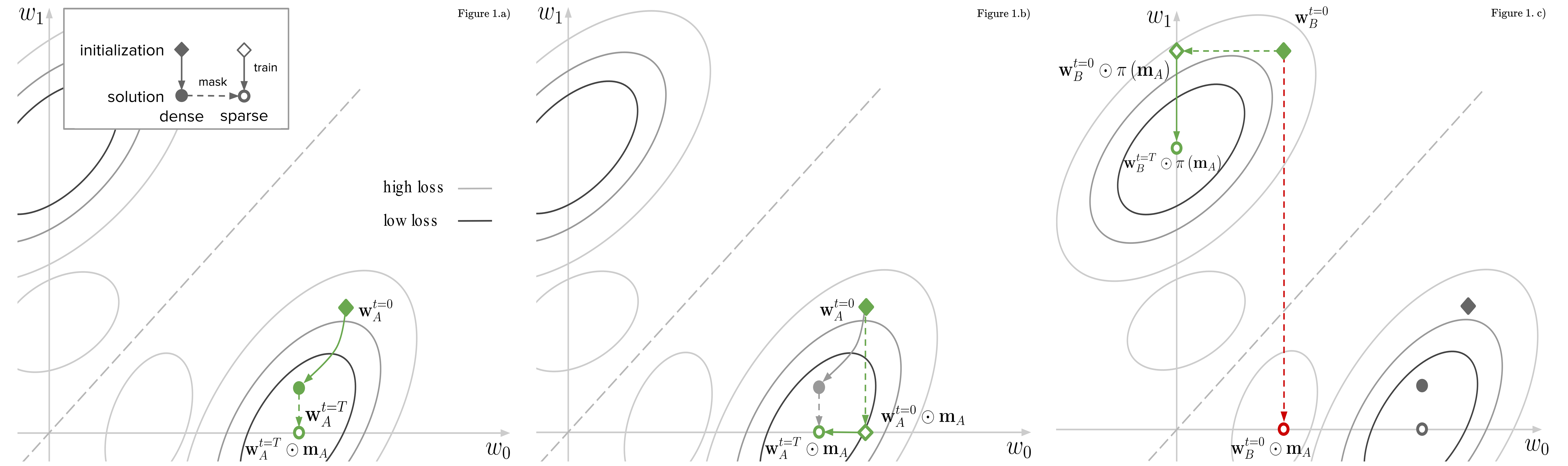
# Sparse Training from Random Initialization: Aligning Lottery Ticket Masks using Weight Symmetry

Mohammed Adnan[*,1,3], Rohan Jain[*,1], Ekansh Sharma[2,3], Yani Ioannou[1]
University of Calgary[1] University of Toronto[2] Vector Institute[3]
* Equally contributed 1st authors




In Figure 1.a), a dense random initialization, $\mathbf{w}_A^{t=0}$, converges to a dense solution, $\mathbf{w}_A^{t=T}$, which is then pruned using weight magnitude resulting in the mask $\mathbf{m}_A = (1, 0)$.
In Figure 1.b), we demonstrate the LTH: re-use the init, $\mathbf{w}_A^{t=0}$, to train model $A$ with the pruned mask, $\mathbf{m}_A$.
In Figure 1.c), permuting the mask, $\pi(\mathbf{m}_A)$, to match the (symmetric) basin in which the new initialization, $\mathbf{w}_B^{t=0}$, is in will enable sparse training.

## Background

1. **Lottery Ticket Hypothesis (LTH):** identifies sparse sub-networks that, when trained independently, *can* match dense model performance. [1]
2. NNs are **permutation invariant**: swapping neurons in a layer doesn't change the function underlying they compute.
3. Git Re-Basin showed that NN loss landscapes nearly contain a **single** solution basin *modulo permutations*. [2]

## Motivation

- Motivated by the goal of training a sparse model from a **truly** random init.
  - [1] demonstrated that training with a highly sparse is mask possibly, proposing the LTH.

