**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH**

**BÁO CÁO MÔN DATA MINING**

**Dataset: Forbes Billionaires n Companies**

**Thành viên:**

**Mai Khánh Huy - 2051052139**

**Võ Thành Tính - 2051052139**

**Hoàng Vũ Thanh - 1951012119**

**TP. HỒ CHÍ MINH, 2022**

MỤC LỤC

[Chương 1. CƠ SỞ LÝ THUYẾT 4](#_Toc143641194)

[1.1. Quá trình khai phá dữ liệu: 4](#_Toc143641195)

[1.2. Thuật toán K-Means: 5](#_Toc143641196)

[1.3. K-NN: 6](#_Toc143641197)

[1.4. Khai phá Luật kết hợp (Association Rules Mining): 6](#_Toc143641198)

[1.4.1. Tổng quan: 6](#_Toc143641199)

[1.4.2. Thuật toán Apriori: 6](#_Toc143641200)

[1.5. Cây quyết định (Decision Tree) 7](#_Toc143641201)

[Chương 2. THỰC HIỆN KHAI PHÁ VÀ KẾT QUẢ 9](#_Toc143641202)

[2.1. Mô tả dữ liệu: 9](#_Toc143641203)

[2.2. Thống kê dữ liệu: 10](#_Toc143641204)

[2.3. Chuẩn hóa dữ liệu: 12](#_Toc143641205)

[2.3.1. Các bộ dữ liệu Forbes Billionaires 2020, 2021, 2022: 12](#_Toc143641206)

[2.3.2. Các bộ dữ liệu Forbes Companies 2020, 2021, 2022: 13](#_Toc143641207)

[2.4. K-Means: 14](#_Toc143641208)

[2.5. K-NN: 22](#_Toc143641209)

[2.5.1. Bộ dữ liệu Forbes Billionaires các năm 2020, 2021, 2022: 22](#_Toc143641210)

[2.5.2. Bộ dữ liệu Companies các năm 2020, 2021, 2022 25](#_Toc143641211)

[2.6. Khai phá Luật kết hợp: 27](#_Toc143641212)

[2.6.1. Đối với các bộ dữ liệu Forbes Billionaires 2020, 2021, 2022: 27](#_Toc143641213)

[2.6.2. Đối với các bộ dữ liệu Forbes Companies 2020, 2021, 2022: 28](#_Toc143641214)

[2.7. Cây quyết định: 29](#_Toc143641215)

[2.7.1. Đối với các bộ dữ liệu Forbes Billionaires 2020, 2021, 2022: 29](#_Toc143641216)

[2.7.2. Đối với các bộ dữ liệu Forbes Companies 2020, 2021, 2022: 31](#_Toc143641217)

[TÀI LIỆU THAM KHẢO 34](#_Toc143641218)

# CƠ SỞ LÝ THUYẾT

## Quá trình khai phá dữ liệu:

Các bước trong quá trình khai phá dữ liệu được mô tả như sau:

**Bước 1** - **Lựa chọn dữ liệu (Selection):** Bước này liên quan đến việc chọn lọc dữ liệu cụ thể từ nguồn dữ liệu gốc để sử dụng trong quá trình khai phá. Mục tiêu là chọn những dữ liệu quan trọng, hữu ích và liên quan để giải quyết vấn đề cụ thể.

**Bước 2** - **Tiền xử lý dữ liệu (Preprocessing):** Ở bước này, dữ liệu được chuẩn bị trước khi áp dụng các phương pháp khai phá. Quá trình này bao gồm việc làm sạch dữ liệu, xử lý dữ liệu bị thiếu, loại bỏ nhiễu, chuẩn hóa dữ liệu và tạo ra các đặc trưng mới từ dữ liệu gốc.

**Bước 3** - **Biến đổi dữ liệu (Transformation):** Bước biến đổi dữ liệu thay đổi cấu trúc hoặc định dạng của dữ liệu để phù hợp với yêu cầu của bài toán khai phá. Điều này có thể bao gồm việc chuyển đổi loại dữ liệu, đơn vị đo lường hoặc thay đổi cấu trúc dữ liệu.

