

✓ 축하합니다! 통과하셨습니다!

받은 성적 100% 통과 점수: 80% 이상

다음 항목으로 이동

Practical aspects of Deep Learning

최근 제출물 성적 100%

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

1 / 1점

- ☐ 60% train . 20% dev . 20% test
- ☒ 98% train . 1% dev . 1% test
- ☐ 33% train . 33% dev . 33% test

✓ 맞습니다

2. The dev and test set should:

1 / 1점

- ☐ Have the same number of examples
- ☒ Come from the same distribution
- ☐ Come from different distributions
- ☐ Be identical to each other (same (x,y) pairs)

✓ 맞습니다

3. If your Neural Network model seems to have high variance, what of the following would be promising things to try?

1 / 1점

☒ Add regularization

✓ 맞습니다

☐ Get more test data

☒ Get more training data

✓ 맞습니다

☐ Make the Neural Network deeper

☐ Increase the number of units in each hidden layer

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 / 1점

☒ Increase the regularization parameter lambda

✓ 맞습니다

☐ Decrease the regularization parameter lambda

☒ Get more training data

✓ 맞습니다

☐ Use a bigger neural network

5. What is weight decay?

1 / 1점

- ☒ A regularization technique (such as L2 regularization) that results in gradient descent shrinking the weights on every iteration.
- ☐ The process of gradually decreasing the learning rate during training.
- ☐ Gradual corruption of the weights in the neural network if it is trained on noisy data.
- ☐ A technique to avoid vanishing gradient by imposing a ceiling on the values of the weights.

✓ 맞습니다

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

1 / 1점

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

✓ 맞습니다

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

1 / 1점

- ☒ 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
- ☐ 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) and do not 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.

✔ 맞습니다

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

1 / 1점

- ☐ Increasing the regularization effect
- ☒ Reducing the regularization effect

✔ 맞습니다

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

✔ 맞습니다

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

1 / 1점

- ☒ Data augmentation

✔ 맞습니다

- ☒ Dropout

✔ 맞습니다

- ☐ Vanishing gradient
- ☐ Xavier initialization
- ☒ L2 regularization

✔ 맞습니다

- ☐ Exploding gradient
- ☐ Gradient Checking

10. Why do we normalize the inputs x ?

1 / 1점

- ☒ It makes the cost function faster to optimize
- ☐ It makes it easier to visualize the data
- ☐ It makes the parameter initialization faster
- ☐ Normalization is another word for regularization--It helps to reduce variance

✔ 맞습니다