Explaining Robustness to Catastrophic Forgetting Through Incremental Concept Formation

Nicki Barari NICKI.BARARI@DREXEL.EDU
Drexel University, Philadelphia, PA 19104 USA

Edward Kim EK826@DREXEL.EDU
Drexel University, Philadelphia, PA 19104 USA

Christopher MacLellan CMACLELL@GATECH.EDU
Georgia Institute of Technology, Atlanta, GA 30332 USA

Abstract

Catastrophic forgetting remains a central challenge in continual learning, where models are required to integrate new knowledge over time without losing what they have previously learned. In prior work, we introduced Cobweb/4V, a hierarchical concept formation model that exhibited robustness to catastrophic forgetting in visual domains. Motivated by this robustness, we examine three hypotheses regarding the factors that contribute to such stability: (1) adaptive structural reorganization enhances knowledge retention, (2) sparse and selective updates reduce interference, and (3) information-theoretic learning based on sufficiency statistics provides advantages over gradient-based backpropagation. To test these hypotheses, we compare Cobweb/4V with neural baselines, including CobwebNN, a neural implementation of the Cobweb framework introduced in this work. Experiments on datasets of varying complexity (MNIST, Fashion-MNIST, MedMNIST, and CIFAR10) show that adaptive restructuring enhances learning plasticity, sparse updates help mitigate interference, and the information-theoretic learning process preserves prior knowledge without revisiting past data. Together, these findings provide insight into mechanisms that can mitigate catastrophic forgetting and highlight the potential of concept-based, information-theoretic approaches for building stable and adaptive continual learning systems.

1. Introduction

Recent advances in computer vision have been driven largely by deep neural networks, which now match or even surpass human performance on tasks such as image classification and object detection (He et al., 2016). Despite these successes, neural networks remain limited in their ability to learn continuously. When trained on tasks sequentially, they often suffer from *catastrophic forgetting* (McCloskey & Cohen, 1989), a phenomenon in which newly acquired knowledge overwrites previously learned information. To mitigate forgetting, networks typically require access to previous training data or surrogate memory mechanisms, which increases computational cost and highlights a persistent trade-off between efficiency and stability.

In contrast, human learning is characterized by the gradual accumulation of knowledge across domains, where new skills and concepts are integrated with minimal disruption to prior knowledge (Barnett & Ceci, 2002; Calvert et al., 2004). While humans do forget, this process is gradual and rarely catastrophic. Achieving similar stability in artificial systems is a central goal of continual learning research, which has produced methods ranging from replay buffers and parameter regularization to dynamic network architectures (French, 1999; Wang et al., 2023). These approaches have advanced the field but face challenges, such as high memory requirements, task-specific tuning, and limited scalability as the number of tasks grows.

In prior work, a novel framework called *Cobweb/4V* was introduced (Barari et al., 2024), extending the psychologically inspired Cobweb family of concept formation models (Fisher et al., 2014; Gennari et al., 1989). Without relying on replay buffers, regularization constraints, or parameter isolation, Cobweb/4V demonstrated empirical robustness to catastrophic forgetting in continual visual learning tasks. These findings suggested that its resilience may arise from fundamental differences in how it learns and organizes knowledge.

This paper builds on those results by investigating the mechanisms that enable Cobweb/4V to retain prior knowledge while learning new concepts. Three hypotheses are considered: first, that adaptive restructuring of the concept hierarchy contributes to stability; second, that sparse, selective updates reduce interference; and third, that the information-theoretic learning mechanism plays a central role in preventing forgetting. Through a series of controlled experiments across multiple datasets, we aim to disentangle these factors and provide a deeper understanding of why Cobweb/4V resists catastrophic forgetting.

2. Background

2.1 Continual Learning and Catastrophic Forgetting

Continual learning refers to the ability of a model to incrementally acquire new knowledge over time without erasing or overwriting previously learned information. This capability is central to building adaptive, generalizable, and data-efficient systems that more closely resemble human learning (McCloskey & Cohen, 1989). In artificial systems, continual learning offers key advantages: it improves adaptability to changing environments, reduces the need for retraining from scratch, and supports more efficient use of data over time. These benefits make it especially valuable across real-world applications and dynamic modeling that must respond to evolving data streams and maintain operational efficiency (Barari & Barari, 2025; Izadkhah et al., 2024, 2025).

Despite its promise, continual learning remains a difficult challenge due to *catastrophic forgetting*, a phenomenon where learning new information interferes with or overwrites previously acquired knowledge. This issue was first identified in early studies on sequential learning in neural networks (McCloskey & Cohen, 1989), and remains a central obstacle in developing flexible, adaptive learning systems. Catastrophic forgetting happens due to a trade-off between plasticity and stability. A model must be plastic enough to adapt to new data, yet stable enough to preserve what it has already learned. Many learning systems, especially those with fixed capacity, struggle to maintain this balance. As new knowledge is encoded, previously learned information may be lost due to interference. The issue of new knowledge overwriting prior patterns arises in many applied domains, such as those that must track evolving user interests or dynamic group preferences (Izadkhah & Rekabdar, 2023, 2024), in ways analogous to catastrophic forgetting. This challenge is not unique to neural networks; it arises in any learning system that must operate under capacity limits. Research in both biological and artificial domains has explored ways to mitigate this constraint, such as using hierarchical or distributed representations to reduce interference and preserve previously acquired knowledge (French, 1999; Parisi et al., 2019). Another possible contributor to forgetting in neural networks is the use of backpropagation, which adjusts all parameters of the model during training. While effective for performance on isolated tasks, these global updates tend to overwrite weights associated with earlier data, increasing the risk of interference (Goodfellow et al., 2013). In contrast, biological systems tend to rely on more localized updates. Mechanisms such as Hebbian learning and other neuro-inspired approaches emphasize gradual and selective modification of memory traces, offering a potential path forward for continual learning models (Miconi et al., 2018). Sparsity is another factor that help the system to retain knowledge. In the brain, only a small subset of neurons activates in response to a given stimulus, which helps limit overlap between representations and preserve past knowledge. Sparse activation has been shown to support memory retention by minimizing interference between tasks (Olshausen & Field, 1996; Barari & Kim, 2021). Similarly, artificial models that use sparse representations, activating only a few components per input, tend to exhibit greater resilience to forgetting (Masse et al., 2018).

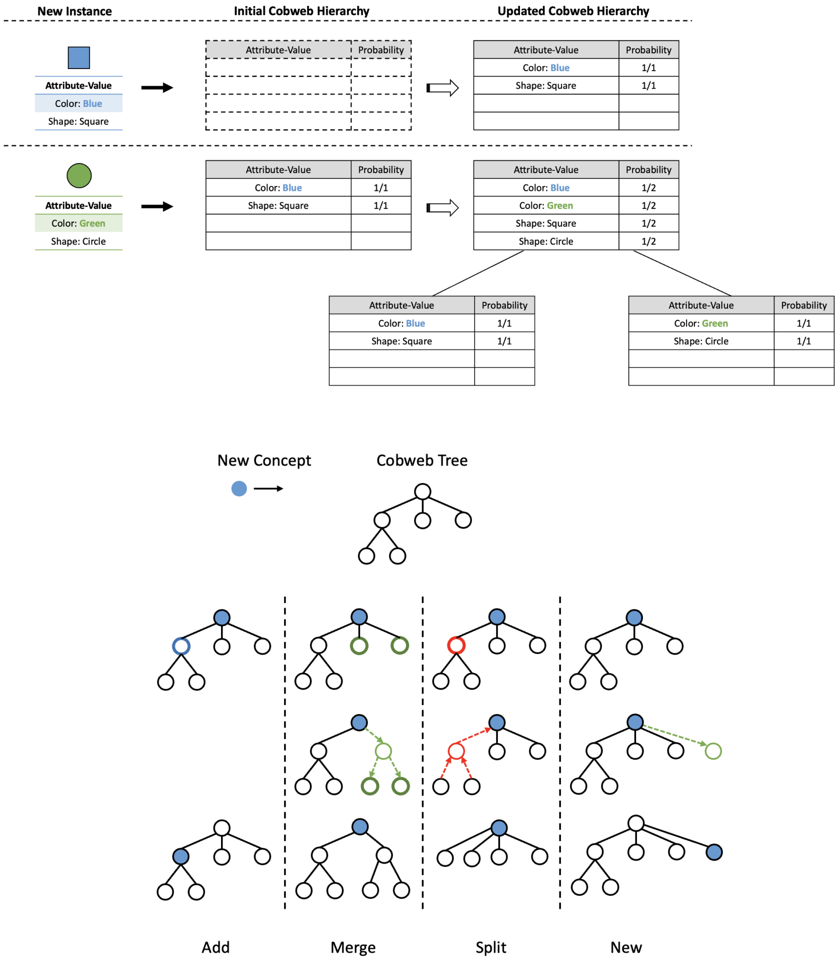
Over the years, a variety of strategies have been proposed to mitigate catastrophic forgetting, ranging from memory-based replay and regularization techniques to dynamic network expansion and sparsity constraints. Although most of these approaches have been developed in the context of deep learning, the challenge extends to a broader class of machine learning systems, including those designed to emulate human-like cognitive processes. Advancing our understanding of the limitations in structural capacity, learning strategies, and representational efficiency may therefore provide critical insights for building more robust and cognitively inspired models of continual learning.

In light of this, we previously introduced *Cobweb/4V* (Barari et al., 2024), a hierarchical concept formation approach inspired by psychological theories of human learning, that differs fundamentally from conventional deep learning approaches. Without relying on replay buffers, regularization constraints, or parameter isolation, Cobweb/4V demonstrated strong empirical robustness to catastrophic forgetting in continual visual learning tasks. To contextualize its performance, we benchmarked Cobweb/4V against a replay-based baseline, one of the most widely adopted strategies for mitigating forgetting, in order to assess its comparative effectiveness.

In the following sections, we provide a brief overview of the Cobweb framework and the key modifications introduced in Cobweb/4V that enable it to operate on high-dimensional image data. We then turn to the central goal of this work: identifying mechanisms that contribute to resilience to forgetting.

2.2 Cobweb - A Hierarchical Concept Formation Approach for Continual Learning

The Cobweb framework offers incremental and unsupervised learning from a continuous stream of examples (Fisher, 1987; Fisher et al., 2014). It processes incoming instances, and builds a hierarchical structure of concepts, drawing inspiration from human concept formation. Each instance is described as a collection of discrete attribute–value pairs, for example {color: blue; shape: square}.



Attribute-Value
Color: Blue
Shape: Square

Attribute-Value	Probability

Attribute-Value	Probability
Color: Blue	1/1
Shape: Square	1/1

Attribute-Value
Color: Green
Shape: Circle

Attribute-Value	Probability
Color: Blue	1/1
Shape: Square	1/1

Attribute-Value	Probability
Color: Blue	1/2
Color: Green	1/2
Shape: Square	1/2
Shape: Circle	1/2

Attribute-Value	Probability
Color: Blue	1/1
Shape: Square	1/1

Attribute-Value	Probability
Color: Green	1/1
Shape: Circle	1/1

Figure 1: Cobweb's Learning Process. (a) How a new instance is incorporated into the concept hierarchy. (b) The four operations Cobweb applies to update its structure during learning.

In the resulting Cobweb tree, each concept node maintains a table of attribute value probabilities that summarize the instances it has adopted.

Cobweb's learning mechanism involves classifying each new instance by recursively traversing the hierarchy. Along the selected path, the algorithm updates the count tables of the concept nodes to record the attribute values of the instance (Figure 1a). At each branching point, Cobweb considers four possible restructuring operations: *adding* the instance to the most appropriate child and updating its attribute-value counts, *merging* the two most similar children and then reevaluating the available options, *splitting* the most similar child and promoting its children to the current level, and *creating* a new child node that initially contains only the new instance (Figure 1b).

For prediction, Cobweb employs a process similar to its learning procedure, but without updating the concept counts along the path. When presented with a new instance, the algorithm begins at the root of the hierarchy and recursively sorts the instance through the tree. At each branching point, Cobweb decides whether to continue the descent into a child node or to halt at the current concept. Once the traversal ends, the count table of the final concept node is used to estimate the values of any unobserved attributes. In both its learning and prediction phases, Cobweb relies on a measure known as *category utility* (Corter & Gluck, 1992) to guide its decisions, selecting the operation that yields the highest value. Category utility quantifies the improvement in predictive power offered by a child node compared to its parent. Formally, the measure is defined as:

$$\frac{\sum_{k=1}^{n} P(C_k) \left[\sum_i \sum_j P(A_i = V_{ij}|C_k)^2 - P(A_i = V_{ij})^2\right]}{n} \tag{1}$$

Here, n denotes the number of child concepts, $P(C_k)$ is the overall probability of the kth child, $P(A_i = V_{ij}|C_k)$ represents the probability that attribute A_i takes value V_{ij} given child C_k, and $P(A_i = V_{ij})$ is the probability of A_i having value V_{ij} in the parent concept. Intuitively, the term $\sum_i \sum_j P(A_i = V_{ij}|C_k)^2$ captures the expected number of attributes correctly predicted within child C_k, while $\sum_i \sum_j P(A_i = V_{ij})^2$ corresponds to the expected number of correct predictions made at the parent level. The category utility score therefore measures the average improvement in predictive accuracy when moving from the parent to its children, with each child weighted by its probability $P(C_k)$. To allow comparison across cases with different branching factors, the score is normalized by dividing by n, the number of children. Although the original Cobweb algorithm supports only nominal attributes, *Cobweb/3* (McKusick & Thompson, 1990) extends the framework to continuous attributes. In this version, each concept models the probability density of continuous attributes using normal distributions, storing the mean and standard deviation for each attribute instead of maintaining nominal count tables.

