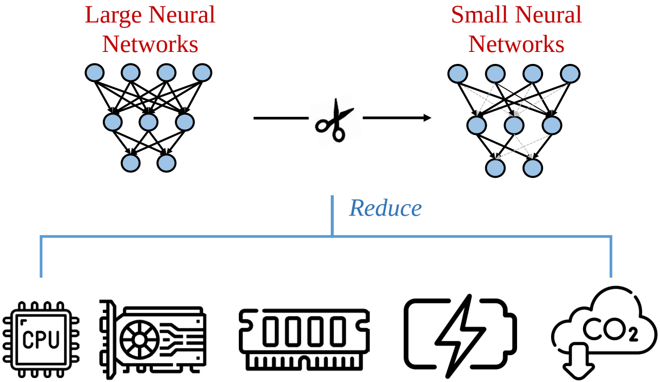
# Neural Network Pruning

- Remove (unnecessary) parameters of the neural networks.
- Reduce compute cost, memory cost, energy consumption, and carbon footprint.



# Neural Network Pruning: Benefits of Pruning

Sparse neural networks:

- 90% of the parameters can be pruned.
- Reduce computational cost by $5\times$.

Table 1: Network pruning makes neural networks sparse. Source from [Han et al.15]

| Neural Network | # Parameters | | | MACs |
|---|---|---|---|---|
| | Before Pruning | After Pruning | Reduction | Reduction |
| Alexnet | 61M | 6.7M | **9** X | **3** X |
| VGG-16 | 138M | 10.3M | **12** X | **5** X |
| GoogleNet | 7M | 2.0M | **3.5** X | **5** X |
| ResNet50 | 26M | 7.47M | **3.4** X | **6.3** X |
| SqueezeNet | 1M | 0.38M | **3.2** X | **3.5** X |

In addition, a good pruned neural network:

- Improved test accuracy
- Faster convergence rate

Table 2: Improved test accuracy of training pruned network. Source: Adapted from [Frankle et al.19], [Chen et al.20], [Chen et al.22].

| Neural Network | Dataset | Accuracy (%) | |
|---|---|---|---|
| | | Before Pruning | After Pruning |
| LetNet-5 | MINST | 98.05 | **98.41** |
| Conv-6 | Cifar-10 | 77.52 | **80.02** |
| ResNet-50 | Cifar-10 | 94.31 | **94.82** |
| ResGCN-28 | Cora | 80.02 | **81.88** |
| BERT | MNLI | 82.39 | **83.08** |

# Related Works: LTH and pruned network

Lottery Ticket Hypothesis (LTH) [Frankle et al.19]

- Existence of a good sub-network, named "winning tickets", such that it benefits the training with
  - A faster convergence rate.
  - A better generalization error.
- Magnitude-based pruning algorithm can find "winning tickets".

# Related Works: LTH and pruned network

Lottery Ticket Hypothesis (LTH) [Frankle et al.19]

- Existence of a good sub-network, named "winning tickets", such that it benefits the training with
  - ☐ A faster convergence rate.
  - ☐ A better generalization error.
- Magnitude-based pruning algorithm can find "winning tickets".

Two limitations.

- Cannot explain the improved performance of the "winning tickets".
  (**Benefits of training "winning tickets"**.)
- Cannot explain why magnitude-based pruning can find the "winning tickets".
  (**Magnitude-based pruning in finding "winning tickets"**.)

# Finding the "Winning Ticket": Iterative Magnitude Pruning

1. Train dense network with initial neuron weights until convergence with weights $\boldsymbol{W}_T$.
2. Pruning network by removing weights with smallest magnitudes in $\boldsymbol{W}_T$
3. Re-train on the pruned network using the same initialization and training set.

