Analyzing Biologically Plausible Alternatives to Backpropagation

Shreya Gupta
Symbolic Systems
Stanford University
shreyagupta@stanford.edu

Abstract

Backpropagation is the foundation of neural networks and deep learning. However, their biological plausibility have been debated for years and the need to design neuroscience-based artificially intelligent models is regaining momentum. In this paper we formulate one such biologically plausible alternative to backpropagation. Based on the principle ”neurons that fire together, wire together”, we use hebbian and anti-hebbian learning rules to update hidden layer weights and delta learning rule to update weights of the output layer. We then contrast the performance of this model with a backpropagation based autoassociator to understand and contrast the learning of the two algorithms.

1 Introduction

Backpropagation is the heart of machine learning, neural networks and the entirety of deep learning field today. It was formulated by Rumelhart et al. (1986), then elaborated and popularized in (McClelland et al., 1987). Backpropagation has been independently rediscovered many times, with the predecessors dating to the 1960s (Kelley, 1960; Goodfellow et al., 2016). Backpropagation is used to update weights in neural networks using partial differentiation of the network loss using gradient descent (Cauchy et al., 1847). However, one of the criticisms of backpropagation has been that it lacks foundation in the biological underpinings of our internal weight updation mechanism (Shrestha et al., 2019). In this paper we propose a biologically plausible alternative to backpropagation, analyze the performance of the method and contrast its performance with the backpropagation based model.

Before elaborating the model, it is worth discussing the need for biological plausibility in designing our AI models. The benefit here is two folds. Designing neuroscientifically founded models helps us understand human visual and information processing mechanisms better. This was reflected in (Yamins et al., 2014; Geirhos et al., 2018; Jagadeesh and Gardner, 2022) where researchers were able to understand that our primary visual cortex is biased towards texture over shape of objects (even though humans recognise images using shapes instead of textures) by training a deep convolutional neural network on imagenet data and fine tuning it to generate fMRI responses. The second gain from this line of study is for deep learning and AI in general. Current AI models have been shown to be susceptible to adversarial attacks while human beings are not as easily fooled (Szegedy et al., 2013; Geirhos et al., 2018). Thus designing models that are rooted in human mechanisms makes the AI models more efficient, accurate, fast and robust.

Our contributions in this report are two folds:

- Proposing a biologically plausible alternative (BPA) model to backpropagation.
- Thorough analysis of the BPA model and the backpropgation based neural network, in isolation and against each other.

In the following sections we elucidate our model architecture (Section 2), the dataset we use for the task (Section 3), the experiments we run (Section 4) and our results from the experiments (Section 5). Finally in Section 6 we give our concluding remarks and enlist future guidelines for the work.

2 Methodology

In this section we start by defining our task for the project. We then describe our models, backpropagation-based autoassociator and the biologically plausible alternative - the BPA model.

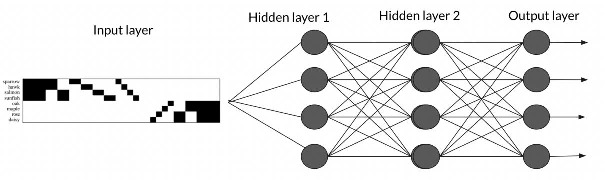

Figure 1: Backpropagation-based autoassociator model architecture.

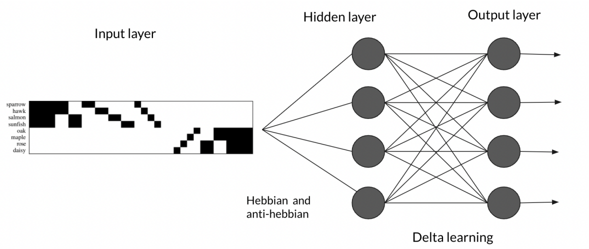

Figure 2: Biologically-plausible model architecture. We analyze the suitability of this model as an alternative to backpropagation. Hidden layer weights are updated using hebbian and anti-hebbian learning rule and output weights are updated using delta learning rule.