2.3 Cobweb/4V - A Novel Version of Cobweb for Image Learning

Cobweb/4V (Barari et al., 2024) is an extension of the Cobweb framework designed to support continual learning in visual domains. This variant demonstrated two notable capabilities: it can learn effectively from limited data and shows strong robustness to catastrophic forgetting in sequential visual learning tasks. Building on the earlier *Trestle* implementation by MacLellan et al. (2016), this variant introduces several key updates, including an information-theoretic learning measure, a multi-concept prediction strategy, and a tensor-based representation that enables efficient processing of image data.

2.3.1 Key Updates

Learning with Mutual Information:
Earlier Cobweb studies used the *probability-theoretic* category utility (Corter & Gluck, 1992), which measures the expected increase in correct attribute predictions given concept membership. This formulation has been described as an unsupervised extension of the *Gini Index* commonly applied in decision tree construction (Fisher, 1996). Cobweb/4V instead employs an *information-*

theoretic category utility (Corter & Gluck, 1992), linking feature predictability with informativeness. The updated measure is defined as:

$$\frac{\sum_{k=1}^{n} P(C_k) \left[H(A = V) - H(A = V|C_k)\right]}{n} \tag{2}$$

where $H(A = V) = \sum_i \sum_j [-P(A_i = V_{ij}) \log(P(A_i = V_{ij})]$ is the entropy over all attribute values in the parent, and $H(A = V|C_k)$ is the entropy for child k. This unsupervised extension of information gain, closely related to mutual information, supports greater precision than the probability-theoretic variant. By expressing utility in terms of entropy, the approach naturally accommodates different attribute distributions, many of which have closed-form entropy expressions.

Predicting with a Combination of Concepts:
Traditional Cobweb prediction assigns an instance to a single subordinate-level concept and uses its counts to infer missing attributes (MacLellan et al., 2016; MacLellan & Thakur, 2022; MacLellan et al., 2022). Other studies have instead used predictions from alternative levels, such as the *basic-level* (Fisher & Langley, 1990; Corter & Gluck, 1992). Cobweb/4V introduces a broader approach that combines predictions from multiple concepts in the hierarchy. Given an instance x with unobserved features and a parameter N_{max} (the number of nodes to expand), the system performs a *best-first* search rather than a single-path greedy search. At each step, it expands the node c^* on the frontier (the set of candidate nodes awaiting exploration) with the highest score $s(c) = P(c|x)P(x|c)$, known as *collocation* (Jones, 1983), which is the product of cue and category validity. Expanded nodes are collected in $\mathcal{C}^*$, and prediction is made via a softmax-weighted combination of their contributions:

$$P(X_i = x_i|\mathcal{C}^*) = \sum_{c \in \mathcal{C}^*} P(x_i|c) \frac{\exp\{-s(c)\}}{\sum_{c \in \mathcal{C}^*} \exp\{-s(c)\}} \tag{3}$$

Although only N_{max} nodes are expanded, this procedure effectively performs a form of Bayesian model averaging (Hinne et al., 2020).

A New Tensor Representation:
In Cobweb/4V, instances are represented as tensors of pixel values paired with class labels, rather than lists of attribute-value pairs as in prior versions. For vision tasks, this design allows inputs to be structured as n-channel 2D images. Each node stores the mean and standard deviation of pixel features in a tensor, similar to Cobweb/3's treatment of continuous attributes, while also maintaining a probability table for class labels. Assuming conditional independence among attributes, the uncertainty of a node is computed as the sum of entropies across its attributes. Implemented with PyTorch, this tensor-based representation enables faster processing than earlier versions of Cobweb that use attribute-value lists.

2.3.2 *Resilient to Catastrophic Forgetting*

Cobweb/4V demonstrates robustness to catastrophic forgetting compared to neural network baselines. The first neural network baseline (fc) employs fully connected layers, and the second baseline

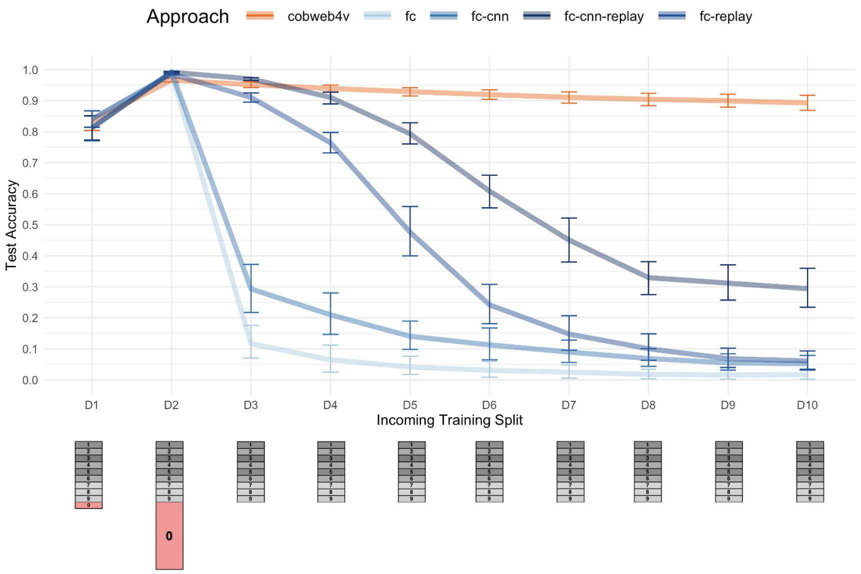


Figure 2: Average test accuracy on the chosen class images from the MNIST test set after each training split (D1–D10). D1 includes a balanced portion of all digits, containing 300 images each for digits 0-9. The second split, includes all remaining data for the chosen digit, along with an additional 300 images from each of the other non-chosen digits. The remaining data for the non-chosen digits are randomly divided across the remaining 8 splits. The color blocks under the x-axis represent the digit distribution in each split when the chosen digit label is 0.

