УДК 514.18

КОМПОЗИЦІЙНА ГЕОМЕТРІЯ ЯК МАТЕМАТИЧНИЙ МЕТОД НЕЙРОННИХ МЕРЕЖ

Верещага В.М., д-р. техн. н.,
vervik1949@gmail.com, ORCID: 0000-0003-0038-8300,
Адоньєв Є.О., д-р. техн. н.,
evgen.adoniev@gmail.com, ORCID: 0000-0003-1279-4138,
Муртазієв Е.Г., канд. пед. н.,
ernest_gaf@ukr.net, ORCID: 0000-0002-2154-5523,
Верещага І.В.,
ivereshchaha@gmail.com
Мелітопольський державний педагогічний університет імені Богдана Хмельницького (Україна, м. Запоріжжя)
Мелітопольська школа прикладної геометрії імені Володимира Найдиша

Порівнюється традиційне (Ньютона-Лейбніца) диференціювання з композиційним диференціюванням, яке є просто іншим, геометрично досконалим, підходом щодо визначення зміни одних величин в залежності від інших. Наголошується, що обчислювальний результат і за традиційним (алгебраїчним), і за композиційним (геометричним) диференціюванням є одинаковим для одних і тих самих вихідних даних.

Показано, що і мова програмування Python і композиційна геометрія передбачають роздрібнення будь-якої складної задачі на низку дрібних задач меншої складності.

Мовою програмування Python передбачено здійснення операцій з алгебраїчними матрицями, утворення точкових поліномів відбувається за використання композиційних матриць, які призначені для формалізації геометричних алгоритмів. При цьому, вказується, що ресурсовитратність операцій над алгебраїчними матрицями є значно більшою у порівнянні з операціями над композиційними матрицями, застосування яких у Python буде ефективнішим.

Показано, що можливості об'єднувальних операцій у Python збігаються з розробленими методами об'єднань у композиційній геометрії. Існуючі записи точкових поліномів у загальних виглядах дозволяють створювати багатошарові нейронні мережі як класи і як об'єкти у цих класах. Через це методи композиційної геометрії ефективніше передаватимуть прямий розподіл сигналів і зворотній розподіл похибок нейронною мережею. Це, у свою чергу, зменшить ресурсовитратність і пришвидчить процес машинного навчання штучного інтелекту.

Робиться висновок, що функціонування нейронних мереж забезпечуються існуючими математичними методами, які існують взагалі і не є підлаштованими для здійснення операції у них. І навпаки, *можливості аналітичної формалізації композиційної геометрії чи не найкращим чином відповідають операціям у нейронних мережах.*

Методом оптимізації у нейронних мережах є метод градієнтного спуску. Вказується на необхідність і можливості розробки власного композиційного методу градієнтного спуску. Якщо існуючий в математиці метод градієнтного спуску ґрунтується на знаходженні дотичної прямої лінії до поверхні похибок, то композиційний метод градієнтного спуску буде будувати відразу дотичну площину у кожній точці поверхні похибок вагових коефіцієнтів. Це у рази зменшить ресурсовитратність, а відтак у рази пришвидшить і машинне навчання штучного інтелекту. Як результат, зменшиться час на прийняття рішень штучним інтелектом у процесі роботи. Таке стає можливим через те, що традиційні (Ньютона-Лейбніца) похідні здобуваються за використання алгебраїчних методів диференціювання, утворення ж композиційних похідних забезпечується геометричними (композиційними) методами диференціювання.

Ключові слова: нейронні мережі, штучний інтелект, композиційна похідна, композиційна геометрія, композиційний метод градієнтного спуску.

