**GANs**

**Latest Submission Grade 100%**

**1.**

**Question 1**

In GANs, the network learns to improve on creating data by the way of knowledge flowing back from the *discriminator* to the *generator.*

**1 / 1 point**



True

**Correct**

Correct! The feedback sent from the discriminator helps the generator in better generation of the new data.

**2.**

**Question 2**

In the process of training a GAN, the *generator* is trained by getting it to produce a batch of fake images, and also labelling them as real images despite them being fake. *While this happens the evaluation performed by the discriminator helps in updating the parameters for the discriminator.*

**1 / 1 point**



False

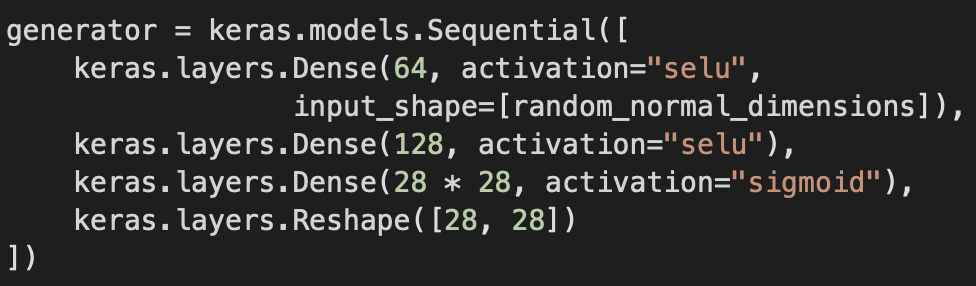
**Correct**

Correct! The parameters of the *discriminator* are frozen during this step.

**3.**

**Question 3**

Consider the following piece of code for a generator, what is the purpose of using the *selu* activation function instead of ReLU?



**1 / 1 point**



ReLU removes the noise within your data, but your intention is to keep it which is why selu is used.

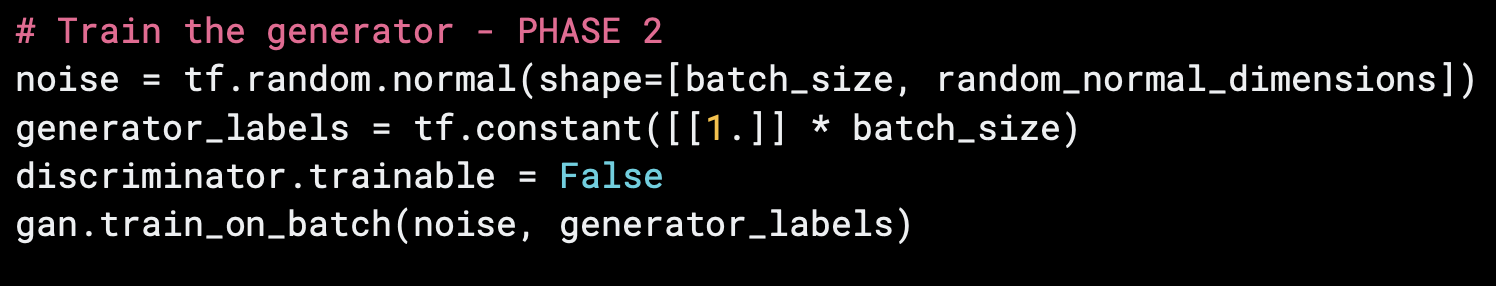
**Correct**

Correct!

**4.**

**Question 4**

Consider the following code for training the generator and check all that are true.



**1 / 1 point**



You set *all* of the generator\_labels=1 and pass in only the fake images in *phase 2* of the training.

**Correct**

Correct! You pass both, only the *fake images, but* set the label of all of them to 1 so you could try to trick the discriminator.



You set the trainable parameters of the discriminator to *false* because updating the discriminator weights will corrupt the training process.

**Correct**

Correct! You set them to false because the discriminator weights will get corrupted because of feeding it fake labels against both, *fake and original* images.

