cs224n

(arthurg)

May 21, 2020

Question 1.

As you can see from the equation, m is $0.9 * m_{prev}$ and $0.1 * gradient$. This means that the amount it varries is mostly dependant on value of m_{prev}.

This is useful because there will probably be lots of noise in our minibatches,, so this approach will on average be less susceptible to noise

Question 2.

Overall, $v < 0, \sqrt(v) < 0$ for terms where the gradient is < 1. Thus, for those terms, there will be a larger update. Conversely, if the gradient > 1, then $v > 0, \sqrt(v) > 0$ and those terms will get smaller update.

This means smaller gradients gets slightly larger updates and larger gradients get slightly smaller learning updates. This might be useful as all the terms are updated at a similar pace, meaning they all converge at the same time.

Question 3.

We know that $E[h_{drop}]_i = E[d \odot h]_i = (1 - p_{drop}) * h_i + p_{drop} * 0 = (1 - p_{drop}) * h_i$ Since we want $E[\gamma d \odot h_i] = h_i$, then $h_i = E[\gamma d \odot h_i] = \gamma E[d \odot h_i] = \gamma(1 - p_{drop})h_i$

$$h_i = \gamma(1 - p_{drop})h_i$$
$$1 = \gamma(1 - p_{drop})$$
$$\frac{1}{(1 - p_{drop})} = \gamma$$

Question 4.

We want to apply dropout during training so that some hidden units will learn similar information as other hidden laters.

During evaluation, we want to use all the units to get the best results. Although some hidden units may have learned the same information, having all of them present is kind of similar to taking a "vote" amongst the hidden units and choosing the best results.