Deep Learning Group Assignment Report (Group No 2)

Q1 You are given the iris dataset. 120 train examples. 30 test examples. Create a simple softmax classifier with 3 neurons in output layer and no hidden layer.

	Run1	**Run2**	**Run3**
Torch.manual_seed	299	798	500
np.random.seed	100	20	9
random.seed	53	50	55
Epochs	1000		
Number of Layers	Single output layer with 3 neurons		
Loss function	CrossEntropy		
Optimiser	Adagrad		
Learning Rate	0.09		
Test Accuracy	93%	73%	93%
Mean	86.33%		
Standard Deviation	0.12		

- Load training and testing data
- Divide training into train and validation
- Define train function using SGD optimiser and CrossEntropy loss (since cross entropy has inbuilt softmax thus we haven't used it explicitly)
- Define test function
- Calculate model accuracy on test data and report results

Q2 You are given the iris dataset. 120 train examples. 30 test examples. Create an MLP classifier with 3 hidden layers of sizes 5, 10, 5.

	Run1	**Run2**	**Run3**
Torch.manual_seed	32	29	94
np.random.seed	1	0	2
random.seed	60	55	61
Epochs	150		
Number of Layers	3 hidden layers, 1 output		
Loss function	CrossEntropy		
Optimiser	SGD		
Learning Rate	0.05		
Test Accuracy	96%	100%	96%
Mean	97.33%		
Standard Deviation	0.02		

- Load training and testing data
- Divide training into train and validation
- Define train function using SGD optimiser and CrossEntropy loss
- Define test function
- Calculate model accuracy on test data and report results

Q3. Create a CNN classifier to classify the pathology images as foreground vs background. The CNN should follow this architecture: CONV layer with 16 3x3 filters with pad 1 stride 1,RELU, POOL 2x2 with stride 2, CONV layer with 8 3x3 filters with pad 1 stride 1, RELU, POOL 2x2 with stride 2, Dense layer of size 64, RELU.

	Run1	**Run2**	**Run3**
Torch.manual_seed	4	6	8
np.random.seed	4	6	8
random.seed	4	6	8
Epochs	10		
Number of Layers	2 convolution, 1 fully connected, 1 output		
Loss function	CrossEntropy		
Optimiser	Adam		
Learning Rate	0.001		
Test Accuracy	99.79%	99.79%	95.67%
Mean	98.42%		
Standard Deviation	0.02		

- Load the background and foreground images folder using datasets.image loader function.
- Assign labels for background and foreground images
- Create the CNN architecture
- Define train function using Adam optimiser and CrossEntropy loss
- Define test function
- Calculate model accuracy on test data and report results

Q4. You are given the reviews dataset. These are 194439 amazon reviews for cell phones and accessories taken from https://jmcauley.ucsd.edu/data/amazon/ Use the reviewText and overall fields from this file. Use fastText embeddings as mentioned in tutorial. Create BiLSTM with 1 layer and 100 as hidden layer size. Use dropout=0.1. Use 1 dense hidden layer of size 50 before the final output layer.

	Run1	**Run2**	**Run3**
Torch.manual_seed	500	600	700
np.random.seed	0	10	20
random.seed	400	450	500
Epochs	10		
Number of Layers	1 BiLSTM Layer, 1 Dense Layer and 1 Output Layer		
Loss function	CrossEntropy		
Optimiser	Adamax		
Learning Rate	0.7 with gamma and Scheduler which helps decreasing learning rate with each epoch		
Test Accuracy	55.886%	55.886%	55.886%
Mean	55.886%		
Standard Deviation	0		

- Load the reviews data and change the reviews rating to 0-4 from 1-5.
- Create training and test datasets.
- Use word2vec to spit the words and generate model ready data.
- Create an embedding layer and then create a BiLSTM layer with 300 embeddings and 100 output layers. The output will go to dense layer of 50.
- Calculate model accuracy on test data and report results