2019 Spring COM526000 Deep Learning - Homework 2

Deep learning Model: Deep Neural Network

Due: Apr 22, 2019

INSTRUCTIONS

1. In this homework, MotionSense Dataset from github **Protecting Sensory Data against Sensitive Inferences** (`https://github.com/mmalekzadeh/motion-sense`) is utilized to build classification models. The datasets were already preprocessed. The last column, *Activities Types* represents participants performed Activities in trials to record the sensors (attitude, gravity, user-acceleration, and rotationRate).

2. **Please use *Data.csv* to train/test your models and report classification results generated from the hidden test set as *test_no_Ac.csv*.** The following columns should be included as predictors: *the max, mim, median, skew, standard deviation, mean of* ***attitude.roll, attitude.pitch, attitude.yaw, gravity.x, gravity.y, gravity.z, rotationRate.x, rotationRate.y, rotationRate.z, userAcceleration.x, userAcceleration.y, userAcceleration.z***. You need to transform *categorical* columns to one-hot encoding vectors. The target is *Activities Types*.

   - **Activities Types :**
     i. dws: downstairs
     ii. ups: upstairs
     iii. sit: sitting
     iv. std: standing
     v. wlk: walking
     vi. jog: jogging

3. **Some of APIs are forbidden in this homework, such asa *TensorFlow.DNNclassifier, Keras, TFlearn, slim, pytorch*.**

4. Name your source code that contains your *main* function as *hw2_StudentID.py* and your report as *hw2_StudentID.pdf*. You should provide your predictions for hidden test set following the format of example submissions (*StudentID_Ac.csv*). Please save the IDs and predictions. Due to Multi-class task, you need to process your answer for using function *argmax* to select the most probable class as your prediction P1.

5. **You should write your own codes independently. Plagiarism is strictly prohibitted.**

PROBLEMS

1. (90%) Classification: In this exercise, you will implement a DNN model for multi-class classification using *Data.csv*. The objective in this exercise is to create and train a neural network to identify *Activities Types* automatically. You need to split the data into training (80%) and validation (20%) data.

   (i) (40%) Please construct a DNN for multi-class classification according to the cross-entropy error function

   $$E(w) = -\frac{1}{m}\sum_{m=1}^{M}\sum_{i=1}^{C} t_{mi} log S_i,$$

   where t_{mi} is the ith target of the mth batch, M is the batch size, C is the classes for each sample, S_i is softmax activation of neural nets output function. Minimize the error function $E(w)$ by running the error backpropagation algorithm using the Adam Optimizer.You should decide the following variables: number of hidden layers, number of hidden units, learning rate, number of iterations and mini-batch size. **Please try to perform grid search over your variables mentioned above and show the best-performing setting for your model**

