# Graph Metanetworks for Processing Diverse Neural Architectures


NVIDIA
MIT CSAIL


Derek Lim

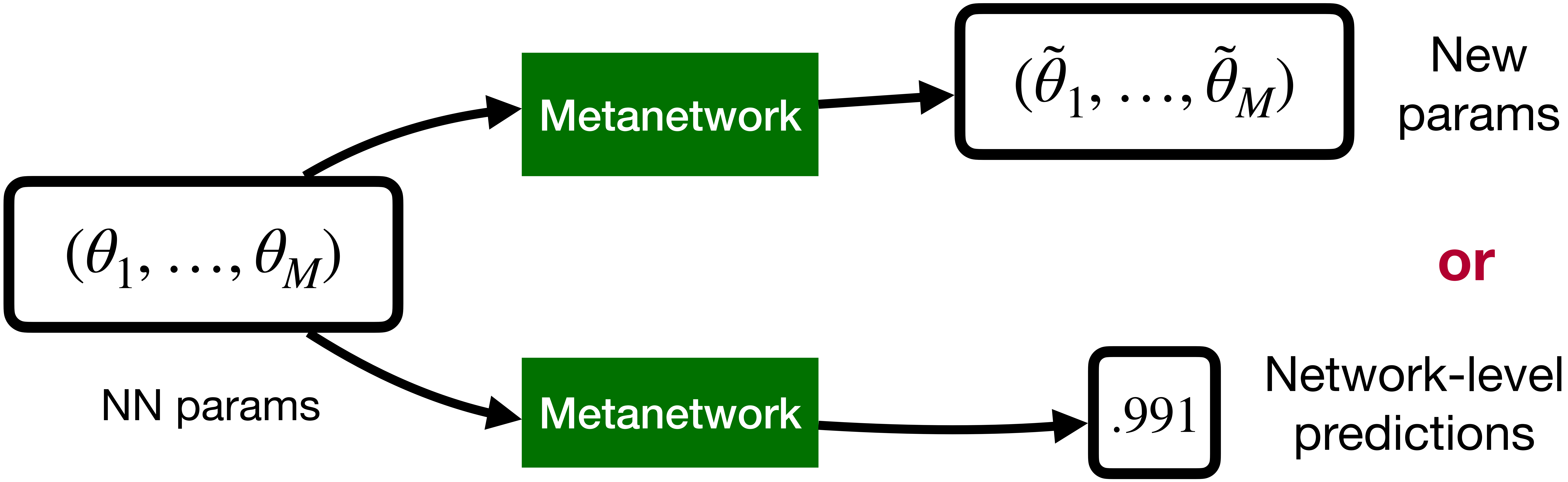

Haggai Maron


Marc Law


Jonathan Lorraine


James Lucas

# Metanetworks take Neural Networks as Input




E.g. interpreting NNs from weights, editing weights, generating NNs

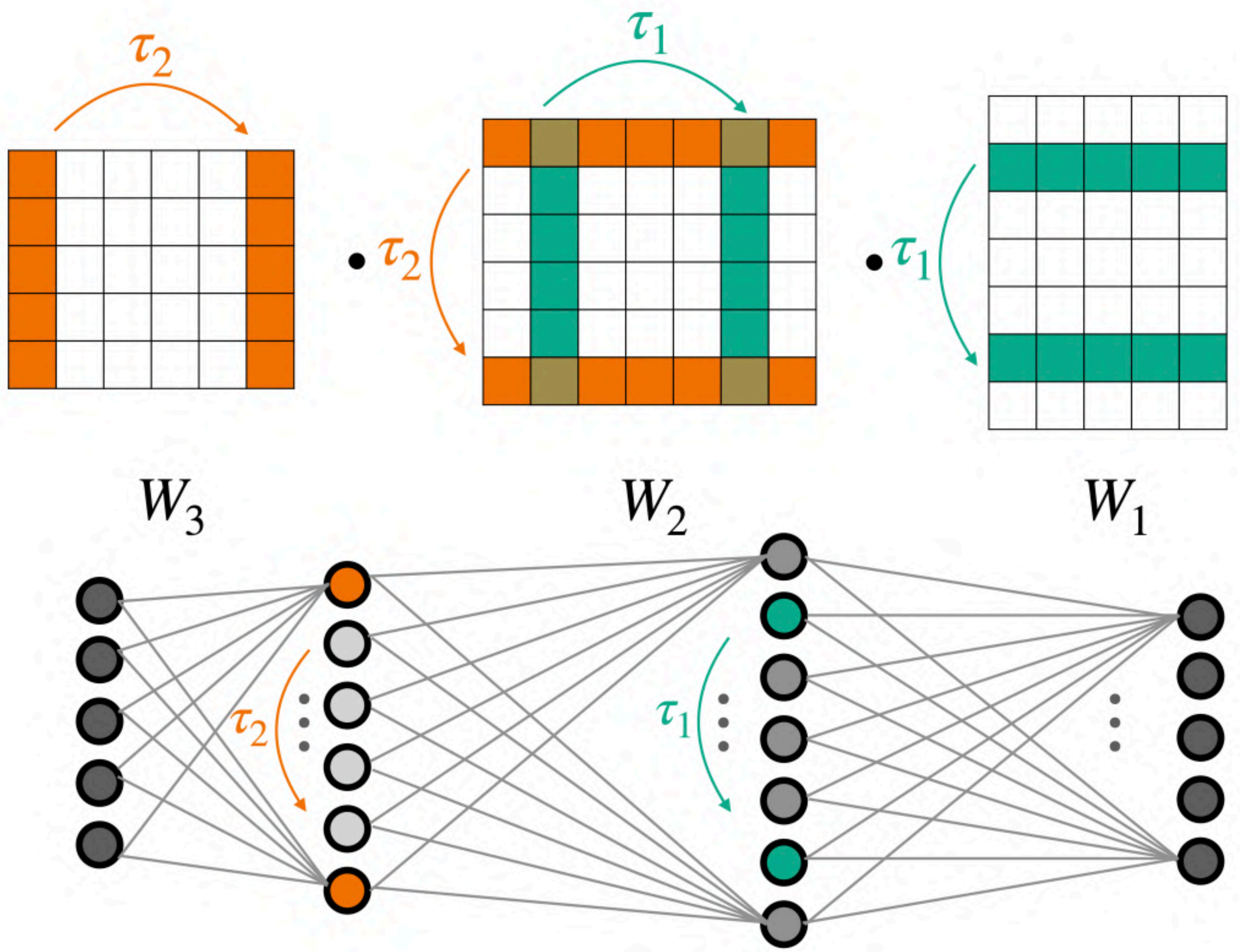
# Symmetries in Neural Network Weights

**MLP**

$$W_2\sigma(W_1x)$$