**Bước 4** - **Khai phá dữ liệu (Data Mining):** Ở bước này, các phương pháp và kỹ thuật tính toán được sử dụng để khám phá thông tin hữu ích từ tập dữ liệu. Điều này bao gồm việc áp dụng các thuật toán máy học, học máy, phân loại, gom cụm và tìm kiếm luật kết hợp để rút trích thông tin từ dữ liệu.

**Bước 5** - **Diễn giải (Interpretation):** Bước diễn giải liên quan đến việc hiểu và giải thích kết quả của quá trình khai phá dữ liệu. Nó liên quan đến việc tìm hiểu ý nghĩa của các mẫu, quy tắc hoặc thông tin quan trọng đã được khám phá.

**Bước 6** - **Đánh giá (Evaluation):** Quá trình đánh giá dùng để đo lường chất lượng và hiệu suất của các mô hình khai phá dữ liệu. Các phép đo, thang điểm hoặc phương pháp so sánh được sử dụng để đánh giá độ chính xác, độ tin cậy và hiệu suất của các phương pháp khai phá dữ liệu đã được áp dụng.

Các bước trong quá trình này có thể được điều chỉnh và lặp lại để đạt được kết quả tốt nhất.

A diagram of data processing process

Description automatically generated

## Thuật toán K-Means:

Thuật toán gom cụm K-Means là một trong những thuật toán gom cụm phổ biến nhất. Nó hoạt động dựa trên việc tìm cách phân chia các điểm dữ liệu vào K cụm sao cho tổng bình phương khoảng cách giữa các điểm và trung tâm cụm là nhỏ nhất.

Khởi tạo ban đầu: Khởi tạo ban đầu có thể ảnh hưởng đến kết quả cuối cùng. K-Means++ là một phương pháp khởi tạo phổ biến hơn, nó giúp chọn các trung tâm cụm ban đầu sao cho chúng phân bố đều trong không gian dữ liệu.

Số lượng lặp: Số lượng lặp cần thiết để thuật toán hội tụ có thể khác nhau. Cần quan sát kết quả và xác định xem cần bao nhiêu lần lặp để đạt được kết quả tốt nhất.

Tiêu chí hội tụ: Tiêu chí hội tụ thường là sự thay đổi nhỏ của trung tâm cụm giữa các lần lặp liên tiếp. Khi sự thay đổi này nhỏ hơn một ngưỡng cố định hoặc khi đạt đến số lượng lặp tối đa, thuật toán dừng lại.

Vấn đề với số lượng cụm K: Lựa chọn số lượng cụm K là một vấn đề quan trọng. Cách đơn giản là thử nghiệm với nhiều giá trị K và sử dụng các chỉ số như Elbow Method (phương pháp khuỷu tay) để tìm giá trị K tốt nhất.

Cách đánh giá chất lượng cụm: Để đánh giá chất lượng của các cụm tạo bởi K-Means, bạn có thể sử dụng các chỉ số như SSE (Sum of Squared Errors), Silhouette Score, hoặc Calinski-Harabasz Index.

Phân tích kết quả: Sau khi thuật toán hoàn tất, hãy kiểm tra trung tâm cụm và xem xét cách mà các cụm được phân bố trong không gian dữ liệu. Điều này có thể giúp bạn hiểu rõ hơn về cấu trúc dữ liệu của mình.

Xử lý dữ liệu nhiễu: K-Means có thể bị ảnh hưởng bởi các điểm nhiễu. Để giảm ảnh hưởng của điểm nhiễu, bạn có thể sử dụng các phiên bản tăng cường như K-Means++ hoặc sử dụng thuật toán khác như DBSCAN.

Thuật toán K-Means có thể được thực hiện bằng nhiều ngôn ngữ lập trình và thư viện, như Python với thư viện scikit-learn. Điều này giúp bạn dễ dàng áp dụng K-Means trên dữ liệu thực tế và thực hiện các phân tích cụ thể.

## K-NN:

Thuật toán K-Nearest Neighbors (KNN) là một phương pháp phân loại phổ biến trong lĩnh vực học máy. Cách KNN hoạt động dựa trên việc tính khoảng cách giữa các điểm dữ liệu để xác định lớp của điểm mới cần phân loại.

