Can Neural Nets Learn the Same Model Twice? Investigating Reproducibility and Double Descent from the Decision Boundary Perspective

Gowthami Somepalli, Liam Fowl, Arpit Bansal, Ping Yeh-Chiang, Yehuda Dar, Richard Baraniuk, Micah Goldblum, Tom Goldstein

CVPR 2022(Oral)

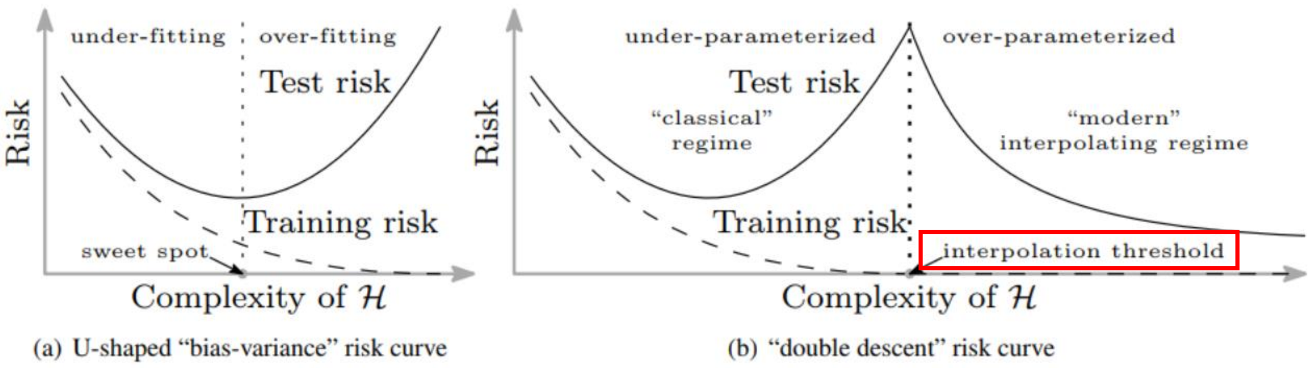
Sanghyeon Lee

13 Jun 2022

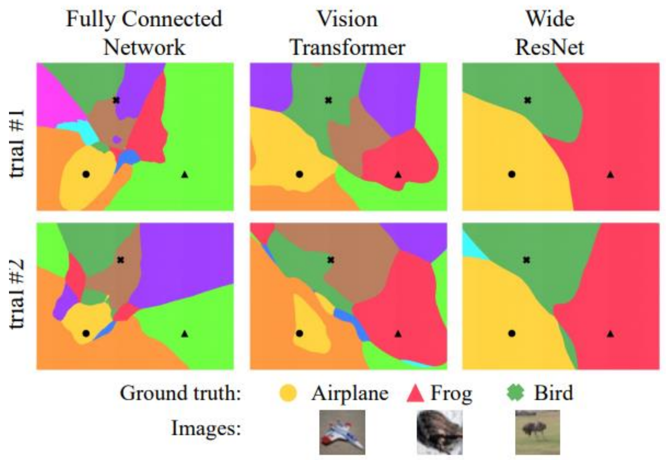
Introduction

- Do neural networks produce decision boundaries that are consistent across random initializations?
- Neural network learn the same model twice?
- Do different neural architectures have measurable differences in inductive bias?
- What happens near the interpolation threshold?

➔ This paper investigate the above questions through proposed decision boundary visualizing methodology!



(a) U-shaped "bias-variance" risk curve (b) "double descent" risk curve



Method

- Authors extend the mixup method
- First, Sample a triplet $(x_1, x_2, x_3) \sim D^3$, $i.i.d$ of images from the data distribution
- Let $\overrightarrow{v_1} = x_2 - x_1, \overrightarrow{v_2} = x_3 - x_1$
- We can span the plane from these two vectors.

$$\alpha \cdot \max\left(\overrightarrow{v_1} \cdot \overrightarrow{v_1}, \left|\mathrm{proj}_{\overrightarrow{v_1}} \overrightarrow{v_2} \cdot \overrightarrow{v_1}\right|\right) \overrightarrow{v_1} + \beta\left(\overrightarrow{v_2} - \mathrm{proj}_{\overrightarrow{v_1}} \overrightarrow{v_2}\right), \qquad -0.1 \le \alpha, \beta, \le 1.1$$

- Authors prove that neural nets are smoothly changing with any random pixels.