(fc-cnn) incorporates additional convolutional neural network (CNN) layers. Figure 2 summarizes the results. Each approach was trained sequentially on ten data splits from MNIST dataset and evaluated only on a chosen class after each split, where the chosen class appeared only in the first two training splits and was absent from the rest. Neural network baselines without replay exhibited a rapid decline in accuracy, eventually approaching zero. With replay, the network was trained on both the current split and the examples stored in the replay buffer. After each split, 1,000 examples were randomly sampled from the union of the buffer and the split and carried forward for the next training iteration. Even under this replay scheme, the performance of the neural networks decayed steadily as training progressed. In contrast, Cobweb/4V maintained high accuracy across splits, with only a gradual decline due to feature interference. These findings indicate that Cobweb/4V preserves prior knowledge effectively, highlighting its resilience to catastrophic forgetting.

2.4 CobwebNN - A Neural Network Version of Cobweb

We introduce *CobwebNN*, a neural architecture inspired by, but distinct from neural taxonomic networks (Wang et al., 2025). Neural taxonomic nets organize concepts hierarchically, using gating functions for branching, classifiers for label prediction, and mechanisms such as temperature-controlled gating, stochastic exploration with Gumbel noise, and regularization for balanced splits.

These techniques make them effective for differentiable concept hierarchies. CobwebNN adapts this hierarchical idea but is designed to approximate the behavior of the Cobweb/4V framework. Unlike taxonomic nets that rely on gating and linear classifiers, CobwebNN represents concepts through reconstruction-based prototypes coupled with class distributions. This design enables controlled comparisons with Cobweb/4V to examine the role of structural non-gradient-based learning in mitigating catastrophic forgetting.

In CobwebNN, each input instance x is matched to concept nodes based on how well it aligns with their prototypes. To support this process, each node maintains three components: (1) Prototype for reconstruction: a vector μ_c that summarizes the typical input associated with a concept c. The match between an input x and this prototype is evaluated using a Gaussian likelihood with unit variance, $p(x \mid c) = \mathcal{N}(x; \mu_c, I)$. This likelihood both reflects reconstruction quality, since μ_c serves as the representative input for the concept, and provides a probabilistic measure of similarity, with higher values assigned to inputs closer to the prototype. The variance is fixed to one so that all concepts are compared on the same scale, ensuring that likelihoods depend only on how close an input is to a prototype. (2) Prior weight for baseline likelihood: a learnable bias term that specifies the baseline probability of selecting a child concept given its parent, denoted $p(c \mid c_{\text{parent}})$. This ensures that every branch retains some probability mass even before considering the input. Intuitively, it is the parent's "default preference" for its children, later modulated by the data. (3) Label information for classification: a set of learnable parameters that define $p(y \mid c)$, the probability of predicting each label when concept c is chosen.

Path selection is made differentiable through the Gumbel–Softmax trick (Jang et al., 2016), which approximates categorical sampling in a continuous and differentiable way. It adds Gumbel noise to the logits, followed by a temperature-controlled softmax. The temperature parameter τ adjusts how close the output is to a one-hot vector: lower τ yields sharper, more discrete selections, while higher τ produces smoother probabilities. This allows gradient flow through otherwise discrete branching. Formally, the path probability at layer L is defined recursively as:

$$p^L(c \mid x) \sim p(x \mid c) \cdot p^{L-1}(c_{\text{parent}} \mid x) \cdot p(c \mid c_{\text{parent}}) \tag{4}$$

with $p^0(\text{root} \mid x) = 1$. Here x is the input instance, y is the label, c is the current concept node, c_{parent} is the parent node of concept c, and C_L is the set of nodes in level L of the hierarchy.

Label prediction marginalizes over all leaves:

$$p(y \mid x) = \sum_{c \in \mathcal{C}_L} p^L(c \mid x) \cdot p(y \mid c) \tag{5}$$

where $\mathcal{C}_L$ denotes the set of concepts in the final level (leaves), and $p(y \mid c)$ is a softmax over learnable logits. Two inference modes are supported: 1. Sparse mode; a single leaf is sampled using Gumbel-Softmax, and prediction is drawn from its label distribution. 2. Dense mode; all leaves contribute, weighted by their path probabilities.

Training follows the same principle, except path probabilities incorporate both features and labels:

$$p^L(c \mid x, y) \sim p(x, y \mid c) \cdot p^{L-1}(c_{\text{parent}} \mid x, y) \cdot p(c \mid c_{\text{parent}}) \tag{6}$$

We factor the joint distribution as $p(x, y \mid c) = p(x \mid c) \cdot p(y \mid c)$, assuming conditional independence of the input features and labels given the concept. In this view, $p(x \mid c)$ captures how well the instance matches the concept's prototype, while $p(y \mid c)$ captures the label distribution associated with that concept. The objective combines reconstruction and classification, with loss:

$$\mathcal{L} = -\frac{1}{N} \sum_{n=1}^{N} \sum_{c \in \mathcal{C}_L} p^L(c \mid x_n, y_n) \cdot \Big(\log p(x_n \mid c) + \log p(y_n \mid c) \Big) \tag{7}$$

Thus, the loss updates each concept in proportion to both its ability to reconstruct the input and its ability to predict the correct label. In dense training, all nodes are updated proportionally to their path probabilities. In sparse training, only nodes along one sampled path are updated, mirroring inference and reducing interference from unrelated concepts.

The results presented in prior work showed that Cobweb/4V is able to learn visual concepts in a continual setting while exhibiting strong robustness to catastrophic forgetting, even when compared against competitive neural network baselines that incorporate replay strategies. While these findings established Cobweb/4V as a promising framework for continual learning, they also raised an important question: *what underlying mechanisms enable this resilience?* Addressing this question is essential not only for understanding the principles behind Cobweb/4V, but also for identifying features that could inform the design of future continual learning systems. Motivated by this goal, the next section introduces three hypotheses that seek to explain Cobweb/4V's robustness to forgetting, each grounded in distinct characteristics of the framework's structure and learning dynamics.

