A short academic research on the relationship between predictive coding and backpropagation

Nikalal T Helessage: SN 7613234; ntkkh958@uowmail.edu.au
Quang Vinh Duong: SN 7918999; qvd976@uowmail.edu.au
Rohini Sutari: SN 7925396; rs490@uowmail.edu.au
Hoang Anh Nguyen: SN 8067867; han990@uowmail.edu.au
Vaibhav Raja: SN 7708865; vr103@uowmail.edu.au
Zawata Afnan Ahmed: SN 7820367; zaa787@uowmail.edu.au

CSCI433/933 Assignment
May 29, 2023

Abstract

Backpropagation (BP) has played a crucial role in the recent achievements of neural networks. However, the neurological learning process following this algorithm's principle is unrealistic. Therefore, there are studies on alternatives that could accurately represent how the human brain accumulates knowledge. This paper investigates an existing candidate for this movement: predictive coding (PC). Unlike backpropagation, the algorithm tries to make predictions to understand the data it receives, which more closely resembles the actual human learning activity. We examine this technique in different scenarios and settings and compare it with backpropagation to gain more insights about its basis and performance. Our experimental results with the MNIST dataset reveal that predictive coding can mostly achieve comparable performance with backpropagation.

1 Introduction

Backpropagation is a popular training method in neural networks that propagates information backwards through the network from its outmost layer towards the inner ones to update parameters and minimize errors. Nonetheless, the neurological implementation of backpropagation is frequently considered biologically implausible due to the nonlocal nature of parameter changes, as stated by Lillicrap, Santoro, Marris, Akerman, and Hinton (2020). On the other hand, predictive coding, a theoretical framework proposing the brain's reliance on internal models or predictions to understand sensory inputs, suggests that the brain propagates the differences between predictions and actual sensory inputs throughout the network, as described by Keller and Mrsic-Flogel (2018). Consequently, it updates the internal models to minimize those errors.

Research on backpropagation and predictive coding has found similar parameter updates under specific conditions. This result suggests that predictive coding may approximate or be a biologically plausible implementation of backpropagation. It provides insights into the feasibility of learning algorithms and their correlation with brain functioning in artificial and biological neural networks.

2 Theory and related work

Our work was inspired by Rosenbaum (2022), which reviewed and extended earlier research on the connection between backpropagation and predictive coding while also discussing some findings regarding predictive coding being a biological learning model. Related to this subject, Spratling (2017) reviewed and compared several distinct algorithms described as predictive coding. On the other hand, Whittington and Bogacz (2017) demonstrated that using predictive coding to perform the training of a feedforward network for supervised learning tasks can result in the updates of parameter similar to those calculated by backpropagation. Additionally, the results from Salvatori, Song, Lukasiewicz, Bogacz, and Xu (2021) proved predictive coding's applicability to more complex architectures, namely convolutional and recurrent neural networks. The work of Millidge, Tschantz, and Buckley (2022) with predictive coding utilizing Hebbian plasticity in similar architectures also coordinates with these outcomes.

Considering a supervised learning problem with an input set of x, a corresponding labeled set of output y. and a neural network f with parameters $\theta = \{\theta_l\}_{l=1}^{L}$ where L is its depth. Consequently, our objective is to reduce the cost function $\mathcal{L}(\hat{y}, y)$ with $\hat{y} = f(x; \theta)$.

Regarding the backpropagation algorithm, utilizing the gradient descent method with learning rate of α, the parameter updates can be described as Equation 1 follows:

$$\theta_l = \theta_l + \alpha d\theta_l \tag{1}$$

where $d\theta_l = -\frac{\partial \mathcal{L}(\hat{y},y)}{\partial \theta_l}$.

