Prediction and Detection of Epileptic Seizures.

University of Coimbra (Masters in Intelligent Systems)
Subject: Computational Lerning

Serafim Barroca 2014213964

Pedro Ferreira 2012149657

Abstract

Exploration of brain signals with occurrences of epilepsy crises to train neural networks with the goal of forecasting and detection. Three types were used of neural networks (Feed Forward, Layer Recurrent and Elman) and comparing the performances of each in different conditions.

1 Introduction

Epilepsy is a disease that affects 1% of the population all around the world. In Portugal there are around 100 000 patients. In the present day, it's not possible to treat patients with any type of antibiotic or surgery. These can suffer attack at anytime, anywhere without prior notice. It may take a few seconds or minutes for this to occur and seriously affect the capabilities motor, perception, language, memory and consciousness.

Currently the possibility of predicting or detecting a crisis, based on information from brain signals, with the help of EEG (Electroencephalogram), is a research challenge.

The possibility of developing algorithms capable of detecting an attack in real time, would give the incentive to strategies for the removal of this crisis. By the other hand crisis forecasting would help the patient to take appropriate precautions in time to ensure their safety and exposure during the crisis.

As a starting point were collected EEG (Electroencephalogram) of patients at the previous, during, post and between seizures. These tests translate into brain activity.

2 Description of Dataset Provided

In this work the set of characteristics that represent the different bands of the electrical signals were extracted and provided by Dr Mojtaba Bandarabadi during his PhD in our department.

The characteristics were extracted using 2 seconds segments of EGE with 50% overlap. In practical terms each seconds a vector with 29 frequency bands is obtained.

With this, we obtained a matrix (FeatVectSel) in which column represents an extracted characteristic and each row a vector of characteristics. A vector (Trg) that is a filled column of 0's and 1's representing two states of the patient:

- The value 0 represents the non-ictal state (interictal, preictal, or postictal);
- The value 1 represents the ictal state.

Two patients were distributed to each student in which we were given the following two datasets with the following attributes:

Dataset ID	Nº of Crisis	Dataset Duration(h)	Crisis Duration(s)		
			Med.	Min.	Max.
44202	22	170.6	131.8	19	199
63502	19	118.9	103	8	155

Table 1: Dataset Description

3 Used Evaluation Methods

The performance of the classifier is calculated through sensitivity (SE) "how many real crises were predicted or detected" and of specificity (SP) "how many false crises were predicted or detected" and of precision (AC) "the proportion of correct states." For clinical use, high sensitivity and high specificity are required.

$$SE = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

$$SP = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}}$$

$$AC = \frac{\text{True}}{\text{True} + \text{False}}$$

4 Pre-Processing of Dataset

In order to be able to calculate the weigHs of the different neural networks we had to normalize the FeatVectSel matrix in the interval [0,1] to be able to have a certain identical influence for each characteristic.

Since the target matrix Trg provided only contains two states (non-ictal and ictal), in order to predict the crises we had to divide it into four states. For this we consider that 10 minutes (equivalent to 600 points) before each crisis as a state of Pre-Ictal "state in which the crisis is about to happen" and 5 minutes after the end of each crisis (equivalent to 300 points) as one state of Post-Ictal "state in which the crisis has just happened". The remaining points at 0 are now represented as Inter-Ictal state "normal patient state" and points at 1 remain as Ictal state "state in which the patient is having a crisis." With this breakdown we have obtained a new target matrix with the following possible columns:

- (1, 0, 0, 0) - "Inter-Ictal" state;
- (0, 1, 0, 0) - "Pre-ictal" state;
- (0, 0, 1, 0) - "Ictal" state;
- (0, 0, 0, 1) - "Post-Ictal" state;

5 Used Neural Networks

First we use the neural networks that were indicated in the project description for a first experimentation. Then after a survey of optional neural networks we choose some that we find interesting by describing them. We then decided to use the following three neural networks.

5.1 Feedforward Neural Network

This network consists of a series of layers. The first layer has the connection to the input of the training set. Each next layer has the connection to the previous layer. The last layer produces the output of the neural network.

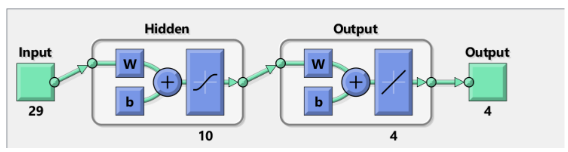

Figure 1: Feedforward Neural Network example.

5.2 Layer Recurrent Neural Network

Similar to the previous one with the exception that each layer has a recurring connection with an associated delay. This allows an infinite dynamic response to the time series input data.

