When Math Meets Life: Unraveling the Secrets of Biology Through Computation

Seyyed Soheil Raeiss Sadati

Abstract

In this paper we explore the place where biology, mathematics and computer science intertwine mainly by focusing on the role of math challenges that biology can't handle by itself. We also discuss how these approaches have led to significant improvements.

Introduction: An Overview of Biology and Its Role in Solving Big Challenges

Biology is a vast science that includes branches such as molecular biology, genetics, cell biology, and ecology. It has led to important achievements in the fields of medicine, public health, agriculture, and the environment, and has deepened our knowledge of life and the natural world. However, many of the complex and big challenges in biology, especially the ones related to complex systems, require the use of mathematical tools and computational modeling to accurately analyze and solve them.

Biological Challenges That Require Mathematics and Computer Science

Stochastic Models in Biology

Stochastic models are needed to capture the randomness of many biological processes. In systems like gene expression and population dynamics randomness plays a big role. For example gene expression can vary because of the probabilistic nature of molecular interactions and you get different outcomes even under the same conditions. Stochastic models incorporate this randomness so biologists can simulate many possible outcomes rather than one expected outcome and get a more realistic view of biological variability. Biology alone often can't capture the complexity of systems driven by randomness. Traditional biological methods are about observing and describing and can fail when dealing with unpredictable processes. Without mathematical tools these analyses might miss important random interactions. And the computational power required to handle these stochastic systems is beyond what biological methods can handle, so we need to bring in biology, mathematics and computational science together to model and understand these complex systems.

Machine Learning and Computational Tools in Biology

As biological data becomes more complex and large-scale, **computer science** plays a central role in analyzing this data. **Machine learning algorithms**, such as **supervised learning**, **unsupervised learning**, and **deep learning**, are widely used in biology to identify patterns and make predictions from biological data.

One of the most important techniques in computational biology is the use of **Convolutional Neural Networks (CNNs)** for image analysis. CNNs are particularly useful in analyzing **biological images**, such as those obtained from **microscopy**. These networks are able to detect important features in the images, such as identifying cells, classifying proteins, or detecting abnormalities associated with diseases like cancer.

The **perceptron**, a basic model of a neural network, has its roots in biology. A perceptron takes multiple inputs, applies weights, sums them up, and passes the result through an activation function:

$$z = \sum_{i=1}^{n} w_i x_i + b$$

$$y = f(z)$$

Where:

- w_i are the weights,
- x_i are the inputs,
- b is the bias term,
- $f(z)$ is the activation function (e.g., **sigmoid**, **ReLU**).

The perceptron learns by adjusting the weights through **gradient descent**:

$$w_i \leftarrow w_i + \eta(y_{\text{actual}} - y_{\text{predicted}})x_i$$

Where η is the learning rate. This iterative learning process allows the perceptron to learn and adapt to data over time, making it a powerful tool in the simulation of **learning processes** in biological systems.

Help in Solving Protein Folding

Proteins are made of chains of amino acids. The problem of protein folding is understanding how proteins fold into specific three-dimensional shapes which determine their function. This process is influenced by the sequence of amino acids and their complicated physical interactions and it is very important for cellular function and biological processes. Solving this problem is essential for advancing biology, medicine, and our understanding of life at the molecular level.

Biology alone can't solve the Protein Folding Problem because the process of folding is extremely complex and involves physical forces that are hard to study just through experiments. Predicting how a protein will fold from its sequence needs advanced math to model the process and computer science to handle the huge amount of data and run simulations. Biology provides the data, but it needs help from other fields to fully understand and predict protein folding. Mathematics can help solving this problem by modeling and predicting the forces and dynamics used in folding. Some examples are :

One of the most significant problems in computational biology has been the protein folding problem. Proteins are essential for cellular function, and their ability to fold into specific 3D shapes is crucial for their function. The folding process is highly complex and determining how a given amino acid sequence folds into a particular shape has been a long-standing challenge in biology.

The AlphaFold system developed by DeepMind uses deep learning to predict the 3D structure of a protein from its amino acid sequence. AlphaFold's model is based on neural networks, and it employs graph-based models to represent the relationships between amino acids in the protein sequence.

The folding process can be modeled mathematically as an optimization problem, where the objective is to minimize the free energy of the protein structure. This is represented by the equation:

$$E_{\text{folded}} = \sum_{i,j} \epsilon_{ij} \left(r_{ij} \right)$$

Where:

- E_{folded} is the free energy of the folded protein,
- ϵ_{ij} represents the interaction energy between amino acids i and j,
- r_{ij} is the distance between residues i and j.

Using **gradient descent**, AlphaFold optimizes the predicted structure by minimizing the free energy, achieving highly accurate predictions for protein structures. This has revolutionized **structural biology** and has broad implications for **drug discovery** and understanding diseases like **Alzheimer's**.

