Emergent representations in networks trained with the Forward-Forward algorithm

Niccolò Tosato[1,2,*], **Lorenzo Basile**[2,*], **Emanuele Ballarin**[2],
Giuseppe de Alteriis[3,4], **Alberto Cazzaniga**[1], **Alessio Ansuini**[1]
[1]AREA Science Park, Italy
[2]University of Trieste, Italy
[3]King's College London, UK
[4]University College London, UK

Abstract

Backpropagation has been criticised for its lack of biological realism. In this work, we show that the internal representations obtained by the Forward-Forward algorithm can organise spontaneously into category-specific *ensembles* exhibiting high sparsity. This situation is reminiscent of what has been observed in cortical sensory areas, where neuronal ensembles are suggested to serve as the functional building blocks for perception and action. Our findings suggest that the learning method used by Forward-Forward may be more biologically plausible than Backpropagation, particularly in terms of the emergent representations it produces.

1. Introduction

The Backpropagation (Backprop) algorithm lacks biological plausibility [23] – leading to many attempts to address the issue. The most recent of such, the Forward-Forward (FF) algorithm [9], eliminates the need to store neural activities and propagate error derivatives along the network.
This work investigates the presence of similarities between artificial and biological neuronal ensembles *i.e.* sets of a relatively low number of units that specifically and consistently co-activate in presence of data samples of a given class [16].
Our experiments demonstrate that: 1) representations in FF show high levels of sparsity, and give raise to well-defined neuronal ensembles (subsection 4.1), 2) the ensembles of related categories often share units (subsection 4.2) as in their biological counterpart, and 3) representations of data from categories not included in the training set (unseen categories) generate distinct ensembles; these ensembles integrate with those from seen categories by sharing units (subsection 4.3). Though optimising the cross-entropy loss with Backprop for the same classification task does not appear to produce sparse ensembles, the phenomenon may not solely be due to the use of FF. In fact, similar results are obtained by optimising the same goodness function of FF with Backprop. This suggests that more focus should be put on the purpose and biological meaning of the loss function rather than the training algorithm [22].

2. Related Work

2.1. Forward-Forward

The Forward-Forward algorithm involves two forward passes executed on different data, named positive and negative data. During training, the objective of Forward-Forward is to maximise a

. *Equal contribution. Correspondence to: alessio.ansuini@areasciencepark.it

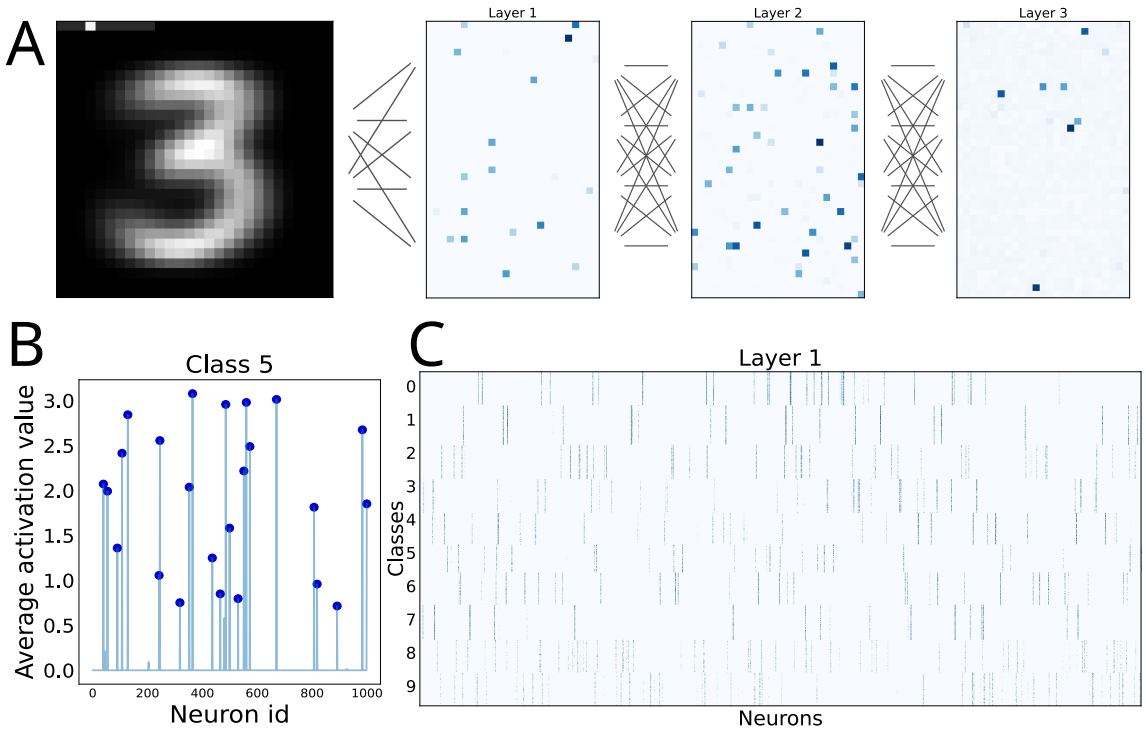

Figure 1: **A**) Activation patterns in different layers of a **FF** model. **B**) Activation values of neurons in a layer averaged across samples in class 5 (MNIST). Blue dots indicate units of the ensemble. **C**) Activation map for all neurons in a layer, grouped by class. The *barcodes* mark different classes.