**Permutation Symmetry**

$$W_2P^T\sigma(PW_1x) = W_2\sigma(W_1x)$$




$(W_2P^\top, PW_1)$ and $(W_2, W_1)$ define same neural network function!

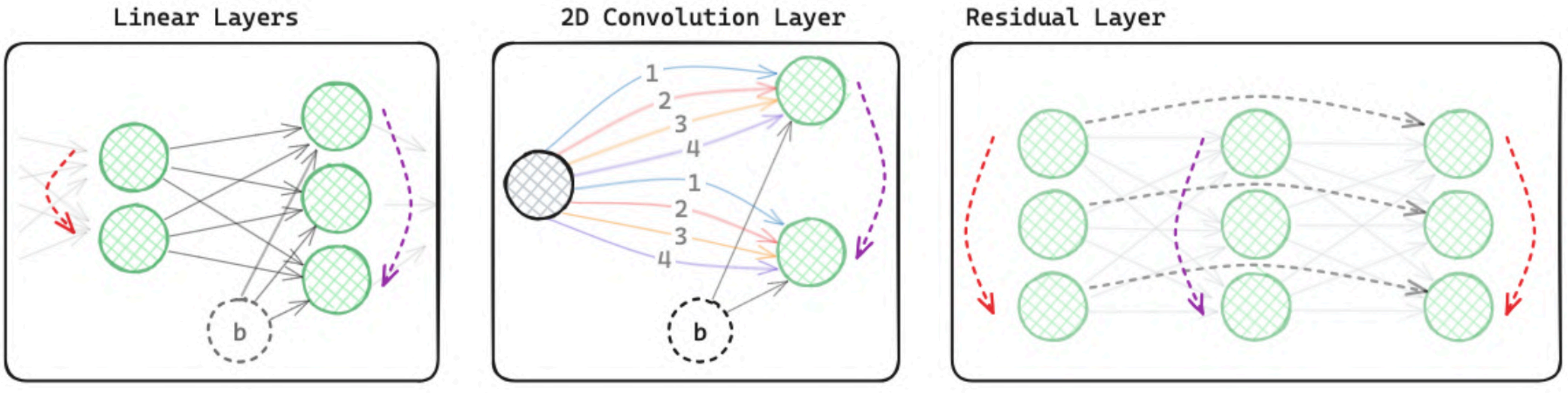
**Want metanet to treat them as equivalent**

Figure from (Navon et al. 2023)

# What are the parameter symmetries of general NNs?

We capture these via automorphisms of a graph associated to the input NN!