Lemma 2.1 *Let $f : [0,1]^n \to [0,1]$ be a neural network satisfying $|f(x) - f(y)| \le \frac{L}{\sqrt{n}}\|x - y\|$. Let $\bar{f}$ denote the median value of f on the unit hypercube. Then, for an image $x \in [0,1]^n$ of uniform random pixels, we have $|f(x) - \bar{f}| \le t$ with probability at least*

$$1 - \frac{Le^{-2\pi n t^2/L^2}}{\pi t\sqrt{n}}.$$

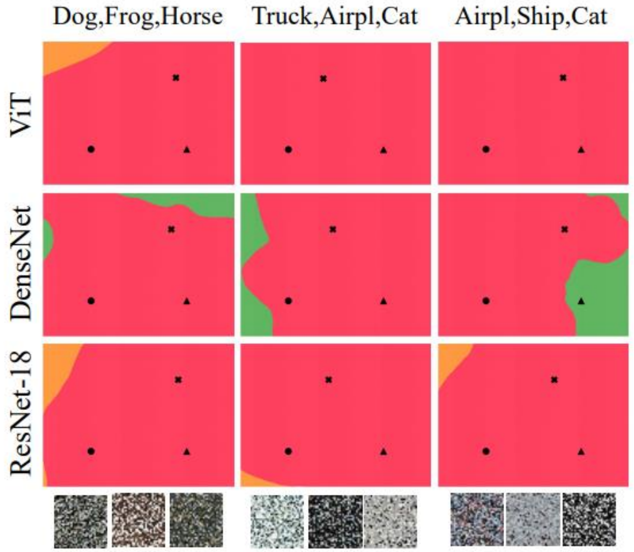
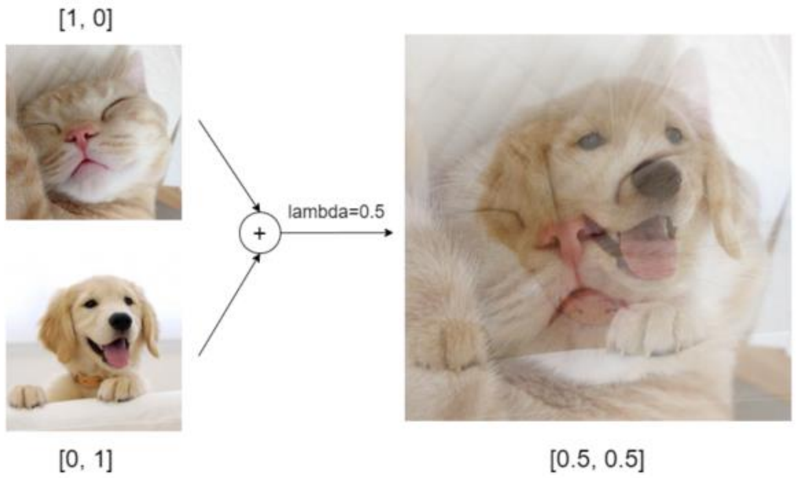


Figure 2. Off-manifold decision boundaries near “random” images created by shuffling pixels in CIFAR-10 images. Each column’s title shows the labels of the unshuffled base images. Below each column we show the shuffled image triplet. Color-class mapping is as follow Red:Frog, Green:Bird, Orange:Automobile.



Zhang et al.mixup: Beyond Empirical Risk Minimization

Experiments

- Do neural networks produce decision boundaries that are consistent across random initializations?
- Do different neural architectures have measurable differences in inductive bias?

