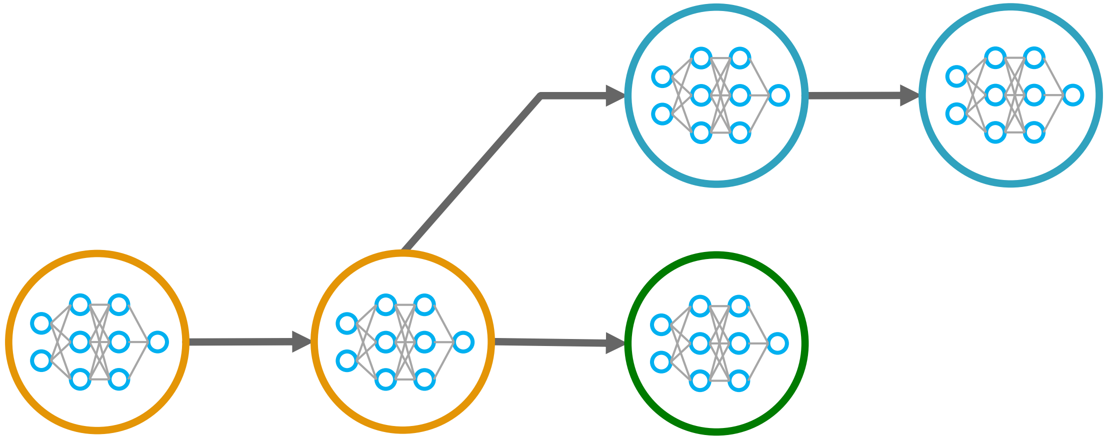


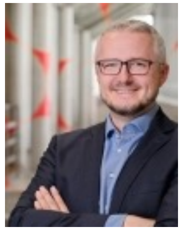
# Self-Supervised Representation Learning on Neural Network Weights for Model Characteristic Prediction


Konstantin Schürholt, Dimche Kostadinov, Damian Borth

AI:ML Lab, School of Computer Science

University of St. Gallen



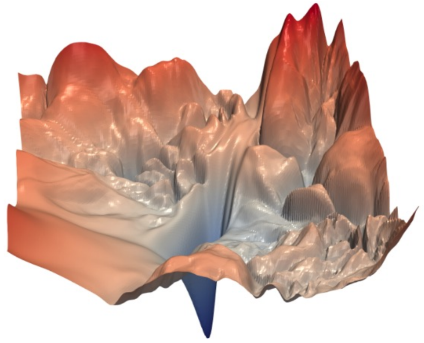



# Problem

## Neural Networks are successfully applied on multiple domains

## Loss surface and optimization problem of Neural Networks are highly non-convex

Goodfellow, Vinyals, Saxe; ICLR 2015; *Qualitatively characterizing neural network optimization problems*
Dauphin et al.; NeurIPS 2014; *Identifying and attacking the saddle point problem in high-dimensional non-convex optimization*
LeCun, Bengion, Hinton; Nature 2015; *Deep Learning*



Li et al.; NeurIPS 2018; *Visualizing the Loss Landscape of Neural Nets*

## Neural Network training optimization is high dimensional

Brown et al.; 2020; *Language Models are Few-Shot Learners*
Larsen et al.; ICML 2021; *How many degrees of freedom do we need to train deep networks: a loss landscape perspective*

## Neural Network training is sensitive to hyperparameters and random initialization

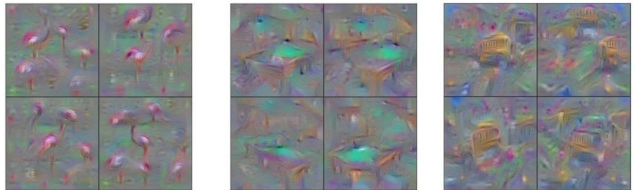
Hanin, Rolnick; NeurIPS 2018; *How to Start Training: The Effect of Initialization and Architecture*

**Relation between characteristics of NN models and their solution in weight space not fully understood**

# Related Work

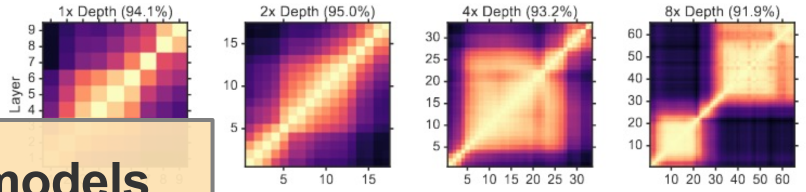
## Visualization of CNN kernels

Yosinski et al.; ICML DL Workshop 2015; *Understanding Neural Networks Through Deep Visualization*
Zintgraf, Cohen, Adel, Welling, ICLR 2017; *Visualizing Deep Neural Network Decisions: Prediction Difference Analysis*



