**ỦY BAN NHÂN DÂN THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC SÀI GÒN**

****

**BÁO CÁO CUỐI KÌ**

**Đề tài: Dự đoán kết quả xổ số kiến thiết**

**MÔN: KHAI PHÁ DỮ LIỆU**

**Giảng viên hướng dẫn: Phạm Hoàng Vương**

**NHÓM 10**

Lê Duy Khanh – 3119560030

Phạm Anh Tuân – 3119560078

Hồ Văn Diện – 3119560004

Nguyễn Thanh Giang – 3119560013

Nguyễn Minh Thắng -  3119560064

**LỜI CẢM ƠN**

Đầu tiên, em xin gửi lời cảm ơn chân thành đến các thầy cô trong khoa Công Nghệ Thông Tin của trường Đại Học Sài Gòn. Đặc biệt, em xin gửi lời cảm ơn sâu sắc đến thầy Phạm Hoàng Vương, giảng viên môn khai phá dữ liệu đã dạy dỗ, truyền đạt những kiến thức quý báu cho em trong suốt thời gian học tập vừa qua. Trong thời gian tham gia lớp học khai phá dữ liệu của thầy, em đã có thêm cho mình nhiều kiến thức bổ ích, tinh thần học tập hiệu quả, nghiêm túc. Đây chắc chắn sẽ là những kiến thức quý báu, là hành trang để em có thể vững bước sau này.

Bộ môn Khai phá dữ liệu là môn học thú vị, vô cùng bổ ích. Đảm bảo cung cấp đủ kiến thức, gắn liền với nhu cầu thực tiễn của sinh viên. Tuy nhiên, do vốn kiến thức còn nhiều hạn chế và khả năng tiếp thu thực tế còn nhiều bỡ ngỡ. Mặc dù em đã cố gắng hết sức nhưng chắc chắn bài tiểu luận khó có thể tránh khỏi những thiếu sót và nhiều chỗ còn chưa chính xác, kính mong thầy xem xét và góp ý để bài tiểu luận của em được hoàn thiện hơn.

**Em xin chân thành cảm ơn Thầy!**

**Nhận xét**

**Chữ ký**

**TP.HCM, Ngày…….Tháng……Năm……**

Contents

[I. Giới thiệu 1](#_Toc134760967)

[1. Lý do chọn đề tài 1](#_Toc134760968)

[2. Mục tiêu của báo cáo 1](#_Toc134760969)

[II. Cơ sở lý thuyết 1](#_Toc134760970)

[1. Khái niệm về dự đoán xổ số 1](#_Toc134760971)

[2. Các phương pháp dự đoán xổ số thông dụng 2](#_Toc134760972)

[3. Các thuật toán phân tích dữ liệu được sử dụng trong dự đoán xổ số 2](#_Toc134760973)

[3.1. Thuật toán K-means clustering (phân cụm K-means). 2](#_Toc134760974)

[3.2. Thuật toán ARIMA(Autoregressive Integrated Moving Average) 8](#_Toc134760975)

[III. Thiết kế phần mềm 12](#_Toc134760976)

[1. Yêu cầu chức năng 12](#_Toc134760977)

[2. Thiết kế giao diện người dùng 13](#_Toc134760978)

[3. Thiết kế cơ sở dữ liệu 13](#_Toc134760979)

[IV. Kết luận 14](#_Toc134760980)

[1. Tổng kết về phần mềm dự đoán kết quả xổ số 14](#_Toc134760981)

[2. Hướng phát triển của phần mềm 14](#_Toc134760982)

[V. Các thuật toán ứng dụng trong phần mềm 15](#_Toc134760983)

# I. Giới thiệu

## 1. Lý do chọn đề tài

* Với sự phát triển của công nghệ và khoa học dữ liệu, việc áp dụng các phương pháp phân tích dữ liệu đang là xu hướng trong nhiều ngành và dự đoán kết quả xổ số cũng là một trong những thứ được hưởng lợi từ các công nghệ này.
* Việc dự đoán kết quả xổ số có thể giúp người chơi có được sự đánh giá chính xác hơn về khả năng trúng thưởng, đồng thời giảm thiểu rủi ro trong việc đầu tư tiền vào các trò chơi xổ số. Do đó, việc phát triển một phần mềm dự đoán kết quả xổ số có thể giúp người dùng tiết kiệm thời gian và tiền bạc trong việc chơi xổ số.

Vì vậy, đề tài "Phần mềm dự đoán kết quả xổ số" là một chủ đề đầy hứa hẹn và có tiềm năng để phát triển trong tương lai.

## 2. Mục tiêu của báo cáo

* Giới thiệu về phương pháp dự đoán kết quả xổ số thông qua việc phân tích dữ liệu và áp dụng các thuật toán máy học. Báo cáo sẽ cung cấp cho độc giả một cái nhìn tổng quan về phương pháp này.

Cụ thể, mục tiêu của báo cáo bao gồm:

* Giới thiệu các khái niệm cơ bản liên quan đến dự đoán kết quả xổ số, bao gồm các phương pháp dự đoán thông dụng và thuật toán phân tích dữ liệu được sử dụng trong việc dự đoán kết quả xổ số.
* Trình bày chi tiết về thiết kế phần mềm dự đoán kết quả xổ số, bao gồm yêu cầu chức năng, thiết kế giao diện người dùng và cơ sở dữ liệu.
* Tổng kết về phần mềm dự đoán kết quả xổ số và đề xuất hướng phát triển của phần mềm trong tương lai.

Từ đó, báo cáo mong muốn giúp cho độc giả hiểu rõ hơn về phương pháp dự đoán kết quả xổ số thông qua phân tích dữ liệu và áp dụng các thuật toán máy học, đồng thời cung cấp cho độc giả một giải pháp phần mềm tiện ích để dự đoán kết quả xổ số một cách hiệu quả và chính xác.

# II. Cơ sở lý thuyết

## 1. Khái niệm về dự đoán xổ số

* Dự đoán xổ số là quá trình sử dụng các phương pháp và công nghệ thống kê, phân tích số liệu và mô hình hóa để dự đoán các số sẽ xuất hiện trong kết quả xổ số trong tương lai. Dự đoán xổ số thường được sử dụng trong việc đánh giá và tính toán xác suất để tìm ra các cơ hội để thắng lớn trong các trò chơi xổ số. Các công nghệ thống kê và phân tích số liệu được sử dụng để tạo ra các mô hình dự đoán, bao gồm các mô hình hồi quy, mô hình chuỗi thời gian, mô hình phân tích yếu tố và các thuật toán máy học khác. Các phương pháp dự đoán xổ số được sử dụng phổ biến trong các hoạt động đánh bạc và trò chơi xổ số, và đã trở thành một ngành công nghiệp đầy tiềm năng.

## 2. Các phương pháp dự đoán xổ số thông dụng

Các phương pháp dự đoán xổ số thông dụng bao gồm:

* Phân tích thống kê: Dựa trên các số xuất hiện trong quá khứ, phân tích thống kê sẽ tìm ra các xu hướng và chu kỳ xuất hiện của các số đó và từ đó đưa ra dự đoán.
* Mô hình chuỗi thời gian : Dựa trên các số xuất hiện trong quá khứ và thời gian, mô hình chuỗi thời gian sẽ tìm ra các mẫu và chu kỳ xuất hiện của các số đó để đưa ra dự đoán.
* Các thuật toán máy học: Sử dụng các thuật toán máy học như mạng neuron nhân tạo, học máy, học sâu để học và dự đoán kết quả xổ số dựa trên dữ liệu lịch sử.

Kết hợp các phương pháp trên: Các phương pháp trên có thể được kết hợp để tăng tính chính xác và hiệu quả của dự đoán xổ số.

## 3. Các thuật toán phân tích dữ liệu được sử dụng trong dự đoán xổ số

3.1. Thuật toán K-means clustering (phân cụm K-means).

* Giới thiệu:

K-means là một thuật toán phân cụm đơn giản thuộc loại học không giám sát(tức là dữ liệu không có nhãn) và được sử dụng để giải quyết bài toán phân cụm. Ý tưởng của thuật toán phân cụm k-means là phân chia 1 bộ dữ liệu thành các cụm khác nhau. Trong đó số lượng cụm được cho trước là k. Công việc phân cụm được xác lập dựa trên nguyên lý: Các điểm dữ liệu trong cùng 1 cụm thì phải có cùng 1 số tính chất nhất định. Tức là giữa các điểm trong cùng 1 cụm phải có sự liên quan lẫn nhau. Đối với máy tính thì các điểm trong 1 cụm đó sẽ là các điểm dữ liệu gần nhau.

