

## A. Hyperparameter tuning

For CNN Zoo 1, we used the hyperparameter sampling listed below.

- Number of layers: uniformly from range [2, 8].<sup>3</sup>
- Hidden size: uniformly from range [256, 512].
- Dropout rate: uniformly from range [0.0, 0.2].
- Weight decay: log-uniformly from range [1e-8, 1e-3].
- Optimizers: uniformly from [adam, adamw, adamax, nadam, radam].
- Batch size: uniformly from [64, 128, 256, 512].
- Initializer: uniformly from [xavier, he, orthogonal, original] (all from the uniform distribution (default in PyTorch).

For CNN Zoo 2, we used the following hyperparameter sampling.

- Number of layers: uniformly from range [2, 8].
- Hidden size: uniformly from range [64, 512].
- Dropout rate: uniformly from range [0.0, 0.7].
- Weight decay: log-uniformly from range [1e-5, 1e-2].
- Optimizers: uniformly from [adam, adamw, adamax, nadam, radam].
- Batch size: uniformly from [64, 128, 256, 512].
- Initializer: uniformly from [xavier, he, orthogonal, original] (all from the uniform distribution (default in PyTorch).

Note that we significantly increase both dropout and weight decay to mitigate overfitting in the regular FC network.

## B. CNN Zoo 1 Performance

### C. Raw weights visualization

See Figure 6 on next page.

### D. Similarity matrices

Architecture	Parameters	Test loss	Test $R^2$
FC network	3538509	0.000171	0.9866
FC <sub>Linear</sub>	1774519	0.000168	0.9867
ComVeX <sub>Linear</sub>	1623967	0.000169	0.9869
ComVeX <sub>Conv</sub>	458153	0.000167	0.9868

Table 6. Predictors' performance on CNN Zoo 1 CIFAR-10.

Architecture	Parameters	Test loss	Test $R^2$
FC network	3848853	0.000558	0.993
FC <sub>Linear</sub>	2017377	0.000564	0.9929
ComVeX <sub>Linear</sub>	842399	0.000505	0.9935
ComVeX <sub>Conv</sub>	966321	0.000513	0.9936

Table 7. Predictors' performance on CNN Zoo 1 Fashion-MNIST.

Architecture	Parameters	Test loss	Test $R^2$
FC network	2817777	0.000658	0.9944
FC <sub>Linear</sub>	1956661	0.000586	0.995
ComVeX <sub>Linear</sub>	1397399	0.000521	0.9955
ComVeX <sub>Conv</sub>	1339361	0.000533	0.9955

Table 8. Predictors' performance on CNN Zoo 1 MNIST.

Architecture	Parameters	Test loss	Test $R^2$
FC network	3675685	0.000265	0.9834
FC <sub>Linear</sub>	2054869	0.000269	0.9832
ComVeX <sub>Linear</sub>	1192327	0.000206	0.9871
ComVeX <sub>Conv</sub>	950801	0.000203	0.9875

Table 9. Predictors' performance on CNN Zoo 1 SVHN.

<sup>3</sup>This does not include the first layer for the proposed architectures, and we add one to the number of layers for the regular fully-connected networks during initialization for fair comparison.

