

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
ФГАОУ ВО НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

Факультет компьютерных наук
Образовательная программа «Прикладная математика и информатика»

УДК 004.8

Отчет об исследовательском проекте

на тему “Сценарии глубокого обучения: нейродифференциальные уравнения и их параметры”

Выполнил:

студент группы БПМИ196

03.06.2021

Дата

Подпись

А. О. Боровский

И.О. Фамилия

Принял:

руководитель проекта

Василий Александрович Громов

Имя, Отчество, Фамилия

Зам. рук. деп., профессор, д. ф.-м. наук

Должность

Департамент анализа данных и искусственного интеллекта

Место работы

Дата 2021

Оценка (по 10-тибалльной шкале)

Подпись

Москва 2021

Реферат

Данное исследование является первым этапом в решении задачи применения концепции Neural ODE для проведения бифуркационного анализа работы ядерного реактора. Изучается эффективность функционирования революционного алгоритма на классической задаче распознавания пар букв, представленных в бинарном виде. В целях повышения значений показателей, по которым результаты обучения сравниваются с полученными при реализации многослойного перцептрона для аналогичной задачи, стандартный метод последовательных прямого и обратного распространений заменяется методом пристрелки. Через сопоставление времени работы и процента верных ответов двух нейронных сетей планируется оценить минусы стандартной концепции перцептрона и преимущества Neural ODE в ее новом прочтении на пригодность для дальнейшего применения в решении более серьезных задач. Результаты позволяют вывить даже при ограниченных тестах и выборках значительные недостатки перцептрона. Однако переход к новой реализации нейродифференциальных уравнений не позволяет на данный момент полностью реализовать поставленную цель.

Содержание

Вступление

| | |
|---|---|
| 1. Основные термины и определения..... | 4 |
| 2. Введение..... | 6 |
| 2.1. Актуальность..... | 6 |
| 2.2. Объект и предмет исследования..... | 7 |
| 2.3. Цель и задачи..... | 7 |

Основная часть

| | |
|---|----|
| 5. Обзор и сравнительный анализ источников..... | 9 |
| 6. Алгоритмы..... | 11 |
| 7. Ход эксперимента..... | 14 |

Заключение

| | |
|----------------------------|----|
| 8. Итоги исследования..... | 19 |
| 9. Библиография..... | 23 |

Вступление

*“Единственная проблема современности заключается в том, сумеет ли человек
пережить свои собственные изобретения.”*

Луи де Бройль, французский физик

Основные термины и определения

- I. Вариационные исчисления - раздел анализа, изучающий методы, позволяющие найти максимальные и минимальные значения функционалов.
- a. Функционал - отображение, ставящее в соответствие каждой функции из некоторого множества определённое число.
 - b. Дифференциальное уравнение (ДУ) - уравнение, связывающее некоторое число функций, их производные и независимую переменную.
 - c. Обыкновенные ДУ (ODE) - уравнения от некоторого числа функций и их производных, зависящие от одной независимой переменной. Обычно ODE имеет вид
$$F(y, y', \dots, y^{(n)}) = 0$$
 - d. Краевая задача - задача, заключающаяся в нахождении решения определённого ДУ или их системы при условии удовлетворения заданным граничным условиям.
 - e. Численные методы - алгоритмы, представляющие решение для определённого вида задач (в множество которых входят и краевые) в численном виде.
 - f. Метод Рунге-Кутты - семейство итеративных численных методов, позволяющих получить приближительное решение ODE. В данном исследовании используется исключительно метод 4-ого порядка.
 - g. Метод Ньютона - итерационный численный метод, целью которого является выявление значений, при которых значение функции становится равным нулю.

h. Метод пристрелки - численный метод, сводящий краевую задачу к классической задаче Коши при сохранении изначальных ДУ.

II. Бифуркация динамической системы - качественное изменение ДУ при непрерывном изменении параметров.

III. Нейронная сеть (ИНС) - компьютерная модель, представляющая из себя систему обрабатывающих блоков (нейронов), которые взаимодействуют путем сигналов, передаваемых через заданное количество связей, каждая наделенная некоторым весовым коэффициентом.

a. Контролируемое обучение - метод обучения, при котором ИНС тренируется сопоставлять входные данные с выходными через анализ подаваемых пар (input — desired output).

b. ИНС прямого распространения (feedforward) - тип архитектуры, в котором поток информации происходит исключительно в направлении от input layer к output layer, не образуя каких-либо циклов.

c. Обратное распространение - модификация метода градиентного спуска, использующаяся с целью распределения ошибки на каждый уровень многослойной ИНС.

IV. Нейродифференциальные уравнения (Neural ODE) - семейство архитектур ИНС, заменяющая дискретную последовательность скрытых слоев на определение производной скрытого состояния:

$$h_{t+1} = h_t + f(h_t, \theta) \implies \frac{dh(t)}{dt} = f(h(t), t, \theta)$$

где h_t - значение нейронов слоя t .

V. Стационарный режим работы реактора - процесс, при котором любой из параметров, определяющих состояние реактора, неизменен во времени.

a. Устойчивое состояние - в результате работы реактора появляющиеся малые возмущения параметров устраняются и реактор возвращается к исходному состоянию. При накоплении отклонения от стационарного режима со временем состояние является неустойчивым.

Введение

Актуальность

Ядерная энергетика является одним из ключевых направлений развития российской промышленности, не только обеспечивая почти 20% всего потребления электроэнергии страны, но и формируя колоссальные портфели зарубежных заказов и подводя к международному сотрудничеству. Однако широко известны риски производства на атомных электростанциях (АЭС), центральными элементами которых являются ядерные реакторы. Вопросы предотвращения повтора прежних катастроф и обеспечения максимальной безопасности процесса актуальны для всей сферы уже несколько десятилетий.