2.1 Task

In this report, we compare two models, a backpropagation based model and a biologically plausible alternative to it. We use the dataset introduced by McClelland et al. (2020) and our goal is to regenerate the input.

2.2 Backpropagation

Since our goal is to regenerate our input, we use an autoassociator. An autoassociator is an unsupervised learning algorithm commonly used when there is no target output available. In such cases the inputs are used as targets. An autoassociator is non-trivial if the model has hidden units which are less than the input units in number. When that is the case, a dimensionality reduction algorithm, like Principal Component Analysis, is performed. To avoid running into an information bottleneck problem, we keep the number of hidden units the same as our input sample size. The backpropagation based model architecture is shown in Figure 1. We feed our input samples to two layers of hidden units and obtain our outputs using weight matrices and biases. We then calculate the error and the loss of information from the expected output. We backpropagate this loss by updating the weights of the intermediate layers using gradient descent. We then re-run the entire network for the updated value of weights until our loss is below a pre-defined threshold or a maximum number of epochs are reached, whichever is sooner. This defines the training regime of the model. The model thus obtained has learned the distinguishing parameters of the input sample and satisfied the goal.

2.3 Biological Alternative

For the biologically plausible model, we use hebbian and anti-hebbian learning rules to generate hidden layer representations and delta learning rules to generate output values. Hebbian learning rule is used to capture statistical relationships between items and their features and attempts to connect the underlying mechanisms behind psychological and neurological properties of learning. It is based on the principle that "neurons that wire together, fire together". Weights in hebbian rule for the hidden layer h are represented by dW_h and are updated using equation 1. Here ϵ represents the value of the learning rate parameter, X represents the input data and H represents the hidden layer units.

$$dW_h = \epsilon H \cdot X \tag{1}$$

Anti-hebbian learning rule is used to decorrelate units in the representation layer. It involves a weakening of an excitatory synapse when its corresponding units are correlated. From equation 1 it is clear that the weights can change without bound. Oja (1982) constrained the unbounded growth of W by introducing an anti-Hebbian decay term. The resultant equation of weight updation using the hebbian and the anti-hebbian decay term is shown in equation 2.

$$dW_h = \epsilon H(X - H \cdot W_h) \tag{2}$$

The output layer is computed using the delta learning rule (Widrow and Hoff, 1960). In delta learning rule, we calculate the difference between the expected output, Y, and the predicted output, Y_{pred}, and use this difference (equation 3), also called the error, to update the weight of the output layer, W_o using equation 4. Here ϵ is a hyper parameter, as before, which defines the learning rate.

$$error = Y - Y_{pred} \tag{3}$$

$$dW_o = \epsilon error \cdot H \tag{4}$$

Our overall model pipeline is illustrated in Figure 2. It looks as follows:

We randomly initialise our weight matrices for hidden and output layer. The units in hidden layer, H are computed using the dot product between the hidden unit weight matrices, W_h and the input, X as is shown in equation 5.

$$H = W_h \cdot X \tag{5}$$

We then compute the output values, Y_{pred}, following the same method. Instead of using the hidden weight matrix, we use the output layer weight matrix, W_o and compute Y_{pred} using a dot product between the hidden units, H and W_o as is shown in equation 6.

$$Y_{pred} = W_o \cdot H \tag{6}$$

Post computation of our predicted output, we calculate the error, i.e. the difference between our expected output and our predicted output using equation 3 and update weight matrix for the output following equation 4. We then update the hidden layer weight matrix following equation 2. We repeat the afore-mentioned steps for the next epoch until all epochs are run. In each epoch we use the (hidden and output) weight matrices that were obtained post updation in the previous epoch.

2.4 Comparison

Final and the most important step in our methodology is comparing model performances. Particularly, we want to compare how the frameworks learned in the course of training and how they contrasted against each other. To do this we use singular value decomposition (SVD). In linear algebra, SVD helps in factorization of a real or complex valued matrix. The factorization of the matrix, M is of the form $M = U\Sigma V^T$, where U and V are square orthogonal unitary matrices and Σ is the diagonal matrix which contains the singular values. We adapt the approach and applicability of SVD from (Saxe et al., 2013, 2019).

