Topic	Confusion matrix	
Class Description	**Students learn about the confusion matrix also called the error matrix. They learn how to check the accuracy of their prediction model. They use the confusion matrix to witness the change in efficiency for logistic regression with one variable vs multiple variables.**	
Class	**C117**	
Class time	**45 mins**	
Goal	● Check for the accuracy of the prediction model in linear and multi linear regression using the confusion matrix.	
Resources Required	● Teacher Resources ○ Google Colab Notebook ○ Laptop with internet connectivity ○ Earphones with mic ○ Notebook and pen ● Student Resources ○ Google Colab notebook ○ Laptop with internet connectivity ○ Earphones with mic ○ Notebook and pen	
Class structure	**Warm Up** **Teacher-led Activity** **Student-led Activity** **Wrap up**	**5 mins** **15 min** **15 min** **5 min**

CONTEXT

- **Learn about the Confusion matrix**

Class Steps	Teacher Action	Student Action

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

Step 1: **Warm Up** **(5 mins)**	Hi <Student Name>! How are you doing today? Can you tell me what we learned in the last class?	**ESR:** - We learned about multilinear regression where the outcome is (yes or no) and it depends on two variables. - We wrote a multilinear algorithm to predict the outcome of the future events which might depend on the state of two variables.
	Wonderful! So till now we have created prediction models or classifiers and used them to predict the output. But how do we know how good the prediction model or machine learning algorithm we created is?	**ESR:** varied
	To check the efficiency of logistic regression, we use a confusion matrix. A confusion matrix is a table that is often used to describe the performance of a prediction or classification model (or "classifier") on a set of test data for which the true values are known. It allows the visualization of the performance of our prediction algorithm. Let's learn more about this.	-
Teacher Initiates Screen Share		

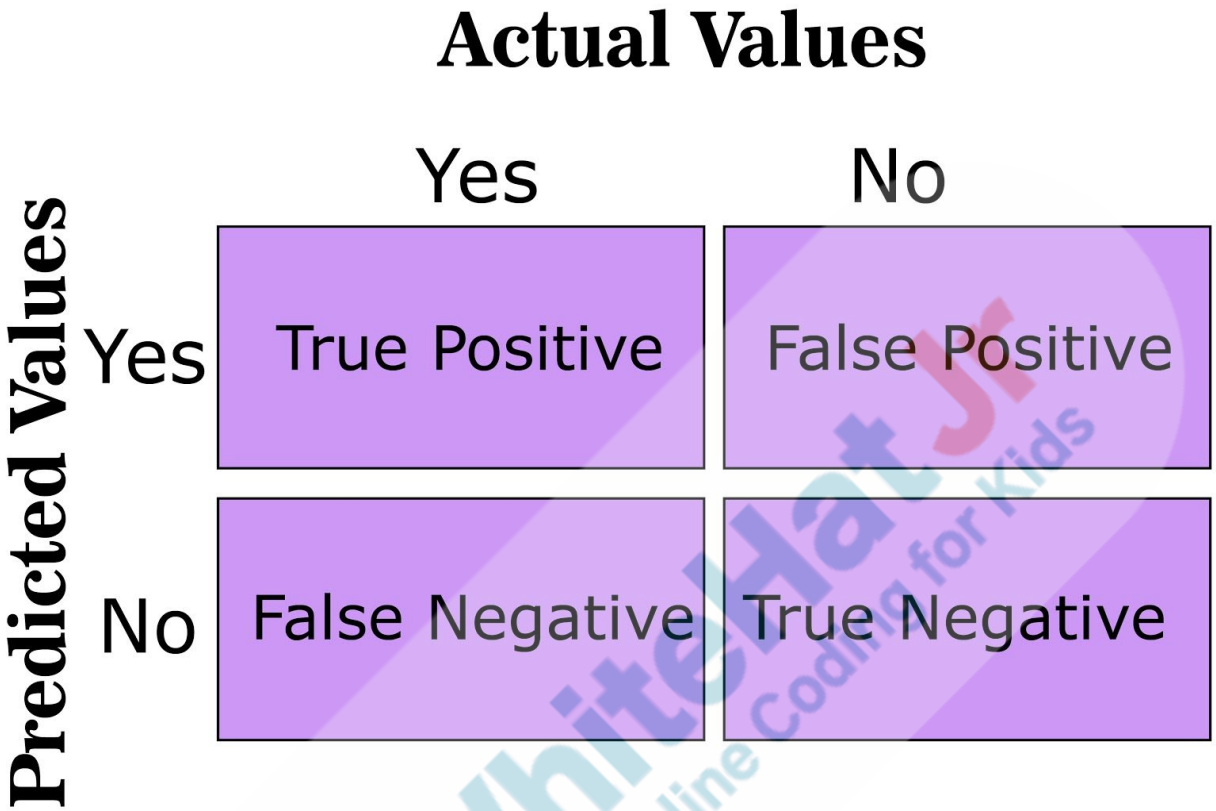
© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

CHALLENGE • **Learn to plot the data in a heat map and visually analyse the accuracy of the prediction model.** • **Find the accuracy of the model using code.**		
Step 2: Teacher-led Activity (15 min)	The confusion matrix is called confusion matrix or error matrix because it allows easy identification of confusion between classes e.g. one class is commonly mislabeled as the other. The number of correct and incorrect predictions are summarized with count values. Most performance measures are computed from the confusion matrix.	-
	Earlier we have studied the binary type of logistic regression, where the outcome is either True, or False. But the confusion matrix has the values as follows: 1. True Positives - The values that were actually True and were predicted to be True as well. 2. True Negatives - The values that were actually False and were predicted to be False as well. If all the data predicted by our algorithm lies in these two buckets, our ML algorithm is 100% accurate. However, that is rarely the case. There are two other buckets in which our predictions can lie: 3. False Positives - The values that were actually False but were	-

