c2W3 - hyperparameter tuning

Hyperparameters:-

most important

second most important

least important

α → most important

Adam:- $\beta_1, \beta_2, \varepsilon$ → 0.9, 0.999, 10^{-8}

#layers

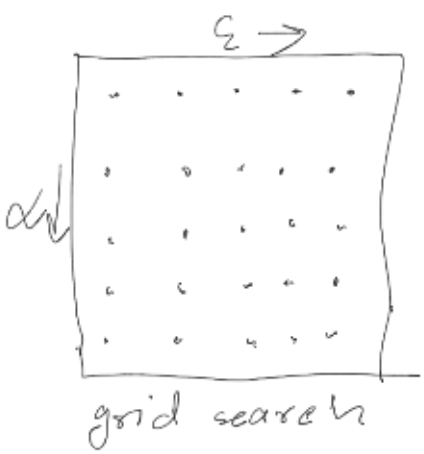
#hidden unit

learning rate decay

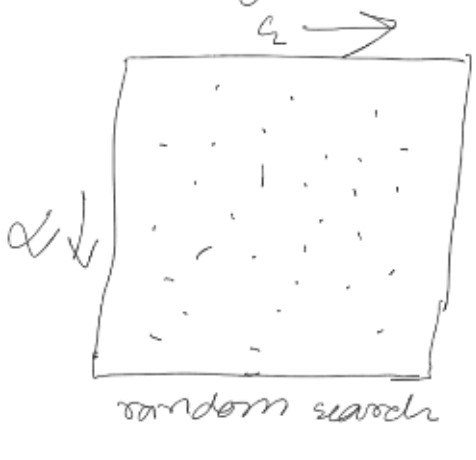
mini-batch size

Hyperparameter:-

Don't use grid search for hyperparameter. instead use some random sample from hyperparameter grid.



grid search



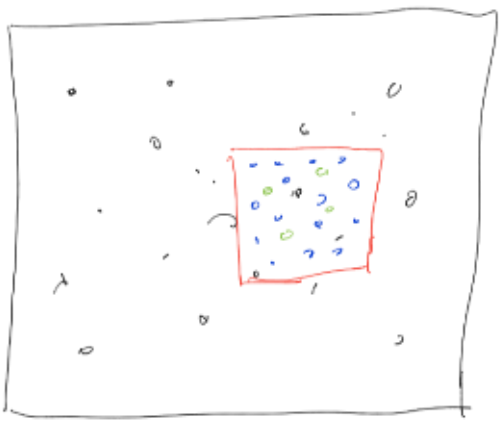
random search.

α will more important than ε. so the efficiency will mostly depend on α. so in grid search above each row will provide same accuracy so we are training 5 model but getting the same result. In the whole search we train 25 model but testing only 5 (α). But if we have choosen those in random then we would have tested 25 different α.

whole point of random search is we don't know which parameter is more important. And we don't want to same parameter(important one) multiple times.

Coarse to fine:-

if we see some hyperparameters close to each other & those works better. then we can zoom in those region and search for more hyperparameter

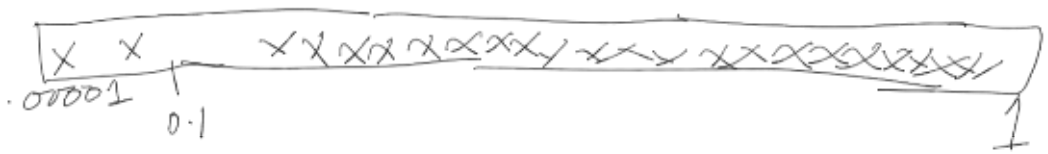


• → works better
• → hyperparameter in zoomed region.
□ → zoomed region.

Picking hyperparameter at random:-

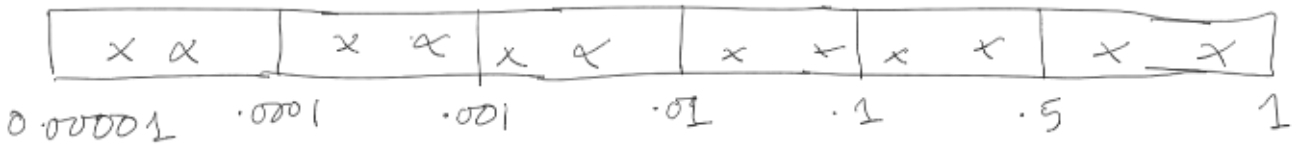
Don't just pick the hyperparameters total random instead make a sub region then pick random parameter.

Let's say we are searching for α which value might be 0.00001 to 1.



we have just two value between 0.00001 to 0.01 but more value from 0.1 to 1. Which is not appropriate as range is more on 0.00001 to 0.1 than 0.1 to 1. so we can deviide that in sub region and take random

value from there.



search for logarithmic scale instead of linear scale.

search for $(1-\beta)$ instead searching for β. because in exponentially weighted sum we take last $\frac{1}{1-\beta}$ average

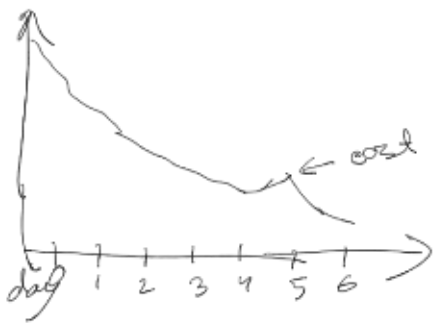
if $\beta = .09 \Rightarrow \frac{1}{1-\beta} = 10$ $\beta = 0.90005 \Rightarrow \frac{1}{1-\beta} = 10$.

$\beta = .999 \Rightarrow \frac{1}{1-\beta} \approx 1000$ $\beta = 0.9995 \Rightarrow \frac{1}{1-\beta} = 2000$

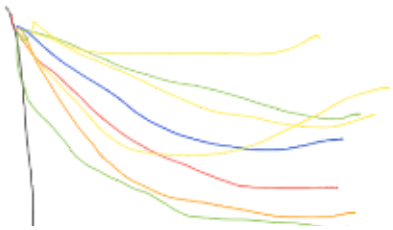
Hyper parameters tuning in practice.

if you don't have enough computing power then you can update the hyperparameters by analyzing previous performance. (baby sitting)

Pandas approach



→ Train many model in parallel



caviar approach