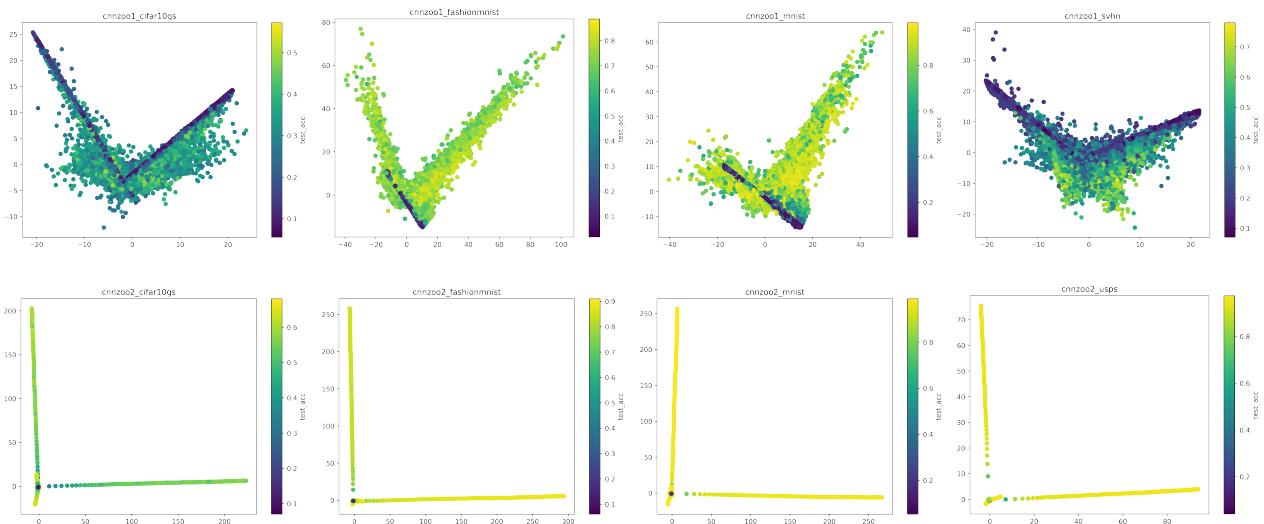


Figure 6. A 2-d visualization of the raw neural network weights in CNN Zoo 1 (top) and CNN Zoo 2 (bottom). Networks with similar performance align inconsistently on lines instead of together as a cluster.

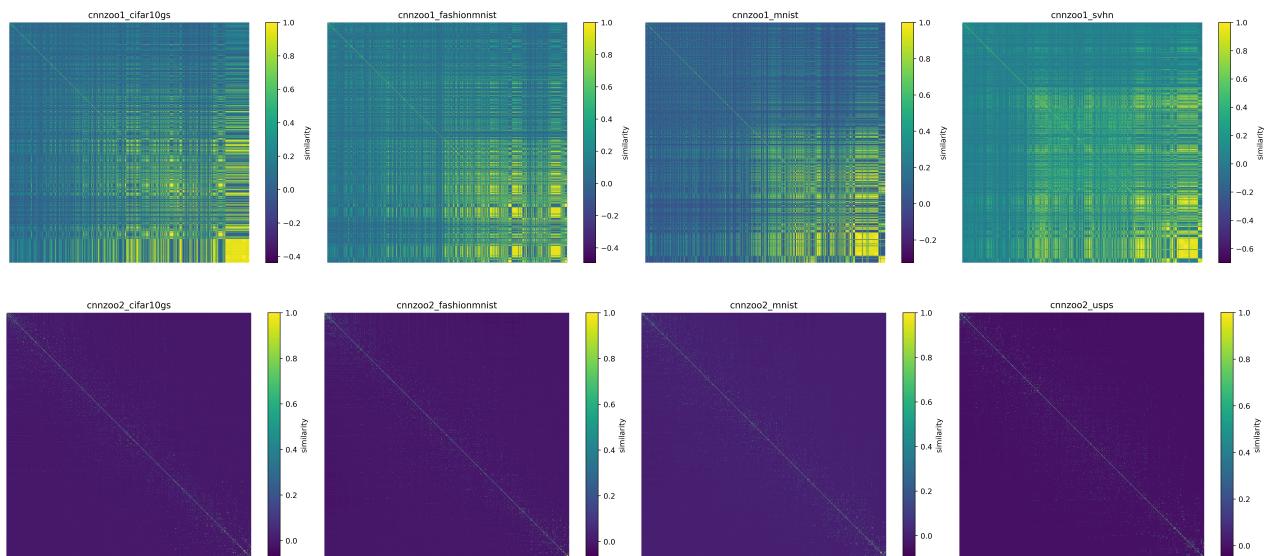


Figure 7. Similarity matrices of raw weights. On both axes, models are sort by their accuracy in descending order (from left to right for x, or from top to bottom for y). Lighter means higher similarity.

