PLOS Complex Systems

RESEARCH ARTICLE

Predictive coding algorithms induce brain-like responses in artificial neural networks

Dirk Gütlin[1*], **Ryszard Auksztulewicz**[1,2]

1 Centre for Cognitive Neuroscience Berlin, Freie Universität Berlin, Germany, **2** Faculty of Psychology and Neuroscience, Maastricht University, Netherlands

* dirk.guetlin@fu-berlin.de

OPEN ACCESS

Citation: Gütlin D, Auksztulewicz R (2025) Predictive coding algorithms induce brain-like responses in artificial neural networks. PLOS Complex Syst 2(11): e0000076. https://doi.org/10.1371/journal.pcsy.0000076

Editor: Juan Gonzalo Barajas-Ramirez, IPICYT: Instituto Potosino de Investigacion Cientifica y Tecnologica AC, MEXICO

Received: January 24, 2025

Accepted: September 22, 2025

Published: November 5, 2025

Peer Review History: PLOS recognizes the benefits of transparency in the peer review process; therefore, we enable the publication of all of the content of peer review and author responses alongside final, published articles. The editorial history of this article is available here: https://doi.org/10.1371/journal.pcsy.0000076

Abstract

Predictive Coding (PC) is a neuroscientific theory that has inspired a variety of training algorithms for biologically inspired deep neural networks (DNN). However, many of these models have only been assessed in terms of their learning performance, without evaluating whether they accurately reflect the underlying mechanisms of neural learning in the brain. This study explores whether predictive coding inspired Deep Neural Networks can serve as biologically plausible neural network models of the brain. We compared two PC-inspired training objectives, a predictive and a contrastive approach, to a supervised baseline in a simple Recurrent Neural Network (RNN) architecture. We evaluated the models on key signatures of PC, including mismatch responses, formation of priors, and learning of semantic information. Our results show that the PC-inspired models, especially a locally trained predictive model, exhibited these PC-like behaviors better than a Supervised or an Untrained RNN. Further, we found that activity regularization evokes mismatch response-like effects across all models, suggesting it may serve as a proxy for the energy-saving principles of PC. Finally, we find that Gain Control (an important mechanism in the PC framework) can be implemented using weight regularization. Overall, our findings indicate that PC-inspired models are able to capture important computational principles of predictive processing in the brain, and can serve as a promising foundation for building biologically plausible artificial neural networks. This work contributes to our understanding of the relationship between artificial and biological neural networks such as the brain, and highlights the potential of PC-inspired algorithms for advancing brain modelling as well as brain-inspired machine learning.

Author summary

Our brains are thought to learn by continuously predicting incoming information and updating these predictions when they are violated. This framework, known as predictive coding, has inspired a range of computational models, yet

Data availability statement: All code to replicate this study are published under github.com/DiGyt/predictive_coding_algorithms. All data and pretrained models to directly replicate the results are published under https://doi.org/10.5281/zenodo.17228391. Hereby, the authors confirm that the mentioned repositories contain all the data needed to replicate the study's findings.

Funding: This study was funded by the German Research Foundation / Deutsche Forschungsgemeinschaft (AU 423/2-1 to RA). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

it remains unclear whether such models capture the mechanisms that underlie neural learning. In this study, we tested whether predictive coding–inspired learning rules in recurrent neural networks can reproduce key features associated with predictive processing in the brain. We evaluated a predictive and a contrastive predictive coding algorithm on their ability to form expectations, generate mismatch responses, and learn meaningful internal representations, in comparison to conventional supervised models. Our results show that relatively simple predictive learning objectives can give rise to these hallmark features better than less biologically plausible supervised learning. These findings suggest that predictive coding may provide a principled way to design neural network models that are not only learning systems but also better models of how the brain processes information. By investigating the integration of neuroscientific learning theories into deep neural networks, this work offers insights for both brain modeling and the development of more biologically grounded machine learning methods.

1. Introduction

The relationship between biological and artificial neural networks is a topic of growing importance, as insights from each field can inform and advance the other. While deep neural networks (DNNs) have emerged as a powerful model for investigating brain processes [1,2], their components and learning mechanisms are not directly aligned with our understanding of the brain as a biological neural network [3–8].

1.1. What exactly are the limitations of current deep neural network models as brain models?

DNN models are essentially machine learning constructs, typically optimized for specific tasks rather than biological plausibility. Consequently, these models rely on learning mechanisms and architectural choices that are only loosely aligned with our knowledge of the brain [3–8]. A key example is supervised backpropagation, which is ubiquitous in deep learning. 'Supervised' means networks are provided with an externally provided ground truth. While supervised learning signals are assumed to guide some (usually motor) processes in the brain [9–11], supervised mechanisms are generally considered much less likely than unsupervised mechanisms for guiding sensory learning in the brain [12–15]. 'Backpropagation' refers to the updating rule used for DNN which updates model parameters hierarchically from target to input, calculating partial derivatives at each layer and propagating them back [16]. In biological networks, such a mechanism becomes increasingly unlikely the more layers are involved [3,17–21]. Despite being powerful functional models, current DNN models (especially when trained with supervised backpropagation), are not plausible models of the brain.

While past neuroconnectionist research has largely focused on architecture [22–29], more recent work has shown that training objectives play a comparably large

role [30]. For example, in visual object recognition neuroscience, models have traditionally used supervised objectives and backpropagation-based optimizers [22–29]. However, unsupervised objectives like SimCLR [31] have emerged as more biologically plausible alternatives that can yield equally effective or better representations [30,32]. Nevertheless, these unsupervised rules are still often based on information-theoretical ideas rather than biological principles. A promising avenue of research is to look to neuroscience for inspiration on biologically plausible learning mechanisms for DNNs. Several such approaches have been proposed.

1. Hebbian learning rules, inspired by synaptic plasticity, are intuitively appealing but often too crude to learn complex relations [33–35].
2. Reinforcement learning aims to capture reward-based learning, but current models are not as effective as classical deep learning optimization [36–39].
3. Bayesian learning rules provide a convenient connection between probability/information theory and neuroscience. Bayesian learning [40,41] in general treats the brain as a 'prediction machine', by actively predicting future states and trying to optimize these predictions. The family of Bayesian learning rules includes more specified theories of neural learning, such as the free energy principle [42,43] and predictive coding [44,45]. These theoretical frameworks are more directly inspired by empirical results from neurophysiology and provide sufficient detail and operationalization to actually implement it in a computer. Due to their biological plausibility and clear cut formalization, they provide a well-suited candidate framework for building biologically plausible artificial neural networks.

In this research article, we will therefore focus on predictive coding models as the most suitable class of biologically plausible DNN models of the brain.

1.2. What is predictive coding?

Predictive coding (PC) posits that the brain is fundamentally engaged in generating predictions about incoming sensory inputs and updating internal models based on the mismatch between predictions and observations. The primary function of PC is to approximate the prediction (prior) as closely as possible to the actual future stimulus (posterior), enabling timely adaptation to new circumstances. Further, as an energy-minimizing procedure, PC proposes that only the residual, unpredicted information should be propagated to higher levels, as the expected information can be suppressed. The main computational operation is a subtraction between the prediction and the incoming stimulus (although more recent formulations also emphasize multiplicative operations such as the precision of prediction errors, linked to biological concepts such as neural gain control [46,47]). This PC framework has gained significant traction in neuroscience, as it provides a principled account for phenomena like hierarchical representations and error signals driving learning and adaptation [44–49].

The most popular PC-inspired algorithm is PredNet by Lotter et al. [50]. PredNet combines a convolutional neural network (CNN) with a long short-term memory (LSTM) recurrent layer and an autoregressive/predictive target, resulting in good performance on video prediction tasks. Recently, an alternative model - namely the Forward-Forward algorithm - has been proposed by Hinton [51]. It integrates ideas from Boltzmann machines, generative adversarial networks, and contrastive learning to form a training setting that can resemble the PC mechanism under the right conditions. Additionally, a range of other algorithms have been proposed, including work such as Millidge et al. [17,52] or Whittington & Bogacz [53]. These approaches often consist of custom architectures and custom updating rules, with errors explicitly calculated. While these approaches are inspired by biological processes, they are typically validated through proof-of-concept demonstrations showing they can learn useful representations or perform specific tasks, rather than through rigorous tests of their mechanistic biological plausibility [48]. Consequently, only a few of these models have been seriously considered as frameworks for understanding actual brain function [50,54].

1.3. What is the main goal of this work?

As interest in using DNNs to model brain processes grows, the field remains largely focused on high-performing architectures with limited biological grounding. It thus remains unclear to what extent PC-inspired DNNs realistically approximate neural dynamics or learning mechanisms in the brain. While there has been research arguing that PC-inspired networks provide various advantages to standard DNN models [55,56], such publications usually focus on theoretical or functional advantages, rather than on whether such models actually reproduce the behaviour expected from PC. To our knowledge, this paper presents the first systematic and independent evaluation of whether such biologically inspired models can serve not only as functional learning systems, but also as mechanistic models of biological neural computation. To address this, we compare several representative algorithms in a tightly controlled experimental setting. While the study does not aim to benchmark large-scale models optimized for performance, it focuses instead on understanding the mechanistic plausibility and dynamic properties of simpler, biologically motivated architectures.

This work aims to investigate whether PC inspired algorithms match phenomena or brain data better than classical supervised backpropagation in a simple perceptive setting. This would situate PC inspired algorithms as a promising alternative to other less biologically plausible approaches. To reach this goal, we compare PC inspired optimization approaches applied to one common simple Recurrent Neural Network (RNN) architecture (consisting of a feedforward input kernel and one square recurrent kernel). Since here we aim at comparing general PC-inspired algorithms and updating rules in a well-controlled setting, we focus on two candidates: A Predictive implementation, inspired by PredNet [57] and a Contrastive implementation, based on Forward-Forward [51]. We train these methods on a simple visual perception task, as a sufficiently complex and intuitively understandable task to demonstrate the effects of PC. As previous literature suggests that applying activity regularization to a network can by itself introduce PC-like effects in a network [58], we introduce an additional condition, where each of these models is trained with activity regularization applied next to the main objective. We then test whether these PC-inspired neural network models exhibit key neural signatures of predictive processing, in comparison to standard supervised and untrained neural networks. Specifically, we define three necessary landmarks which a PC network should exhibit:

1. Mismatch responses (MMR), which reflect the network's ability to detect and respond to unexpected or deviant stimuli, and is core to the concept of saving energy in PC literature [59–61].
2. Prior expectations (i.e., actively maintaining predictions about upcoming states), which are a necessary precondition for a PC system [62–68].
3. Learning of semantic information (i.e., the association of a stimulus to an abstract encoded concept over the course of training) without explicit supervision. This is a necessary attribute of any learning system.

We use these landmarks as dependent variables, quantifying to what extent the mentioned phenomena emerge in different model types (see Fig 1). Furthermore, since gain control is considered a crucial component of PC and enables flexible context-sensitive weighting of prediction errors [46,47], we perform an additional analysis to investigate whether ANN weight regularization can be used to simulate gain control.

2. Results

To investigate which neural network model aligns most closely with PC, we compared PC-inspired training objectives (mentioned in Introduction 1.3.) to a Supervised and Untrained baseline. We created several different training conditions:

1. Predictive global, where a predictive loss is applied after the final network output.
2. Predictive local, where a predictive loss is applied after each layer.

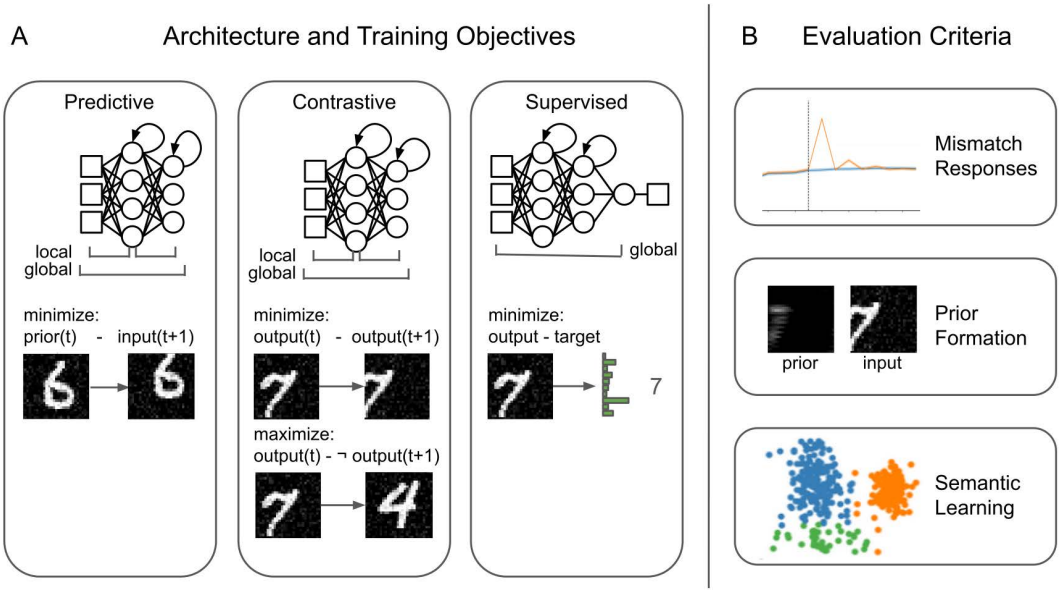

Fig 1. Illustration of the different training objectives (main experimental conditions) and evaluation criteria (dependent variables). A. Network Conditions: Depiction of each training condition (Predictive, Contrastive, Supervised), illustrating the simple Recurrent Neural Network Architecture (RNN) used as well as the optimization objective of each network. For the predictive algorithm, the supervised objective can only be applied using "global" gradient propagation (backpropagation) through the entire network. However, the Predictive and the Contrastive objective can be applied on a "local" scale, with each layer being trained on the inputs and outputs of the current layer. **B.** Evaluation criteria: Illustration of the three dependent variables to assess phenomenological hallmarks of PC in the networks.

https://doi.org/10.1371/journal.pcsy.0000076.g001

3. Contrastive global, where a contrastive loss is applied after the final network output.
4. Contrastive local, where a contrastive loss is applied after each layer.
5. Supervised, a control condition where a supervised loss is applied after the final layer.
6. Untrained, a control condition in which the network is not trained at all.

