

1 A Appendix

2 Contents

| | | |
|----|--|---|
| 3 | A.1 MIRA: A Multi-Perspective Analysis | 1 |
| 4 | A.1.1 Meta-learning (Hypernetworks) | 1 |
| 5 | A.1.2 Functional Interpolation/Extrapolation | 1 |
| 6 | A.1.3 Test Time Adaptation | 2 |
| 7 | A.1.4 Biological Perspective | 2 |
| 8 | A.2 Additional Experiments | 3 |
| 9 | A.3 Experimental Details | 4 |
| 10 | A.4 Dataset Details | 4 |
| 11 | A.5 Limitations | 5 |

12 A.1 MIRA: A Multi-Perspective Analysis

13 Our method, **MIRA** is a generic framework that can be viewed from four contemporary, broad
14 perspectives in the general context of machine learning, going beyond the perspective introduced in
15 the main paper. We elaborate on those perspectives below.

16 A.1.1 Meta-learning (Hypernetworks)

17 Hypernetworks are models that predict weights of other models [HDL16, CZL⁺24]. They offer a
18 single-loop approach to meta-learning by eliminating the need for an inner-loop adaptation (as in
19 MAML [FAL17]), instead learning to output task-specific parameters in one forward pass. Since
20 **MIRA** predicts adapters at each consecutive block hierarchically, the associative memories in **MIRA**'s
21 architecture can be viewed as a specific type of hypernetwork. However, hypernetwork training is
22 often unstable and is unable to support large models such as ViT architectures. We address this
23 issue by providing a "supervision" to guide the direction of learning of the underlying hypernetwork.
24 Hypernetworks then essentially become models that map a deterministic set of inputs to their outputs
25 deterministically, resembling the behavior of associative memories, and can be implemented as
26 such. We posit this perspective in this work, wherein weight retrieval is achieved by storing the
27 adapter weights over tasks defined over the training set, and retrieving them differentially, instead
28 of attempting to both learn and memorize them simultaneously as done in traditional hypernetwork-
29 based learning. Thus Hopfield Networks [RSL⁺21], Predictive Coding Networks [YW22, TSM⁺23],
30 and any such **AM** may in theory be used in this framework. Practical experiments however, indicated
31 that despite PCNs [YW22] exhibiting better compression properties than Hopfield Nets, they usually
32 have unsatisfactory retrieval quality when the vectors to be retrieved have very high dimension. This is
33 expected behavior especially in the context of storing weight adapters of large foundational models. In
34 addition, their reliance on algorithms incompatible with backpropagation makes it difficult to integrate
35 them into models that need to be trained end-to-end. Hopfield Nets, on the other hand, provide high
36 fidelity in retrieving such high-dimensional vectors at scale, and are implicitly differentiable.

37 A.1.2 Functional Interpolation/Extrapolation

38 In this work, we utilize affine combinations of task-specific adapters for retrieval across different
39 domains. However, our associative memory-centric learned retrieval framework is versatile and can
40 seamlessly accommodate richer, non-linear retrieval mechanisms by modifying the underlying simi-
41 larity metric strategy. In principle, one could design much more expressive (non-linear) combination
42 schemes to merge knowledge from multiple domains, rather than restricting to linear interpolation.
43 This is an interesting direction of future work for this paper. Such non-linear retrieval approaches have
44 been explored in recent literature, such as in sparsely-gated mixture-of-experts models [SMM⁺17]
45 use a learned gating function to dynamically select only a subset of expert parameters for each
46 input (instead of a fixed weighted average) [He24], or in using learned retrieval functions (e.g.,
47 trainable hashing or routing instead of standard nearest-neighbor search) yields better scaling and can
48 capture latent structure in the memory, outperforming fixed similarity measures [He24]. Crucially,

there is both practical and theoretical evidence of directly retrieving such non-linear ensembles of experts/adapters in Hopfield Networks, such as in [SNMM24].

