

- 1) Describe what problems you encountered and how did you solve them when implementing the basic and advanced functions. (2%)
  - a) Basic functions:
    - i) Not really understand Glorot uniform initialization.
      - (1) After checking the resources from internet, I can complete the function.
    - ii) When using the parameter  $W$  and  $b$  in Dense layer, I didn't realize I have to use `self.parameters["W"]` and `self.parameters["b"]`, instead I used  $W$  and  $b$  directly.
      - (1) After checking the template code again, I found that `self.parameters = {"W":  $W$ , "b":  $b$ }`. So, I changed it.
    - iii) I had face overflow encountered in `exp` when doing the softmax forward.
      - (1) After checking the resources from internet, I changed the code to:  
$$A = \frac{\exp(Z - \max(Z))}{\sum(\exp(Z - \max(Z)), \text{axis}=0)}$$
    - iv) Unclear about the Model class especially when implementing the initialize parameter using function call, I do it:  
$$A = \text{self.forward}(\text{self.linear}[i]) \text{ and } \text{self.forward}(\text{self.activation}[i])$$
  
Correct way:  $A = \text{self.linear}[i].\text{forward}(A) \text{ and } \text{self.activation}[i].\text{forward}(A)$
  - b) Advance functions:
    - i) Got the output of nan
      - (1) Correct the dimension to [10, 60000]
      - (2) Do one-hot encoding to the `y_train`
      - (3) Set the `layer_dims = [784, 10]` (After that changed to [748,50,10])
      - (4) Correct the variable name when splitting the training and validation data
- 2) Briefly describe the structure of your binary and multi-class classifiers. (2%)
  - a) Binary classifiers: one layer network with `layer_dims = [30,1]` and `activation_fn = ["sigmoid"]`
  - b) Multi-class classifiers: 2 layer network with `layer_dims = [784, 50, 10]` and `activation_fn = ["relu", "softmax"]`
- 3) Describe effort you put to improve your model (e.g., hyperparameter finetuning). (1%)
  - a) For binary classification, I tried learning rate = 0.001 and number of iteration = 4000, and I found that the accuracy is not high, only 0.78. So I changed to some other learning rate, and found that it is all not very high, for example 0.0017, 0.003, 0.99 (accuracy = 0.95, but the learning rate is too high). At last, I changed the learning rate to 0.1 and number of iteration = 8500, and I am able to get the accuracy = 0.96.
  - b) For multi-class classification, I tried learning rate = 0.99 and number of iteration = 4000, and I get an accuracy = 0.94 something, but the learning rate is too high, so I changed it to 0.0001 and number of iteration = 3000, but the accuracy is low. Then I tried for learning rate = 0.05 and number of iteration = 20000, but the runtime reached max duration. So, I changed the learning rate to 0.001 and number of iteration = 3000, and get accuracies of 0.9923 for `pred_train` and 0.9702 for `pred_val`.