CS 316: Introduction to Deep Learning

Regularization Techniques

Week 6

Dr Abdul Samad

Outline

- Weight Decay
  - Regularized Loss
  - l_2 norm vs l_1 norm
- Early Stopping
- Dropout

Weight Decay

- Weight decay is the one of most widely used technique for regularizing parametric machine learning models.
- Instead of directly manipulating the number of parameters, weight decay, operates by restricting the values that the parameters can take.
- The technique is motivated by the basic intuition that among all functions f, the function f = 0 (assigning the value 0 to all the inputs) is in some sense the simplest, and that we can measure the complexity of a function by the distance of its parameters from zero.
- One simple interpretation might be to measure the complexity of a linear function $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$ by some norm of its weight vector e.g. $||\mathbf{w}||^2$

Regularized Loss

- The most common method for ensuring a small weight vector is to add its norm as a penalty term to the problem of minimizing the loss.
- Thus, our new objective is minimizing the sum of prediction loss and penalty term.
- Our loss was given by

$$L(\mathbf{w}, b) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{2}\left(\mathbf{w}^\top \mathbf{x}^{(i)} + b - y^{(i)}\right)^2.$$

  where $\mathbf{x}^{(i)}$ are the features, $y^{(i)}$ is the label for any data example i, and $(\mathbf{w}, b)$ are the weight and bias parameters.
- To penalize the size of the weight vector, we must somehow add $||\mathbf{w}||^2$ to the loss function while also allowing the model to trade off the standard loss for this new penalty.
- In practice, we characterize this tradeoff via the regularization constant λ, a non-negative hyperparameter that we fit using validation data:

$$L(\mathbf{w}, b) + \frac{\lambda}{2}\|\mathbf{w}\|^2.$$

Updating $\mathbf{w}$

- The minibatch stochastic gradient descent update for l_2 regularized regression is as follows:

$$\mathbf{w} = (1 - \eta\lambda)\,\mathbf{w} - \frac{\eta}{|\mathcal{B}|}\sum_{i\in\mathcal{B}}\mathbf{x}^{(i)}\left(\mathbf{w}^\top\mathbf{x}^{(i)} + b - y^{(i)}\right).$$

- As before, we update $\mathbf{w}$ based on the amount by which our estimate differs from our observation. However, we also shrink the size of $\mathbf{w}$ towards zero.
- Given the penalty term alone, our optimization algorithm decays the weight at each step of training
- In contrast to feature selection, weight decay offers us a continuous mechanism for adjusting the complexity of a function.
- Smaller values of λ correspond to less constrained $\mathbf{w}$, whereas larger values of λ constrain $\mathbf{w}$ more considerably.

l_2 norm vs l_1 norm

- One reason to work with l_2 norm is that it places an outsize penalty on large components of the weight vector.
- This biases our learning algorithm towards models that distributes weights evenly across a large number of features.
- By contrast, l_1 penalties lead to models that concentrates weights on a small set of features by clearing other weights to zero.
- This gives us an effective method for feature selection.
- For example,

$$\mathbf{x} = [1, 1, 1, 1]^\top$$
$$\mathbf{w_1} = [1, 0, 0, 0]^\top$$
$$\mathbf{w_2} = [0.25, 0.25, 0.25, 0.25]^\top$$
$$\mathbf{w_1}^\top x = \mathbf{w_2}^\top x = 1$$

Early stopping

- Early stopping is a classical technique for regularizing deep neural networks.
- Rather than directly constraining the values of the weights, one constrains the number of epochs of training.
- The most common way to determine the stopping criteria is to monitor validation error throughout the training and to cut off training when the validation error has not decreased by more than some small amount of ϵ for some number of epochs.
- Besides the potential to lead to better generalization, in the setting of noisy labels, another benefit of early stopping is the time saved.

Dropout

- Classical generalization theory states that to close the gap between train and test performance, we should for a simple model.
- One useful notion of simplicity is smoothness i.e., the function should not be sensitive to small changes to its input. For instance, when we classify images, we should except that adding some random noise to the pixels should be mostly harmless.
- Dropout involves injecting noise while computing each internal layer during forward propagation.
- The method is called dropout because we literally drop out some neurons during training. Throughout training, on each iteration, standard dropout consists of zeroing out some fraction of the nodes in each layer before calculating the subsequent layer.

