|  |  |
| --- | --- |
| Logo T3H  **PHÒNG LẬP TRÌNH & MẠNG**  **TRUNG TÂM TIN HỌC**  **ĐẠI HỌC KHOA HỌC TỰ NHIÊN**  **⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯**  **BÁO CÁO ĐỒ ÁN**  **Chuyên viên Data Science (Applied Data Science Certificate)**  **Đồ án:**  **Dự đoán giá vàng**      Giáo viên hướng dẫn: Nguyễn Quan Liêm  Học viên thực hiện: Lê Thị Thuý Hằng    ***TP. Hồ Chí Minh, ngày 20 tháng 08 năm 2021*** | |
| ***NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN***  ***⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯***  **………………………………………………………………………...**  **………………………………………………………………………...**  **………………………………………………………………………...**  **………………………………………………………………………...**  **………………………………………………………………………...**  **………………………………………………………………………...**  **………………………………………………………………………...**  **………………………………………………………………………...**  **………………………………………………………………………...**  **Ngày …. tháng …. năm …**  **Giáo viên hướng dẫn**  *(ký và ghi rõ họ tên)* |

|  |
| --- |
| ***GIỚI THIỆU ĐỒ ÁN***  ⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯  Trong vài năm trở lại đây, những rủi ro và bất ổn diễn ra khắp nơi khiến vàng trở thành một kênh đầu tư, một phương tiện phòng tránh rủi ro được ưa chuộng hơn bao giờ hết. Tuy nhiên, chính bản thân vàng cũng mang trong nó những diễn biến phức tạp và khó lường.  Từ đó, yêu cầu nắm bắt được diễn biến và dự báo giá vàng trở nên vô cùng quan trọng và cần thiết đối với các nhà đầu tư hoặc quỹ tài chính. |

LỜI CẢM ƠN

Em xin trân trọng cảm ơn Thầy Nguyễn Văn Liêm đã giúp em định hướng, hỗ trợ tận tình trong quá trình thực hiện đồ án này.

Đồng thời, gửi lời cảm ơn đến các Thầy, Cô đã giảng dạy truyền đạt những kiến thức và kinh nghiệm trong quãng thời gian em học tập tại Trung tâm.

**MỤC LỤC**

[CHƯƠNG 1: GIỚI THIỆU 6](#_Toc80715838)

[1.1 Lý do lựa chọn đề tài 6](#_Toc80715839)

[1.2. Mục tiêu của nghiên cứu 6](#_Toc80715840)

[1.3. Phạm vi, đối tượng nghiên cứu 6](#_Toc80715841)

[1.4. Nguồn dữ liệu 6](#_Toc80715842)

[1.5. Phương pháp nghiên cứu 7](#_Toc80715843)

[CHƯƠNG 2: MÔ TẢ DỮ LIỆU 8](#_Toc80715844)

[2.1. Giới thiệu về Dataset 8](#_Toc80715845)

[2.2. Tiền xử lý dữ liệu 8](#_Toc80715846)

[CHƯƠNG 3: CƠ SỞ LÝ THUYẾT VÀ XÂY DỰNG MÔ HÌNH 13](#_Toc80715847)

[3.1. Cơ sở lý thuyết 13](#_Toc80715848)

[3.1.1. Gới thiệu chuỗi thời gian (Time Series) 13](#_Toc80715849)

[a. Định nghĩa model Long short-term memory 15](#_Toc80715850)

[b. Lịch sử hình thành 16](#_Toc80715851)

[c. Cấu trúc model¶ 17](#_Toc80715852)

[**3.1.2. Giới thiệu mô hình Artificial Neural Network (ANN)** 18](#_Toc80715853)

[3.2. Xây dựng mô hình 21](#_Toc80715854)

[**3.2.1. Mô hinh ARIMA** 21](#_Toc80715855)

[CHƯƠNG 4: ĐÁNH GIÁ MÔ HÌNH VÀ SO SÁNH 32](#_Toc80715856)

[4.1. Đánh giá chung các mô hình 32](#_Toc80715857)

[Tài liệu tham khảo 33](#_Toc80715858)

# CHƯƠNG 1: GIỚI THIỆU

## 1.1 Lý do lựa chọn đề tài

Theo dòng chảy của lịch sử, nhiều hình thái tiền tệ đã được ra đời nhằm đáp ứng nhu cầu giao thương ngày một phát triển. Cùng với đó là sự bùng nổ của các công cụ đầu tư trên thị trường tài chính trong nền kinh tế toàn cầu hóa.

Tuy nhiên, bên cạnh những lợi ích to lớn mà những hình thái tiền tệ và công cụ đầu tư hiện đại này mang lại thì chúng cũng hàm chứa không ít bất ổn. Tiền dấu hiệu thiếu cơ sở ổn định vững chắc, luôn đứng trước áp lực giảm giá. Các sản phẩm, công cụ đầu tư tài chính tiềm ẩn nhiều rủi ro do bản chất phức tạp. Khủng hoảng tài chính Châu Á 1997 – 1998 và gần đây hơn là những ảnh hưởng to lớn đến kinh tế tài chính toàn do đại dịch Covid gây ra là các minh chứng cụ thể cho điều này. Trong bối cảnh như vậy, vai trò của vàng với tư cách của một công cụ tiền tệ, một công cụ đầu tư, một kênh trú ẩn an toàn, bảo đảm được nguồn vốn đã phát triển mạnh mẽ.

Việc giữ vàng là hiện tượng rất phổ biến trong các quỹ đầu tư, tài chính hoặc cá nhân riêng lẻ. Thị trường vàng lại luôn biến động. Từ đó có thể thấy rằng, vàng kênh đầu tư sinh lời cũng như là một kênh đầu tư an toàn. Tuy nhiên cũng ẩn chứa không ít rủi ro mà nhà đầu tư cần phải nắm bắt, dự báo được chiều dao động của giá vàng nhằm giúp khoản đầu tư có thể sinh lời hay ít nhất cũng hạn chế một phần rủi ro do thị trường mang lại.

## 1.2. Mục tiêu của nghiên cứu

Phân tính và xây dựng mô hình phù hợp để dự đoán giá vàng dựa trên các dữ liệu thu thập được.

## 1.3. Phạm vi, đối tượng nghiên cứu

Giá vàng thế giới trong gia đoạn 07/2016 – 06/2021. Bao gồm 1818 quan sát tương ứng với 1818 ngày trong giai đoạn này.

## 1.4. Nguồn dữ liệu

Dữ liệu được thực hiện cào về dựa trên trên lịch sử giá vàng thế giới tại trang web <https://vn.investing.com/commodities/gold-historical-data>.

## 1.5. Phương pháp nghiên cứu

Phương pháp sử dụng kết hợp mô hình chuỗi thời gian ARIMA và mạng neural nhân tạo hay thường gọi ngắn gọn là mạng neural(tiếng Anh là Artificial Neural network - ANN hay Neural Network) sử dụng thư viện MLPRegressor.

**1.6. Kết cấu khái quát của Đồ án**

Đồ án được chia thành 5 chương gồm:

Chương 1: Giới thiệu

Chương 2: Mô tả dữ liệu

Chương 3: Cơ sở lý thuêts và xây dựng mô hình

Chương 4: Kết quả dựa báo

Chương 5: Demo

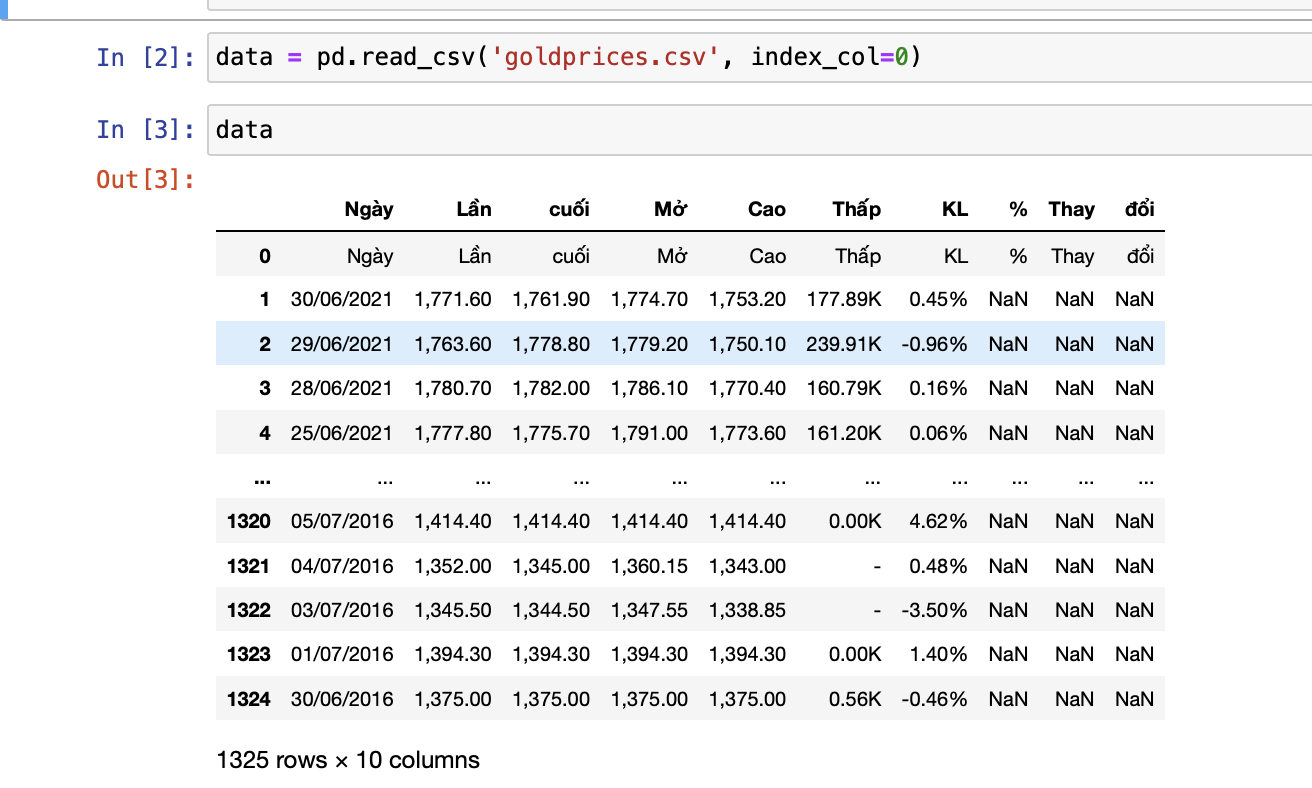
# CHƯƠNG 2: MÔ TẢ DỮ LIỆU

## 2.1. Giới thiệu về Dataset

Dữ liệu thu thập được bao gồm giá vàng lần cuối, giá mở, giá cao, giá thấp, khối lượng thực hiện (KL) và phần trăm thay đổi theo ngày được cào về từ trang web: <https://vn.investing.com/commodities/gold-historical-data> trong khoảng thời gian từ tháng 07/2016 đến tháng 06/2021.