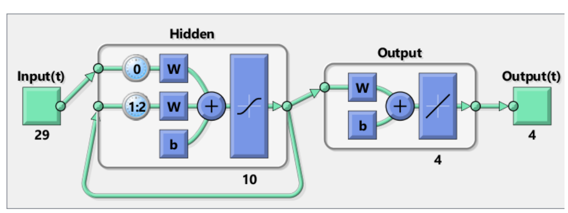

Figure 2: Layer Recurrent Neural Network example.

5.3 Elman Neural Network

Elman networks are Feedforward with the addition of recurring late connections associated.

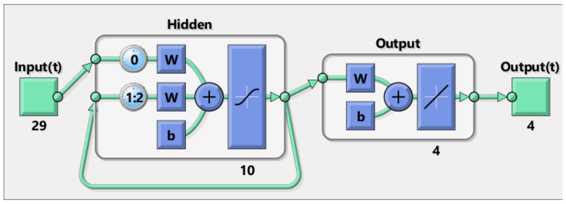

Figure 3: Elman Neural Network example.

6 Experimentation

For this experimentation we used Matlab R2017b, on which we implemented several scripts to analyze the selected metrics. With this we created two Dataset handlers:

6.1 Equal Number of Classes

This handler as the name suggests will make sure that the number of points of different states will be the same of Ictal points, for example, if we have a 200 points Ictal points then the Inter-Ictal, Pre-Ictal and Pos-Ictal have 200 points each.

Results with training(100%), testing(0%), validation(0%):

Dataset	TrainFunction	HiddenSize	SE	SP	PIA	IA	AC
44202	traincgb	10	77%	96%	0%	77%	2%
63502	traincgb	10	91%	69%	33%	67%	2%
44202	traincgb	20	85%	98%	0%	78%	81%
63502	traincgb	20	71%	96%	2%	66%	9%
44202	traincgb	29	97%	54%	76%	76%	2%
63502	traincgb	29	75%	95%	3%	67%	1%
44202	traincgp	10	92%	78%	21%	74%	57%
63502	traincgp	10	73%	96%	2%	68%	3%
44202	traincgp	20	82%	96%	1%	72%	80%
63502	traincgp	20	72%	96%	2%	67%	4%
44202	traincgp	29	96%	58%	63%	74%	3%
63502	traincgp	29	74%	95%	2%	67%	20%
44202	trainscg	10	96%	58%	65%	75%	3%
63502	trainscg	10	69%	96%	0%	67%	10%
44202	trainscg	20	97%	55%	77%	72%	3%
63502	trainscg	20	71%	95%	2%	67%	4%
44202	trainscg	29	96%	57%	67%	75%	3%
63502	trainscg	29	73%	96%	1%	68%	5%

Table 2: Feedfoward Neural Network Results.

Dataset	TrainFunction	HiddenSize	SE	SP	PIA	IA	AC
44202	traincgb	10	94%	68%	38%	77%	3%
63502	traincgb	10	70%	96%	0%	68%	3%
44202	traincgb	20	96%	60%	58%	73%	21%
63502	traincgb	20	73%	95%	2%	68%	2%
44202	traincgb	29	97%	55%	76%	73%	3%
63502	traincgb	29	80%	93%	6%	67%	2%
44202	traincgp	10	79%	92%	0%	79%	1%
63502	traincgp	10	84%	89%	11%	68%	7%
44202	traincgp	20	97%	54%	78%	72%	3%
63502	traincgp	20	69%	96%	0%	68%	7%
44202	traincgp	29	97%	56%	73%	75%	2%
63502	traincgp	29	77%	94%	4%	68%	13%
44202	trainscg	10	96%	58%	66%	74%	7%
63502	trainscg	10	78%	94%	6%	67%	7%
44202	trainscg	20	95%	65%	48%	74%	13%
63502	trainscg	20	76%	95%	4%	67%	1%
44202	trainscg	29	96%	57%	67%	75%	3%
63502	trainscg	29	68%	96%	0%	67%	4%

Table 3: Layer Recurrent Neural Network results.