3 Dataset

The input data consists of different plants and animals and their properties. We consider 8 samples and 34 features for each sample. The dataset is taken from (McClelland et al., 2020). The species included sparrow, hawk, salmon, sunfish, oak, maple, rose and daisy. For the sake of understanding, the features can be interpreted as properties of these samples like (in chronological ordering for input-output correlation matrix for Figure 3) Can move, Has eyes, Has skin, Has bones; Has wings, Can fly; Has fins, Can swim; Is small, Is meek; Is large, Is fierce; then likely Lives in rivers, Lives in the sea; Is pink, Is yellow. Finally, the idiosyncratic features could be the property names.

Each cell of the item-property matrix has values from 0 and 1 depending upon the extent a particular property is true for a particular item. Each row of this matrix consists of a vector X (consisting of the elements x_1, x_2,...,x_n), where X represents the item property matrix that consists of a set of items and their properties. For example, items such as 'robin' and 'oak' may be the rows of this matrix, and properties such 'can fly' or 'has leaves' might be the columns.

4 Experiments

We consider the matrix X consisting of the input samples (like 'sparrow') and their properties (like 'can fly'). Since we are regenerating our input, our expected output, Y, is the same as input, X.

For SVD analysis, we conduct various experiments. In the first experiment, we compare the SVD plots of the final predicted output with the data SVD for both the models. In the next set of experiments, we compare the SVD plot of the predicted output of the autoassociator with the predicted output of the BPA model. Finally, we analyze the strength of each dimension in the SVD over the training epochs, first for individual models and then against each other.

5 Results and Analysis

In this section, we report the experimental results for SVD analysis of the final outputs for the data, backpropagation-based autoassociator and biologically plausible alternative model, BPA. We also report and analyze the change in dimension strength with learning for autoassociator and BPA.

5.1 Final Output SVD

In the first experiment we compare the SVD plot of the data (Figure 3) with SVD of the predicted output for the autoassociator (Figure 4) and the BPA model (Figure 5). As we can see the input-output correlation matrix in our data and the autoassociator is almost the same indicating that the model was able to learn the features of our samples. We further observe that the U matrix learns the same properties for the initial modes, learns the dimensionally-complimentary properties for the middle modes

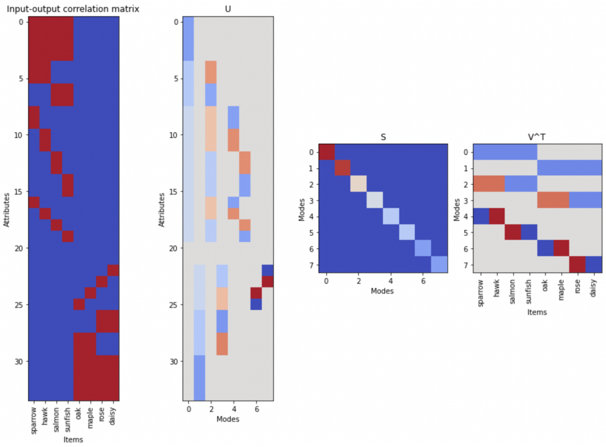

Figure 3: SVD of expected output values of the data

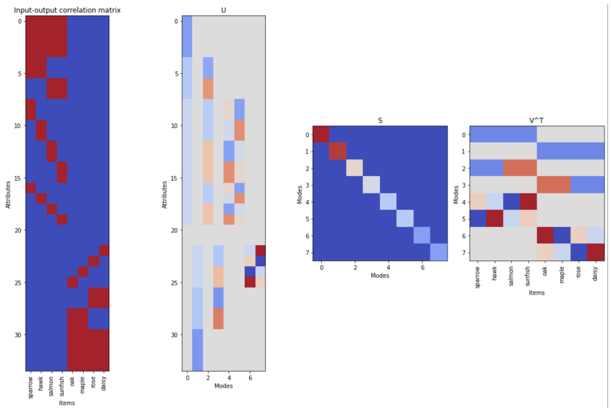

Figure 4: SVD of predicted output values of the backpropagation-based Autoassociator