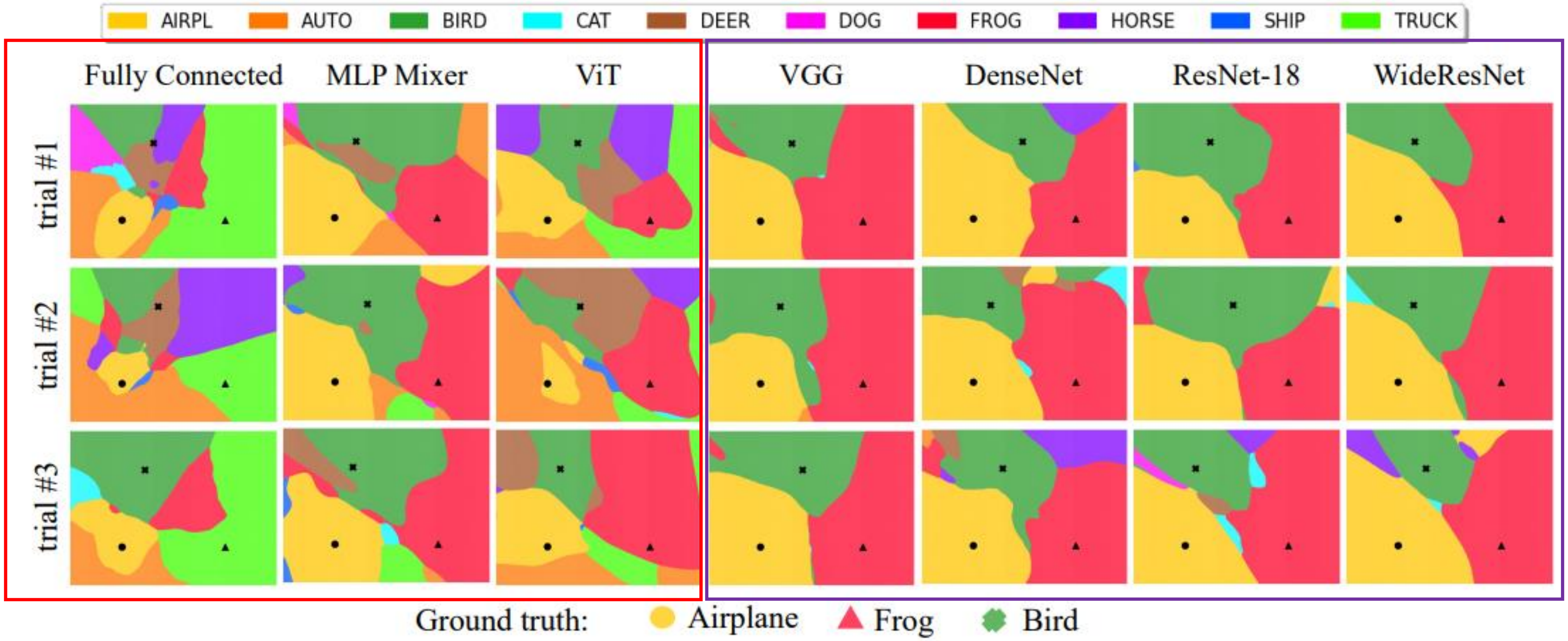


Figure 3. Decision regions through a triplet of images, for various architectures (columns) and initialization seeds (rows).

Experiments

Quantitative analysis of decision regions

- Reproducibility Score: 500 the random triplets, 2500 points

$$R(\theta_1, \theta_2) = \mathbb{E}_{T_i \sim \mathcal{D}} \left[(|f(S_i, \theta_1) \cap f(S_i, \theta_2)|)/|S_i| \right]$$

1) The inductive biases of all the convolutional architectures are highly similar

2) Wider convolutional models appear to have higher reproducibility in their decision regions

3) Skip connections have little impact on the shape of decision regions (ResNet, DenstNet, and VGGNet)