Постановка проблеми. У сучасному діджиталізованому світі нейронні мережі набувають якнайбільшого застосування за різними напрямками. Наразі для операцій з нейронними мережами застосовуються алгебраїчні методи традиційної математики. Відсутність спеціального математичного методу нейронних мереж багато у чому роблять їх ресурсомісткими через застосування малоефективних математичних методів. Наприклад, для сигналів у нейронах (вузлах) вагові коефіцієнти міжшарових зв'язків обираються за алгоритмами випадкових чисел. Композиційні методи дозволяють обґрунтувати вибір цих вагових коефіцієнтів. Крім того, у традиційному машинному навчанні штучного інтелекту, для вдалої і швидкої оптимізації вагових коефіцієнтів і зменшення відсотку похибок, вимагається достатньо високий рівень досвіду і професіоналізму оператора через недосконалий математичний метод, що застосовується. Отже, існує проблема удосконалення математичних методів для нейронних мереж.

Аналіз останніх досліджень і публікацій. Композиційна геометрія є новим науковим напрямом, який створений і розвивається у Мелітопольській школі прикладної геометрії при МДПУ імені Богдана Хмельницького. Шлях її розвитку можна побачити у наступних роботах [1-12]. Мову програмування Python [13] покладено в основу створення нейронних мереж і машинного навчання штучного інтелекту.

Формулювання цілей статті. Дослідити і обґрунтувати можливість застосування композиційної геометрії як математичного методу нейронних мереж з метою розробки, у подальшому, методик машинного навчання штучного інтелекту з її використанням.

Основна частина. У порівнянні з традиційним (Ньютона-Лейбніца) диференціюванням, композиційне диференціювання – це просто інший, геометрично досконалий, підхід щодо визначення зміни одних величин в залежності від інших. Як показали проведені нами дослідження, значення композиційних перших похідних і традиційних перших похідних відрізняються одні від одних у поточних точках на величину обчислювальних помилок. При цьому, композиційне диференціювання, основою якого є елементарні геометричні алгоритми і за рахунок геометричної візуалізації, є більш наглядним, зрозумілим і менш ресурсовитратним у порівнянні з традиційним алгебраїчним диференціюванням. З погляду на це, композиційне диференціювання має перевагу над традиційним (Ньютона-Лейбніца) диференціюванням, яке широко використовується у мові програмування Python.

Крім того, мова програмування Python передбачає роздрібнення будь-якої складної задачі на низку дрібних задач меншої складності. Такий, правильний на наш погляд, підхід завжди дозволяє структурувати логіку складної задачі, в цілому. А у разі схиблення у записах кодів, дає можливість швидко знайти помилку. Точкові поліноми композиційної геометрії саме і передбачають складання алгоритмів, які утворюються дрібними геометричними алгоритмами – модулями, які всередині також можуть бути подрібненими на окремі операції. В термінології Python такі частини називаються чарунками (комірками) (cells).

Утворення точкових поліномів композиційної геометрії використовує набори часто повторюваних виразів за алгоритмами на основі аналогій. У той же час, мова програмування Python спрощує повторно використовувані інструкції і організує у багаторазово задіяні фрагменти коду. У разі, коли ці фрагменти визначені належним чином, то діятимуть автоматично і забезпечуватимуть утворення коротших і, як наслідок, менш ресурсовитратних кодів. Скорочення коду відбувається через те, що замість повторного написання елементу коду, просто здійснюється звернення до виразу (функції), які зберігаються окремо. Отже, принципи методів утворення точкових поліномів збігаються з філософією мови програмування Python.

Крім того, у мові програмування Python передбачено здійснення операцій з матрицями. Утворення точкових поліномів ґрунтується на використанні композиційних матриць різної спрямованості, які утворені на основі елементарних геометричних алгоритмів. Відмінність між композиційними і традиційними матрицями полягає у тому, що традиційні матриці призначені для обслуговування розв’язків лінійних алгебраїчних систем і операцій з ними є набагато ресурсовитратнішими ніж з композиційними матрицями, утворення яких ґрунтується на формалізації геометричних алгоритмів і через це операції над ними є менш ресурсовитратними у порівнянні з традиційними матрицями. З погляду на таке, вважаємо можливим застосування композиційних матриць у мові програмування Python. Це надасть багато переваг щодо скорочення часу на здійснення операцій і, як наслідок, штучний інтелект швидше прийматиме рішення для виконання необхідних дій, що в умовах війни є надзвичайно важливим.