Call them neural DAG automorphisms, see paper for more



# Our Model: Graph Metanetworks

**Step 1:** View input neural network as graph

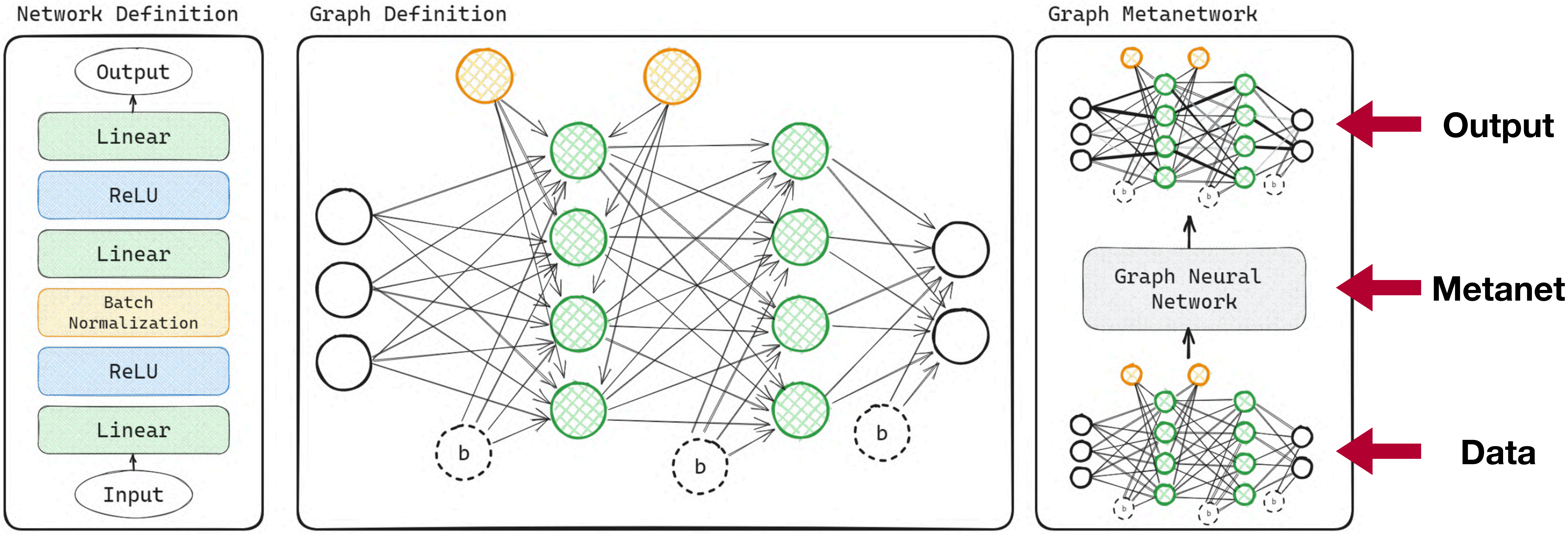
Node = neurons, Edges = parameters

**Step 2:** Apply graph neural network

New parameter prediction = edge prediction

Network-level prediction = graph-level prediction

Neuron-level prediction = node prediction



# Graph Metanets have the desired equivariances

**Strength 1**

- GNNs are equivariant to graph automorphisms
- Thus, Graph Metanets are equivariant to Neural DAG Automorphisms