## Comparing Neural Network models

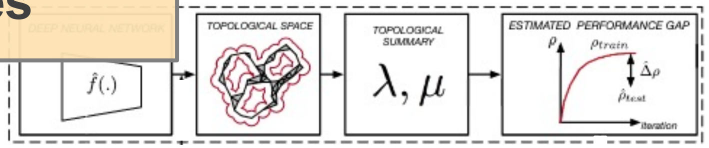
Raghu et al.; NeurIPS 2017; *SVCCA: Singular Vector Canonical Correlation Analysis for Deep Learning Dynamics and Interpretability*
Kornblith et al.; ICML 2019; *Similarity of Neural Network Representations Revisited*
Mehrer et al.; Nature 2020; *Individual Differences among Deep Neural Network Models*




- investigate/compare only single/pairs of models
- rely on expressivity of data
- supervised learning may overfit few features

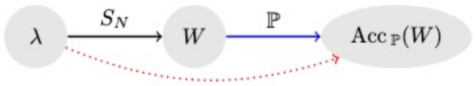
## Activation-based Prediction of Neural Network properties

Yak et al.; ICML 2019; *Towards Task and Architecture-Independent Generalization Gap Predictors*
Jiang, Krishnan, Mobahi and Bengio; ICLR 2019; *Predicting the Generalization Gap in Deep Networks with Margin Distributions*
Corneanu et al.; CVPR 2020; *Computing the Testing Error Without a Testing Set*
Mellor et al.; ICML 2021; *Neural Architecture Search Without Training*



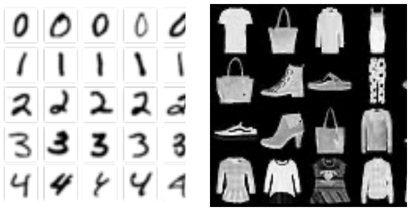

## Prediction of Neural Network properties from weights

Martin and Mahoney; ICML 2019; *Traditional and Heavy-Tailed Self Regularization in Neural Network Models*
Unterthiner et al.; 2020; *Predicting Neural Network Accuracy from Weights*
Eilertsen et al; ECAI 2020; *Classifying the Classifier*

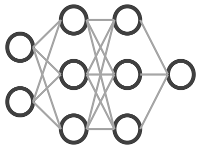


# Investigating Populations of NN Models

**Dataset**



**Architecture**

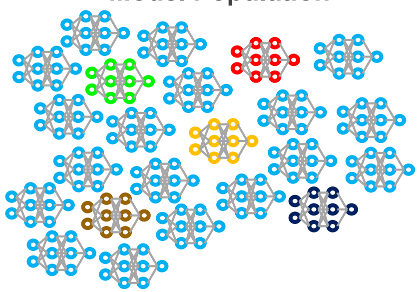


**Hyperparamerters**

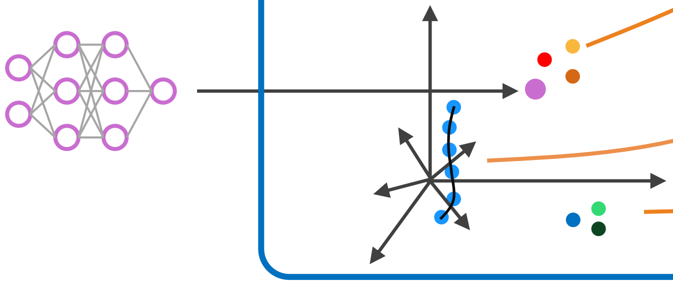
- Optimizer
- Activation
- Initialization Method
- Learning Rate
- L2-Regularization

**Hypothesis:**
1. Neural Networks populate a structure in weight space
2. That structure contains information on properties and generating factors of the models

**Model Population**



**Representation Space**

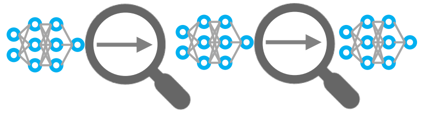


**Goal:** Learn meaningful representations of populations of Neural Network models

**Model Analysis**

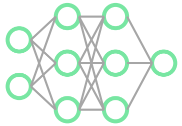
versioning, diagnostics, ...