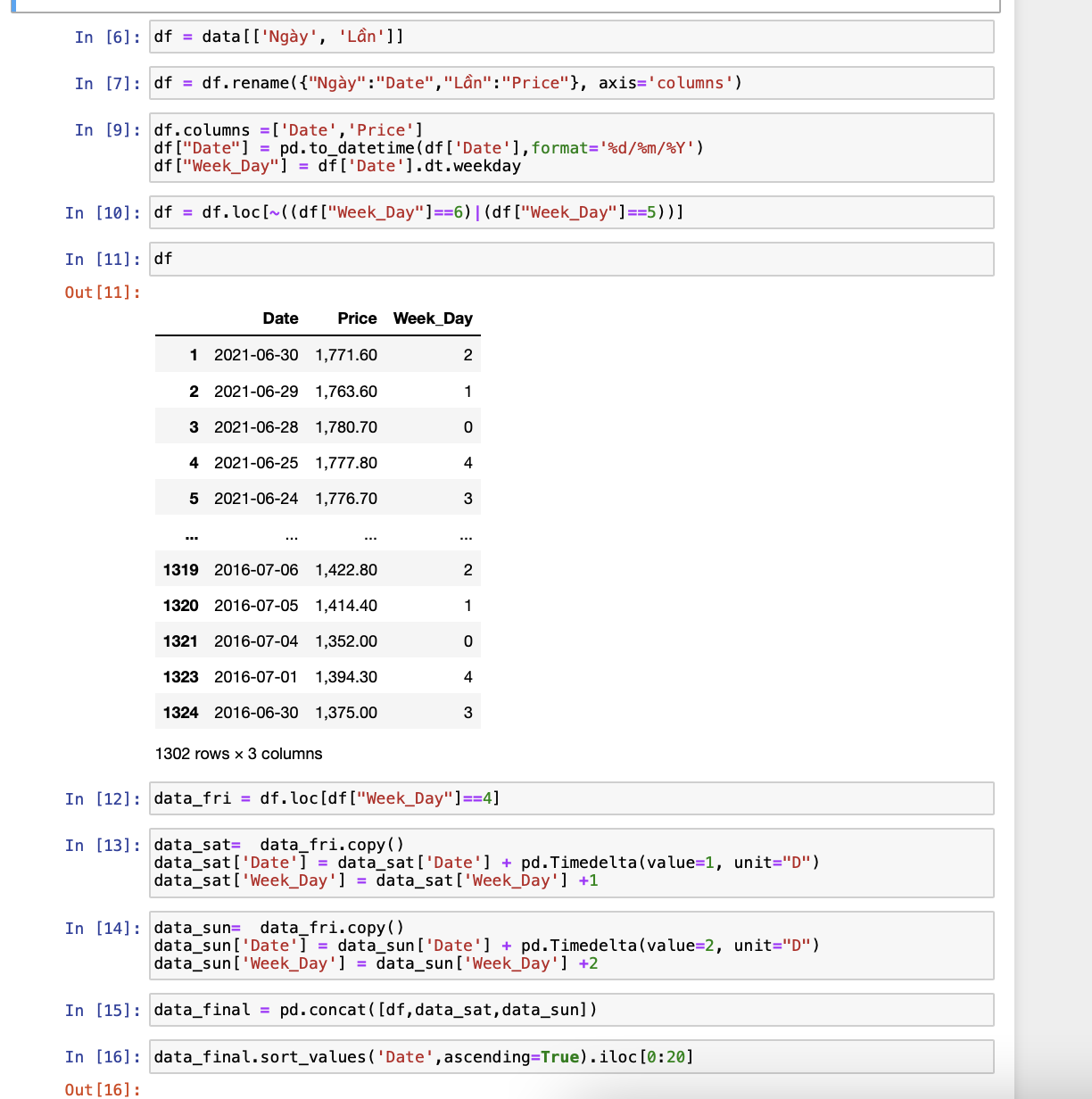
## 2.2. Tiền xử lý dữ liệu

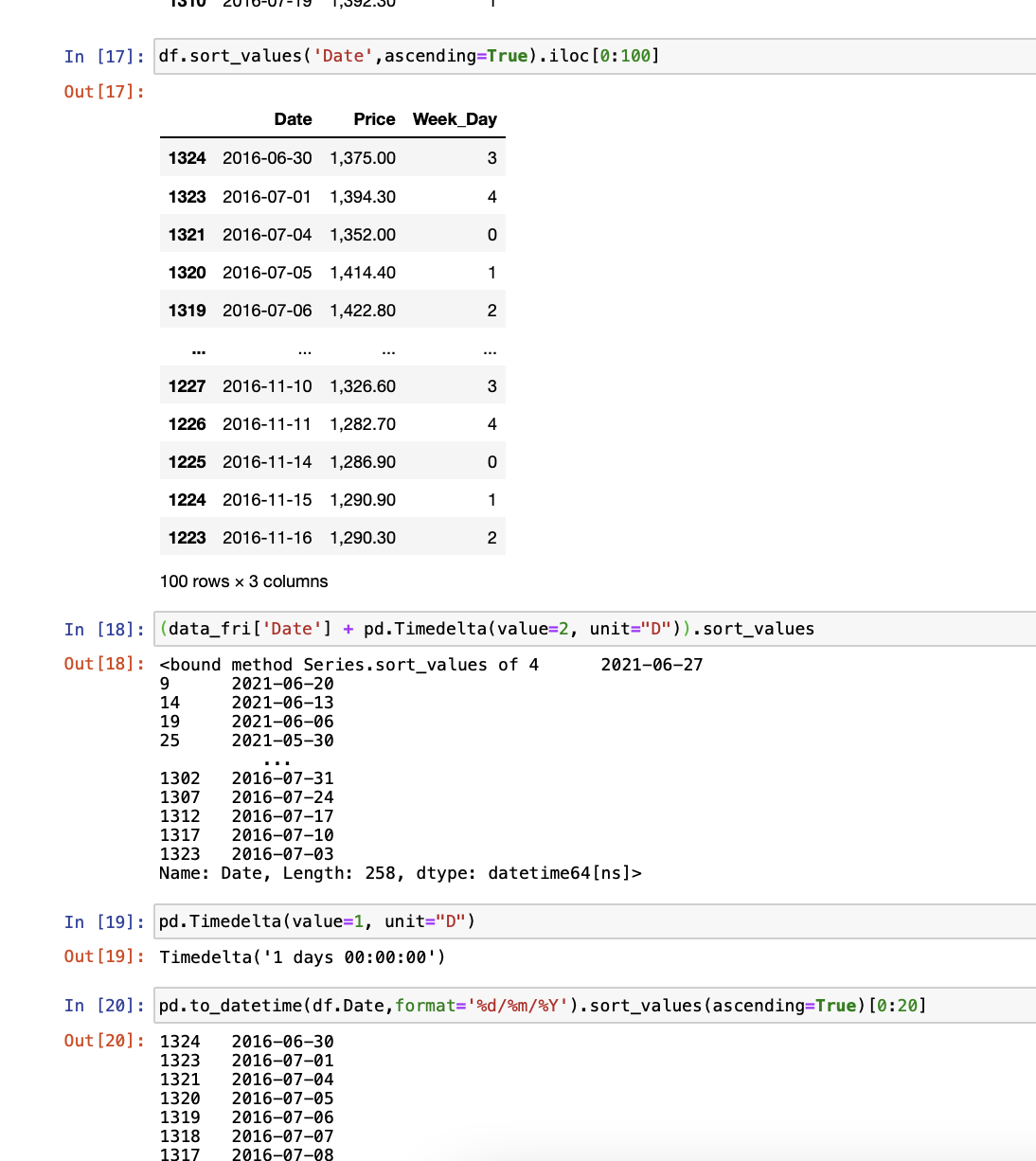
Dữ liệu ban đầu được cào về với tên cột chưa chính xác, chỉ tập trung lấy dữ liệu cột ngày và cột giá vàng lần cuối.



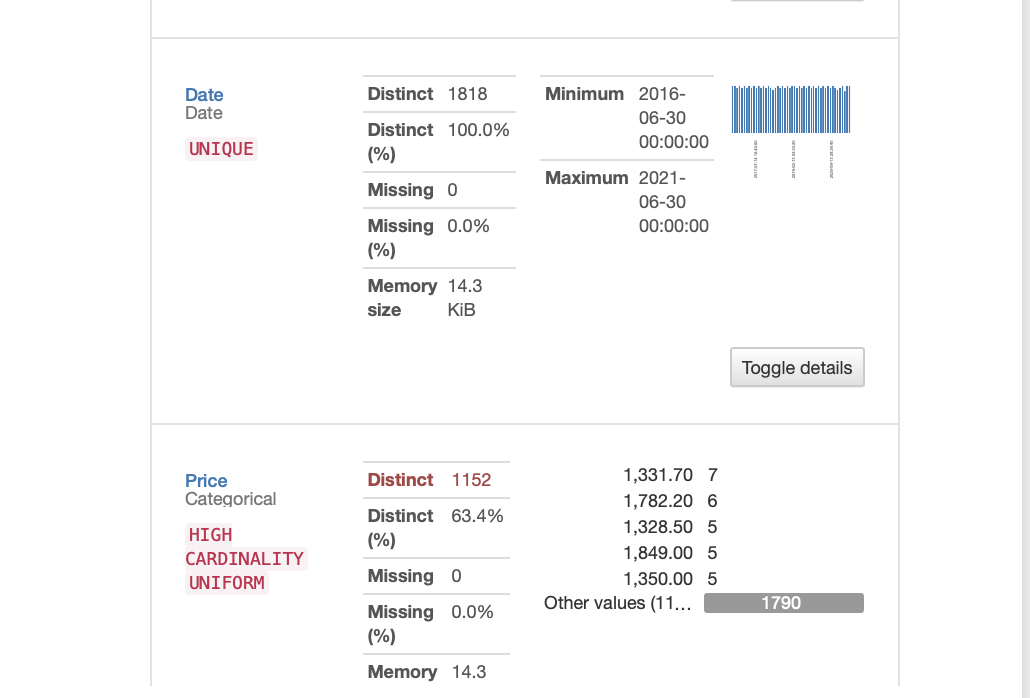
Tiến hành thực hiện đổi tên cột để thuận tiện hơn trong quá trình xử lý. Bên cạnh đó, trong quá trình xử lý nhận thấy số ngày được cào về chưa bao gồm đầy đủ dữ liệu của tất cả các ngày trong tháng. Thông thường giá vàng sẽ không được cập nhật vào cuối tuần nên dữ liệu sẽ thiếu mất các ngày cuối tuần. Tuy nhiên sẽ có cả những ngày được cập nhật

Do đó tiến hàng thực hiện thêm ngày thứ bảy và chủ nhật còn thiếu vào bảng dữ liệu lấy giá vàng cuối cùng là ngày thứ sáu trong tuần để đưa vào những ngày này.