3. Forgetting Hypotheses

Although Cobweb/4V has demonstrated strong resilience to catastrophic forgetting in continual visual learning tasks, the reasons for this robustness remain unclear. To investigate, we propose three complementary hypotheses that each target a different potential contributing factor. The first examines the role of Cobweb's adaptive hierarchical structure, the second considers its use of sparse and localized updates during learning, and the third focuses on its information-theoretic learning mechanism as an alternative to gradient-based optimization. By testing these hypotheses across multiple datasets and controlled variations of the framework, we aim to identify the key properties that enable Cobweb/4V to mitigate catastrophic forgetting and assess their broader relevance to continual learning models.

3.1 Adaptive Structure

A well-known challenge in continual learning is the limited capacity of learning models. Systems with fixed or constrained structures often struggle to integrate new knowledge without overwriting previously learned information, as interference arises when old and new data compete for the same representational resources (French, 1999; Parisi et al., 2019). This trade-off between plasticity and stability is a central factor in catastrophic forgetting. Cobweb's adaptive structure provides a potential means of addressing this challenge. During learning, Cobweb dynamically reorganizes its hierarchy by creating, merging, or splitting nodes in response to new data. This allows the model to allocate representational capacity where it is most needed and to adjust its concept hierarchy as distributions shift. In principle, such structural flexibility could mitigate forgetting by reducing interference between old and new knowledge. Based on these observations, we hypothesize that Cobweb's adaptive structure plays an important role in its robustness to catastrophic forgetting. To evaluate this hypothesis, we design experiments that compare the original adaptive Cobweb to a fixed-structure variant, allowing us to isolate the effect of structural adaptivity on continual learning performance.

3.2 Sparse Updates

Another major contributor to catastrophic forgetting is how models update their internal representations when new data arrive. In neural networks, learning typically occurs through backpropagation, where all parameters are adjusted at once for each new batch. This dense updating can cause interference: changes made for new information may overwrite weights that were essential for earlier knowledge, leading to forgetting over time (Goodfellow et al., 2013). Cobweb, in contrast, relies on sparse and selective updates. When an input is categorized, only a single path, or a small localized part of the hierarchy, is updated, leaving unrelated knowledge intact. This localized adaptation reduces interference and mirrors the sparse activation patterns observed in biological learning, which help preserve memory by limiting overlap across tasks (Olshausen & Field, 1996; Masse et al., 2018). We therefore hypothesize that Cobweb's reliance on sparse, localized updates plays a key role in its robustness to catastrophic forgetting. To test this, we compare the standard sparse-update process to a dense-update alternative, where a broader portion of the hierarchy is modified during learning, allowing us to isolate the effect of update sparsity on interference.

3.3 Information-Theoretic Learning Mechanism

A further potential explanation for Cobweb's robustness to catastrophic forgetting lies in its learning mechanism. Most neural networks rely on backpropagation, which performs iterative gradient-based updates over mini batches of data. While effective for task-specific optimization, this process introduces a recency bias: parameter estimates are influenced more strongly by the most recent data, leading to a gradual disruption of information about earlier experiences (Goodfellow et al., 2013). Since old data are not revisited in continual learning settings, gradient descent updates often approximate a moving average rather than a true posterior, which increases the risk of forgetting. In contrast, Cobweb employs a closed-form, information-theoretic learning mechanism that leverages sufficiency statistics under the assumption of normal distributions. Each concept tracks the number of instances seen, as well as the mean and variance of their feature values. These statistics are sufficient in the statistical sense: they retain all the information the data provide about the distribution's parameters. As a result, Cobweb can update its concept representations incrementally with each new instance while maintaining unbiased estimates of the mean and variance across all data observed, without the need to revisit earlier examples. This approach effectively avoids the recency bias inherent in stochastic gradient descent, allowing the system to preserve prior knowledge more faithfully. Based on these observations, we hypothesize that Cobweb's information-theoretic learning mechanism, supported by the use of sufficiency statistics, plays a critical role in its robustness to catastrophic forgetting. To evaluate this hypothesis, we design experiments that directly compare Cobweb's closed-form updates with gradient-based optimization methods, allowing us to assess the extent to which the choice of learning mechanism contributes to its stability in continual learning.

4. Experiments

Our experiments aim to investigate the mechanisms underlying Cobweb/4V's robustness to catastrophic forgetting. To ensure the results are not tied to a single benchmark, we evaluate the model on a range of image datasets that differ in content and complexity. We introduce three main sets of experiments that each test one of the proposed hypotheses.

4.1 Datasets

To evaluate generalizability, we test on four widely used image datasets that differ in content and complexity: handwritten digits, clothing items, natural images, and medical imagery.

- MNIST (LeCun, 1998) contains 70,000 grayscale images of handwritten digits of size 28×28. It is a long-standing benchmark in continual learning due to its simplicity and balanced classes.
- Fashion-MNIST (Xiao et al., 2017) has the same format as MNIST but depicts clothing items (e.g., shirts, trousers, sneakers, bags). With 70,000 grayscale images at 28×28, it poses a more visually challenging task than handwritten digits.
- CIFAR-10 (Krizhevsky et al., 2009) consists of 60,000 color images of size 32×32 from ten classes of natural objects such as animals, vehicles, and household items. Its greater visual diversity makes it a stronger test of robustness compared to grayscale benchmarks.
- MedMNIST (OrganA subset) (Yang et al., 2023) provides multi-class abdominal organ images derived from medical scans. Designed as a lightweight medical benchmark, it introduces more realistic scenarios where resilience to forgetting is critical. The OrganA subset contains 58,850 grayscale images at 28×28 resolution, spanning eleven abdominal organ classes such as liver, spleen, kidney, and stomach.

Together, these datasets form a diverse testbed for assessing both the stability and adaptability of Cobweb/4V and its neural variants in continual learning.

4.2 Training Splits

Following the protocol from our previous work (Barari et al., 2024), we partition each dataset into ten splits (D1–D10). The first split (D1) contains a balanced sample of all classes, with 300 images per class. The second split (D2) consists of all remaining data for the chosen class, together with an additional 300 images from each of the other non-chosen classes. The remaining data from the non-chosen classes are then randomly and evenly distributed across the last eight splits (D3–D10).