**Learning Dynamics**



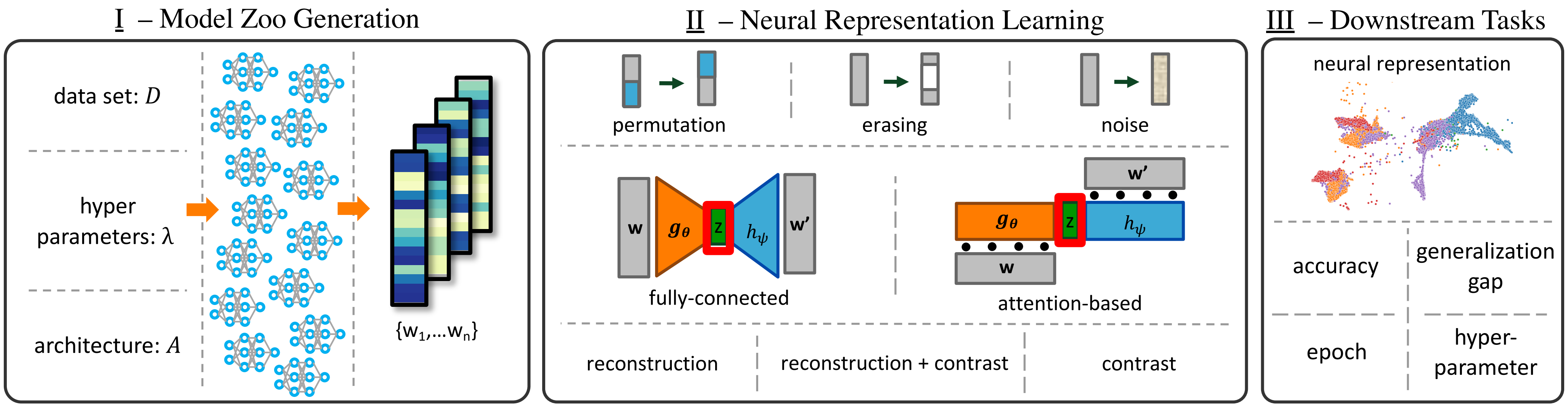
early-stopping, model selection, ...

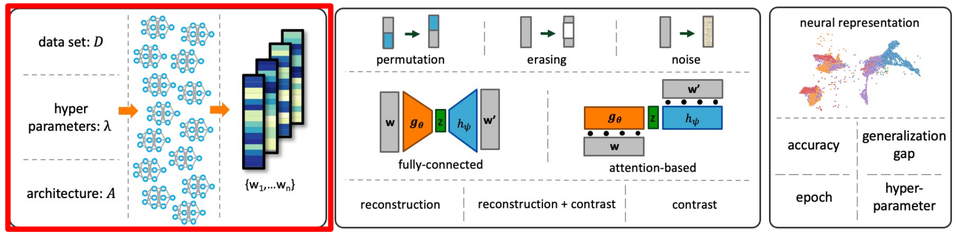
**Model Generation**



initialization,
transfer-learning, meta-learning, ...

# Approach

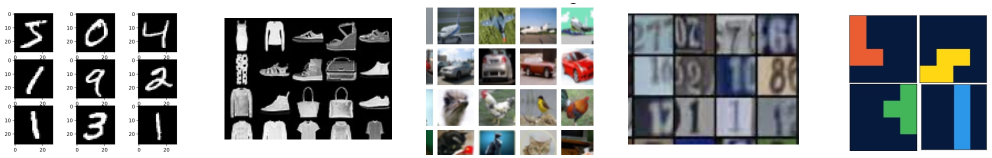

I – Model Zoo Generation
data set: $D$
hyper parameters: $\lambda$
architecture: $A$
$\{w_1,\ldots w_n\}$
II – Neural Representation Learning
permutation
erasing
noise
w
$g_\theta$
z
$h_\psi$
w'
fully-connected
attention-based
reconstruction
reconstruction + contrast
contrast
III – Downstream Tasks
neural representation
accuracy
generalization gap
epoch
hyper-parameter