Thuật toán phân cụm k-means thường được sử dụng trong các ứng dụng cỗ máy tìm kiếm, phân đoạn khách hàng, thống kê dữ liệu và có thể áp dụng trong rất nhiều lĩnh vực như:

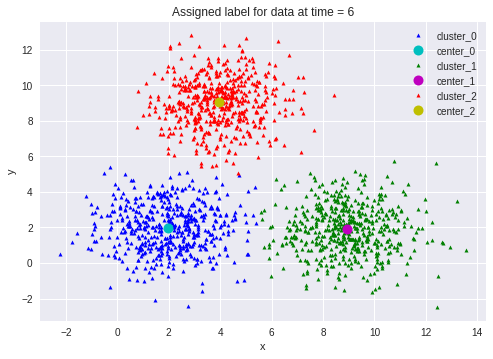
+ Marketing: Xác định các nhóm khách hàng (khách hàng tiềm năng, khách hàng giá trị, phân loại và dự đoán hành vi khách hàng,…) sử dụng sản phẩm hay dịch vụ của công ty để giúp công ty có chiến lược kinh doanh hiệu quả hơn;

+ Biology: Phận nhóm động vật và thực vật dựa vào các thuộc tính của chúng;

+ Libraries:  Theo dõi độc giả, sách, dự đoán nhu cầu của độc giả…;

+ Insurance, Finance: Phân nhóm các đối tượng sử dụng bảo hiểm và các dịch vụ tài chính, dự đoán xu hướng (trend) của khách hàng, phát hiện gian lận tài chính (identifying frauds);

+ WWW:  Phân loại tài liệu (document classification);  phân loại người dùng web (clustering weblog);…

****

* Định nghĩa:

Thuật toán phân cụm k-means là một phương pháp được sử dụng trong phân tích tính chất cụm của dữ liệu. Nó đặc biệt được sử dụng nhiều trong khai phá dữ liệu và thống kê. Nó phân vùng dữ liệu thành k cụm khác nhau. Giải thuật này giúp chúng ta xác định được dữ liệu của chúng ta nó thực sử thuộc về nhóm nào.

Để các bạn dễ hình dung ứng dụng của thuật toán. Chúng ta hãy quan sát một ví dụ thực tế như sau:

Trong các mô hình kinh doanh, doanh nghiệp sẽ chia nhỏ tệp khách hàng ra thành những nhóm đối tượng khác nhau để có thể áp dụng những chiến lược kinh doanh cụ thể cho từng nhóm đối tượng. Điều này giúp cho khách hàng được tiếp cận với các sản phẩm thật sự phù hợp với bản thân họ. Sự phù hợp đó sẽ kéo doanh số của chúng ta tăng lên. Vấn đề đặt ra là làm sao có thể chia nhỏ tệp khách hàng đó ra khi mà số lượng hóa đơn là rất lớn và chúng ta không thể ngồi để phân tích từng vị khách.

Và mục tiêu của các thuật toán phân cụm là từ tập dữ liệu khổng lồ đó. Làm sao chúng ta biết có những nhóm dữ liệu đặc trưng nào trong đó? Từng dữ liệu trong đó thuộc vào nhóm nào? Đó là cái mà thuật toán phân cụm của chúng ta cần đi tìm câu trả lời.

- Ý tưởng:

* Khởi tạo K điểm dữ liệu trong bộ dữ liệu và tạm thời coi nó là tâm của các cụm dữ liệu của chúng ta.
* Với mỗi điểm dữ liệu trong bộ dữ liệu, tâm cụm của nó sẽ được xác định là 1 trong K tâm cụm gần nó nhất.
* Sau khi tất cả các điểm dữ liệu đã có tâm, tính toán lại vị trí của tâm cụm để đảm bảo tâm của cụm nằm ở chính giữa cụm.
* Bước 2 và bước 3 sẽ được lặp đi lặp lại cho tới khi vị trí của tâm cụm không thay đổi hoặc tâm của tất cả các điểm dữ liệu không thay đổi.
* Trên thực tế, sẽ có 1 vài lưu ý cần phải giải quyết khi áp dụng thuật toán k-means:
* Lựa chọn số lượng cụm: Chỉ việc lựa chọn số cụm k đã có thể tách thành 1 bài toán riêng. Không có 1 con số k nào là hợp lý cho tất cả các bài toán. Bạn có thể đọc hiểu tập dữ liệu của mình để xác định xem trong đó có thể có bao nhiêu cụm? Nhưng không phải lúc nào bạn cũng có thể làm thế. Cách làm duy nhất là bạn hãy thử với từng giá trị k=1,2,3,4,5,… để xem kết quả phân cụm thay đổi như thế nào. Một số nghiên cứu cho thấy việc thay đổi k sẽ có hiệu quả nhưng sẽ dừng lại ở 1 con số nào đó. Như vậy bạn hoàn toàn có thể thử xem dữ liệu của mình tốt với giá trị k nào đó.
* Khởi tạo k vị trí ban đầu: Bằng cách nào đó, hãy có gắng khởi tạo k tâm cụm này phân bố đồng đều trên không gian của bộ dữ liệu. Điều đó có thể làm khi bạn có thể xác định được không gian và tính chất của dữ liệu. Nhưng ít nhất, các tâm cụm mà bạn khởi tạo cũng đừng quá gần nhau, cũng đừng trùng nhau.

Còn 1 cách cuối cùng là bạn sẽ chạy thuật toán nhiều lần để lấy kết quả tốt nhất trong các lần chạy đó. Với điều kiện là bạn khởi tạo tâm của k cụm ngẫu nhiên.

* Về vấn đề tính dừng(hội tụ): Đối với những trường hợp dữ liệu phức tạp, thuật toán k-means sẽ rất lâu hoặc không bao giờ hội tụ. Tức là sẽ không bao giờ xác định được tâm cụm cố định để kết thúc bài toán. Hoặc là phải chạy qua rất nhiều bước lặp. Trong những trường hợp như vậy, thay vì phải tìm được k tâm cụm cố định thì ta sẽ dừng bài toán khi sự thay đổi ở một con số chấp nhận được. Tức là giữa hai lần cập nhật tâm cụm thì chênh lệch vị trí giữa tâm cũ và mới nhỏ hơn một số delta cho phép nào đó.

**\* Một số ảnh hưởng đến thuật toán K-means**

*- Ảnh hưởng của outlier*

- Outlier là gì?

+ Hiểu đơn giản thì Outliers là một hoặc nhiều cá thể khác hẳn đối với các thành viên còn lại của nhóm. Sự khác biệt này có thể dựa trên nhiều tiêu chí khác nhau như giá trị hay thuộc tính. Ví dụ về outlier có thể như là nhiễu trong các cảm biến hay lỗi trong quá trình nhập liệu của người dùng ảnh hưởng đến chất lượng của dữ liệu.

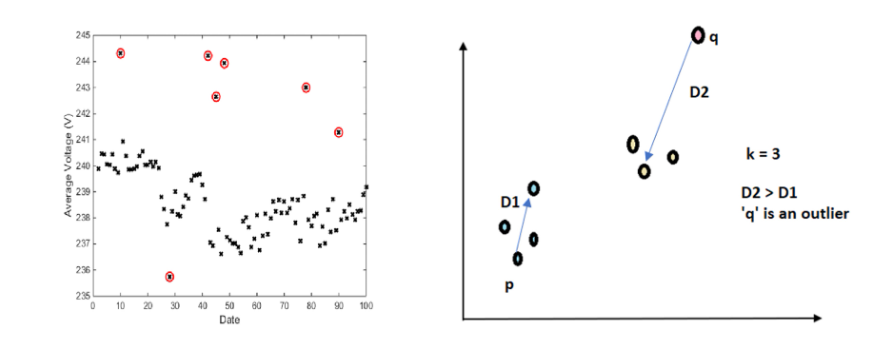
- Xem xét ảnh hường

- K-means nhạy cảm với các điểm outlier, ví dụ: Các điểm dữ liệu outlier ảnh hưởng lớn đến kết quả của việc phân cụm:

+ Các điểm dữ liệu outlier có khoảng cách đến các điểm dữ liệu chuẩn rất lớn.

