**AutoEncoders**

**Latest Submission Grade 100%**

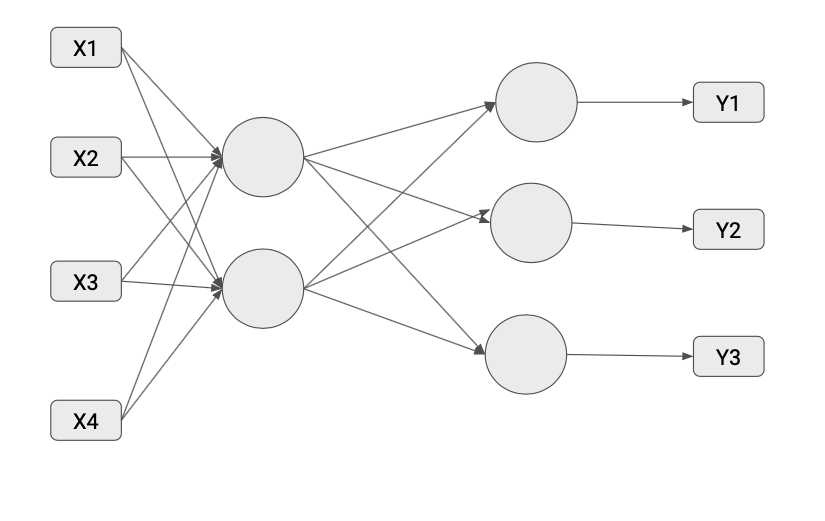
**1.**

**Question 1**

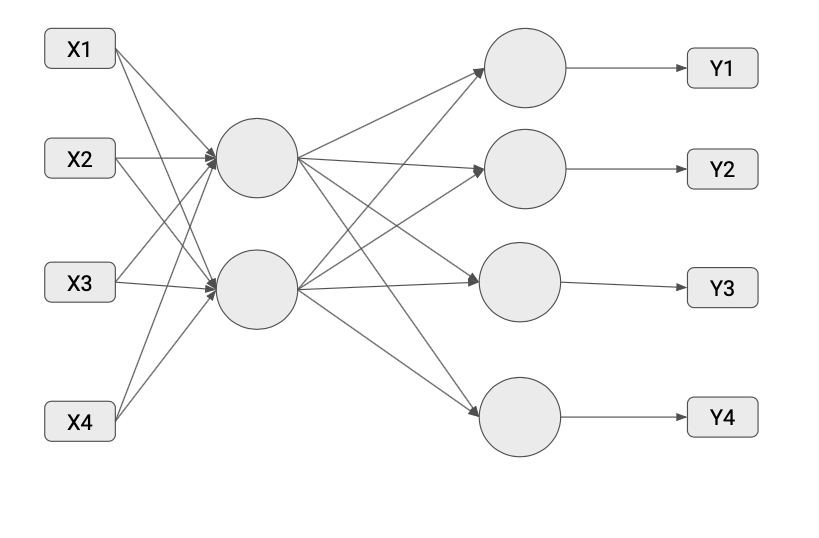
Which of the following is a valid architecture for an AutoEncoder? Check all that apply.

**1 / 1 point**





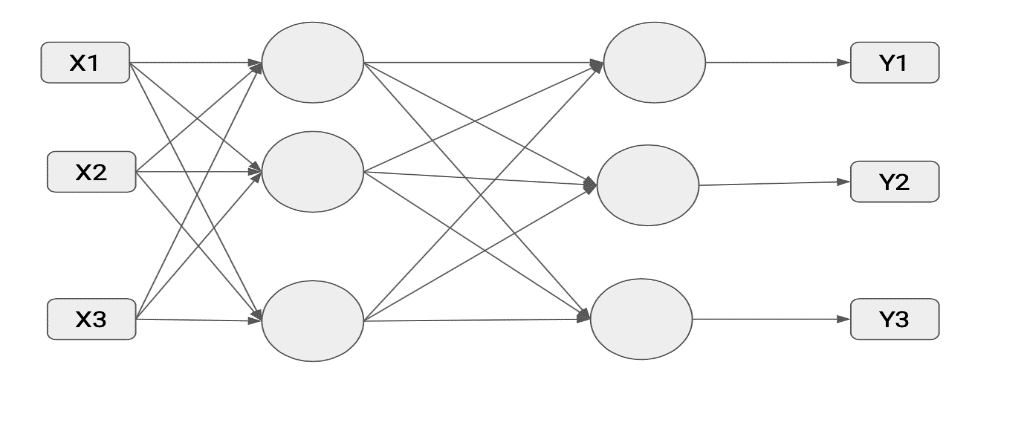




**Correct**

Correct!

