Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)
Кафедра Штучного інтелекту
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Розпізнавання зображень згортковими нейронними мережами
з використанням різних активаційних функцій
(тема)

Виконав:
студент 2 курсу, групи СШІм-22-2
Піхуля Д.О.
(прізвище, ініціали)

Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)
Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)
Освітня програма Системи штучного інтелекту
(повна назва спеціалізації)
Керівник ст. викл. Попов С.В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____ В.О. Філатов
(підпис) (прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Штучного інтелекту
(повна назва)

Рівень вищої освіти другий (магістерський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма Системи штучного інтелекту
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«_____» _____ 20 ___ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Піхулі Дмитру Олексійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Розпізнавання зображень згортковими нейронними мережами з використанням різних активаційних функцій

затверджена наказом університету від 1 квітня 20 24 р. № 260Ст
2. Термін подання студентом роботи до екзаменаційної комісії 10 червня 20 24 р.
3. Вихідні дані до роботи Наукові публікації з машинного навчання, інформація з машинного навчання з ресурсів мережі Internet, документація Keras, документація TensorFlow

4. Перелік питань, що потрібно опрацювати в роботі
1) Визначити конкретні набори даних та моделі для експериментів
2) Визначити гіперпараметри для проведення експериментів
3) Визначити набір стандартних АФ для порівняння
4) Визначити кандидатів для нових АФ
5) Методика оцінки ефективності нових АФ
6) Експериментальна оцінка ефективності нових АФ
7) Аналіз результатів експериментів

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	01.04.2024	виконано
2	Вибір потенційних категорій нових АФ	5.04.2024 – 12.04.2024	виконано
3	Попередні експерименти із обраними	15.04.2024 – 19.04.2024	виконано
	категоріями АФ		
4	Розробка методики оцінки ефективності АФ	22.04.2024 – 25.04.2024	виконано
5	Розробка моделей та обирання варіантів	26.04.2024 – 30.04.2024	виконано
	гіперпараметрів для експериментів		
6	Детальні експерименти із новими АФ	1.04.2024 – 30.04.2024	виконано
7	Аналіз результатів експериментів	17.05.2024 – 20.05.2024	виконано
8	Оформлення пояснювальної записки	21.05.2024 – 5.06.2024	виконано
9	Захист кваліфікаційної роботи	13.06.2024	

Дата видачі завдання 1 квітня 20 24 р.

Студент _____
(підпис)

Керівник роботи _____ ст.викл. Попов С.В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 91 с., 46 рис., 8 табл., 1 дод., 22 джерела.

АКТИВАЦІЙНА ФУНКЦІЯ, ЕФЕКТИВНІСТЬ, ЗГОРТКА, ЗГОРТКОВА НЕЙРОННА МЕРЕЖА, ЗОБРАЖЕННЯ, КЛАСИФІКАЦІЯ.

Об’єкт дослідження – розпізнавання зображень згортковими нейронними мережами.

Предмет дослідження – ефективність використання деяких активаційних функцій при розпізнаванні зображень згортковими нейронними мережами.

Мета роботи – дослідження існуючих активаційних функцій, розглядання можливих нових активаційних функцій, та оцінка їх ефективності в задачі класифікації зображень з використанням згорткових нейронних мереж.

Методи дослідження – аналіз літератури та визначення закономірностей, що можуть робити деякі існуючі активаційні функції успішними, та відповідне створення можливих нових активаційних функцій, із проведенням експериментів з використанням реальних даних для оцінки їх ефективності.

У роботі були сформовані прості емпіричні правила для створення нових активаційних функцій, та створена і протестована низка нових активаційних функцій. Загалом було випробувано 37 різновидів нових функцій та порівняно їх із 35 існуючими. Були проведені експерименти із наборами даних CIFAR-10, MNIST, Fashion-MNIST. Два варіанта нових адаптивних активаційних функцій $ASiSoAsinh$ та $AGeSoAsinh$ показали стійку високу якість класифікації зображень CIFAR-10, що на експериментальній моделі була вищою порівняно із популярними функціями, такими як $ReLU$, $SiLU$, $GeLU$, $Swish$, $PreLU$, та ін.

ABSTRACT

Master's thesis contains: 91 pp., 46 fig., 8 tabl., 1 ann., 22 references.

ACTIVATION FUNCTION, CLASSIFICATION, CONVOLUTION, CONVOLUTIONAL NEURAL NETWORK, EFFECTIVENESS, IMAGE.

The object of research is image recognition using convolutional neural networks.

The subject of research is the study of effectiveness of using certain activation functions for image recognition using convolutional neural networks.

The goal of the work is to study existing activation functions and consider potential new activation functions with measuring their effectiveness in the image classification task using convolutional neural networks.

Research methods include the analysis of literature for identifying patterns that might make certain activation functions successful, creating new activation functions, conducting experiments using real data sets and evaluating their effectiveness.

This work considers prior research works, which review existing activation functions as well as propose new functions. Some simple rules were formulated, which could be useful for creating new potentially effective activation functions, and some new activation functions were proposed, and their effectiveness compared to other activation functions. Overall, 37 variants of new activation functions were considered and compared with 35 existing ones. The experiments were conducted with the CIFAR-10, MNIST, and Fashion-MNIST datasets. Two variants of new adaptive activation functions $ASiSoAsinh$, and $AGeSoAsinh$ have demonstrated a stable high accuracy of CIFAR-10 images classification, which have outperformed in terms of accuracy other popular activation functions, such as $ReLU$, $SiLU$, $GeLU$, $Swish$, $PreLU$, and others.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ААФ – адаптивна активаційна функція;

АФ – активаційна функція;

БШП – багатошаровий перцептрон;

ЗНМ – згорткова нейронна мережа;

РЕ – ранг ефективності;

ШНМ – штучна нейронна мережа.

ВСТУП

У ці часи ми є свідками того, як розвиток та використання різноманітних систем штучного інтелекту виявляє вибуховий зріст у, мабуть, майже усіх галузях людської діяльності. Це включає широкий спектр застосування, включаючи промислові та побутові задачі, медицину, дозвілля, комунікації, транспорт, тощо. При цьому, враховуючи перспективи, що може дати застосування штучного інтелекту, можна стверджувати, що розвиток технологій штучного інтелекту мабуть знаходиться в його початку.

В переважній більшості в основі цих технологій використовується машинне навчання із штучними нейронними мережами. Концепція штучних нейронних мереж натхненна тим, як нервова система тварин здатна до ефективного розв’язання багатьох задач у природі. До того ж, як виявилось, багато з таких задач, які для людини або тварин виявляються тривіальними, дуже складно або практично не можливо надійно вирішувати за допомогою традиційних методів програмування, де алгоритм вирішення задачі складається людиною-програмістом, ефективно вирішуються за допомогою штучних нейронних мереж. Наприклад, проста задача вирішення чи на зображенні зображений кіт чи собака здається практично неможливою для надійного вирішення шляхом розробки відповідного алгоритму у зв’язку з необхідністю створити правила, які б враховували надвелику кількість можливих комбінацій зображень котів та собак та їх відповідних аспектів, які потрібно врахувати для визначення класу зображення.

Говорячи про аспекти задач, які стоять перед системами штучного інтелекту, розпізнавання, обробка, та генерація зображень є великою частиною таких задач. Це зумовлено тим, що зір є одним із найважливіших чуттів людини, та, подібно до людини, системи штучного інтелекту стоять перед необхідністю вирішувати широкий спектр задач, де людина опирається на це чуття. Одним із видів нейронних мереж є згорткові

нейронні мережі. Ці мережі є ефективним способом роботи із зображеннями, та у деяких аспектах їхня робота схожа із тим, як зображення обробляються у мозку. Подібно до БШП, згорткова нейронна мережа може містити довільну кількість шарів, та кожен шар виявляє деякі патерни у вхідному зображенні. Перші шари виявляють найбільш прості аспекти, таки як лінії під різними кутами, кольори, текстури, та ін., і більш глибокі шари можуть виявляти більш складні ознаки, такі як окремі частини об’єктів, цілі об’єкти, та ін.

Подібно до БШП, окремі згортки у згорткової мережі можна також вважати аналогами нейронів, які концептуально працюють таким самим чином, хоча, на відміну від перцептрону мають обмежений діапазон простору зображення (або даних з попереднього шару), з якими він працює, та натомість вони обробляють все зображення за принципом ковзаючого вікна. Як і в перцептроні, задля здатності апроксимувати складні функції, обчислення згортки потребує використання активаційної функції. Ця робота досліджує ефективність навчання та роботи згорткових нейронних мереж в залежності від використання різних активаційних функцій.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1 Принцип роботи згорткових нейронних мереж

Штучні нейронні мережі є одним із класів моделей машинного навчання, у яких процес вирішення певної задачі відбувається за допомогою мережі взаємозв’язаних обчислювальних блоків (штучних нейронів), подібно тому, як це відбувається у біологічних нейронних мережах. На цей час було розроблено багато моделей штучних нейронних мереж, наприклад, мережі векторного квантування, неокогнітрон, перцептрон, та ін. Однією з дуже успішних видів моделей, які зарекомендували себе як ефективний засіб вирішення широкого спектру задач є багатошаровий перцептрон (БШП), та його модифікації.

Багатошаровий перцептрон, подібно до перцептрону, складається із шарів, у яких кожен нейрон приймає сигнал від кожного нейрону попереднього шару. При цьому кожен із входів має індивідуальне числове значення, асоційоване з відповідним входом (ваги відповідних входів). Значення на виході із нейрону обчислюється в три етапи:

– значення, яке подається з виходу відповідного нейрону попереднього шару примножується на значення вагів відповідного входу нейрону, та як результат отримується зважене значення кожного із входів нейрону;

– обчислюється сума усіх зважених значень входів нейрону. Слід зазначити, що ця сума є лінійно залежною від значень входів нейрону;

– значення, яке прийме вихід нейрону у такій мережі, визначається як значення, яке приймає деяка функція, яка приймає на вхід зважену суму входів, обчислену вище. Цю функцію називають активаційною функцією.

Це схематично можна відобразити як показано на рисунку 1.1.

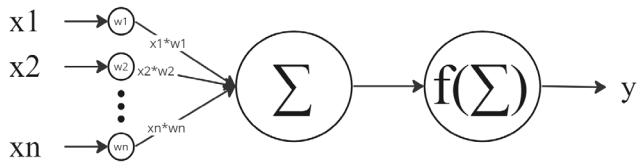


Рисунок 1.1 – Обчислення значення виходу нейрона у багатошаровому перцептроні

Проводячи паралель із біологічними нейронами, активаційна функція фактично визначає коли нейрон «активується», залежно від зважених значень, що надаються на входи цього нейрону. При цьому важливо, щоб ця функція була нелінійною, що дозволить багатошаровій мережі апроксимувати складні залежності між вхідними та вихідними значеннями мережі.

Загалом, модель БШП вважається здатною до апроксимації довільних функцій [1]. Проте на практиці процедура створення відповідної конкретної моделі для конкретних потреб не є формалізованою, та успішність створення придатної моделі залежить від багатьох факторів, серед яких є належне налаштування гіперпараметрів моделі. Це може бути, серед іншого, обирання кількості шарів у мережі, ширини (кількості нейронів) кожного шару, а та також темп навчання, алгоритм оптимізації, та ін. Одним із важливих гіперпараметрів є функція активації, що використовується в нейронах мережі. На практиці вибір функції активації може радикальним образом змінювати здатність моделі до навчання на відповідній вибірці даних, а також призводити до різного рівню здатності виконувати відповідні задачі.

Згорткові нейронні мережі (ЗНМ) є спеціалізованим типом глибоких нейронних мереж, які знайшли широке застосування в обробці та аналізі зображень. Основна їх особливість полягає в ефективності виявлення візуальних ознак на різних рівнях абстракції, що дозволяє автоматично класифікувати та інтерпретувати зображення. Така схема їх роботи нагадує

механізм зору живих організмів, який також має ієрархічну побудову, та здатний розпізнавати більш складні аспекти вхідних зображень на більш глибоких рівнях мережі.

ЗНМ були інспіровані неокогнітроном [2], а також мають загальні риси з БШП. На відміну від перцептрону, кожен шар ЗНМ приймає на вхід багатовимірний масив, залежно від типу даних, що обробляє мережа. Наприклад, у разі обробки монохромного зображення, вхідний шар мережі буде отримувати двовимірний масив, а у разі кольорового зображення, тривимірний (третій вимір у такому разі використовується задля представлення двомірного зображення у трьох кольорових каналах).

Типова ЗНМ складається з наступних ключових частин:

– згорткові шари. Кожен шар складається з набору фільтрів, які сканують вхідні зображення (або активаційні карти, у разі більш глибоких шарів) за принципом ковзаючого вікна, та завдяки процесу навчання мережі здатні визначати специфічні ознаки у вхідних даних, таких як краї, кути або текстури. Завдяки ковзаючим згорткам, мережа здатна виявити ознаки незалежно від їхнього місцеположення на зображенні;

– активаційні функції (АФ). Результат згортки є лінійною функцією від відповідних значень вхідних пікселів зображення. Проте, подібно до механізму роботи БШП, такий результат згортки обробляється за допомогою АФ для формування активаційної карти. АФ часто практично розглядається як частина згорткового шару. Таким чином, кожна згортка згорткового шару, яка отримала зображення на вхід, генерує відповідний двовимірний масив, який відображає використання відповідного фільтру для всіх пікселів вхідного зображення, та застосування АФ до результатів згортки. Такий двовимірний масив згенерований кожною згорткою називається активаційною картою;

– шари об’єднання (pooling layers). Такі шари виконують операцію зменшення розмірності даних за обраним правилом, наприклад, вибірково зберігаючи лише найбільш значні ознаки, що відповідають до більших

значень в активаційній карті (у разі max pooling layer). Використання таких шарів допомагає зменшити обчислювальне навантаження та запобігти перенавчанню;

– повнозв’язні шари. В залежності від задачі, що вирішує ЗНМ, вона може включати також звичайні повнозв’язні шари, які використовуються в звичайному БШП. Наприклад, для задачі класифікації, наприкінці згорткової мережі підключають один або кілька повнозв’язних шарів, які об'єднують всі виявлені ознаки для прийняття відповідного рішення.

Перевагами ЗНМ в задачах обробки зображень, порівняно з БШП є такі аспекти:

– кожен нейрон (згортка) наступного шару з’єднаний тільки з невеликою кількістю нейронів попереднього шару, що прискорює навчання та вивід за навченою мережею. При цьому такі з’єднання відбуваються тільки із відповідними нейронами попереднього шару, які є близькими в просторі зображення до просторової позиції відповідного нейрону в наступному шарі, що дозволяє мінімізувати набор корисних з’єднань та відповідних параметрів моделі, що підлягають тренуванню, та при цьому мати здатність виявляти окремі ознаки зображення, які можна виявити у певному локальному діапазоні вхідного зображення;

– завдяки тому, що навчання та вивід мережі працює за принципом ковзаючого вікна (розглядаючи кожен нейрон як такий, що може використовуватись для довільної позиції зображення), мережа, що вивчила якусь ознаку, здатна розпізнати її у будь-якої позиції вхідного зображення;

– загалом, зважаючи на аспекти викладені в попередніх пунктах, мережа має набагато меншу кількість змінних параметрів, що навчаються, що робить модель більш компактною, та, між іншим, допомагає запобігати проблемі перенавчання.

Слід зазначити, що, зважаючи на деякі споріднені принципи роботи ЗНМ до БШП (багатошарова структура, агрегація сигналу з попереднього шару з використанням АФ для визначення виходу нейрону), принципово у

ЗНМ можна використовувати ті самі типи АФ, що використовуються в БШП.

Проте, попри схожості роботи ЗНМ до БШП, суттєві особливості побудови та функціонування ЗНМ наведені вище можуть мати певний вплив на ефективність роботи тих чи інших АФ само із ЗНМ. Тому важливо між іншим дослідити ефективність роботи різних АФ само із ЗНМ, що може дати корисну інформацію для створенні більш ефективних моделей, що базуються на ЗНМ.

1.2 Огляд деяких активаційних функцій

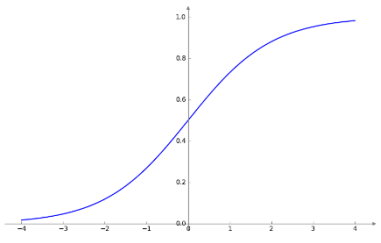
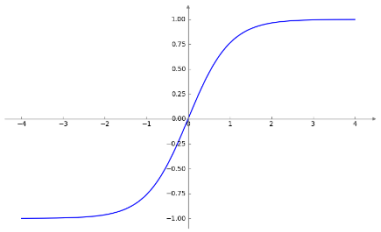
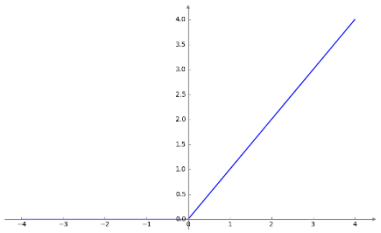
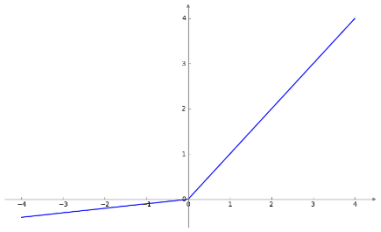
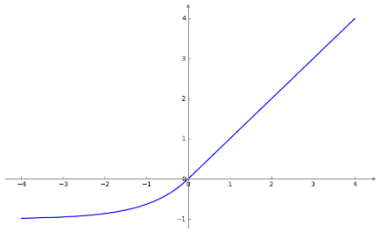
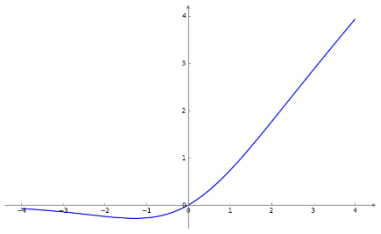
Передусім, слід зазначити, що задля використання переваг багатошарової структури мережі, АФ має бути нелінійною. Внесення нелінійності на кожному шарі є важливою функцією присутності АФ в мережі в цілому.

За відсутності АФ, або за наявності АФ, яка є лінійною, кожен вихід мережі буде мати лінійну залежність від її входів, що нівелювало би наявність будь якої кількості прихованих шарів в мережі та суттєво зменшувало б здатність мережі до вирішування складних задач. Натомість, внесення нелінійності за допомогою нелінійної АФ дозволяє мережі відтворювати все більш складні відображення із збільшенням кількості шарів в мережі.

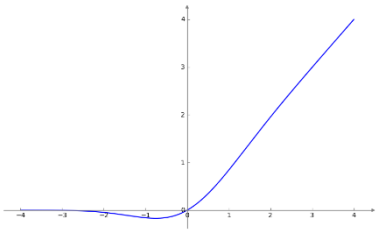
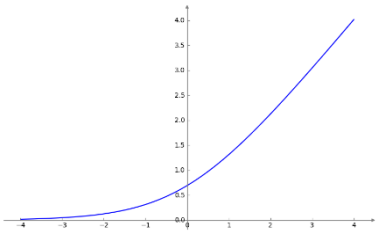
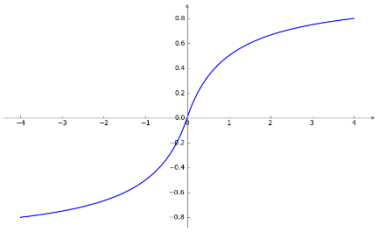
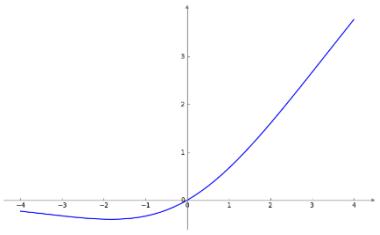
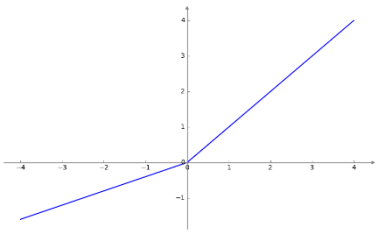
До того ж, слід зауважити, що у зв’язку з механізмом навчання мережі методом градієнтного спуску із зворотнім поширенням похибки, важливо щоб АФ була безперервною, а також такою, що диференціюється у кожній її точці (за деякими практично можливими виключеннями, такими як нульова точка в ReLU).

У таблиці 1.1 наведений огляд деяких типових активаційних функцій.

Таблиця 1.1 – Перелік деяких популярних активаційних функцій

Активаційна функція	Формула	Графік
Логістична сигмоїда (Logistic Sigmoid)	$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}}$	
Гіперболічний тангенс (Tanh)	$f(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$	
ReLU	$f(x) = \max(0, x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases}$	
Leaky ReLU	$f(x) = \begin{cases} x, & x \geq 0 \\ \alpha x, & x < 0 \end{cases}$	
ELU	$f(x) = \begin{cases} x, & x \geq 0 \\ \alpha(e^{x} - 1), & x < 0 \end{cases}$	
SiLU [3]	$f(x) = x\sigma(x) = x\frac{1}{1 + e^{-x}}$	

Продовження таблиці 1.1

GELU [4, с. 2]	$f(x) = x\Phi(x) = x\frac{1}{2}\left[1 + \operatorname{erf}(x/\sqrt{2})\right]$	
Softplus	$f(x) = \log(1 + e^x)$	
Softsign	$f(x) = \frac{x}{\lvert x\rvert + 1}$	
Swish	$f(x) = x\sigma(\beta x) = x\frac{1}{1 + e^{-\beta x}}$	
PReLU	$f(x_i) = \begin{cases} x_i, & x_i \geq 0 \\ \alpha_i x_i, & x_i < 0 \end{cases}$	

Варто зазначити, що, як ми можемо бачити у таблиці 1.1, ми можемо виділити певні групи активаційних функцій. Один клас функцій є S-подібними функціями. Це такі функції, як Logistic Sigmoid, Tanh, та Softsign. Ці функції, зазвичай фактично стискають необмежений діапазон аргументу функції у певний обмежений діапазон значень. Інший, доволі великий, клас функцій – це ReLU-подібні функції, такі як ReLU, Leaky ReLU, SiLU, GELU, Swish, та PreLU. У цих функціях, їх частина, що відповідає негативним

значенням x, тяжіє до різної міри близькості до осі x, та частина з позитивними значеннями x тяжіє до лінійної функції $f(x) = x$.

Іншим аспектом, за яким можна розділити АФ – це те, що деякі функції залежать тільки від значення x, що фактично є зваженою сумою синаптичних вагів. Проте є група функцій, які мають додаткові параметри, що можна вважати такими, які належать до функції як такої, та можуть навчаються згідно із тренувальними даними подібно до навчанню синаптичних вагів мережі. Наприклад, параметр α_i, що навчається у функції PReLU регулює нахил негативної частини функції, та у функції Swish також є параметр, що навчається β, який регулює форму функції навколо значення $x = 0$.

1.3 Огляд літератури

Оскільки ми не можемо виключати, що ефективність різних АФ може змінюватись залежно від вирішуваної задачі та відповідної архітектури мережі, для мети цієї роботи ми концентруємося на розгляді існуючих робіт, в яких досліджуються АФ саме в контексті ЗНМ для розпізнавання та обробки зображень. Втім, слід зазначити, що оскільки розпізнавання зображень є доволі популярною потребою, коли йдеться про системи штучного інтелекту, та враховуючи той факт, що ЗНМ є одним із основних та фактично стандартних методів роботи з зображеннями за допомогою нейронних мереж, ми можемо бачити, що майже кожна стаття, яка розглядає АФ, розглядає серед іншого і роботу з використанням саме ЗНМ.

1.3.1 Популярні АФ, їх особливості та ефективність

В процесі пошуку літератури за темою було знайдено значну кількість існуючих робіт, що присвячені як окремим активаційним функціям, так і

огляду наборів активаційних функцій різного масштабу. Розглянемо декілька таких робіт, що розглядають широкий набір АФ.

В роботі [5] робиться огляд значного набору активаційних функцій, таких як: Sigmoid, Hardsigmoid, SiLU, dSiLU, Tanh, Hardtanh, Softmax, Softplus, Softsign, ReLU, LReLU, PReLU, RReLU, SReLU, ELU, PELU, SELU, Maxout, Swish, EliSH, HardELiSH. Ця робота не містить власного експериментального дослідження ефективності чи особливостей цих функцій, проте натомість підсумовує та викладає загальну інформацію про ці активаційні функції, яка існувала на час подання цієї роботи. Це включає таку інформацію як загальний опис функцій, мету їх створення або переваги, сферу їх типового використання, та деякі відповідні історичні довідки. Також вона містить інформацію про те які функції знайшли собі місце в деяких популярних моделях нейронних мереж (наприклад за результатами змагань, таких як ImageNet та інші).

Робота [6] також містить огляд деяких активаційних функцій, таких як: Step, Linear Activation, Sigmoid, Tanh, Softsign, ReLU, Leaky ReLU, Maxout, Softplus, та Swish. На відміну від попередньої, ця робота також детально розглядає проблеми притаманні різним АФ, а також дає деякі емпіричні рекомендації щодо використання окремих функцій. Також ця робота проводить експериментальне порівняння цих функцій, використовуючи точність та час тренування як критерії. Це робиться за допомогою тренування класифікатора на наборі даних CIFAR-10 на мережі з двома згортковими шарами. За результатами експерименту, точність натренованої моделі за результатом 25 епох тренування була у діапазоні від 0.6166 (для Sigmoid) до 0.7295 (для Leaky ReLU). Ці результати створювалися як усереднені за результатами трьох запусків [6].

Доклад [7] містить дослідження ефективності тренування простої згорткової мережі на наборі даних CIFAR-10 з використанням п’яти різних АФ. Результат, який отримали автори можна бачити на рисунку 1.2.