All conditions were trained on identical series of images (for details see Methods 4.2.) in an identical simple RNN architecture (see Fig 1). While the Supervised condition was trained to correctly identify the number presented in each image of the series, the PC-inspired conditions were trained according to unsupervised PC objectives (for details see Methods 4.3.1.). We then compared all six training conditions in respect to the hallmarks of predictive coding defined in Introduction 1.3.: (1) generation of mismatch responses, (2) formation of prior expectations, and (3) learning of class-specific semantic information. In the following section, we review each evaluation criterion and explain which model condition shows the corresponding hallmarks of PC. Since activity regularization may also evoke PC-like dynamics in networks, we first present the results for networks trained without activity regularization, and then for those trained with additional activity regularization.

The main analysis reported here was performed on a moving MNIST [69] paradigm, which was specifically designed to present the models with a challenging yet controlled set of input stimuli. As a confirmatory procedure, all parts of the main analysis were additionally performed on the BAIR robot pushing dataset [70], consisting of real-life video data of a robot arm interacting with objects. We used the BAIR dataset to ascertain that the effects found in the artificially generated MNIST setting can be replicated in a more ecologically valid real-life video perception setting. The results of this confirmatory analysis are summarized in Results 2.4. and fully reported in Section A of S1 Appendix.

2.1. Mismatch responses

In PC, the occurrence of MMR is directly tied to deviations from what was expected, resulting in elevated activity [71]. We expect similar MMR patterns in PC-inspired networks. To assess the models' ability to detect and respond to unexpected or deviant stimuli, we evaluated their MMR (measured by deviations in the evoked neural response) to a series of matching and mismatching input sequences (see Fig 2). The matching series condition consisted of a sequence of MNIST images with random augmentations, while the mismatching series condition included random augmentations and random switches in the semantic content of the images (e.g., a 3 suddenly changing to a 7). We ensured that the match and mismatch condition did not show any base level differences that might result in a difference in mean-squared activation from the network's layers (for details see Methods 4.4.1.). We then measured the mean square layer activations of the networks over time, similar to evoked responses. Our metric of interest was whether the match and the mismatch condition produced significant deviations in the model output for the sample immediately following the stimulus switch.

For networks without activity regularization (see Fig 3; for details see Section D1 of S1 Appendix), most of the PC-inspired conditions showed significant MMR. Specifically, the Contrastive global (T(11998)=-41.58071, p(corrected)<1e-5), Contrastive local (T(11998)=-30.00066, p(corrected)<1e-5), and Predictive local (T(11998)=-40.36738, p(corrected)<1e-5)

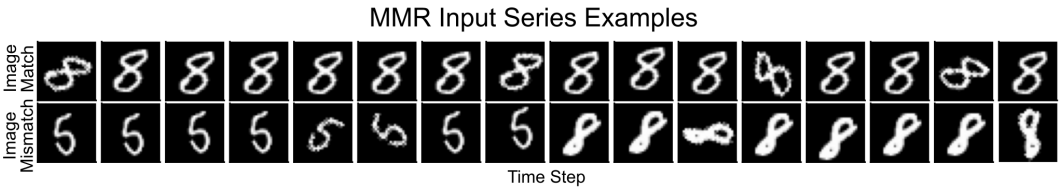

Fig 2. Example image sequences for the matching and mismatching conditions. The top row shows a sequence of MNIST digit images with random shifts and noise, but no changes in the semantic content (matching condition). The bottom row shows a sequence with the same types of visual transformations, but with random switches in the digit identity (mismatching condition).

https://doi.org/10.1371/journal.pcsy.0000076.g002

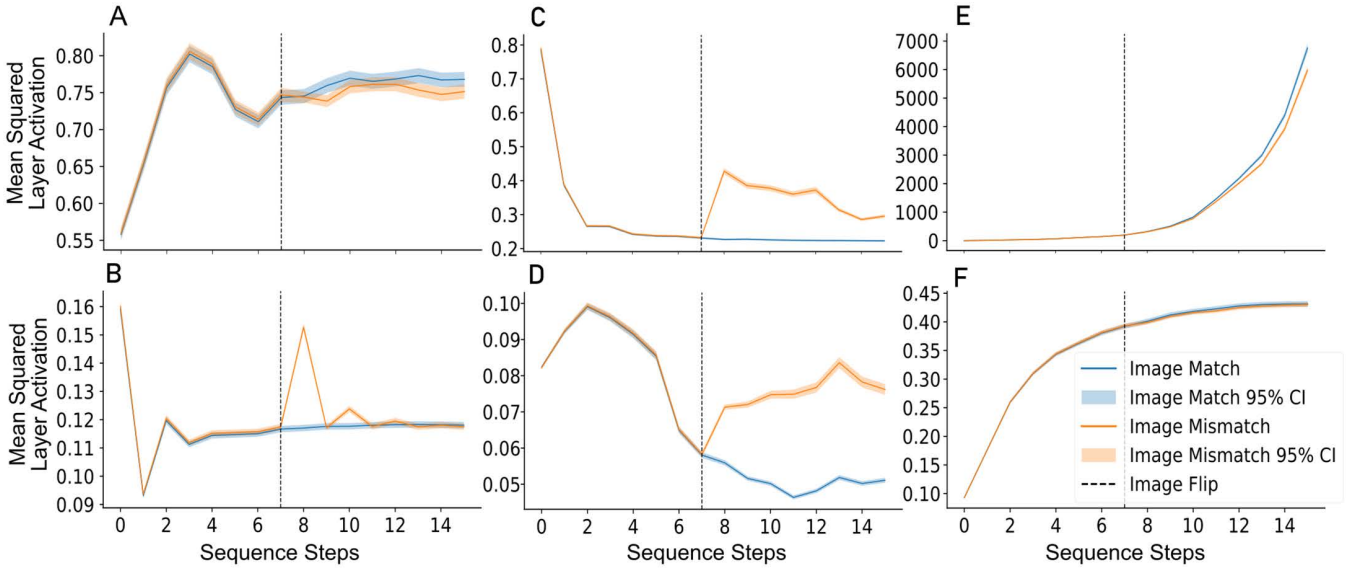

Fig 3. Mismatch responses for unregularized conditions: A. Predictive global; B. Predictive local; C: Contrastive global; D: Contrastive local, E: Supervised; F: Untrained. Each subfigure shows the Mean Squared Average activation of a layer over different time steps, separated for sequences when an unexpected change is happening versus when an expected change is happening. Shaded areas show 95% Confidence intervals.

https://doi.org/10.1371/journal.pcsy.0000076.g003

conditions exhibited significant changes in mean squared layer activity in response to a semantic stimulus switch. The Predictive global condition did not show a significant overall MMR (T(11998)=0.22222, p(corrected)=1). Visual inspection (see Fig C2 in S1 Appendix) revealed that this was due to a positive MMR in the first layer and a negative MMR in the second layer, which canceled each other out in the overall sum (for further discussion, see Discussion 3.1.). Neither the Supervised (T(11998)=2.28285, p(corrected)=1) nor the Untrained (T(11998)=0.72614, p(corrected)=1) conditions showed a significant MMR immediately after the stimulus switch.

To investigate whether these effects persist if an additional energy constraint is added, we turned to activity-regularized networks. For activity-regularized networks (see Fig 4; for details see Section D2 in S1 Appendix), all trained conditions, including Predictive global (T(11998)=-11.68170, p(corrected)<1e-5), Predictive local (T(11998)=-20.63438, p(corrected)<1e-5), Contrastive global (T(11998)=-23.97432, p(corrected)<1e-5), and Contrastive local (T(11998)=-15.89864, p(corrected)<1e-5). Even Supervised (T(11998)=17.60957, p(corrected)<1e-5), exhibited a significant MMR, however it is important to note that in this condition the effect was inverted, meaning that the mismatch condition produced less overall activity than the matching condition. The overall effects suggest that activity regularization reinforces or even produces MMR-like effects in artificial neural networks (this effect is further elaborated in Discussion 3.1.). The Untrained condition did not show significant MMR effects (T(11998)=0.72614, p(corrected)<1e-5).

In summary, the PC-inspired conditions, especially the locally trained ones, were generally able to generate clear mismatch responses, with the exception of the Predictive global condition. Activity regularization appeared to induce MMR-like patterns across all trained models, and even result in (inverted) MMR-like behavior in the Supervised condition.

2.2. Prior expectations

Prior predictions are a necessary component of a PC system. They allow the network to match up the prior prediction with incoming information and save energy by only propagating relevant information [62–68]. To measure to what degree the neural network models formed an inherent prior representation, we evaluated the similarities between the neural network's prior state (negative latent state times recurrent kernel) and the expected future input. If a neural network works according to PC principles, these prior states should start to approximate the future input states over the course of learning. To

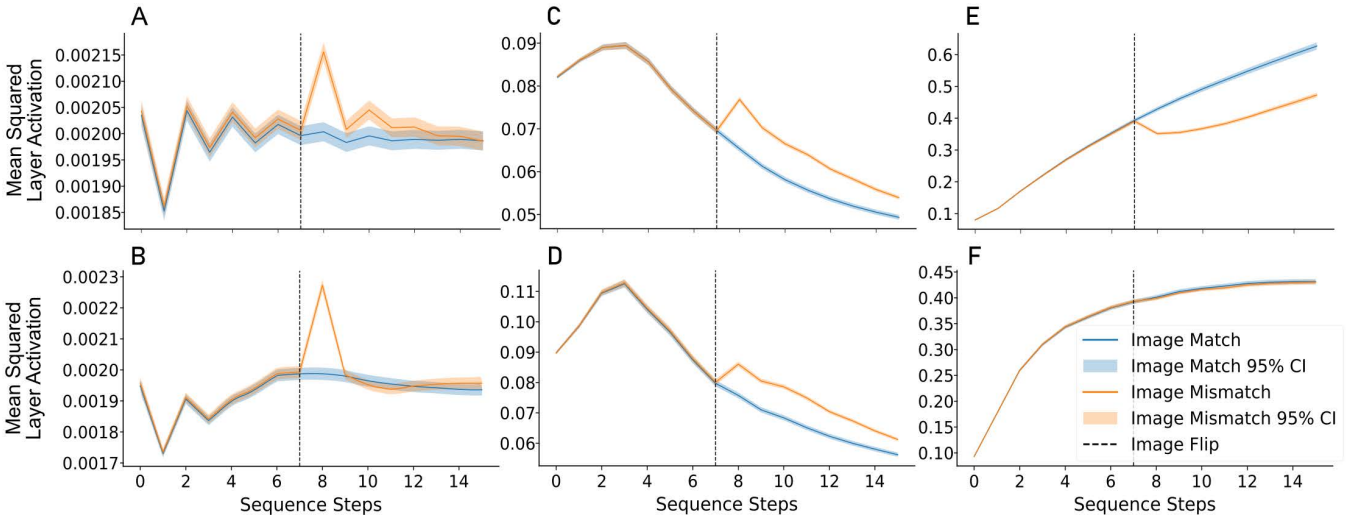

Fig 4. Mismatch responses for activity regularization condition: A. Predictive global; B. Predictive local; C: Contrastive global; D: Contrastive local, E: Supervised; F: Untrained. Each subfigure shows the Mean Squared Average activation of a layer over different time steps, separated for sequences when an unexpected change is happening versus when an expected change is happening. Shaded areas show 95% Confidence intervals.

https://doi.org/10.1371/journal.pcsy.0000076.g004

investigate this effect, we introduced a series of stimuli to all networks and correlated its prior state (latent state multiplied by recurrent kernel) with the actual incoming stimulus. A high correlation in this context means that the prior state approximates the future input well, while a zero correlation means that the prior state and the future incoming input are unrelated (for details, see Methods 4.4.2.).

