 Giảng viên: VÕ XUÂN THỂ: voxuanthephd@gmail.com (công việc) tranhailua@yahoo.com (cá nhân)

G?.stt

HỌ TÊN: **MSSV:** **KÝ TÊN:**

TRƯỜNG: HCMUTE

MÔN: Lập trình Python

NGÀY: ..../...../..... (BUỔI HỌC SỐ ?)

Riêng buổi học Phòng máy ghi thêm, SỐ MÁY: PHÒNG MÁY:

==========

Buổi 13_... = Bài tập 3 Phần 2 = GĐ 3 EDA: Phân tích dữ liệu thăm dò

Giai đoạn 3 = Bước 7, 8, 9, 10

1. MỘT SỐ LÝ THUYẾT

CÁC THƯ VIỆN CHÍNH

```
from sklearn import preprocessing # Thư viện tiền xử lý DL (XL ngoại lệ: Isolated)

from sklearn.feature_selection import SelectKBest, chi2
# Nạp hàm Thư viện hỗ trợ Mô hình phân tích dữ liệu thăm dò
```

2. FULL CODES THAM KHẢO

(Chỉ là phần core: sv phát triển & hoàn thiện lên Form = Bài tập ĐAHP theo cá nhân + nên cá nhân hóa thông tin bài làm)

PHẦN BUỔI 4 = TIỀN XỬ LÝ

```
"""
# -*- coding: utf-8 -*-
"""
Created on Fri Mar 10 16:49:20 2023

@author: VOXUAN
"""

# GIAI ĐOẠN 1: NẠP DỮ LIỆU GỐC (PRIMARY INPUT DATA LOAD)

# Bước 1: Nạp các thư viện cần thiết

import numpy as np #Numeric Python: Thư viện về Đại số tuyến tính tính

import pandas as pd #Python Analytic on Data System: For data processing (Thư viện xử lý dữ liệu)

from scipy import stats # thư viện cung cấp các công cụ thống kê [statistics] sub-lib của science python [các công cụ khoa học]
```

```
from sklearn import preprocessing # Thư viện tiền xử lý DL (XL ngoại lệ: Isolated)

from sklearn.feature_selection import SelectKBest, chi2 # Nạp hàm Thư viện hỗ trợ Mô hình
phân tích dữ liệu thăm dò

# Bước 2: Tải tập dữ liệu: Load the data set (Nạp tập dữ liệu)
# ./sttHoTen_weatherAUS.csv
df = pd.read_csv('./weatherAUS.csv')
# Display the shape of the data set (xem lượng dòng & cột dữ liệu của tập DL gốc)
print('Độ lớn của bảng [frame] dữ liệu thời tiết:',df.shape)
#Display data (Hiển thị dữ liệu dạng mảng 5 dòng đầu của tập DL gốc)
print(df[0:5])

# GIAI ĐOẠN 2: TIỀN XỬ LÝ (PRE-PROCESSING)

# Bước 3: Xử lý CỘT dữ liệu NULL quá nhiều OR không có giá trị phân tích
# Checking for null values (Kiểm tra giá trị null = đếm số dòng có dữ liệu ứng từng thuộc#
tính)
print(df.count().sort_values()) #df.count(): đếm số lượng dòng có dữ liệu của df,
.sort_values() sx tăng dần

df =
df.drop(columns=['Sunshine','Evaporation','Cloud3pm','Cloud9am','Location','Date','RISK_MM'],axis=1)
#df = df.drop(columns=['Sunshine','Evaporation','Cloud3pm','Cloud9am','Pressure9am',#
'Pressure3pm','WindDir3pm', 'WindDir9am', 'WindGustDir',#
'WindGustSpeed','Location','Date','RISK_MM'],axis=1)
print(df.shape) # kiểm tra lại số lượng cột & dòng của df sau khi XL NULL cột

# Bước 4: Xử lý DÒNG dữ liệu NULL
# Removing null values (Xóa tất cả các dòng có giá trị null trong tập FRAME dữ liệu.)
df = df.dropna(how='any')
print(df.shape) # kiểm tra lại số lượng cột & dòng của df sau khi XL NULL các dòng DL

# Bước 5: Xử lý loại bỏ các giá trị ngoại lệ (cá biệt): isolated
#kiểm tra tập dữ liệu có bất kỳ ngoại lệ nào không
z = np.abs(stats.zscore(df._get_numeric_data())) # Dò tìm và lấy các giá trị cá biệt trong
tập dữ liệu gốc thông qua điểm z (z_score)
print('MA TRAN Z-SCORE\n')
print(z) # in ra tập (ma trận) các giá trị z-score từ tập dữ liệu gốc
df= df[(z < 3).all(axis=1)] # kiểm tra và chỉ giữ lại trong df các giá trị số liệu tương ứng
với z-score < 3  # {loại các giá trị >= 3} vì các giá trị z-score >=3 tướng ứng với số liệu
quá khác biệt so với các số liệu còn lại ("cá biệt" = "ngoại lệ" = isolated}
print(df.shape) # xác định số dòng & cột dữ liệu sau khu xử lý các giá trị cá biệt

# Bước 6: Thay thế các vị trí giá trị  0 và 1 bởi CÓ (Yes) và KHÔNG (No).
#Thay thế vào vị trí giá trị 1 (Y) và 0 (N) tương ứng cột|biến RainToday và# RainTomorrow
df['RainToday'].replace({'No': 0, 'Yes': 1},inplace = True)
df['RainTomorrow'].replace({'No':0, 'Yes':1},inplace = True)

print(df[0:5])

#Bước 7: Chuẩn hóa (Rời rạc hóa) tập dữ liệu Input dùng ..MaxMin
rr = preprocessing.MinMaxScaler() # xác định thang đo
rr.fit(df)
df = pd.DataFrame(rr.transform(df), index=df.index, columns=df.columns)
df.iloc[4:10]
print(df)
```

PHẦN BUỔI 13 = EDA

```
# GIAI ĐOẠN 3: PHÂN TÍCH DỮ LIỆU THĂM DÒ : EDA [CƠ SỞ = HỌC CÁC MÔN data Science, AI, ML và DeepML,... ]

#Bước 8: Xác định mô hình trích lọc các thuộc tính đặc trưng: EDA
X = df.loc[:,df.columns!='RainTomorrow'] # xác định tập DL Input (X) = All trừ (chú ý !) cột DL đoán đầu ra RainTomorrow
y = df[['RainTomorrow']] # xác định tập DL ra RainTomorrow

selector = SelectKBest(chi2, k=3) # sd các hàm ... trong thư viện sklearn = Mô hình xác định các Thuộc tính quan trọng quyết định việc dự đoàn DL output = trích lọc Đặc trưng = Feature Extraction
selector.fit(X, y) # Áp dụng mô hình trên vào ....
X_new = selector.transform(X) # Chuyên DL Input teho mô hình
print(X_new)
print(y)
########################################
print(X.columns[selector.get_support(indices=True)])

#Bước 9: Xác định mô hình trích lọc các thuộc tính đặc trưng
# XĐ data frame = Chiếu lấy các thuộc tính đặc trưng đã xđ trong B8
df = df[['Humidity3pm','Rainfall','RainToday','RainTomorrow']]

#Bước 10: EDA theo nhu cầu thực tế => input vào các mô hình AI, ML,...
# Đơn giản nhất là lấy 1 thuộc tính đầu vào (Humidity3pm) để XD Mô hình

X = df[['Humidity3pm']]
y = df[['RainTomorrow']]

print(X)
print(y)
############CÁC KQ TRÊN SẼ INPUT VÀO CÁC  MÔ HÌNH AI & ML,...
```