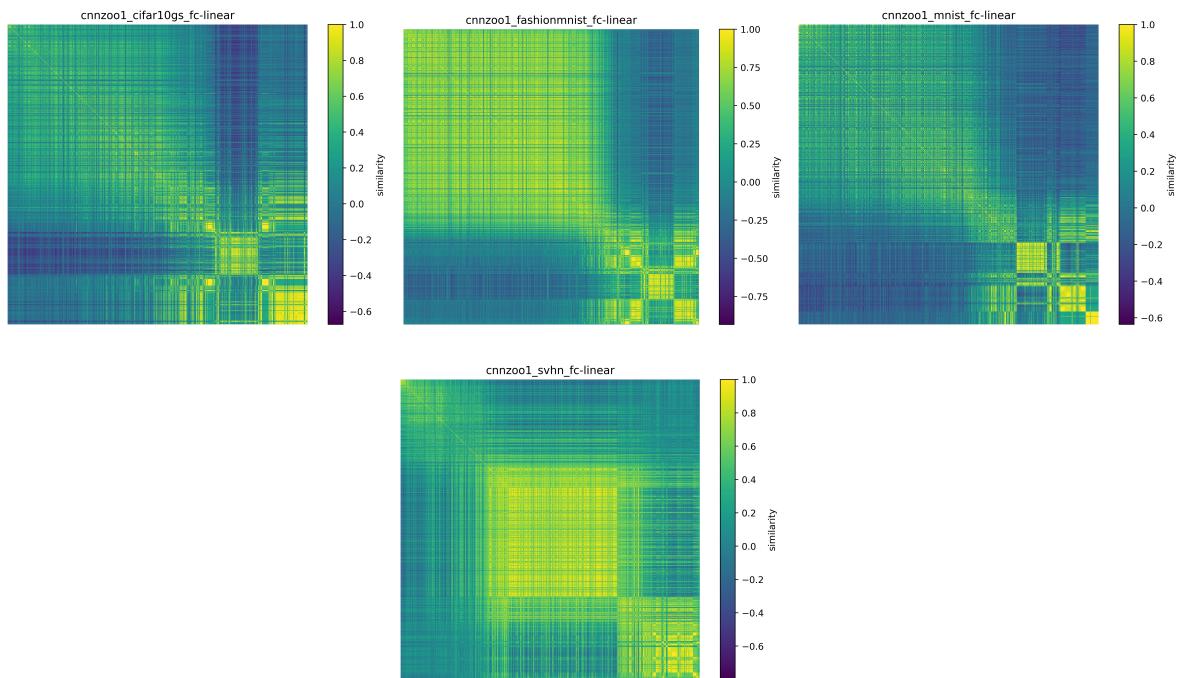


Figure 8. CNN Zoo 1 similarity matrices

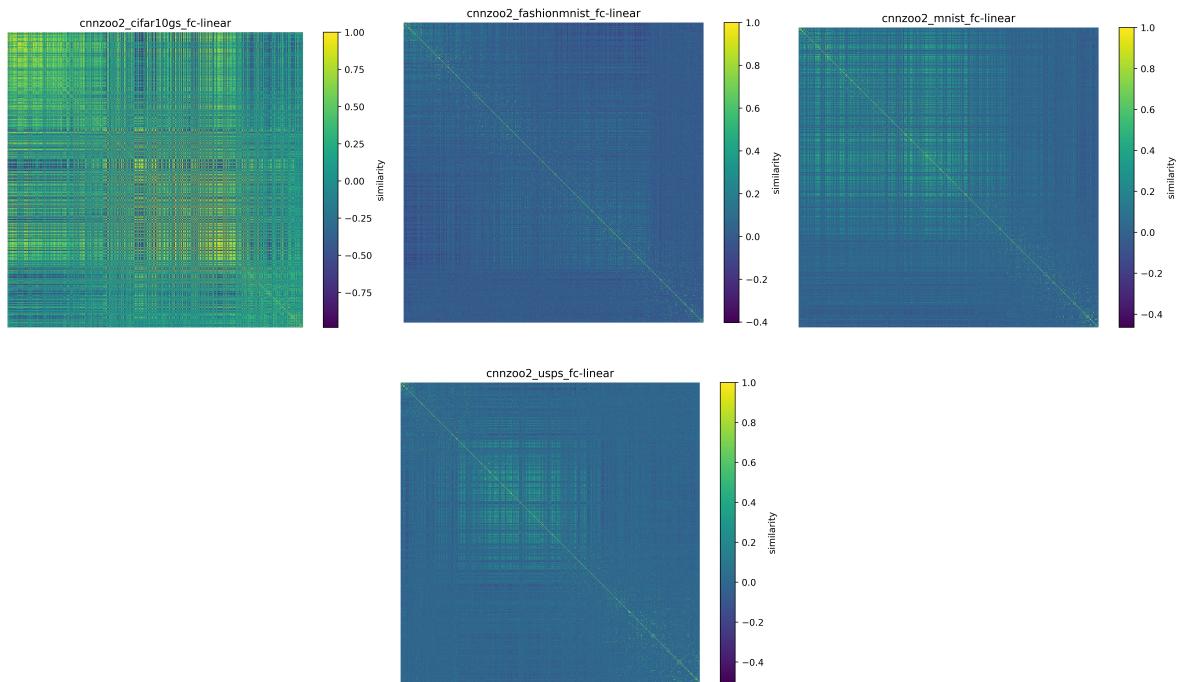