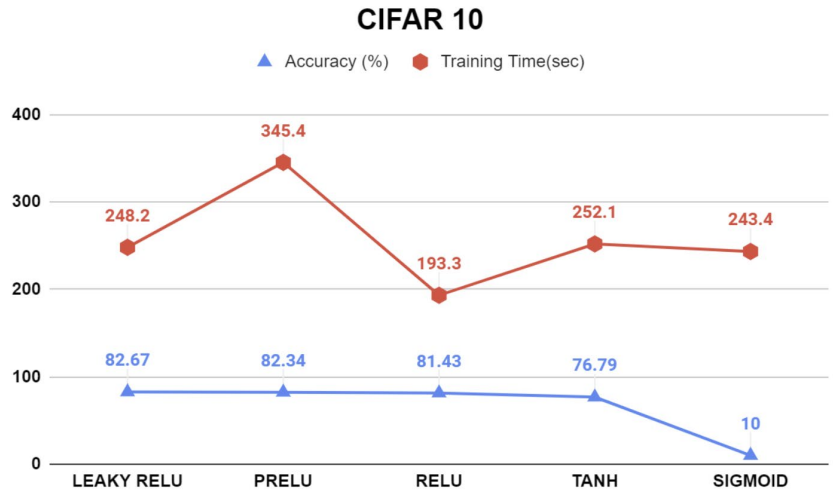


Рисунок 1.2 – Ефективність тренування згорткової мережі на наборі даних CIFAR-10 за результатами дослідження [7, с. 6]

Дуже ретельно розглядається широкий діапазон АФ у роботі [8]. В ній детально розглядають різні класи активаційних функцій, історію та мотиви їх розробки, включаючи опис проблем притаманним тим чи іншим класам АФ, експериментально оцінюється ефективність роботи різних функцій. Це включає такі класи активаційних функцій, як:

– сигмоїдні функції (Logistic Sigmoid, Tanh);
– функції, які базуються на ReLU;
– експоненційні функції, такі як ELU;
– адаптивні АФ, які містять додаткові параметри, що навчаються;
– інші функції, які були запропоновані нещодавно.

Серед проблем логістичних функцій Sigmoid і Tanh вказуються проблеми із зникаючим градієнтом, складнощі із тренуванням глибоких мереж, зв’язані з насиченням мережі. Також описуються деякі інші модифікації цих АФ, такі як масштабований гіперболічний тангенс (sTanh), параметрична сигмоїда (Parametric Sigmoid Function), масштабована сигмоїда (sSigmoid), пеналізований тангенс (pTanh), зашумлені АФ, та ін. [8, с. 2-4].

Функція ReLU вказана як така, що вирішує проблеми АФ Logistic Sigmoid та Tanh, проте їй притаманна інша проблема зі зникаючим градієнтом у негативній частині функції. Проте було запропоновано значну кількість модифікацій, які між іншим спрямовані на подолання цієї проблеми. Це такі АФ, як Leaky ReLU (LRELU), Parametric ReLU (PReLU), Randomized ReLU (RReLU), Concatenate ReLU (CReLU), Parametric Tan Hyperbolic Linear Unit (P-TELU), Flexible ReLU (FReLU), Random Translation ReLU (RTReLU), Average Biased ReLU (AB-ReLU), DualReLU, PairedReLU, Displaced ReLU (DisReLU), Bendable Linear Unit (BLU), Lipschitz ReLU (L-ReLU) [8, с. 4-6]. Також розглядаються такі питання, як обмежена нелінійність функції ReLU та відповідні пропозиції, такі як S-shaped ReLU (SReLU), та Multi-bin Trainable Linear Unit (MTLU). Інше питання, яке окремо розглядається – це необмежений діапазон функції ReLU та відповідна функція Bounded ReLU (BreLU), яка вирішує цю проблему та покращує ситуацію з нестабільністю тренування, яке може виникати через необмежений діапазон функції [8, с. 6].

Іншим різновидом модифікацій ReLU, який розглядається в роботі [8] є експоненційні АФ, які спрямовані на вирішення зникаючого градієнту в ReLU. Це такі функції, як Exponential Linear Unit (ELU), Scaled ELU (SELU), Parametric ELU (PELU), Continuously differentiable ELU (CELU), Multiple PELU (MPELU), Shifted ELU (ShELU), Fast ELU (FELU), Parametric Deformable ELU (PDELU), Exponential Linear Sigmoid SquasHing (EliSH), та HardELiSH [8, с 6-7].

Також робота [8] розглядає окремий клас АФ, що складається з адаптивних АФ. Адаптивні АФ – це АФ, які містять власні параметри, що навчаються згідно з відповідним навчальним набором даних. Цей клас АФ не є взаємовиключним із попередньо розглянутими класами, що були виділені за критерієм форми функції та окремих особливостей їх функцій. Тобто деякі з розглянутих АФ є також адаптивними [8, с. 7]. Інші адаптивні функції, які розглядаються – це функції Adaptive Piecewise Linear (APL), яка

фактично є кусково-лінійною функцією зі значеннями в діапазоні $[0, \infty]$, Swish, E-Swish [8, с. 7-8].

Інші АФ, які розглядаються в [8] включають Softplus, Softplus Linear Unit (SLU), Rectified Softplus (ReSP), Rand Softplus (RSP), Mish, стохастичні АФ, такі як Randomized ReLU (RReLU), Elastic ReLU (EReLU), Randomly Translational ReLU (RTReLU), Gaussian Error Linear Unit (GELU), а також поліноміальну АФ Smooth Adaptive AF (SAAF) [8, с. 9].

Щодо аналізу ефективності АФ у роботі [8] проводиться експериментальний аналіз за наступними параметрами:

− моделі, що розглядаються: MobileNet, VGG, GoogLeNet, ResNet, SENet, DenseNet, та ін.;

− активаційні функції: 18 активаційних функцій;

− набори даних: CIFAR-10, CIFAR-100, переклад з мови на мову, розпізнавання мовлення.

1.3.2 Деякі нові запропоновані АФ

В роботі [9] описує процес створення нової АФ Soft-ReLU, яка була створена завдяки аналізу поверхні помилок у моделях натренованих з використанням різних існуючих АФ [9, с. 5]. У роботі стверджується, що така поверхня помилок є характерною для кожної із активаційних функцій, та здається цікавим методом дослідження впливу вибору АФ на роботу натренованої моделі. Запропонована в роботі функція Soft-ReLU показала себе успішною порівняно із деякими іншими існуючими АФ на різних моделях та наборах даних [9, с. 7-9].

Також інша робота [10] розглядає створення двох нових активаційних функцій, та аналізує їх за результатами експериментів. Вибір функцій базується на загальних прагматичних міркуваннях щодо відомих причин виникнення проблем у відомих функціях активації, таких як запобігання необмеженому діапазону, використанню комбінування функцій, що добре

зарекомендувало себе в разі з існуючими АФ, та ін. [10, с. 6]. Запропоновані функції називаються IpLU та AbsLU [10, с. 11-12]. Функції експериментально випробовуються на різних моделях, таких як Alex, VGG-Network, Residual Network, Dense-Network, Inception-Network, а також на деякій згортковій мережі з різним рівнем глибини на наборах даних MNIST, Fashion MNIST, CIFAR-10, CIFAR-100, Covid-19, та інших [10, с. 12-13]. За результатами експериментів, функція AbsLU та IpLU показали кращі результати у порівнянні з набором існуючих АФ на деяких моделях з тренуванням на деяких наборах даних [10, с. 14-18].

В роботі [11] проводяться експерименти із модифікованими функціями логістичної сигмоїди та гіперболічного тангенсу ($tanh$). По-перше в роботі модифікується сигмоїдальна функція шляхом її масштабування та зсунення для отримання покращеник якостних характеристик із такою АФ, і, по-друге, в цій роботі розглядається асиметрична модифікація функції tanh, що є аналогічною до асиметричної модифікації в функції Leaky ReLU. Отримана модифікована АФ (яка в цій роботі називається leaky tanh, та penalized tanh) показує рівень ефективності приблизно на рівні АФ Leaky ReLU.

Робота [12] розглядає розташування регуляризації після та перед функцією активації та досліджує переваги асиметричної сатурації активацій АФ. Серед іншого в роботі випробовується зсунута версія функції гіперболічного тангенса, який при цьому співставлений таким чином, щоби перетинати точку початку координат під назвою Shifted Tanh [12, с. 7]. Робота демонструє, що така зсунута версія тангенсу показує кращі результати за незсунутий.

1.3.3 Адаптивні АФ

Було розглянуто декілька робіт, що пропонують адаптивні АФ, тобто функції, що містять параметри, що навчаються згідно з набором даних.

У роботі [13] пропонується низка адаптивних функцій, таких як Cosinu-Sigmoidal Linear Unit (CosLU), DELU, Linear Combination (LinComb), Normalized Linear Combination (NormLincomb), Rectified Linear Unit N (ReLUN), Scaled Soft Sign (ScaledSoftSign), Shifted Rectified Linear Unit (ShiLU). Автор визнає, що ці функції мають свої переваги та недоліки, та деяким з них притаманні проблеми ReLU у зв'язку з тим, що вони були розроблені саме як модифікації ReLU, та деяким може бути притаманна проблема зникаючого градієнта у разі ScaledSoftSign, а також LinComb та NormLinComb програють в швидкості [13, с. 3].

З цих АФ виділяються та є особливо цікавими дві функції LinComb та NormLinComb, які по суті є сумішшю (лінійною комбінацією, що є зваженою сумою) довільних інших активаційних функцій, та коефіцієнти, за якими кожна з функцій має вплив на сумарну функцію є такими, що навчаються. Таким чином ця активаційна функція фактично має змогу навчатися або вибирати якусь функцію у процесі навчання автоматично, або, скоріше, генерувати таку їх комбінацію, що призводить до кращих результатів. В цій роботі ці функції комбінували наступні АФ: ReLU, Sigmoid, Tanh, SoftSign [13, с. 3].

Експерименти проводилися з наборами даних MNIST та CIFAR-10 на моделі ResNet. За результатами [13, с. 11] ми можемо бачити, що на наборі даних CIFAR здебільшого функція CosLU конкурує з ReLU та на деяких мережах показує трохи кращі результати. Також функція NormLinComb показала трохи кращі результати, або результати на рівні з ReLU для деяких мереж, але тільки в тренувальному наборі і не в перевірочному.

Дещо схожий підхід на функції LinComb та NormLinComb ми можемо бачити в іншій роботі [14]. В цій роботі також використовується комбінована активаційна функція, але цього разу не для комбінування інших активаційних функцій, а для формування поліномів Ерміта, які сумуються для отримання комбінованої АФ [14, с. 3]. Див. схематичне зображення організації такої АФ на рисунку 1.3.

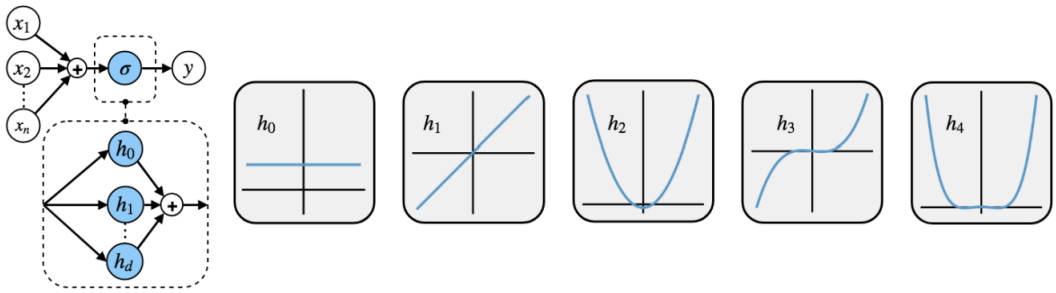


Рисунок 1.3 – Схематичне зображення роботи адаптивної АФ, що складається з поліномів Ерміта [14, с. 2]

За результатами експериментів автори показують, що така АФ потребує менше епох тренування ніж ReLU для досягнення певного рівня точності на різних наборах даних [14, с. 6].

2 ПОСТАНОВКА ЗАДАЧІ

2.1 Вибір напрямку дослідження

В процесі дослідження літератури за темою були розглянуті як активаційні функції, що можна вважати класичними та стандартними (такі як Logistic Sigmoid, Tanh, ReLU, та ін.), так і велика різноманітність їх модифікацій. Між усім, ми можемо бачити деякі закономірності.

Як вже було зазначено в попередньому розділі, такі функції, як Logistic Sigmoid та Tanh мають систематичну проблему із зникаючим градієнтом, та мають проблеми із тренуванням глибоких мереж, які пов’язані із зменшенням градієнту оновлення мережі під час її навчання, що особливо має вплив на навчання перших (найдальших від останнього) шарів мережі, якщо мережа містить багато шарів.

Також зазвичай зазначається, що ReLU та його модифікації вирішують цю проблему і фактично є поточним стандартом на те, які активаційні функції обираються для згорткових нейронних мереж. Такі мережі загалом навчаються помітно швидше, та досягають під час тренування більшої точності, та втім мають свої недоліки.

Окрім цього, згідно з оглядом літератури, дослідниками було запропоновано багато інших активаційних функцій та підходів до їх створення, включаючи використання елементів випадковості, додавання довільної кількості параметрів, що належать активаційній функції та навчаються за набором даних, а також використання комбінацій різних функцій як у просторовому вимірі (з’єднання окремих відрізків різних функцій), так і об’єднання декількох функцій як їх лінійні комбінації, та ін.

Щодо оцінки ефективності АФ, існують роботи, які дуже ретельно розглядають широкий набір стандартних активаційних функцій (наприклад у роботі [9]). Також, при розробці нових активаційних функцій загальною

практикою є порівняння ефективності створених АФ із існуючими стандартними АФ на різних наборах даних та різних моделях.

Враховуючи вищенаведене, можна зробити висновок, що потенційно найбільш доцільним напрямком подальших досліджень в цій області буде випробовування нових нетипових активаційних функцій, або модифікацій та/або комбінацій існуючих, включаючи порівняння ефективності використання нових функцій із вже існуючими.

Тому в цій роботі ставиться на меті випробування різних варіантів можливих нових активаційних функцій та оцінка їх придатності та ефективності у порівнянні із деякими популярними існуючими функціями при використанні їх для розпізнавання зображень за допомогою згорткових нейронних мереж.

Також слід зазначити, що згідно з загальними міркуваннями, а також із практикою, що в цілому використовується в багатьох інших дослідженнях, ефективність окремих активаційних функцій може бути різною в залежності від багатьох факторів.

Тому дослідження буде більш повним, надійним, а також потенційно більш корисним, якщо в ньому робота АФ буде досліджуватися з врахуванням впливу таких факторів. Наприклад, різні навчальні данні можуть значно впливати на можливість та ефективність тренування мережі, та використання різних АФ може мати різний вплив на процес тренування в залежності від вирішуваної задачі та відповідних даних.

Також відомо, що деякі активаційні функції більш чутливі до ступеня глибини мережі, що навчається [6], тож різні АФ можуть бути більш або менш ефективними з певними особливостями структури мережі, що використовується, та ін. Тому виявляється корисним також розглядання впливу таких факторів на ефективність АФ, що пропонуються зокрема.

2.2 Формулювання задачі

Беручи до уваги вищенаведені міркування, сформулюємо такі аспекти задачі:

– запропонувати потенційно можливі варіанти нових активаційних функцій;

– провести експерименти з такими функціями для оцінки їх придатності для тренування мереж, та оцінити параметри їх ефективності, а саме точність, якої досягає натренована модель: на тренувальному та перевірочному наборі даних;

– порівняти ефективність запропонованих активаційних функцій із популярними існуючими АФ та між собою;

– перевірити ці активаційні функції з різними гіперпараметрами моделі (наприклад, різні оптимізатори, темп навчання), та, якщо можливо, різними наборами даних.

3 МЕТОДИ ТА ТЕХНОЛОГІЇ РОЗВ'ЯЗАННЯ ЗАДАЧІ

3.1 Критерії вибору кандидатів для нових активаційних функцій

Враховуючи, що однією із головних цілей цієї роботи є створення та дослідження нових активаційних функцій, однією із задач, яку потрібно вирішити є методика або спосіб, згідно з яким ми можемо пропонувати нові активаційні функції, що можуть бути варті розглядання. Для цієї мети спробуємо сформулювати набір правил, які можна використовувати для емпіричного створення кандидатів нових АФ. Згідно з загальними міркуваннями, а також із інформацією наведеною в дослідженій літературі, при створенні активаційної функції виявляється важливим врахування таких аспектів:

– функція повинна бути безперервною та диференційованою у кожній її точці ([5]);

– з вимоги диференційованості в кожній точці можуть бути деякі виключення, які практично не заважають тренуванню моделі, та можуть вважатися дозволеними. Це випадки, коли безперервна функція має окрему точку (або окремі точки), у яких перша похідна не визначена, наприклад у разі шматково-лінійних функцій, таких, як ReLU, BreLU [15], SReLU [16], APL, MTLU [17], або подібних до них;

– наявність декількох локальних мінімумів або максимумів у АФ може викликати складнощі із знаходження глобального мінімуму функції похибки під час тренування;

– інтервали функції, на яких перша похідна дорівнює 0 не дозволяють алгоритму градієнтного спуску визначити напрямок зміни вагів та можуть перешкоджати навчанню [8];

– функції, які обмежені в своїх значеннях певним діапазоном значень (наприклад, Logistic Sigmoid або Tanh), у разі якщо вони не періодичні матимуть містити частини, у яких перша похідна дорівнює, або

тяжіє до нуля. Це здатно провокувати проблему зникаючого градієнта, уповільнює навчання, та здатно провокувати сатурацію вагів, що також може бути проблемою для навчання мережі [8];

– як зазначається в деяких роботах, функції, значення яких не дорівнює нулю у точці $x = 0$ можуть мати гірші здатності навчання [6];

– це здається парадоксальним протиріччям, але деякі роботи згадують, що функції з відкритим діапазоном значень також можуть викликати проблеми із навчанням [15].

Враховуючи ці особливості, в цій роботі, серед іншого, попередньо можна пропонувати експериментально дослідити різні варіанти АФ з використанням функції логарифма та функції експоненти, а також розглянути модифікації S-подібних функції, які попередньо можуть бути модифіковані таким чином, щоб задовільняти критеріям наведеним вище, а також, враховуючи успішність родини функцій ReLU із ЗНМ, ми можемо розглянути кандидати нових АФ, які можуть мати схожість з такими ReLU-подібними функціями при цьому враховуючи критерії наведені вище.

У разі логарифму, ми можемо бачити наступні особливості:

– він повільно зростає, але не обмежений зверху як Logistic Sigmoid та Tanh;

– він стрімко прагне до мінус нескінченності в негативній частині функції, проте це може бути вирішено шляхом заміни лівої частини на іншу функцію.

У разі експоненційної функції:

– Вона природнім чином плавно зростає переходячи із негативної в позитивну частину функції, що потенційно може бути корисною властивістю.

– Вона відносно швидко та необмежено зростає у додатній частині графіку, що напевно має бути проблемою, але це можна спробувати вирішити частково за допомогою горизонтального масштабування функції, та частково за допомогою фіксованого обмеження значення функції зверху.

У разі обох цих функцій, їх можна комбінувати з іншими функціями або шляхом стиковки у певній точці (наприклад, як у ReLU-подібних функціях), або шляхом деяких математичних операцій з іншими функціями.

3.2 Вибір наборів даних

Враховуючи, що мета цієї роботи – це дослідження різних АФ з задачами розпізнавання зображень згортковими нейронними мережами, а також з поглядом на необхідність проводити значну кількість експериментів та відносно обмежені обчислювальні можливості, прагматичним вибором щодо наборів даних можуть бути відносно невеликі набори даних, що містять невеликі зображення.

Для цілей зазначених вище добре підійдуть наступні відомі набори даних:

– MNIST [18];

– Fashion MNIST [19];

– CIFAR-10 [20].

Набір даних MNIST є, мабуть, найпопулярнішим простим набором даних для класифікації зображень. Він містить набор монохромних зображень рукописних цифр розміром 28x28 пікселів: 60000 тренувальних екземплярів та 10000 перевірочних екземплярів картинок. Кожен екземпляр картинки містить асоційоване з нею цифрове значення, що визначає яка цифра міститься у цьому екземплярі. Це дозволяє використовувати цей набір даних для навчання моделей здатності класифікації зображень.

Набір даних Fashion MNIST – це за структурою цілком ідентичний набір даних тієї ж розмірності та в такої самої кількості. Різниця тільки в тому, що на кожному екземплярі в цьому наборі даних, замість зображення цифри, міститься зображення елемент одягу, який належать одному із 10 класів. Попри ідентичну структуру набору даних, цей набір зазвичай показує дещо гірші результати на простіших мережах, які працюють добре

з оригінальним MNIST. Це зумовлено дещо більшою варіативністю даних, що містяться в цьому наборі (більша площа зображень, більше пікселів з відтінками сірого), що добре використовувати як додатковий компактний набір даних для швидких експериментів.

Набір даних CIFAR-10 також містить набір із 10 класів маленьких зображень, проте цей набір даних складніший для тренування ніж Fashion MNIST:

– він містить зображення трохи більшого розміру (розміром 32x32);

– замість монохромних зображень, він містить кольорові зображення;

– цей набір даних містить більш складні зображення. По-перше замість синтетичних зображень він містить зменшені фотографії, і по-друге, подібно звичайній фотографії, окремо від самого об’єкту відповідного класу він містить також і довільний фон на якому цей об’єкт зображений.

Цей набір складається загально з 50000 тренувальних зображень, та 10000 перевірочних зображень.

3.3 Моделі для експериментів

Залежно від навчального набору, було створено декілька моделей для проведення експериментів для визначення ефективності нових АФ.

3.3.1 Модель для класифікації з набором даних CIFAR-10

Для експериментів з класифікацією за набором даних CIFAR-10 використовувалась модель ЗНМ, яка містить 4 згорткових шари, 2 MaxPooling шари, а також два прихованих повнозв’язних шари, які під’єднані до останнього згорткового шару із використанням dropout під час навчання із коефіцієнтом 0.5. Схематичне зображення структури цієї моделі показане на рисунку 3.1.

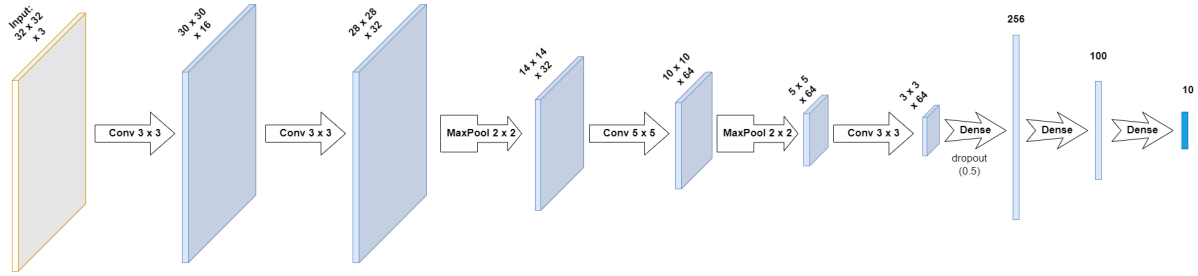


Рисунок 3.1 – Модель для класифікації за набором даних CIFAR-10

3.3.2 Модель для класифікації з наборами даних MNIST

Для експериментів з класифікацією за наборами даних MNIST та Fashion-MNIST використовувалась модель ЗНМ, яка містить 3 згорткових шари, 2 MaxPooling шари, а також два прихованих повнозв'язних шари (рисунок 3.2).

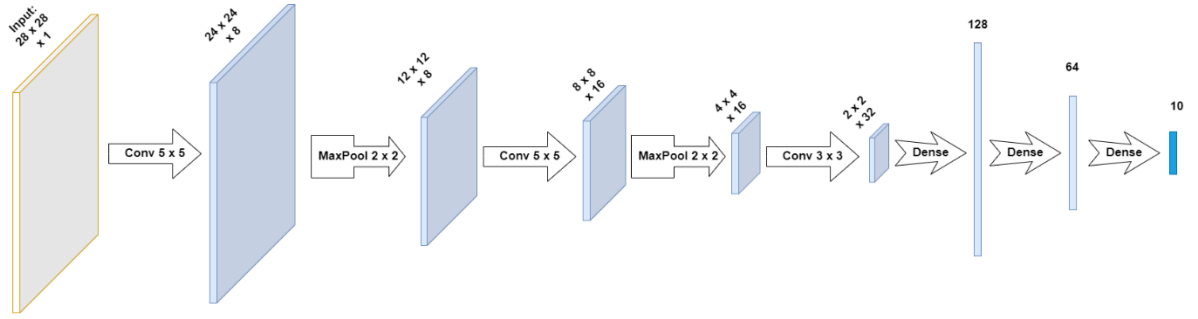


Рисунок 3.2 – Модель для класифікації за наборами даних MNIST

3.3.3 Модель для класифікації з набором даних Fashion-MNIST

Для експериментів з класифікацією за набором даних Fashion-MNIST використовувалась модель ЗНМ, яка містить 3 згорткових шари, 2 MaxPooling шари, а також два прихованих повнозв'язних шари, які під'єднані до останнього згорткового шару із використанням dropout під час навчання із коефіцієнтом 0.3 (рисунок 3.3).

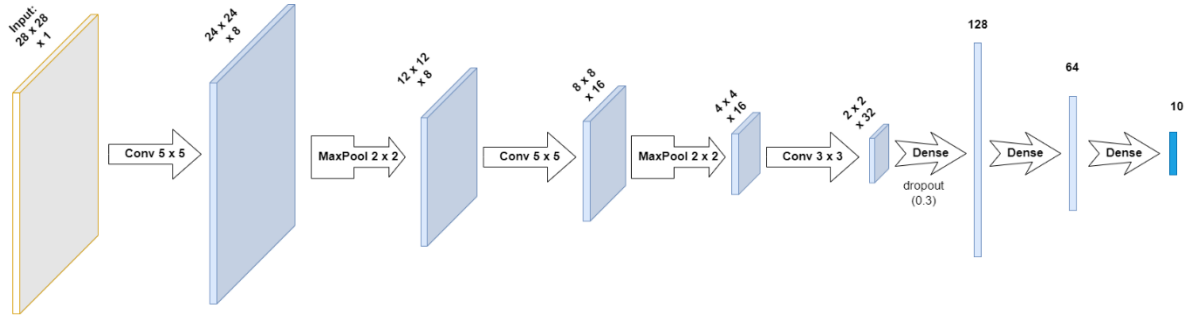


Рисунок 3.3 – Модель для класифікації за наборами даних Fashion-MNIST

3.4 Методика проведення експериментів

Розглянемо деякі особливості та деталі щодо проведення експериментів в цій роботі.

3.4.1 Процес вибору кандидатів нових АФ

Як було зазначено в попередніх розділах, ефективність АФ може залежати від багатьох факторів, включаючи вид задачі (класифікація, регресія, та ін.), набір даних, що використовується, оптимізатор, вид, та структура мережі, включно з модифікаціями відповідних гіперпараметрів, та ін. З поглядом на це, випробовування АФ може бути складною задачею, яка в ідеалі має враховувати таке різноманіття відповідних аспектів.

З погляду на обмеження часових та апаратних можливостей наявних для проведення в цій роботі, використовується компромісний підхід, у якому більшість експериментів з кандидатами на нові АФ робиться з однією конфігурацією моделі, яка попередньо розглядається як така, що даватиме найбільш показові результати, які мають бути найбільш наближеними до вірогідних сценаріїв реального використання. При цьому, після проведення основної серії експериментів з такою основною експериментальною моделлю, та визначення найбільш перспективних або показових кандидатів

АФ, проводяться додаткові експерименти з визначеними кандидатами на додаткових моделях та задачах.

