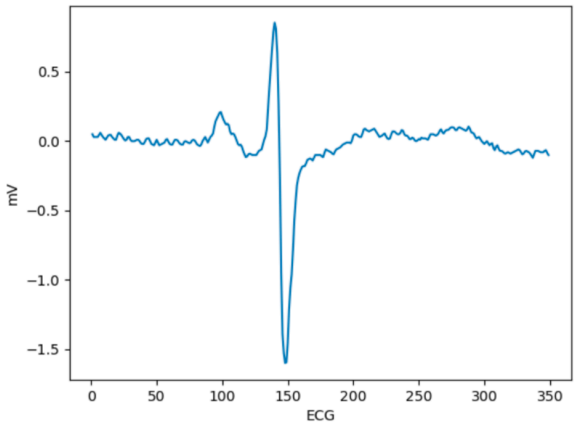
Shirin Mohebbi 9935597

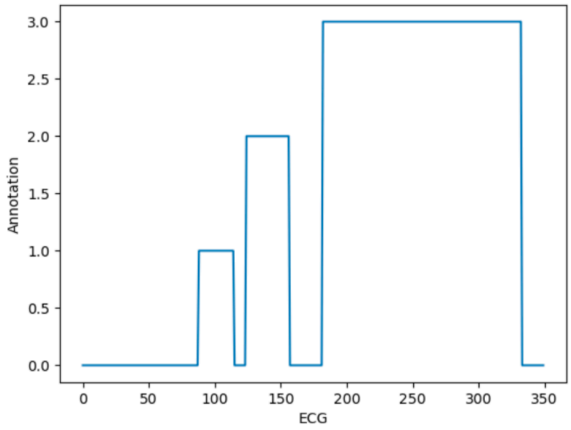
Homework#3: Detection and Identification of Electrocardiogram Signals using Recurrent Neural Networks (RNN)

Wednesday, June 23, 2021

In this assignment, we predict the secondary sequences (annotations) from their primary sequences (samples) .each sample sequence contains continuous data and it’s annotation sequence consist of discrete value 0 to 3.



sample



annotation

first we read data from sample files and their corresponding annotation files. then we have to extract patterns from each sample and it’s annotation based on the window size. for example if we have

sample: 1.5, 2.5, 3.5

annotation: 0, 1, 1

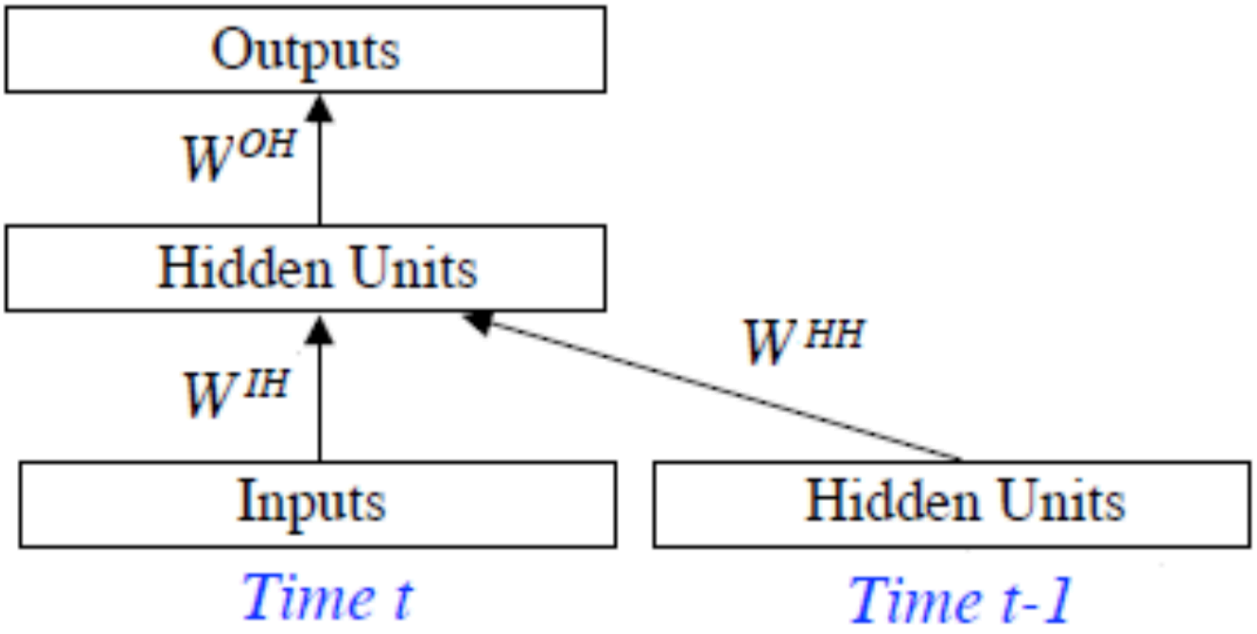
pattern with window 3:

1.5, 0, 2.5, 1, 3.5 and label 1

we extract all patterns for all input data. we use first 80% patterns of each data patterns for training and 20% rest for test.

for representing output, i used one 4 bit hot encoding since we have discrete values between 0 to 3.