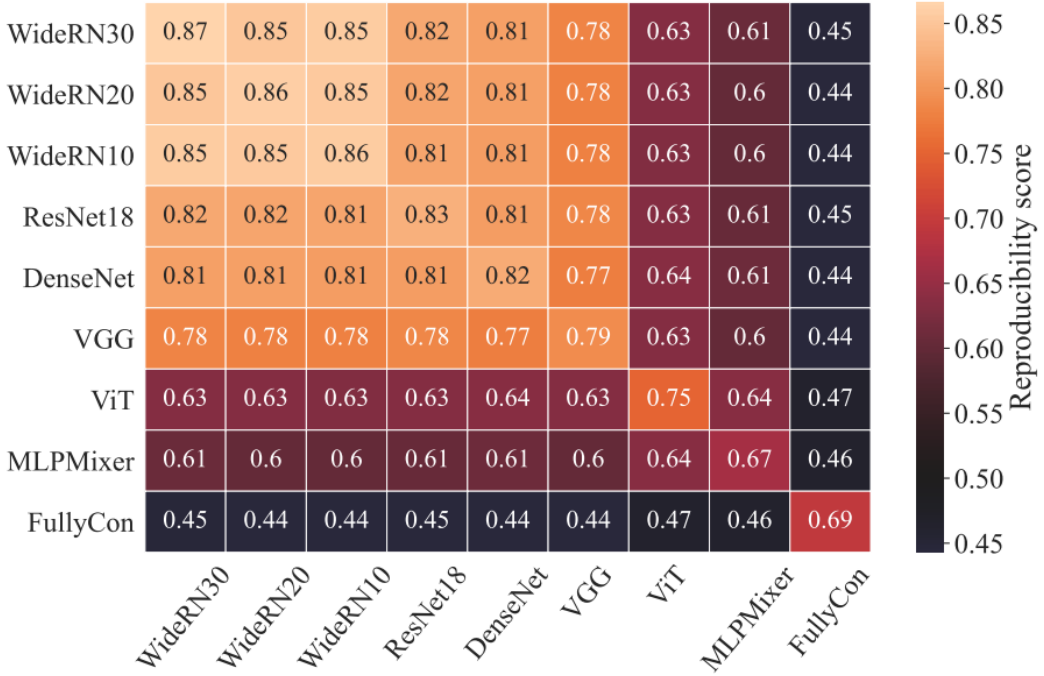


Figure 4. Reproducibility across several popular architectures.

Experiments

- Dose distillation preserve decision boundries?
- The effect of the optimizer

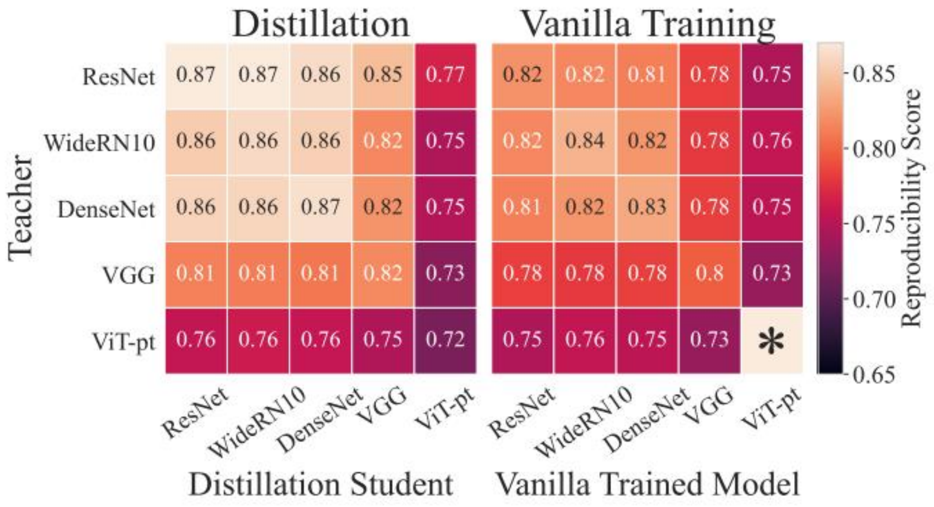


Figure 5. Differences in reproducibility comparing distilled model to vanilla trained model. *The reproducibility score is not applicable for this diagonal entry because we start from the same pretrained model.