Такий процес має потенційний недолік пов’язаний з тим, що в такий попередній відбір потенційно може не війти кандидат АФ, який не показує добрий результат на основній експериментальній моделі, але міг би показати добрі результати на більшості інших моделей. Аналогічно, добрий результат на основній моделі може виявитися гіршим на більшості інших моделей.

Таким чином такий процес певною мірою зменшує вірогідність знаходження кандидата, який був би добрим за середньою оцінкою по всім задачам, моделям, та параметрам, що розглядаються, але значно прискорює темп експериментів, тож розглядається як продуктивний для цілей цієї роботи.

У якості основної експериментальної конфігурації, з якою проводилось більшість експериментів для відбирання кандидатів АФ в цій роботі була обрана конфігурація наведена в таблиці 3.1.

Таблиця 3.1 – Основна експериментальна конфігурація

Модель	Набор даних	Оптимізатор	Темп навчання	Кількість епох	Кількість повторень
Див. розділ 3.3.1	CIFAR-10	Adam	0.03	30	10

У якості додаткових конфігурацій були обрані такі, що зазначені в таблиці 3.2.

Таблиця 3.2 – Додаткові експериментальні конфігурації

Модель	Набор даних	Оптимізатор	Темп навчання	Кількість епох	Кількість повторень
1	2	3	4	5	6
Див. розділ 3.3.1	CIFAR-10	SGD	0.03	30	10

Продовження таблиці 3.2

1	2	3	4	5	6
Див. розділ 3.3.2	MNIST	Adam	0.001	30	10
Див. розділ 3.3.3	Fashion-MNIST	Adam	0.001	30	10
Див. розділ 3.3.3	Fashion-MNIST	Adam	0.01	30	3
Див. розділ 3.3.3	Fashion-MNIST	SGD	0.3	30	3
Див. розділ 3.3.3	Fashion-MNIST	SGD	0.03	30	10
Див. розділ 3.3.4	CIFAR-10	Adam	0.002	30	3
Див. розділ 3.3.4	CIFAR-10	SGD	0.06	30	3

3.4.2 Набор існуючих АФ для порівняння з новими

З метою порівняння ефективності нових функцій з існуючими були обрані існуючі АФ, що можна розділити на декілька категорій. В таблицях нижче наведені назви АФ в тому виді, в якому вони використовуються в роботі при експериментах та порівняннях.

Перша категорія (таблицю 3.3) містить деякі прості математичні функції, для яких навіть не було знайдено згадок про їх успішність при використанні із ЗНМ, проте вони були включені в експерименти для кращого розуміння властивостей АФ такого виду, а також для того, щоб мати більшу базу незвичних функцій для порівняння із новими.

Таблиця 3.3 – Прості АФ для порівняння

Назва АФ	Формула
1	2
Linear	$f(x) = x$
Sin	$f(x) = \sin(x)$
Cos	$f(x) = \cos(x)$

Продовження таблиці 3.3

1	2
Gaussian	$f(x) = e^{-x^2}$
SGaussian	$f(x) = e^{-x^2} - 0.2$
Tanh3	$f(x) = tanh^3(x)$

На рисунку 3.4 показані відповідні графіки цих функцій.

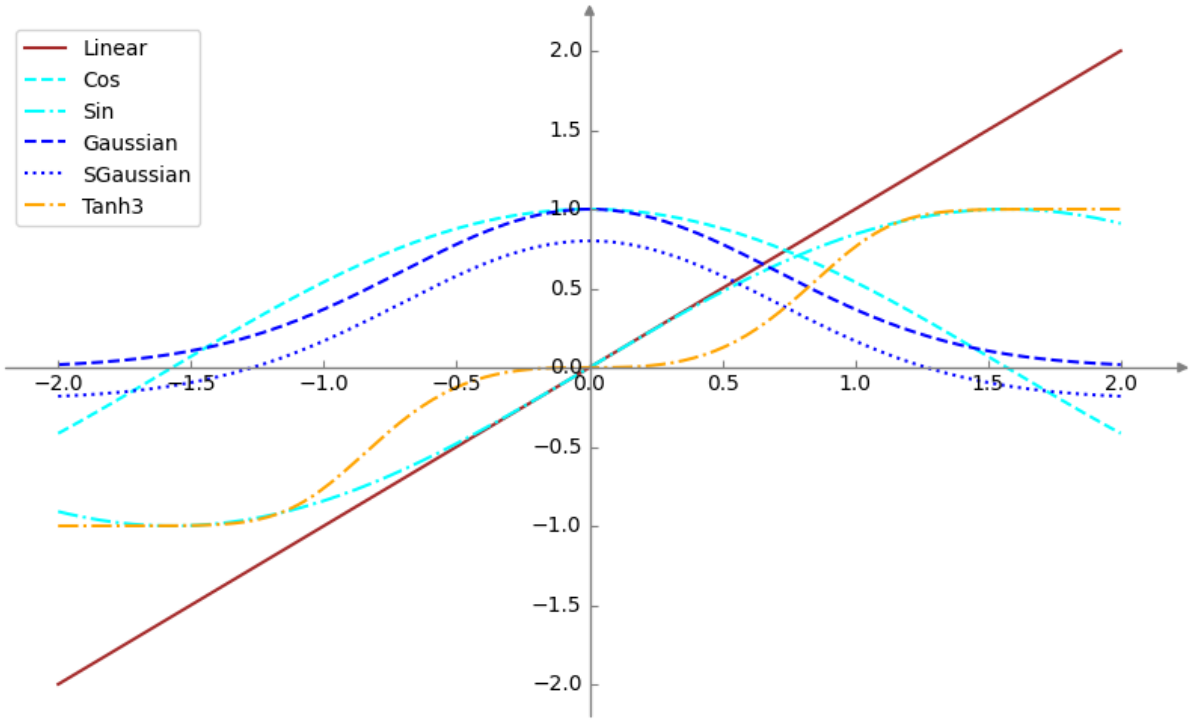


Рисунок 3.4 – Набір простих АФ для порівняння

Серед існуючих S-подібних функцій були обрані наступні функції, що наведені у таблиці 3.4. В цей перелік також було включено декілька суто «експериментальних» функцій, які не були в цьому виді взяти з інших робот, та не претендують на успіх, проте є простими та очевидними, та, подібно до попередньої категорії АФ, були включені в експерименти для кращого розуміння та різноманіття представлених функцій (див. OSSigmoid, Tanh3, SSTanh).

Таблиця 3.4 – Існуючі S-подібні АФ для порівняння

Назва АФ	Формула
Sigmoid	$f(x) = \sigma(x) = \frac{1}{1+e^{-x}}$
OSigmoid	$f(x) = Sigmoid(x) - 0.5$
OSSigmoid	$f(x) = 2 \cdot OSigmoid(x)$
Tanh	$f(x) = tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}}$
SSTanh	$f(x) = 0.5 \cdot Tanh(x) + 0.5$
Atan	$f(x) = arctan(x)$
SinAtan	$f(x) = \sin(\arctan(x))$
Erf	$f(x) = Erf(x)$
Asinh	$f(x) = \sin(\arctan(x))$
Softsign	$f(x) = \frac{x}{\lvert x \rvert + 1}$
Shifted Tanh [12], SoTanh	$f(x) = \tanh\alpha + \tanh(x - \tanh\alpha)$
Asinh	$f(x) = arcsinh(x)$

На рисунках 3.5 та 3.6 показані графіки цих функцій на двох різних масштабах. Як ми можемо бачити, ці функції мають різну динаміку зростання, та різний діапазон значень, який ці функції охоплюють.

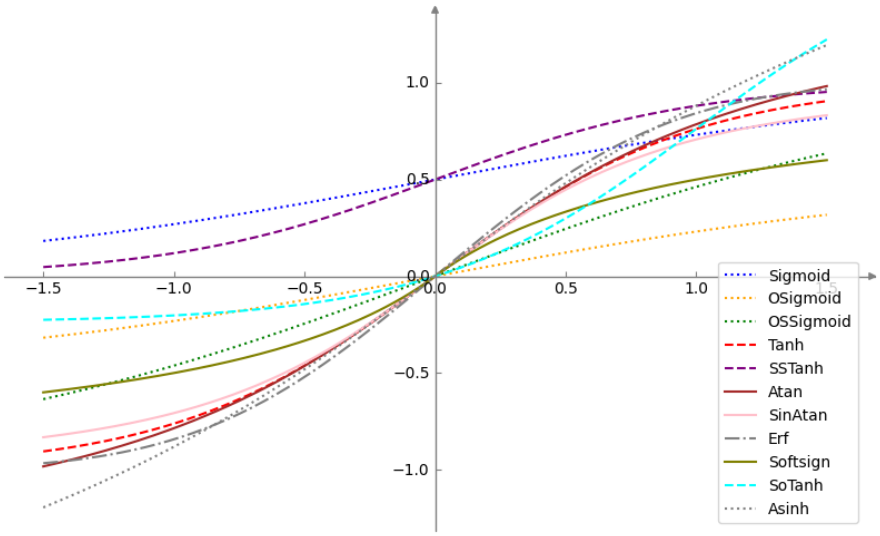


Рисунок 3.5 – S-подібні АФ для порівняння (у проміжку близькому до $x = 0$)

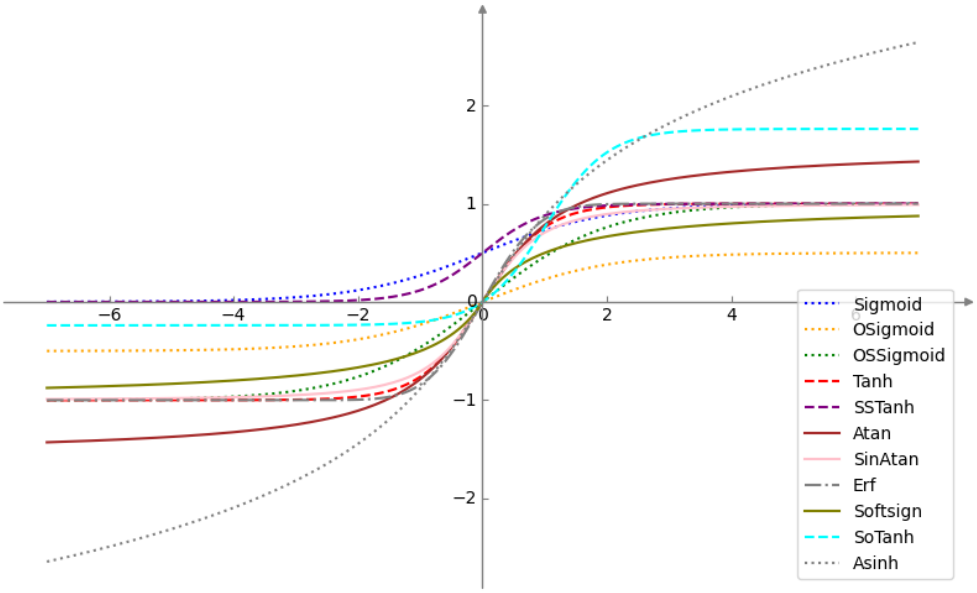


Рисунок 3.6 – S-подібні АФ для порівняння (у більшому масштабі)

Із переліка існуючих ReLU-подібних функцій були взяти функції, що наведені у таблиці 3.5.

Таблиця 3.5 – Існуючі ReLU-подібні АФ для порівняння

Назва АФ	Формула
1	2
ReLU	$f(x) = \max(0, x) = \begin{cases} x, & x \geq 0 \\ 0, & x < 0 \end{cases}$
ReLU-6 [21]	$f(x) = min(ReLU(x), 6)$
LeakyReLU(0.01)	$f(x) = \begin{cases} x, & x \geq 0 \\ \alpha x, & x < 0 \end{cases}\Big\vert_{\alpha=0.01}$
LeakyReLU(0.1)	$f(x) = \begin{cases} x, & x \geq 0 \\ \alpha x, & x < 0 \end{cases}\Big\vert_{\alpha=0.1}$
LeakyReLU(0.3)	$f(x) = \begin{cases} x, & x \geq 0 \\ \alpha x, & x < 0 \end{cases}\Big\vert_{\alpha=0.3}$
LeakyReLU(0.5)	$f(x) = \begin{cases} x, & x \geq 0 \\ \alpha x, & x < 0 \end{cases}\Big\vert_{\alpha=0.5}$

Продовження таблиці 3.5

1	2
PreLU	$f(x_i) = \begin{cases} x_i, & x_i \geq 0 \\ \alpha_i x_i, & x_i < 0 \end{cases}$
ELU	$f(x) = \begin{cases} x, & x \geq 0 \\ \alpha(e^x - 1), & x < 0 \end{cases}$
SiLU	$f(x) = x\sigma(x) = x\frac{1}{1+e^{-x}}$
Swish	$f(x) = x\sigma(\beta x) = x\frac{1}{1+e^{-\beta x}}$
GELU	$f(x) = x\Phi(x) = x\frac{1}{2}[1 + \mathrm{erf}(x/\sqrt{2})]$

На рисунках 3.7 та 3.8 показані графіки цих функцій на двох різних масштабах.

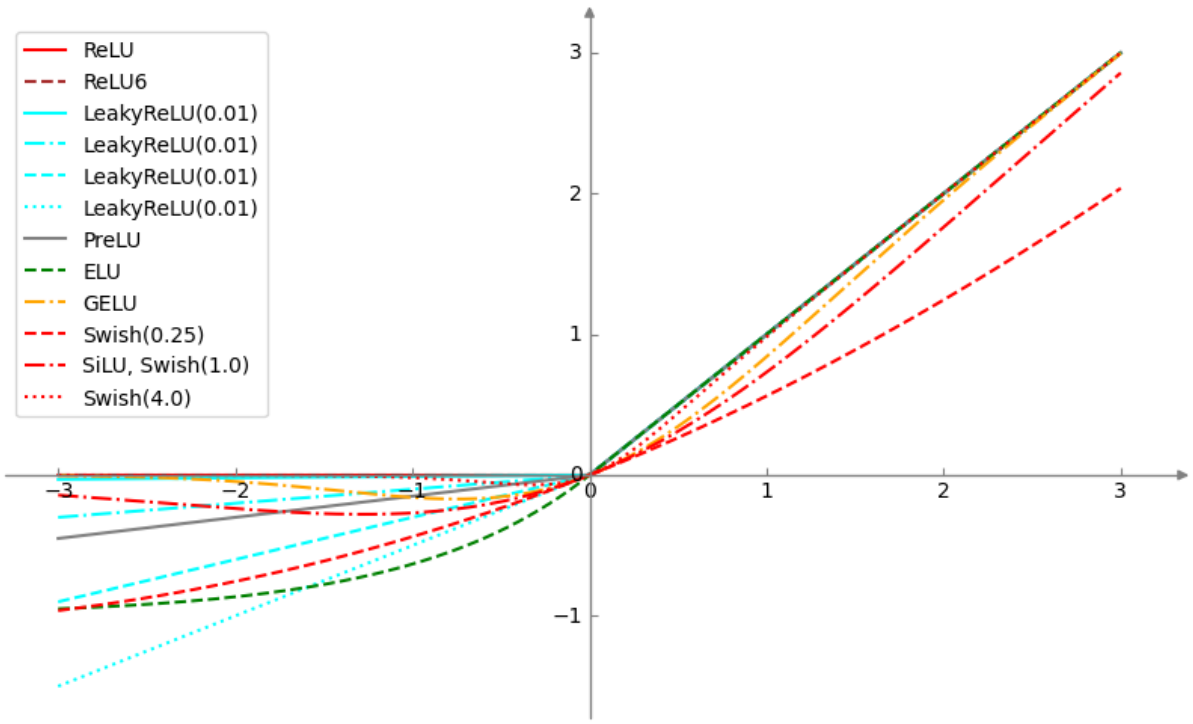


Рисунок 3.7 – ReLU-подібні АФ для порівняння (у проміжку близькому до $x = 0$)

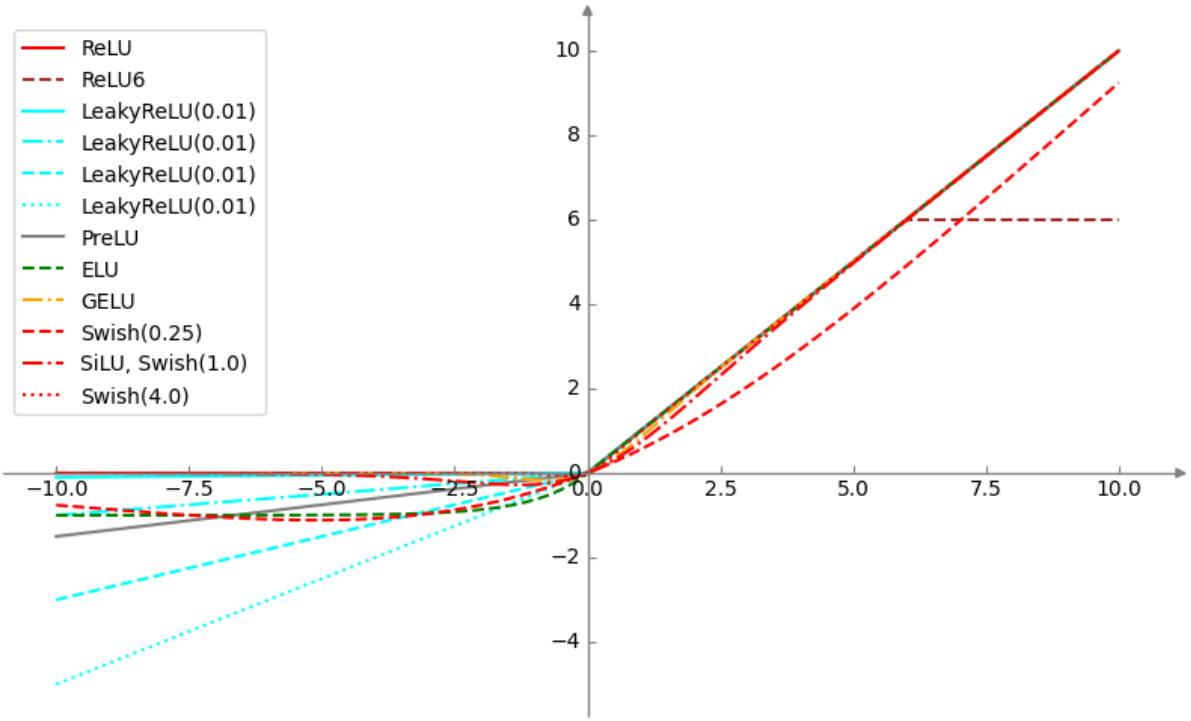


Рисунок 3.8 – ReLU-подібні АФ для порівняння (у більшому масштабі)

3.5 Методика вимірювання ефективності АФ

В цій роботі використовуються наступні міркування коли йдеться про оцінку ефективності ЗНМ в залежності від вибору АФ:

а) для оцінки ефективності кожної обраної нової АФ необхідно провести порівняння її ефективності із ідентичним набором вже існуючих функцій;

б) параметри, що мають бути порівняні – це:

– точність класифікації у разі задач класифікації;

– значення функції втрат у разі задач регресії та автокодувальника;

– динаміка зміни цих параметрів у часі задля можливості порівняння швидкості навчання з використанням різних АФ шляхом співставлення різних графіків;

в) задля мінімізації випадкових відхилень під час експериментальних замірів слід робити заміри декілька раз із створенням нового екземпляру моделі під час кожного заміру. За результатами низки замірів потрібно розрахувати середні значення параметрів, що оцінюються та стандартне відхилення від середніх значень. Окрім того, усі значення гіперпараметрів (кількість епох, темп навчання, та ін.) мають бути однаковими у всіх запусках та між запусками для рівноцінного співставлення результатів;

г) у попередніх експериментах попередній відбір потенційно придатних АФ та відкидання таких, що виглядають безперспективними може не містити стандартну кількість повторів у зв’язку з тим, що значення ефективності з високою мірою точності у таких випадках не важливо, та може бути визначено вже у відібраних АФ.

Для оцінки ефективності кожної АФ та її порівняння з іншими було вирішено використовувати наступні метрики:

а) максимальна ефективність (незалежно від епохи, на якої вона досягається). Для цього відповідні експерименти мають містити достатньою кількістю епох, що дозволить всім АФ досягнути у процесі тренування стану найбільшої можливої для цієї АФ ефективності;

б) оцінка узагальнених рангів ефективності кожної АФ (див. розділ 3.5.2), що може бути доцільним робити для різних контекстів застосування кожної АФ, наприклад:

– ранг ефективності кожної АФ при застосуванні з окремими оптимізаторами;

– узагальнений ранг ефективності кожної АФ по всім оптимізаторам, та ін. гіперпараметрам, що використовувались під час експериментів.

Визначення рангів ефективності описано в наступному розділі 3.5.2.

3.5.2 Ранги ефективності АФ

Як було зазначено в розділі 3.5.1, поняття рангу ефективності (РЕ) було введено як одна із метрик для оцінки ефективності АФ. РЕ – це фактично спосіб визначити відносну ефективність АФ порівняно із низкою інших конкретних АФ в здатності вирішення конкретної задачі або набора конкретних задач, і фактично визначається як місце, яке займає якась АФ за якимось обраним критерієм серед певного переліка інших АФ у здатності вирішення відповідної задачі.

З погляду на потреби цієї роботи визначення РЕ для кожної АФ із певного набору АФ полягатиме в сортуванні АФ за їх середньою точністю класифікації в певній задачі та конфігурації та фактично визначенні порядкового номеру АФ в такому відсортованому переліку. АФ із найвищою точністю отримує РЕ, що дорівнює 1, наступна за ефективністю АФ матиме РЕ , що дорівнює 2, і так далі.

Корисність такого способу оцінки ефективні полягає в здатності визначати узагальнені значення РЕ, шляхом усереднення інших РЕ, отриманих для того самого набору АФ при вирішенні інших задач, або в інших конфігураціях мережі/моделі та ін. РЕ не залежить від окремих значень ефективності, підходить як для задач у яких доречно оцінювати точність, так і задач, у яких потрібно оцінювати значення функції похибки, зберігаючи при цьому можливість комбінування різних РЕ для формування узагальнених варіантів РЕ.

Недоліком рангу ефективності може бути те, що він не враховує саме різницю в точності або значенні функції похибки між кращою та гіршою АФ, а натомість враховує тільки сам факт переваги однієї АФ, відносно відповідного параметру, над іншою. Тому РЕ може використовуватись як доповнення до аналізу абсолютних метрик ефективності АФ, а також як один із емпіричних критеріїв для обирання потенційно найбільш

ефективних АФ для подальшої більш конкретної оцінки із відповідною конкретною моделлю та задачею.

Слід зауважити, що коли йдеться про співставлення РЕ (порівняння, або усереднення рангів за декількома серіями експериментів), слід співставляти тільки РЕ, які були отримані із однаковими наборами АФ. Якщо кількість або набор АФ в різних експериментах розрізняється хоча б на одну АФ, відповідні ранги вже не можуть бути комбіновані для отримання узагальнених РЕ.

3.6 Іменування нових АФ

Використання традиційних існуючих типових підходів, а також абревіатури, що виходять із іменування функцій англійською мовою для будь-яких інших аспектів взагалі.

Серед іншого використовуються такі існуючі абревіатури:

– Re – Rectified (випрямлений), як в ReLU;

– Ge – Gaussian error weighted (зважена функцією Гаусової похибки), як в GELU. Зазначимо, що на відміну від оригінального написання «GE», у цій роботі було обране спрощене написання «Ge» у зв’язку з тим, що цей префікс як правило суміщається із іншими префіксами, і таким чином ми підкреслюємо його нероздільність та спрощуємо відокремлення від інших префіксів;

– Si – Sigmoid-weighted (зважена сигмоїдою), як в SiLU;

– LU – Linear Unit (випрямлена одиниця), як в ReLU, SiLU, та ін.

Арктагненс визначається як Atan (замість Arctan), та гіперболічний арксінус визначається як Asinh (замість Arcsinh) з поглядом на те, що вони часто присутні в назвах нових АФ, які нерідко вже мають чималий префікс окрім цього (наприклад, AGeSoAtan, ASiSoAsinh).

Також використовуються наступні префікси-скорочення для типових модифікацій нових АФ в цій роботі:

– S – Scaled (масштабований);

– O – Origin-aligned (підогнаний під начало координат);

– So – Shifted, origin-aligned (зсунутий, та підігнаний під начало координат);

– A – Adaptive (адаптивний).

3.7 Інші особливості

Слід зазначити наступні обмеження роботи та інші особливості:

– в межах цієї роботи не ставиться на меті порівняння роботи АФ в зв’язку з використанням чи невикористанням таких технік як dropout та регуляризація. Ці техніки можуть бути застосовані, якщо це доцільно для певної моделі, але експерименти не включать перевірку з різними конфігураціями в цих аспектах;

– для оцінки ефективності ЗНМ в залежності від обраної АФ, активаційна функція застосовується для всіх шарів окрім останнього. Останній шар залишається фіксованим в обраної моделі і фактично залежить від задачі, що вирішується моделлю.

3.8 Вибір програмної платформи

3.8.1 Вибір мови програмування

Мова програмування Python мабуть фактично є найбільш популярним вибором, що стосується наукових досліджень, а також штучного інтелекту та нейронних мереж зокрема. Завдяки такому статусу впродовж значного періоду часу, ця мова акумулювала великий багаж різноманітних бібліотек, що тим чи іншим чином є корисними у цих галузях. До того ж ця мова є доволі простою, та достатньо гнучкою, а також зручною до швидкого прототипування та експериментування.

З урахуванням таких міркувань для мети цієї роботи була обрана саме мова Python, а саме використовувалась одна з її актуальних на момент часу виконання роботи версій 3.11.8.

3.8.2 Вибір бібліотеки машинного навчання

Враховуючи простоту програмування та експериментування, а також широкий спектр існуючих інструментів для створення різноманітних моделей нейронних мереж була обрана бібліотека машинного навчання Keras. Між іншим ця бібліотека дозволяє дуже легко створювати нестандартні користувацькі активаційні функції завдяки механізму автоматичного диференціювання, що дуже доречно для цієї роботи. Що також є значним фактором, ця бібліотека підтримує акселерацію процесів тренування та виводу нейронних мереж за допомогою апаратних прискорювачів, таких як графічні процесори (GPU – Graphics Processing Unit), що є дуже корисним для прискорення експериментів.

В цій роботі використовується найсвіжіша на час початка роботи версія Keras 3.1.1.