Thus, our method can be viewed as retrieving an interpolated task-specific knowledge proxy (adapter) in a space defined by the chosen functional form of combination. With a sufficiently expressive interpolation function, this approach can even enable extrapolation to out-of-distribution tasks. In settings like **DG**, the target domain may lie outside the convex hull of the source domains, wherein the model must generalize beyond any seen domain mixture. A suitably rich, non-linear combination strategy could, in principle, can facilitate extrapolation of the memorized adapter weights to novel task distributions [ABLR23]. Our approach thus offers a generalizable framework capable of both capturing nuanced relationships between known tasks and extending learned knowledge to new domains beyond the scope of training distributions. We leave the exploration of such function schemes to future work.

A.1.3 Test Time Adaptation

At test time, the optimal combination of adapters for a given input may not be obtained using pre-defined weight coefficients. In general, determining the adapter coefficients that best serve a new downstream sample can require solving an optimization problem on a per-sample basis, as posited in Equation 3. In other words, Equation 3 formalizes the idea that the best adapter composition $\{\alpha_{t,i}^*(x)\}^{T,L}$ for an input x is obtained by minimizing a suitable objective for that specific sample at inference time (instead of applying a fixed combination rule). This perspective aligns with the paradigm of test-time adaptation in literature, wherein models trained only on source data are adapted to target data during inference time [XS24, IM21]. As an example, Tent (Fully Test-Time Entropy Minimization) performs online model updates during testing by minimizing the entropy of its predictions for each test batch, thereby adjusting normalization parameters to increase the model’s confidence on the target distribution [WSL⁺20]. One could view our **MIRA** framework as implementing this idea via a memory-based inference mechanism. Since **MIRA** learns a set of key representations (i.e. associative memory slots in a Hopfield network) during training that are used to derive weights on a per-sample basis, it can also use the learned keys at test time on a per-sample basis effectively, performing adaptation via associative recall. By casting test-time adaptation as an integral part of inference (through solving a Hopfield memory retrieval optimization akin to Equation 3), **MIRA** can be viewed also as a test-time adaptation strategy within a unified, optimized memory-based framework.

A.1.4 Biological Perspective

Notably, **MIRA** can also be perceived as a biologically plausible framework that solves multiple settings such as **DG**, **CIL**, and **DIL**. While Hopfield Nets are well known as biologically implementable **AM** mechanisms, even the affine combinations that we adopt are well-founded in biological mechanisms. Specifically, Tolman-Eichenbaum Machine (TAM) [WMM⁺20, WWB22], a model of the Hippocampus, proposes linear combinations of stored memories as implementable (illustrated in Section A.2). We further observe that the incremental learning settings, when accompanied by DualGPM [LL23] as the strategy to mitigate catastrophic forgetting, resolve into Generalized Hebbian learning principles [San89] in the gradient space of stored memories.

Lemma 1 (DualGPM–Hebbian Gradient Subspace Equivalence). *Let $\mathcal{G}_t = \{g_i\}_{i=1}^{N_t} \subset \mathbb{R}^d$ be the set of gradient vectors observed while training on task t and let the empirical second-moment matrix be*

$$\Sigma_t = \frac{1}{N_t} \sum_{i=1}^{N_t} g_i g_i^\top.$$

In the DualGPM algorithm, for a given energy budget $\varepsilon \in (0, 1)$, we define:

$$k := \arg \min_{k \in [d]} \frac{\sum_{j=1}^k \lambda_j}{\sum_{j=1}^d \lambda_j} \geq \varepsilon.$$

where $\lambda_1 \geq \dots \geq \lambda_d$ are the eigenvalues of Σ_t .

Let $U_t \in \mathbb{R}^{d \times k}$ the orthonormal basis produced by the DualGPM memory update for the energy budget ε . Independently, let $W_t \in \mathbb{R}^{d \times k}$ be the weight matrix obtained as a stationary point of the Generalised Hebbian (Oja) update averaged over each gradient in \mathcal{G}_t (with row sums equal to 1).

