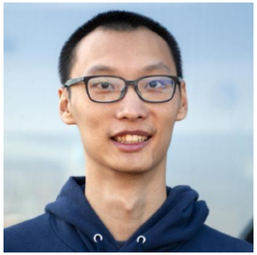
FMI
Friedrich Miescher Institute
for Biomedical Research

Finding trainable sparse networks through Neural Tangent Transfer



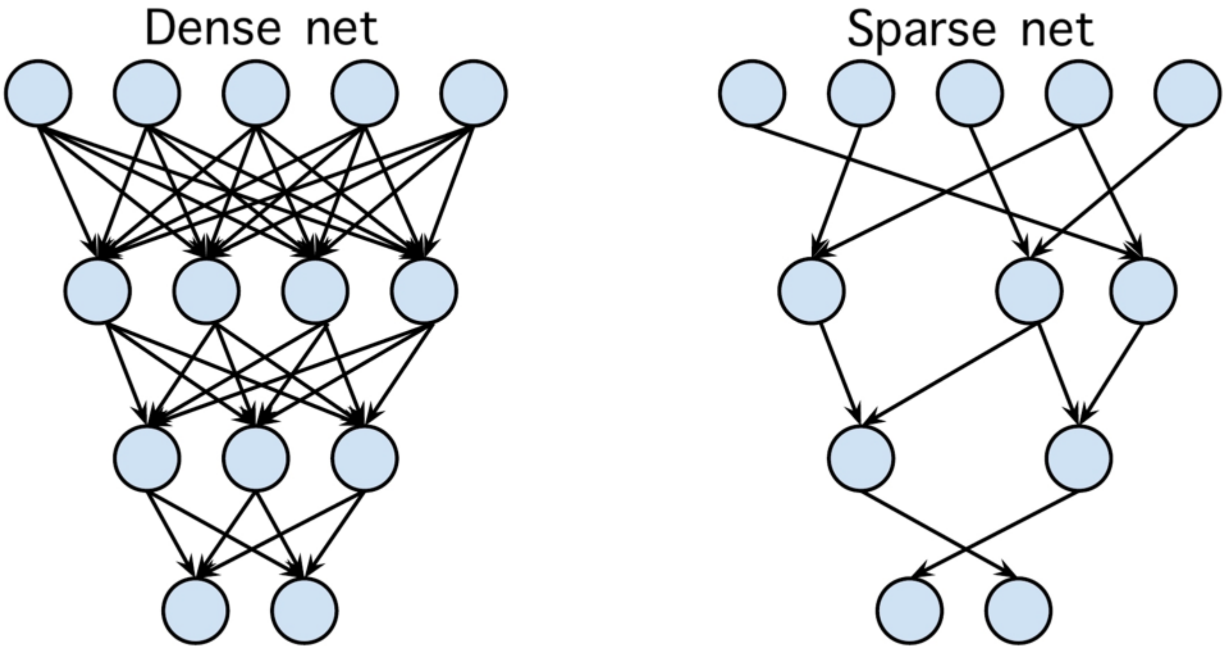
Tianlin Liu[1 2]

Friedemann Zenke[1]

[1]Friedrich Miescher Institute for Biomedical Research, Basel, Switzerland

[2]University of Basel, Basel, Switzerland

Sparse neural nets

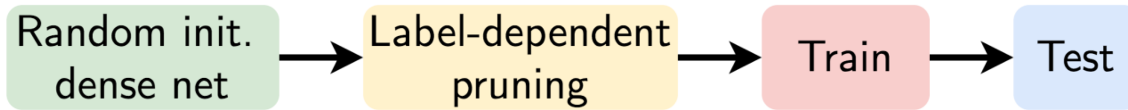


Sparse nets are computationally efficient but difficult to train.

Friedrich Miescher Institute
for Biomedical Research

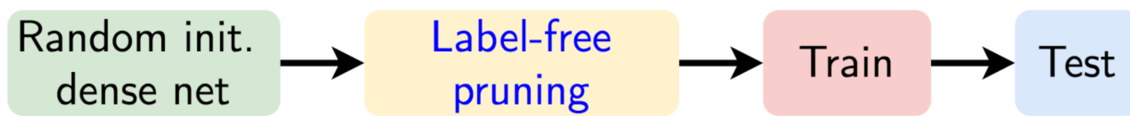
Foresight pruning

- Existing foresight pruning methods:



Lee, et al. SNIP: Single-shotnetwork pruning based on connection sensitivity, ICLR2019.
Wang, el al. Picking winningtickets before training by preserving gradient flow, ICLR2020.