goodness function of the neural activations on positive data and minimise it on negative data. In a simple image classification setting, one could encode a class label at the border of images, by one-hot encoding it with a white pixel. Positive data are those for which the encoded label matches the ground truth label, while the opposite holds for negative data.
Layers are trained separately and sequentially. Crucially, activations are normalised before being passed to the subsequent layer, to prevent layers from relying on the goodness computed by their predecessors. At test time, when a new unlabelled sample has to be classified, many copies of the image are created, each with a different one-hot encoded label, and the predicted label is the one that receives the highest goodness.

2.2. Neuronal ensembles

In Neuroscience, ensembles are defined as sparse groups of neurons that co-activate during spontaneous activity or in response to sensory stimuli. Ensembles – rather than single neurons – have long been suggested to be emergent functional blocks of cortical activity [6–8, 26] and have a prominent role in sensory processing, memory [10] and behaviour [2]. Single neurons can participate in more than one ensemble, thus maximising the encoding potential of the network. These neurons of ensembles consistently and reliably encode the same inputs across multiple trials.
The idea of neuronal ensembles has inspired the development of computational models of sparsity and redundancy in neural representations [19], [4], [5], [18].

3. Methods

We compare the representations in three models: 1) a MLP classifier like the one in [9] trained with Forward-Forward (**FF**), 2) the same classifier but trained end-to-end with Backprop to optimise the same goodness function (**BP/FF**), 3) the same classifier trained with Backprop on the cross-entropy loss (**BP**). The datasets we use to train and test the models described are MNIST [14], FASHIONMNIST [24], SVHN [17] and CIFAR10 [1] (see section B for details).

3.1. Model trained with Forward-Forward (FF)

Our **FF** model is inspired by [9] – and trained according to the Forward-Forward algorithm. It consists of three fully-connected layers, each composed by 1000 units in the case of MNIST and FASHIONMNIST, and 3072 units in the case of SVHN and CIFAR10. Each linear layer is followed by element-wise ReLU non-linearities. Both during training and inference, the layer-wise ℓ_2 norm is used as the goodness function of choice; correspondingly, ℓ_2 normalisation is performed between subsequent layers.
To define positive and negative data, a one-hot-encoded class vector is embedded at the top-left corner of images (randomly assigned in the case of negative data). During training, the weights are optimised by minimising the loss function $L = \log(1 + e^{G_{neg} - G_{pos}})$, where G_{neg} and G_{pos} are, respectively, the goodness value for negative and positive data. At inference time, for each layer, the goodness values corresponding to every possible label are converted into probabilities using softmax.

3.2. Model trained with Backpropagation on the goodness objective (BP/FF)

The architecture of the **FF** model can be trained seamlessly with Backpropagation on the same goodness maximisation/minimisation objective. Positive and negative data are fed to the network and the overall goodness of the internal representation is evaluated. The backward pass is then executed. In this case, the goodness is optimised globally instead of layer-by-layer.

3.3. Model trained with Backpropagation on the cross-entropy loss (BP)

The **FF** and **BP/FF** models are also compared to a standard neural classifier with the same architecture (**BP**), serving as a baseline. The only difference between **BP** and the other two is the addition of a final softmax layer. The model is trained with Backprop on the cross-entropy loss.

3.4. Analysis of representations

We analyse representations of test set data correctly classified. The representation of a sample is a n-dimensional vector composed by the activations (after the ReLU non-linearity) of all the units in the layer. For each layer, we extract a representation matrix X of size (M, n), where M is the total number of test samples (correctly classified) and n is the number of neurons in the layer considered.

Sparsity For each vector x we compute a *sparsity* following [11]: $S(x) = \frac{\sqrt{n} - \frac{\|x\|_1}{\|x\|_2}}{\sqrt{n} - 1}$. When $S(x) = 1$ the vector x contains only one non-zero component, the other limiting case is the one in which all the components of x are equal in magnitude, where $S(x) = 0$. $S(x)$ interpolates smoothly between these two extremes. The sparsity of a layer is the average on its component vectors $S = \frac{1}{M} \sum_{i=1}^{M} S(x_i)$.