+ Phân bố của các điểm outliner rất khác so với các điểm dữ liệu chuẩn

+ Nhiễu hoặc lỗi của dữ liệu được thể hiện trong các điểm outlier

****

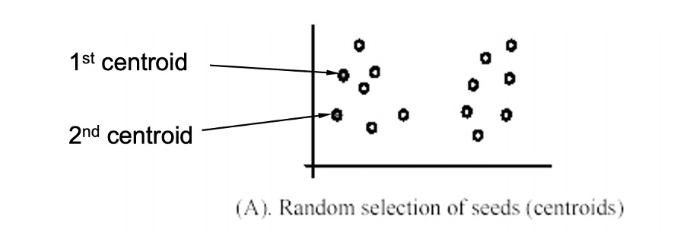
- Khắc phục outlier

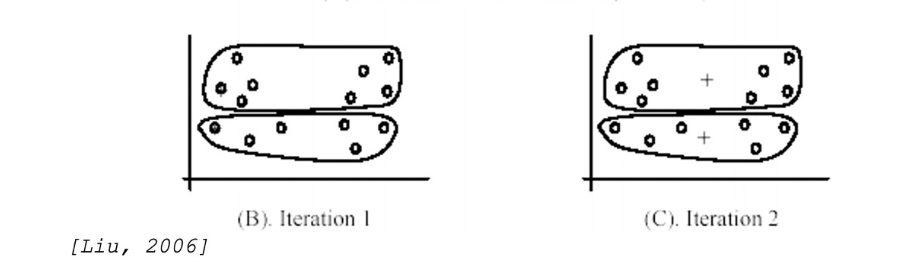
+ Outlier removal: Có thể loại bỏ các điểm dữ liệu xa đáng kể so với điểm trung tâm (centroid) của các cluster so với các điểm dữ liệu khác. Việc loại bỏ có thể được thực hiện trước hoặc trong khi phân cụm.

+ Random sampling: Thay vì phân cụm toàn bộ tập dữ liệu, chúng ta sẽ lấy ngẫu nhiên tập con S từ tập dữ liệu huấn luyện. S được sử dụng để phân cụm, tập S lúc này sẽ có ít các điểm outlier hơn tập dữ liệu gốc. Sau khi phân cụm xong, tập dữ liệu còn lại sẽ được gán vào các cụm đã học được

- Ảnh hưởng của việc khởi tạo trung tâm

Chất lượng của K-means phụ thuộc vào việc khởi tạo các điểm centroid

****

****

- Giải pháp 1: Lặp lại nhiều lần thuật toán K-means:

+ Mỗi lần chạy lại thuật toán K-means sẽ khởi tạo các điểm centroid khác nhau

+ Sau quá trình học, tiến hành gộp các kết quả từ các lần chạy thành kết quả cuối cùng

- Giải pháp 2: Thuật toán K-means++ : Để tìm ra cụm tốt nhất, chúng ta có thể lần lượt khởi tại các điểm trung tâm từ tập D tuần tự như sau:

+ Lấy ngẫu nhiên điểm centroid đầu tiên m1

+ Lấy điểm centroid tiếp theo là điểm xa nhất so với m1

..

+ Lấy điểm centroid thứ i (Mi) là điểm xa nhất so với {M1,...,Mi-1}

…

+ Bằng cách này K-means sẽ hội tụ về gần kết quả tối ưu (Arthur, D.; Vassilvitskii, 2007)

**\* Ưu điểm và nhược điểm của thuật toán K-means clustering**  
 Phân cụm (clustering) là quá trình phân chia các đối tượng vào các nhóm có tính chất tương tự nhau. K-mean là một trong những phương pháp phổ biến để phân cụm dữ liệu. Dưới đây là một số ưu điểm và nhược điểm của phương pháp K-mean:

**- Ưu điểm:**

+ Dễ hiểu và triển khai: Phương pháp K-mean là một trong những phương pháp phân cụm đơn giản và dễ triển khai nhất.

+ Thời gian tính toán nhanh: Vì phương pháp này chỉ cần tính toán khoảng cách giữa các điểm dữ liệu và các trung tâm cụm, nên thời gian tính toán sẽ nhanh hơn so với một số phương pháp khác.

+ Phù hợp với dữ liệu lớn: Phương pháp K-mean có thể được áp dụng với tập dữ liệu lớn.

+ Tính linh hoạt: Phương pháp K-mean có thể được sử dụng với nhiều loại dữ liệu khác nhau, bao gồm cả dữ liệu số và dữ liệu không số.

+ Đảm bảo hội tụ trong thời gian đa thức [Manthey & Röglin, JACM, 2011]

**- Nhược điểm:**

+ Yêu cầu xác định số lượng cụm trước: Phương pháp K-mean yêu cầu người dùng phải xác định trước số lượng cụm cần phân chia, điều này có thể gây khó khăn đối với người dùng nếu không biết rõ về dữ liệu.

+ Nhạy cảm với giá trị khởi tạo ban đầu: Kết quả của phương pháp K-mean có thể thay đổi phụ thuộc vào giá trị khởi tạo ban đầu. Điều này có thể dẫn đến việc thu được kết quả phân cụm không tối ưu.

+ Chỉ áp dụng được cho dữ liệu có tính chất tuyến tính: Phương pháp K-mean chỉ thích hợp với dữ liệu có tính chất tuyến tính, điều này có nghĩa là các cụm được xác định sẽ có hình dạng tròn hoặc cầu. Việc áp dụng K-mean với dữ liệu phi tuyến sẽ cho kết quả không tốt.

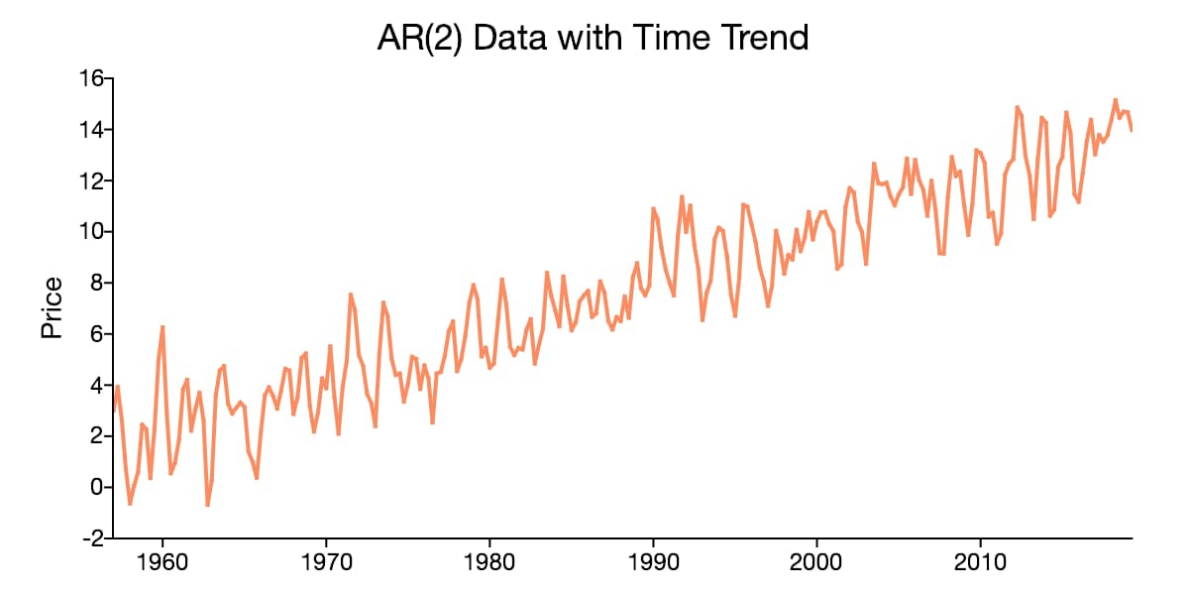
+ phải chỉ ra số lượng cluster và yêu cầu CSDL cần phân nhóm phải xác định được tâm. Thuật toán này không phù hợp với việc khai phá các dữ liệu gồm các cluster có hình dạng không lồi và KMeans hay gặp lỗi với các dữ  liệu có phần tử ngoại lai (outliers).

+ Đối với các tập dữ liệu có số chiều lớn dữ liệu có nhiều phần tử nhiễu như các tập dữ liệu biểu hiện gien thì giải thuật Kmeans thực hiện không đạt hiệu quả cao (Dipti 2015).

