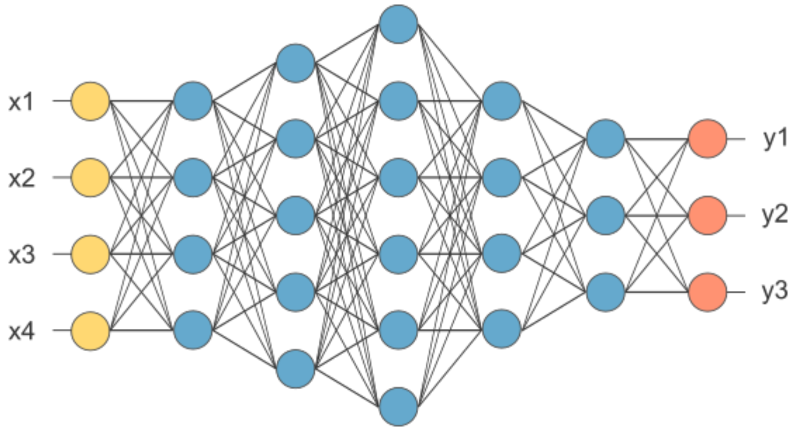
Universidade Federal
do Rio de Janeiro

Escola Politécnica



Weightless Neural Network for Hand-Written Digit Classification

Guillaume Jeusel

Palestra NCG013 Priscila Lima

June 17, 2017

1 Introduction

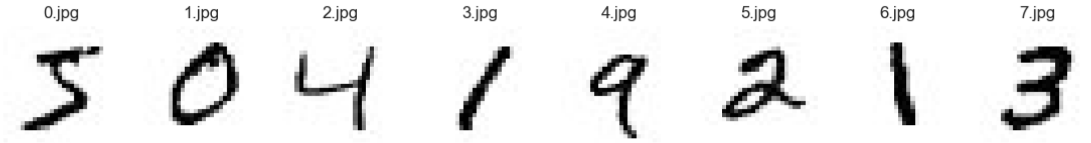
The purpose of this work is to study WiSARD's performances for handwritten digits classification.

1.1 Dataset Presentation

The handwritten digits dataset used is provided by MNIST. It has a training set of 60,000 examples, and a test set of 10,000 examples. More informations about the construction of this dataset can be obtained directly on the website.

These files are not in any standard image format. Consequently I converted them into jpgs using the following python script : mnist-to-jpg.py. It necessitates the installation TensorFlow machine-learning python library from google.

Finally, using the Python Imaging Library (PIL), I was able to get a look into the digits :



Many methods have been tested with this training set and test set. A lot of them can be found with the corresponding paper in the MNIST website. Here are a few of them:

Classifier	Accuracy
linear classifier (1-layer NN)	0.88
K-nearest-neighbors, Euclidean (L2)	0.9691
SVM, Gaussian Kernel	0.986
2-layer NN, 300 hidden units, mean square error	0.953

1.2 Parameters of interest

We are going to limit the number of digits for training and testing sets in order to ease the study.

- Number of figures used for **training** : 900
- Number of figures used for **testing** : 100

Another remark that can be done regarding the dataset is that digits are **28x28 pixels** size and represented in **Luminance space**. We will need to binarize the digits using a **threshold** interval [0,256].

The following algorithm is then utilized to binarize each pixel value :

```
function BINARIZE(threshold, pixel_value)
    if $pixel\_value \leq threshold$ then
        return 0
    else
        return 1
    end if
end function
```

Additionally, the WiSARD parameters that will be studied are :

- **num_bits_addr** : number of bits used to build the memories
- **bleaching** : whether to activate bleaching technique or not

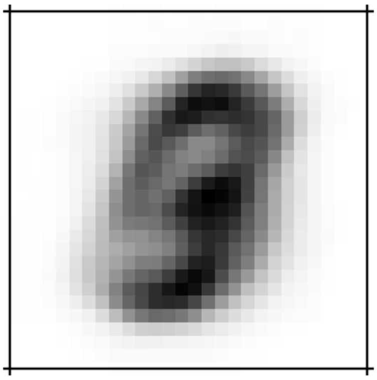
2 Results

The code can be found at `https://github.com/gjeusel/NCG013_redes_neurais_sem_peso`.

2.1 Dataset Analysis

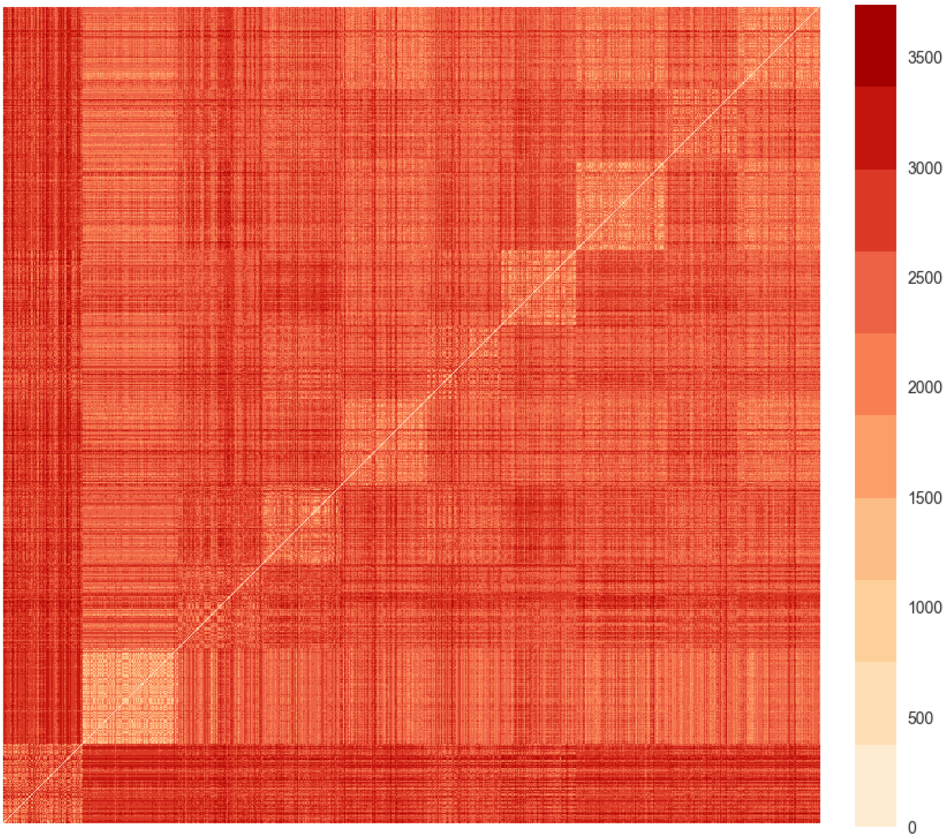
In this section, some interesting results about the dataset itself will be discussed.

- **Mean value per pixel** for the training set in Luminance space:

  This show us that the corners are likely to influence the classification results. We could imagine a pre-processing step involving the cutting of this blank pixels.

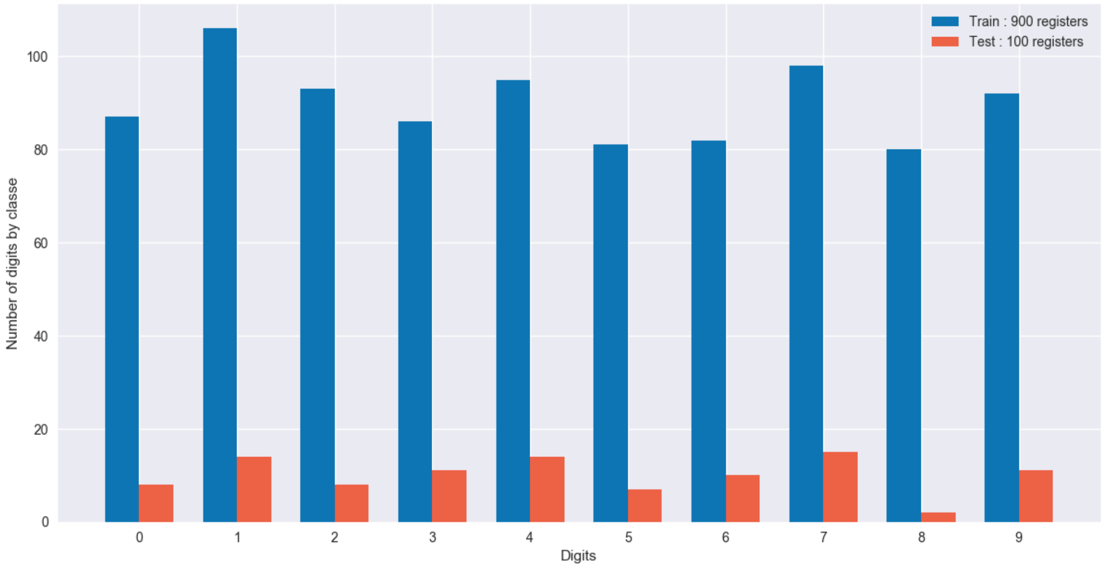
- **Distance Matrix** using the euclidean norm L2 metric. Only the training set will be considered. Each register is previously **sorted** according to the digit it represents. Consequently, we can distinguish 10 clusters formed by the digits [0,..,9].



Interpretation :
- the digit 0 is well distant from the others digits.
- the digit 1 seems to shortly differs across handwritten styles.
- the digit 2 appears to be difficult to identify because distances inside this cluster are akin to distances with others clusters.

The lecture of this matrix gives insights about problems that are likely to occure at the time of classifying this digits.

- **Bar plot** of classes. This gives the number of registers per classes for training set and test set.



