

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

**======\*\*\*======**

****

**BÁO CÁO BÀI TẬP LỚN**

**MÔN: KHO DỮ LIỆU VÀ CÁC PHƯƠNG PHÁP KHAI PHÁ**

**ĐỀ TÀI: Phân cụm bằng K-Means: Tìm hiểu giải thuật K-Means; ưu, nhược điểm của k-means; thu thập và tiền xử lý dữ liệu phân cụm UCI; cài đặt giải thuật k-means trên dữ liệu đã xử lý; tổng hợp và đánh giá kết quả thu được.**

**Giáo viên hướng dẫn : An Văn Minh**

**Nhóm** : 3

**Sinh viên:**

1. Đàm Thị Kim Phượng
2. Tô Anh Quân
3. Đoàn Mạnh Cường

**Lớp:** **Khóa:** K12

**Hà Nội – 2021**

MỤC LỤC

[Chương 1. TỔNG QUAN VỀ KHAI PHÁ DỮ LIỆU 2](#_Toc72366730)

[1.1 Giới thiệu về khai phá dữ liệu và khám phá tri thức 2](#_Toc72366731)

[1.1.1 Ứng dụng của khai phá dữ liệu 2](#_Toc72366732)

[1.1.2 Quá trình khám phá tri thức từ cơ sở dữ liệu 2](#_Toc72366733)

[1.2 Giới thiệu chung về weka 3](#_Toc72366734)

[1.2.1 Khái niệm 3](#_Toc72366735)

[1.2.2 Ưu điểm của Weka 4](#_Toc72366736)

[1.2.3 Những tính năng vượt trội của Weka 5](#_Toc72366737)

[1.2.4 Kiến trúc thư viện Weka 6](#_Toc72366738)

[1.2.5 Cài đặt Weka 7](#_Toc72366739)

[1.2.6 Loại phiên bản 7](#_Toc72366740)

[1.2.7 Môi trường hỗ trợ 7](#_Toc72366741)

[1.3 Tìm hiểu chung về Weka 8](#_Toc72366742)

[1.3.1 Menu thành phần của Weka 8](#_Toc72366743)

[1.3.2 Các môi trường chức năng chính của Weka 13](#_Toc72366744)

[1.4 Các định dạng dữ liệu của WEKA 14](#_Toc72366745)

[1.4.1 Định dạng ARFF 15](#_Toc72366746)

[1.4.2 Sparse ARFF Files 17](#_Toc72366747)

[Chương 2. PHÂN CỤM BẰNG K-MEANS 18](#_Toc72366748)

[2.1 Lịch sử 18](#_Toc72366749)

[2.2 Mô tả chung 18](#_Toc72366750)

[2.3 Thuật toán K-Means 19](#_Toc72366751)

[2.4 Cách chọn nhóm khởi tạo 20](#_Toc72366752)

[2.5 Nhận xét ưu nhược điểm 20](#_Toc72366753)

[2.6 Ứng dụng của thuật toán 21](#_Toc72366754)

[2.7 Ví dụ minh họa : 21](#_Toc72366755)

[Chương 3. ỨNG DỤNG BÀI TOÀN PHÂN LOẠI RƯỢU 25](#_Toc72366756)

[3.1 Tiến hành 25](#_Toc72366757)

[3.1.1 Thực hiện chỉnh sửa dữ liệu 25](#_Toc72366758)

[*3.1.2* Kết quả thu được 30](#_Toc72366759)

[**KẾT LUẬN** 31](#_Toc72366760)

[**TÀI LIỆU THAM KHẢO** 32](#_Toc72366761)

**LỜI MỞ ĐẦU**

Ngày nay, khi mà thời buổi công nghiệp hóa, hiện đại hóa đang ngày càng phát triển trên toàn cầu thì công nghệ thông tin cũng trở thành trợ thủ đắc lực của con người trong cuộc sống và ở bất kỳ quốc gia nào để có thể hòa nhập với thế giới. Đứng trước những thách thức, trước sự bùng nổ mạnh mẽ của thời đại công nghệ 4.0, cũng như việc ứng dụng công nghệ thông tin trong nhiều lĩnh vực của đời sống, kinh tế xã hội thì Khai phá tri thức là một lĩnh vực được tạo ra và phát triển để trợ giúp công tác ra quyết định, đặc biệt đối với các hoạt động ra quyết định kinh doanh hoặc tổ chức.

Cùng với sự ra đời của Khai phá tri thức, phần mềm Weka cũng được tạo ra để hỗ trợ đắc lực cho con người trong việc phân tích dữ liệu và lập các mô hình dự đoán, phục vụ tốt nhất cho quá trình tổ chức, ra quyết định. Chính vì vậy bọn em xin chọn đề tài Tìm hiểu giải thuật Kmeans để nghiên cứu.

Trong suốt quá trình hoàn thành bài tập lớn cũng không tránh khỏi những sai sót. Rất mong nhận được sự góp ý của thầy An Văn Minh để bài tập lớn của bọn em có thể hoàn thiện hơn.

*Chúng em xin chân thành cảm ơn!*

# 

# TỔNG QUAN VỀ KHAI PHÁ DỮ LIỆU

## Giới thiệu về khai phá dữ liệu và khám phá tri thức

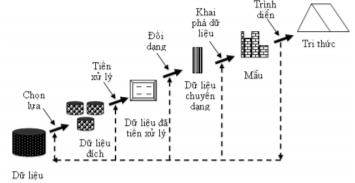
**Khai phá dữ liệu** (*data mining*) Là quá trình tính toán để tìm ra các mẫu trong các bộ dữ liệu lớn liên quan đến các phương pháp tại giao điểm của máy học, thống kê và các hệ thống cơ sở dữ liệu. Đây là một lĩnh vực liên ngành của khoa học máy tính.Mục tiêu tổng thể của quá trình khai thác dữ liệu là trích xuất thông tin từ một bộ dữ liệu và chuyển nó thành một cấu trúc dễ hiểu để sử dụng tiếp.Ngoài bước phân tích thô, nó còn liên quan tới cơ sở dữ liệu và các khía cạnh quản lý dữ liệu, xử lý dữ liệu trước, suy xét mô hình và suy luận thống kê, các thước đo thú vị, các cân nhắc phức tạp, xuất kết quả về các cấu trúc được phát hiện, hiện hình hóa và cập nhật trực tuyến.Khai thác dữ liệu là bước phân tích của quá trình "**khám phá tri thức** trong cơ sở dữ liệu.

Khám phá hay phát hiện tri thức trong cơ sở dữ liệu là một quy trình nhận biết các mẫu hoặc các mô hình trong dữ liệu với các tính năng: Phân tích, tổng hợp, hợp thức, khả ích, và có thể hiểu được. Còn khai phá dữ liệu là một bước trong quy trình khám phá tri thức gồm có các thuật toán khai thác dữ liệu chuyên dùng dưới một số quy định về hiệu quà tinh toán chấp nhận được để tìm ra các mâu hoặc các mô hình trong dữ liệu. Nói một cách khác, mục đích của phát hiện trí thức và khai phá dữ liệu chính là tìm ra các và/hoặc các mô hình đang tồn tại trong các cơ sở dữ liệu nhưng vẫn còn bị che khuất bởi hàng núi dữ liệu.