Figure 9. CNN Zoo 2 similarity matrices

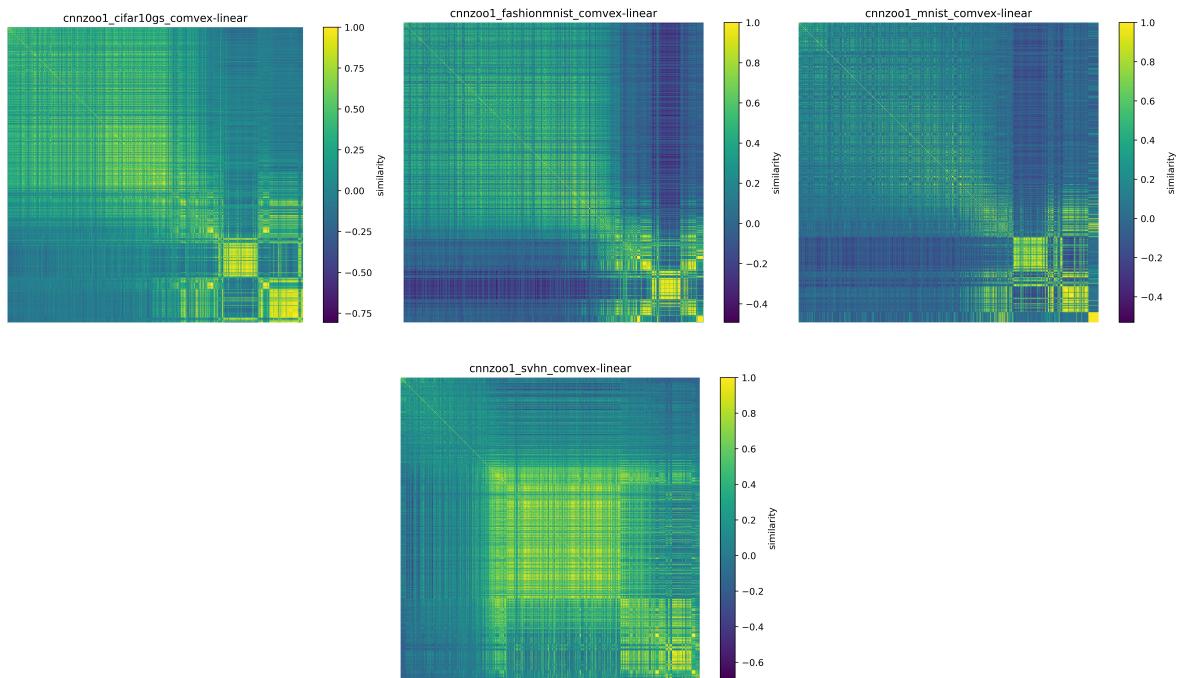


Figure 10. CNN Zoo 1 similarity matrices