96 Then,

$$\text{span}(U_t) = \text{span}(W_t).$$

97 **Proof. Step 1: Optimality of DualGPM.** We first note that the precise objective that DualGPM tries
98 to solve is given by:

$$U_t = \arg \min_{U^\top U = I_k} \text{Tr}[(I - UU^\top)\Sigma_t]$$

99 By the Eckart-Young-Mirsky theorem, the optima is attained at those U , whose columns are the
100 k eigenvectors (subject to permutation and sign inversion) of Σ_t corresponding to the k largest
101 eigenvalues, $\lambda_1, \dots, \lambda_k$. Hence U_t satisfies the eigenvalue equation

$$\Sigma_t U_t = U_t \Lambda, \quad \Lambda = \text{diag}(\lambda_1, \dots, \lambda_k). \quad (1)$$

102 **Step 2: Fixed points of the Hebbian rule.** The Generalized Hebbian update rule [?] for a gradient
103 vector $g \in \mathcal{G}_t$ is given by:

$$\Delta W(g) = \eta(gg^\top W - W \text{diag}(W^\top gg^\top W)).$$

104 Considering the weight update averaged over all $g \in \mathcal{G}_t$ and imposing stationary conditions on the
105 same yields:

$$\Sigma_t W - W \text{diag}(W^\top \Sigma_t W) = 0.$$

106 Pre-multiplying by W_t^\top and noting that each row sum of W_t^\top equals one, shows that the diagonal
107 term on the right is precisely the eigenvalue matrix of the projected covariance, so the above is
108 identical to (1) with U_t replaced by W_t .

109 Consequently, U_t and W_t have the same k -dimensional eigenbasis, corresponding to the k eigenvec-
110 tors of Σ_t with the k -largest eigenvalues, which proves the assertion. \square

111 Thus, Lemma 1 implies that there exists an orthogonal matrix $R_{k \times k}$ such that $U_t = W_t R$.

112 **Implications.** This lemma elevates the biological analogy behind DualGPM into a provable equiv-
113 alence: the algorithm’s batch SVD update computes *exactly* the same principal gradient subspace
114 that an online Oja-style Hebbian learner would converge to. The result has three immediate conse-
115 quences: (i) *Theoretical grounding*: Any optimal variance or noise filtering guarantees enjoyed by
116 Hebbian PCA now apply to DualGPM’s memory, providing a principled basis for its strong empirical
117 resistance to catastrophic forgetting; (ii) *Algorithmic unification*: Projection-based continual learn-
118 ing methods can be re-interpreted through the lens of gradient-space Hebbian consolidation; and
119 (iii) *Neuro-inspired design*: By demonstrating that protecting past tasks is tantamount to a Hebbian
120 consolidation step, the lemma bridges continual learning research with synaptic consolidation theories
121 in neuroscience, motivating biologically grounded extensions such as local online updates or neural
122 gating via associative memory.

123 A.2 Additional Experiments

124 **Choice of $g(\cdot)$.** In the tables in the main paper, we set g described in the conceptual framework to
125 be an identity function. We tabulate the results below for different choices of g . We find that this
126 function has minimal impact on performance, indicating that the transformations performed within
127 the different layers of ViT are strong and fairly sufficient by themselves to constitute an eigenbasis
128 wherein the appropriate keys can be found via gradient descent.

