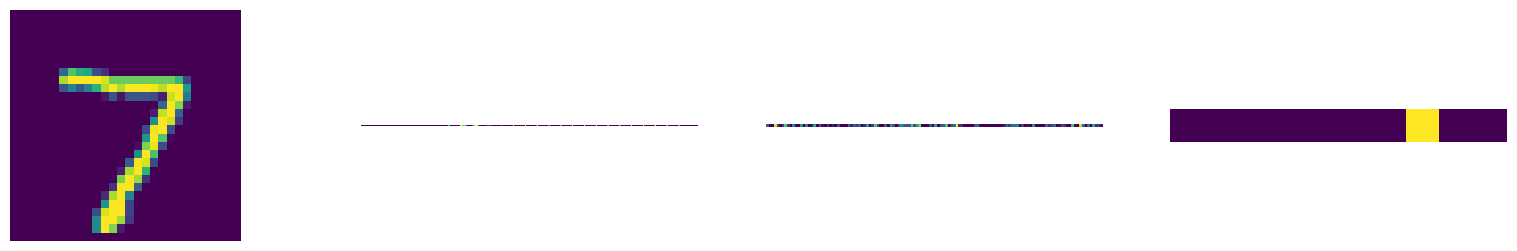
After Intensive prep assessment,  
Advised to follow Geometric analysis implementation

**Geometric analysis**

First deviated and tried methods such as

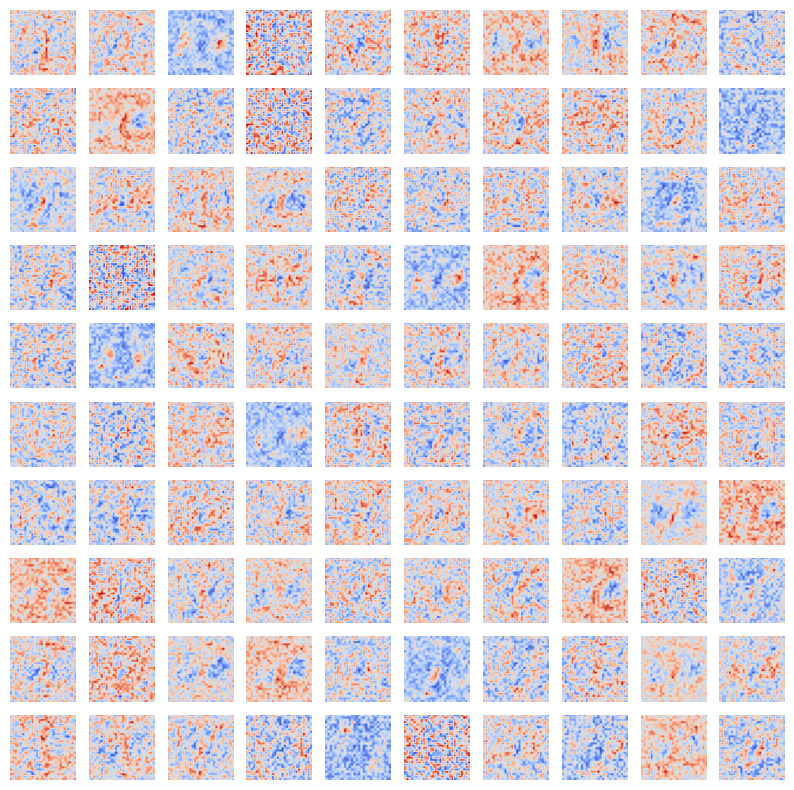
1. Visualising weights for all the dense layers, and looking for patterns → Failed implementation in Trial.ipynb