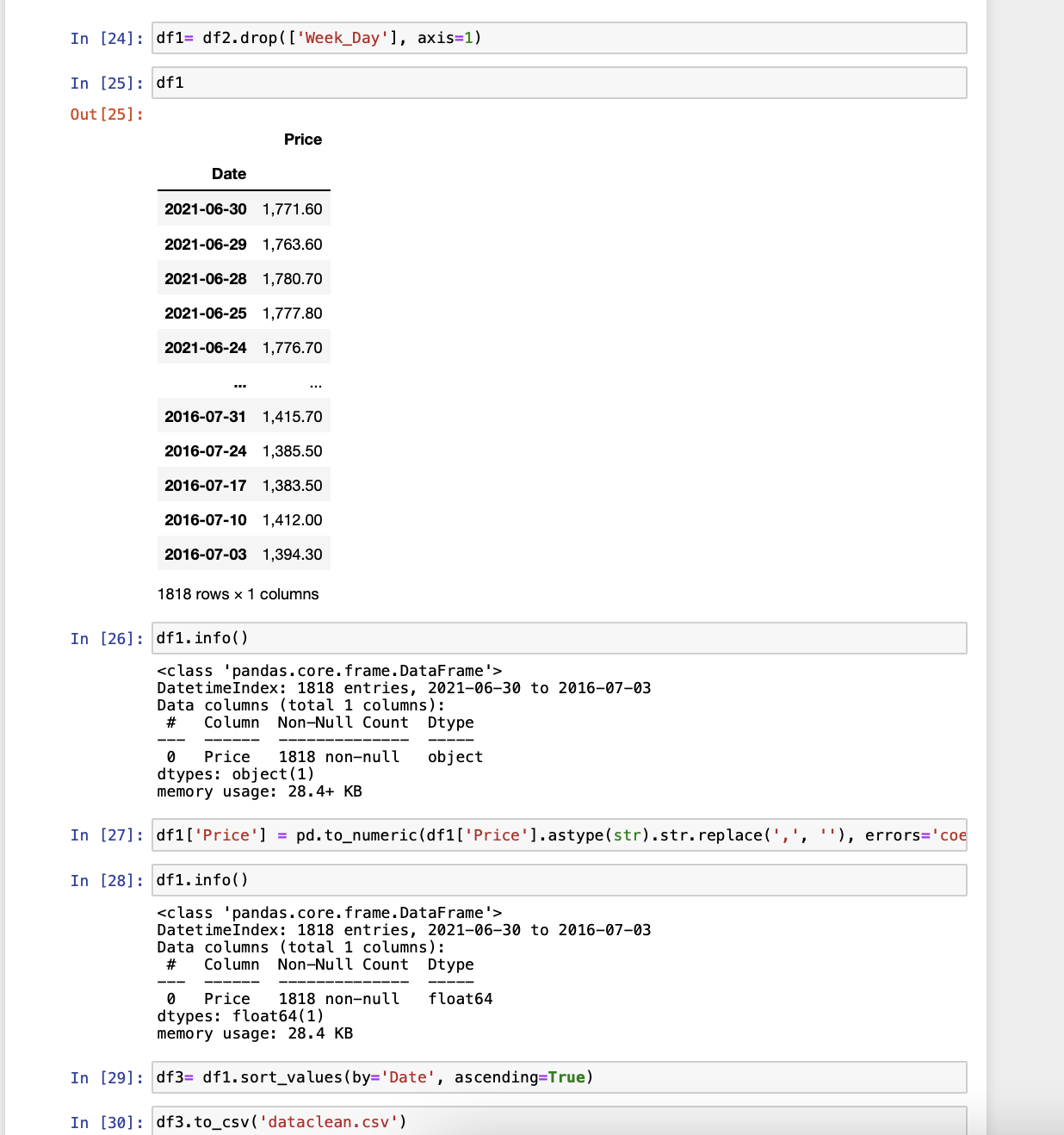


Sau khi đã xử lý các bước ban đầu thực hiện, sử dụng pandas\_profiling kiểm tra các dữ liệu đã xử lý để xem có dữ liệu thiếu, trùng lặp không.



Dữ liệu ngày đã không còn trùng lặp, không có missing value, không có dữ liệu null. Đảm bảo dữ liệu tương đổi ổn để có thể sử dụng.

Tiến hành loại bỏ cột không cần thiết và định dạng lại cột dữ liệu sau đó lưu trữ dữ liệu đã được tiền xử lý.



Sau khi hoàn tất, chúng ta có được một bộ dữ liệu tương đối hoàn chỉnh gồm bao gồm ngày và giá vàng theo từng ngày trong khoảng thời gian từ ngày 30/06/2016 đến ngày 30/06/2021.

# CHƯƠNG 3: CƠ SỞ LÝ THUYẾT VÀ XÂY DỰNG MÔ HÌNH

## 3.1. Cơ sở lý thuyết

## 3.1.1. Gới thiệu chuỗi thời gian (Time Series)

Dự báo chuỗi thời gian là một lớp mô hình quan trọng trong thống kê, kinh tế lượng và machine learning. Sở dĩ chúng ta gọi lớp mô hình này là chuỗi thời gian (*time series*) là vì mô hình được áp dụng trên các chuỗi đặc thù có yếu tố thời gian. Một mô hình chuỗi thời gian thường dự báo dựa trên giả định rằng các qui luật trong quá khứ sẽ lặp lại ở tương lai. Do đó xây dựng mô hình chuỗi thời gian là chúng ta đang mô hình hóa mối quan hệ trong quá khứ giữa biến độc lập (biến đầu vào) và biến phụ thuộc (biến mục tiêu). Dựa vào mối quan hệ này để dự đoán giá trị trong tương lai của biến phụ thuộc.

Do là dữ liệu chịu ảnh hưởng bởi tính chất thời gian nên chuỗi thời gian thường xuất hiện những qui luật đặc trưng như : yếu tố chu kỳ, mùa vụ và yếu tố xu hướng. Đây là những đặc trưng thường thấy và xuất hiện ở hầu hết các chuỗi thời gian.

* Yếu tố chu kỳ, mùa vụ (seasonal) là những đặc tính lặp lại theo chu kỳ. Ví dụ như nhiệt độ trung bình các tháng trong năm sẽ chịu ảnh hưởng bởi các mùa xuân, hạ, thu, đông. Hay xuất nhập khẩu của một quốc gia thường có chu kỳ theo các quí.
* Yếu tố xu hướng (*trend*) thể hiện đà tăng hoặc giảm của chuỗi trong tương lai. Chẳng hạn như lạm phát là xu hướng chung của các nền kinh tế, do đó giá cả trung bình của giỏ hàng hóa cơ sở hay còn gọi là chỉ số CPI luôn có xu hướng tăng và xu hướng tăng này đại diện cho sự mất giá của đồng tiền.

Các dự báo chuỗi thời gian có tính ứng dụng cao và được sử dụng rất nhiều lĩnh vực như tài chính ngân hàng, chứng khoán, bảo hiểm, thương mại điện tử, marketing, quản lý chính sách. Bên dưới là một số ứng dụng của dự báo chuỗi thời gian:

* Dự báo nhu cầu thị trường để lập kết hoạch sản xuất kinh doanh cho hãng.
* Dự báo lợi suất tài sản tài chính, tỷ giá, giá cả hàng hóa phái sinh để thực hiện trading hiệu quả trong market risk.
* Dự báo giá chứng khoán, các chuỗi lợi suất danh mục để quản trị danh mục đầu tư.
* Dự báo giá bitcoin, giá dầu mỏ, giá gas,…
* Dự báo nhiệt độ, lượng mưa để lập kế hoạch sản xuất nông, lâm, ngư nghiệp.
* Dự báo tác động của các nhân tố vĩ mô như lãi suất, cung tiền, đầu tư trực tiếp nước ngoài, chi tiêu chính phủ, lạm phát,… tác động lên tăng trưởng GDP để điều hành nền kinh tế.

Vai trò của chuỗi thời gian rất quan trọng đối với nền kinh tế và hoạt động của doanh nghiệp nên trong machine learning và thống kê có những ngành học nghiên cứu chuyên sâu về chuỗi thời gian như kinh tế lượng, định giá tài sản tài chính.

**3.1.1.1 Mô hình ARIMA**

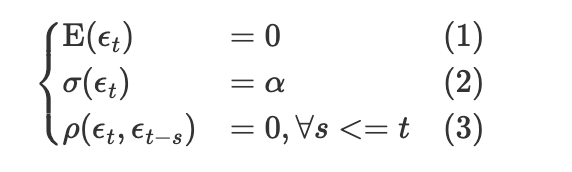
Chúng ta biết rằng hầu hết các chuỗi thời gian đều có sự tương quan giữa giá trị trong quá khứ đến giá trị hiện tại. Mức độ tương quan càng lớn khi chuỗi càng gần thời điểm hiện tại. Chính vì thể mô hình ARIMA sẽ tìm cách đưa vào các biến trễ nhằm tạo ra một mô hình dự báo fitting tốt hơn giá trị của chuỗi.

ARIMA model là viết tắt của cụm từ Autoregressive Intergrated Moving Average. Mô hình sẽ biểu diễn phương trình hồi qui tuyến tính đa biến (multiple linear regression) của các biến đầu vào (còn gọi là biến phụ thuộc trong thống kê) là 2 thành phần chính:

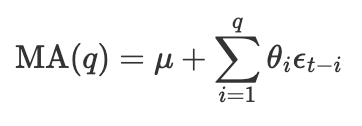
* **Auto regression**: Kí hiệu là AR. Đây là thành phần tự hồi qui bao gồm tợp hợp các độ trễ của biến hiện tại. Độ trễ bậc p chính là giá trị lùi về quá khứ p bước thời gian của chuỗi. Độ trễ dài hoặc ngắn trong quá trình AR phụ thuộc vào tham số trễ p. Cụ thể, quá trình AR(p) của chuỗi xt được biểu diễn như bên dưới:

AR(p)=ϕ0 + ϕ1xt – 1 + ϕ2xt−2 + ⋯ + ϕpxt−p

* **Moving average**: Qúa trình trung bình trượt được hiểu là quá trình dịch chuyển hoặc thay đổi giá trị trung bình của chuổi theo thời gian. Do chuỗi của chúng ta được giả định là dừng nên quá trình thay đổi trung bình dường như là một chuỗi nhiễu trắng. Qúa trình moving average sẽ tìm mối liên hệ về mặt tuyến tính giữa các phần tử ngẫu nhiên ϵt (stochastic term). Chuỗi này phải là một chuỗi nhiễu trắng thỏa mãn các tính chất:



Vế (1) có nghĩa rằng kì vọng của chuỗi bằng 0 để đảm bảo chuỗi dừng không có sự thay đổi về trung bình theo thời gian. Vế (2) là phương sai của chuỗi không đổi. Do kì vọng và phương sai không đổi nên chúng ta gọi phân phối của nhiễu trắng là phân phối xác định (identical distribution) và được kí hiệu là ϵt ∼ WN(0,σ2). Nhiễu trắng là một thành phần ngẫu nhiên thể hiện cho yếu tố không thể dự báo của model và không có tính qui luật. Quá trình trung bình trượt được biểu diễn theo nhiễu trắng như sau:



Qúa trình này có thể được biểu diễn theo [dịch chuyển trễ - backshift operator](https://en.wikipedia.org/wiki/Lag_operator) B như sau:

MA(q) = μ + (1 + θ1B + ⋯ + θqBq)ϵt

Như vậy bạn đọc đã hình dung ra moving average là gì rồi chứ? Về mặt ý tưởng thì đó chính là quá trình hồi qui tuyến tính của giá trị hiện tại theo các giá trị hiện tại và quá khứ của sai số nhiễu trắng (white noise error term) đại diện cho các yếu tố shock ngẫu nhiên, những sự thay đổi không lường trước và giải thích bởi mô hình.