The **key** limitation of LTH: a dense model must be first trained to get a mask, which is *only* usable with its original random init.

- Obtaining winning tickets requires *rewinding* — requiring significantly more compute.
- Lottery tickets do **not** generalize well to new random init's.

We seek to answer:

- **How can we train a LTH mask from a different random init. while maintaining good generalization?**

## Our Findings

To reuse an LTH winning ticket mask with a truly different random init. ...

We leverage permutation symmetries, to **permute** the mask to align with the new random init's optimization basin.

- We find that a sparse model (with the permuted mask) can *nearly* match generalization performance of the LTH solution.
- We show for a fixed init., the dense solution and corresponding LTH solution reside within the same loss basin when variance collapse is considered. This conclusion presents a new perspective compared to the work of [3].
- Models trainied from random init. using the permuted mask are more functionally diverse in the solutions they learn vs. LTH.
- We empirically demonstrate this on CIFAR-10/100 and ImageNet with VGG11 and ResNet models of varying widths.

## Methodology

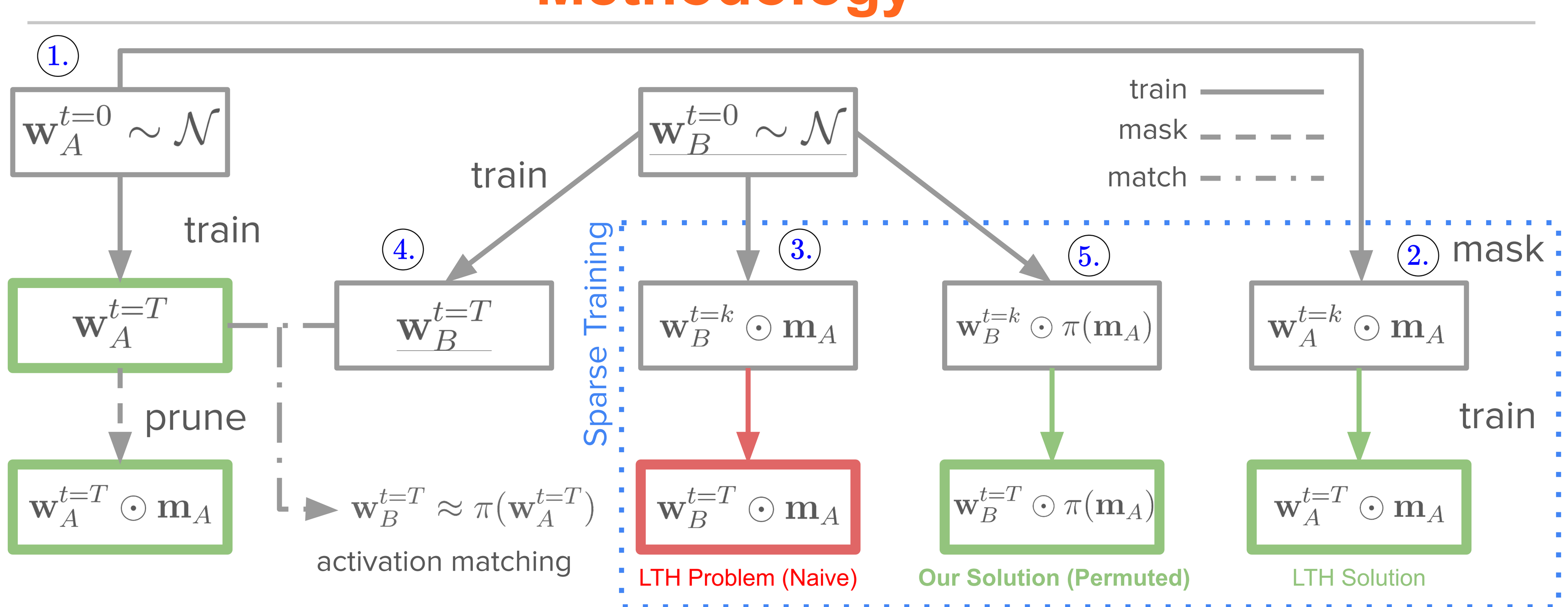



Figure 1. The overall framework of our training procedure.

