Voice recognition

LABO 3

Berney Alec & Ferrari Teo
MLG – 25.04.2022

Questions

HOLD_OUT_VALIDATION NOTEBOOK

Q1. Determine where do we define all the parameters mentioned above.

We can find them here:

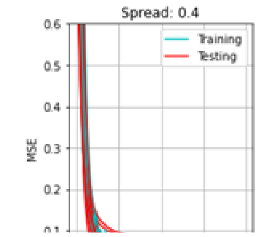
Experiment

In this experiment we create datasets with different degrees of complexity and we test the behaviour of hold-out validation with each one of them. For each dataset, we split the dataset several times, which generates different partitions training/testing. We also initializes the neural networks several times with each partition in order to be sure that the results are not a special case of a lucky initialization.

```
Entrée [7]: N_INITS = 2
            N_SPLITS = 10
            DATASET_SIZE = 200
            EPOCHS = 100
            N_NEURONS = 2
            LEARNING_RATE = 0.001
            MOMENTUM = 0.7
            TRAIN_TEST_RATIO = 0.8
            DATA_PARAMS = np.arange(0.4, 0.71, 0.1)
```

Observe that we run the evaluation procedure on four different problems. Each problem is a two-class two-dimensional problem, where the two sets are more and more overlapped (e.g., the synthetic datasets are randomly generated using variances of 0.4, 0.5, 0.6 and 0.7).

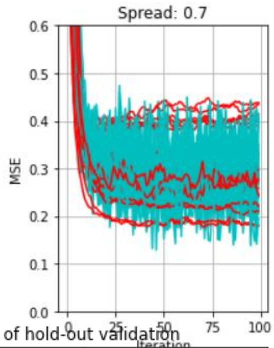
Q2. What are the cyan and red curves in those plots? Why are they different?



As indicated in the first image (on the right) :

- The blue or cyan curve, show the mean square error (MSE) evolution during **training**
- The red curve, show the mean square error (MSE) evolution during **testing**.

Q3. What happens with the training and test errors (MSE) when we have the two sets more overlapped?



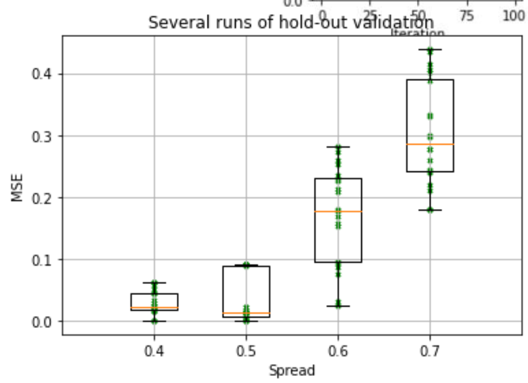
We see that the MSE variation increases compared to lower spread data.

For a spread of 0.7, the MSE is also higher at his lowest point compared to lower spread data.

Q4. Why sometimes the red curves indicate a higher error than the cyan ones?

Because the testing is done with values that haven't been used for training.

Q5. What is showing the boxplot summarizing the validation errors of the preceding experiments?



We see that the variation / gap between highest and lowest MSE is increasing with the spread of data. It is easily visible with the boxplot interquartile range also increasing with the spread of data. As already said before in the document, the MSE lowest point is also increasing.

CROSS_VALIDATION NOTEBOOK

Q1. Determine where do we define all the above-mentioned parameters.

We can find them here:

Experiment

In this experiment we create datasets with different degrees of complexity and we test the behaviour of k-fold cross-validation with each one of them. For each dataset, we split the dataset several times, which generates different partitions training/testing.

Entrée [7]:
```
N_SPLITS = 10
DATASET_SIZE = 200
EPOCHS = 20
N_NEURONS = 2
K = 5
LEARNING_RATE = 0.001
MOMENTUM = 0.7
DATA_PARAMS = np.arange(0.4, 0.71, 0.1)
```

Q2. What is the difference between hold-out and cross-validation? What is the new parameter that has to be defined for cross-validation?

Hold-out use only a percentage of data to learn the model and keep the other part for testing. So, it doesn't use all te data for learning, that is potentially a probleme.

Cross-validation use all the data to learn, but for each EPOCH of testing, will select the k ième data of the data set to test the model.

The new parameter from cross-validation is the value: K.

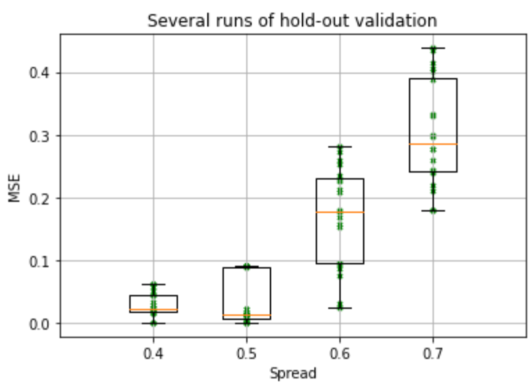
Q3. Observe the boxplots summarizing the validation errors obtained using the cross-validation method and compare them with those obtained by hold-out validation

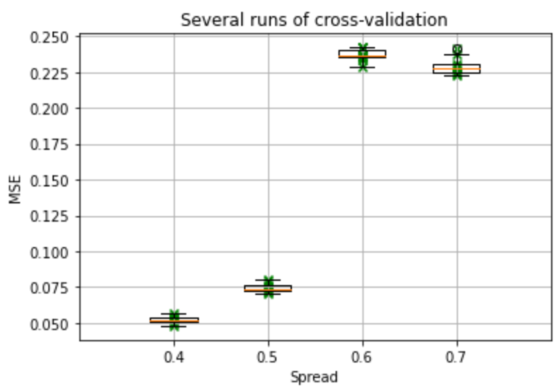
For the cross-validation, we see that the variation / gap between lowest and higher point isn't increasing with the spread of data. That was the case for hold-out method.

It can be easily seen by interquartile range.

But the lowest point of MSE is increasing in both methods.

The main difference is that cross-validation, by his low variation / gap, validate that the result.





Experiments

For our experiments we have some parameter that are the same for each experiment. For each experiment we have 13 neurons in input, a variable number of hidden neurons and 1 or 3 output neurons depending on the number of input classes. The attributed synaptic weights are random and the activation function is always tanh. We used the mean and not the standard deviation because it gave better results.

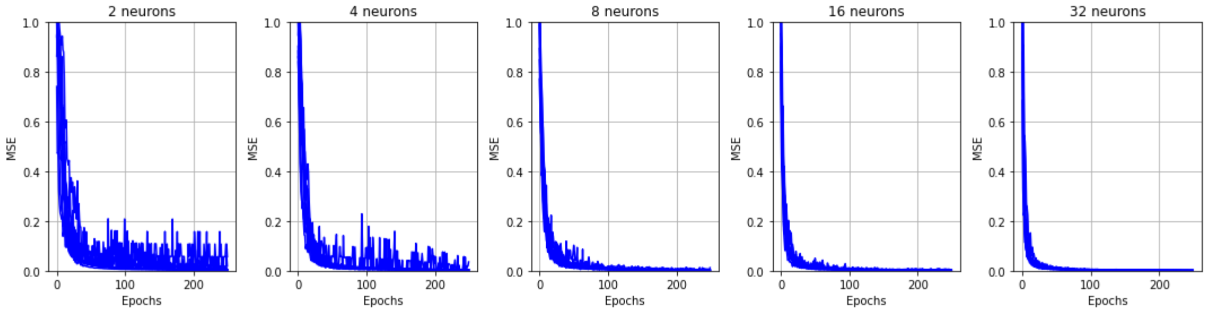
Man vs Woman using only natural voices

For this experiment we will train a model to differentiate the natural voices of men and women.

Exploring number of epochs

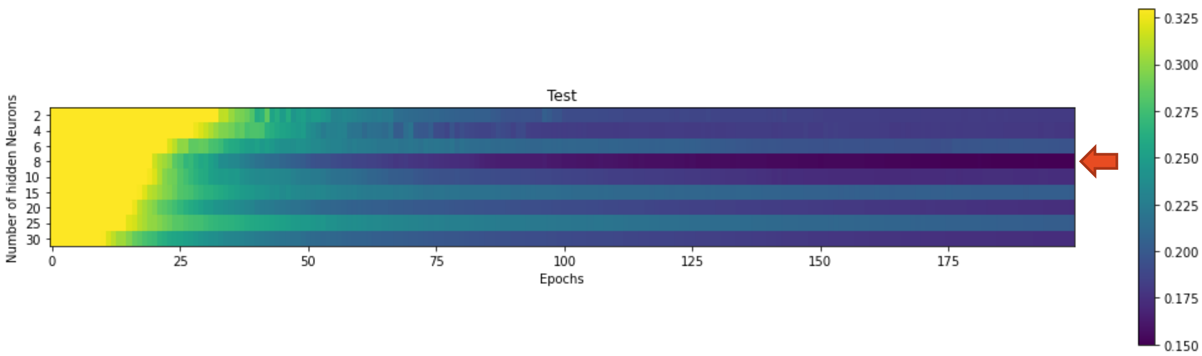
We tested up to 250 epochs.