3.2. Thuật toán ARIMA(Autoregressive Integrated Moving Average)

1. Định nghĩa

* 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 quy 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 quy 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 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.

****

Đồ thị về yếu tố xu hướng trong chuỗi thời gian của chuỗi giá.

• 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:

1. Vai trò của thuật toán ARIMA :

- Thuật toán ARIMA được sử dụng để mô hình hóa và dự đoán các chuỗi thời gian. Nó là viết tắt của AutoRegressive Integrated Moving Average và kết hợp giữa ba phương pháp khác nhau để xây dựng mô hình.

- Cụ thể, ARIMA kết hợp ba yếu tố sau đây:

• Autoregressive (AR): Mô hình phụ thuộc vào các giá trị trước đó của chuỗi thời gian.

• Integration (I): Mô hình sử dụng việc chuyển đổi chuỗi thời gian không ổn định thành chuỗi thời gian ổn định.

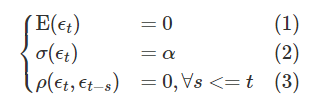
• Moving Average (MA): Mô hình dựa trên các sai số dự báo trước đó để dự đoán giá trị tiếp theo.

- Tùy thuộc vào cách kết hợp các yếu tố này, ARIMA có thể được sử dụng để xây dựng mô hình cho các loại chuỗi thời gian khác nhau, bao gồm chuỗi dừng, chuỗi không dừng và chuỗi có xu hướng. Các mô hình ARIMA được sử dụng rộng rãi trong các lĩnh vực như tài chính, kinh tế học, dự báo sản xuất và chính trị, để dự đoán các biến số kinh tế và xã hội, giúp các nhà quản lý và chính phủ đưa ra các quyết định hiệu quả hơn.

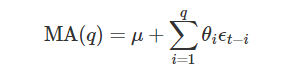
1. Lý thuyết về mô hình ARIMA

Lý thuyết: 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 Integrated Moving Average. Mô hình sẽ biểu diễn phương trình hồi quy 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 quy bao gồm tập hợp các độ trễ của biến hiện tại. Độ trễ bậc chính là giá trị lùi về quá khứ bước thời gian của chuỗi. Độ trễ dài hoặc ngắn trong quá trình AR(p) phụ thuộc vào tham số trễ. Cụ thể, quá trình AR(p) của chuỗi xt được biểu diễn như bên dưới:  
  ****
* Moving average: Quá trình trung bình trượt được hiểu là quá trình dịch chuyển hoặc thay đổi giá trị trung bình của chuỗi theo thời gian. Do chuỗi của chúng ta được giả định là đừng nên quá trình thay đổi trung bình dường như là một chuỗi nhiễu trắng. Quá trình moving average sẽ tìm mối liên hệ về mặt tuyến tính giữa các phần tử ngẫu nhiên (stochastic term). Chuỗi này phải là một chuỗi nhiễu trắng thỏa mãn các tính chất:  
  ****

Vế (1) có nghĩa rằng kỳ 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 kỳ 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 ∊t~WN(0,𝟃^2). Nhiễu trắng là một thành phần ngẫu nhiên thể hiện cho yếu tố không thể dự báo của model và không có tính quy luật. Quá trình trung bình trượt được biểu diễn theo nhiễu trắng như sau:

****

Quá trình này có thể được biểu diễn theo dịch chuyển trễ - backshift operator

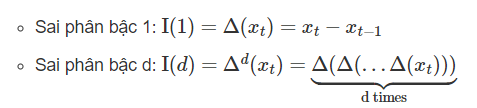
 như sau:

****

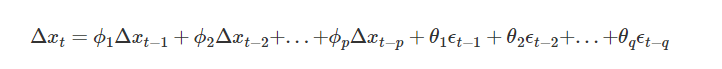
Như vậy bạn đọc đã hình dung ra moving average là gì rồi chứ? Về mặt ý tưởng thì đó chính là quá trình hồi quy tuyến tính của giá trị hiện tại theo các giá trị hiện tại và quá khứ của sai số nhiễu trắng (white noise error term) đại diện cho các yếu tố shock ngẫu nhiên, những sự thay đổi không lường trước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

 của chuỗi được thực hiện như sau:

****

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:

****

Trong đó ∆xt là giá trị sai phân bậc d và ∊t 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. Đó là lý thuyết của kinh tế lượng. Còn theo trường phái machine learning thì tôi chỉ cần quan tâm đến làm sao để lựa chọn một mô hình có sai số dự báo là nhỏ nhất. Tiếp theo chúng ta sẽ sử dụng package vnquant, một package được tôi viết để hỗ trợ cộng đồng khai thác dữ liệu chứng khoán thuận tiện hơn.

# III. Thiết kế phần mềm

## 1. Yêu cầu chức năng

\* Dưới đây là một số chức năng cơ bản có thể có trong ứng dụng dự đoán kết quả xổ số:

-    Hiển thị kết quả xổ số mới nhất: thống kê được kết quả 90 ngày gần đây nhất  
 -    Dự đoán kết quả xổ số: Bằng cách sử dụng các thuật toán Machine Learning, ứng dụng có thể dự đoán kết quả xổ số trong tương lai.

-    Nhập số dự đoán: Người dùng có thể nhập các số mà họ muốn dự đoán vào ứng dụng.

-    Hiển thị kết quả dự đoán: Sau khi nhập số dự đoán, ứng dụng sẽ hiển thị kết quả dự đoán tương ứng.

-    Lưu số dự đoán: Người dùng có thể lưu các số dự đoán của mình để theo dõi và so sánh với kết quả thực tế.

-   Cập nhật kết quả xổ số: Ứng dụng có thể tự động cập nhật kết quả sổ số mới nhất theo khoảng thời gian nhất định.

## 2. Thiết kế giao diện người dùng

- Giao diện người dùng của phần mềm cần được thiết kế sao cho đơn giản, dễ sử dụng và thân thiện với người dùng. Các chức năng cần được sắp xếp một cách logic và trực quan trên giao diện để người dùng có thể dễ dàng tương tác và sử dụng phần mềm một cách thuận tiện.

- Giao diện của phần mềm cần hiển thị thông tin cần thiết về các số liệu và kết quả dự đoán một cách rõ ràng, dễ hiểu để người dùng có thể dễ dàng theo dõi và đánh giá kết quả.

## 3. Thiết kế cơ sở dữ liệu

- Cơ sở dữ liệu của phần mềm cần được thiết kế sao cho có thể lưu trữ và quản lý các thông tin cần thiết về các kết quả xổ số, các số đã trúng và các thông tin liên quan khác. Cơ sở dữ liệu cần được cấu trúc rõ ràng, đơn giản và dễ quản lý.

- Cơ sở dữ liệu cần có các tính năng hỗ trợ truy vấn và phân tích dữ liệu để phục vụ cho quá trình dự đoán kết quả. Các thao tác truy vấn cần được tối ưu hóa để đảm bảo tốc độ truy vấn nhanh và hiệu quả.

Tóm lại, thiết kế phần mềm dự đoán kết quả xổ số cần đáp ứng các yêu cầu chức năng, thiết kế giao diện người dùng đơn giản, thân thiện và thiết kế cơ sở dữ liệu

# IV. Kết luận

## 1. Tổng kết về phần mềm dự đoán kết quả xổ số

- Tổng kết lại, công nghệ ngày càng phát triển và ứng dụng của nó trong lĩnh vực xổ số đã đem lại nhiều tiện ích cho người chơi. Các ứng dụng dự đoán kết quả xổ số đã giúp người chơi có thêm cơ hội để chiến thắng và giảm thiểu rủi ro khi đặt cược.

- Tuy nhiên, cần lưu ý rằng các phương pháp dự đoán kết quả xổ số còn tồn tại nhiều hạn chế, không đảm bảo tính chính xác và hiệu quả tuyệt đối. Do đó, người chơi cần hiểu rõ các ưu điểm và nhược điểm của từng phương pháp, cân nhắc trước khi sử dụng.

- Để tăng tính chính xác và hiệu quả của ứng dụng dự đoán kết quả xổ số, cần có sự đẩy mạnh công tác nghiên cứu và phát triển các phương pháp mới, kết hợp sử dụng trí tuệ nhân tạo và máy học. Đồng thời, cần có sự hợp tác giữa các doanh nghiệp, các nhà nghiên cứu, các chuyên gia trong lĩnh vực xổ số để xây dựng và phát triển các sản phẩm ứng dụng dự đoán kết quả xổ số tốt nhất, đáp ứng nhu cầu của người chơi.