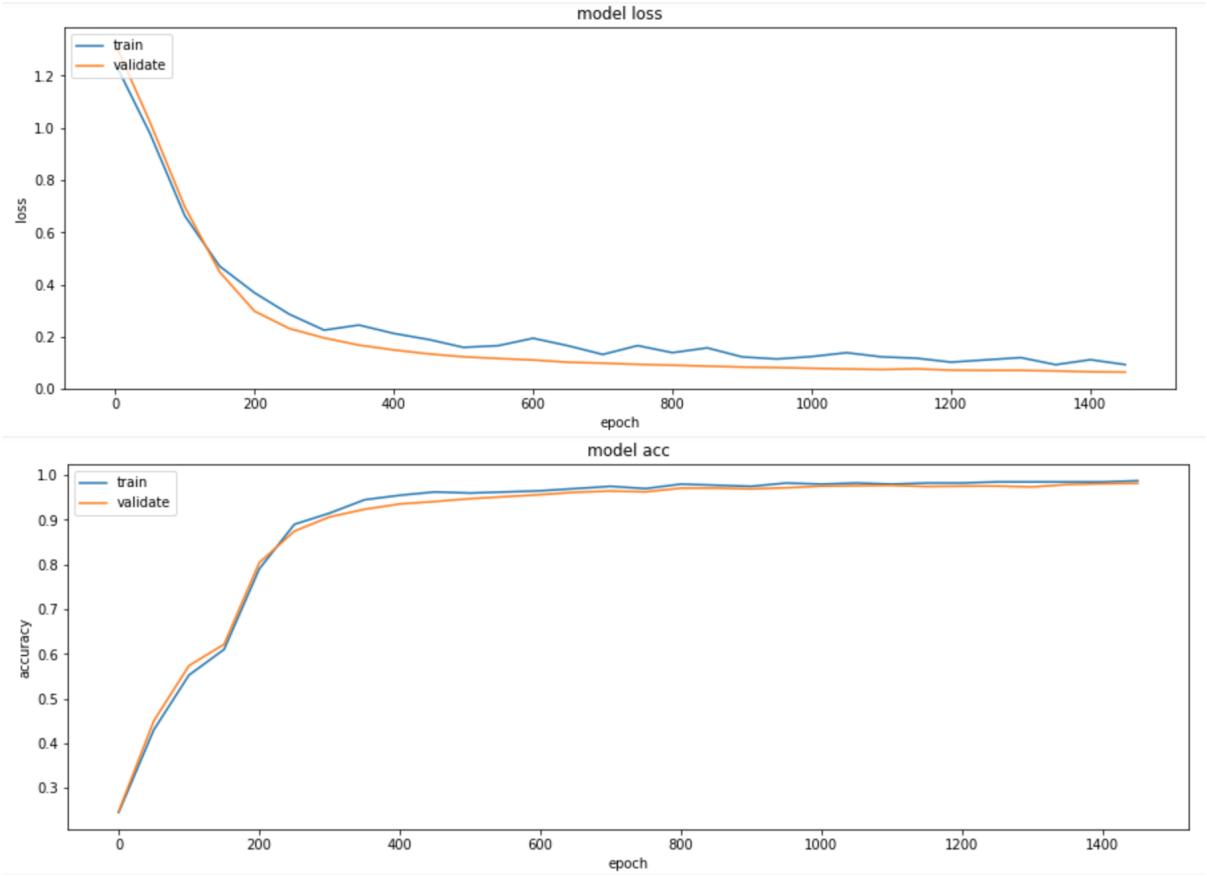

Figure 1: Example of loss and accuracy curve.

in the report. You also have to show your (a) training accuracy, (b) testing accuracy, (c) training loss and (d) testing loss in the report. An example is detailed in Figure 1.

(ii) (10%) Precision, recall, F1-score are other ways to evaluate model performance. For each class, **please record precision, recall and F1-score as well as the averages of those criteria** (***micro-average*** that will compute the metric independently for each class and then take the average and ***macro-average*** that will will aggregate the contributions of all classes to compute the average metric) over all classes in your report.

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

$$F - score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

(iii) (5%) Repeat (i), please compare with **different optimizers (Gradient descent, Adagrad)**. Explain what you observe.

(iv) (15%) **Please visualize the validation set through PCA and plot the data using 2 principal components**. Please use different colors to mark data points from the different classes. An example of a PCA plot in Figure 2(a).

(v) (15%) **Please project the validation set onto a 2D space by t-SNE**. Please use different colors to mark data points from the different classes. An example of a t-SNE plot in Figure 2(b).

(vi) (5%) Please explain what you observe from (iv) and (v).

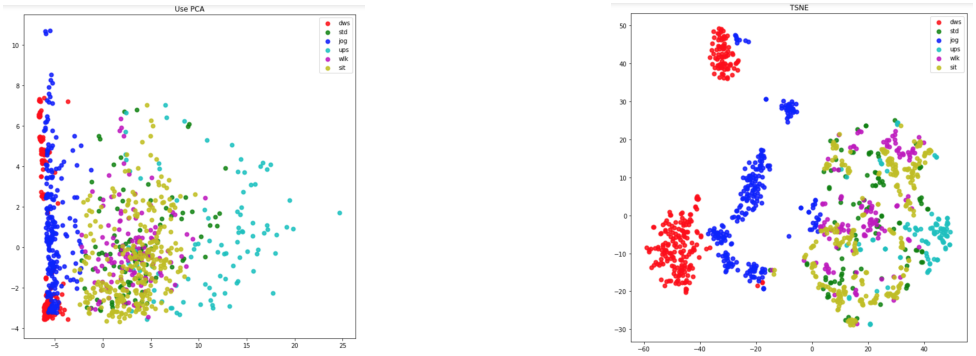

Figure 2: Example of PCA (a) and t-SNE (b).

2. (10%) Hidden Test Set :

(i) (10%) Apply the model to *test_no_Ac.csv* and save your classify results as *StudentID_answer.txt*.