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

	predicted to be True. 4. False Negatives - The values that were actually True but were predicted to be False.	
	How can you know when a prediction model or classification model you have created is working fine?	**ESR:** When the model is making predictions with a great percentage of accuracy.
	Yes! The accuracy of a model is equal to: **(True Positives + True Negatives) / (True Positives + True Negatives + False Positives + False Negatives)**	-

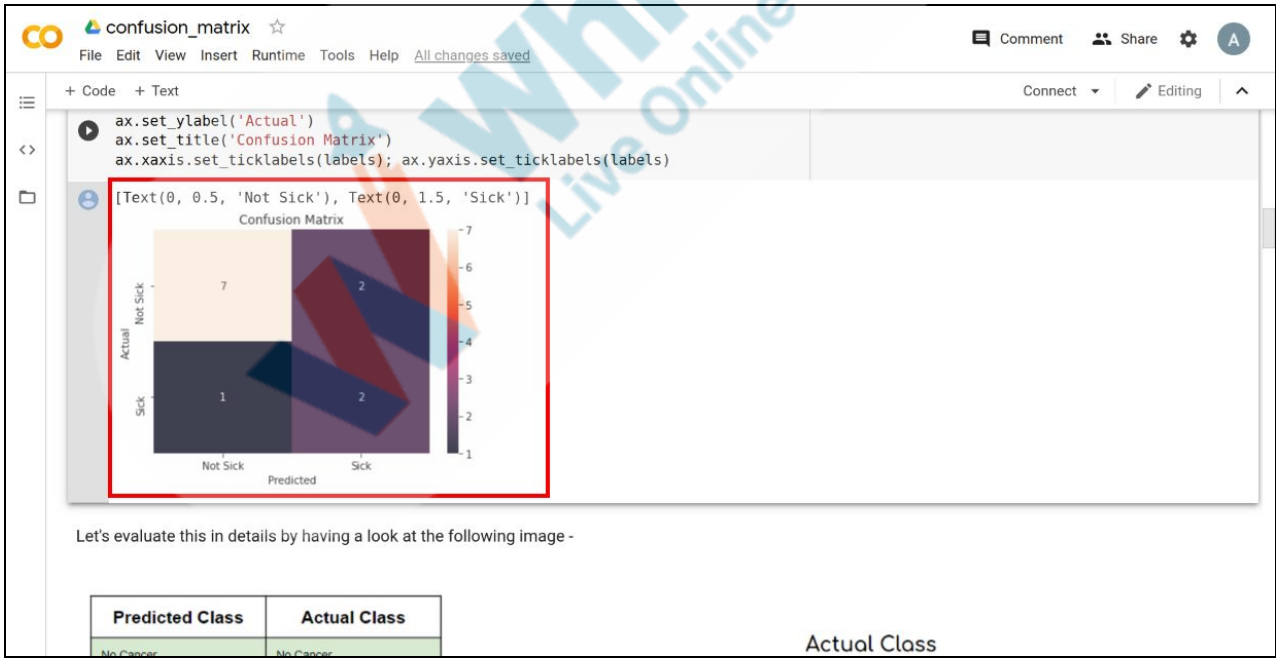
© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



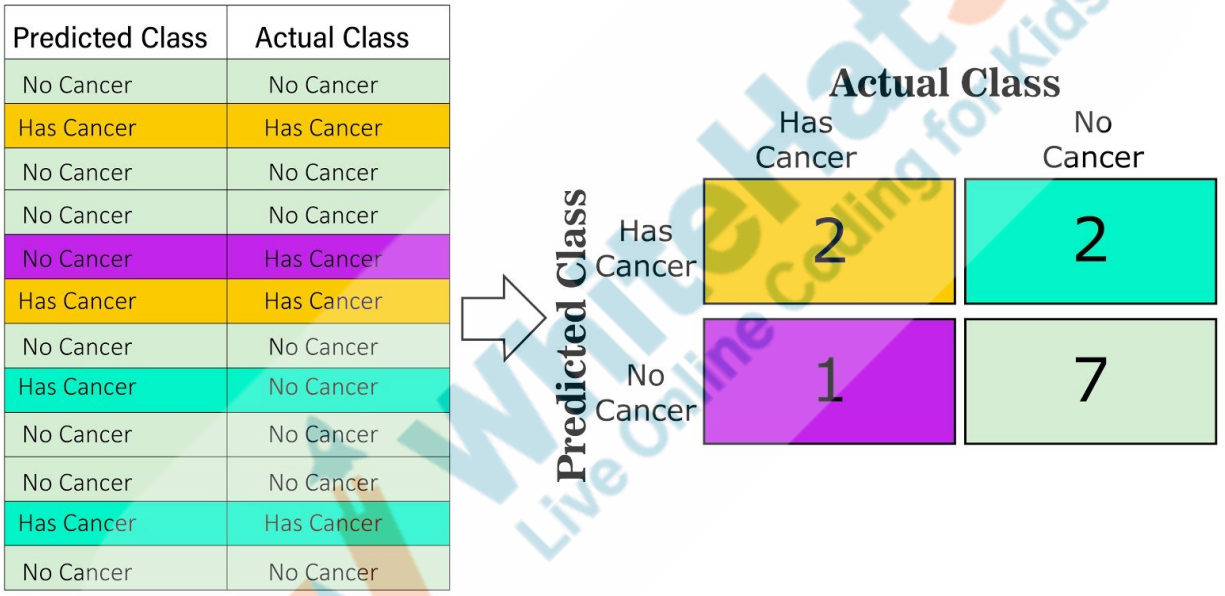
	Alright now let's see how the confusion matrix can be formed. *<Teacher opens the Google Colab notebook from **Teacher Activity 1**>* *Teacher imports the **confusion_matrix** function from sklearn.metrics.* This function takes the actual data and the predicted data as parameters along with labels if available. *Teacher also **imports seaborn as sns** and **imports matplotlib.pyplot***	*The student observes and learns.*