- Our new foresight pruning, **Neural Tangent Transfer**:



Advantages:

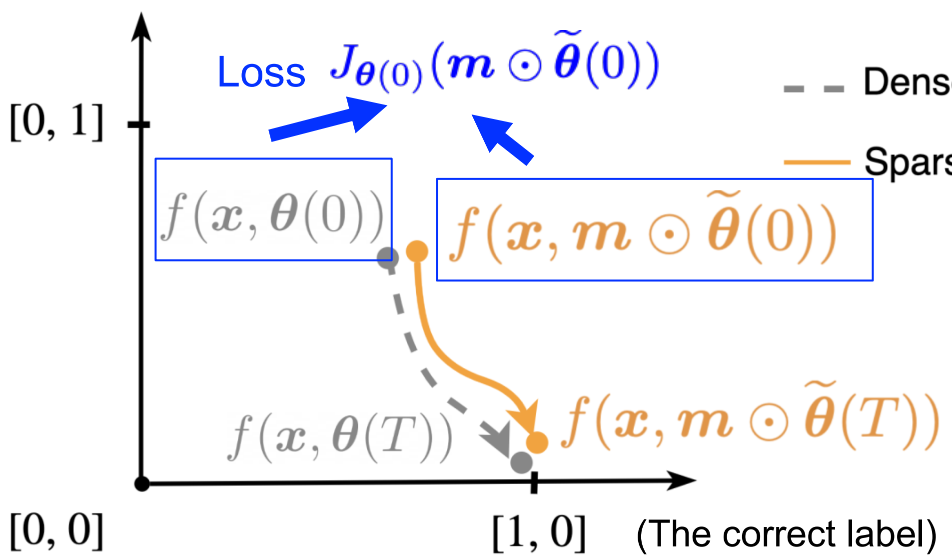
- Does not require labels for pruning.
- Per-layer sparsity levels of pruned networks are controllable.

The idea of our approach

Notation: **dense net** $f(x, \theta)$ **sparse net** $f(x, m \odot \tilde{\theta})$

Want: initialize a **sparse net** that is as "trainable" as a **dense net:**

$$f(x, m \odot \tilde{\theta}(T)) \approx f(x, \theta(T))$$



We want a small $J_{\theta(0)}(m \odot \tilde{\theta}(0))$ to imply $f(x, m \odot \tilde{\theta}(t)) \approx f(x, \theta(t))$ for all t > 0.

Our loss function (first try)

Given $\mathcal{X}$ and $f(\cdot, \boldsymbol{\theta}(0))$, choose $\{\boldsymbol{m}, \widetilde{\boldsymbol{\theta}}(0)\}$ to minimize

$$\sum_{t=t_0}^{t_T} \| \underbrace{f(\boldsymbol{\mathcal{X}}, \boldsymbol{m} \odot \widetilde{\boldsymbol{\theta}}(t))}_{\text{Output of the sparse net under supervised training}} - \underbrace{f(\boldsymbol{\mathcal{X}}, \boldsymbol{\theta}(t))}_{\text{Output of the dense net under supervised training}} \|_2^2$$

Impossible to evaluate without using labels!

,

Modified loss through linearization

Given $\mathcal{X}$ and $f(\cdot, \boldsymbol{\theta}(0))$, choose $\{\boldsymbol{m}, \widetilde{\boldsymbol{\theta}}(0)\}$ to minimize

$$\sum_{t=t_0}^{t_T} \| f^{\mathrm{lin}}(\boldsymbol{\mathcal{X}}, \boldsymbol{m} \odot \widetilde{\boldsymbol{\theta}}(t)) - f^{\mathrm{lin}}(\boldsymbol{\mathcal{X}}, \boldsymbol{\theta}(t)) \|_2^2$$