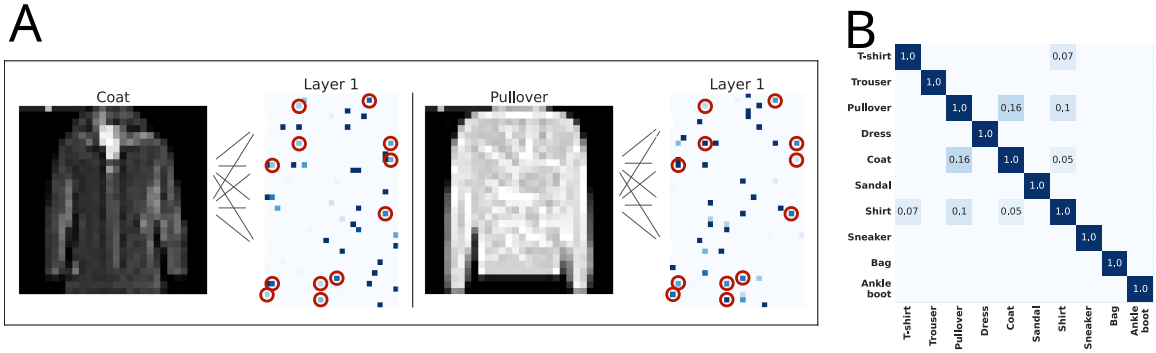

Figure 2: Ensembles of similar classes in FASHIONMNIST share units. **A**) Ensembles in **FF** elicited by two visually related inputs. Red circles indicate the shared units. **B**) Similarity of the ensembles: $\frac{|\mathcal{E}^i \cap \mathcal{E}^j|}{|\mathcal{E}^i \cup \mathcal{E}^j|}$. The results are referred to a single training run.

Ensembles We consider a neuron as part of an ensemble for a category c if it activates consistently when the network receives data that belongs to c. We consider category-specific representation matrix X_c, of shape (M_c, n), where M_c is the number of correctly classified test images of c. Then, we compute the average activation of each hidden unit across all these samples: $\overline{x_j} = \frac{1}{M_c} \sum_{i=1}^{M_c} (X_c)_{ij}$; and the leave-one-out average of the averages $\text{LOO}_j = \frac{1}{n-1} \sum_{i \neq j} \overline{x_j}$. We then classify a neuron i as part of the ensemble ($\mathcal{E}^c$) if $\overline{x_i} > \text{LOO}_i$. An example of an average activation profile ($\overline{x_j}$) and its ensemble is reported in Figure 1, **B**. When the sparsity S is low, ensembles are ill-defined. We will consider values of S below 0.5 as not-sparse, and in these cases we do not define ensembles.

4. Results

4.1. Forward-Forward elicits sparse neuronal ensembles

The **FF** and **BP/FF** models exhibit typically high sparsity levels in their representations, in clear contrast with **BP**. As an example, for MNIST we have $S \approx 0.9$ (**FF**), $S \approx 0.9$ (**BP/FF**) and $S \approx 0.3$ (**BP**), in the first hidden layer. The trend identified by these values generalises well across all our experiments (see Table 3).

When $S > 0.5$ we are typically able to identify ensembles composed by few participating units. Figure 1, **B** shows an example of average neuron activations ($\overline{x_j}$), and its ensemble (blue dots). Ensembles are category-specific, as evidenced by the barcode-like structure of the activation maps in Figure 1, **C**. Other activation maps, across different layers and models are reported in section G, section H for similar visualisations. The fraction of units that participate to ensembles is reported in Table 4, where a slight increasing trend in the ensemble size with the complexity of the data is observed.

4.2. Similar classes elicit ensembles with shared neurons

Visually related categories can be expected to share units of their ensembles in real brains [25]. This is what we systematically observe in **FF** and **BP/FF**. An example is shown in Figure 2. Remarkably, units can also be shared across two ensembles even if one of these refers to an unseen category (*i.e.*, excluded from the training) as we show in Figure 3.

4.3. Representations of unseen categories can elicit well-defined ensembles

Trained **FF** models respond to unseen categories with a well-defined ensemble. We repeatedly train **FF** on FASHIONMNIST, removing one category at a time. In each case a new ensemble is formed, with the same characteristics of the "regular" ones, except a lower average activation (see Figure 3 **A**). We also find that the ensembles of unseen categories share units with the ensembles of seen categories, when the two are related by visual similarity (Figure 3, **B**). A more extensive exploration of these cases is reported in section I.

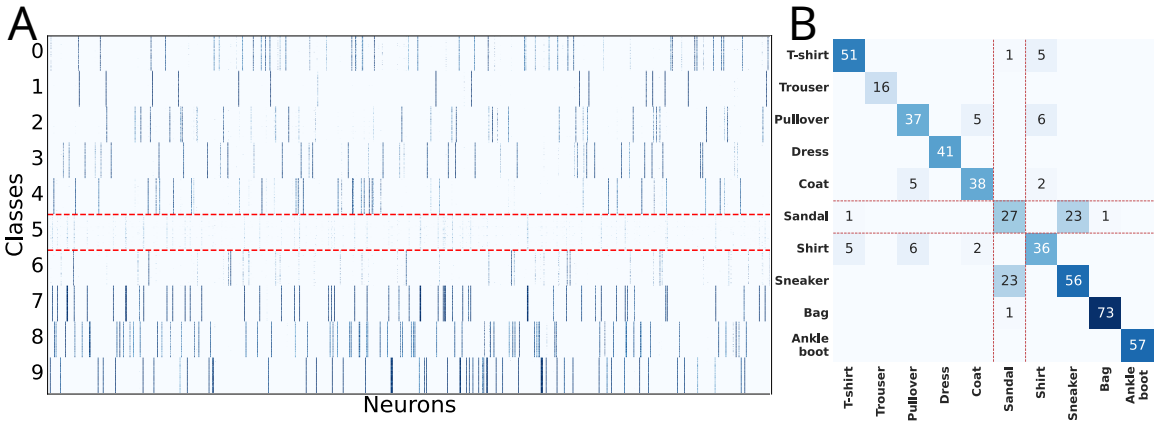

Figure 3: The representation of unseen categories forms a valid ensemble in **FF** trained on FASHIONMNIST. **A**) Activation patterns in response to the different categories in the first hidden layer. The unseen category (`Sandal`), surrounded by red lines, produces a relatively weaker but well-defined ensemble-like activation pattern. **B**) The unseen category integrates well with related seen categories by sharing units. Number of units shared between category pairs (i, j): $\mid \mathcal{E}^i \cap \mathcal{E}^j \mid$. The results are referred to a single training run.