For unregularized networks, Predictive global (r = 0.80507) and Predictive local (r = 0.39563) showed clear correlations between the prior and the next stimulus. In contrast, Contrastive global (r = 0.01178), Contrastive local (r = -0.01456), Supervised (r = -0.00260), and Untrained (r = -0.00249) did not exhibit any prior-stimulus correlations. Accordingly, only Predictive global (Z(11998)=61.07541, p(corrected)<1e-5) and Predictive local (Z(11998)=23.05043, p(corrected)<1e-5) correlated significantly stronger than the Untrained condition. Supervised (Z(11998)= -0.00612, p(corrected)=1.), Contrastive global (Z(11998)=0.78126, p(corrected)=1.) and Contrastive local (Z(11998)=-0.66089, p(corrected)=1.) did not show prior-stimulus correlations significantly stronger than the untrained condition (see Fig 5; for details see Section D1 in S1 Appendix).

The same pattern held true for the activity-regularized networks: Predictive global (r = 0.78028) and Predictive local (r = 0.57463) demonstrated strong prior-stimulus correlations. Meanwhile, Contrastive global (r = 0.00826), Contrastive local (r = -0.00059), Supervised (r = -0.01706), and Untrained (r = -0.00249) did not show any clear prior formation. Like in the unregularized condition, only Predictive global (Z(11998)=57.41855, p(corrected)<1e-5) and Predictive local (Z(11998)=35.97105, p(corrected)<1e-5) correlated significantly stronger than the Untrained condition. Supervised (Z(11998)= -0.79785, p(corrected)=1.), Contrastive global (Z(11998)=0.58876, p(corrected)=1.) and Contrastive local (Z(11998)=0.10407, p(corrected)=1.) did not show prior-stimulus correlations significantly stronger than the untrained condition (see Fig 5; for details see Section D2 in S1 Appendix).

In summary, the Predictive condition was the only one that consistently exhibited the formation of meaningful prior expectations across the different network configurations. The other conditions, including Contrastive, Supervised, and Untrained, did not show evidence of prior formation.

2.3. Learned representations

To assess each model's ability to learn abstract representations, we examined the encoded information content in each model by decoding the original number classes represented by the input from the model's output state. While the

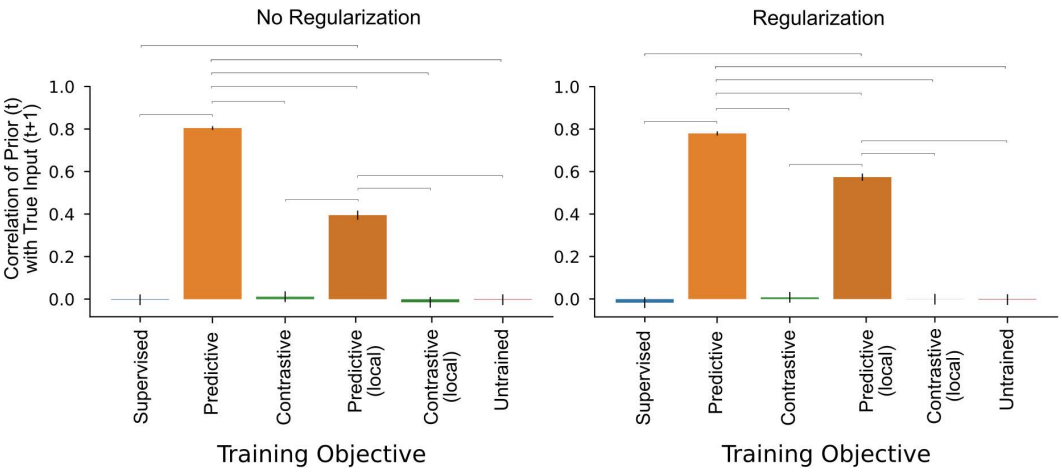

Fig 5. Correlations between priors (project to input level) and future input. Left: No regularization, Right: With activity regularization. High correlation means that the prior (negative recurrent state) projected back to the input level correlates highly with the next (unseen) stimulus in the sequence. Low correlation means the state and the next stimulus are uncorrelated. Solid line: p < 0.0001.

https://doi.org/10.1371/journal.pcsy.0000076.g005

Supervised condition has been explicitly trained to fulfil this task, the PC conditions were optimized on unrelated unsupervised targets. Accordingly, we use better-than-untrained decoding performance as an indicator of the PC network's capability of learning semantic information under unsupervised learning conditions.

The Predictive global condition (accuracy = 0.38667, SE = 0.01232) not only performed notably better than the untrained network (T(11998)=7.57907, p(corrected)<1e-5), but not significantly better (T(11998)=2.26123, p(corrected)=0.35644) than the Supervised condition (accuracy = 0.36667, SE = 0.01219), which still performed notably better than Untrained (T(11998)=5.31126, p(corrected)<1e-5). The Contrastive global condition (accuracy = 0.36550, SE = 0.01219) also showed substantial learning beyond the Untrained condition (T(11998)=5.17835, p(corrected)<1e-5), though insignificantly lower than Predictive global (T(11998)=2.39393, p(corrected)=0.25026). The Contrastive local condition (accuracy = 0.35217, SE = 0.01209) (T(11998)=3.65364, p(corrected)=0.00389) as well as the Predictive local condition (accuracy = 0.34700, SE = 0.01204) (T(11998)= 3.05970, p(corrected)=0.03331) still exhibited significant learning above the Untrained baseline (accuracy = 0.32000, SE = 0.01180). These results are illustrated in Fig 6 (For details see Section D1 in S1 Appendix).

Combining activity regularization with the predictive algorithm resulted in a notable decline of learning performance (see Discussion 3.1.). In this condition, the learning effects diverged more across between models: Predictive global (accuracy = 0.33300, SE = 0.01192) and Predictive local (accuracy = 0.23017, SE = 0.01065) both showed decreased learning performance, with Predictive global performing worse than Supervised (T(11998)=2.82368, p(corrected)=0.07133) and on par with an Untrained Network (T(11998)=1.42093, p(corrected)=1.) Predictive local even performing notably worse than the Untrained condition (T(11998)=-12.62837, p(corrected)<1e-5). In contrast, Contrastive global (accuracy = 0.36600, SE = 0.01219)(T(11998)=5.23532, p(corrected)<1e-5) and Contrastive local (accuracy = 0.37350, SE = 0.01224) (T(11998)=6.08825, p(corrected)<1e-5) exhibited increased learning compared with the Untrained condition. The Supervised condition still performed notably better than the untrained condition (T(11998)=4.24578, p(corrected)=0.00033). These results are illustrated in Fig 6 (For details see Section D2 in S1 Appendix).

To conclude, our analysis of semantic learning effects showed that all algorithms in almost all settings exhibited significant learning of category information beyond the untrained baseline condition. In the unregularized setting, all training conditions showed significant learning effects beyond the untrained baseline. While this is to be expected for the Supervised condition, it shows that the PC objectives as well are capable of instilling semantic information into a model despite their unsupervised training objective.

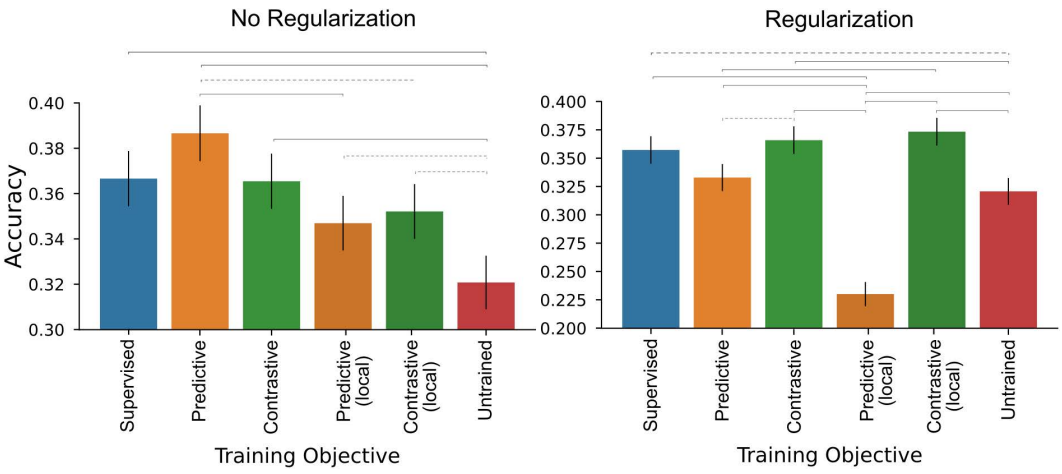

Fig 6. Learning Performance evaluated by classification accuracy of the original stimulus classes based on the output encoding of the neural network models. Left: No regularization, Right: With activity regularization. Dotted line: $p < 0.05$. Solid line: $p < 0.0001$.

https://doi.org/10.1371/journal.pcsy.0000076.g006

2.4. Generalization to real video data

To assess whether the main findings generalize to more complex, real-world data, we repeated the full analysis on the BAIR Robot Pushing dataset. This dataset introduces greater visual and semantic complexity, as well as continuous target variables. The results confirmed our main findings: both Predictive conditions exhibited strong prior formation, with high correlations between the internal prior and the upcoming stimulus. Significant mismatch responses were also observed in the Predictive local and both Contrastive conditions, mirroring the main analysis. While the learning evaluation showed similar trends—with Predictive and Supervised conditions outperforming Contrastive and Untrained models—the differences were not statistically significant under multiple comparison correction. Nonetheless, the pattern of results closely replicates the structure observed in the original analysis, indicating that key effects of prior formation and mismatch sensitivity robustly extend to more complex visual input and continuous output spaces. Overall, this supplementary analysis supports the generalizability of our core conclusions beyond the original experimental setting. The full replication is reported in Section A in S1 Appendix.

2.5. Gain control

In PC, the noise level of the output signal is correlated with the noise level of the input. Gain control is the ability to manipulate the output noise level independently of the input noise [46,47]. As gain control is an important mechanism in PC literature [46], we investigated whether it is possible to simulate this mechanism in PC networks using weight regularization. We created two input conditions with high and low noise levels and evaluated whether weight regularized networks significantly reduce the difference in noise level of the model's output. We did this by comparing the variance ratio between low and high noise input for unregularized versus weight regularized networks.

Our results showed that weight regularization results in a gain control-like effect, reducing the neural activity's variance ratio between high noise and low noise input stimuli. Contrastive ($Z = -67.08483$, p(corrected)<1e-5), Predictive ($Z = -64.67718$, p(corrected)<1e-5), and Supervised ($Z = -67.08477$, p(corrected)<1e-5) all showed significant reduction in output variance ratios between high noise and low noise when weight regularized. From visual inspection of Fig 7, it can be seen that all algorithms have undergone notable reduction in variance differences with applied weight regularization. However, weight regularization did not lead to perfectly identical output variance distributions for any algorithm, with Contrastive ($Z = -67.08478$, p(corrected)<1e-5), Predictive ($Z = -66.92948$, p(corrected)<1e-5), and Supervised ($Z = -36.19967$, p(corrected)<1e-5) all still showing significance variance difference between high noise and low noise samples (for details see Section D4 in S1 Appendix).

3. Discussion

The overarching aim of this research was to investigate whether common neural network architectures can be adapted to exhibit key signatures of PC. Specifically, we were interested in exploring the models' ability to form prior expectations, generate mismatch responses, and learn robust representations in an unsupervised manner. The results of this study suggest several key insights: (1) Networks trained with predictive coding-inspired objectives generally exhibited more characteristics of predictive coding compared to non-PC-inspired objectives. (2) The local Predictive condition (without activity regularization) was the most likely candidate for predictive-coding like effects due to showing clear MMR, priors, and learning effects. (3) The contrastive approach showed very pronounced mismatch responses and good learning performance, especially with activity regularization. However, it did not lead to the explicit formation of prior expectations. (4) Activity regularization seemed to evoke mismatch-like effects across all trained networks, but decreased the learning performance of the predictive and locally predictive networks. (5) Weight regularization assimilates output variance of networks, irrespective of the noise level of the input. This mechanism can be used to model and investigate the biological process of gain control.

In the following section, we discuss these findings in more detail.