## Khai phá Luật kết hợp (Association Rules Mining):

### Tổng quan:

Khai phá Luật kết hợp là một kỹ thuật Machine Learning với mục tiêu nhằm phát hiện mối quan hệ giữa các mục dữ liệu trong cơ sở dữ liệu. Một luật kết hợp có 2 phần: antecedent và consequent.

Khi khai phá luật kết hợp, chúng ta tìm kiếm các luật kết hợp thỏa mãn các ngưỡng độ hỗ trợ và độ tin cậy cho trước. Trong đó:

* Support: là tần suất xuất hiện của một tập hợp các mục dữ liệu trong cơ sở dữ liệu.
* Confidence: là xác suất xuất hiện của một tập hợp các mục dữ liệu trong cơ sở dữ liệu khi biết rằng một tập hợp khác đã xuất hiện.

Ngoài ra, chúng ta còn có các phương pháp để đánh giá Luật kết hợp như:

* Lift: được sử dụng để so sánh các luật kết hợp có chung Consequent hay nhấn mạnh vào tầm quan trọng của Antecedent.
* Leverage: được sử dụng để giảm bớt những luật kết hợp đảm bảo tỷ lệ Support, tỷ lệ Confidence. Leverage đo lường sự khác biệt trong tỷ lệ support (A, B) và tỷ lệ support của A và B nếu chúng độc lập. Leverage bằng 0 khi A và B thực sự độc lập.

### Thuật toán Apriori:

Là thuật toán rất nổi tiếng trong kỹ thuật Khai phá Luật kết hợp. Bao gồm các bước:

Bước 1: Tìm tập phổ biến 1-items (tập chỉ có 1 items).

Bước 2: (Loop) Tạo các tập ứng viên k-items từ các tập k-1 items.

Bước 3: Kiểm tra độ phổ biến của ứng viên trên CSDL. Chỉ lấy các Large Items làm tập ứng viên, bỏ qua Small Items.

Bước 4: Dừng khi không tạo ra được tập phổ biến hay tập ứng viên.

## Cây quyết định (Decision Tree)

Cây quyết định là một thuật toán khai phá dữ liệu trong lĩnh vực học máy và khoa học dữ liệu. Nó là một loại mô hình học máy giúp tạo ra các quyết định dựa trên việc tạo ra một loạt các câu hỏi tách dữ liệu thành các nhóm nhỏ hơn, cuối cùng là các quyết định hoặc dự đoán.

Cây quyết định hoạt động theo cách xây dựng một cây có các nút đại diện cho các quyết định và các nút lá đại diện cho các kết quả dự đoán. Quá trình xây dựng cây bắt đầu từ nút gốc và tiến hành tạo ra các nhánh (branches) dựa trên các câu hỏi về các đặc trưng của dữ liệu. Các câu hỏi này giúp tách dữ liệu thành các phần nhỏ hơn và tiếp tục được áp dụng đến các nút con. Quá trình này tiếp tục cho đến khi đạt được điều kiện dừng, ví dụ như một độ sâu tối đa hoặc không còn sự tách biệt hiệu quả nữa.

Các thành phần quan trọng của cây quyết định

Nút Gốc (Root Node): Đây là nút đầu tiên của cây, nơi quyết định ban đầu được thực hiện dựa trên một đặc trưng. Tất cả các nhánh con bắt đầu từ nút gốc.

Nút Quyết Định (Decision Node): Là các nút trong cây nơi quyết định về việc tách dữ liệu dựa trên một hoặc nhiều đặc trưng. Các nút này là mối nối giữa các nhánh khác nhau trong cây.

Nút Lá (Leaf Node): Các nút lá đại diện cho kết quả cuối cùng hoặc lớp dự đoán. Khi cây đã thực hiện tách dữ liệu đến mức tối ưu, kết quả của dự đoán sẽ nằm ở các nút lá.

Nhánh (Branches): Các nhánh là các đường dẫn từ nút gốc đến nút lá. Các quyết định và tách dữ liệu được thực hiện thông qua các nhánh này.

Gini Impurity hoặc Entropy (Độ Tạp Hóa Gini hoặc Entropy): Là các phép đo độ tạp hóa của dữ liệu tại mỗi nút. Cây quyết định cố gắng tối thiểu hóa độ tạp hóa bằng cách chọn cách tách dữ liệu sao cho các nhóm con có độ tạp hóa thấp hơn.