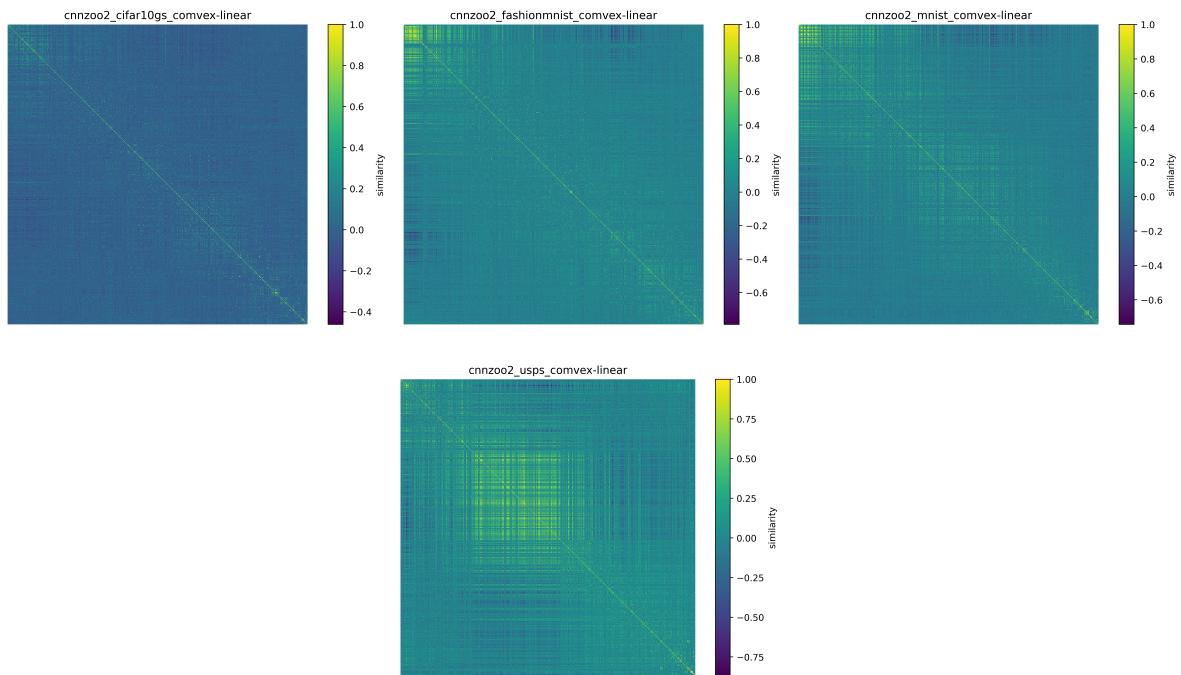


Figure 11. CNN Zoo 2 similarity matrices

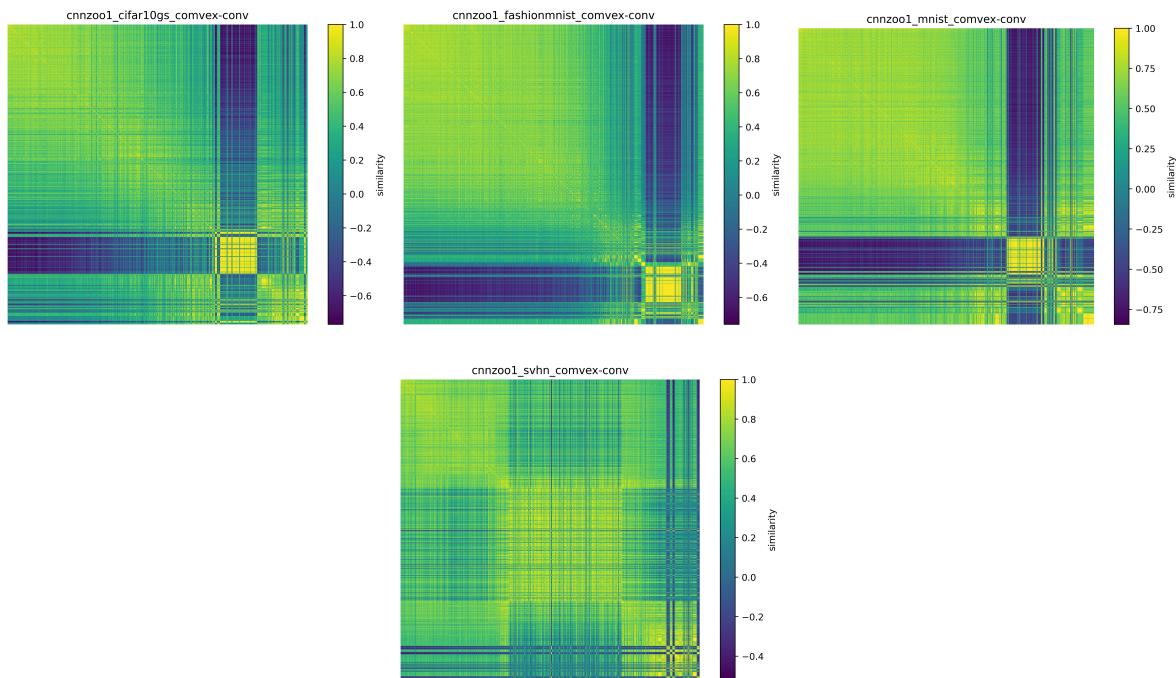


