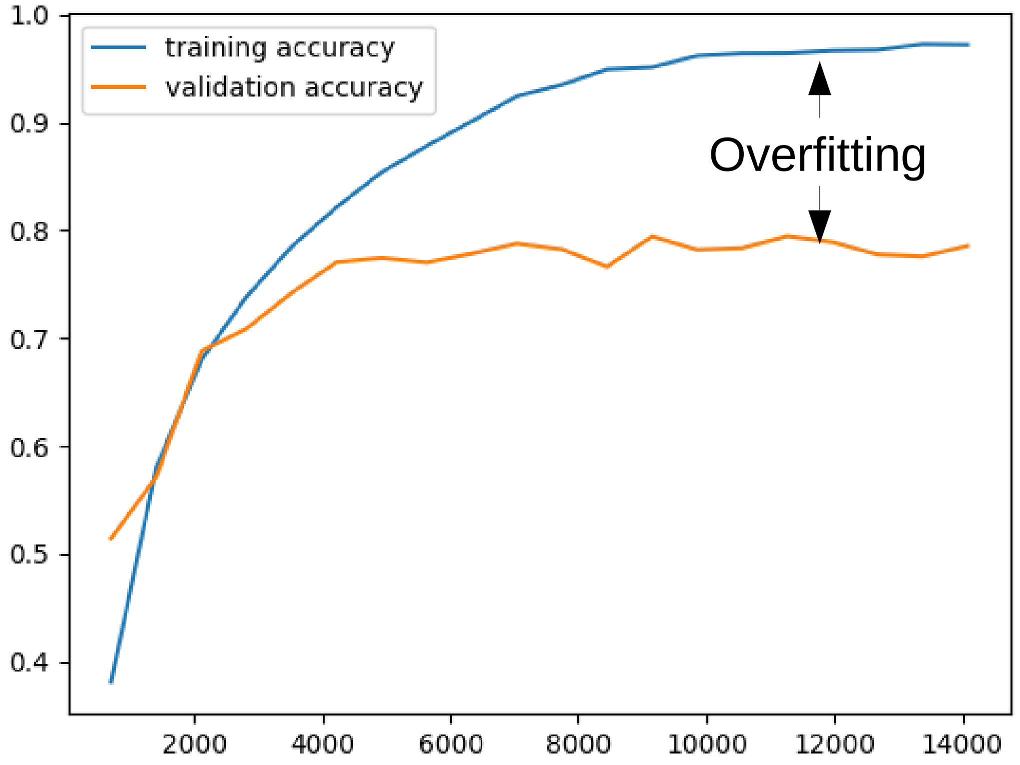
Overfitting

What is Overfitting

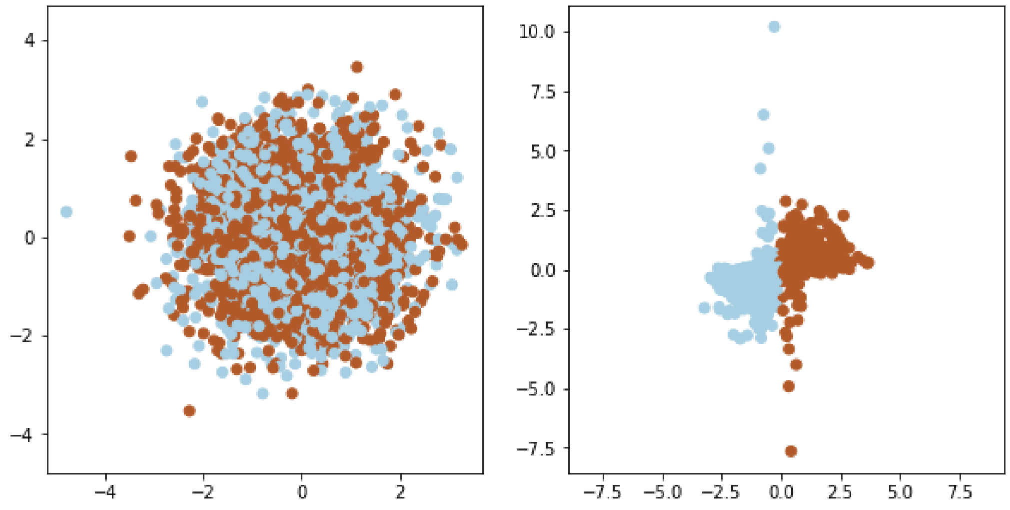
- The network performs better on the training set than the validation and test sets.
- Not *necessarily* bad as long as your validation error is low.



Why do Networks Overfit

Patterns exist in the training set which are not there in the test set.

