**Assignment\_3**

1. After each stride-2 conv, why do we double the number of filters?

**Ans: A stride 2 conv with the default padding (1) and ks (3) will reduce the activation map dimension by half. Formula: (n + 2\*pad - ks)//stride + 1. As the activation map dimension reduces by half we double the number of filters. This results in no overall change in computation as the network gets deeper and deeper.**

2. Why do we use a larger kernel with MNIST (with simple cnn) in the first conv?

3. What data is saved by ActivationStats for each layer?

**Ans: Adding one additional hidden layer doesn't have much effect, but adding 2 layers makes the output identical regardless of the input.**4. How do we get a learner's callback after they've completed training?

6. Draw up the benefits and drawbacks of practicing in larger batches?

Ans:

**Each batch can be subject to meticulous quality control and assurances, potentially causing increased employee downtime**. **Increased storage costs for large quantities of produced products. Errors with the batch produced will incur wasted time and cost.**

7. Why should we avoid starting training with a high learning rate?

**Ans: 1) It may increase training time.**

**2) If your learning rate is set too low, training will progress very slowly as you are making very tiny updates to the weights in your network. However, if your learning rate is set too high, it can cause undesirable divergent behavior in your loss function.**

8. What are the pros of studying with a high rate of learning?

**Ans: 1) It makes the model more accurate.**

**2) It increases the efficiency of model.**

9. Why do we want to end the training with a low learning rate?

**Ans: A lower learning rate means more training time. More time results in increased cloud GPU costs. A higher rate could result in a model that might not be able to predict anything accurately.**