

**Question 1 (20 points):**

Construct the decision tree for the following dataset and show your derivation process.

ID	Features		Label
	X1	X2	Y
1	T	T	+
2	T	T	+
3	T	F	-
4	F	F	+
5	F	T	-
6	F	T	-

**Question 2 (80 points):**

The dataset consists of taxi trips spanning a period of days in Chicago. The training data records normalized taxi pick up counts for 100 regions at each hour of the day. We provide a sequence of 8 hours for each sample.

The feature matrix has dimensions (72000 x 8 x 49), where 72,000 = 30 days \* 24 hours \* 100 regions, 8 is the sequence length, and 49 features extracted from neighbors' information. The dimension of the label is (72000, 1). In addition, we provide the region location as (72000, 2), where each row represents the location of each grid (e.g., (7,7), (8,7), (6,7) are nearby grids). The time matrix is also provided as a (72000, 1) matrix, where each row represents the hour index of the time.

To load the training data, use the numpy.load function:

```
data = np.load('train.npz')
x = data['x'] #feature matrix
y = data['y'] #label matrix
location = data['locations'] #location matrix
times = data['times'] #time matrix
```

For hyperparameter tuning, the validation set (val.npz) has the same format of training. However, we do not provide the label (i.e., y) for the testing set (test.npz).

For you to compare the performance of your proposed model, you could implement a very simple baseline by predicting the average of historical records at a region.

You are required to predict the traffic demand value (i.e., pick-up volume) for the next time step. You can use any model to achieve the best performance. Some recommended modules: CNN, RNN.

Please use PyTorch/Tensorflow.

You need to submit:

1. Code
2. running script
3. predicted value on the testing data
4. A brief summary of your motivation, model structure and loss (400~1000 words)