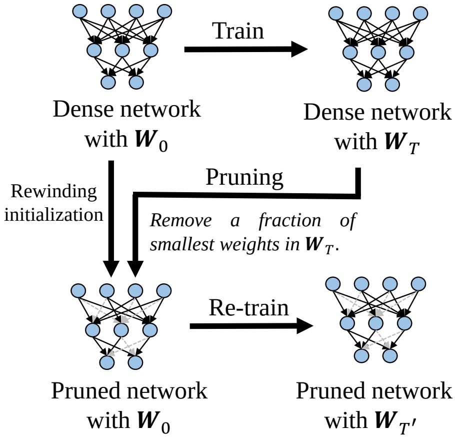



Figure 3: Iterative Magnitude Pruning (IMP) Method [Frankle et al.19]

# Finding the "Winning Ticket": Iterative Magnitude Pruning

1. Train dense network with initial neuron weights until convergence with weights $\boldsymbol{W}_T$.
2. Pruning network by removing weights with smallest magnitudes in $\boldsymbol{W}_T$
3. Re-train on the pruned network using the same initialization and training set.
4. Repeat the pruning process until observing the desired sparsity or a significant drop in performance.

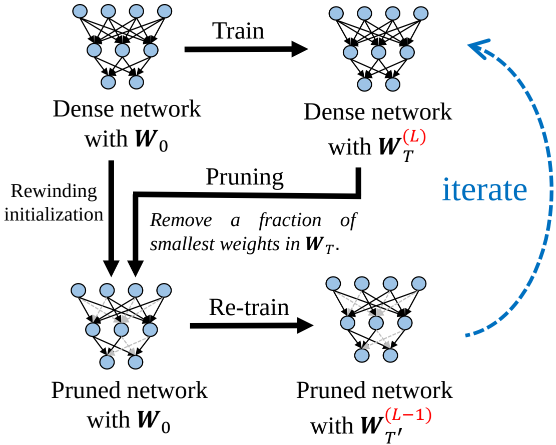



Figure 3: Iterative Magnitude Pruning (IMP) Method [Frankle et al.19]

# Highlights of Results

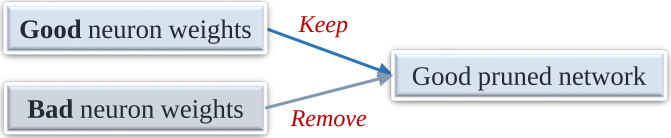
1. Benefits of “winning tickets”. (Training on “winning tickets”)

2. Magnitude-based pruning in finding “winning tickets”. (Training on dense networks.)

# Highlights of Results

1. Benefits of "winning tickets". (Training on "winning tickets")

2. Magnitude-based pruning in finding "winning tickets". (Training on dense networks.)



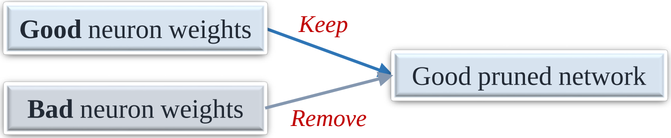
# Highlights of Results

1. Benefits of “winning tickets”. (Training on “winning tickets”)

2. Magnitude-based pruning in finding “winning tickets”. (Training on dense networks.)

*How to distinguish good neuron weights?*

*From the magnitudes!*



# Highlights of Results

1. Benefits of “winning tickets”. (Training on “winning tickets”)

What is a “winning ticket”

↓

A pruned neural network

↓

Reduced Generalization Gap

2. Magnitude-based pruning in finding “winning tickets”. (Training on dense networks.)

*How to distinguish good neuron weights?*

*From the magnitudes!*

**Good** neuron weights — *Keep* → Good pruned network

**Bad** neuron weights — *Remove* →

# Highlights of Results

1. Benefits of "winning tickets". (Training on "winning tickets")

What is a "winning ticket"

A pruned neural network → Reduced Generalization Gap

Similar expressive power as dense network → Similar Training Error

2. Magnitude-based pruning in finding "winning tickets". (Training on dense networks.)

*How to distinguish good neuron weights?*

*From the magnitudes!*

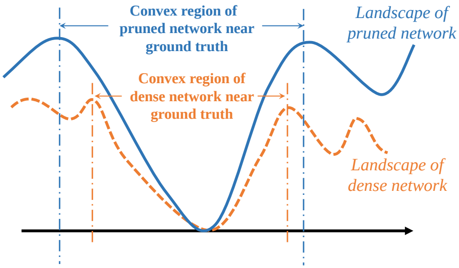
**Good** neuron weights $\xrightarrow{\textit{Keep}}$ Good pruned network

**Bad** neuron weights $\xrightarrow{\textit{Remove}}$ Good pruned network

# Benefits of Pruning (Part 1)

**1. (Convex region near ground truth)** Compared with the dense neural network, the pruned network has a larger and steeper convex region.

