TRƯỜNG ĐẠI HỌC THỦY LỢI

**KHOA CÔNG NGHỆ THÔNG TIN**

A blue and white logo

Description automatically generated with low confidence

**BÀI TẬP LỚN**

HỌC PHẦN: HỌC MÁY

**ĐỀ TÀI: DỰ ĐOÁN THỜI TIẾT**

**Phần 1: Lý thuyết**

Perceptron là 1 thuật toánClassification cho trường hợp đơn giản nhất: chỉ có 2 class (Bài toán chỉ với 2 class được gọi là binary classification) và cũng chỉ hoạt động được trong một trường hợp rất cụ thể.

1. Input: Là 1 tập DL đã được gán nhãn
2. Output: Là 1 vecto dự đoán
3. Method:

* Phương trình đường boundary:

*f*w(x) = w1.x1 + w2.x2 + … + wd.xd + w0

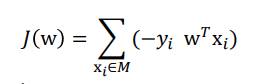
= wT.x = 0

+ w = (w0, w1,…, wd) : vector hệ số

+ w0 : số hạng tự do

+ x = (x0, x1,…, xd): vector đặc trưng

* Hàm mất mát:



* Nghiệm của bài toán w: Tìm w sao cho hàm mất mát đạt GTNN = 0 (không có điểm dữ liệu nào bị phân lớp sai)
* Với mỗi điểm phân lớp đúng -> kết thúc thuật toán
* Với mỗi điểm phân lớp sai nghiệm w được cập nhật lại:



1. Điều kiện dừng

* Tất cả các mẫu của nút thuộc cùng 1 lớp
* Không còn thuộc tính nào để phân chia
* Không còn mẫu nào

**Phần 2: Ứng dụng trong thực tế**

***1. Mô tả bài toán***

* **Tên bài toán**: Dự đoán thời tiết.
* **Mục đích**: Dự đoán tình trạng thời tiết khi nhập các thông số thời tiết như precipitation, temp, humidity, wind.
* **Input**: Thông số thời tiết (precipitation, temp, humidity, wind).
* **Output**: Thời tiết (weather: sun, rain, driz, snow, fog).

Bài toán này nhằm dự đoán các kiểu thời tiết (sun, rain, driz, snow, fog) dựa trên thông tin về lượng mưa (precipitation), nhiệt độ (temp), độ ẩm (humidity), và sức gió (wind).

***2. Mô tả tập dữ liệu của bài toán***

* Dữ liệu:
* Gồm những chiều thông tin: precipitation, temp, humidity, wind, weather.
* Các nhãn lớp gồm: sun, rain, driz, snow, fog.
* Mô tả ma trận dữ liệu (X), nhãn lớp (Y).
  + Ma trận dữ liệu (X) gồm các cột:
    - Precipitation: Thể hiện lượng mưa.
    - Temp: Thể hiện nhiệt độ.
    - Humidity: Thể hiện độ ẩm.
    - Wind: Thể hiện sức gió.
  + Nhãn lớp (Y) gồm:
    - Weather: Thể hiện tình trạng thời tiết (sun, rain, driz, snow, fog).
* Chia tập dữ liệu thành 2 phần với tỷ lệ 3:7 gồm:
* 439 dòng đầu: kiểm tra mô hình.
* 1022 dòng cuối: huấn luyện mô hình.

***3. Viết ứng dụng***

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

from sklearn.linear\_model import Perceptron

from sklearn.metrics import ConfusionMatrixDisplay, confusion\_matrix, precision\_score, recall\_score, f1\_score

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import LabelEncoder

*# Đọc tệp dữ liệu*

data = pd.read\_csv(r'C:\Users\LENOVO\Downloads\Học máy\seattle-weather.csv', *index\_col*=0)

*# Sử dụng các cột: precipitation, temp\_max, temp\_min, wind*

features = ["precipitation", "temp\_max", "temp\_min", "wind"]

X = data[features]

*# Sử dụng LabelEncoder để mã hóa cột 'weather'*

le = LabelEncoder()

y = le.fit\_transform(data["weather"])

*# Chia dữ liệu thành tập huấn luyện và kiểm tra*

X\_train, X\_valid, y\_train, y\_valid = train\_test\_split(X, y, *train\_size*=0.7, *test\_size*=0.3, *shuffle*=False)

*# Khởi tạo và huấn luyện mô hình Perceptron*

clf = Perceptron()

clf.fit(X\_train, y\_train)

*# Dự đoán trên tập kiểm tra*

y\_pred = clf.predict(X\_valid)

*# Hiển thị kết quả thực tế và dự đoán dưới dạng số*

print("Thực tế \t Dự đoán")

for true, pred in zip(y\_valid, y\_pred):

    print(true, "\t\t", pred)

*# Tính toán tỷ lệ dự đoán chính xác*

accuracy = np.mean(y\_valid == y\_pred) \* 100

print(f'Tỷ lệ dự đoán đúng: {accuracy:.2f} %')

*# Tính toán precision, recall và F1-score*

print("Độ đo precision:", precision\_score(y\_valid, y\_pred, *average*='micro'))

print("Độ đo recall:", recall\_score(y\_valid, y\_pred, *average*='micro'))

print("Độ đo F1:", f1\_score(y\_valid, y\_pred, *average*='micro'))

*# Hiển thị ma trận nhầm lẫn (Confusion Matrix)*

cm = confusion\_matrix(y\_valid, y\_pred)

disp = ConfusionMatrixDisplay(*confusion\_matrix*=cm, *display\_labels*=le.classes\_)

disp.plot(*cmap*=plt.cm.Blues)

plt.show()

*# Dự đoán với dữ liệu mới*

new\_input = pd.DataFrame({

    'precipitation': [4.3],

    'temp\_max': [13.9],

    'temp\_min': [10],

    'wind': [2.8]

})

new\_prediction = clf.predict(new\_input)

print("Dự đoán thời tiết cho dữ liệu mới:", new\_prediction)