Sự Tách (Splitting): Quá trình tách dữ liệu tại các nút quyết định. Điều này bao gồm việc chọn một hoặc nhiều đặc trưng để tách dữ liệu thành các nhóm con.

Chọn Đặc Trưng (Feature Selection): Quá trình chọn đặc trưng tốt nhất để tách dữ liệu dựa trên tiêu chí tối thiểu hóa độ tạp hóa, thông tin lượng dữ liệu, hoặc các phép đo khác.

Cắt Tác Động (Pruning): Là quá trình loại bỏ các nhánh hoặc nút lá không cần thiết khỏi cây để tránh quá khớp (overfitting).

Độ Sâu Cây (Tree Depth): Số lượng nút quyết định từ nút gốc đến nút lá. Các cây có độ sâu lớn có khả năng bị quá khớp dữ liệu.

Các Tham Số Điều Khiển (Hyperparameters): Là các tham số có thể điều chỉnh để kiểm soát quá trình xây dựng cây, chẳng hạn như độ sâu tối đa của cây, ngưỡng độ tạp hóa, cách tách dữ liệu, và nhiều tham số khác.

# THỰC HIỆN KHAI PHÁ VÀ KẾT QUẢ

## Mô tả dữ liệu:

Bộ dữ liệu **‘Forbes Billionaires’** là bảng xếp hạng thường niên những người được coi là có giá trị ròng từ 1 tỷ USD trở lên, do tạp chí kinh doanh Forbes của Mỹ thực hiện. Trong khi đó, bộ dữ liệu **‘Forbes Companies’** là một bảng xếp hạng hàng năm của 2000 công ty đại chúng lớn nhất trên thế giới theo tạp chí Forbes.

Đối với bộ dữ liệu **‘Forbes Billionaires’**, nhóm sử dụng bộ dữ liệu từ năm 2020 đến năm 2022 gồm các đặc trưng:

* Rank: Thứ hạng của tỷ phú trong danh sách dựa theo tài sản ròng
* Name: Tên tỷ phú được liệt kê trong danh sách
* Net Worth: Tài sản ròng của tỷ phú trong một khoảng thời gian nhất định
* Age: Tuổi
* Country/Territory: Quốc gia của tỷ phú
* Source: Doanh nghiệp mà tỷ phú đang công tác
* Industries: Ngành hàng của doanh nghiệp

Đối với bộ dữ liệu **‘Forbes Companies’**, nhóm sử dụng bộ dữ liệu từ năm 2020 đến năm 2022, bao gồm các đặc trưng:

* Rank: Xếp hạng của công ty trong danh sách dựa trên một tiêu chí cụ thể, chẳng hạn như doanh số, lợi nhuận hoặc giá trị thị trường.
* Global Company: Tên công ty quốc tế được liệt kê trong danh sách.
* Country: Quốc gia mà công ty đó có trụ sở hoặc quốc gia mà nền kinh tế công ty thuộc về.
* Sales: Doanh số của công ty trong một khoảng thời gian nhất định. Đây là tổng số tiền công ty đã thu được từ việc bán hàng hoặc dịch vụ.
* Profit: Lợi nhuận của công ty trong một khoảng thời gian nhất định. Đây là số tiền công ty kiếm được sau khi trừ đi các chi phí hoạt động, thuế, và các yếu tố khác.
* Assets: Tổng tài sản của công ty, bao gồm các khoản tiền, tài sản cố định (như nhà xưởng, máy móc), nợ phải thu và các tài sản khác.
* Market Value: Giá trị thị trường của công ty, tức là tổng giá trị của tất cả các cổ phiếu của công ty tính theo giá cổ phiếu hiện tại trên thị trường.