Reproducibility			
	Adam	SGD	SGD + SAM
ResNet-18	79.81%	83.74%	**87.22%**
VGG	81.19%	80.92%	**84.21%**
MLPMixer	67.80%	66.51%	**68.06%**
VIT	69.55%	75.13%	**75.19%**

Test Accuracy			
	Adam	SGD	SGD + SAM
ResNet-18	93.04	95.30	**95.68**
VGG	92.87	93.13	**93.90**
MLPMixer	**82.22**	82.04	82.18
VIT	70.89	**75.49**	74.72

Table 1. Reproducibility of different models when using different optimizers. SGD produces more reproducible decision boundaries relative to Adam, and SGD+SAM almost always consistently increase reproducibility of the model relative to SGD.

Experiments

Double descent

- ResNet-18 with width = k(1~64)
- Label noise is important for creating easily observable double descent in realistic models
- **How do decision boundaries change as we cross the interpolation threshold?**

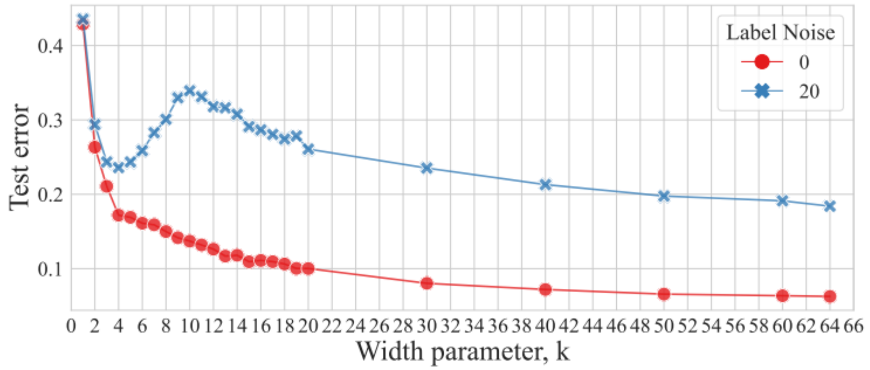
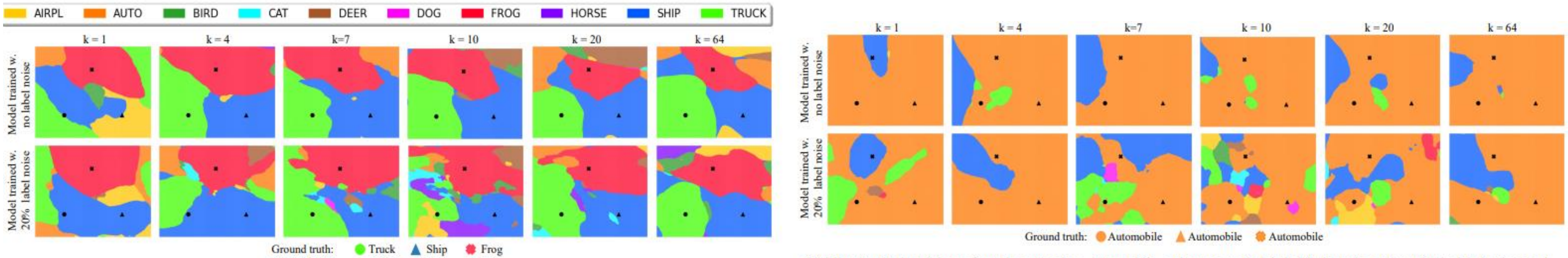


Figure 6. Test error curves with 0 and 20% label noise in training.



(a) All the points in the triple are from different classes, and are correctly labeled in the train set (even in the label noise case).

(b) All points in the triple are from the same class, Automobile, and are correctly labeled in the train set (even in the label noise case).

Figure 7. **Decision boundaries for models of varying width.** Label noise induces chaotic fragmentation of decision regions as we the threshold of interpolation (k=10), while very narrow and wide models remain smooth.