До того ж, у композиційній геометрії безпомилково утворені окремі модулі можна об’єднувати у інші крупніші, попередньо узгодивши для них відповідні вхідні та вихідні значення величин. Також і мова програмування Python для таких операцій містить фундаментальне поняття: «Об’єкти», яких, за допомоги точкових поліномів, можна створювати безліч. Як бачимо, і з цього питання не виникає розбіжностей між методами композиційної геометрії і можливостями мови програмування Python.

Однотипність утворення точкових поліномів і наявність їхніх записів у загальних виглядах якнайкраще накладаються на будь-які структури нейронних мереж різних конфігурацій і розмірів. Записи точкових поліномів у загальному вигляді дозволяють створювати багатошарові нейронні мережі як класи і як об’єкти у цих класах. Сума вхідних сигналів помножених на відповідні вагові коефіцієнти у першому шарі нейронної мережі за виглядом і за сутністю у повній мірі збігається із записами точкових поліномів.

Утворення нейронної мережі відбувається шляхом складання вузлів всередині кожного шара та встановлення зв’язків між відповідними вузлами у різних шарах. Цьому у композиційній геометрії відповідають базисні точки вихідної геометричної композиції та їх параметризація для створення чи то лінії, чи то поверхні, чи то геометричного тіла, які завжди утворюються за структурами, що відповідають структурам нейронних мереж. На наш погляд, методи композиційної геометрії ефективніше передаватимуть прямий розподіл сигналів, а також зворотній розподіл похибок нейронною мережею. Це зменшить ресурсовитратність і пришвидшить процес машинного навчання штучного інтелекту.

Однак, для нас є незрозумілим, чому вагові коефіцієнти обираються за алгоритмами випадкових чисел? Композиційні методи дозволяють обґрунтувати вибір вагових коефіцієнтів виходячи із вихідних даних (сигналів) конкретної нейронної мережі.

Підсумовуючи сказане у цій частині, складається таке враження, що штучний інтелект отримав вибуховий розвиток після того, як розробники відмовились від використання традиційних алгебраїчних методів утворення обчислювальних алгоритмів, а звернулися до геометричних методів аналізу, результатом яких стали нейронні мережі. Або іншими словами, кинули комбінаційні (алгебраїчні) методи утворення алгоритмів і перейшли до композиційних (геометричних) методів їхнього утворення. Традиційні алгебраїчні методи утворення математичних виразів для визначення сигналу на виході із мережі (обчислювальних значень розв’язку), за відомих вхідних сигналів і відповідних вагових коефіцієнтів, є доволі складними. Для розв’язання комбінаційними методами задач штучного інтелекту, кількість можливих комбінацій у яких є неймовірно великою, можуть знадобитися години, дні, тижні, місяці та, навіть, роки.

У той же час, нейронні мережі використовують елементарні математичні поняття, такі як функції, лінійні класифікатори, інтерактивні покращення, матричну алгебру, оптимізацію методом градієнтного спуску і таке інше. При цьому, алгебра матриць у цих мережах використовує операції традиційних алгебраїчних матриць, що обслуговують лінійні системи, метод градієнтного спуску використовує традиційне диференціювання. Тобто, нейронні мережі не мають власного математичного методу. На наш погляд, композиційну геометрію можна довести до стану спеціального математичного методу для нейронних мереж.

Отже, нейронні мережі створені без свого математичного методу і використовують методи традиційної математики. Композиційна геометрія створена нами безвідносно нейронних мереж, однак її методи знаходяться у повній відповідності з математичними вимогами, які покладено у основу функціонування нейронних мереж.

Таким чином, встановлюємо, що: «Композиційна геометрія є математичним методом нейронних мереж», ЯКА може бути покладена в основу машинного навчання штучного інтелекту.

Для цього, розроблена нами теорія композиційних матриць може без обмежень використовуватись у нейронних мережах.

