Mathematical Principles of Artificial Neural Networks and Machine Learning for Bioinformatics

2023-03-03

Konstantinos Krampis*[1], Eric Ross[2], Olorunseun O. Ogunwobi[1], Grace Ma,[3] Raja Mazumder,[4] Claudia Wultsch[1]

[1]Belfer Research Facility, Biological Sciences, Hunter College, City University of New York [2]Fox Chase Cancel Center [3]Temple University [4]George Washington University

*Corresponding Author, *agbiotec@gmail.com*

ABSTRACT

Following the exponential growth of the machine and deep learning field in recent years, artificial neural network models have found significant applications towards bioinformatics data analysis. The various omics datasets are commonly represented in a graph format, such as for example protein or gene interaction networks, molecular structures and cellular signalling pathways. Unlike structured images, video and text that are commonly used for training artificial neural networks, graph data are in the non-euclidean domain and require significantly different algorithmic approaches. The machine learning research community, has developed a range of new algorithms for training artificial neural networks with graph data. These novel approaches and their importance for the bioinformatics field is established herein, through exhibition of the undelying mathematical foundations from group theory, functional analysis and linear algebra. Furthermore, it is argued that the most recent developments in the field such as geometric deep learning on data manifolds, can also significantly accelerate scientific discovery in bioinformatics by enabling new approaches to understand complex datasets. Finally, we conclude this opinion article with the options for implementations through the available software frameworks, as guideline for transitioning from the mathematical principles to practical graph machine learning tools for bioinformatics applications.

INTRODUCTION

Symmetry and invariance is a central concept in physical, mathematical and biological systems, in addition to other areas of human endeavor such as social networks. It has been established since the early 20th century that fundamental laws of nature originate in symmetry, for example through the work of Noether and Wigner [1].In the last decade, technologies such as genomic sequencing have enabled an exponential increase [2] of the data that describe the fundamental elements, structure and function of biological systems. Other endeavors from fields as diverse as physics, and the rise of social media platforms [3], have resulted in datasets of scale not previously available to scientists. This data explosion, has also been fundamental for the ever accelerating advancements in the field of machine learning, deep learning and artificial intelligence, where we now have algorithms that can be trained to make discoveries from the data at a level that matches closely human intuition.

As in any field (including bioinformatics) that develops rapidly whithin the span of a few years,deep learning and artificial intelligence researchers have developed hundreds of successful algorithms, however with a few unifying principles . In a seminal `proto-book` by Bronstein et al. [4], a range of systematization principles for the different artificial neural network architectures from the deep learning field was presented, based on the concepts of symmetry that is formalized within the mathematical field of group theory. The authors also introduced the concept of Geometric Deep Learning, and demonstrated how the group theoretic principles of iso- and auto-morphism, along with invariance and equivariance of functions, can be used in composing and describing various deep learning algorithms.