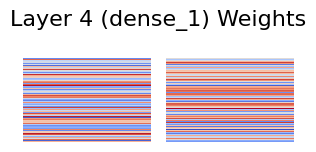
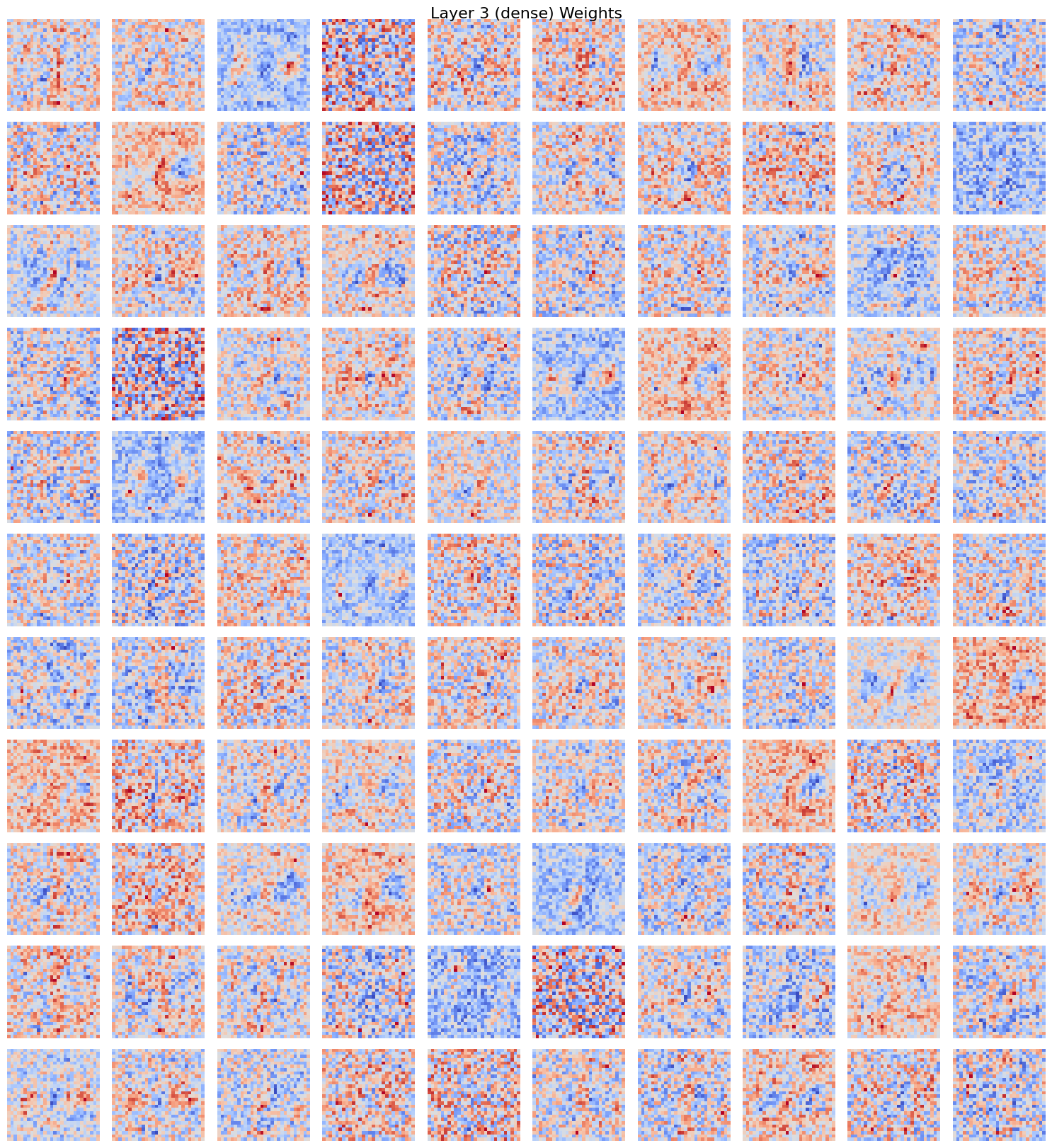


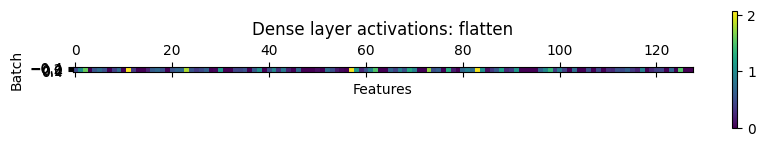
First Dense Layer Visualisation

Layer 2 Activations Visualisations



Layer 3 Activations Visualisations





Implementations in Trial\_Deep\_Layer\_Visualisation.ipynb

1. Tried Visualizing Convolutional Neural Networks its training

<https://github.com/far1din/manim?tab=readme-ov-file#visualizing-convolutional-neural-networks--training-a-cnn> - Accessed on : 20th Novembe, 2024

<https://www.youtube.com/watch?v=jDe5BAsT2-Y> - Accessed on : 20th Novembe, 2024

1. Found out statistical method that helps visualize high-dimensional data in two or three dimensions

### 

D. A. Senanayake, W. Wang, S. H. Naik and S. Halgamuge, "Self-Organizing Nebulous Growths for Robust and Incremental Data Visualization," in IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 10, pp. 4588-4602, Oct. 2021, doi: 10.1109/TNNLS.2020.3023941.

### GPU Optimizable:

These techniques generally benefit significantly from GPU acceleration, primarily due to the parallelization of computations such as distance calculations, matrix operations, and iterative optimization procedures.