Один из главных факторов благополучной работы АЭС заключается в устойчивости стационарных режимов реактора. Однако не всегда возможно обеспечивать существенный запас устойчивости, более того, имеют место случаи и неустойчивого состояния работы реактора. Вследствие этого вопрос высокой надежности прогнозирования динамики рабочей точки необычайно важен.

В то же время, сложные математические модели, к которым применяются методы теории бифуркаций динамических систем, требуют значительного времени для описания системы, а обоснованность результата, определенного без использования ЭВМ, признано считать крайне низким. [14] И несмотря на то, что уже много лет именно за компьютеризированными системами стоит задача расчета, проблема не теряет своей актуальности. Причина заключается в огромном множестве комбинаций развития ситуации.

Именно поэтому уже несколько лет компания Росатом проводит исследования по внедрению ИНС в процесс регулирования работы реактора, которые показывают на порядок более высокие показатели, чем прежние методы и в скорости расчетного моделирования, и в безошибочности результатов. Представляется, что применение для данных задач концепции Neural ODE в качестве механизма прогнозирования может значительно повысить эффективность развития данного направления и приблизить возможное внедрение ИНС в процесс управления реактором. Отмечается значительное преимущество нейродифференциальных уравнений перед остальными ИНС по затратам на память, росту ошибки аппроксимации и количеству проводимых операций, согласно статье авторов

концепции. Данный тип архитектуры был представлен в 2019 году, что лишь подчеркивает актуальность применения новых подходов в столь колоссальном по значимости вопросе оценки состояния реактора и его дальнейшей динамики.

Данное исследование представляется как первый этап разрешения проблемы. Изучив эффективность работы нейродифференциального уравнения в сравнении со стандартными алгоритмами, возможно будет в дальнейшем применить новую концепцию при анализе работы реактора. Особая ценность исследования заключается в том, что, в отличие от стандартных реализаций Neural ODE, данная работа предполагает пересмотр авторских идей и предложение новых подходов, предполагающих качественный скачок в эффективности. Тогда как достоверность результатов исследования представляется крайне высокой, возможности, в свою очередь, ограничены как временными ресурсами, так и физическими, что сказывается и на относительно небольшой выборке при тестировании, и на решении только одной классической задачи вместо более комплексного сопоставления, и на сравнении только с многослойным перцептроном вместо ряда современных алгоритмов нейронных сетей.

Объект и предмет исследования

Объектом исследования является концепция Neural ODE с внедренной предметной областью. В качестве предметов исследования можно выделить эффективность данного алгоритма в контексте таких показателей, как скорость вычислений и надежность результатов.

Цель и задачи

Цель данного исследования заключается в изучение нейродифференциальных уравнений и их показателей в сравнении со стандартными алгоритмами перцептрона на классической задаче различия изображений.

Задачи можно распределить по трем группам. Стоит отметить, что предполагается продолжение работы над данным исследованием в контексте моделирования работы реактора. Вследствие этого, объем изученных материалов превосходит узкой цели проекта второго курса, но является необходимым для дальнейших исследований.

I. Изучение математической теории

- Курс Дифференциальных Уравнений: виды и алгоритмы решения ДУ, задача Коши и краевая задача, численные методы: Рунге-Кутта, Ньютона, пристрелки
- Курс Вариационных Исчислений: метод вариаций в задачах с неподвижными границами, задача со свободными концами
- Курс Теории Бифуркаций: виды бифуркаций, решение бифуркационных задач

II. Изучение принципов работы ИНС

- Fundaments: Перцептрон, Рекуррентные нейронные сети, обратное распространение
- Концепция Neural ODE: статья авторов и их реализация

III. Реализация проекта

- Реализация многослойного перцептрона, распознающего буквы
- Пересмотр авторской концепции реализации нейродифференциального уравнения
- Реализация Neural ODE, распознающего буквы, в рамках новой парадигмы
- Анализ результатов работы и итоговых характеристик двух ИНС

Основная часть

1. Обзор и сравнительный анализ источников

Можно утверждать, что все использованные в рамках исследования источники подразделяются на три категории, обзор и анализ которых должен проходить в изоляции друг от друга:

1.1 Теоретические учебные материалы

Данный раздел включает в себя различного рода учебники или переработанные лекции изученных на требуемом уровне курсов Дифференциальных Уравнений [4; 9; 10; 16], Вариационных исчислений [13; 16] и Теории Бифуркаций [2; 4]. С одной стороны, все приведенные здесь материалы не могут быть сопоставлены с точки зрения надежности не только потому, что каждый признан достоверным, но и ввиду того, что источники по разным разделам анализа исключительно дополняют друг друга, соединяясь в одну общую картину. Более того, нельзя не отметить, что выбранный учебник для курса ДУ в рамках обучения на ФКН [8] совпадает во всех пройденных главах с остальными упомянутыми. Стоит отметить, что источники и по ДУ, и по Вариационным исчислениям были многократно переизданы (некоторые с общим временным промежутком в более полувека), что лишь подчеркивает их достоверность.

С другой стороны, имеет смысл обговорить полезность каждого из источников для исследования. Конечно, для понимания природы дифференциальных уравнений на требуемом уровне информации из исследования по теории бифуркаций [4], включившим лишь базовые формулы ODE, было недостаточно. Однако, по сравнению с более объемными учебниками [8; 16], содержащими, в том числе, излишнее для данных временных сроков количество теории, прикладная польза от повторения ранее изученного в них материала при прочтении краткого исследования [4] была крайне высока.

Имеет смысл отметить, что в изучении теории бифуркаций отдельное исследование [4] так же играло скорее ознакомительный характер. Поэтому для более полного понимания требуемых к применению в дальнейшем методов был изучен курс лекций по данной теме [5].

В вопросах базовых концепций машинного обучения [3] и химической стороны работы реакторов [17] мною были изучены единичные источники, позволившие в дальнейшем анализировать различные концепции и статьи с необходимой степенью понимания именно благодаря достоверности и актуальности изученной информации. В особенности это касается реализованного мною многослойного Перцептрона, высокая эффективность которого была достигнута именно благодаря опоре на исследованный материал.