4 РОЗГЛЯДАННЯ МОДИФІКОВАНИХ ТА НОВИХ S-ПОДІБНИХ АФ

Розглядаючи властивості S-подібних АФ, можна зазначити, що вони в цілому задовольняють критеріям для створення можливих нових АФ, що викладені у розділі 3.1.

Для створення нових функцій ми можемо розглядати або модифікації існуючих S-подібних АФ, а також розглядати нові. Щодо модифікацій, очевидними модифікаціями можуть бути наступні:

– горизонтальне та/або вертикальне масштабування АФ;

– горизонтальний та/або вертикальний зсув.

Особливо цікавою, та потенційно корисною модифікацією можуть бути варіанти зсуву S-подібних функцій, що зсунути таким чином вправо та догори, щоб при цьому функція зберігала перетин з точкою початка координат, що вводить потенційно корисну асиметрію (рисунок 4.1). Як зазначалось у розділі 3.1, перетин функцією точки початку координат може бути корисною властивістю, що може призводити до підвищення ефективності порівняно з варіантами АФ без такого перетину, у зв’язку з тим, що розподіл активацій буде залишатися навколо нульової точки, що має спростити та прискорити тренування.

До того ж асиметрія, що виникає при перенесенні S-подібної функції таким чином робить таку функцію дещо схожою за формою до ReLU-подібних функцій (малі значення у від’ємній частині, та значення, що близькі до функції $f(x) = x$ у додатній частині), принаймні у певному проміжку близькому до $x = 0$. До того ж, той факт, що функція припиняє свій зріст та має обмеження у додатній частині, згідно з критеріями викладеними в розділі 3.1 також може бути потенційно корисною властивістю АФ. Насправді, така форма АФ є дещо схожою на деякі інші успішні ReLU-подібної функції, такі як BreLU [15]та ReLU-6 [21], але при

цьому мають безперервну першу похідну, що може бути корисною властивістю порівняно з цими АФ.

Робота [13] описує одну з таких АФ (Shifted Tanh – зсунутий гіперболічний тангенс), та демонструє її більшу ефективність відносно використання звичайного незсунутого гіперболічного тангенсу. Спробуємо між іншим модифікувати таку функцію таким чином, щоб більше наблизити її за зовнішнім виглядом до ReLU-подібної на більшому інтервалі значень шляхом її масштабування, а також розглянемо інші S-подібні функції зсунуті і масштабовані аналогічним чином.

4.1 Модифікації існуючих S-подібних АФ

Як основу для модифікованих функцій використовуватемо такі S-подібні функції:

- $Tanh$ (гіперболічний тангенс);
- $Atan$ (арктагненс);
- $Asinh$ (гіперболічний арксінус).

Кожна з цих функцій має різну динаміку зростання, та деякі функції також мають різні діапазони значень. Між іншим серед цих функцій можна виділити $Arcsinh$ у тій її особливості, що вона на відміну від інших має необмежений діапазон значень, що потенційно може мати позитивний ефект у боротьбі з проблемою зникаючого градієнта. Інші функції також мають свої унікальні властивості, які також теоретично можуть виявитись корисними у певних випадках.

Також слід зазначити, що функція гіперболічного тангенсу є ідентичною до зсунутої та масштабованої функції логістичної сигмоїди [22]:

$$Tanh(x) = 2 \cdot Sigmoid(2x) - 1. \tag{4.1}$$

Тому у разі, якщо ми розглядаємо версії таких функцій, що включають довільний зсув та масштабування, ми технічно можемо обмежитись випробовуванням тільки однієї із функцій $Tanh$ чи $Sigmoid$. Тому для більшості відповідних експериментів з цих двох функцій була обрана функція $Tanh$, з поглядом на те, що її форма «за замовчанням» (без зсуву та масштабування) краще задовольняє умовам наведеним в розділі 3.1, та, відповідно, може бути більш ефективною у зв'язку з тим, що вона проходить через точку початка координат, та має додатні та від'ємні значення. Втім, функція Sigmoid ччастково також представлена у експериментальних замірах в деяких її формах з метою порівняння.

4.1.1 Зсунуті співставлені S-подібні АФ

Категорія функцій Fso (shifted, origin-aligned – зсунуті, співставлені із початком координат), яка розглядається в цьому підрозділі містить модифікації S-подібних функцій, які загально можуть бути описані за допомогою формули 4.2.

$$Fso(x) = S(\alpha) + S(x - \alpha), \tag{4.2}$$

де S – будь-яка S-подібна функція;

α – параметр, який визначає величину горизонтального зсуву цієї функції.

Таким чином це фактично є незмінна за формою функція S, яка при цьому зсунута горизонтально та вертикально таким чином, щоб зберігати перетин із початком координат незалежно від величини горизонтального зсуву.

Як також було наведено раніше, один із різновидів такої функції, що базується на функції $tanh$ ретельно досліджений та описаний у роботі [13], як функція із назвою Shifted Tanh. В поточній роботі також включений

ідентичний варіант АФ, який в цій роботі має назву $SoTanh$, і, окрім цього, також розглядаються інші S-подібні функції, як основа для аналогічних модифікованих АФ. Загалом, враховуючи, що у якості функції S використовувались функцій $Tanh$, $Atan$, $Asinh$ відповідні конкретні АФ можуть бути описані формулами (4.3), (4.4), (4.5).

$$SoTanh(x) = Tanh(\alpha) + Tanh(x - \alpha), \tag{4.3}$$

$$SoAtan(x) = Atan(\alpha) + Atan(x - \alpha), \tag{4.4}$$

$$SoAsinh(x) = Asinh(\alpha) + Asinh(x - \alpha). \tag{4.5}$$

Подивимось на ці функції із значенням $\alpha = 1.0$, та оцінимо їх ефективність. Графіки цих АФ можна побачити на рисунку 4.1.

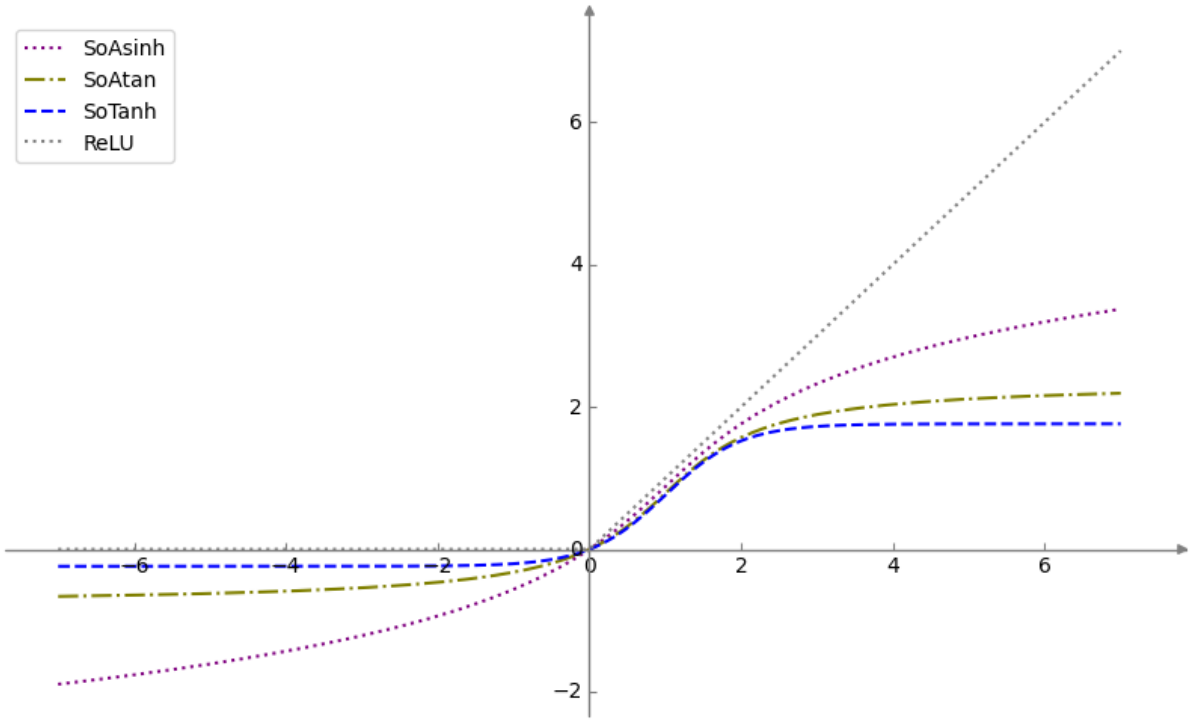


Рисунок 4.1 – Графіки АФ $SoTanh$, $SoAtan$, $SoAsinh$ співставлені із графіком $ReLU$ для порівняння

Ефективність моделі (див. розділ 3.3.1) виконувати задачу класифікації зображень з набору CIFAR-10 в залежності від АФ з якою вона була натренована можна побачити на рисунку 4.2. З кожною АФ модель була натренована впродовж 30 епох 10 разів, та дані приведені на рисунку відображають усереднену точність роботи натренованої моделі згідно цим 10 тренуванням. Як ми можемо бачити, модифіковані АФ $SoTanh$, $SoAtan$, $SoAsinh$ мають суттєво більшу точність класифікації, яка приблизно дорівнює точності аналогічної моделі, яка натренована з використанням АФ $ReLU$, $ReLU6$, та $PreLU$.

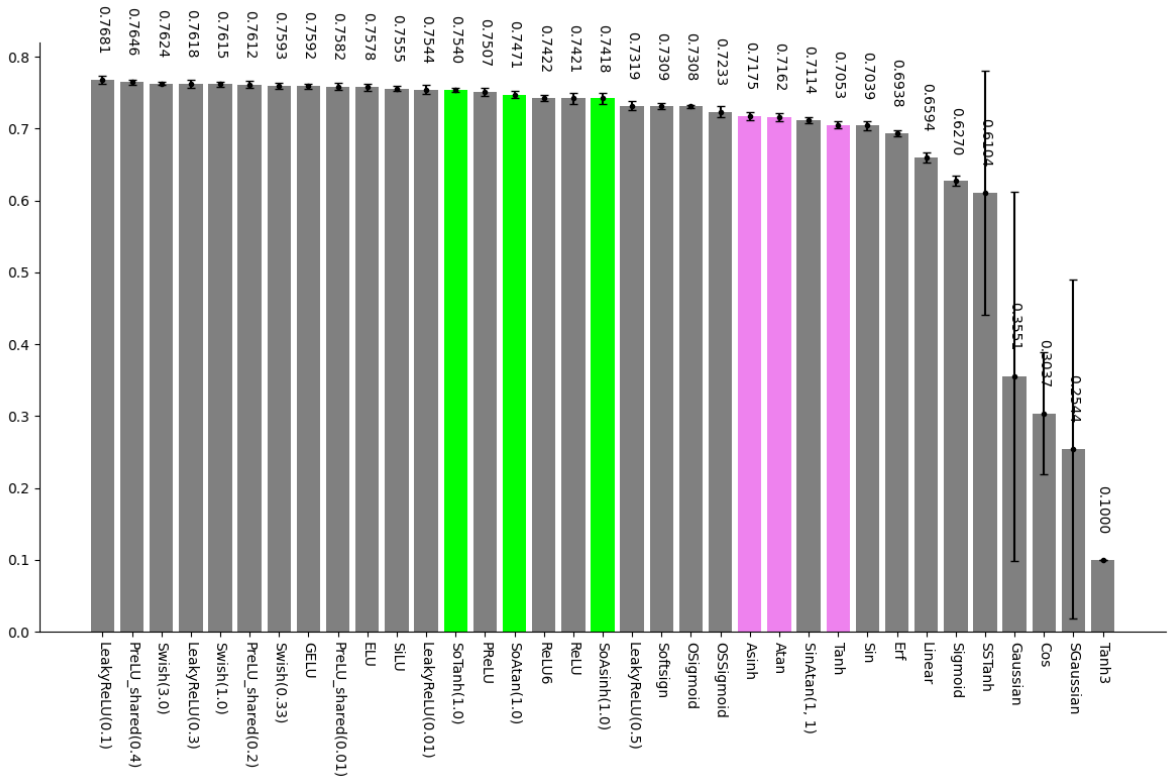


Рисунок 4.2 – Порівняння ефективності модифікованих S-подібних функцій із немодифікованими аналогами та іншими існуючими АФ

Загалом це узгоджується із виводами приведеними з приводу АФ $Shifted\ Tanh$ (яка фактично є ідентичною до $SoTanh$) в роботі [13]. Тож ми можемо бачити, що значне підвищення точності класифікації також

притаманне і принаймні двом іншим S-подібним функціям ($Arctan$ та $Asinh$), коли вони змінюються згідно з формулою (3.3). Втім, за цими результатами покращення для цих двох функцій було дещо меншим ніж для функції $Tanh$.

4.2 Активаційна функція $SymLog$

Як зазначалось в розділі 3.1, при створенні нових АФ між іншим може бути потенційно корисним враховувати такі два аспекта:

– S-образні функції, такі як Sigmoid, Tanh, та ін. мають обмежений діапазон значень та першу похідну, що прагне до нуля із збільшенням значення параметру АФ, що може бути причиною проблеми зникаючого градієнта;

– втім, певні обмеження на значення необмежених функцій також можуть бути корисними [19].

Ці два критерії вступають у певне протиріччя одне з одним, тому спробуємо створити функцію, яка буде певним компромісом між цими двома вимогами.

В цьому розділі для цієї мети оберемо натуральний логарифм як основу функцію. Його специфіка, яка може бути корисною в цьому аспекті – це те, що ця функція фактично не є обмеженою, та при цьому із збільшенням аргументу вона зростає повільніше, тому її неформально можна назвати «м’яко обмеженою».

Загалом це забезпечує більші значення першої похідної порівняно із функціями $Tanh$ та $Atan$, що має призводити до більшого градієнту змін ваг на великих значення аргументу, що може сприяти зменшенню проблеми зникаючого градієнта.

На рисунку 4.3 наведено графіки перших похідних функцій $ln(x+1)$, $tanh(x)$, та $arctan(x)$.

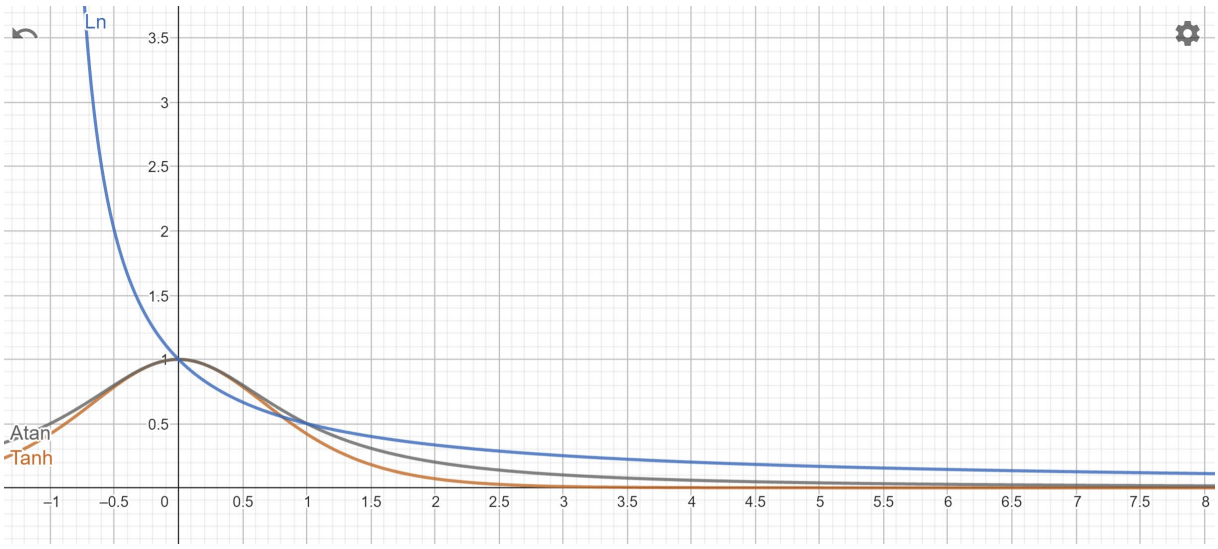


Рисунок 4.3 – Графік перших похідних функцій $ln(x+1)$, $tanh(x)$, та $arctan(x)$

Втім, як ми можемо бачити, функція логарифму дуже стрімко зменшується у від'ємній частині графіка (рисунок 4.4).

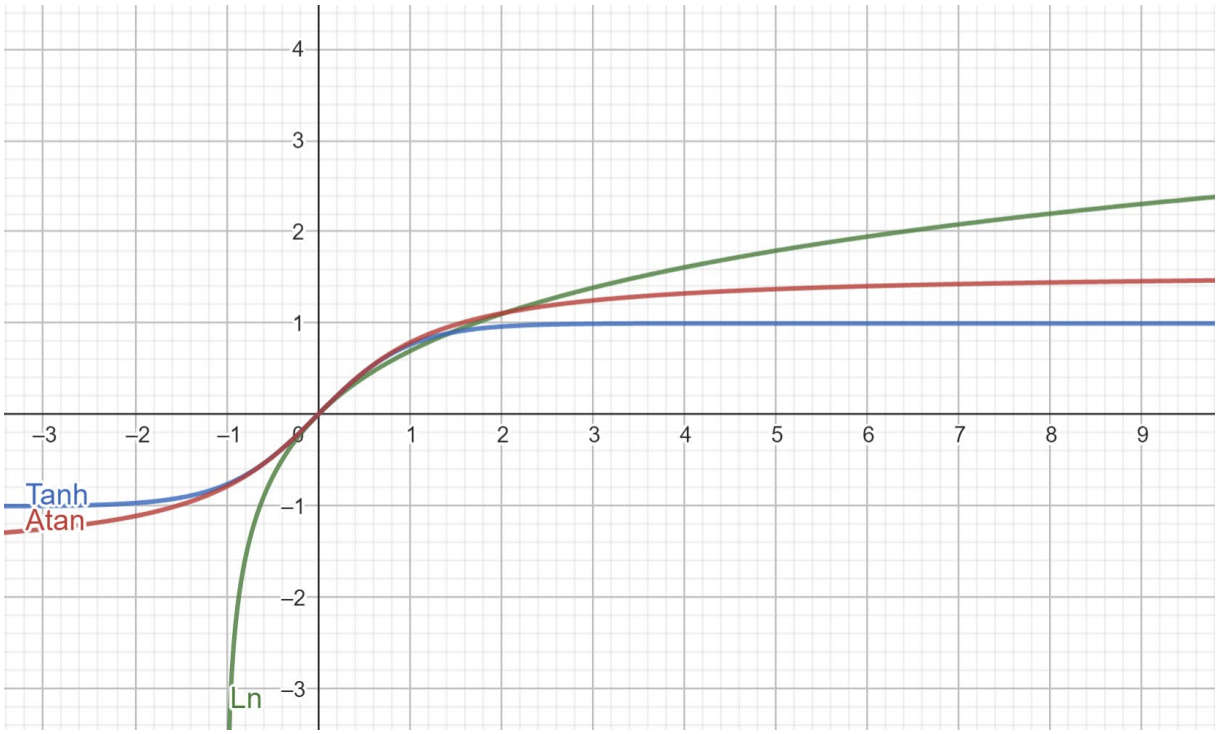


Рисунок 4.4 – Графіки функцій $ln(x+1)$, $tanh(x)$, та $arctan(x)$

В новій експериментальній АФ, яка розглядається в цьому розділі ми вирішимо цю проблему фактично відобразивши додатну частину графіку відносно початку координат, та отримаємо функцію відповідної АФ, яку ми назвемо $SymLog$ (symmetric logarithm – симетричний логарифм), як показано у формулі (4.6).

$$SymLog(x) = sign(x) \cdot \alpha \cdot \ln(|x| + 1) \tag{4.6}$$

Графік цієї функції можна побачити на рисунку 4.5.

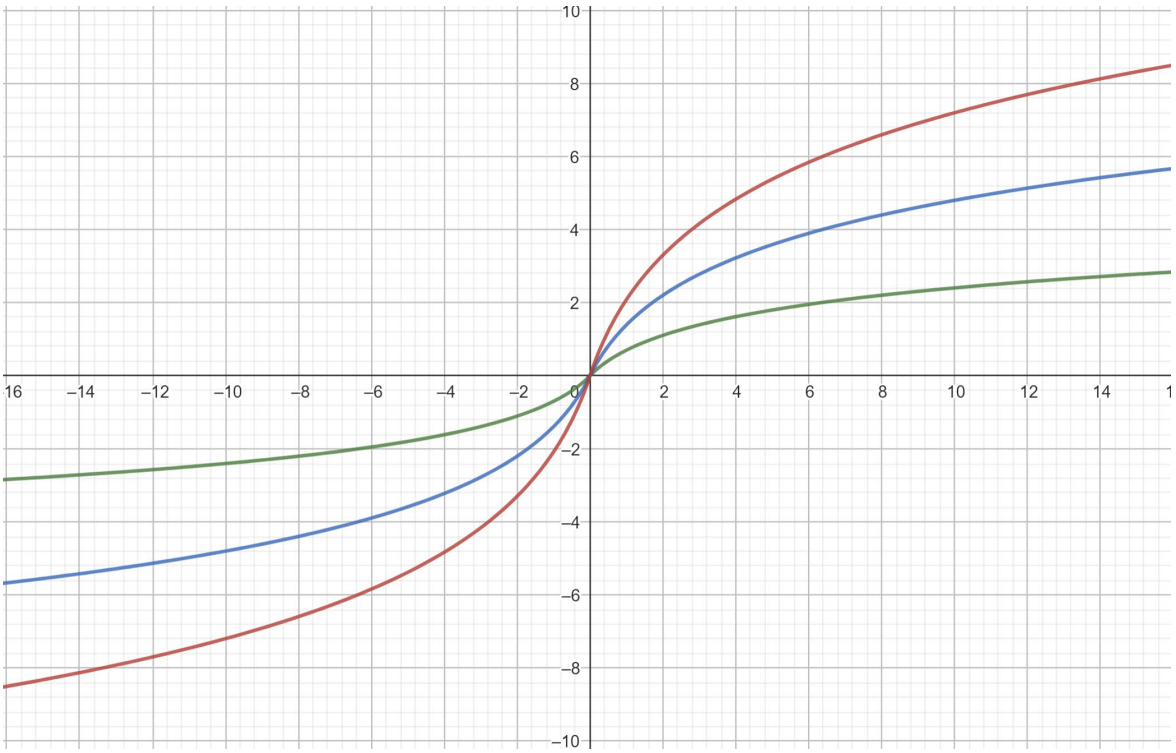


Рисунок 4.5 – Графік функції $SymLog$ із значеннями $\alpha = 1, 2, 3$

Поглянемо на точність класифікації зображень CIFAR-10 моделі з розділу 3.3.1 з цією АФ, та порівняємо її з моделями, які натренованими з іншими існуючими S-подібними, та іншими існуючими функціями взагалі. Усі моделі тренувались впродовж 30 епох з оптимізатором Adam, темпом навчання 0.001, та розміром пакета 32.

Моделі з кожною АФ тренувались 10 раз для отримання усереднених результатів точності для кожної АФ. Відповідні порівняльні результати точності можна бачити на рисунку 4.6.

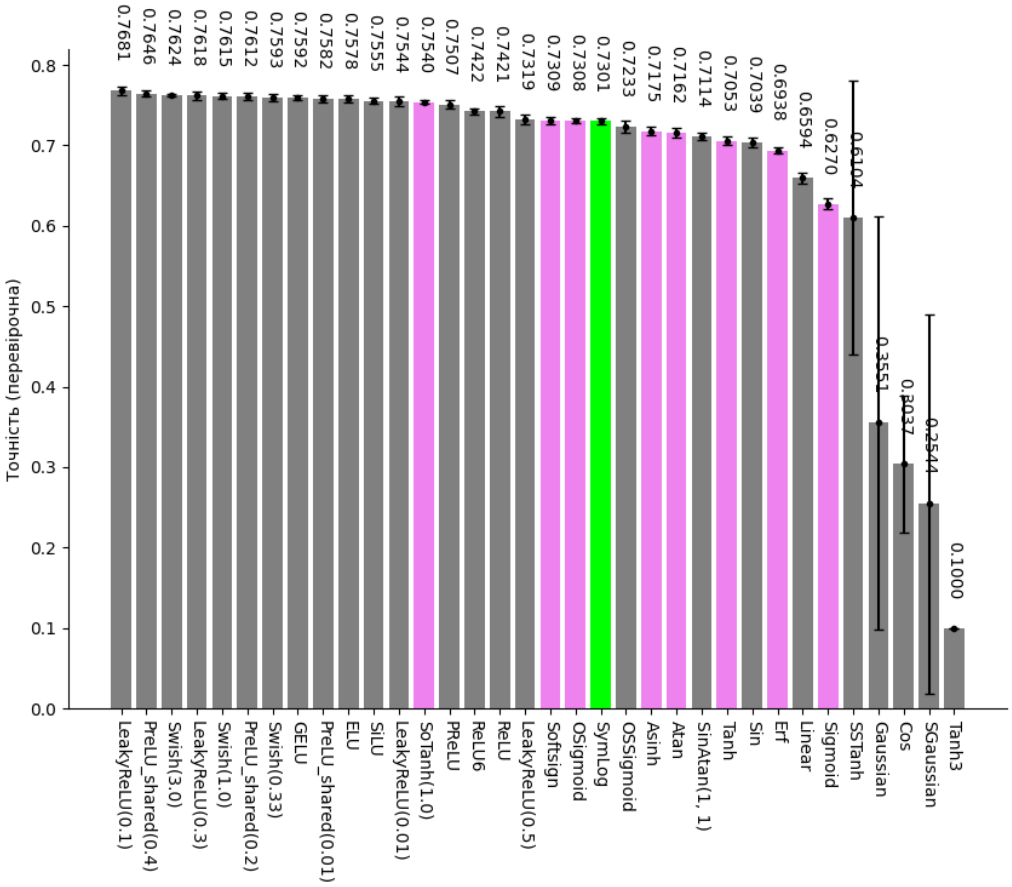


Рисунок 4.6 – Порівняння точності класифікації з використанням $SymLog$ порівняно з іншими АФ (фіолетовим відмічені існуючі S-подібні АФ)

Як ми можемо бачити, точність за цим варіантом функції $SymLog$ на цій моделі виявилась майже ідентичною до функцій $OSigmoid$ та $Softsign$. Вона краще інших популярних S-подібних АФ, втім помітно гірше ніж функція $SoTanh$.

