REI602M **Quiz 5.3.2021**

Hand in on Gradescope before 22:00 on Feb. 27 (Saturday). Each question will be given 1, 0.5 or 0 points as follows. If the question is more or less correct it gets 1 point. If it is partly correct it gets 0.5, and if it is missing or completely wrong it gets 0 points.

1) Linear activations (ísl. virkjunarföll) , $f(x)=w^{\top}x$, are not used in hidden nodes in neural networks. Why?

Answer:

Because the data can usally not be display'd in a linear way. When a neural network is more than one layer, we cannot use linear activation, even if the data can can be display'd in a linear transformation, real world data is usally not linear.

2)

a) Initializing all parameters in a neural network to zero prior to training is not recommended. Why? [you may want to refresh material from last weeks lecture]

Answer:

if all weights are zero, then every neuron in every hidden layer will get zero independence. Get stuck in local minimum.

b) Initializing all parameters in a neural network to constant value is not recommended. Why? [you may want to refresh material from last weeks lecture]

Answer:

If every single parameter is the same, then there will be no change in the gradient and and the model continues doing the same thing over and over again. This also applies to a, where all weights are 0.

3)

a) Why is the mini-match size (n_b) usually not set equal to one in mini-batch stochastic gradient descent?

Answer:

number of samples that will be spread through the network are define by the size of the mini-batch. If a mini-batch size is 50, then the algorithm looks at the first 50 samples of the training data and trains the network. Then next 50, and so on. If the size of the batch is 1, we cannot compare 1 sample to anything.

b) Why is the mini-match size (n_b) usually not set equal to *n* in mini-batch stochastic gradient descent?

Answer

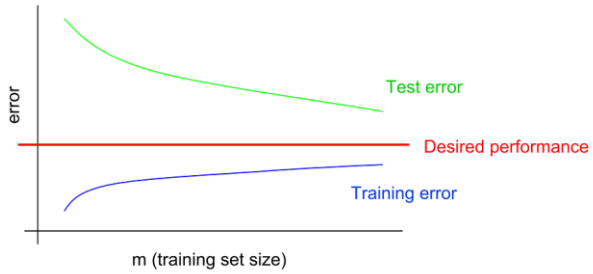
If n is the number of samples in the training data, then we will split the data to one batch witch takes away the meaning of using mini-batch. Explanation in a on how mini-batch works.

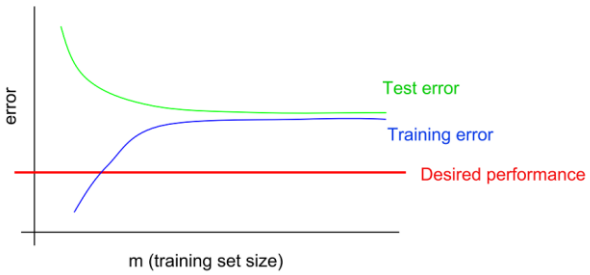
4) A neural network is trained on a dataset and the performance is not as we had been hoping for. The results of the training are summarized in the left figure below. What do you think has gone wrong? What can be done to remedy the situation?

Answer:

There is one thing missing on these graphs, that's the validation error, Validation set is used to minimize overfitting witch is the most likely reason for the test error to be so much larger than the training error.

b) Repeat a) for the right figure below.





like I explained in 2b, the the network could be stuck in local minimum, where there is no change of the weights in the backpropagation.