Розглянемо можливість створення композиційного методу градієнтного спуску.

Якщо на поверхні, що відображає значення похибок на виході із нейронної мережі, за використання методу градієнтного спуску, знайти мінімум, то це означатиме, що мінімізовано похибку результату на відповідному етапі машинного навчання. Мінімізована похибка використовується для уточнення вагових коефіцієнтів зв’язків між сигналами відповідних вузлів (нейронів) у сусідніх шарах. Для цього утворюється поверхні похибок на виході із нейронної мережі як різниця між правильним (тобто, існуючим об’єктом, який у подальшому підлягатиме розпізнаванню штучним інтелектом) і результатом, який надають тренування цієї нейронної мережі. Утворення такої поверхні та її диференціювання традиційними методами (Ньютона-Лейбніца) є доволі складною задачею, яку, практично, не можливо здійснити в аналітичній формі.

Застосування методів композиційної геометрії дозволяють, у неперервному вигляді, легко відтворити, за наперед визначеними даними, рельєфні поверхні довільної форми. Для застосування цих композиційних поверхонь у оптимізації похибок вагових коефіцієнтів необхідно розробити композиційний метод градієнтного спуску, який використовуватиме композиційні похідні, а не традиційні (Ньютона-

Лейбніца) похідні. На наш погляд, наразі це є цілком вирішуваним завданням.

Геометричним сенсом похідної для кривої лінії у певній точці є кут нахилу між віссю та дотичною до графіка цієї кривої у цій точці. Геометричним сенсом похідної для поверхні є дотична площина до цієї поверхні у відповідній її точці. При цьому, кут нахилу дотичної площини відносно координатної площини визначається за допомоги лінії найбільшого нахилу цієї площини. На відміну від традиційних (Ньютона-Лейбніца) похідних, які здобуваються за використання алгебраїчних методів диференціювання, утворення композиційних похідних забезпечується геометричними методами диференціювання, які дозволяють для цієї поверхні похибок утворити неперервну композиційну поверхню дотичних площин. Кожна точка якої визначатиме, у відповідній точці на поверхні похибок дотичну площину, а, відповідно, і напрям для градієнта, який збігається з лінією найбільшого нахилу цієї дотичної площини.

Ресурсовитратність утворення композиційного градієнта на усіх кроках спуску, за описаною методикою, є мінімальною у порівнянні з градієнтами, що здійснюються на використанні методів традиційного алгебраїчного диференціювання. Крім того, традиційний метод градієнтного спуску для знаходження дотичного вектору градієнта, застосовує сигмоїду у матричній формі. Основною операцією, при цьому, є поелементне множення алгебраїчних матриць, яке у порівняння з множенням композиційних матриць є ресурсовитратнішим. Крім того, у традиційному машинному навчанні штучного інтелекту для вдалої і швидкої оптимізації вагових коефіцієнтів і зменшення відсотку похибок, багато у яких питаннях вимагається достатньо високий рівень досвіду і професіоналізму оператора. Тобто, під час навчання багато і чому приймаються рішення навмання, не маючи упевненості, вийде чи не вийде, типу: «А давайте спробуємо так». І навпаки у композиційній геометрії, гадаємо, можна буде розробляти методики утворення геометричних алгоритмів цілеспрямованого пошуку оптимізації рішень.

Єдиною проблемою, на наш погляд, у композиційній геометрії є те, що наразі треба розробити методику визначення базисних точок для усіх шарів обраної схеми нейронної мережі.

Висновки. Фрагментарно надано інформацію щодо сполучення вимог нейронних мереж і можливостей композиційної геометрії. Встановлено, що функціональні можливості нейронних мереж у частині їхнього утворення і машинного навчання у повній мірі відповідають методам композиційного геометричного моделювання. Зроблено узагальнення, що композиційна геометрія є математичним методом нейронних мереж. У зв’язку з цим, подальші дослідження методів композиційної геометрії можуть спрямовуватися на розробку методик щодо її застосування у нейронних мережах у частині машинного навчання штучного інтелекту з метою розпізнавання об'єктів.