- Tóm lại, ứng dụng dự đoán kết quả xổ số là một tiện ích hữu ích cho người chơi xổ số. Tuy nhiên, cần hiểu rõ các hạn chế và sử dụng đúng phương pháp để đảm bảo tính chính xác và hiệu quả. Cần có nhiều nghiên cứu và phát triển để nâng cao tính chính xác và hiệu quả của ứng dụng dự đoán kết quả xổ số trong tương lai.

## 2. Hướng phát triển của phần mềm

\* Một số hướng phát triển có thể thực hiện

* Cải thiện độ chính xác: Để đạt được độ chính xác cao hơn, phần mềm có thể được cải thiện bằng cách sử dụng các thuật toán phân tích dữ liệu mới và tối ưu hóa mô hình máy học hiện tại.
* Tăng tính tiện dụng: Phần mềm có thể được tăng tính tiện dụng bằng cách cung cấp thêm các tính năng mới cho người dùng, chẳng hạn như cập nhật kết quả xổ số tự động, thông báo kết quả qua email hoặc tin nhắn.
* Tích hợp trí tuệ nhân tạo: Phần mềm có thể được tích hợp với các công nghệ trí tuệ nhân tạo như học sâu và học máy để cải thiện độ chính xác và độ tin cậy của kết quả dự đoán.
* Cải thiện giao diện người dùng: Giao diện người dùng của phần mềm có thể được cải thiện để dễ sử dụng và trực quan hơn, bao gồm các tính năng tương tác và đồ họa trực quan để giúp người dùng hiểu rõ hơn về kết quả dự đoán.

# **V. Các thuật toán ứng dụng trong phần mềm**

Dữ liệu ở trong file XSMB.csv gồm nhiều dòng, mỗi dòng bao gồm 3 giá trị được cách bởi dấu phẩy là ["Ngày","Đặc biệt","Giải nhất"] với "Ngày" được định dạng như sau "Ngày-Tháng-Năm" ví dụ "25-01-2023", "Đặc biệt" và "Giải nhất" là một số có 5 chữ số chứa thông tin giải thưởng

*Linear Regression*

Để áp dụng thuật toán Linear Regression cho dữ liệu trên, bạn có thể làm như sau:

Chuẩn bị dữ liệu:

Đọc file dữ liệu CSV và chuyển đổi cột "Ngày" sang định dạng datetime.

Tách dữ liệu thành hai biến độc lập (input) và phụ thuộc (output) để đưa vào mô hình Linear Regression. Trong trường hợp này, ta có thể sử dụng cột "Ngày" và "Giải đặc biệt" để dự đoán "Giải nhất".

Xây dựng mô hình Linear Regression:

Sử dụng thư viện scikit-learn để xây dựng mô hình Linear Regression.

Tách dữ liệu thành tập huấn luyện và tập kiểm tra.

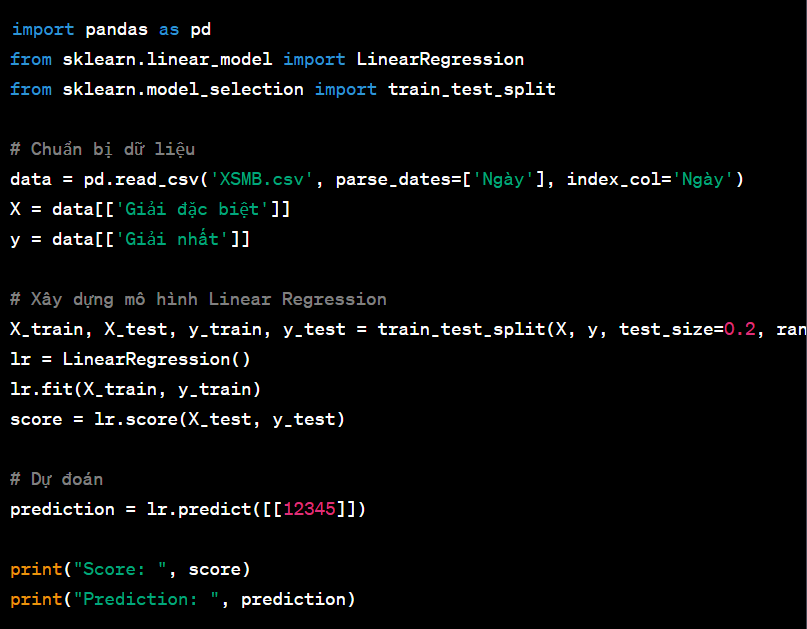
Sử dụng tập huấn luyện để huấn luyện mô hình Linear Regression.

Sử dụng tập kiểm tra để đánh giá độ chính xác của mô hình.

Dự đoán:

Sau khi đã xây dựng và huấn luyện mô hình Linear Regression, ta có thể sử dụng nó để dự đoán giải thưởng "Giải nhất" cho các ngày tiếp theo.

Dưới đây là một ví dụ code cho quá trình áp dụng thuật toán Linear Regression cho dữ liệu này bằng Python:



Trong đó:

Dòng 3-4: Đọc file dữ liệu CSV và tách dữ liệu thành X (độc lập) và y (phụ thuộc).

Dòng 7-10: Tách dữ liệu thành tập huấn luyện và tập kiểm tra.

Dòng 12-13: Xây dựng mô hình Linear Regression và huấn luyện trên tập huấn luyện.

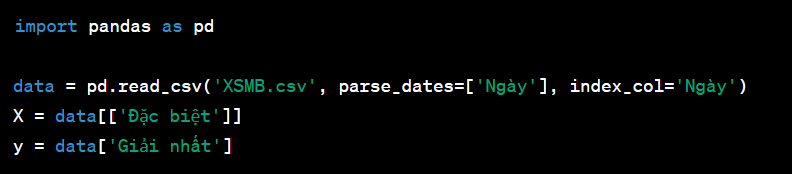
Dòng 14: Đánh giá độ chính xác của mô hình trên tập kiểm tra.

Dòng 17: Dự đoán giải thưởng "Giải nhất" cho một giải đặc biệt có giá trị là 12345.

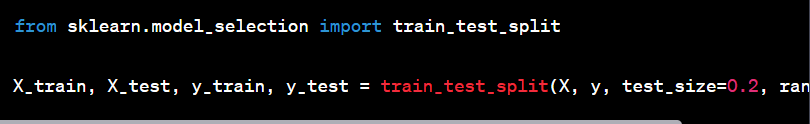
*Decision Tree*

Để áp dụng thuật toán Decision Tree trên dữ liệu XSMB.csv, bạn có thể làm như sau:

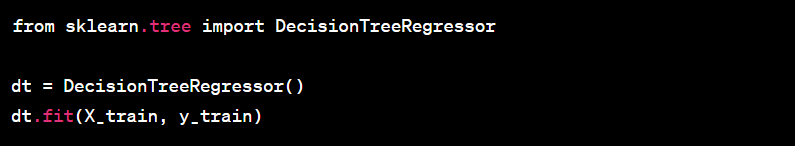
Chuẩn bị dữ liệu



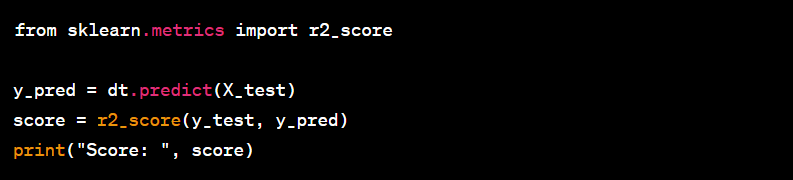
Chia dữ liệu thành tập huấn luyện và tập kiểm tra