Solving Complex Problems That Brain Can't

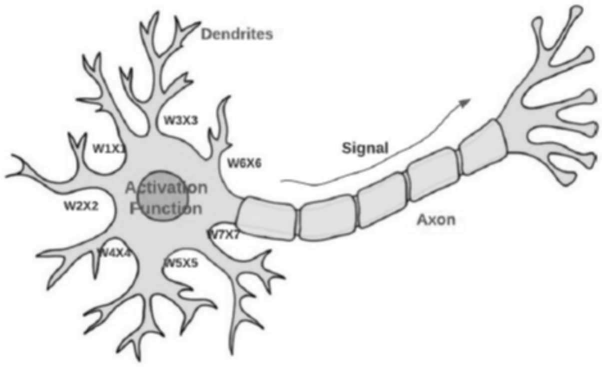
Biological systems, such as the human brain, have amazing abilities in processing information and learning but the still are limits with complex tasks and when they need to process a large amount of data. Neural networks, especially perceptrons, provide a mathematical framework by offering faster and more efficient learning models to solve these limits . Even though biological neurons are highly adaptable, they cannot compete with computational power and speed of artificial neural networks in handling problems. Now we will give a more detailed explanation of this problem in the next chapter.

Solution of a Biological Problem in a Mathematical Way: The Role of Activation Functions in Perceptrons

What Is a Perceptron?

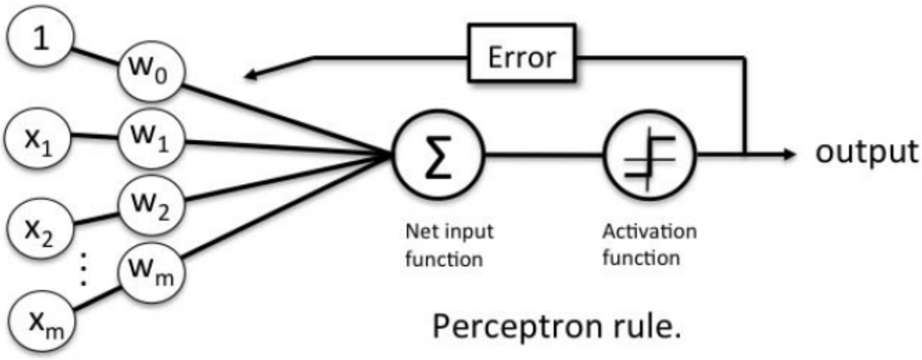
A perceptron is a machine learning algorithm that mimics how a neuron in the brain works. It is also called a single-layer neural network, as the output is decided based on the outcome of just one activation function, which represents a neuron.

To understand how a neuron works, consider the following diagram representing a neuron in the brain. The input signals $x_1, x_2, \ldots$ of different strengths (weights $w_1, w_2, \ldots$) are fed into the neuron cell via dendrites. The net input (weighted sum) is processed by the neuron, and the output signal (observed signal in the axon) is appropriately fired. If the combined signal strength is not adequate based on the decision function within the neuron cell (activation function), the neuron does not fire any output signal.



Neuron in Human Brain

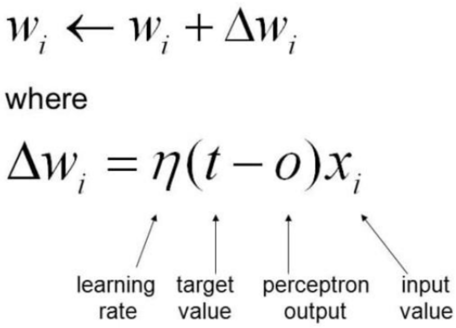
The perceptron when represented as a line diagram would look like the following:



Perceptron - Single layer Neural Network

- **Step 1: Input signals are weighted and combined as a net input:** Weighted sums of input signal reaches the neuron cell through dendrites. The weighted inputs represent the fact that different input signal may have different strength, and thus, weighted sum. This weighted sum can as well be termed as net input to the neuron cell.
- **Step 2: Net input fed into activation function:**Weighted The weighted sum of inputs or net input is fed as input to what is called as activation function.
- **Step 3.1: Activation function outputs binary signal appropriately:**The activation function processes the net input based on unit step (heaviside) function and outputs the binary signal appropriately as either 1 or 0. The activation function for perceptron can said to be unit step function.