Что касается численных методов - Рунге-Кутта, Ньютона и пристрелки - были использованы различные интернет-ресурсы и учебники, не только представляющие сами формулы и алгоритмы, но и поясняющие применяемые подходы.

1.2 Neural ODE: концепции и реализации

Определив центральной темой всего исследования идею нейродифференциальных уравнений, нельзя не сказать о первостепенной важности статьи ее авторов [7] и их реализации собственной библиотеки с применением PyTorch [1]. Стоит отметить, что в рамках данного исследования было решено отказаться от применения авторских программных материалов, на которые ссылается статья. Конечно, самостоятельная реализация заняла существенное время и не позволила продвинуться дальше в решении рабочей задачи с ядерным реактором. Однако принятое решение позволило полностью осознать принцип работы нейродифференциального уравнения, а также предложить новый подход в осуществлении определенных шагов. Необходимо упомянуть целый ряд обзоров на указанную статью [6; 18], позволивший не только более глубоко проанализировать общую идею, но и обратить внимание на каждый из ее аспектов ввиду отличающихся акцентов в разных материалах.

Изучив все представленные реализации Neural ODE, призванные заменить широко используемые ResNet, можно выделить две категории: авторская и схожие с ней [7; 18]. Последняя делится, в свою очередь, на две подгруппы: нацеленные на масштабные проекты и более элементарные, ориентированные на узко направленные небольшие задачи [12]. Тогда как официальная документация библиотеки torchdiffeq позволила осознать методы переноса абстрактных математических конструкций в конкретные классы и функции,

именно более простой код стал изначальной основой для написания собственной реализации ввиду ограниченных временных сроков.

1.3 Дискуссия вокруг предмета исследования

В ходе изучения природы поведения реакторов, были проанализированы статьи научного журнала “Атомная энергия” [14-15], авторы которых не только демонстрируют важность использования бифуркационного анализа при работе с ядерным реактором, который в реальности нередко приходит в состояние неустойчивого стационарного режима, но и описывают конкретные кейсы. Более того, подчеркивается важность использования ЭВМ для более точных и надежных результатов.

В то же время, примеров конкретного использования Neural ODE для данных задач не было обнаружено в мировой практике. Более того, кейсы применения ИНС как таковой единичны и носят скорей экспериментальный характер [19].

С одной стороны, в самом эксперименте отсутствует какая-либо прямая связь с вопросами бифуркационного анализа или работы ядерного реактора. В то же время, с самого начала подразумевалось комплексное изучение всей теоретической базы ввиду намерений продолжить исследование даже по окончании срока проекта. Поэтому не относящиеся напрямую к цели изученные материалы и решенные задачи, тем не менее, не только потребовали достаточно больших усилий, но и позволили осознать план большего исследования, выходящего за рамки курса.

2. Алгоритмы

2.1 Многослойный Перцептрон

Данная концепция нейронной сети является одной из самых базовых среди алгоритмов машинного обучения. Общая идея состоит в создании структуры, своего рода графа, вершины которого называются нейронами. Нейрон, в данном случае, получает некоторое значение на вход, обрабатывает его с помощью единой для всей структуры функции, а после передает полученное значение на выход. В свою очередь, слой

перцептрона есть некоторый блок из нейронов, одновременно и независимо друг от друга обрабатывающих поданные значения.

Работа перцептрона основана на принципе прямого и обратного распространения. Изначально в структуру поступает некоторый вектор входных данных. Любому из нейронов первого слоя соответствует ровно одно конкретное значение из вектора. После обработки функцией результат каждого нейрона передается каждому нейрону из последующего слоя - hidden layer (число этих слоев зачастую больше одного, они идут друг за другом, выполняя схожую функцию). Пересылка значений происходит через ребра, связывающие нейроны. У любого ребра есть свое конкретное значение веса, случайно заданное на некотором промежутке при инициализации структуры. Следовательно, дальнейшие нейроны получают сумму из значений выхода нейронов предыдущего слоя, каждое умноженное на значение весов.

Прямой ход, начавшись с поступления информации в input layer, ее обработки и передаче в некоторое число последовательных hidden layers, заканчивается на output layer. В моей изначальной реализации использовался один выходной нейрон, который обрабатывал сумму всех значений, переданных им от последнего из скрытых слоев, и давал итоговый ответ структуры.

Результат прямого распространения сравнивается с заданным учителем значением (в этом и состоит идея контролируемого обучения). В случае несоответствия рассчитывается значение ошибки L . От него мы и будем отталкиваться в back-propagation. Для минимизации потерь и, соответственно, корректировки выдаваемых структурой результатов необходимо изменить значения весов. После каждого нового поданного вектора используется стохастический градиентный спуск. Его суть состоит в минимизации ошибки через вычисление $\Delta w_{ij} = -\alpha \frac{\partial L}{\partial w_{ij}}$, где w_{ij} - вес ребра между i -ым и j -ым нейронами, α - learning rate (коэффициент, уменьшающий разброс в изменении весов).

Таким образом, обратное распространение состоит в проходе от конце перцептрона к его началу и последовательном обновлении всех весов, слой за слоем. После завершения данного процесса подается новый вектор значений и весь алгоритм повторяется снова. Итоговая цель - пропустить через перцептрон все имеющиеся в обучающей выборке вектора

значений определенное количество раз (эпох), в итоге получив максимально эффективную структуру при заданном числе слоев и нейронов в них.

2.2 Neural ODE

Как было ранее отмечено, алгоритм работы Neural ODE во многом основывается на принципах Residual network, прообразом которой можно назвать многослойный перцептрон. В ResNet последнее значение нового слоя рассчитывается как сумма выхода предыдущего слоя и функции от весов и этого значения. Концепция нейродифференциального уравнения состоит в устремлении к бесконечности числа слоев. Другими словами, происходит переход от дискретного принципа существования структуры к континуальному, где количество слоев становится тождественно понятию времени. Далее будет описан классический алгоритм функционирования Neural ODE.