* **Intergrated**: Là quá trình đồng tích hợp hoặc lấy sai phân. Yêu cầu chung của các thuật toán trong time series là chuỗi phải đảm bảo tính dừng. Hầu hết các chuỗi đều tăng hoặc giảm theo thời gian. Do đó yếu tố tương quan giữa chúng chưa chắc là thực sự mà là do chúng cùng tương quan theo thời gian. Khi biến đổi sang chuỗi dừng, các nhân tố ảnh hưởng thời gian được loại bỏ và chuỗi sẽ dễ dự báo hơn. Để tạo thành chuỗi dừng, một phương pháp đơn giản nhất là chúng ta sẽ lấy sai phân. Một số chuỗi tài chính còn qui đổi sang logarit hoặc lợi suất. Bậc của sai phân để tạo thành chuỗi dừng còn gọi là bậc của quá trình đồng tích hợp (order of intergration). Qúa trình sai phân bậc d của chuỗi được thực hiện như sau:
  + Sai phân bậc 1: I(1) = Δ(xt)=xt−xt−1
  + Sai phân bậc d: I(d) = Δd(xt) = Δ(Δ(…Δ(xt))) = d times

Thông thường chuỗi sẽ dừng sau quá trình đồng tích hợp I(0) hoặc I(1). Rất ít chuỗi chúng ta phải lấy tới sai phân bậc 2. Một số trường hợp chúng ta sẽ cần biến đổi logarit hoặc căn bậc 2 để tạo thành chuỗi dừng. Phương trình hồi qui ARIMA(p, d, q) có thể được biểu diễn dưới dạng:

Δxt = ϕ1Δxt−1 + ϕ2Δxt−2 + ... + ϕpΔxt−p + θ1ϵt−1 + θ2ϵt−2 + ... + θqϵt−q

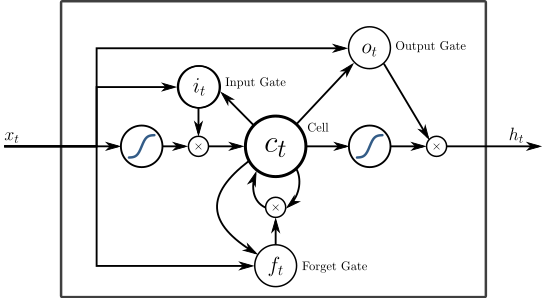
Trong đó Δxt là giá trị sai phân bậc d và ϵt là các chuỗi nhiễu trắng.

Như vậy về tổng quát thì ARIMA là mô hình kết hợp của 2 quá trình tự hồi qui và trung bình trượt. Dữ liệu trong quá khứ sẽ được sử dụng để dự báo dữ liệu trong tương lai. Trước khi huấn luyện mô hình, cần chuyển hóa chuỗi sang chuỗi dừng bằng cách lấy sai phân bậc 1 hoặc logarit. Ngoài ra mô hình cũng cần tuân thủ điều kiện ngặt về sai số không có hiện tượng tự tương quan và phần dư là nhiễu trắng.

**3.1.1.2 Mô hình LSTM**

## a. Định nghĩa model Long short-term memory

**Long short-term memory** là một dạng mô hình Recurrent neural network (RNN) mà mạng network của nó được tổng hợp từ nhiều các đơn vị Long short-term memory. Các đơn vị LSTM đều có điểm chung đó là được cấu tạo từ một cell, một input gate, một output gate và một forget gate. Cell sẽ có tác dụng ghi nhớ giá trị tùy ý của một khoảng thời gian bất kì và 3 cổng sẽ điều chỉnh giá trị của luồng thông tin vào và ra của cell. Bên dưới là sơ đồ của một đơn vị Long short-term memory:



Mạng LSTM rất phù hợp với các bài toán phân loại và dự báo dựa trên dữ liệu dạng chuỗi thời gian bởi vì model có khả năng ghi nhớ tức thời các sự kiện xảy ra ở gần nó. LSTM được thiết kế để giải quyết sự bùng nổ và triệt tiêu gradient, hiện tượng mà khiến cho các mô hình truyền thống của RNNs có thể gặp phải. Sự mở rộng của gap độ dài là một tiến bộ của model LSTM so với RNN, [model ẩn Markov](https://en.wikipedia.org/wiki/Hidden_Markov_models), và các phương pháp học chuỗi khác.

### b. Lịch sử hình thành

LSTM đã được giới thiệu lần đầu tiền vào năm 1997 bởi Sepp Hochreiter (lĩnh vực y sinh và học máy) và Jurgen Schmidhuber (lĩnh vực trí tuệ nhân tạo), sau đó công trình đã được phát triển bởi nhóm của giáo sư Felix Gers vào năm 2000 bằng việc đưa thêm forget gate vào mạng LSTM vào cấu trúc ban đầu.

LSTM đã đạt được hiệu quả cao trong các mô hình xử lý ngôn ngữ tự nhiên, nhận diện chữ viết tay và dành chiến thắng trong cuộc thi ICDAR được tổ chức vào năm 2009. LSTM cũng là một trong những thành phần chính của mạng đạt được 17.7% phoneme error rate (một chỉ số được dùng để đo mức độ sai khác giữa các âm) trên bộ dữ liệu âm thanh [TIMIT](https://en.wikipedia.org/wiki/TIMIT). Các hãng công nghệ lớn như Google, Facebook, Apple, Microsoft đều sử dụng LSTM như một nên tảng trong các ứng dụng nhận diện giọng nói của mình.

### c. Cấu trúc model[¶](https://render.githubusercontent.com/view/ipynb?color_mode=auto&commit=87c56dedaa6ff4a2f8b8559dfefb07679c5f9993&enc_url=68747470733a2f2f7261772e67697468756275736572636f6e74656e742e636f6d2f7068616d64696e686b68616e682f4c53544d2f383763353664656461613666663461326638623835353964666566623037363739633566393939332f4c53544d2e6970796e62&nwo=phamdinhkhanh%2FLSTM&path=LSTM.ipynb&repository_id=149889998&repository_type=Repository#1.1.2.-C%E1%BA%A5u-tr%C3%BAc-model)

Có một vài các cấu trúc khác nhau của một đơn vị LSTM. Nhưng cấu trúc chung đều được tổng hợp từ một memory cell, một input gate, một output gate và một forget gate.

Một LSTM cell sẽ nhậm một đầu vào và lưu nó trong một số giai đoạn. Nó tương đương với việc áp dụng hàm f(x) = x cho giá trị đầu vào. Chúng ta gọi hàm này là hàm xác định do có giá trị hàm bằng giá trị của biến. Bởi vì đạo hàm của hàm xác định là một hằng số nên khi thực hiện quá trình lan truyền ngược để training mô hình thì gradient sẽ không bị triệt tiêu. Hàm kích hoạt của các cổng LSTM thường là logistic function. input gate sẽ kiểm soát sự mở rộng để giá trị mới được đưa vào một cell. forget gate sẽ kiểm soát sự mở rộng của các giá trị tồn tại trong một cell và output gate sẽ kiểm soát sự mở rộng các giá trị trong một cell được sử dụng để tính toán hàm activation của một đơn vị LSTM.

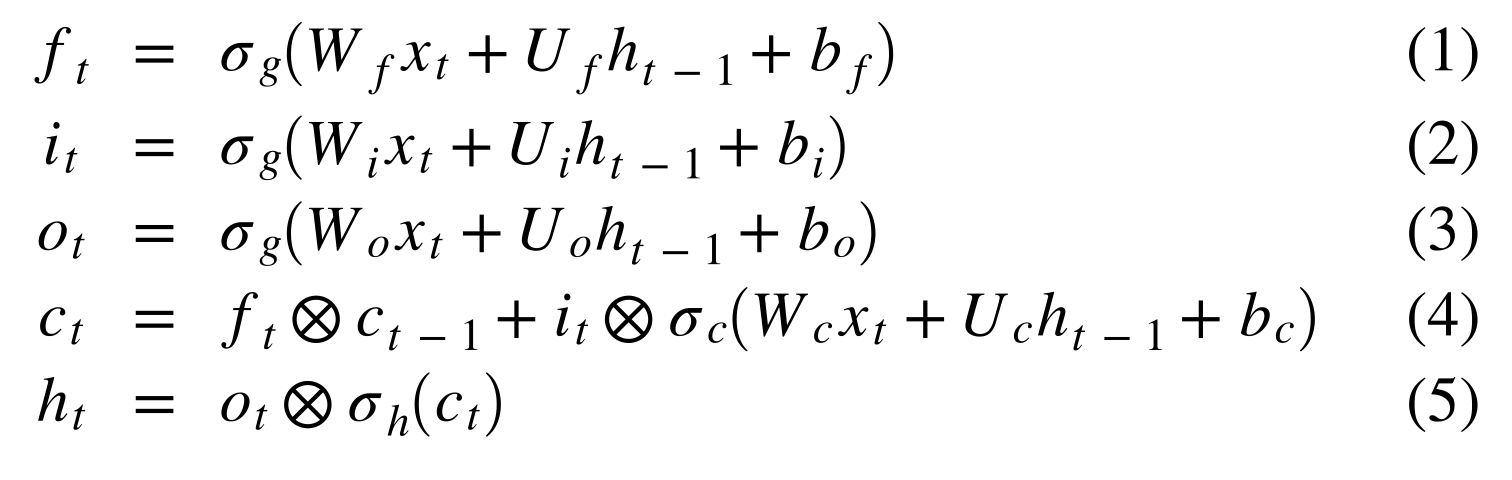
Có những kết nối vào và ra trong các cổng LSTM và một số trong các kết nối này tái tục. Trọng số của những kết nối này sẽ được học trong suốt quá trình training để xác định các cổng được thực hiện như thế nào.

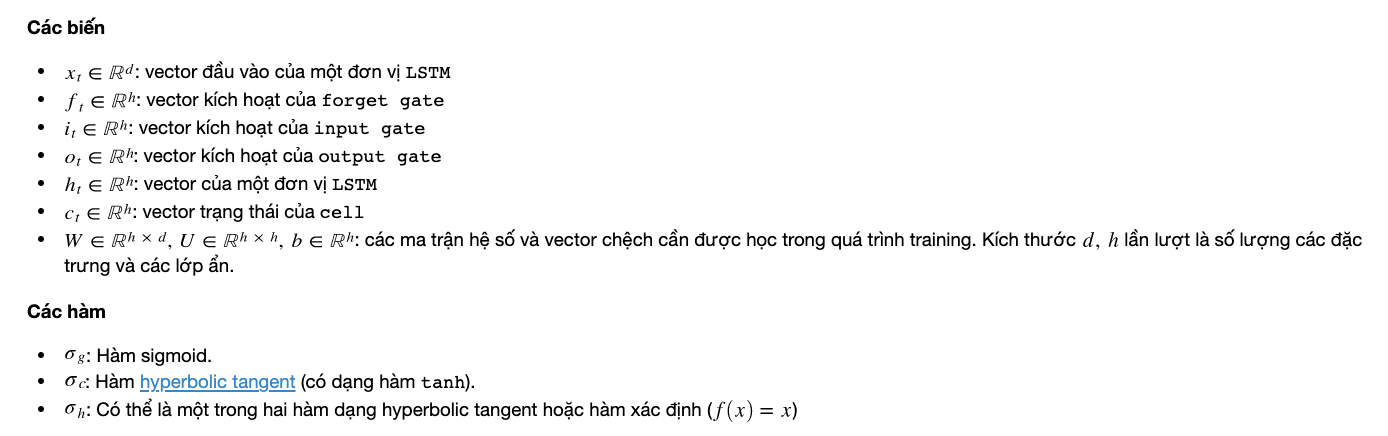
d. Các biến số

Trong các hệ phương trình được trình bày bên dưới thì và chứa lần lượt các trọng số của đầu vào và các kết nối luân hồi (recurrent connections). Ở đây có thể là input gate, output gate, foreget gate, hoặc memory cell dựa trên hàm kích hoạt được sử dụng là gì.

