**Bài 13: Analyzing Time Series Data phần 1 - Lập trình AI bằng Python**

Dự đoán tiếp theo trong một chuỗi đầu vào nhất định là một khái niệm quan trọng khác trong học máy. Bài này cung cấp cho bạn lời giải thích chi tiết về phân tích dữ liệu chuỗi thời gian.

**1. Giới thiệu :**

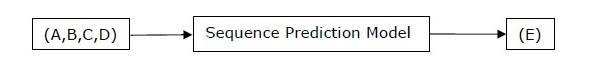
Dữ liệu chuỗi thời gian có nghĩa là dữ liệu nằm trong một chuỗi các khoảng thời gian cụ thể. Nếu chúng ta muốn xây dựng dự đoán trình tự trong học máy, thì chúng ta phải xử lý dữ liệu tuần tự và thời gian. Dữ liệu chuỗi là một bản tóm tắt của dữ liệu tuần tự. Thứ tự dữ liệu là một tính năng quan trọng của dữ liệu tuần tự.

**2. Khái niệm cơ bản về Sequence Analysis hay Time Series Analysis**

Sequence Analysis hay Time Series Analysis là dự đoán tiếp theo trong một chuỗi đầu vào nhất định dựa trên những gì đã quan sát trước đó. Dự đoán có thể là bất cứ điều gì có thể xảy ra tiếp theo: một biểu tượng, một con số, thời tiết ngày hôm sau, thuật ngữ tiếp theo trong bài phát biểu, v.v. Phân tích trình tự có thể rất hữu ích trong các ứng dụng như phân tích thị trường chứng khoán, dự báo thời tiết và đề xuất sản phẩm.

**Ví dụ :**

Hãy xem xét ví dụ sau để hiểu dự đoán chuỗi. Ở đây A, B, C, D là các giá trị đã cho và bạn phải dự đoán giá trị E bằng Mô hình dự đoán trình tự.



**3. Cài đặt các package quan trọng :**

Để phân tích dữ liệu chuỗi thời gian bằng Python, chúng ta cần cài đặt các package sau:

**a. Pandas**

Pandas một thư viện mã nguồn mở được cấp phép BSD cung cấp các công cụ phân tích dữ liệu và hiệu suất cao, dễ sử dụng cấu trúc dữ liệu cho Python. Bạn có thể cài đặt Pandas như sau :

**pip** install pandas

Nếu bạn sử dụng conda thì cài đặt với câu lệnh sau :

conda install -c anaconda pandas

**b. hmmlearn**

Là một thư viện mã nguồn mở được cấp phép BSD bao gồm các thuật toán và mô hình đơn giản để học Mô hình Markov ẩn (HMM) trong Python.Cài đặt hmmlearn với câu lệnh sau :

**pip** install hmmlearn

Nếu bạn sử dụng conda thì cài đặt với câu lệnh sau :

conda install -c omnia hmmlearn

**c. PyStruct**

Là một thư viện dự đoán và học tập có cấu trúc. Các thuật toán học tập được thực hiện trong PyStruct có tên như trường ngẫu nhiên có điều kiện (CRF), Mạng ngẫu nhiên Markov Margin tối đa (M3N) hoặc máy vectơ hỗ trợ cấu trúc. Cài đặt với câu lệnh sau :

**pip** install pystruct

**d. CVXOPT :**

Được sử dụng để tối ưu hóa lồi dựa trên ngôn ngữ lập trình Python. Nó cũng là một gói phần mềm miễn phí

**pip** install cvxopt

anaconda:

conda install -c anaconda cvdoxt

**4. Pandas : Xử lý, cắt và trích xuất thống kê từ dữ liệu chuỗi thời gian**

Pandas : là một công cụ rất hữu ích nếu bạn phải làm việc với dữ liệu chuỗi thời gian. Với sự trợ giúp của Pandas, bạn có thể thực hiện những việc sau

