TRƯỜNG ĐẠI HỌC THỦY LỢI

**KHOA CÔNG NGHỆ THÔNG TIN**

A blue and white logo

Description automatically generated with low confidence

**BÀI TẬP LỚN**

HỌC PHẦN: HỌC MÁY

**ĐỀ TÀI:**

**Dự đoán khả năng khách hàng gửi tiền tiết kiệm kỳ hạn ngân hàng**

Giáo viên hướng dẫn: Nguyễn Thị Kim Ngân

Sinh viên/nhóm sinh viên thực hiện:

1. Lê Hoàn, lớp 63CNTT.NB

2. Trình Hữu Tuấn, lớp 63CNTT.VA

3. Nguyễn Quang Thắng, lớp 63CNTT.NB

**Hà Nội, năm 2023**

**MỤC LỤC**

**Phần 1. Tổng quan**

*1. Giới thiệu về học máy*

*2. Phương pháp học máy*

**Phần 2. Thực nghiệm**

*1. Mô tả tập dữ liệu*

*2. Mô tả cách giải bài toán bằng phương pháp học máy*

*3. Đánh giá mô hình*

*4. Mô tả các chức năng của chương trình*

**Phần 3. Kết luận**

**Tài liệu tham khảo**

# **Phần 1: Tổng quan**

# ***1. Giới thiệu về học máy***

- Lịch sử và vai trò của machine learning:

+ Những năm gần đây, AI - Artificial Intelligence (Trí Tuệ Nhân Tạo), và cụ thể hơn là Machine Learning (Học Máy hoặc Máy Học) nổi lên như một bằng chứng của cuộc cách mạng công nghiệp lần thứ tư (1 - động cơ hơi nước, 2 - năng lượng điện, 3 - công nghệ thông tin). Trí Tuệ Nhân Tạo đang len lỏi vào mọi lĩnh vực trong đời sống mà có thể chúng ta không nhận ra. Xe tự hành của Google và Tesla, hệ thống tự tag khuôn mặt trong ảnh của Facebook, trợ lý ảo Siri của Apple, hệ thống gợi ý sản phẩm của Amazon,...

+ Machine Learning là một tập con của AI,là một lĩnh vực nhỏ của Khoa Học Máy Tính, nó có khả năng tự học hỏi dựa trên dữ liệu đưa vào mà không cần phải được lập trình cụ thể.

+ Những năm gần đây, khi mà khả năng tính toán của các máy tính được nâng lên một tầm cao mới và lượng dữ liệu khổng lồ được thu thập bởi các hãng công nghệ lớn, Machine Learning đã tiến thêm một bước dài và một lĩnh vực mới được ra đời gọi là Deep Learning. Nó đã giúp máy tính thực thi những việc tưởng chừng như không thể như phân loại cả ngàn vật thể khác nhau trong các bức ảnh, tự tạo chú thích cho ảnh, bắt chước giọng nói và chữ viết của con người, giao tiếp với con người.

- Ưu điểm của Học có giám sát và không giám sát:

+ Học có giám sát: Supervised learning là thuật toán dự đoán đầu ra (outcome) của một dữ liệu mới (new input) dựa trên các cặp (*input, outcome*) đã biết từ trước. Cặp dữ liệu này còn được gọi là (*data, label*), tức (*dữ liệu, nhãn*). Supervised learning là nhóm phổ biến nhất trong các thuật toán Machine Learning. Chúng học thông qua một bộ dữ liệu đã được gán nhãn chính xác và cố gắng đưa ra các dự đoán đầu ra mục tiêu chính xác nhất có thể trên bộ dữ liệu cho trước.

+ Học không giám sát: Thuật toán unsupervised learning sẽ dựa vào cấu trúc của dữ liệu để thực hiện một công việc nào đó, ví dụ như phân nhóm (clustering) hoặc giảm số chiều của dữ liệu (dimension reduction) để thuận tiện trong việc lưu trữ và tính toán.

- Hạn chế của Học có giám sát và không giám sát: Có một phương pháp học rơi vào giữa 2 phương pháp này là học bán giám sát (semi-supervised learning) khi dữ liệu học chỉ có một phần nhỏ là có đầy đủ cặp đầu vào - đầu ra tương ứng và phần dữ liệu còn lại chỉ có đầu vào. Phương pháp này phù hợp trong trường hợp việc gắn đầu ra cho từng đầu vào là không khả thi hoặc tốn rất nhiều tài nguyên và việc sử dụng dữ liệu không được gắn đầu ra sẽ hiệu quả hơn.