LSTM với một forget gate

Dạng ngắn gọn của phương trình cho quá trình truyền tiến bước của một đơn vị LSTM với một forget gate là



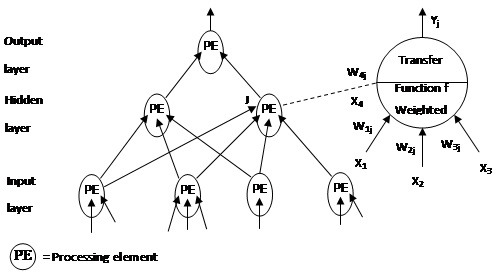
Với các giá trị khởi tạo lần là c0 = 0. H0 = 0. Toán tửlà kí hiệu của tích Hadamard product (hay còn gọi là element wise - product). Tích này sẽ áp dụng trên 2 ma trận cùng kích thước và kết quả thu được là một ma trận mới có kích thước bằng với 2 ma trận số nhân sao cho phần tử bất kì của ma trận kết quả bằng tích của 2 phần tử có vị trí tương ứng ở 2 ma trận số nhân.

**3.1.2. Giới thiệu mô hình Artificial Neural Network (ANN)**

**Mạng Nơron nhân tạo (Artificial Neural Network- ANN) là mô hình xử lý thông tin được mô phỏng dựa trên hoạt động của hệ thống thần kinh của sinh vật, bao gồm số lượng lớn các Nơron được gắn kết để xử lý thông tin. ANN giống như bộ não con người, được học bởi kinh nghiệm (thông qua huấn luyện), có khả năng lưu giữ những kinh nghiệm hiểu biết (tri thức) và sử dụng những tri thức đó trong việc dự đoán các dữ liệu chưa biết (unseen data).**

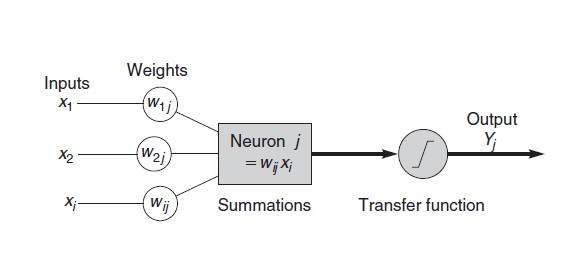
Kiến trúc chung của một mạng nơron nhân tạo (ANN) gồm 3 thành phần đó là: Input Layer, Hidden Layer và Output Layer.

Trong đó, lớp ẩn (Hidden Layer) gồm các Nơron nhận dữ liệu input từ các Nơron ở lớp (Layer) trước đó và chuyển đổi các input này cho các lớp xử lý tiếp theo. Trong một ANN có thể có nhiều lớp ẩn.

Kiến trúc tổng quát của một ANN

Trong đó các Processing Elements (PE) của ANN gọi là Nơron, mỗi Nơron nhận các dữ liệu vào (Inputs) xử lý chúng và cho ra một kết quả (Output) duy nhất. Kết quả xử lý của một Nơron có thể làm Input cho các Nơron khác.

- Quá trình xử lý thông tin của một ANN:

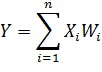


+ Inputs (dữ liệu vào): Mỗi Input tương ứng với 1 thuộc tính (attribute) của dữ liệu (patterns).

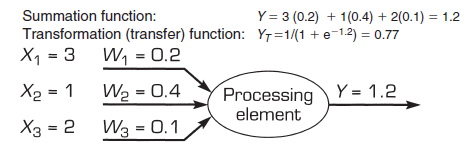
+ Output (kết quả): Kết quả của một ANN là một giải pháp cho một vấn đề.

+ Connection Weights (Trọng số liên kết) : Đây là thành phần rất quan trọng của một ANN, nó thể hiện mức độ quan trọng (độ mạnh) của dữ liệu đầu vào đối với quá trình xử lý thông tin (quá trình chuyển đổi dữ liệu từ Layer này sang layer khác). Quá trình học (Learning Processing) của ANN thực ra là quá trình điều chỉnh các trọng số (Weight) của các input data để có được kết quả mong muốn.

+ Summation Function (Hàm tổng): Tính tổng trọng số của tất cả các input được đưa vào mỗi Nơron (phần tử xử lý PE). Hàm tổng của một Nơron đối với n input được tính theo công thức sau:



+ Transfer Function (Hàm chuyển đổi): Hàm tổng (Summation Function) của một Nơron cho biết khả năng kích hoạt (Activation) của Nơron đó còn gọi là kích hoạt bên trong (internal activation). Các Nơron này có thể sinh ra một output hoặc không trong ANN (nói cách khác rằng có thể output của 1 Nơron có thể được chuyển đến layer tiếp trong mạng Nơron hoặc không). Mối quan hệ giữa Internal Activation và kết quả (output) được thể hiện bằng hàm chuyển đổi (Transfer Function).



Việc lựa chọn Transfer Function có tác động lớn đến kết quả của ANN. Hàm chuyển đổi phi tuyến được sử dụng phổ biến trong ANN là sigmoid (logical activation) function.

YT= 1/(1 + e-Y)

Trong đó :

YT: Hàm chuyển đổi

Y: Hàm tổng

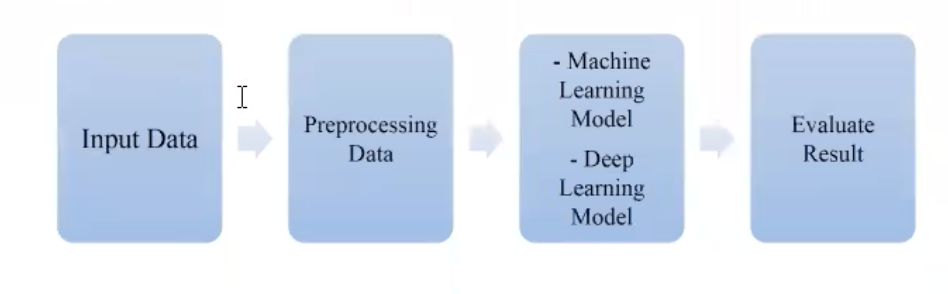
Kết quả của Sigmoid Function thuộc khoảng [0,1] nên còn gọi là hàm chuẩn hóa (Normalized Function).

Kết quả xử lý tại các Nơron (Output) đôi khi rất lớn, vì vậy transfer function được sử dụng để xử lý output này trước khi chuyển đến layer tiếp theo. Đôi khi thay vì sử dụng Transfer Function người ta sử dụng giá trị ngưỡng (Threshold value) để kiểm soát các output của các Nơron tại một layer nào đó trước khi chuyển các output này đến các Layer tiếp theo. Nếu output của một nơron nào đó nhỏ hơn giá trị ngưỡng thì nó sẽ không được chuyển đến Layer tiếp theo.

Tham khảo cụ thể thêm về Neural network models Regression tại: “[https://scikit-learn.org/stable/modules/neural\_networks\_supervised.html#regression](https://scikit-learn.org/stable/modules/neural_networks_supervised.html" \l "regression)”.

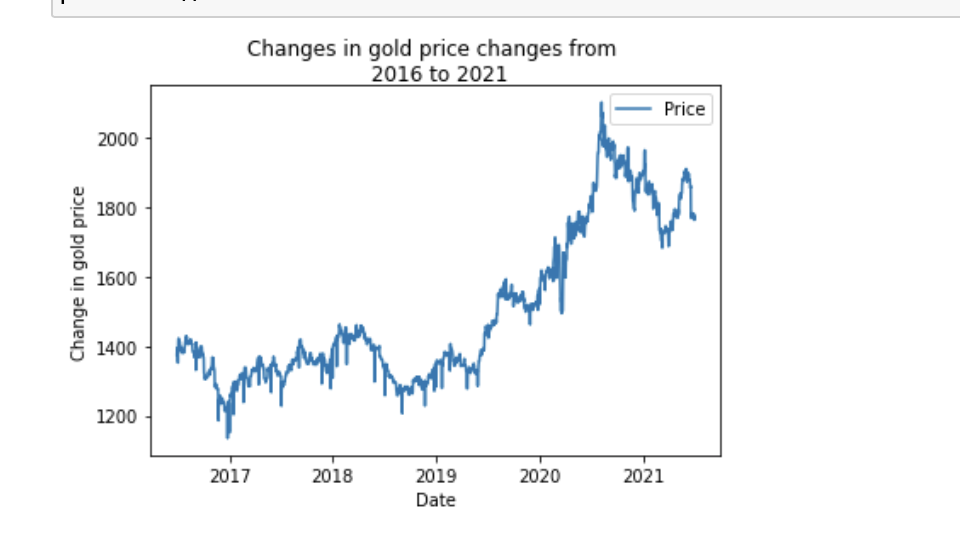
**3.2. Xây dựng mô hình**

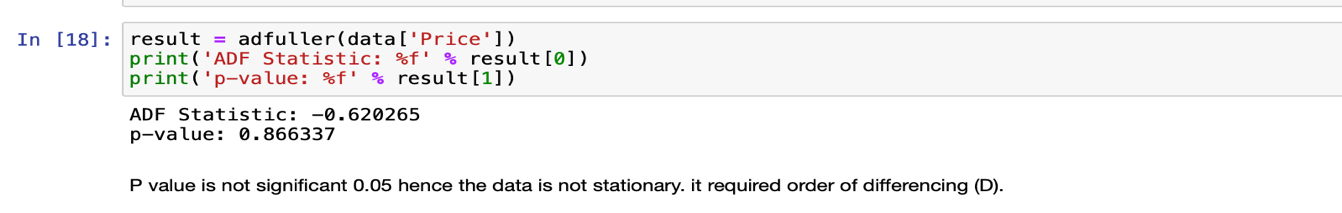
Mô hình tổng quát



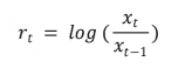
**3.2.1. Mô hinh ARIMA**

Áp dụng mô hình ARIMA trong time series để dự báo giá vàng thế giới từ giữa năm 2016 đến giữa năm 2021.



Các chuỗi thời gian thường sẽ có xu hướng tăng hoặc giảm theo thời gian, do đó yếu tố chuỗi dừng thường không đạt được.