- The radius of the convex region scales in the order of $1-\sqrt{\frac{r}{N}}$.
  - *With pruned networks, we need fewer samples for initialization.*
- The eigenvalues of the Hessian matrix scales in the order of $1-\sqrt{\frac{r}{N}}$.
  - *With pruned networks, we have a faster convergence rate.*

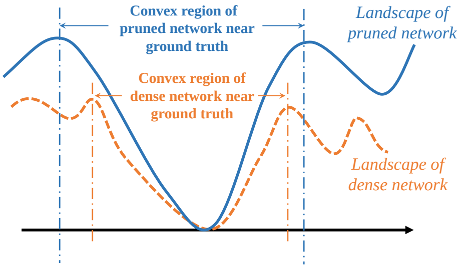



*[Zhang et al. NeurIPS'21]*

# Benefits of Pruning (Part 1)

**1. (Convex region near ground truth)** Compared with the dense neural network, the pruned network has a larger and steeper convex region.

- The radius of the convex region scales in the order of $1 - \sqrt{\frac{r}{N}}$.
  - *Sample complexity scales in the order of $r$.*
- The eigenvalues of the Hessian matrix scales in the order of $1 - \sqrt{\frac{r}{N}}$.
  - *Convergence rate scales in the order of $1 - \sqrt{\frac{r}{N}}$.*

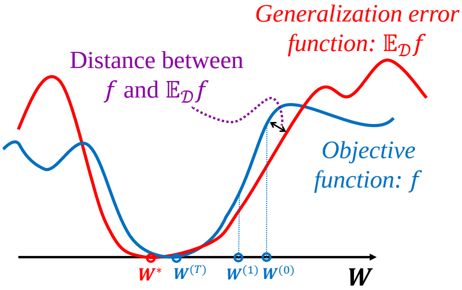



*[Zhang et al. NeurIPS'21]*

# Benefits of Pruning (Part 2)

**2. (Landscape near the ground truth)** Compared with the dense neural network, the objective function of pruned network $f$ is closer to the generalization error function $\mathbb{E}_{\mathcal{D}} f$.

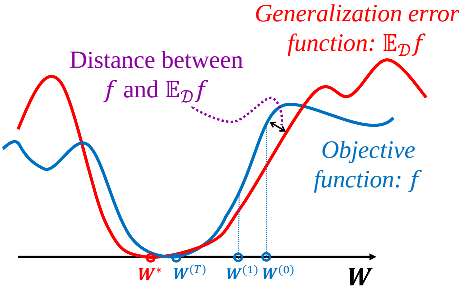
- Distance between the $\mathbb{E}_{\mathcal{D}} f$ and $f$ scales in the order of $\sqrt{\frac{r}{N}}$.
  - *With pruned networks, we need fewer samples for converging.*
- The distance of convergent point $W^{(T)}$ and ground truth $\boldsymbol{W}^*$ scales in the order of $\sqrt{\frac{r}{N}}$.
  - *With pruned networks, we have a better generalization error.*




*[Zhang et al. NeurIPS'21]*

**2. (Landscape near the ground truth)** Compared with the dense neural network, the objective function of pruned network $f$ is closer to the generalization error function $\mathbb{E}_{\mathcal{D}} f$.