## Results: ResNet50/ImageNet + VGG11/CIFAR-10

- **VGG11**: increasing the rewind point, the permuted solution closely matches the accuacy of LTH, while naive solution significantly plateaus.
- **ResNet50**: permuted solution beats the naive solution across all sparsity levels, validating our hypothesis on large datasets.

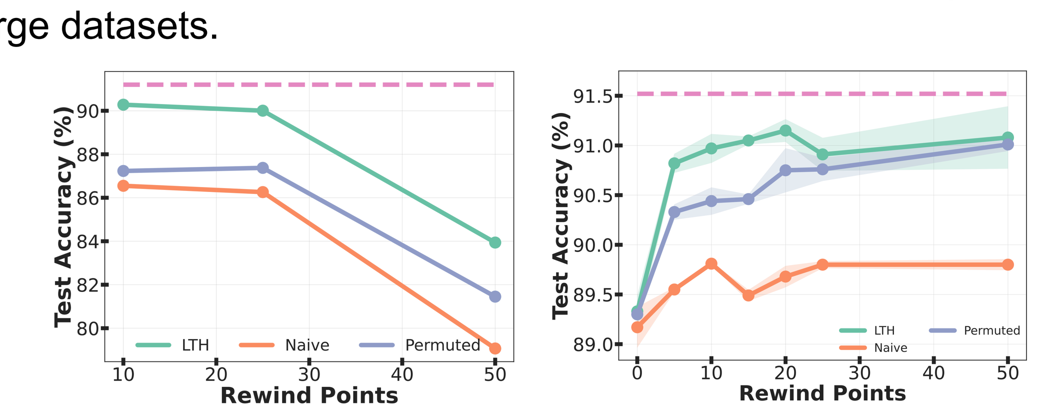



Figure 2. Permuted solutions behaviour over ResNet50/ImageNet & VGG11/CIFAR-10 at 90% sparsity, with width = 1.

## Results: ResNet20/CIFAR-10

- Permuted solution outperforms the naive solution. As sparsity increases, training becomes harder, widening the gap between permuted and naive solutions.
- Both the LTH & permuted solution do not perform well at a truly random init (k = 0) but **improves** on increasing the rewind point until plateauing.
- As width increases, the gap between training from random init. with the permuted mask & the LTH/ dense baseline decreases, unlike training with the naive mask.