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

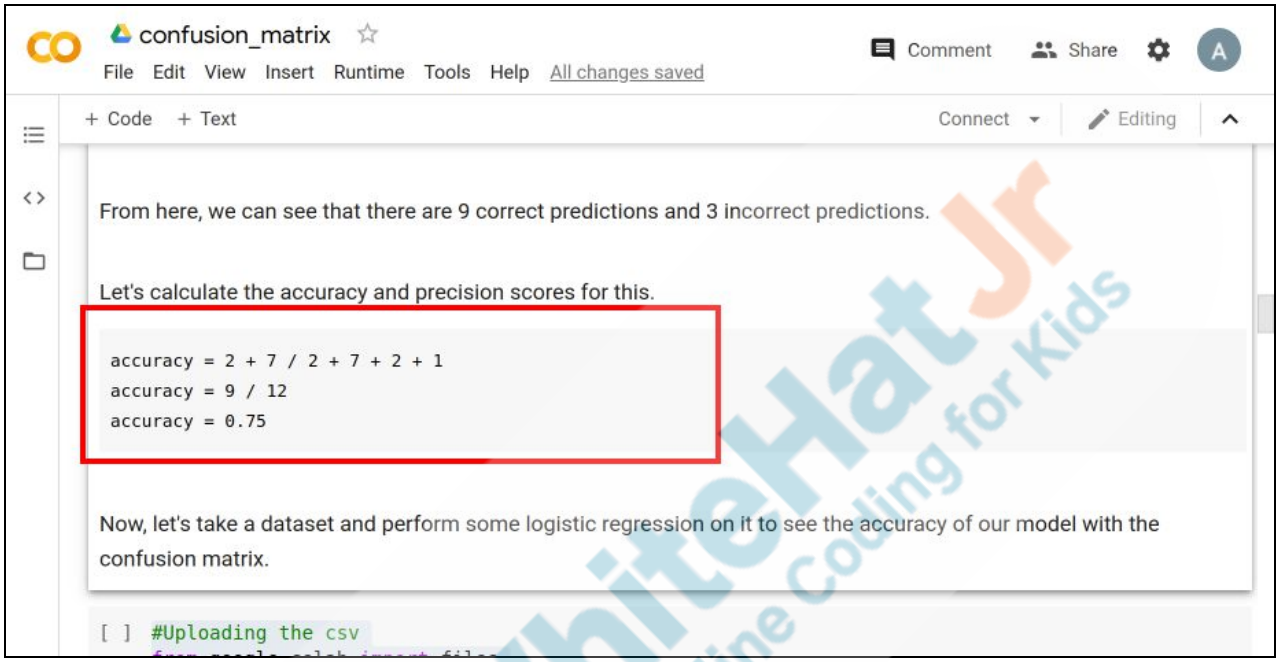
	as plt. *Teacher creates 2 different data frames for* ***actual_data*** *of* ***sick*** *and* ***not sick*** *people and* ***predicted data*** *of sick and not sick people and a* ***labels*** *list which will have the* ***labels sick*** *and* ***not sick****.* Then using the **confusion_matrix** function and passing **actual_data**, **predicted_data** and **labels** as parameters store it in variable cm. Now we are going to plot this data on a heat map. Heat map plots rectangular data as a color-encoded matrix also adds labels using the plt.subplot().	



© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

	What do we see here? Yes! lets evaluate it more by taking a look at the image. *<Teacher opens the image from the Teacher Activity 2>* What can you make out from the image?	**ESR:** We see a rectangle plot with small rectangles in it with different numbers in them. **ESR:** We can see that there are 9 correct predictions and 3 incorrect predictions.
	 	
	Yes! Now let's calculate the accuracy of this model. We have seen the formula earlier: True Positives + True Negatives / True Positives + True Negatives + False Positives + False Negatives Using this we'll calculate the accuracy. accuracy = 2 + 7 / 2 + 7 + 2 + 1	-

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

	accuracy = 9 / 12 accuracy = 0.75 The accuracy we got is 0.75.	
		
	Now, let's take a dataset and perform some logistic regression on it to see the accuracy of our model with the confusion matrix. *Teacher downloads the data from Teacher Activity 3 and uploads the file in the Colab notebook.* Code for reference:- **#Uploading the csv** **from google.colab import files** **data_to_load = files.upload()**	-

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

	Let's print the data. **import pandas as pd** **df = pd.read_csv("heart.csv")** **print(df.head())**	
		
	Now from this data we are going to use the age and target. We are going to split this data into 75% and 25% to train our prediction model and then test it. *Teacher codes to split the data into 75% and 25%.* Code for reference:- **from sklearn.model_selection import train_test_split** **age = df["age"]** **heart_attack = df["target"]**	*The student helps the teacher to split the data.*

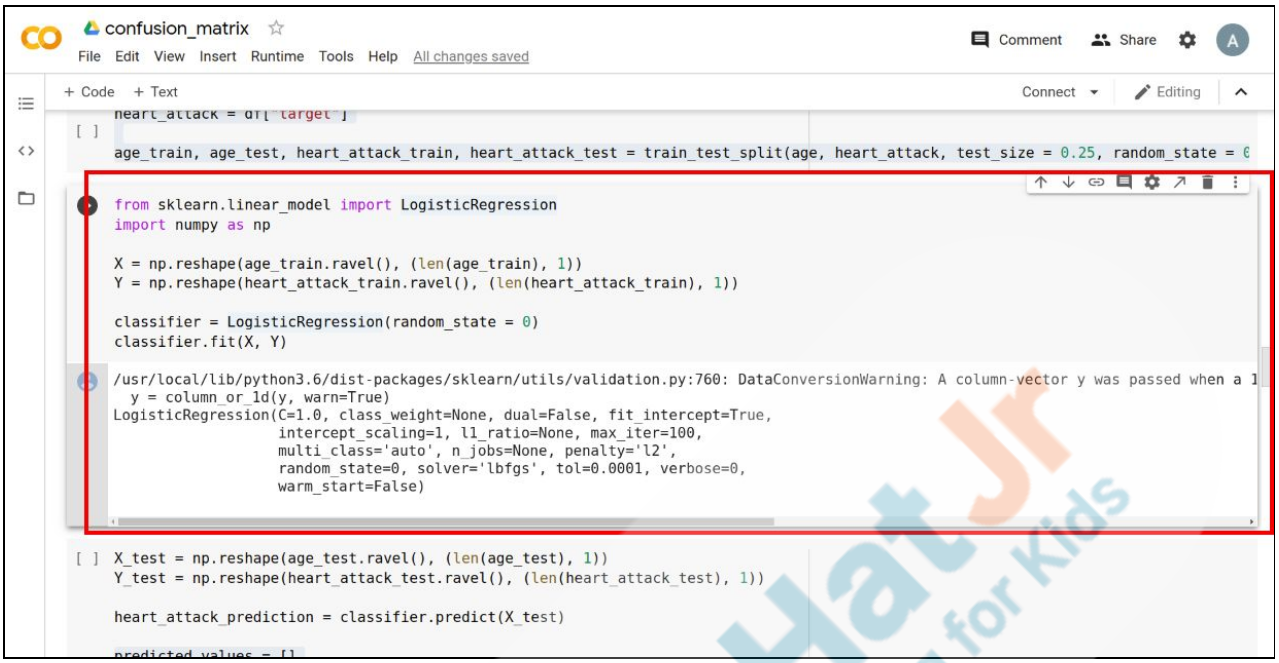
© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