Література

1. Адоньєв Є.О. Композиційний метод геометричного моделювання багатофакторних систем: дис. ... д-ра техн. наук. К.: КНУБА, 2018, 512 с.
2. Верещага В.М. Композиційне геометричне моделювання: Моногафія. Мелітополь: ФОП Однорог Т.В., 2017, 108с.
3. Верещага В.М., Найдиш А.В., Адоньєв Є.О., Лисенко К.Ю. Основи композиційного геометричного моделювання: навчальний посібник. Мелітополь: ФОП Однорог Т.В., 2019. 255 с.
4. Лисенко К.Ю. Теоретичні основи методів утворення композиційних ліній і поверхонь: дис…к.т.н. Київ. КНУБА, 2022. 267с.
5. Павленко О.М. Порівняльний аналіз композиційної інтерполяції з традиційними методами. *Прикладна геометрія та інженерна графіка*. К., 2022. Вип. 103. С. 162-174.
6. Павленко О.М. Параметричні композиційні матриці. Збірник тез доповідей XVII Міжнародної науково-практичної конференції «Обухівські читання» 30 березня 2023 р. НУБІП. Київ, 2023, с. 91-96.
7. Лисенко К.Ю. Точкові композиційні матриці. Збірник тез доповідей XVII Міжнародної науково-практичної конференції «Обухівські читання» 30 березня 2023 р. НУБІП. Київ, 2023, с. 97-99.
8. Верещага В.М. О поле дифпроекций эмпирической кривой. *Начертательная геометрия и черчение:* межвузовский сборник - Алма-Ата, 1979. С. 63-66.
9. Верещага В.М. Про необхідність розробки методів композиційного диференціювання та композиційного інтегрування. Збірник тез доповідей XVII Міжнародної науково-практичної конференції «*Обухівські читання*» 30 березня 2023 р. НУБІП. Київ, 2023, с. 108-110.
10. Лисенко К.Ю., Верещага В.М. Елементи композиційного диференціювання у точковій формі. *Прикладна геометрія, інженерна графіка*. Випуск 103. КНУБА, 2023 р. 114-122 с.
11. Муртазієв Е.Г., Верещага В.М. Узагальнений графічний аналіз кривих з використанням їхніх похідних. *Прикладна геометрія та інженерна графіка*. К., 2022, Вип. 103. С. 142-150.
12. Муртазієв Е.Г. Алгоритм утворення смуги дифпроєкцій та визначення композиційних похідних у базисних точках. Збірник тез доповідей XVII Міжнародної науково-практичної конференції «*Обухівські читання*» 30 березня 2023 р. НУБІП. Київ, 2023, с. 102-105.
13. Шолле Ф. Глубокое обучение на Python. СПб: Питер, 2018. 400с.: ил. – (Серия «Библиотека программиста»). ISBN 978-5-4461-0770-4.

COMPOSITE GEOMETRY AS A MATHEMATICAL METHOD OF NEURAL NETWORKS

Viktor Vereshchaha, Yevhen Adoniev, Ernest Murtaziiev, Ivan Vereshchaha

Traditional (Newton-Leibnitz) differentiation is compared with compositional differentiation, which is simply another, geometrically perfect, approach to determining the change of some quantities depending on others. It is emphasized that the computational result according to both traditional (algebraic) and compositional (geometric) differentiation is the same for the same initial data.

It is shown that both the Python programming language and compositional geometry involve breaking down any complex problem into a number of small problems of lower complexity.

The Python programming language provides for the implementation of operations with algebraic matrices, the formation of point polynomials occurs using composite matrices, which are intended for the formalization of geometric algorithms. At the same time, it is indicated that the resource consumption of operations on algebraic matrices is much higher compared to operations on composite matrices, the use of which in Python will be more efficient.