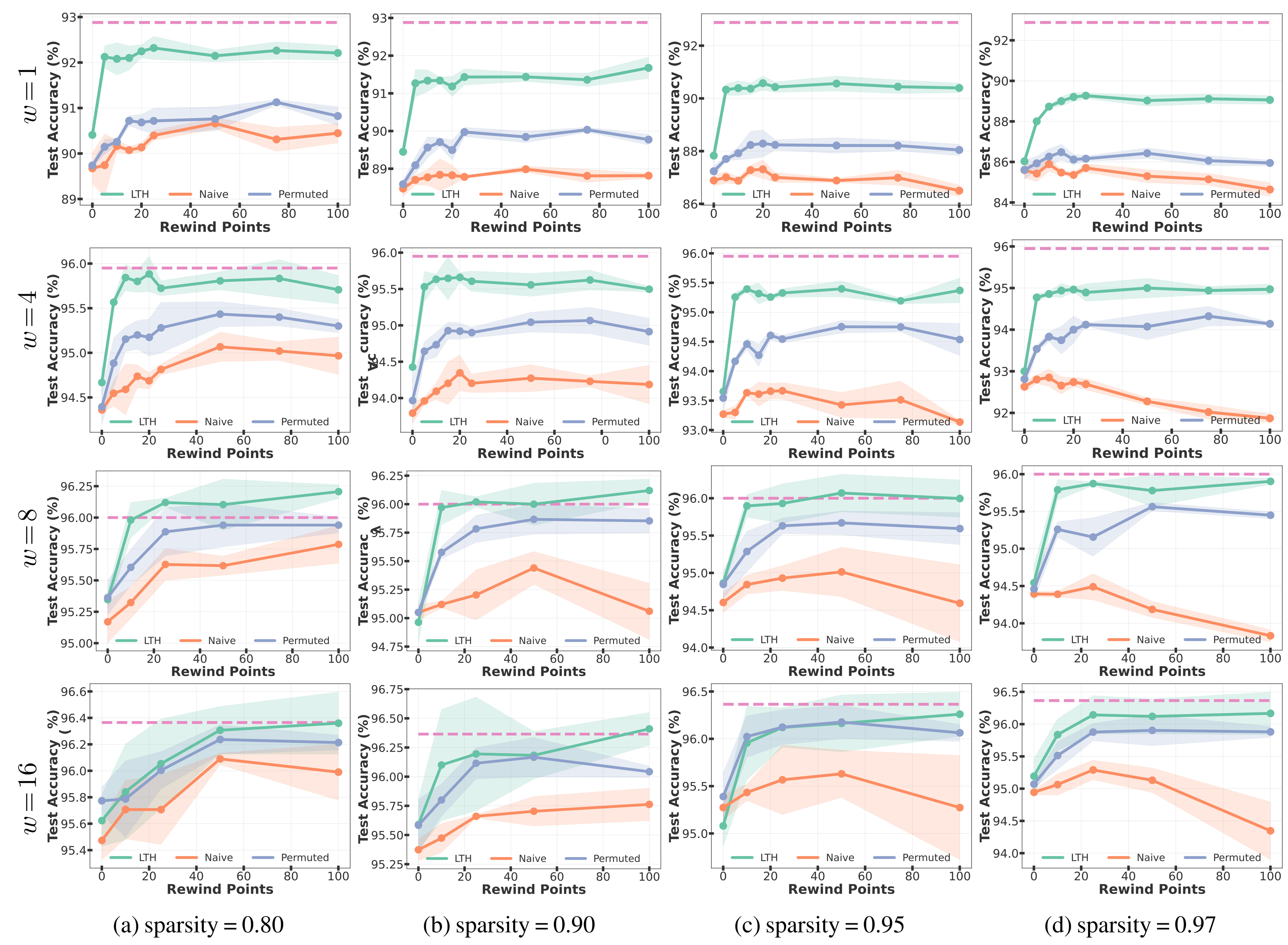



Figure 3. Test accuracy of sparse networks solutons vs. increasing rewind points for different sparsity levels and widths, $w$.

## Effect of Model Width

- Larger width exhibits better linear mode connectivity (LMC). As the width of the model increases, the permutation matching algorithm gets more accurate, thereby reducing the loss barrier.
- This leads to an improvement in performance of our permuted solution.