Получив некоторый вектор значений на первый слой (а вернее сказать, на слой, соответствующий начальному моменту), структура одним из известных способов вычисляет

ДУ для каждого j -ого нейрона следующего временного рубежа: $\frac{dz_{1j}(t)}{dt} = f(z(t), t, \theta)^1$.

Заметим, что и значения весов θ , и число нейронов в слое N - есть функции от времени. Задавая изначально определенные моменты времени, которым соответствует каждый слой, тем самым определяем структуру всей ИНС.

После окончания прямого прохода, точно так же, как и в перцептроне, вычисляется значение ошибки L . В свою очередь, обратное распространение имеет кардинальные отличия от предыдущих вариантов. Так как необходимо определять значения $\frac{dL}{d\theta(t)}$, а метод дискретного случая не отвечает запросам нового подхода, авторы описывают adjoint method и доказывают видоизмененные формулы.

Значения самого adjoint $a(t) = \frac{dL}{dz(t)}$ рассчитываются упомянутым методом Рунге-

Кутта по следующему ДУ:

¹ Авторы статьи, где впервые была представлена эта концепция, используют метод Рунге-Кутта 4 порядка, что остается неизменным и для данного исследования.

$$\frac{da(t)}{dt} = -a(t) \frac{\partial f(z(t), t\theta(t))}{\partial z(t)}$$

Имея подходящую функцию и значения adjoint в необходимом интервале времени, мы аналогично классическому методу обратного распространения обновляем значения θ . После этого весь алгоритм повторяется заново.

Ход эксперимента

1. Этап подготовки

Для сопоставления эффективности двух ИНС было решено изначально провести эксперимент в рамках классической задачи распознавания изображения. Тогда как структуры способны работать с любыми парами, наличие лишь двух категорий важно для представляемых реализаций, однако легко может быть увеличено при расширении количества возможных исходов в численном виде. Символично были выбраны две буквы - "А" и "О". Особенно важно было найти диаметрально разные почерки: в противном случае, алгоритм обучения был бы крайне прост и обе ИНС заведомо выдали бы высокие результаты, не имеющие особой связи с задачей, обсуждаемой в исследовании. Необходимо отметить, что ввиду ограниченных физических ресурсов было собрано лишь 100 экземпляров.

По окончании первого этапа начался процесс обработки изображений. Каждое из них сжималось до фиксированных размеров (10×10), после чего производился перевод в бинарный вид. Другими словами, оценивался цвет каждого из пикселей относительно некоего пограничного значения. При значении более высоком пиксель читался как достаточно темный и в бинарной матрице изображения в соответствующую ячейку записывалась единица, в противном случае - 0.

Финальным этапом в рамках подготовки было разбиение общего множества изображений на две выборки: обучающую (train) и тестовую (test) в процентном соотношении 70 на 30 соответственно. С помощью определенных библиотек были созданы

отдельные папки и специальная система наименований, из которой можно было в дальнейшем понять требуемый результат (на изображении буква “А” или “О”).

2. Реализация многослойного Перцептрона

В отличие от в дальнейшем описываемых преобразований в алгоритме Neural ODE, в случае перцептрона был реализован самый классический вариант, ранее описанный в методах. Входной слой содержал 100 нейронов (по одному на каждое значение из матрицы 10×10). Так как целью исследования было сопоставление эффективности двух ИНС, в рамках эксперимента проверялись варианты от одного до пяти скрытых слоев, в каждом из которых ставилось от 1 до 120 нейронов. Одни из самых высокие результаты показала модель

$$\begin{array}{ccccccc} \underline{100} & - & \underline{60} & - & \underline{60} & - & \underline{1} \\ \text{input layer} & & \text{hidden layer 1} & & \text{hidden layer 2} & & \text{output layer} \end{array}$$

Подчеркнем, что хоть финальный вариант представлялся именно в таком устройстве, в результатах будут сопоставлены несколько наиболее успешных комбинаций. Что касается программной части реализации класса, то после создания ИНС требуемых размеров и с случайно заданными значениями весов, для обучающей части запускалась функция `request`. Получив запрос в виде матрицы значений некоторого изображения, она вызывала функцию прямого распространения `forward_propagate`. Итогом последней было некоторое выходное значение, которое сравнивалось с ответом учителя: единица для “А” и ноль для “О”. Стандартно реализуется выход в виде вектора длины, равной количеству алфавита, где должна быть лишь одна единица на позиции заданной буквы. Однако в условиях сравнения двух вариантов более логично было прибегнуть к одному выходному нейрону.

В случае ошибки, превышающей 0.1, передавался сигнал в функцию `back_propagation`, которая совершала обратный проход по перцептрону и с помощью `hidden_propagation` обновляла значения весов. После алгоритм рекурсивно повторялся до правильного ответа.

Для дальнейшего тестирования были написаны функции `testing` и `forward_test`, которые помимо выдачи итогового ответа согласно уже установившейся системе весов,

также представляли различного рода дополнительные данные: общий счетчик правильных ответов, номер проведенного теста, буква, исследуемая в тесте и каждый ответ ИНС.

Отдельно заметим, что были реализованы функции и части кода, которые не вошли в итоговую версию. С их помощью проводились отдельные расчёты, позволившие выявить комбинации нейронов с наиболее точными результатами, время работы при каждом устройстве, а также среднюю эффективность на промежутках. Эти результаты будут так же представлены в заключительной части исследования.

3. Реализация Neural ODE

Как это было указано ранее, в планах исследования состояло, в том числе, самостоятельное выведение формул, связанных с обратным распространением в Neural ODE, и его дальнейшее видоизменение в сторону синхронного решения системы ДУ.

