AI/ML theory (CS 337) **Midsem**
Computer Science and Engineering **16-09-2022 Friday, Total time 2 hours 30 mins**
Indian Institute of Technology Bombay

Instructions

1. It is an OPEN BOOK and OPEN INTERNET examination.
2. This Lab has three questions. The maximum marks is 20.
3. You need to upload your code solution and output files in Moodle before the deadline.
4. Be sure to follow the upload instructions.
5. Total time for the examination is 2 hours 30 minutes.
6. This is an open-ended assignment. You are free to use any API from the following libraries to solve the problems: `pytorch`, `numpy`, `scipy`, `matplotlib`, `time`.

1. **Multi-Task Learning** For this problem, we are given with a dataset $D = \{(\mathbf{x}^i, y_1^i, y_2^i)\}_{i=1}^N$ of $N = 469$ samples where each training feature $\mathbf{x} \in \mathbb{R}^d$ with $d = 28$ features. Notice that unlike the problems explored in lab so far, this problem involves two target variables $y_1 \in \{0, 1\}$ and $y_2 \in \mathbb{R}$. The first task corresponding to y_1 is a simple binary classification task and the second task corresponding to y_2 involves a univariate regression. You are expected to build a multi-task learning (MTL) model that aims to solve both the tasks by exploiting the similarity between them. Towards this end, you will build a machine learning model that has an embedding network $\phi : \mathbb{R}^d \to \mathbb{R}^k$ where k is the embedding dimension. ϕ is shared across both the tasks. From ϕ, there are two task-specific network heads $\eta_1 : \mathbb{R}^k \to \{0, 1\}$ and $\eta_2 : \mathbb{R}^k \to \mathbb{R}$ that emits the output. You will design a loss function of the form $\mathcal{L}_1 + \lambda\mathcal{L}_2$ where $\lambda \in \mathbb{R}$ is a hyperparameter that trades-off between the losses corresponding to both these tasks.

   Note: You will not receive any credit if you solve these tasks using two independent models.

   You can train a model of your choice.You need to complete the function `predict_labels(model, X)` that takes in 2 arguments. `model` is the trained machine learning model and $X \in \mathbb{R}^{n \times d}$ are the test features for which we make predictions. This function returns two vectors `y1, y2` of shape $\mathbb{R}^n$ where `y1` is the class predictions and `y2` is the regression predictions.

   The following will be checked for evaluation:

   1.a The dumped test set predictions `y1_pred` and `y2_pred`, will be evaluated for accuracy in the classification task, and mean squared error in the regression task. (6 marks for classification, and 6 marks for regression)

   1.b The dumped model pickle will be checked to ensure that the task has been solved using a single model.

   1.c We will use the `predict_labels` function, along with the trained model pickle and the test dataset, to ensure consistency with the dumped predictions `y1_pred` and `y2_pred`.

c	/6+6	

2. This is a simple problem where you are expected to complete the code for computing the test metrics that are usually used to evaluate classification and regression tasks. Each of these functions receive two arguments namely `preds, targets` where `preds` $\in \mathbb{R}^n$ is a vector consisting of predictions and `targets` $\in \mathbb{R}^n$ is a vector consisting of the ground-truth. These functions should return a single scalar as output.

   - `accuracy`
   - `precision`
   - `recall`
   - `f1_score`
   - `mean_squared_error`
   - `mean_absolute_error`

c	/6	

3. It is a common practice to split the training dataset into train and validation splits and use the validation dataset to tune hyper-parameters. So in this question, you have to write a function that splits the training dataset into train and validation dataset. You need to complete the function `train_val_split(x, y1, y2, train_pc)` that takes the arguments as training features, classification target, regression target, train_pc where train_pc $\in [0, 1]$ represents the percentage of samples to be present in the training dataset. This function should return `x_trn, y1_trn, y2_trn, x_val, y1_val, y2_val`.

c	/2	

1 Submission instructions

Complete the functions in `assignment.py`. Do not modify the function signatures. Keep the file in a folder named `<ROLL_NUMBER>_quiz` and compress it to a tar file named `<ROLL_NUMBER>_quiz.tar.gz` using the command

```
tar -zcvf <ROLL_NUMBER>_quiz.tar.gz <ROLL_NUMBER>_quiz
```

Submit the tar file on Moodle. The directory structure should be -

```
<ROLL_NUMBER>_quiz
| - - - - assignment.py
| - - - - output.pkl
| - - - - model.pkl
```

Replace `ROLL_NUMBER` with your own roll number. If your Roll number has alphabets, they should be in *"small"* letters.

Total: 20