# Approach: Model Zoos

## Datasets:

- MNIST, Fashion-MNIST, SVHN, CIFAR, Tetris



## Architectures

- **MLP:** 100 parameters (ours)
- **CNN:** 2464 paramters (ours)
- **CNN:** 4970 paramters (Unterthiner et al., 2020)

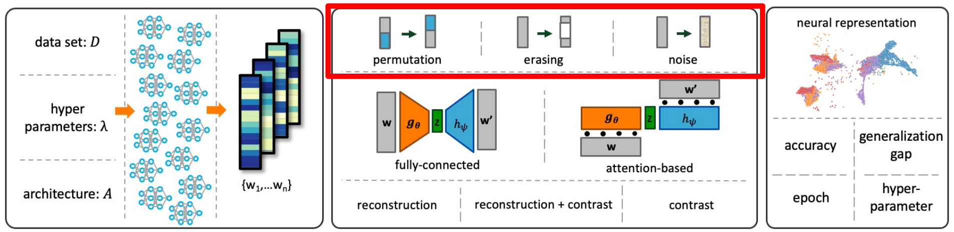
## Hyperparamters

- Seed, activation, initialization method, learning rate, regularization, …

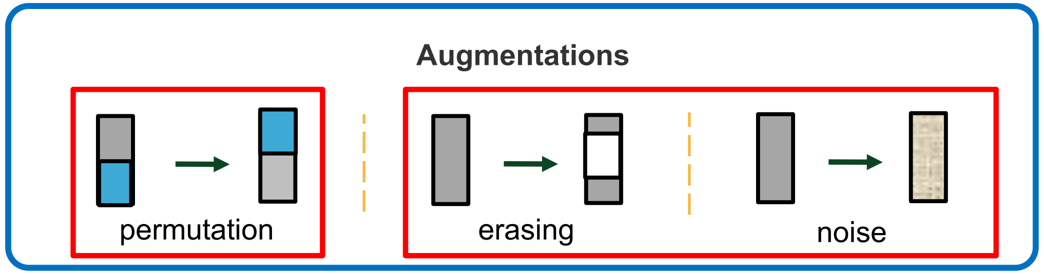
- **More than 635k model samples**
- **Zoos are open source**

| Our Zoos | Data | Architecture | Samples |
|---|---|---|---|
| Tetris-Seed | Tetris | MLP (100 params.) | 75k |
| Tetris-Hyp | Tetris | MLP (100 params.) | 217.5k |
| MNIST-Seed | MNIST | CNN (2464 params.) | 50k |
| F-MNIST-Seed | F-MNIST | CNN (2464 params.) | 50k |
| MNIST-Hyp-1-Fix-Seed | MNIST | CNN (2464 params.) | ~57.6k |
| MNIST-Hyp-1-Rand-Seed | MNIST | CNN (2464 params.) | ~57.6k |
| MNIST-Hyp-5-Fix-Seed | MNIST | CNN (2464 params.) | ~64k |
| MNIST-Hyp-5-Rand-Seed | MNIST | CNN (2464 params.) | ~64k |

| Zoos from Unterthiner et al., 2020 | Data | Architecture | Samples |
|---|---|---|---|
| MNIST-Hyp | MNIST | CNN (4970 params.) | 270k |
| F-MNIST-Hyp | F-MNIST | CNN (4970 params.) | 270k |
| CIFAR-Hyp | CIFAR10 | CNN (4970 params.) | 270k |
| SVHN-Hyp | SVHN | CNN (4970 params.) | 270k |




# NN Weights Augmentations




**Augmentations:**

- Multiply # of samples
- Encode inductive bias

**Erasing & Noise:**

- Adaptations from computer vision

**Permutation Augmentation:**

- Leverages symmetries in weight space
- Proof: equivalence holds forward & backward
- Scales with faculty of # neurons/kernels
- Fully-connected and convolutional layers
- Full Details: Appendix A

**Assumptions**

$$(\mathbf{P}^l)^\mathrm{T}\mathbf{P}^l = \mathbf{I}, \qquad \mathbf{P}^l\sigma(\mathbf{n}^l) = \sigma(\mathbf{P}^l\mathbf{n}^l),$$