On the other hand, considering the predictive coding algorithm with $\mathbf{v}_l$ being the predicted activation set of the neural network f at layer l-th, which based on the known input $x = \mathbf{v}_0$, the predictive error ϵ_L could be formularized as Equation 2:

$$\epsilon_L = f_L(\mathbf{v}_{L-1}; \theta_L) - y \tag{2}$$

With η being the step size, the algorithm aims to minimize the error by updating $\mathbf{v}_l$. In another phrase, the model's parameters can be updated as following in Equation 3:

$$\theta_l = \theta_l + \eta_\theta d\theta_l \tag{3}$$

where $d\theta_l = -\epsilon_l \frac{\partial f_l(\mathbf{v}_{l-1}, \theta_l)}{\partial \theta_l}$.

The PC algorithm is called "fixed prediction assumption" if the value of $\mathbf{v}_l$ is fixed to be the result $\hat{\mathbf{v}}_l$ of the original forward propagation starting from $\hat{\mathbf{v}}_0 = x$. As a result, the gradient is modified as shown in Equation 4 below:

$$d\theta_l = -\epsilon_l^* \frac{\partial f_l(\hat{\mathbf{v}}_{l-1}, \theta_l)}{\partial \theta_l} = -\frac{\partial \mathcal{L}(\hat{y}, y)}{\partial \hat{\mathbf{v}}_l} \frac{\partial \hat{\mathbf{v}}_l}{\partial \theta_l} = -\frac{\partial \mathcal{L}(\hat{y}, y)}{\partial \theta_l} \tag{4}$$

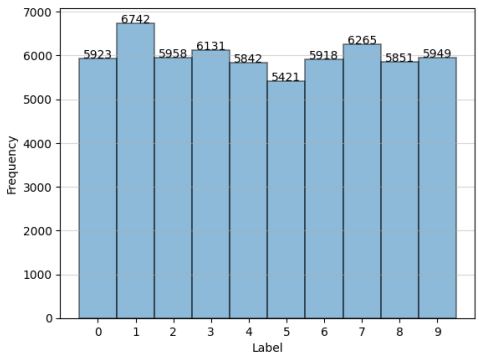
3 Dataset and Experiments

3.1 Dataset

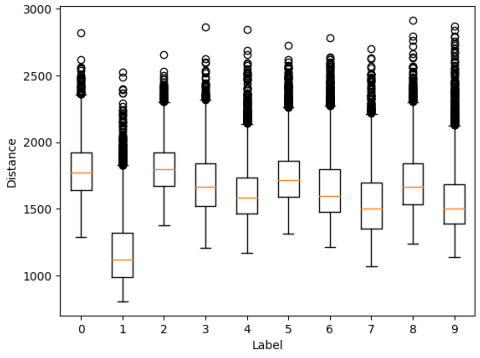
The MNIST dataset (Deng (2012)) is a well-known and widely used benchmark dataset in machine learning. It consists of a large collection of handwritten digits, ranging from 0 to 9, that have been carefully labeled. The dataset was created by remixing samples from the original datasets compiled by the National Institute of Standards and Technology (NIST). Each image in the MNIST dataset is a grayscale image with a resolution of 28x28 pixels. The dataset is divided into two main subsets: the training and testing sets contain 60,000 and 10,000 samples, respectively. The MNIST dataset has been instrumental in advancing and assessing numerous image classification and pattern recognition algorithms.

In this research, we observed that the testing dataset exhibited an uneven distribution, as illustrated in Figure 1a. Additionally, when the digits deviate significantly from the norm, it poses a considerable challenge for machine learning algorithms. Exploring atypical cases is beneficial as it provides insights into the method's limitations and assists in selecting suitable approaches and designing effective features. We calculated the Euclidean distance (square root of the sum of squares) between each image and its corresponding label's centroid to conduct the dataset analysis further. Figure 1b presents the visualization of this analysis using box plots.

The analysis reveals that 1s have particularly low distances to their centroid, indicating little variation in how people typically draw this digit. On the other hand, 0s and 2s exhibit higher variability based on this measure. However, it is important to note that every digit category has several instances with unusually large distances from their centroid, indicating some outliers. To better understand these cases, we can visually examine the six-digit instances that deviate the most from their central digit, as illustrated in Figure 2.