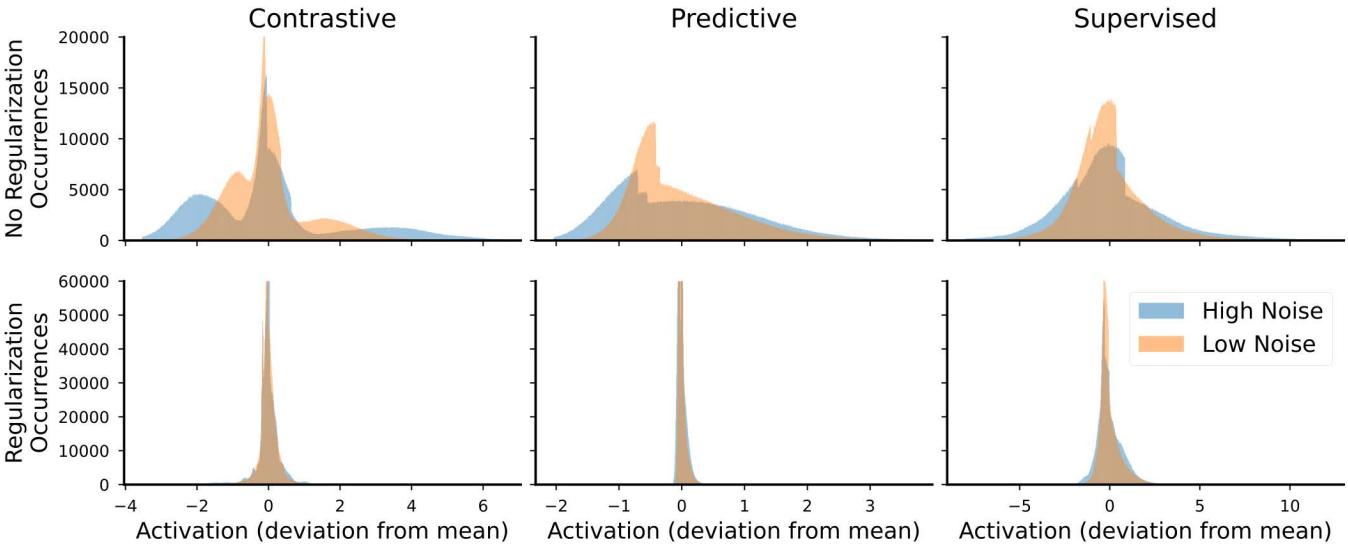

Fig 7. Distributions of output activations for unregularized and weight-regularized networks, for high noise and low noise inputs. Output activations from high noise inputs are shown in blue, output activations from low noise inputs are shown in orange. Weight regularization leads not only to normalization of the neural output distribution, but also reduces the ratio in output variance in response to high noise and low noise inputs.

https://doi.org/10.1371/journal.pcsy.0000076.g007

3.1. To what extent did the PC inspired models exhibit landmarks of biological PC?

Mismatch Responses: All PC-inspired networks showed some form of MMR (i.e., stimulus mismatch induced positive activity) in at least one layer. We found the strongest MMR in the Contrastive condition, likely due to the fact that the Forward-Forward implemented here is directly optimized towards maximizing the output activity after occurrence of mismatching stimuli. The Predictive local condition also evoked clear MMR. Interestingly, the Predictive global condition produced a negative MMR in the first layer and a positive MMR in the second layer, which cancelled out when added together (see Fig C2 in S1 Appendix). This phenomenon likely arises from the fact that the globally defined predictive function is defined to facilitate MMR at the final layer, which might lead to opposite effects in the previous layers, meaning that (goal-oriented but negative) weight patterns that are punished in the final layer are more likely to appear in previous layers. A comparable pattern is observed in the brain as well, where repetition suppression and repetition enhancement can occur at different levels of the cortex [47]. However, further research is needed to determine if it is analogous to neurophysiological MMR or an unrelated occurrence in the artificial network. Overall, the clear MMR spikes in PC-inspired networks sets them apart from Supervised networks, which does not show such clear spikes. These results provide further evidence that MMR are a consequence of prediction-based learning, as claimed in PC literature [59]. They further confirm that complex semantic information can be learned in an unsupervised fashion by maximizing the contrast in MMR (as proposed in [51]).

MMR responses were strongly enhanced or even evoked by activity regularization. This suggests activity regularization may be a promising approximation of the energy-saving principle behind PC [58]. Interestingly, we find that activity regularization evokes an inverted MMR-like effect in Supervised recurrent networks, where the output activity for mismatching stimuli is lower than for matching stimuli. This does not directly align with the idea of PC, where energy should be saved by only propagating unexpected information, meaning that mismatch activity is expected to be larger than expected activity. This effect can be explained by the combination of the Supervised learning objective in combination with activity regularization: Supervised RNN models have the tendency to build up activity over time through accumulating activation over the recurrent state (this is an issue related to the exploding gradients problem and discussed in [72,73]). In unregularized

networks, this leads to an exponential increase in activity (see Fig 3) as long as the network is presented with unchanching patterns that accumulate and amplify in the recurrent state. Now when a mismatching input appears, the activity builds up slightly slower because the new input tends to cancel out activity rather than further amplifying existing patterns. In the activity regularization setting, this activity accumulation is decreased due to increased penalization of high activity (see Fig 4). However, similar to the unregularized condition, the stimulus switch tends to cancel out activity rather than amplifying existing patterns, leading to reduced activity for the mismatch condition. This might explain the effect of the MMR-like pattern in the Supervised model with activation regularization.

Further, we found that the application of activity regularization in combination with a Supervised objective, resulted in unexpected behavior. The Predictive condition performed well overall, but was heavily impaired when combined with activity regularization. An interesting aspect of this phenomenon is that the Predictive models with activity regularization still exhibited very strong correlations between the current prior and the future state, but seemed to encode drastically less class-specific semantic information. This suggests that the activity regularization forced the loss in the Predictive objective to only focus on the Predictive information which is shared between classes, creating relatively high predictions from this generally shared variance, while at the same time cutting all the activity that would carry important information for specific classes, but which occur less often and therefore contribute too little signal to withstand the activity regularization.

Prior expectations: Contrary to the broad occurrence of MMR, only the Predictive condition showed clear prior formation, while the Contrastive, Supervised, and Untrained conditions did not. Empirical evidence shows that predictions in the brain are usually organized topographically. This has been widely shown in the visual, auditory, sensory, and other systems [74–79]. As PC builds upon this idea of neuron-by-neuron topographically ordered predictions, this suggests that a population-activity based contrastive model such as Forward-Forward [51] might not be a perfectly suitable model of PC. Further, this suggests that explicit definition of a prior in the Predictive algorithm may lead to the appearance of an MMR as a secondary effect, while the opposite (i.e., the explicit definition of a MMR in the Contrastive algorithm implying the formation of a topographically organized prediction) does not hold. This insight can be further used to investigate alternative models of PC and Bayesian brain theories [48,49,59,80–83]. MMR are among the main phenomena that lead to the development of the theory of PC and are frequently used as evidence for it [47,59]. If there are learning systems that exhibit MMR, but do not exhibit explicit prior predictions, then models such as the ones presented here can help in investigating the theory by evaluating whether a predictive MMR model or a non-predictive MMR model fits the brain better.

Learned Representations: Finally, to assess the learning effects of the different training approaches, we compared the performance of the models on the task of classifying the original MNIST digit classes based on the encoded representations in the networks. In general, predictive and contrastive models showed better-than-random learning effects. Supervised showed the strongest learning effects, because the algorithm was specifically optimized to perform a classification task on a similar set of input stimuli. One important aspect to consider is that the Predictive model used in this study performed well at encoding semantic information about stimuli, while similar investigation into the classical PredNet architecture [84] did not show strong encoding capabilities. This is possibly related to the lack of semantic encoding occurring in the convolution-based PredNet architecture, which has been fixed in the presented predictive model (see Methods 4.1.). While the Supervised algorithm still showed the best learning performance, our results show that PC-inspired algorithms might be useful for many learning contexts, adding to other research confirming PC's usefulness for tasks like transfer learning and generalization [85].

Gain Control: In an additional analysis we found that weight regularization added to neural networks during the training process results in a variance normalization of the output activations. In unregularized networks, the variance of the network's output activity correlates with the variance of the input. However, when weight regularization is applied, this correlation is weakened, meaning that noisy inputs almost produce the same output activity variance as clean inputs. We successfully showed that weight regularization can be used to implement gain control mechanisms in DNN models of the brain. Such a gain control mechanism may achieve two aims, consistent with the PC framework: 1. scaling output variance to compensate for input variance, and 2. encouraging the network to learn sparse representations and minimize

activity to conserve energy [43,45,86]. Additionally, this effect should be taken into account when using DNN as neuro-connectionist models while changing the noise levels in the input (such as [87,88]). In such cases, networks trained using regularization could lead to different distributional properties of the activation, which might affect outcome or decodability in downstream (encoding, decoding, RSA) analyses.

Summary: Taken together, these results indicate that under certain circumstances, the neural network models trained with PC-inspired objectives were able to display hallmarks of predictive processing. Specifically, the locally trained predictive objective (without activity regularization) in a RNN exhibited these traits most closely. This model was able to fulfill the expected landmarks of PC, including the formation of meaningful priors, the generation of mismatch responses, and the learning of informative representations, without requiring extensive gradient propagation throughout the network hierarchy. This strongly supports the idea that simple predictive objectives can serve as valid approximations of PC. These models can exhibit behaviors resembling information processing in the brain better than standard DNN models. Our results not only show that PC is effective as a learning mechanism (which has been demonstrated in other studies; [17,50–53,89–92]), but also connect the learning effects to the phenomenological markers theorized as part of PC theory (such as priors and MMR), confirming them as a possible and logically coherent consequence of the PC mechanism.

3.2. What are the key limitations of the current study and how can they be addressed?

While this study provides valuable insights into the relationship between artificial and biological learning, it highlights the need for further exploration and refinement of PC-inspired neural network models. Future research should scale up the model complexity by exploring more sophisticated architectures with additional layers and increased capacity. This could uncover new emergent properties and better align the artificial networks with the brain's hierarchical structure. Specifically, incorporating deeper networks with longer propagation of backwards errors could better mimic the hierarchical information processing theorized by PC.

Another very common way of showing alignment between DNN models and the brain are encoding models and benchmarks like Brain-Score [1], where models are compared in terms of how well they predict neurophysiological or behavioral data. Over recent years, there has been much progress using functionally performant DNN models like CNN or Transformers as models to investigate brain semantics and representations. While these functionally performant models are very successful at brain encoding or representational similarity tasks (i.e., explaining representational variance in neurophysiological data of humans performing semantic tasks) [1], using them to investigate specific mechanisms of neural functions is inherently limited by their mechanistic capability to adequately represent the corresponding mechanism in the human brain. For example the investigation of feedback mechanisms in the brain requires a DNN model containing manipulable variables that mechanistically represent such a feedback mechanism. In this research, we chose to investigate PC networks specifically, as a potential future model category allowing researchers to investigate variables which cannot be investigated using less biologically plausible DNN models. However, PC networks are rarely investigated in respect of semantic or representational likeness to the brain, largely due to being mostly implemented at a very small scale and therefore often performing worse than larger DNN models in representational aspects [48]. Therefore, as a starting point, we decided to focus on the phenomenological and dynamical behavior of PC networks as this mechanistic and biological interpretability is the main advantage of PC networks compared to larger DNN models. Nevertheless, it would be helpful to address semantic and representational capabilities of these networks in further research.

Another possible improvement is to test these algorithms on more challenging and diverse tasks beyond the moving MNIST and BAIR robot pushing datasets used in this work. The PredNet model [57], for example, has been evaluated on high level visual data including more complex and more diverse naturalistic inputs [93]. However, PredNet has been criticized for issues related to its biological plausibility [84]. The predictive model used in the current study aims to address these biological plausibility concerns, but it has not yet been tested on larger-scale tasks. Similarly,

the contrastive model, which is based on the Forward-Forward algorithm [51], as well as other related algorithms [91], have only been evaluated on MNIST or moving MNIST, which are relatively simplified tasks. MNIST or moving MNIST may not generalize well to more complex computer vision tasks [94]. While we show that the effects presented in this publication do indeed largely generalize to real life video data, further evaluation on larger and more complex datasets which include more naturalistic conditions, background, and object interactions. By testing these PC-inspired models on a wider range of tasks, including those with more naturalistic and diverse inputs, researchers can better assess their generalization capabilities and further explore the computational principles underlying biological neural information processing.

3.3. Implications for biological and artificial neural learning

The insights from this research can be used to create better DNN models of the brain. The neuroconnectionist framework relies on creating neural network models and experimentally altering different mechanisms to investigate how changes in the model improve or decrease its similarity to the brain [1,22]. However, research into PC often relies on biologically implausible machine learning networks [26,84,95] despite previous research showing that well-performing machine learning models are not always better neuroconnectionist brain models [1,30]. The results presented here suggest that the PC-inspired models, while currently only sparsely being used as neuroconnectionist research models [50], might be a promising future direction in brain modelling.

From a machine learning perspective, the results from this paper support existing evidence that PC-inspired algorithms based on Predictive as well as Contrastive objectives [51,57] are promising methods for building effective learning algorithms. The models presented in this work perform well in their role as unsupervised machine learning models, even after addressing many of the biologically implausible aspects that have been criticized in classical deep neural networks. Specifically, such models could serve as a foundation for developing new artificial neural network models for time-dissolved or sequential tasks such as time series or video prediction. While PC-inspired models currently may not surpass existing state-of-the-art models in unsupervised learning [96–98], the elegant way in which PC integrates a prediction-based framework with complex semantic learning systems poses a strong foundation for creating simple and effective theory-driven models for unsupervised learning.

