**Training Neural nets/ Things Need to know about Neural Nets**

**Keep Trying to train data until you are satisfied by lowering the learning rate and changing the hypermeters**

**Backprop: What You Need To Know**

* Gradients are important
  + If it's differentiable, we can probably learn on it
* Gradients are important
  + If it's differentiable, we can probably learn on it
* Gradients can vanish
  + Each additional layer can successively reduce signal vs. noise
  + ReLus are useful here
* Gradients can explode
  + Learning rates are important here
  + Batch normalization (useful knob) can help
* ReLu layers can die
  + Keep calm and lower your learning rates

**Normalizing Feature Values**

* We'd like our features to have reasonable scales
  + Roughly zero-centered, [-1, 1] range often works well
  + Helps gradient descent converge; avoid NaN trap
  + Avoiding outlier values can also help
* Can use a few standard methods:
  + Linear scaling
  + Hard cap (clipping) to max, min
  + Log scaling

**Dropout Regularization**

* Dropout: Another form of regularization, useful for NNs
* Works by randomly "dropping out" units in a network for a single gradient step
  + There's a connection to ensemble models here
* The more you drop out, the stronger the regularization
  + 0.0 = no dropout regularization
  + 1.0 = drop everything out! learns nothing
  + Intermediate values more useful