## Thống kê dữ liệu:

* Top 10 Country có Profit cao nhất và thấp nhất

A screenshot of a graph

Description automatically generated

A graph of orange squares

Description automatically generated

* Top 10 quốc gia Sales cao nhất và thấp nhất

**A white rectangular object with orange squares

Description automatically generatedA graph of orange bars

Description automatically generated with medium confidence**

* Top 10 Quốc Gia có Assets cao nhất và thấp nhất

A graph with orange squares

Description automatically generated

A graph with orange rectangular bars

Description automatically generated with medium confidence

* Top 10 quốc gia có MARKET VALUE cao nhất và thấp nhất

A graph with orange and white bars

Description automatically generated with medium confidence

A graph of orange rectangular objects

Description automatically generated with medium confidence

* Tần suất xuất hiện của các quốc gia

A white rectangular object with black lines

Description automatically generated with medium confidence

## Chuẩn hóa dữ liệu:

### Các bộ dữ liệu Forbes Billionaires 2020, 2021, 2022:

* Chuẩn hóa cột dữ liệu Net Worth về dạng số.

A screenshot of a computer code

Description automatically generated

### Các bộ dữ liệu Forbes Companies 2020, 2021, 2022:

* Loại bỏ các cột dữ liệu gây nhiễu đồng thời đổi tên các cột dữ liệu.



* Chuẩn hóa các cột dữ liệu Sales, Profit, Assets, Maket value về dạng số.

A screenshot of a computer screen

Description automatically generated

A table with numbers and symbols

Description automatically generated

A table of numbers and numbers

Description automatically generated with medium confidence

* Thêm vào các dữ liệu bị Null

A screenshot of a computer

Description automatically generated

## K-Means:

Ở đây chúng ta sẽ đi phân tích về bộ dữ liệu 2020\_forbes\_billionaires.csv. Sau khi đã chuẩn hóa dữ liệu chuẩn ta sẽ có các cột Net Worth, Age, Country, Source, Industries để gom cụm, sau đó đưa bộ dữ liệu về một không gian mẫu đó là số để dễ dàng cho việc gom cụm.

A screen shot of a computer code

Description automatically generated

Đầu tiên chúng ta sẽ import thư viện sklearn.linear\_model, LinearRegression, import module LinearRegression từ thư viện scikit-learn để sử dụng thuật toán Linear Regression.

Sau đó chúng ta sử dụng một công cụ gọi là "LabelEncoder" để mã hóa các giá trị trong các cột cụ thể của một DataFrame được gọi là df\_bln1. Mục đích của việc này có thể là để chuyển đổi các giá trị chuỗi hoặc phân loại thành các số nguyên để có thể sử dụng chúng trong các mô hình học máy hoặc thuật toán yêu cầu dữ liệu số hóa.

* label\_encoder = LabelEncoder(): Đây là bước tạo một đối tượng LabelEncoder. Đối tượng này sẽ được sử dụng để thực hiện quá trình mã hóa.
* df\_bln1['Net Worth'] = label\_encoder.fit\_transform(df\_bln1['Net Worth']): Ở dòng này, cột 'Net Worth' trong DataFrame df\_bln1 đang được mã hóa. Phương thức
* fit\_transform của đối tượng LabelEncoder được sử dụng để chuyển đổi các giá trị trong cột 'Net Worth' thành các số nguyên tương ứng. Kết quả sau khi mã hóa sẽ được gán lại vào cột 'Net Worth'.
* Tương tự như trên, các dòng tiếp theo thực hiện mã hóa các cột 'Age', 'Country', 'Source' và 'Industries' trong cùng DataFrame df\_bln1.
* Ta được kết quả như sau :

A screenshot of a number table

Description automatically generated

Sau khi chuyển đổi về một mẫu chung ta đến giai đoạn xác đinh K để gom cụm dữ liệu. Ở đây để chọn k ta sẽ vẽ tính giá trị trung bình SSE sau đó vẽ biểu đồ khuỷu tay để xác định đoạn gấp khúc nhằm đưa ra k phù hợp.

Ta thực hiện tìm k bằng phương pháp "Elbow Method" để xác định số lượng cụm tốt nhất cho dữ liệu trong DataFrame df\_bln1. Đây là một cách thường được sử dụng để xác định số lượng cụm tối ưu cho thuật toán KMeans.