4. Methods

4.1. General model architecture

To implement the PC-inspired neural learning algorithms, and to facilitate comparisons between distinct models, we used a simple RNN architecture as a basis for all models (see Fig 8). We chose this straightforward RNN architecture to demonstrate the core properties of PC in a maximally simple and well-controlled setting. Specifically, all conditions were trained on a model with two vanilla RNN layers - both composed of a forward kernel as well as a recurrent kernel. The recurrent connections in the RNN allow the model to generate some form of prior expectations, which can then be integrated with incoming sensory input. Specific details on the architecture and hyperparameters are given in Section B in S1 Appendix. All code to replicate this study are published under github.com/DiGyt/predictive_coding_algorithms. Further data and pretrained models to directly replicate the results are published under https://doi.org/10.5281/zenodo.17228391.

By using a simple RNN rather than more complex architectures, we aimed to isolate the key PC principles without the confounding factors that could arise in deeper or more specialized network designs [99]. Further, this constant architecture allows us to isolate training effects induced by the training objective and optimization procedure, omitting effects created by varying architectures. Further details and an overview table on architectural hyperparameters for the main analysis as well as the BAIR replication analysis are reported in Section B in S1 Appendix.

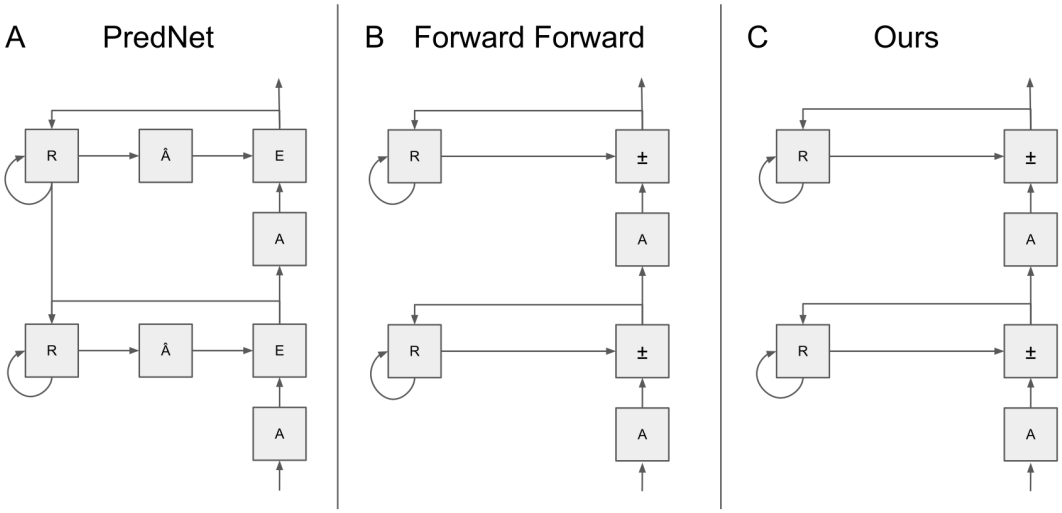

Fig 8. A. Illustration of the PredNet Architecture [57]. B. Illustration of the Recurrent version of the Forward Forward architecture [51]. **C.** Illustration of the Simple RNN Architecture used here. While PredNet is a high level approach for self-supervised video learning, we use SimpleRNN to create the most minimalistic and assumption-free type of network capable of expressing prior predictions. There are 3 main differences between the two models: (1) The top-down connections presented in PredNet are omitted in the SimpleRNN for the sake of simplicity. (2) The layers in PredNet are specified for learning of visual information, using convolutional layers, convolutional LSTMs, max-pooling layers. In the Simple RNN, we use only a minimal amount of weights, consisting of one input kernel and one recurrent kernel. (3) PredNet maintains the pixel-level state of the image throughout the entire network hierarchy, meaning that the network does not encode the image sequence, but only applies changes to it. This means that PredNet output states do not provide a biologically plausible representation of neurons that might be interpreted as pure positive firing patterns of neurons. This maintenance of the pixel-level variable in combination with using ReLU activation functions (which are only capable of expressing positive values) also requires an explicit handling and splitting up of positive and negative correction errors. In the SimpleRNN model, the states are encoded and maintained as purely positive values (interpretable as firing patterns). In this network, the expression of positive and negative error corrections takes place through positive and negative weights in the input and recurrent kernel, which are applied *before* the activation function is applied.

https://doi.org/10.1371/journal.pcsy.0000076.g008

4.2. Training environment

The neural network models in this study were trained on a moving MNIST [69] paradigm, which was specifically designed to present the models with a challenging yet controlled set of input stimuli. This paradigm involves a series of MNIST digit images that move across the input field in a particular order, with the digits moving at different speeds (comparable to [91]) (see Fig 9). Specifically, the movement of the MNIST digits follows sine wave patterns, with the phase and direction of the sine waves on the horizontal and vertical axes determined by the class of the number being presented. This approach encourages the networks to develop robust representations that link the digit semantics to their dynamic visual features, rather than just memorizing static pixel-level details. We use MNIST because it is a typical dataset to test new DNN architectures on, and has been used as a test dataset for various novel DNN architectures [51,91]. However, as a dataset for semantic classification, MNIST is notoriously easy to solve [100]. In order to enhance classification difficulty and improve generalization, we applied a variety of random augmentations to the input images. These augmentations included shifts (up to ~±21% of the receptive field with a standard deviation of ~14% of the receptive field) and the addition of Gaussian noise (+33% standard deviation relative to noise-free images). By introducing these transformations, we ensured that the networks could not simply memorize the pixel-level details of the inputs, but had to learn more abstract representations. Additionally, all input images were normalized to a 0–1 scale, rather than the original 0–255 range, to help standardize the data.

The main dataset consisted of 54,000 MNIST samples drawn from the original training set, with an additional 6,000 images held out for testing. During training, the data was shuffled each epoch and then batched up in batches of 512

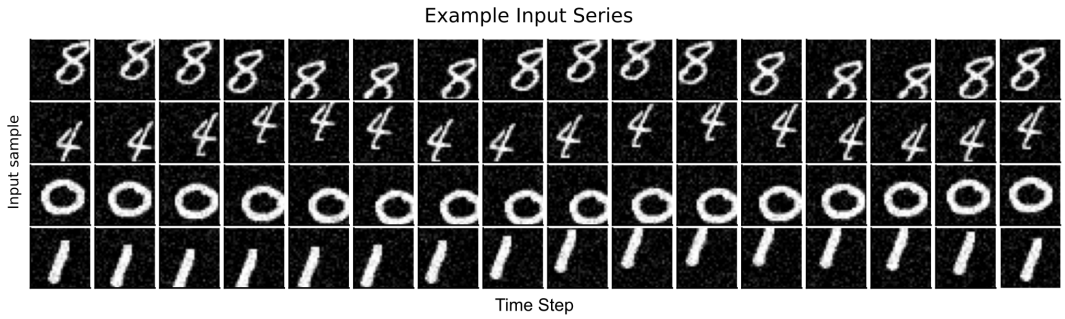

Fig 9. Illustration of the input series. Networks were trained according to a moving MNIST paradigm. For each batch, multiple samples were fed to the network. Each sample contained a series of images, depicting a number moving across the visual field. Each separate number class moves across the visual field according to specific rules (direction and magnitude in change of x and y coordinates per time step). These rules were shared between samples of the same number class, but differed from samples of other number classes.

https://doi.org/10.1371/journal.pcsy.0000076.g009

samples per batch (and 240 samples for the final batch). The preprocessing for the BAIR robot pushing dataset which was used for our confirmatory analysis are reported in Section A in S1 Appendix. Further details and an overview table on training hyperparameters for the main analysis as well as the BAIR replication analysis are reported in Section B in S1 Appendix.

4.3. Model variations

4.3.1. Training objectives. In order to easily compare different predictive-coding like training objectives with supervised backpropagation while omitting potential nuisance factors from the network architectures, we adapted two common PC-inspired algorithms to work with a simple RNN architecture (see Fig 8). This is especially relevant since common PC algorithms such as PredNet are specifically designed to work with Image-type data (including convolutional network layers) and therefore comparisons to other algorithms on a fundamental level are not trivial. In this section, we describe the core inspiration for these algorithms and how we adapted them. The different training conditions are illustrated in Fig 1.

Predictive: One of the training procedures we investigated was a predictive, or autoregressive, objective. This approach was inspired by the PredNet model, which aims to predict the next frame in a sequence of inputs. PredNet is a predictive-coding inspired DNN architecture published by Lotter et al. [57]. The architecture consists of a series of stacked modules, each comprising a convolutional input layer, a recurrent layer, a prediction layer, and an error representation layer. Each module makes local predictions about the input, and the deviations from these predictions are forwarded to higher layers, allowing the network to learn the temporal structure of visual data. This hierarchical approach enables the model to predict future frames in video sequences while capturing essential aspects of object movement and scene dynamics, meaning the loss is simply the mean squared error (MSE) between the model's final output and the stimulus image of the next time step. However with its architecture, PredNet is explicitly designed to create pixel level predictions as their output. Due to the fact that the objective of PredNet is to approximate the output representations as close as possible to the future input, the output representations of PredNet are necessarily forced to retain the same level of representation as the input, without encoding the input to more condensed representations, as it would usually happen in image recognition DNN (including typical CNN architectures). This means that PredNet is limited to operating on image-like data and performing semantic predictions at the sensory input level, rather than learning more abstract, encoded representations.

To solve this problem and to make the predictive objective more generally applicable to a wider range of neural network architectures, we implemented an alternative version of the predictive training approach. This implementation addresses the previously noted limitations of PredNet by introducing a novel training approach that enables the prediction of future states within simpler RNN architectures. In our implementation, the loss function consisted of a mean squared error (MSE) loss between the current "prior" (the previous hidden state multiplied by the recurrent kernel and projected back through the inverse input kernel) and the actual input stimulus one step into the future. This change allows us to encode representations into an arbitrary set of neurons, rather than maintaining the representations in the input space like PredNet does. Further, this loss is adaptable to various DNN model architectures, allowing us to implement the predictive objective in the more comparable setting of a simple RNN architecture.

By using this predictive, autoregressive objective, we encourage the neural network models to learn to generate predictions about upcoming inputs and update their internal representations in a way that they approximate the future input as close as possible given the current information. This training procedure aligns with the key principles of PC, where the brain is thought to constantly generate predictions about sensory inputs and update its models based on the mismatch between predictions and observations. Implementing this predictive objective in a more general way, rather than relying on specialized architectures, allowed us to explore the potential of this approach across a broader range of neural network models.

Contrastive: Another training procedure we investigated was a contrastive objective, inspired by the Forward-Forward algorithm proposed by Hinton [51]. The Forward-Forward algorithm works as a type of contrastive optimization loss, where an arbitrary network learns about stimuli by being able to separate valid (positive) data from invalid (negative) data. This happens on a layer-wise level, such that for example layer-wise activity is minimized for valid data, and maximized for invalid data. In this setting, the algorithm is comparable to the general concept of PC, where expected (valid) stimuli should evoke no activity, while unexpected (invalid) stimuli evoke maximum activity. In the original paper, a broad concept for a multi-hierarchical implementation of the algorithm is presented, however, as a preliminary investigation, the paper presents only working examples of more simple, feedforward context, where the network effectively performs contrastive objective, separating coherent input stimuli from incoherent ones.The Contrastive algorithm used in this paper closely follows the principle proposed by Hinton [51], applied to a standard RNN architecture, facilitating comparisons to other algorithms.

The training algorithm for this condition was derived from Hinton's original description. The loss function was defined to minimize a layer's activity when the next input image moves as expected (e.g., the numbers move according to predictable rules), and to maximize activity for stimuli that are unexpected (e.g., the numbers move according to non-predictable rules). In contrast to the predictive objective, this does not happen on a neuron-by-neuron level, where the resulting neuronal activity is expected to match activational patterns that predict the upcoming stimulus, but instead only considers the overall activity of the layer, which should be minimized for expected stimuli and maximized for unexpected stimuli. To implement this contrastive objective, the input data was split into positive samples (expected stimuli) and negative samples (unexpected stimuli). The positive stimuli consisted of normal number sequences that would make the MNIST numbers move as expected, while the negative stimuli included random semantic number switches, resulting in numbers that are not aligned with the rules according to which they should move. By implementing this contrastive objective, we aimed to encourage the neural network models to learn representations that are sensitive to expected versus unexpected stimuli, a key characteristic of PC. This training procedure provides an alternative approach to the predictive objective, allowing us to explore different ways of instilling PC-like principles into the models.

Supervised: As a first control condition, we used a standard supervised backpropagation approach. In this case, the target objective for the neural network models was to classify the image class of the current number shown in the input sequence. The model architecture for this condition is identical to the other models, with the only exception that, for this model, the final layer output is mapped onto a one-hot representation of the stimulus classes, allowing to train the network in a supervised fashion.

The use of supervised backpropagation provides an important baseline for comparison against the PC-inspired training procedures. Supervised learning, which relies on classical deep learning optimization rules like stochastic gradient descent and backpropagation, is known to be effective at training artificial neural networks to perform a wide range of tasks.

