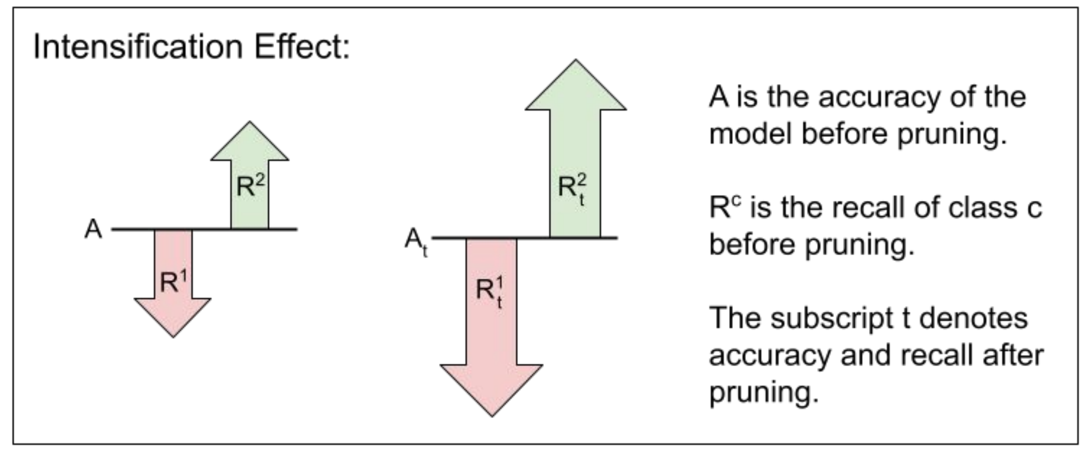
Recall Distortion in Neural Network Pruning and the Undecayed Pruning Algorithm

Aidan Good[1*], Jiaqi Lin[1*], Hannah Sieg[1], Mikey Ferguson[1], Xin Yu[2], Shandian Zhe[2], Jerzy Wieczorek[3], Thiago Serra[1]

[1]Bucknell University; [2]The University of Utah; [3]Colby College

1. Problem Statement

Does pruning induce an intensification effect on neural network models that causes a distortion in their recall performance?



How does pruning strategy, model size, and task complexity (dataset) affect this intensification effect?

2. Definitions

Let the recall balance be denoted by:

$$B^c_t(m) = R^c_t(m) - A_t(m)$$

Where $A(m)$ is accuracy for model m, $R^c(m)$ is recall for class c, and t is the pruning ratio (default $t = 1$).

Let the normalized recall balance be denoted by:

$$\bar{B}^c_t(m) = \frac{B^c_t(m)}{A_t(m)} = \frac{R^c_t(m) - A_t(m)}{A_t(m)}$$

The further away this value is from 1, the more pronounced the difference in performance is between class c and the other classes in model m at the pruning ratio t.

Let the intensification ratio be denoted by:

$$I^c_t(m) := \frac{\bar{B}^c_t(m)}{\bar{B}^c(m)} \equiv \frac{\text{Normalized recall balance } \textbf{after} \text{ pruning}}{\text{Normalized recall balance } \textbf{before} \text{ pruning}}$$

This metric is used to evaluate if pruning widens the performance gap between classes, and our focus is on if E[I] = 1 (no intensification) **or** if E[I] <> 1, then we can analyse when E[I] > 1 (intensification) but also when E[I] < 1 (de-intensification).