Figure 12. CNN Zoo 1 similarity matrices

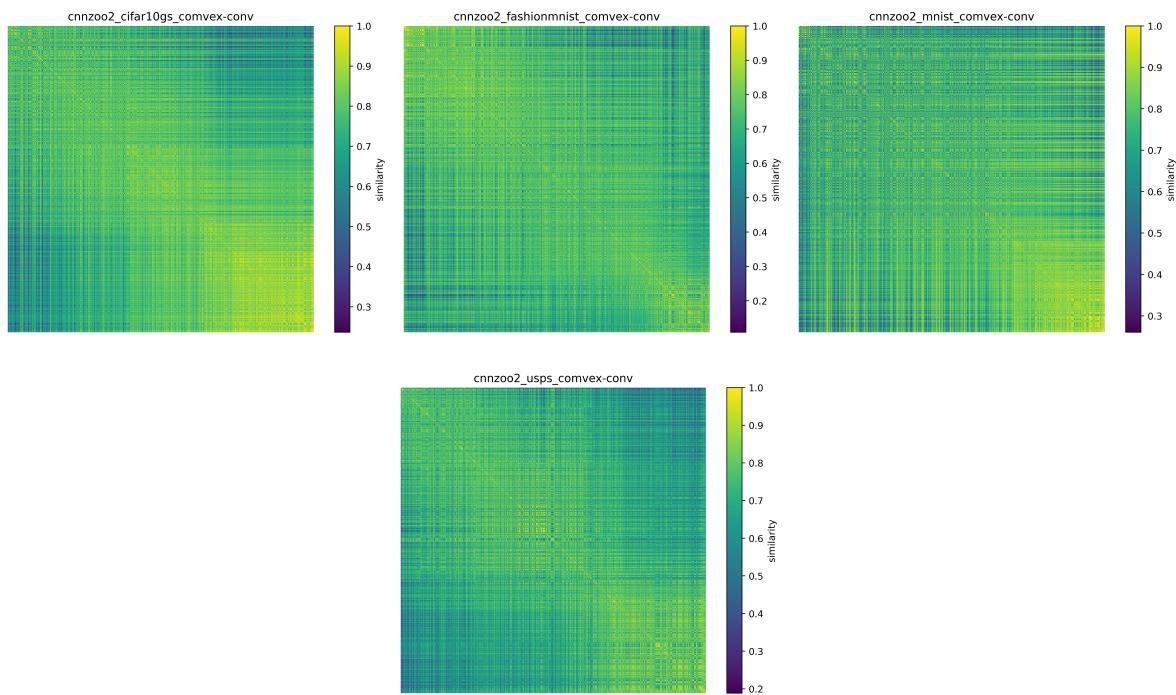


Figure 13. CNN Zoo 2 similarity matrices