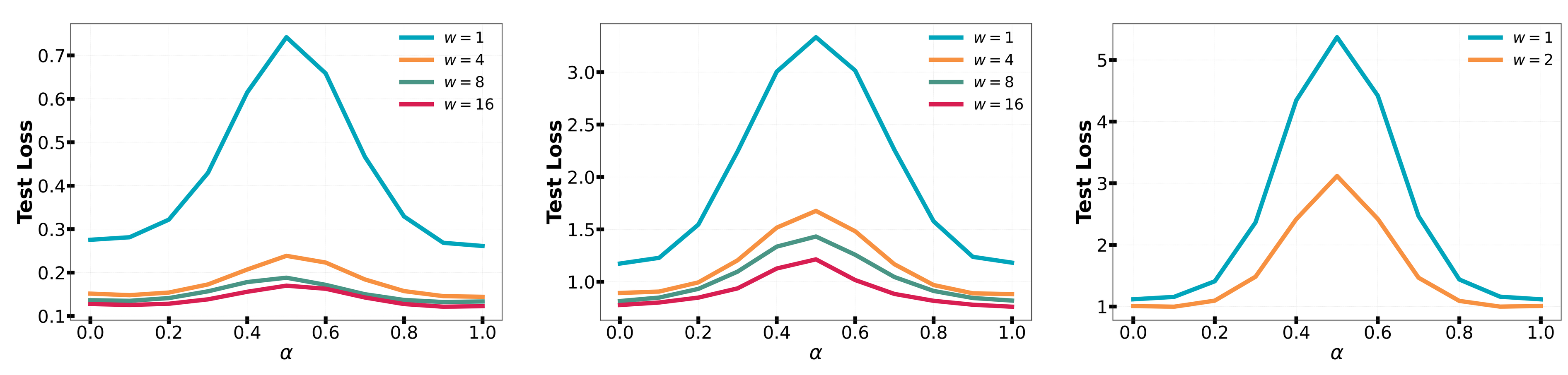



Figure 4. Plots showing the linear interpolation between $\pi(\mathbf{w}_A^{t=T})$ and $\mathbf{w}_B^{t=T}$ for various widths.

## Ensemble Diversity & Loss Landscape Analysis

- A limitation of LTH: consistently converges to very similar solutions to the original pruned model, effectively relearning the same solution. [4]
- Although the mean test acc. of LTH is higher, ensemble of permuted models acheives better test acc. due to better functional diversity of permuted models.
- We also show, *modulo permutations* reusing the permuted mask leads to convergence in the same mode as the LTH solution.

| Mask | Test Accuracy (%) | Ensemble Acc. (%) | Disagree-ment | KL | JS |
|---|---|---|---|---|---|
| ResNet20×{4}/CIFAR-100 | | | | | |
| none (dense) | $78.37 \pm 0.059$ | - | - | - | - |
| IMP | $74.46 \pm 0.321$ | 79.27 | 0.259 | 1.005 | 0.372 |
| LTH | $\mathbf{75.35 \pm 0.204}$ | 75.99 | 0.117 | 0.134 | 0.038 |
| permuted | $72.48 \pm 0.356$ | **77.85** | 0.278 | 0.918 | 0.327 |
| naive | $71.05 \pm 0.366$ | 76.15 | **0.290** | **0.970** | **0.348** |

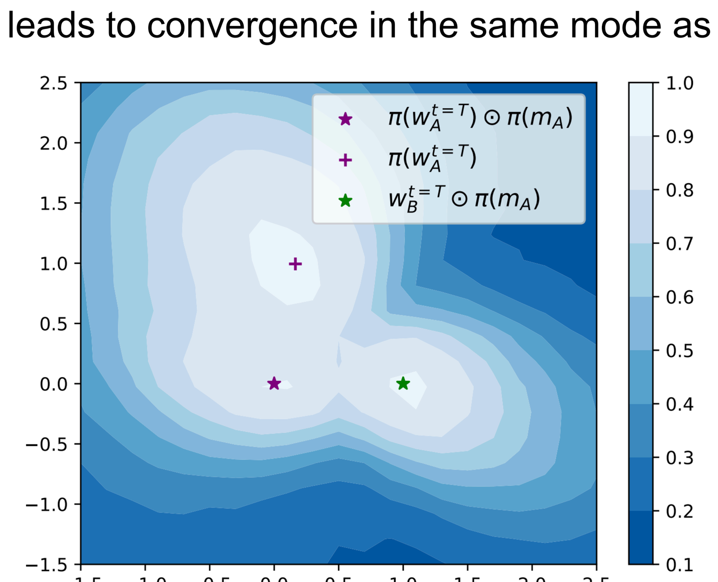


Figure 5. (left): Various measures of function space similarity between the models. (right): 0-1 loss landscape of ResNet20x{4}/CIFAR-100.

[1] Frankle et al. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks, 2019.
[2] Ainsworth et al. Git re-basin: Merging models modulo permutation symmetries, 2023.
[3] Paul et al. Unmasking the lottery ticket hypothesis: What's encoded in a winning ticket's mask?, 2023.
[4] Evci et al. Gradient flow in sparse neural networks and how lottery tickets win, 2022.