Table A1: Comparison of different choices of g for DIL and DG settings.

| g | DIL | DG |
|-------------|-------|-------|
| Identity | 69.18 | 61.19 |
| Linear | 69.22 | 60.98 |
| 3-layer MLP | 69.22 | 61.12 |

Prefixes as memories. The proposed **MIRA** framework bears resemblance to the complementary learning system implemented by hippocampal-cortical connections in the brain [STBBN17]. This framework, however, models the hippocampus as a memory storage for representations, rather than neural overlays. The analogue to such a mechanism in contemporary deep learning architectures is Prompt Tuning [WZL⁺22] in PEFT literature. In particular, the prefix tuning [LL21] variant of prompt tuning can be directly integrated into the **MIRA** framework to implement a system analogous to a neuroscientific framework, with the pretrained network serving the role of the cortical circuits with powerful generalization capabilities, and the prefixes - stored in associative memories - serving as task-specific representations. We compare the prefix-tuning based approach with our original **MIRA** framework. Our results indicate that this variant maintains comparable performance to storing overlay weights, showing that **MIRA** can be adapted to different PEFT methods.

Table A2: Comparison of Prefix Tuning vs LoRA tuning in **MIRA**.

| MIRA Variant | DIL | DG |
|---------------|-------|-------|
| MIRA-default | 69.18 | 61.19 |
| MIRA-Prefixes | 69.61 | 60.72 |

A.3 Experimental Details

For training task-specific adapters in the *Adaptation* stage across all datasets and settings, we use rank-4 LoRA adapters trained for 5 epochs with a learning rate of 1e-3. For CIL and DIL experiments, we set the DualGPM threshold to 0.7. The AdamW optimizer is used with a weight decay of 1e-3 across all experiments as well. All our experiments are performed on a single RTX A6000 Ada GPU with 48GB VRAM, on a machine having a 96-core Intel Xeon CPU and 128GB RAM.

In the *Consolidation* stage, all experiments in DIL and CIL settings ran for 2 epochs. In addition, we initialized the CIL classifiers in the *Consolidation* stage with the weights learned in the *Adaptation* stage for the corresponding label set. Note that this is not effective in the DIL setting as even though the label sets are the same, the distribution of inputs to the classifier changes, and hence the scope of knowledge transfer in the linear classifier head is limited in this setting. In the *Consolidation* stage of the DG setting, we run PACS, OfficeHome and VLCS for 10 epochs, while DomainNet is just run for one epoch. We rescale all images to 256×256 during both training and evaluation and take a 224×224 crop from this rescaled image (random crop during training, center crop at inference) as input to the model. We apply a random horizontal flip as a training augmentation in all cases, and an additional mixup augmentation for the DG setting. In all CIL and DIL settings, we use the AdamW optimizer with a weight decay and learning of 1e-3, while in the DG setting, we set the learning rate to 7e-4.

A.4 Dataset Details

DomainNet. DomainNet is a large-scale benchmark comprising approximately 600,000 images across 345 categories, distributed over six distinct domains: Real, Clipart, Infograph, Painting, Quickdraw, and Sketch. Each domain introduces a unique visual style, presenting significant domain shifts. In the DIL setup, each domain is treated as a separate experience, with the model sequentially exposed to data from one domain at a time while maintaining a consistent label space. This setup challenges models to generalize across diverse visual domains without forgetting previously learned knowledge. In the CIL setup, the dataset is divided into 5 experience, each experience containing 69 classes from all 6 domains combined. Unlike the DIL setting, the label space in the CIL setting grows with each experience. In the DG setup, models are trained on 5 domains conjointly and evaluated on the 6th unseen domain.

DN4IL. DN4IL is a curated subset of DomainNet, specifically designed for evaluating domain-incremental learning methods. It retains the six domains from DomainNet but focuses on a reduced set of 100 classes to facilitate controlled experiments on domain shifts. The dataset emphasizes the challenges posed by significant distributional differences between domains, making it a suitable benchmark for assessing the robustness of continual learning algorithms .

iDigits. iDigits is a domain-incremental benchmark constructed by combining four digit recognition datasets: MNIST, SVHN, MNIST-M, and SYN. Each dataset represents a distinct domain with varying visual characteristics. In the DIL setting, the model is trained sequentially on each domain, with the objective of maintaining performance across all domains despite the domain shifts. This benchmark is particularly useful for studying the effects of domain shifts in simpler classification tasks. In the CIL setting, all datasets are jointly split into 5 training experiences, each experience containing 2 classes from each of the 4 datasets.

