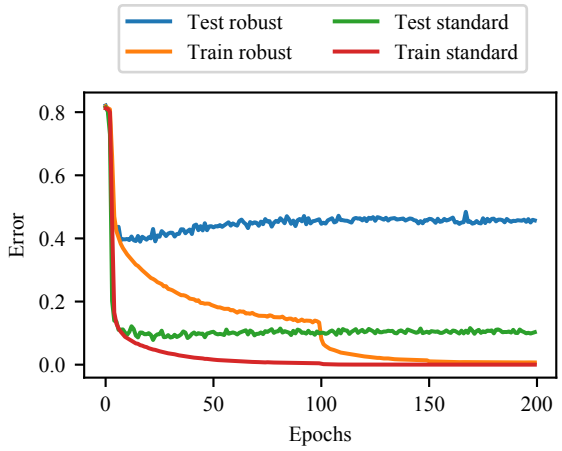

Figure 9. Learning curves for training an SVHN classifier which is adversarially robust to ℓ_∞ perturbations of radius 8/255. Note that robust overfitting occurs before the learning rate has decayed, likely due to the lower initial learning rate.

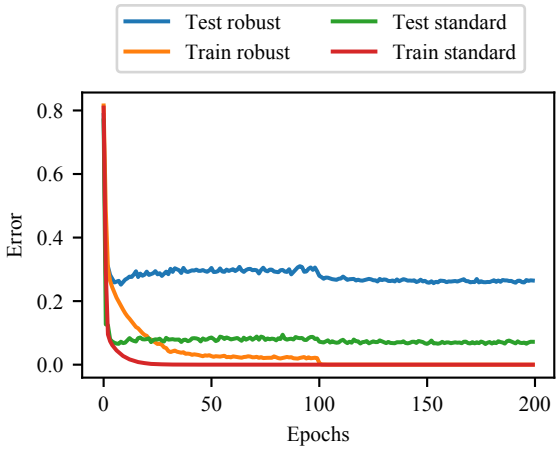

Figure 10. Learning curves for training an SVHN classifier which is adversarially robust to ℓ_2 perturbations of radius 128/255. Robust overfitting occurs early here as well, with robust test error increasing after the 9th epoch.

A. Full set of results for Table 1

In this section, we extend Table 1 to additionally include standard error and results from different adversarial training schemes (FGSM and TRADES), as shown in Table 3. The final error is an average over the final 5 epochs of when the model has converged, along with the standard deviation. The best error is the lowest test error of all model checkpoints during training. For convenience we also show the difference in the final model's error and the best model's error, which indicates the amount of degradation incurred by robust overfitting.

The remainder of this section contains the experimental details for reproducing these experiments, as well as the learning curves for each experiment as visual evidence of robust overfitting. We default to using pre-activation ResNet18s for our experiments, with the exception of Wide ResNets with width factor 10 for ℓ_∞ adversaries on CIFAR-10 (for a proper comparison to what is reported for TRADES), and ResNet50s for ImageNet. For CIFAR-10 and CIFAR-100, we train with the SGD optimizer using a batch size of 128, a step-wise learning rate decay set initially at 0.1 and divided by 10 at epochs 100 and 150, and weight decay $5 \cdot 10^{-4}$. For SVHN, we use the same parameters except with a starting learning rate of 0.01 instead. For ImageNet, we use the same learning configuration used to train the pretrained models and simply run them for longer epochs and lower learning rates using the publicly released repository available at `https://github.com/madrylab/robustness`.

ℓ_∞ adversary We consider the ℓ_∞ threat model with radius 8/255, with the PGD adversary taking 10 steps of size 2/255 on all datasets except for ImageNet. For ImageNet, we fine-tune the pretrained model from `https://github.com/madrylab/robustness` (Engstrom et al., 2019) and continue training with the exact same parameters with a learning rate of 0.001, which uses an adversary with 5 steps of size 0.9/255 within a ball of radius 4/255.

ℓ_2 adversary We consider the ℓ_2 threat model with radius 128/255, with the PGD adversary taking 10 steps of size 15/255 on all datasets except for ImageNet. For Imagenet, we fine-tune the pretrained model from `https://github.com/madrylab/robustness` (Engstrom et al., 2019) and continue training with the exact same parameters with a learning rate of 0.001, which uses an adversary with 7 steps of size 0.5 within a ball of radius 3.