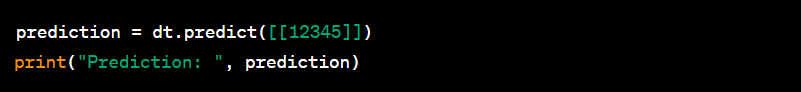
Xây dựng mô hình Decision Tree



Đánh giá mô hình



Dự đoán

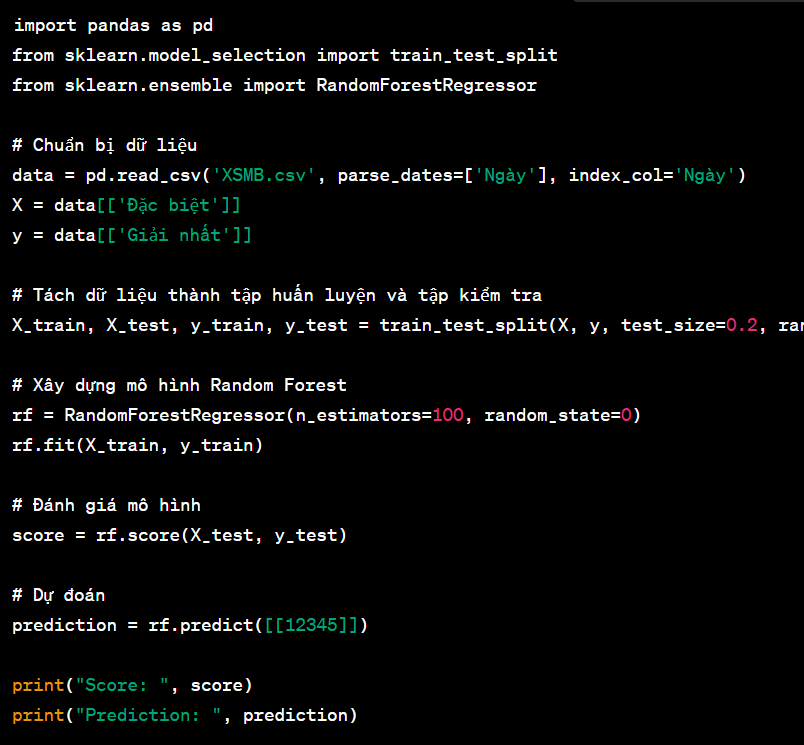


Trong đó, X là ma trận các đặc trưng (trong trường hợp này chỉ có một đặc trưng là Giải đặc biệt), y là vector các nhãn (trong trường hợp này là Giải nhất). Tập dữ liệu được chia thành tập huấn luyện và tập kiểm tra với tỷ lệ 80:20. Mô hình Decision Tree được xây dựng và đánh giá bằng hàm r2\_score dựa trên tập kiểm tra, và được sử dụng để dự đoán số có thể ra hôm nay.

Kết quả score có ý nghĩa là độ chính xác của mô hình Decision Tree trên tập dữ liệu kiểm tra. Kết quả score càng gần 1 thì mô hình càng chính xác.

*Random Forest*

Để áp dụng thuật toán Random Forest để phân tích dữ liệu của bạn, bạn có thể sử dụng thư viện scikit-learn trong Python



Trong đó:

Đối tượng RandomForestRegressor được sử dụng để xây dựng mô hình Random Forest với tham số n\_estimators là số cây trong rừng và random\_state là seed cho random state.

fit được sử dụng để huấn luyện mô hình trên dữ liệu huấn luyện X\_train và y\_train.

score được sử dụng để đánh giá mô hình trên dữ liệu kiểm tra X\_test và y\_test.

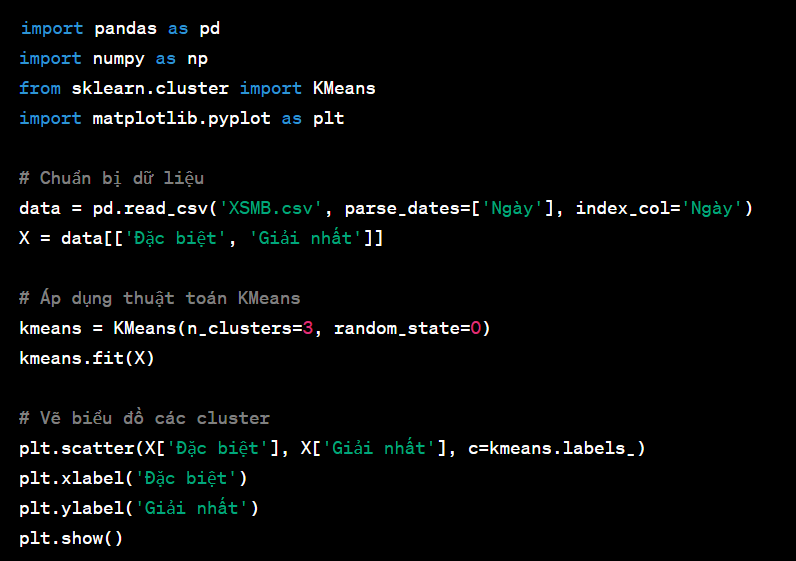
predict được sử dụng để dự đoán giá trị cho một mẫu mới.

*Clustering*

Clustering là một kỹ thuật trong Machine Learning, cho phép chia dữ liệu vào các nhóm (cluster) dựa trên tính chất tương tự của các điểm dữ liệu. Thuật toán clustering không cần có nhãn (label) cho dữ liệu và là một kỹ thuật phổ biến trong việc khám phá cụm dữ liệu.

Để áp dụng Clustering cho dữ liệu của bạn, bạn có thể sử dụng thuật toán KMeans của thư viện scikit-learn trong Python.

Dưới đây là ví dụ code áp dụng KMeans cho dữ liệu của bạn:



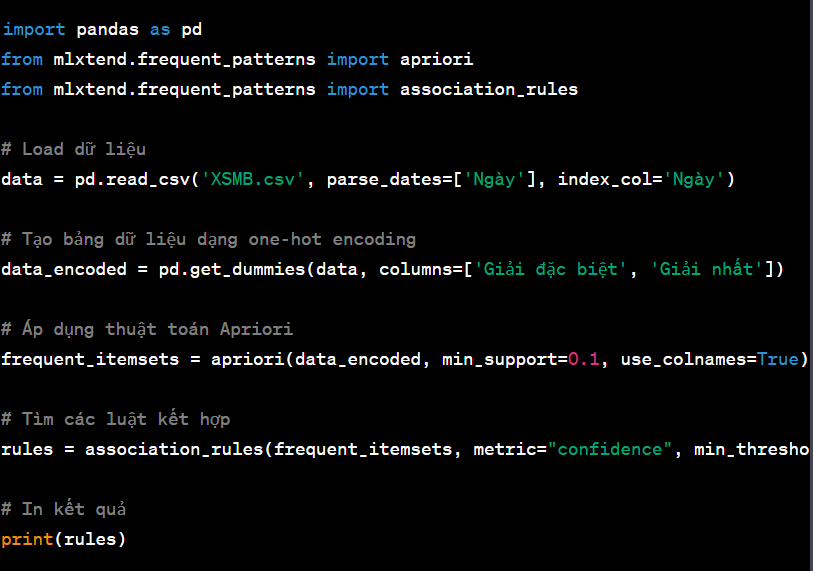
Trong đó, n\_clusters là số cluster bạn muốn phân chia dữ liệu thành, và labels\_ là nhãn của từng điểm dữ liệu sau khi được phân chia vào các cluster. Biểu đồ sẽ cho thấy phân chia của các điểm dữ liệu vào các cluster. Bạn có thể thay đổi số cluster để tìm kiếm phân chia tốt nhất cho dữ liệu của mình.

Association Rule Mining

Association Rule Mining là một kỹ thuật phân tích dữ liệu để khám phá các mối quan hệ giữa các mặt hàng trong một tập dữ liệu. Kỹ thuật này được sử dụng để tìm kiếm các quy luật liên quan đến các sản phẩm hay dịch vụ được mua cùng nhau, từ đó có thể đưa ra các khuyến nghị cho các sản phẩm hay dịch vụ liên quan đến nhau.

Trong trường hợp của bạn, bạn có thể áp dụng kỹ thuật Association Rule Mining để tìm kiếm các mối quan hệ giữa các giải thưởng trong các ngày khác nhau. Ví dụ, bạn có thể sử dụng thuật toán Apriori để tìm các cặp giải thưởng thường được trúng cùng nhau trong các ngày khác nhau.

Dưới đây là một ví dụ về cách áp dụng thuật toán Apriori trên dữ liệu của bạn:



Bạn có thể điều chỉnh các tham số như min\_support (ngưỡng hỗ trợ tối thiểu) và min\_threshold (ngưỡng độ tin cậy tối thiểu) để tìm ra các kết quả phù hợp với dữ liệu của bạn.

