
SEMINARS PROJECT

Matteo Silvestri

ID: 1774987

Master in Engineering in Computer Science
`silvestri.1774987@studenti.uniroma1.it`

August 26, 2020

1 Introduction

First of all we need to explain the environment where this project is developed, is a project created in order to predict a final score of a precisely final survey only with the score of the first survey, this survey is the same both the time, but the first one is made during the first meeting the other one is made during the last one, the answers of this survey are number between 1 and 4, and the name of this kind of survey is SCL 90, this project is developed with the idea to help the doctor to have a predicted estimation of the score in the final survey knowing only the answers of the first one, in order to do this i used a regression algorithm.

2 Dataset Info

The original dataset is composed with two different tuples for every patient(one for the first survey and one for the last), in the columns the most useful data are the answers to the question of the survey(90) and the final score of it, so in order to execute the script we need to manipulate the data in order to have a better representation of these data.

2.1 Preprocessing

In order to use the dataset in a better way we needed to start from the **TestSCLSF36BESTERESA.xlsx** file and commute it in a CSV file(**TERESATraining.csv**), after this operation through Pandas(python framework for handle the csv file) i take only the odd tuples, representing the answers of all the tuples of the first survey and the final score of the last survey, in this way:

```
df.drop(df.index[[1,3,5,7,9,11,13,15,17,19,21,23,25,27,29]], inplace=True)

for index, label in dataset["Total Score"].iteritems():
    if( index%2 != 0):
        y.append(label)
```

Figure 1: Take only the odd tuples and save the final score of the final survey

3 Regression

I choose to solve this situation with a Regression algorithm, in this kind of problem we have an output that could be real or continuous(in our case the prediction of the finale score of the last survey knowing only the answers to the first survey), so in order to realize this idea in a more practical way i used the Sklearn framework of python, that help with the training and the test of the dataset, in this way:

```
X = df[["1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15", "16", "17", "18", "19", "20", "21",
Y = df["Total Score"].values

X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=1, random_state=0)

regressor = LinearRegression()
regressor.fit(X_train, Y_train)
```

Figure 2: Training and Test of the Regression

After this training we can use the script in order to predict the final score of a patient, and i take the input from the GUI that i will explain later. Obviously i use Sklearn also for the prediction:

```
print(inp.split())
X_ins = np.array(list(inp.split()), dtype=int)
print(X_ins.reshape(1, -1))
print(X_test)
Y_pred = regressor.predict(X_ins.reshape(1, -1))
df = pd.DataFrame({'Predicted': Y_pred.flatten()})
print(df)
return df
```

Figure 3: Prediction of the Regression with the input of the GUI

4 Insert the Answer

In order to help the the user to have a easier experience with this script i wanted to use a GUI for insert the data and to show the output, so in this way the user can only copy and past all the answers of the survey directly in the textbox and see the prediction, in the following i show the GUI:

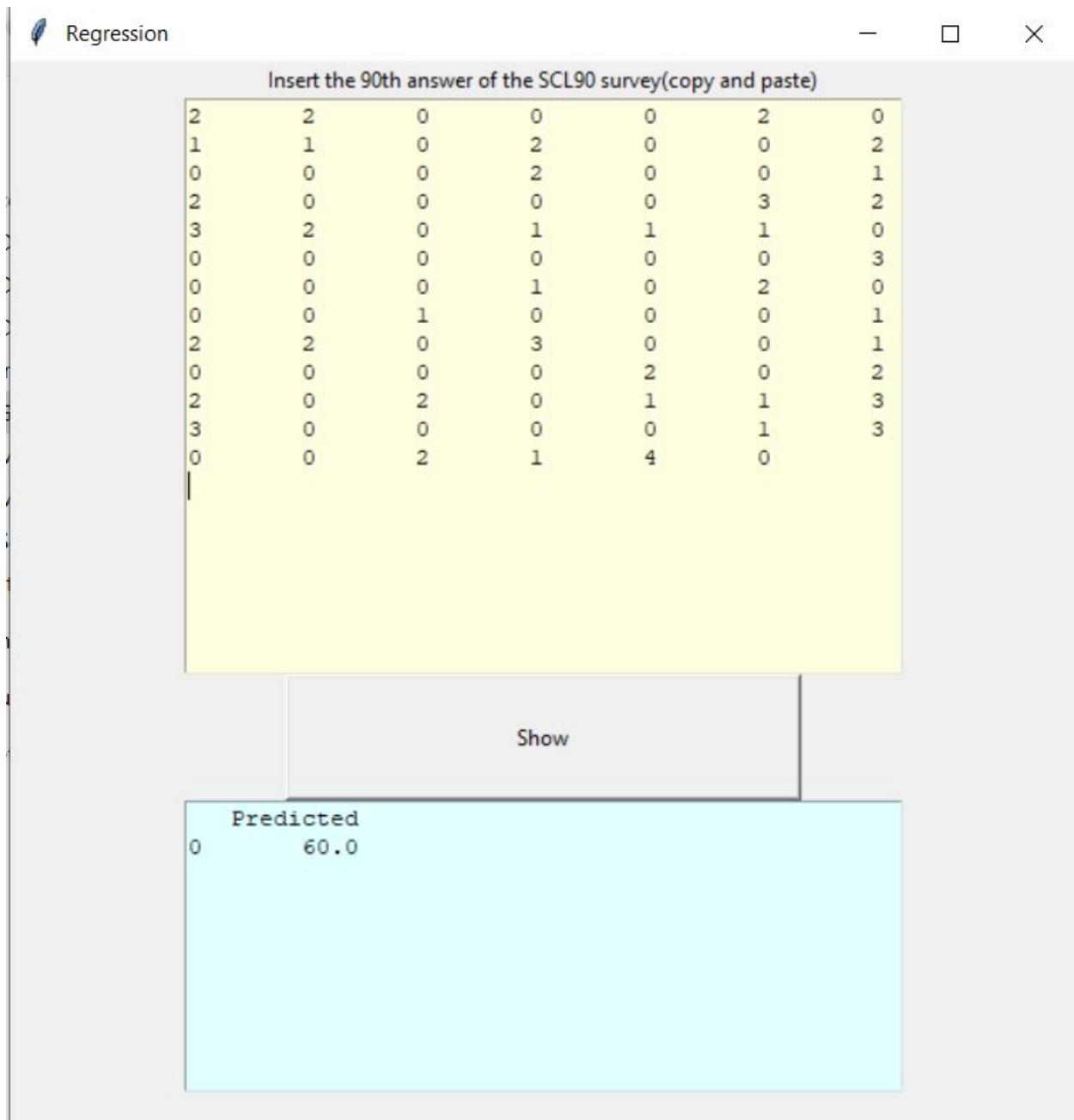


Figure 4: GUI where the user can copy and paste the answers of the first survey

The framework of python that i used for this GUI is tkinter, and at follow we can see how:

```

def Take_input():
    INPUT = inputtxt.get("1.0", "end-1c")
    out = Regre_Sol(INPUT)
    Output.insert(END, out)
    #print(INPUT)

l = Label(text = "Insert the 90th answer of the SCL90 survey(copy and paste)")
inputtxt = Text(root, height = 10*2,
                width = 25*2,
                bg = "light yellow")

Output = Text(root, height = 5*2,
              width = 25*2,
              bg = "light cyan")

Display = Button(root, height = 2*2,
                 width = 20*2,
                 text = "Show",
                 command = lambda:Take_input())

```

Figure 5: Code in order to have the GUI

5 Launch It

In order to launch the script we need to do:

- Open the terminal and execute this command "python.exe regre.py"
- Fill the text are with the value of the SCL90 survey answers(copy and paste)
- Click the show button, and saw the prediction