Dropout

- The key challenge is how to inject this noise.
- One idea is to inject the noise in an unbiased manner such that the expected valuer of each layer – while fixing the others – equals to the value it would have taken absent the noise.
- At each iteration, noise sampled from a distribution with mean zero $\epsilon \sim \mathcal{N}(0, \sigma^2)$ is added to the input $\mathbf{x}$.

$$\mathbf{x}' = \mathbf{x} + \epsilon$$

$$E[\mathbf{x}'] = E[\mathbf{x}]$$

- In standard dropout regularization, one zeros out some fraction of the nodes in each layer and then debiases each layer by normalizing by the fraction of nodes that were retained.
- In other words, with dropout probability p, each intermediate activation h is replaced by a random variable h'

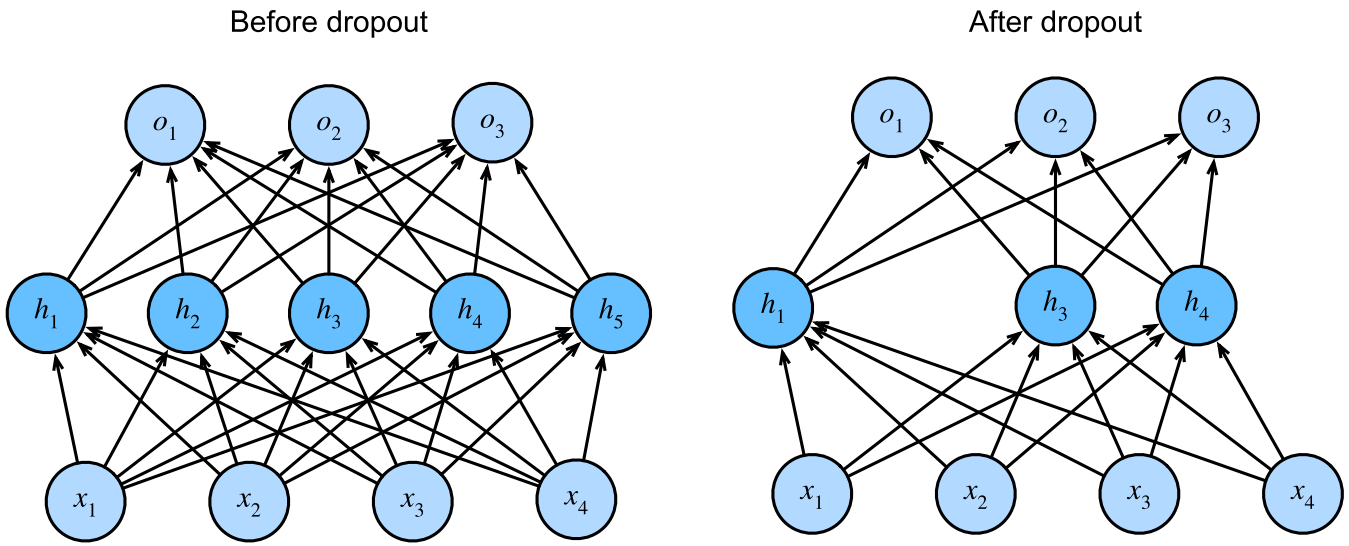
$$h' = \begin{cases} 0 & \text{with probability } p \\ \frac{h}{1-p} & \text{otherwise} \end{cases}$$

$$E[h'] = E[h]$$

Dropout in MLP

- When we apply drop out to a hidden layer, zeroing out each hidden unit with the probability p , the result can be viewed as a network containing only a subset of the original neurons.
- Consider a MLP with a single hidden layer and 5 hidden units. After dropout, h_2 and h_5 are removed.
- Consequently, the calculation of the outputs no longer depends on h_2 or h_5 and their respective gradient also vanishes when performing backpropagation.
- In this way, the calculation of the output layer cannot be overly dependent on any one element of $h_1, \ldots, h_5$

Dropout in MLP



Source