Interpretation : It seems that the training set is more or less equilibrated, but we can notice a lack of the digit 8 in the testing set.

2.2 WiSARD

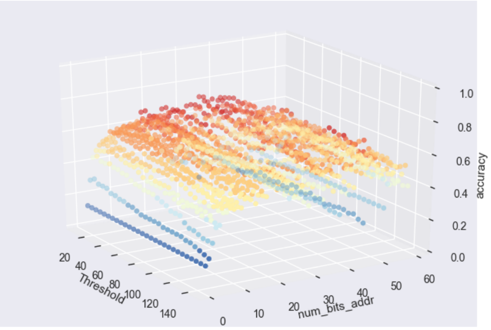
In this section, we will present the results obtained using the WiSARD classifier, by varying the threshold from 10 to 150 by step of 5, and the num_bits_addr from 1 to 64 by step of 1.

Basically, the algorithm used was :

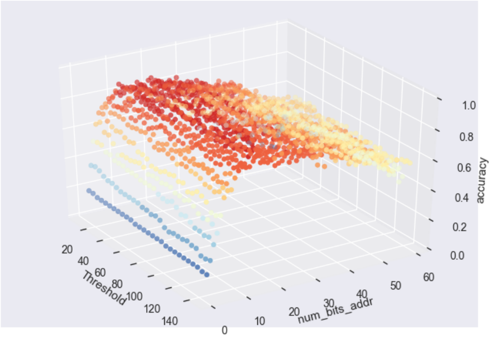
```
for t in range(10,150,5) do
    for n in range(1,64) do
        WiSARD(threshold=t, num_bits_addr=n)
    end for
end for
```

Each **accuracy** obtained is then stored. The results are given in the form of scatter plots with a color scheme to ease the reading of results with high accuracy.
- x axe : num_bits_addr
- y axe : threshold
- z axe : accuracy

- Scatter plot **without** bleaching :



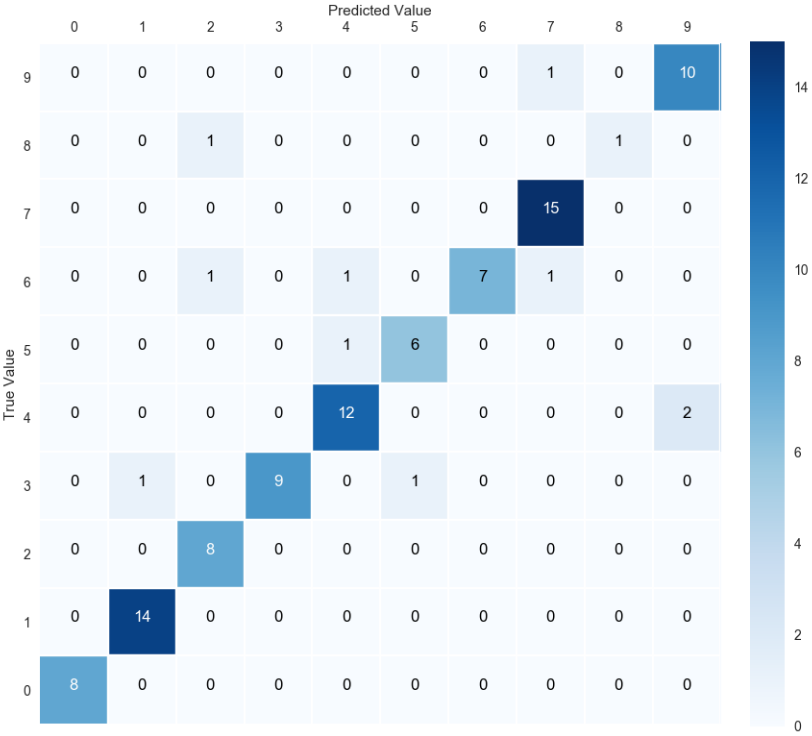
- Scatter plot **with** bleaching :



Following, a table that resume the best results obtained :

Classifier	Threshold	num_bits_addr	Accuracy
WiSARD without bleaching	35	38	0.75
WiSARD with bleaching	45	27	0.90
NearestNeighbors	15	None	0.86
MultiLayerPerceptron (layers=(100))	50	None	0.93

The confusion matrix obtained for WiSARD with :
- bleaching
- threshold = 45
- num_bits_addr = 27



	0	1	2	3	4	5	6	7	8	9
9	0	0	0	0	0	0	0	1	0	10
8	0	0	1	0	0	0	0	0	1	0
7	0	0	0	0	0	0	0	15	0	0
6	0	0	1	0	1	0	7	1	0	0
5	0	0	0	0	1	6	0	0	0	0
4	0	0	0	0	12	0	0	0	0	2
3	0	1	0	9	0	1	0	0	0	0
2	0	0	8	0	0	0	0	0	0	0
1	0	14	0	0	0	0	0	0	0	0
0	8	0	0	0	0	0	0	0	0	0