Học không giám sát: chúng ta không biết câu trả lời chính xác cho mỗi dữ liệu đầu vào. Giống như khi ta học, không có thầy cô giáo nào chỉ cho ta biết đó là chữ A hay chữ B. Cụm *không giám sát* được đặt tên theo nghĩa này. Các dữ liệu không được "hướng dẫn" trước như trong trường hợp học có giám sát.

***2. Phương pháp học máy*** - ***thuật toán Decision Tree***

* **Mục đích của phương pháp:** là phân loại dữ liệu. Phương pháp này sử dụng một cây quyết định để phân loại dữ liệu. Cây quyết định là một cấu trúc dữ liệu phân cấp, trong đó mỗi nút đại diện cho một câu hỏi hoặc tiêu chí. Các nút con của một nút đại diện cho các kết quả có thể của câu hỏi hoặc tiêu chí đó.
* **Input:** tập dữ liệu huấn luyện
* **Output:** là một cây quyết định. Cây quyết định có thể được sử dụng để phân loại các mẫu dữ liệu mới bằng cách duyệt qua cây và trả về nhãn mục tiêu của nút lá mà mẫu dữ liệu kết thúc.
* **Cách thực hiện:**
  + Chúng ta cần xác định thứ tự của thuộc tính cần được xem xét tại mỗi bước
  + Tại mỗi bước, một thuộc tính tốt nhất sẽ được chọn ra dựa trên một tiêu chuẩn nào đó
  + Với mỗi thuộc tính được chọn, ta chia dữ liệu vào các nút con tương ứng với các giá trị của thuộc tính đó rồi tiếp tục áp dụng phương pháp này cho mỗi nút con
  + Việc chọn ra thuộc tính tốt nhất ở mỗi bước như thế này được gọi là cách chọn tham lam (greedy).
* Thuật toán GenDecTree(Mẫu S, thuộc tính A)

1. Tạo một nút N

2. Nếu tất cả các mẫu thuộc cùng lớp C thì N được gán nhãn C; dừng thuật toán;

3. Nếu A là rỗng thì N được gán nhãn C là nhãn phổ biến nhất trong S; dừng thuật toán;

4. Chọn a thuộc A, có độ đo **information gain** cao nhất; Gán nhãn N theo a;

5. Với mỗi giá trị v của a:

a. Phát triển 1 nhánh từ N với điều kiện a=v;

b. Đặt Sv là tập con của S với a=v;

c. Nếu Sv là rỗng thì gắn một lá có nhãn phổ biến nhất trong S;

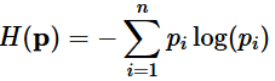
d. Ngược lại gắn một nút được tạo bởi GenDecTree(Sv, A-a)

* **Hàm số entropy**
  + Cho một phân phối xác suất của một biến rời rạc x có thể nhận n giá trị khác nhau 𝑥1, 𝑥2, … , 𝑥𝑛.
  + Giả sử rằng xác suất để x nhận các giá trị này l:

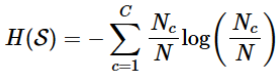


Ký hiệu phân phối này là 𝑝 = (𝑝1, 𝑝2, … , 𝑝𝑛).

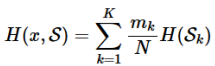
* + Entropy của phân phối này được định nghĩa là



* Xét một bài toán với C class khác nhau:
  + GS ta đang làm việc với một non-leaf node với các điểm dữ liệu tạo thành một tập 𝑆 với số phần tử là |𝑆| = 𝑁.
  + Giả sử rằng trong số 𝑁 điểm dữ liệu này, 𝑁𝑐, 𝑐 = 1,2, … , 𝐶 điểm thuộc vào class 𝑐. Xác suất để mỗi điểm dữ liệu rơi vào một class 𝑐 được xấp xỉ bằng 𝑁𝑐/𝑁 (maximum likelihood estimation).
  + entropy tại:



* + Giả sử thuộc tính được chọn là 𝑥. Dựa trên 𝑥, các điểm dữ liệu trong 𝑆 được phân ra thành 𝐾 child node 𝑆1, 𝑆2, … , 𝑆k với số điểm trong mỗi child node lần lượt là 𝑚1, 𝑚2, … , 𝑚𝐾. Ta định nghĩa