5 НОВІ АФ З ЕКСПОНЕНЦІЙНОЮ ФУНКЦІЄЮ

Як зазначалось в розділі 3.1, експоненційна функція потенційно може бути гідним варіантом АФ, якщо вирішити проблему швидкого зростання цієї функції після певних значень її аргументу. Також, згідно з критеріями в розділі 3.1 також потенційно корисним буде використовувати саме зсунуту версію експоненційної функції, таким чином, щоби вона перетинала точку початка координат. В цьому розділі розглядається три варіанта обмеження такої функції зверху та оцінюється відповідна ефективність таких варіантів функцій порівняно із стандартними АФ.

5.1 Активаційна функція $BOExp$

Цей варіант експоненційної функції, окрім зсунення її таким чином, щоб вона перетинала початок координат, вона модифікована таким чином, щоб мати обмежену верхню частину за аналогією з $Bounded\ ReLU$ [9, с. 6].

Така функція $BOExp$ (Bounded, Origin-aligned Exponent – обмежена, співставлена з початком координат експонента) може бути виражена формулою (5.1).

$$BOExp(x) = \min(U, e^{\alpha x} - 1), \tag{5.1}$$

де U – це значення, яким функція обмежується зверху;

α –масштабний коефіцієнт функції.

На рисунку 5.1 наведено графік функції $BOExp$ із значеннями $\alpha = 0.2, 0.3, 0.4$ та $U = 7.0$, співставлений із графіками АФ $ReLU$, $ReLU6$.

Подивимось на точність класифікації зображень з набора даних CIFAR-10 з використанням функції $BOExp$ із параметрами $\alpha = 0.3$, $U = 7.0$. Відповідне порівняння точності використанням такої функції, порівняно з використанням інших стандартних функцій приведене на рисунку 5.2.

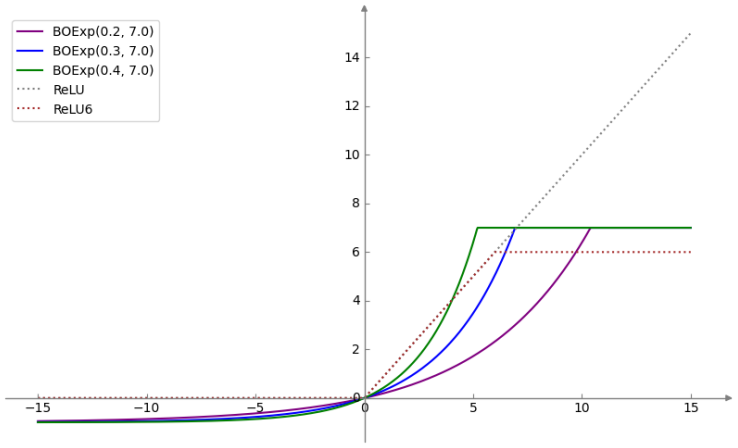


Рисунок 5.1 – Графік функції $BOExp$ із значеннями $\alpha = 0.2, 0.3, 0.4$ та $U = 7.0$, співставлений із графіками АФ $ReLU$, $ReLU6$

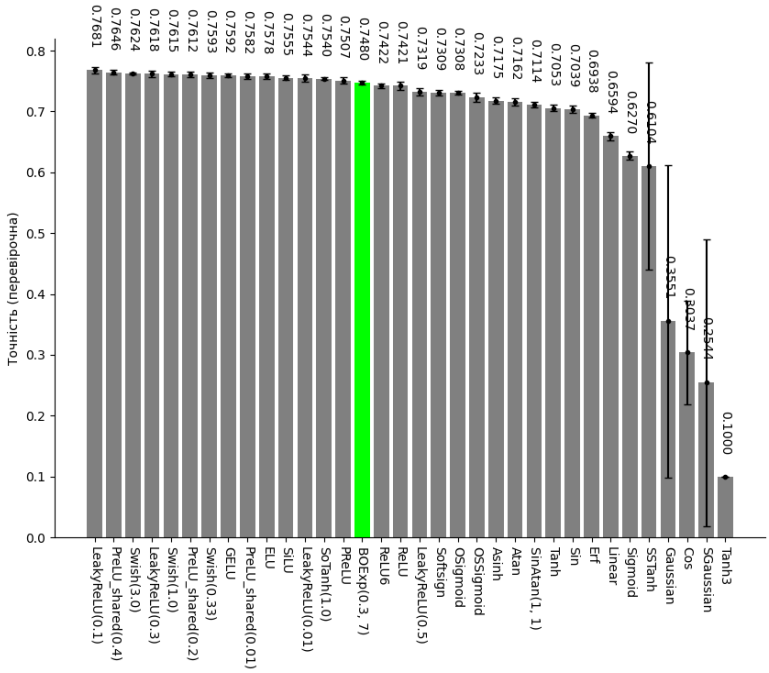


Рисунок 5.2 – Порівняння ефективності $BOExp$ із набором стандартних АФ

Як ми можемо бачити, на цій моделі такий варіант функції приблизно дорівнює за точністю класифікації стандартним функціям ReLU, ReLU-6, трохи перевершивши їх за усередненою точністю, а також функції PReLU, яка має трохи більшу середню точність.

5.2 Активаційна функція $LOExp$

Слід зазначити, що функція $BOExp$, яка описується в розділі попередньому розділі 5.1 має потенційний недолік у тому, що ділянка при значеннях, що перевищує певне значення x є горизонтальною з нульовою першою похідною, що потенційно може призводити до проблем навчання (проблема зникаючого градієнта). Тому в цьому розділі ми розглянемо модифіковану версію цієї АФ, яка змінює обмеження експоненційної функції із використання фіксованого значення на лінійну функцію, що має певне зростання, що може бути задано окремим коефіцієнтом. Префікс «LO» в назві означає Linearly-bound, Origin-aligned (лінійно-обмежена, співставлена з початком координат). Така функція може бути описана формулою (5.2).

$$LOExp(x) = \begin{cases} U + \beta(x - X_u), & x \geq X_u \\ e^{\alpha x} - 1, & x < X_u \end{cases} \Big|_{X_u = e^{U+1}/\alpha} \tag{5.2}$$

де U – це значення функції, починаючи з якого припиняється експоненційний зріст та починаючи з якого починається лінійний зріст;

α –масштабний коефіцієнт функції;

β – коефіцієнт нахилу лінійної частини функції.

Графік функції $LOExp$ наведено на рисунку 5.3.

Точність класифікації зображень CIFAR-10 із базовою тестовою конфігурацією (див. таблицю 3.1) у порівнянні із $BOExp$ та стандартними АФ можна побачити на рисунку 5.4.

За усередненою точністю $LOExp$ показала майже ідентичний але трохи більший результат порівняно із функцією $BOExp$, що у середньому близько за точністю до $ReLU$, ReLU-6 та $PReLU$.

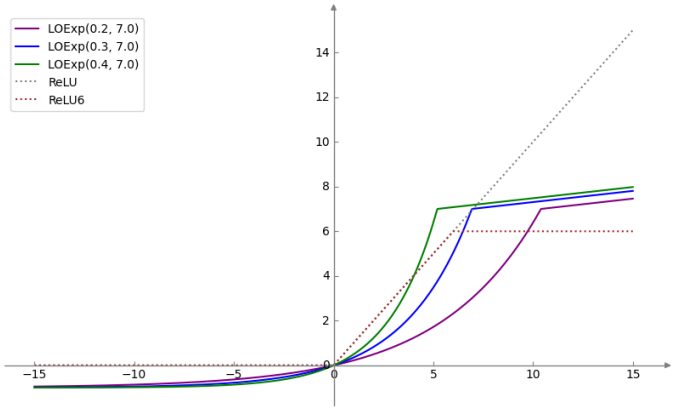


Рисунок 5.3 – Графік функції $LOExp$ із значеннями $\alpha = 0.2, 0.3, 0.4$, $\beta = 0.1$, та $U = 7.0$, співставлений із графіками АФ $ReLU$, $ReLU6$

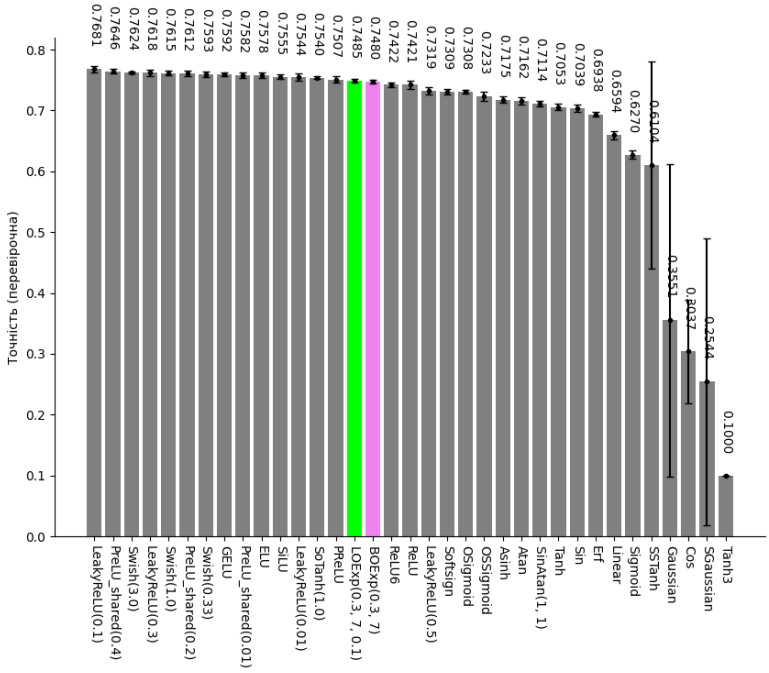


Рисунок 5.4 – Порівняння ефективності $LOExp$ із набором стандартних АФ, та $BOExp$

5.3 Активаційна функція $AOExp$

Додатковим експериментальним різновидом такої функції розглянемо інший різновид обмеження функції зверху – замість обмеження лінійною

функцією розглянемо обмеження функцією відповідним чином зсунутого арктангенсу (його додатної половини), що робить таке обмеження більш поступовим. Мотивацією для випробовування такого різновиду є припущення, що нелінійна функція може показувати кращу динаміку тренування на цій ділянці ніж лінійна, принаймні із оптимізаторами, що враховують другу похідну темпу навчання (Adam), а також така функція дещо «пом'якшує» зріст із зростанням значень параметру функції, порівняно з лінійною функцією. Така функція $AOExp$ (Arctan-limited Origin-aligned Exponent – експонента співставлена із початком координат та обмежена арктангенсом) має формулу (5.3).

$$AOExp(x) = \begin{cases} U + \beta arctan(x - X_u), & x \geq X_u \\ e^{\alpha x} - 1, & x < X_u \end{cases} |_{X_u = e^{U+1}/\alpha} \qquad (5.3)$$

де U – це значення функції, починаючи з якого припиняється експоненційний зріст, та починаючи з якого починається зріст за формулою арктагенсу;

α –масштабний коефіцієнт функції.

Арктангенс був обраний у ролі обмежуючої функції серед інших S-подібних функцій як такий, що має більш повільний зріст ніж всі інші S-подібні АФ, та має менший діапазон значень ніж $arcsinh$. Зовнішній вигляд функції $AOExp$ можна побачити на рисунку 5.5.

Розглядаючи ефективність цієї функції у порівнянні її із $BOExp$, $LOExp$ та стандартними АФ ми можемо бачити, що точність класифікації з використанням $AOExp$ незначною мірою в середньому в тестовій експериментальній конфігурації (таблиця 3.1) краща ніж $BOExp$ та $LOExp$, та, подібно до цих АФ, розташована між функціями $ReLU6$ та $PReLU$.

Результати відповідних порівнянь точності класифікації зображень CIFAR-10 з використанням цієї АФ на основній тестовій конфігурації ми можемо бачити на рисунку 5.6.

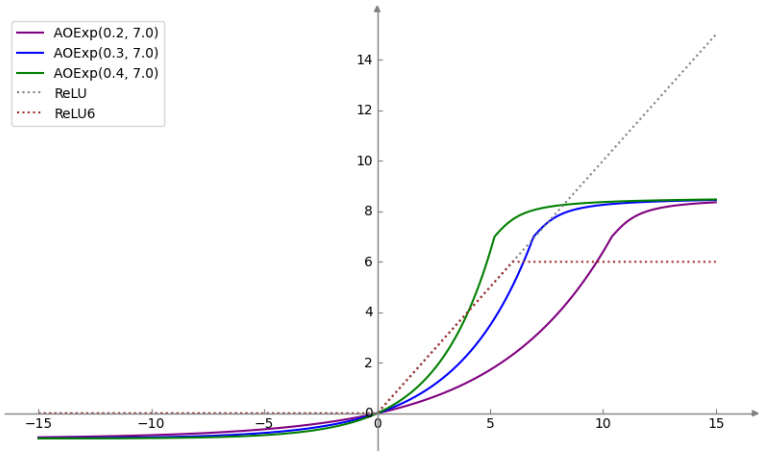


Рисунок 5.5 – Графік функції $AOExp$ із значеннями $\alpha = 0.2, 0.3, 0.4$, $\beta = 1.0$ та $U = 7.0$, співставлений із графіками АФ $ReLU$, $ReLU6$

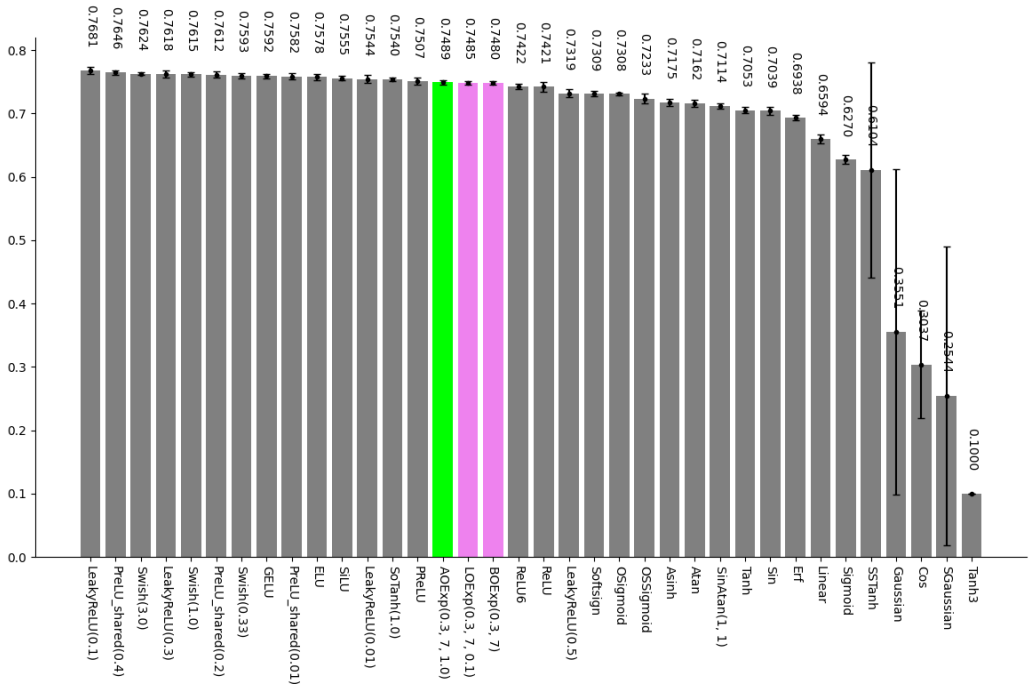


Рисунок 5.6 – Порівняння ефективності $AOExp$ із набором стандартних АФ, а також із $BOExp$ та $LOExp$

Як ми можемо бачити, заміна лінійного обмеження експоненційної АФ на більш поступове дійсно певною мірою покращило результат і показало вищу точність класифікації.

6 НОВІ АФ ІЗ ЗВАЖЕННЯМ ЗСУНУТИХ S-ПОДІБНИХ ФУНКЦІЙ

6.1 Загальна форма нових зважених АФ

В цьому розділі розглядаються нові активаційні функції, які є спробою комбінувати успішність зсунутих S-подібних функцій с перетином із точкою початка координат (див. розділ 4.1.1), із успіхом $ReLU$-подібних функцій, таких як $SiLU$, $GELU$, та інші. В основі створення відповідних нових АФ є наступні припущення:

– у зв'язку з тим, що такі зсунуті S-подібні функції як SoTanh, SoAtan, SoAsinh за своєю формою є схожими на функції, подібні до $BReLU$, $ReLU6$, або одна із форм $SReLU$ [16], такі нові АФ фактично можна вважати «м'якими» формами таких S-подібних ReLU функцій, та можна припустити, що в обох цих класах функцій спільні причини їх ефективності, що перевищують ефективність звичайних S-подібних функцій. Одне із припущень що це можуть бути за причини – це факт того, що $ReLU$-подібні функції відносно різко змінюють значення своєї першої похідної у точці $x = 0$ при незначних змінах на решті діапазонів значень (або у разі $SReLU$-подібних функцій мають ще один «перелам» у додатній частині графіку). Проте в рамках цього міркування точна причина є не стільки важливою, скільки факт зовнішньої схожості функцій, що супроводжується схожістю в ефективності;

– близькість від'ємної частини графіків функцій $ReLU$, $SiLU$, $GELU$, та ін. до нулю може бути однією з причин її ефективності;

– створення м'якого переходу від від'ємної до додатної частини функції можливо є одним із факторів того, що функції SiLU та $GELU$ можуть показувати кращі результати точності порівняно з $ReLU$.

Враховуючи здатність простого зважування лінійної функції $f(x) = x$ S-подібною функцією та отримання у результаті ефективну АФ, як у разі АФ $SiLU/Swish$ та $GELU$, ідея нового класу АФ, що розглядаються в цьому

розділі полягає в тому, щоб зважити іншу успішну категорію АФ (зсунуті, співставлені з початком координат S-подібні функції, що вже, як зазначено, мають деякі спільності з $BBreLU/SReLU$ та подібними функціями) аналогічним чином S-подібною функцією, та дослідити ефективність таких функцій у порівнянні з існуючими функціями.

Таким чином загальною загальним чином новий клас АФ, який розглядається в цьому розділі можна представити формулою (6.1).

$$F(x) = \gamma S(x)M(x), \tag{6.1}$$

де S –S-подібна функція, що має діапазон значень $S(x) \in (0;\ 1)$, та яка є симетричною навколо точки $(x = 0, y = 0.5)$;

M – зсунута та співставлена з початком координат S-подібна функція, яка має загальну форму (6.2);

γ – масштабний коефіцієнт функції.

$$M(x) = \beta(B(\alpha) + B(x/\beta - \alpha)), \tag{6.2}$$

де B – «базова» S-подібна функція, вибір якої фактично визначає динаміку зростання в додатній частині функції F;

α – горизонтальний зсув «базової» функції S, що фактично визначає ділянку цієї функції, що буде присутня в додатній частині функції F;

β – масштабний коефіцієнт зсунутої функції, що головним чином впливає на діапазон значень, який приймає функція F в її додатній частині, але і відповідним чином також впливає на розмір «виступу» функції в її від'ємній частині, який порівняно з додатною частиною є невеликим, втім, вірогідно потенційно здатним значно впивати на роботу АФ з поглядом на те, що ця ділянка від'ємної частини графіка близька до 0, та де попередньо може концентруватися більшість значень аргументу АФ для прихованих шарів мережі.

6.2 Конкретні варіанти нових зважених АФ, та їх властивості

В цій роботі в якості функції S у формулі (6.1) розглядаються такі два варіанти:

– функція логістичної сигмоїди (Logistic Sigmoid), див. формулу (6.3);

– функція помилок Гауса (Erf), модифікована таким чином щоб мати значення у діапазоні (0; 1), див. формулу (6.4).

$$S_{sig} = \frac{1}{1+e^{x}} \tag{6.3}$$

$$S_{erf} = \frac{1}{2}(Erf(x) + 1) \tag{6.4}$$

В якості «базової» функції B у формулі (6.2) були випробувані наступні функції:

– функція гіперболічного тангенсу ($tanh$);

– функція арктангенсу ($arctan$);

– функція гіперболічного арксінусу ($arcsinh$).

Були створені та випробувані функції, що наведені у таблиці 6.1.

Таблиця 6.1 – нові АФ із зваженням зсунутих S-подібних функцій

Назва АФ	Формула АФ
$GeSoTanh$	$\gamma\frac{1}{2}(Erf(x)+1)\,\beta(tanh(\alpha)+tanh(x/\beta-\alpha))$
$GeSoAtan$	$\gamma\frac{1}{2}(Erf(x)+1)\,\beta(arctan(\alpha)+arctan(x/\beta-\alpha))$
$GeSoAsinh$	$\gamma\frac{1}{2}(Erf(x)+1)\,\beta(arcsinh(\alpha)+arcsinh(x/\beta-\alpha))$
$SiSoTanh$	$\gamma\sigma(x)\,\beta(tanh(\alpha)+tanh(x/\beta-\alpha))$
$SiSoAtan$	$\gamma\sigma(x)\,\beta(arctan(\alpha)+arctan(x/\beta-\alpha))$
$SiSoAsinh$	$\gamma\sigma(x)\,\beta(arcsinh(\alpha)+arcsinh(x/\beta-\alpha))$

Графіки цих функцій із значеннями $\alpha = 1.0;\ \beta = 1.0;\ \gamma = 1.0$ зображені на рисунку 6.1.

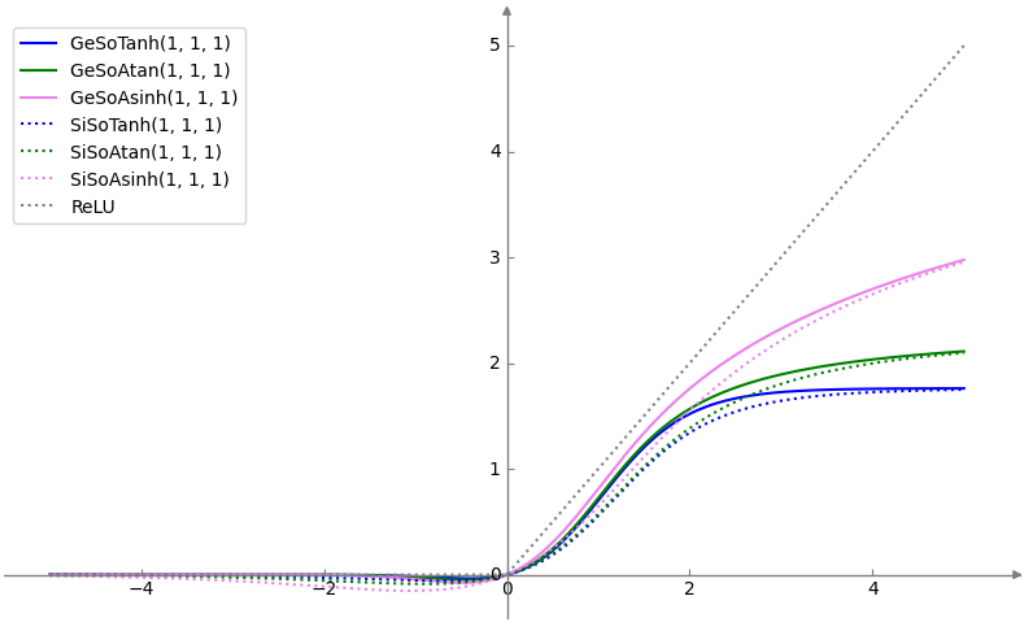


Рисунок 6.1 – Графіки АФ $GeSoTanh$, $GeSoAtan$, $GeSoAsinh$, $SiSoTanh$, $SiSoAtan$, $SoSoAsinh$ із значеннями всіх коефіцієнтів 1.0

Як ми можемо бачити, графіки цих функцій візуально схожі на функції обмежених варіантів ReLU-подібних функцій, що мають від'ємну частину близьку до 0, та початок додатної частини близький до функції $f(x) = x$, проте на відміну від BreLU, SReLU, та ін., це обмеження є «м'яким» із поступовим зменшенням темпу зросту.

Також поглянемо на відповідні графіки цих функцій із більшими значеннями $\alpha = 1.0;\ \beta = 10.0;\ \gamma = 2.0$, що зображені рисунку 6.2, який показує діапазон функції у проміжку $(-30; 30)$.

Як ми можемо бачити, таке масштабування цих функцій робить їх ще більше схожими до функцій $ReLU$, $GELU$, $SiLU$, та ін., проте також включає м'яке обмеження у районі великих значень x. Рисунок 6.3, що показує ці функції в великому масштабі, також демонструє їх схожість до обмежених зверху варіантів $ReLU$, але з м'яким обмеженням.

