

# Лабораторна робота №5

## з алгоритмів машинного навчання.

### Варіант №1

Горбунов Даниїл

7 червня 2020 р.

## 1 Вступ.

У даній роботі наведені результати використання нейронної мережі прямого зв'язку (FFNN) в задачі бінарної класифікації даних. Покажемо структуру даних, коротко опишемо архітектуру побудованої нейронної мережі та поговоримо про отримані результати. Програмна реалізація була виконана за допомогою Python.

## 2 Геометрична інтерпретація даних.

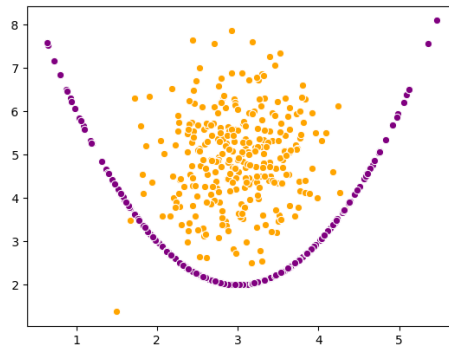


Рис. 1: Діаграма розсіювання вхідних даних.

Маємо матрицю  $X$ , яка складається з двох регресорів та 600 спостережень. Елементи одного класу утворюють параболічну форму на діаграмі розсіювання, а іншого є гаусовими величинами на площині. У цьому легко переконатися, якщо застосувати критерій Шапіро-Вілکا та подивитися на розподіли відповідних елементів  $X$ .

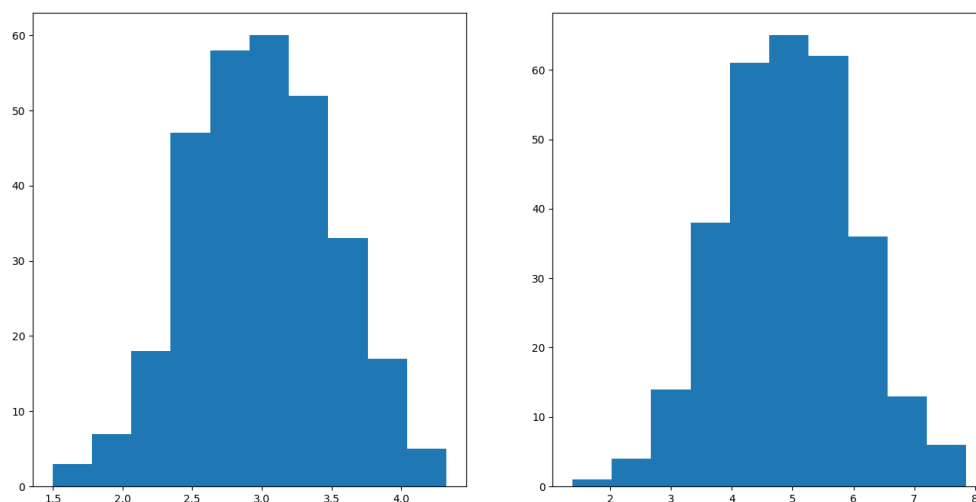


Рис. 2: Гістограми відповідних координат, які відносяться до другого класу.

Отримані статистики та ймовірнісні значення за вказаних критерієм:  
Для елементів першого регресора:

Shapiro-Wilk statistic: 0.9973, p-value: 0.9074

Для елементів другого регресора:

Shapiro-Wilk statistic: 0.9976, p-value: 0.9417

Очевидно, що дані не є лінійно сепарабельними, але це не буде заважати в цілому. Насправді тут можна спокійно використати модель опорних векторів з радіальним ядром та коефіцієнтом  $C = 1$ , отримавши достатньо хороші метрики на тренувальних даних:

	precision	recall	f1-score	support
0	0.97	1.00	0.99	300
1	1.00	0.97	0.99	300
accuracy			0.9867	600
macro avg	0.99	0.99	0.99	600
weighted avg	0.99	0.99	0.99	600

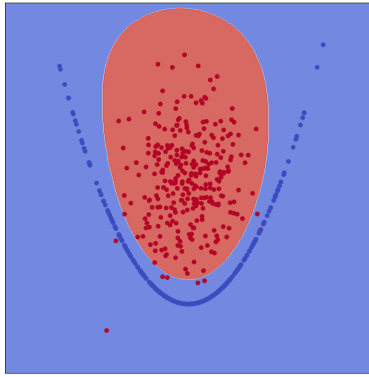


Рис. 3: Діаграма розсіювання даних та область класифікації, використовуючи машину опорних векторів з нелінійним ядром.