5. Discussion and conclusions

Artificial neural networks trained with the Forward-Forward algorithm can elicit biologically plausible representations in the form of sparse neuronal ensembles [16, 26]. The category-specific ensembles we find in Forward-Forward models share similarities with their biological counterparts as: 1) are composed of a few units, 2) can share units across visually/semantically related inputs [25], 3) can be formed in response to an unseen input category, and 4) when the unseen category is characterised by a certain degree of visual similarity, the corresponding ensembles often share one or more units. These results suggest that Forward-Forward can perform well in zero-shot classification tasks, which is also a biologically relevant feat [13]. Overall, our findings – focused on the emerging properties of representations – corroborate the idea that Forward-Forward might be a better model than Backprop for learning in the cortex [9]. As shown in section 4, besides being a non biological learning rule, Backprop elicits non-sparse and less biologically plausible representations.
We emphasise that biologically plausible learning algorithms should generate biologically plausible representations. In this context, the emergence of sparse ensembles – similar to those we observed – is likely an important requisite.

References

[1] Krizhevsky Alex. Learning multiple layers of features from tiny images. *https://www. cs. toronto. edu/kriz/learning-features-2009-TR. pdf*, 2009.

[2] Luis Carrillo-Reid, Shuting Han, Weijian Yang, Alejandro Akrouh, and Rafael Yuste. Controlling visually guided behavior by holographic recalling of cortical ensembles. *Cell*, 178(2): 447–457, 2019.

[3] Dan Claudiu Cireşan, Ueli Meier, Luca Maria Gambardella, and Jürgen Schmidhuber. Deep, big, simple neural nets for handwritten digit recognition. *Neural Computation*, 22(12): 3207–3220, 12 2010.

[4] Eizaburo Doi and Michael Lewicki. Sparse coding of natural images using an overcomplete set of limited capacity units. *Advances in Neural Information Processing Systems*, 17, 2004.

[5] David J Field. What is the goal of sensory coding? *Neural Computation*, 6(4):559–601, 1994.

[6] Buzsáki György. Neural syntax: cell assemblies, synapsembles, and readers. *Neuron*, 68(3): 362–385, 2010.

[7] Kenneth D Harris. Neural signatures of cell assembly organization. *Nature reviews neuroscience*, 6(5):399–407, 2005.

[8] Donald Olding Hebb. *The organization of behavior: A neuropsychological theory*. Psychology press, 2005.

[9] Geoffrey Hinton. The forward-forward algorithm: Some preliminary investigations. *arXiv preprint arXiv:2212.13345*, 2022.

[10] John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.

[11] Patrik O Hoyer. Non-negative matrix factorization with sparseness constraints. *Journal of machine learning research*, 5(9), 2004.

[12] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*, 2017.

[13] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

[14] Yann LeCun and Corinna Cortes. The MNIST handwritten digit database, 2010.

[15] Zhouhan Lin, Roland Memisevic, and Kishore Konda. How far can we go without convolution: Improving fully-connected networks. *arXiv preprint arXiv:1511.02580*, 2015.

[16] Jae-eun Kang Miller, Inbal Ayzenshtat, Luis Carrillo-Reid, and Rafael Yuste. Visual stimuli recruit intrinsically generated cortical ensembles. *Proceedings of the National Academy of Sciences*, 111(38):E4053–E4061, 2014.

[17] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. *Deep Learning and Unsupervised Feature Learning Workshop, NeurIPS*, 2011.

[18] Bruno A Olshausen and David J Field. Sparse coding of sensory inputs. *Current Opinion in Neurobiology*, 14(4):481–487, 2004.

[19] Christos H Papadimitriou, Santosh S Vempala, Daniel Mitropolsky, Michael Collins, and Wolfgang Maass. Brain computation by assemblies of neurons. *Proceedings of the National Academy of Sciences*, 117(25):14464–14472, 2020.

[20] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library, 2019.

[21] Stamatis Pitsios. SVHN number recognition using deep learning. https://github.com/pitsios-s/SVHN, 2017.

[22] Blake A Richards, Timothy P Lillicrap, Philippe Beaudoin, Yoshua Bengio, Rafal Bogacz, Amelia Christensen, Claudia Clopath, Rui Ponte Costa, Archy de Berker, Surya Ganguli, et al. A deep learning framework for neuroscience. *Nature Neuroscience*, 22(11):1761–1770, 2019.