and learns internally-complimentary values with slight noise for the last dimensions. Dimensionally-complimentary in this context means that the forth and fifth dimension are complimentary to each other between data SVD and autoassociator SVD; Internally-complimentary means that the attribute values are complimentary between the data SVD and autoassociator SVD within the same dimension (or mode in this case). This is also reflected in the V^T matrix where the first few modes are learned the same way for the model as is for the data, the middle modes are dimensionally-complimentary to each other and the last few modes are the internally-complimentary with slight noise. When the three matrices are multiplied, they however generate the same resultant matrix since complimentary of a complimentary is the same value as the original one.

We now contrast SVD of data (Figure 3) with SVD of the BPA model (Figure 5). As we observe in the input-output correlation matrix, for

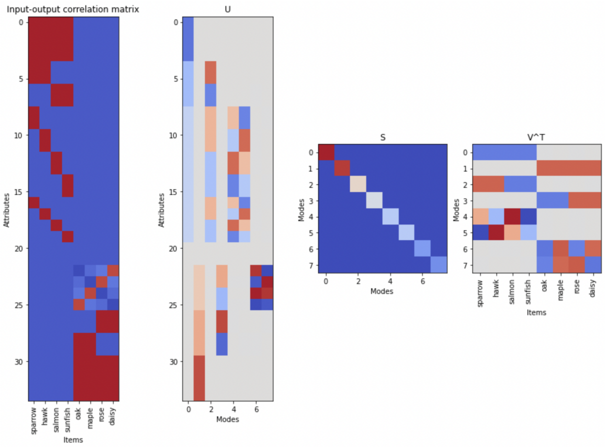

Figure 5: SVD of predicted output values of the biologically plausible alternative

the later items, there is some noise for the learned attributes. This is also reflected in the SVD matrices. Firstly we observe that the second mode is learned complimentarily, i.e. if the data learned positive values for the modes against the attributes, the BPA learned negative values of similar magnitude. Secondly, as for the autoassociator, the BPA also kearned dimensionally-complimentary values for the middle modes but with more noise than the autoassociator. Finally, for the final modes we observe that the BPA model understands the spectrum of attribute range but is not able to differentiate the attributes as distinctly as the original data inherits or as the autoassociator was able to. All these trends are also observed in the V^T matrix. Analyzing autoassociator (Figure 4) and BPA (Figure 5), in addition to the differences noted above, we also observe that the final modes (mode 7 and 8) in the BPA model for oak, maple, rose and daisy are not as distinctly differentiable.

5.2 Change in Dimension Strength With Learning

In the second set of experiments, we analyze the change in singular dimension strength as training progresses. We observe this graph for the autoassociator and BPA individually, and then contrast them against each other. In both the graphs we notice that two dimensions are distinguished early in the learning curve. These dimensions are most likely to correspond to plant and animal distinction. We then observe the next two dimensions are distinguishable shortly. These distinctions are most likely to correspond to tree and flower and bird and fish. Finally, we observe the last few dimensions

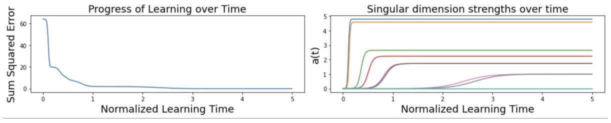

Figure 6: Single dimensional strength with training time in backpropagation-based autoassociator

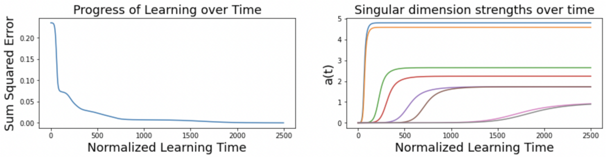

Figure 7: Single dimensional strength with training time in biologically plausible alternative (BPA) model

overlap each other in pairs, more for the autoassociator than for the BPA. These dimensions are more likely to represent more intricate class distinguishing properties like colour is yellow or can fly.

