NN: what and why?

Training:

- Stochastic gradient descent (SGD)
- Random initialization
- How to compute gradients

Regularization

- Early stopping
- Dropout

Hidden layer: a bunch of logistic regression classifiers

- parameters: w_j and b_j for each classifier
- equivalently: matrix W (h x d) and vector b (length h)
- Produces “activations” = learned feature vector
- Parameters of model are theta = (W, b, v, c)

Final layer: a linear classifier

- E.g. if logistic regression, has parameter vector v and bias b

Do we need “non-linearity”?

- Without sigmoid, it becomes a linear function of x, not desired
- So we need a non-linear function between two layers
- Options
    - Sigmoid(z) = 1/(1+e^-z)
    - Tanh(z) = (e^2z-1)/(e^2z+1)
    - ReLU(z) = max(z, 0)

Expressiveness of NN

- 2 layer NN can solve XOR (which can’t be solved by linear classifier)

Universal Approximation

- any function can be approximated by a 2 layer neural network with enough hidden units

Multi-layer perceptron

Training objectives

- loss function is the same as that of logistic regression
    - g(x) = w^Tx+b   g(x) = v^T sigma(Wx+b) + c
    - Loss = 1/n (sum -log sigma (y^i g(x^i)))
    - More generally, loss = 1/n (sum L(y^i, g(x^i)))
- SGD
    - Sample a batch of B of examples from the training dataset
    - Do the update on gradient using only the Batch (much faster than normal GD with large dataset)
    - In practice, partition training examples into batches -> use all examples
    - Batch size
        - Large -> more accurate gradient, slower
        - Smaller -> faster, less accurate updates

For NN, initialization is important because it’s non-convex, may stuck in local minimum.

- initialize every entry in W to a small random number
    - depends on “fan-in”, “fan-out”

    - Xavier initialization
    - He initialization
    - Pytorch

Regularization

- Weight decay AKA L2 Regularization
- Prevent overfitting by stopping training before overfit too much
    - save checkpoints, if dev set starts to increase continuously, stop training
- Dropout
    - randomly drop out some neurons by seting their values to 0