Output of the
linearized sparse
net
under supervised

Output of the
linearized dense
net
under supervised
training

$$f^{\mathrm{lin}}(\boldsymbol{x}, \boldsymbol{m} \odot \widetilde{\boldsymbol{\theta}}) := f\left(\boldsymbol{x}, \boldsymbol{m} \odot \widetilde{\boldsymbol{\theta}}(0)\right) + \left\langle \widetilde{\boldsymbol{\theta}} - \widetilde{\boldsymbol{\theta}}(0), \nabla_{\widetilde{\boldsymbol{\theta}}} f\left(\boldsymbol{x}, \boldsymbol{m} \odot \widetilde{\boldsymbol{\theta}}(0)\right) \right\rangle.$$

$$f^{\mathrm{lin}}(\boldsymbol{x}, \boldsymbol{\theta}) := f\left(\boldsymbol{x}, \boldsymbol{\theta}(0)\right) + \left\langle \boldsymbol{\theta} - \boldsymbol{\theta}(0), \nabla_{\boldsymbol{\theta}} f\left(\boldsymbol{x}, \boldsymbol{\theta}(0)\right) \right\rangle.$$

Lee et al. Wide Neural Networks of Any Depth Evolve as Linear Models Under Gradient Descent, NeurIPS, 2019.

The neural tangent transfer loss

Given $\mathcal{X}$ and $f(\cdot, \boldsymbol{\theta}(0))$, choose $\{\boldsymbol{m}, \tilde{\boldsymbol{\theta}}(0)\}$ to minimize

$$J_{\boldsymbol{\theta}(0)}\left(\boldsymbol{m} \odot \tilde{\boldsymbol{\theta}}(0)\right) = \overbrace{\frac{1}{n}\left\| f\left(\boldsymbol{\mathcal{X}}, \boldsymbol{m} \odot \tilde{\boldsymbol{\theta}}(0)\right) - f\left(\boldsymbol{\mathcal{X}}, \boldsymbol{\theta}(0)\right)\right\|_2^2}^{\text{Target distance:}}$$
$$+ \underbrace{\frac{\gamma^2}{n^2}\left\| \boldsymbol{H}_{\boldsymbol{m}\odot\tilde{\boldsymbol{\theta}}(0)} - \boldsymbol{H}_{\boldsymbol{\theta}(0)}\right\|_F^2}_{\text{Neural Tagent Kernel (NTK) distance}},$$

Jacot et al. Neural Tangent Kernel: Convergence and Generalization in Neural Networks, NeurIPS, 2018.

where
$$\boldsymbol{H}_{\boldsymbol{\theta}(0)}(i,j) = \left\langle \nabla_{\boldsymbol{\theta}} f\left(\boldsymbol{x}_i, \boldsymbol{\theta}(0)\right), \nabla_{\boldsymbol{\theta}} f\left(\boldsymbol{x}_j, \boldsymbol{\theta}(0)\right)\right\rangle$$
$$\boldsymbol{H}_{\boldsymbol{m}\odot\tilde{\boldsymbol{\theta}}(0)}(i,j) = \left\langle \nabla_{\tilde{\boldsymbol{\theta}}} f\left(\boldsymbol{x}_i, \boldsymbol{m} \odot \tilde{\boldsymbol{\theta}}(0)\right), \nabla_{\tilde{\boldsymbol{\theta}}} f\left(\boldsymbol{x}_j, \boldsymbol{m} \odot \tilde{\boldsymbol{\theta}}(0)\right)\right\rangle$$

The algorithm that minimizes this loss is called **Neural Tangent Transfer (NTT)**

Experiment

- **Datasets:** MNIST, Fashion-MNIST, CIFAR-10, SVHN

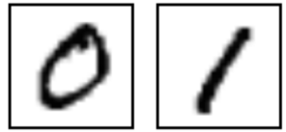
- **Network architecture:**
  Lenet-300-100 (MLP)
  Lenet-5-Caffe (CNN)
  Conv-4 (a CNN with 4 convolution layers followed by 2 dense layers with dropout)

- **General experimental procedure:**

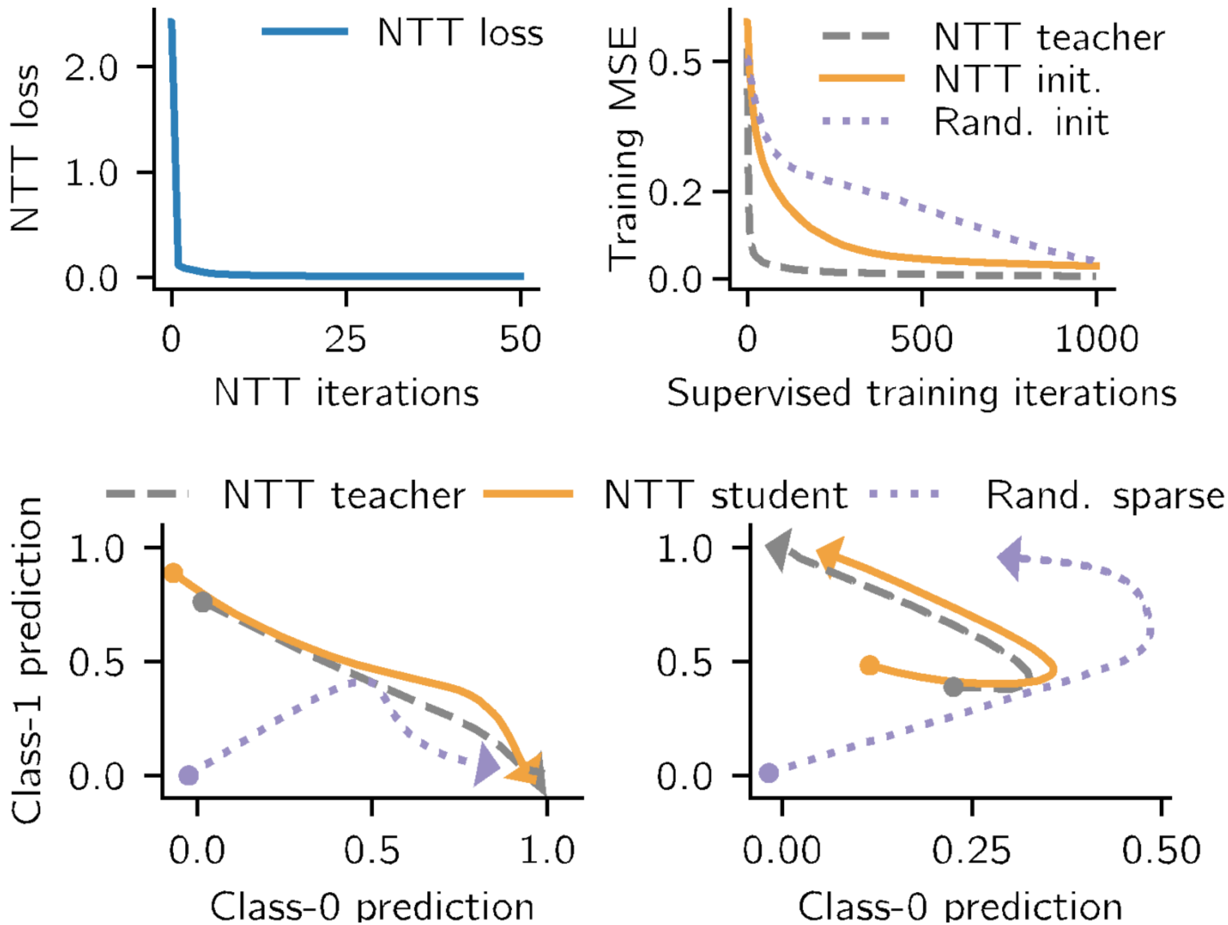
  Random init. dense net → Label-free pruning → Train → Test

- **Layerwise & global pruning**

A toy example

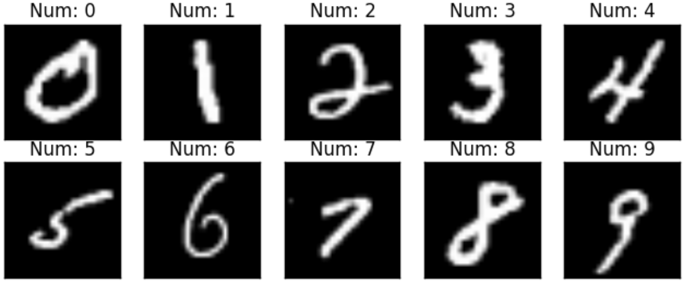


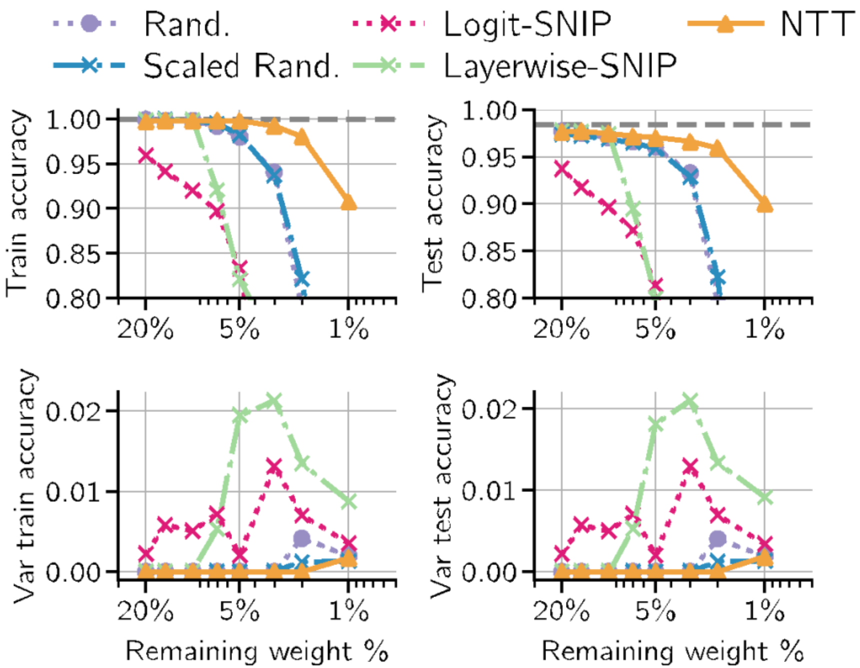
- A small dataset of 0 and 1 digits from MNIST dataset



MNIST: MLP Lenet-300-100

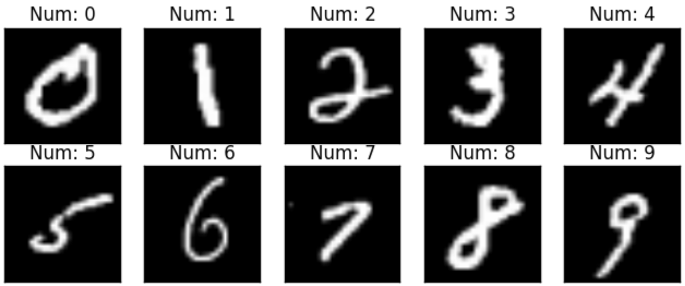
Friedrich Miescher Institute
for Biomedical Research

