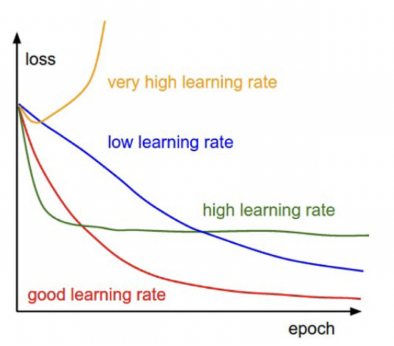
CS 5242
Prof. You's Neural Networks and Deep Learning
August 30, 2022

Homework 3

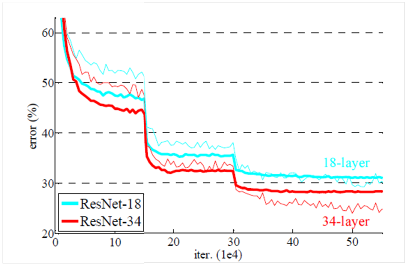
DEADLINE: 12 Sept 2022

This Homework have 5 questions, each is worth 2 points. There are 2 point Bonus in Question 5. You can write the answer in LaTeX, word, or handwriting (take a photo), and submit it to the system.

Question 1. 1) In the plot of loss vs. epoch number (as shown on the left), why does the loss increase for a very high learning rate (yellow curve)?



2) Why the schedule of learning rate as in the figure below for some training and what are the advantages of such a schedule.

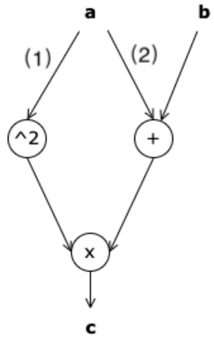


Question 2. You are presented with the following four activation functions:

1) $f(x) = max(x, 0) + min(x, 0) * 0.1$
2) $f(x) = ln(e^{3x} + 1)$
3) $f(x) = ln(e^{3x+1})$

Which one is not suitable as an activation function? Which one is prone to gradient vanishing?

Question 3. You are presented with the computational graph on the below. Suppose that a = -1 and b = 4. 1) calculate the gradient dc/da 2) What is the gradient component of dc/da at location (1) and (2)?



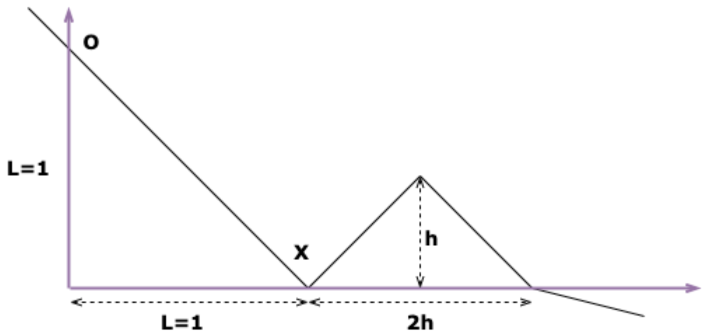
Question 4. You have a binary classification problem with input $x \in R^{100\times 1}$. You consider designing a multi-layer perceptron (MLP) with two hidden layers and one output layer. Each hidden layer perceptron unit has no bias and uses a ReLU activation function. The output layer perceptron uses logistic regression, again with no bias.

1) MLP A has 100 units in the first hidden layer, 20 units in the second hidden layer.

2) MLP B has 20 units in the first hidden layer, 100 units in the second hidden layer.

How many parameters does each MLP have?

Question 5. The diagram below shows a plot of a function f and gradient descend is applied to minimise the function at the point O. there is a bump a distance L away with bump dimensions given as $h \times 2h$. Let $L = 1$, $a = 0.3$ and $h > a$ where a is the learning rate



1) What is the lowest value f could reach in 1000 steps of standard gradient descend? Please show your explanation.

2) (**Bonus 2 points**) If you apply Adam optimizer with parameters given in the following figure, what is the max height h of the bump in which the Adam optimizer will escape the local min at x? use $\epsilon = 0$ instead of $\epsilon = 1e-8$ in your calculations. (You can also write code to calculate the answer. If so please attach your code when submit.)

Algorithm 1: *Adam*, our proposed algorithm for stochastic optimization. See section 2 for details, and for a slightly more efficient (but less clear) order of computation. g_t^2 indicates the elementwise square $g_t \odot g_t$. Good default settings for the tested machine learning problems are $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. All operations on vectors are element-wise. With β_1^t and β_2^t we denote β_1 and β_2 to the power t.

Require: α: Stepsize
Require: $\beta_1, \beta_2 \in [0, 1)$: Exponential decay rates for the moment estimates
Require: $f(\theta)$: Stochastic objective function with parameters θ
Require: θ_0: Initial parameter vector
  $m_0 \leftarrow 0$ (Initialize 1st moment vector)
  $v_0 \leftarrow 0$ (Initialize 2nd moment vector)
  $t \leftarrow 0$ (Initialize timestep)
  while θ_t not converged **do**
    $t \leftarrow t + 1$
    $g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$ (Get gradients w.r.t. stochastic objective at timestep t)
    $m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Update biased first moment estimate)
    $v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ (Update biased second raw moment estimate)
    $\widehat{m}_t \leftarrow m_t / (1 - \beta_1^t)$ (Compute bias-corrected first moment estimate)
    $\widehat{v}_t \leftarrow v_t / (1 - \beta_2^t)$ (Compute bias-corrected second raw moment estimate)
    $\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \widehat{m}_t / (\sqrt{\widehat{v}_t} + \epsilon)$ (Update parameters)
  end while
  return θ_t (Resulting parameters)