A.1. SVHN experiments

Figures 9 and 10 contain the convergence plots for the PGD-based adversarial training experiments on SVHN for ℓ_∞ and ℓ_2 perturbations respectively. We find that robust overfitting occurs even earlier on this dataset, before the initial learning rate decay, indicating that the learning rate threshold at which robust overfitting begins to occur has already been passed. The best checkpoint for ℓ_∞ achieves 39.0% robust error, which is a 6.6% improvement over the 45.6% robust error achieved at the end of training.

A.2. CIFAR-100 experiments

Figures 11 and 12 contain the convergence plots for the PGD-based adversarial training experiments on CIFAR-100 for ℓ_∞ and ℓ_2 perturbations respectively. We find that ro-

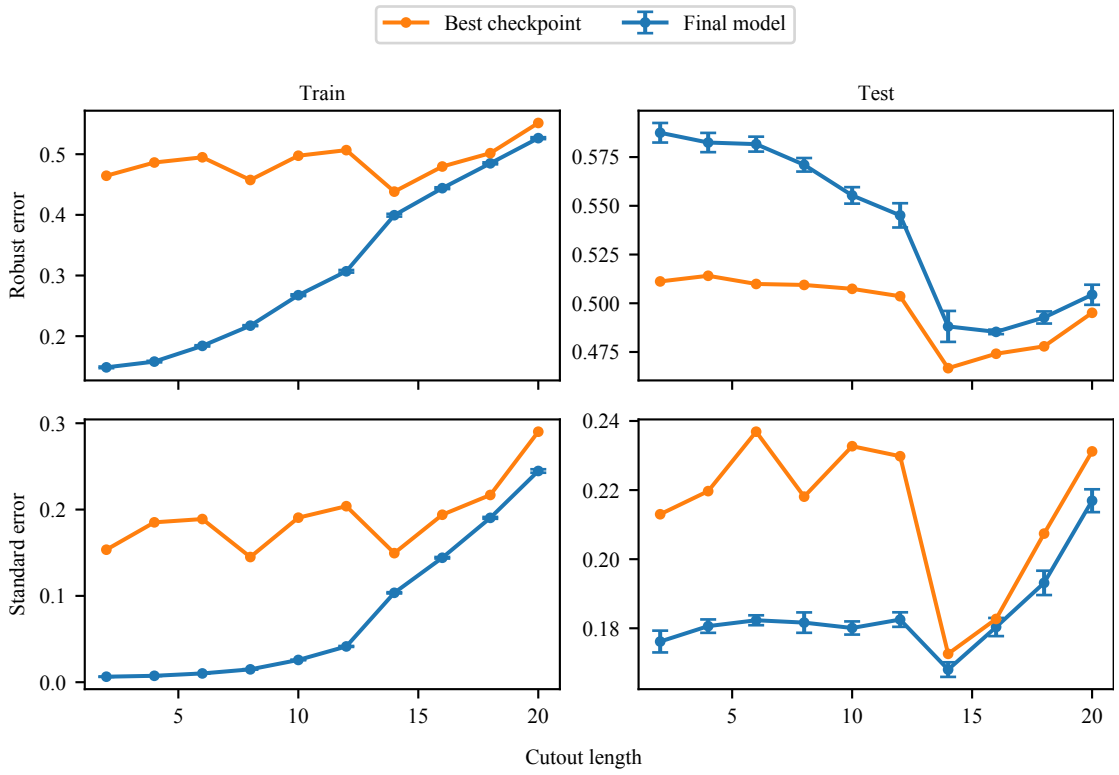

Figure 27. Standard and robust performance on the train and test set for varying cutout patch lengths.