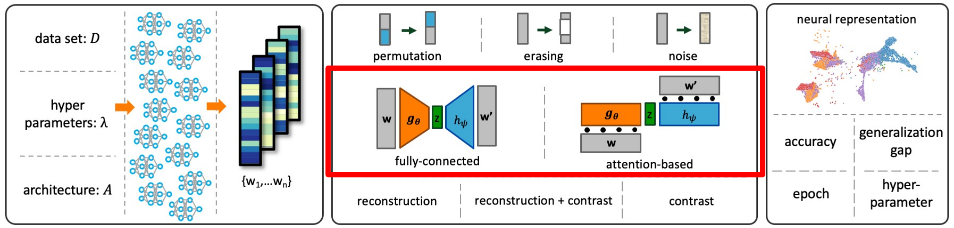
**Forward pass**

$$\begin{aligned}\mathbf{n}^{l+1} &= \mathbf{W}^{l+1}\,\mathbf{I}\,\sigma(\mathbf{W}^l\,\mathbf{a}^{l-1} + \mathbf{b}^l) + \mathbf{b}^{l+1}\\ &= \mathbf{W}^{l+1}\,(\mathbf{P}^l)^\mathrm{T}\,\mathbf{P}^l\,\sigma(\mathbf{W}^l\,\mathbf{a}^{l-1} + \mathbf{b}^l) + \mathbf{b}^{l+1}\\ &= \mathbf{W}^{l+1}\,(\mathbf{P}^l)^\mathrm{T}\,\sigma(\mathbf{P}^l\,\mathbf{W}^l\,\mathbf{a}^{l-1} + \mathbf{P}^l\,\mathbf{b}^l) + \mathbf{b}^{l+1}\\ &= \hat{\mathbf{W}}^{l+1}\,\sigma(\hat{\mathbf{W}}^l\,\mathbf{a}^{l-1} + \hat{\mathbf{b}}^l) + \mathbf{b}^{l+1},\end{aligned}$$

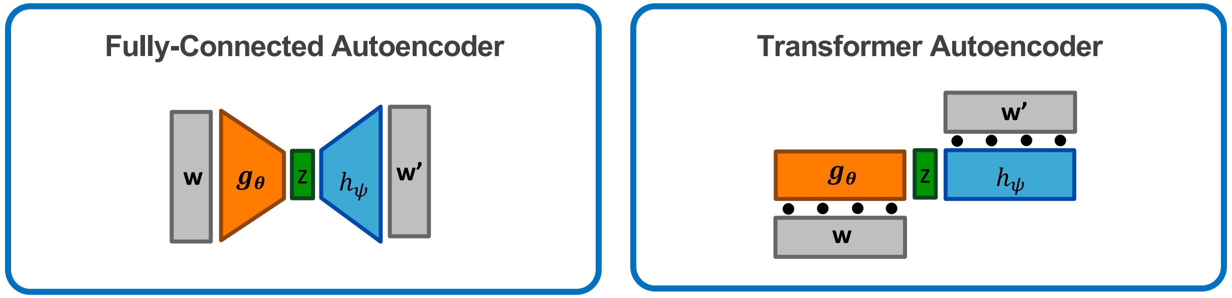
**Backward pass**

$$\begin{aligned}(\mathbf{P}^l\mathbf{W}^l)_{\mathrm{new}} &= \mathbf{P}^l\mathbf{W}^l - \alpha\mathbf{P}^l\nabla_{\mathbf{W}^l}\mathcal{L}\\ &= \mathbf{P}^l\mathbf{W}^l - \alpha\mathbf{P}^l\delta^l(\mathbf{a}^{l-1})^\mathrm{T}\\ &= \mathbf{P}^l\mathbf{W}^l - \alpha\mathbf{P}^l\left[(\mathbf{W}^{l+1})^\mathrm{T}\delta^{l+1}\odot\sigma'(\mathbf{n}^l)\right](\mathbf{a}^{l-1})^\mathrm{T}\\ &= \mathbf{P}^l\mathbf{W}^l - \alpha\left[(\mathbf{W}^{l+1}\mathbf{P}^\mathrm{T})^\mathrm{T}\delta^{l+1}\odot\sigma'(\mathbf{P}^l\mathbf{n}^l)\right](\mathbf{a}^{l-1})^\mathrm{T}\\ &= \mathbf{P}^l\mathbf{W}^l - \alpha[(\mathbf{W}^{l+1}(\mathbf{P}^l)^\mathrm{T})^\mathrm{T}\delta^{l+1}\odot\sigma'(\mathbf{P}^l\mathbf{W}^l\mathbf{a}^{l-1} + \mathbf{P}^l\mathbf{b}^l)](\mathbf{a}^{l-1})^\mathrm{T}.\end{aligned}$$

$$(\hat{\mathbf{W}}^l)_{\mathrm{new}} = \hat{\mathbf{W}}^l - \alpha\left[(\hat{\mathbf{W}}^{l+1})^\mathrm{T}\delta^{l+1}\odot\sigma'(\hat{\mathbf{W}}^l\mathbf{a}^{l-1} + \hat{\mathbf{b}}^l)\right](\mathbf{a}^{l-1})^\mathrm{T}\square$$