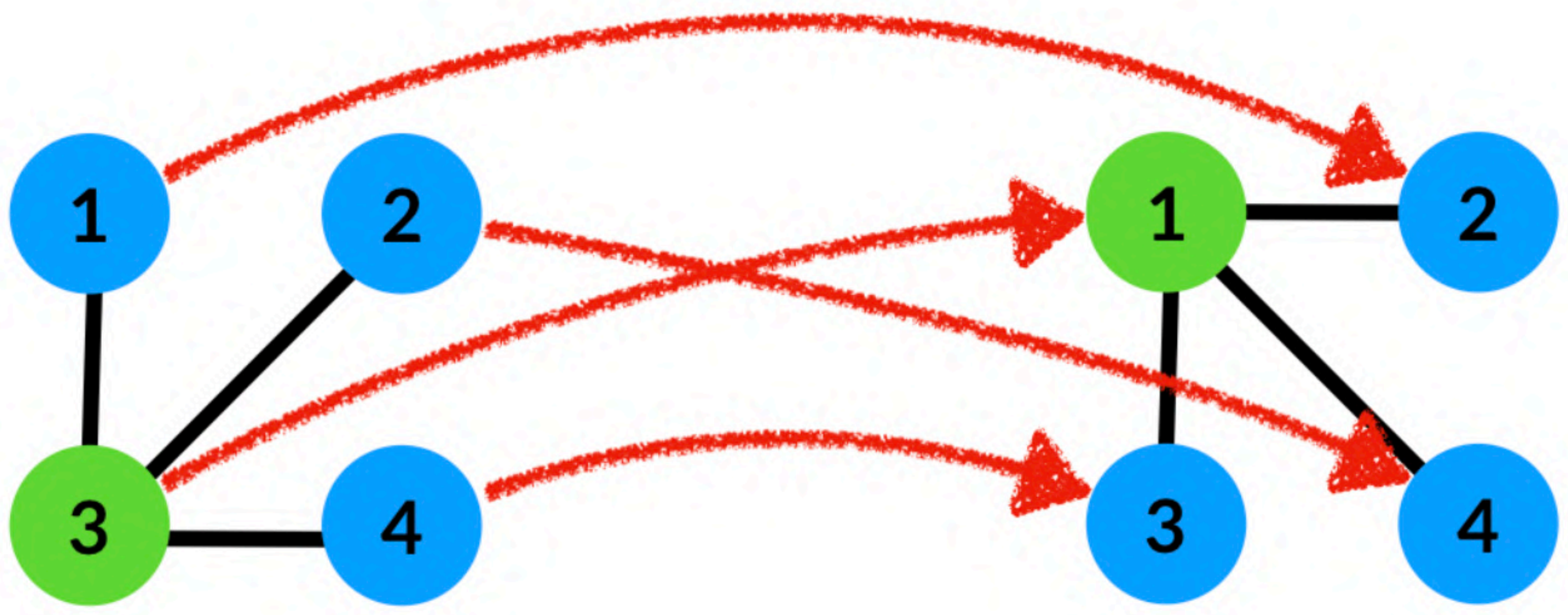
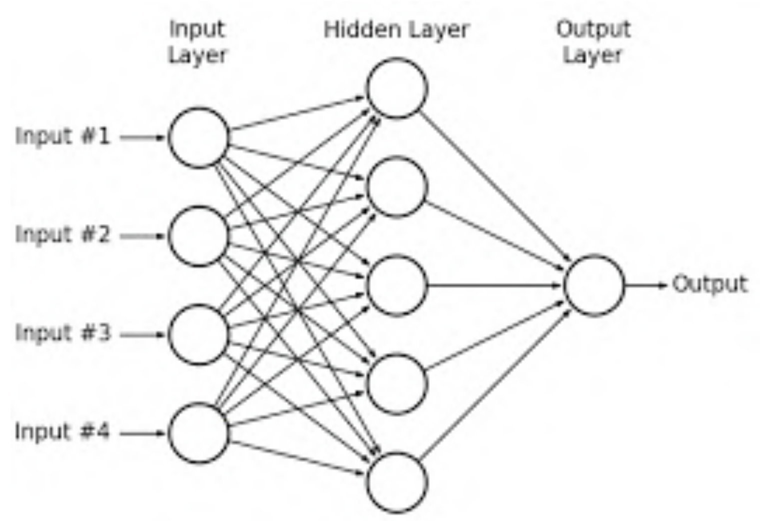