	age_train, age_test, heart_attack_train, heart_attack_test = train_test_split(age, heart_attack, test_size = 0.25, random_state = 0) Here, we are taking the columns of **age and target** and storing them in variables. We are then using the **train_test_split()** function to split these columns for training and testing purposes in ratio of 75% and 25% respectively.	
		
	Now we'll train our prediction model on the data. *Teacher codes to train the prediction model.* Code for reference:	*The student helps the teacher to train the prediction model.*

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

	from sklearn.linear_model import LogisticRegression **import numpy as np** **X = np.reshape(age_train.ravel(), (len(age_train), 1))** **Y = np.reshape(heart_attack_train.ravel(), (len(heart_attack_train), 1))** **classifier = LogisticRegression(random_state = 0)** **classifier.fit(X, Y)** Here, we are first creating the list of training data using the reshape() function. reshape() function takes 2 arguments that are compulsory. The first one is the list that needs to be reshaped and the second on is the new shape. Here, we are first converting the list of lists that the age_train and heart_attack_train have into simple lists, and then we are reshaping it into an array with just one row and n-number of columns (same as the total number of data points) We are then creating a LogisticRegression classifier and fitting our reshaped arrays into this to find a relation and make predictions!	

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

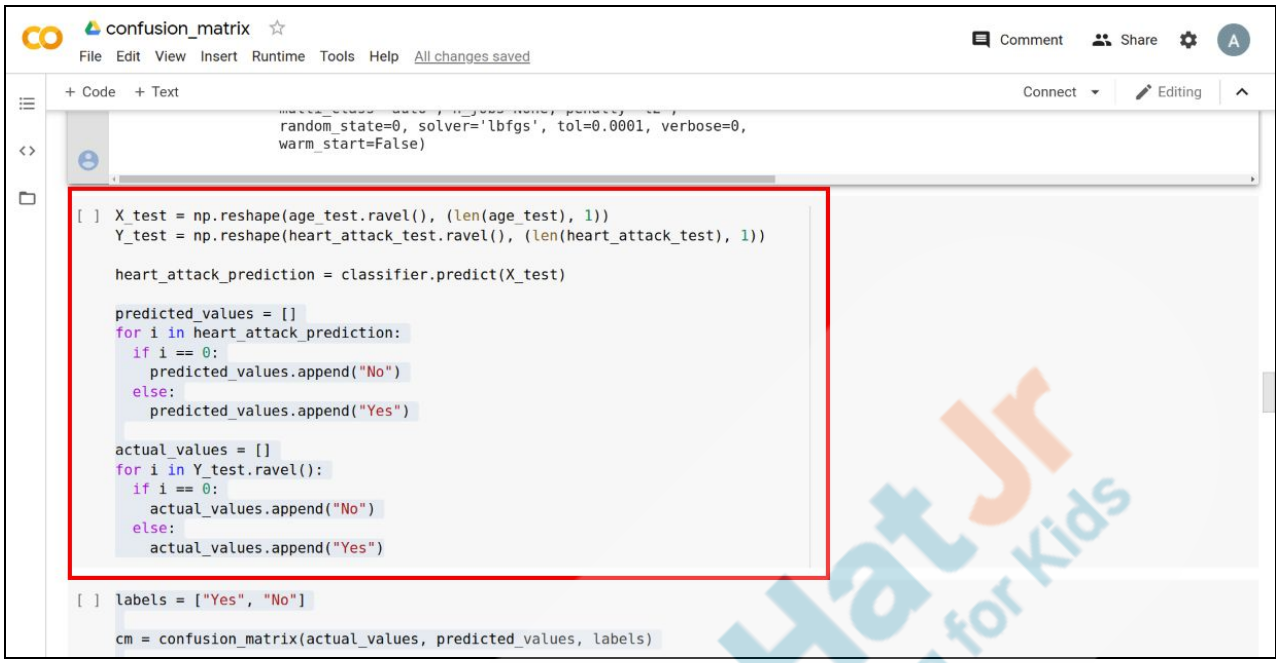


	Now that we have trained the model we'll test it using the test data. *Teacher codes to test the prediction.* Code for the reference:- **X_test = np.reshape(age_test.ravel(), (len(age_test), 1))** **Y_test = np.reshape(heart_attack_test.ravel(), (len(heart_attack_test), 1))** **heart_attack_prediction = classifier.predict(X_test)** Here again, we are repeating what we did earlier with the training data, this time for the testing data and finally we are using our classifier that we created to make predictions on the	-

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

	testing data, whether the patient will get a heart attack or not. In the result, we'll substitute the value of 0 as No and 1 as Yes and create 2 arrays for the values in heart attack predictions and actual_values and create a labels array with values Yes and No. Code:- **predicted_values = []** **for i in heart_attack_prediction:** **if i == 0:** **predicted_values.append("No")** **else:** **predicted_values.append("Yes")** **actual_values = []** **for i in Y_test.ravel():** **if i == 0:** **actual_values.append("No")** **else:** **actual_values.append("Yes")** Here, we are creating the lists of actual values and predicted values on whether the patient would get a heart attack or not.	

