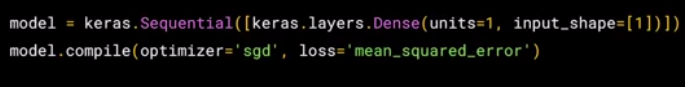
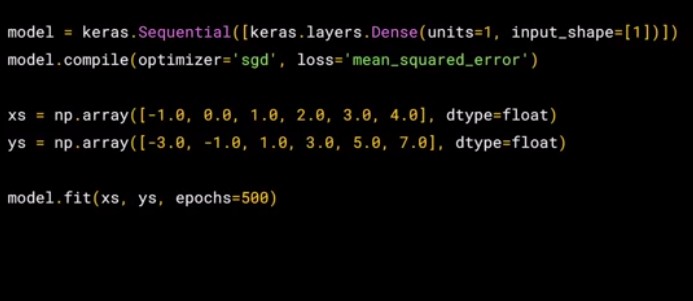
**ML on Tensorflow and GCP**

**DeepLearning**

Simplest FFNN with only one layer, one neuron and one input value.



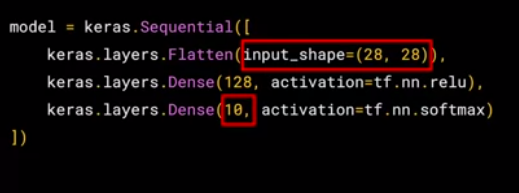
Important: Errors and loss functions, optimizers



Loading dataset from a source (Ex: MNIST)



First CNN:

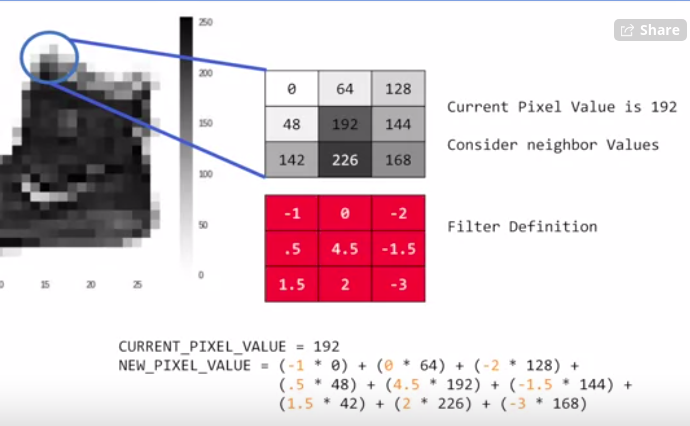


For CNN, it’s important to Normalize Data

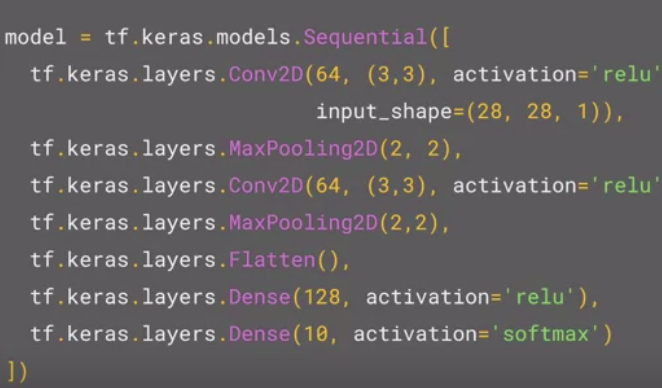
(0-255 color values ) -> values between 0-1

Text-lables → Numeric classes

**Convolutional Pooling**



Next step: keep the biggest value (max pooling) or the smallest value(min pooling) only.