* import matplotlib.pyplot as plt: Thư viện matplotlib được sử dụng để vẽ biểu đồ.
* from sklearn.cluster import KMeans: Thư viện sklearn được sử dụng để thực hiện thuật toán phân cụm KMeans.
* from sklearn.preprocessing import LabelEncoder: Đoạn mã trước đã sử dụng LabelEncoder để mã hóa dữ liệu, tuy nhiên, trong đoạn mã này, không cần thiết vì chúng ta chỉ quan tâm đến việc xác định số lượng cụm.
* sse\_values = []: Đây là một danh sách rỗng để lưu giá trị Sum of Squared Errors (SSE) tương ứng với số lượng cụm từ 1 đến 9.
* Vòng lặp for k in range(1, 10): Vòng lặp này sẽ duyệt qua các số lượng cụm từ 1 đến 9.
* Trong vòng lặp, kmeans = KMeans(n\_clusters=k): Một đối tượng KMeans được khởi tạo với số lượng cụm là k.
* kmeans.fit(df\_bln1): Đối tượng KMeans được huấn luyện trên dữ liệu trong DataFrame df\_bln1.
* sse\_values.append(kmeans.inertia\_): Giá trị SSE của mô hình KMeans sau khi huấn luyện được thêm vào danh sách sse\_values.
* Sau khi vòng lặp kết thúc, danh sách sse\_values sẽ chứa các giá trị SSE tương ứng với số lượng cụm từ 1 đến 9.
* plt.plot(range(1, 10), sse\_values, marker='o'): Đoạn mã này vẽ biểu đồ điểm của các giá trị SSE với số lượng cụm trên trục x và giá trị SSE trên trục y.
* plt.grid(True): Đoạn mã này thêm lưới vào biểu đồ.
* plt.show(): Hiển thị biểu đồ.

A screenshot of a computer code

Description automatically generated

Biểu đồ sẽ giúp bạn xác định "khuỷu tay" trong biểu đồ, nơi giá trị SSE bắt đầu giảm chậm sau khi tăng. Điều này thường xác định số lượng cụm tốt nhất cho dữ liệu.

Nhìn vào biểu đồ ta thấy giá trị SSE giảm mạnh khi có số cụm là 1 -> 4 và bắt đầu giảm chậm lại khi từ cụm 5 trở lên nên ta sẽ chọn k = 5 để phân cụm.

A graph with a line

Description automatically generated

Tiếp theo ta tiến hành gom cụm dữ liệu bằng phương pháp K-Means theo 5 cột Net Worth, Age

* kmeans = KMeans(n\_clusters=6): Đoạn này khởi tạo một đối tượng KMeans với tham số n\_clusters được đặt là 6. Điều này có nghĩa là chúng ta đang cố gắng tìm 6 cụm trong dữ liệu.
* kmeans.fit: Bước này, đối tượng KMeans được huấn luyện trên một phần của dữ liệu trong DataFrame df\_bln1, chỉ bao gồm các cột trong dữ liệu. Điều này đặt ra mục tiêu là tìm cách phân chia các dòng dữ liệu dựa trên các đặc trưng này thành 6 cụm.

A screenshot of a computer code

Description automatically generatedA screenshot of a computer program

Description automatically generatedSau đó ta gán thêm cột sẽ xuất các cụm để nhận xét các cụm.

* + labels = kmeans.labels\_: Đây là bước gán nhãn cho các điểm dữ liệu sử dụng thuật toán K-means. kmeans ở đây là một đối tượng đã được tạo ra và huấn luyện từ thuật toán K-means trước đó. Thuộc tính labels\_ của đối tượng kmeans chứa thông tin về nhãn mà thuật toán đã gán cho mỗi điểm dữ liệu trong dãy số.
  + df\_bln1['Cluster'] = labels: Dòng này thêm một cột mới có tên là 'Cluster' vào DataFrame df\_bln1. Các giá trị trong cột này sẽ tương ứng với nhãn mà thuật toán K-means đã gán cho mỗi điểm dữ liệu.
  + df\_bln1[df\_bln1['Cluster'] == 0]: Đây là cách để lọc các dòng trong DataFrame df\_bln1 chỉ hiển thị những hàng (điểm dữ liệu) mà giá trị trong cột 'Cluster' là 0. Điều này sẽ trả về một bộ dữ liệu con (sub DataFrame) chứa các điểm thuộc vào cluster 0.