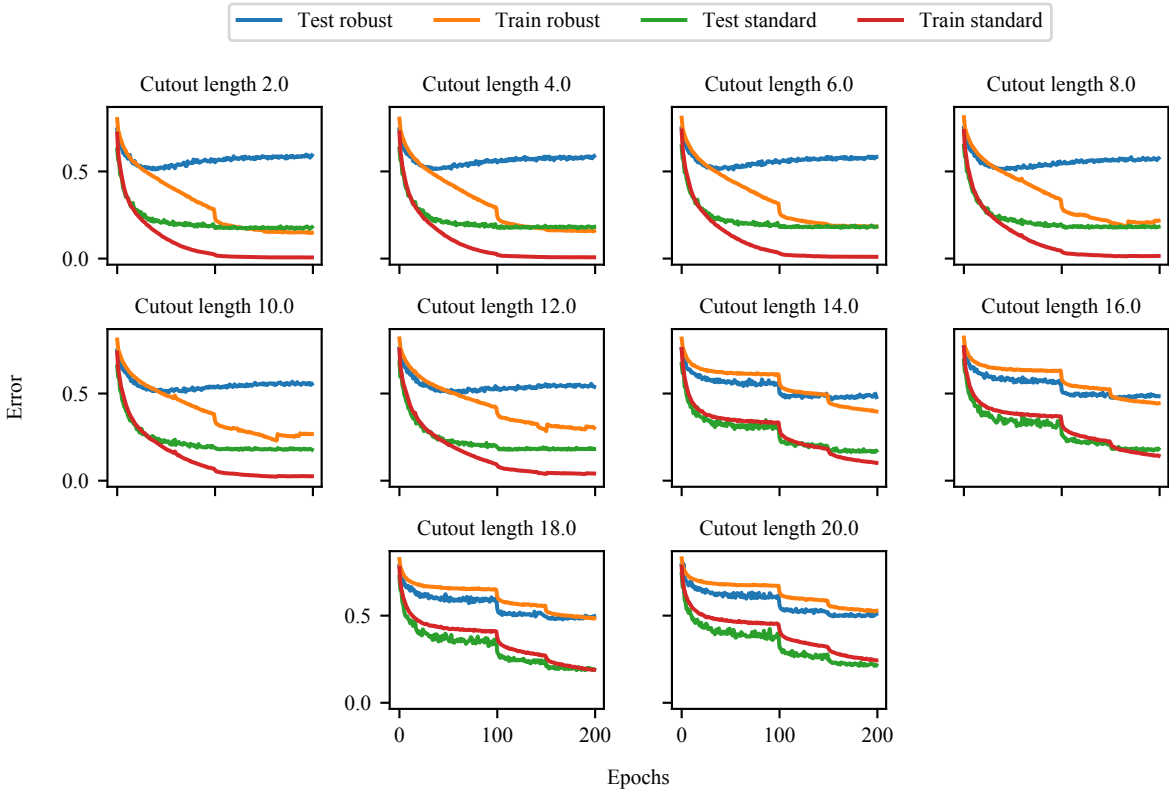

Figure 28. Learning curves for adversarial training using cutout data augmentation with different cutout patch lengths.