### 3 Архітектура, інші параметри щільної нейронної мережі.

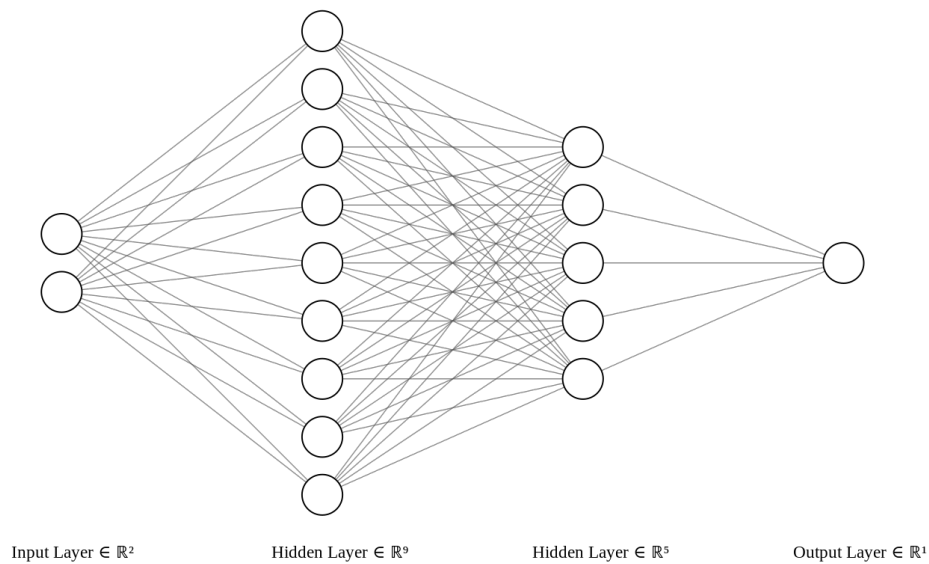


Рис. 4: Візуалізація будови нейронної мережі.

Нейронна мережа, яка побудована у рамках поставленої задачі, досить проста. З метою зосередитися лише на основному, наведемо значення параметрів та архітектуру мережі у вигляді списку.

- Кількість шарів  $L = 3$ ; кількість нейронів на кожному шарі: 2, 9, 5, 1.
- Кількість ітерацій  $T = 8350$ . Якщо збільшити кількість епох більше заданої, маємо перенавчання моделі.
- Параметр навчання  $\alpha = 0.01$ . При більших значеннях, ентропія не збігається до менших значень, а при менших процес навчання не завжди досягає оптимального результату з точки зору витраченого часу та використаної моделі в цілому.
- Алгоритм обчислення вагів: стохастичний градієнтний спуск. Досить простий в сенсі реалізації та можливих модифікацій в подальшому.

## 4 Результати.

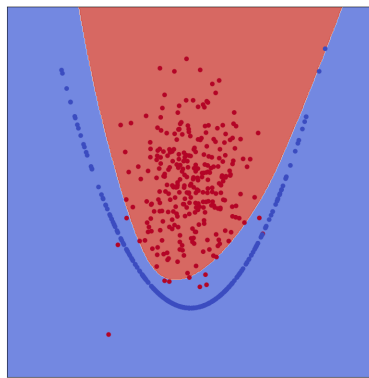


Рис. 5: Діаграма розсіювання даних та область класифікації.

	precision	recall	f1-score	support
0	0.96	1.00	0.98	300
1	1.00	0.96	0.98	300
accuracy			0.98	600
macro avg	0.98	0.98	0.98	600
weighted avg	0.98	0.98	0.98	600

Iteration #8300 Cost: 0.0933

Iteration #8300 RMSE: 0.1414

Видно, область класифікації нейронної мережі менш точна порівняно з тою, що була отримана для моделі опорних векторів. На останній ітерації, ентропія та середньоквадратична похибка досить малі, що свідчить про те, що кількість ітерацій добре підібрана та архітектура мережі, що була наведена вище, є доречною.

## 5 Висновки.

Модель нейронної мережі прямого зв'язку дає непогані результати на тих даних, що маємо. Переконалися в тому, що метод градієнтного спуску досить повільний з більшою кількістю ітерацій.

Порівняли класифікатори на базі SVM та FFNN. Видно, що на вибірках малого обсягу метод опорних векторів також дає змогу побудувати якісну модель.