(a) Mnist dataset label distribution



(b) Mnist dataset Euclidean distance to label centroid

Figure 1: Mnist training dataset label distribution and Box plot of Euclidean distance to label centroid

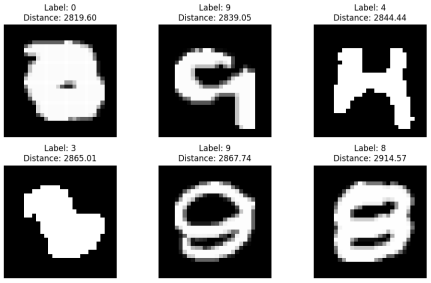


Figure 2: Six digits which are having the largest distance to the centeroid

3.2 Experiment

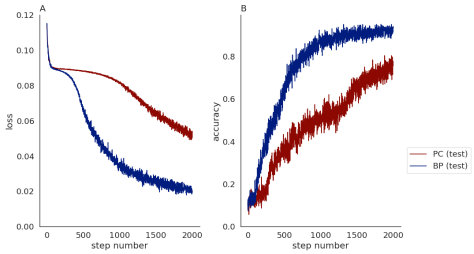
In this report, we conduct three more experiments besides replicating the test cases and results from Rosenbaum (2022) utilizing MNIST mentioned above dataset. There are two objectives in each of these test cases. Firstly, we compared the performance of the proposed algorithms to a backpropagation model with identical settings. Secondly, we examined the parameter updates of the algorithms concerning the backpropagation algorithm's exact gradient. Specifically, the predictive coding model will be trained using four different η (0.1, 0.5, 1.0, and 1.5). Although Rosenbaum (2022) had tested the model with η in the range from 0 to 1, we proposed another testing value of η outside that range (1.5).

Our solution in the first test scenario is an exact interpretation of the PC method. The model used in this case is a deep neural network with five layers. The first layer performs convolution and pooling, followed by another convolution and flattening in the second layer. The last three layers are linear, reducing features to ten in the final output. The Rectified Linear Unit (ReLU) is the activation function for all layers. The Mean Square Error (MSE) loss function calculates dissimilarities between predicted and actual outcomes. Additionally, we replaced the Adam optimizer with Stochastic Gradient Descent (SGD) to analyze its impact on parameter updates. We apply a modified predictive coding algorithm in the second experiment that regulates predictions through forward propagation. The model structure remains remains unchanged from the initial experiment, utilizing the SGD optimizer for parameter updates. However, the loss function used here is cross-entropy instead of mean square error. Moreover, we changed the activation function ReLU to Exponential Linear Unit (ELU) to investigate more but still kept all the original settings in the second experiment. To explore the convergence of gradients in the modified prediction assumption scheme, a third experiment is conducted with a 4-layer convolutional neural network. This network includes different convolutions and two linear layers. The convergence rate is also analyzed by adjusting the batch size and learning rate.

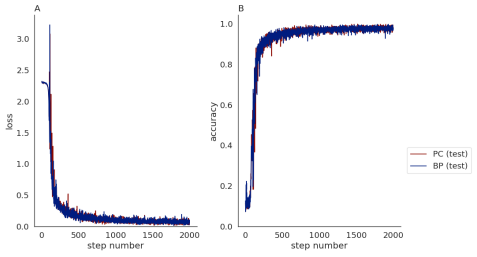
4 Results

In experiment 1, we found that the accuracy obtained through predictive coding was lower than training with backpropagation on the MNIST dataset, as shown in figure 3a. Additionally, the predictive coding loss slowly decreases compared to the loss of the backpropagation method. This indicates that backpropagation was more effective in this particular scenario compared to the strict interpretation of predictive coding. Furthermore, figure 4 demonstrates the transition of the updates of parameter from PC to actual gradients. The results indicated that for smaller η values, the updates of parameter converged near the gradients after sufficient iterations. However, for larger η values, the updates of parameter diverged from the gradients, suggesting instability in the training process.