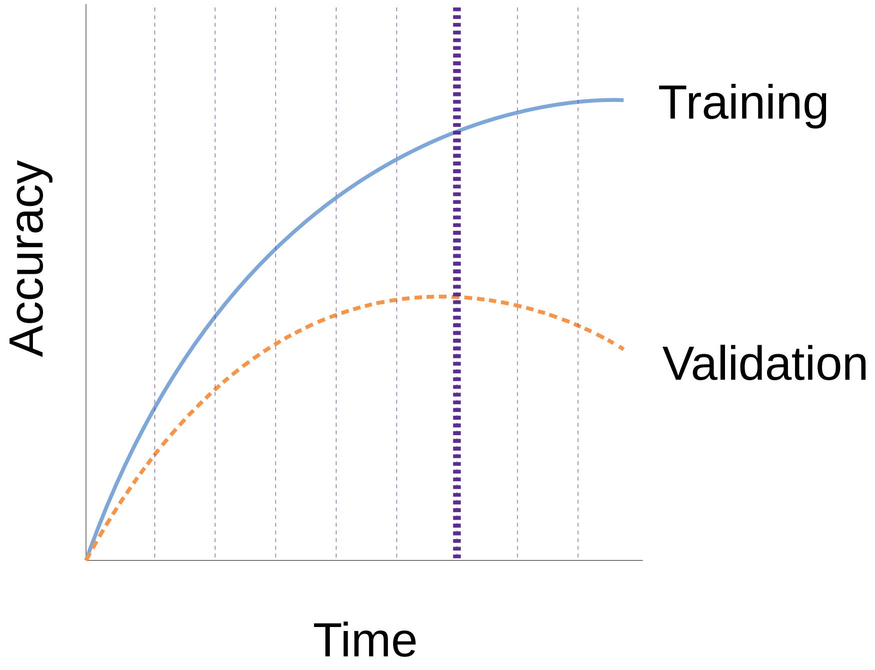
Sampling bias



- 4000 points
- 4000D space
- Linear classifier achieves 99+% accuracy

Early Stopping

- Stop training when the validation accuracy starts to decrease.