Experiments

- K= 10 (the interpolation threshold), random nosie, we can observe the instability (Fig 8.)
- What happens to the decision boundaries around mislabeled images? (Fig 9.)

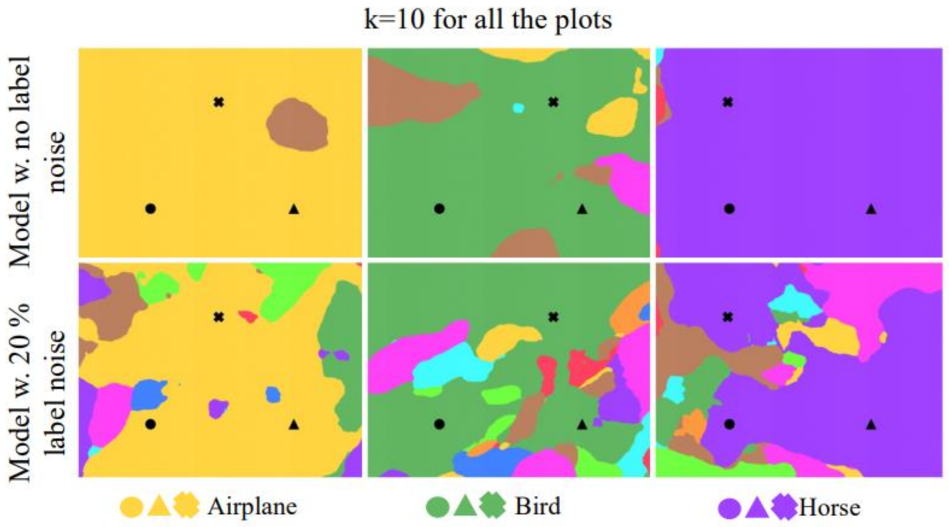


Figure 8. Decision boundaries of 3 correctly labeled points at $k = 10$ on models with and without label noise.

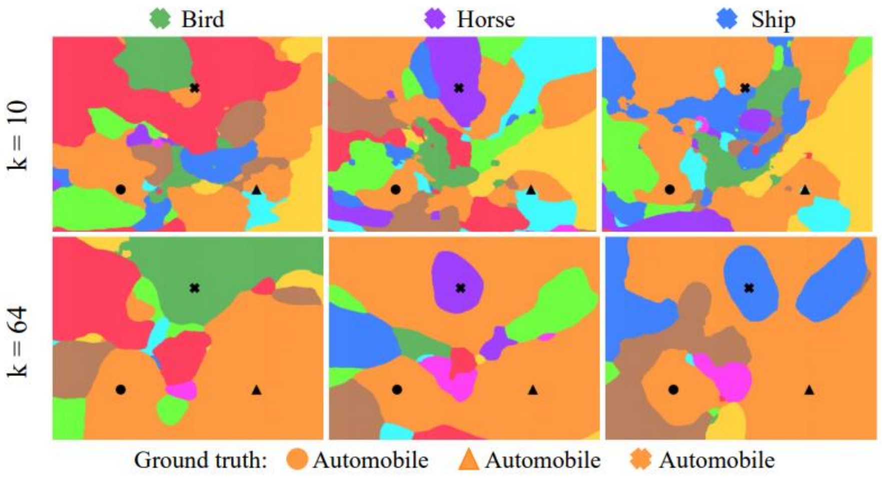


Figure 9. Decision boundaries with 1 mislabeled automobile and 2 correctly labeled automobiles. Each column represents a different image triplet. The mislabeled point is marked by **x**.

Conclusion

- Do neural networks produce decision boundaries that are consistent across random initializations?
  ➔ Yes

- Neural network learn the same model twice?
  ➔ CNN>VIT

- Do different neural architectures have measurable differences in inductive bias?
  ➔ Yes

- What happens near the interpolation threshold?
  ➔ We can observe the dramatic fragmentation of class regions

Thank you