It is shown that the possibilities of unifying operations in Python coincide with the developed methods of unification in composite geometry. Existing records of point polynomials in general forms allow creating multilayer neural networks as classes and as objects in these classes. Because of this, the methods of composite geometry will more effectively convey the direct distribution of signals and the inverse distribution of errors by the neural network. This, in turn, will reduce resource consumption and speed up the process of machine learning of artificial intelligence.

It is concluded that the functioning of neural networks is provided by existing mathematical methods that exist in general and are not adapted to perform operations in them. Conversely, the possibilities of analytical formalization of composite geometry do not best correspond to operations in neural networks.

The optimization method in neural networks is the gradient descent method. The necessity and possibilities of developing one's own compositional method of gradient descent are indicated. If the gradient descent method existing in mathematics is based on finding a tangent straight line to the error surface, then the composite gradient descent method will immediately construct a tangent plane at each point of the error surface of the weighting coefficients. This will reduce resource consumption many times, and therefore will speed up machine learning of artificial intelligence many times. As a result, the time for decision-making by artificial intelligence in the work process will decrease. This becomes possible due to the fact that traditional (Newton-Leibnitz) derivatives are obtained using algebraic methods of differentiation, while the formation of composite derivatives is ensured by geometric (composite) methods of

differentiation.

Keywords: neural networks, artificial intelligence, composite derivative, composite geometry, composite gradient descent method.

References

1. Adoniev, Ye. (2018) Composite method of geometric modeling of multifactorial systems: Dr. thesis K.: KNUBA. [in Ukrainian].
2. Vereshchaga, V. (2017) Composite geometric modeling: Monograph. Melitopol: FOP Odnorog T.V. [in Ukrainian].
3. Vereshchaga, V., Naidysh, A., Adoniev, Ye., Lysenko, K. (2019) Fundamentals of composite geometric modeling: a textbook. Melitopol: FOP Odnorog T.V. [in Ukrainian].
4. Lysenko, K. (2022) Theoretical foundations of the methods of formation of compositional lines and surfaces: Ph.D. Kyiv. KNUBA. [in Ukrainian].
5. Pavlenko, O. (2022) Comparative analysis of composite interpolation with traditional methods. Applied geometry and engineering graphics. K. ,103, 162-174 [in Ukrainian].
6. Pavlenko, O. (2023) Parametric composite matrices. Collection of theses of reports of the XVII International Scientific and Practical Conference "Obukhov Readings" March 30, 2023. KNUBA. Kyiv, 91-96 [in Ukrainian].
7. Lysenko, K. (2023) Point composite matrices. Collection oftheses of reports of the XVII International Scientific and Practical Conference "Obukhov Readings" March 30, 2023. KNUBA. Kyiv, 97-99 [in Ukrainian].
8. Vereshchaga, V. (1979) On the field of diffrictions of the empirical curve. Sketchy geometry and drawing (interuniversity collection) - Alma-Ata [in Russian].
9. Vereshchaga, V. (2023) On the need to develop methods of compositional differentiation and compositional integration. Collection of theses of reports of the XVII International Scientific and Practical Conference "Obukhov Readings" March 30, 2023.KNUBA. Kyiv, 108-110 [in Ukrainian].
10. Lysenko, K., Vereshchaga, V. (2023) Elements of compositional differentiation in point form. Applied geometry and engineering graphics. K., 103, 114-122 [in Ukrainian].
11. Murtaziiev, E., Vereshchaga, V. (2023) Generalized graphic analysis of curves using their derivatives. Applied geometry and engineering graphics. K., 103, 142-150 [in Ukrainian].
12. Murtaziiev, E. (2023) Algorithm for the formation of a band of diffrictions and determination of composite derivatives at base points. Collection of theses of reports of the XVII International Scientific and Practical Conference "Obukhov Readings" March 30, 2023.KNUBA. Kyiv, 102-105 [in Ukrainian].
13. Scholle F. (2018) Deep learning in Python. St. Petersburg: Peter, (Series "Programmer's Library"). ISBN 978-5-4461-0770-4 [in Russian].