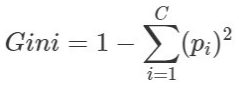
* + Ta định nghĩa information gain dựa trên thuộc tính 𝑥



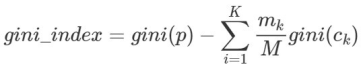
* + Tại mỗi node, thuộc tính được chọn được xác định dựa trên:



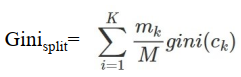
* Điều kiện dừng
  + nếu ta tiếp tục phân chia các node chưa tinh khiết, ta sẽ thu được một tree mà mọi điểm trong tập huấn luyện đều được dự đoán đúng.
  + Khi đó, tree có thể sẽ rất phức tạp (nhiều node) với nhiều leaf node chỉ có một vài điểm dữ liệu. Như vậy, nhiều khả năng overfitting sẽ xảy ra.
  + Để tránh overfitting, chúng ta phải có điều kiện dừng.
* Tại một node, nếu một trong số các điều kiện sau đây xảy ra, ta không tiếp tục phân chia node đó và coi nó là một leaf node:
  + Nếu node đó có entropy bằng 0, tức mọi điểm trong node đều thuộc một class.
  + Nếu node đó có số phần tử nhỏ hơn một ngưỡng nào đó. Trong trường hợp này, ta chấp nhận có một số điểm bị phân lớp sai để tránh overfitting. Class cho leaf node này có thể được xác định dựa trên class chiếm đa số trong node.
  + Nếu khoảng cách từ node đó đến root node đạt tới một giá trị nào đó. Việc hạn chế chiều sâu của tree này làm giảm độ phức tạp của tree và phần nào giúp tránh overfitting.
  + Nếu tổng số leaf node vượt quá một ngưỡng nào đó.
  + Nếu việc phân chia node đó không làm giảm entropy quá nhiều (information gain nhỏ hơn một ngưỡng nào đó)
* **Thuật toán CART (Classification and regression tree)**
* Decision Tree quyết định khi nào sẽ phân nhánh
  + Các quyết định phân nhánh sẽ ảnh hưởng đến độ chính xác của Cây.
  + Cây hồi quy và cây phân lớp có các thuật toán phân nhánh khác nhau.
  + Có nhiều thuật toán phân nhánh, tùy vào kiểu của biến mục tiêu mà sử dụng thuật toán như thế nào: Gini Index (CART)
    - Gini Index



* + - * Trong đó:
        + C: số lớp cần phân loại
        + pi=ni/N,
        + ni là số lượng phần tử ở lớp thứ i
        + N là tổng số lượng phần tử ở node đó



* + - * Trong đó
        + gini(p): chỉ số gini ở node cha
        + K: số node con được tách ra
        + gini(ck): chỉ số gini ở node con thứ k
        + M: số phần tử ở node p
        + mi: là số phần tử ở node con thứ i 
      * Chọn thuộc tính có hệ số Gini split nhỏ



* ***Các độ đo để đánh giá mô hình:***
  + Accuracy là chỉ số đo lường tỷ lệ phần trăm các mẫu dữ liệu được phân loại chính xác. Nó được tính bằng công thức sau:

Accuracy = số lượng dự đoán chính xác / tổng số mẫu dữ liệu

* + Precision là chỉ số đo lường tỷ lệ phần trăm các mẫu dữ liệu được phân loại là dương tính thực sự là dương tính. Nó được tính bằng công thức sau:

Precision = số lượng dự đoán dương tính thực sự / tổng số dự đoán dương tính

* + Recall là chỉ số đo lường tỷ lệ phần trăm các mẫu dữ liệu dương tính thực sự được phân loại là dương tính. Nó được tính bằng công thức sau:

Recall = số lượng dự đoán dương tính thực sự / tổng số mẫu dữ liệu dương tính thực sự

* + F1-score là chỉ số kết hợp độ chính xác và độ thu hồi. Nó được tính bằng công thức sau:

F1-score = 2 \* precision \* recall / (precision + recall)