Первым делом было определено граничное значение для сопряженного. Заметим, что используется стандартная функция потерь - среднеквадратическая ошибка:

$$L(z(t_0 + \int_{t_0}^{t_1} f(z(t), \theta(t), t) - z^*) = \frac{1}{2}(z(t_1) - z^*)^2 \implies a(t_1) = \frac{\partial L}{\partial z(t_1)} = z_1 - z^*$$

Несмотря на представление авторами ДУ для весов в стандартном виде, правильной проведение анализа данного уравнения в качестве функционала. Следовательно, для выявления формулы, по которой обновляются веса, необходимо было варьировать наш функционал по $\theta(t)$. После этого вычисление итоговых значений $\Delta \theta(t, i, j)$, умноженных на learning rate, не представляло особых проблем.

Касательно внедрения метода пристрелки для синхронного решения полученной системы необходимо сказать, что именно эта часть повлекла за собой основные проблемы в исследовании. Имеющуюся краевую задачу, состоящую из системы ДУ на $a(t)$ и $z(t)$, а также ранее представленных граничных значений ($a(t_1)$ и $z(t_0) = \text{input}$) было обговорено решать из центра временного промежутка. Изначально задавались случайным образом значения параметров для наших функций в выбранный момент времени t^* - time_middle. Отметим, что обновление параметров происходило по методу Ньютона, в многомерном случае оперирующей следующей формулой:

$$x^{k+1} = x^k - [F(x^k)]^{-1} f(x^k)$$

где x^k - значения параметров на k-ой итерации, $F(x^k)$ - матрица Якоби, $f(x^k)$ - значения невязок на k-ой итерации.

Невязки - это функции, в которых сравнивается полученное при данном параметре значение на границе и правильное. Очевидно, что метод Ньютона при удачно заданных параметрах сходится тогда и только тогда, когда значение невязок нулевое (или максимально близко к нему). Из приведенной формулы видно, что матрица Якоби должна быть квадратной и вырожденной. Так как количество параметров равно $2 \cdot n_{l*}$, где последнее - количество нейронов на среднем слое, число невязок должно в точности соответствовать удвоенному числу нейронов. Более того, определитель матрицы Якоби обязан быть ненулевым. В ходе исследований было обнаружен тривиальный факт, что производные левых невязок по сопряженным параметрам равны 0 (ДУ для $z(t)$, по которое и вычисляются значения для левых невязок, не зависят от $a(t)$). Поэтому количество входных нейронов не должно превышать количество выходных.

Подчеркнем, что если сами значения параметров задаются в функции `prepare_shooting`, то процесс пристрелки происходит в функции `newton_method`, тогда как обновление параметров - в `newton_update`. Далее происходит прохождение континуальных слоев по очереди в двух направлениях (`forward_shooting` и `back_shooting`), на каждом одновременно просчитывая значения для обоих ДУ по методу рунге-кутты для систем. После получения значений на границах, рассчитываются результаты на данной итерации для невязок, после вычисляется матрица производных. Для этих целей поочередно возмущается каждый параметр, определяются новые значения невязок и высчитываются значения производных по формуле:

$$\frac{\partial f_i}{\partial param_i} = \frac{f_i(a_1, a_2, \dots, param_i + 0.1, \dots, z_n) - f_i(a_1, \dots, z_n)}{0.1}$$

После уже стандартно вычисляем по ранее приведенной формуле для метода Ньютона. Заметим, что итерации происходят до тех пор, пока $|x^{k+1} - x^k| \geq 0.1$. Рядовая ситуация для данного подхода - обнаружить, что изначальные значения неспособны привести к сходимости метода. При превышении константного значения потолка количества итераций программа завершалась, выводя соответствующее сообщение и запрашивая новые вектора параметров.

Как только метод Ньютона успешно завершен, сигнал передается в функцию `correct_weights`, которая обновляет значения весов, пользуясь результатами внутри ИНС, полученных после пристрелки последними параметрами перед выходом.

Также отметим, что выходной слой представляет определенное множество нейронов, где каждой букве соответствуют единицы на выходе из некоторых идущих подряд нейронов (кол-во = $\frac{\text{input size}}{26 = \text{alph. size}}$), стоящих на позиции соответствующей этой букве в алфавите.

Все остальное предполагалось провести аналогичным перцептронным образом: поочередно подавать экземпляры из обучающей выборки, а после проверять результаты на тестовой.

Заключение

Итоги исследования

Подводя итоги работы, необходимо сказать, что говорить о полном выполнении цели невозможно.

С одной стороны, имеются разнообразные комплексные выводы по работе перцептрона. Был многократно проведен анализ работы данной ИНС с точки зрения времени и успешности при разных количествах слоев:

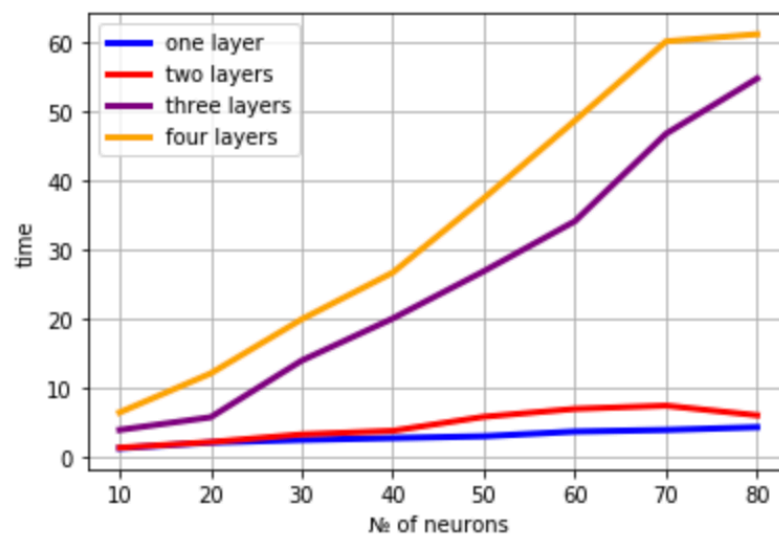


Рисунок 1 - График времени работы перцептрона в зависимости от количества слоев и нейронов

