Which of the following regularization in NN (implemented in PyTorch) lead to sparse

(A) **[Ans]** L1 regularization

(B) L2 regularization

(C) **[Ans]** Dropout

(D) Data Augmentation

(E) None of the above

A sparse set of weights in a Deep MLP is preferred:

(A) **[Ans]** it could lead to better generalization

(B) **[Ans]** it is compact and fit in lesser memory

(C) **[Ans]** it has many zeros and lesser amount of operations in forward pass

(D) **[Ans]** it is easy to train when the number of weights/parameters are less

(E) **[Ans]** All the above

While re-using a trained network for a new task:

(A) We always prefer to take the later (towards the end) layer

(B) **[Ans]** We always prefer to take an early(in the beginning) layer

(C) Which layer is more appropriate depends on the tasks.

(D) All the layers are equally useful.

(E) None of the above.

It is believed that adding noise is some sort of regularization.

(A) **[Ans]** Adding noise to the input is useful.

(B) Adding noise to the output/labels is useful. (for simplicity, assume the task is regression!).

(C) **[Ans]** Adding noise to the weights is useful.

(D) **[Ans]** Higher the noise the better the regularization.

(E) Lower the noise the better the regularization

Consider a problem where we do data augmentation and early stopping.

(A) With data augmentation, training accuracy is expected to increase.

(B) **[Ans]** With data augmentation, training accuracy may decrease.

(C) **[Ans]** With data augmentation, performance on the validation set is expected to increase.

(D) With data augmentation, the iteration where we do early stop, will increase.

(E) **[Ans]** With data augmentation, the iteration where we do early stop, will decrease.