- Distance between the $\mathbb{E}_{\mathcal{D}} f$ and $f$ scales in the order of $\sqrt{\frac{r}{N}}$.
  - *Sample complexity scales in the order of $r$.*
- The distance of convergent point $W^{(T)}$ and ground truth $\boldsymbol{W}^*$ scales in the order of $\sqrt{\frac{r}{N}}$.
  - *Generalization error scales in order of $\sqrt{\frac{r}{N}}$.*




[Zhang et al. NeurIPS'21]

# Benefits of "Winning Tickets": Major Theoretical Findings

*Takeaways*: Compared with dense network, training on a "winning ticket":

1. Fewer samples for initialization and convergence.
2. A faster convergence rate.
3. A reduced generalization error.

*[Zhang et al. NeurIPS'21]*

# Benefits of “Winning Tickets”: Major Theoretical Findings

*Takeaways*: Compared with dense network, training on a “winning ticket”:

1. Fewer samples for initialization and convergence.
2. A faster convergence rate.
3. A reduced generalization error.

Quantitative characterizations of the improved performance and the sparsity of ”winning ticket”:

- The radius of the convex region scales in the order of $1 - c \cdot \sqrt{r}$ ($c$ is a small constant).
- Distance between the gradient direction of objective function and generalization functions scales in the order of $\sqrt{r}$.
  - *Sample complexity scales in the order of $r$.*
- The eigenvalues of the Hessian matrix scales in the order of $1 - c \cdot \sqrt{r}$.
  - *Convergence rate scales in the order of $1 - c \cdot \sqrt{r}$.*
- The distance of convergent point $\boldsymbol{W}_T$ and ground truth $\boldsymbol{W}^*$ scales in the order of $\sqrt{r}$.
  - *Generalization error scales in order of $\sqrt{r}$.*

---

*[Zhang et al. NeurIPS’21]*

# Magnitude-based Pruning Finds "Winning tickets"

Assumption: Data is a mixture of $\underbrace{\textit{class-relevant features}}_{\text{contain labeling info}}$ and $\underbrace{\textit{class-irrelevant features}}_{\text{contain background/noise info}}$.

*[Zhang et al. ICLR'23]*

# Magnitude-based Pruning Finds "Winning tickets"

Assumption: Data is a mixture of *class-relevant features* and *class-irrelevant features*.
contain labeling info — contain background/noise info

Two types of neuron weights:

- "Good" neuron weights:
  - small angle $\longrightarrow$ learns class-relevant features.
  - have large magnitudes.
- "Bad" neuron weights:
  - large angle $\longrightarrow$ learns class-irrelevant features.
  - have small magnitudes.

**(Informal ver.) Theorem 3.1**

For a "good" neuron with weights $\boldsymbol{w}_1$ and "bad" neuron with weights $\boldsymbol{w}_2$, we have

$$\|\boldsymbol{w}_1\| - \|\boldsymbol{w}_2\| \geq 1 - \Theta(1/\sqrt{N}).$$

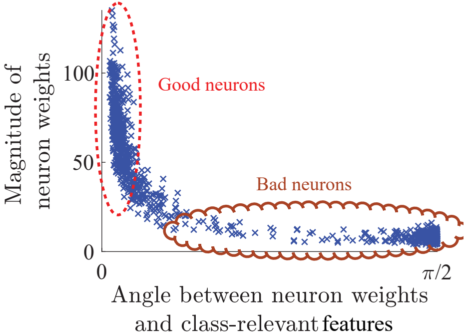



Figure 4: Distribution of the neuron weights.

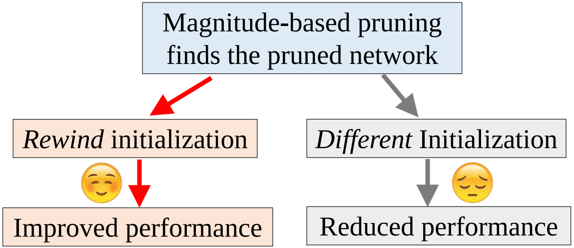
*[Zhang et al. ICLR'23]*

# Magnitude-based Pruning + Rewinding Initialization

The initial weights determine whether a neuron is “good” or ”bad”.

**(Informal ver.) Theorem 3.2**

A “good” neuron at initialization is still “good” at the next iterations.