### Ứng dụng của khai phá dữ liệu

* Phân tích dữ liệu tài chính (Financial Data Analysis)
* Công nghiệp bán lẻ (Retail Industry)
* Công nghiệp viễn thông (Telecommunication Industry)
* Phân tích dữ liệu sinh học (Biological Data Analysis)
* Phát hiện xâm nhập (Intrusion Detection)
* Một số ứng dụng trong khoa học (Scientific Applications)

### Quá trình khám phá tri thức từ cơ sở dữ liệu



*Hình 1.1: Quá trình phát hiện tri thức từ cơ sở dữ liệu*

* Làm sạch dữ liệu: Loại bỏ dữ liệu nhiều hoặc dữ liệu không thích hợp.
* Tích hợp dữ liệu: Tích hợp dữ liệu từ các nguồn khác nhau.
* Chọn dữ liệu: Chọn những dữ liệu liên quan trực tiếp đến nhiệm vụ.
* Chuyển đổi dữ liệu: Chuyển dữ liệu về những dạng phù hợp cho việc khai phá
* Khai phá dữ liệu.
* Đánh giá mẫu: Đánh giá mẫu hoặc tri thức đã thu được.
* Trình diễn dữ liệu: Biểu diễn những tri thức khai phá được cho người sử dụng.

## Giới thiệu chung về weka

### Khái niệm

Weka là một bộ phần mềm mã nguồn mở miễn phí khai thác dữ liệu, được xây dựng bằng ngôn ngữ lập trình Java, theo kiến trúc hướng đối tượng, được tổ chức thành thư viện phục vụ cho lĩnh vực học máy và khai phá dữ liệu.

Weka là phần mềm thuộc dự án phát triển của Đại học Waikato, New Zealand, nó là tên viết tắt của Waikato Environment for Knowledge Analysis. Ngoài khuôn khổ trường đại học, Weka được phát âm vần điệu với Mecca, đây là một loài chim không bay với bản chất tò mò chỉ tìm thấy trên các đảo ở New Zealand, đồng thời, loài chim này cũng chính là biểu tượng cho phần mềm Weka.



*Hình 1.2****:*** *Loài chim Mecca được chọn là biểu tượng của ứng dụng Weka*

Weka cung cấp các thuật toán học tập mà bạn có thể dễ dàng áp dụng cho các bộ dữ liệu (dataset). Nó cũng bao gồm một loạt các công cụ để chuyển đổi các bộ dữ liệu như các thuật toán để phân loại và lấy mẫu mà không cần phải viết bất kỳ mã chương trình nào.

Theo KDNuggets (2005) đánh giá, Weka là sản phẩm khai thác dữ liệu được sử dụng nhiều nhất, nổi bật và hiệu quả nhất năm 2005.

Weka chứa một bộ sưu tập các công cụ trực quan và các thuật toán để phân tích dữ liệu cũng như lập các mô hình dự đoán, cùng với các giao diện người dùng đồ họa để dễ dàng truy cập các chức năng này.

### Ưu điểm của Weka

Phần mềm mã nguồn mở Weka bắt đầu phát triển mạnh mẽ vào năm 1997 và được sử dụng trong nhiều lĩnh vực ứng dụng khác nhau, đặc biệt cho mục đích giáo dục và nghiên cứu bởi những ưu điểm sau:

* Tính khả dụng miễn phí theo Giấy phép Công cộng GNU (giấy phép cung cấp bản quyền phần mềm miễn phí, đảm bảo cho người dùng được tự do chạy, nghiên cứu, chia sẻ và thay đổi phần mềm).
* Được thực hiện bởi ngôn ngữ lập trình Java nên Weka có tính di động khi nó chạy được trên hầu hết bất kỳ nền tảng điện toán hiện đại nào đã được thử nghiệm như Linux, Windows hay hệ điều hành Macintosh.
* Tổng hợp toàn diện các kỹ thuật tiền xử lý và mô hình hóa dữ liệu.
* Giao diện đồ họa giúp người dùng dễ sử dụng.

Weka hỗ trợ một số nhiệm vụ khai thác dữ liệu tiêu chuẩn, cụ thể hơn là xử lý trước dữ liệu, phân cụm, phân loại, hồi quy, trực quan hóa và lựa chọn tính năng. Tất cả các kỹ thuật của Weka được dựa trên giả định rằng dữ liệu có sẵn dưới dạng một tệp phẳng hoặc quan hệ, trong đó mỗi điểm dữ liệu được mô tả bằng một số thuộc tính cố định (thông thường là thuộc tính số hoặc danh nghĩa, nhưng một số loại thuộc tính khác cũng được hỗ trợ). Weka cung cấp quyền truy cập vào cơ sở dữ liệu SQL bằng cách sử dụng Java Database Connectivity và có thể xử lý kết quả được trả về bởi truy vấn cơ sở dữ liệu.

### Những tính năng vượt trội của Weka

Như đã nói ở trên, Weka được đánh giá là sản phẩm khai thác dữ liệu nổi bật là bởi những tính năng đặc biệt sau:

* Hỗ trợ nhiều thuật toán máy học và khai thác dữ liệu
* Được tổ chức theo dạng mã nguồn mở để người dùng dễ dàng download và sử dụng
* Độc lập với môi trường do sử dụng các máy ảo Java (JVM – Java virtual machine)
* Dễ sử dụng, kiến trúc dạng thư viện dễ dàng cho việc xây dựng các ứng dụng thực nghiệm
* Giao diện đồ họa (gồm cả tính năng hiển thị hóa dữ liệu)
* Môi trường cho phép so sánh các giải thuật học máy và khai phá dữ liệu.

### Kiến trúc thư viện Weka

Để thuận tiện cho việc sử dụng của người dùng, kiến trúc thư viện của Weka được xây dựng với hơn 600 lớp và được tổ chức thành 10 gói (package), trong đó, mỗi package sẽ thực hiện một nhiệm vụ chức năng riêng biệt trong quá trình khai thác dữ liệu. Để người dùng có thể tự xây dựng ứng dụng của bản thân mình thì người dùng có thể trực tiếp sử dụng những package này để xây dựng các chương trình.

Weka  có một số lượng lớn các công cụ hồi quy và phân loại. Các gói bản địa là các gói được bao gồm trong phần mềm Weka thực thi, trong khi các gói không phải gốc khác có thể được tải xuống và sử dụng trong môi trường R.Weka. Trong số các gói bản địa, công cụ nổi tiếng nhất là gói cây mô hình M5p. Một số công cụ hồi quy phổ biến là:

* M5Rules (thuật toán M5' được trình bày dưới dạng hàm toán học không có cây)
* DecisionStump (giống như M5' nhưng với một đầu ra số duy nhất trong mỗi nút)
* M5P (tách miền thành các vùng nhị phân kế tiếp và sau đó phù hợp với các mô hình tuyến tính cho mỗi nút cây)
* RandomForest (một số cây mô hình kết hợp)
* RepTree (một số cây mô hình kết hợp)
* ZeroR (giá trị trung bình của kết quả đầu ra)
* DecisionRules (chia tách dữ liệu thành nhiều vùng dựa trên một biến độc lập và cung cấp một giá trị đầu ra cho mỗi phạm vi)
* LinearRegression
* Sự hồi quy vector hỗ trợ
* SimpleLinearRegression (sử dụng một biến chặn và chỉ 1 biến đầu vào cho dữ liệu đa biến)
* MultiLayerPerceptron (mạng nơron)
* GaussianProcesses

### Cài đặt Weka

Trang chủ Weka <https://www.cs.waikato.ac.nz/ml/weka/> cung cấp đầy đủ các loại phiên bản trong các môi trường khác nhau. Đồng thời, nó cũng cung cấp hướng dẫn sử dụng ngay trên trang chủ của Weka hoặc trong các file dưới dạng PDF ngay sau khi cài đặt phần mềm. Không những vậy, Weka còn cung cấp các file dữ liệu đã xây dựng sẵn theo định dạng ARFF để tiết kiệm thời gian trong quá trình xây dựng bộ dữ liệu.

