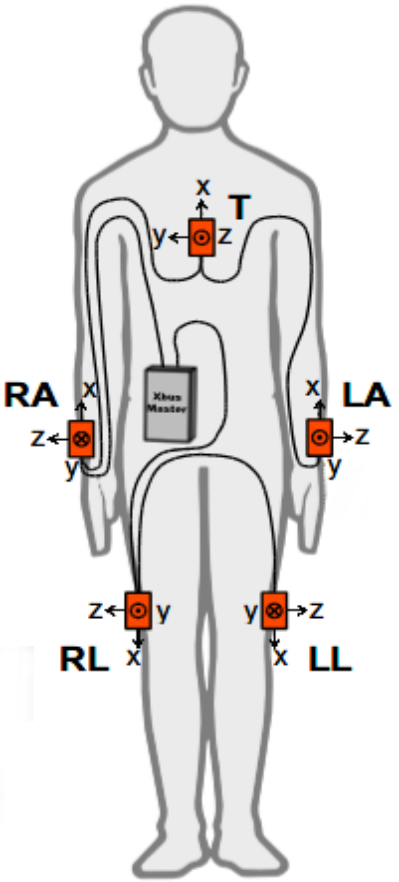
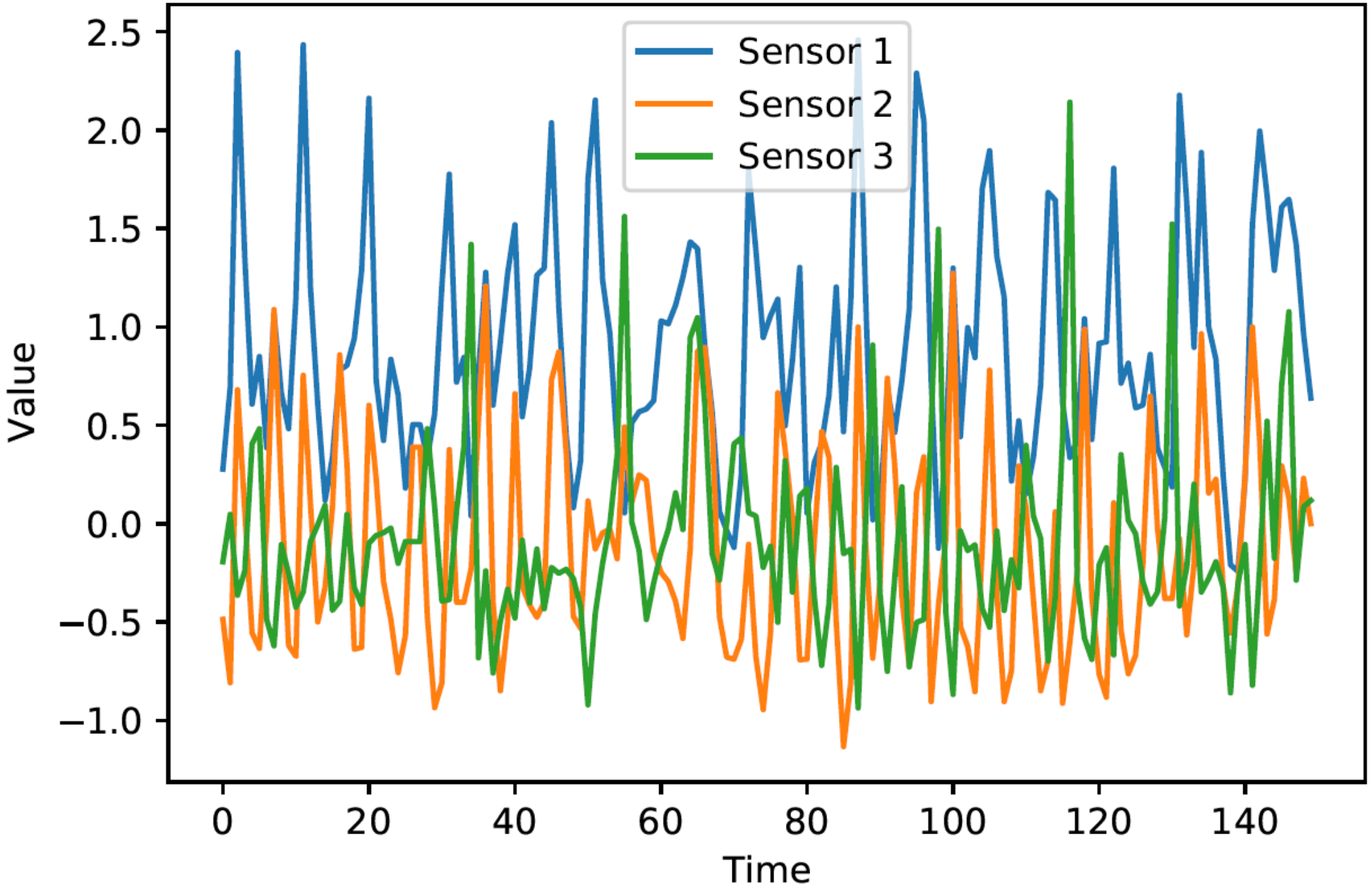
In this project we will consider an example of Human Activity Recognition (abbreviated as HAR) problem. The data set contains samples from 6 different human activities : Class 1 as downstairs motion, Class 2 as jogging motion, Class 3 as sitting motion, Class 4 as standing motion, Class 5 as upstairs motion, Class 6 as walking motion. The data set contains movement signals measured by three different sensors simultaneously. A sample plot of such an activity as sensor measurement vs. time is attached. The data set file, data-Mini Project 2.h5, contains time series of training and testing data (trX, tstX) and their corresponding labels (trY, tstY). The length of each time series is 150 units. The training set consists of 3000 samples and test set contains 600 samples. You are going to implement a fully recurrent neural network architecture trained with Back Propagation Through Time (BPTT) algorithm to solve the given multi-class time series classification problem. For details, refer to the lecture notes on Chapter 7.