Dataset	TrainFunction	HiddenSize	SE	SP	PIA	IA	AC
44202	traincgb	10	78%	96%	0%	76%	7%
63502	traincgb	10	64%	97%	0%	65%	1%
44202	traincgb	20	68%	78%	0%	67%	1%
63502	traincgb	20	75%	95%	3%	68%	2%
44202	traincgb	29	80%	94%	3%	75%	63%
63502	traincgb	29	78%	94%	5%	68%	6%
44202	traincgp	10	97%	58%	64%	71%	14%
63502	traincgp	10	68%	96%	0%	68%	5%
44202	traincgp	20	97%	55%	76%	74%	5%
63502	traincgp	20	68%	96%	0%	68%	32%
44202	traincgp	29	97%	54%	77%	75%	3%
63502	traincgp	29	70%	96%	0%	68%	10%
44202	trainscg	10	96%	58%	65%	75%	3%
63502	trainscg	10	72%	96%	1%	67%	7%
44202	trainscg	20	96%	57%	67%	75%	3%
63502	trainscg	20	80%	93%	7%	67%	3%
44202	trainscg	29	95%	63%	52%	74%	19%
63502	trainscg	29	74%	96%	2%	67%	29%

Table 4: Elman Neural Network results.

Results with training(70%), testing(30%), validation(0%):

Dataset	TrainFunction	HiddenSize	SE	SP	PIA	IA	AC
44202	traincgb	10	83%	98%	0%	81%	44%
63502	traincgb	10	59%	92%	3%	52%	36%
44202	traincgb	20	94%	71%	72%	74%	59%
63502	traincgb	20	51%	97%	0%	50%	37%
44202	traincgb	29	97%	55%	74%	71%	3%
63502	traincgb	29	69%	96%	1%	66%	2%
44202	traincgp	10	96%	69%	80%	76%	61%
63502	traincgp	10	90%	55%	66%	53%	35%
44202	traincgp	20	84%	98%	0%	74%	56%
63502	traincgp	20	58%	94%	2%	53%	38%
44202	traincgp	29	86%	88%	8%	74%	66%
63502	traincgp	29	84%	88%	11%	68%	1%
44202	trainscg	10	93%	76%	58%	72%	55%
63502	trainscg	10	59%	93%	2%	54%	36%
44202	trainscg	20	92%	83%	45%	75%	53%
63502	trainscg	20	57%	94%	2%	52%	36%
44202	trainscg	29	94%	69%	37%	76%	2%
63502	trainscg	29	73%	96%	2%	67%	2%

Table 5: Feedfoward Neural Network Results.

Dataset	TrainFunction	HiddenSize	SE	SP	PIA	IA	AC
44202	traincgb	10	93%	75%	62%	75%	57%
63502	traincgb	10	58%	93%	1%	53%	36%
44202	traincgb	20	85%	98%	0%	82%	47%
63502	traincgb	20	63%	90%	6%	53%	38%
44202	traincgb	29	72%	98%	0%	70%	6%
63502	traincgb	29	79%	94%	6%	67%	1%
44202	traincgp	10	85%	97%	2%	78%	51%
63502	traincgp	10	55%	96%	0%	55%	37%
44202	traincgp	20	94%	74%	65%	80%	59%
63502	traincgp	20	64%	90%	6%	54%	38%
44202	traincgp	29	97%	55%	75%	73%	3%
63502	traincgp	29	67%	97%	0%	66%	5%
44202	trainscg	10	88%	89%	23%	75%	56%
63502	trainscg	10	54%	95%	0%	51%	36%
44202	trainscg	20	91%	85%	43%	76%	53%
63502	trainscg	20	59%	93%	3%	53%	38%
44202	trainscg	29	96%	60%	59%	75%	7%
63502	trainscg	29	68%	96%	0%	66%	38%

Table 6: Layer Recurrent Neural Network results.

Dataset	TrainFunction	HiddenSize	SE	SP	PIA	IA	AC
44202	traincgb	10	77%	99%	0%	77%	43%
63502	traincgb	10	52%	97%	0%	51%	36%
44202	traincgb	20	94%	77%	61%	78%	57%
63502	traincgb	20	51%	97%	0%	51%	36%
44202	traincgb	29	97%	56%	71%	73%	5%
63502	traincgb	29	82%	90%	9%	68%	1%
44202	traincgp	10	87%	93%	15%	75%	46%
63502	traincgp	10	56%	95%	1%	53%	39%
44202	traincgp	20	95%	72%	74%	75%	59%
63502	traincgp	20	63%	91%	5%	53%	39%
44202	traincgp	29	97%	56%	72%	73%	4%
63502	traincgp	29	71%	96%	1%	68%	10%
44202	trainscg	10	93%	78%	59%	73%	56%
63502	trainscg	10	58%	92%	2%	52%	38%
44202	trainscg	20	93%	77%	56%	76%	55%
63502	trainscg	20	57%	94%	1%	53%	38%
44202	trainscg	29	96%	59%	62%	75%	6%
63502	trainscg	29	72%	96%	1%	67%	5%

Table 7: Elman Neural Network results.