One significant difference between the learning over time is the learning of the final dimensions. In autoassociator the final dimension segregation occurs sooner in the learning curve. However, in BPA the last dimensions are distinguishable towards the end of learning. This is in sync with the final SVD plot we obtained for BPA (Figure 5). Juxtapositioning both the figures together, we can deduce that these dimensions are most likely to correspond to unique identifying properties of oak, maple, rose and daisy.

6 Conclusion and Future Work

In this paper we propose a biologically plausible alternative to backpropagation using hebbian and anti-hebbian learning rule to obtain the hidden layer representations and delta learning rule to obtain output values. We refer to this model as the BPA model and contrast its learning with a backpropagation based autoassociator. For this task we choose a regeneration problem where the objective is to accurately reproduce the input samples, thereby learning the inherent properties in the input data. The crux of our paper lies in the analysis. We analyze the performance of each of these models, independently, and contrast their performances against each other. We do so using Singular Value Decomposition (SVD) analysis. We first contrast the SVD of the final outputs of the two models with the data and then with each other after the learning is complete. We also contrast the strength of each dimension during learning for both the models, first in isolation and then against each other. Through the analysis, we can deduce that while BPA is able to distinguish the sample properties with a considerable strength, it falls short with respect to backpropagation based autoassociator in making cleaner distinctions for some samples.

Future work include performing further analysis on the two models. One of the ways to do it is by plotting dendrograms. Another method is by analysing SVD plots for hidden layers for both the models. Even though we performed hyper parameter tuning for both the models, there is a possibility to tune it further to achieve more optimal performances that are more qualitatively contrastable. Finally, we can also try other alternatives to biologically plauisble methods and can experiment them for tasks other than regeneration.

7 Acknowledgements

The author would like to thank Prof. Jay McClelland for his consistent support and guidance for the entirety of the project. The author is also thankful to Prof. Gaurav Suri for his contribution in designing the BPA model.

References

Augustin Cauchy et al. 1847. Méthode générale pour la résolution des systemes d'équations simultanées. *Comp. Rend. Sci. Paris*, 25(1847):536–538.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. 2018. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep learning*. MIT press.

Akshay V. Jagadeesh and Justin L. Gardner. 2022. Texture-like representation of objects in human visual cortex. *bioRxiv*.

James E Kelley, Jr. 1960. The cutting-plane method for solving convex programs. *Journal of the society for Industrial and Applied Mathematics*, 8(4):703–712.

James L McClelland, Bruce L McNaughton, and Andrew K Lampinen. 2020. Integration of new information in memory: new insights from a complementary learning systems perspective. *Philosophical Transactions of the Royal Society B*, 375(1799):20190637.

James L McClelland, David E Rumelhart, PDP Research Group, et al. 1987. *Parallel Distributed Processing, Volume 2: Explorations in the Microstructure of Cognition: Psychological and Biological Models*, volume 2. MIT press.

Erkki Oja. 1982. Simplified neuron model as a principal component analyzer. *Journal of mathematical biology*, 15(3):267–273.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. 1986. Learning representations by back-propagating errors. *nature*, 323(6088):533–536.

Andrew M Saxe, Yamini Bansal, Joel Dapello, Madhu Advani, Artemy Kolchinsky, Brendan D Tracey, and David D Cox. 2019. On the information bottleneck theory of deep learning. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124020.

Andrew M Saxe, James L McClelland, and Surya Ganguli. 2013. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*.

Amar Shrestha, Haowen Fang, Qing Wu, and Qinru Qiu. 2019. Approximating back-propagation for a biologically plausible local learning rule in spiking neural networks. In *Proceedings of the International Conference on Neuromorphic Systems*, ICONS '19, New York, NY, USA. Association for Computing Machinery.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Bernard Widrow and Marcian E Hoff. 1960. Adaptive switching circuits. Technical report, Stanford Univ Ca Stanford Electronics Labs.

Daniel LK Yamins, Ha Hong, Charles F Cadieu, Ethan A Solomon, Darren Seibert, and James J DiCarlo. 2014. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the national academy of sciences*, 111(23):8619–8624.