1. Consider the regularization technique dropout (with hyper-parameter $p$, the probability of setting a particular input to 0). In class, we discussed scaling the inputs at inference time.

(a) Why is that scaling necessary? (Be specific and complete.)

**Solution:** Without any scaling, the average weighted input of the next layer would be $\frac{1}{1-p}$ higher during inference (without dropout), than during training (with dropout).

(b) How could scaling be done at training time instead? (Be specific in what scaling factor would be used.)

**Solution:** Scale each input in the dropout layer by $p$ at inference time.

(c) List any advantages or disadvantages of scaling at training time rather than inference time.

**Solution:**
- By scaling at training time, we avoid extra computation at inference time. Inference time may have strict latencies, or be done on resource-constrained hardware (e.g., limited CPU power, limited battery life).
- If scaling at training time, we can modify the dropout hyper-parameter $p$ over time without changing the average weighted input at the next layer.

2. A Generative Adversarial Network (GAN) consists of a generator, $G$, generating fakes, and a discriminator, $D$, that discriminates real objects from fake objects. The training of $G$ and $D$ consists of rounds where some training of one is done, and then training of the other is done. Please circle the true items:

A. $D$ is given training instances from: ground truth real objects with a label of 1, and from generated fake objects with a label of 0.

B. $D$ is given training instances consisting of pairs of objects: either one ground truth real object and one generated fake object, or two ground truth real objects.

C. When doing a round of training $G$, ground truth real objects are necessary.

D. **When doing a round of training $D$, ground truth real objects are necessary.**

E. Backpropagating error from the loss function $L_G$ to weights in $G$ require back-propagating errors from $L_G$ to weights in $D$.

F. **Backpropagating error from the loss function $L_D$ to weights in $D$ require back-propagating errors from $L_D$ to weights in $G$.**