1.Question 1

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

60% train . 20% dev . 20% test

**98% train . 1% dev . 1% test**

33% train . 33% dev . 33% test

2.Question 2

The dev and test set should:

**Come from the same distribution**

Come from different distributions

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

Have the same number of examples

3.Question 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.)

Get more test data

**Increase the number of units in each hidden layer**

Add regularization

**Make the Neural Network deeper**

Get more training data

4.Question 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.)

**Increase the regularization parameter lambda**

Decrease the regularization parameter lambda

**Get more training data**

Use a bigger neural network

5.Question 5

What is weight decay?

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

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.

The process of gradually decreasing the learning rate during training.

6.Question 6

What happens when you increase the regularization hyperparameter lambda?

**Weights are pushed toward becoming smaller (closer to 0)**

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.Question 7

With the inverted dropout technique, at test time:

You apply dropout (randomly eliminating units) but 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 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 apply dropout (randomly eliminating units) and do not keep the 1/keep\_prob factor in the calculations used in training

8.Question 8

Increasing the parameter keep\_prob from (say) 0.5 to 0.6 will likely cause the following: (Check the two that apply)

Increasing the regularization effect

**Reducing the regularization effect**

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**

9.Question 9

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

**Dropout**

Xavier initialization

**Data augmentation**

Vanishing gradient

Exploding gradient

Gradient Checking

**L2 regularization**

10.Question 10

Why do we normalize the inputs xxx?

**It makes the cost function faster to optimize**

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

It makes the parameter initialization faster

It makes it easier to visualize the data