x
T
y
z
RA
x
z
y
Xbus Master
x
LA
z
y
z
y
y
z
RL
x
x
LL


Sensor 1
Sensor 2
Sensor 3
Value
Time
2.5
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
0
20
40
60
80
100
120
140

a : The network architecture is as follows (refer to the class notes of Chp. 7). The hidden layer contains N neurons, hence W_{HH} is $N \times N$ matrix, W_{IH} is $N \times (3+1)$ matrix, where $+1$ is for bias terms. For hidden layer activation functions (f_H in the notes), use tangent hyperbolic functions. Output layer contains 6 output neurons, hence W_{HO} is $6 \times (N+1)$ matrix, where $+1$ is for bias terms. Each output corresponds to the class membership of one human activity. Choose output i for the indicator of class i, $i = 1, 2, \ldots, 6$. The desired output for class i is a one-hot vector which has 1 at output i and 0 for other outputs. Use sigmoidal activity functions at output layer and multi category cross entropy as the cost function, (refer to class notes on cross entropy).

b : Initialize the weights/biases by using uniform distribution in $[-0.1, 0.1]$. The hyperparameters are chosen as :

- The learning constant $\eta = 0.05, 0.1$.
- $N = 50, 100$,
- The mini batch size as $10, 30$.

For each of these cases run the BPTT algorithm for 50 epochs. Report the following : training error as a function of epochs, accuracy measured over the test set, list of patterns which are misclassified for both training and test set, and discuss your results. Note that the Top-1 accuracy is the percentage of accurate classification where correct output is maximum, which is the accuracy mentioned above. Top-2 accuracy is the percentage of the cases in which the correct output is among the top 2 maximum outputs, and so on. Discuss these Top-1, Top-2 and Top-3 correct classification accuracies for the test set.

c : After the step b, keep the best performing and second best performing configurations based on accuracy on test set. For these configurations only, do the following.

- Choose 10% of the training set data a validation set (i.e. 50 patterns for each activity). Run the BPTT algorithm on the training set (i.e. not on the validation set), but record the validation set error per epoch and perform early stopping. Report the validation error as a function of epoch number + the same items mentioned at the end of part b.

Instructions:

1. Prepare a report (including your answers/plots) to be uploaded on Moodle.
2. The report should be typeset (no handwriting allowed except for lengthy derivations, which may be scanned and embedded into the report).
3. Show all steps of your work clearly.
4. Unclear presentation of results will be penalized heavily.
5. No partial credits for unjustified answers.
6. **Use of any toolbox or library for neural networks is prohibited**.
7. Return all Matlab/Python code that you wrote in a single `.m/.py` file.
8. Code should be commented,
9. The code file should NOT return an error during runtime.
10. If the code returns an error at any point, the remaining part of your code will not be evaluated (i.e., 0 points).