A screenshot of a number table

Description automatically generatedTa được kết quả là:

Sau đó ta xuất dữ liệu ra thể hiện từng cột để nhận xét về các cụm

A graph with colorful dots

Description automatically generated with medium confidenceA computer code with many colorful text

Description automatically generated with medium confidence

## K-NN:

### Bộ dữ liệu Forbes Billionaires các năm 2020, 2021, 2022:

A screenshot of a computer program

Description automatically generated

Hàm train\_test\_split được sử dụng để chia tập dữ liệu ban đầu thành tập huấn luyện (X\_train\_1, y\_train\_1) và tập kiểm tra (X\_test\_1, y\_test\_1). Tỉ lệ tập kiểm tra là 20% và random\_state được sử dụng để đảm bảo rằng việc chia dữ liệu là nhất quán.

Trong trường hợp này, mô hình KNN sẽ xem xét 11 láng giềng gần nhất và chọn nhãn dựa trên đa số nhãn trong những láng giềng này.

Kết quả bộ dữ liệu năm 2020:

A graph with lines and numbers

Description automatically generated

Có thể thấy rằng khi K tăng quá lớn ( ví dụ: K = 11), sai số trên cả tập train và tập test tăng lên. Điều này cho thấy mô hình có thể bắt đầu mất khả năng tổng quát khi K quá lớn , và đây là hiện tượng underfitting.

Kết quả bộ dữ liệu 2021 và 2022 tương tự nhau

A graph with numbers and lines

Description automatically generated

Nhận xét:

* Sai số rất thấp: Sai số trên cả tập huấn luyện và tập kiểm tra đều rất thấp và gần với 0. Điều này cho thấy mô hình KNN đã hoạt động khá tốt trên tập dữ liệu của bạn.
* Khả năng tổng quát hóa tốt: Độ chênh lệch giữa sai số trên tập huấn luyện và tập kiểm tra là nhỏ, cho thấy mô hình có khả năng tổng quát hóa tốt trên dữ liệu chưa từng thấy (tập kiểm tra). Điều này là một tín hiệu tích cực về sự hiệu quả của mô hình.
* Giá trị K ảnh hưởng tới sai số: Dựa trên bảng sai số, có vẻ như mô hình có hiệu suất tốt nhất với K bằng 3 hoặc 5, do có sai số thấp nhất trên tập kiểm tra. Tuy nhiên, sai số rất thấp cho tất cả các giá trị K, cho thấy mô hình khá ổn định.
* Sự ổn định với K lớn: Có thể thấy rằng sai số trên tập kiểm tra không tăng đáng kể khi K tăng lên (ví dụ: từ 5 đến 11). Điều này cho thấy mô hình có khả năng tổng quát hóa tốt và ít bị ảnh hưởng bởi thay đổi giá trị K.

### Bộ dữ liệu Companies các năm 2020, 2021, 2022

**Bộ dữ liệu năm 2020**

A graph with numbers and lines

Description automatically generated

- Sai số thấp: Tất cả các giá trị sai số đều thấp trên cả tập huấn luyện và tập kiểm tra. Điều này cho thấy mô hình KNN đã đạt được độ chính xác cao trên tập dữ liệu của bạn.

- Khả năng tổng quát hóa tốt: Sai số trên tập kiểm tra có xu hướng tăng nhẹ khi giá trị K tăng. Tuy nhiên, sự chênh lệ giữa sai số trên tập huấn luyện và tập kiểm tra vẫn khá nhỏ, cho thấy mô hình có khả năng tổng quát hóa tốt trên dữ liệu chưa từng thấy.

- Giá trị K tối ưu: Dựa vào sai số trên tập kiểm tra, giá trị K = 3 hoặc 5 dường như cho hiệu suất tốt nhất. Sai số trên tập kiểm tra thấp nhất ở giá trị K = 3 và 5, nhưng cần xem xét thêm các yếu tố khác như hiện tượng overfitting và underfitting.

- Tăng sai số với K lớn: Có thể thấy rằng khi K tăng lên, sai số trên cả tập huấn luyện và tập kiểm tra tăng dần. Điều này thể hiện mô hình có thể mất đi khả năng tổng quát hóa khi K quá lớn.

