

Practical aspects of deep learning

10/10 points (100%)

Quiz, 10 questions

Congratulations! You passed!

1. If you have 10,000,000 examples, how would you split the train/dev/test set?

1 / 1
points

☒ 98% train . 1% dev . 1% test

Correct

☐ 60% train . 20% dev . 20% test

☐ 33% train . 33% dev . 33% test

2. The dev and test set should:

1 / 1
points

☒ Come from the same distribution

Correct

☐ Come from different distributions

☐ Be identical to each other (same (x,y) pairs)

☐ **Have the same number of examples**

3. If your Neural Network model seems to have high bias, what of the following would be promising things to try? (Check all that apply.)

1 / 1
points

☐ Get more test data

Un-selected is correct

☒ Increase the number of units in each hidden layer

Correct

☐ Add regularization

Un-selected is correct

☒ Make the Neural Network deeper

Correct

☐ Get more training data

Un-selected is correct

4. You are working on an automated check-out kiosk for a supermarket, and are building a classifier for apples, bananas and oranges. Suppose your classifier obtains a training set error of 0.5%, and a dev set error of 7%. Which of the following are promising things to try to improve your classifier? (Check all that apply.)

1 / 1
points

☒ Increase the regularization parameter lambda

Correct

☐ Decrease the regularization parameter lambda

Un-selected is correct

☒ Get more training data

Correct

☐ Use a bigger neural network

Un-selected is correct

5. What is weight decay?

1 / 1
points

☐ A technique to avoid vanishing gradient by imposing a ceiling on the values of the weights.

☒ A regularization technique (such as L2 regularization) that results in gradient descent shrinking the weights on every iteration.

Correct

☐ Gradual corruption of the weights in the neural network if it is trained on

- ☒ noisy data.
- ☐ The process of gradually decreasing the learning rate during training.
-

6. What happens when you increase the regularization hyperparameter lambda?

1 / 1
points

- ☒ Weights are pushed toward becoming smaller (closer to 0)
- Correct**
- ☐ Weights are pushed toward becoming bigger (further from 0)
- ☐ Doubling lambda should roughly result in doubling the weights
- ☐ Gradient descent taking bigger steps with each iteration (proportional to lambda)
-

7. With the inverted dropout technique, at test time:

1 / 1
points

- ☐ You do not apply dropout (do not randomly eliminate units), but keep the 1/keep_prob factor in the calculations used in training.
- ☐ You apply dropout (randomly eliminating units) but keep the 1/keep_prob factor in the calculations used in training.
- ☐ You apply dropout (randomly eliminating units) and do not keep the 1/keep_prob factor in the calculations used in training
- ☒ You do not apply dropout (do not randomly eliminate units) and do not keep the 1/keep_prob factor in the calculations used in training
- Correct**
-

8. Increasing the parameter keep_prob from (say) 0.5 to 0.6 will likely cause the following:
(Check the two that apply)

1 / 1
points

- ☐ Increasing the regularization effect
- Un-selected is correct**

- ☒ Reducing the regularization effect
- Correct**

- ☐ Causing the neural network to end up with a higher training set error
- Un-selected is correct**

☒ Causing the neural network to end up with a lower training set error

Correct

9. Which of these techniques are useful for reducing variance (reducing overfitting)?
(Check all that apply.)

1 / 1
points

☐ Exploding gradient

Un-selected is correct

☒ Dropout

Correct

☒ Data augmentation

Correct

☒ L2 regularization

Correct

☐ Vanishing gradient

Un-selected is correct

☐ Gradient Checking

Un-selected is correct

☐ Xavier initialization

Un-selected is correct

10. Why do we normalize the inputs x ?

1 / 1
points

☐ It makes the parameter initialization faster

☐ It makes it easier to visualize the data

☐ Normalization is another word for regularization--It helps to reduce variance

☒ It makes the cost function faster to optimize



It makes the cost function faster to optimize

Correct