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



Now using the confusion_matrix function we'll plot this data on the heat map.

Teacher codes to plot the data on the heat map.

Code for reference:-

labels = ["Yes", "No"]

cm = confusion_matrix(actual_values, predicted_values, labels)

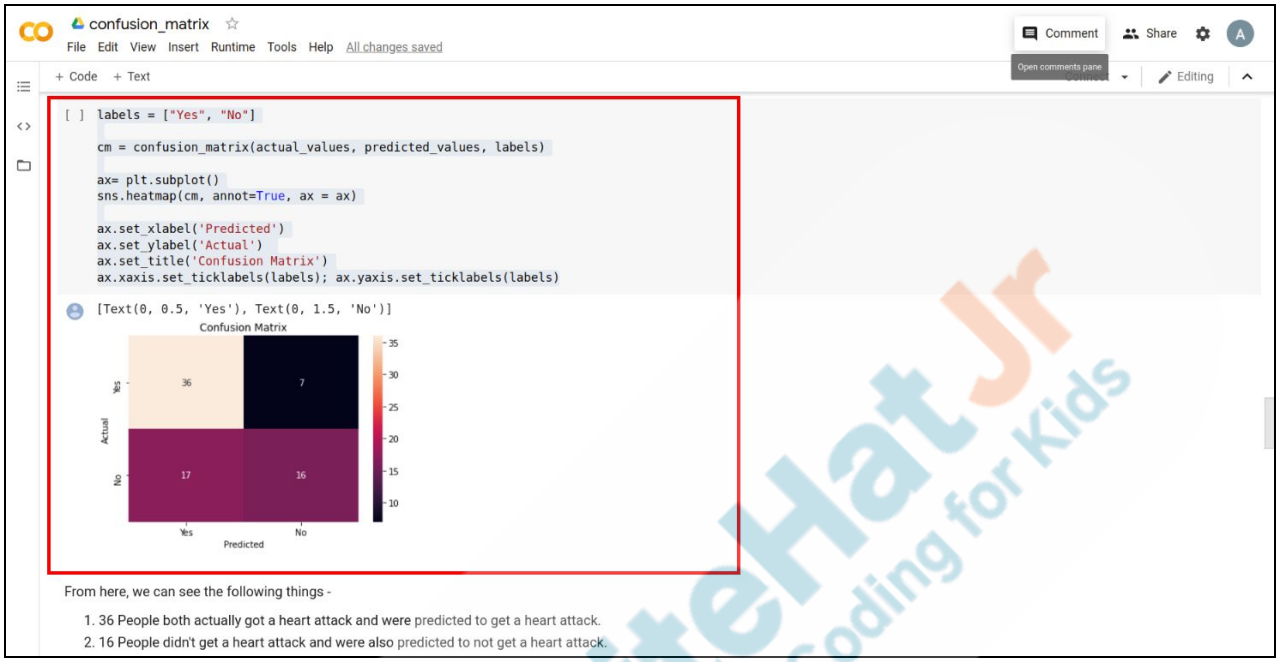
ax= plt.subplot()
sns.heatmap(cm, annot=True, ax = ax)

ax.set_xlabel('Predicted')
ax.set_ylabel('Actual')
ax.set_title('Confusion Matrix')

-

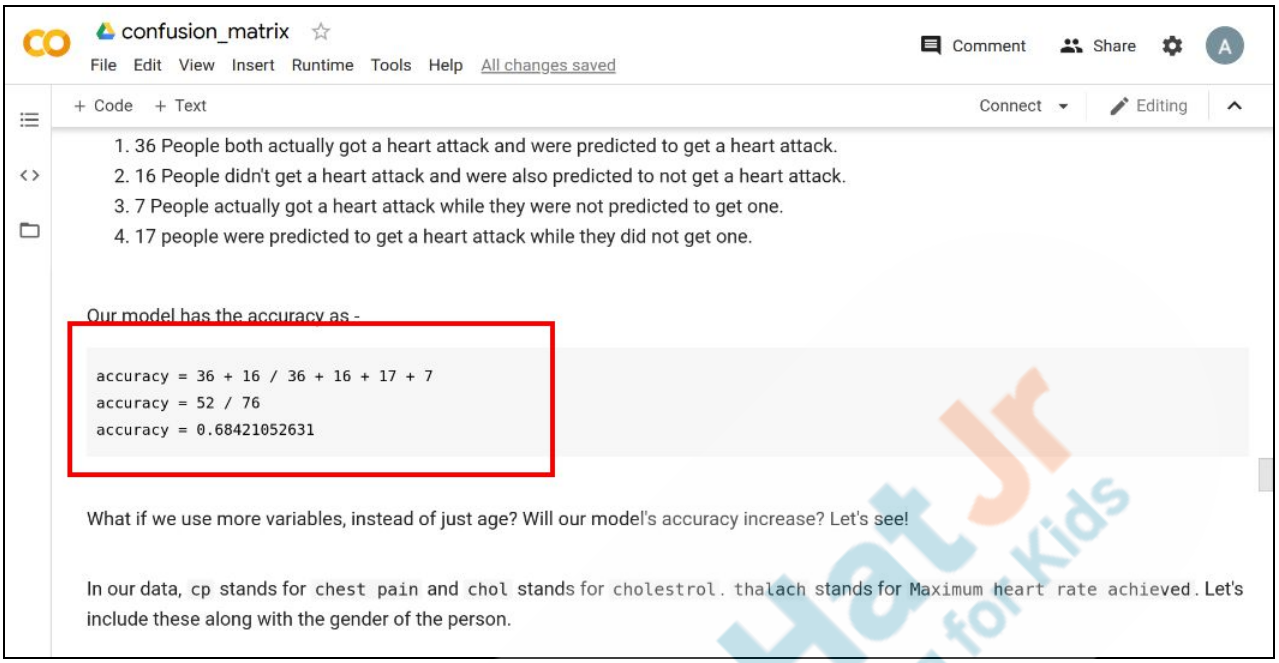
© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

