

Multi-layer Perceptrons (MLP)



LE MÉE Aurel
SAILLY Gregory

Table of contents

01

Perceptrons and MLPs

02

Applications of MLPs

03

Simple use of MLPs

04

Pyrat Predictions



01

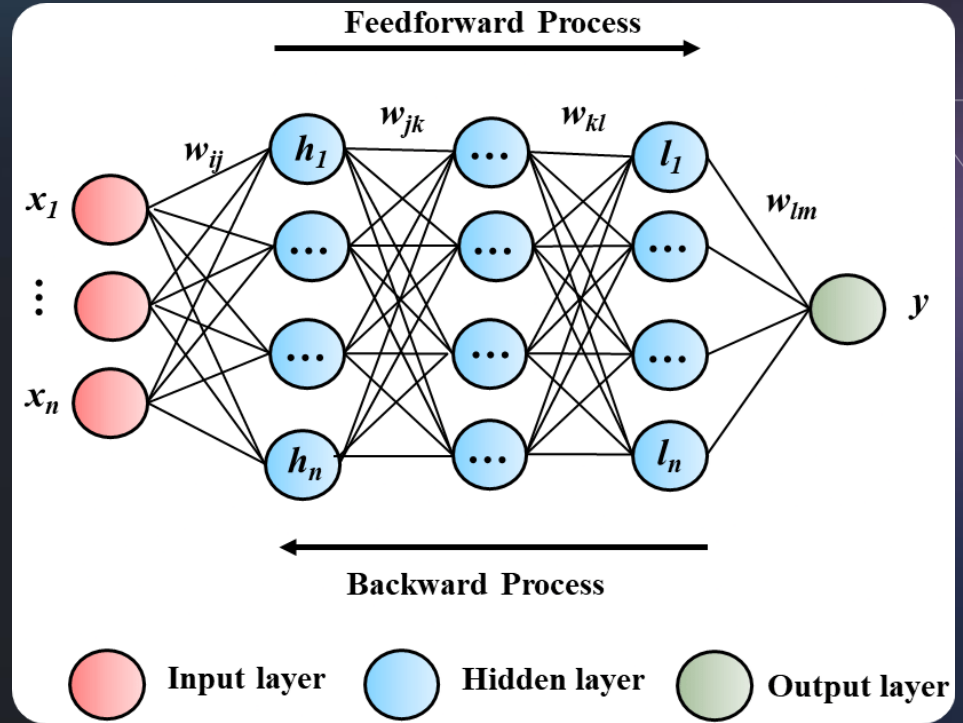
Perceptrons and MLPs

Perceptrons and MLPs

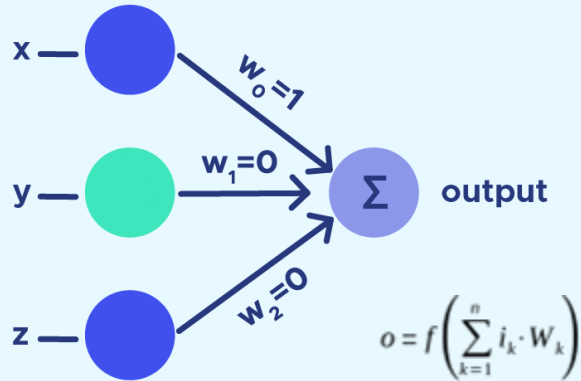
For every MLPs :