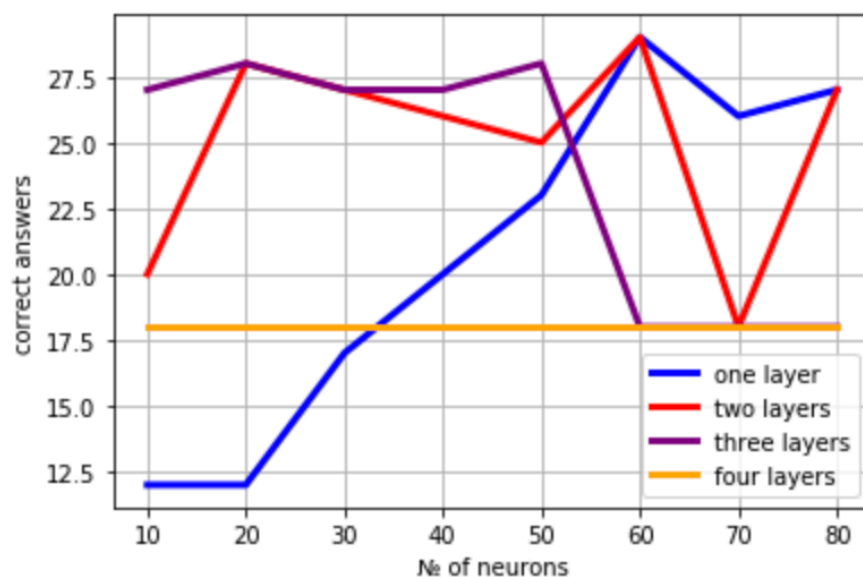


Рисунок 2 - График успешности перцептрона в зависимости от количества слоев и нейронов

Несмотря на сильные колебания, было выявлено общее поведение. Согласно первоначальному анализу было решено прекратить исследования для трех и четырех слоев ввиду крайне высокой неэффективности по времени и сомнительным результатам в плане узнавания букв. Можно предположить, что ввиду малой выборки имелись разные категории “трудности” букв: чем менее эффективно устройство перцептрона, тем менее трудную категорию он может распознать.

Анализ работы перцептрона с одним и двумя скрытыми слоями представлен ниже:

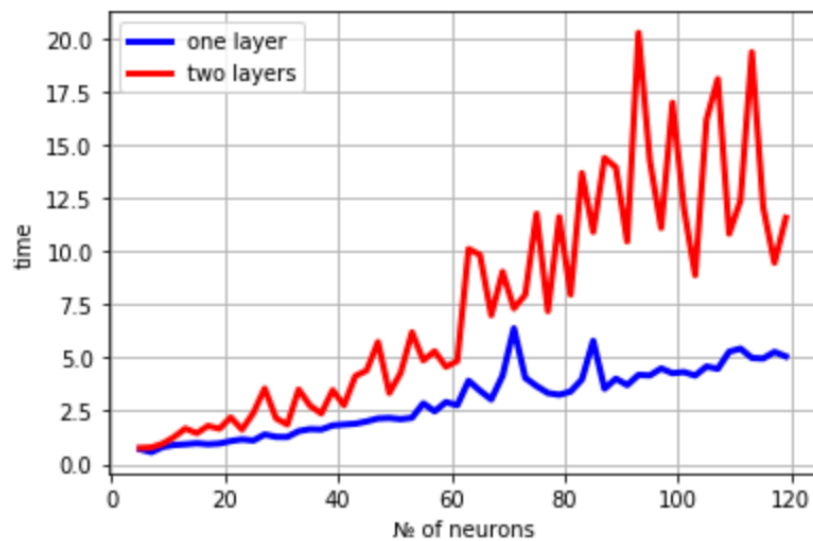


Рисунок 3 - График времени работы перцептрона из одного и двух скрытых слоев

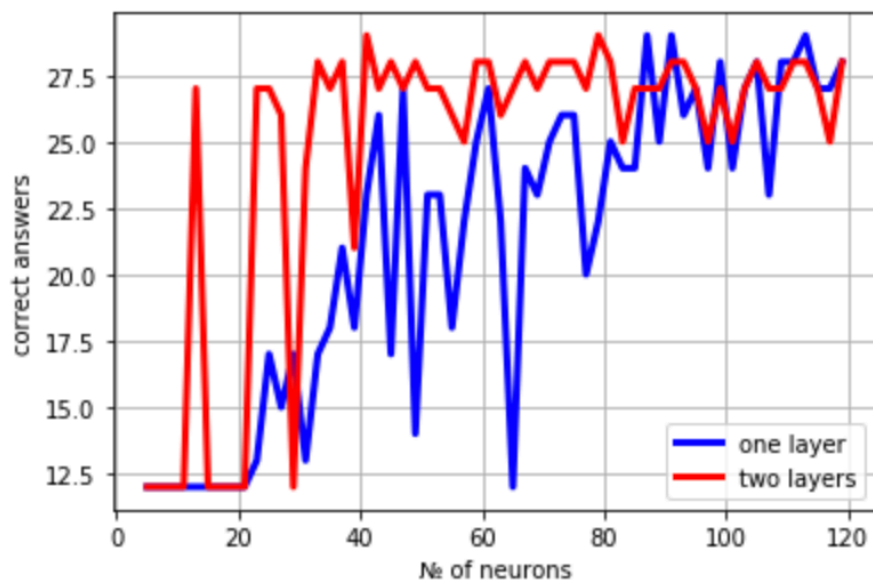


Рисунок 4 - График успешности работы перцептрона из одного и двух скрытых слоев

Можно сразу отнести к минусам перцептрона высокую амплитуду значений как для времени, так и для правильных ответов, которая сохранялась от эксперимента к эксперименту при разных составлениях обучающих и тестовых выборок. Подобное поведение не позволяет с высокой надежностью выбрать оптимальную конструкцию. Тем не менее, в контексте времени можно выявить определенную корреляцию. Время работы перцептрона из двух скрытых слоев заметно больше, чем из одного, при этом их значения уравниваются для достаточно больших значений числа нейронов в слое.

