Question 1: what are differences between training dataset and test dataset and how do you manage the ratios between them? Explain.

⊕ The differences between training dataset and test dataset:

- Training dataset: the sample of data used to fit the model. The actual dataset that we use to train the model (weights and biases in the case of Neural Network). The mode sees and learns from this data.
- Test dataset: the sample of data (unseen data) used to provide an unbiased evaluation of a final model fit on the training dataset. The test set is generally what is used to evaluate or give prediction to the models.

⊕ Once the model has been chosen, it is trained on the entire dataset and tested on the unseen test set. In our example, the training dataset has a training set of 60,000 examples and a test set of 10,000 examples from MNIST database of handwritten digits 0-9.

Question 2: What are transponsed convolutions and their functional mechanism? Which applications do transponsed convolutions apply to? Explain.

⊕ Transposed convolutions are: also named fractionally-strided convolution or deconvolution, is the prediction of values for each pixel of layers. Compared to convolutions that reduce inputs through kernels, transposed convolutions broadcast inputs.

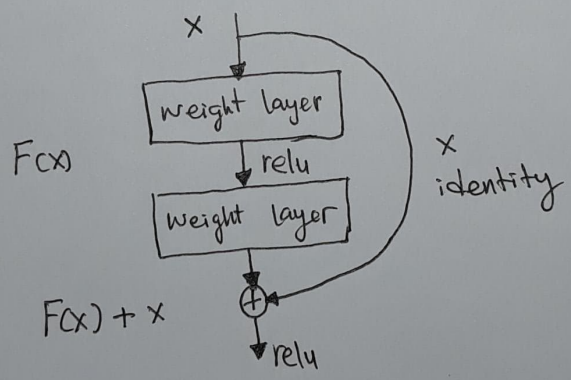
⊕ Functional mechanism: if a convolution layer reduces the input width and height by n_w and h_h time, respectively. Then a transponsed convolution layer with the same kernel sizes, padding and strides will increase the input width and height by n_w and $h_{h/}$ respectively.

⊕ Applications: we can implement convolution operations by the matrix multiplication, the corresponding transposed convolutions can be done by transposed matrix multiplication.

Question 3: Why is vanishing gradient problem so important? Are there solutions and how? Explain.

⊕ Vanishing gradient problem is so important because very deep "plain" networks do not work in practice, they are hard to train due to the vanishing gradients.

⊕ The skip-connections help to address the Vanishing Gradient problem. They also make it easy for a ResNet block to learn an identity function. There are two main types of blocks: the identity block and the convolutional block. Very deep Residual Networks are built by stacking these block together. Every two layers, there is an identity mapping via an element-wise addition. This proved to be very helpful for gradient propagation, as the error can be back-propaged through multiple paths.



Question 4: How can you avoid over-fitting? show some examples.

⊕ Over-fitting means a model that learns well from training dataset but does not perform well on a hold out sample.

To avoid over-fitting:

- Use enough training data: Example, minimum 10,000 data points to test and train a model
- Use accurate data
- Add Dropout layer to our model. Example: model.add(Dropout(as))