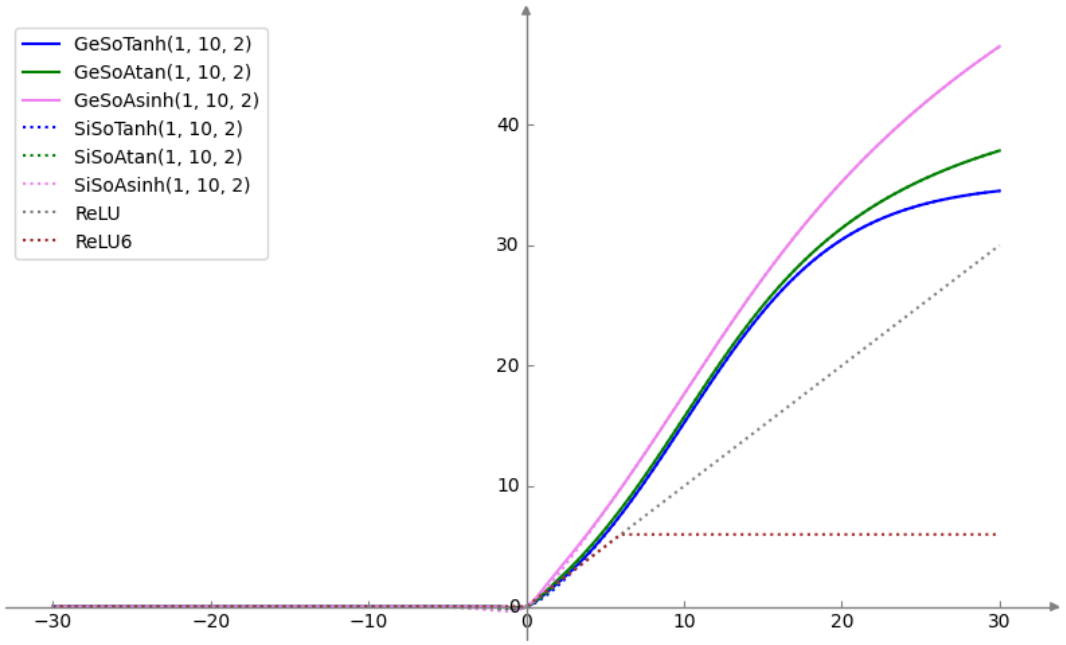


Рисунок 6.2 – Графіки АФ $GeSoTanh$, $GeSoAtan$, $GeSoAsinh$, $SiSoTanh$, $SiSoAtan$, $SoSoAsinh$ із значеннями $\alpha = 1.0;\ \beta = 10.0;\ \gamma = 2.0$ у діапазоні $x \in (-30; 30)$

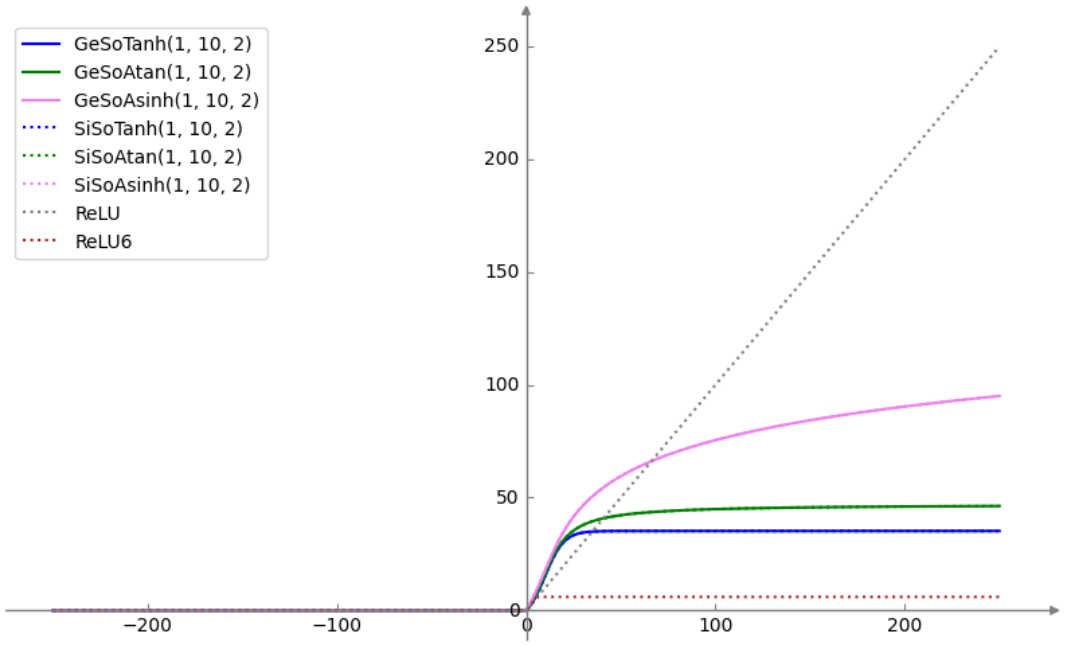


Рисунок 6.3 – Графіки АФ $GeSoTanh$, $GeSoAtan$, $GeSoAsinh$, $SiSoTanh$, $SiSoAtan$, $SoSoAsinh$ із значеннями $\alpha = 1.0;\ \beta = 10.0;\ \gamma = 2.0$ у діапазоні $x \in (-200; 200)$

Також слід зазначити, що варіанти цих АФ, що базуються на функції гіперболічного арксінусу (АФ $GeSoAsinh$, $SiSoAsinh$) не є обмеженими, та постійно зростають, хоча із все меншим темпом, що також добре видно на рисунку 6.3. Це робить верхнє обмеження відповідних функцій ще менш вираженим, та зберігає відносно великі значення похилу функції (її першої похідної), що теоретично може бути корисним у певних обставинах для запобігання проблемі зникаючого градієнта.

Щодо різниці між варіантами функцій, що зважені сигмоїдою ($SiSoTanh$, $SiSoAtan$, $SiSoAsinh$), та функцією помилок Гауса ($GeSoTanh$, $GeSoAtan$, $GeSoAsinh$), на графіках, зображених на рисунках 6.1 – 6.4 можна бачити, що різниця між цими категоріями функцій більше помітна тільки у діапазоні значень x, що є відносно близькими до 0, та на великих масштабах ці категорій виглядають однаково. Втім, саме ця різниця може бути важливою, адже саме в близькості до нуля ми бачимо найбільш значну різницю в від'ємній частині графіку.

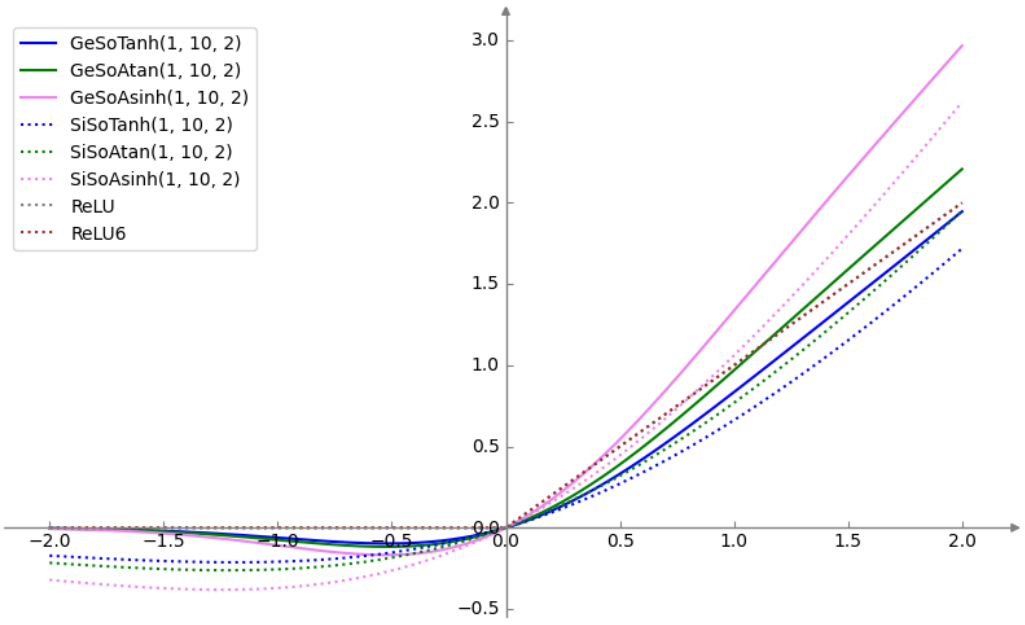


Рисунок 6.4 – Графіки АФ $GeSoTanh$, $GeSoAtan$, $GeSoAsinh$, $SiSoTanh$, $SiSoAtan$, $SoSoAsinh$ із значеннями $\alpha = 1.0;\ \beta = 10.0;\ \gamma = 2.0$ у діапазоні $x \in (-2.0; 2.0)$

6.3 Точність класифікації з новими зваженими АФ

Подивимось на точність класифікації зображень у основній тестовій конфігурації (див. таблицю 3.1) із усіма шістьма новими зваженими АФ. На рисунках 6.5 та 6.6 показані однакові заміри із різницею в підсвітці категорій нових функцій. Загалом ми можемо бачити, що 4 із цих варіантів конфігурації нових АФ виявилися гірше ніж $ReLU$, та інші 12 виявилися краще ніж $ReLU$, що попередньо можна вважати добрим результатом на цій тестовій конфігурації. Щодо закономірностей на цьому наборі важко побачити однозначні результати, проте ми можемо бачити, що гіршими ніж $ReLU$ виявилися варіанти функцій $GeSoAtan/GeSoTan$ та $SiSoAtan/SiSoTan$, що мають відносно невеликі значення для параметрів β у діапазоні (1.2 та 1.5), та помірні значення параметрів γ (3.2 та 3.64).

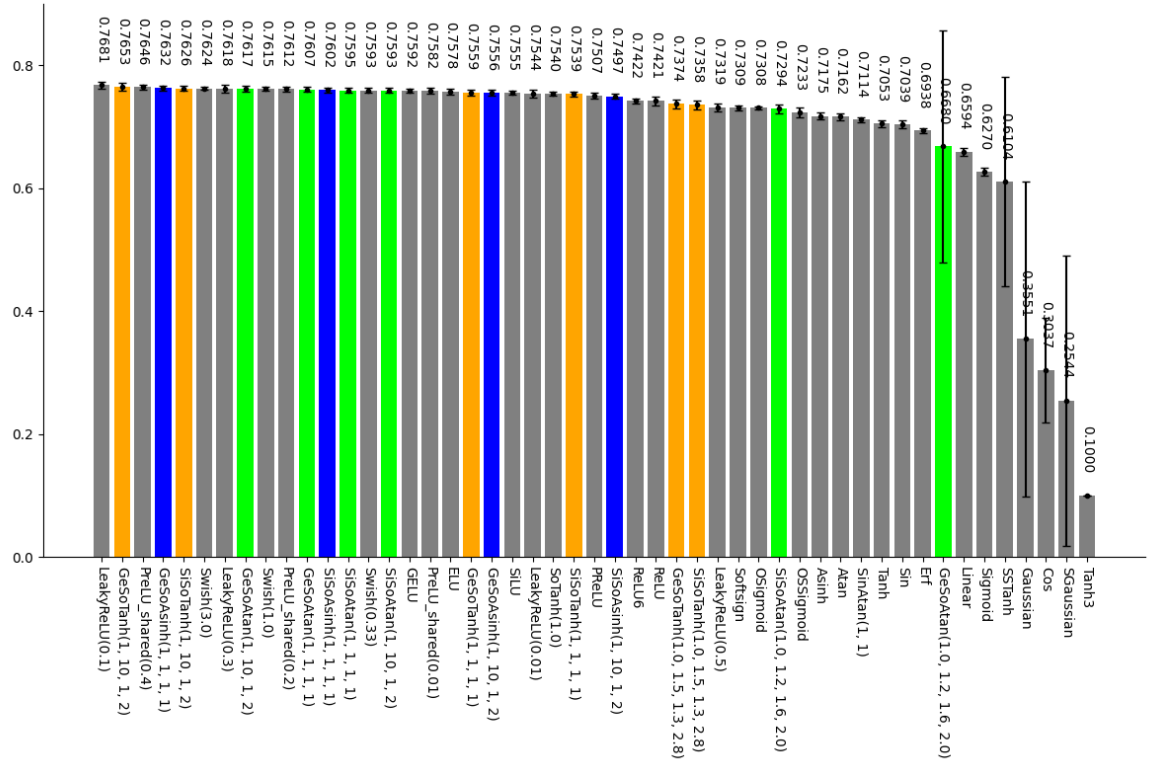


Рисунок 6.5 – Розподіл оцінок точності нових зважених АФ порівняно із існуючими АФ (колір відображає кожну із трьох базових S-функцій)

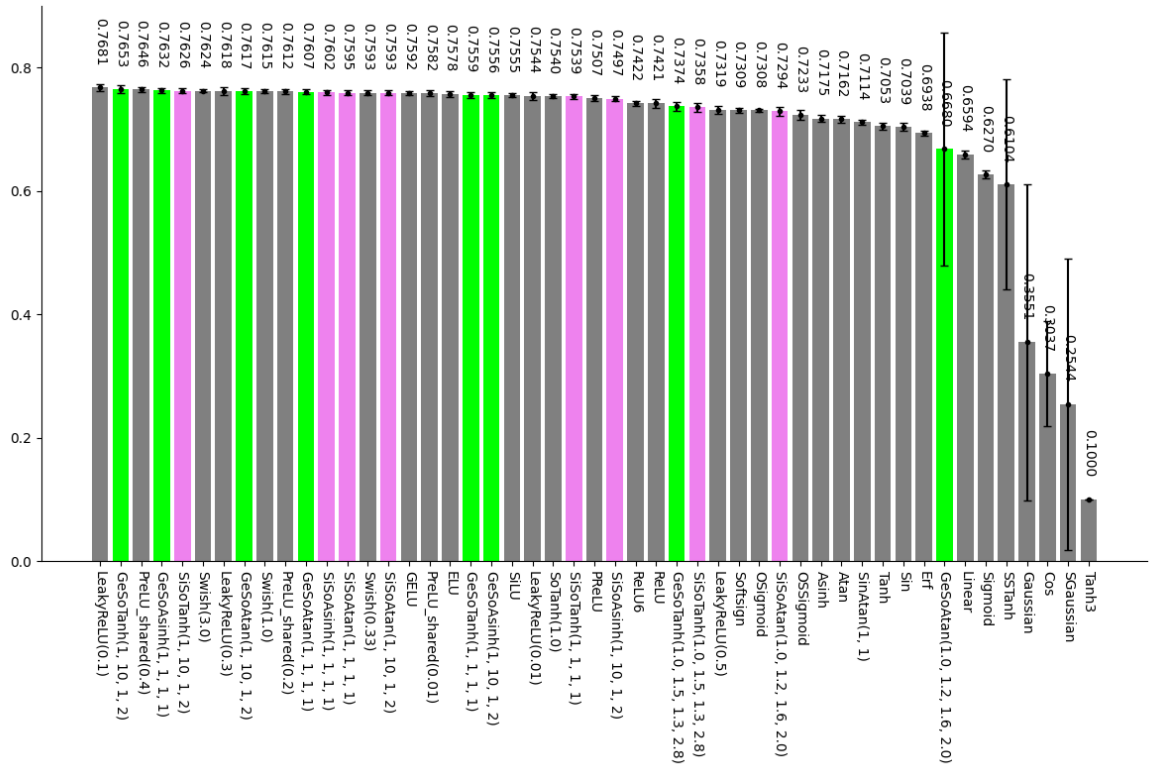


Рисунок 6.6 – Розподіл оцінок точності нових зважених АФ порівняно із існуючими АФ (колір відображає розподіл на GeSo* та SiSo* функції)

Проте ті ж функції із значно меншими (1), або значно більшими (10) значеннями параметру β показали кращі результати. Всі варіанти функції $SiSoAsinh$ показали кращі результати ніж функція $ReLU$.

Щодо порівняння ефективності між $GeSo-$ та $SiSo-$ функціями (рисунок 6.5) також важко бачити однозначний результат, проте дві найкращих в цьому тесті функції виявилися $GeSo-$ варіантами. Всі великі значення параметру $\beta = 10$ виявилися кращі ніж ReLU. Три переможця із цього тестового заміру є такими конкретними конфігураціями функцій:

– $GeSoTanh(\alpha = 1, \beta = 10, \gamma = 2)$;

– $GeSoAsinh(\alpha = 1, \beta = 1, \gamma = 1)$;

– $SiSoTanh(\alpha = 1, \beta = 10, \gamma = 2)$.

7 АДАПТИВНІ ВЕРСІЇ НОВИХ АФ

В розділах 4 – 6 були представлені модифіковані та нові активаційні функції, які між іншим включають зсунуті співставлені з початком координат S-подібні АФ (див. розділ 4.1.1), та нові зважені S-подібні функції, що розглядаються в розділі 6. В цьому розділі ми розглянемо адаптивні модифікації таких АФ та порівняємо їх ефективність із існуючими АФ, та неадаптивними варіантами цих АФ. Загалом адаптивні АФ (ААФ) у додаток до звичайних АФ мають один або більше параметрів, що навчаються під час тренування моделі подібно до синаптичних ваг моделі. Втім, то, яким чином змінні для цих параметрів розподілені між нейронами мережі (та їх відповідними активаційними функціями) може відрізнятись в залежності від потреб, та реалізації відповідних АФ. Серед іншого, можливо принаймні два варіанту того, як відповідні змінні можуть зберігатися: можна мати відповідний набір параметрів ААФ для кожного нейрону мережі, або можна мати змінні для цих параметрів, які є загальними для всієї мережі, та у такому разі, фактично, ААФ всіх нейронів у мережі використовують одні і ті ж значення параметрів, та, відповідно, усі задіяні в тренуванні цих значень. В цьому розділі розглядаються тільки такі реалізації ААФ, що мають один набор параметрів для тренування для всієї мережі.

7.1 Адаптивні зсунуті співставлені S-подібні АФ

Адаптивні активаційні функції $ASoTanh$, $ASoAtan$, $ASoAsinh$ фактично є ідентичними їх звичайним аналогам, що мають параметр горизонтального зсуву α з тим виключенням, що цей параметр не є зафіксованим, а навчається із тренуванням мережі (див. формули (4.3) – 4.5)).

На рисунку 7.1 приведені результати усереднених замірів точності класифікації зображень із набору даних CIFAR-10 на основній тестовій конфігурації (див. таблицю 3.1.1), включаючи порівняння цих ААФ із їх неадаптивними варіантами та іншими існуючими АФ.

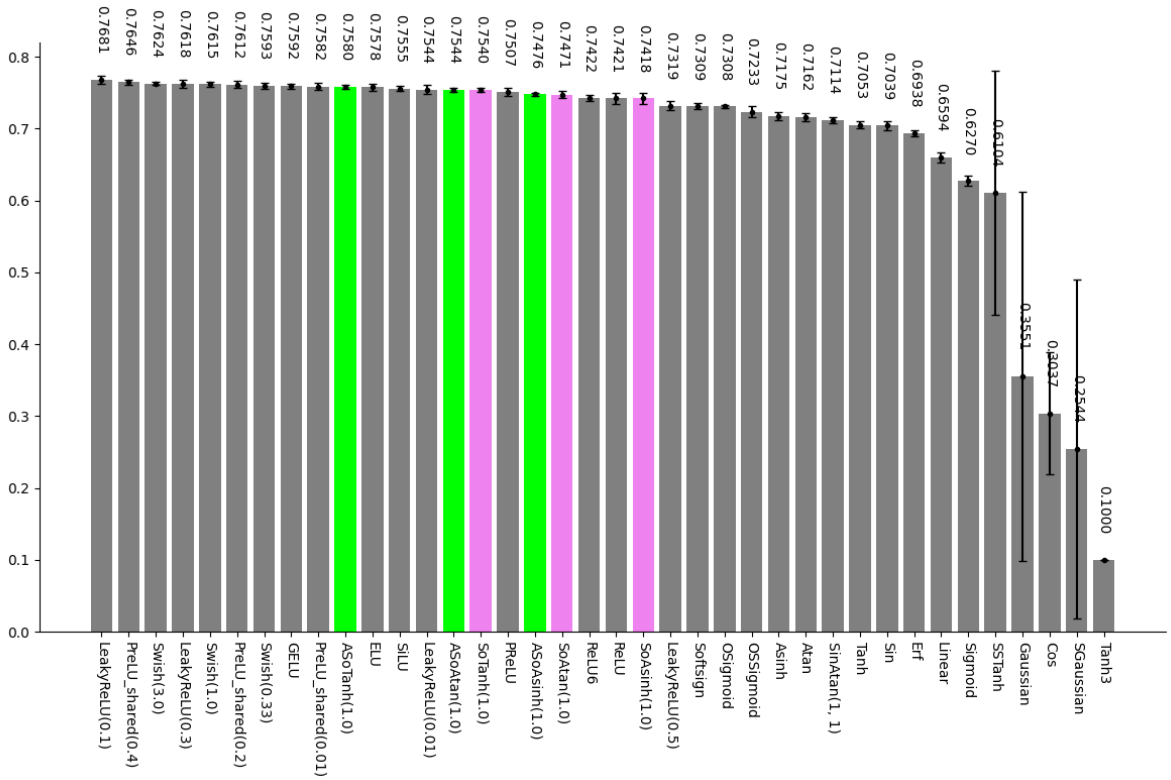


Рисунок 7.1 – Результати заміру точності функцій $ASoTanh$, $ASoAtan$, $ASoSinh$ (зелені стовпці) у порівнянні із неадаптивними версіями цих АФ (фіолетові стовпці), та існуючими АФ (сірі стовпці)

Як ми можемо бачити, кожна із адаптивних версій цих АФ перевершує відповідний неадаптивний варіант АФ, що покращує ранг точності їх класифікації на декілька сходинок, обходячи $ReLU$ та $ReLU6$ для всіх цих ААФ, а найкраща з них ААФ $ASoTanh$ за цією серією тестових замірів у середньому виявилась трохи кращою за функції ELU та $SiLU$.

Поглянемо також на відповідні графіки функцій, що належать відповідним ААФ, із параметрами, що були визначені в однієї з таких натренованих моделей (див. рисунок 7.2).

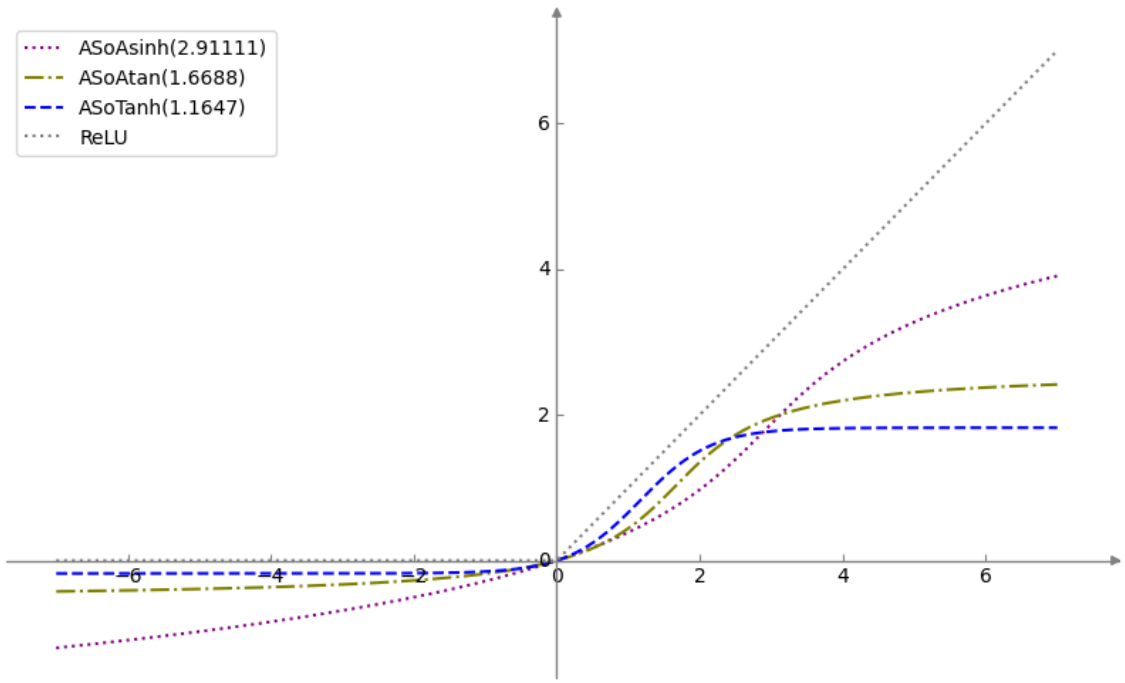


Рисунок 7.2 – Графіки ААФ з параметрами, що були автоматично визначені після тренування моделі

Як ми можемо бачити у випадках всіх цих ААФ, «базові» S-подібні функції були зсунуті дещо більше ніж початкове значення 1.0 перед тренуванням, що між іншим призводить до приближення значень цих ААФ у їх від'ємній частині до нуля. Такий зсув є особливо вираженим із функцією $ASoAsinh$, що, вірогідно, пов'язано із більшим (фактично необмеженим, або «м'яко» обмеженим) загальним діапазоном значень цієї ААФ.

7.2 Адаптивні версії нових зважених S-подібні АФ

Аналогічним чином розглянемо адаптивні варіанти нових зважених функцій $GeSoTanh$, $GeSoAtan$, $GeSoAsinh$, $SiSoTanh$, $SiSoAtan$,

$SiSoAsinh$. Адаптивні варіанти цих АФ (додаючи до назви префікс «А», мають назви $AGeSoTanh$, $AGeSoAtan$, $AGeSoAsinh$, $ASiSoTanh$, $ASiSoAtan$, $ASiSoAsinh$) також є ідентичними за формулою та набором параметрів до їх неадаптивних аналогів, але їх параметри також не зафіксовані та беруть участь у тренуванні мережі.

Усереднені заміри точності класифікації зображень CIFAR-10 на основній тестовій конфігурації із використанням нових зважених ААФ можна бачити на рисунку 7.3.

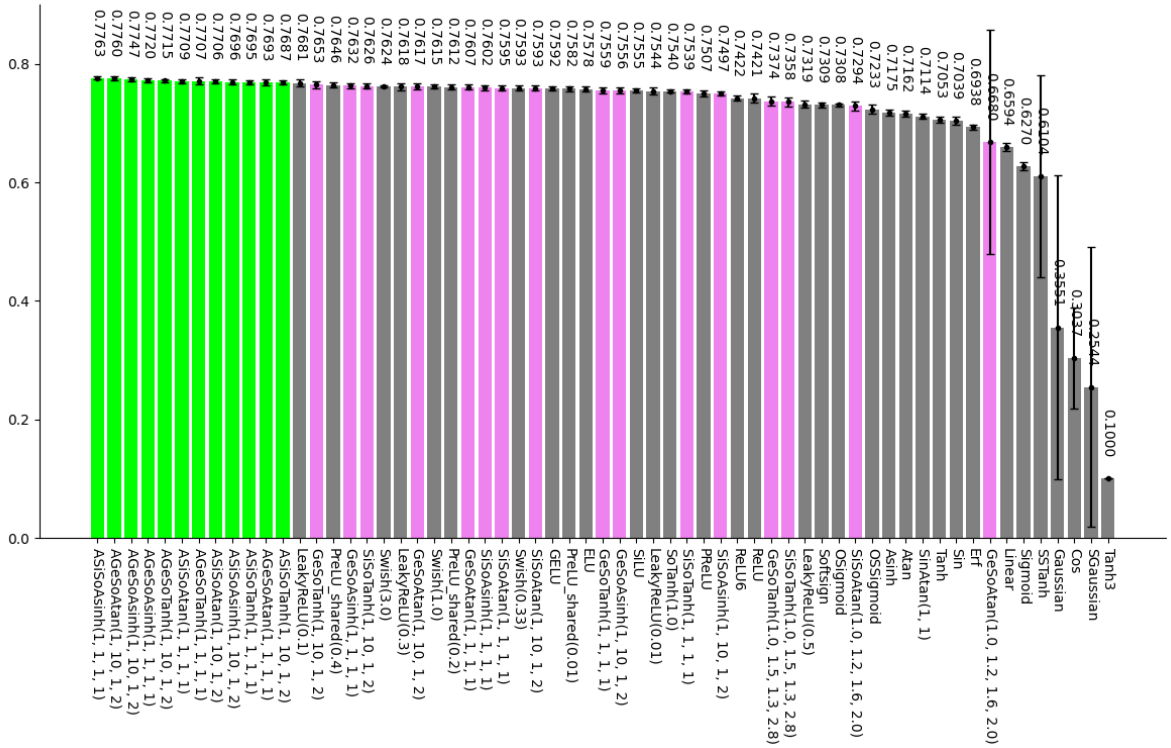


Рисунок 7.3 – результати заміру точності функцій нових зважених ААФ (зелені стовпці), у порівнянні з їх неадаптивними варіантами (фіолетові стовпці, та існуючими АФ (сірі стовпці)

Як ми можемо бачити усі відповідні ААФ у цій тестовій сесії у середньому перевершують всі існуючі АФ, які були розглянуті в цій роботі та неадаптивні версії цих ААФ.