### Loại phiên bản

* Snapshot: là những bản vá lỗi mới nhất, những bản vá lỗi này được phát sinh tự động.
* Book version: là các phiên bản được người sử dụng đánh giá là ổn định nhất tính cho đến thời điểm hiện tại.
* Developer version: là các phiên bản mới nhất đã được phát triển, luôn được cập nhật thường xuyên và có nhiều tính năng hơn Book version nhưng vẫn mang khuyết điểm khi có thể chưa được ổn định bằng các phiên bản của Book version.

### Môi trường hỗ trợ

Weka hỗ trợ chạy được trên rất nhiều hệ điều hành khác nhau, trong đó được chia thành hai nhóm chính là những phiên bản dành cho môi trường Windows và các platform (nền tảng) khác như Linux, Macintosh…

Điều kiện để khởi chạy được Weka chính là chương trình của bạn phải nhất định phải đi kèm với máy ảo Java (JVM – Java virtual machine).

## Tìm hiểu chung về Weka

### Menu thành phần của Weka

Phiên bản mới nhất của Weka được cập nhật vào ngày 9/4/2018 thì ứng dụng đã được cập nhật lên phiên bản 3.8.3. Trong phiên bản mới nhất, menu của Weka gồm 5 thành phần chính:

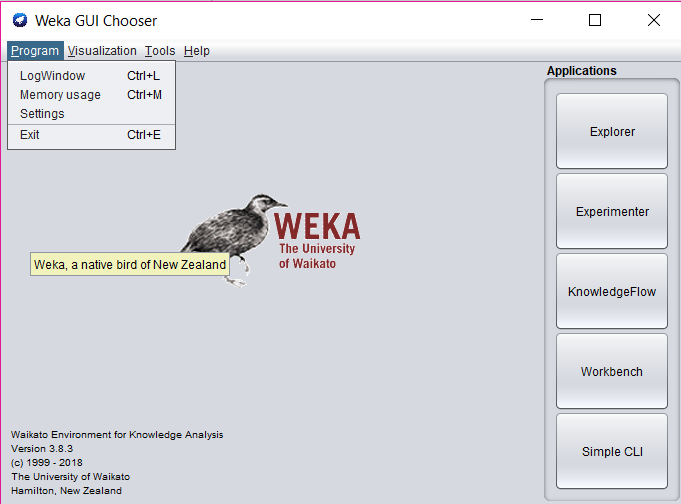


*Hình 1.3:**Màn hình Menu của Weka*

#### Program

Chương trình Program bao gồm:

* LogWindow (Ctrl+L): là nơi ghi lại thông tin hệ thống hay lỗi phát sinh trong suốt quá trình sử dụng phần mềm.
* Memory usage (Ctrl+M): sử dụng bộ nhớ.
* Settings: cài đặt Weka.
* Exit (Ctrl+E): chức năng đóng chương trình Weka.

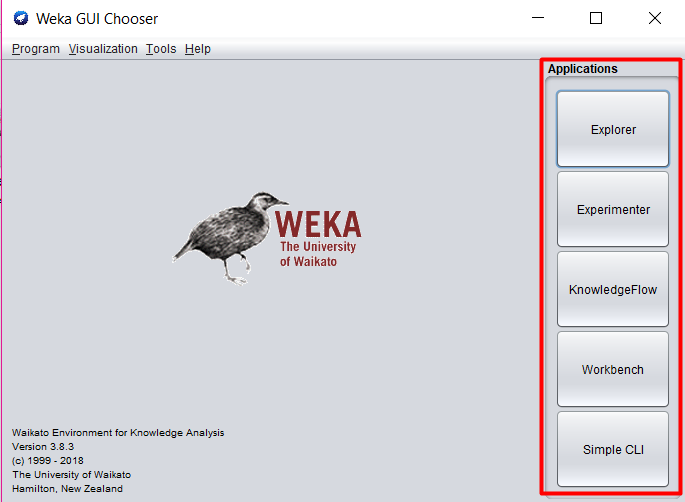
**

*Hình 1.4:**Các chức năng của Program*

#### Applications

Applications chứa các môi trường chức năng chính của Weka

* Explorer: cung cấp môi trường để khảo sát dữ liệu
* Experimenter: môi trường tiến hành các thực nghiệm và kiểm chứng giữa các mô hình học
* Knowledge Flow: cũng có các chức năng tương tự như Explorer nhưng Knowledge Flow lại có giao diện kéo – thả tiện lợi. Đồng thời còn hỗ trợ chức năng học tăng dần (incremental learning).
* Workbench: được coi là ứng dụng tất cả trong một, Workbench là ứng dụng kết hợp tất cả các ứng dụng ở bên trong nó.
* Simple CLI: cung cấp giao diện dòng lệnh để thực thi Weka, tính năng này tương tự như cửa sổ Command Prompt của Windows, nó hỗ trợ cho các hệ điều hành không có chức năng của các cửa sổ dòng lệnh, SimpleCLI sẽ thực thi trực tiếp các lệnh của Weka mà không cần thông qua giao diện.

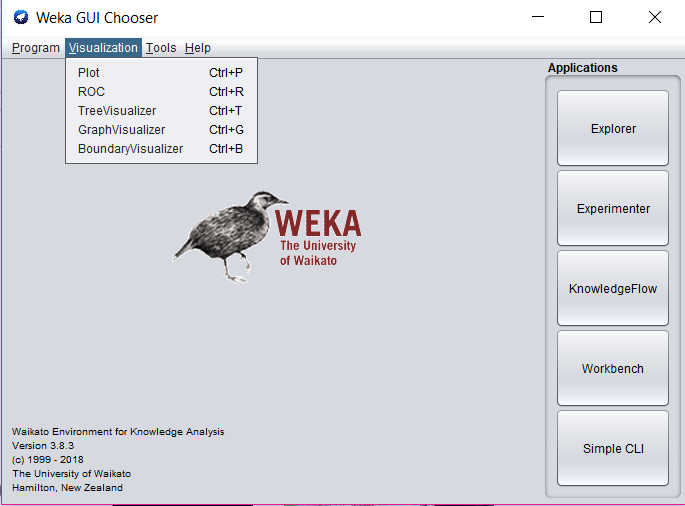
**

*Hình 1.5: Các môi trường chức năng của Application*

#### Visualization

Visualization bao gồm các phương pháp biểu diễn trực quan giúp ta nhận rõ được bản chất của tập dữ liệu:

* Plot (Ctrl+P): vẽ biểu đồ 2D cho một tập dữ liệu
* ROC (Ctrl+R): biểu diễn một đường cong ROC đã lưu trước đó
* TreeVisualizer (Ctrl+T): biểu diễn đồ thị có hướng (ví dụ như cây quyết định…)
* GraphVisualizer (Ctrl+G): biểu diễn đồ thị tổng quát
* BoundaryVisualizer (Ctrl+B): biểu diễn 2D về biên giữa các lớp.