***Phần 2. Thực nghiệm***

***1. Mô tả tập dữ liệu***

* Tên bài toán: Dự đoán ung thư vú
* **Số mẫu dữ liệu:** 569 mẫu
* **Thông tin của từng mẫu (Input) gồm:**
* 30 thuộc tính giá trị thực được tính toán cho mỗi nhân tế bào:
  + a) radius: Bán kính (trung bình của khoảng cách từ tâm đến các điểm trên chu vi)
  + b) texture: Độ nhám (độ lệch chuẩn của các giá trị thang độ xám)
  + c) perimeter: Chu vi
  + d) area: Diện tích
  + e) smoothness: Độ mịn (biến đổi cục bộ về độ dài bán kính)
  + f) compactness: Mật độ (chu vi^2 / diện tích - 1.0)
  + g) concavity: Độ lõm (mức độ nghiêm trọng của các phần lõm của đường viền)
  + h) concave points: Số lượng các phần lõm của đường viền
  + i) symmetry: Độ đối xứng
  + j) fractal dimension: Kích thước fractal
* **Tập nhãn (Output):**
* Diagnosis (M = lành tính, B = ác tính)



*1 mẫu nhỏ trong data*

***2. Mô tả cách giải bài toán bằng phương pháp học máy***

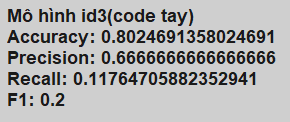
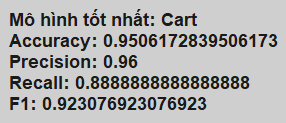
Phương pháp CROSS-VALIDATION

Bước 1: Chia toàn bộ tập dữ liệu thành k phần (phương pháp k-fold cross validation).

Bước 2: Chọn ngẫu nhiên k-1 phần làm training data, 1 phần còn lại làm test data. Sử dụng phương pháp ID3, CART tập training data và test data để xây dựng và đánh giá mô hình. Bước 2 này được làm k lần.

Bước 3: Chọn mô hình có (train error + validation error) là nhỏ nhất.

***3. Đánh giá mô hình***



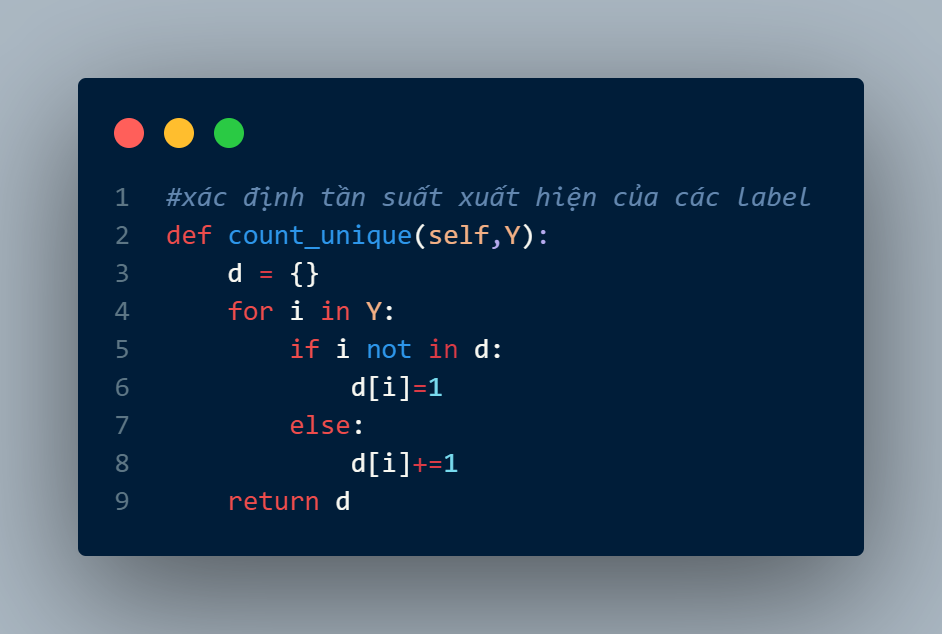
***4. Mô tả các chức năng của chương trình***

