**Quiz 1**

* **Due** Oct 16, 2020 at 11:59

* **Points** 5

* **Questions** 12

* **Available** Oct 15, 2020 at 12:00 - Oct 16, 2020 at 11:59 about 24 hours

* **Time Limit** 60 Minutes

**Instructions**

Total Points: 5

Duration: 60 Minutes

Read the questions and follow instructions **very carefully**.  Questions can carry different marks. There are 12 questions, including 4 numericals.

This quiz was locked Oct 16, 2020 at 11:59.

Attempt History

|  | **Attempt** | **Time** | **Score** |
| --- | --- | --- | --- |
| **LATEST** | [Attempt 1 Links to an external site.](https://bits-pilani.instructure.com/courses/580/quizzes/1066/history?version=1) | 60 minutes | 3.75 out of 5 |

Score for this quiz: **3.75** out of 5

Submitted Oct 16, 2020 at 10:52

This attempt took 60 minutes.

**Question 1**

**0 / 0.25 pts**

During training a deep network, initially a high learning rate helps in avoiding local minima and at later stage, a low learning rate helps in moving to a local minima.

**Correct AnswerCorrect Answer**



True

**You AnsweredYou Answered**



False

**Question 2**

**0.25 / 0.25 pts**

Which of the statements about Adam is False?

**Correct!Correct!**



Adam should be used with batch gradient computations, not with min-batches.



We usually use “default” values for the hyperparameters β1, β2 and ε in Adam (β1= 0.9, β2= 0.999, ε = 10-8).



Adam combines the advantages of RMSprop and momentum.



The learning rate hyperparameter α in Adam usually needs to be tuned.

**Question 3**

**0.25 / 0.25 pts**

Which of the following(s) is(are) correct regarding Nesterov Accelerated Gradient Descent (NAG)?

**Correct!Correct!**



It evaluates the gradient after adding momentum



It evaluates the gradient independent of momentum



None of the other answers



It evaluates the gradient before adding momentum

**Question 4**

**0.25 / 0.25 pts**

Which of these statements about mini-batch gradient descent is true?



None of the given statements.



You should implement mini-batch gradient descent without an explicit for-loop over different mini-batches, so that the algorithm processes all mini-batches at the same time (vectorization).



Training one epoch (one pass through the training set) using mini-batch gradient descent is faster than training one epoch using batch gradient descent.

**Correct!Correct!**



One iteration of mini-batch gradient descent (computing on a single mini-batch) is faster than one iteration of batch gradient descent.

**Question 5**

**0.25 / 0.25 pts**

Suppose k is the mini batch size and N is the number of data points. How many steps will be there in one epoch of mini-batch gradient descent?



k\*N



k



N

**Correct!Correct!**



N/k

**Question 6**

**0.25 / 0.25 pts**

[0.5] Which of these is NOT a good learning decay rate scheme? Here, t is the epoch number.

1. α=(1/(1+2\*t))α0
2. α=0.95t α0
3. α=(1/√t )α0
4. α=etα0



α=α0/(1+2\*t)



α=α0/√t



α=0.95t α0

**Correct!Correct!**



α=et α0

**Question 7**

**0.25 / 0.25 pts**

Convergence rate of batch gradient descent is lower than that of stochastic gradient descent



True

**Correct!Correct!**



False

**Question 8**

**0.25 / 0.25 pts**

The learning rate hyperparameter α in Adam usually needs to be tuned.

**Correct!Correct!**



True



False

**Question 9**

**0 / 0.5 pts**

Error surface is given by E(w) = 2 w2+ 5w + 4 . Assume gradient descent is used to find the minimum of this error surface. What is the maximum value of the learning rate that allows convergence? **Enter a real number (a.xy) only with 2 digits after decimal.**

**You AnsweredYou Answered**



**Correct AnswersCorrect Answers**

0.5 (with precision: 2)

**Question 10**

**0 / 0.5 pts**

Error surface is given by E(x,y) = 2x2+3y2+6.  What is the smallest value of the learning rate that leads to divergence? **Enter a real number (a.xyz) only with 3 digits after decimal without rounding.**

**You AnsweredYou Answered**



**Correct AnswersCorrect Answers**

0.333 (with precision: 3)