- Input layers
- Multiple hidden layers
- Output layers

weights are set to the vertices



Perceptrons and MLPs

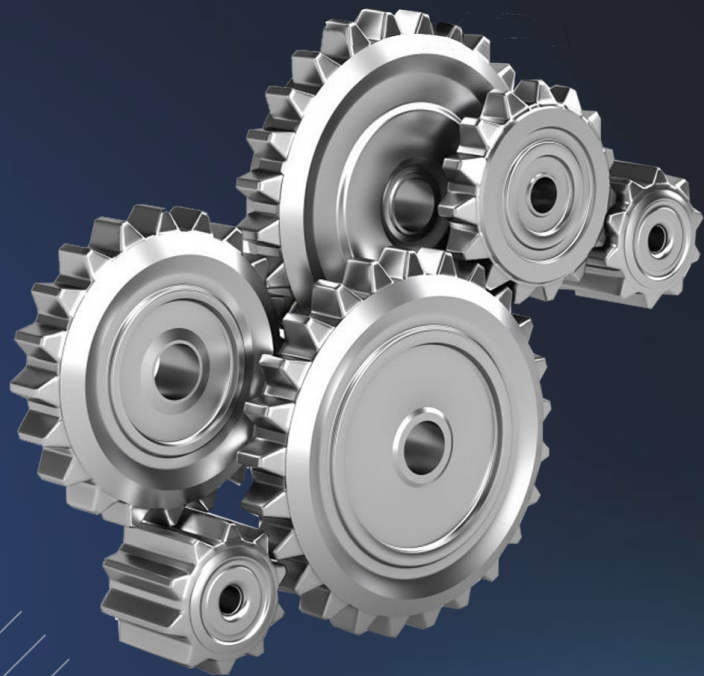


Perceptrons

algorithm for supervised learning of binary classifiers.

It uses an activation function to know their importance.

These weights can be modified with the gradient descent.

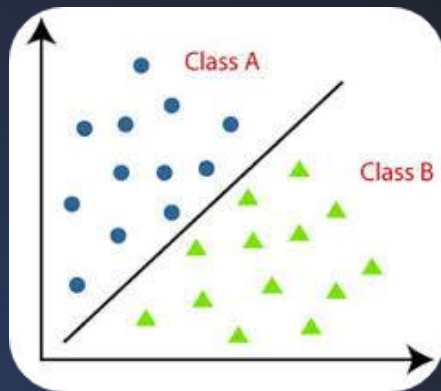


02

Applications of MLPs



Image Classification



Financial Forecasting



Natural Language Processing (NLP)





03

Simple use of MLPs



Parameters



23 parameters with sklearn function



Focus on 4

	Strengths	Risks
Maximum number of iterations	<ul style="list-style-type: none">• Training duration control• Avoid overfitting	<ul style="list-style-type: none">• May reduce model accuracy• May not converge to an optimal solution
Number of hidden layers	<ul style="list-style-type: none">• ↗ Capacity for learning complex data representations• Capture non-linear relationships	<ul style="list-style-type: none">• Overfitting• ↗ Time
Number of neurons per layer	<ul style="list-style-type: none">• Learns easily complex representations• Enhance complex ability to capture complex patterns	<ul style="list-style-type: none">• Overfitting• ↗ Time
Activation function	<ul style="list-style-type: none">• Introduce nonlinearities	<ul style="list-style-type: none">• Can cause troubles with gradients