Nếu chuỗi không dừng thì sẽ chuyển chuỗi giá sang chuỗi dừng bằng cách lấy lợi suất theo công thức sai phân bậc 1 của logarit như sau:

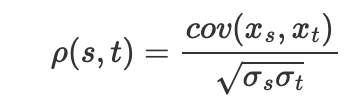


Mục tiêu của mô hình sẽ là dự báo giá lần cuối của vàng. Hàm data.shift() sẽ giúp lấy trễ theo bậc của chuỗi giá.

Nếu như chọn sai phân bậc 2 thì dường như độ trễ đi ra ngoài khoảng tin cậy khá nhanh điều này cho thấy chuỗi có thể đã sai lệch quá mức. Do vậy có thể lựa chọn sai phân bậc 1.

Lựa chọn tham số của mô hình ARIMA, gồm tham số p, d và q.

Tự tương quan Tự tương quan (ACF - AutoCorrelation Function**)**: Tự tương quan là một khái niệm quan trọng trong chuỗi thời gian. Hầu hết các chuỗi thời gian sẽ có sự tương quan với giá trị trễ của nó và các giá trị càng gần nhau thì tương quan càng mạnh hoặc các giá trị cùng thuộc 1 chu kì của chuỗi thì sẽ có tương quan cao (chẳng hạn như cùng tháng trong chu kì năm hay cùng quí trong chu kì năm). Chính vì vậy hệ số này mới có tên là tự tương quan. Hệ số tự tương quan được viết tắt là ACF và thường dùng để tìm ra độ trễ của quá trình trung bình trượt MV(q) để xây dựng các mô hình như ARIMA, GARCH, ARIMAX,… và kiểm tra yếu tố mùa vụ. Hệ số tự tương quan bậc s được xác định như sau:



Giá trị ρ(s,t) đo lường khả năng dự báo của biến xt nếu chỉ sử dụng biến xs. Trong trường hợp 2 đại lượng có tương quan hoàn hảo tức ρ(s,t) = ± 1 ta có thể biểu diễn xt = β0 + β1xs. Hệ số của β1 sẽ ảnh hưởng lên chiều của hệ số tương quan. Theo đó ρ(s,t) = 1 khi β1 > 0 và ρ(s,t) = −1 khi β1 < 0.

Trục hoành là độ trễ, trục tung là giá trị của hệ số tự tương quan tương ứng với độ trễ. Dải màu hồng chính là khoảng tin cậy 95% để giá trị hệ số tự tương quan bằng 0. Nếu tại một độ trễ nhỏ nhất mà đoạn thẳng (vuông góc với trục hoành) mà độ dài đại diện cho giá trị của hệ số tự tương quan nằm ngoài khoảng tin cậy thì đó chính là độ trễ phù hợp lớn nhất mà ta nên lựa chọn cho quá trình trung bình trượt . Nhìn chung bậc không nên quá lớn. Dựa trên biểu đồ trên, Đối với bài toán này có hệ số tương quan với bậc 1 có giá trị nằm ngoài khoảng tin cậy 95% của 0. Do đó chúng ta có thể lựa chọn bậc q = 1 dựa trên biểu đồ AutoCorrelation.

**Tự tương quan riêng phần (PACF - Partitial AutoCorrelation Function)**: Về cơ bản tương quan riêng phần cũng là chỉ số đo lường hệ số tương quan như ACF. Tuy nhiên vẫn có sự khác biệt đó là hệ số tương quan này loại bỏ ảnh hưởng của các chuỗi độ trễ trung gian (là các chuỗi trễ x t−1,…,xt−k+1 nằm giữa xt và xt−k). Một phương trình hồi qui tuyến tính giữa chuỗi hiện tại với các chuỗi độ trễ trung gian được xây dựng nhằm đánh giá ảnh hưởng của các chuỗi độ trễ lên chuỗi hiện tại. Sau đó, để tính hệ số tương quan riêng phần chúng ta sẽ loại bỏ ảnh hưởng của các độ trễ trung gian khỏi chuỗi hiện tại bằng cách trừ đi giá trị ước lượng từ phương trình hồi qui. Lấy ví dụ: Để tính tự tương quan riêng phần PACF bậc k của chuỗi xt. Đầu tiên ta sẽ hồi qui tuyến tính xt theo các chuỗi trễ của nó là xt−1,…,xt−k. Khi đó ta thu được phương trình hồi qui tuyến tính tổng quát bậc k là:

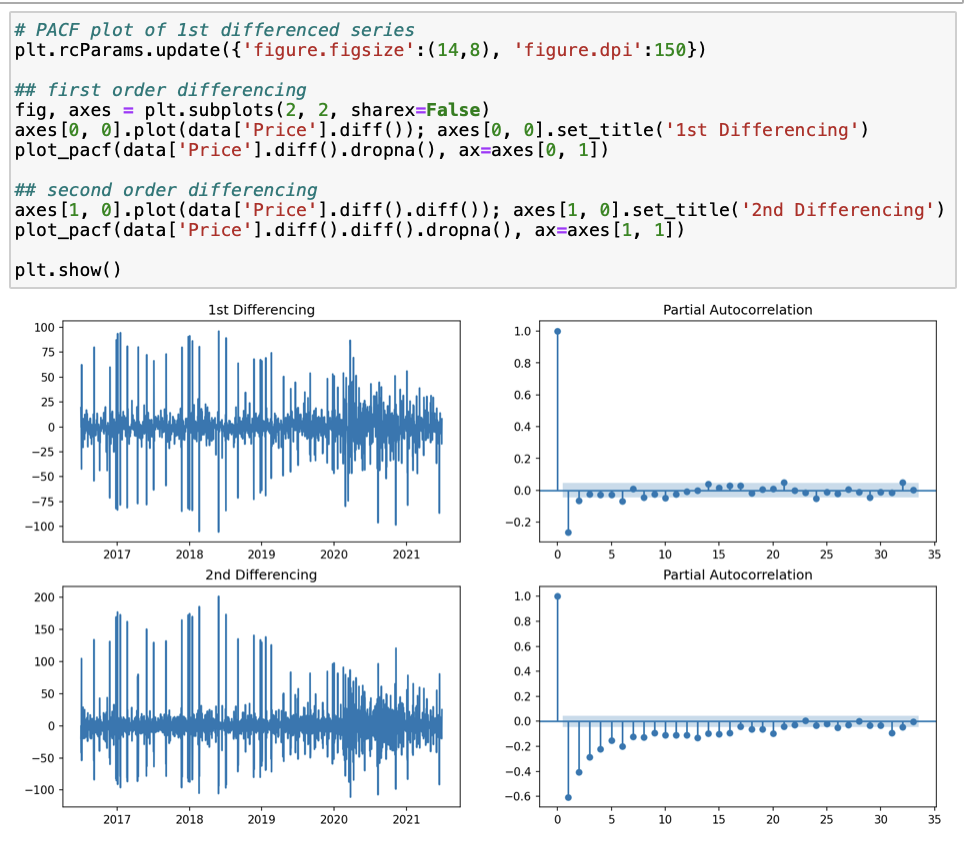
xt = ϵt + α0 + α1xt−1 + ⋯ + αkxt−k

ϵt là thành phần đại diện cho sai số. Gía trị ước lượng của mô hình đối với xt chính là:

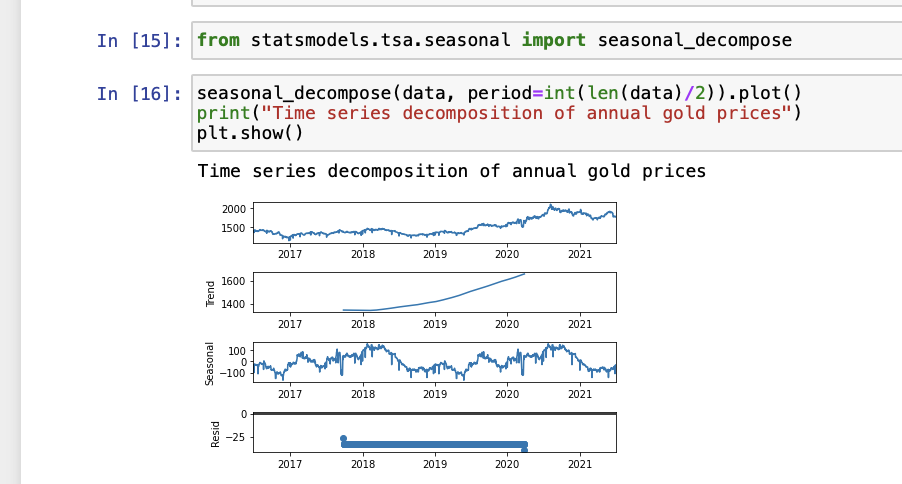
Pt,k(xt) = α0 + α1xt−1 + ⋯ + αkxt−k

Hệ số tự tương quan tuyến tính sau đó sẽ chính bằng:

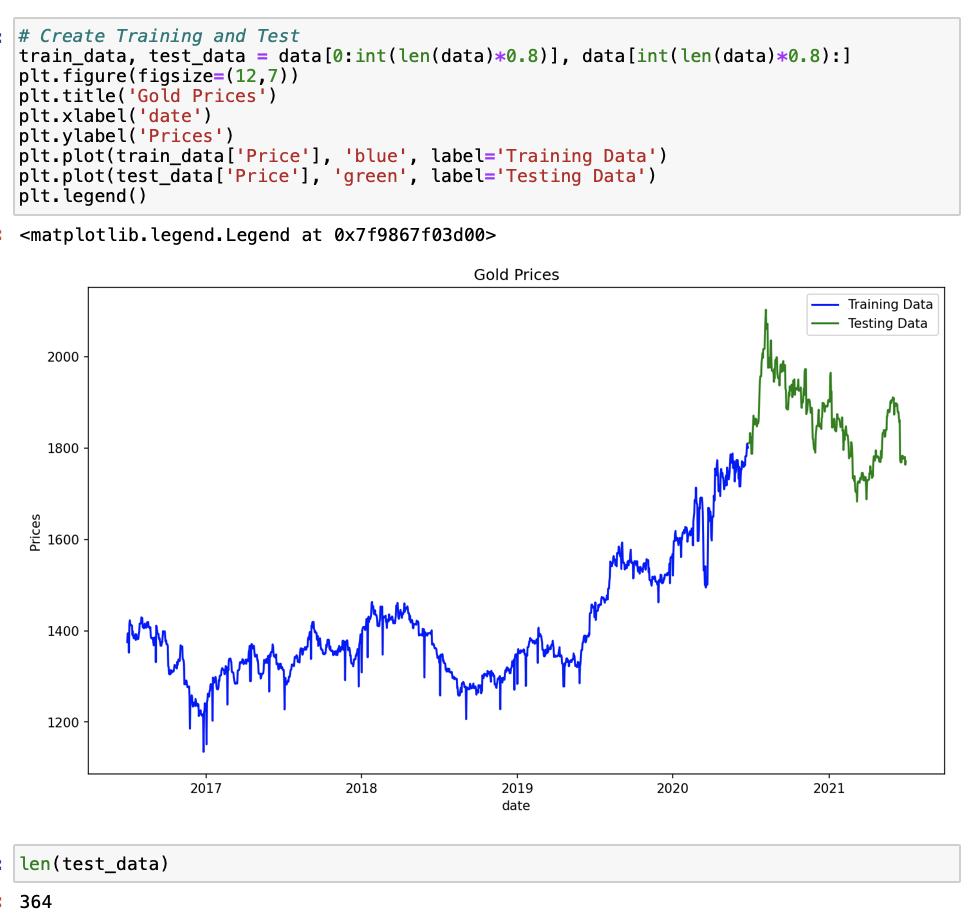
ϕk = corr(xt − Pt,k(xt),xt−k − Pt,k(xt−k))