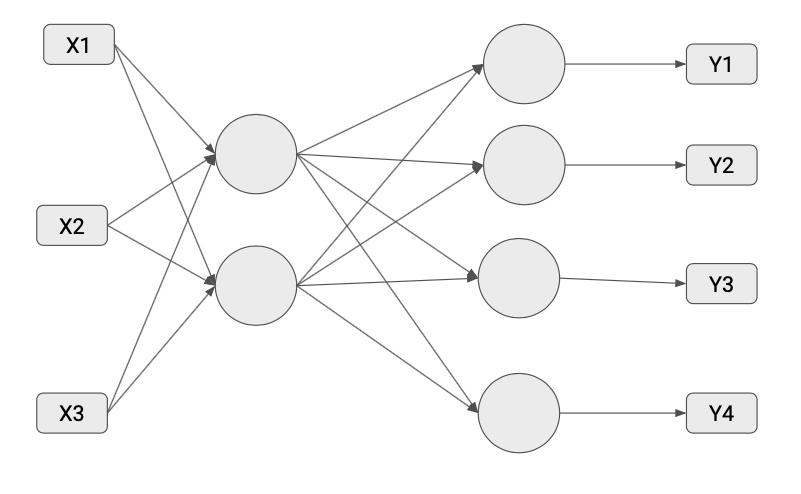




**Correct**

Correct! While the encoder layer and decoder layer have the same number of units which might take a straight pass through the layers, resulting in poor learning of latent representation, the architecture is still valid.

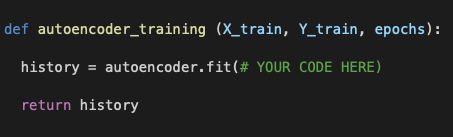




**2.**

**Question 2**

After initializing your AutoEncoder you are all set to train it. Which of the following pieces of code will you use ?



**1 / 1 point**



autoencoder.fit(X\_train, X\_train, epochs=epochs)

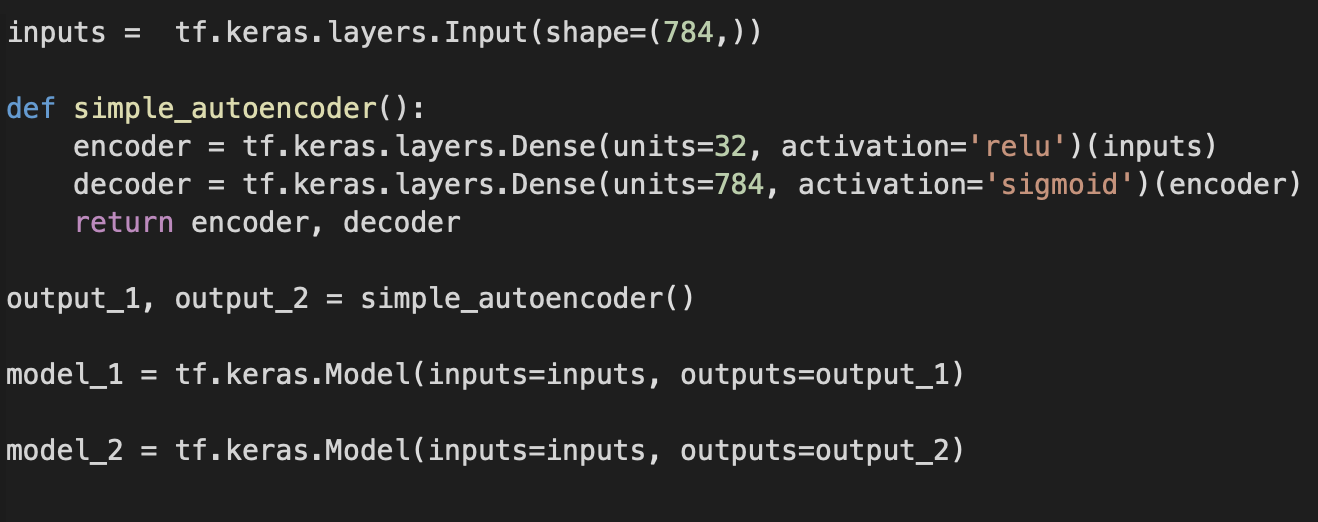
**Correct**

Correct! For data reconstruction purposes you fit input data values to input data values (as opposed to fitting them to output data values), this way the model learns best to replicate the data.

**3.**

**Question 3**

Consider the following code for a simple *AutoEncoder*, what is *model\_1* outputting ?



**1 / 1 point**



Displaying the internal representation of the input the model is learning to replicate.