Phân tích thành phần chính (PCA)

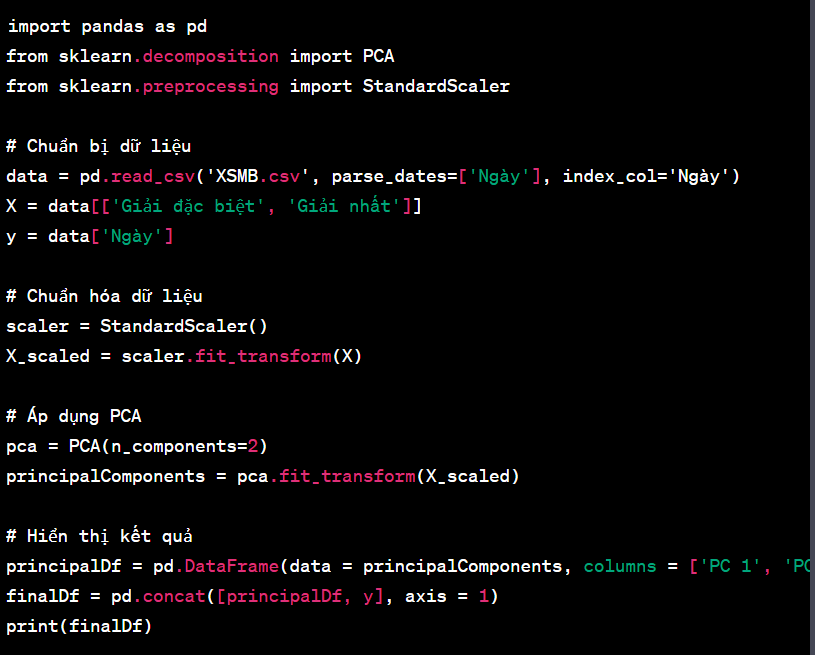
Để áp dụng thuật toán PCA cho dữ liệu của bạn, bạn có thể làm theo các bước sau:

Chuẩn bị dữ liệu: Đọc dữ liệu vào và chuẩn bị dữ liệu bằng cách loại bỏ các giá trị thiếu (nếu có) và chuẩn hóa dữ liệu (nếu cần).

Áp dụng PCA: Áp dụng thuật toán PCA để phân tích dữ liệu. Các thành phần chính sẽ được tạo ra bằng cách giảm số chiều của dữ liệu ban đầu.

Hiển thị kết quả: Hiển thị kết quả bằng cách biểu diễn các thành phần chính và giải thích ý nghĩa của chúng.

Dưới đây là ví dụ code Python minh họa cho việc áp dụng PCA cho dữ liệu của bạn:



Trong ví dụ trên, chúng ta sử dụng 2 thành phần chính để giảm số chiều của dữ liệu ban đầu. Kết quả được hiển thị dưới dạng một dataframe có 3 cột, trong đó 2 cột đầu tiên là các thành phần chính và cột cuối cùng là thông tin về ngày. Bạn có thể sử dụng các cột đầu tiên để phân tích dữ liệu của mình.

Hồi quy tuyến tính

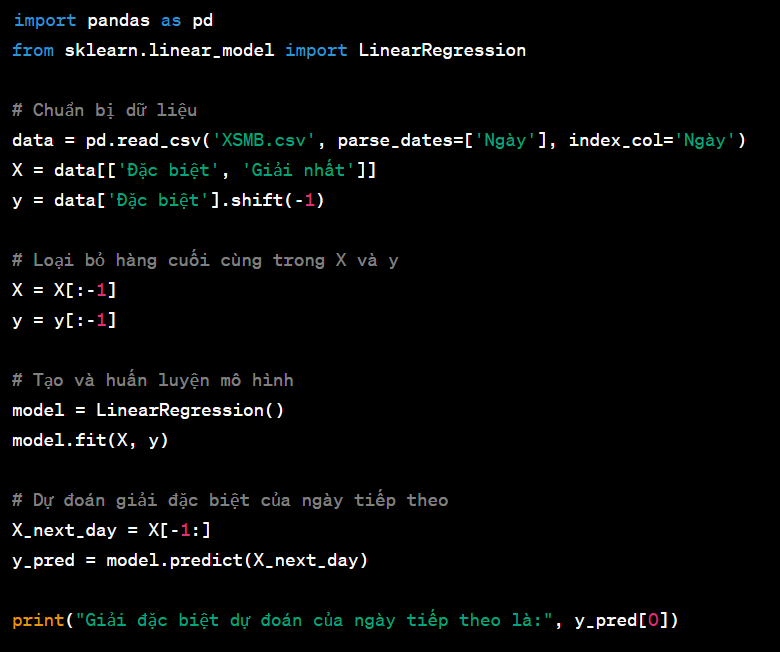
Để áp dụng thuật toán Hồi quy tuyến tính, trước tiên chúng ta cần chuẩn bị dữ liệu bao gồm:

X: là ma trận các đặc trưng, có kích thước (m, n), trong đó m là số lượng mẫu và n là số lượng đặc trưng. Trong ví dụ này, các đặc trưng có thể là các giá trị đặc biệt và giải nhất của các ngày trước đó.

y: là vector đầu ra, có kích thước (m, 1), chứa giá trị dự đoán cần tìm. Trong ví dụ này, giá trị dự đoán có thể là giải đặc biệt của ngày tiếp theo.

Sau khi chuẩn bị dữ liệu, ta có thể sử dụng thư viện scikit-learn để huấn luyện mô hình Hồi quy tuyến tính.

Ví dụ code:



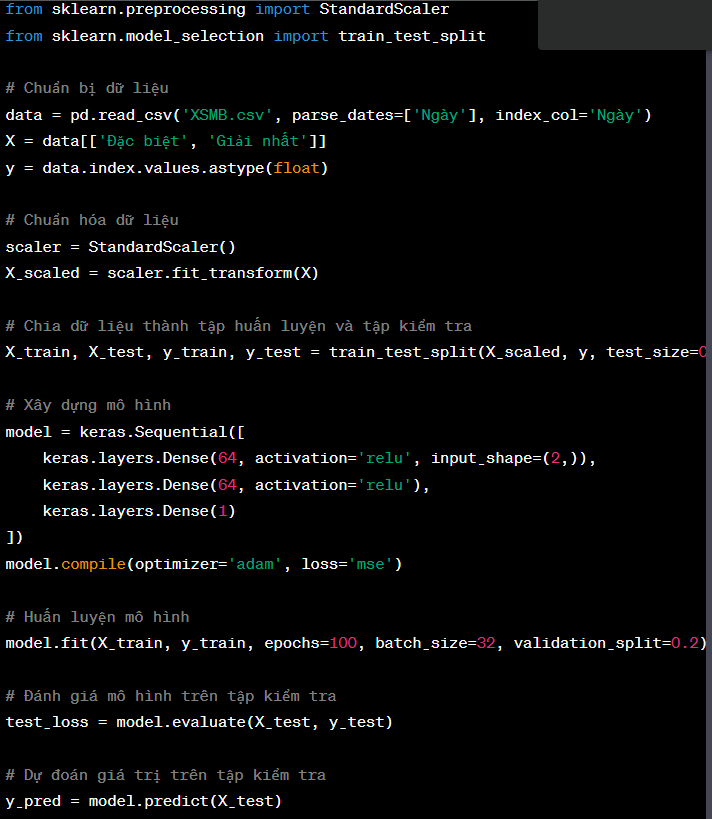
Lưu ý: Trong ví dụ này, để dự đoán giải đặc biệt của ngày tiếp theo, ta sử dụng giá trị đặc biệt của ngày hôm trước và giải đặc biệt của 2 ngày trước đó. Các đặc trưng có thể được chọn khác tùy thuộc vào bài toán cụ thể.