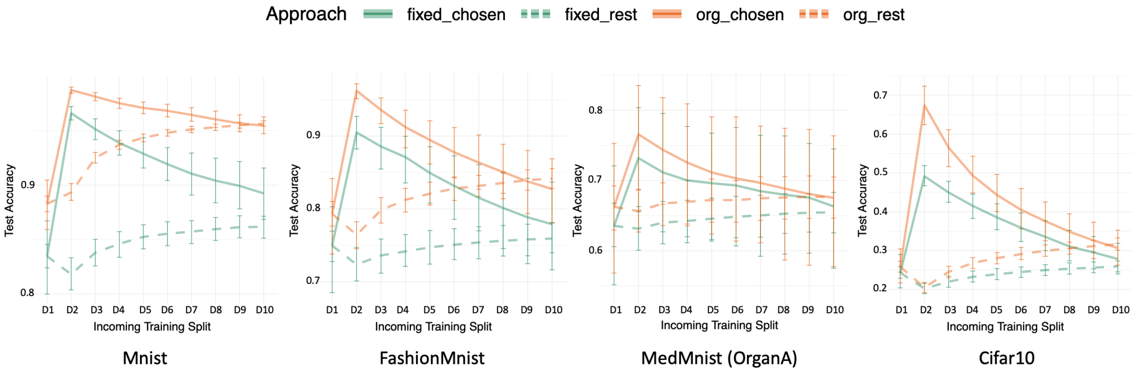


Figure 3: Average accuracy of (Fixed vs. Adaptive)-structure Cobweb/4V on Chosen and Non-chosen classes across datasets, after each training split (D1–D10). Solid lines represent accuracy on the chosen class; dashed lines represent average accuracy on non-chosen classes. *fixed_* refers to the fixed-structure Cobweb/4V and *org_* refers to the original Cobweb/4V with adaptive structure.

4.3 Experiment 1 - Adaptive vs. Fixed Structure

4.3.1 Method

This experiment evaluates the first hypothesis: that Cobweb's adaptive structure contributes to its robustness to catastrophic forgetting. Cobweb/4V normally employs dynamic restructuring operations, including creating, merging, and splitting nodes, which allow the hierarchy to expand and reorganize as new data are introduced. To test the role of structural adaptivity, we compare the standard adaptive version of Cobweb/4V with a fixed-structure variant. In the fixed version, the depth and branching factor of the tree are predetermined, and merge and split operations are disabled. This design removes the system's ability to dynamically reorganize its structure while maintaining all other aspects of the learning process. Both variants are trained under the continual learning protocol described earlier, using sequential data splits across multiple dataset. Performance is evaluated on both the chosen class and the non-chosen classes after each split, allowing us to assess differences in knowledge retention and the ability to incorporate new information.

4.3.2 Results

Figure 3 shows that fixing the structure consistently reduced accuracy compared to adaptive Cobweb/4V, underscoring the importance of reorganization for both stability and plasticity. Even so, the fixed-structure model maintained relatively stable performance across training. It did not exhibit sharp drops in accuracy, and forgetting remained gradual. Earlier knowledge was largely preserved, suggesting that factors beyond structural adaptivity also contribute to Cobweb/4V's robustness.

4.3.3 Discussion

These findings show that Cobweb/4V's adaptive structure strengthens both memory stability and learning plasticity. By allowing nodes to be created, merged, and split, the model integrates new concepts while preserving prior knowledge, leading to consistently higher performance than the fixed-structure variant on both chosen and non-chosen classes. At the same time, the fixed-structure model still demonstrated notable robustness to forgetting. Its accuracy declined gradually rather than collapsing sharply, suggesting that structural adaptivity is important but not the sole driver of Cobweb/4V's stability. Importantly, robustness cannot be explained simply by Cobweb's instance-based design. Even when leaves no longer corresponded to individual training examples, the model retained stable performance by relying on intermediate concepts rather than memorized exemplars. Taken together, these results support the view, consistent with cognitive science, that learning involves updating and reorganizing internal structures to accommodate new information while maintaining continuity of past knowledge.

4.4 Experiment 2 - Sparse vs. Dense Updates

4.4.1 Method

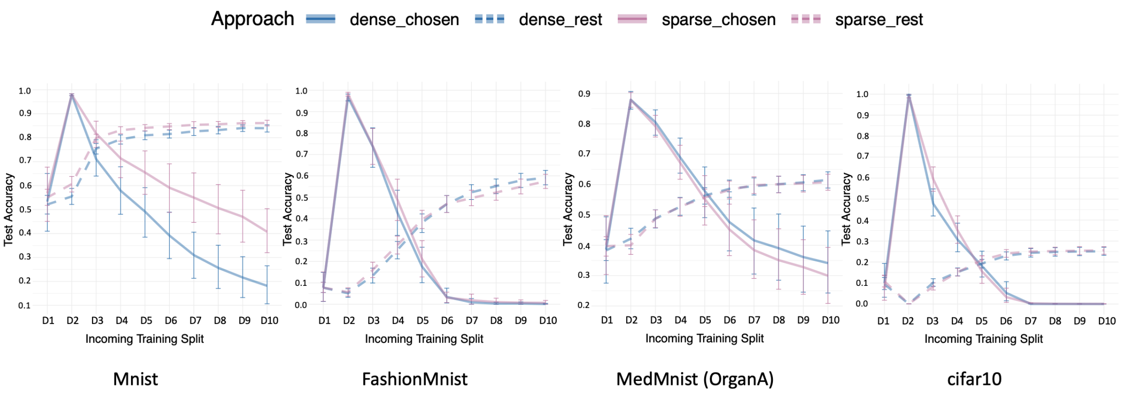
This experiment tests the hypothesis that Cobweb's resistance to catastrophic forgetting arises from its sparse and selective updates. In Cobweb/4V, a new instance updates only a single path or a small subset of nodes in the hierarchy, unlike neural networks where backpropagation adjusts all parameters at once. Such localized updates are expected to reduce interference between old and new knowledge. To evaluate this idea, we used CobwebNN, a neural architecture that mimics Cobweb while allowing explicit control over update sparsity via the Gumbel-Softmax trick (Jang et al., 2017; Wang et al., 2025). By adjusting the temperature parameter τ and sampling mode, CobwebNN can be run in either sparse-update mode (one path) or dense-update mode (multiple paths). Both variants were trained under the continual learning protocol described earlier. Performance was then measured on the chosen class to track forgetting, and on non-chosen classes to test generalization to new data across splits.

4.4.2 Results

Figure 4 compares sparse- and dense-update variants of CobwebNN on both chosen and non-chosen classes. Results show no substantial accuracy difference between the two modes. In both cases, chosen-class performance declined gradually across training splits, reflecting forgetting, while non-chosen class performance followed similar trends. These findings suggest that, in this implementation, update sparsity did not measurably affect either memory retention or learning of new information.