Щодо значень параметрів зважених функцій на зображеннях, у разі ААФ, вони задають початкові значення параметрів α, β, та γ відповідно, див. формули (6.1) та (6.2). При цьому слід зауважити, що чотири параметри на зображенні є наслідком помилково доданого зайвого параметру. При цьому, якщо вважати третій та четвертий параметри зважених функцій на зображенні як γ_1 та γ_2, то параметр γ у формулі 6.1 фактично був замінений на добуток $\gamma_1\gamma_2$. З математичної точки зору це не змінює поведінку функції, та ми повинні це враховувати для правильної інтерпретації відповідних визначень функцій на цих графіках точності класифікації.

7.2.1 Попередній аналіз деяких успішних конфігурацій АФ та ААФ

Розглянемо графіки деяких вибіркових з найбільш успішних, та менш успішних конфігурацій нових зважених ААФ та АФ. На рисунках 7.4, та 7.5 показані графіки ААФ із параметрами, що були визначені як результат тренування мережі, співставлені із графіками декількох неадаптивних конфігурацій АФ із визначенням точності класифікації з кожною із ААФ/АФ. Обидва рисунка містять однакові ААФ/АФ з різними масштабами.

Цільними лініями на цьому рисунку зображені графіки ААФ, що, як можна бачити, мають значення точності у найвищому діапазоні, а переривчасті графіки відповідають неадаптивним варіантам АФ з різними наборами параметрів, що, як можна бачити мають помітно нижчу точність класифікації.

Щодо закономірностей, та попередніх гіпотез щодо причин ефективності деяких конфігурацій АФ/ААФ, ми можемо припустити, що ті варіанти, які на значному діапазоні у додатній частині графіку зберігають максимальну близькість до функції $f(x) = x$ є більш ефективними.

Проте, як ми можемо бачити, є деякі виключення, що мають дуже несхожі графіки, при цьому також маючи точність у найвищому діапазоні,

як у разі конфігурацій ААФ $ASiSoAsinh(0.12, 0.92, 0.56)$, та $ASiSoTanh(0.38, 1.35, 0.54)$, та виявлення причини ефективності таких конфігурацій ААФ зокрема, потребує більш глибокого подальшого аналізу.

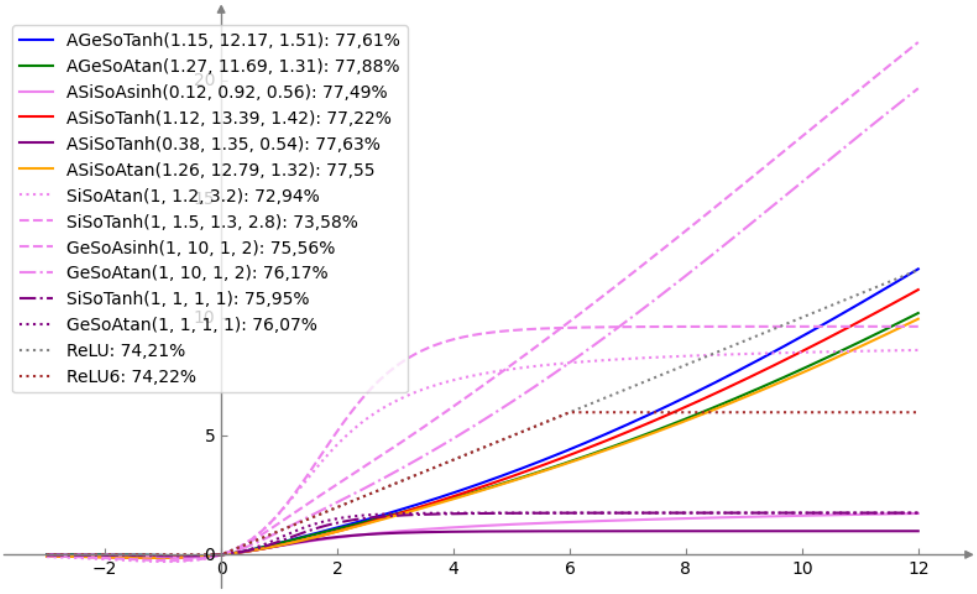


Рисунок 7.4 – Графіки деяких конфігурацій ААФ та АФ із відповідними визначеннями точності класифікації з цими ААФ/АФ

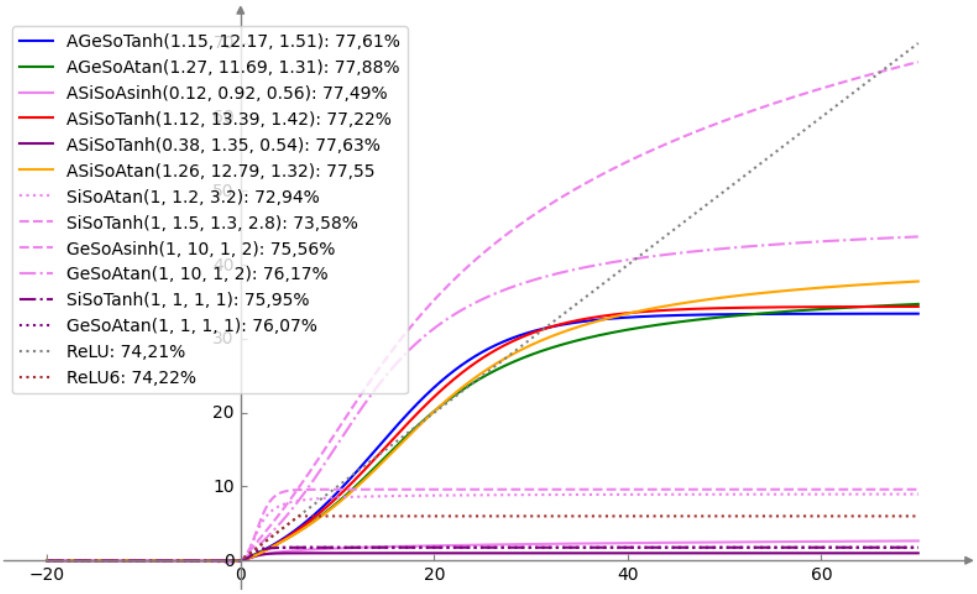


Рисунок 7.5 – Графіки деяких конфігурацій ААФ та АФ із відповідними визначеннями точності класифікації з цими ААФ/АФ (великий масштаб)

8 ЗАГАЛЬНЕ ПОРІВНЯННЯ ЕФЕКТИВНОСТІ НОВИХ ТА ІСНУЮЧИХ АФ

8.1 Оцінки точності усіх АФ в межах класифікації CIFAR-10

Поглянемо на усереднені результати замірів точності для всіх АФ для додаткових конфігурацій в межах класифікації CIFAR-10, включаючи використання оптимізаторів Adam та SGD, а також використання збільшеного вдвічі темпу навчання з кожним із оптимізаторів.

Загалом ефективність кожної функції була усереднена за результатами кількох замірів (тренувань нових моделей) для чотирьох експериментальних конфігурацій з CIFAR-10 (див. таблиці 3.1.1 та 3.1.2). Дивіться рисунки 8.1 – 8.4 для перегляду зведених оцінок точності всіх АФ в кожній з цих конфігурацій.

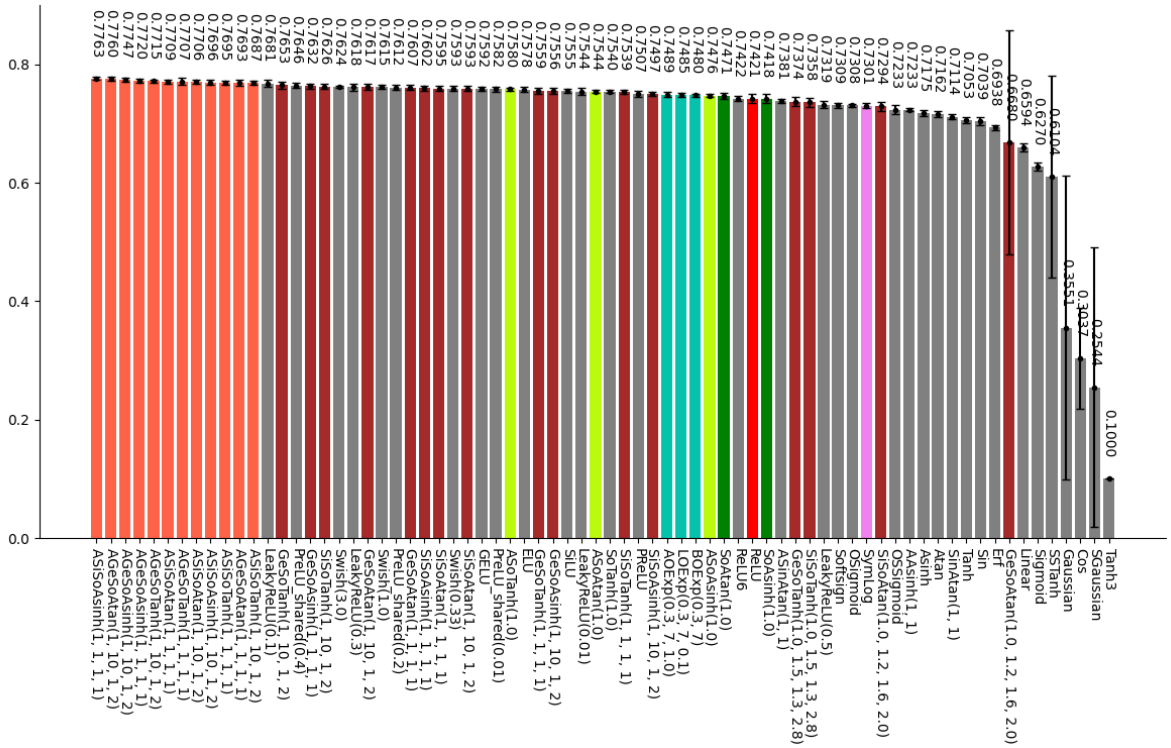


Рисунок 8.1 – Результати замірів точності всіх АФ для класифікації зображень CIFAR-10 з оптимізатором Adam та темпом навчання 0.001

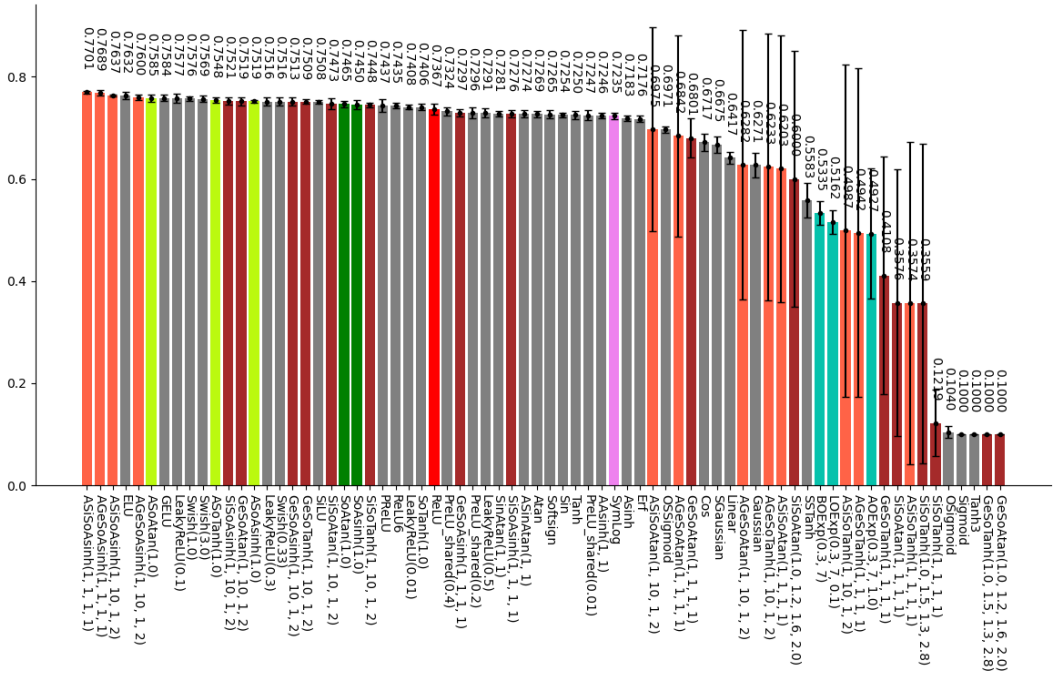


Рисунок 8.2 – Результати замірів точності всіх АФ для класифікації зображень CIFAR-10 з оптимізатором SGD та темпом навчання 0.03

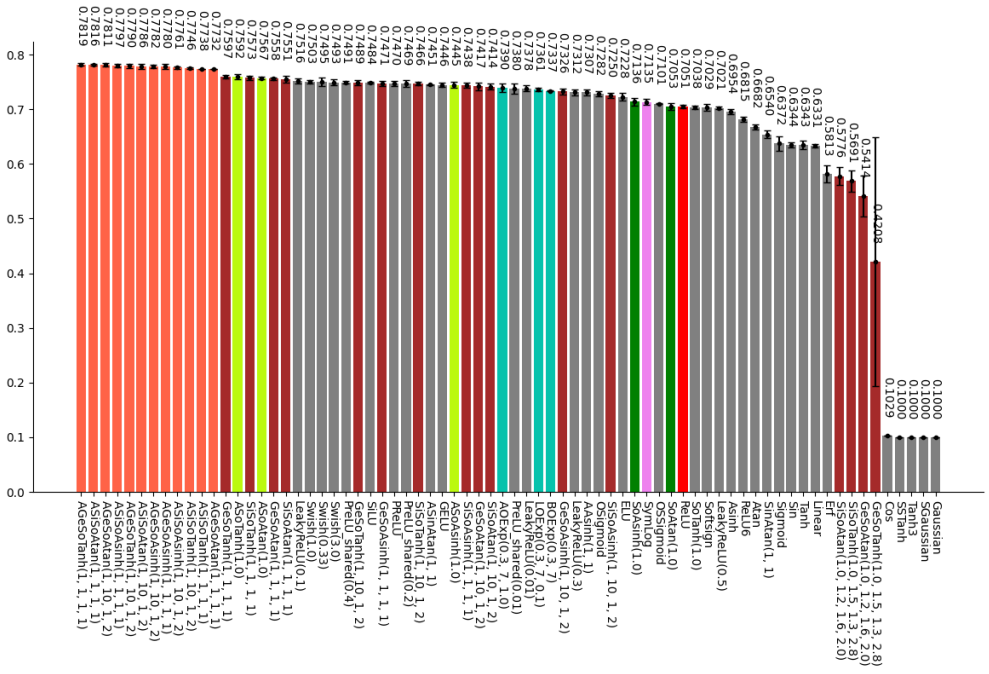


Рисунок 8.3 – Результати замірів точності всіх АФ для класифікації зображень CIFAR-10 з оптимізатором Adam та темпом навчання 0.002

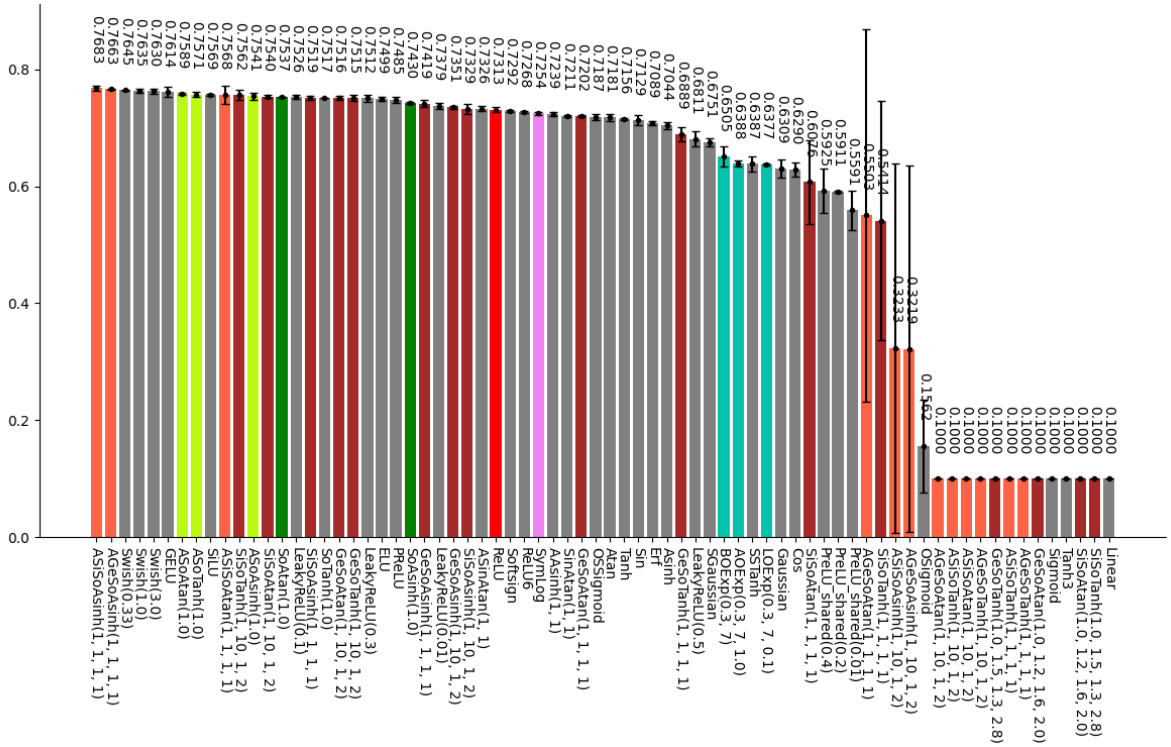


Рисунок 8.4 – Результати замірів точності всіх АФ для класифікації зображень CIFAR-10 з оптимізатором SGD та темпом навчання 0.06

Всі кольорові стовпчики на цих діаграмах відповідають новим АФ та ААФ, окрім красного, який відповідає до ReLU. Сірі стовпчики відповідають до існуючих АФ та ААФ.

Як ми можемо бачити на перших місцях у всіх цих замірів розташовуються ААФ із «родини» нових зважених ААФ. При цьому деякі АФ призводять до виникнення проблеми вибухаючого градієнту (всі АФ, точність яких дорівнює, або приблизно дорівнює до 0.1). Кількість функцій з вибухаючим градієнтом збільшується із збільшенням темпу навчання.

Також, розглядаючи точність нових ААФ з оптимізатором SGD, ми можемо бачити, що з цим оптимізатором, на відміну від Adam нові зважені ААФ не є дуже стабільними (показуючи гірші результати, іноді призводячи до появи вибухаючого градієнту). Втім, одна із функцій в цій серії експериментів показала незмінно добрі результати у всіх цих конфігураціях. Це ААФ $ASiSoAsinh$ із початковим значенням всіх її параметрів 1.0.

Подивимось також на відповідну кількісну різницю в середньої точності класифікації зображень з набора даних CIFAR-10 з використанням деяких нових, та стандартних АФ. В таблиці 8.1 наведена різниця відповідних значень точності, що включає три нових АФ, які показали найбільші значення точності у цій конфігурації, та деякі найбільш популярні із стандартних АФ (включаючи найкращий у цій серії експериментів із них). А разі якщо деяка популярна АФ була присутня в декількох варіантах параметрів, був включений той варіант, який показав найкращу точність в цій конфігурації. На цій тестовій конфігурації, варіант АФ $LeakyReLU(0.1)$ показав найкращий результат із всіх стандартних АФ.

Таблиця 8.1 – Різниця в точності класифікації за основною тестовою конфігурацією CIFAR-10 між деякими новими та стандартними АФ

	$LeakyReLU$ (0.1)	$Swish$ (3.0)	$PreLU\_shared$ (0.4)	$GELU$	$SiLU$	$ReLU$	$Sigmoid$
$ASiSoAsinh$ (1,1,1)	0.82%	1.39%	1.17%	1.71%	2.08%	3.42%	14.93%
$AGeSoAtan$ $(1, 10, 2)$	0.80%	1.36%	1.14%	1.69%	2.05%	3.40%	14.91%
$AGeSoAsinh$ $(1, 10, 2)$	0.66%	1.22%	1.01%	1.55%	1.91%	3.26%	14.77%

Як ми можемо бачити, нові АФ перевершили за точністю класифікації найкаращий варіант із переліку стандартних АФ ($LeakyReLU(0.1)$) на величину 0.66% – 0.82%.

8.2 Ранги ефективності всіх ААФ в межах класифікації CIFAR-10

Як було зазначено в розділі 3.5.2, ранги ефективності можуть бути зручною мірою оцінки ефективності АФ у порівнянні з іншими та дозволяє їх усереднення між різними задачами або її конфігураціями, для яких самі

оцінки ефективності (наприклад точність) не можуть бути співставлені між такими задачами.

Враховуючи значну різницю в тому, як ефективність різних АФ та ААФ змінюється із зміною оптимізатора, по-перше може бути корисним мати оцінки рангів ефективності по кожному оптимізатору окремо.

На рисунку 8.5 показана діаграма рангів ефективності класифікації для всіх функцій із оптимізатором Adam, та на рисунку 8.6 зображена аналогічна діаграма для оптимізатора SGD.

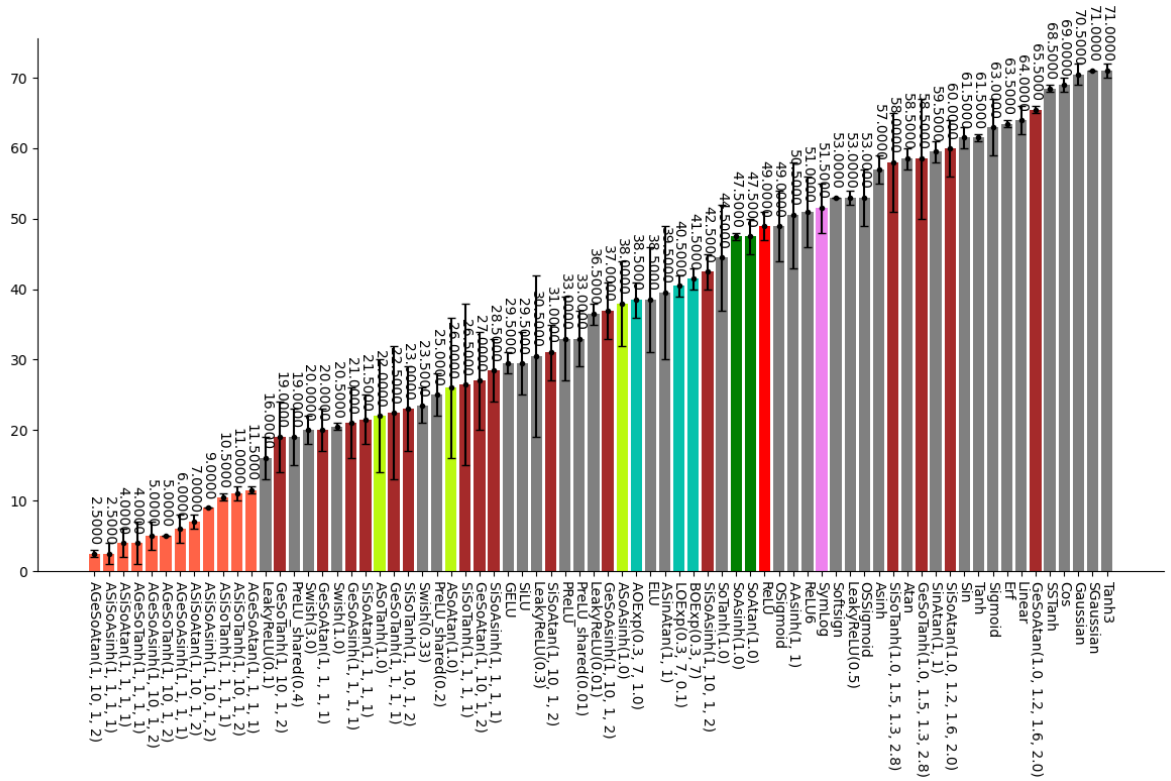


Рисунок 8.5 – Оцінка рангів ефективності всіх АФ із набором CIFAR-10, оптимізатором Adam із двома рівнями темпу навчання (менше – краще)

Як ми також можемо бачити на рисунку 8.6, більшість конфігурацій нових зважених ААФ не є дуже ефективними з оптимізатором SGD, при цьому маючи дві конфігурації із використанням $ASiSoAsinh$ та $AGeSoAsinh$, які при цьому показують найкращу точність з SGD при

встановленні початкових значень всіх їх параметрів у значення 1.0. Ці факти означають, що усі нові зважені АФ та ААФ є дуже чутливими до початкових значень параметрів при використанні із SGD, що має враховуватись при використанні цих АФ та ААФ з цим оптимізатором.