[23] David G. Stork. Is backpropagation biologically plausible? In *Proceedings of the International Joint Conference on Neural Networks*, 1989.

[24] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017.

[25] Takashi Yoshida and Kenichi Ohki. Natural images are reliably represented by sparse and variable populations of neurons in visual cortex. *Nature communications*, 11(1):872, 2020.

[26] Rafael Yuste. From the neuron doctrine to neural networks. *Nature Reviews Neuroscience*, 16 (8):487–497, 2015.

Appendix A. Computational resources

Training and subsequent experiments were conducted on an NVIDIA DGX A100 system. The system is equipped with 8 NVIDIA A100 GPUs, interconnected by NVLink technology, two AMD EPYC 7742 64-core CPUs, 1TB of RAM, and a 3TB NVME storage configured in RAID-0. Each GPU is equipped with 6912 CUDA cores, 432 Tensor cores and 40 GB of high-bandwidth memory.

Appendix B. Data

The MNIST dataset consists of pictures of handwritten Arabic numerals, from 0 to 9, each represented as a grayscale image of size 28×28. FASHIONMNIST has been designed as a drop-in replacement to MNIST, offering a more challenging classification task. It consists of ten classes of clothing items, still represented as grayscale images with a resolution of 28×28. Both datasets provide 60000 training and 10000 test images, balanced in terms of per-class numerosity.
SVHN contains colored images of digits from house numbers, captured by Google StreetView. The images are composed of 32×32 RGB-encoded pixels. This dataset is slightly larger than the previous two, as it contains 73257 data-points in the training set and 26032 in the test set.
The SVHN images have been cropped in order to center the digit of interest within the frame. However, the presence of adjacent digits and other distracting elements, that have been kept within the images, introduces an additional layer of complexity when compared to MNIST and FASHIONMNIST, where the subjects are prominently displayed against a uniform black background. The CIFAR10 consists of 60000 coloured natural images categorised in 10 balanced classes. The dataset is split in 50000 training images and 10000 test images. Each image, like SVHN has a resolution of 32×32 for each channel. Compared to previous datasets, this is the most challenging one for a fully connected network. The dataset split employed is provided by the TORCHVISION framework.

Appendix C. Classification accuracy

We evaluate the performances of our models. Table 1 contains results in terms of test set classification accuracy for all combinations of models and datasets. While some of these accuracy values are far from the state-of-the-art (*i.e.*, respectively, 0.997 [3], 0.931 [24], 0.860 [21] and approximately 0.7 [15], for fully-connected networks), they are a solid ground on which to build our subsequent investigations. Training details and hyperparameters for all models are reported in section D.

Table 1: Test-set classification accuracy. Results expressed as *mean* $\pm$ *std. dev.* over 10 runs with independent randomised weight initialisation.

Dataset	**FF**	**BP/FF**	**BP**
MNIST	0.94 ± 0.008	0.969 ± 0.001	0.982 ± 0.001
FASHIONMNIST	0.849 ± 0.002	0.877 ± 0.002	0.892 ± 0.004
SVHN	0.716 ± 0.002	0.799 ± 0.004	0.793 ± 0.145
CIFAR10	0.484 ± 0.004	0.521 ± 0.006	0.564 ± 0.004

Appendix D. Training details

All our models (**FF**, **BP/FF** and **BP**), on all datasets (MNIST, FASHIONMNIST, SVHN and CIFAR10), have been optimised using Adam [12] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ implemented in PYTORCH [20]. A hyperparameter search has been performed to achieve sufficient accuracy for each model across all datasets. Every model was trained using batches of size 1024.

Table 2: Hyperparameters selected to train our models.

Model		MNIST	FASHIONMNIST	SVHN	CIFAR10
FF	Epochs	1200	100	1000	1000
	Learning rate	0.01	0.01	0.0001	0.0001
BP/FF	Epochs	300	300	200	200
	Learning rate	0.0001	0.0001	0.0001	0.0001
BP	Epochs	80	80	80	80
	Learning rate	0.0001	0.0001	0.0001	0.0001

Appendix E. Sparsity measures

Table 3: Average sparsity measured with the method described in subsection 3.4. Results are expressed as *mean* $\pm$ *std. dev.* computed over 10 runs with independent random weights initialization.

Model	Layer	MNIST	FASHIONMNIST	SVHN	CIFAR10
FF	1	0.922 ± 0.001	0.85 ± 0.002	0.83 ± 0.001	0.77 ± 0.001
	2	0.813 ± 0.019	0.605 ± 0.015	0.706 ± 0.001	0.728 ± 0.002
	3	0.618 ± 0.074	0.628 ± 0.013	0.489 ± 0.004	0.566 ± 0.002
BP/FF	1	0.895 ± 0.005	0.81 ± 0.007	0.783 ± 0.003	0.753 ± 0.004
	2	0.747 ± 0.013	0.851 ± 0.007	0.95 ± 0.003	0.932 ± 0.003
	3	0.131 ± 0.011	0.065 ± 0.009	0.133 ± 0.011	0.135 ± 0.009
BP	1	0.315 ± 0.003	0.352 ± 0.003	0.47 ± 0.02	0.478 ± 0.016
	2	0.193 ± 0.004	0.241 ± 0.005	0.524 ± 0.212	0.3 ± 0.18
	3	0.225 ± 0.006	0.248 ± 0.006	0.232 ± 0.106	0.164 ± 0.006

Appendix F. Average fraction of units

Table 4: Average fraction of units taking part in ensembles. Ensemble sizes are averaged across all categories, divided by the number of neurons in a layer, and then expressed in %. Results expressed as *mean* $\pm$ *std. dev.*. In the third layer of **BP/FF**, as well as in **BP**, the representation is not-sparse.