* Xây dựng xây dựng cây thuật toán ID3

A screen shot of a computer code

Description automatically generated

**Tạo node của cây**

****

**Xác định tần suất xuất hiện của labels**

A computer screen shot of a program code

Description automatically generated

**Tính error để tính tổng error của train và validation**

A computer screen shot of a computer code

Description automatically generated

**Tính entropy**

A computer screen shot of a program

Description automatically generated

**Tính information gain**

A screen shot of a computer screen

Description automatically generated

**Train mô hình**

****

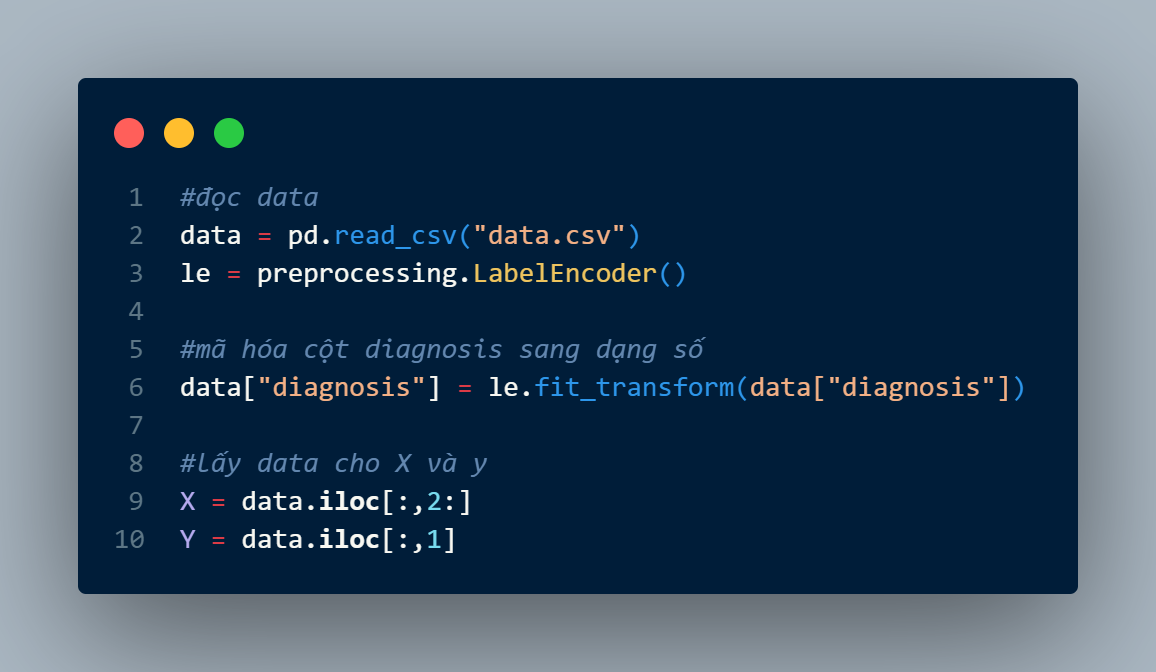
**Dự đoán nhãn**

A screen shot of a computer screen

Description automatically generated

**Xây dựng cây quyết định**

* Lấy và xử lý dữ liệu từ file data.csv



* Chia dữ liệu và xây dựng 2 mô hình id3 và cart

A screen shot of a computer program

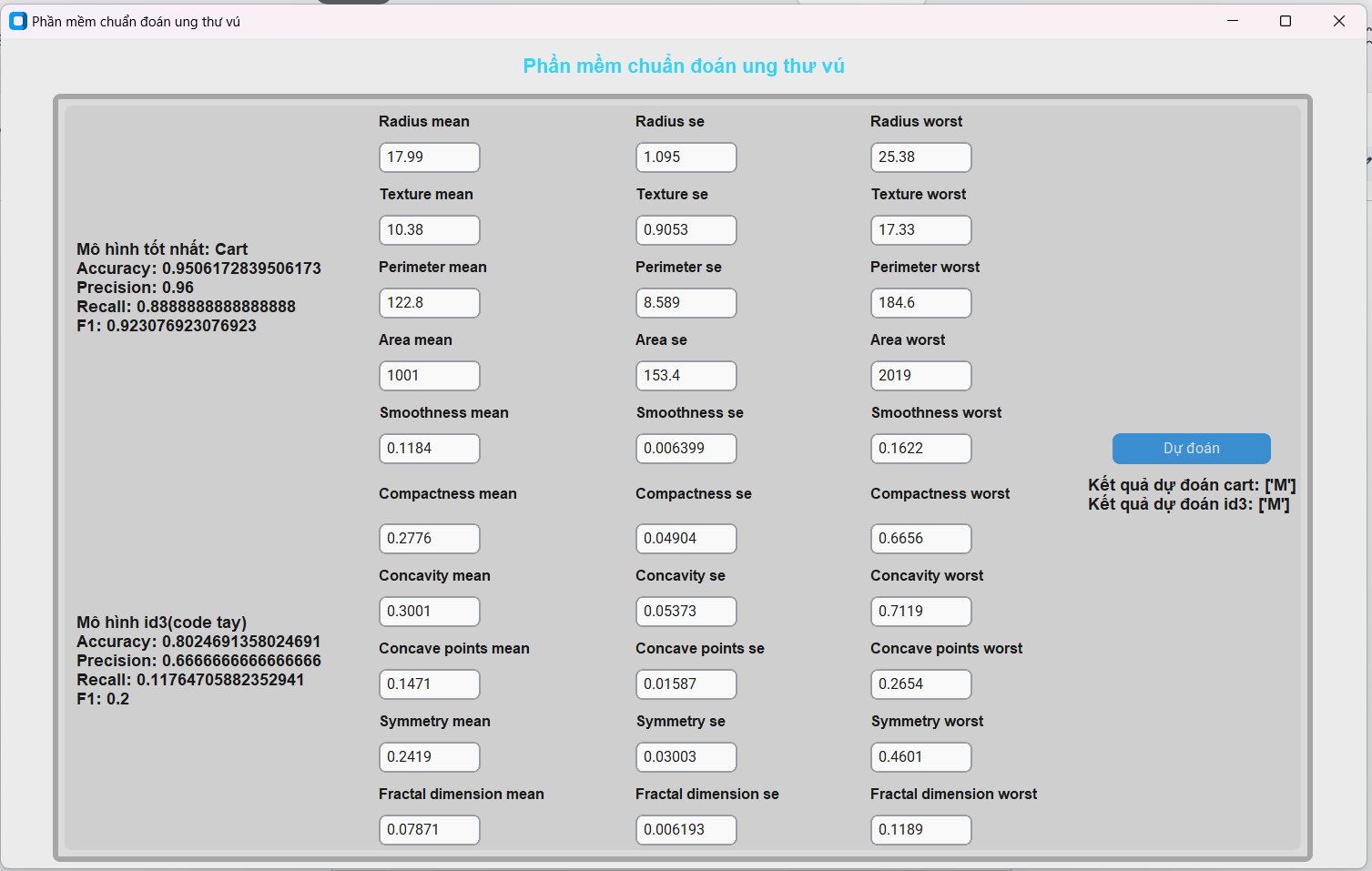
Description automatically generated

* Tìm fold có mô hình chất lượng nhất

A screenshot of a computer program

Description automatically generated

* Tạo giao diện để người dùng nhập dữ liệu đầu vào và chương trình sử dụng cây quyết định dự đoán kết quả ung thư vú