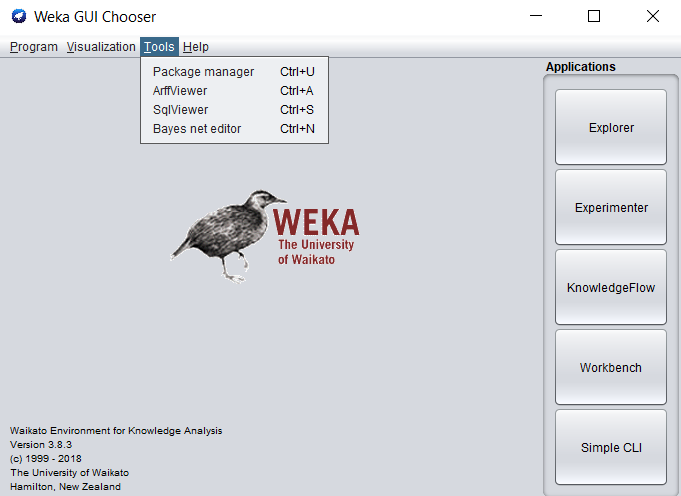


*Hình 1.6: Các chức năng của Visualization*

#### Tools

Một số ứng dụng tiện ích khác được tích hợp trong Tools:

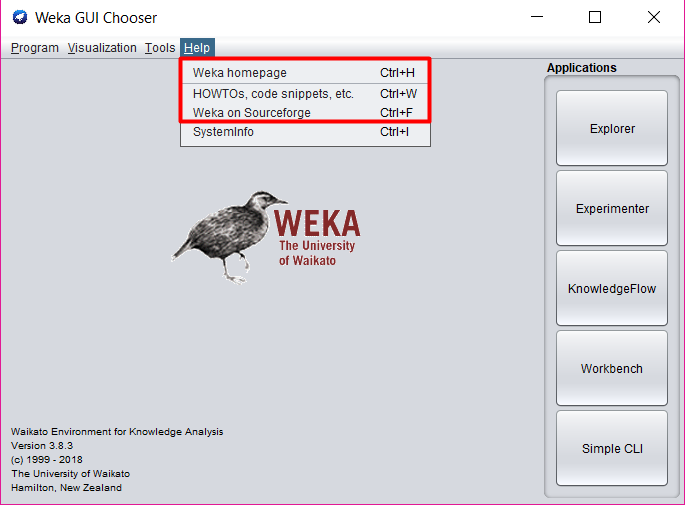
* Package manager (Ctrl+U): Một giao diện đồ họa cho trình quản lý gói của Weka
* ArffViewer (Ctrl+A): là ứng dụng cho phép xem các file ARFF dưới dạng bảng tính, nó giống như một file text với các dữ liệu được lưu nối tiếp nhau, giúp ta có cái nhìn trực quan hơn thay vì nhìn trực tiếp vào nội dung của file ARFF
* SqlViewer (Ctrl+S): là chức năng dùng để kết nối các dữ liệu từ bên ngoài khi biểu diễn bảng SQLđể truy vấn cơ sở dữ liệu.
* Bayes net editor (Ctrl+N): ứng dụng để chỉnh sửa, hiển thị và tìm hiểu các mạng lưới của Bayes.



*Hình 1.7:**Các chức năng trong Tools*

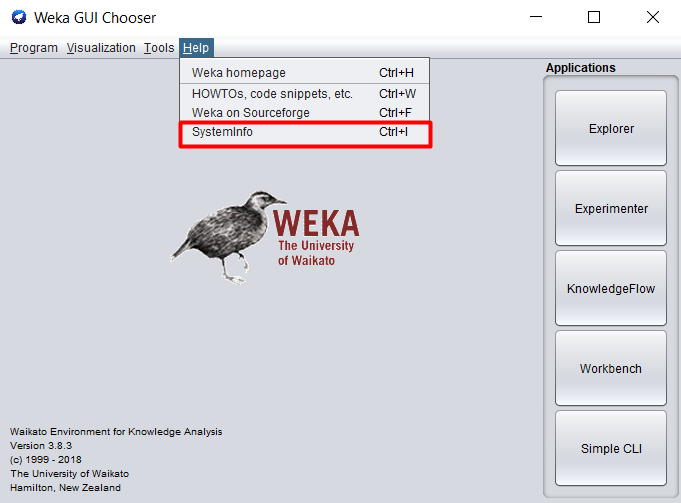
#### Help

Help chứa đường dẫn đến trang chủ của Weka và các trang web khác liên quan đến tài liệu tham khảo cũng như các ví dụ mẫu…



*Hình 1.8:**Help chứa đường dẫn đến trang chủ Weka*

Systeminfor là thông tin về hệ thống, nó liệt kê một số tham số nội bộ cần thiết để chạy trong môi trường Java/Weka. Ví dụ như CLASSPATH, đây là tham số liên quan đến vấn đề của ngôn ngữ Java…



*Hình 1.9:**Systeminfo là thông tin về hệ thống*

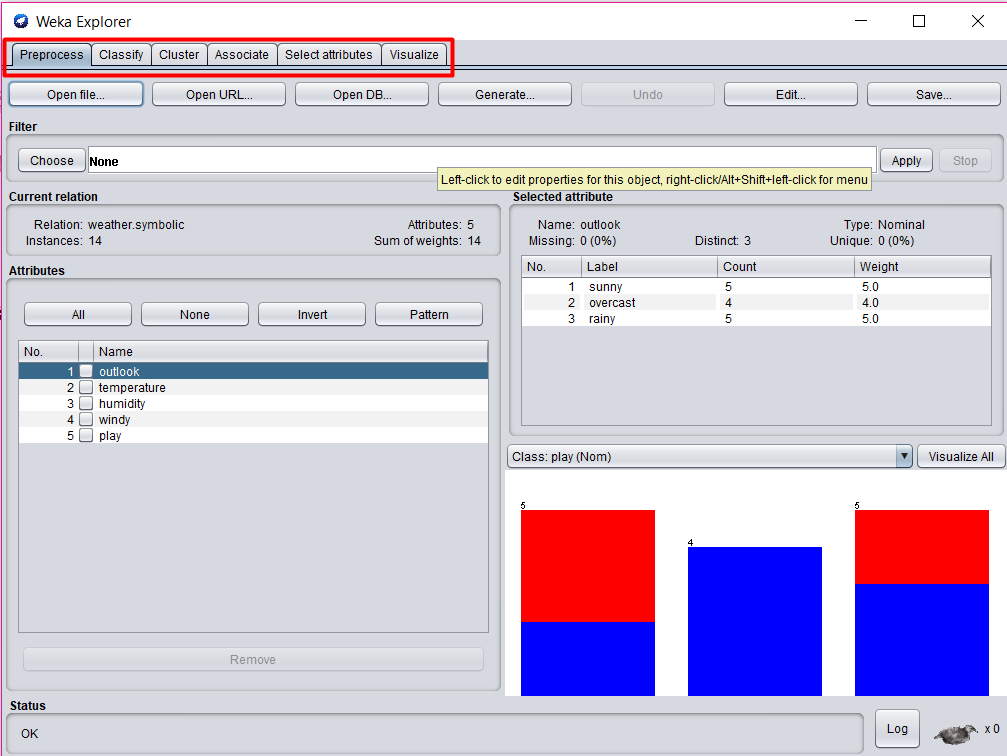
### Các môi trường chức năng chính của Weka

Như đã nói ở trên, trong Applications có chứa 4 môi trường là:

* Simple CLI với giao diện đơn giản và kiểu dòng lệnh (như MS-DOS)
* Explorer là môi trường cho phép thực hiện tất cả các chức năng của Weka để khám phá dữ liệu.
* Experimenter là môi trường cho phép tiến hành các thí nghiệm và thực hiện các kiểm tra thống kê (statistical tests) giữa các mô hình học máy.
* KnowledgeFlow là môi trường cho phép bạn tương tác đồ họa kiểu kéo/thả để thiết kế các bước (các thành phần) của một thí nghiệm.

Trong số 4 môi trường trên thì môi trường Explorer là môi trường được sử dụng nhiều nhất. Trong Explorer có các chức năng chính là:

* Preprocess: để chọn và thay đổi (xử lý) dữ liệu làm việc
* Classify: để huấn luyện và kiểm tra các mô hình học máy (phân loại, hoặc hồi quy/dự đoán)
* Cluster: để học các nhóm từ dữ liệu (phân cụm)
* Associate: để khám phá các luật kết hợp từ dữ liệu
* Select attributes: để xác định và lựa chọn các thuộc tính liên quan (quan trọng) nhất của dữ liệu
* Visualize: để xem (hiển thị) biểu đồ tương tác 2 chiều đối với dữ liệu

**

*Hình 1.10:**Các chức năng trong Explorer*

## Các định dạng dữ liệu của WEKA

Weka hỗ trợ khá nhiều định dạng như:

* ARFF: \*.arff
* CVS files: \*.cvs
* C4.5 files: \*.name, \*.data
* XRFF files: \*.xrff

Trong đó, ARFF là định dạng chính được sử dụng phổ biến nhất trong Weka.