	ax.xaxis.set_ticklabels(labels); **ax.yaxis.set_ticklabels(labels)**	



	What can we see from this plot? Alright now let's calculate the accuracy of the model. *Teacher codes to calculate the accuracy.* Code for reference:- accuracy = 36 + 16 / 36 + 16 + 17 + 7 accuracy = 52 / 76 accuracy = 0.68421052631	**ESR:** We can see that 36 People got heart attack and were actually predicted to get an heart attack 16 People didn't get a heart attack and were also predicted to not get a heart attack. 7 People actually got a heart attack while they were not predicted to get one. 17 people were predicted to get a heart attack while they did not get one.

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



	There is one more thing that we have to check and that is if we use multiple variables will the accuracy change or not. Can you try doing that?	**ESR:** Yes
	Let's begin then.	
Teacher Stops Screen Share		
	Now it's your turn. Please share your screen with me.	

- **Ask Student to press ESC key to come back to panel**
- **Guide Student to start Screen Share**
- **Teacher gets into Fullscreen**

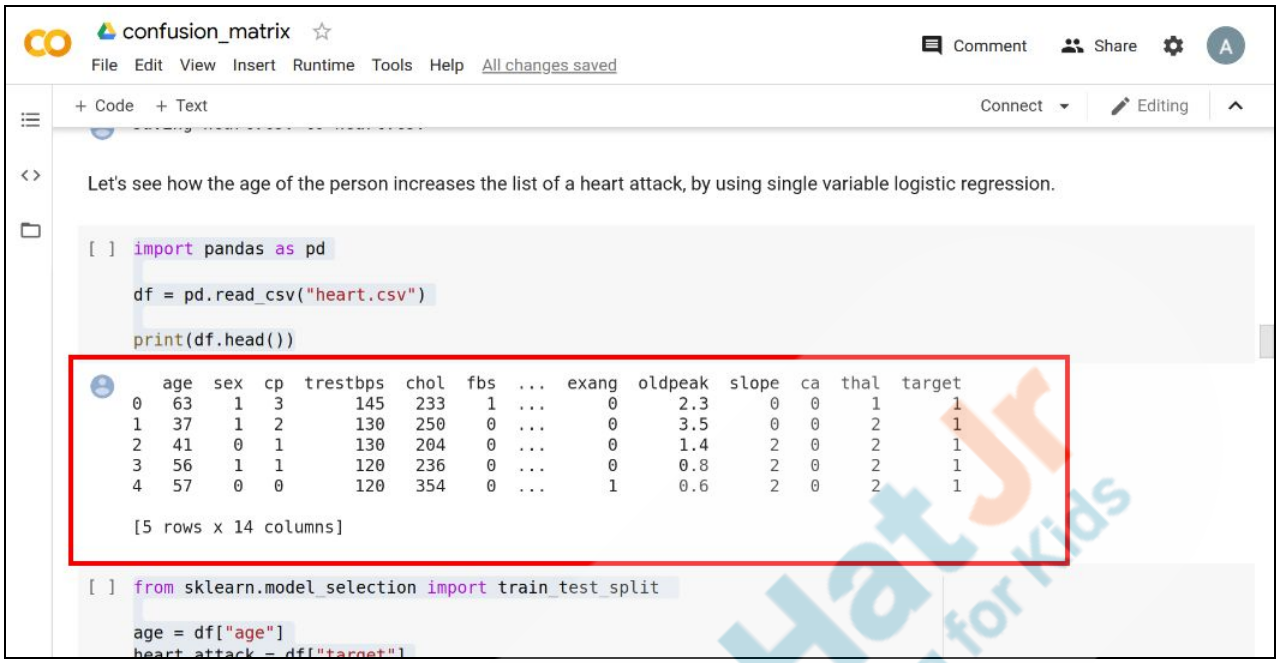
ACTIVITY

- **Plot the data on a heat map and visually analyze the accuracy of the prediction model of multilinear regression**
- **Check the difference in the accuracy of linear and multilinear regression**

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

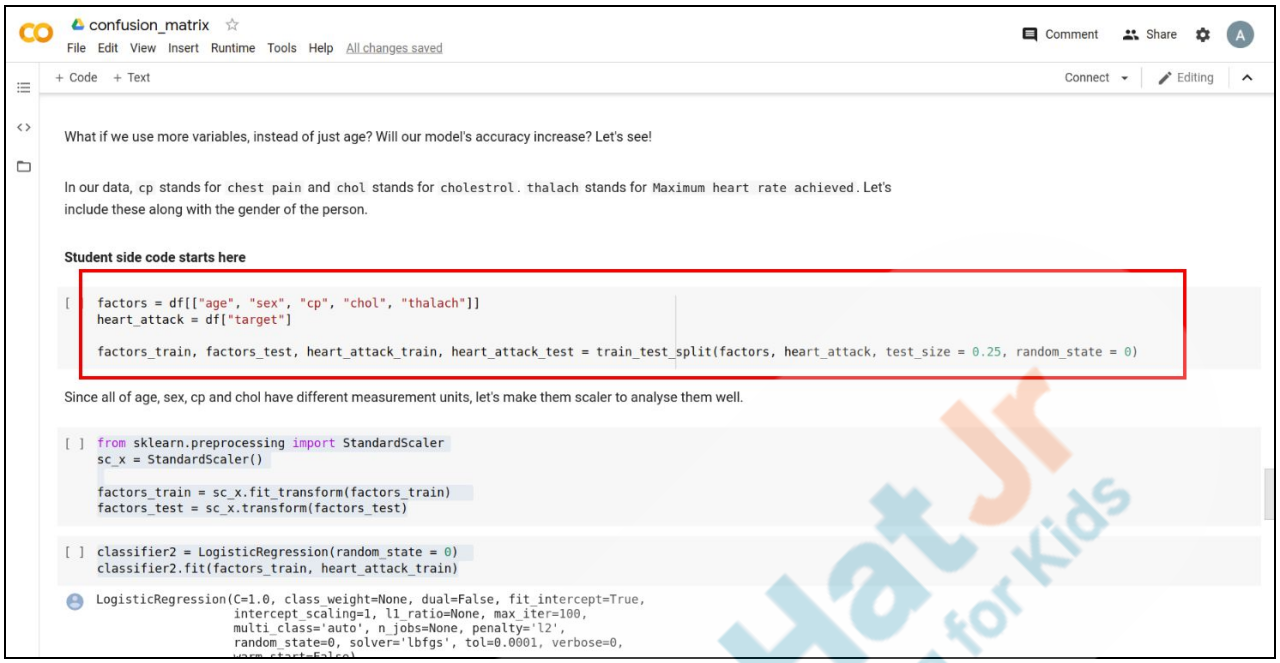
Step 3: Student-Led Activity (15 min)	*Teacher helps the student to download the data and open a new google colab notebook.*	*Student downloads the data from Student Activity 1* *Student opens the Google colab notebook from Student Activity 2.*
	Teacher helps student to import all the useful libraries: Code for reference:- **from sklearn.metrics import confusion_matrix** **import seaborn as sns** **import matplotlib.pyplot as plt**	*Student uploads the data in the notebook.* *And imports the useful libraries in the notebook like Sklearn, pandas, matplotlib.*
	As we know that our data contains multiple factors such as age, gender, cp (chest pain), cholesterol, thalach (maximum heart rate achieved). And we are going to be making use of all these factors.	-

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



	Teacher helps the student create data of all the factors and heart attack, and split it into 75% and 25%.	*Student codes to create 2 data frames of all the factors and heart attack and then split this data into 75% and 25%.*

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



	As we can see that our factors have different measurements so we need to make them scalar for our analysis. *Teacher helps the student make the data scalar.* Code for reference: **from sklearn.preprocessing import StandardScaler** **sc_x = StandardScaler()** **factors_train = sc_x.fit_transform(factors_train)** **factors_test = sc_x.transform(factors_test)**	*Student codes to make the data scalar.*

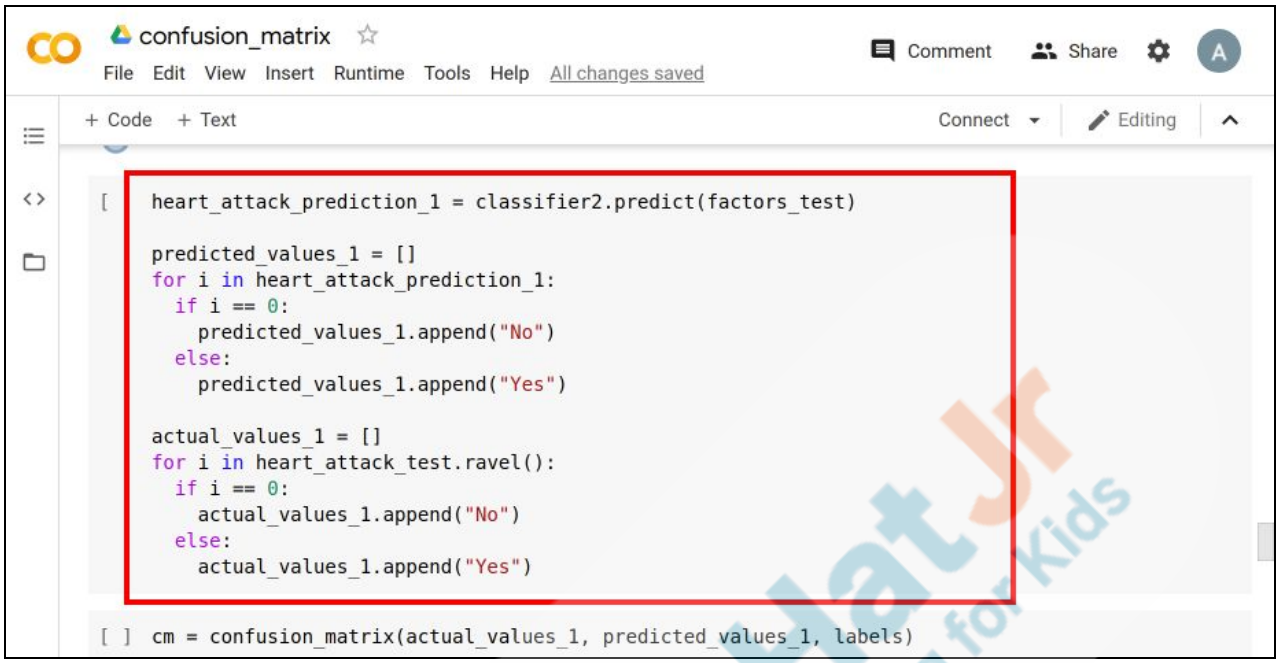
© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

	We need to train the model on the 75% data that we separated. *Teacher helps the student to train the model on the separated data.* Code for reference:- **classifier2 = LogisticRegression(random_state = 0)** **classifier2.fit(factors_train, heart_attack_train)**	*Student codes to train the model on the 75% data that was separated.*

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

	Teacher helps the student to code to make predictions and substitute the value of 0 as No and 1 as Yes and create 2 arrays for the values in heart attack predictions and actual_values. and create a labels array with values Yes and No.	*Student codes to make a prediction and then substitute the value of 0 as No and 1 as Yes and create 2 arrays for the values in heart_attack_predictions_1 and actual_values_1 and create a labels array with values Yes and No.*

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.