**Kết luận**

* + Giới thiệu về học máy như lịch sử ra đời của học máy và vai trò của học máy, ưu và nhược điểm của 2 phương pháp học có giám sát và học không giám sát.
  + Trình bày Mục đích phương pháp, Input, Output, Method của xây dựng cây quyết định ID3, CART và các độ đo để đánh giá chất lượng mô hình.
  + Mô tả về tập dữ liệu gồm những thông tin gì và sử dụng thông tin của những thuộc tính nào để dự đoán giá trị của thuộc tính nào.
  + Mô tả lại cách giải bài toán bằng việc sử dụng phương pháp k-fold cross validation để chọn ra mô hình huấn luyện và dùng ID3, CART để dự đoán dữ liệu mới.
  + Đánh giá chất lượng của 2 mô hình ID3 và CART qua các độ đo Accuracy, Precision, Recall, F1-score và mô tả các chức năng và giao diện của chương trình.

**Tài liệu tham khảo**

* Slide của giáo viên hướng dẫn Nguyễn Thị Kim Ngân
* Sách Machine Learning cơ bản của Vũ Hữu Tiệp
* Tiền xử lý dữ liệu: [sklearn.preprocessing.LabelEncoder](https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html)
* Thuật toán ID3, CART: [sklearn.tree.DecisionTreeClassifier](https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html)
* Link dataset: [kaggle](https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data/data)