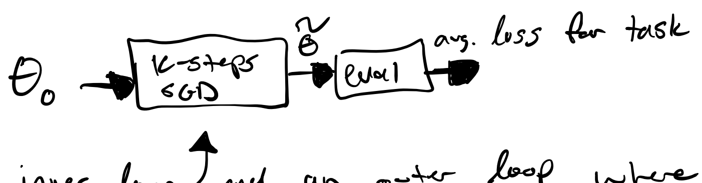
Generative models

Recap MAML: want multitask learning by fine-tuning

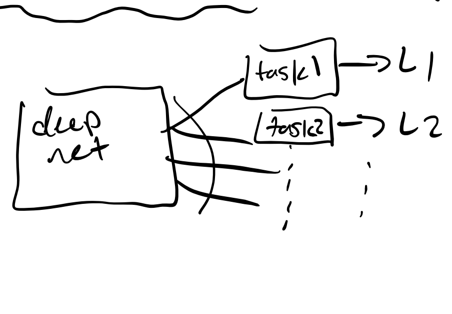
- training data: lots of example tasks, each has labeled data
- optimize w/ SGD
  - A.) Pick minibatch of random of tasks from train data
  - B.) Use current params and batch to compute gradients
  - C.) take $\eta \times$ grad step in negative dir
  - D.) Repeat until stop



- inner loop and an outer loop where we iterate on different task, updating params every time

- As k increases, we get a deeper net, possible exploding gradients
  - memory issues with large k as well
  - we must limit the size of k
- we are learning "good features" for the deep network
  - few-shot fine-tuning will succeed if the dimensions features appropriately capture the tasks we are interested in

- Alternative Baseline: "Just pretrain on union of training tasks"



- Standard supervised learning
- implicitly focuses on conventional view of features
- same as k=0 in MAML

↳ also like setting $\eta_{in} = 0$

↳ people started wondering best choice of η_{in}

↳ what if we made $\eta_{in} < 0$? (gradient ascent)

ANIL approach:

- motivation: inner task head isn't good at initialization so gradients aren't super helpful

- approach:

  A.) "freeze" "feature extractor network"

  ↳ just train linear task head (convex)

  B.) Now differentiate w.r.t. parameters in feature extractor

Reptile: take k-steps of SGD, but update params to be $\eta_{outer}(\theta_k - \theta_0)$