Heart Disease Prediction

Model based on Logistic Regression

Sara Carolina Gómez Delgado
Artificial Intelligence Student
Email: 0226594@up.edu.mx

Paty Yarely López Méndez
Artificial Intelligence Student
Email: 0226482@up.edu.mx

***Abstract*—In this paper, a method for the detection of cardiac diseases based on Logistic Regression, is presented. We also tried other models such as Decision Tree and Naive Bayes.**

In the dataset used, thirteen descriptors are addressed: age, sex, chest pain type, resting blood pressure, serum cholestoral(mg/dl), fasting blood sugar, resting electrocardiographic results, maximum heart rate achieved, exercise induced angina, oldpeak, the slope of peak exercise, number of major vessels colored by flourosopy and thal (normal, fixed defect, reversable defect). To demonstrate the functionality of our proposal, we asked Dr. Juan de Dios Rivera Zambrano (general doctor and the general director of the IMSS of Lagos de Moreno), to test our prediction model with real data.

Keywords—Logistic regression, Decision Tree, Naive Bayes.

I. Introduction

Image segmentation is an important concept in image processing and computer vision. It involves dividing an image into regions with similar features such as color, texture, and orientation. More specific, it is the process of labeling each pixel in the image with the objective that same label pixels have similar features. Particularly, binary segmentation divides an image in two regions and has applications in object and background discrimination, medical imaging, autonomous vehicles and intelligent systems, tracking, among others [1-5]. Some algorithms and techniques of general purpose have been developed for image segmentation [6-9]. Unfortunately, these methods can only be used in images with a small number of bands such as RGB images.

RGB images are the most common type of multiband images. In these images, the pixel value is obtained as the combination of three bands: red, green, and blue.

Satellite images are another example of multiband images. They are the result of the information of the land cover captured by sensors in artificial satellites. They are very useful for the study of weather, land cover, etc. Binary segmentation of satellite images could be used for land cover classification, detection of zones with particular features like agricultural areas, urban spaces, etc. In [10], seasonal data from several years were used for the classification of the land cover using feature selection techniques for the reduction of the dimensionality and the minimum distance to the center of the classes or maximum likelihood for classification. In [11], the environment's vegetation was detected using a supervised learning technique for classification and the NDVI (Norma- lized Difference Vegetation Index) coefficient.

Formally, an image with *n* bands is an array of *n* bidimensionality arrays where each array has the information of the corresponding band, so a multiband pixel can be defined as (equation (1)):

$$x_{ij} = [x_{ij1}, x_{ij2}, \ldots, x_{ijk}, \ldots, x_{ijn}]$$

(1) where x_{ijk} represents the value of the pixel (i, j) in the k band.

Contrary to other methods, the proposed method is capable of making binary segmentation in different kinds of multiband images.

II. Brief review of the methods used in this work

A. Descriptors

A pixel descriptor is a set of data that represents color, texture, and orientation of a specific pixel and its neighbors. The descriptors used in this work are:

Color, this descriptor has pixel values in all bands.

DCT (Discrete Cosine Transform), described in [12], is a transformation that compresses the information of an image. This descriptor captures the texture information using few values. To calculate the DCT descriptor, it is necessary to obtain a pixel window, as a matrix, centered in the main pixel. Then, the matrix B, which represents the DCT of the pixel window is calculated with equation (2).

$$B_{pq} = \alpha_p\alpha_q \sum_i \sum_j Img(i, j)c_pc_q \qquad (2)$$

where matrix B has the same size of the pixel window, p and q represent rows and column, respectively, so that $1 \leq p \leq M$ and $1 \leq q \leq N$. Constants c_p, c_q, α_p, and α_q are calculated with equations (3) and (4).

$$c_p = cos\frac{\pi(2i+1)p}{2M}, \quad c_q = cos\frac{\pi(2j+1)q}{2N}$$

(3)

$$\alpha_p = \begin{cases} \frac{1}{\sqrt{M}} & p = 1 \\ \sqrt{\frac{2}{M}} & 2 \le p \le M \end{cases} \qquad \alpha_q = \begin{cases} \frac{1}{\sqrt{N}} & q = 1 \\ \sqrt{\frac{2}{N}} & 2 \le q \le N \end{cases} \qquad (4)$$

Finally, the descriptor DCT of the pixel is the vectorization of all the DCT matrices; one for each band.

GF (Gradient Fields), this descriptor is formed with the histogram of the magnitude and the orientation of the GF of the pixel and its neighbors. A pixel gradient can be obtained using equation (5). It is possible to obtain information on the image's texture orientation with this descriptor.

$$\nabla Img(i, j) = \left(\frac{\partial Img(i, j)}{\partial x}, \frac{\partial Img(i, j)}{\partial y} \right) \qquad (5)$$

AD (Adjacency Matrix), this descriptor contains information on the number of times that a specific color is next to another color. It can be used to obtain texture information. The horizontal and vertical adjacency matrices are calculated with the information of the pixel window centered in the main pixel as shown in equations (6) and (7), respectively.

$$H_{rs} = \sum_{i=1}^{M} \sum_{j=2}^{N} \delta(Img(i, j), r)\delta(Img(i, j-1), s) \qquad (6)$$

M N

is needed. Each neighbor has n values too. So, if a pixel window of v v having n bands is used, the dimensionality of the pixel is v v n, this considering the use of only one descriptor. If more descriptors are used (with the aim of improving the dimensionality results), the problem becomes bigger. Dimensionality reduction becomes then very important to solve this problem.

WPCA (Weighted Principal Component Analysis) described in [13], is a technique for dimensionality reduction based on PCA that assumes that each dimension contributes in a different proportion to represent an information set. Specifically, segmentation can be seen as a classification problem. Therefore the contribution of each variable depends on its classification capability.

Assuming the data $X = x_{ij}$, where N is the data number and D is the dimensionality of the data, $i = 1, 2, ..., N$ and $j = 1, 2, ..., D$. The goal of WPCA is to project the data in a space of dimensionality $M < D$. The steps are:

1) Calculate a vector with the variables' weight $W = [w_1, w_2, ..., w_D]^T$. These values can be calculated with a dependency measurement between the variable and the class, for example with the Pearson Correlation Coefficient.
2) Normalize the vector W in order to have $w_j > 0 \ \forall j$ $\sum_{j=1}^{D} w_j = 1$.
3) Calculate the mean vector $\bar{X} = [\bar{x}_1, \bar{x}_2, ..., \bar{x}_D]^T$ and the variance vector $s = [s_1, s_2, ..., s_D]^T$.

$V_{rs} = \sum \sum \delta(Img(i, j), r)\delta(Img(i - 1, j), s)$ (7)

$i{=}2\ j{=}1$

where $H = \{H\}$ and $V = \{V\}$

and N are the number of rows and columns in the pixel window, b is the number of different values in the image. Thus, the values in the image must be discrete to use this descriptor, if not, they must be discretized before calculation, δ is the Kronecher delta where the result is 1 if the parameters are equal or 0 if they are different.

In order to increase the response of the adjacency, the exponential matrices of the adjacency matrices are calculated using equations (8) and (9).

$$EH = I + H + \frac{H^2}{2!} + \frac{H^3}{3!} \quad (8)$$

$$EV = I + V + \frac{V^2}{2!} + \frac{V^3}{3!} \quad (9)$$

Finally, the descriptor is formed by the vectorization of all values in EH and EV.

B. Dimensionality reduction

Segmentation of multiband images is a difficult problem mainly because of data dimensionality. The information of each pixel is composed of n values; one for each band. However, in order to obtain information about the texture and orientation, information on the pixel and its neighbors

4) Standardize the data using the means and variances $z_{ij} = \frac{x_{ij} - \bar{x}_j}{\sqrt{s_j}}$.

5) Weight the data using the vector W, $z^*_{ij} = z_{ij}$ $\sqrt{}$ w_j

6) Calculate the projection vector P as the M eigenvector . M of $(Z^*)^T (Z^*)$, where $Z^* = z^*_{ij}$, $i = 1, 2, ..., N$ and $j = 1, 2, ..., D$.

7) Relocate data to the origin $\hat{x}_{ij} = x_{ij}$ $\bar{x}_j$.

8) Calculate the main components projecting the data $Y = \hat{X} P$, where $\hat{X} = \{\hat{x}_{ij}\}$.

Finally, the new data are $Y = y_{ij}$ with $i = 1, 2, ..., N$ and $j = 1, 2, ..., M$.

QPFS (Quadratic Programming Feature Selection) is a feature selection method for classification problems using Quadratic Programming. It reduces the redundancy between variables and maximizes the dependency between the vari- ables and the class variable. The main goal is to provide a method of reasonable complexity for classification problems of high dimensionality. The importance of each variable w_i can be calculated solving the optimization problem defined in equation (10).

$$\min_w \frac{1}{2}(1 - \alpha)w^T Qw - \alpha F^T w \quad s.t. \quad w_i \geq 0, \quad \sum_{i=1}^{D} w_i = 1 \quad (10)$$

where w is the resultant vector that contains the importance of all variables, Q is a quadratic symmetric matrix that represents the redundancy between variables, and F is a vector that represents the dependency between the variables and the class variable. The size of Q is DxD and the size of F is $Dx1$, where D is the variable number. $Q = \{q_{ij}\}$, where q_{ij} represents the dependency between the i and j variables. $F = f_i$, each f_i represents the dependency between the i variable and the class variable. The dependency between variables can be calculated for discrete variables using mutual information [14] or for continuous variables with the Pearson Correlation Coefficient [15], α controls the relevance in front of the redundancy, it must be $0 \leq \alpha \leq 1$. Finally, only the variables with the bigger weights are selected.

$$\min_{p(r)} \sum_r Q(p(r), \hat{v}(r)) - \frac{\mu}{2} \sum_r ||p(r)||^2 + \frac{\lambda}{2} \sum_{<r,s>} R(p) \quad (12)$$

C. Classification

Classification consists of setting a class to a pixel, so it becomes important for the segmentation problem. The fact of using multiband images requires techniques that are capable of working with high dimensional data.

GMM (Gaussian Mixture Model), described in [16], allows to model complex distributions of data sets based on a linear combination of Gaussians. The parameters of a GMM are calculated with the EM algorithm. Formally, the method for calculating the likelihood with a GMM is with equation (11).

$$p(x) = \sum_{k=1}^{K} \pi_k N(x|\mu_k, \Sigma_k) \quad (11)$$

where x is the vector to evaluate, k is the Gaussian number, π_k is the proportion of the k Gaussian, μ_k and Σ_k are the mean vector and covarΣiance matrix of the k Gaussian, respectively, $0 \leq \pi_k \leq 1 \; \forall k$, $\sum_k \pi_k = 1$.

RF (Random Forest), described in [17], is a set of random decision trees, each one created with a random subset of the training data set. A decision tree is a prediction model based on a series of questions about the variable values to predict the class [18]. In a decision tree, data are organized in rectangular regions, product of the questions, with the objective that data in the same region are of the same class too.

D. Segmentation

The segmentation problem consists of setting the pixel class by trying that neighboring pixels have the same class. This process is important to avoid isolated assignments.

QMMF (Quadratic Markov Measure Field Model), described

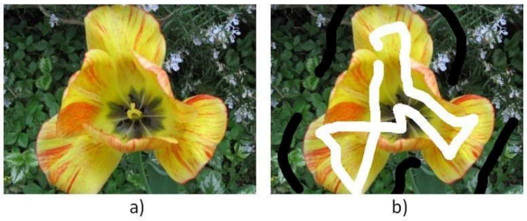


Fig. 1. User clues for segmentation: a) image and b) user clues. White pixels are a sample of pixels of class one and black pixels are another sample of pixels of class two.

where μ and λ control the contribution to each term. The first term relates the solution p with the likelihood $\hat{v}$, $Q(p(r), \hat{v}(r)) = p(r)^T D_r p(r)$, $D_r = diag(\log(\hat{v}(r)))$. The second term controls the solution entropy with the objective of keeping it small. Finally, the third term produces soft spatially solution, and r and s are first neighbors. The optimal solution can be calculated with the Gauss-Seidel projection method as described in [19].

III. Proposed Method

The proposed method performs binary segmentation of an image based on user clues at the beginning of the process with the objective of learning the features of each class. The user

in [8], calculates the probability of setting a label to the pixel (unlike hard segmentation that sets the label to the pixel). Once the likelihood of each pixel is calculated as a normalized vector $\hat{v}(r)$, QMMF calculates the probability $p(r)$ as a normalized vector that shows the probability of belonging to each class based on the likelihood, the neighbor probabilities, and the entropy. QMMF is based on the Quadratic Programming equation defined in (12).

clues consist of marking a sample of pixels as class one and marking another sample of pixels as class two. Figure 1 shows an example of user clues.

The method is divided in two phases:

1) Training with the marked pixels
2) Classification of the non marked pixels and segmentation

A. Training with the marked pixels

The first phase consists of recognizing the features of each class based on user clues, this is to find the descriptors that identify the features that separate the classes, calculate the parameters to an optimal dimensional reduction, and calculate the parameters to the classification models.

In this phase we only use the information of the marked pixels, the user clues, to learn the models for the classification. The steps, shown in the figure 2, are:

1) Create the descriptors: Color, DCT, GF, and AD for the marked pixels.
2) Reduce the descriptors' dimensionality using: WPCA and QPFS. The result is eight reduced descriptors in two groups:
    a) Group 1: Color reduced with QPFS, DCT reduced with QPFS, GF reduced with QPFS, and AD reduced with QPFS.
    b) Group 2: Color reduced with WPCA, DCT reduced with WPCA, GF reduced with WPCA, and AD reduced with WPCA.

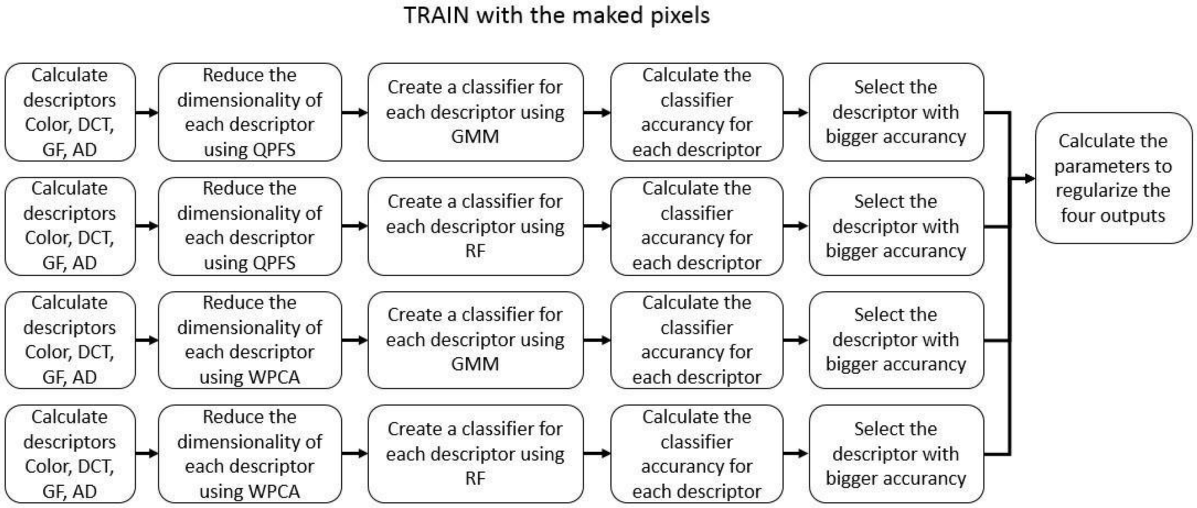

TRAIN with the maked pixels
Calculate descriptors Color, DCT, GF, AD
Reduce the dimensionality of each descriptor using QPFS
Create a classifier for each descriptor using GMM
Calculate the classifier accurancy for each descriptor
Select the descriptor with bigger accurancy
Calculate descriptors Color, DCT, GF, AD
Reduce the dimensionality of each descriptor using QPFS
Create a classifier for each descriptor using RF
Calculate the classifier accurancy for each descriptor
Select the descriptor with bigger accurancy
Calculate descriptors Color, DCT, GF, AD
Reduce the dimensionality of each descriptor using WPCA
Create a classifier for each descriptor using GMM
Calculate the classifier accurancy for each descriptor
Select the descriptor with bigger accurancy
Calculate descriptors Color, DCT, GF, AD
Reduce the dimensionality of each descriptor using WPCA
Create a classifier for each descriptor using RF
Calculate the classifier accurancy for each descriptor
Select the descriptor with bigger accurancy
Calculate the parameters to regularize the four outputs

Fig. 2. Proposed method, first phase.

3) Create the models for the classification. As we have two dimensionality reduction techniques and two classification methods, four combinations appear:
   a) Group 1: GMM of descriptors reduced with QPFS
   b) Group 2: RF of descriptors reduced with QPFS
   c) Group 3: GMM of descriptors reduced with WPCA
   d) Group 4: RF of descriptors reduced with WPCA
4) Classification of the marked pixels with the models created in the previous step; 16 different likelihoods for each marked pixel are calculated (four groups of four descriptors each).
5) Select the descriptor and the model for each group. Thus, for each group:
   a) Calculate the accuracy of the classifier of each descriptor with equation (13).

$$Accuracy = \frac{\sum_{i} \phi(V_i^1, V_i^2, C_i)}{Nmp} \quad (13)$$

$$\phi(V_i^1, V_i^2, C_i) = \begin{cases} 1 & if \quad C_i = 1, V_i^1 \geq V_i^2 \\ 1 & if \quad C_i = 2, V_i^2 \geq V_i^1 \\ 0 & otherwise \end{cases} \quad (14)$$

$$1 \quad \frac{V_i^1 + \varepsilon}{V_i^1 + V_i^2 + \varepsilon} \quad (15)$$

6) Calculate the weight of each group for classification.
   a) Calculate the classification efficiency for each group a^g, where $g = 1, 2, 3, 4$, a^g represents the classification efficiency of the group g, and it is calculated as the accuracy of the normalized likelihood.
   b) Regularize the accuracy, this is, $\hat{a}^1 + \hat{a}^2 + \hat{a}^3 + \hat{a}^4 = 1$, with equation (16)

$$1 \quad \frac{a^1}{a_1+a_2+a_3+a_4} \qquad 2 \quad \frac{a^2}{a_1+a_2+a_3+a_4} \quad (16)$$
$$3 \quad \frac{a^3}{a_1+a_2+a_3+a_4} \qquad 4 \quad \frac{a^4}{a_1+a_2+a_3+a_4}$$

B. *Classification of the non marked pixels and segmentation*

The second phase performs classification of the non marked pixels and segmentation of the result. It means: to calculate the descriptor of the non marked pixels, to reduce the descriptors dimensionality, and to calculate the likelihood of the non marked pixels with the models created in the first phase, where Nmp is the number of marked pixels, $< i >$ represents only the marked pixels, V_i^1 and V_i^2 are calculated with the classifier and they represent the likelihood that the pixel i be part of the class 1 and class 2, respectively. Finally, C_i is the real class of the pixel i.

b) Select the descriptor which corresponds to the classifier with the bigger accuracy.

c) Select the model that corresponds to the selected descriptor.

d) Normalize the likelihoods of the descriptor selected, with the goal of $V^1 + V_2 = 1$, using equation (15).

combination of the four classifiers results, and segmentation. The steps of this phase, shown in figure 3, are:

1) Create the selected descriptors for the non marked pixels.
2) Reduce the descriptors dimensionality using WPCA and QPFS.
3) Classify the non marked pixels using the reduced descriptors and the models created and selected in the first phase. The result of this step must be four likelihoods for each non marked pixel, one for each group:
   a) Group 1: QPFS and GMM
   b) Group 2: QPFS and RF
   c) Group 3: WPCA and GMM

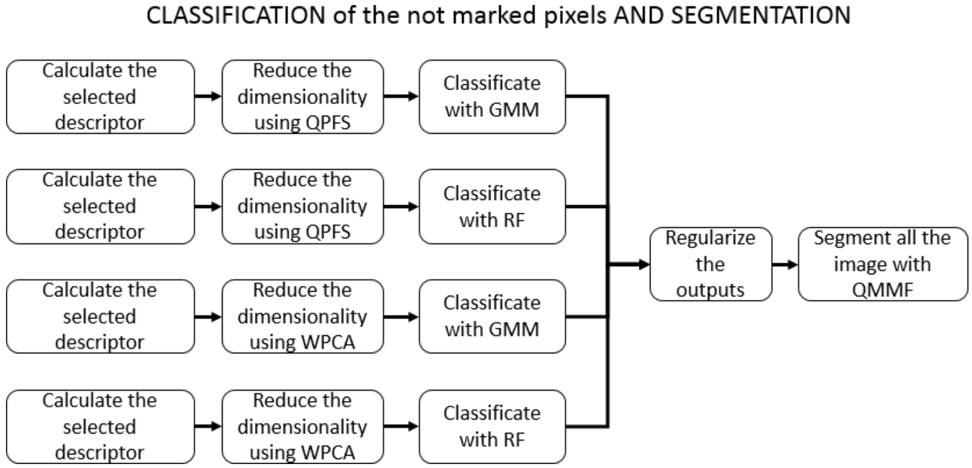

CLASSIFICATION of the not marked pixels AND SEGMENTATION
Calculate the selected descriptor
Reduce the dimensionality using QPFS
Classificate with GMM
Calculate the selected descriptor
Reduce the dimensionality using QPFS
Classificate with RF
Calculate the selected descriptor
Reduce the dimensionality using WPCA
Classificate with GMM
Calculate the selected descriptor
Reduce the dimensionality using WPCA
Classificate with RF
Regularize the outputs
Segment all the image with QMMF

Fig. 3. Proposed method, second phase.

d) Group 4: WPCA and RF

4) Calculate the general likelihood using the accuracy of each group.
   a) Normalize the likelihood of each group, as mentioned in equation (15).
   b) Combine the classifiers results (equation (17)).

$$\hat{V}^{1} = \sum_{g=1}^{4} \hat{a}^{g} V^{1}_{gi} \qquad (17)$$

5) Segment the likelihood of all pixels (marked and non marked) using QMMF.

IV. Experiments and Results

A. Statlog dataset

The Statlog data set [20] has information of satellite images of the Landsat satellite. This data set consists of multivariate data of pixels in 3x3 neighborhoods and the classification of the central pixel. The objective is to predict the classification. The pixel class is coded with a number that represents: 1 red soil, 2 cotton crop, 3 gray soil, 4 damp gray soil, 5 soil with vegetation stubble, 6 mixture class, and 7 very damp gray soil. Each neighborhood is represented by 36 variables plus the class, the data set is composed by 6,435 data. The objective of this experiment is to verify that the classifiers combination is better than each individual classifier when we have high dimensional data. The results are shown in the table I .

B. Images with real textures

Images with real textures were obtained from the Microsoft Research Cambridge data set [21]. The objective of this experiment is to measure the efficiency of the proposed method with real images in RGB format. The data set consists of 50 images. Three are shown in the figure 4. The mean percentage of error in the segmentation of all these images obtained with the proposed method is 3.6%.

TABLE I
Percentage of Error in the Classification of the Statlog Data set

C. Images with different textures

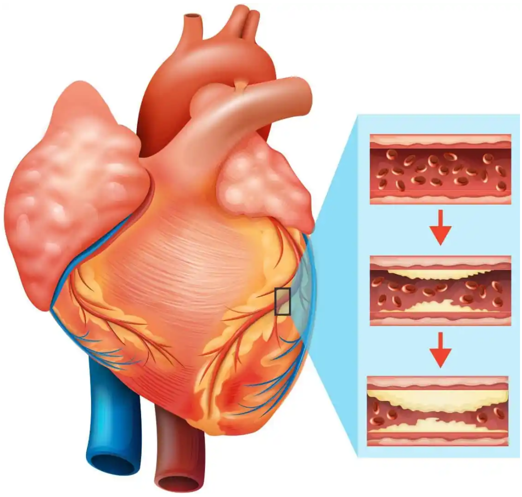
Images with different textures were created based on the Microsoft Research Cambridge data set [21] by replacing the main object and the background with texture images. The objective of this experiment is to measure the efficiency of the proposed method for segmentation of color, texture, and orientation. Results are shown in figure 5.

V. Conclusion

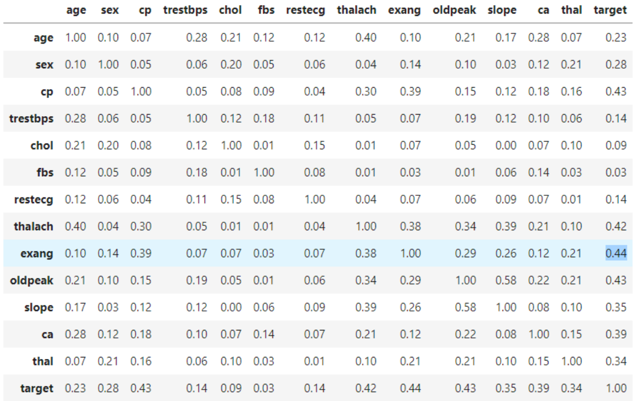
This paper has presented a method for the detection of heart diseases based on Logistic Regression. This model got much better results than the Decision Tree model. Naive Bayes almost reached the good results of Logistic Regression. However, this last one outperformed the other models.

