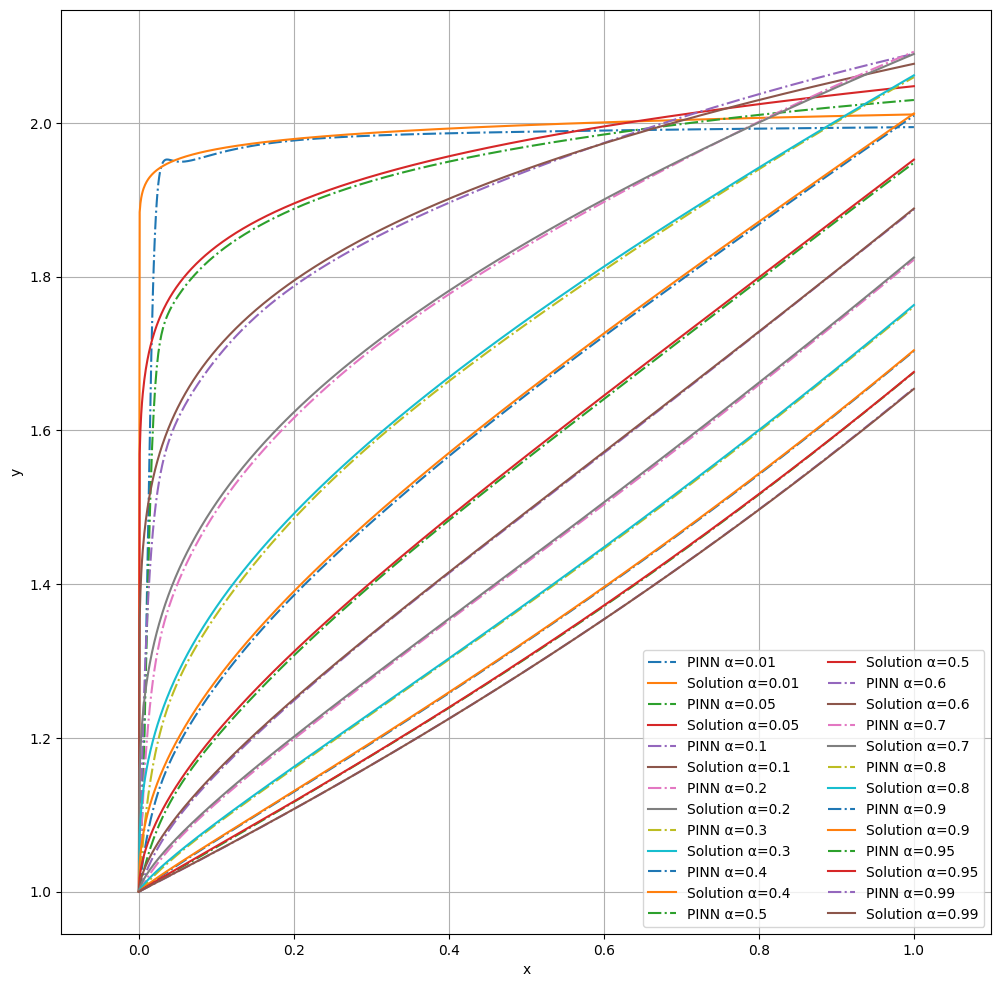
**

Рисунок Б.1 – Графики решения для различных порядков дробной производной (линейная задача)