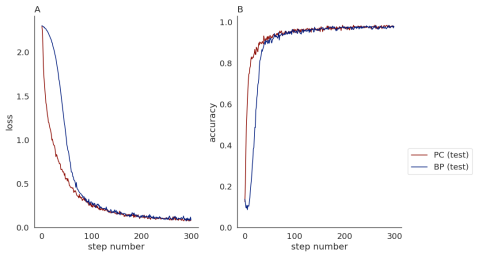
In experiment 2, we aimed to train the MNIST dataset using PC modified by the fixed prediction assumption. We used η of 1 and compared the results with training using backpropagation. Figure 3b shows



(a) Comparison of loss and accuracy of BP and PC in the experiment 1.



(b) Comparison of loss and accuracy of BP and PC modified by the fixed prediction assumption in the experiment 2.



(c) Comparison of loss and accuracy of BP and PC modified by the fixed prediction assumption in the experiment 3.

Figure 3: Loss and accuracy obtained from 3 experiments

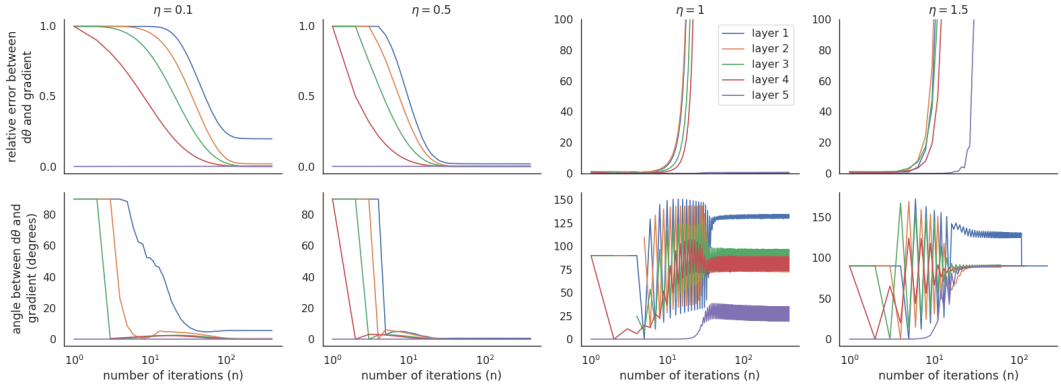


Figure 4: Comparison of parameter updates from PC to exact gradients in the experiment 1.

that the accuracy obtained through the modified predictive coding approach was approximately comparable to training with backpropagation. In this experiment, we also examined the angle and the relative error between the updates of parameter and the exact gradients for different values of η (0.1, 0.5, 1, 1.5). The results revealed that the updates of parameters rapidly reached convergence with the actual gradients when η had larger values, as shown in figure 5. After that, we change the activation function to ELU and the modified predictive coding and backpropagation accuracy slightly increases.

In experiment 3, the modified predictive coding method achieved an accuracy of 0.977, indicating a high level of performance on the MNIST dataset that is approximate to the accuracy of backpropagation technique, which achieved an accuracy of 0.980. All the accuracy values are obtained in the table 1.

Table 1: Accuracy of each experiment

	Experiment 1	Experiment 2	Experiment 2 with ELU	Experiment 3
PC	0.756			
Modified PC		0.983	0.986	0.977
Backpropagation	0.936	0.976	0.980	0.980

