Short Manual for training an ANN

1. ANN architecture

The general ANN architecture consists of five input branches (Figure 1). The first three correspond to the long-period spectral accelerations of the two horizontal components and the vertical component, respectively. The fourth branch includes source parameters, while the fifth represents the EC8 site classification (categories A, B, C, or D), encoded in binary form. Each input branch is first processed independently through fully connected layers, after which they are merged via a concatenation layer. The merged representation is then passed to a fully connected hidden layer. Finally, this shared hidden layer connects to three separate output layers, each predicting the short-period spectral accelerations of horizontal and vertical components. To ensure comparability across variables, z-score normalization is applied to the first three input branches. Hidden layers employ hyperbolic tangent activation functions, whereas the output layers use linear activation functions, with the number of nodes matching the dimensionality of the target values. Input and output spectral accelerations are given in $\log_{10}(SA)$, with SA in m/s^2. Distances are given in kilometers. The architecture of the ANN is defined inside the code train_ann_basics_h12.m.

It is possible to train ANNs only for the vertical component or both horizontal components. In these cases the corresponding branches are deactivated.

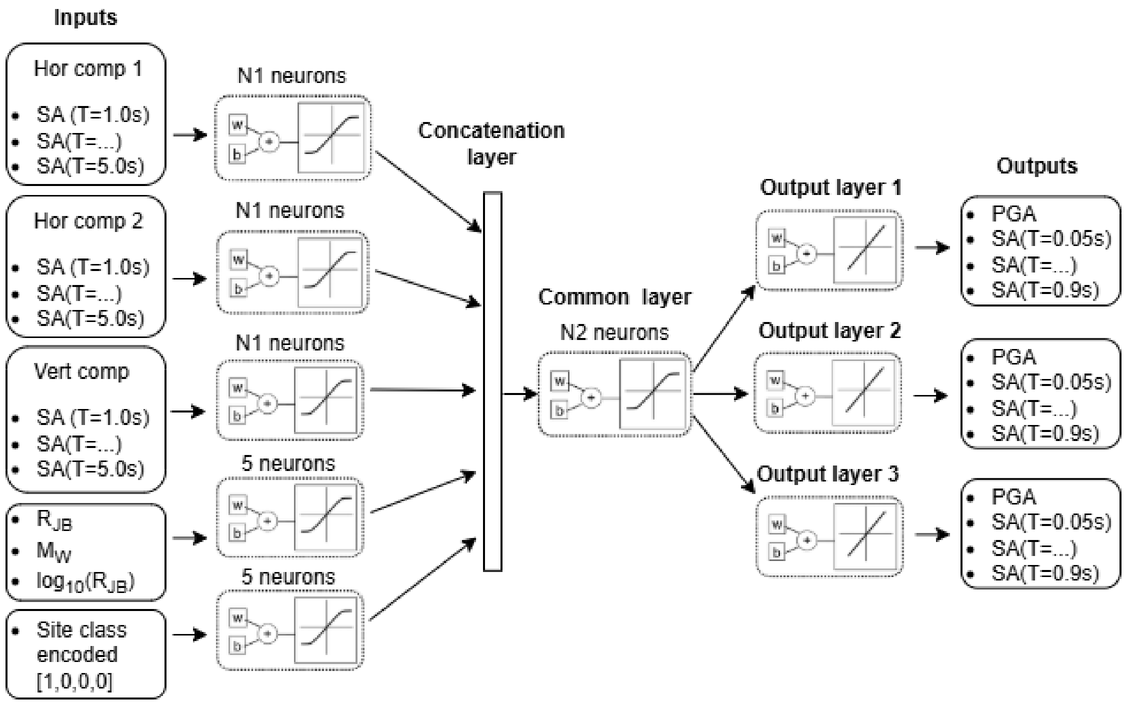

Figure 1. Architecture of the ANN.

2. Training

For training the ANNs you have to run the code drive.m. Inside you have the change the following options:

- dbn_name = 'ESM_SIMBADs'; Name of the database with the spectral accelerations for training. It should be located inside the subfolder database.
- TransferLearning = 'False'; Use 'True' if you want to do a two-step training with transfer learning. In that case you need to define also dbn_name2, which is the database for the second training (fine tunning), otherwise dbn_name2 is ignored.
- dbn_name2 = 'ESM_SIMBADs'; If TransferLearning = 'False' just use the same name as in dbn_name.
- num_nets = 10; Number of individual nets to be trained.
- n_LoopsANN = 1; Number of trained nets before choosing the best one. (usually 1).
- add_distance = 'True'; 'True' if you want to add R_{JB} as input for training.
- add_m = 'True'; 'True' if you want to add M_W as input for training.
- add_lndistance ='True'; 'True' if you want to add log10(R_{JB}) as input for training.
- separate_classes = 'False'; 'True' for activating the fourth branch of the ANN, i.e., EC8 site classes.
- nnr = [18,20]; Number of neurons per input and output component of spectral accelerations. With reference to Figure 1, N1=18 and N2=3*20.
- ann.trn.nr = 2; Choose 2 for training ANNs for horizontal (h12) and vertical (ud) components at the same time.
- TnC = [1,1]; Corner periods T* of the ANNs.
- cp = {'h12','ud'}; Components (ud=vertical;h12=both horizontals; or h12v= three components simultaneously, as in Figure 1).
- vTn; Vector with the periods at which the spectral accelerations of the database are computed. It should at least include these 30 periods: [0,0.04,0.05,0.07,(0.1:0.05:0.5), 0.6,0.7, 0.75,0.8,0.9,1.0:0.2:2.0,(2.5:0.5:5)].

The trained nets are stored in a Matlab structure. They are saved in the folder *ANNs*, with names having the following format:

net_(100*Tnc)_ALL_(cp)_(extra input variables)_(dbn_name)_v##

For instance, for the set of selected options above, the names of the files will be:

- Horizontal components:net_100_ALL_h12_Rjb_Mw_logRjb_ESM_SIMBADs_v##
- Vertical component:net_100_ALL_ud_Rjb_Mw_logRjb_ESM_SIMBADs_v##

Ten trained nets for the horizontal, vertical, and 3 simultaneous components are included in subfolder ANNs, for corner periods T*=0.8, 1, and 1.2 seconds.

3. Database

The database(s) for training should be inside the subfolder *database*. As a minimum it should contain spectral accelerations for the three components (psa_h1, psa_h2, and psa_v) in cm/s^2. Mw, site class and R_{JB} (in km) are only needed if they are going to be included in the training.

A combination of SIMBAD and the ESM databases is included (ESM_SIMBADs.mat), consisting of 2497 strong-motion records, with magnitudes M_W from 5 to 7.8, and R_{JB} lower than 110 km. Figure 2 shows the magnitude vs distance distribution of the records.

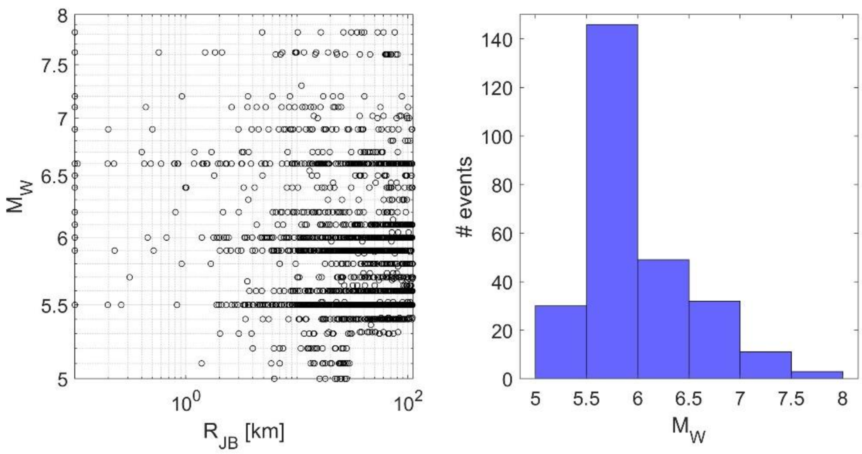

Figure 2. Magnitude vs distance R_{JB} scatter plot of recordings in the ESM+SIMBAD data set used for training the ANN (left). Histogram of the number of events per magnitude bin (right).

The format of the database should be as follows (a Matlab structure named SIMBAD), with these variable names:

1x2497 struct with 36 fields

Fields	rec	event_name					Mw						site_EC8				Rjb												psa_h1	psa_h2	psa_v
1	1	'AM-1988-00...	...	...	...	...	6.70...	"	'...	'...	...	...	'C'	...	...	...	30.8735	...	'...	'...	'...	'...	'...	'A'	[...	[...	[...	[...	37x1 dou...	37x1 dou...	37x1 dou...
2	2	'AM-1990-00...	...	...	...	...	5.50...	"	'...	'...	...	...	'B'	...	...	...	5.9553	...	'...	'...	'...	'...	'...	'A'	[...	[...	[...	[...	37x1 dou...	37x1 dou...	37x1 dou...
3	3	'AM-1990-00...	...	...	...	...	5.50...	"	'...	'...	...	...	'A'	...	...	...	14.5794	...	'...	'...	'...	'...	'...	'A'	[...	[...	[...	[...	37x1 dou...	37x1 dou...	37x1 dou...
4	4	'AM-1990-00...	...	...	...	...	5.50...	"	'...	'...	...	...	'B'	...	...	...	25.8610	...	'...	'...	'...	'...	'...	'A'	[...	[...	[...	[...	37x1 dou...	37x1 dou...	37x1 dou...
5	5	'AM-1990-00...	...	...	...	...	5.50...	"	'...	'...	...	...	'C'	...	...	...	46.1043	...	'...	'...	'...	'...	'...	'A'	[...	[...	[...	[...	37x1 dou...	37x1 dou...	37x1 dou...
6	6	'EMSC-2004...	...	...	...	...	5.50...	"	'...	'...	...	...	'A'	...	...	...	102.8533	...	'...	'...	'...	'...	'...	'...	[...	[...	[...	[...	37x1 dou...	37x1 dou...	37x1 dou...
7	7	'EMSC-2004...	...	...	...	...	5.50...	"	'...	'...	...	...	'A'	...	...	...	80.0241	...	'...	'...	'...	'...	'...	'...	[...	[...	[...	[...	37x1 dou...	37x1 dou...	37x1 dou...
8	8	'EMSC-2004...	...	...	...	...	5.50...	"	'...	'...	...	...	'E'	...	...	...	82.7041	...	'...	'...	'...	'...	'...	'...	[...	[...	[...	[...	37x1 dou...	37x1 dou...	37x1 dou...
9	9	'EMSC-2004...	...	...	...	...	5.50...	"	'...	'...	...	...	'B'	...	...	...	98.6419	...	'...	'...	'...	'...	'...	'...	[...	[...	[...	[...	37x1 dou...	37x1 dou...	37x1 dou...
10	10	'EMSC-2007...	...	...	...	...	5.90...	"	'...	'...	...	...	'B'	...	...	...	103.1640	...	'...	'...	'...	'...	'...	'...	[...	[...	[...	[...	37x1 dou...	37x1 dou...	37x1 dou...
11	11	'EMSC-2007...	...	-...	-...	...	5.50...	"	'...	'...	-...	-...	'A'	...	...	...	44.8743	...	'...	'...	'...	'...	'...	'...	[...	[...	[...	[...	37x1 dou...	37x1 dou...	37x1 dou...
12	12	'EMSC-2008...	...	-...	-...	...	5.70...	"	'...	'...	-...	-...	'A'	...	...	...	48.5003	...	'...	'...	'...	'...	'...	'...	[...	[...	[...	[...	37x1 dou...	37x1 dou...	37x1 dou...
13	13	'EMSC-2008...	...	-...	...	67	5.70...	"	'...	'...	-...	...	'B'	...	...	...	25.6043	...	'...	'...	'...	'...	'...	'...	[...	[...	[...	[...	37x1 dou...	37x1 dou...	37x1 dou...
14	14	'EMSC-2008...	...	-...	...	81	5.60...	"	'...	'...	-...	...	'C'	...	...	...	102.0132	...	'...	'...	'...	'...	'...	'...	[...	[...	[...	[...	37x1 dou...	37x1 dou...	37x1 dou...

Figure 3. Format of the Matlab structure with the data used for training.

4. Making predictions

For instance, for the ANN in net_100_ALL_h12_Rjb_Mw_logRjb_ESM_SIMBADs_v01, you can make predictions with the Matlab function *predict*:

[out_1,out_2] = predict(NNs.net,inp_1,inp_2,inp_3);

Where inp_1 and inp_2 are arrays with long-period spectral accelerations for each horizontal component. They should be given as $\log_{10}$(SA), with SA in m/s^2. Since in this case T*=1 s, there should be 12 input SA, at periods [1.0:0.2:2.0,(2.5:0.5:5)]. The predictions will be given also in log10, and they will be given for these periods [0,0.04,0.05,0.07, (0.1:0.05:0.5), 0.6,0.7, 0.75,0.8,0.9]. More than one prediction can be obtained at the same time, in such case the inputs and outputs are given as rows of the arrays. In this case the array inp_3 should have these three columns [R_{jb},M_w,$\log_{10}(R_{jb})$]. Rjb should be in km and limited to values larger than 0.01 km.

Paolucci, R., Gatti, F., Infantino, M., Smerzini, C., Ozcebe, A. G., and Stupazzini, M. (2018). Broadband ground motions from 3D physics-based numerical simulations using artificial neural networks. Bull. Seismol. Soc. Am., 108(3):1272–1286.

Hernández-Aguirre, V. M., Rupakhety, Paolucci, R., & Smerzini, C. (2025). Physics-based Broadband Ground Motion Simulations Enhanced by Transfer Learning. *Manuscript in Preparation*.