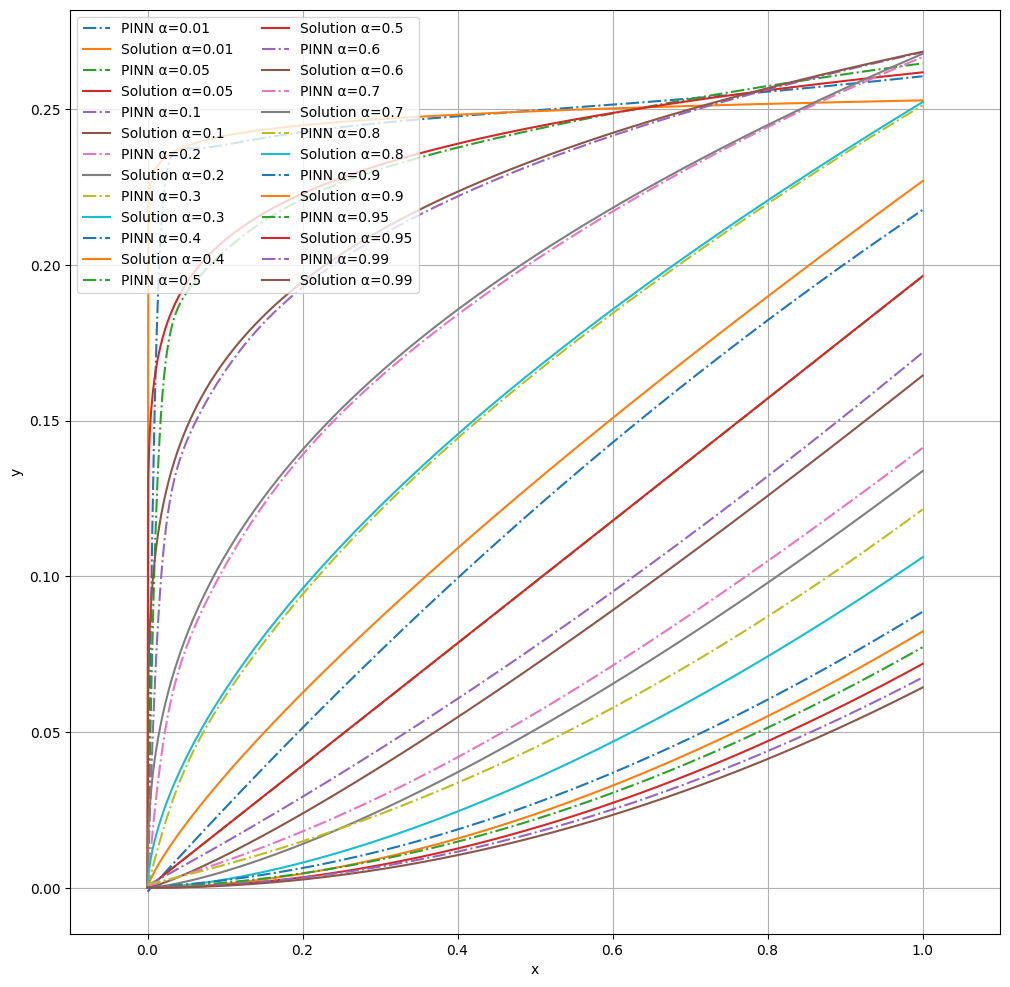
**

Рисунок Б.2 – Графики решения для различных порядков дробной производной (нелинейная задача)

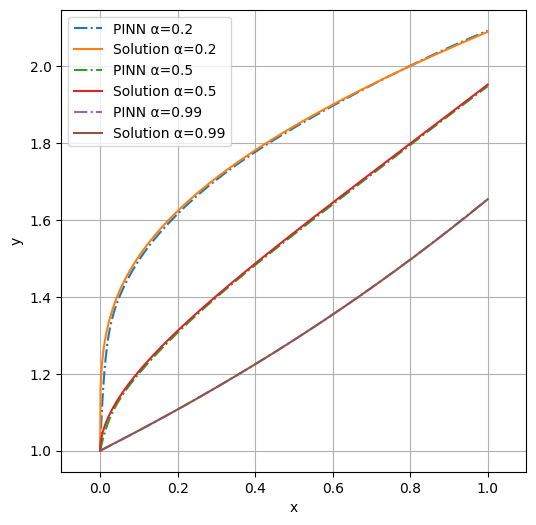
**

Рисунок Б.3 – Графики решения для выборочных порядков дробной производной (линейная задача)

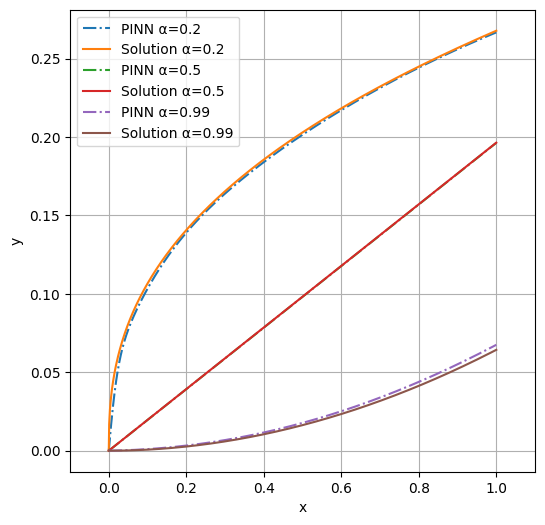
**

Рисунок Б.4 – Графики решения для выборочных порядков дробной производной (нелинейная задача)