Symptoms of Overfitting

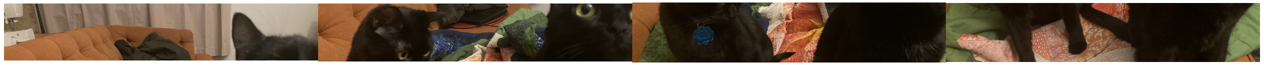
Cat

Dog

Airplane

Capturing Important Patterns

- Using the network structure
  - All-convolutional networks



- Using the data
  - Data augmentation

Data Augmentation

Randomly transform input data, keeping the same labels

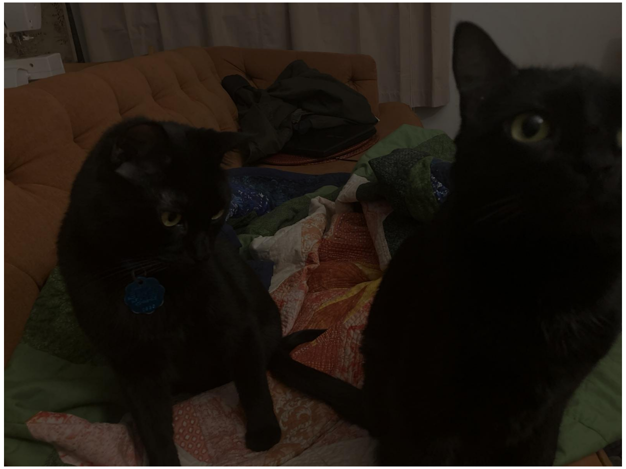
Original
Label: Cat

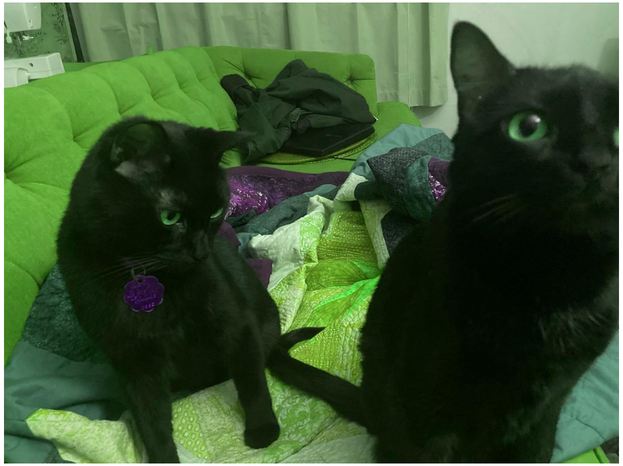
Flipped
Label: Cat

Scaled
Label: Cat

Color Augmentation

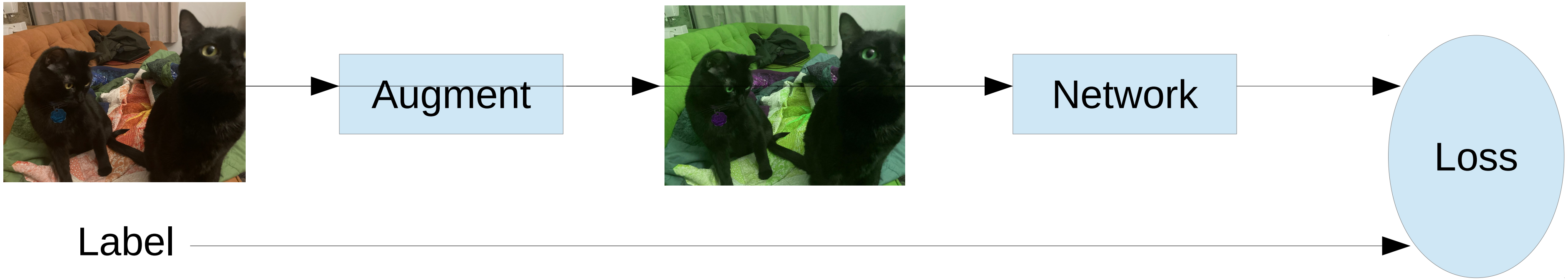
In addition to the geometric augmentations we just saw, we also have color augmentations





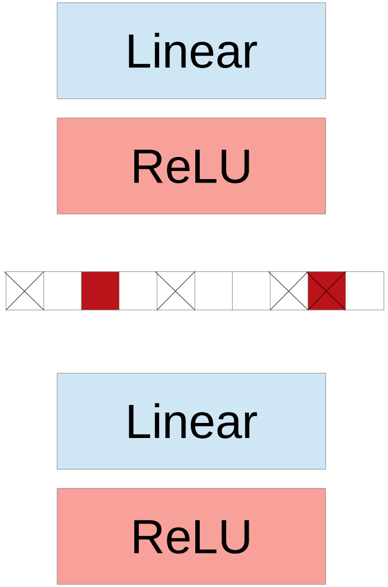
Data Augmentation in Practice

- Randomly change each input in each iteration



- Network never sees the same image twice
- For segmentation, the labels need to be augmented accordingly

Dropout



Set some activations to zero with probability p

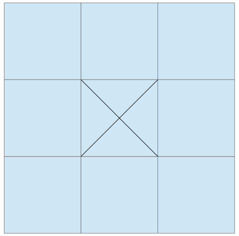
At test time, keep all activations

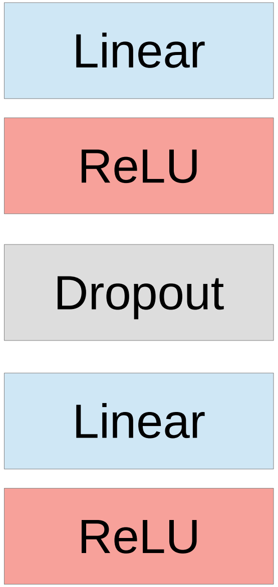
$$E\left[\left\|\text{dropout}_p(x)\right\|\right]=(1-p)\,E\left[\left\|x\right\|\right]$$

PyTorch handles scaling

Dropout in Practice

- Add before large fully-connected layers
- Sometimes before 1x1 convolutions
- Not before most convolutions