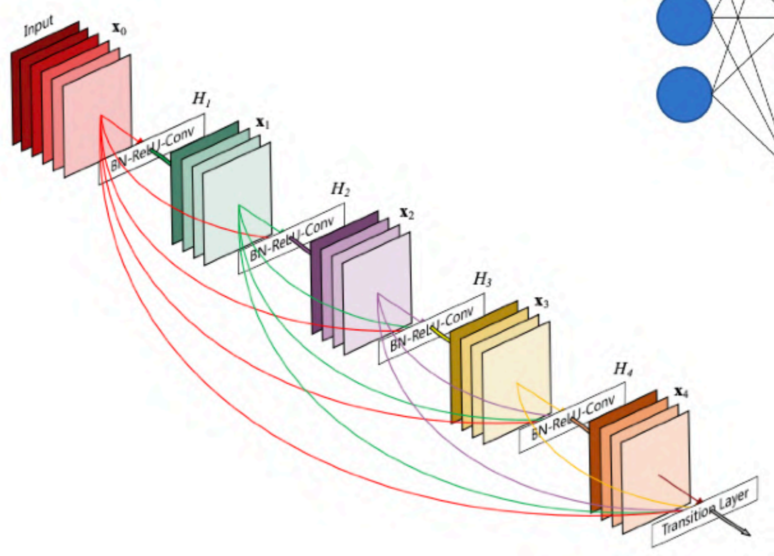

Figure from Maron and Lipman, 2019

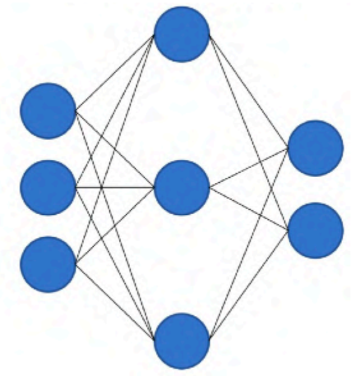
# Graph Metanets are General!

**Strength 2**

- To apply to new input NN, just view it as a graph
- One GNN can process different architectures, sizes, etc





# Graph Metanets are Provably Expressive!

**Strength 3**

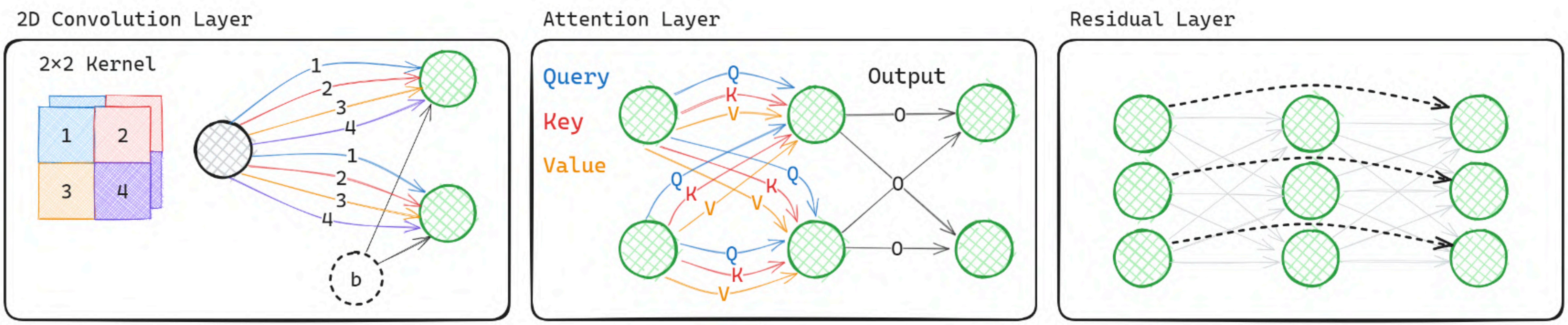
**Theorem:**
Graph Metanets can simulate any NP-NFNet *(Zhou et al. 2023)* or StatNN (Unterthiner et al. 2020)

$$
\begin{aligned}
H(U) &\coloneqq (Y, z) \in \mathcal{W} \times \mathcal{V} \\
Y^{(i)}_{jk} &\coloneqq \sum_s \left( a^{i,s}_{\vartheta} W^{(s)}_{\star,\star} + a^{i,s}_{\phi} v^{(s)}_{\star} \right) + b^{i,i}_{\vartheta} W^{(i)}_{\star,k} + b^{i,i-1}_{\vartheta} W^{(i-1)}_{k,\star} \\
&\quad + b^{i}_{\phi} v^{(i)}_{j} + c^{i,i}_{\vartheta} W^{(i)}_{j,\star} + c^{i,i+1}_{\vartheta} W^{(i+1)}_{\star,j} + c^{i}_{\phi} v^{(i-1)}_{k} + d^{i}_{\vartheta} W^{(i)}_{jk} \\
z^{(i)}_{j} &\coloneqq \sum_s \left( a^{i,s}_{\varphi} W^{(s)}_{\star,\star} + a^{i,s}_{\psi} v^{(s)}_{\star} \right) + b^{i,i}_{\varphi} W^{(i)}_{j,\star} + b^{i,i+1}_{\varphi} W^{(i+1)}_{\star,j} + b^{i}_{\psi} v^{(i)}_{j}
\end{aligned}
$$