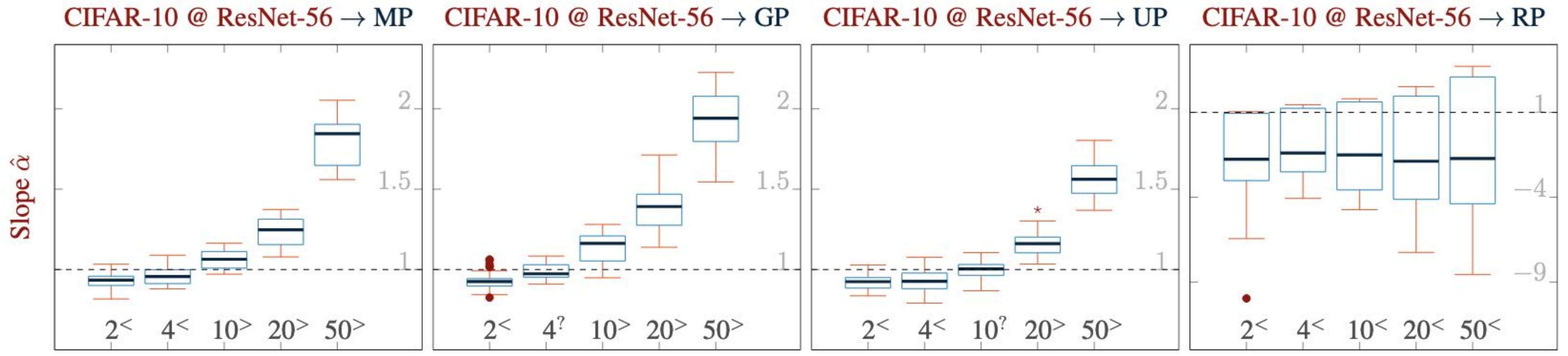
$\hat{\alpha}$ is the slope of the linear regression of $\bar{B}^c_t(m)$ on $\bar{B}^c(m)$, giving a weighted mean of $I^c_t(m)$ (across c for a given m and t).

For boxplots, means below 1 (dashed-line) show a de-intensification effect
For scatter plots, slopes below 1 show a de-intensification effect

3. What Affects the Intensification Ratio?

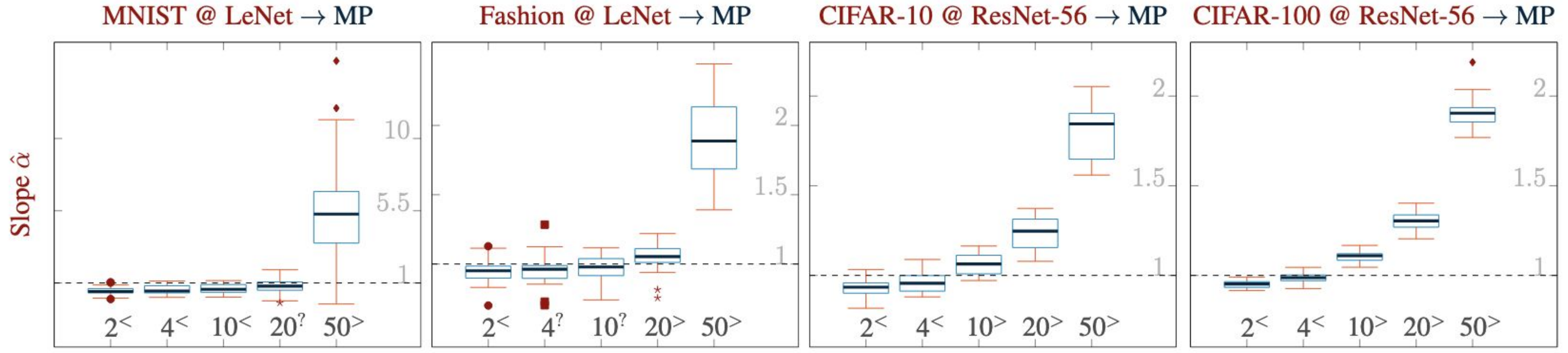
Superscripts <, >, or ? denote where 99% CIs were below 1, above 1, or overlapped 1. MP is magnitude, GP is gradient, UP is undecayed, and RP is random pruning.

1. **How does pruning strategy affect the intensification ratio?**



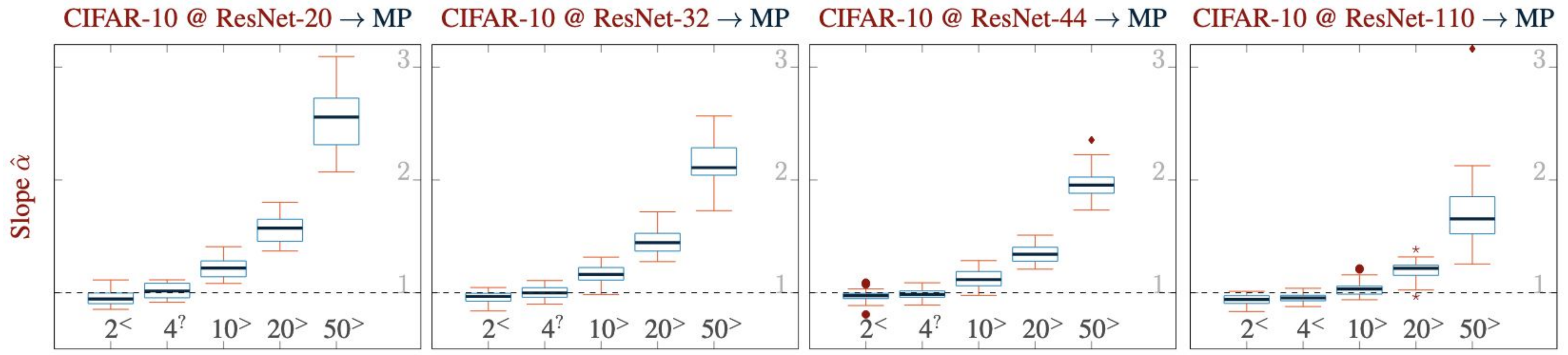
We observe an intensification effect for all pruning strategies except RP.

2. **How does task complexity affect the intensification ratio?**



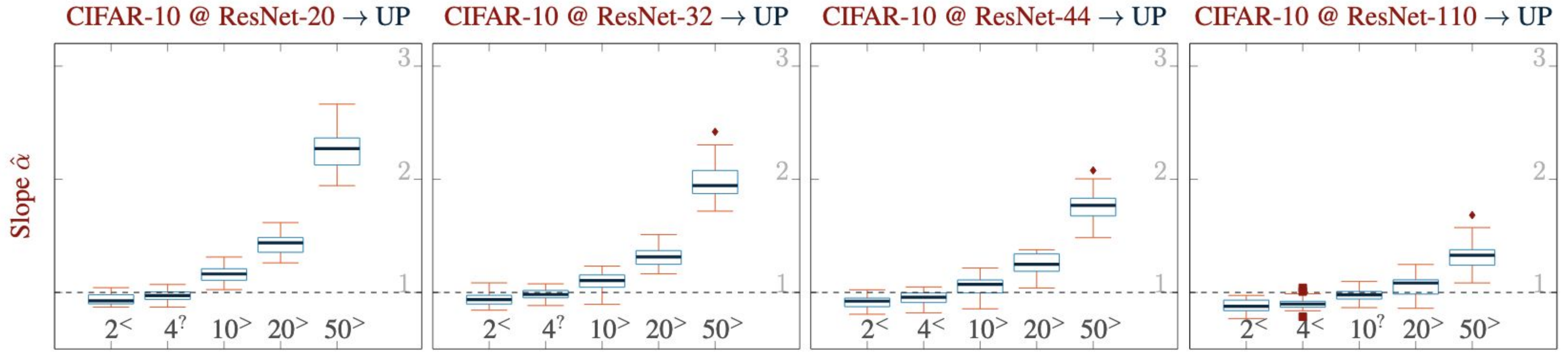
As datasets get more complex, we see higher intensification per pruning rate.

3. **How does model size affect the intensification ratio?**



Smaller model sizes show more intensification per pruning rate.

4. **How does undecayed pruning perform?**



Comparing to boxplot 3, UP has less of an intensification effect than MP.

4. Undecayed Pruning vs Magnitude Pruning

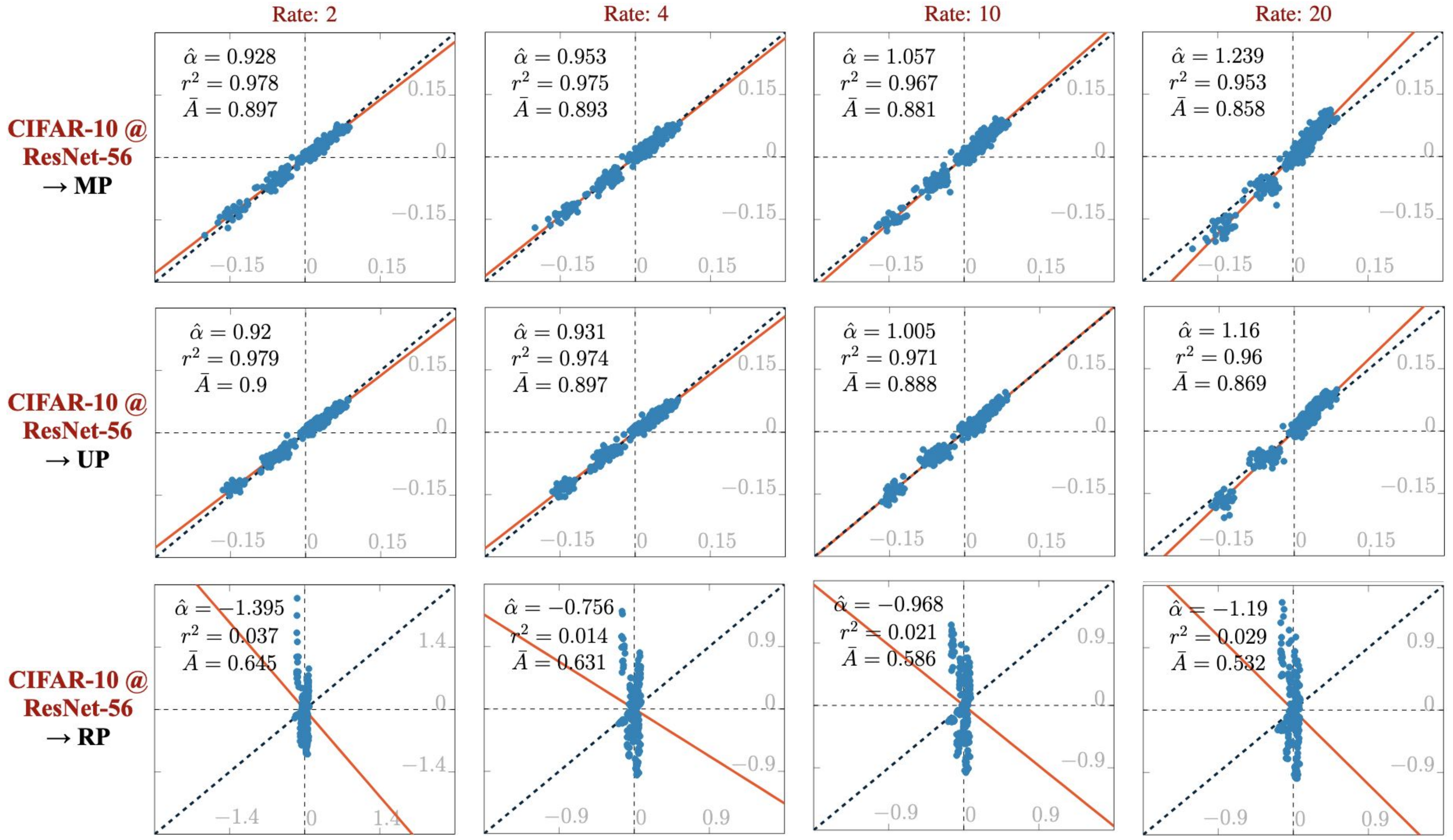
To better determine the impact of parameters for pruning, we propose a combination of magnitude and gradient pruning:

$$UP = GP + \varepsilon MP$$

Where ε is the weight decay hyperparameter.

To determine its effectiveness, we compare it with MP and include RP:

x-axis is $\bar{B}^c_t(m)$ before pruning, y-axis is $\bar{B}^c_t(m)$ after pruning.



We find that **UP** has a **smaller mean intensification ratio** ($\hat{\alpha}$) than **MP**, while having **better accuracy** ($\bar{A}$), at the same pruning rate.
RP has the **lowest intensification ratio** of them all, implying that it heavily reduces recall distortion, but the model accuracies are below any usable threshold.

5. Conclusion

- **We find statistically significant evidence for I >1 at high pruning rates.**
- **Different pruning strategies have different effects, with UP performing best.**
- **More complex tasks and smaller model sizes tend to have higher I at same pruning rates.**
- **At low pruning rates (t <= 4) we see a de-intensification effect.**

Main Paper: https://arxiv.org/abs/2206.02976