**5.**

**Question 5**

With regards to GANs, what does the term *mode collapse* mean?

**1 / 1 point**



When the model starts to generate more and more of the same data with which it was able to fool the discriminator.

**Correct**

Correct!

**6.**

**Question 6**

Which of the following are some of the *best practices* when building GANs (**DCGans**) which help us avoid the problem of *mode collapse* ? Check all that apply.

**1 / 1 point**



Batch normalization should be used in the generator except in the output layer.

**Correct**

Correct!



Avoid the use of *Dense* layer in both the discriminator and the generator.

**Correct**

Correct!

**7.**

**Question 7**

You can apply a 3x3 stride filter of 1 on a 3x3 image using Conv2DTranspose (Process of deconvolution).

**1 / 1 point**



True

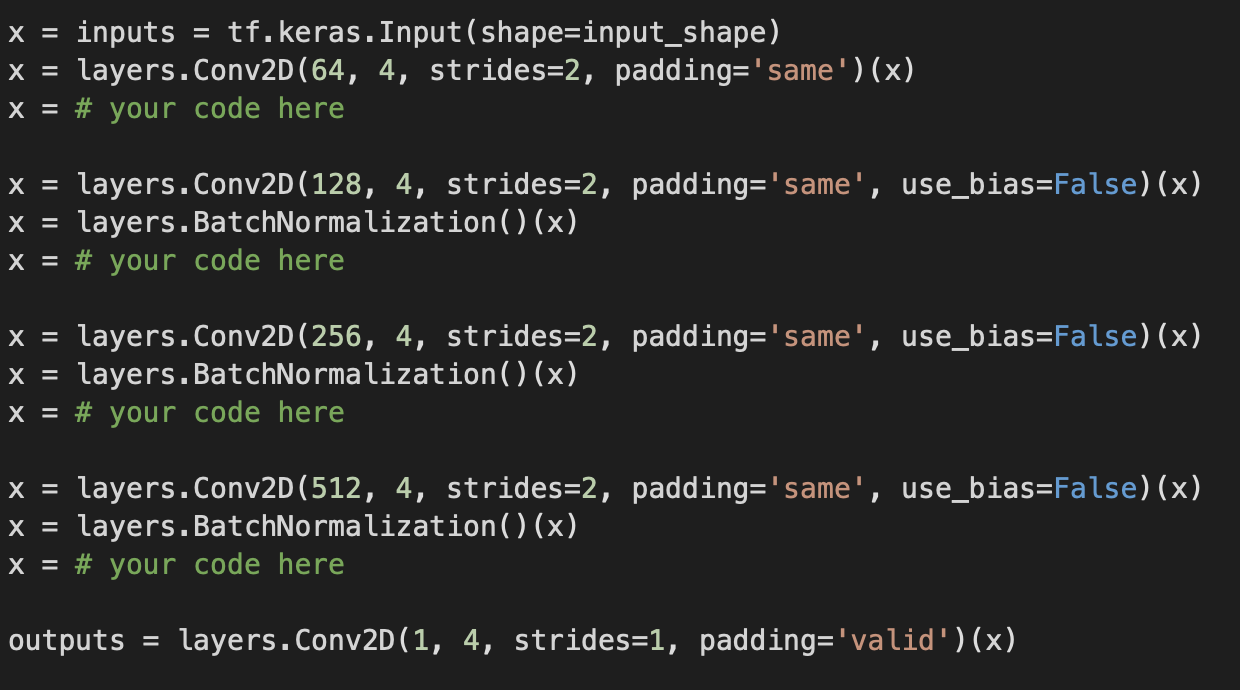
**Correct**

Correct! While it may not sound possible, Conv2DTranspose makes it possible by filling more data in the 3x3 image, making it a 9x9 image.

**8.**

**Question 8**

Following is the code of a *discriminator*. According to *best practices*, which activation function should be used?



**1 / 1 point**



LeakyReLU

**Correct**

Correct! You want to maintain some values when learning, instead of zeroing them out, which is what ReLU does.