[Zhang et al. ICLR’23]

A numerical justification on a shallow neural network.

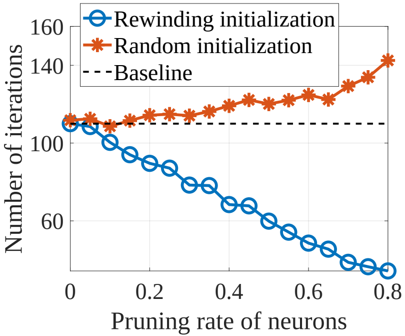



Figure 5: Number of iterations against the pruning rate.

# Empirical Results: ResNet-50 on CIFAR-10

- Ten-class image classification: Labeled data from CIFAR-10, 50-layer ResNet.
- CIFAR-10 dataset contains 60,000 32x32 color images in 10 different classes.
- ResNet: network with Residual blocks via skip connections.

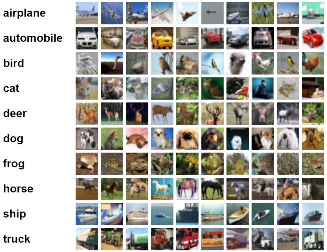



Figure 6: Illustration of the CIFAR-10 dataset (labeled subsets of Tiny Images)

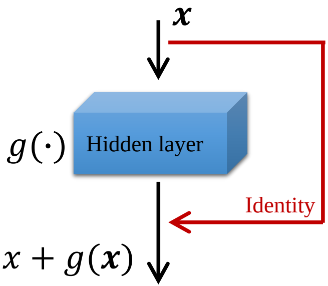



Figure 7: Illustration of Residual modular

# Empirical Results: ResNet-50 on CIFAR-10

Magnitude-based network pruning finds a good pruned neural network.

- Magnitude-based pruned network (blue line) achieves an improved test accuracy over the dense network (baseline), while a random pruning network (purple line) suggests a reduced test accuracy.
- Up to 80% neuron weights can be pruned with a similar performance as training on the dense network.

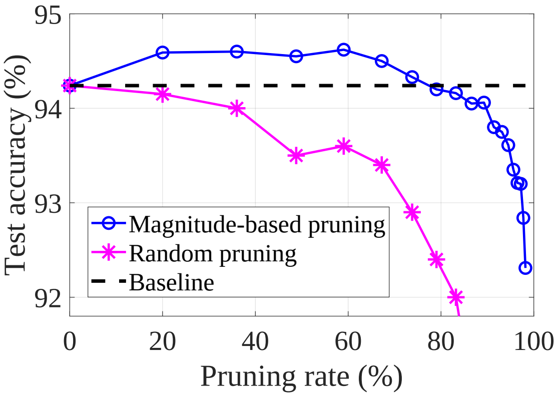



Figure 7: The test accuracy against the pruning ratio.

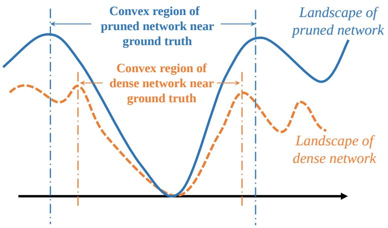
# Network Pruning for Efficient Deep Learning

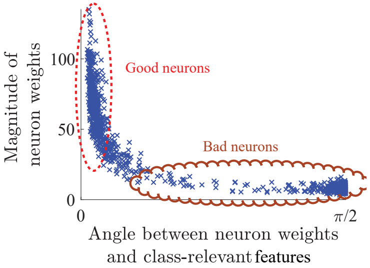
Magnitude-based network pruning $\Longrightarrow$ Improved computation efficiency.

– Sparsify neural network – Improved test accuracy – Faster convergence rate

Our contributions:

- Theoretical guarantees for using magnitude-based pruning in finding the "winning ticket".
- Theoretical explanations for a good pruned network in achieving improved test accuracy and accelerated convergence rate.




[Zhang et al. NeurIPS'21], [Zhang et al. ICLR'23]