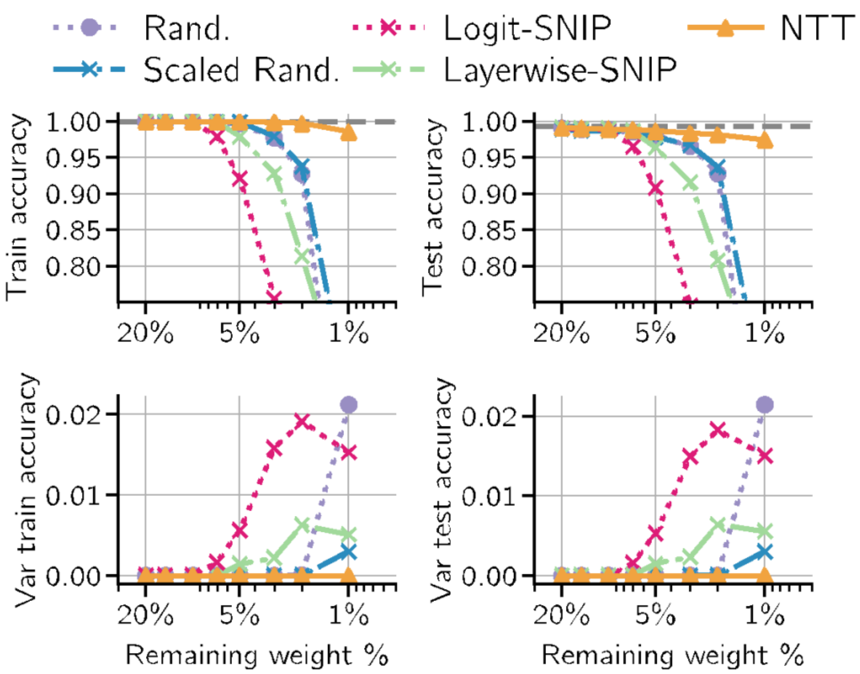




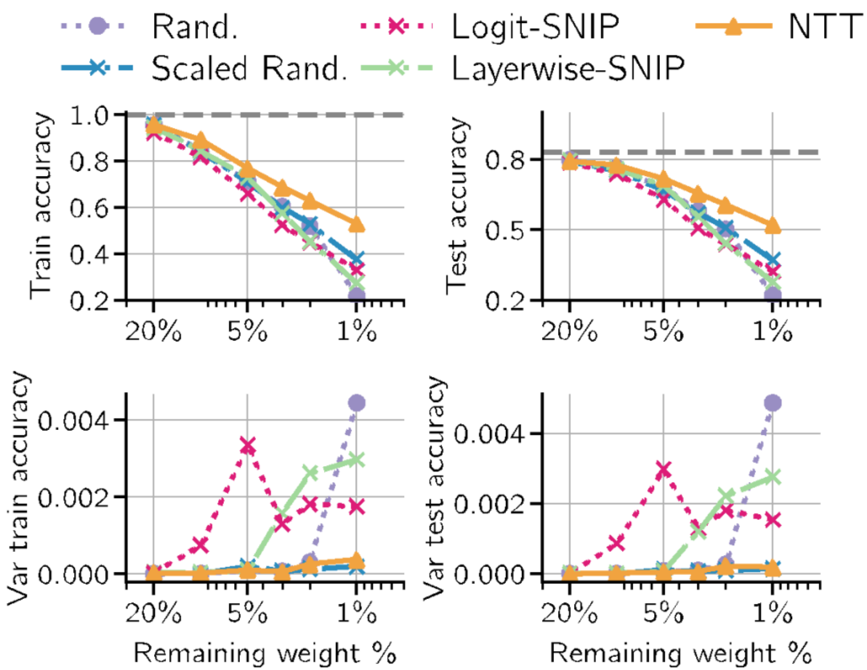
MNIST: CNN Lenet-5-Caffe

Friedrich Miescher Institute
for Biomedical Research

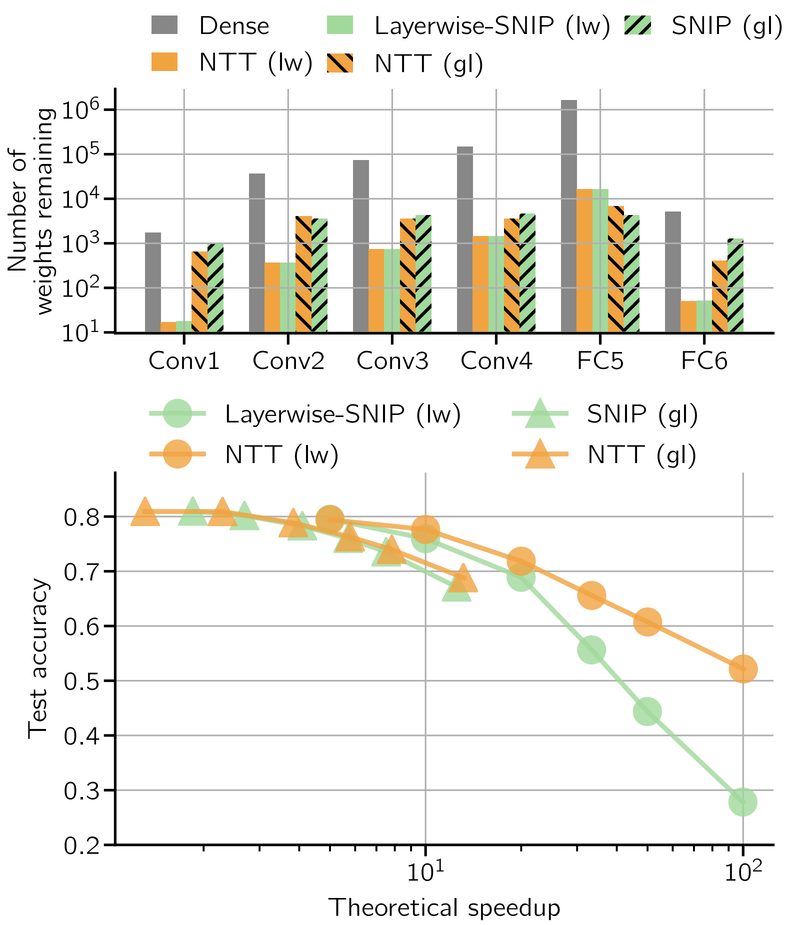




CIFAR-10: CNN Conv-4



Layerwise vs global pruning



Conclusions

- We proposed NTT, a label-free method that finds trainable sparse networks.
- Idea: transfer the training dynamics of a dense net onto a sparse net.
- Theoretical handle: Neural Tangent Kernel [Jacot et al. 2018].
- We showed that the resulting sparse nets are highly trainable on supervised learning tasks.
- We showed that NTT can significantly sparsify convolutional layers.