NP-NFNet

**Theorem:**
Graph Metanets can simulate forward pass of input neural nets.

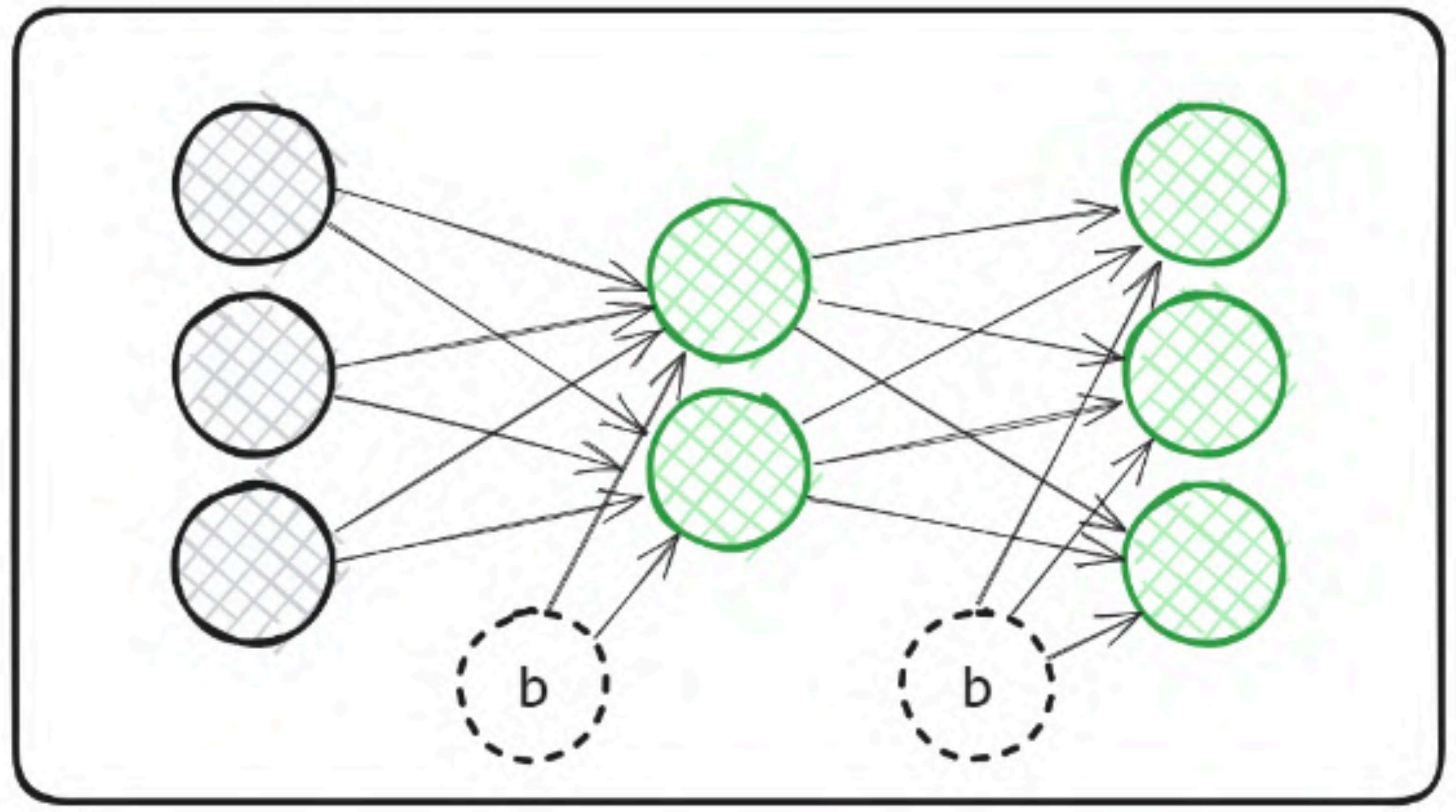