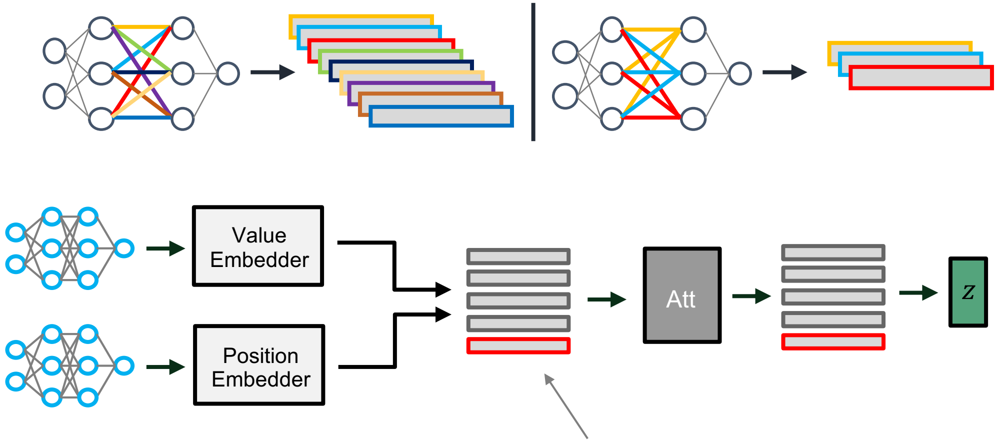



# Representation Learning Architecture




## Challenge:

- little intuition on inductive biases
- scalability for larger samples

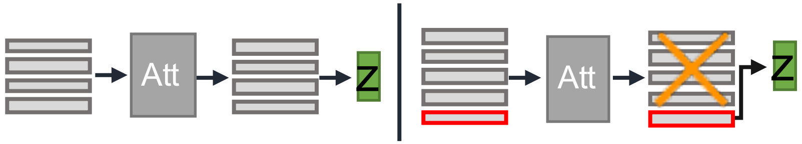


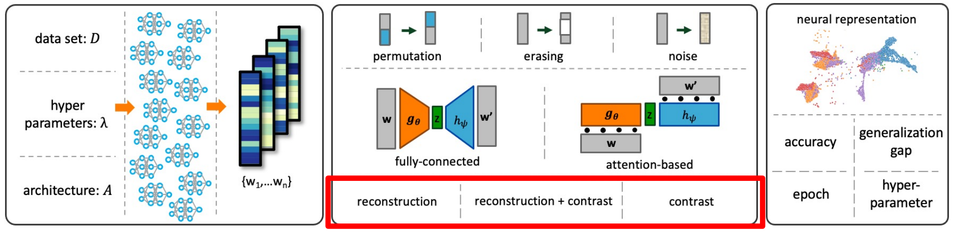

## Fully-Connected AE

- **+** low inductive bias
- **-** doesn't scale well for large inputs

## Transformer AE

- **+** low inductive bias
- **+** scales to larger models
- Two encodings: sequences of
  - Weights
  - Neurons (all weights of one neuron/kernel)
- Compression token






# Representation Learning Task

**Goal:**

- rich, generalizing representation

**Reconstruction:**

- Full representation of samples

$$\mathcal{L}_{MSE} = \frac{1}{M}\sum_{i=1}^{M} \|\mathbf{w}_i - h_\psi(g_\theta(\mathbf{w}_i))\|_2^2.$$

