

UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II  
Dipartimento di Ingegneria Elettrica e delle Tecnologie  
dell'Informazione



**Neuro BackPropagation Lab**

Giuliano Aiello

2025



# Contents

<b>I</b>	<b>First part</b>	<b>1</b>
<b>1</b>	<b>Prolusion</b>	<b>3</b>
1.1	Goal . . . . .	3
1.2	Software Stack . . . . .	4
1.3	Project Structure . . . . .	5
	<b>Acronyms</b>	<b>9</b>



# Part I

## First part



# Chapter 1

## Prolusion

### 1.1 Goal

This report provides a comprehensive overview of a Python project whose goal is to develop and compare different adaptive backpropagation techniques involved in a machine learning process, as Rprop (Resilient backpropagation).

The project follows the “Empirical evaluation of the improved Rprop learning algorithms” article by Christian Igel and Michel Hüsken (2001).

## 1.2 Software Stack

- Python 3.9.6
- PyTorch 2.6.0

The project is equipped with a `requirements.txt` file which allows for seamless installation of dependencies, by executing `pip install -r requirements.txt`.



## 1.3 Project Structure

```
neuro-backprop-lab/
├── model/
├── tester/
│   ├── tester.py
│   └── trained_model.pt
├── trainer/
│   ├── irpropplus/
│   ├── rpropplus/
│   └── trainer.py
├── utils/
├── test_model.py
└── train_model.py
```

- **model** includes the neural network model architecture.
- **tester** handles the testing flow of the ready-to-use **trained\_model.pt**.
- **trainer** handles the examined backpropagation techniques and the training flow of the model, saving it as **trained\_model.pt**.
- **utils** offers utility functions designed to support the root project scripts.







# Acronyms

**Rprop** Resilient backpropagation 3