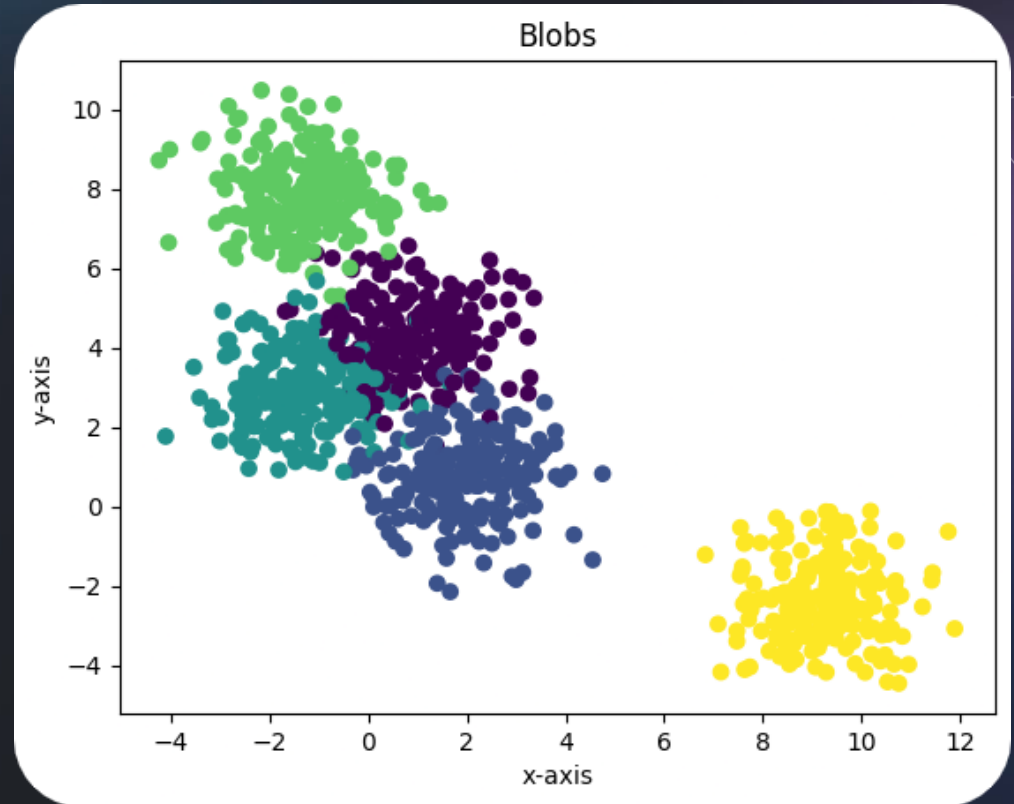
Training on blobs

Blobs Characteristics :

- 1000 samples
- 5 centers

MLP Parameters trained :

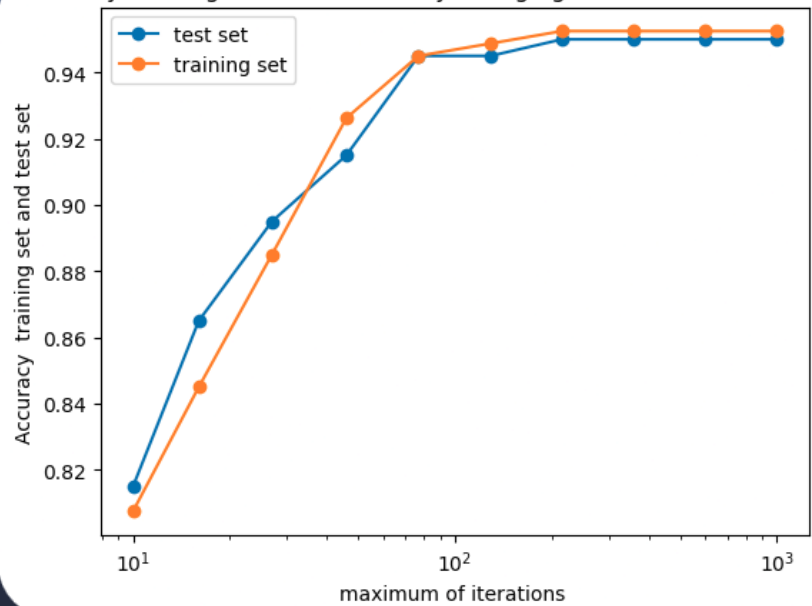
- Activation functions :
 - Relu
 - Logistic(Sigmoid)
 - Tanh
- Max iterations
- Neurons
- Number of hidden Layers



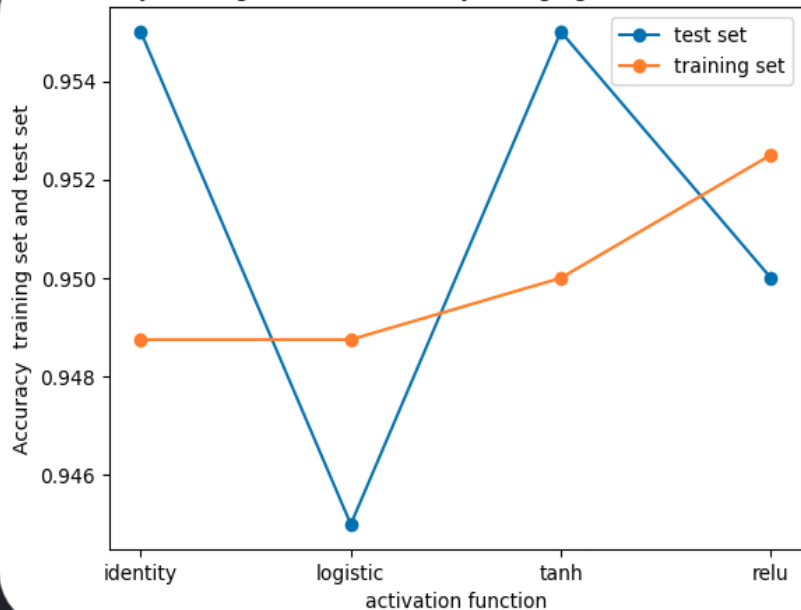


Results

Accuracy training set and test set by changing the maximum of iterations



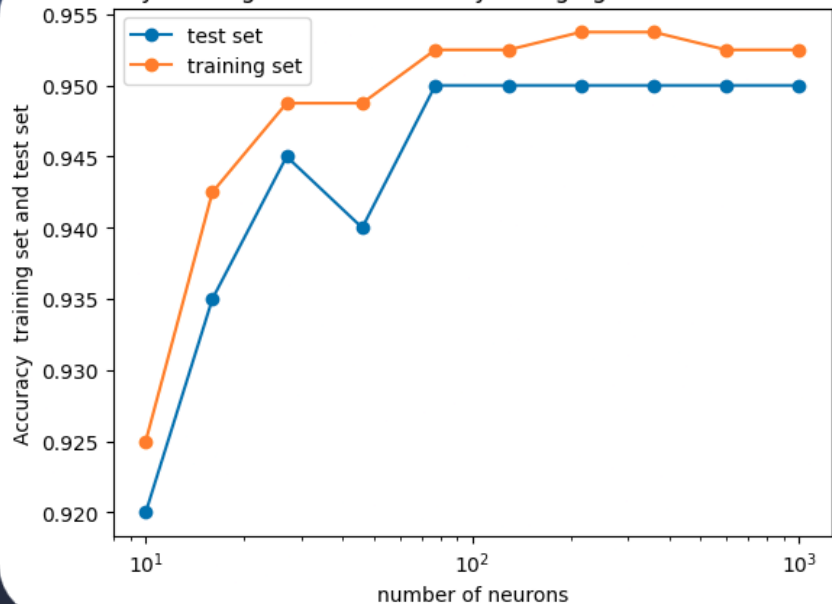
Accuracy training set and test set by changing the activation function



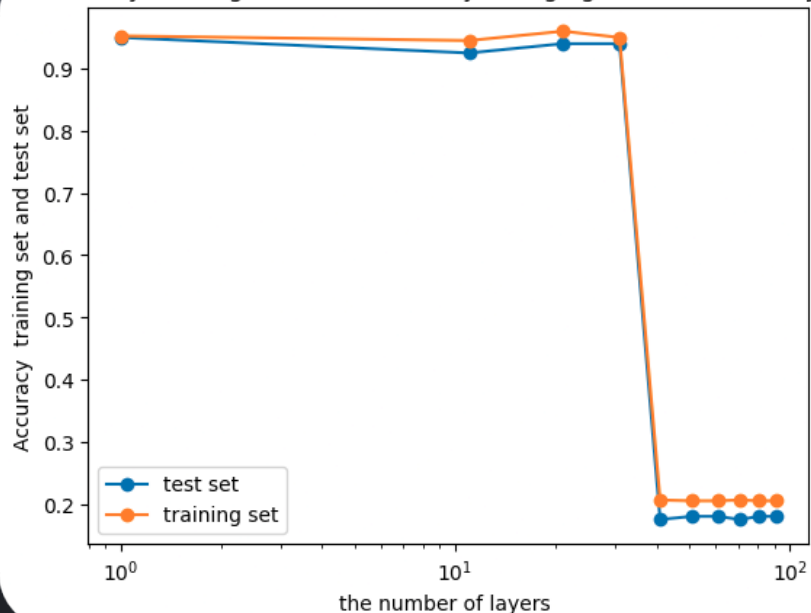


Results

Accuracy training set and test set by changing the number of neurons



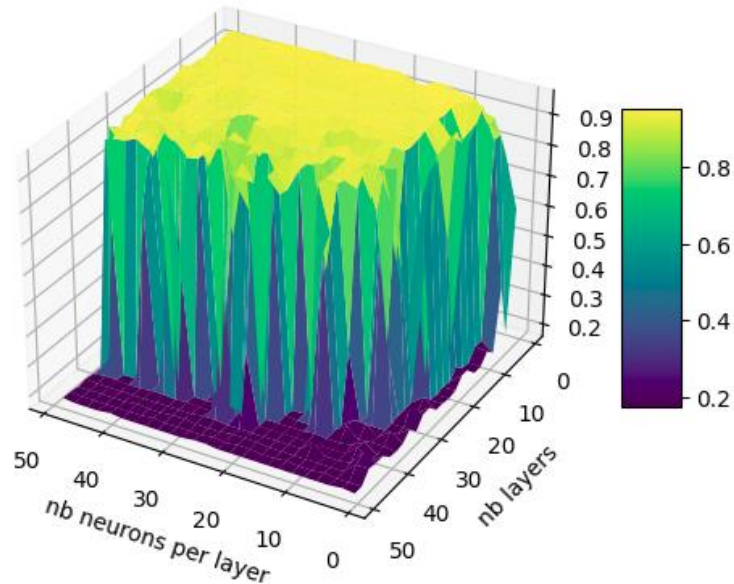
Accuracy training set and test set by changing the number of layers





Results

Accuracy of the classifier on the test set by changing the number of layers and neurons per layer





03

Pyrat Predictions



1th Training

Data : 500 games in a 5*7 grid with 4 cheeses

MLP Parameters : (Bo81 combinations)

- Max iterations = 300
- Activation function = relu
- Number of hidden layers = 3
- Number of neurons = 150

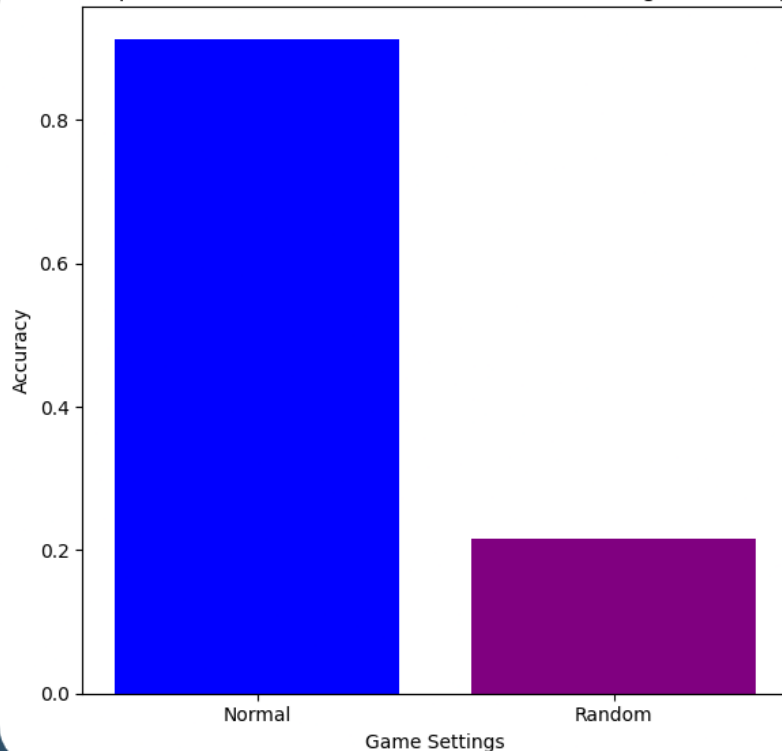
Normal game settings :

- 5*7 grid
- 4 cheeses

Random game settings :

- ?*? grid
- ? cheeses

Comparison of Accuracies on Normal and Random game settings





2nd Training

Data : 117 files of 50 games each with different game settings for each file

MLP Parameters : (Bo81 combinations)

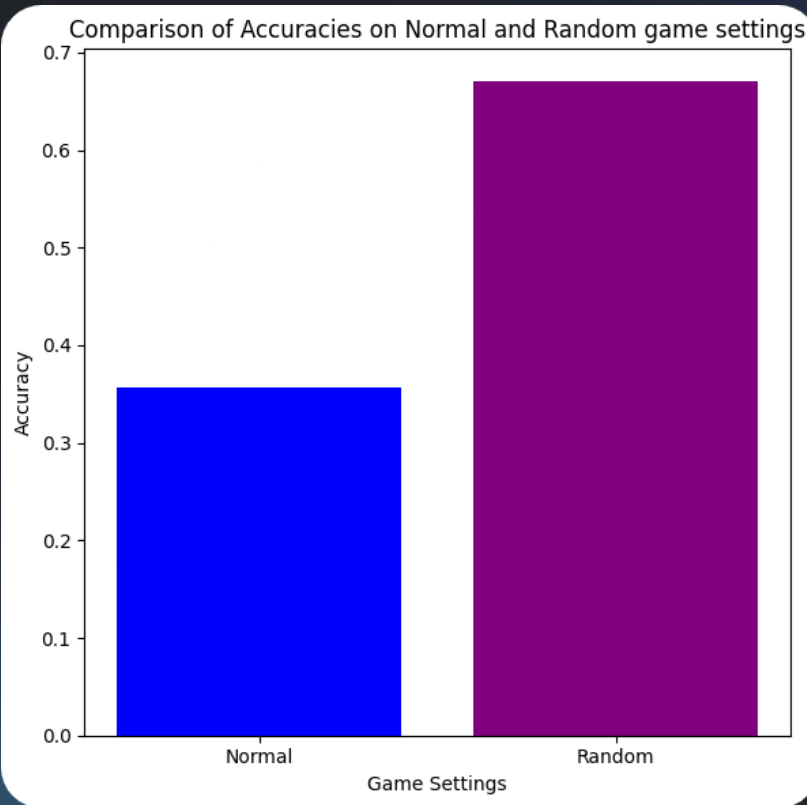
- Max iterations = 300
- Activation function = relu
- Number of hidden layers = 5
- Number of neurons = 500