6.2 Low InterIctal Points

This final handler will make sure that the for each Ictal Phase , the InterIctal, Pre-Ictal and Post-Ictal have 300,600,300 corresponding points.

Results with training(100%), testing(0%), validation(0%) and input simulation from opposite dataset:

Dataset	TrainFunction	HiddenSize	SE	SP	PIA	IA	AC
44202	traincgb	10	100%	49%	99%	65%	2%
63502	traincgb	10	99%	49%	98%	53%	2%
44202	traincgb	20	98%	50%	97%	61%	3%
63502	traincgb	20	99%	49%	98%	56%	2%
44202	traincgb	29	98%	51%	94%	64%	3%
63502	traincgb	29	99%	49%	98%	56%	2%
44202	traincgp	10	98%	51%	94%	67%	3%
63502	traincgp	10	99%	50%	98%	57%	2%
44202	traincgp	20	98%	51%	95%	62%	3%
63502	traincgp	20	99%	49%	98%	57%	2%
44202	traincgp	29	98%	51%	96%	63%	3%
63502	traincgp	29	99%	49%	98%	57%	2%
44202	trainscg	10	100%	49%	99%	66%	2%
63502	trainscg	10	99%	49%	98%	56%	2%
44202	trainscg	20	99%	50%	99%	66%	2%
63502	trainscg	20	99%	49%	98%	56%	2%
44202	trainscg	29	99%	50%	99%	65%	2%
63502	trainscg	29	99%	49%	98%	57%	2%

Table 8: Feedfoward Neural Network Results.

Dataset	TrainFunction	HiddenSize	SE	SP	PIA	IA	AC
44202	traincgb	10	98%	51%	94%	64%	3%
63502	traincgb	10	99%	49%	98%	54%	2%
44202	traincgb	20	98%	50%	97%	62%	3%
63502	traincgb	20	99%	49%	98%	53%	2%
44202	traincgb	29	98%	51%	95%	62%	3%
63502	traincgb	29	99%	49%	98%	53%	2%
44202	traincgp	10	99%	50%	99%	64%	2%
63502	traincgp	10	99%	49%	98%	56%	2%
44202	traincgp	20	98%	51%	94%	65%	3%
63502	traincgp	20	99%	50%	98%	57%	2%
44202	traincgp	29	98%	50%	97%	65%	3%
63502	traincgp	29	99%	49%	98%	59%	2%
44202	trainscg	10	99%	50%	98%	62%	3%
63502	trainscg	10	99%	49%	98%	58%	2%
44202	trainscg	20	99%	50%	98%	64%	3%
63502	trainscg	20	99%	49%	98%	57%	2%
44202	trainscg	29	99%	50%	99%	64%	2%
63502	trainscg	29	99%	49%	98%	57%	2%

Table 9: Layer Recurrent Neural Network results.

Dataset	TrainFunction	HiddenSize	SE	SP	PIA	IA	AC
44202	traincgb	10	100%	49%	99%	67%	2%
63502	traincgb	10	99%	49%	98%	56%	2%
44202	traincgb	20	100%	49%	99%	53%	2%
63502	traincgb	20	99%	49%	98%	57%	2%
44202	traincgb	29	98%	50%	97%	65%	3%
63502	traincgb	29	99%	49%	98%	56%	2%
44202	traincgp	10	100%	49%	99%	66%	2%
63502	traincgp	10	99%	49%	98%	55%	2%
44202	traincgp	20	98%	50%	97%	66%	3%
63502	traincgp	20	99%	49%	98%	56%	2%
44202	traincgp	29	98%	51%	94%	61%	3%
63502	traincgp	29	99%	49%	98%	57%	2%
44202	trainscg	10	98%	51%	94%	60%	3%
63502	trainscg	10	99%	50%	98%	56%	2%
44202	trainscg	20	98%	50%	98%	64%	3%
63502	trainscg	20	99%	49%	98%	57%	2%
44202	trainscg	29	99%	50%	99%	65%	2%
63502	trainscg	29	99%	50%	98%	56%	2%

Table 10: Elman Neural Network results.