4.4.3 Discussion

This experiment tested whether sparse updates, as in Cobweb/4V, reduce interference and improve retention in a neural network setting. The results are inconclusive: no clear performance gap emerged between sparse and dense updates. While sparsity may provide benefits under certain conditions, in CobwebNN other factors, such as the learning mechanism and absence of hierarchical restructuring, likely dominate forgetting. Thus, we find no definitive evidence for the effect of sparsity on continual learning in neural networks. Further work that isolates sparsity from such confounding factors will be needed to clarify its role.



Mnist FashionMnist MedMnist (OrganA) cifar10

Figure 4: Average accuracy of (Sparse vs. Dense)-update configurations of CobwebNN on Chosen and Non-chosen classes across datasets. Each subplot shows the test accuracy after each training split. Solid lines represent accuracy on the chosen class; dashed lines represent average accuracy on non-chosen classes.

4.5 Experiment 3 - Information-Theoretic vs. Backpropagation

4.5.1 Method

The third experiment examines whether Cobweb/4V's resilience to catastrophic forgetting stems from its information-theoretic learning mechanism rather than the gradient-based optimization used in neural networks. To isolate this effect, we compare fixed-structure Cobweb/4V with the sparse-update variant of CobwebNN, ensuring both have similar structural sparsity. In Cobweb/4V, each concept node is modeled as a multivariate normal distribution with diagonal covariance. The model maintains sufficiency statistics (count, mean, variance), which allow incremental Bayesian updates without revisiting past data. For example, given N prior observations (the number of examples already seen for the concept) the mean and variance can be updated with a new input x_{new} as:

$$\mu_{\text{new}} = \mu_{\text{old}} + \frac{1}{N+1}(x_{\text{new}} - \mu_{\text{old}}) \tag{8}$$

$$\sigma^2_{\text{new}} = \sigma^2_{\text{old}} + \frac{1}{N+1}\big((x_{\text{new}} - \mu_{\text{old}})(x_{\text{new}} - \mu_{\text{new}}) - \sigma^2_{\text{old}}\big) \tag{9}$$

As data accumulate, the effect of each new input diminishes, reflecting the growing evidence base. This process yields unbiased posterior estimates equivalent to batch computation, while avoiding storage or replay of past examples. These updates are mathematically equivalent to a gradient-based update (Equation 10) with a learning rate of $1/(1+N)$. In contrast, sparse CobwebNN relies on gradient descent and backpropagation. Even with structural sparsity, its updates resemble an exponential moving average, where each input contributes a fixed proportion regardless of prior experience:

$$\theta_{t+1} = \theta_t - \alpha \nabla_\theta L(x_t). \tag{10}$$

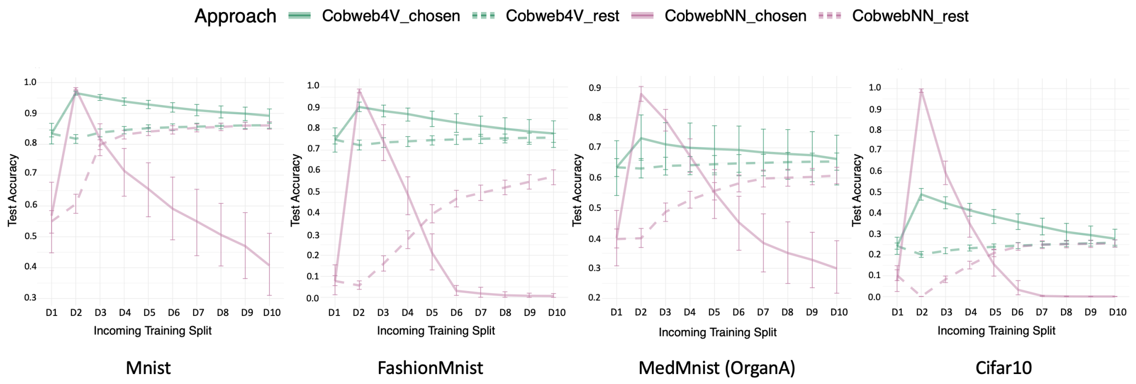


Figure 5: Average accuracy of fixed Cobweb4V vs. sparse CobwebNN on Chosen and Non-chosen classes across datasets. Each subplot shows the test accuracy after each training split. Solid lines represent accuracy on the chosen class; dashed lines represent average accuracy on non-chosen classes.

Here, θ denotes the learnable parameters of the network, t indexes the training step, and x_t is the input processed at step t. This uniform weighting makes the network more prone to interference from recent inputs and thus more vulnerable to forgetting. Both models were evaluated under the same continual learning protocol, with performance on the chosen and non-chosen classes tracked after each training split.

4.5.2 Results

Figure 5 compares fixed Cobweb/4V with sparse CobwebNN across datasets. For the chosen class (solid lines), CobwebNN showed a sharp accuracy decline over successive splits, a clear sign of catastrophic forgetting. In contrast, Cobweb/4V maintained stable accuracy, demonstrating stronger resistance to forgetting. For the non-chosen classes (dashed lines), both models improved as new tasks were introduced, but Cobweb/4V consistently outperformed CobwebNN, with the gap narrowing as training progressed. Overall, these results indicate that Cobweb/4V preserves earlier knowledge more effectively while still supporting new learning, underscoring the stability-plasticity balance provided by its information-theoretic learning mechanism.

4.5.3 Discussion

These findings suggest that Cobweb/4V's resistance to catastrophic forgetting stems largely from its information-theoretic learning process. By maintaining sufficiency statistics and updating parameters in closed form, the model incorporates new data while preserving essential information about past inputs, eliminating the need to revisit earlier examples. In contrast, CobwebNN relies on parameter updates similar to a moving average, which gradually overwrite older experiences with recent ones. Even when structural configurations are matched, the two models diverge in performance, showing that the learning mechanism itself plays a key role in knowledge retention. Cobweb/4V's probability-based updates allow it to integrate new information in a principled and stable manner, highlighting that algorithmic design, beyond structural adaptivity or sparsity, is central to mitigating catastrophic forgetting in continual learning.