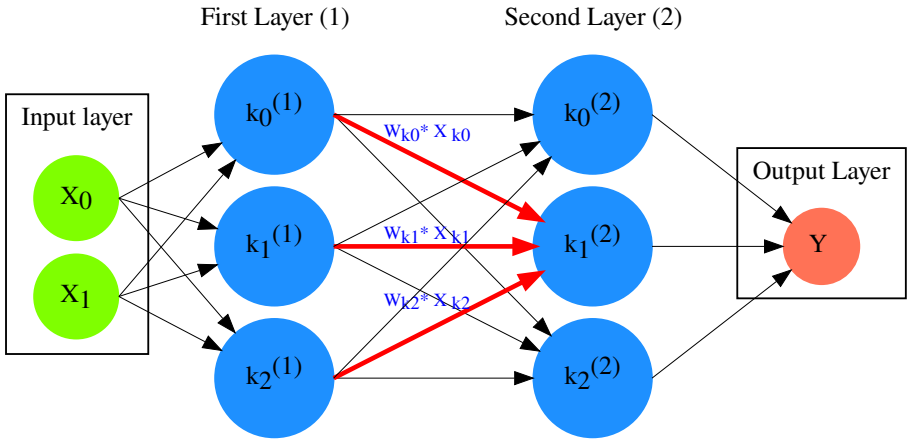
THE STRUCTURE OF ARTIFICIAL INTELLIGENCE AND NEURAL NETWORKS

Before proceeding towards introducing these concepts further, we will describe the structure and function of deep learning and Artificial Neural Networks (ANNs) that are the foundation of artificial intelligence [5],through a simple mathematical description. Assume a dataset consisting of n pairs of $(x_i, y_i)_n$, with the x_i being n data points and y_i their labels. Each x_i data point can be for example be a number, a vector (array of numbers), a matrix (grid of numbers) storing the data for image pixels, or graph data structures composed by nodes and edges representing drugs or other chemical molecules. The data labels can be of various formats, such as binary (two-option) for example $y_i = 1$ "inhibits cancer growth", or $y_i = 0$ "does not inhibit cancer". The labels can also be continuous numbers such as for example $y_i = 0.3$ meaning 30% inhibition, while also each y_i could be a composite label such as $y_i = [0, 1, 0]$ representing respectively drug attributes such as 'no inhibition', 'yes for toxicity', 'not metabolized', with 'not' for 0 and 'yes' 1.

Similar to a brain neural network, the computational abstractions used in

machine learning and artificial intelligence, model neurons as signal aggregators and thresholding units. Specifically, each artificial neuron performs a summation of incoming signals from its connected neighbooring neurons in the network, and "fires" when the aggregated signals reach a certain threshold. This can be described with the summation $\sigma_k = \sum_{1}^{k} w_k * x_k - b$ (**Fig.1**), where the w_k represents the connection weight of neighboring neuron k. The next term x_k multiplied by the weight strength, is the incoming signal from the neigbooring neuron for which the weighted connection is established. As it will be mentioned in a subsquent section, the weight value is the only variable element in the ANNs that is adjusted by the algorithms in order to fit the data, similarly to the biological brain where learning takes place by strengthening connections among neurons. Unlike the biological brain, the ANNs used in practice for data analysis have fixed connections between the neurons and structure of the neural network. The last term b in the summation, represents a threshold that needs to be surpassed as $\sum_{1}^{k} w_k * x_k > b$, in order for the neuron to activate and "fire" a signal.

The simplest structure of an artificial neural network as shown on **Fig.1** is "fully connected", with each neuron k in the network having a number of incoming and outgoing connections corresponding to the number of neurons in previous and next layer in the neural network. For example the first neuron $(k_0)^1$ of Layer 1 on **Fig.1**, has $n_{\in} = 2$ and $n_{out} = 3$ connections, corresponding to the two neurons of the preceeding Input Layer and the three neurons of the subsequent Layer 2. Finally, the output value of the neuron, is determined by applying to the summation value a thresholding or otherwise "logit" function $\varphi(\sigma_k)$ which is non-linear [5]. This final value is the output of the firing neuron, and is transmitted to its connected neurons in the next layer through the outgoing connections (or "axons" in the biological analogy). Multiple layers of such computational models of neurons connected together in layers (**Fig.1**), along with multiple connections per layer each having each own weight w_k, forms an Artificial Neural Network (ANN).



Mathematically, a trained ANN is a function f that predicts the labels y_{pred_i} for the input data x_i, such as for example 'no inhibition', 'yes for toxicity' etc. for data representing drug molecules. The function f is non-linear and is estimated by fitting a training dataset, which correlates labels y_i to data points x_i. Most important, f is a composition of the $\varphi(\sigma_k)$ thresholding functions of the each neuron interconnected in the artificial network graph [5]. This is key aspect of the ANNs, since the non-linearities present in each individual logit neuron function σ_k, are aggregated across the neural network and on the function f, which enables fitting complex multi-dimensional input data with non-linear distributions. This is the key fact that enables ANNs to achieve higher clasification power compared to traditional statistical methods [6]. With hundreds of papers and monographs that have been written on the technical details of training ANNs,and since the focus of the present review manuscript are networks designed with graph and other non-euclidean datasets, we will next attempt to summarize the training in few sentences and refer the reader to the citations for further details.