Results with training(70%), testing(30%), validation(0%):

Dataset	TrainFunction	HiddenSize	SE	SP	PIA	IA	AC
44202	traincgb	10	100%	48%	100%	70%	51%
63502	traincgb	10	100%	47%	98%	41%	48%
44202	traincgb	20	99%	62%	98%	62%	70%
63502	traincgb	20	99%	47%	98%	42%	48%
44202	traincgb	29	99%	56%	99%	61%	63%
63502	traincgb	29	99%	47%	97%	43%	48%
44202	traincgp	10	99%	63%	96%	63%	69%
63502	traincgp	10	100%	47%	98%	43%	48%
44202	traincgp	20	99%	62%	98%	59%	70%
63502	traincgp	20	99%	47%	97%	44%	48%
44202	traincgp	29	99%	63%	97%	61%	69%
63502	traincgp	29	100%	47%	98%	42%	48%
44202	trainscg	10	99%	63%	97%	58%	69%
63502	trainscg	10	99%	47%	98%	41%	48%
44202	trainscg	20	99%	64%	95%	62%	69%
63502	trainscg	20	99%	47%	97%	44%	48%
44202	trainscg	29	99%	64%	95%	67%	69%
63502	trainscg	29	100%	47%	98%	42%	48%

Table 11: Feedfoward Neural Network Results.

Dataset	TrainFunction	HiddenSize	SE	SP	PIA	IA	AC
44202	traincgb	10	100%	48%	99%	75%	52%
63502	traincgb	10	99%	47%	98%	41%	48%
44202	traincgb	20	99%	63%	96%	59%	68%
63502	traincgb	20	99%	47%	98%	42%	48%
44202	traincgb	29	99%	49%	99%	58%	51%
63502	traincgb	29	100%	47%	98%	42%	48%
44202	traincgp	10	99%	61%	98%	33%	67%
63502	traincgp	10	99%	47%	98%	43%	48%
44202	traincgp	20	99%	61%	99%	59%	70%
63502	traincgp	20	99%	47%	98%	43%	48%
44202	traincgp	29	99%	48%	99%	62%	51%
63502	traincgp	29	99%	47%	98%	42%	48%
44202	trainscg	10	99%	63%	95%	55%	68%
63502	trainscg	10	99%	47%	97%	42%	48%
44202	trainscg	20	99%	62%	97%	64%	70%
63502	trainscg	20	99%	47%	97%	43%	48%
44202	trainscg	29	99%	61%	99%	69%	70%
63502	trainscg	29	99%	47%	97%	42%	48%

Table 12: Layer Recurrent Neural Network results.

Dataset	TrainFunction	HiddenSize	SE	SP	PIA	IA	AC
44202	traincgb	10	99%	61%	98%	55%	69%
63502	traincgb	10	100%	46%	98%	39%	48%
44202	traincgb	20	99%	61%	99%	45%	68%
63502	traincgb	20	100%	46%	98%	33%	48%
44202	traincgb	29	99%	64%	95%	68%	69%
63502	traincgb	29	99%	47%	98%	40%	48%
44202	traincgp	10	99%	61%	98%	59%	70%
63502	traincgp	10	99%	47%	98%	41%	48%
44202	traincgp	20	99%	64%	95%	69%	69%
63502	traincgp	20	99%	47%	98%	42%	48%
44202	traincgp	29	99%	59%	99%	59%	68%
63502	traincgp	29	99%	47%	97%	43%	48%
44202	trainscg	10	99%	61%	99%	60%	70%
63502	trainscg	10	100%	47%	98%	43%	48%
44202	trainscg	20	99%	62%	98%	63%	70%
63502	trainscg	20	99%	47%	97%	42%	48%
44202	trainscg	29	99%	53%	99%	58%	59%
63502	trainscg	29	99%	47%	98%	43%	48%

Table 13: Elman Neural Network results.

7 Best Neural Networks

NNtype	Handler	Dataset	TrainFunction	HiddenSize	SE	SP	PIA	IA	AC
Recurrent	Even	44202	traincgb	20	85%	98%	0%	82%	47%
Elman	Low	44202	trainscg	20	99%	62%	98%	63%	70%
Elman	Even	44202	traincgp	20	95%	72%	74%	75%	59%

Table 14: Best Neural Networks

There is a few interesting things we can extract from this table. The best Neural Network were made with 44202.mat and the possible reason is that it's the dataset with the most Ictal phases. The best hidden layer size is 20. The Low InterIctal Points handler is very good to provide the NN the best Pre-Ictal training but at the cost of specificity. We noticed a drastic downfall in Network performance when the simulation was done by the opposite dataset, and after a small research we found that the possible reason to justify this is that not all Epilepsy attacks are the same, as the project description show the 44202.mat and 63502.mat don't have similar seizures types.

8 Conclusion

In terms of data balancing, the pre-processing handlers are essential to provide the Neural Network better results. The Elman Neural Network is the best to provide a good Pre-Ictal detection with the trainscg function as well a hidden layer size of 20. Finally the best way to predict and detect a seizure is to make sure that the Neural Network used was trained with the same type of seizure as the patient in question.