However, these supervised training approaches are often defined based on information-theoretical ideas rather than implementing biologically plausible objectives. By including a supervised backpropagation condition, we can assess how the performance and characteristics of the PC-inspired models compare to a standard, widely-used training method that is not explicitly designed to capture the computational principles of biological neural learning.

Untrained: As another baseline control condition, we used a model initialized of the same architecture as all the other models, but without undergoing any training process. Comparing the results of the trained models to this untrained random network baseline will help us determine the specific contributions of the different training objectives, ensuring that any observed signatures of PC are indeed a result of the learning process, rather than simply inherent in the recurrent model architecture itself.

4.3.2. Localized optimization procedure. In backpropagation, the error gradients are propagated through a long nested chain of partial derivatives that are all directed towards optimizing the output of the final layer [16]. A common critique on the biological plausibility of backpropagation is the implausibility of conveying such learning gradients backwards through a high number of biological neural layers [3,17,21]. In contrast to that, learning signals in the brain are usually transmitted using chemical or electrical signals [101–104], which are too diffuse and imprecise to transmit exact gradients over long hierarchies of layers. While it is theoretically possible to send gradient-like learning signals through a long hierarchy of neural layers [21], this becomes more unlikely, the longer the hierarchy becomes [3,17–21]. Accordingly, the biological brain is widely believed to use local learning gradients [7,17,18]. Importantly, both the predictive and the contrastive training conditions can be implemented in a localized, layer-by-layer fashion rather than using an end-to-end backpropagation approach throughout the network. For the contrastive condition, this localized training is possible because of the use of positive versus negative samples. The contrast in firing patterns can be learned at every level of the hierarchy, allowing a training procedure that first creates maximum contrast in output activity in the first layer, then the second layer, and so on. Similarly, the predictive objective can also be applied in a localized manner. By approximating the current "prior" (the previous hidden state projected back to the next lower level) to the next input state, this predictive loss can be computed at any given level of encoding. This allows for a training procedure that enables learning the first layer, then the second layer, and so on, without the need for full end-to-end backpropagation.

Accordingly, we introduced additional localized conditions: besides the Predictive/Contrastive global conditions, which predict/contrast only at the last layer and use backpropagation for propagating the gradients through the network, we additionally introduce a Predictive/Contrastive local condition, where gradients are only propagated within one layer. By exploring these localized training procedures, we aim to investigate whether PC-inspired objectives can be effectively implemented in a more brain-aligned fashion, without relying on the biologically implausible backpropagation algorithm.

4.3.3. Activity regularization. One key hyperparameter we investigated was the use of regularization. Specifically, we compared the results of the models trained with and without activity regularization. Activity regularization enforces sparsity in the layer activations, which provides an interesting analogy to the energy-saving principle behind PC [58,105].The rationale for exploring activity regularization is that it can serve as an approximation to one of the underlying assumptions of PC, namely that the brain is thought to minimize the energy required for information and only propagate the residual, or "prediction error" information [106]. Similarly, activity regularization encourages the neural network models to learn sparse, efficient representations which minimize the activity required to represent stimuli.

By comparing the performance of the models trained with and without activity regularization, we aimed to gain insights into the role of this type of regularization as a proxy for the computational mechanisms underlying PC. This analysis

allowed us to explore whether the incorporation of this energy-saving principle, even in a simplified form, can lead to neural network models that better exhibit the hallmarks of predictive processing in the brain.

4.4. Model evaluation

We selected three specific evaluation metrics - mismatch responses, prior expectations, and learned representations - to align with the key behaviors we would expect from a neural network model exhibiting the principles of PC. In the following, we describe how each of these metrics are quantified.

4.4.1. Mismatch responses. In PC, the occurrence of MMR is directly tied to deviations from what was expected, resulting in elevated activity [71]. In neuroscientific literature, mismatch responses are a widely observed phenomenon, occurring in various modalities [61,107,108]. We expect similar MMR patterns in PC-inspired networks. To assess the models' ability to detect and respond to unexpected or deviant stimuli, we evaluated their MMR (measured by deviations in the evoked neural response) to a series of matching and mismatching input sequences (see Fig 2).

The matching series condition consisted of a sequence of MNIST images with random shifts and rotations, but without any semantic changes to the digit content. In contrast, the mismatching series condition included shifts, rotations, and random switches in the semantic content of the images (e.g., a 3 suddenly changing to a 7). We ensured that the augmentation between match and mismatch condition did not show any base level differences that might result in a difference in mean-squared activation from the network's layers. We performed a Kolmogorov-Smirnov ($D = 0.000$, $p = 1.000$) on the pixel-level distribution of the two image conditions, to make sure that the core statistics of the matching group does not differ significantly from the mismatching group. We then measured the mean square layer activations of the networks over time, approximating evoked responses for the matching and mismatching conditions. Our metric of interest was whether the model output produced significant deviations between the match and the mismatch condition for the input immediately following the stimulus switch. This deviation was compared using independent sample t-tests, corrected for multiple comparisons across models using the Bonferroni correction. By analyzing the mismatch responses of the neural network models, we aimed to assess their ability to detect and respond to prediction errors, a key characteristic of PC. The comparison of the matching and mismatching conditions allowed us to isolate the models' sensitivity to unexpected changes in the input stimuli. As a metric for inference we evaluate the difference between the matching and the mismatching condition, immediately following a stimulus switch. We performed undirected independent t-tests with significance levels corrected for each time step and each condition using a Bonferroni correction.

4.4.2. Prior expectations. Prior predictions are a necessary precondition for a PC system, allowing to match up the prior prediction with incoming information, to save energy by only propagating relevant information [62–68]. Priors in the DNN context mean that there exists a representation of what input values the network expects to come up in the next step, which is then matched up and integrated with the actual incoming stimulus. If a neural network works according to PC principles, these prior states should start to approximate the input states over the course of learning. This would mean that the neural network forms expectations which not only convey information to the next state, but explicitly model what will happen in the next state (in order to implement expectation-related suppression of predicted stimuli [47,109]).

To measure to what degree the neural network models formed an inherent prior representation, we evaluated the similarities between the neural network's prior state (negative latent state times recurrent kernel) and the expected future input. We assessed the prior formation by projecting the prior state back through the network to the input level. This projected prior was then correlated with the future input to express the similarity between the prior and the future incoming input. As a second confirmatory analysis, we also performed this analysis at the encoded level by propagating the future inputs through the network in a forward direction and correlating the prior state with the encoded future state. We perform this additional analysis to make sure that this effect does not only occur at the stimulus level, which is more directly linked to the objective of the predictive algorithm. This confirmatory analysis is shown in Section C1 in S1 Appendix (with tables

in Section D in S1 Appendix). Since the actual PC operation subtracts the prior from the input, while our simple RNN models add the prior to the input, we expected the "prior" to be the negative latent state multiplied by the recurrent kernel. This transformation allows the PC subtraction to be implemented in a standard RNN architecture.

For statistical inference, we applied a Fisher-Z transformation to the correlation coefficients, inferred standard errors from the coefficients as described in [110], followed by Z-tests to compare all possible combinations of models with each other. We use Z-tests for this analysis to match the previously applied Fisher-Z transform. Confidence levels were corrected for all 36 possible between-model comparisons, using a Bonferroni correction.

4.4.3. Learned representations. Any learning system should exhibit the capability to learn and form abstract representations [35,111,112]. Specifically, we are interested in whether the PC-inspired algorithms condense meaningful semantic information about the category of objects shown (in our case: numbers), without explicitly being trained to do so. This would demonstrate the network's capacity to learn without supervision - a hallmark of biological neural learning systems such as PC [113–115].

To investigate learning performance of the networks, we examined the encoded information content in each of the models after being shown a single image. Each image was randomly augmented before feeding it to the network. Then, the encoded information was determined by using the latent output state of the network to decode the original number classes represented by the input. While the supervised condition has been explicitly trained to perform this task, for the predictive and the contrastive algorithms, no stimulus-class specific information has been fed to the network. Any better-than-untrained information in the network is therefore a result of the unsupervised learning procedures that these algorithms employ.

To encode information from the output state of the network, we used a simple logistic regression that classifies the original stimulus class based on the output features. From this, we calculated the overall decoding accuracy for all the stimulus classes for each model. For statistical inference, we calculated standard errors from the prediction accuracy using [116], followed by independent-sample T-tests. We performed these T-tests between all possible combinations of each of the 6 different models. Confidence levels were corrected for all 36 possible between-model comparisons, using a Bonferroni correction.

All the above tests were performed on the 6,000 test samples that were not used for training the models.

4.5. Analysis of gain control

We additionally investigated whether weight regularization can be used to simulate gain control. In PC, the noise level of the output signal is correlated with the noise level of the input. Gain control is the ability to manipulate the output noise level independently of the input noise [46,47]. Investigating gain control in our PC-inspired DNNs can provide insights into corresponding brain mechanisms. Weight regularization in neural networks is a widely known and used mechanic that constrains the magnitude of a network's weights [117]. From preliminary investigations, we expected this to be a suitable mechanism to implement gain control, as weight regularization reduces the network's sensitivity to variance in the input, potentially normalizing the statistical attributes of the output.

For this analysis, we trained all of the algorithms according to the standard setting without activity regularization, but with weight regularization. We compared the unregularized networks to the weight regularization networks. For weight regularized and unregularized networks of all algorithms, we passed single stimulus images while adding high (noise-to-signal standard deviation ratio ~1.7) or low (noise-to-signal standard deviation ratio ~0.3) Gaussian noise to the networks and collected the output activity. In order to create a distribution of variance ratios, we calculated the variances of the output activity and divided the variance of the high noise input images through the variances of the low noise input images for each sample. To evaluate whether the ratio of variance for both high and low noise images is significantly smaller for weight regularized networks than for unregularized networks (indicating gain control), we performed a Wilcoxon signed-rank test. We chose to use a nonparametric test for this analysis as the distribution of variance ratios was skewed and

the requirement of normality was not given. Further we investigated whether despite weight regularization - there are still differences in variance distributions between low and high noise stimuli. We did this by evaluating if the variance distribution in the weight regularized conditions are indistinguishable between low noise and high noise stimuli. For this, we used another Wilcoxon signed-rank test. Significance levels were corrected for 6 multiple comparisons (2 tests with 3 comparisons) using a Bonferroni correction.

4.6. Other computational settings

This study was implemented in Python, using tensorflow [118], keras [119], scikit-learn [120], and scipy [121]. Other details on hyperparameters and software used are mentioned in Section B in S1 Appendix.

Supporting information

S1 Appendix. A. Generalization to real video data. B. Technical details. B1. Network architecture. B2. Network and training hyperparameters. B3. Computational Environment. C. Figures. C1. Prior correlation at encoding level. C2. Predictive Condition MMR spread over layers. D. Tables. D1. No Regularization. D2. Activity Regularization. D3. Bair robot pushing dataset generalization. D4. Gain control.
(PDF)

Acknowledgments

We thank Mahdi Enan (Maastricht University) and Rasmus Bruckner (Hamburg University/Center for Cognitive Neuroscience Berlin) for feedback on this work. Further, we thank Johannes Singer (Center for Cognitive Neuroscience Berlin) for proofreading.

Author contributions

Conceptualization: Dirk Gütlin, Ryszard Auksztulewicz.

Data curation: Dirk Gütlin.

Formal analysis: Dirk Gütlin.

Funding acquisition: Ryszard Auksztulewicz.

Investigation: Dirk Gütlin.

Methodology: Dirk Gütlin, Ryszard Auksztulewicz.

Project administration: Ryszard Auksztulewicz.

Software: Dirk Gütlin.

Supervision: Ryszard Auksztulewicz.

Validation: Dirk Gütlin.

Visualization: Dirk Gütlin.

Writing – original draft: Dirk Gütlin.

Writing – review & editing: Dirk Gütlin, Ryszard Auksztulewicz.