When obtaining the relationship between the target (1= has heart disease, 0= does not have heart disease) and the other variables, we realize that there are some that stand out more than others. For example: exercise-induced angina.

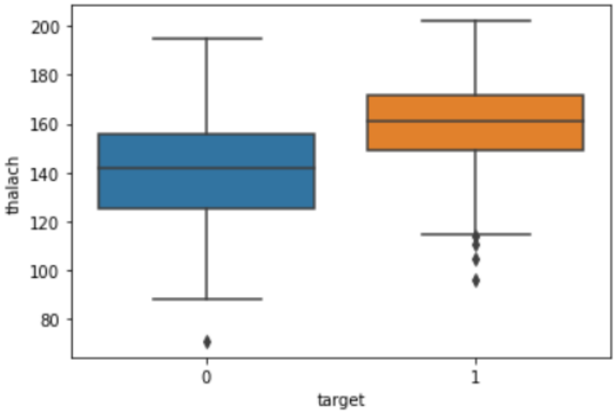
During the interview with Dr. Juan de Dios confirmed that those variables with the highest correlation against target, are those that can become critical in the face of a possible heart disease.



This is the example of a heart with ischemia, a disease caused by lack of blood circulation in some part of the heart, which has exercise-induced angina as a symptom. This was used as training data. And it was one of the strongest correlations between whether or not a patient had heart disease (0.44).



	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
age	1.00	0.10	0.07	0.28	0.21	0.12	0.12	0.40	0.10	0.21	0.17	0.28	0.07	0.23
sex	0.10	1.00	0.05	0.06	0.20	0.05	0.06	0.04	0.14	0.10	0.03	0.12	0.21	0.28
cp	0.07	0.05	1.00	0.05	0.08	0.09	0.04	0.30	0.39	0.15	0.12	0.18	0.16	0.43
trestbps	0.28	0.06	0.05	1.00	0.12	0.18	0.11	0.05	0.07	0.19	0.12	0.10	0.06	0.14
chol	0.21	0.20	0.08	0.12	1.00	0.01	0.15	0.01	0.07	0.05	0.00	0.07	0.10	0.09
fbs	0.12	0.05	0.09	0.18	0.01	1.00	0.08	0.01	0.03	0.01	0.06	0.14	0.03	0.03
restecg	0.12	0.06	0.04	0.11	0.15	0.08	1.00	0.04	0.07	0.06	0.09	0.07	0.01	0.14
thalach	0.40	0.04	0.30	0.05	0.01	0.01	0.04	1.00	0.38	0.34	0.39	0.21	0.10	0.42
exang	0.10	0.14	0.39	0.07	0.07	0.03	0.07	0.38	1.00	0.29	0.26	0.12	0.21	0.44
oldpeak	0.21	0.10	0.15	0.19	0.05	0.01	0.06	0.34	0.29	1.00	0.58	0.22	0.21	0.43
slope	0.17	0.03	0.12	0.12	0.00	0.06	0.09	0.39	0.26	0.58	1.00	0.08	0.10	0.35
ca	0.28	0.12	0.18	0.10	0.07	0.14	0.07	0.21	0.12	0.22	0.08	1.00	0.15	0.39
thal	0.07	0.21	0.16	0.06	0.10	0.03	0.01	0.10	0.21	0.21	0.10	0.15	1.00	0.34
target	0.23	0.28	0.43	0.14	0.09	0.03	0.14	0.42	0.44	0.43	0.35	0.39	0.34	1.00



We also found that the faster the heart rate, the more likely it is that a heart attack will occur. It is really uncommon that people with a heart rate under 120 bpm, have heart problems.

REFERENCES

[1] *Heart disease - Symptoms and causes*. (2021, 9 febrero). Mayo Clinic. https://www.mayoclinic.org/diseases-conditions/heart-disease/symptoms-causes/syc-20353118