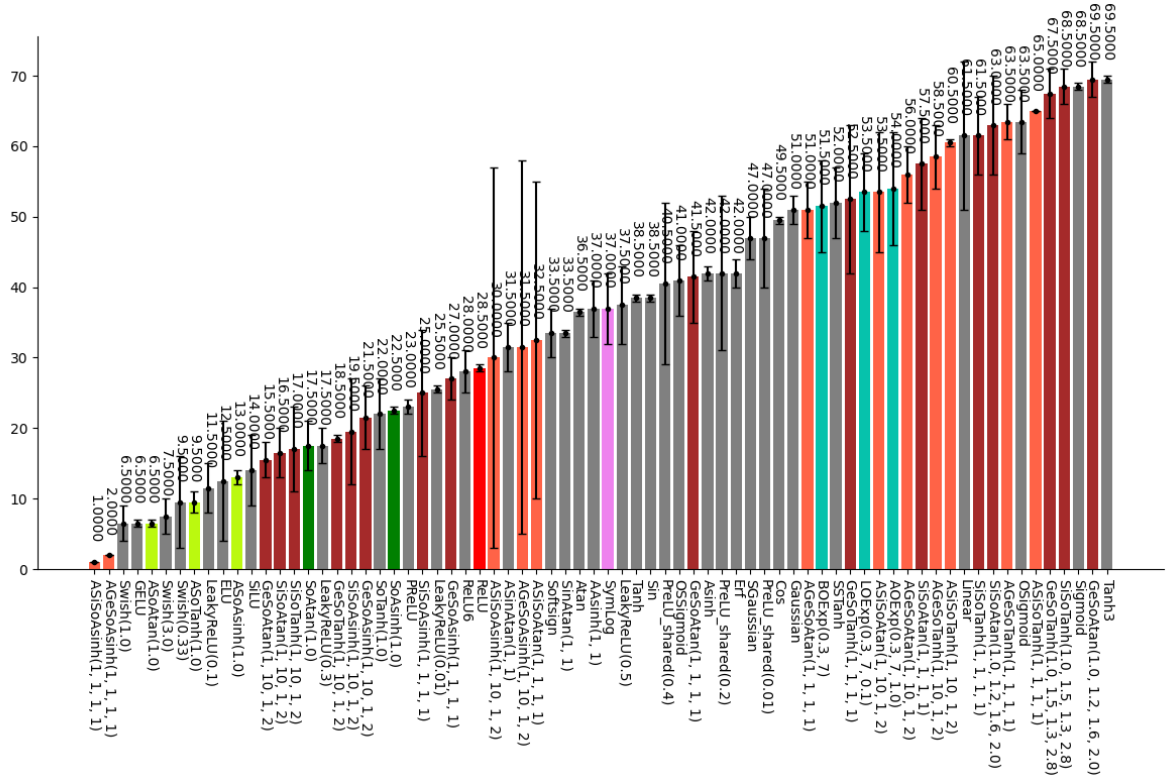


Рисунок 8.6 – Оцінка рангів ефективності всіх АФ із набором CIFAR-10, оптимізатором SGD із двома рівнями темпу навчання (менше – краще)

Загальні ранги ефективності класифікації зображень CIFAR-10, що враховують як результати експериментів на обох оптимізаторах так і результати експериментів з двома рівнями темпів навчання показані на рисунку 8.7.

Як ми можемо бачити, за таким порівнянням більшість рангів ефективності має дуже значний розкид. При цьому ААФ $ASiSoAsinh(1,1,1)$ та $AGeGeAsinh(1,1,1)$ зберегли стабільно найкращі результати за всіма цими чотирма конфігураціями.

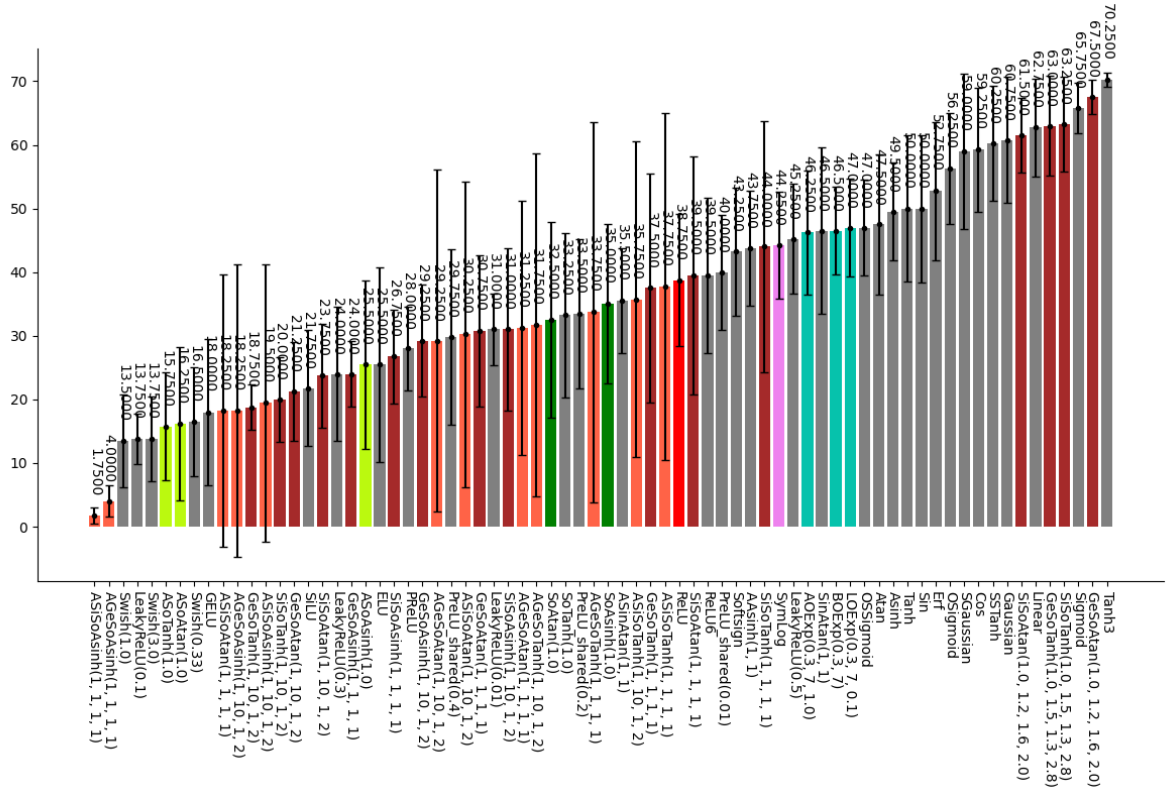


Рисунок 8.7 – Оцінка рангів ефективності всіх АФ із набором CIFAR-10, включаючи всі експерименти з двома оптимізаторами та двома темпами навчання (менше – краще)

8.2 Оцінки точності деяких АФ з іншими наборами даних

В рамках цієї роботи також була проведена часткова серія експериментів із наборами даних MNIST та Fashion-MNIST. Ці експерименти не містили перевірки повного набор АФ для цих конфігурацій, тому вони на поточному стані не можуть бути співставлені в загальному зведеному порівнянні рангів ефективності АФ, який міг би включати всі моделі.

Втім, результати цих експериментальних замірів можна бачити на рисунках 8.8 – 8.12. Слід зазначити, що значення точності, що дорівнюють 0 на цих зображення означають, що відповідна функція на цей час не була протестована із відповідною конфігурацією.

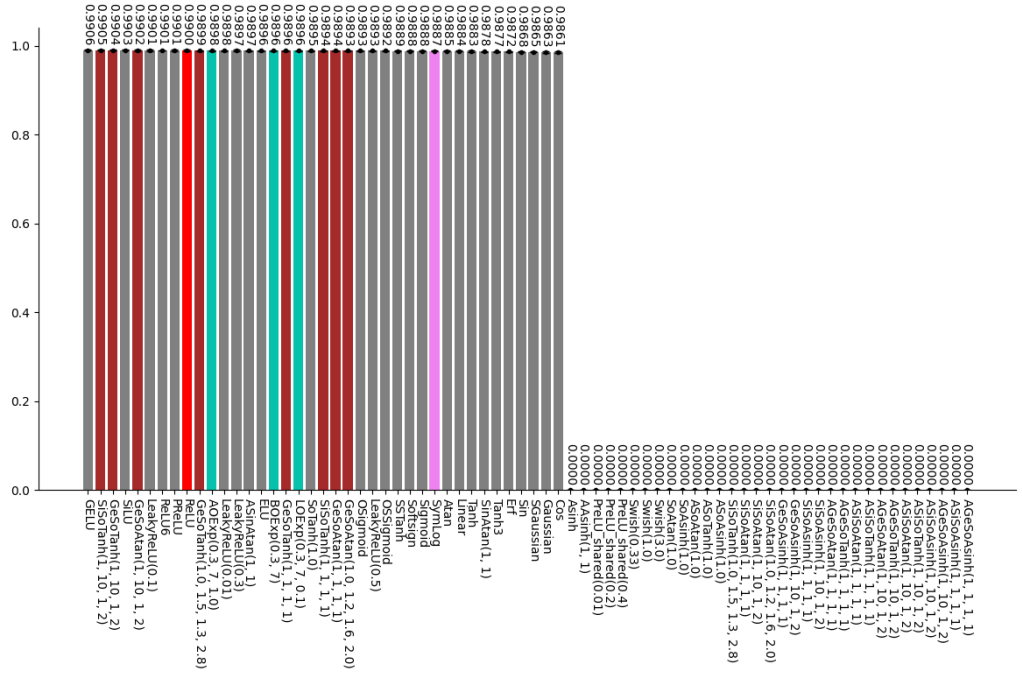


Рисунок 8.8 – Результати замірів точності деяких АФ для класифікації зображень MNIST з оптимізатором Adam та темпом навчання 0.001

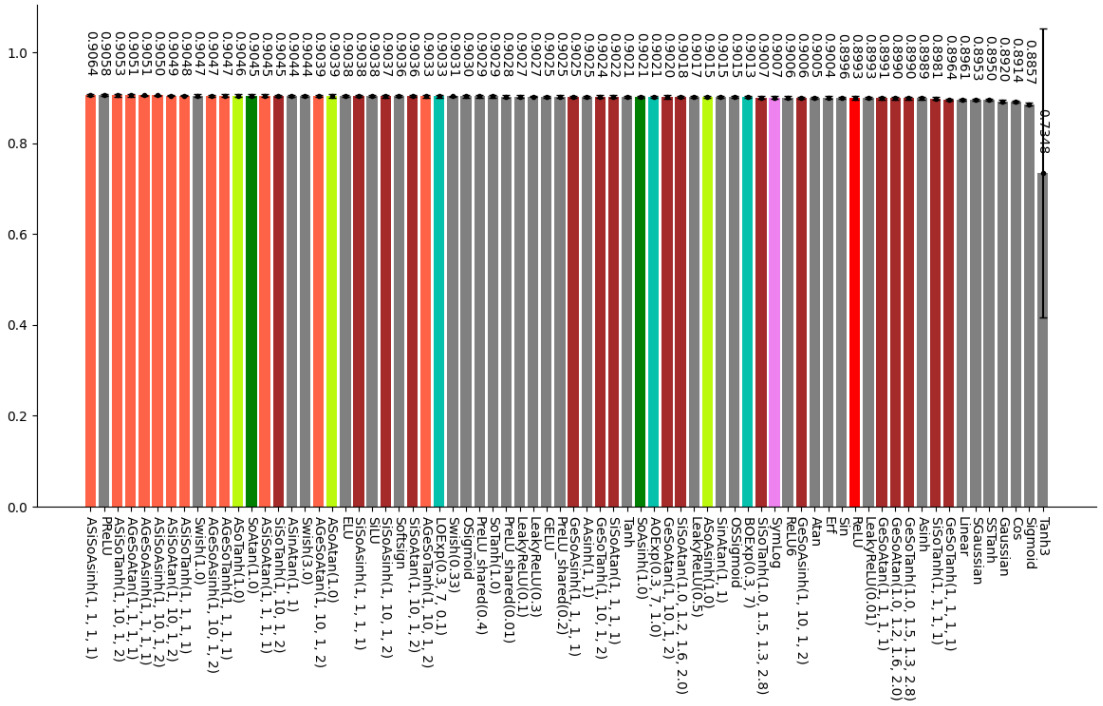


Рисунок 8.9 – Результати замірів точності деяких АФ для класифікації зображень Fashion-MNIST з оптимізатором Adam та темпом навчання 0.001

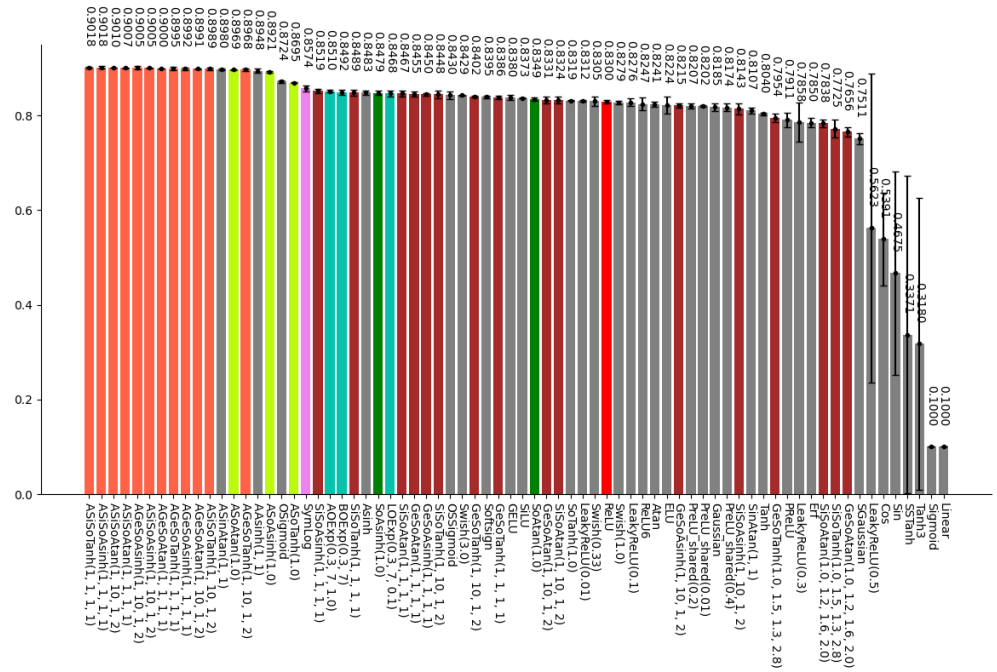


Рисунок 8.10 – Результати замірів точності деяких АФ для класифікації зображень Fashion-MNIST з оптимізатором Adam та темпом навчання 0.01

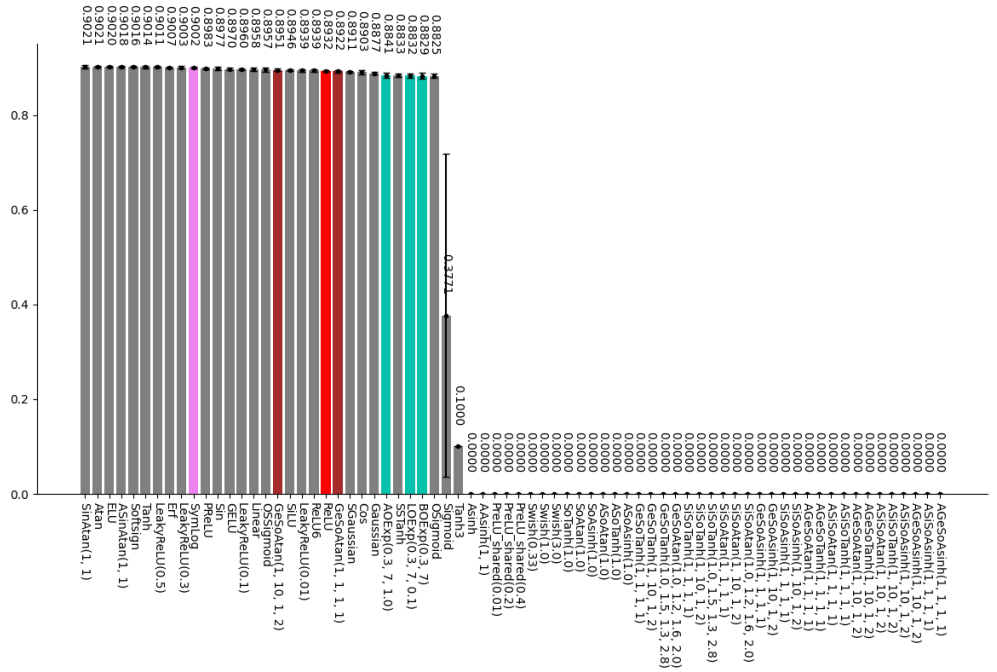


Рисунок 8.11 – Результати замірів точності деяких АФ для класифікації зображень Fashion-MNIST з оптимізатором SGD та темпом навчання 0.03

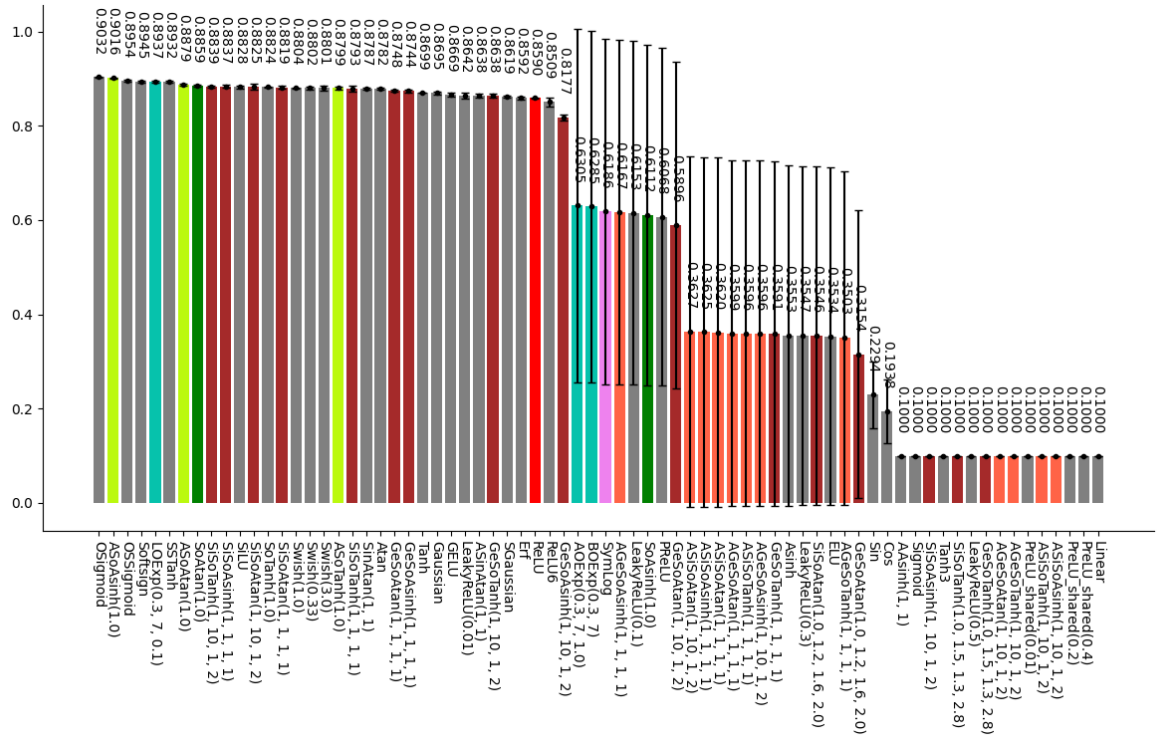


Рисунок 8.12 – Результати замірів точності деяких АФ для класифікації зображень Fashion-MNIST з оптимізатором SGD та темпом навчання 0.3

У більшості цих експериментальних замірів на цей час не було випробувано більшість з нових зважених ААФ, які показали непогані результати на наборі даних CIFAR-10. В тих випадках, коли вони були випробувані ми бачимо схожі закономірності, щодо цих ААФ, що ми бачили із набором даних CIFAR-10 – вони показують добрі результати з оптимізатором Adam, та в переважної більшості погані з оптимізатором SGD. Загалом, щодо використання нових АФ та ААФ з цими та іншими наборами даних, це потребує подальших експериментів і досліджень для кращого розуміння ефективності нових функцій поза межами набору даних CIFAR-10, а також поза межами задач класифікації загалом.

ВИСНОВКИ

В межах цієї роботи була досліджена література щодо поточного стану наукових знань та досягнень у галузі застосування активаційних функцій у згорткових нейронних мережах, та між іншим складений набор критеріїв, що можуть бути корисними при створення нових ефективних активаційних функцій. Згідно з цими критеріями було створено та випробувано декілька видів нових АФ, та деякі модифікації існуючих АФ. Нові АФ включають зсунуті S-подібні функції, АФ, що базуються на експоненційній функції, та зважені зсунені S-подібні функції, включаючи адаптивні варіанти цих АФ.

Із створеними АФ та їх модифікаціями були проведені експерименти з метою визначення точності класифікації зображень при використанні цих АФ в ЗНМ, у порівнянні з використанням низки популярних існуючих АФ. За результатами експериментів було виявлено, що три варіанта нових АФ, що показали найвищу точність класифікації зображень CIFAR-10, а саме $ASiSoAsinh(1,1,1)$, $AGesoAtan(1,10,2)$, та $AGeSoAsinh(1,10,2)$, перевищили точність стандартної АФ ($LeakyReLU(0.1)$), що показала найвищу точність за цією конфігурацію із усіх стандартних АФ, що розглядалися, на 0.82%, 0.80%, та 0.66% відповідно (див. таблицю 8.1). Загалом, нові зважені зсунуті S-подібні ААФ показують кращу точність класифікації зображень CIFAR-10, що за проведеними експериментами перевищила точність класифікації з використанням усіх розглянутих існуючих популярних АФ та ААФ, які увійшли в експерименти (включаючи $ReLU$, $SiLU$, $GeLU$, $Swish$, $PreLU$, та ін.), за умови, що модель тренується із використанням оптимізатора Adam.

Використання оптимізатора SGD робить ці ААФ дуже чутливими до початкових значень їх параметрів, та значно погіршує якість більшості їх варіантів, проте деякі варіанти таких функцій також показали на

експериментальній моделі найкращий результат у порівнянні із іншими існуючим АФ, з якими робились експерименти.

Подальші дослідження, які можуть виявитися корисними, включають більш глибокі дослідження ефективності модифікацій цих АФ/ААФ із SGD та іншими оптимізаторами, дослідження залежності ефективності нових АФ від ініціалізації ваг, структури мережі, випробовування на більш складних мережах та задачах і наборах даних, що є більш наближені до реальних, та ін.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Hornik K., Stinchcombe M., White H. Multilayer feedforward networks are universal approximators. *Neural Networks*. 1989. Vol. 2, no. 5. P. 359–366. URL: https://doi.org/10.1016/0893-6080(89)90020-8 (дата звернення: 18.03.2024).

2. Fukushima K. Neocognitron: A hierarchical neural network capable of visual pattern recognition. *Neural Networks*. 1988. Vol. 1, no. 2. P. 119–130. URL: https://doi.org/10.1016/0893-6080(88)90014-7 (дата звернення: 18.03.2024).

3. Elfwing S., Uchibe E., Doya K. Sigmoid-Weighted Linear Units for Neural Network Function Approximation in Reinforcement Learning. *arXiv.org*. URL: https://arxiv.org/abs/1702.03118v3 (дата звернення: 20.03.2024).

4. Hendrycks D., Gimpel K. Gaussian Error Linear Units (GELUs). 10 p. (Preprint. arXiv:1606.08415 [cs.LG]).

5. Activation Functions: Comparison of trends in Practice and Research for Deep Learning / C. Nwankpa et al. *arXiv.org*. URL: https://arxiv.org/abs/1811.03378 (дата звернення: 21.03.2024).

6. Szandała T. Review and Comparison of Commonly Used Activation Functions for Deep Neural Networks. *Bio-inspired Neurocomputing*. Singapore, 2020. P. 203–224. URL: https://doi.org/10.1007/978-981-15-5495-7_11 (дата звернення: 22.03.2024).

7. Analyzing the Impacts of Activation Functions on the Performance of Convolutional Neural Network Models. *ResearchGate.net*. URL: https://www.researchgate.net/publication/343193175_Analyzing_the_Impacts_of_Activation_Functions_on_the_Performance_of_Convolutional_Neural_Network_Models (дата звернення: 22.03.2024).

8. Dubey S. R., Singh S. K., Chaudhuri B. B. Activation functions in deep learning: A comprehensive survey and benchmark. *Neurocomputing*. 2022.

Vol. 503. P. 92–108. URL: https://doi.org/10.1016/j.neucom.2022.06.111 (дата звернення: 22.03.2024).

9. Rethinking the activation function in lightweight network / L. Yang et al. *Multimedia Tools and Applications*. 2022. URL: https://doi.org/10.1007/s11042-022-13217-z (дата звернення: 25.03.2024).

10. Alkhouly A. A., Mohammed A., Hefny H. A. Improving the Performance of Deep Neural Networks Using Two Proposed Activation Functions. *IEEE Access*. 2021. Vol. 9. P. 82249–82271. URL: https://doi.org/10.1109/access.2021.3085855 (дата звернення: 25.03.2024).

11. Xu B., Huang R., Li M. Revise Saturated Activation Functions. *arXiv.org*. URL: https://arxiv.org/abs/1602.05980 (дата звернення: 26.03.2024).

12. Kim D., Kim W., Kim S. Tanh Works Better With Asymmetry. *OpenReview.net*. URL: https://openreview.net/pdf?id=1WpmOipyYI (дата звернення: 07.04.2024).

13. Pishchik E. Trainable Activations for Image Classification. *Preprints.org - The Multidisciplinary Preprint Platform*. URL: https://www.preprints.org/manuscript/202301.0463/v1 (дата звернення: 26.03.2024).

14. Generating Accurate Pseudo-labels in Semi-Supervised Learning and Avoiding Overconfident Predictions via Hermite Polynomial Activations / V. S. Lokhande et al. *arXiv.org*. URL: https://arxiv.org/abs/1909.05479 (дата звернення: 26.03.2024).

15. Liew S. S., Khalil-Hani M., Bakhteri R. Bounded activation functions for enhanced training stability of deep neural networks on visual pattern recognition problems. *Neurocomputing*. 2016. Vol. 216. P. 718–734. URL: https://doi.org/10.1016/j.neucom.2016.08.037 (дата звернення: 07.06.2024).

16. Deep Learning with S-shaped Rectified Linear Activation Units / X. Jin et al. *arXiv*. URL: https://doi.org/10.48550/arXiv.1512.07030 (date of access: 03.05.2024).

17. Gu S., Timofte R., Van Gool L. Multi-bin Trainable Linear Unit for Fast Image Restoration Networks. *arXiv*. URL: https://doi.org/10.48550/arXiv.1807.11389 (дата звернення: 25.05.2024).

18. LeCunn Y., Cortes C., Burges C. J. The MNIST database of handwritten digits. *yann.lecun.com*. URL: https://yann.lecun.com/exdb/mnist/index.html (дата звернення: 25.05.2024).

19. Zalando Research. A MNIST-like fashion product database. *GitHub/zalandoresearch*. URL: https://github.com/zalandoresearch/fashion-mnist (дата звернення: 27.03.2024).

20. Krizhevsky A., Nair V., Hinton G. CIFAR-10 and CIFAR-100 datasets. *Department of Computer Science, University of Toronto*. URL: https://www.cs.toronto.edu/~kriz/cifar.html (дата звернення: 27.03.2024).

21. Krizhevsky A. Convolutional Deep Belief Networks on CIFAR-10. *Department of Computer Science, University of Toronto*. URL: https://www.cs.utoronto.ca/~kriz/conv-cifar10-aug2010.pdf (дата звернення: 27.03.2024).

22. Raschka S. Is the logistic sigmoid function just a rescaled version of the hyberpolic tangent (tanh) function?. *Machine Learning FAQ*. URL: https://sebastianraschka.com/faq/docs/tanh-sigmoid-relationship.html (дата звернення: 01.06.2024).