### Định dạng ARFF

**Tổng quan về ARFF**

ARFF là định dạng được sử dụng chính trong Weka, được biểu diễn dưới một dạng file text có thể đọc được một cách đơn giản bằng các chương trình như notepad… mà không cần phải có chương trình chuyên dụng. Các dữ liệu trong file là giá trị thuộc tính của các thể hiện (instance) có chung tập hợp các thuộc tính.

ARFF dễ thao tác và dễ hiểu, người dùng có thể tự tạo cho mình một file ARFF khi hiểu về những quy định chuẩn của nó. Tuy nhiên, vì là định dạng chuyên dụng cho mỗi chương trình Weka nên những ứng dụng khác không thể đọc được nó. Do đó, khi sử dụng kết hợp Weka với các chương trình khác thì người dùng buộc phải xây dựng các module để chuyển kiểu dữ liệu từ ARFF sang kiểu dữ liệu tương ứng với chương trình đó.

**ARFF gồm 2 phần riêng biệt:**

* Header: chứa tên quan hệ, danh sách các thuộc tính và kiểu dữ liệu của chúng.
* Data: chứa thông tin của các instance được tổ chức theo đúng thứ tự khai báo thuộc tính.

**Ký hiệu trong file ARFF**

* Khai báo **@relation** đặt tên quan hệ:

**@relation <relation-name>**

Trong đó:

* relation-name là một chuỗi
* Nếu relation-name chứa khoảng trắng thì phải đặt trong dấu ngoặc kép

*Ví dụ*: @relation 67DCHT22

@relation “class 67DCHT22”

* Khai báo **@attribute**: khai báo tên thuộc tính và kiểu dữ liệu của thuộc tính đó:

**@attribute <attribute-name> <datatype>**

Trong đó:

* attribute-name phải bắt đầu bằng ký tự chữ cái, nếu chứa khoảng trắng thì phải đặt trong dấu ngoặc kép
* datatype là 1 trong 4 loại sau:

- numeric: real, integer

*Ví dụ:* @attribute 67DCHT22 real

@attribute 67HT22 integer

@attribute HT22 numeric

- nominal

*Ví dụ:* @attribute class {67DCHT21,67DCHT22,67DCHT23}

- string

- date: **@attribute <name> date [<date format>]**

Trong đó:

**<date format>** là định dạng dữ liệu ngày tháng, nó cho phép biểu diễn ngày, tháng, năm và giờ, phút, giây: yyyy-MM-dd HH:mm:ss

* Khai báo **@data**: báo hiệu bắt đầu phần dữ liệu:

**@data**

Trong đó:

* Cụm @data sẽ nằm trên một dòng đơn
* Mỗi instance được thể hiện bằng một dòng, kết thúc bằng cặp ký tự xuống dòng, giá trị các thuộc tính xuất hiện theo đúng thứ tự khai báo
* Các thuộc tính thiếu giá trị được biểu diễn bằng dấu (?)

### Sparse ARFF Files

Sparse ARFF là một dạng mở rộng của file ARFF. Cấu trúc phần header của sparse ARFF hoàn toàn tương tự với file ARFF nguyên thủy, tuy nhiên, nó chỉ khác ở những instance được biểu diễn trong phần data.

Phần data không biểu diễn tường mình số 0.

* Mỗi thể hiện được đặt trong dấu {}
* Cú pháp: **<index> <space> <value>**
* Với index là chỉ mục thuộc tính (bắt đầu từ 0)

*Ví dụ:*

ARFF: @data

0, X, 0, Y “class A”

0, 0, W, 0 “class B”

Sparse ARFF: @data

{1 X, 3 Y, 4 “class A”}

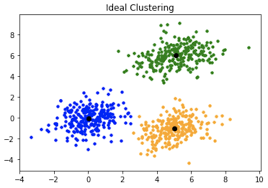
{2 W, 4 “class B”}

# PHÂN CỤM BẰNG K-MEANS

**Phân cụm k-means** là 1 phương pháp [lượng tử hóa vector](https://vi.wikipedia.org/w/index.php?title=L%C6%B0%E1%BB%A3ng_t%E1%BB%AD_h%C3%B3a_vector&action=edit&redlink=1) dùng để phân các điểm dữ liệu cho trước vào các cụm khác nhau. Phân cụm k-means có nhiều ứng dụng, nhưng được sử dụng nhiều nhất trong [Trí tuệ nhân tạo](https://vi.wikipedia.org/wiki/Tr%C3%AD_tu%E1%BB%87_nh%C3%A2n_t%E1%BA%A1o) và [Học máy](https://vi.wikipedia.org/wiki/H%E1%BB%8Dc_m%C3%A1y)

## Lịch sử

Thuật ngữ “K-means" được [James MacQueen](https://vi.wikipedia.org/w/index.php?title=James_MacQueen&action=edit&redlink=1) sử dụng lần đầu tiên vào năm 1967,  mặc dù ý tưởng này quay trở lại Hugo Steinhaus vào năm 1956.  [Thuật toán](https://vi.wikipedia.org/wiki/Thu%E1%BA%ADt_to%C3%A1n) tiêu chuẩn được đề xuất lần đầu tiên bởi Stuart Lloyd của [Bell Labs](https://vi.wikipedia.org/w/index.php?title=Bell_Labs&action=edit&redlink=1) vào năm 1957 như một kỹ thuật cho [điều chế mã xung](https://vi.wikipedia.org/w/index.php?title=%C4%90i%E1%BB%81u_ch%E1%BA%BF_m%C3%A3_xung&action=edit&redlink=1), mặc dù nó không được xuất bản dưới dạng một bài báo cho đến năm 1982.  Năm 1965, Edward W. Forgy đã công bố về cơ bản cùng một phương pháp, đó là lý do tại sao nó đôi khi được gọi là Lloyd-Forgy



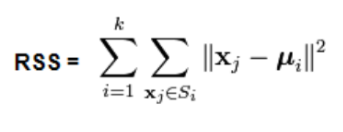
*Hình 2.1: Hình ảnh phân cụm Kmeans*

## Mô tả chung

Thuật toán k-means sử dụng phương pháp tạo và cập nhật trung tâm để phân nhóm các điểm [dữ liệu](https://vi.wikipedia.org/wiki/D%E1%BB%AF_li%E1%BB%87u) cho trước vào các nhóm khác nhau. Đầu tiên chúng sẽ tạo ra các điểm trung tâm ngẫu nhiên. Sau đó gán mỗi điểm trong tập dữ liệu vào trung tâm gần nó nhất. Sau đó chúng sẽ cập nhật lại trung tâm và tiếp tục lặp lại các bước đã kể trên. Điều kiện dừng của thuật toán: Khi các trung tâm không thay đổi trong 2 vòng lặp kế tiếp nhau. Tuy nhiên, việc đạt được 1 kết quả hoàn hảo là rất khó và rất tốn thời gian, vậy nên thường người ta sẽ cho dừng thuật toán khi đạt được 1 kết quả gần đúng và chấp nhận được.

## Thuật toán K-Means

* Cho tập đối tượng D ( X1 , X2 , X3 ... Xn ) có n đối tượng , thuật toán K-Means sẽ phân hoạch n đối tượng vào trong k nhóm (k <= n), U = { c1 , c2 , ... , ck } sao cho tổng sự tương tự (khoảng cách) từ các đối tượng tới trọng tâm của cụm trong tất cả các cụm giảm tối thiểu (RSS Residual Sum of Squares giảm tối thiểu).