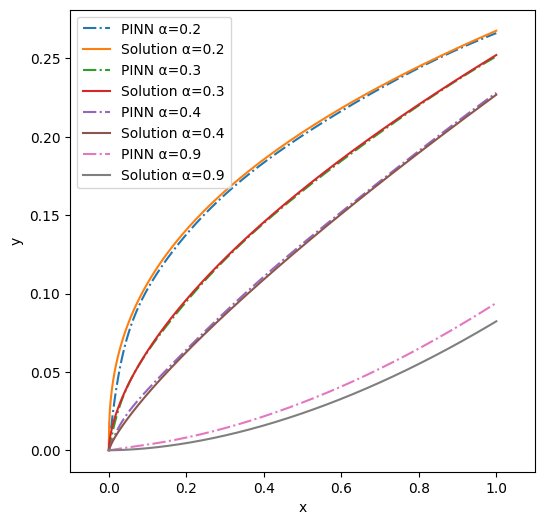


Рисунок Б.5 – Решения нелинейной задачи с различными входными значениями порядка дробной производной

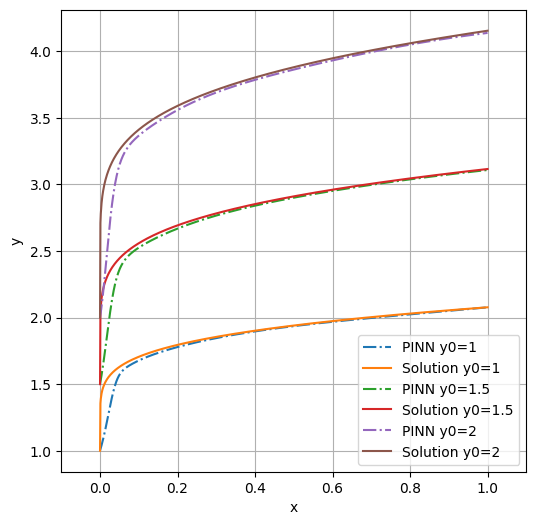


Рисунок Б.6 – Решения линейной задачи с различными начальными условиями ()

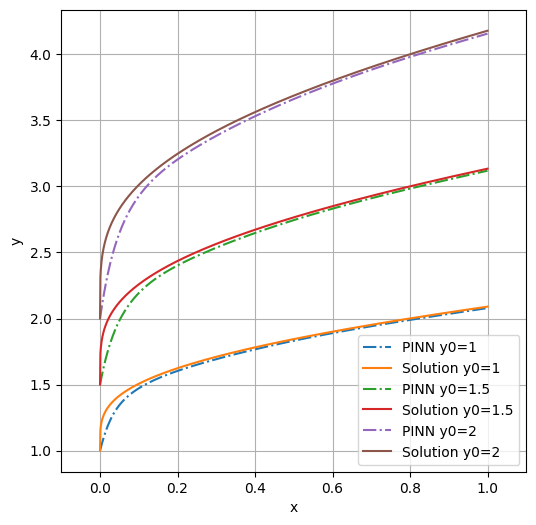


Рисунок Б.7 – Решения линейной задачи с различными начальными условиями ()

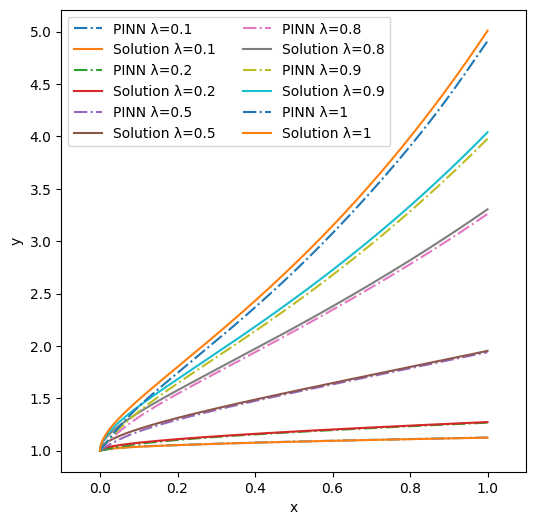


Рисунок Б.8 – Решения линейной задачи с различными входными значениями параметра