In summary, the ANN algorithm is using the training data to identify a function f that predicts labels from the data such as $y_{pred_i} = f(x_i)$. As mentioned previously, f is a composition of the $\varphi\left(\sigma_{(k_0)^1}\right)$ functions of the each neuron in the ANN, and as such the training is the estimation of the weights w_k of the neuron connections, while minimizing the error E based on the difference between y_{pred_i} and y_i. This error quantifies the neural network precision by comparing predicted y_{pred_i} and actual labels y_i. The neuron connection weight w_k estimation by the algorithm takes place through fitting the network function f on a large training dataset of $\{x_i, y_i\}_i^n$, pairs of input data and labels, while it evaluates the error E using smaller testing and validation data subsets. The training algorithm works by making an initial estimated guess for initializing the weights, and then performing multiple cycles (called "epochs") of fitting x_i

the training data to the network. At the end of each cycle "backpropagation" is performed [6], which involves gradient descent optimization, in order to fine tune the weights of the individual neurons in $\sigma_k = \sum_{k=1}^{n} w_k * x_k + b$. The gradient descent (REF) searches the possible combinations of weight values, and since it is a heuristic algorithm it minimizes but cannot reach zero error E. At the completion of multiple training cycles the training algorithm identifies a set of weights which best fit the data model, and the ANN settles on the optimal parameters that estimate the $\varphi(\sigma_k)$ function for each interconnected neuron. Consequently, the overall $f(x_i)$ is also estimated,since it is the composition of the individual $\varphi(\sigma_k)$ neuron functions. Once the artificial neural network training has ben completed by finding the most optimal set of weights, it is now ready to be used for label prediction with new, unknown x_i data.

ARTIFICIAL INTELLIGENCE, GROUP THEORY, SYMMETRY AND INVARIANCE

We conclude, by briefly reviewing how the principles of group theory, symmetry and invariance, have been recently utilized as a foundational framework to explain learning algorithms for ANNs [4].

Following the terminology of Bronstein et al., we consider the input x_i from a data domain Ω, which has a specific structure corresponding to the data type used for training the ANN. For example, microscopy images are essentially 2-dimensional numerical grids (matrices) of n x n pixels, with each pixel having a value for light intensity. In this case the data domain is composed of integers ($\mathbb{Z}$) as grid $\Omega : \mathbb{Z}_n \times \mathbb{Z}_n$, which can have all possible combinations of pixel intensities. Similarly, for color images the data domain is $x_i : \Omega \to \mathbb{Z}_n^3 \times \mathbb{Z}_n^3$, with 3 integer grids each representing the green, blue and red layers composing the color image. The ANN data fitting and label prediction function $y_{pred_i} = f(x_i)$ is applied on a "signal" $\mathrm{X}(\Omega)$ from the domain, which is a subset of the domain Ω with the specific images used for training the neural network.

Concluding this review, we will briefly discuss the concepts of symmetry and invariance through the lens of group theory, in order to examine the classifying power of ANNs in relation to statistical variance in the data. In summary, symmetry is the study of space and structure, with examples referring to to geometrical and algebraic constructs in mathematics, matter configurations in physics and molecular biology structures. Invariance of an object under transformation, is the property of changing the position of the object in space, such as rotating a drug molecule or shifting a cancer histology image, while leaving the properties of the object unchanged [4]. In these examples, the drug remains potent following rotation of the molecule, and the tissue is still recognized as cancerous based on the histology image.

When artificial neural networks act as function estimators $f(\mathrm{X}(\Omega) \to \mathrm{Y})$ to

predict output labels (i.e y_i = potent drug / not potent,

[1] E. Noether, "Invariante variationsprobleme, math-phys," *Klasse, pp235-257*, 1918.

[2] K. Katz, O. Shutov, R. Lapoint, M. Kimelman, J. R. Brister, and C. O'Sullivan, "The sequence read archive: a decade more of explosive growth," *Nucleic acids research*, vol. 50, no. D1, pp. D387–D390, 2022.

[3] L. Clissa, "Survey of Big Data sizes in 2021." 2022.

[4] M. M. Bronstein, J. Bruna, T. Cohen, and P. Veličković, "Geometric deep learning: Grids, groups, graphs, geodesics, and gauges," *arXiv preprint arXiv:2104.13478*, 2021.

[5] Y. Li, C. Huang, L. Ding, Z. Li, Y. Pan, and X. Gao, "Deep learning in bioinformatics: Introduction, application, and perspective in the big data era," *Methods*, vol. 166, pp. 4–21, 2019.

[6] B. Tang, Z. Pan, K. Yin, and A. Khateeb, "Recent advances of deep learning in bioinformatics and computational biology," *Frontiers in genetics*, vol. 10, p. 214, 2019.