CORE50. CORE50 is a dataset designed for continuous object recognition, consisting of 50 household objects recorded under 11 different environmental conditions. Each condition introduces variations such as background changes, lighting, and occlusions. In the domain-incremental setup, each environmental condition is treated as a separate domain, and the model learns to recognize the same set of objects across these varying conditions. A key difference from other DIL datasets is that CORE50 uses 3 of the 11 domains as the test set, and incrementally trained on the other 8 domains. This setup evaluates a model’s ability to generalize object recognition across different real-world scenarios. It can thus also be viewed as a combination of DIL and DG settings, where the test set comprises of unseen domains. A forgetting of ≤ 0 indicates that the models’ performance remains the same or improves on the unseen domains as new domains are incrementally learned. The CIL setting is similar to DomainNet and iDigits - the dataset is split into 5 experiences of 10 classes each, encompassing all 11 training domains.

CDDDB. CDDDB (Continual Deepfake Detection Benchmark) is a dataset aimed at evaluating continual learning methods in the context of deepfake detection. It comprises a collection of deepfake videos generated using various known and unknown generative models. In the DIL framework, each generative model represents a different domain, and the model is sequentially trained to detect deepfakes from these diverse sources. CDDDB challenges models to adapt to new types of deepfakes while retaining the ability to detect previously encountered ones. We particularly evaluate on the CDDDB-hard subset, comprising of five domains: GauGAN, BigGAN, WildDeepfake, WhichFaceReal, and SAN.

ImageNet-R. ImageNet-R is a dataset comprising 30,000 images of 200 ImageNet classes, with images rendered in various styles such as art, cartoons, graffiti, embroidery, and video games. This dataset is designed to evaluate the robustness of models to distribution shifts. In the CIL setup, the 200 classes are divided into 5 or 10 tasks, each containing 40 or 20 unique classes. The model is trained sequentially on these tasks, with the goal of learning new classes while maintaining performance on previously learned ones.

VLCS. VLCS is a benchmark dataset for domain generalization, comprising images from four distinct domains: PASCAL VOC2007, LabelMe, Caltech-101, and SUN09. Each domain contains images labeled across five shared object categories: bird, car, chair, dog, and person. The dataset includes a total of 7,510 images, with domain-specific distributions. In the DG setup, models are trained on three domains and tested on the remaining one, evaluating their ability to generalize to unseen domains.

PACS. PACS is an image dataset designed for domain generalization, consisting of four domains: Photo, Art Painting, Cartoon, and Sketch. Each domain contains images from seven categories: dog, elephant, giraffe, guitar, horse, house, and person. The dataset comprises a total of 9,991 images, with varying numbers across domains. PACS introduces significant domain shifts due to the diverse visual styles, making it a challenging benchmark for DG methods.

OfficeHome. OfficeHome is a benchmark dataset for domain adaptation and generalization, containing images from four domains: Art, Clipart, Product, and Real-World. Each domain includes 65 categories of everyday objects, totaling approximately 15,500 images. The dataset presents substantial domain shifts due to differences in image styles and acquisition methods. In the DG setup, models are trained on three domains and evaluated on the fourth, assessing their ability to generalize to unseen domains.

A.5 Limitations

This work highlights the benefits of incorporating neuroscientific insights into deep learning architectures, especially in the context of biologically plausible memory mechanisms. In particular, it proposes a potential mechanism in which such task-switching can occur in biological systems with

the aid of associative memories. The work constraints to task settings such as CIL, DIL and DG; extensions to related settings such as Versatile Incremental Learning or Multi-Task Learning, or even other PEFT methods, would be interesting future extensions of our framework. All experiments provided are based on computational models from deep learning research; analogous neuroscience experiments may need to be conducted to confirmatively declare if memory mechanisms are indeed used in the stated manner in biological systems. Besides, validating this framework on non-ViT architectures such as ResNets is also possible, and may help extend this work more generally to all architectures.