- **Step 3.2: Learning input signal weights based on prediction vs actual:**A parallel step is neuron sending the feedback to strengthen the input signal strength (weights) appropriately such that it could create output signal appropriately that matches the actual value. The feedback is based on the outcome of the activation function which is a unit step function. Weights are updated based on the gradient descent learning algorithm. Here is the equation based on which the weights get updated:



Weight update rule of Perceptron learning algorithm

Pay attention to some of the following details in the equation above with respect to the Perceptron learning algorithm:

- Weights get updated by δ_w
- δ_w is derived by taking first order derivative of loss function (gradient) and multiplying the output with negative (gradient descent) of learning rate. The output is what is shown in the above equation - product of learning rate, difference between actual and predicted value (perceptron output) and input value.
- The weights are updated based on each training example such that perceptron can learn to predict closer to actual output for next input signal. This is also called as stochastic gradient descent (SGD). That said, one could also try batch gradient descent to learn the weights of input signals

Example: AND Gate

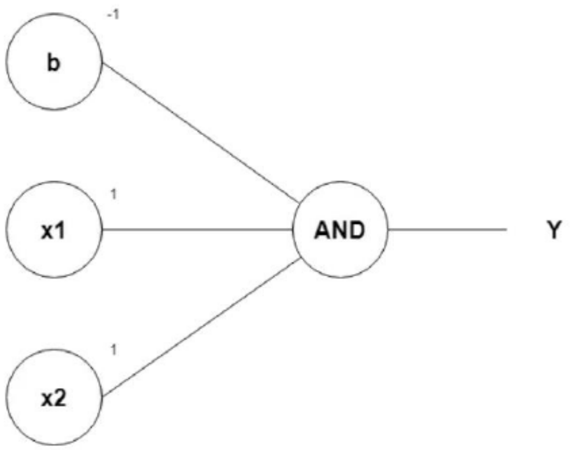
From our knowledge of logic gates, we know that an AND logic table is given by the diagram below:

A	B	Out
0	0	0
0	1	0
1	0	0
1	1	1

AND Gate

The question is, what are the weights and bias for the AND perceptron? First, we need to understand that the output of an AND gate is 1 only if both inputs (in this case, $x1$ and $x2$) are 1. So, following the steps listed above:

- **Row 1:**
  - From $w_1 * x_1 + w_2 * x_2 + b$, initializing w_1, w_2, as 1 and b as –1, we get: $x_1(1) + x_2(1)$–1
  - Passing the first row of the AND logic table ($x_1 = 0$, $x_2 = 0$), we get: 0+0–1 = –1
  - From the Perceptron rule, if $Wx + b \leq 0$, then $y' = 0$. Therefore, this row is correct, and no need for Back-propagation.
- **Row 2:** For $x_1 = 0$, $x_2 = 1$:
$$0 + 1 - 1 = 0$$
  - From the Perceptron rule, if $Wx + b \leq 0$, then $y' = 0$. This row is correct, as the output is 0 for the AND gate.
  - From the Perceptron rule, this works (for both row 1, row 2 and 3).
- **Row 4:** For $x_1 = 1$, $x_2 = 1$:
$$0 + 1 - 1 = 1$$
  Again, from the perceptron rule, this is still valid. Therefore, we can conclude that the model to achieve an AND gate, using the Perceptron algorithm is: $x_1 + x_2$–1



What We Learned?

- The perceptron mimics neurons in the human brain.
- Perceptron is termed as machine learning algorithm as weights of input signals are learned using the algorithm.
- Perceptron algorithm learns the weight using gradient descent algorithm. Both stochastic gradient descent and batch gradient descent could be used for learning the weights of the input signals.

- The activation function of Perceptron is based on the unit step function which outputs 1 if the net input value is greater than or equal to 0, otherwise it is 0.
- The prediction is also based on the unit step function.

Conclusion

Mathematics uses tools such as modeling, data analysis and theoretical proofs to help biologists understand and analyze the complex systems and even solve the problems that they can't solve on their own. This tools help to predict biological problems and analyze complex data. With increasing progressions in computing power of and being able to model complex systems with math, the connection between mathematics and computer science with biology continues to deepen. This helps biologists to make new discoveries in fields like medicine and even neuroscience. The interesting thing is that mathematics and biology help each other grow meaning this is a two sided relationship . So we can conclude that this connection can improve the quality of life.

References

[1] David F. Anderson. *An Introduction to Stochastic Models in Biology*. `https://people.math.wisc.edu/~dfanderson/RecentTalks/2010/CIBM.pdf`. 2010.

[2] Adeline Samson. *Mathematics and Modern Biology: An Overview*. `https://adeline.e-samson.org/wp-content/uploads/2013/10/9783642321566-c1-1.pdf`. 2013.

[3] Emory University Biomathematics Center. *Mathematics Addressing Biological Challenges*. `https://www.biomath.emory.edu`. 2024.

[4] University of California Berkeley Garcia Lab. *Applications of Mathematics in Biology*. `https://garcia.berkeley.edu`. 2024.

[5] University of Nebraska. *Mathematical Modeling in Biology: SIR and SEIR Models*. `https://www.unl.edu/mathmodels`. 2024.