We see that generally when we hit 200 epochs the MSE becomes stable and low from 8 neurons onwards, so we chose this value of epochs.



Exploring number of hidden neurons

The important part of the graph below is the Test and not the Training part that has been cropped out. By looking at the graph we can see that at 200 epochs 8 hidden neurons gives the best result. Thus, we chose this value for the final model.



Results

Parameters for the final model:

Hidden neurons	Number of epochs	Learning rate	Momentum	N_Inits / N_Tests	K
8	200	0,001	0.5	10	5

With a network of 13 neurons in input, 8 neurons in the hidden layer and 1 neuron in output. With a threshold of 0, we obtain the following final performances.

	M	**F**
Accuracy:	0.972	
Precision:	1	0.947
Recall:	0.944	1
F-score:	0.971	0.972

```
MSE training:  0.0037702041276576866
MSE test:  0.1280555993415202
Confusion matrix:
[[34.  2.]
 [ 0. 36.]]
```

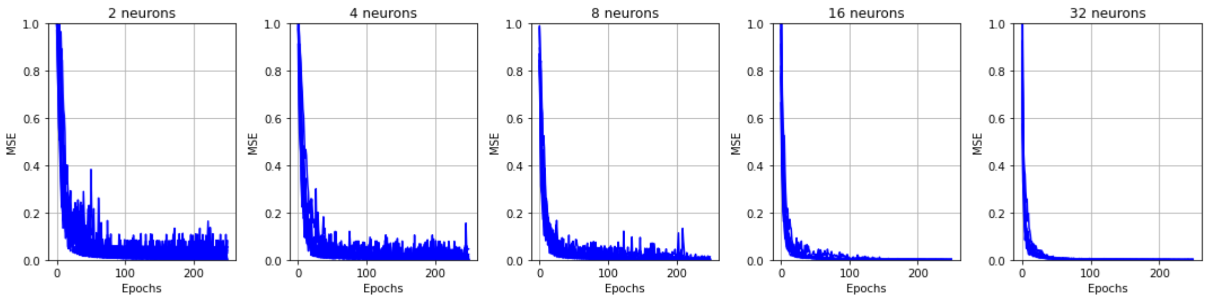
Man vs Woman using both natural and synthetic voices

For this experiment we will train a model to differentiate the voices of men and women, be it natural or synthetic.

Exploring number of epochs

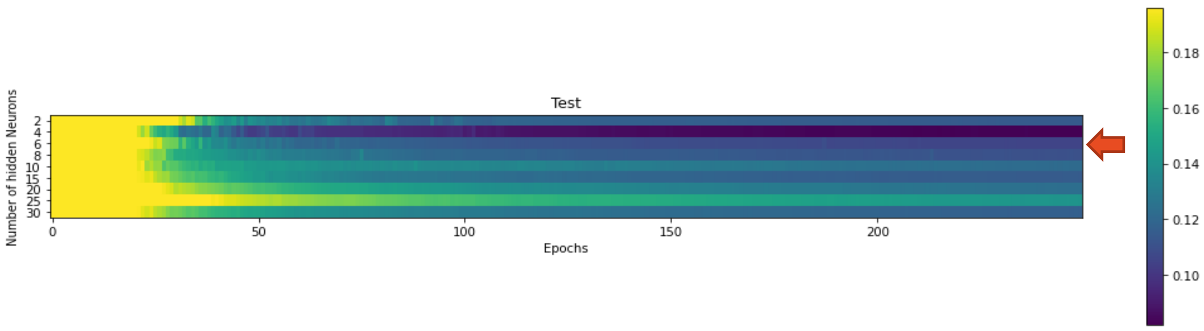
We tested up to 250 epochs.