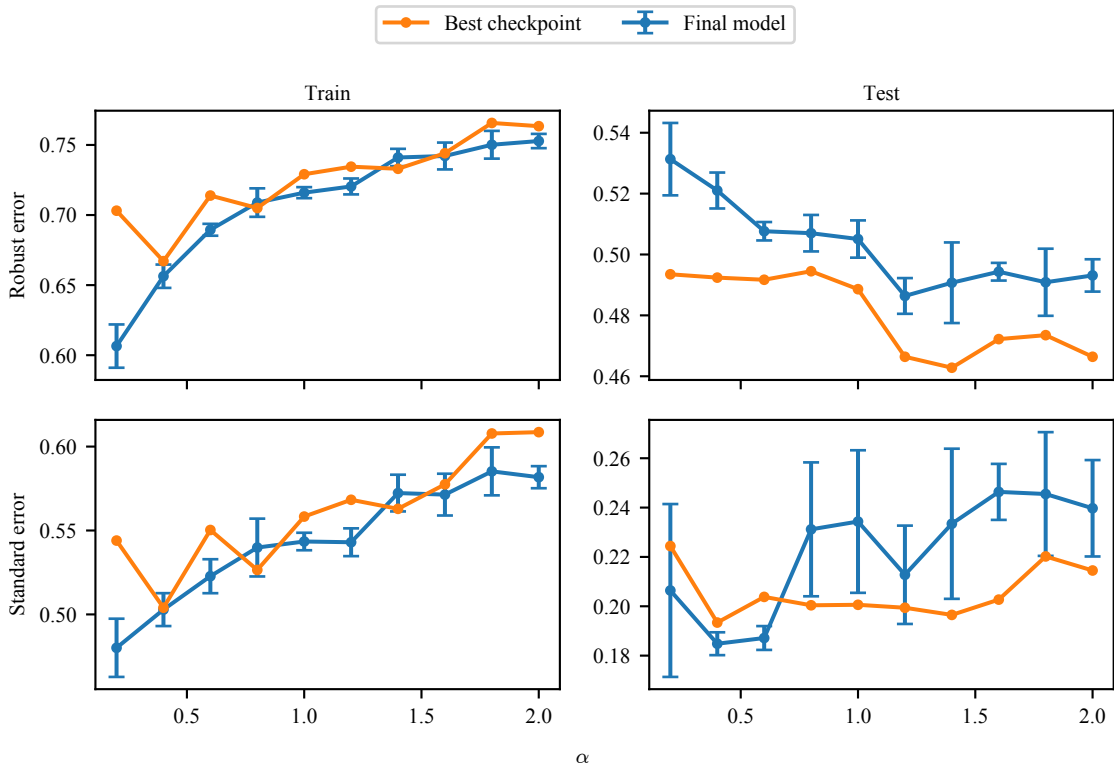

Figure 29. Standard and robust performance on the train and test set for varying degrees of mixup.

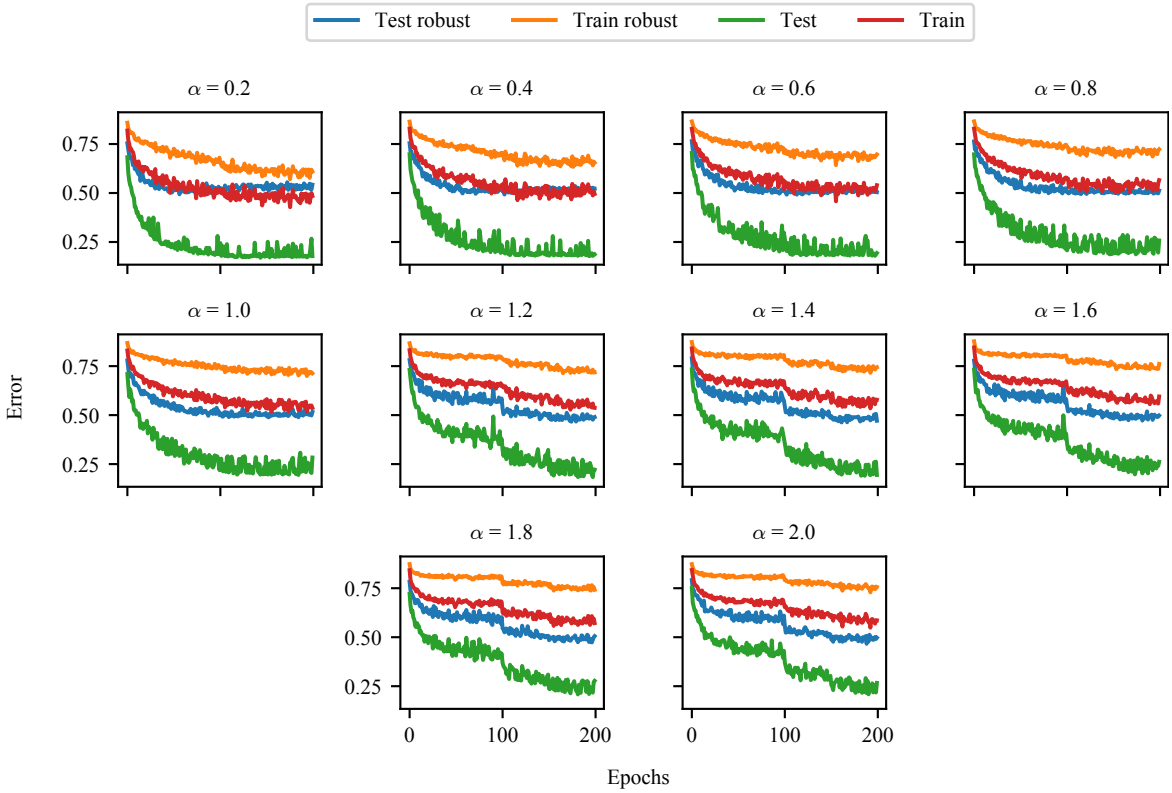

Figure 30. Learning curves for adversarial training using mixup with different choices of hyperparameter α.