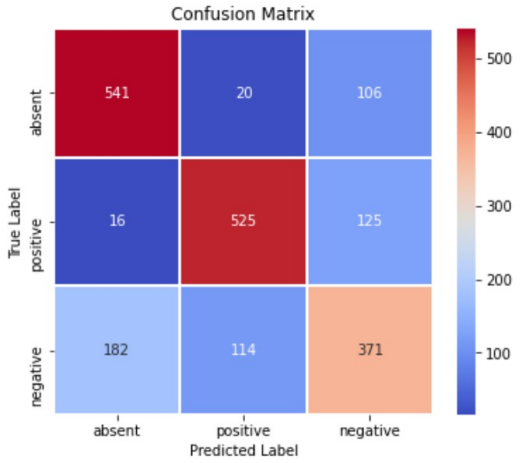
Exercise 3, convolutional neural networks

1. Experiment with different architectures (which was left as an open point when “NCONV” variable was introduced), in particular try at least another version where only one convolutional layer is introduced instead of two, and where any number of dense layers may be used, with the global constraint of using a network with at most 600 trainable parameters. The filter length in the second convolutional layer may be shorter.
   Is the number of parameters scaling more quickly by adding Dense layers or Conv1D layers?
   Check which configuration works better by comparing the performances quantitatively.
2. Visualize the “confusion matrix” (look for similar examples in the material available or by googling)

```
import seaborn
from sklearn import metrics
```



3. With two convolutional layers, plot also the weights of the second layer and try to see if regularization may improve the performances by varying the parameter lambda of the L1 or of the L2 regularization; see lambda in eqs.(43) and (52) in the review. There is also a mixed version (l1_l2?) that can be tried. Is any value of lambda good? Is there any intermediate value where the performances of the network are better? Is there any improvement in the visualization and understanding of the weights in the filters? Note that the regularization we introduced acts on the w’s, not on the biases. One can also try the equivalent procedure for biases or for the output of the relu units (see Keras doc.), if there is any reason for suspecting that it may help. In our case, the logic was to let the weights of the filters go to zero if not needed, hence that kind of regularization was selected.
4. By reducing the signal-to-noise ratio, namely the amplitude of the external signal in the data (A) with respect to the step typical size of the jump process, check where the network starts to fail discriminating the categories.

As usual, if there is time after completing the points, also other aspects (relu, adam, data normalization, weight initialization, filter lengths, filter number, etc.) may be modified to test the performances in different regimes.