References

1. Schrimpf M, Kubilius J, Hong H, Majaj NJ, Rajalingham R, Issa EB, et al. Brain-Score: Which Artificial Neural Network for Object Recognition is most Brain-Like? Cold Spring Harbor Laboratory. 2018. https://doi.org/10.1101/407007

2. Richards BA, Lillicrap TP, Beaudoin P, Bengio Y, Bogacz R, Christensen A, et al. A deep learning framework for neuroscience. Nat Neurosci. 2019;22(11):1761–70. https://doi.org/10.1038/s41593-019-0520-2 PMID: 31659335
3. Stork. Is backpropagation biologically plausible? In: International Joint Conference on Neural Networks, 1989. 241–6 vol.2. https://doi.org/10.1109/ijcnn.1989.118705
4. Pulvermüller F, Tomasello R, Henningsen-Schomers MR, Wennekers T. Biological constraints on neural network models of cognitive function. Nat Rev Neurosci. 2021;22(8):488–502. https://doi.org/10.1038/s41583-021-00473-5 PMID: 34183826
5. van Albada SJ, Morales-Gregorio A, Dickscheid T, Goulas A, Bakker R, Bludau S, et al. Bringing Anatomical Information into Neuronal Network Models. Adv Exp Med Biol. 2022;1359:201–34. https://doi.org/10.1007/978-3-030-89439-9_9 PMID: 35471541
6. Kietzmann TC, McClure P, Kriegeskorte N. Deep Neural Networks in Computational Neuroscience. Cold Spring Harbor Laboratory. 2017. https://doi.org/10.1101/133504
7. Bengio Y, Lee DH, Bornschein J, Mesnard T, Lin Z. Towards biologically plausible deep learning. In: 2016. https://doi.org/10.48550/arXiv.1502.04156
8. Salvatori T, Mali A, Buckley CL, Lukasiewicz T, Rao RPN, Friston K. Brain-Inspired Computational Intelligence via Predictive Coding. 2023. https://doi.org/10.48550/arXiv.2308.07870
9. Raymond JL, Medina JF. Computational Principles of Supervised Learning in the Cerebellum. Annu Rev Neurosci. 2018;41:233–53. https://doi.org/10.1146/annurev-neuro-080317-061948 PMID: 29986160
10. Chandrashekar A, Granger R. Derivation of a novel efficient supervised learning algorithm from cortical-subcortical loops. Front Comput Neurosci. 2012;5:50. https://doi.org/10.3389/fncom.2011.00050 PMID: 22291632
11. Knudsen EI. Supervised learning in the brain. J Neurosci. 1994;14(7):3985–97. https://doi.org/10.1523/JNEUROSCI.14-07-03985.1994 PMID: 8027757
12. Anselmi F, Poggio T. Representation Learning in Sensory Cortex: a theory. Center for Brains, Minds and Machines (CBMM). 2014. https://dspace.mit.edu/handle/1721.1/100191
13. Matteucci G, Piasini E, Zoccolan D. Unsupervised learning of mid-level visual representations. Curr Opin Neurobiol. 2024;84:102834. https://doi.org/10.1016/j.conb.2023.102834 PMID: 38154417
14. Anselmi F, Leibo JZ, Rosasco L, Mutch J, Tacchetti A, Poggio T. Unsupervised learning of invariant representations. Theoretical Computer Science. 2016;633:112–21. https://doi.org/10.1016/j.tcs.2015.06.048
15. Maiello G. Unsupervised learning in biological brains. Nat Rev Psychol. 2023;2(4):201–201. https://doi.org/10.1038/s44159-023-00166-z
16. Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. Nature. 1986;323(6088):533–6. https://doi.org/10.1038/323533a0
17. Millidge B, Tschantz A, Seth AK, Buckley CL. Activation Relaxation: A Local Dynamical Approximation to Backpropagation in the Brain. 2020. https://doi.org/10.48550/arXiv.2009.05359
18. Mazzoni P, Andersen RA, Jordan MI. A more biologically plausible learning rule than backpropagation applied to a network model of cortical area 7a. Cereb Cortex. 1991;1(4):293–307. https://doi.org/10.1093/cercor/1.4.293 PMID: 1822737
19. Bohte SM, Kok JN, La Poutré H. Error-backpropagation in temporally encoded networks of spiking neurons. Neurocomputing. 2002;48(1–4):17–37. https://doi.org/10.1016/s0925-2312(01)00658-0
20. Sacramento J, Costa RP, Bengio Y, Senn W. Dendritic error backpropagation in deep cortical microcircuits. 2017. https://doi.org/10.48550/arXiv.1801.00062
21. Lillicrap TP, Santoro A, Marris L, Akerman CJ, Hinton G. Backpropagation and the brain. Nat Rev Neurosci. 2020;21(6):335–46. https://doi.org/10.1038/s41583-020-0277-3 PMID: 32303713
22. Doerig A, Sommers RP, Seeliger K, Richards B, Ismael J, Lindsay GW, et al. The neuroconnectionist research programme. Nat Rev Neurosci. 2023;24(7):431–50. https://doi.org/10.1038/s41583-023-00705-w PMID: 37253949
23. Cichy RM, Khosla A, Pantazis D, Torralba A, Oliva A. Comparison of deep neural networks to spatio-temporal cortical dynamics of human visual object recognition reveals hierarchical correspondence. Sci Rep. 2016;6:27755. https://doi.org/10.1038/srep27755 PMID: 27282108
24. Sexton NJ, Love BC. Reassessing hierarchical correspondences between brain and deep networks through direct interface. Sci Adv. 2022;8(28):eabm2219. https://doi.org/10.1126/sciadv.abm2219 PMID: 35857493
25. Masse NY, Grant GD, Freedman DJ. Alleviating catastrophic forgetting using context-dependent gating and synaptic stabilization. Proc Natl Acad Sci U S A. 2018;115(44):E10467–75. https://doi.org/10.1073/pnas.1803839115 PMID: 30315147
26. Kietzmann TC, Spoerer CJ, Sörensen LKA, Cichy RM, Hauk O, Kriegeskorte N. Recurrence is required to capture the representational dynamics of the human visual system. Proc Natl Acad Sci U S A. 2019;116(43):21854–63. https://doi.org/10.1073/pnas.1905544116 PMID: 31591217
27. Rajaei K, Mohsenzadeh Y, Ebrahimpour R, Khaligh-Razavi S-M. Beyond core object recognition: Recurrent processes account for object recognition under occlusion. PLoS Comput Biol. 2019;15(5):e1007001. https://doi.org/10.1371/journal.pcbi.1007001 PMID: 31091234
28. Tang H, Schrimpf M, Lotter W, Moerman C, Paredes A, Ortega Caro J, et al. Recurrent computations for visual pattern completion. Proc Natl Acad Sci U S A. 2018;115(35):8835–40. https://doi.org/10.1073/pnas.1719397115 PMID: 30104363

29. Spoerer CJ, Kietzmann TC, Mehrer J, Charest I, Kriegeskorte N. Recurrent neural networks can explain flexible trading of speed and accuracy in biological vision. PLoS Comput Biol. 2020;16(10):e1008215. https://doi.org/10.1371/journal.pcbi.1008215 PMID: 33006992
30. Conwell C, Prince JS, Kay KN, Alvarez GA, Konkle T. What can 1.8 billion regressions tell us about the pressures shaping high-level visual representation in brains and machines? Cold Spring Harbor Laboratory. 2022. https://doi.org/10.1101/2022.03.28.485868
31. Chen T, Kornblith S, Norouzi M, Hinton G. A simple framework for contrastive learning of visual representations. 2020. https://doi.org/10.48550/arXiv.2002.05709
32. Prince JS, Alvarez GA, Konkle T. Contrastive learning explains the emergence and function of visual category-selective regions. Sci Adv. 2024;10(39):eadl1776. https://doi.org/10.1126/sciadv.adl1776 PMID: 39321304
33. Fernandez JG, Hortal E, Mehrkanoon S. Towards biologically plausible learning in neural networks. In: 2021 IEEE Symposium Series on Computational Intelligence (SSCI), 2021. 01–8. https://doi.org/10.1109/ssci50451.2021.9659539
34. Gupta M, Ambikapathi A, Ramasamy S. HebbNet: A Simplified Hebbian Learning Framework to do Biologically Plausible Learning. In: ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021. 3115–9. https://doi.org/10.1109/icassp39728.2021.9414241
35. Hebb DO. The Organization of Behavior. Psychology Press. 2005. https://doi.org/10.4324/9781410612403
36. Neftci EO, Averbeck BB. Reinforcement learning in artificial and biological systems. Nat Mach Intell. 2019;1(3):133–43. https://doi.org/10.1038/s42256-019-0025-4
37. Song HF, Yang GR, Wang X-J. Reward-based training of recurrent neural networks for cognitive and value-based tasks. Elife. 2017;6:e21492. https://doi.org/10.7554/eLife.21492 PMID: 28084991
38. Gershman SJ, Ölveczky BP. The neurobiology of deep reinforcement learning. Curr Biol. 2020;30(11):R629–32. https://doi.org/10.1016/j.cub.2020.04.021 PMID: 32516607
39. Botvinick M, Wang JX, Dabney W, Miller KJ, Kurth-Nelson Z. Deep Reinforcement Learning and Its Neuroscientific Implications. Neuron. 2020;107(4):603–16. https://doi.org/10.1016/j.neuron.2020.06.014 PMID: 32663439
40. Friston K. The history of the future of the Bayesian brain. Neuroimage. 2012;62(2):1230–3. https://doi.org/10.1016/j.neuroimage.2011.10.004 PMID: 22023743
41. Knill DC, Pouget A. The Bayesian brain: the role of uncertainty in neural coding and computation. Trends Neurosci. 2004;27(12):712–9. https://doi.org/10.1016/j.tins.2004.10.007 PMID: 15541511
42. Friston K. The free-energy principle: a unified brain theory? Nat Rev Neurosci. 2010;11(2):127–38. https://doi.org/10.1038/nrn2787 PMID: 20068583
43. Friston K, Kilner J, Harrison L. A free energy principle for the brain. J Physiol Paris. 2006;100(1–3):70–87. https://doi.org/10.1016/j.jphysparis.2006.10.001 PMID: 17097864
44. Rao RP, Ballard DH. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. Nat Neurosci. 1999;2(1):79–87. https://doi.org/10.1038/4580 PMID: 10195184
45. Friston K, Kiebel S. Predictive coding under the free-energy principle. Philos Trans R Soc Lond B Biol Sci. 2009;364(1521):1211–21. https://doi.org/10.1098/rstb.2008.0300 PMID: 19528002
46. Shipp S. Neural Elements for Predictive Coding. Front Psychol. 2016;7:1792. https://doi.org/10.3389/fpsyg.2016.01792 PMID: 27917138
47. Auksztulewicz R, Friston K. Repetition suppression and its contextual determinants in predictive coding. Cortex. 2016;80:125–40. https://doi.org/10.1016/j.cortex.2015.11.024 PMID: 26861557
48. Millidge B, Seth A, Buckley CL. Predictive Coding: a Theoretical and Experimental Review. 2022. https://doi.org/10.48550/arXiv.2107.12979
49. Hodson R, Mehta M, Smith R. The empirical status of predictive coding and active inference. Neurosci Biobehav Rev. 2024;157:105473. https://doi.org/10.1016/j.neubiorev.2023.105473 PMID: 38030100
50. Lotter W, Kreiman G, Cox D. A neural network trained for prediction mimics diverse features of biological neurons and perception. Nat Mach Intell. 2020;2(4):210–9. https://doi.org/10.1038/s42256-020-0170-9 PMID: 34291193
51. Hinton G. The Forward-Forward Algorithm: Some Preliminary Investigations. arXiv. 2022. https://doi.org/10.48550/arXiv.2212.13345
52. Millidge B, Tang M, Osanlouy M, Harper NS, Bogacz R. Predictive Coding Networks for Temporal Prediction. Cold Spring Harbor Laboratory. 2023. https://doi.org/10.1101/2023.05.15.540906
53. Whittington JCR, Bogacz R. An Approximation of the Error Backpropagation Algorithm in a Predictive Coding Network with Local Hebbian Synaptic Plasticity. Neural Comput. 2017;29(5):1229–62. https://doi.org/10.1162/NECO_a_00949 PMID: 28333583
54. Zhuang C, Yan S, Nayebi A, Schrimpf M, Frank MC, DiCarlo JJ, et al. Unsupervised neural network models of the ventral visual stream. Proc Natl Acad Sci U S A. 2021;118(3):e2014196118. https://doi.org/10.1073/pnas.2014196118 PMID: 33431673
55. Lee J, Jo J, Lee B, Lee J-H, Yoon S. Brain-inspired Predictive Coding Improves the Performance of Machine Challenging Tasks. Front Comput Neurosci. 2022;16:1062678. https://doi.org/10.3389/fncom.2022.1062678 PMID: 36465966
56. Salvatori T, Mali A, Buckley CL, Lukasiewicz T, Rao RPN, Friston K. A Survey on Brain-Inspired Deep Learning via Predictive Coding. 2025. https://doi.org/10.48550/arXiv.2308.07870

