**Chapter 2: Representation (12 câu)**

**Outline**

[**[2.1 → 2.3: Long. 2.4 → 2.6: Thành] 2**](#_tee5xhdfloji)

[2.1 Question: 2](#_2tphertd6v2t)

[2.1 Answer: 2](#_u29jarl1pfm1)

[2.2 Question 3](#_3pqzy7qz48p6)

[2.2 Answer 3](#_fifpodem329b)

[2.3. Question 3](#_kjhx6kkjrrla)

[2.3. Answer 4](#_3idm62ifvhtl)

[2.4 Question 4](#_20awxxymn1pf)

[2.4 Answer 4](#_izp128zdfj0n)

[2.5 Question 5](#_k57enlbpczu5)

[2.5 Answer 5](#_4tm1zku2mot9)

[2.6 Question 5](#_ur5428ssacp)

[2.6 Answer 5](#_70unpbkuvrqc)

[Answer 1: 5](#_jiqswv64vdqc)

[Answer 2: 6](#_rtbgtmb4e2uz)

[Answer 3: [Trâm] 7](#_zhcrcamzj60z)

[**Tâm + Huân 7**](#_m1zx1gf43vkx)

[2.7 Question 7](#_8zzd61pbo3e6)

[2.7 Answer 7](#_5f4aa4v1mumo)

[2.8 Question 8](#_v6ke2877re7d)

[2.8 Answer 9](#_do1jphhmss4k)

[2.9 Question 10](#_3kpa3qap8grf)

[2.9 Answer 10](#_9v4za83v5n6e)

[2.10 Question 11](#_z2h2tu34b2q9)

[2.10 Answer 11](#_aetpxkmicc1w)

[2.11 Question 12](#_bmjaxue7m7qj)

[2.11 Answer 13](#_idib6s40qy45)

[**Toàn 14**](#_2p3zx2sy97zn)

[2.7 Question 14](#_rfqrrh6evypa)

[2.7 Answer 14](#_31zo8yvnmv3j)

[2.8 Question 15](#_k9prwbnx8j4z)

[2.8 Answer 15](#_w757m5ev090t)

[2.9 Question 17](#_rs71n72yupnd)

[2.9 Answer 17](#_iknx60nb97ar)

[2.10 Question 17](#_c5c6not65lbz)

[2.10 Answer 17](#_qujpbjdcf9sl)

[2.11 Question 18](#_i0tat4qpy2h7)

[2.11 Answer 18](#_1kimlssiit55)

[2.12 Question 19](#_azhk1u7e05sn)

[2.12 Answer 19](#_q6t7ad5s0d7h)

### **[2.1 → 2.3: Long. 2.4 → 2.6: Thành]**

#### **2.1 Question:**

Find the centroid and medoid (most centrally located pattern) for the following set of patterns:

*Tìm trọng tâm và medoid (mẫu nằm ở vị trí trung tâm nhất) cho tập hợp các mẫu sau:*

*(1, 1), (1, 3), (1, 4), (2, 2), (2, 3), (3,1), (3,4), (4,2)*

#### **2.1 Answer:**

Centroid được tính bằng cách tính trung bình tất cả các chiều của các điểm dữ liệu. Đặt tọa độ của centroid cần tìm là , trong đó:

Vậy centroid của tập dữ liệu là

Để tìm metroid của tập dữ liệu, ta cần tìm điểm có tổng khoảng cách tính từ nó đến các điểm dữ liệu khác là nhỏ nhất. Ta có thể dùng công thức Euclid hoặc Manhattan để tính khoảng cách giữa 2 điểm. Để đơn giản, ta sử dụng công thức Manhattan để tính khoảng cách:

Ví dụ, khoảng cách Manhattan giữa 2 điểm và là .

Ta lập bảng sau để tính theo công thức khoảng cách Manhattan.

|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | Total Cost |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 0 | 2 | 3 | 2 | 3 | 2 | 5 | 4 | 21 |
| 2 | 2 | 0 | 1 | 2 | 1 | 4 | 3 | 4 | 17 |
| 3 | 3 | 1 | 0 | 3 | 2 | 5 | 2 | 5 | 21 |
| 4 | 2 | 2 | 3 | 0 | 1 | 2 | 3 | 2 | 15 |
| 5 | 3 | 1 | 2 | 1 | 0 | 3 | 2 | 3 | 15 |
| 6 | 2 | 4 | 5 | 2 | 3 | 0 | 3 | 2 | 21 |
| 7 | 5 | 3 | 2 | 3 | 2 | 3 | 0 | 3 | 21 |
| 8 | 4 | 4 | 5 | 2 | 3 | 2 | 3 | 0 | 23 |

Như vậy ta có thể chọn điểm 4 hoặc 5 2, 3) làm metroid do hai điểm này có tổng khoảng cách từ nó tới các điểm khác ( là nhỏ nhất.

\* Nếu áp dụng công thức Euclid để tính khoảng cách: , ta có bảng sau:

|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | Total Cost |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | - | 2.00 | 3.00 | 1.41 | 2.24 | 2.00 | 3.61 | 3.16 | 17.42 |
| 2 | 2.00 | - | 1.00 | 1.41 | 1.00 | 2.83 | 2.24 | 3.16 | 13.64 |
| 3 | 3.00 | 1.00 | - | 2.24 | 1.41 | 3.61 | 2.00 | 3.61 | 16.86 |
| 4 | 1.41 | 1.41 | 2.24 | - | 1.00 | 1.41 | 2.24 | 2.00 | 11.71 |
| 5 | 2.24 | 1.00 | 1.41 | 1.00 | - | 2.24 | 1.41 | 2.24 | 11.54 |
| 6 | 2.00 | 2.83 | 3.61 | 1.41 | 2.24 | - | 3.00 | 1.41 | 16.50 |
| 7 | 3.61 | 2.24 | 2.00 | 2.24 | 1.41 | 3.00 | - | 2.24 | 16.73 |
| 8 | 3.16 | 3.16 | 3.61 | 2.00 | 2.24 | 1.41 | 2.24 | - | 17.82 |

Ở đây ta chọn metroid là điểm 5 do tổng khoảng cách từ điểm này tới các điểm khác trong tập dữ liệu là nhỏ nhất.

#### **2.2 Question**

Find the edit distance between the two strings and .

*Tìm khoảng cách chỉnh sửa giữa hai chuỗi “HOUSE” và “MOUND”.*

#### **2.2 Answer**

Khoảng cách edit distance (Levenshtein distance) là số bước thao tác nhỏ nhất để biến đổi chuỗi ký tự cho trước sang chuỗi ký tự mục tiêu. Các thao tác biến đổi được cho phép bao gồm: insertion, replacement hoặc deletion. Để biến đổi chuỗi về ta đi qua các thao tác sau:

Như vậy khoảng cách

#### **2.3. Question**

Given the following distance measures. Let state which one is metric and which is non-metric.

Euclidean distance, edit-distance, Manhattan distance, Dynamic Time Warping distance.

*Cho các số đo khoảng cách sau. Hãy nêu rõ cái nào là số liệu và cái nào không phải số liệu.*

*Khoảng cách Euclide (***Euclidean distance**)*, khoảng cách chỉnh sửa(***edit-distance**)*, khoảng cách Manhattan (***Manhattan distance**)*, khoảng cách Cong vênh thời gian động (***Dynamic Time Warping distance**).

#### **2.3. Answer**

Một metric distance cần thỏa mãn các tính chất sau:

* Positive reflexivity:
* Symetric:
* Triangular inequality:

Nếu không thỏa mãn dù chỉ 1 trong 3 tính chất trên thì độ đo được gọi là non-metric distance. Như vậy:

* Metric distance: Euclidean distance, Manhattan distance
* Non-metric distance: Edit distance, Dynamic Time Wraping distance

#### **2.4 Question**

A dataset consists of the following patterns:

(1, 1, 1), (2, 2, 1), (1.5, 0.5, 1), (1, 3, 1), (4, 4, 2), (5, 5, 2), (4, 5, 2), (4, 6, 2)

Where each pattern consists of the x coordinate, the y coordinate and the class label. Find the direction of the w vector associated with Fisher’s Linear Discriminant.

*Một bộ dữ liệu bao gồm các mẫu sau:*

*(1, 1, 1), (2, 2, 1), (1,5, 0,5, 1), (1, 3, 1), (4, 4, 2), (5, 5, 2), (4 , 5, 2), (4, 6, 2)*

*Trong đó mỗi mẫu bao gồm tọa độ x, tọa độ y và nhãn lớp. Tìm hướng của vectơ w được liên kết với Phân biệt tuyến tính của Fisher.*

#### **2.4 Answer**

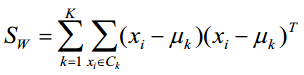
**Anwser 1: [Thành]**

Tính mean lớp 1 và 2:

**μ1** = ((1 + 2 + 1.5 + 1) / 4, (1 + 2 + 0.5 + 3) /4) = (1.375, 1.625)

**μ2** = ((4 + 5 + 4 + 4) /4, (4 +5+5+6) /4) = (4.25 ,5)

Công thức tính **SW** (Ghi chú: lấy tọa độ của 1 điểm nằm trong 1 class trừ đi tọa độ của mean của class đó và sau đó nhân cho ma trận chuyển vị (transpose) tương ứng, và lấy tổng tất cả các điểm của các class)



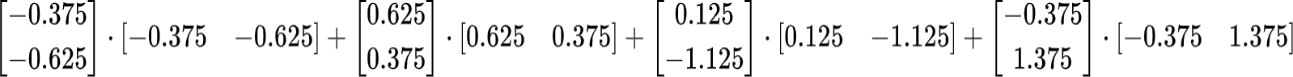
Lấy ví dụ cho điểm x1 = (1,1) có mean là μ1 = (1.375, 1.625)

x1 - μ1 = (1, 1) - (1.375, 1.625) = (-0.375, -0.625)

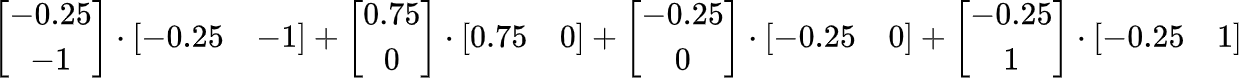
= (-0.375, -0.625) \* (-0.375, -0.625)T = ((0.140625, 0.234375), (0.234375, 0.390625))Tính toán tương tự cho các điểm còn lại.

**Sw = S1 + S2**

**=**



+



= ((0.83, 0.42), (0.42, 2.83)) + ((0.75, 0), (-0.25, 2))

= ((1.58, 0.42), (0.17, 5.83))

=> **SW-1** = ((4.02, -0.7), (-0.7, 0.21)) (Inverse matrix of **SW**)

**W = SW-1 (**μ**1 -** μ**2)**

= ((4.02, -0.7), (-0.7, 0.21)) \* (-2.875, -3.375) = (-9.195, 1.30375)

**Anwser 2: [Thanh Nhan Nguyen]**

Đáp án: **u1 - u2 = (-2.875, -3.375)**

#### **2.5 Question**

If there are 10 features and it is necessary to reduce the number of features to 6 so that the best set of six features is chosen, what is the number of feature sub-sets to be evaluated to find the optimal set of six features in the exhaustive search?

*Nếu có 10 thuộc tính và cần phải giảm số thuộc tính xuống còn 6 để chọn được bộ sáu thuộc tính tốt nhất, thì số lượng tập con thuộc tính cần đánh giá để tìm ra bộ sáu thuộc tính tối ưu trong tìm kiếm đầy đủ?*

#### **2.5 Answer**

C(m, d) = d!/[m!(d – m)!]

C(6, 10) = 10!/(6!\*4!)=210

#### **2.6 Question**

Give some differences between two feature extraction methods LDA (Fisher Linear Discriminant) and PCA.

*Nêu một số khác biệt giữa hai phương pháp trích chọn đặc trưng LDA (Fisher Linear Discriminant) và PCA.*

#### **2.6 Answer**

##### **Answer 1:**

LDA (Fisher Linear Discriminant) and PCA are both feature abstraction methods used in machine learning and data analysis. However, they differ in their goals, assumptions, and methods. Here are some key differences between LDA and PCA:

**Goal:**

* LDA: LDA is a supervised dimensionality reduction method, which means that it seeks to find a projection of the data that maximizes the separation between classes, while minimizing the variance within each class. The goal of LDA is to find the best features to separate the classes.
* PCA: PCA is an unsupervised dimensionality reduction method, which means that it seeks to find a projection of the data that captures the maximum amount of variance in the data. The goal of PCA is to find the most important features that capture the variability of the data.

**Assumptions:**

* LDA: LDA assumes that the data is normally distributed and that the covariance matrix is the same for all classes. LDA assumes that the classes have equal covariance matrices, and the feature values are normally distributed.
* PCA: PCA assumes that the data is linearly related, and there are no assumptions regarding the distribution of the data.

**Method:**

* LDA: LDA calculates the between-class and within-class scatter matrices to find the projection that maximizes the separation between classes while minimizing the variance within each class. It tries to find a linear combination of features that best separates the classes in the data.
* PCA: PCA calculates the covariance matrix and finds the eigenvectors of the covariance matrix. It then projects the data onto the eigenvectors with the highest eigenvalues. It tries to find the most important directions of variance in the data.

**Applicability:**

* LDA: LDA is generally used for classification problems where the goal is to separate data into different classes. It is often used in face recognition or biometric authentication systems.
* PCA: PCA is generally used for data visualization, data compression, or noise reduction. It is often used in image compression, data compression, and data visualization.

**In summary**:

* LDA and PCA are both feature abstraction methods used to reduce the dimensionality of data. However, LDA is a supervised method that tries to find a projection that maximizes the separation between classes, while PCA is an unsupervised method that tries to find the most important directions of variance in the data.

##### **Answer 2:**

LDA and PCA are both linear transformation techniques for dimensionality reduction. However, they have some key differences12:

* LDA is supervised while PCA is unsupervised. LDA uses class labels while PCA ignores them.
* LDA maximizes the separability between classes while PCA maximizes the variance in the data set.
* LDA forms a new set of components that are linear discriminants while PCA forms a new set of components that are principal components.

1: What is the difference between LDA and PCA for dimensionality reduction? (sebastianraschka.com) <https://sebastianraschka.com/faq/docs/lda-vs-pca.html>

2: LDA vs. PCA – Towards AI <https://towardsai.net/p/data-science/lda-vs-pca>

##### **Answer 3: [Trâm]**

* **Giống:** Cả LDA (Fisher Linear Discriminant) và PCA đều là các phương pháp trích chọn đặc trưng phổ biến được sử dụng trong lĩnh vực xử lý tín hiệu và học máy.
* **Khác:** được so sánh cụ thể ở bảng bên dưới

| **STT** | **Tiêu chí so sánh** | **LDA (Fisher Linear Discriminant)** | **PCA** |
| --- | --- | --- | --- |
| 1 | Mục đích | LDA được thiết kế để giữ lại các đặc trưng tốt nhất cho việc phân loại | PCA giữ lại các đặc trưng tốt nhất cho việc biểu diễn dữ liệu. |
| 2 | Tiêu chuẩn lựa chọn đặc trưng | LDA sử dụng phương pháp tối đa hoá độ tách biệt giữa các lớp | PCA sử dụng phương pháp tối thiểu hoá tổng bình phương sai số giữa các điểm dữ liệu và phân tách đường trung bình. |
| 3 | Kiểu dữ liệu | LDA được sử dụng cho các bài toán phân loại hai lớp hoặc nhiều lớp | PCA thường được sử dụng cho các bài toán giảm chiều dữ liệu trong không gian Euclid. |
| 4 | Đối tượng | LDA sử dụng thông tin lớp để trích chọn đặc trưng | PCA không sử dụng thông tin lớp. |
| 5 | Kết quả | LDA cho ra các đặc trưng mới mà có thể được sử dụng để phân loại các điểm dữ liệu mới | PCA chỉ cho ra các trục mới trong không gian dữ liệu ban đầu. |
| 6 | Số lượng đặc trưng | LDA cho ra số lượng đặc trưng ít hơn hoặc bằng số lượng lớp | PCA cho ra số lượng đặc trưng tối đa bằng số lượng chiều dữ liệu ban đầu. |

### **Tâm + Huân**

#### **2.7 Question**

Given a data set ***X*** consisting of 10 patterns (each pattern has 3 features) as follows: (7, 4, 3), (4, 1, 8), (6, 3, 5), (8, 6, 1), (8, 5, 7), (7, 2, 9), (8, 2, 2), (7, 4, 5), (9, 5, 8) và (5, 3, 3).

Compute covariance matrix of the dataset ***X***

*Cho tập dữ liệu X gồm 10 mẫu (mỗi mẫu có 3 đặc trưng) như sau: (7, 4, 3), (4, 1, 8), (6, 3, 5), (8, 6, 1) , (8, 5, 7), (7, 2, 9), (8, 2, 2), (7, 4, 5), (9, 5, 8) và (5, 3, 3).*

*Tính ma trận hiệp phương sai của tập dữ liệu X*

#### **2.7 Answer**

The covariance matrix:

mean\_x1 = (7+4+6+8+8+7+8+7+9+5)/10 = 6.9

mean\_x2 = (4+1+3+6+5+2+2+4+5+3)/10 = 3.5

mean\_x3 = (3+8+5+1+7+9+2+5+8+3)/10 = 5.2

The mean of each feature is (6.9, 3.5, 5.2).

D = X\_centered = X - mean

D =

[[ 0.1, 0.5, -2.2],

[-2.9, -2.5, 2.8],

[-0.9, -0.5, -0.2],

[ 1.1, 2.5, -4.2],

[ 1.1, 1.5, 1.8],

[ 0.1, -1.5, 3.8],

[ 1.1, -1.5, -3.2],

[ 0.1, 0.5, -0.2],

[ 2.1, 1.5, 2.8],

[-1.9, -0.5, -2.2]]

**cov = 1/n.DT.D**

covariance matrix is:

C =1/9 **DT.D =**

#### **2.8 Question**

We need to classify a person as male or female, based on the three features: height, weight and foot length. The training set is given as follows (assume that the data set satisfies Gaussian distribution):Given a test pattern with height = 6, weight = 130 and foot length = 8. Use covariance matrix and Mahalanobis distance to classify the test pattern.

*Chúng ta cần phân loại một người là nam hay nữ dựa trên ba đặc điểm: chiều cao, cân nặng và chiều dài bàn chân. Tập huấn luyện được đưa ra như sau (giả sử rằng tập dữ liệu thỏa mãn phân phối Gaussian): Cho một mẫu thử nghiệm có chiều cao = 6, cân nặng = 130 và chiều dài bàn chân = 8. Sử dụng ma trận hiệp phương sai và khoảng cách Mahalanobis để phân loại mẫu thử nghiệm.*

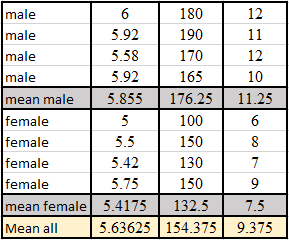
| Gender | Height (feet) | Weight(lbs) | Foot length (inch) |
| --- | --- | --- | --- |
| M  M  M  M  F  F  F  F | 6  5.92  5.58  5.92  5  5.5  5.42  5.75 | 180  190  170  165  100  150  130  150 | 12  11  12  10  6  8  7  9 |

#### **2.8 Answer**

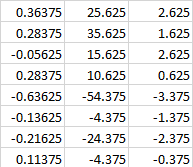
First we find the mean for each group of male and female

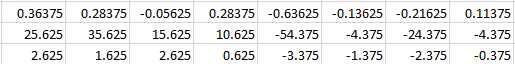
*Đầu tiên chúng tôi tìm giá trị trung bình cho từng nhóm nam và nữ*

#### 



Tương tự như câu 1 D = X - mean all

D = 

DT = 

**cov(X,Y) = 1/n.DT.D = 1/7 .DT.D**

C = [[ 0.11136964, 8.79017857, 0.60732143],

[ 8.79017857, 838.83928571, 60.26785714],

[ 0.60732143, 60.26785714, 5.125 ]]

C-1= [[ 53.9974745, -0.684057525, 1.64543567 ],

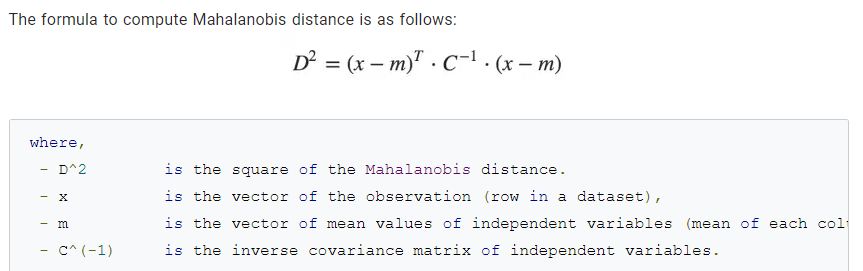
[-0.684057525, 0.016351402, -0.111223643],

[ 1.64543567, -0.111223643, 1.30807849 ]]

We have 2 mean vector: M [5.855 176.25 11.25]T ;F [5.4175 132.5 7.5]T

Test point: X=[6 130 8]T

**Mahalanobis distance from test point to 2 mean**

****

To mean Male:D2m(X, M) = (X-M)T. C-1.(X-M)

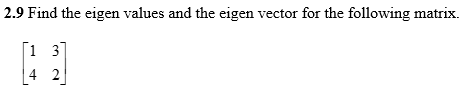
= [0.145 -46.25 -3.25]. C-1[0.145 -46.25 -3.25]T =24.116037726522777

To mean female:D2m(X, F) = (X-F)T. C-1.(X-F)

= [0.5825 -2.5 0.5]. C-1[0.5825 -2.5 0.5]T = 21.979739401247166

**Compare the 2 distance which is smaller then select the closer → Assigned to class Female**

#### **2.9 Question**



#### **2.9 Answer**

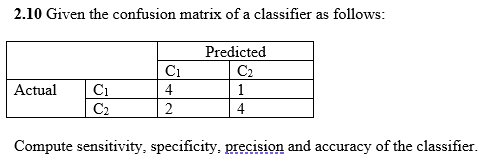
det| A - λ I | = 0

(1-λ)(2-λ)-3.4 =0

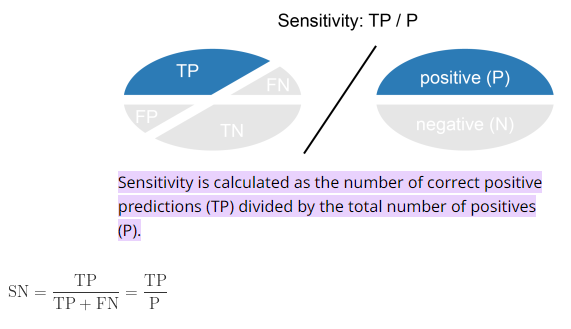
λ= -2

λ= 5

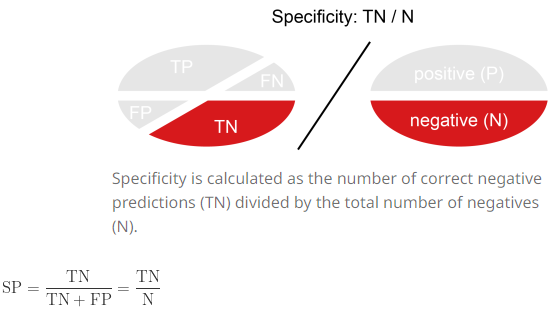
#### **2.10 Question**



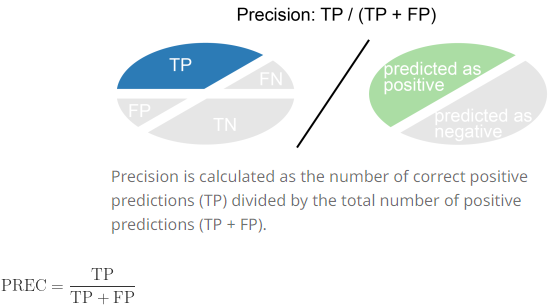
#### **2.10 Answer**



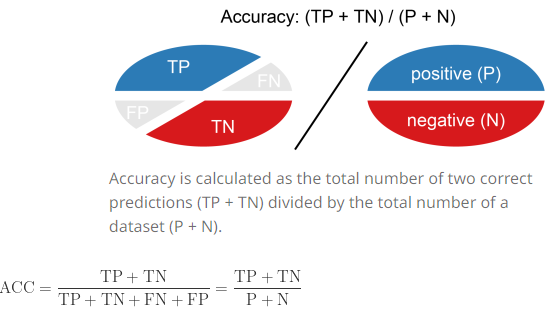
Sensitivity = 4/(4+1) =4/5



Specificity = 4/(4+2) = 4/6

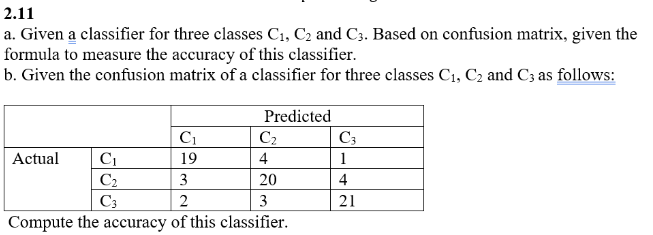


Precision = 4/(4+2)=4/6



Accuracy = (4+4)/(4+4+2+1) = 8/11

#### **2.11 Question**

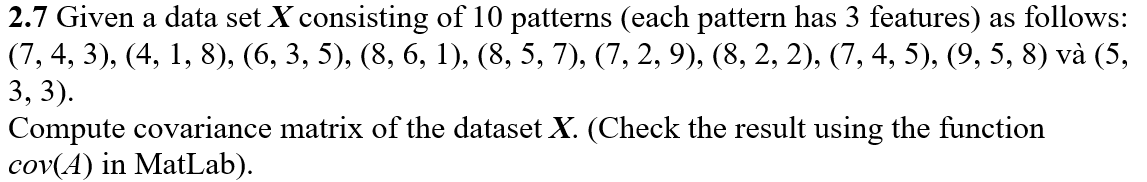


#### **2.11 Answer**

Accuracy = TP(C1 + C2 + C3)/ total = (19+20+21)/(19+20+21+4+1+3+4+2+3)

### **Toàn**

#### **2.7 Question**



*Cho tập dữ liệu X gồm 10 mẫu (mỗi mẫu có 3 đặc trưng) như sau: (7, 4, 3), (4, 1, 8), (6, 3, 5), (8, 6, 1) , (8, 5, 7), (7, 2, 9), (8, 2, 2), (7, 4, 5), (9, 5, 8) và (5, 3, 3).*

*Tính ma trận hiệp phương sai của tập dữ liệu X*

#### **2.7 Answer**

*(Nâng cao, nên dùng máy tính vì tính toán phức tạp độ lớn)*

|  | X1 | X2 | X3 |
| --- | --- | --- | --- |
|  | 7 | 4 | 3 |
|  | 4 | 1 | 8 |
|  | 6 | 3 | 5 |
|  | 8 | 6 | 1 |
|  | 8 | 5 | 7 |
|  | 7 | 2 | 9 |
|  | 8 | 2 | 2 |
|  | 7 | 4 | 5 |
|  | 9 | 5 | 8 |
|  | 5 | 3 | 3 |
| mean | **6.90** | **3.50** | **5.10** |

The matrix X can be written as:

7 4 6 8 8 7 8 7 9 5

4 1 3 6 5 2 2 4 5 3

3 8 5 1 7 9 2 5 8 3

After subtracting off the mean, the matrix X become:

*Sau khi trừ đi giá trị trung bình, ma trận X trở thành:*

0.1 -2.9 -0.9 1.1 1.1 0.1 1.1 0.1 2.1 -1.9

0.5 -2.5 -0.5 2.5 1.5 -1.5 -1.5 0.5 1.5 -0.5

-2.1 2.9 -0.1 -4.1 1.9 3.9 -3.1 -0.1 2.9 -2.1

And the transposed matrix :

0.10 0.50 -2.10

-2.90 -2.50 2.90

-0.90 -0.50 -0.10

1.10 2.50 -4.10

1.10 1.50 1.90

0.10 -1.50 3.90

1.10 -1.50 -3.10

0.10 0.50 -0.10

2.10 1.50 2.90

-1.90 -0.50 -2.10

Finally, the covariance matrix:

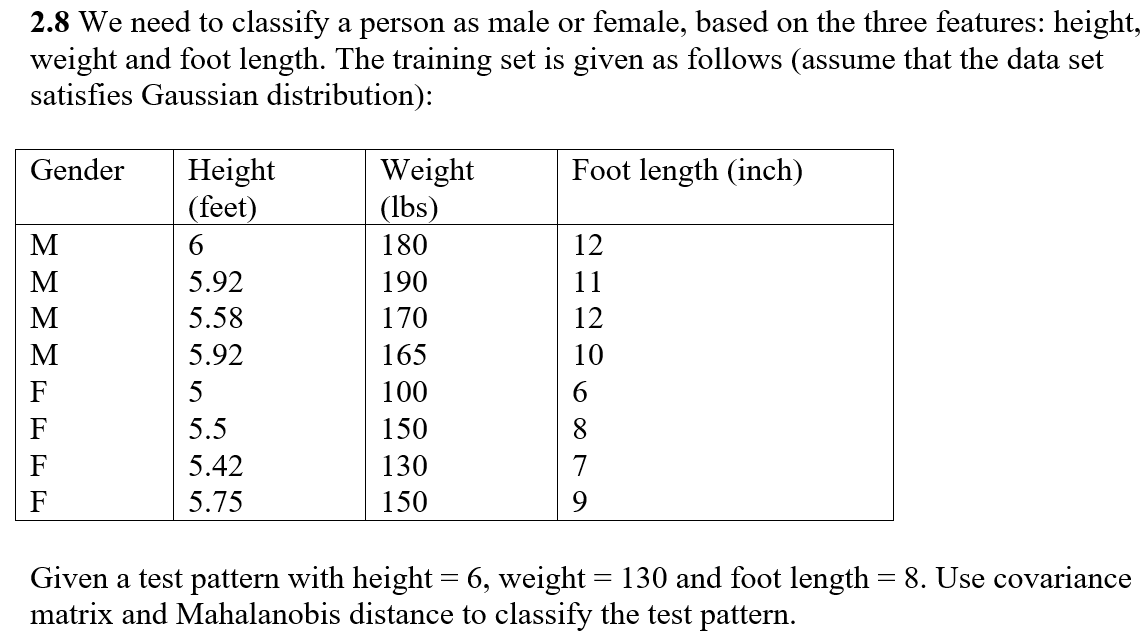
*hiệp phương sai*

2.32 1.61 -0.43

1.61 2.50 -1.28

-0.43 -1.28 7.88

#### **2.8 Question**



#### **2.8 Answer**

*(không cùng hệ giá trị nên phải normaline trc)*

Normalize A:

1.00 0.89 1.00

0.92 1.00 0.83

0.58 0.78 1.00

0.92 0.72 0.67

0.00 0.00 0.00

0.50 0.56 0.33

0.42 0.33 0.17

0.75 0.56 0.50

mean: 0.64 0.60 0.56

After subtracting off the mean, the matrix A become:

0.36 0.28 -0.06 0.28 -0.64 -0.14 -0.22 0.11

0.28 0.40 0.17 0.12 -0.60 -0.05 -0.27 -0.05

0.44 0.27 0.44 0.10 -0.56 -0.23 -0.40 -0.06

And the transposed matrix :

0.36 0.28 0.44

0.28 0.40 0.27

-0.06 0.17 0.44

0.28 0.12 0.10

-0.64 -0.60 -0.56

-0.14 -0.05 -0.23

-0.22 -0.27 -0.40

0.11 -0.05 -0.06

The covariance matrix:

0.11 0.10 0.10

0.10 0.10 0.11

0.10 0.11 0.14

⇒ =

54.00 -61.57 9.87

-61.57 132.45 -60.06

9.87 -60.06 47.09

X= [6 130 8] ⇒ X’= [1.00 0.33 0.33]



= 24.12

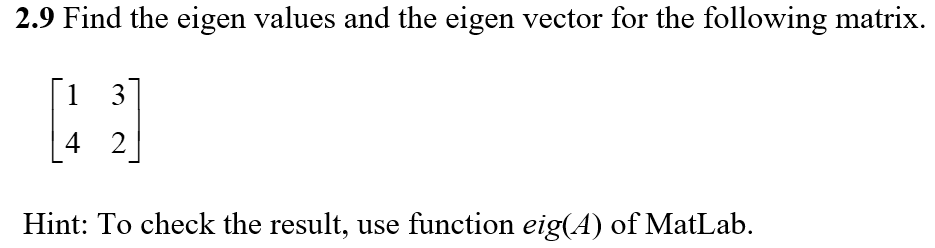
= 21.98 ⇒ The vector is assigned to class 2, since X is Female

*(Slide 14 + phụ lục) ⇒ nên để 2 bài dạng này làm sau cùng*

**[HX Long]:** Theo mình bài này trước tiên cần tách dữ liệu theo các nhóm M, F, sau đó tính u\_M, uF, tiếp theo tính D\_M = X - u\_M, D\_F = X - u\_F, tiếp theo tính được cov matrix cho từng nhóm C\_M, C\_F. Cuối cùng tính khoảng cách theo từng nhóm. Khi tách ra mình chỉ thao tác với các matrix size max là 4 => Cũng ko quá khó để tính.

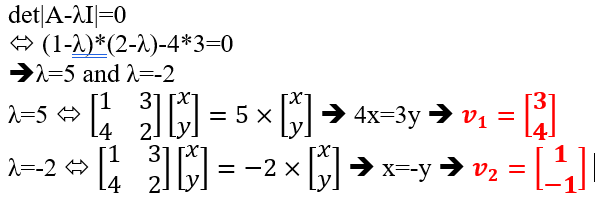
Kết quả: d\_M = 5.046 vs. d\_F = 7.325 -> **Chọn nhóm M**.