**Contrast:**

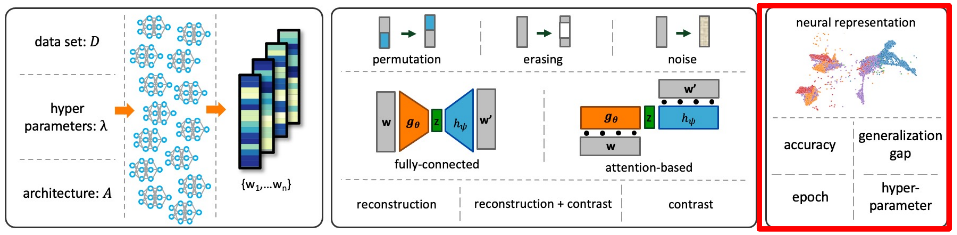
- Include inductive bias

$$\mathcal{L}_c = \sum_{(i,j)} -\log \frac{\exp(\mathrm{sim}(\bar{\mathbf{z}}_i, \bar{\mathbf{z}}_j)/\tau}{\sum_{k=1}^{2M_B} \mathbb{I}_{k\neq i} \exp(\mathrm{sim}(\bar{\mathbf{z}}_i, \bar{\mathbf{z}}_j)/\tau}$$

**Four Taks:**

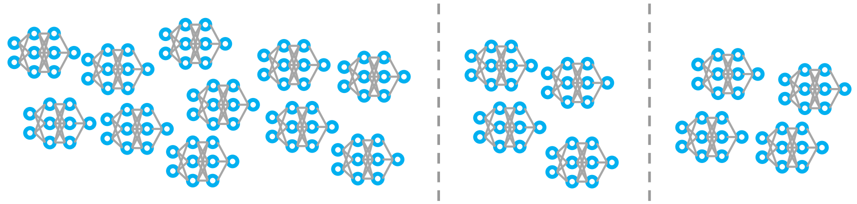
1. Reconstruction only
2. Contrast only
3. Reconstruction + Contrast
4. Reconstruction + ‘positive’ Contrast

$$\mathcal{L}_{c+} = \sum_i -\log\left(\exp(\mathrm{sim}(\bar{\mathbf{z}}_i^j, \bar{\mathbf{z}}_i^k))/\tau\right) = \sum_i -\mathrm{sim}(\bar{\mathbf{z}}_i^j, \bar{\mathbf{z}}_i^k) + \log(\tau).$$



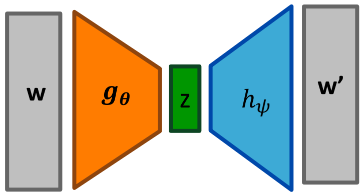

# Downstream Tasks

## Split Zoos in train | val | test



## Train Representation

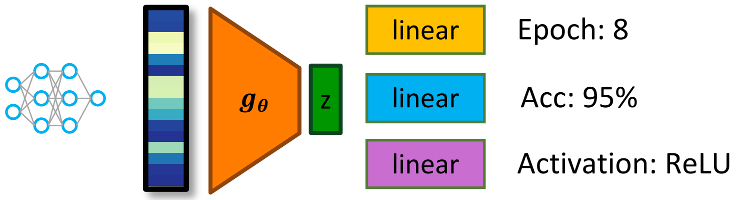
- Evaluation: Reconstruction R^2




Train and freeze

## Downstream Task:

- Linear probe for model characteristics
- Evaluation: Accuracy / R^2



# Experiment Results: Ablation Studies

**Augmentation:**

- Aggregated performance
- Permutation augmentation most useful
- Combination of augmentations beneficial

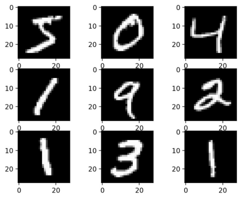
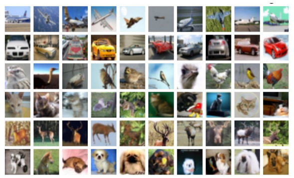
| | TETRIS-SEED | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | - | P | E | N | P,E | P,N | E,N | P,E,N |
| $E_c$ | 46.5 | 71.2 | 27.7 | 23.2 | 71.2 | 57.5 | 27.9 | 71.8 |
| ED | 66.4 | 68.2 | 65.9 | 66.2 | 64.9 | 67.9 | 65.6 | 64.4 |
| $E_cD$ | 60.1 | 75.4 | 66.3 | 61.5 | 82.0 | 79.1 | 70.7 | 82.2 |
| $E_{c+}D$ | 64.7 | 69.9 | 65.8 | 64.0 | 66.2 | 67.3 | 65.8 | 63.6 |