* Với µi là trọng tâm của cụm Si
* Ý tưởng chính của thuật toán là định nghĩa điểm trọng tâm của cụm (centroids µi)
* Thuật toán k - means :

Nhập : số cụm k , tập đối tượng D ( X1 , X2 , X ...... x ) có n đối tượng

Bước 1 : Khởi tạo phân hoạch tập đối tượng thành k cụm khác rỗng, tạo ra ma trận phân hoạch khởi tạo

Bước 2 : Tính trọng tâm cho từng cụm (seed, centroid)

Bước 3 : Gán (hoặc gán lại) từng đối tượng vào cụm mà có trọng tâm gần với đối tượng đang xét nhất. Đo khoảng cách lần lượt đối tượng tới các trọng tâm , khoảng cách nào nhỏ nhất thì xếp đối tượng vào cụm có trọng tâm đó, tạo ra ma trận phân hoạch mới

Bước 4 : Nếu không thỏa điều kiện dừng thì quay lại bước 2.

Nếu thỏa mãn điều kiện hội tụ thì dừng lại.

* Điều kiện dùng của thuật toán
* Đạt tới số lượng vòng lặp giới hạn t . Chất lượng gom nhóm có thể sẽ tệ nếu số lượng vòng lặp không đủ
* Ma trận phân hoạch không có thay đổi
* Dừng khi mà RSS giảm tới một ngưỡng cho phép , khi đó phần hoạch đã gần hội tụ , điều này sẽ giảm thời gian chạy thuật toán xuống , hay thuật toán không thể cải thiện làm giảm độ phân biệt hơn được nữa |Un–Un-1| < epsilon thì dừng

## Cách chọn nhóm khởi tạo

Thuật toán K-Means không đảm bảo tối ưu toàn cục . Đối với những trường hợp mà có những đối tượng nằm rất xa so với nhóm đối tượng khác (outlier mẫu cá biệt), nó sẽ không phù hợp với bất cứ nhóm khác thì nên để khởi tạo nó riêng một nhóm , như vậy sẽ không cần phải gán lại trong suốt quá trình lặp . Để lựa chọn một khởi tạo hiệu quả thì nên loại bỏ các đối tượng cá biệt ( outlier ), áp dụng thuật giải di truyền để phát sinh các bộ khởi tạo ...

## Nhận xét ưu nhược điểm

Có thể sử dụng kĩ thuật threading để tính toán song song ma trận trọng tâm , ma trận phân hoạch mới làm tăng tốc độ tính toán

Thuật toán có thể gặp lỗi , sau vài lần lặp sẽ có nhóm không có điểm nào thuộc về , lúc đó phải chạy lại thuật toán với khởi tạo phân hoạch khác , khác với các ma trận phân hoạch đã chạy thất bại trước đó ( bao gồm ma trận phần hoạch khởi tạo và ma trận phân hoạch trong vòng lặp ) hoặc giảm / tăng số nhóm k cần phần hoạch

Ưu điểm của thuật toán là:

* Thực hiện tương đối nhanh
* Độ phức tạp của thuật toán là O (tkn) (t : số lần lặp , k : số cụm cần phân hoạch , n : số điểm cần gom cụm)
* Phù hợp với các cụm có dạng hình cầu

Khuyết điểm của k - means là:

* + Cần xác định số cụm k
  + Khó xác định no toàn cục
  + Phụ thuộc vào điểm khởi tạo
  + Không tốt trong một số trường hơp: Dữ liệu không cân bằng, dữ liệu có quá nhiều clusters, dữ liệu có phân bố đặc biệt.

## Ứng dụng của thuật toán

K-means được sử dụng nhiều trong máy học (học không giám sát) để phân nhóm dữ liệu. Chúng cũng thường được dùng trong phân vùng ảnh.

## Ví dụ minh họa :

**2.7.1 Ví dụ 1 :**

Bộ dữ liệu mà ta xét là một phần của kết quả phân tích các chất trong rượu. Bảng dưới đây chứa số đo etanol, acid malic, magnesium, phenol, độ đậm màu, sắc thái của các mẫu thuộc 1 trong 3 loại rượu.



*Hình 2.2: Ví dụ minh họa 1*

Cột loại rượu chỉ có ý nghĩa tham khảo để đánh giá hiệu quả thuật toán. Do là bài toán nhóm Unsupervised Learning nên các mẫu dữ liệu không có output.Ta sẽ dùng thuật toán K-Means Clustering để phân chia các mẫu dữ liệu thành 3 cụm.

Các bước cài đặt thuật toán

**Bước 1:**Chọn ngẫu nhiên 3 trong số 15 điểm dữ liệu làm trung tâm cụm μ1,μ2,μ3.

**Bước 2:** Gán mỗi điểm dữ liệu về cụm có trung tâm cụm gần nhất.

c(i)=argmink∥x(i)−μk∥2

**Bước 3:** Di chuyển các trung tâm cụm về vị trí trung bình của các điểm dữ liệu trong cụm tương ứng.

μk=1n(x(k1)+x(k2)+…+x(kn))

**Bước 4:** Lặp lại hai bước trên tới khi vị trí các trung tâm cụm không đổi sau một bước lặp nào đó.

**Bước 5:** Lặp lại tất cả các bước trên một số lần và chọn phương án có hàm mất mát nhỏ nhất.

argminc(1),…,c(15),μ1,μ2,μ3J(c(1),…,c(15),μ1,μ2,μ3)

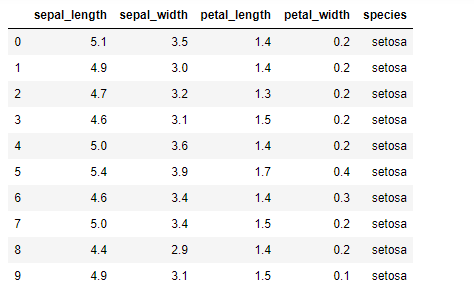
Kết quả :



*Hình 2.2: Kết quả phân cụm*

**2.7.2 Ví dụ 2 :**

Bộ dữ liệu mà ta xét là một phần của kết quả phân tích các loại hoa. Bảng dưới đây chứa số đo sepal\_length, sepal\_width, petal\_length, petal\_width ,các mẫu thuộc 1 trong 2 loại hoa.



*Hình 2.3: Ví dụ minh họa 2*

Cột loại hoa chỉ có ý nghĩa tham khảo để đánh giá hiệu quả thuật toán. Do là bài toán nhóm Unsupervised Learning nên các mẫu dữ liệu không có output.Ta sẽ dùng thuật toán K-Means Clustering để phân chia các mẫu dữ liệu thành 2 cụm.