While with 16 hidden neurons and above the MSE is stable from 150 epochs on we saw that with 8 hidden neurons there is a spike at 200 epochs. Thus, we decided to choose 250 epochs for the final value to be safe.



Exploring number of hidden neurons

By looking at the graph below we saw that, generally, the best results came with 4 hidden neurons.



Results

Parameters for the final model:

Hidden neurons	Number of epochs	Learning rate	Momentum	N_Inits / N_Tests	K
4	250	0,001	0.5	10	5

With a network of 13 neurons in input, 4 neurons in the hidden layer and 1 neuron in output. With a threshold of 0, we obtain the following final performances.

	M	F
Accuracy:	0.979	
Precision:	0.985	0.972
Recall:	0.972	0.986
F-score:	0.978	0.978

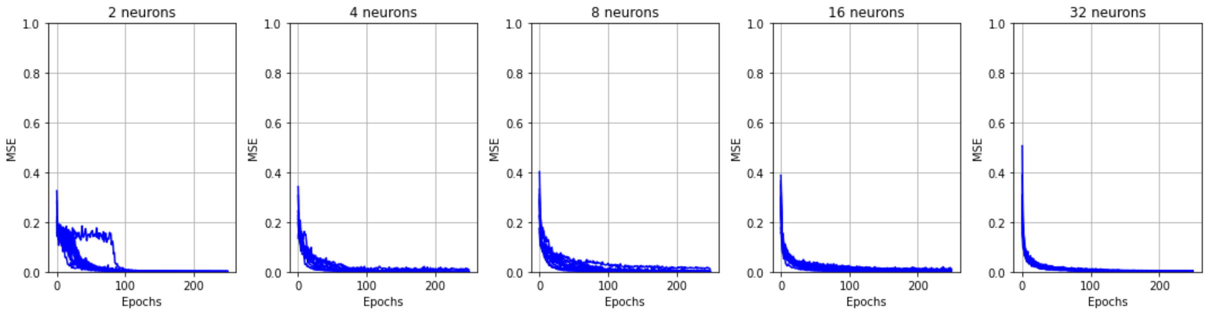
```
MSE training:  0.01074182720952276
MSE test:  0.07967828307281768
Confusion matrix:
[[70.  2.]
 [ 1. 71.]]
```

Man vs. Woman vs. Children using only the natural voices

For this experiment we will train a model to differentiate the voices of men and women and children, using only their natural voices.

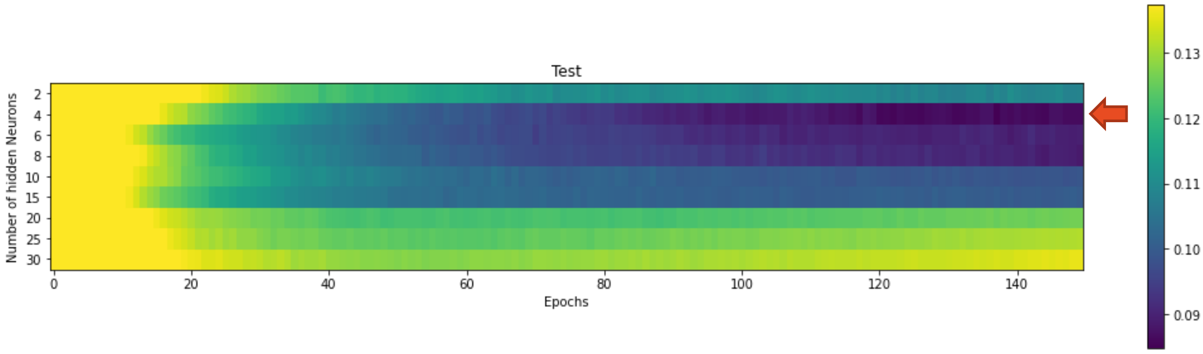
Exploring number of epochs

In this case we see that generally the MSE stabilises at 100 epochs except with 8 neurons where it stabilises at 150 epochs. We decided to use 150 epochs.



Exploring number of hidden neurons

The result with 4, 6 and 8 hidden neurons are good but the results with 4 hidden neurons are slightly better. We will choose 4 hidden neurons for the final model



Results

- Parameters for the final model:

Hidden neurons	Number of epochs	Learning rate	Momentum	N_Inits / N_Tests	K
4	150	0,001	0.5	10	5

With a network of 13 neurons in input, 8 neurons in the hidden layer and 3 neurons in output. With a threshold of 0.55, we obtain the following final performances. It's interesting to note that the difference between the voices of children and women is easier to miss.

	M	F	C
Accuracy:		0.978	
Precision:	0.942	0.634	0.897
Recall:	0.942	0.702	0.864
F-score:	0.942	0.666	0.880

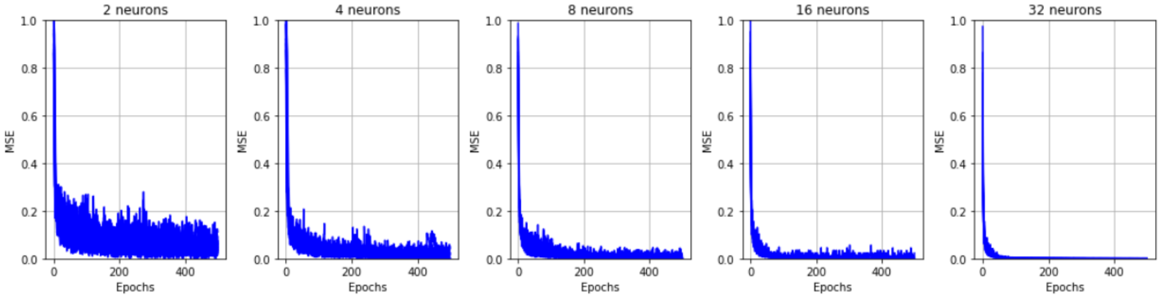
```
MSE training:  0.025517095358083712
MSE test:  0.08199207669434756
Confusion matrix:
[[33.  2.  0.]
 [ 0. 26. 11.]
 [ 2. 13. 96.]]
```

Design a final experiment of your choice

We chose to test all natural voice types against all artificial ones. We thought it would be an interesting experiment that could also maybe tell us something on the quality of synthetised voice.

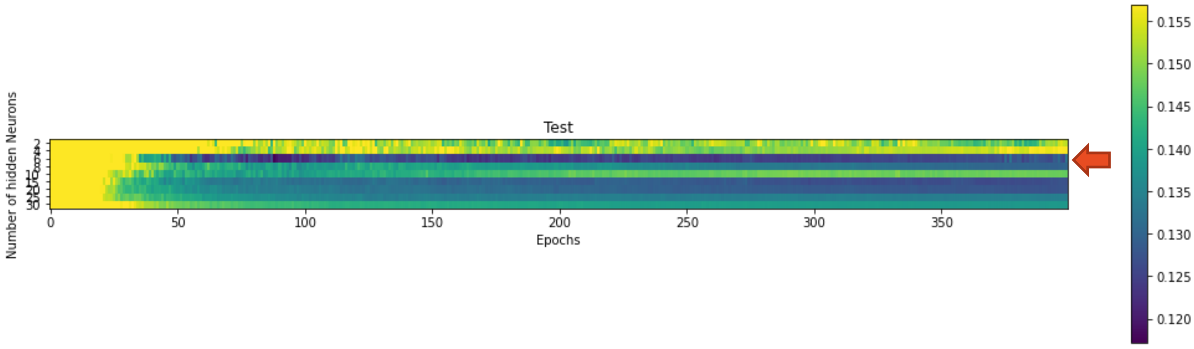
Exploring number of epochs

We clearly had some problems here. The MSE was unstable up to 450 epochs in most cases except the 32 hidden neurons one.



Exploring number of hidden neurons

We didn't have a clear best result in this case. We can see that 6, 8, 15 and 20 hidden neurons seem to give good results. We decided to minimise the number of hidden neuron and to use 6. It's also the result that's best at a lower number of epochs.



Results

• Hidden neurons	Number of epochs	Learning rate	Momentum	N_Inits / N_Tests	K
6	400	0,001	0.5	10	5

With a network of 13 neurons in input, 8 neurons in the hidden layer and 1 neuron in output. With a threshold of 0, we obtain the following final performances.

	N	S
Accuracy:	0.955	
Precision:	0.971	0.940
Recall:	0.939	0.972
F-score:	0.955	0.956

```
MSE training:  0.01252448841338274
MSE test:  0.14455165969837552
Confusion matrix:
[[169.  11.]
 [  5. 175.]]
```