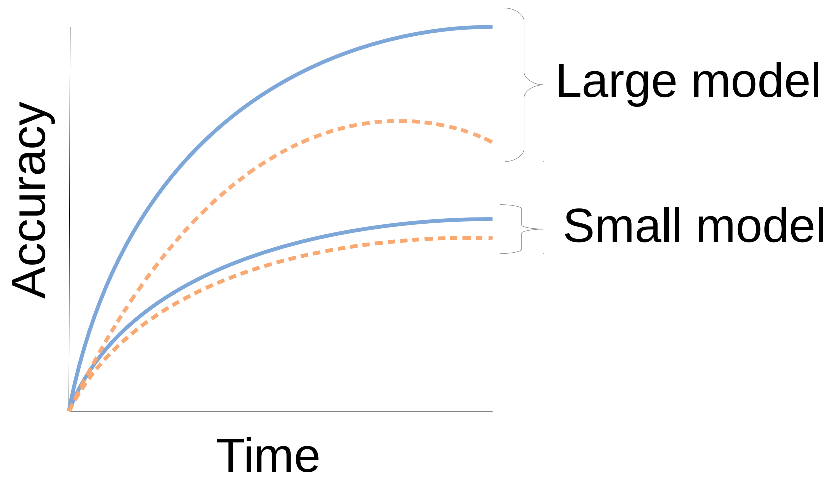




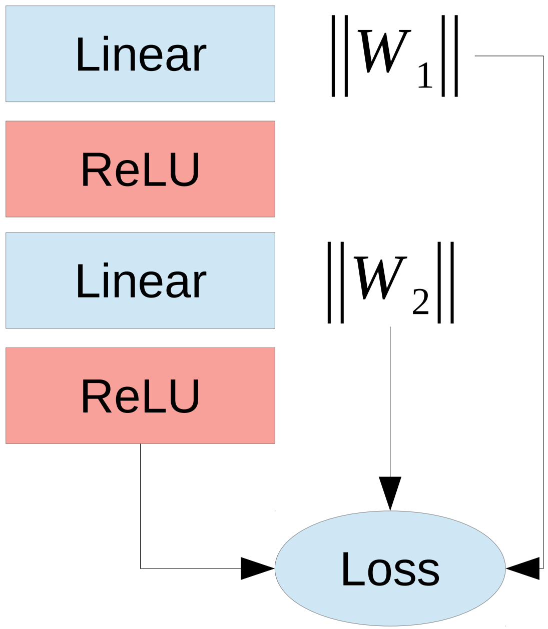
Weight Decay

Simple models overfit less

Small models

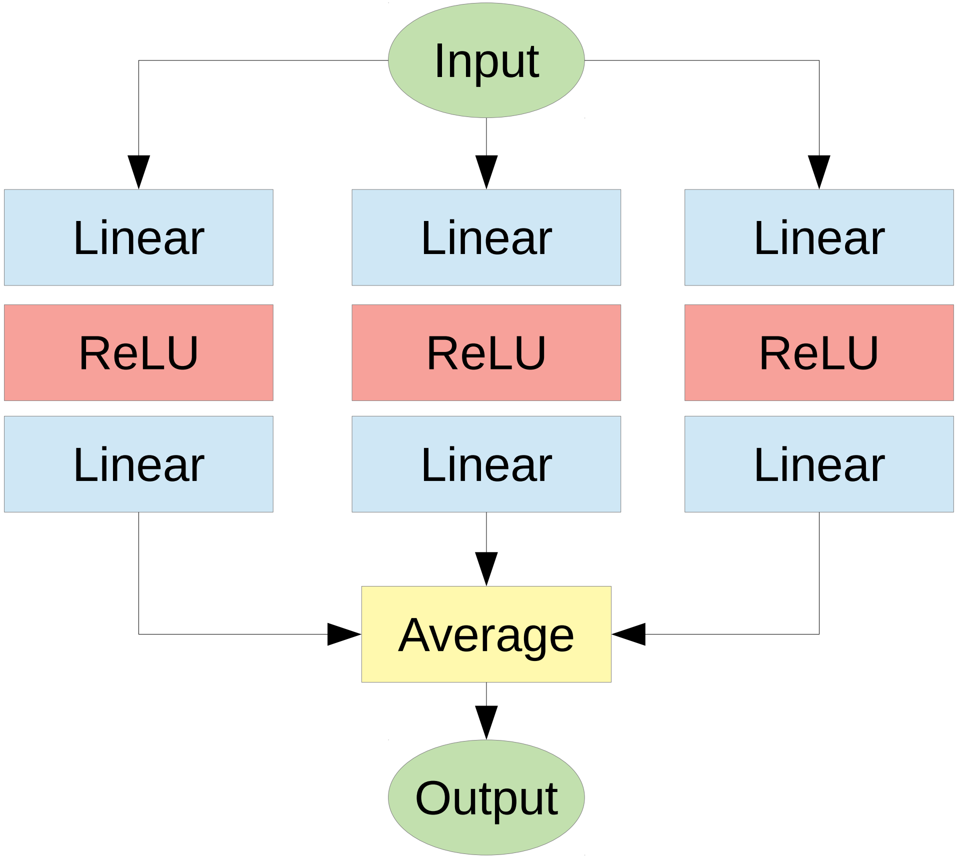


Regularization



In practice: use 1e-4

Ensembles



- Reliable bump in accuracy
- Expensive