57. Lotter W, Kreiman G, Cox D. Deep Predictive Coding Networks for Video Prediction and Unsupervised Learning. 2017. https://doi.org/10.48550/arXiv.1605.08104
58. Ali A, Ahmad N, de Groot E, Johannes van Gerven MA, Kietzmann TC. Predictive coding is a consequence of energy efficiency in recurrent neural networks. Patterns (N Y). 2022;3(12):100639. https://doi.org/10.1016/j.patter.2022.100639 PMID: 36569556
59. Garrido MI, Kilner JM, Stephan KE, Friston KJ. The mismatch negativity: a review of underlying mechanisms. Clin Neurophysiol. 2009;120(3):453–63. https://doi.org/10.1016/j.clinph.2008.11.029 PMID: 19181570
60. Wacongne C, Changeux J-P, Dehaene S. A neuronal model of predictive coding accounting for the mismatch negativity. J Neurosci. 2012;32(11):3665–78. https://doi.org/10.1523/JNEUROSCI.5003-11.2012 PMID: 22423089
61. Stefanics G, Stephan KE, Heinzle J. Feature-specific prediction errors for visual mismatch. Neuroimage. 2019;196:142–51. https://doi.org/10.1016/j.neuroimage.2019.04.020 PMID: 30978499
62. Perrinet L. From the Retina to Action: Dynamics of Predictive Processing in the Visual System. The Philosophy and Science of Predictive Processing. Bloomsbury Publishing Plc. 2020. https://doi.org/10.5040/9781350099784.ch-005
63. Williams MA, Baker CI, Op de Beeck HP, Shim WM, Dang S, Triantafyllou C, et al. Feedback of visual object information to foveal retinotopic cortex. Nat Neurosci. 2008;11(12):1439–45. https://doi.org/10.1038/nn.2218 PMID: 18978780
64. Clark A. Expecting the world: perception, prediction, and the origins of human knowledge. J Philos. 2013;110:469–96.
65. O'Callaghan C, Kveraga K, Shine JM, Adams RB Jr, Bar M. Predictions penetrate perception: Converging insights from brain, behaviour and disorder. Conscious Cogn. 2017;47:63–74. https://doi.org/10.1016/j.concog.2016.05.003 PMID: 27222169
66. Summerfield C, Egner T. Expectation (and attention) in visual cognition. Trends Cogn Sci. 2009;13(9):403–9. https://doi.org/10.1016/j.tics.2009.06.003 PMID: 19716752
67. Herwig A, Weiß K, Schneider WX. Feature prediction across eye movements is location specific and based on retinotopic coordinates. J Vis. 2018;18(8):13. https://doi.org/10.1167/18.8.13 PMID: 30372762
68. Carvalho W, Tomov MS, de Cothi W, Barry C, Gershman SJ. Predictive representations: building blocks of intelligence. 2024. https://doi.org/10.48550/arXiv.2402.06590
69. LeCun Y, Cortes C, Burges CJC. The MNIST Database of Handwritten Images. https://yann.lecun.com/exdb/mnist/. 1998. Accessed 2025 January 14.
70. Finn C, Goodfellow I, Levine S. Unsupervised learning for physical interaction through video prediction. 2016. https://doi.org/10.48550/arXiv.1605.07157
71. Näätänen R, Alho K. Mismatch negativity--a unique measure of sensory processing in audition. Int J Neurosci. 1995;80(1–4):317–37. https://doi.org/10.3109/00207459508986107 PMID: 7775056
72. Pascanu R, Mikolov T, Bengio Y. On the difficulty of training Recurrent Neural Networks. 2013. https://doi.org/10.48550/arXiv.1211.5063
73. Krueger D, Memisevic R. Regularizing RNNs by Stabilizing Activations. In: 2016. https://doi.org/10.48550/arXiv.1511.08400
74. Schwartz EL, Christman DR, Wolf AP. Human primary visual cortex topography imaged via positron tomography. Brain Res. 1984;294(2):225–30. https://doi.org/10.1016/0006-8993(84)91033-3 PMID: 6608398
75. Hubel DH, Wiesel TN. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. J Physiol. 1962;160(1):106–54. https://doi.org/10.1113/jphysiol.1962.sp006837 PMID: 14449617
76. Lauter JL, Herscovitch P, Formby C, Raichle ME. Tonotopic organization in human auditory cortex revealed by positron emission tomography. Hear Res. 1985;20(3):199–205. https://doi.org/10.1016/0378-5955(85)90024-3 PMID: 3878839
77. Romani GL, Williamson SJ, Kaufman L. Tonotopic organization of the human auditory cortex. Science. 1982;216(4552):1339–40. https://doi.org/10.1126/science.7079770 PMID: 7079770
78. Penfield W, Boldrey E. Somatic motor and sensory representation in the cerebral cortex of man as studied by electrical stimulation. Brain. 1937;60(4):389–443. https://doi.org/10.1093/brain/60.4.389
79. Braun C, Heinz U, Schweizer R, Wiech K, Birbaumer N, Topka H. Dynamic organization of the somatosensory cortex induced by motor activity. Brain. 2001;124(Pt 11):2259–67. https://doi.org/10.1093/brain/124.11.2259 PMID: 11673326
80. Friston K. Does predictive coding have a future? Nat Neurosci. 2018;21(8):1019–21. https://doi.org/10.1038/s41593-018-0200-7 PMID: 30038278
81. Millidge B, Salvatori T, Song Y, Bogacz R, Lukasiewicz T. Predictive Coding: Towards a Future of Deep Learning beyond Backpropagation? arXiv. 2022. https://doi.org/10.48550/arXiv.2202.09467
82. Spratling MW. A review of predictive coding algorithms. Brain Cogn. 2017;112:92–7. https://doi.org/10.1016/j.bandc.2015.11.003 PMID: 26809759
83. Pezzulo G, Parr T, Friston K. The evolution of brain architectures for predictive coding and active inference. Philos Trans R Soc Lond B Biol Sci. 2022;377(1844):20200531. https://doi.org/10.1098/rstb.2020.0531 PMID: 34957844
84. Rane RP, Szügyi E, Saxena V, Ofner A, Stober S. PredNet and Predictive Coding: A Critical Review. In: Proceedings of the 2020 International Conference on Multimedia Retrieval, 2020. 233–41. https://doi.org/10.1145/3372278.3390694
85. Salvatori T, Song Y, Hong Y, Sha L, Frieder S, Xu Z. Associative Memories via Predictive Coding. In: Advances in Neural Information Processing Systems, 2021. 3874–86. https://proceedings.neurips.cc/paper/2021/hash/1fb36c4ccf88f7e67ead155496f02338-Abstract.html

86. Brown HR, Friston KJ. Dynamic causal modelling of precision and synaptic gain in visual perception - an EEG study. Neuroimage. 2012;63(1):223–31. https://doi.org/10.1016/j.neuroimage.2012.06.044 PMID: 22750569
87. Geirhos R, Janssen DHJ, Schütt HH, Rauber J, Bethge M, Wichmann FA. Comparing deep neural networks against humans: object recognition when the signal gets weaker. 2018. https://doi.org/10.48550/arXiv.1706.06969
88. Jang H, Tong F. Improved modeling of human vision by incorporating robustness to blur in convolutional neural networks. Nat Commun. 2024;15(1):1989. https://doi.org/10.1038/s41467-024-45679-0 PMID: 38443349
89. Tscshantz A, Millidge B, Seth AK, Buckley CL. Hybrid predictive coding: Inferring, fast and slow. PLoS Comput Biol. 2023;19(8):e1011280. https://doi.org/10.1371/journal.pcbi.1011280 PMID: 37531366
90. Li TE, Tang M, Bogacz R. Predictive coding model can detect novelty on different levels of representation hierarchy. Cold Spring Harbor Laboratory. 2024. https://doi.org/10.1101/2024.06.10.597876
91. Jiang LP, Rao RPN. Dynamic predictive coding: A model of hierarchical sequence learning and prediction in the neocortex. PLoS Comput Biol. 2024;20(2):e1011801. https://doi.org/10.1371/journal.pcbi.1011801 PMID: 38330098
92. Oord A van den, Li Y, Vinyals O. Representation Learning with Contrastive Predictive Coding. In: 2019.
93. Geiger A, Lenz P, Stiller C, Urtasun R. Vision meets robotics: The KITTI dataset. The International Journal of Robotics Research. 2013;32(11):1231–7. https://doi.org/10.1177/0278364913491297
94. Cech T, Wegen O, Atzberger D, Richter R, Scheibel W, Döllner J. Standardness clouds meaning: A position regarding the informed usage of standard datasets. arXiv. 2025. https://doi.org/10.48550/arXiv.2406.13552
95. Caucheteux C, Gramfort A, King J-R. Evidence of a predictive coding hierarchy in the human brain listening to speech. Nat Hum Behav. 2023;7(3):430–41. https://doi.org/10.1038/s41562-022-01516-2 PMID: 36864133
96. Chen Y, Mancini M, Zhu X, Akata Z. Semi-Supervised and Unsupervised Deep Visual Learning: A Survey. 2022. https://doi.org/10.48550/arXiv.2208.11296
97. Meng Q, Qian H, Liu Y, Xu Y, Shen Z, Cui L. Unsupervised Representation Learning for Time Series: A Review. arXiv. 2023. https://doi.org/10.48550/arXiv.2308.01578
98. Qi G-J, Luo J. Small Data Challenges in Big Data Era: A Survey of Recent Progress on Unsupervised and Semi-Supervised Methods. 2021. https://doi.org/10.48550/arXiv.1903.11260
99. Shao F, Shen Z. How can artificial neural networks approximate the brain? Front Psychol. 2023;13. https://doi.org/10.3389/fpsyg.2022.970214
100. Shevlyakov AN, Berezin AA. Recognition of the MNIST dataset with defective rows. J Phys: Conf Ser. 2022;2182(1):012031. https://doi.org/10.1088/1742-6596/2182/1/012031
101. Bairy LK, Kumar S. Neurotransmitters and neuromodulators involved in learning and memory. Int J Basic Clin Pharmacol. 2019;8(12):2777. https://doi.org/10.18203/2319-2003.ijbcp20195296
102. Myhrer T. Neurotransmitter systems involved in learning and memory in the rat: a meta-analysis based on studies of four behavioral tasks. Brain Res Brain Res Rev. 2003;41(2–3):268–87. https://doi.org/10.1016/s0165-0173(02)00268-0 PMID: 12663083
103. Harley CW. Norepinephrine and dopamine as learning signals. Neural Plast. 2004;11(3–4):191–204. https://doi.org/10.1155/np.2004.191 PMID: 15656268
104. Kennedy MB. Synaptic Signaling in Learning and Memory. Cold Spring Harb Perspect Biol. 2013;8(2):a016824. https://doi.org/10.1101/cshperspect.a016824 PMID: 24379319
105. Lässig F, Aceituno PV, Sorbaro M, Grewe BF. Bio-inspired, task-free continual learning through activity regularization. Biol Cybern. 2023;117(4–5):345–61. https://doi.org/10.1007/s00422-023-00973-w PMID: 37589728
106. Auksztulewicz R, Garrido MI, Malmierca MS, Tavano A, Todd J, Winkler I. Editorial: Sensing the World Through Predictions and Errors. Front Hum Neurosci. 2022;16:899529. https://doi.org/10.3389/fnhum.2022.899529 PMID: 35529776
107. Müller D, Widmann A, Schröger E. Object-related regularities are processed automatically: evidence from the visual mismatch negativity. Front Hum Neurosci. 2013;7:259. https://doi.org/10.3389/fnhum.2013.00259 PMID: 23772212
108. Hesse PN, Schmitt C, Klingenhoefer S, Bremmer F. Preattentive Processing of Numerical Visual Information. Front Hum Neurosci. 2017;11:70. https://doi.org/10.3389/fnhum.2017.00070 PMID: 28261078
109. de-Wit L, Machilsen B, Putzeys T. Predictive coding and the neural response to predictable stimuli. J Neurosci. 2010;30(26):8702–3. https://doi.org/10.1523/JNEUROSCI.2248-10.2010 PMID: 20592191
110. Eid M. Statistik und Forschungsmethoden: mit Online-Materialien. 4., überarbeitete und erweiterte Auflage ed. Beltz. 2015.
111. Anderson JR. The Adaptive Character of Thought. Psychology Press. 2013. https://doi.org/10.4324/9780203771730
112. Marr D. Vision: A computational investigation into the human representation and processing of visual information. MIT Press. 2010.
113. Gershman SJ, Niv Y. Learning latent structure: carving nature at its joints. Curr Opin Neurobiol. 2010;20(2):251–6. https://doi.org/10.1016/j.conb.2010.02.008 PMID: 20227271
114. Tenenbaum JB, Kemp C, Griffiths TL, Goodman ND. How to grow a mind: statistics, structure, and abstraction. Science. 2011;331(6022):1279–85. https://doi.org/10.1126/science.1192788 PMID: 21393536

115. Hohwy J. The predictive mind. OUP Oxford. 2013.
116. Newcombe RG. Two-sided confidence intervals for the single proportion: comparison of seven methods. Statist Med. 1998;17(8):857–72. https://doi.org/10.1002/(sici)1097-0258(19980430)17:8<857::aid-sim777>3.0.co;2-e
117. Goodfellow I, Bengio Y, Courville A. Deep learning. MIT Press. 2016.
118. Developers T. TensorFlow. Zenodo. 2021. https://doi.org/10.5281/zenodo.4758419
119. Chollet F. Keras: The python deep learning library. Astrophys Source Code Libr. 2018. https://doi.org/ascl-1806
120. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine learning in Python. The Journal of Machine Learning Research. 2011;12:2825–30.
121. Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, et al. SciPy 1.0: fundamental algorithms for scientific computing in Python. Nat Methods. 2020;17(3):261–72. https://doi.org/10.1038/s41592-019-0686-2 PMID: 32015543