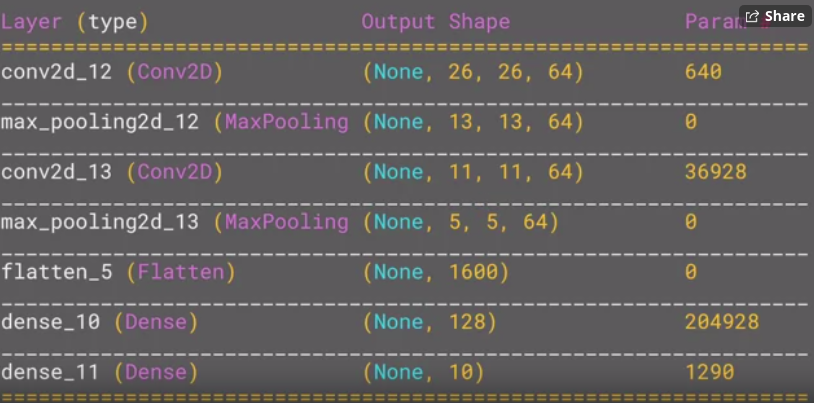
1st layer: Convo2D - 64 (3\*3) layers , input shape should always hv one dimension more than given. (colorIndex)

Grayscale images: 1

Color Images: 3

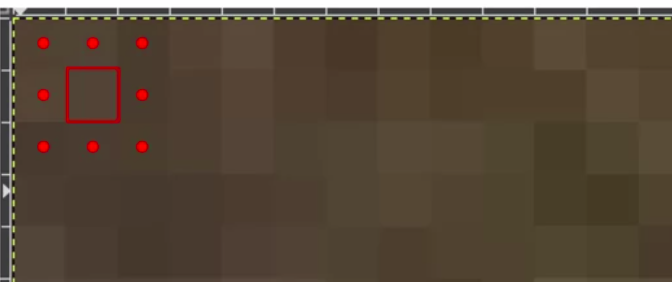
Note: **model.summary()**

This will give an overview of layers and how image pixel sizes hv reduces while pooling

Ex: 

Why 28 → 26 ???

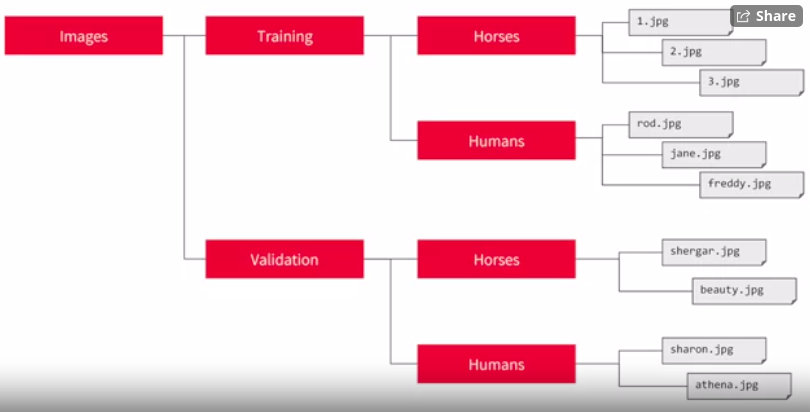
Reason: to fit the kernel on image one pixel of area should be omitted from all around the image (2 pixels per each side)

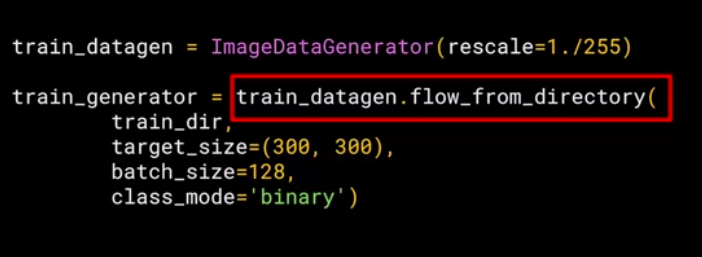


Data Pre-processing in GCP Tensorflow

1. Image Generator of Tensorflow

Create subdirectories and categorize images with labels





Train\_dir : Parent Directory

Class\_mode: classifier mode (Ex: binary, only 2 categories)

Ex: ConvoNet for complex images:

