

TRƯỜNG ĐẠI HỌC THỦY LỢI
KHOA CÔNG NGHỆ THÔNG TIN



BÀI TẬP LỚN

HỌC PHẦN: HỌC MÁY

ĐỀ TÀI: NHẬN BIẾT GIỚI TÍNH BẰNG GIỌNG NÓI

Giáo viên hướng dẫn:

Sinh viên thực hiện:

Lương Chung Hội, 2051060527, lớp 62THNB

Hà Nội, năm 2022

Phần 1: Tổng quan

1. Mô tả bài toán

Tên bài toán: Nhận biết giới tính bằng giọng nói bằng phương pháp CART và phân tích thành phần chính (PCA).

Input: Bộ dữ liệu bao gồm 3.168 mẫu giọng nói được ghi lại, được thu thập từ những người nói nam và nữ. Mỗi mẫu gồm các đặc tính âm thanh của mỗi giọng nói được đo và đưa vào file csv

- **meanfreq:** tần số trung bình (tính bằng kHz)
- **sd:** độ lệch chuẩn của tần số
- **median:** tần số trung bình (tính bằng kHz)
- **Q25:** lượng tử đầu tiên (tính bằng kHz)
- **Q75:** lượng tử thứ ba (tính bằng kHz)
- **IQR:** dải ký tự (tính bằng kHz)
- **skew:** độ lệch
- **kurt:** kurtosis (see note in specprop description)
- **sp.ent:** entropy quang phổ
- **sfm:** độ phẳng quang phổ
- **mode:** tần số chế độ
- **centroid:** tần số trung tâm (see specprop)
- **peakf:** tần số đỉnh (tần số có năng lượng cao nhất)
- **meanfun:** trung bình của tần số cơ bản được đo trên tín hiệu âm thanh
- **minfun:** tần số cơ bản tối thiểu được đo trên tín hiệu âm thanh
- **maxfun:** tần số cơ bản tối đa được đo trên tín hiệu âm thanh
- **meandom:** trung bình của tần số chi phối được đo trên tín hiệu âm thanh
- **mindom:** tần số tối thiểu được đo trên tín hiệu âm thanh
- **maxdom:** tối đa của tần số ưu thế được đo trên tín hiệu âm thanh
- **dfrange:** dải tần số chính được đo trên tín hiệu âm thanh
- **modindx:** chỉ số điều chế. Được tính bằng hiệu số tuyệt đối tích lũy giữa các phép đo lân cận của các tần số cơ bản chia cho dải tần số

Output: xác định một giọng nói là nam hay nữ

- **label:** male or female

Tóm tắt công việc thực hiện của bài toán.

Bước 1: Thu thập dữ liệu bài toán

Bước 2: Xác định tập dữ liệu của bài toán

Bước 3: Mô tả ma trận dữ liệu (X), Nhãn lớp (Y)

Bước 4: Chia bộ dữ liệu Train (70%), Test (30%) và chạy với các mô hình

Bước 5: Chọn mô hình có tỷ lệ chính xác tốt nhất và có tỷ lệ sai thấp nhất

2. Phương pháp học máy

2.1. Classification and regression tree (cây phân loại và hồi quy)

2.1.1 Khái niệm

CART hay Classification and regression tree (cây phân loại và hồi quy) là một thuật toán cây quyết định, được giới thiệu bởi Leo Breiman. Nó có thể giải quyết cả hai vấn đề phân loại và hồi quy.

2.1.2 Cách hoạt động của CART

CART thường sử dụng phương pháp Gini để tạo các điểm phân chia.

Gini impurity

Là phương pháp hướng đến đo lường tần suất một đối tượng dữ liệu ngẫu nhiên trong tập dữ liệu ban đầu được phân loại không chính xác, trên cơ sở đối tượng dữ liệu đã nằm trong một tập con được phân ra từ tập dữ liệu ban đầu, có dán nhãn thể hiện thuộc tính chung bất kỳ của các đối tượng còn lại trong tập con này, giá trị phân loại chính là nhãn của tập con.

Gini impurity chính là chỉ số đo lường mức độ đồng nhất hay nhiễu loạn của thông tin, hay sự khác biệt về các giá trị mà mỗi điểm dữ liệu trong một tập con, hoặc một nhánh của cây quyết định. Công thức Gini có thể dùng cho cả dữ liệu rời rạc và liên tục. Nếu điểm dữ liệu thuộc về một node và có chung thuộc tính bất kỳ thì node này thể hiện sự đồng nhất lúc này $gini=0$ và ngược lại gini sẽ lớn.

Công thức tổng quát của Gini:

$$G(p) = \sum_{i=1}^n p_i(1 - p_i) = 1 - \sum_{i=1}^n (p_i)^2$$

Công thức trên để tính độ vẩn đục của một node, khi có nhiều cách phân nhánh mỗi cách có thể phân ra một số node nhất định. Cho nên, lúc này có thêm công thức thứ 2 để tìm ra các phân chia tối ưu nhất:

$$G_{split} = \sum_{i=1}^k \frac{N_i}{N} G(i)$$

Trong đó:

- N_i là số điểm dữ liệu có trong node của nhánh được phân
- N là số điểm dữ liệu có trong node được dùng để phân nhánh
- Hệ số G_{split} càng nhỏ thì cách phân nhánh đó càng tối ưu.

2.1.3 Ưu điểm của CART

Cây quyết định có thể thực hiện phân loại đa lớp.

Cung cấp hầu hết khả năng diễn giải mô hình bởi vì chúng đơn giản như là một loạt các điều kiện if-else.

Có thể xử lý cả dữ liệu số và dữ liệu phân loại.

Mối quan hệ phi tuyến (Nonlinear relationships) giữa các tính năng không ảnh hưởng đến hiệu suất của Cây quyết định

2.1.4 Nhược điểm của CART

Nhược điểm lớn nhất của Cây quyết định là vấn đề Overfitting.

Một thay đổi nhỏ trong bộ dữ liệu có thể làm cho cấu trúc cây không ổn định có thể gây ra phương sai.

Cây quyết định có thể bị underfit nếu dữ liệu mất cân bằng. Do đó, nên cân bằng tập dữ liệu trước khi phù hợp với Cây quyết định.

2.2. PCA - Principal Components Analysis

2.2.1 Khái niệm.

PCA là phương pháp biến đổi giúp giảm số lượng lớn các biến có tương quan với nhau thành tập ít các biến sao cho các biến mới tạo ra là tổ hợp tuyến tính của những biến cũ không có tương quan lẫn nhau. Ví dụ, chúng ta có 100 biến ban đầu có tương quan tuyến tính với nhau, khi đó chúng ta sử dụng phương pháp PCA xoay chiều không gian cũ thành chiều không gian mới mà ở đó chỉ còn 5 biến không có tương quan tuyến tính mà vẫn giữ được nhiều nhất lượng thông tin từ nhóm biến ban đầu.

2.2.2 Đặc tính PCA.

Một số đặc tính của PCA được kể đến như:

- ❖ Giúp giảm số chiều dữ liệu - Giúp visualization khi dữ liệu có quá nhiều chiều thông tin.
- ❖ Do dữ liệu ban đầu có số chiều lớn (nhiều biến) thì PCA giúp chúng ta xoay trục tọa độ xây một trục tọa độ mới đảm bảo độ biến thiên của dữ liệu và giữ lại được nhiều thông tin nhất mà không ảnh hưởng tới chất lượng của các mô hình dự báo. (Maximize the variability).
- ❖ Do PCA giúp tạo 1 hệ trục tọa độ mới nên về mặt ý nghĩa toán học, PCA giúp chúng ta xây dựng những biến factor mới là tổ hợp tuyến tính của những biến ban đầu.
- ❖ Trong không gian mới, có thể giúp chúng ta khám phá thêm những thông tin quý giá mới khi mà tại chiều thông tin cũ những thông tin quý giá này bị che mất (Điền hình cho ví dụ về chú lạc đà phía trên).

2.2.3 Mô hình PCA.

Xét tập không gian (dữ liệu) k biến, k biến này được biểu qua j thành phần chính sao cho ($j < k$). Xét thành phần chính đầu tiên có dạng:

$$PC_1 = a_1X_1 + a_2X_3 + a_4X_5 + \dots a_kX_k$$

Thành phần chính đầu tiên chứa đựng hầu hết thông tin từ k biến ban đầu (được hình thành là 1 tổ hợp tuyến tính của các biến ban đầu) và lúc này tiếp tục xét thành phần chính thứ 2 được biểu diễn tuyến tính từ k biến ban đầu tuy nhiên thành phần chính thứ 2 phải không trực giao với thành phần chính ban đầu hay (thành phần chính thứ 2 không có mối tương quan tuyến tính với thành phần chính đầu tiên). Về lý thuyết chúng ta có thể xây dựng nhiều thành phần chính từ nhiều biến ban đầu. Tuy nhiên chúng ta cần tìm được trục không gian sao cho ít thành phần nhất mà có thể biểu diễn được hầu hết thông tin từ những biến ban đầu

2.2.4 Các bước thực hiện PCA

Tính vector kỳ vọng của toàn bộ dữ liệu:

$$\bar{\mathbf{x}} = \frac{1}{N} \sum_{n=1}^N \mathbf{x}_n$$

Trừ mỗi điểm dữ liệu đi vector kỳ vọng của toàn bộ dữ liệu:

$$\hat{\mathbf{x}}_n = \mathbf{x}_n - \bar{\mathbf{x}}$$

Tính ma trận hiệp phương sai:

$$\mathbf{S} = \frac{1}{N} \hat{\mathbf{X}} \hat{\mathbf{X}}^T$$

Tính các trị riêng và vector riêng có norm bằng 1 của ma trận này, sắp xếp chúng theo thứ tự giảm dần của trị riêng.

Chọn K vector riêng ứng với K trị riêng lớn nhất để xây dựng ma trận \mathbf{U}_K có các cột tạo thành một hệ trực giao. K vector này, còn được gọi là các thành phần chính, tạo thành một không gian con gần với phân bố của dữ liệu ban đầu đã chuẩn hoá.

Chiếu dữ liệu ban đầu đã chuẩn hoá $\hat{\mathbf{X}}$ xuống không gian con tìm được

Dữ liệu mới chính là toạ độ của các điểm dữ liệu trên không gian mới

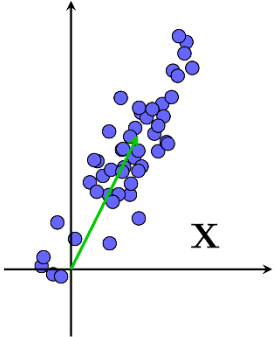
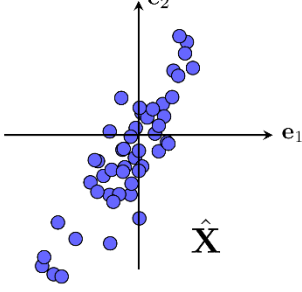
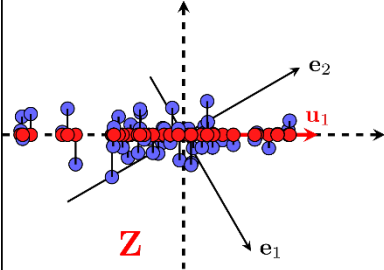
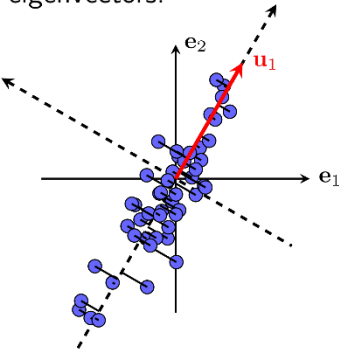
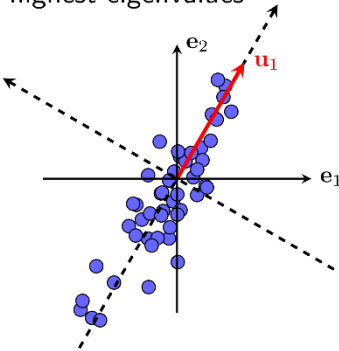
$$\mathbf{Z} = \mathbf{U}_K^T \hat{\mathbf{X}}$$

Dữ liệu ban đầu có thể tính được xấp xỉ theo dữ liệu mới như sau:

$$\mathbf{x} \approx \mathbf{U}_K \mathbf{Z} + \bar{\mathbf{x}}$$

Các bước thực hiện PCA có thể được xem trong Hình dưới đây:

PCA procedure

<p>1. Find mean vector</p> 	<p>2. Subtract mean</p> 	<p>3. Compute covariance matrix: $\mathbf{S} = \frac{1}{N} \hat{\mathbf{X}} \hat{\mathbf{X}}^T$</p> <p>4. Computer eigenvalues and eigenvectors of \mathbf{S}: $(\lambda_1, \mathbf{u}_1), \dots, (\lambda_D, \mathbf{u}_D)$ Remember the orthonormality of \mathbf{u}_i.</p>
<p>7. Obtain projected points in low dimension.</p> 	<p>6. Project data to selected eigenvectors.</p> 	<p>5. Pick K eigenvectors w. highest eigenvalues</p> 

Phần 2: Thực nghiệm

1. Mô tả tập dữ liệu của bài toán

```
RangeIndex: 3168 entries, 0 to 3167
Data columns (total 21 columns):
#   Column      Non-Null Count  Dtype
---  -
0   meanfreq    3168 non-null   float64
1   sd          3168 non-null   float64
2   median      3168 non-null   float64
3   Q25         3168 non-null   float64
4   Q75         3168 non-null   float64
5   IQR         3168 non-null   float64
6   skew        3168 non-null   float64
7   kurt        3168 non-null   float64
8   sp.ent      3168 non-null   float64
9   sfm         3168 non-null   float64
10  mode        3168 non-null   float64
11  centroid    3168 non-null   float64
12  meanfun     3168 non-null   float64
13  minfun      3168 non-null   float64
14  maxfun      3168 non-null   float64
15  meandom     3168 non-null   float64
16  mindom      3168 non-null   float64
17  maxdom      3168 non-null   float64
18  dfrange     3168 non-null   float64
19  modindx     3168 non-null   float64
20  label       3168 non-null   object
dtypes: float64(20), object(1)
```

- **meanfreq**: tần số trung bình (tính bằng kHz)
- **sd**: độ lệch chuẩn của tần số
- **median**: tần số trung bình (tính bằng kHz)
- **Q25**: lượng tử đầu tiên (tính bằng kHz)
- **Q75**: lượng tử thứ ba (tính bằng kHz)
- **IQR**: dải ký tự (tính bằng kHz)
- **skew**: độ lệch
- **kurt**: kurtosis (see note in specprop description)
- **sp.ent**: entropy quang phổ
- **sfm**: độ phẳng quang phổ
- **mode**: tần số chế độ
- **centroid**: tần số trung tâm (see specprop)
- **peakf**: tần số đỉnh (tần số có năng lượng cao nhất)
- **meanfun**: trung bình của tần số cơ bản được đo trên tín hiệu âm thanh
- **minfun**: tần số cơ bản tối thiểu được đo trên tín hiệu âm thanh
- **maxfun**: tần số cơ bản tối đa được đo trên tín hiệu âm thanh
- **meandom**: trung bình của tần số chi phối được đo trên tín hiệu âm thanh
- **mindom**: tần số tối thiểu được đo trên tín hiệu âm thanh

- **maxdom:** tối đa của tần số ưu thế được đo trên tín hiệu âm thanh
- **dfrange:** dải tần số chính được đo trên tín hiệu âm thanh
- **modindx:** chỉ số điều chế. Được tính bằng hiệu số tuyệt đối tích lũy giữa các phép đo lân cận của các tần số cơ bản chia cho dải tần số có 3.168 mẫu giọng nói.

Nhãn của dữ liệu: **label:** male or female

Mô tả ma trận dữ liệu (X): 20 cột và 3168 hàng

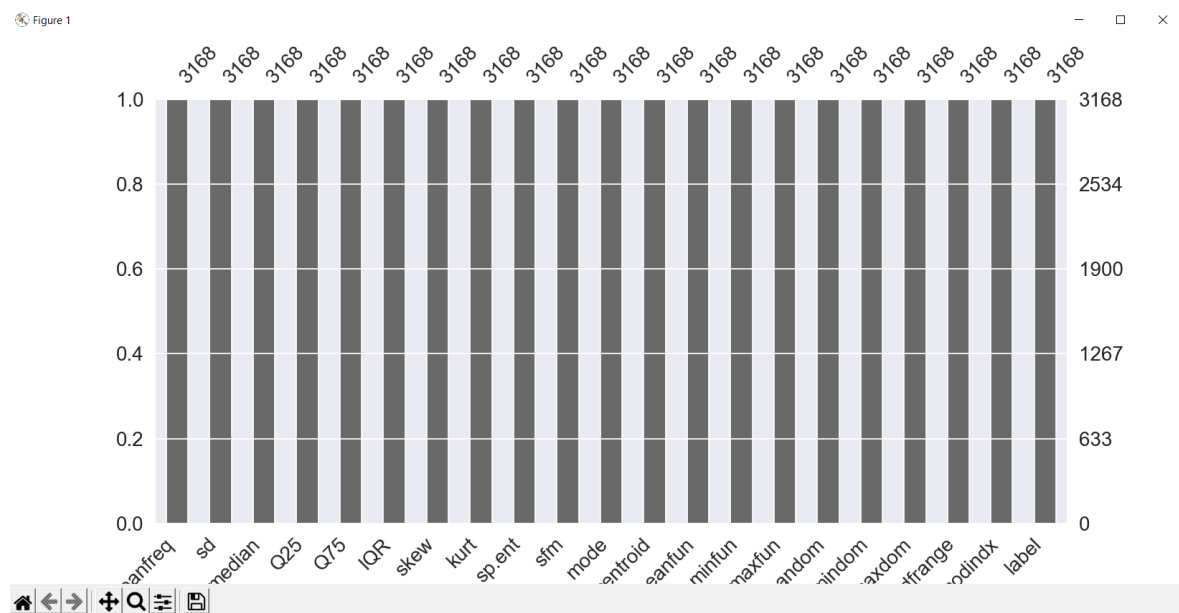
Mô tả ma trận nhãn lớp (Y): 1 cột và 3168 hàng

Chia tập dữ liệu thành 2 phần:

- 70% dùng để huấn luyện mô hình: 2117 vector
- 30% dùng để kiểm tra sự phù hợp của mô hình: 951 vector.

2. Phân tích kết quả của chương trình

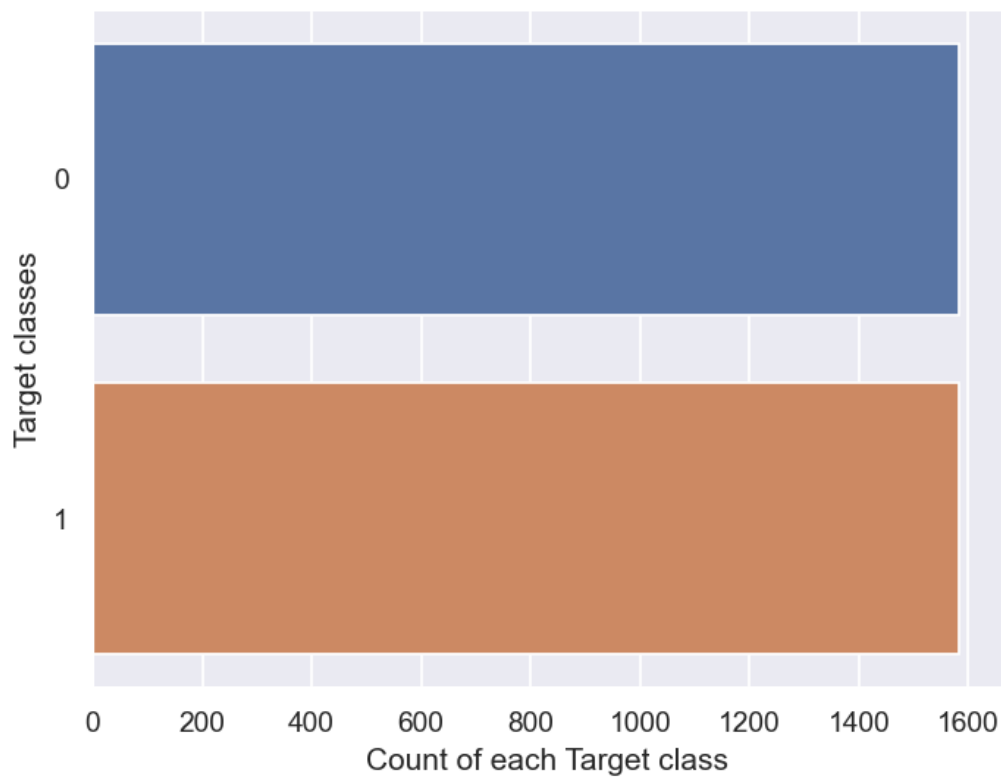
Kiểm tra tập dữ liệu có bị thiếu giá trị hay không



⇒ Dữ liệu không bị thiếu.

Kiểm tra tập dữ liệu nhãn:

Figure 1



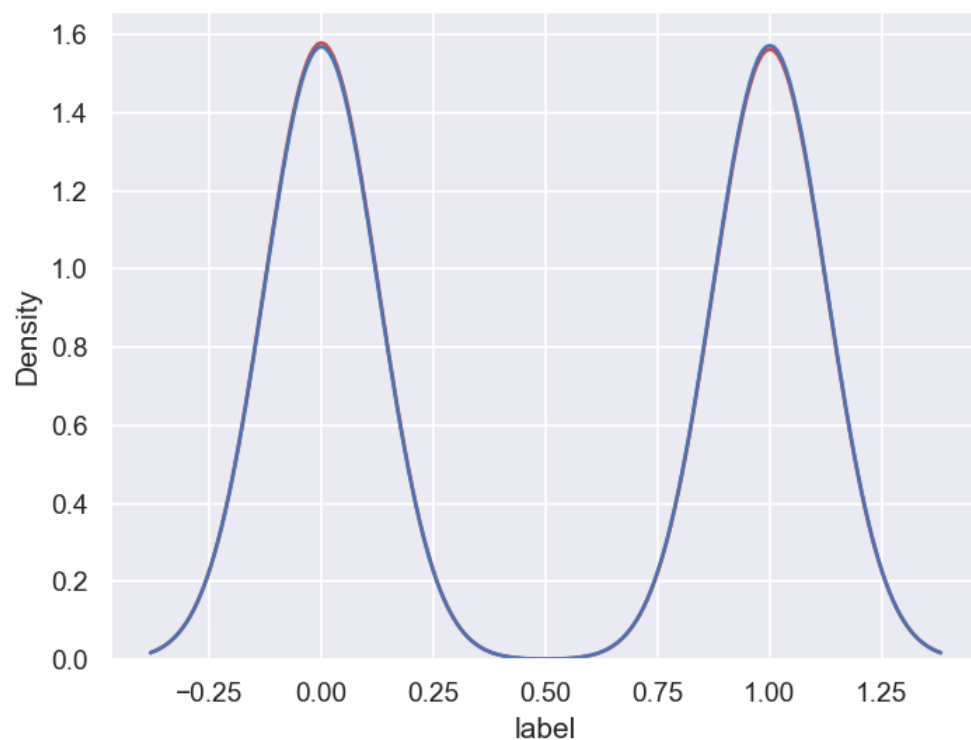
Nhãn 0 có 1584 dữ liệu

Nhãn 1 có 1584 dữ liệu

2.1 Kết quả phương pháp học máy CART với tập dữ liệu gốc.

```
Accuracy_CART: 0.9579390115667719
MAE_CART: 0.04206098843322818
MSE_CART: 0.04206098843322818
RMSE_CART: 0.20508775788239575
confusion_matrix:
[[453  25]
 [ 15 458]]
Precision: 0.958094441790094
Recall: 0.9682875264270613
F1_score: 0.9581589958158996
```

Figure 1



x=1.434 y=1.106

- Tỷ lệ dự đoán đúng: 95.8%
- Tỷ lệ dự đoán sai: 4.2%
- Phân tích kết quả: Có 453 nam dự đoán đúng, 25 dự đoán sai
Có 458 nữ dự đoán đúng, 15 nữ dự đoán sai
- Precision: 95.8%
- Recall: 96%
- Trung bình điều hòa (F1-score): 95.8%

2.2 Kết quả phương pháp học máy CART với phân tích thành phần chính PCA.

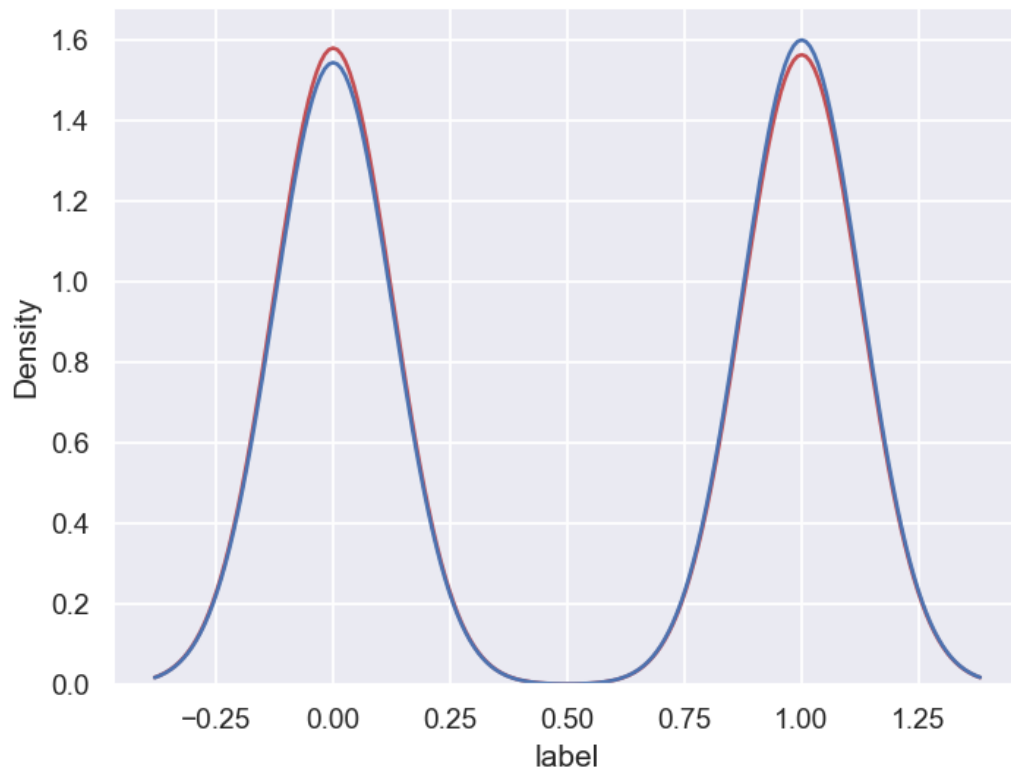
2.2.1 Giảm chiều dữ liệu bằng phân tích thành phần chính

```
(3168, 13)
[[ 2.38081294e+02 -4.80224677e+00 -2.24803910e+00 ... -2.59390854e-03
  -6.64500448e-02  5.48516462e-03]
 [ 5.98396593e+02 -1.02137403e+00 -8.81787888e-01 ... -2.27223557e-02
  -5.09705775e-02  2.50928162e-03]
 [ 9.88762667e+02  3.02260542e+00  2.62021449e+00 ... -7.56012824e-03
  -4.91887292e-02  4.02360371e-04]
 ...
 [-2.99567425e+01 -3.26518493e+00  4.68605195e-01 ...  1.06142212e-01
   6.79281143e-02 -1.39756821e-02]
 [-3.11908741e+01 -2.32061968e+00  6.91941205e-01 ...  6.00527589e-02
   4.15054549e-02 -2.21735068e-02]
 [-3.07626982e+01 -6.64483239e+00  7.32112467e-01 ...  9.15270941e-02
   2.59931595e-02  2.58594989e-03]]
```

2.2.2 Kết quả chương trình chạy phương pháp học máy CART với bộ dữ liệu đã được phân tích

```
Accuracy_CART_PCA: 0.9221871713985279
MAE_CART_PCA: 0.07781282860147214
MSE_CART_PCA: 0.07781282860147214
RMSE_CART_PCA: 0.27894950905400806
confusion_matrix:
[[437  41]
 [ 33 440]]
Precision_new: 0.9147609147609148
Recall_new: 0.9302325581395349
F1_score_new: 0.9224318658280923
```

Figure 1



- Tỷ lệ dự đoán đúng: 92%

- Tỷ lệ dự đoán sai: 8%
- Phân tích kết quả: Có 437 nam dự đoán đúng, 41 dự đoán sai
Có 440 nữ dự đoán đúng, 33 nữ dự đoán sai
- Precision: 91.4%
- Recall: 93%
- Trung bình điều hòa (F1-score): 92.2%

Kết luận

Qua các kiến thức được học về Machine Learning và tìm hiểu thêm em đã thực hiện thành công bài toán “ : Nhận biết giới tính bằng giọng nói. ” với mô hình bằng phương pháp CART và phân tích thành phần chính (PCA)

Trong quá trình hoàn thành bài tập lớn, em đã tìm hiểu và tham khảo các tài liệu liên quan. Đạt được một kết quả tương đối tốt với 96% tập kiểm tra với mô hình mạng “CART với tập dữ liệu gốc” đây là mô hình đạt các chỉ số đánh giá (độ chính xác, f1-score, AUC) tương đối cao. Kết quả của mô hình dự đoán CART khi kết hợp với dữ liệu đã được rút gọn bằng PCA cho kết quả đúng 92%. Chúng em nhận thấy với bài toán này mô hình dự đoán bằng phương pháp CART với dữ liệu gốc cho kết quả tốt hơn và khá là phù hợp với bài toán.

Tài liệu tham khảo