Normal game settings :

- 5*7 grid
- 4 cheeses

Random game settings :

- ?*? grid
- ? cheeses





2nd Training

Data : 117 files of 50 games each with different game settings for each file

MLP Parameters : (Bo81 combinations)

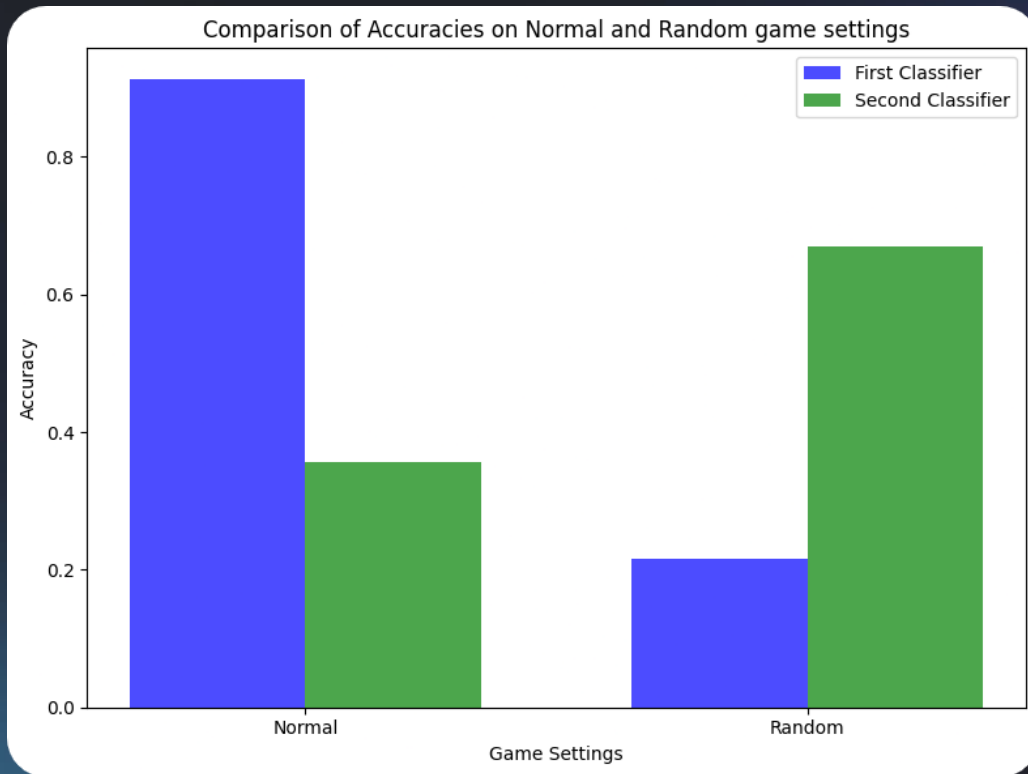
- Max iterations = 300
- Activation function = relu
- Number of hidden layers = 5
- Number of neurons = 500

Normal game settings :

- 5*7 grid
- 4 cheeses

Random game settings :

- ?*? grid
- ? cheeses





Potential explanations to low accuracies

35	19	25	6
13	22	16	53
4	3	7	10
9	8	1	3

0	0	0	0	0	0
0	35	19	25	6	0
0	13	22	16	53	0
0	4	3	7	10	0
0	9	8	1	3	0
0	0	0	0	0	0

Zero-padding :

- 5x7 matrix placed into a 40x40 etc..
- Small shift of data when different parity.



Non-optimal parameters :

- Computational time too high



Lack of exhaustive training data

THANK YOU !

Sources :



- Sklearn : https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html



WIKIPEDIA
The Free Encyclopedia

- Wikipedia : https://en.wikipedia.org/wiki/Multilayer_perceptron



- DataCamp : <https://www.datacamp.com/tutorial/multilayer-perceptrons-in-machine-learning>