Model	Layer	MNIST	FASHIONMNIST	SVHN	CIFAR10
FF	1	3.69 ± 0.09	5.02 ± 0.14	10.3 ± 0.15	16.08 ± 0.09
	2	5.31 ± 0.35	18.46 ± 0.66	21.28 ± 0.23	21.2 ± 0.3
	3	1.36 ± 0.36	20.59 ± 0.63	4.48 ± 0.52	4.86 ± 0.51
BP/FF	1	8.58 ± 0.23	13.24 ± 0.31	15.07 ± 0.16	13.3 ± 0.13
	2	13.18 ± 0.67	8.45 ± 0.47	5.08 ± 0.19	5.55 ± 0.28
	3	-	-	-	-

Appendix G. Activation patterns in deeper layers

In subsection 4.1 we claimed that in **FF** and **BP/FF** the images of a given category activate consistently a small set of units that we named ensembles, that share similarities to what is observed in sensory cortices. We reported in Figure 1 the activation map for Layer 1 (the first hidden layer) of **FF** trained on the MNIST dataset, and observed that very sparse ensembles emerge. In this section we show, in a similar fashion, the representations for Layers 2 and 3 (Figure 4 and Figure 5, respectively). We found high sparsity also for deeper layers of this specific network; a qualitatively similar conclusion is reached for **FF** models trained on FASHIONMNIST, SVHN and CIFAR10. In **BP/FF** models a similar sparsity levels are observed, with the exception of the last layer that turns out to be not-sparse in each of the datasets we considered (see Table 4).

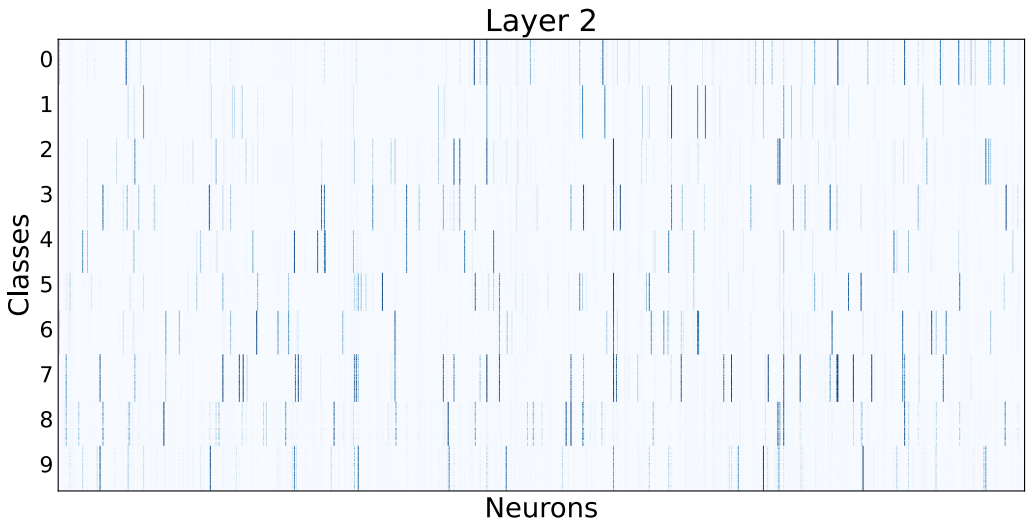

Figure 4: Activation patterns in a Multi-Layer Perceptron trained with the Forward-Forward algorithm, on the MNIST dataset. The image represents the activation map for neurons in Layer 2 for all images, grouped by class. A blue dot in position (x, y) indicates that neuron x is activated by input y; colorscale represents the intensity of such activation (incorrectly classified samples have been removed). Horizontal bands mark different categories; dark blue vertical lines mark active neurons. Each input category activates consistently a specific sets of neurons (ensemble). The sparsity measured according with the definition provided in subsection 3.4 is 0.84.