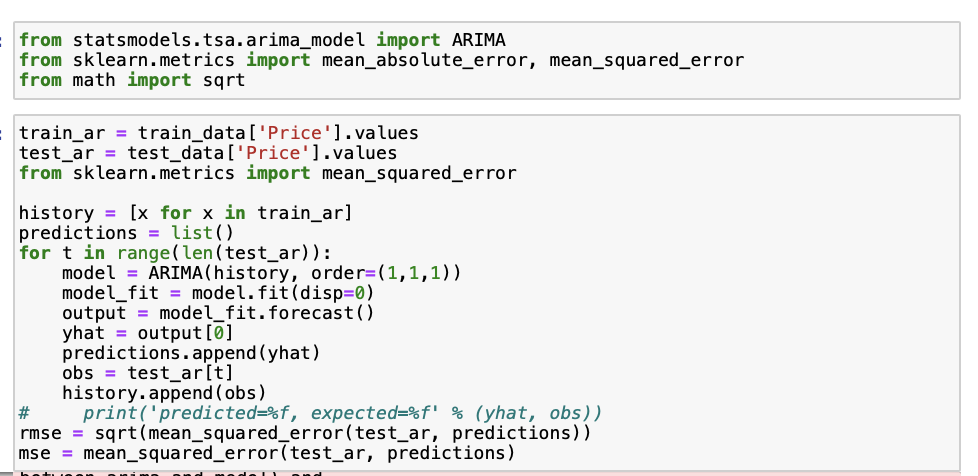
Trong đó corr() là hàm tính hệ số tương quan. Đó là tất cả về PACF. Khá dễ hiểu phải không nào? PACF sẽ có tác dụng tìm ra hệ số bậc tự do p của quá trình tự hồi qui AR(p).

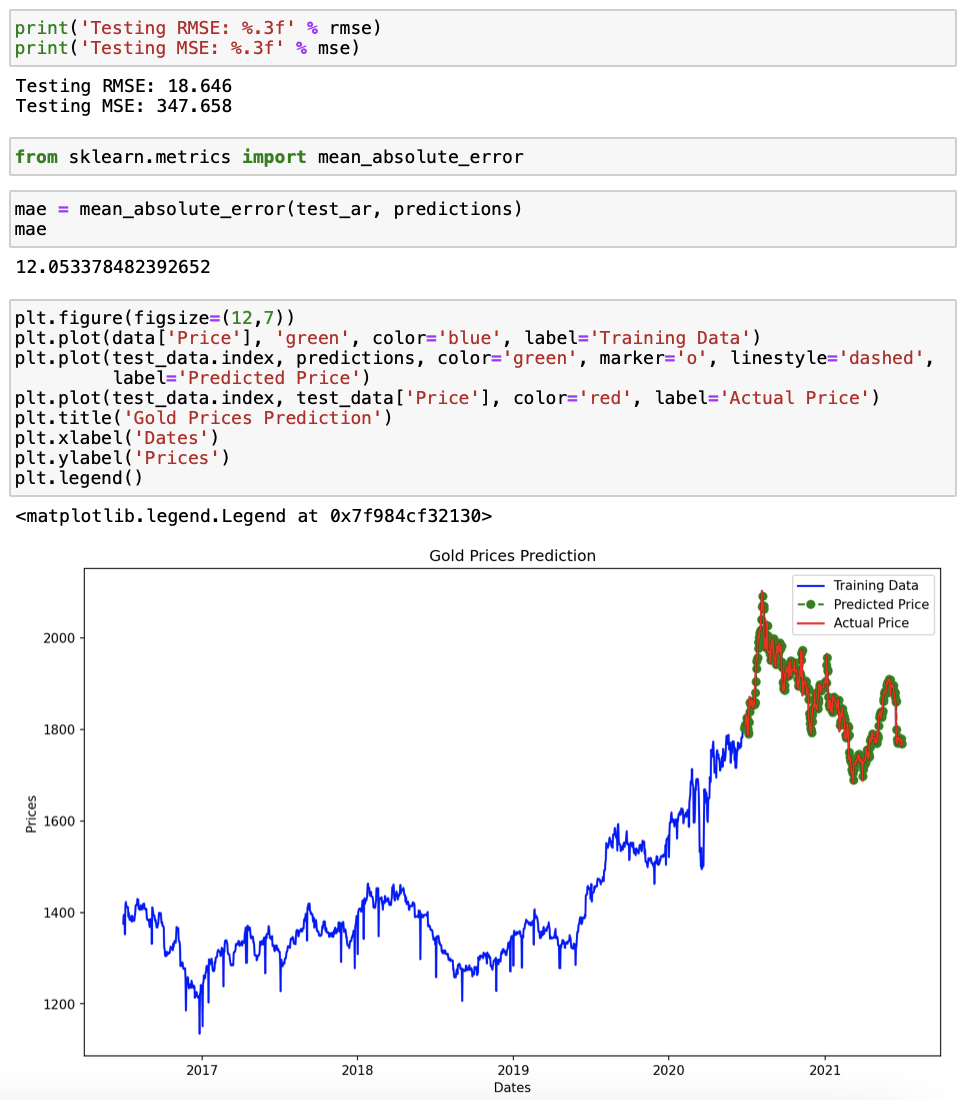
Tương tự như ACF, thông qua một biểu đồ PACF về giá trị các hệ số tương quan riêng phần tương ứng với các độ trễ khác nhau, chúng ta sẽ tìm ra được các bậc tự do p phù hợp. Đó chính là vị trí mà giá trị của hệ số tương quan riêng phần nằm ngoài ngưỡng tin cậy 95% của giả thuyết hệ số tương quan riêng phần bằng 0. Lựa chọn p = 1.

Kiểm tra yếu tố mùa vụ của dữ liệu.

Tiến hành xây dựng mô hình ARIMA theo các thông số đã lựa chọn:

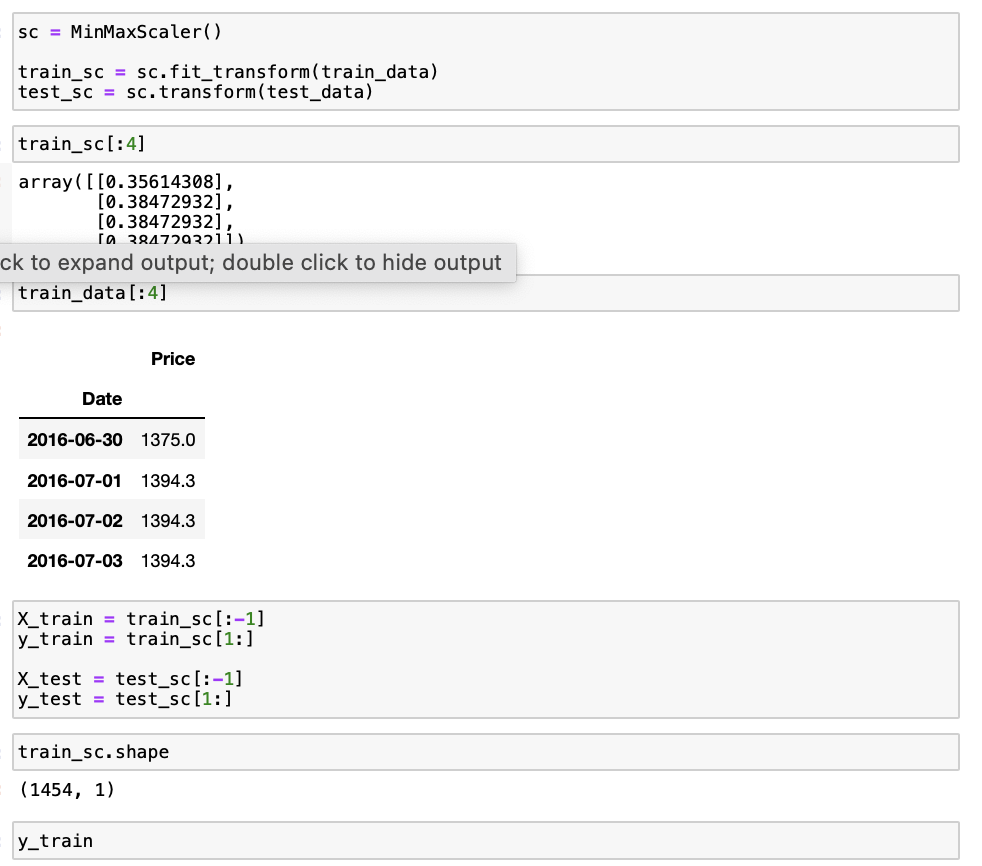




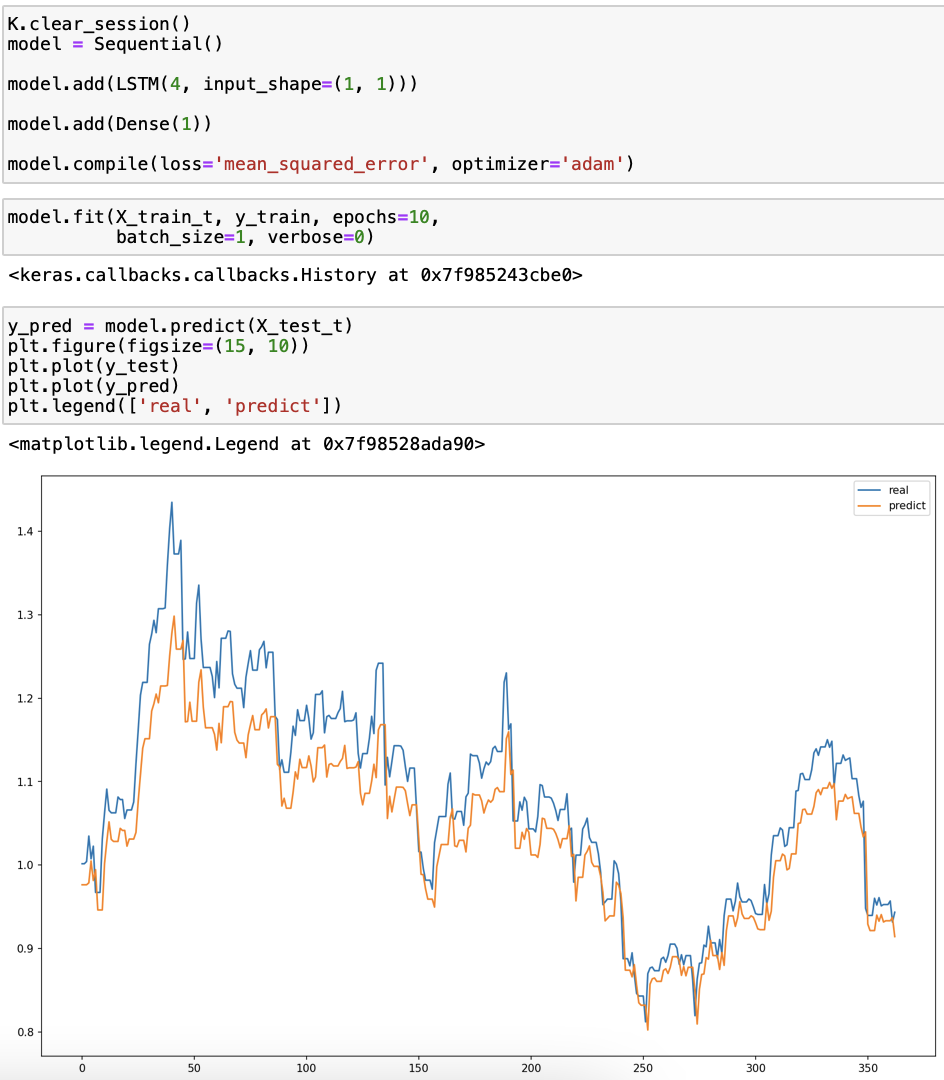


Nhìn biểu đồ ta có thể thấy, đường dự báo bám khá gần so với đường thực tế 🡺 Mô hình có thể phù hợp, tuy nhiên mse lại chưa đạt được ở mức thấp nhất. Áp dụng các mô hình khác để có thể so sánh.

**3.2.2. Mô hình LSTM**

Sử dụng MinMaxScaler để tạo ra array mới từ train\_dat và test\_data đã được phân chia trong quá trình thực hiện mô hình ARIMA.

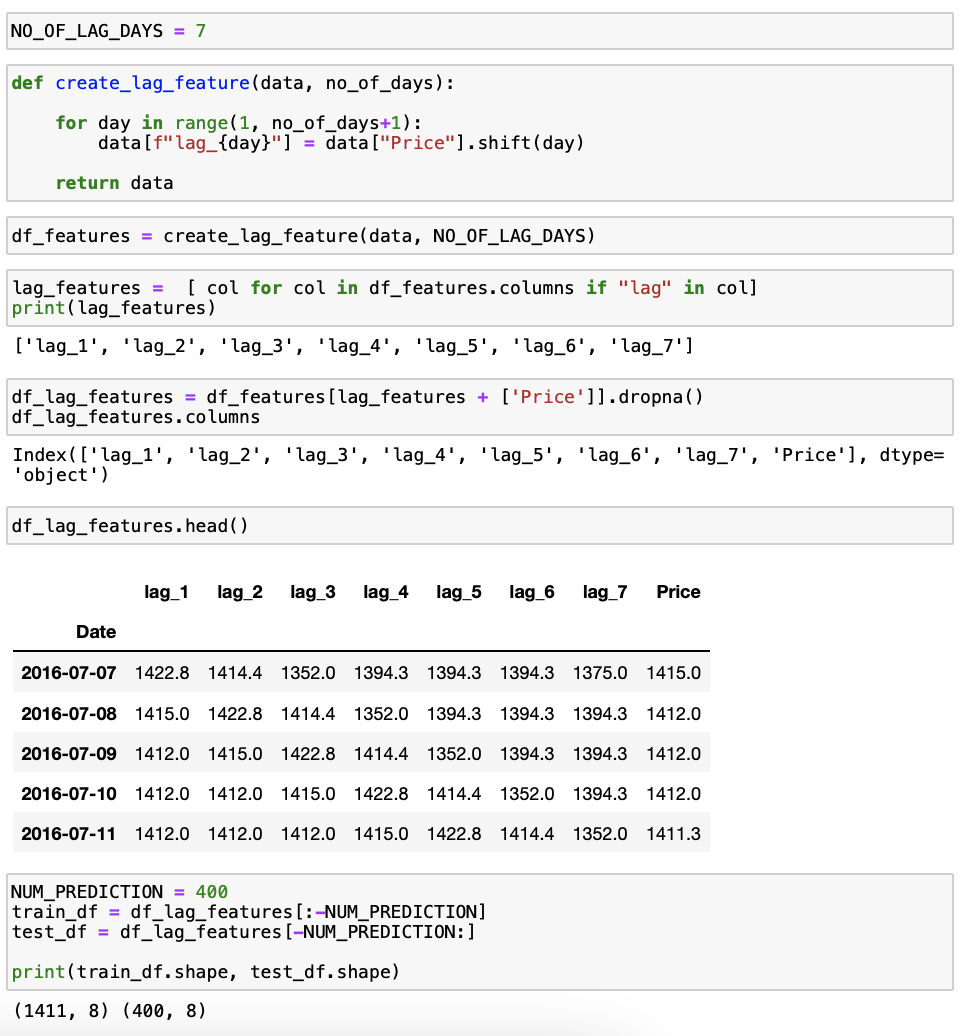
Xây dựng model



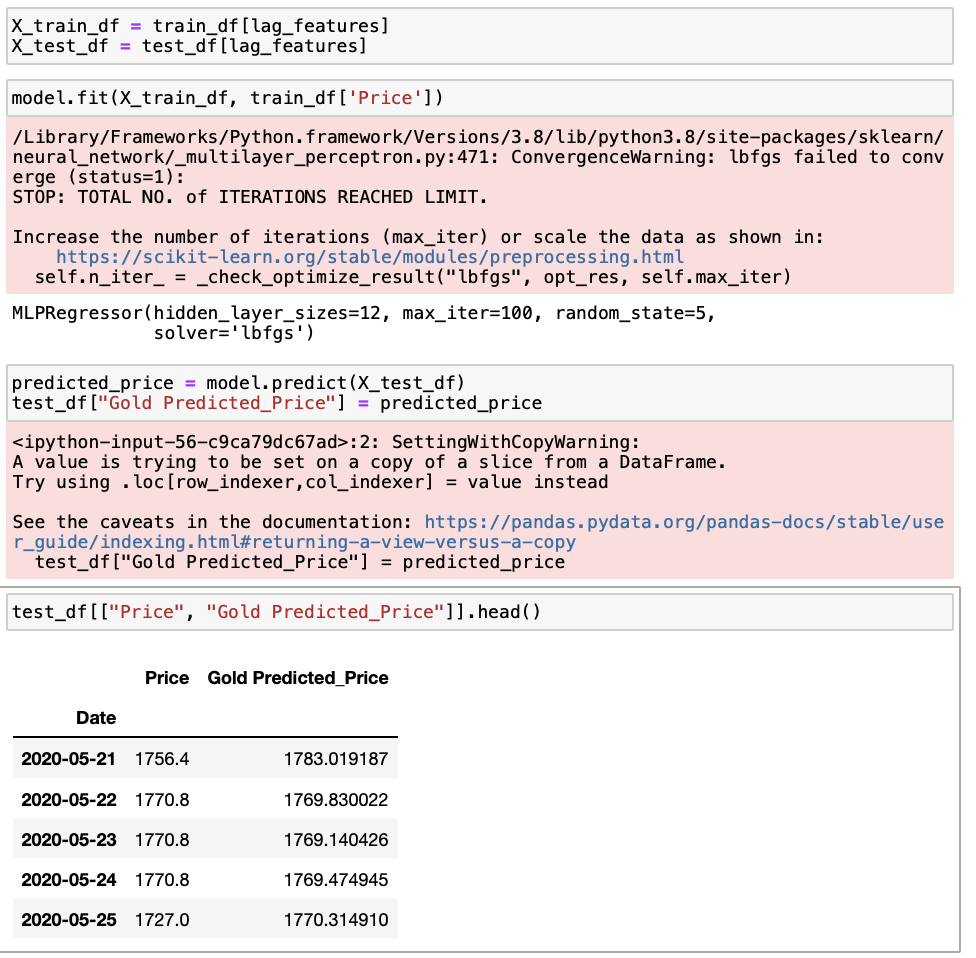
Nhìn chung kết quả dự báo bám gần với đường giá thực tế.

**3.2.3. Mô hình Neural Network ( sử dụng MLP Regressor)**

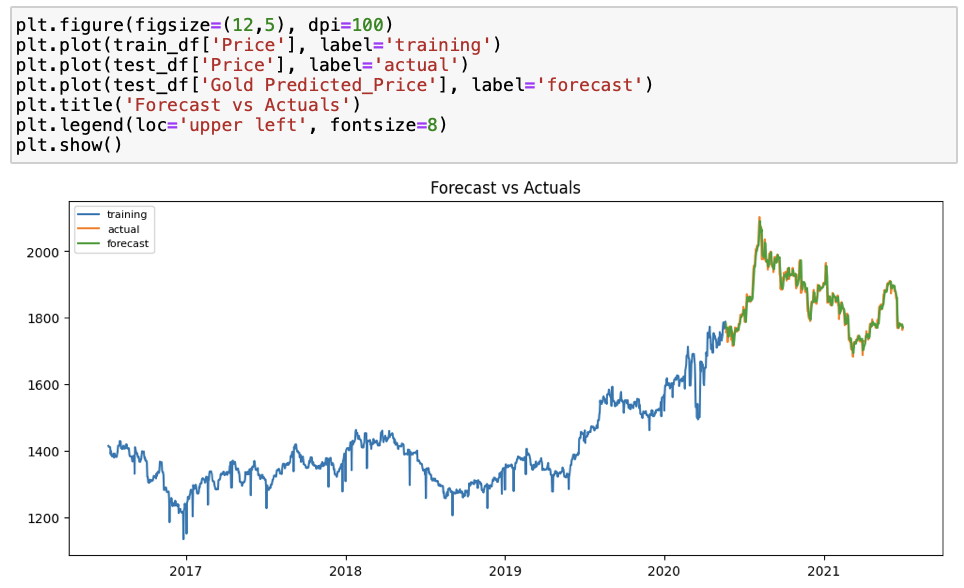
Sử dụng feature Engineering để tạo ra multipe lag features với độ trễ là 7.



Tiến hành xây dựng model từ dữ liêụ train và test đã được xử lý.



Vẽ biểu đồ hiển thị giá dự báo và giá thực tế của model

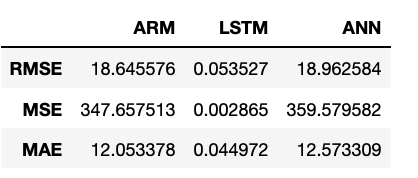


🡺 Giá dự báo và giá thực tế cũng khá gần nhau.

# CHƯƠNG 4: ĐÁNH GIÁ MÔ HÌNH VÀ SO SÁNH

## 4.1. Đánh giá chung các mô hình

Chúng ta có được kết quả so sánh sau khi chạy các mô hình như sau:



Trong 3 mô hình thì LSTM cho kết quả MSE và RMSE khá thấp và tương đối tốt. Nếu chỉ dựa vào mse nhỏ nhất thì LSTM có thể phù hợp nhất.

# Tài liệu tham khảo

Pham Dinh Khanh blog: <https://phamdinhkhanh.github.io/home>

Thông tin tại trang: <https://www.analyticsvidhya.com/blog/2019/12/6-powerful-feature-engineering-techniques-time-series/>

Cũng như các thông tin tìm hiểu được trên các trang web khác.