#### **2.9 Question**

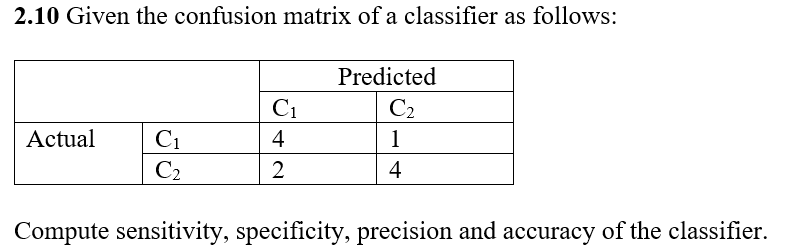


#### **2.9 Answer**

**⇒ bài dễ**

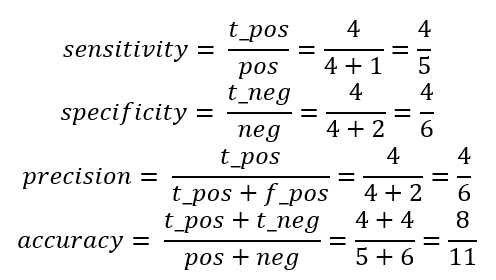


#### **2.10 Question**

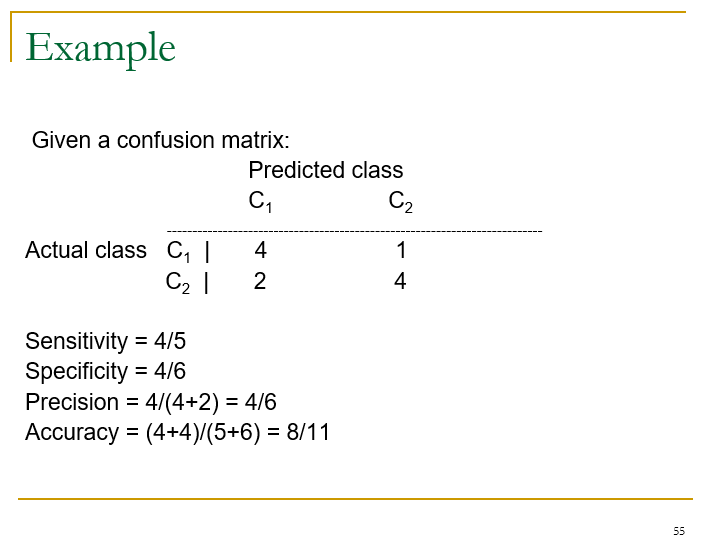


#### **2.10 Answer**

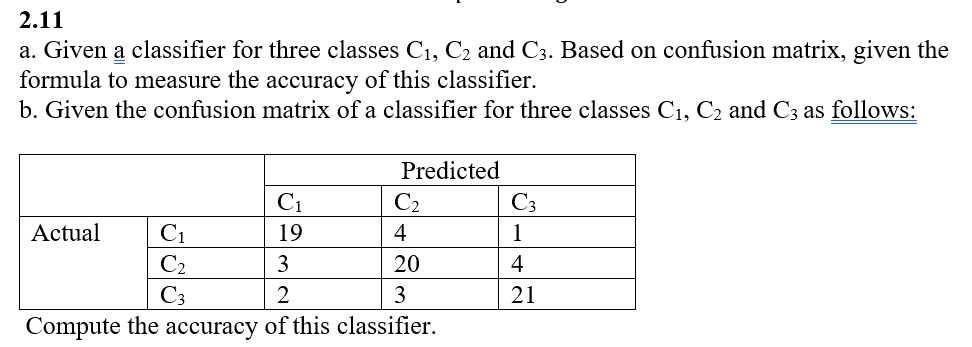
⇒ Bài dễ



**[HX Long]** Xem slide chapter 2, page 55

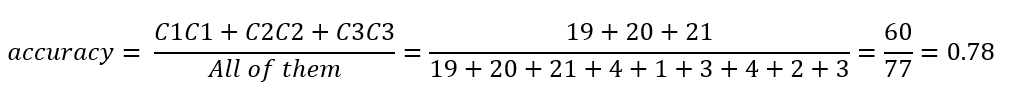


#### **2.11 Question**

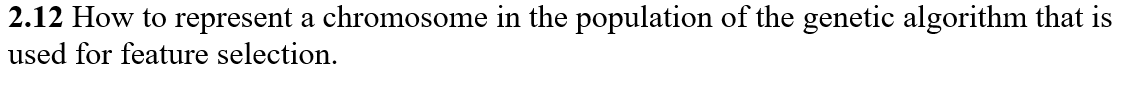


#### **2.11 Answer**

*⇒ chỉ cần lấy đường chéo chình chia tổng giá trị (có thể ra thi)*



#### **2.12 Question**



#### **2.12 Answer**

The *population* in the GA consists of strings which is binary.

Each string (*chromosome*) is of length *d*, with each position *i* being zero or one depending on the absence or presence of feature *i* in the set.

Each feature subset is coded as a *d*-element bit string. Each string in the population is a feature selection vector *a*, where *a* = *a1*,…, *ad* and *ai* assumes a value 0 if the *i*th feature is excluded and 1 if it is present in the subset.

To compute the *fitness* of a chromosome, it is evaluated by determine its performance on the training set.

*(Dân số trong GA bao gồm các chuỗi nhị phân.*

*Mỗi chuỗi (nhiễm sắc thể) có độ dài d, với mỗi vị trí i là 0 hoặc 1 tùy thuộc vào sự vắng mặt hoặc hiện diện của đặc điểm i trong tập hợp.*

*Mỗi tập hợp con tính năng được mã hóa dưới dạng chuỗi bit phần tử d. Mỗi chuỗi trong quần thể là một vectơ lựa chọn đặc trưng a, trong đó a = a1,…, ad và ai nhận giá trị 0 nếu đặc trưng thứ i bị loại trừ và 1 nếu đặc trưng thứ i có trong tập hợp con.*

*Để tính toán mức độ phù hợp của một nhiễm sắc thể, nó được đánh giá bằng cách xác định hiệu suất của nó trên tập huấn luyện.)*

