Week 1 Quiz

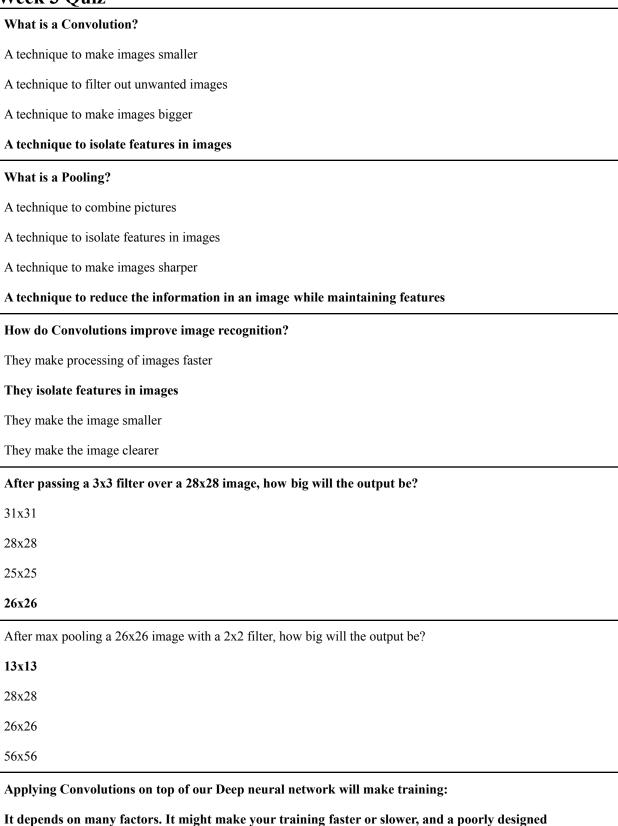
| The diagram for traditional programming had Rules and Data In, but what came out? |
|--|
| Bugs |
| Machine Learning |
| Binary |
| Answers |
| The diagram for Machine Learning had Answers and Data In, but what came out? |
| Bugs |
| Rules |
| Binary |
| Models |
| When I tell a computer what the data represents (i.e. this data is for walking, this data is for running), what is that process called? Learning the Data |
| Labelling the Data |
| Programming the Data |
| Categorizing the Data |
| What is a Dense? |
| A layer of connected neurons |
| Mass over Volume |
| A single neuron |
| A layer of disconnected neurons |
| What does a Loss function do? |
| Generates a guess |
| Measures how good the current 'guess' is |
| Decides to stop training a neural network |
| Figures out if you win or lose |
| What does the optimizer do? |
| Figures out how to efficiently compile your code |

Measures how good the current guess is Decides to stop training a neural network Generates a new and improved guess What is Convergence? The bad guys in the next 'Star Wars' movie A dramatic increase in loss The process of getting very close to the correct answer A programming API for AI What does model.fit do? It trains the neural network to fit one set of values to another It makes a model fit available memory It determines if your activity is good for your body It optimizes an existing model Week 2 Quiz What's the name of the dataset of Fashion images used in this week's code? **Fashion Tensors** Fashion MN Fashion Data **Fashion MNIST** What do the above mentioned Images look like? 28x28 Color 100x100 Color 28x28 Greyscale 82x82 Greyscale How many images are in the Fashion MNIST dataset? 70,000

60,000

| 10,000 |
|---|
| 42 |
| Why are there 10 output neurons? |
| To make it classify 10x faster |
| To make it train 10x faster |
| There are 10 different labels |
| Purely arbitrary |
| What does Relu do? |
| It only returns x if x is greater than zero |
| It returns the negative of x |
| For a value x, it returns 1/x |
| It only returns x if x is less than zero |
| Why do you split data into training and test sets? |
| To make training quicker |
| To train a network with previously unseen data |
| To test a network with previously unseen data |
| To make testing quicker |
| What method gets called when an epoch finishes? |
| on_end |
| On_training_complete |
| on_epoch_end |
| on_epoch_finished |
| What parameter to you set in your fit function to tell it to use callbacks? |
| callback= |
| oncallback= |
| callbacks= |
| oncallbacks= |
| |

Week 3 Quiz



Convolutional layer may even be less efficient than a plain DNN!

Faster

Stay the same

Slower

Week 4 Quiz

Using Image Generator, how do you label images?

TensorFlow figures it out from the contents

It's based on the file name

It's based on the directory the image is contained in

You have to manually do it

What method on the Image Generator is used to normalize the image?

normalize

rescale

Rescale image

normalize image

How did we specify the training size for the images?

The target size parameter on the training generator

The training_size parameter on the training generator

The training_size parameter on the validation generator

The target size parameter on the validation generator

When we specify the input shape to be (300, 300, 3), what does that mean?

There will be 300 images, each size 300, loaded in batches of 3

Every Image will be 300x300 pixels, with 3 bytes to define color

Every Image will be 300x300 pixels, and there should be 3 Convolutional Layers

There will be 300 horses and 300 humans, loaded in batches of 3

If your training data is close to 1.000 accuracy, but your validation data isn't, what's the risk here?

You're overfitting on your training data

You're underfitting on your validation data

You're overfitting on your validation data

No risk, that's a great result

Convolutional Neural Networks are better for classifying images like horses and humans because:

In these images, the features may be in different parts of the frame

There's a wide variety of horses

There's a wide variety of humans

All of the above

After reducing the size of the images, the training results were different. Why?

There was more condensed information in the images

There was less information in the images

We removed some convolutions to handle the smaller images

The training was faster