**Bộ dữ liệu năm 2021**

A graph with numbers and lines

Description automatically generated

- Sai số rất thấp: Cả sai số trên tập huấn luyện và tập kiểm tra đều rất thấp và gần với 0. Điều này cho thấy mô hình KNN đã hoạt động rất tốt trên tập dữ liệu của bạn

- Khả năng tổng quát hóa tốt: Sai số trên cả tập huấn luyện và tập kiểm tra đều rất thấp và gần nhau. Điều này cho thấy mô hình có khả năng tổng quát hóa tốt trên dữ liệu chưa từng thấy.

- Sự ổn định với K tăng lên: Có thể thấy rằng sai số trên tập kiểm tra không tăng quá nhiều khi K tăng lên. Điều này thể hiện mô hình có khả năng tổng quát hóa tốt và không bị ảnh hưởng mạnh bởi thay đổi giá trị K.

- Tăng sai số với K lớn: Có một sự tăng nhẹ về sai số trên tập kiểm tra khi K tăng lên (ví dụ: từ 9 đến 11). Điều này thể hiện khả năng mất khả năng tổng quát hóa khi K quá lớn.

- Cân nhắc lựa chọn K: Đối với bài toán của bạn, việc lựa chọn giá trị K = 3 hoặc 5 có thể là lựa chọn tốt vì đều cho sai số rất thấp trên cả tập huấn luyện và tập kiểm tra.

## Khai phá Luật kết hợp:

### Đối với các bộ dữ liệu Forbes Billionaires 2020, 2021, 2022:

Khai phá dựa trên các đặc trưng: Country, Sales, Profit, Assets, Market Value.

Lựa chọn các ngưỡng cho Age < 30, Net Worth > 20. Đồng thời lựa chọn ngưỡng Support là 0.035 và Confidence là 0.5.

A screen shot of a computer code

Description automatically generated

A screen shot of a computer code

Description automatically generated

Kết quả sau khi chạy đoạn code trên:

1. Đối với bộ dữ liệu 2020\_forbes\_billionaires:

A close-up of a number

Description automatically generated

2. Đối với bộ dữ liệu 2021\_forbes\_billionaires:

A close-up of a number

Description automatically generated

3. Đối với bộ dữ liệu 2022\_forbes\_billionaires:

A close-up of a number

Description automatically generated

### Đối với các bộ dữ liệu Forbes Companies 2020, 2021, 2022:

Khai phá dựa trên các đặc trưng: Country, Sales, Profit, Assets, Market Value.

Lựa chọn các ngưỡng cho Sales > 100, Profit > 5, Assets > 100 và Market Value > 100. Đồng thời lựa chọn ngưỡng Support là 0.035 và Confidence là 0.5.

A computer code with text

Description automatically generated

A screen shot of a computer code

Description automatically generated

Kết quả sau khi chạy đoạn code trên:

1. Đối với bộ dữ liệu Forbes\_2020:

A close-up of a number

Description automatically generated

2. Đối với bộ dữ liệu Forbes\_2021:

A screenshot of a computer

Description automatically generated

3. Đối với bộ dữ liệu Forbes\_2022:

A screenshot of a computer screen

Description automatically generated

## Cây quyết định:

### Đối với các bộ dữ liệu Forbes Billionaires 2020, 2021, 2022:

Lựa chọn các cột đặc trưng bao gồm: Net Worth, Age, Country, Industries

A computer screen shot of a code

Description automatically generated

A screen shot of a computer

Description automatically generated

Kết quả sau khi chạy đoạn code:

1. 2020\_forbes\_billionaires: Accuracy = 0.9964



2. 2021\_forbes\_billionaires: Accuracy = 0.9991



3. 2022\_forbes\_billionaires: Accuracy = 0.9971



### Đối với các bộ dữ liệu Forbes Companies 2020, 2021, 2022:

Lựa chọn các cột đặc trưng bao gồm: Country, Sales, Profit, Assets, Market Value

A screen shot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Kết quả sau khi chạy code:

1. Đối với bộ dữ liệu Forbes-2020: Accuracy = 0.9912

A diagram of a flowchart

Description automatically generated

2. Đối với bộ dữ liệu Forbes-2021: Accuracy = 0.9970

A diagram of a graph

Description automatically generated with medium confidence

3. Đối với bộ dữ liệu forbes\_companies\_2022: Accuracy = 0.9981

A diagram of a diagram

Description automatically generated with medium confidence

TÀI LIỆU THAM KHẢO