Các bước cài đặt thuật toán

**Bước 1:**Chọn ngẫu nhiên 2 trong số 10 điểm dữ liệu làm trung tâm cụm μ1,μ2

**Bước 2:** Gán mỗi điểm dữ liệu về cụm có trung tâm cụm gần nhất.

c(i)=argmink∥x(i)−μk∥2

**Bước 3:** Di chuyển các trung tâm cụm về vị trí trung bình của các điểm dữ liệu trong cụm tương ứng.

μk=1n(x(k1)+x(k2)+…+x(kn))

**Bước 4:** Lặp lại hai bước trên tới khi vị trí các trung tâm cụm không đổi sau một bước lặp nào đó.

**Bước 5:** Lặp lại tất cả các bước trên một số lần và chọn phương án có hàm mất mát nhỏ nhất.

argminc(1),…,c(15),μ1,μ2J(c(1),…,c(15),μ1,μ2)

Kết quả :



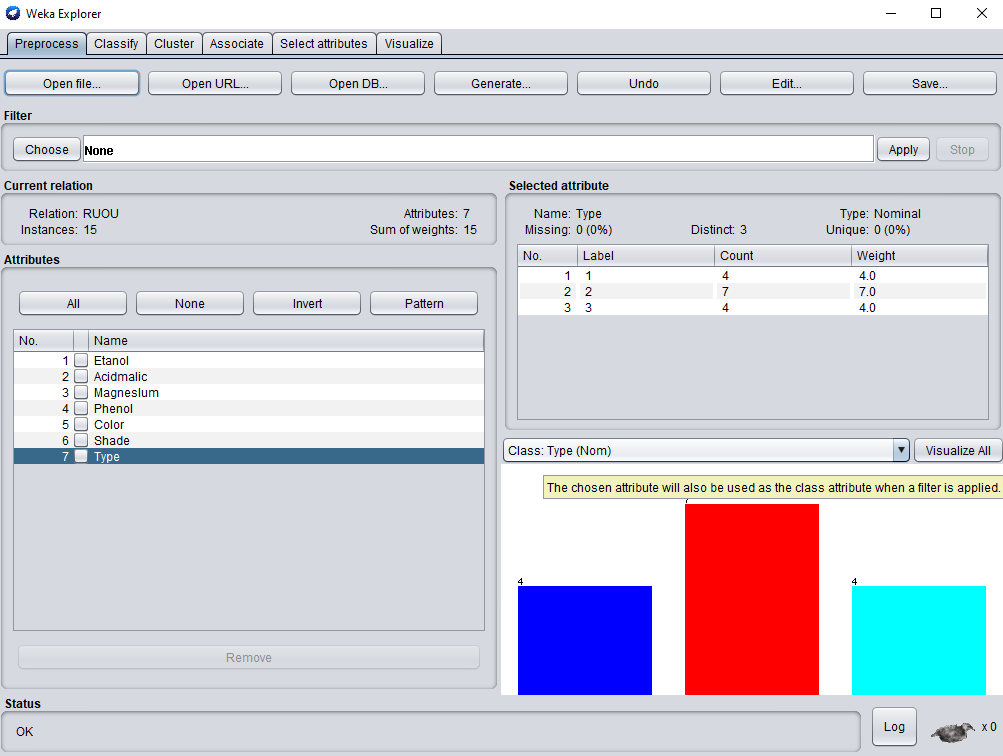
*Hình 2.4: Kết quả phân cụm*

# ỨNG DỤNG BÀI TOÀN PHÂN LOẠI RƯỢU

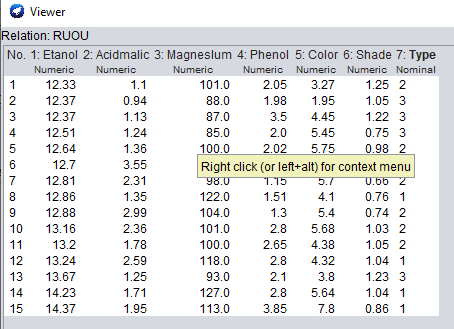
Để có thể hiểu rõ hơn về chức năng cũng như cách thức thực hiện trong ứng dụng Weka, nhóm chúng em đã ứng dụng vào bài toán phân loại rượu.

## Tiến hành

### Thực hiện chỉnh sửa dữ liệu



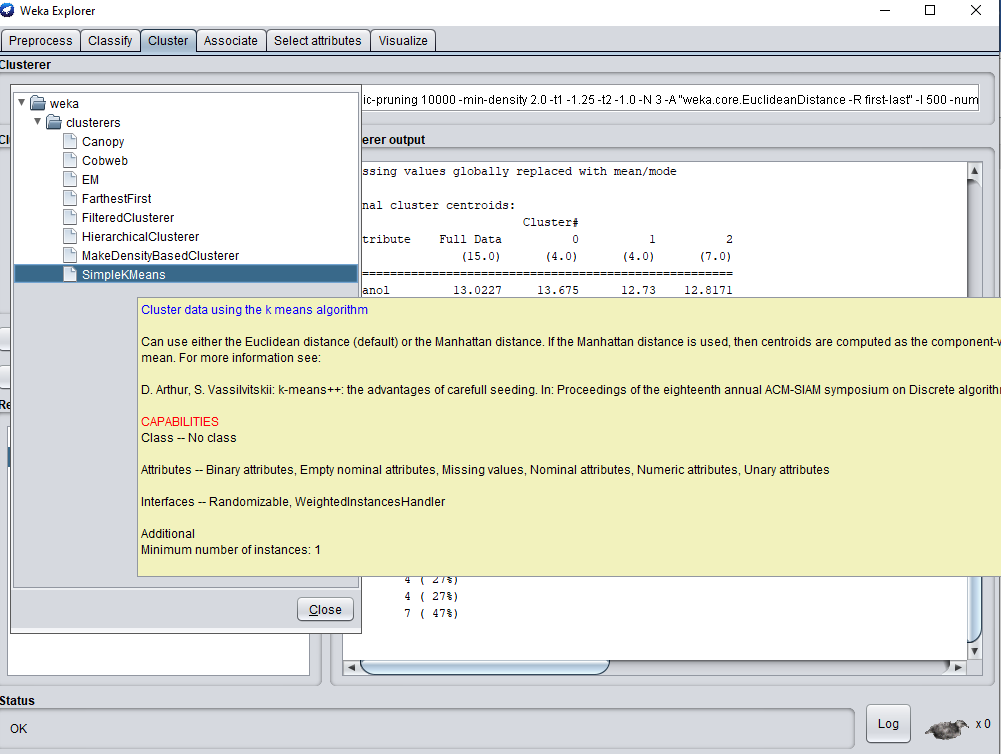
*Hình 3.1: Các thuộc tính của rượu*



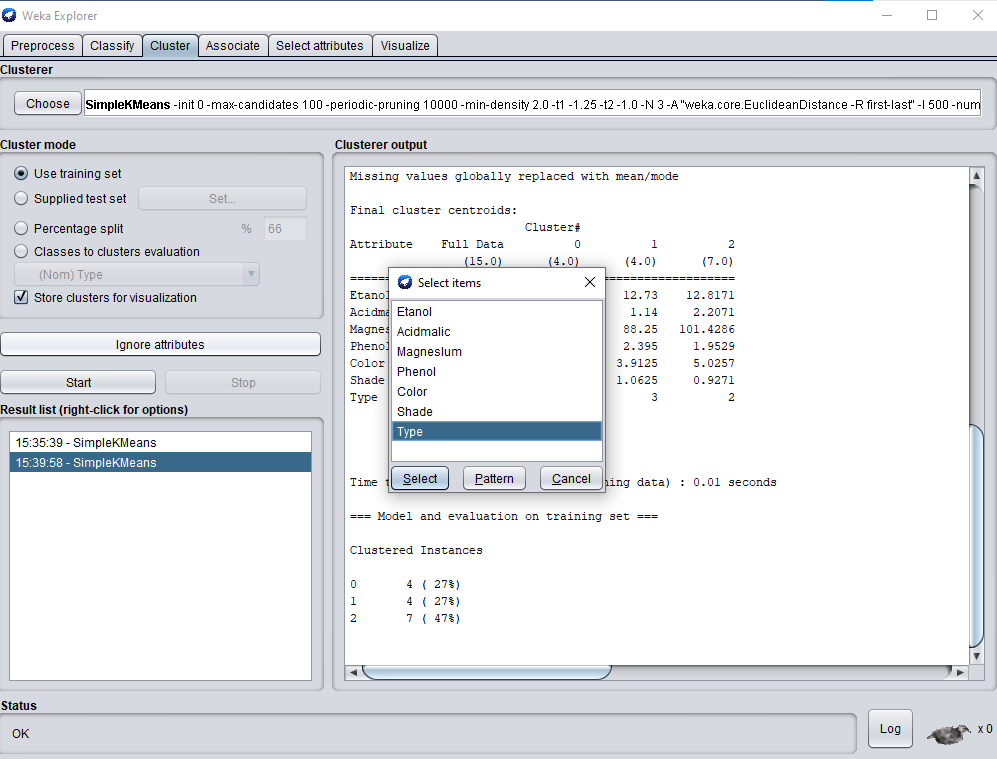
*Hình 3.2:**Dữ liệu chính*

Chúng ta thấy rằng trong *Hình 3.2* có tất cả 15 giá trị và 7 thuộc tính. Để thêm các giá trị còn thiếu, người dùng có thể nhấp chuột vào từng ô để chọn giá trị tương ứng.