После целого ряда одинаковых экспериментов были выявлены поля, в которых наиболее часто обе структуры выдают более 85% правильных ответов. После в рамках этих полей проводились многочисленные вычисления с целью установки среднего в данных диапазонах:

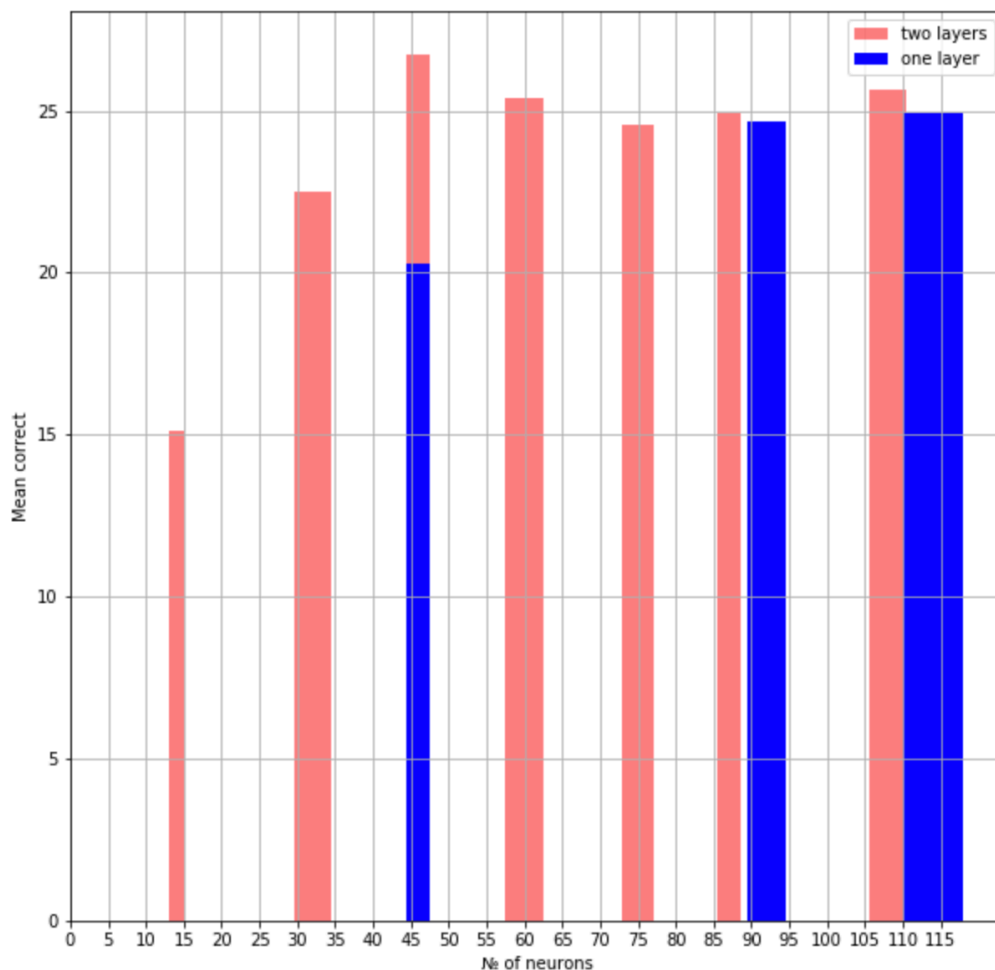


Рисунок 5 - Диаграмма правильных ответов при разных значениях числа нейронов

Нельзя не заметить, что несмотря на отдельные успехи, точность однослойного перцептрона меньше: почти на каждом промежутке двухслойный более эффективен. Однако высокие перепады присутствуют у обеих структур - это свойство перцептрона, от которого никуда не деться. Более того, ни одна из реализаций ни разу не показала 100% эффективности, не справляясь с наиболее трудной категорией.

Отдельно добавим про результаты структуры, описанной в ходе эксперимента и представленной на внутренней защите: ее успешность на финальных тестах превысила 90%, а на контрольных составила 100%. Однако это обуславливается маленькой выборкой и удачными тестами - колебательный характер перцептрона был описан ранее.

В то же время, все эти результаты не столь показательны без сравнения с Neural ODE. К срокам реализации так и не был представлен исправно работающий код. Тогда как все математические расчеты и алгоритмы, соединяющие в себе многочисленные методы и концепции, а также функции и общая структура были реализованы, видоизменение стандартного подхода в сторону метода пристрелки открыло на своем пути целый ряд локальных проблем - появлении вырожденных матриц Якоби, несходимость метода Ньютона, несопоставимость размеров входного и выходного слоя, технические проблемы.

Однако заметим, что программа действительно находится на финальном этапе корректировки, проделанную работу в этом направлении возможно представить. Несмотря на невыполненный внутренний дедлайн, реализация функционирующего нейродифференциала будет основной задачей в ближайшее время. Это особенно важно как для демонстрации потраченных усилий и эффективности нового подхода, так и для дальнейших работ.

Переходя к последнему, нельзя не упомянуть весь тот спектр изученных материалов, которые были представлены в изученной литературе. Был проделан комплексный подход для подготовки многоуровневого исследования. Сохраняя глобальную цель - применение нейродифференциальных уравнений для бифуркационного анализа работы реактора - видится категорически важным справиться с одним из ключевых этапов: представить итоговую версию кода, проанализировать ее с точки зрения заявленных критериев и начать применение для решения обговоренных рабочих задач.