5. Conclusion and Future Work

This study examined the factors behind Cobweb/4V's resilience to catastrophic forgetting in continual learning. Three hypotheses were tested: (1) structural reorganization enhances stability, (2) sparse and selective updates reduce interference, and (3) an information-theoretic learning mechanism supports memory retention. Results show that while restructuring improves flexibility, it is not the main driver of stability. Strongest support emerges for the third hypothesis: Cobweb/4V's use of sufficiency statistics enables accurate, incremental updates without revisiting past data, substantially reducing forgetting. Overall, Cobweb/4V's stability appears to arise from interacting factors, with its information-theoretic learning as the key contributor. These findings highlight the value of concept-based, probabilistic models as an alternative to gradient-based methods for continual learning. Future work should investigate how neural models can adopt Cobweb's adaptive learning dynamics by adjusting their update rules to scale with accumulated experience. Instead of using a fixed learning rate, such models could gradually reduce update magnitudes as evidence grows, similar to Cobweb's incremental learning behavior. This could combine the representational power of neural networks with the statistical stability of concept-based learning, advancing the development of continual learning models that maintain prior knowledge while adapting to new information.

References

Barari, N., & Barari, G. (2025). Continual learning models in aviation systems. *The Collegiate Aviation Review International*, *43*.

Barari, N., & Kim, E. (2021). Linking sparse coding dictionaries for representation learning. *2021 International Conference on Rebooting Computing (ICRC)* (pp. 84–87). IEEE.

Barari, N., Lian, X., & MacLellan, C. J. (2024). Incremental concept formation over visual images without catastrophic forgetting. *arXiv preprint arXiv:2402.16933*.

Barnett, S. M., & Ceci, S. J. (2002). When and where do we apply what we learn?: A taxonomy for far transfer. *Psychological bulletin*, *128*, 612.

Calvert, G., Spence, C., & Stein, B. E. (2004). *The handbook of multisensory processes*. MIT press.

Corter, J. E., & Gluck, M. A. (1992). Explaining basic categories: Feature predictability and information. *Psychological bulletin*, *111*, 291.

Fisher, D. H. (1987). Knowledge acquisition via incremental conceptual clustering. *Machine learning*, *2*, 139–172.

Fisher, D. H. (1996). Iterative optimization and simplification of hierarchical clusterings. *Journal of artificial intelligence research*, *4*, 147–178.

Fisher, D. H., & Langley, P. (1990). The structure and formation of natural categories. *Psychology of Learning and Motivation*, *26*, 241–284.

Fisher, D. H., Pazzani, M. J., & Langley, P. (2014). *Concept formation: Knowledge and experience in unsupervised learning*. Morgan Kaufmann.

French, R. M. (1999). Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, *3*, 128–135.

Gennari, J. H., Langley, P., & Fisher, D. H. (1989). Models of incremental concept formation. *Artificial intelligence*, *40*, 11–61.

Goodfellow, I. J., Mirza, M., Xiao, D., Courville, A., & Bengio, Y. (2013). An empirical investigation of catastrophic forgetting in gradient-based neural networks. *arXiv preprint arXiv:1312.6211*.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Hinne, M., Gronau, Q. F., van den Bergh, D., & Wagenmakers, E.-J. (2020). A conceptual introduction to bayesian model averaging. *Advances in Methods and Practices in Psychological Science*, *3*, 200–215.

Izadkhah, S., & Rekabdar, B. (2023). Deep reinforcement learning based group recommendation system with multi-head attention mechanism. *2023 Fifth International Conference on Transdisciplinary AI (TransAI)* (pp. 120–127). IEEE.

Izadkhah, S., & Rekabdar, B. (2024). Enhanced deep reinforcement learning based group recommendation system with multi-head attention for varied group sizes. *ESANN*.

Izadkhah, S., Rekabdar, B., Wagner, A., Broach, J., & Kothuri, S. (2025). A time series transformer attention model for enhancing bicyclist volume estimation using data fusion and feature selection techniques. *2025 19th International Conference on Semantic Computing (ICSC)* (pp. 60–67). IEEE Computer Society.

Izadkhah, S., Wagner, A., Rekabdar, B., Broach, J., & Kothuri, S. (2024). Enhancing bicyclist volume estimation with data fusion and deep learning techniques. *2024 Conference on AI, Science, Engineering, and Technology (AIxSET)* (pp. 35–44). IEEE.

Jang, E., Gu, S., & Poole, B. (2016). Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

Jang, E., Gu, S., & Poole, B. (2017). Categorical reparameterization with gumbel-softmax. From `https://arxiv.org/abs/1611.01144`.

Jones, G. V. (1983). Identifying basic categories. *Psychological Bulletin*, *94*, 423.

Krizhevsky, A., Hinton, G., et al. (2009). *Learning multiple layers of features from tiny images*. Technical report.

LeCun, Y. (1998). The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*.

MacLellan, C. J., Harpstead, E., Aleven, V., Koedinger, K. R., et al. (2016). Trestle: a model of concept formation in structured domains. *Advances in Cognitive Systems*, *4*, 131–150.

MacLellan, C. J., Matsakis, P., & Langley, P. (2022). Efficient induction of language models via probabilistic concept formation. *Proceedings of the Tenth Annual Conference on Advances in Cognitive Systems*.

MacLellan, C. J., & Thakur, H. (2022). Convolutional cobweb: A model of incremental learning from 2d images. *Proceedings of the Ninth Annual Conference on Advances in Cognitive Systems*.

Masse, N. Y., Grant, G. D., & Freedman, D. J. (2018). Alleviating catastrophic forgetting using context-dependent gating and synaptic stabilization. *Proceedings of the National Academy of Sciences*, *115*, E10467–E10475.

McCloskey, M., & Cohen, N. J. (1989). Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, 109–165. Elsevier.

McKusick, K., & Thompson, K. (1990). *Cobweb/3: A portable implementation*. Technical report.

Miconi, T., Stanley, K., & Clune, J. (2018). Differentiable plasticity: training plastic neural networks with backpropagation. *International Conference on Machine Learning* (pp. 3559–3568). PMLR.

Olshausen, B. A., & Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, *381*, 607–609.

Parisi, G. I., Kemker, R., Part, J. L., Kanan, C., & Wermter, S. (2019). Continual lifelong learning with neural networks: A review. *Neural networks*, *113*, 54–71.

Wang, L., Zhang, X., Su, H., & Zhu, J. (2023). A comprehensive survey of continual learning: Theory, method and application. *arXiv preprint arXiv:2302.00487*.

Wang, Z., Haarer, E. L., Barari, N., & MacLellan, C. J. (2025). Taxonomic networks: A representation for neuro-symbolic pairing. *arXiv preprint arXiv:2505.24601*.

Xiao, H., Rasul, K., & Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*.

Yang, J., Shi, R., Wei, D., Liu, Z., Zhao, L., Ke, B., Pfister, H., & Ni, B. (2023). Medmnist v2-a large-scale lightweight benchmark for 2d and 3d biomedical image classification. *Scientific Data*, *10*, 41.