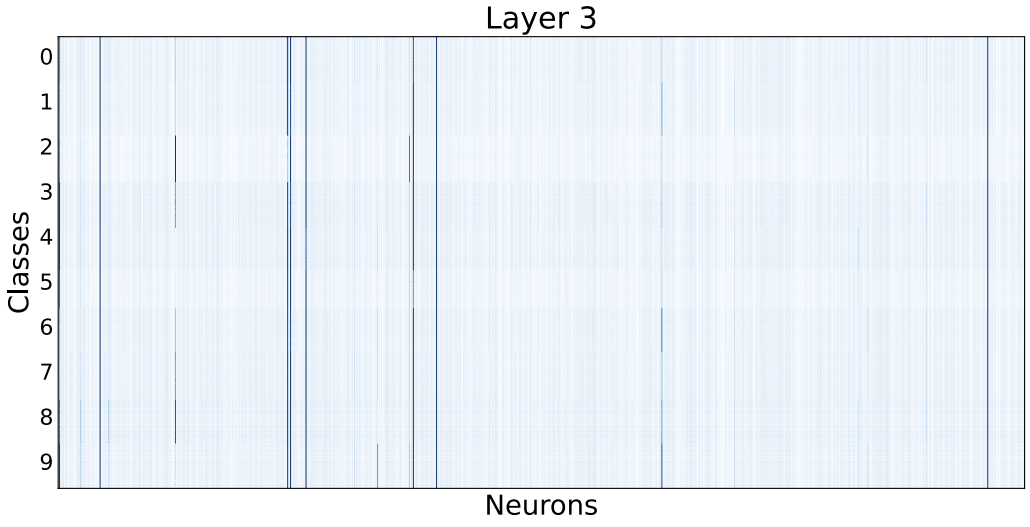

Figure 5: Activation reported as in Figure 4, for Layer 3. Notice that there are only few units that activate significantly and do not play a role in discriminating categories. The role of this layer, in this experiment, seems not related to the classification task. Despite the low number of active units, the sparsity level of the representation is lower than that of Layer 2 ($S = 0.67$), due to the noise of the inactive units.

Appendix H. Activation patterns in different models

In Figure 1 **C** we show an example of activation map in Layer 1 of **FF** trained on MNIST. For the purpose of a qualitative comparison, we show here analogous patterns for **BP/FF** and **BP** (see Figure 6 and Figure 7).

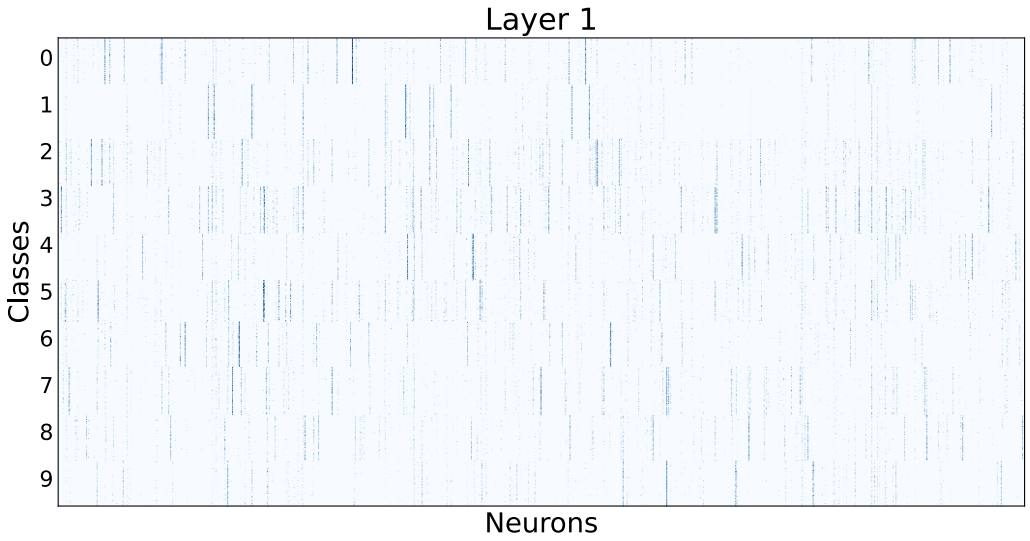

Figure 6: Activation pattern in Layer 1 of the **BP/FF** model trained on the MNIST dataset. The sparsity measure is 0.89, comparable with the correspondent first layer of the **FF** model.

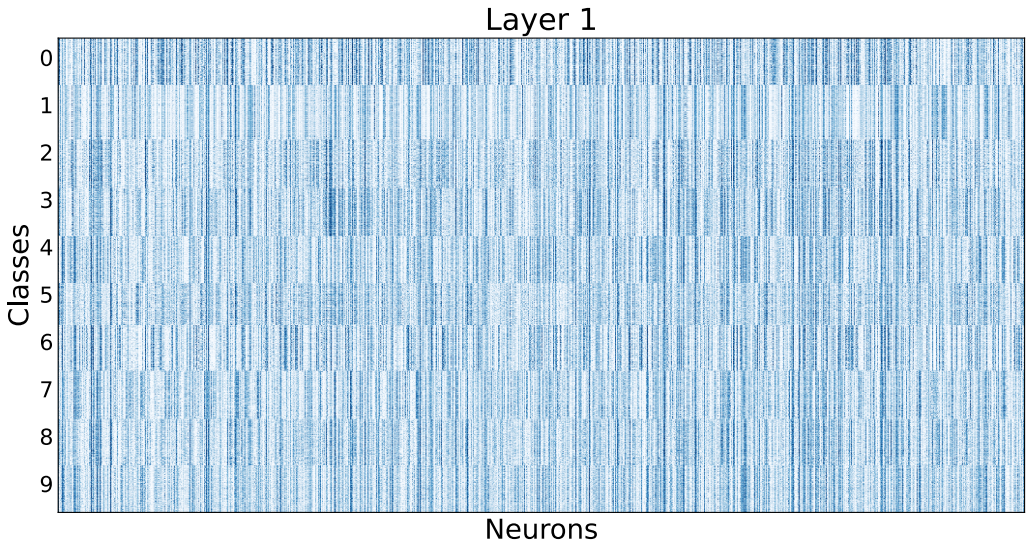

Figure 7: Activation pattern in Layer 1 of the **BP** model trained on the MNIST dataset. The sparsity measure is 0.32 (non-sparse representation), about $\frac{1}{3}$ of the sparsity level measured in the analogous experiment with **FF** and **BP/FF**.