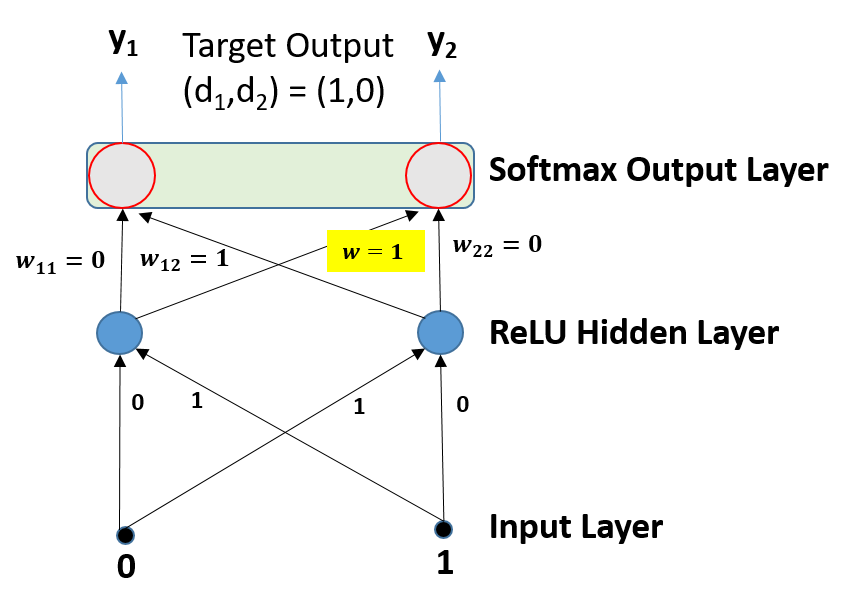
**Question 11**

**0.5 / 0.5 pts**

Refer to the Figure below.  Hidden nodes use ReLU activation function. Output nodes are softmax. Bias at all nodes is **0**.

Calculate the absolute value of the cross-entropy loss for the given input data, bias and weights.

**Write the answers as  REAL numbers only with 3 digits after decimal  (e.g., a.xyz) without any rounding.**



**Correct!Correct!**



**Correct AnswersCorrect Answers**

1.31 (with precision: 3)

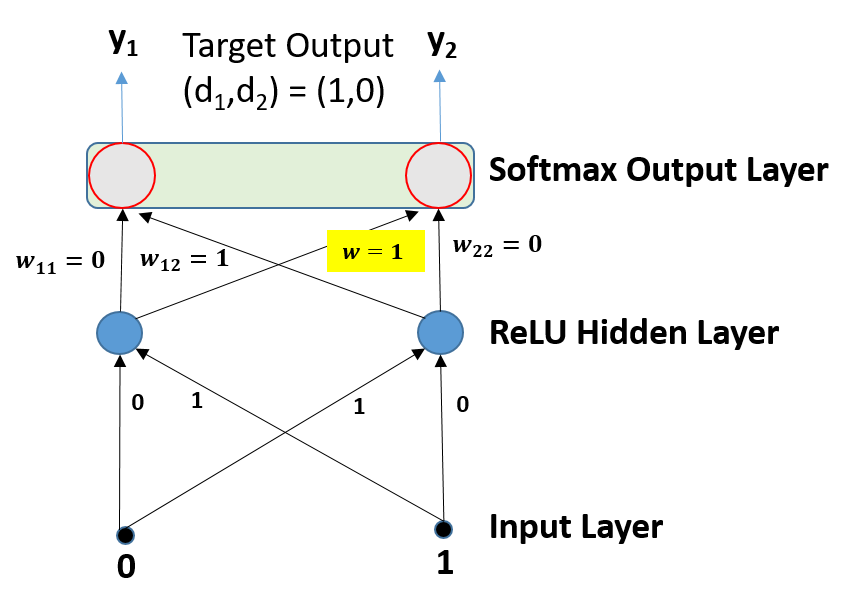
**Question 12**

**1.5 / 1.5 pts**

Refer to the Figure below.  Hidden nodes use ReLU activation function. Output nodes are softmax. Backpropagation is used to update the weights. Learning rate is **0.1**. Bias at all nodes is **0**.

Calculate the absolute value of **change** in weight **w** (marked in yellow) for the given input data, weights and target outputs (d1,d2). (y1, y2) are actual outputs from the network.

**Write the answers as  REAL numbers only with 3 digits after decimal  (e.g., a.xyz) without any rounding.**



**Correct!Correct!**



**Correct AnswersCorrect Answers**

0.073 (with precision: 3)