* Tạo phạm vi ngày bằng package pd.date\_range
* Lập index pandas với ngày tháng bằng cách sử dụng gói pd.Series
* Thực hiện lấy mẫu lại bằng cách sử dụng gói ts.resample
* Thay đổi tần số

Ví dụ sau cho thấy bạn xử lý và cắt dữ liệu chuỗi thời gian bằng cách sử dụng Pandas. Lưu ý rằng ở đây ta đang sử dụng dữ liệu Dao động Bắc Cực hàng tháng, dữ liệu này có thể được tải xuống từ [month.ao.index.b50.current.ascii](http://www.cpc.ncep.noaa.gov/products/precip/CWlink/daily_ao_index/monthly.ao.index.b50.current.ascii)và có thể được chuyển đổi sang định dạng văn bản .

**Xử lý dữ liệu time series :**

Để xử lý dữ liệu chuỗi thời gian, bạn sẽ phải thực hiện các bước sau:

import các package cần thiết :

**import** numpy as np

**import** matplotlib.pyplot as plt

**import** pandas as pd

xác định một hàm sẽ đọc dữ liệu từ tệp đầu vào, như được hiển thị trong đoạn code dưới đây:

**def** **read\_data**(input\_file):

input\_data = np.loadtxt(input\_file, delimiter = None)

Bây giờ, hãy chuyển đổi dữ liệu này thành chuỗi thời gian. Đối với điều này, hãy tạo phạm vi ngày trong chuỗi thời gian . Trong ví dụ này,ta giữ một tháng làm tần suất dữ liệu. Tệp của ta có dữ liệu bắt đầu từ tháng 1 năm 1950.

dates = pd.date\_range('1950-01', periods = input\_data.shape[0], freq = 'M')

Tạo dữ liệu chuỗi thời gian với sự trợ giúp của Chuỗi Pandas, như được hiển thị bên dưới -

output = pd.Series(input\_data[:, **index**], **index** = dates)

**return** output

**if** \_\_name\_\_=='\_\_main\_\_':

Nhập đường dẫn của tệp đầu vào như sau :

input\_file = "/Users/admin/AO.txt"

chuyển đổi cột sang định dạng thời gian

timeseries = read\_data(input\_file)

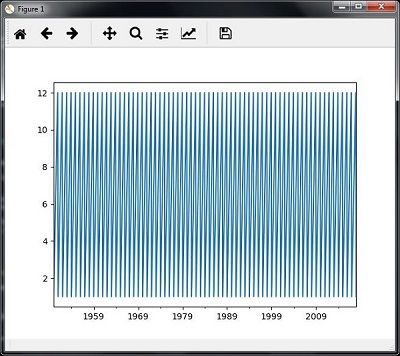
Cuối cùng, plot và visualize dữ liệu :

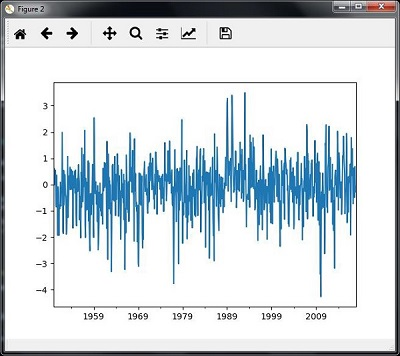
**plt**.figure()

**timeseries**.plot()

**plt**.show()

plots hiển thị như sau :





**Cắt dữ liệu time series :**

Cắt chỉ liên quan đến việc truy xuất một số phần của dữ liệu chuỗi thời gian. Là một phần của ví dụ, mình chỉ cắt dữ liệu từ năm 1980 đến năm 1990. Hãy quan sát đoạn mã sau thực hiện tác vụ này

**timeseries**['1980':'1990'].plot()

<**matplotlib**.axes.\_subplots.AxesSubplot **at** 0**xa0e4b00**>

**plt**.show()

Khi bạn chạy đoạn code để cắt dữ liệu chuỗi thời gian, bạn có thể quan sát biểu đồ sau như được hiển thị trong hình ảnh ở đây:

