

## Data Analysis

DATA ANALYSIS WITH STUDENT INFORMATION DATABASE

Table of Contents

1. [Aim of the Project 2](#_bookmark0)
2. [Dataset Description 2](#_bookmark1)
3. [Explanation of Data Analysis Technique 3](#_bookmark2)
4. [MATLAB code 4](#_bookmark3)
5. [Execution Outcome 12](#_bookmark4)
6. [Evaluating Results 12](#_bookmark5)

# Aim of the Project

Finding a safe blood donor is a challenging quest. Every person should know the importance of blood donation. Hospitals or government cannot maintain health care without adequate blood donation. In order to prevent such situations, blood donor organizations gain importance. The problem is whether or not the characteristics of blood donors meet the required standards. Blood data is not always very large. Therefore, classification should be applied to a data of this nature. Thus, the results sought can be easily estimated. The classification algorithm will find the relationships between the donors' characteristics to predict the outcome.

This project aims to create a data-driven system for predicting and observing the blood to be donated. Bayes classification algorithm was preferred to realize this prediction.

# Dataset Description

Title: Blood Transfusion Service Center Data Set Size: 748 x 5

Attribute Type of Dataset: Numerical Default Task of Dataset: Classification Number of Instances: 748

Area: Business

Attribute Characteristics: Real

Number of Attributes: 5 (without classification column it is 4) Date Donated: 2008-10-03

Associated Tasks: Classification

Data taken from the Blood Transfusion Service Center in Taiwan. Recency - months since last donation,

Frequency - total number of donation, Monetary - total blood donated in cc., Time - months since first donation,

a binary variable representing whether he/she donated blood in March 2007 (1 stand for donating blood; 0 stands for not donating blood).

The dataset is associated with blood donations at a blood transfusion service center. In the continuation of the project, it will be easier to estimate the characteristics of the blood of future blood donors by classifying the dataset into different classes

The dataset research consists of 748 people in total. This number is quite large. Therefore, consistent results will be obtained. However, since the dataset is taken from a place in Taiwan, there may be slight differences all over the world.

Blood donation is very important. However, classification is important for easier and more accurate results. This dataset was chosen to see how the Naive Bayes classification method would help with blood donation.

# Explanation of Data Analysis Technique

The Bayes theorem was discovered in 1812 by Thomas Bayes to calculate conditional probability.

The Naive Bayes classification used in this project is based on Bayes theorem. The way in which the algorithm works generally calculates the probability of each state for an element and classifies it according to the highest probability value. The difference between the Bayes rule and the Naive Bayes rule is that the features are independent.

Step 1: Calculate the previous possibilities for given class labels. Step 2: Find the probability for each attribute for each class

Step 3: Replace these values in the Bayesian formula and calculate posterior probability

Step 4: Find out which class has the highest probability. The input given belongs to the higher probability class.

# MATLAB code

classdef DA\_Project < matlab.apps.AppBase

% Properties that correspond to app components properties (Access = public)

BloodTransfussionClassificationUIFigure matlab.ui.Figure Image matlab.ui.control.Image

EntervaluesLabel matlab.ui.control.Label firstVal matlab.ui.control.NumericEditField title matlab.ui.control.Label

ValuesLabel matlab.ui.control.Label LetsCalculateButton matlab.ui.control.Button Image2 matlab.ui.control.Image

end

% Callbacks that handle component events methods (Access = private)

% Code that executes after component creation function startupFcn(app)

%Reading dataset and getting the inputs from user by using input operation

matrix = xlsread("Transfusion.xlsx"); attributesNum = size(matrix,2) - 1 ; X = zeros(attributesNum,1);

for i=1:attributesNum disp("value") waitfor(app.firstVal, 'Value'); value = app.firstVal.Value; X(i,1) = value; app.firstVal.Value = Inf;

%Update text of ValuesLabel (for demostrating the concept). text = ['Values: ', sprintf('%.1f, ', X(1:i))]; app.ValuesLabel.Text = text(1:end-2);

end

%By using this, we create 2 matrices. class 0 and class 1

mask = matrix(:,end) == 0; c0 = matrix(mask,:);

c1 = matrix(~mask,:);

%By using the input data we will first find the mean of each classes means

%of columns. To do that we are going to use mean() operation. This

%operation takes the mean of each column of each class

% m0=mean(c0(:,attributesNum));

% m1=mean(c1(:,attributesNum)); m0 = zeros(attributesNum,1);

m1 = zeros(attributesNum,1); for i=1:attributesNum

for j=1:attributesNum m0(i)= mean(c0(i,j));

m1(i)= mean(c1(i,j));

end end

%By using those mean values we are going to find centered data matrix of

%the dataset for each column of each class

z0=zeros();

meanC0 = mean(c0); Z0 = c0' - meanC0'; for i=1:attributesNum

for j=1:attributesNum z0(i,j)= Z0(i,j);

end end

z1 = zeros(); meanC1 = mean(c1); Z1 = c1' - meanC1';

for i=1:attributesNum for j=1:attributesNum

z1(i,j)= Z1(i,j); end

end

%At this part we are calculating the standard deviation of each column of

%each classes centered data matrix columns arr0 = [];

arr0=size(attributesNum); stdC0 = [];

for a=1:size(z0,1) for b=1:size(z0,1)

arr0 = [arr0 z0(a,b)]; stdC0(1,b) = var(arr0);

end end

arr1 = [];

arr1=size(attributesNum); stdC1 = [];

for a=1:size(z1,1) for b=1:size(z1,1)

arr1 = [arr1 z1(a,b)]; stdC1(1,b) = var(arr1);

end

end

%In this part we are calculating the size of both classes and the whole

%matrix. Then after we found those values we are going to find the

%probability of each classes

n0 = size(c0,1); n1 = size(c1,1);

n = size(matrix,1);

Pc0 = n0/n;

Pc1 = n1/n;

%In this part we compute the probability density function (pdf) values evaluated at the values in xi i=1,2,3,4 for the normal distribution with mean mu and standard deviation sigma.

xes = zeros(attributesNum,1); for p = 1 : attributesNum

xes(p) = X(p,1); end

% x1 = X(1,:);

% x2 = X(2,:);

% x3 = X(3,:);

% x4 = X(4,:);

probs0 = zeros(attributesNum,1); for q=1:attributesNum

for w=1:attributesNum

probs0(q,1) = 1/sqrt(2\*pi\*stdC0(1,w))\*exp(-(xes(q)-m0(q))^2/(2\*stdC0(1,w)));

end end

probs1 = zeros(attributesNum,1); for q=1:attributesNum

for w=1:attributesNum

probs1(q,1) = 1/sqrt(2\*pi\*stdC1(1,w))\*exp(-(xes(q)-m1(q))^2/(2\*stdC1(1,w)));

end end

disp(probs1);

%Then we are going to multiply all the probabilities of each classes with

%probability density function of the class( For each class seperately).

%Then we take the maximum of the two values that we calculate. Then answer

%is the maximum one. Which one is maximum, it is the class what the user

%input included.

disp("fc"); fc0=1;

for t=1:attributesNum fc0 = fc0 \* probs0(t);

end

Class0 = fc0 \* Pc0;

fc1=1;

for t=1:attributesNum fc1 = fc1 \* probs1(t);

end

Class1 = Pc1 \* fc1;

%

% disp(Class0);

% disp(Class1);

y = max(Class0,Class1); if y == Class0

didnotdonate

disp("According to input data, He/she did not donate blood in March 2017"); else

diddonate

disp("According to input data, He/she did donate blood in March 2017");

end

end

% Image clicked function: Image2

function Image2Clicked(app, ~)

% Make current instance of app invisible app.BloodTransfussionClassificationUIFigure.Visible = 'off';

% Open 2nd instance of app

app1(); % < The name of your app

% Delete old instance close(app.BloodTransfussionClassificationUIFigure)

end end

% Component initialization methods (Access = private)

% Create UIFigure and components function createComponents(app)

% Create BloodTransfussionClassificationUIFigure and hide until all components are created

app.BloodTransfussionClassificationUIFigure = uifigure('Visible', 'off'); app.BloodTransfussionClassificationUIFigure.Color = [0.8588 0.7569 0.8039];

app.BloodTransfussionClassificationUIFigure.Colormap = [0.2431 0.149

0.6588;0.2431 0.1529 0.6745;0.2471 0.1569 0.6863;0.2471 0.1608 0.698;0.251 0.1647

0.7059;0.251 0.1686 0.7176;0.2549 0.1725 0.7294;0.2549 0.1765 0.7412;0.2588 0.1804

0.749;0.2588 0.1843 0.7608;0.2627 0.1882 0.7725;0.2588 0.1882 0.7804;0.2627 0.1961

0.7922;0.2667 0.2 0.8039;0.2667 0.2039 0.8157;0.2706 0.2078 0.8235;0.2706 0.2157

0.8353;0.2706 0.2196 0.8431;0.2745 0.2235 0.851;0.2745 0.2275 0.8627;0.2745 0.2314

0.8706;0.2745 0.2392 0.8784;0.2784 0.2431 0.8824;0.2784 0.2471 0.8902;0.2784 0.2549

0.898;0.2784 0.2588 0.902;0.2784 0.2667 0.9098;0.2784 0.2706 0.9137;0.2784 0.2745

0.9216;0.2824 0.2824 0.9255;0.2824 0.2863 0.9294;0.2824 0.2941 0.9333;0.2824 0.298

0.9412;0.2824 0.3059 0.9451;0.2824 0.3098 0.949;0.2824 0.3137 0.9529;0.2824 0.3216

0.9569;0.2824 0.3255 0.9608;0.2824 0.3294 0.9647;0.2784 0.3373 0.9686;0.2784 0.3412

0.9686;0.2784 0.349 0.9725;0.2784 0.3529 0.9765;0.2784 0.3569 0.9804;0.2784 0.3647

0.9804;0.2745 0.3686 0.9843;0.2745 0.3765 0.9843;0.2745 0.3804 0.9882;0.2706 0.3843

0.9882;0.2706 0.3922 0.9922;0.2667 0.3961 0.9922;0.2627 0.4039 0.9922;0.2627 0.4078

0.9961;0.2588 0.4157 0.9961;0.2549 0.4196 0.9961;0.251 0.4275 0.9961;0.2471 0.4314

1;0.2431 0.4392 1;0.2353 0.4431 1;0.2314 0.451 1;0.2235 0.4549 1;0.2196 0.4627

0.9961;0.2118 0.4667 0.9961;0.2078 0.4745 0.9922;0.2 0.4784 0.9922;0.1961 0.4863

0.9882;0.1922 0.4902 0.9882;0.1882 0.498 0.9843;0.1843 0.502 0.9804;0.1843 0.5098

0.9804;0.1804 0.5137 0.9765;0.1804 0.5176 0.9725;0.1804 0.5255 0.9725;0.1804 0.5294

0.9686;0.1765 0.5333 0.9647;0.1765 0.5412 0.9608;0.1765 0.5451 0.9569;0.1765 0.549

0.9529;0.1765 0.5569 0.949;0.1725 0.5608 0.9451;0.1725 0.5647 0.9412;0.1686 0.5686

0.9373;0.1647 0.5765 0.9333;0.1608 0.5804 0.9294;0.1569 0.5843 0.9255;0.1529 0.5922

0.9216;0.1529 0.5961 0.9176;0.149 0.6 0.9137;0.149 0.6039 0.9098;0.1451 0.6078

0.9098;0.1451 0.6118 0.9059;0.1412 0.6196 0.902;0.1412 0.6235 0.898;0.1373 0.6275

0.898;0.1373 0.6314 0.8941;0.1333 0.6353 0.8941;0.1294 0.6392 0.8902;0.1255 0.6471

0.8902;0.1216 0.651 0.8863;0.1176 0.6549 0.8824;0.1137 0.6588 0.8824;0.1137 0.6627

0.8784;0.1098 0.6667 0.8745;0.1059 0.6706 0.8706;0.102 0.6745 0.8667;0.098 0.6784

0.8627;0.0902 0.6824 0.8549;0.0863 0.6863 0.851;0.0784 0.6902 0.8471;0.0706 0.6941

0.8392;0.0627 0.698 0.8353;0.0549 0.702 0.8314;0.0431 0.702 0.8235;0.0314 0.7059

0.8196;0.0235 0.7098 0.8118;0.0157 0.7137 0.8078;0.0078 0.7176 0.8;0.0039 0.7176

0.7922;0 0.7216 0.7882;0 0.7255 0.7804;0 0.7294 0.7765;0.0039 0.7294 0.7686;0.0078

0.7333 0.7608;0.0157 0.7333 0.7569;0.0235 0.7373 0.749;0.0353 0.7412 0.7412;0.051

0.7412 0.7373;0.0627 0.7451 0.7294;0.0784 0.7451 0.7216;0.0902 0.749 0.7137;0.102

0.7529 0.7098;0.1137 0.7529 0.702;0.1255 0.7569 0.6941;0.1373 0.7569 0.6863;0.1451

0.7608 0.6824;0.1529 0.7608 0.6745;0.1608 0.7647 0.6667;0.1686 0.7647 0.6588;0.1725

0.7686 0.651;0.1804 0.7686 0.6471;0.1843 0.7725 0.6392;0.1922 0.7725 0.6314;0.1961

0.7765 0.6235;0.2 0.7804 0.6157;0.2078 0.7804 0.6078;0.2118 0.7843 0.6;0.2196 0.7843

0.5882;0.2235 0.7882 0.5804;0.2314 0.7882 0.5725;0.2392 0.7922 0.5647;0.251 0.7922

0.5529;0.2588 0.7922 0.5451;0.2706 0.7961 0.5373;0.2824 0.7961 0.5255;0.2941 0.7961

0.5176;0.3059 0.8 0.5059;0.3176 0.8 0.498;0.3294 0.8 0.4863;0.3412 0.8 0.4784;0.3529

0.8 0.4667;0.3686 0.8039 0.4549;0.3804 0.8039 0.4471;0.3922 0.8039 0.4353;0.4039

0.8039 0.4235;0.4196 0.8039 0.4118;0.4314 0.8039 0.4;0.4471 0.8039 0.3922;0.4627 0.8

0.3804;0.4745 0.8 0.3686;0.4902 0.8 0.3569;0.5059 0.8 0.349;0.5176 0.8 0.3373;0.5333

0.7961 0.3255;0.5451 0.7961 0.3176;0.5608 0.7961 0.3059;0.5765 0.7922 0.2941;0.5882

0.7922 0.2824;0.6039 0.7882 0.2745;0.6157 0.7882 0.2627;0.6314 0.7843 0.251;0.6431

0.7843 0.2431;0.6549 0.7804 0.2314;0.6706 0.7804 0.2235;0.6824 0.7765 0.2157;0.698

0.7765 0.2078;0.7098 0.7725 0.2;0.7216 0.7686 0.1922;0.7333 0.7686 0.1843;0.7451

0.7647 0.1765;0.7608 0.7647 0.1725;0.7725 0.7608 0.1647;0.7843 0.7569 0.1608;0.7961

0.7569 0.1569;0.8078 0.7529 0.1529;0.8157 0.749 0.1529;0.8275 0.749 0.1529;0.8392

0.7451 0.1529;0.851 0.7451 0.1569;0.8588 0.7412 0.1569;0.8706 0.7373 0.1608;0.8824

0.7373 0.1647;0.8902 0.7373 0.1686;0.902 0.7333 0.1765;0.9098 0.7333 0.1804;0.9176

0.7294 0.1882;0.9255 0.7294 0.1961;0.9373 0.7294 0.2078;0.9451 0.7294 0.2157;0.9529

0.7294 0.2235;0.9608 0.7294 0.2314;0.9686 0.7294 0.2392;0.9765 0.7294 0.2431;0.9843

0.7333 0.2431;0.9882 0.7373 0.2431;0.9961 0.7412 0.2392;0.9961 0.7451 0.2353;0.9961

0.7529 0.2314;0.9961 0.7569 0.2275;0.9961 0.7608 0.2235;0.9961 0.7686 0.2196;0.9961

0.7725 0.2157;0.9961 0.7804 0.2078;0.9961 0.7843 0.2039;0.9961 0.7922 0.2;0.9922

0.7961 0.1961;0.9922 0.8039 0.1922;0.9922 0.8078 0.1922;0.9882 0.8157 0.1882;0.9843

0.8235 0.1843;0.9843 0.8275 0.1804;0.9804 0.8353 0.1804;0.9765 0.8392 0.1765;0.9765

0.8471 0.1725;0.9725 0.851 0.1686;0.9686 0.8588 0.1647;0.9686 0.8667 0.1647;0.9647

0.8706 0.1608;0.9647 0.8784 0.1569;0.9608 0.8824 0.1569;0.9608 0.8902 0.1529;0.9608

0.898 0.149;0.9608 0.902 0.149;0.9608 0.9098 0.1451;0.9608 0.9137 0.1412;0.9608

0.9216 0.1373;0.9608 0.9255 0.1333;0.9608 0.9333 0.1294;0.9647 0.9373 0.1255;0.9647

0.9451 0.1216;0.9647 0.949 0.1176;0.9686 0.9569 0.1098;0.9686 0.9608 0.1059;0.9725

0.9686 0.102;0.9725 0.9725 0.0941;0.9765 0.9765 0.0863;0.9765 0.9843 0.0824];

app.BloodTransfussionClassificationUIFigure.Position = [100 100 640 480]; app.BloodTransfussionClassificationUIFigure.Name = 'Blood Transfussion

Classification';

% Create Image

app.Image = uiimage(app.BloodTransfussionClassificationUIFigure); app.Image.Position = [46 269 161 128];

app.Image.ImageSource = '1297099.png';

% Create EntervaluesLabel

app.EntervaluesLabel = uilabel(app.BloodTransfussionClassificationUIFigure); app.EntervaluesLabel.HorizontalAlignment = 'right'; app.EntervaluesLabel.Position = [237 230 86 22];

app.EntervaluesLabel.Text = 'Enter values: ';

% Create firstVal

app.firstVal = uieditfield(app.BloodTransfussionClassificationUIFigure, 'numeric'); app.firstVal.Position = [348 230 100 22];

% Create title

app.title = uilabel(app.BloodTransfussionClassificationUIFigure); app.title.FontName = 'Californian FB';

app.title.FontSize = 20; app.title.FontWeight = 'bold'; app.title.Position = [237 269 340 133];

app.title.Text = 'Blood Transfussion Classification';

% Create ValuesLabel

app.ValuesLabel = uilabel(app.BloodTransfussionClassificationUIFigure); app.ValuesLabel.FontName = 'Bell MT';

app.ValuesLabel.FontSize = 15;

app.ValuesLabel.Position = [46 67 552 46]; app.ValuesLabel.Text = 'Values';

% Create LetsCalculateButton

app.LetsCalculateButton = uibutton(app.BloodTransfussionClassificationUIFigure, 'push');

app.LetsCalculateButton.BackgroundColor = [0.0745 0.6235 1]; app.LetsCalculateButton.FontName = 'Lucida Sans'; app.LetsCalculateButton.FontSize = 16; app.LetsCalculateButton.FontWeight = 'bold'; app.LetsCalculateButton.Position = [274 151 138 41]; app.LetsCalculateButton.Text = 'Let''s Calculate!';

% Create Image2

app.Image2 = uiimage(app.BloodTransfussionClassificationUIFigure); app.Image2.ImageClickedFcn = createCallbackFcn(app, @Image2Clicked, true); app.Image2.Tooltip = {'Restart '};

app.Image2.Position = [561 12 63 56]; app.Image2.ImageSource = '26868-8-restart-file.png';

% Show the figure after all components are created app.BloodTransfussionClassificationUIFigure.Visible = 'on';

end end

% App creation and deletion methods (Access = public)

% Construct app

function app = DA\_Project

% Create UIFigure and components createComponents(app)

% Register the app with App Designer

registerApp(app, app.BloodTransfussionClassificationUIFigure)

% Execute the startup function runStartupFcn(app, @startupFcn)

if nargout == 0 clear app

end end

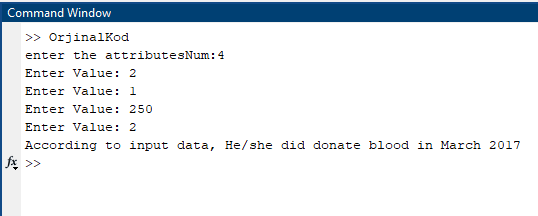
% Code that executes before app deletion function delete(app)

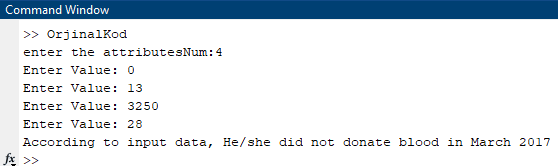
% Delete UIFigure when app is deleted delete(app.BloodTransfussionClassificationUIFigure)

end end

end

# Execution Outcome





# Evaluating Results

As a result of this project, if you enter the desired values. Recency (months since last donation), Frequency (total number of donation), Monetary (total blood donated in cc.), Time (months since first donation),

According to these input value this program will give whether he/she donated blood in March 2007.

So the result of this project fits the purpose of the project’s aim.