Sau khi có được dữ liệu , chúng ta tiến hành chọn thuật toán bên tab Cluster, sau đó chọn Simple K-Means

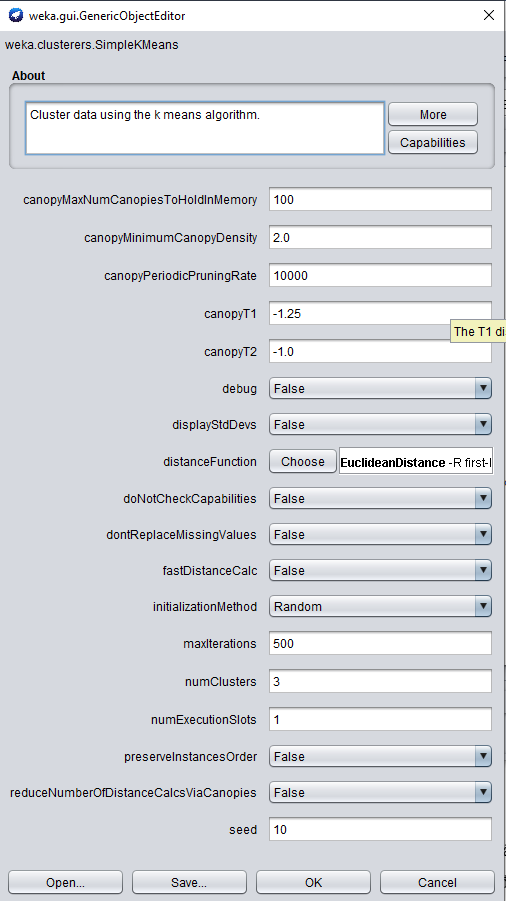
**

*Hình 3.3:**Thuật toán được chọn*

Chọn thuật toán xong , tiếp tục chỉnh sửa một số cài đặt cần thiết như số trọng tâm K=3 , ignore attribute 

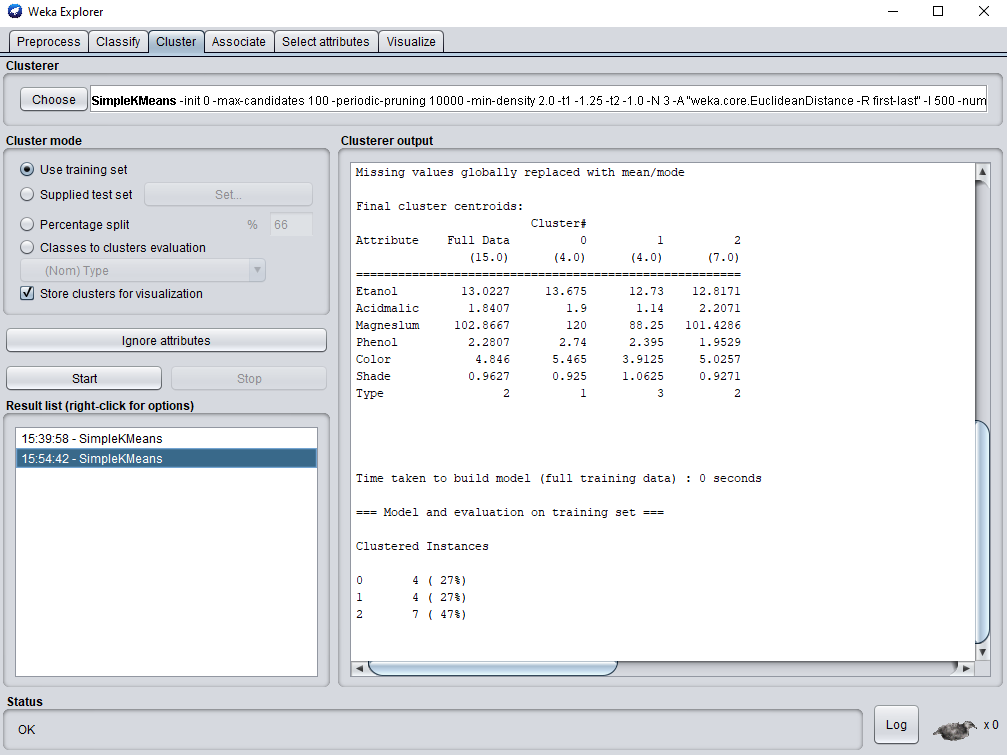
*Hình 3.4:**Ignore Attribute*

Điều chỉnh số trọng tâm tại cột numClusters = 3



*Hình 3.5:**Điều chỉnh số trọng tâm bằng 3*

Tiến hành bấm Start , ta thu được kết quả :



*Hình 3.6****:*** *Kết quả thu được*

### Kết quả thu được

Tại cột Full Data nhận giá trị 15 đồng nghĩa với 15 giá trị được nhập vào từ dữ liệu , 0 ,1 ,2 là các tâm cụm nhận lần lượt các giá trị 4 ,4, 7 có nghĩa rằng :

Cụm 0 có 4 điểm gần nhất : có nghĩa loại rượu 0 có 4 giá trị

Cụm 1 có 4 điểm gần nhất : có nghĩa loại rượu 1 có 4 giá trị

Cụm 2 có 7 điểm gần nhất : có nghĩa loại rượu 2 có 7 giá trị

**KẾT LUẬN**

Việc quyết định luôn cần xử lý một lượng kiến thức lớn nhưng con người lại bị giới hạn bởi nhận thức, kinh tế, thời gian hay áp lực cạnh tranh. Hệ trợ giúp quyết định (DSS) ra đời chính là sự kết hợp trị tuệ của con người và năng lực máy tính để cải thiện chất lượng quyết định, giải quyết vấn đề.

Trong bài báo cáo trên, chúng em đã trình bày tổng quan về ứng dụng Weka, một trong những ứng dụng hỗ trợ thiết yếu trong việc ra quyết định. Trên cơ sở đó để đưa ra một số ví dụ để giải quyết vấn đề, cung cấp kiến thức cho bạn đọc một cái nhìn khái quát về môn học cũng như ứng dụng. Như việc phân tích dữ liệu về các thành phần để đưa ra kết luận về những loại rượu…

Do kiến thức còn hạn hẹp nên bài báo cáo của chúng em còn nhiều thiết xót, rất mong nhận được sự giúp đỡ, góp ý của cô và các bạn.

Chúng em xin chân thành cảm ơn!

**TÀI LIỆU THAM KHẢO**

1. Bài giảng Hệ trợ giúp quyết định:

<https://vi.scribd.com/doc/51949793/Bai-giang-he-tro-giup-quyet-dinh>

1. Trang chủ Weka: <https://www.cs.waikato.ac.nz/ml/weka/>
2. Wikipedia Weka:

<https://en.wikipedia.org/wiki/Weka_(machine_learning)>

1. Data set UCI Machine Learning Repository

<https://archive.ics.uci.edu/ml/datasets.php>