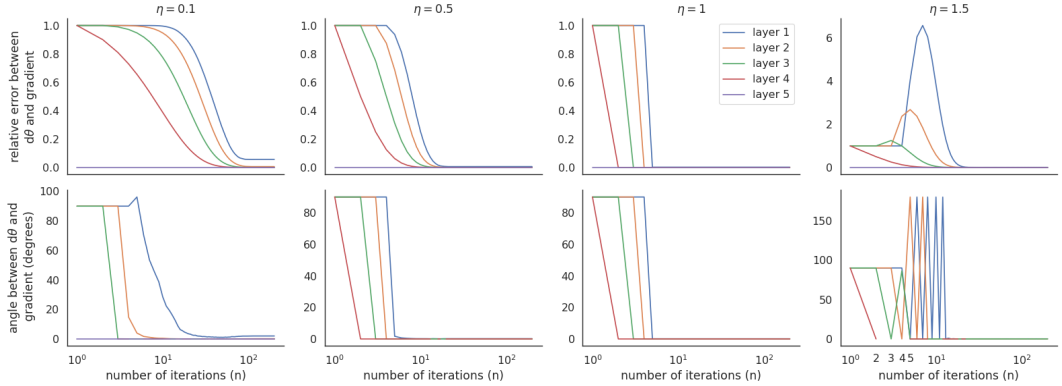


Figure 5: Comparison of parameter updates from PC modified by the fixed prediction assumption to exact gradients in the experiment 2.

5 Discussion

In experiment 1, the result shows that the training process of PC was less efficient in comparison to that of backpropagation. The slower rate of decrease in loss during the training process of predictive coding compared to backpropagation indicates that predictive coding may require more iterations or adjustments to reach convergence. This suggests that predictive coding might be more computationally expensive or time-consuming, which could limit its practical applicability. It is an important factor to consider when finding alternative methods to backpropagation - not only the final performance but also the efficiency and practical feasibility. Moreover, the divergence between our results and the original paper's findings might be due to the use of different optimizers. Therefore, the choice of optimizer can significantly influence the effectiveness of predictive coding or any alternative method.

In experiments 2 and 3, an examination of the modified prediction assumption with $\eta = 1$ revealed its algorithmic equivalence to direct backpropagation implementation. This provides promising evidence for the viability of predictive coding as an alternative to backpropagation. This result is consistent with a more general objective of discovering training algorithms that can compete with or enhance backpropagation. The quick convergence of parameter updates to true gradients with larger η values implies that a carefully chosen step size (η) can enhance the training dynamics of the modified predictive coding method. Experiment 2 involved a small test with ELU, but the results showed minimal change. According to Clevert, Unterthiner, and Hochreiter (2015) and experiment 2 with ELU, it has been observed to enhance the learning process in the modified predictive coding method and result in higher classification accuracies compared to RELU.

Overall, while predictive coding and its modified forms show promise, their performance can be sensitive to various factors, such as hyperparameters, activation functions or optimization choices. The search for alternative training algorithms should consider not only accuracy but also efficiency, convergence properties, and generalizability to diverse datasets and tasks.

References

Clevert, D.-A., Unterthiner, T., & Hochreiter, S. (2015). Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*.

Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, *29*(6), 141–142.

Keller, G. B., & Mrsic-Flogel, T. D. (2018). Predictive processing: A canonical cortical computation. *Neuron (Cambridge, Mass.)*, *100*(2), 424-435.

Lillicrap, T. P., Santoro, A., Marris, L., Akerman, C. J., & Hinton, G. (2020). Backpropagation and the brain. *Nature Reviews Neuroscience*, *21*(6), 335–346.

Millidge, B., Tschantz, A., & Buckley, C. L. (2022). Predictive coding approximates backprop along arbitrary computation graphs. *Neural computation*, *34*(6), 1329-1368.

Rosenbaum, R. (2022). On the relationship between predictive coding and backpropagation. *PloS One*, *17*(3).

Salvatori, T., Song, Y., Lukasiewicz, T., Bogacz, R., & Xu, Z. (2021). Predictive coding can do exact backpropagation on convolutional and recurrent neural networks.

Spratling, M. (2017). A review of predictive coding algorithms. *Brain and Cognition*, *112*, 92-97. Retrieved from `https://www.sciencedirect.com/science/article/pii/S027826261530035X` (Perspectives on Human Probabilistic Inferences and the 'Bayesian Brain') doi: https://doi.org/10.1016/j.bandc.2015.11.003

Whittington, J. C. R., & Bogacz, R. (2017). An approximation of the error backpropagation algorithm in a predictive coding network with local hebbian synaptic plasticity. *Neural computation*, *29*(5), 1229-1262.