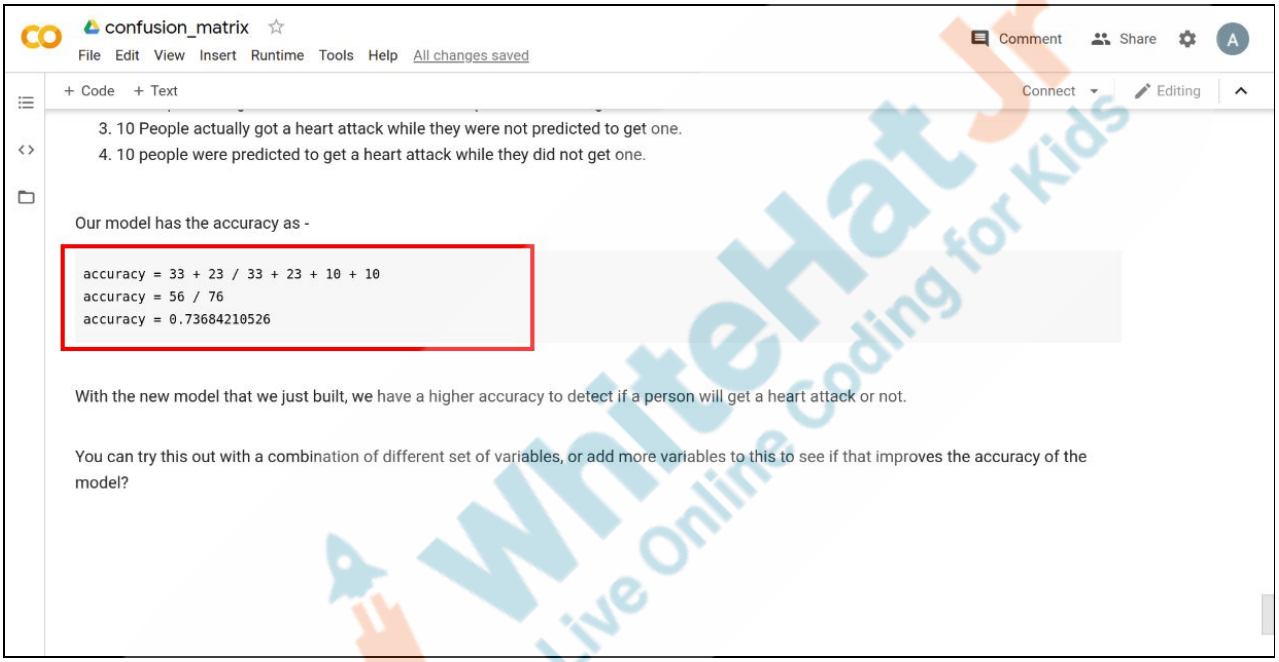
	Teacher helps the student to plot the data on the heat map.	*Using the confusion_matrix function student plots the data on the heat map.*

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

	What can we make out from this plot?	**ESR:** 33 People actually got a heart attack and were also predicted to get a heart attack. 23 People didn't get a heart attack and were also predicted to not get a heart attack. 10 People actually got a heart attack while they were not predicted to get one. 10 people were predicted to get a heart attack while they did not get one.

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

	Let's check for accuracy. *Teacher helps the student to check for accuracy.* Code for reference: accuracy = 33 + 23 / 33 + 23 + 10 + 10 accuracy = 56 / 76 accuracy = 0.73684210526	*Student codes to check the accuracy.*



	Our accuracy rate has increased. So we can conclude that with the new model that we just built, we have a higher accuracy to detect if a person will get a heart attack or not.	

Teacher Guides Student to Stop Screen Share

FEEDBACK

- **Appreciate the student for their efforts**
- **Identify 2 strengths and 1 area of progress for the student**

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

Step 4: Wrap-Up (5 min)	So let's quickly go over what we did in today's class.	**ESR:** - We learned about the confusion matrix. - We saw how to create a confusion matrix and check for the accuracy in linear regression and multilinear regression.
	Very good. You must have wondered how a phone can detect a gesture or an AI detect an object. All this is done using clustering. In our upcoming classes we'll learn about clustering.	
Teacher Clicks 		
Additional Activities	*Encourage the student to write reflection notes in their reflection journal using markdown.* Use these as guiding questions: ● What happened today? - Describe what happened - Code I wrote ● How did I feel after the class? ● What have I learned about programming and developing games?	*The student uses the markdown editor to write her/his reflection in a reflection journal.*

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

	• What aspects of the class helped me? What did I find difficult?	
Project Overview	## Confusion Matrix **Goal of the Project:** In this project you will apply what you learned in the class and write an algorithm to create a confusion matrix In this project you will apply what you learned in the class and create your own prediction model. **Story:** Working as a bank official you have asked your team to make a Machine learning model to find out whether a note is fake or real.Your team created an AI model, but how would you know if the model is working properly or not, well you can plot a Confusion matrix to check, how well the model is performing, you have the data, make a model and plot a confusion matrix to find the accuracy of the model. I am very excited to see your project solution and I know you will do really well. Bye Bye!	

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.

Activity	Activity Name	Links
Teacher Activity 1	Google Colab Notebook	https://colab.research.google.com/notebooks/intro.ipynb#recent=true
Teacher Activity 2	Confusion matrix image	https://miro.medium.com/max/1000/1*TWXtKH_4trfKz7sexoadiw.png
Teacher Activity 3	data file	https://raw.githubusercontent.com/whitehatjr/datasets/master/c117/heart.csv
Teacher Activity 4	Teacher reference link	https://colab.research.google.com/drive/1Z4S6QGYh3JEzzN2AxG7SxeIgWqMhaOLq?usp=sharing
Student Activity 1	data file	https://raw.githubusercontent.com/whitehatjr/datasets/master/c117/heart.csv
Student Activity 2	Google Colab Notebook	https://colab.research.google.com/notebooks/intro.ipynb#recent=true

© 2020 - WhiteHat Education Technology Private Limited.
Note: This document is the original copyright of WhiteHat Education Technology Private Limited.
Please don't share, download or copy this file without permission.