first we train our network with Elman network



In Elman networks, input of hidden layer is input and value of previous state of hidden layer. output calculated as before. for simplification i truncated unfolded network to just one time step. to implement this part i used ElmanSimpleRecurrent from pyneurgen.recurrent library. we define number of neurons in each layer. (i choose w to be number of hidden layer because its half of the input size) and create an elman network. then we set learning rate and activation function type(i use sigmoid). then we give out training data and test data to model, we set

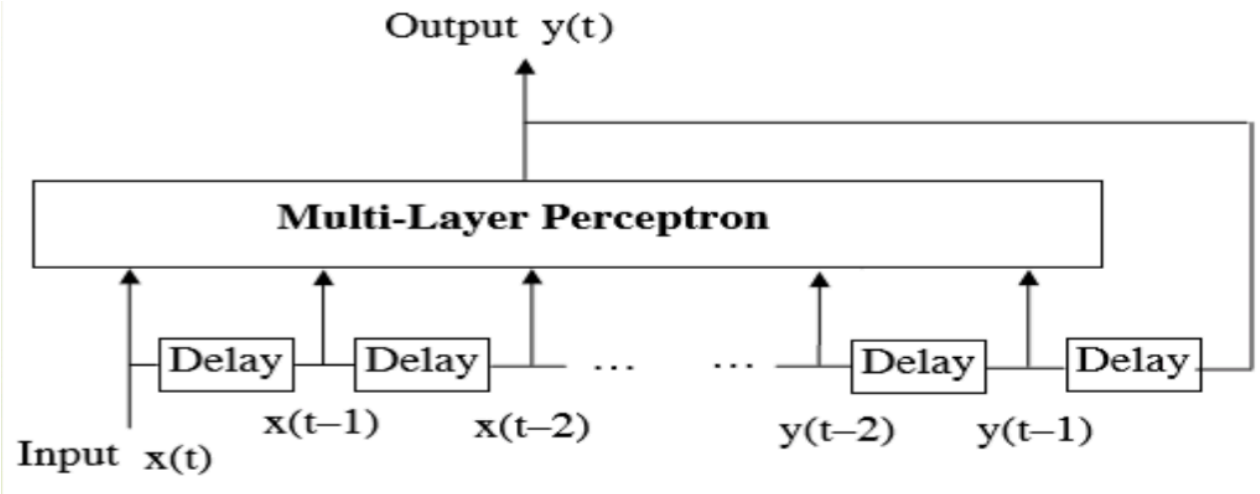
our arbitrary epoch number. and we learn and test the model. here is the result for w = 5, 11, 21

epoch: 0 MSE: 0.0340570942463748
epoch: 1 MSE: 0.0275189141479150
epoch: 2 MSE: 0.0270267797794185
epoch: 3 MSE: 0.0269956955508889
epoch: 4 MSE: 0.0269320385038898
accuracy test of elman rnn is: 70.67% and mse is: 0.08736 with window size: 21 and epoch number: 5
Runtime of the program is 7:53:49

epoch: 0 MSE: 0.035403748570942 46
epoch: 1 MSE: 0.026915075189141 47
epoch: 2 MSE: 0.026926779779418 54
epoch: 3 MSE: 0.026995695550888 97
epoch: 4 MSE: 0.026932038503889 88
accuracy test of elman rnn is: 91.23% and mse is: 0.08709 with window size: 11 and epoch number: 5
Runtime of the program is 5:03:51

epoch: 0 MSE: 0.0310678942463901
epoch: 1 MSE: 0.0265108914147431
epoch: 2 MSE: 0.0268012607796779
epoch: 3 MSE: 0.0259956955508889
epoch: 4 MSE: 0.0258320385038988
accuracy test of elman rnn is: 92.52% and mse is: 0.08603 with window size: 5 and epoch number: 5
Runtime of the program is 4:24:31

In narx networks, input of hidden layer is input and value of previous state of output layer with desired delay. output calculated as before. to implement this part i used NARXRecurrent from pyneurgen.recurrent library. we define number of neurons in each layer. (i choose w to be number of hidden layer because its half of the input size) and create a narx network. then we set learning rate and activation function type(i use sigmoid). then we give out training data and test data to model, we set our arbitrary epoch number. and we learn and test the model. here is the result for w = 5, 11, 21



epoch: 0 MSE: 0.0290367047604096
epoch: 1 MSE: 0.0285128848418794
epoch: 2 MSE: 0.0280424175156491
epoch: 3 MSE: 0.0279064707103217
epoch: 4 MSE: 0.0278872798865026
accuracy test of narx rnn is: 78.01% and mse is: 0.08376 with window size: 21 and epoch number: 5
Runtime of the program is 7:04:49

epoch: 0 MSE: 0.029405462644736 56
epoch: 1 MSE: 0.028956100850300 78
epoch: 2 MSE: 0.028924371194962 15
epoch: 3 MSE: 0.028919644496012 36
epoch: 4 MSE: 0.027980601940900 55
accuracy test of narx rnn is: 93.64% and mse is: 0.08245 with window size: 11 and epoch number: 5
Runtime of the program is 6:53:51

epoch: 0 MSE: 0.029065012688562 1
epoch: 1 MSE: 0.028568499094158 3
epoch: 2 MSE: 0.028086434885566 7
epoch: 3 MSE: 0.027985568159369 9
epoch: 4 MSE: 0.027850935225172 1
accuracy test of narx rnn is: 94.02% and mse is: 0.07980 with window size: 5 and epoch number: 5
Runtime of the program is 5:51:31