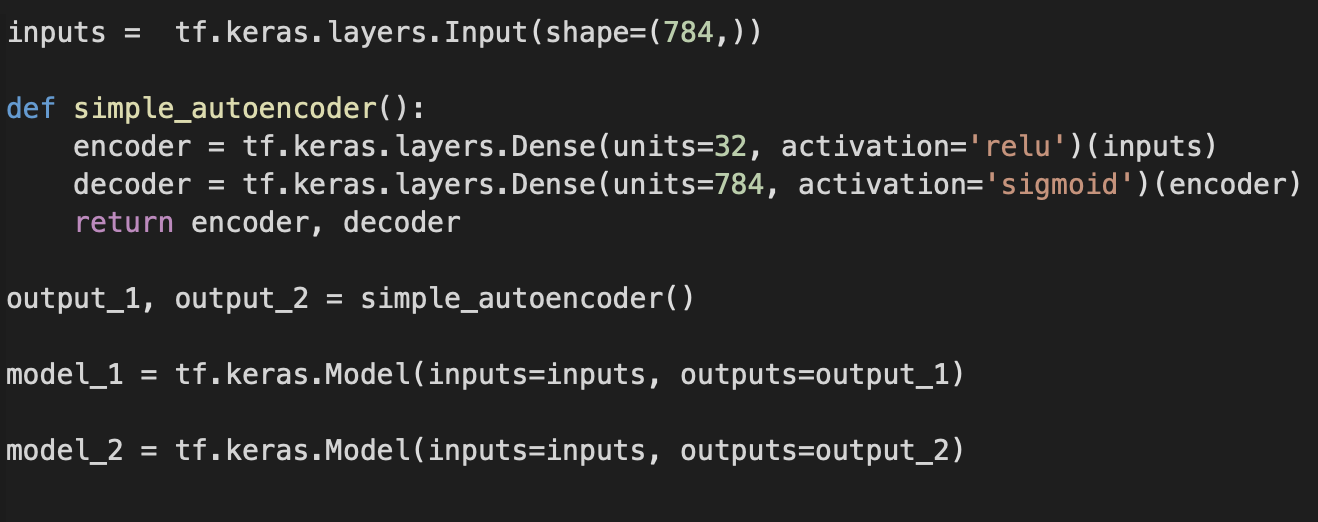
**Correct**

Correct! *model\_1* is returning the encoded representation of your input values, which are being fed to the decoder as input.

**4.**

**Question 4**

Consider the following code for a simple *AutoEncoder*, which of these is *model\_1*’s output ?



**1 / 1 point**

















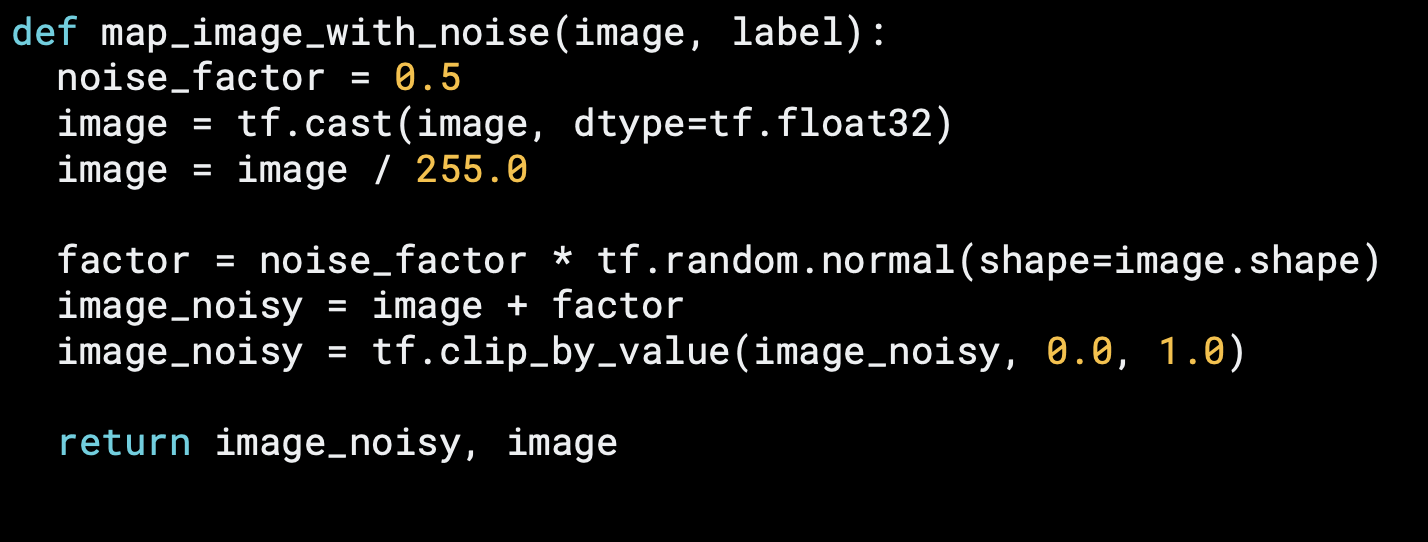
**Correct**

Correct!

**5.**

**Question 5**

Consider the following code for adding noise in an image. You use *tf.clip\_by\_value*to constrain the output image to values between 0 & 1.



**1 / 1 point**



False



True

**Correct**

Correct!