# Graph Constructions 1



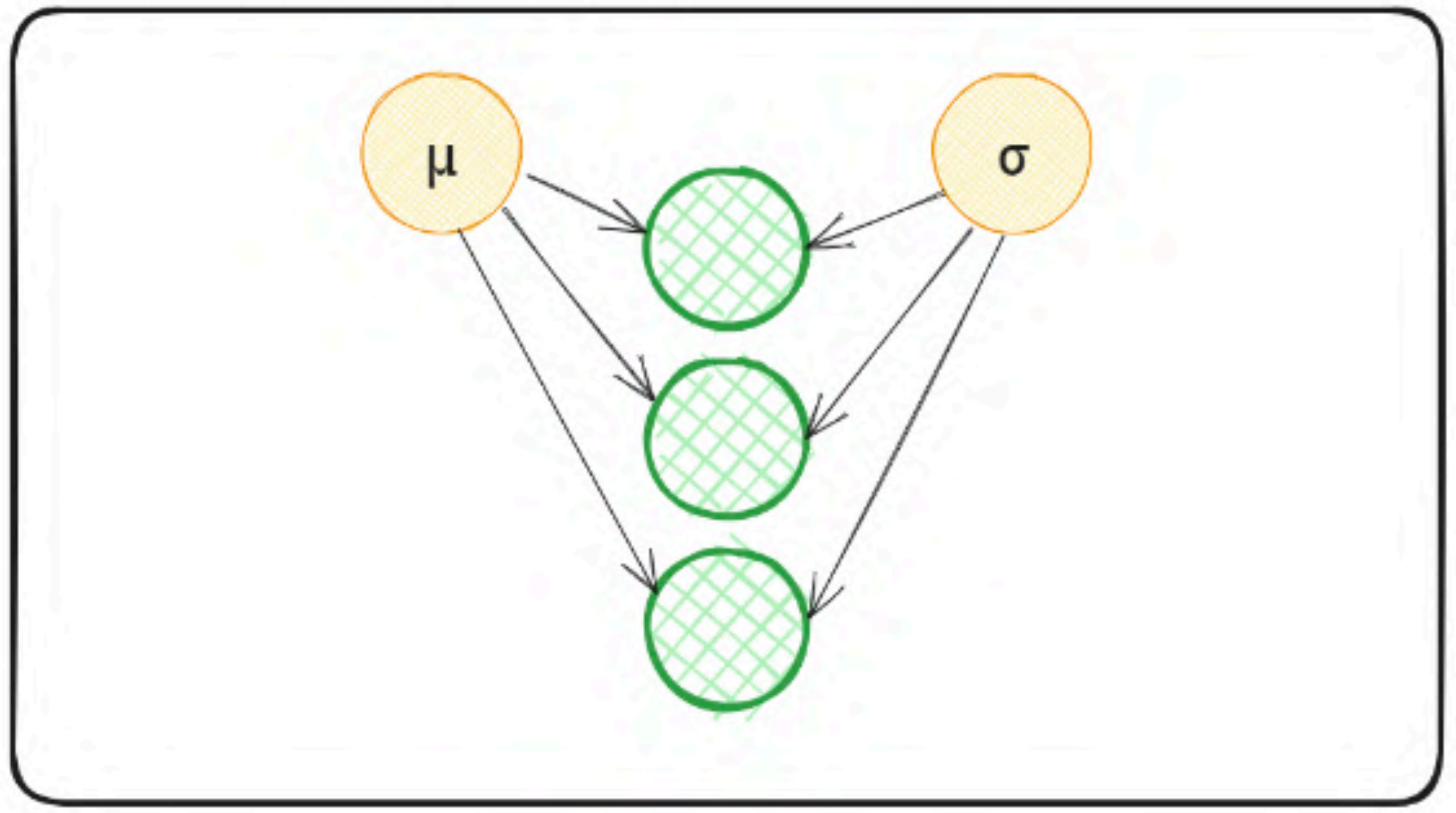

Can also handle general group-equivariant linear layers like this!