Mạng neural

Để áp dụng thuật toán Mạng neural cho bài toán của bạn, bạn có thể sử dụng thư viện Tensorflow của Python. Đầu tiên, bạn cần cài đặt Tensorflow bằng cách chạy lệnh sau trong terminal:

“pip install tensorflow”

Sau đó, bạn có thể sử dụng mã sau để áp dụng thuật toán Mạng neural cho bài toán của mình:

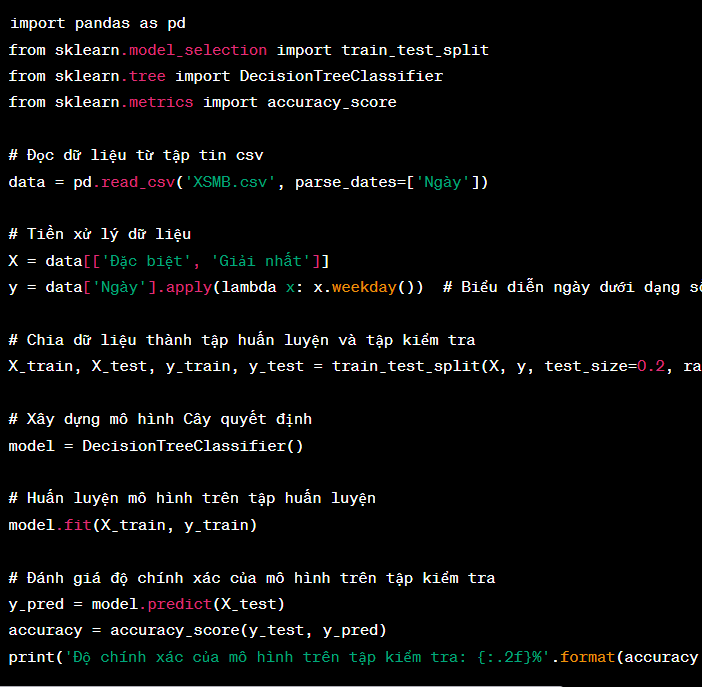


Trong đoạn mã trên, chúng ta sử dụng một mạng neural đơn giản với 2 tầng ẩn có 64 nơ-ron, và hàm kích hoạt là relu. Chúng ta sử dụng hàm mất mát là mean squared error (MSE) và tối ưu hóa bằng thuật toán Adam. Mô hình được huấn luyện trên tập huấn luyện và đánh giá trên tập kiểm tra.

Cây quyết định

Để áp dụng thuật toán Cây quyết định cho dữ liệu của bạn, trước tiên bạn cần phải tiến hành tiền xử lý dữ liệu và chia dữ liệu thành tập huấn luyện và tập kiểm tra. Sau đó, bạn có thể sử dụng thư viện scikit-learn để xây dựng mô hình và đánh giá độ chính xác của mô hình trên tập kiểm tra.

Dưới đây là mã Python để áp dụng thuật toán Cây quyết định cho dữ liệu của bạn:

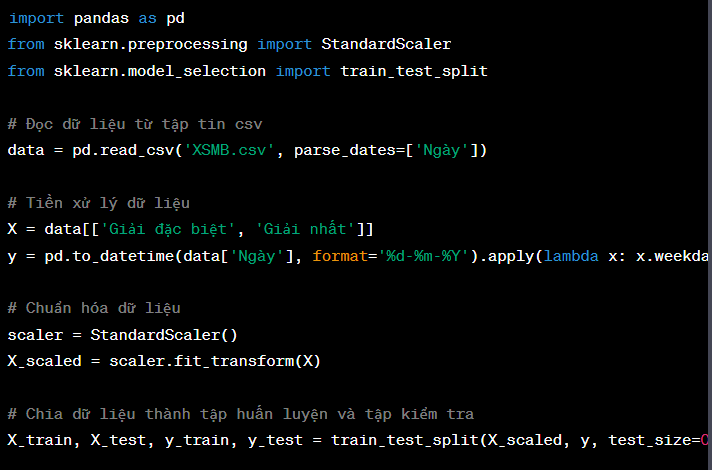


Trong đoạn mã trên, chúng ta sử dụng cột 'Ngày' để biểu diễn ngày dưới dạng số nguyên từ 0 đến 6 (từ thứ 2 đến Chủ nhật), bằng cách sử dụng phương thức apply() và hàm lambda. Sau đó, chúng ta chia dữ liệu thành tập huấn luyện và tập kiểm tra với tỷ lệ 80:20 bằng hàm train\_test\_split() của thư viện scikit-learn. Chúng ta sử dụng mô hình DecisionTreeClassifier để xây dựng mô hình và đánh giá độ chính xác của mô hình trên tập kiểm tra bằng hàm accuracy\_score() của thư viện scikit-learn.

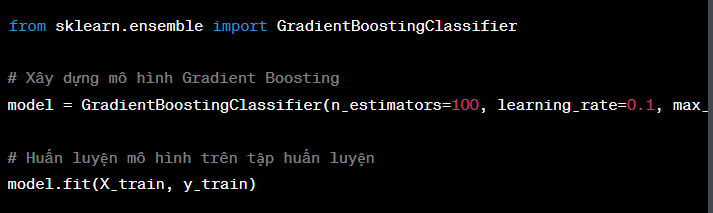
Gradient Boosting

Để áp dụng thuật toán Gradient Boosting trên dữ liệu của bạn, bạn có thể làm như sau:

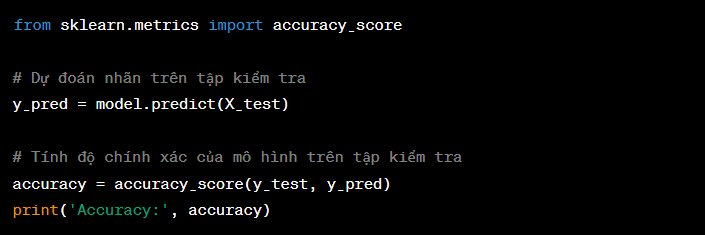
Đọc dữ liệu từ tập tin csv và tiền xử lý dữ liệu



Xây dựng và huấn luyện mô hình Gradient Boosting



Đánh giá mô hình trên tập kiểm tra

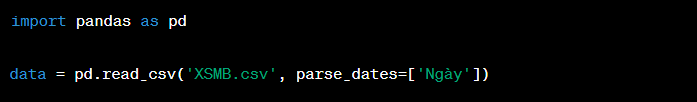


Lưu ý rằng trong ví dụ này, chúng ta muốn dự đoán ngày trong tuần của một kết quả xổ số dựa trên giải đặc biệt và giải nhất của nó. Do đó, chúng ta sử dụng mô hình Gradient Boosting để phân loại các ngày trong tuần, chứ không phải để dự đoán giá trị của giải đặc biệt hoặc giải nhất.

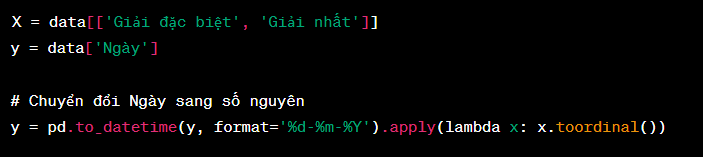
Máy vector hỗ trợ (SVM)

Để áp dụng thuật toán Máy vector hỗ trợ (SVM) cho dữ liệu XSMB.csv của bạn, bạn có thể sử dụng thư viện Scikit-learn trong Python. Cụ thể, bạn có thể làm như sau:

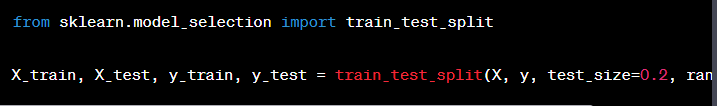
Đọc dữ liệu từ tập tin csv:



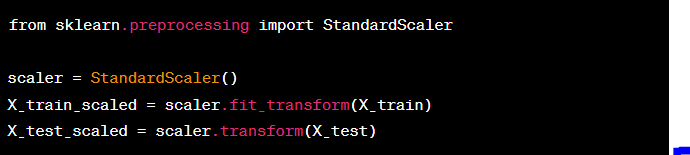
Tiền xử lý dữ liệu:



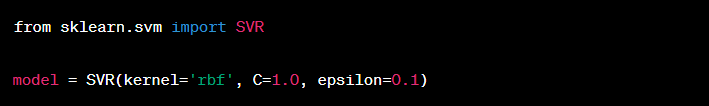
Chia dữ liệu thành tập huấn luyện và tập kiểm tra:



Chuẩn hóa dữ liệu:



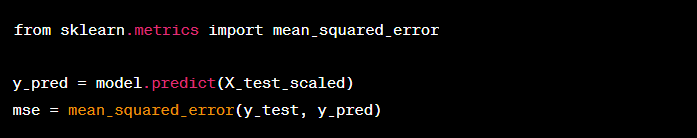
Xây dựng mô hình SVM:



Huấn luyện mô hình:



Đánh giá mô hình trên tập kiểm tra:



Lưu ý rằng đối với bài toán dự đoán ngày, giải đặc biệt và giải nhất, bạn cần chuyển đổi giá trị của Ngày thành số nguyên bằng phương thức toordinal() của đối tượng datetime trong Python.