**Architectures:**

- Fully-connected lowest performance
- Weight-encoding works, but doesn't scale
- Neuron-encoding has best performance
- Compression token generally improves performance

| | TETRIS-SEED | | | | |
|---|---|---|---|---|---|
| | Rec. | Eph | Acc | Ggap | $F1_{AVG}$ |
| FF | 0.0 | 80.0 | 85.3 | 79.8 | 63.9 |
| $Att_W$ | 6.8 | 95.3 | 71.1 | 71.2 | 57.0 |
| $Att_{W+t}$ | 74.1 | 95.4 | 88.6 | **82.4** | 73.0 |
| $Att_N$ | **89.4** | **97**.1 | 88.4 | 80.6 | **75.9** |
| $Att_{N+t}$ | 84.1 | 97.0 | **90.2** | 81.9 | 75.6 |

# Experiment Results

| | MNIST-HYP | | | FASHION-HYP | | | CIFAR10-HYP | | | SVHN-HYP | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | W | s(W) | $E_cD$ | W | s(W) | $E_cD$ | W | s(W) | $E_cD$ | W | s(W) | $E_cD$ |
| EPH | 25.8 | 33.2 | **50.0** | 26.6 | 34.6 | **51.3** | 25.7 | 30.3 | **53.3** | 22.8 | 37.8 | **52.6** |
| ACC | 74.7 | 81.5 | **94.9** | 70.9 | 78.5 | **96.2** | 76.4 | 82.9 | **92.7** | 80.5 | 82.1 | **91.1** |
| GGAP | 23.4 | 24.4 | **27.4** | 48.1 | 41.1 | **49.0** | 37.7 | 37.4 | **40.4** | 38.7 | 42.2 | **44.2** |
| LR | 29.3 | 34.3 | **37.1** | 33.5 | 35.6 | **42.4** | 27.4 | 32.3 | **44.7** | 24.5 | 33.4 | **49.1** |
| $\ell_2$-REG | 12.5 | 16.5 | **20.1** | 11.9 | 16.3 | **25.0** | 08.7 | 13.8 | **28.0** | 09.0 | 13.6 | **28.0** |
| DROP | 28.5 | 19.2 | **35.8** | 26.7 | 21.3 | **38.3** | 16.7 | 16.5 | **33.8** | 09.0 | 14.6 | **23.3** |
| TF | 03.8 | 07.8 | **15.9** | 08.1 | 08.2 | **22.1** | 08.4 | 06.9 | **35.4** | 03.2 | 08.8 | **21.4** |
| ACT | 88.6 | 81.1 | **88.7** | 89.8 | 82.4 | **90.1** | 88.3 | 80.3 | **90.0** | 86.9 | 78.8 | **87.2** |
| INIT | **94.6** | 72.0 | 80.6 | **95.7** | 76.5 | 86.7 | **93.5** | 73.3 | 82.6 | **91.0** | 73.0 | 82.8 |
| OPT | **76.7** | 65.4 | 66.4 | **79.9** | 67.4 | 73.0 | **74.0** | 65.5 | 71.0 | **72.5** | 68.2 | 72.3 |

# Out-of-Distribution

**Experiment Setup**

- Train Representation & Linear Probe on ID Zoo
- Use learned representation & linear probe on OOD Zoos
- Evaluation: Kendall's tau

**Results**

- Approach generalizes to OOD settings
- Outperforms baselines in the majority of cases

| | MNIST-HYP | | | FASHION-HYP | | | SVHN-HYP | | | CIFAR10-HYP | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | W | s(W) | $E_{c+}D$ | W | s(W) | $E_{c+}D$ | W | s(W) | $E_{c+}D$ | W | s(W) | $E_{c+}D$ |
| MNIST-HYP | **.36** | .29 | **.36** | .21 | .14 | **.27** | **.26** | .12 | .23 | -.01 | -.04 | **.02** |
| FASHION-HYP | -.02 | **.08** | .02 | .54 | .48 | **.56** | .06 | **.14** | .01 | .07 | .10 | **.27** |
| SVHN-HYP | .05 | **.15** | -.04 | -.02 | **.27** | .10 | .44 | .34 | **.45** | -.02 | .08 | **.10** |
| CIFAR10-HYP | **.11** | .09 | .06 | .38 | .36 | **.39** | .14 | .14 | **.15** | **.41** | .28 | .35 |

# Acknowledgements

Find our work at **hsg.ai/neurips21**

*Thanks to*
Marco Schreyer
Xavier Giró-i-Nieto
Pol Caselles Rico

*Funding:* HSG Basic Research Fund


# Thanks for your attention!