# Graph Constructions 2

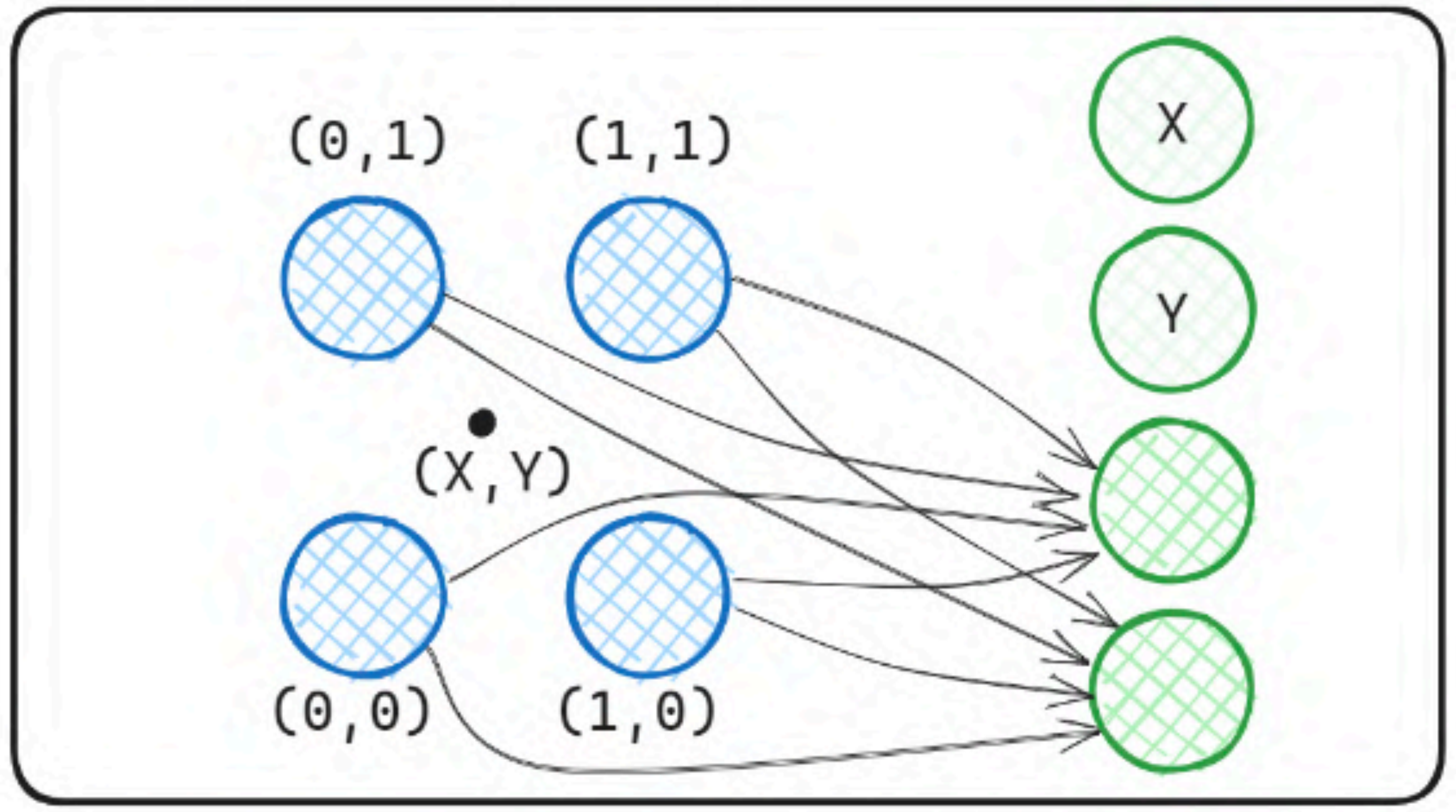
Linear Layers




Normalization Layer




Spatial Parameter Grid



# Graph Metanets outperform other Metanets!

| | Metanet | Diverse Architectures $R^2$ | Diverse Architectures $\tau$ |
|---|---|---|---|
| 100% | DeepSets (Zaheer et al., 2017) | .562±.020 | .559±.011 |
| | DMC (Eilertsen et al., 2020) | .957±.009 | .883±.007 |
| | GMN (Ours) | **.973**±.001 | **.905**±.003 |
| 10% | DeepSets (Zaheer et al., 2017) | .126±.015 | .290±.010 |
| | DMC (Eilertsen et al., 2020) | .810±.046 | .758±.013 |
| | GMN (Ours) | **.912**±.013 | **.827**±.007 |
| OOD | DeepSets (Zaheer et al., 2017) | .128±.071 | .380±.014 |
| | DMC (Eilertsen et al., 2020) | −.134 ± .147 | .566±.055 |
| | GMN (Ours) | **.694**±.047 | **.757**±.019 |

- **Task:** Predict CIFAR-10 accuracy of input image classifier, from weights.
- **Dataset**: diverse image classifiers (CNNs, DeepSets, ViTs, ResNets) of varying sizes

DeepSets: Equivariant
DMC: Expressive
GMN: Both

**Graph Metanets generalize much better**