Appendix I. Further results on representations of unseen categories and their ensembles

We showed in subsection 4.3 that a **FF** model trained on the FASHIONMNIST dataset – deprived of one category – can respond at test time to this unseen category with a valid ensemble (Figure 3). We report here the results of similar experiments, removing one category at a time. It turns out that, in each of the ten possible cases (we performed a single run for each category), the representations of the unseen category form a valid ensemble; we show three examples in Figure 8, different from the example shown in (Figure 3). It is with this situation in mind that we refer to "the ensembles related to unseen categories".

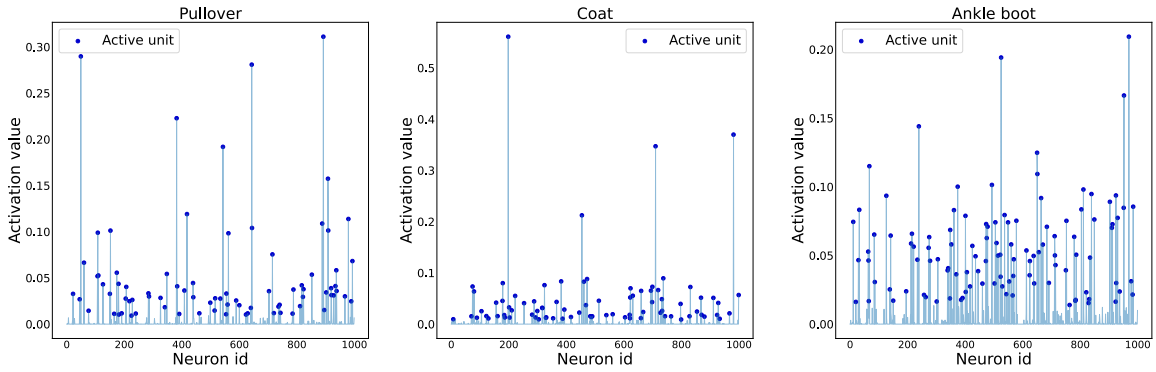

Figure 8: Ensembles elicited by the **FF** model trained on FASHIONMNIST deprived of one category (we show three examples: `Pullover`, `Coat` and `Ankle boot`). We report for the three categories, the activation value of each neuron in the first hidden layer (Layer 1), averaged on all images of the unseen category. Neuron index on the x axis; average activation on the y axis. Blue dots indicate units that are considered active according to the method described in subsection 3.4.

When an unseen category forms a valid ensemble, it generally exhibits a high level of integration with the ensembles associated with the categories encountered during training. This integration implies that it can share common units with ensembles belonging to related categories. We show in Figure 9 how the ensembles of missing categories (same examples as in Figure 8) integrate – by sharing units – with the other ensembles. Overall, these result relates to biological neural networks [25, 26], where ensembles appear to be the functional building block of brain representations even in the absence of known stimuli.

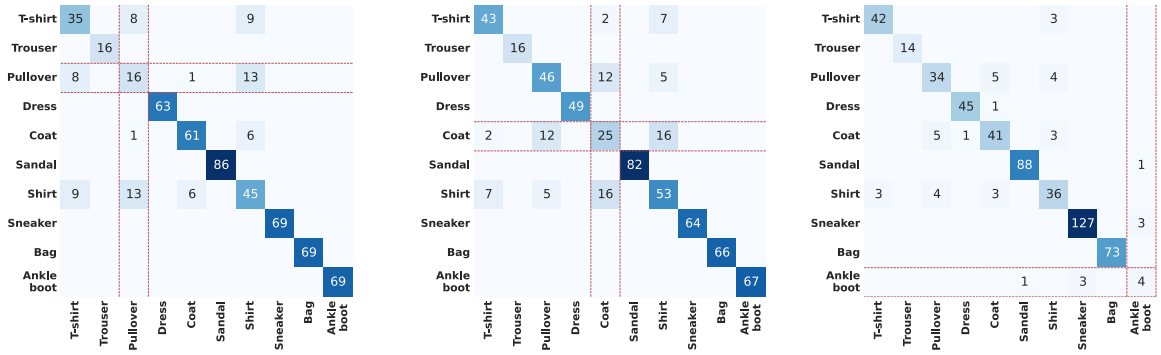

Figure 9: Shared units between the ensembles of unseen categories and the ensembles of categories seen during training (stripes delimited by the red lines). The results for `Pullover`, `Coat` and `Ankle boot` are shown.