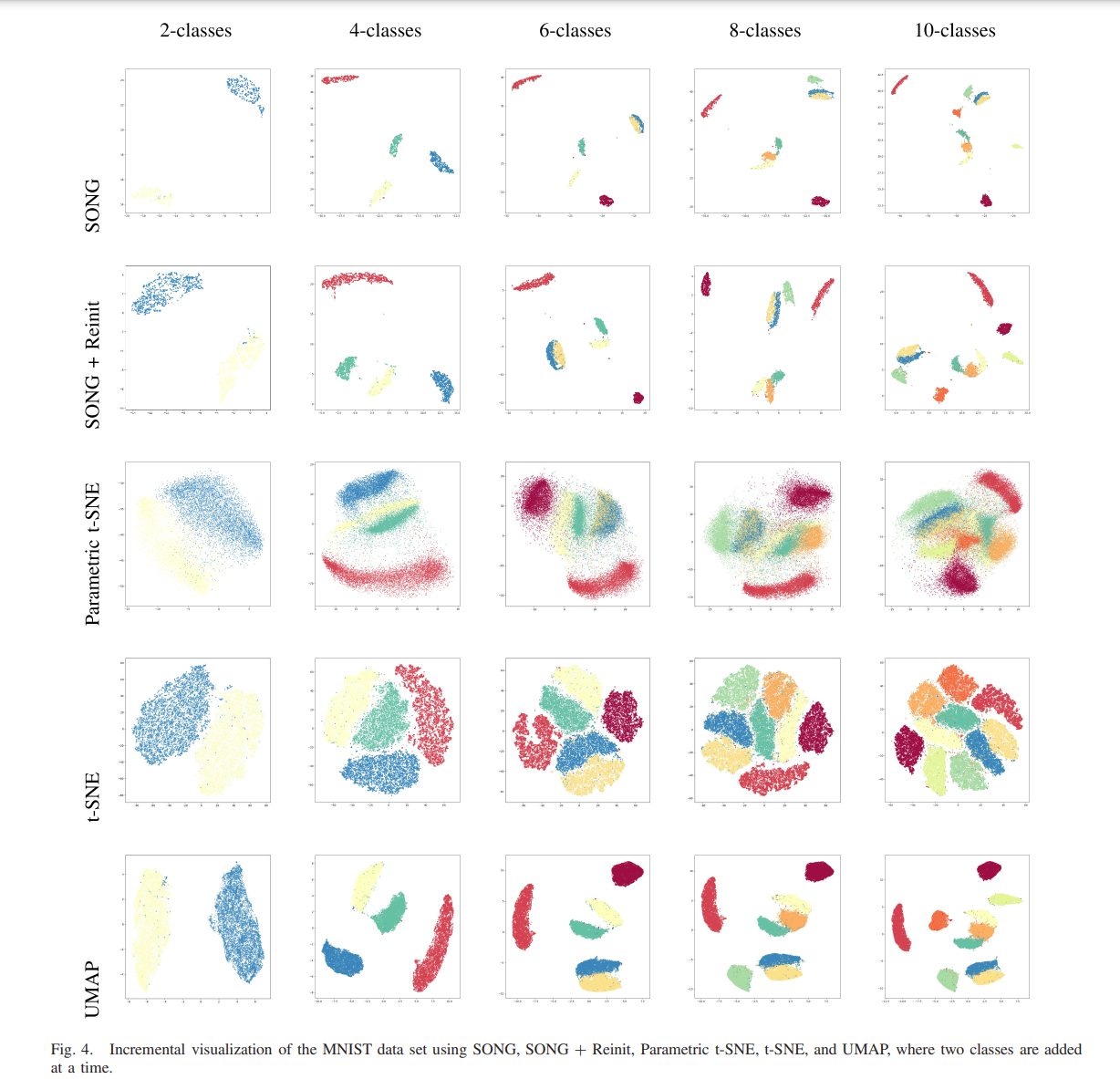
1. t-SNE
2. UMAP
3. SOMs (Self-Organizing Maps)
4. Spectral Embedding
5. SONGs (Self-Organizing Neural Graphs)
6. SONGs + Reinit
7. MDS (Multidimensional Scaling)

### Less Commonly GPU Optimizable:

These techniques either do not typically utilize GPU acceleration in standard implementations or their typical computations do not parallelize as effectively on GPUs.

1. IsoMAP - While it could theoretically benefit from GPU acceleration, standard implementations usually don't employ it.
2. NCA (Neighborhood Components Analysis) - Some implementations might use GPUs for parts of the computation, but it's less common.
3. LDA (Linear Discriminant Analysis) - While matrix operations can be accelerated, the standard applications of LDA often do not require the heavy computational resources that benefit from GPUs.

Implementation of MDS in MDS\_in\_Colab.ipynb



Tao, X., Chang, X., Hong, X., Wei, X., Gong, Y. (2020). Topology-Preserving Class-Incremental Learning. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, JM. (eds) Computer Vision – ECCV 2020. ECCV 2020. Lecture Notes in Computer Science(), vol 12364. Springer, Cham. <https://doi.org/10.1007/978-3-030-58529-7_16>

Y. Yang et al., "Learning Adaptive Embedding Considering Incremental Class," in IEEE Transactions on Knowledge and Data Engineering, vol. 35, no. 3, pp. 2736-2749, 1 March 2023, doi: 10.1109/TKDE.2021.3109131.

keywords: {Data models;Task analysis;Labeling;Streaming media;Training;Prototypes;Manuals;Class-incremental learning;Novel class detection;incremental model update},