References

- [ABLR23] Emmanuel Abbe, Samy Bengio, Aryo Lotfi, and Kevin Rizk. Generalization on the unseen, logic reasoning and degree curriculum june 2023. In *URL https://openreview.net/forum*, 2023.
- [CZL⁺24] Vinod Kumar Chauhan, Jiandong Zhou, Ping Lu, Soheila Molaei, and David A Clifton. A brief review of hypernetworks in deep learning. *Artificial Intelligence Review*, 57(9):250, 2024.
- [FAL17] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.
- [HDL16] David Ha, Andrew Dai, and Quoc V Le. Hypernetworks. *arXiv preprint arXiv:1609.09106*, 2016.
- [He24] Xu Owen He. Mixture of a million experts. *arXiv preprint arXiv:2407.04153*, 2024.
- [IM21] Yusuke Iwasawa and Yutaka Matsuo. Test-time classifier adjustment module for model-agnostic domain generalization. *Advances in Neural Information Processing Systems*, 34:2427–2440, 2021.
- [LL21] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online, August 2021. Association for Computational Linguistics.
- [LL23] Yan-Shuo Liang and Wu-Jun Li. Adaptive plasticity improvement for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7816–7825, 2023.
- [RSL⁺21] Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Thomas Adler, David Kreil, Michael K Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. Hopfield networks is all you need. In *International Conference on Learning Representations*, 2021.
- [San89] Terence D Sanger. Optimal unsupervised learning in a single-layer linear feedforward neural network. *Neural networks*, 2(6):459–473, 1989.
- [SMM⁺17] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarczyk, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017.
- [SNMM24] Saul Santos, Vlad Niculae, Daniel McNamee, and Andre FT Martins. Sparse and structured hopfield networks. *arXiv preprint arXiv:2402.13725*, 2024.
- [STBBN17] Anna C Schapiro, Nicholas B Turk-Browne, Matthew M Botvinick, and Kenneth A Norman. Complementary learning systems within the hippocampus: a neural network modelling approach to reconciling episodic memory with statistical learning. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 372(1711):20160049, 2017.

- 277 [TSM⁺23] Mufeng Tang, Tommaso Salvatori, Beren Millidge, Yuhang Song, Thomas Lukasiewicz,
278 and Rafal Bogacz. Recurrent predictive coding models for associative memory employ-
279 ing covariance learning. *PLoS computational biology*, 19(4):e1010719, 2023.
- 280 [WMM⁺20] James C R Whittington, Timothy H Muller, Shirley Mark, Guifen Chen, Caswell Barry,
281 Neil Burgess, and Timothy E J Behrens. The Tolman-Eichenbaum machine: Unifying
282 space and relational memory through generalization in the hippocampal formation.
283 *Cell*, 183(5):1249–1263.e23, November 2020.
- 284 [WSL⁺20] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Dar-
285 rell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint*
286 *arXiv:2006.10726*, 2020.
- 287 [WWB22] James C. R. Whittington, Joseph Warren, and Tim E.J. Behrens. Relating transformers
288 to models and neural representations of the hippocampal formation. In *International*
289 *Conference on Learning Representations*, 2022.
- 290 [WZL⁺22] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren,
291 Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for
292 continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision*
293 *and Pattern Recognition*, pages 139–149, 2022.
- 294 [XS24] Zehao Xiao and Cees GM Snoek. Beyond model adaptation at test time: A survey.
295 *arXiv preprint arXiv:2411.03687*, 2024.
- 296 [YW22] Jinsoo Yoo and Frank Wood. Bayespcn: A continually learnable predictive coding
297 associative memory. *Advances in Neural Information Processing Systems*, 35:29903–
298 29914, 2022.