Библиография

1. Chen, R. T. Q. [и др.] PyTorch Implementation of Differentiable ODE Solvers [Электронный ресурс] / MIT License — URL: <https://github.com/rtqichen/torchdiffeq>, свободный. (дата обращения 08.02.2021)
2. Homburg, A. J. Bifurcation Theory [Электронный ресурс]: учебник для вузов / Universiteit van Amsterdam, Vrije Universiteit Amsterdam — URL: <https://staff.fnwi.uva.nl/a.j.homburg/BifTh/bifurcatietheorie.pdf>, свободный. (дата обращения 29.01.2021)
3. Krose, B. An Introduction to Neural Networks [Электронный ресурс] / B. Krose, P. v. d. Smagt / University of Amsterdam — Электр. дан. — 1996, 8 изд., свободный. (дата обращения 08.02.2021)
4. McCann, C. Bifurcation Analysis of Non-linear Differential Equations [Электронный ресурс] / Dr. Vasiev — URL: https://pcwww.liv.ac.uk/~bnvasiev/Past%20students/Caitlin_399.pdf, свободный. (дата обращения 08.02.2021)
5. Nathan Kutz, J. Advanced Differential Equations: Asymptotics & Perturbations [Электронный ресурс]: курс лекций / University of Washington — URL: <https://faculty.washington.edu/kutz/am568/am568.html>, свободный. (дата обращения 08.02.2021)
6. Sinai, J. Understanding Neural ODE's [Электронный ресурс] — URL: <https://jontysinai.github.io/jekyll/update/2019/01/18/understanding-neural-odes.html>, свободный. (дата обращения 08.02.2021)
7. Neural Ordinary Differential Equations [Электронный ресурс]: науч. статья / R. T. Q. Chen, Y. Rubanova, J. Bettencourt, D. Duvenaud / University of Toronto, Vector Institute — Электр. дан. — 2019, свободный. (дата обращения 08.02.2021)
8. PyTorch Documentation [Электронный ресурс]. URL: <https://pytorch.org/docs/stable/index.html>, свободный. (дата обращения 08.02.2021)
9. Амосов А.А., Дубинский Ю.А., Копченкова Н.В. Вычислительные методы для инженеров: Учеб. пособие. — М.: Высш. шк., 1994. — 544 с.
10. Арнольд В.И. Обыкновенные дифференциальные уравнения [Электронный ресурс] — Московский центр непрерывного математического образования — 2012 — 341с. — ISBN: 978-5-4439-2007-8 — URL: <https://mccme.ru/arnold/books/odu-12.pdf>, свободный. (дата обращения 08.02.2021)

11. Садчикова, Н. В. Разработка архитектуры нейронной сети с динамической глубиной: 02.04.02: защищена 20.06.19 / Садчикова Н. В. - Казань, 2019. - 105 с.
12. Сурцуков, М. Знакомство с Neural ODE [Электронный ресурс] / Open Data Science — URL: <https://habr.com/ru/company/ods/blog/442002/>, свободный. (дата обращения 08.02.2021)
13. Гельфанд, И. М. Вариационные исчисления [Электронный ресурс]: учебник для вузов / И. М. Гельфанд, С. В. Фомин / под ред. С. М. Половинкина — М: Гос. изд. физ.-мат. лит., 1961 — URL: <https://yadi.sk/i/31UaANeZkIC9Yw>, свободный. (дата обращения 08.02.2021)
14. Колчин, В. А. Бифуркации и автоколебания в ядерных реакторах с линейной обратной зависимостью [Электронный ресурс] / под ред. О. А. Казачковского / Электронная библиотека “История Росатома” // Атомная энергия — 1982 — №6 (52) — С. 393 - 398 — URL: http://elib.biblioatom.ru/text/atomnaya-energiya_t52-6_1982/go,0/, свободный. (дата обращения 08.02.2021)
15. Шихов, С. Б. Исследование устойчивости стационарных температурного и нейтронного полей активной зоны реактора в распределенных моделях [Электронный ресурс] / С. Б. Шихов, А. В. Крянев / под ред. Н. Н. Пономарева-Степного / Электронная библиотека “История Росатома” // Атомная энергия — 1990 — №2 (68) — С. 85 - 90 — URL: http://elib.biblioatom.ru/text/atomnaya-energiya_t68-2_1990/go,13/, свободный. (дата обращения 08.02.2021)
16. Эльсгольц, Л. Э. Дифференциальные уравнения и вариационное исчисление: учебник для вузов — URL: http://phys.nsu.ru/balakina/El%27sgol%27dz_Dif_ur_i_var_isch.pdf, свободный. (дата обращения 08.02.2021)
17. Инженерная химия. Химические реакторы [Электронный ресурс] / Ю.Н. Жилин, А.Н. Зарубина, Г.Л. Олиференко, А.Н. Иванкин — М. : МГУИ, 2016 — URL: https://mf.bmstu.ru/UserFiles/File/7_IVANKIN/HIM_REAKTORI_Zhil-Zar-Ol-Iv2016.pdf, свободный. (дата обращения 08.02.2021)
18. Нейронные обыкновенные дифференциальные уравнения [Электронный ресурс]: науч. обзоры / под ред. А. Г. Дьяконова — URL: <https://dyakonov.org/2019/06/13/нейронные-обыкновенные-дифференциал/>, свободный. (дата обращения 08.02.2021)

19. Нейронные сети на службе безопасности АЭС [Электронный ресурс]: многопред. науч. журн. / Страна Росатом — Электрон. журн. — URL: <https://strana-rosatom.ru/2017/01/23/нейронные-сети-на-службе-безопасност/>, свободный. (дата обращения 08.02.2021)