Arjun Koshal

CS 110 - A

Professor Ryan

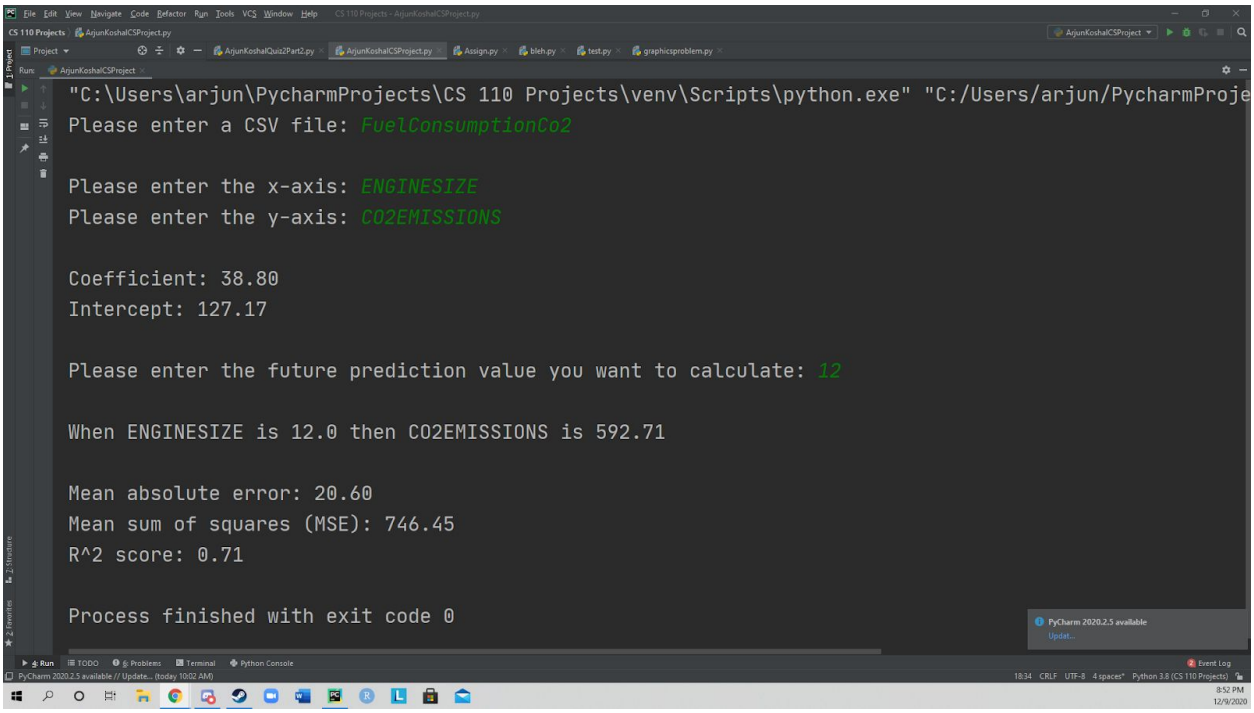
December 13, 2020

Final Project Report

Section 1: "Overview and Summary of Project."

The purpose of the program that I designed primarily attempts to figure out exactly what the least squares regression line is. Based on a csv file that the user inputs, I wanted to take all of the points and plot all of them, using matplotlib.pyplot. The program first utilizes the pandas library in order to read the csv file. If the file does not have .csv at the end, the program adds the .csv at the end of the file. Once the file is read, the user is prompted to input the x axis and the y axis. In order for the program to run properly, the user must confirm that the x and y axis exist and the alignment is proper, or else the user will encounter an error. Once the user enters the information, a data set is generated which requires training and testing. 80% of the data was trained and the rest was tested, and each axis was transformed into an array. The arrays were assigned a new variable and from sklearn, I imported a linear regression model. The package from sklearn allowed me to display the coefficient and intercept for the data on the graph. Once that was given, the user was able to see the line of best fit along with the data they inputted. Once the line of best fit was shown, the user was given the opportunity to make a prediction based on the given x value that the user stated. Finally, the program displays the mean absolute error, mean sum of squares, and the r^2 score, which determines the accuracy of the model.

Pictures


"C:\Users\arjun\PycharmProjects\CS 110 Projects\venv\Scripts\python.exe" "C:/Users/arjun/PycharmProje
Please enter a CSV file: FuelConsumptionCo2
Please enter the x-axis: ENGINESIZE
Please enter the y-axis: CO2EMISSIONS
Coefficient: 38.80
Intercept: 127.17
Please enter the future prediction value you want to calculate: 12
When ENGINESIZE is 12.0 then CO2EMISSIONS is 592.71
Mean absolute error: 20.60
Mean sum of squares (MSE): 746.45
R^2 score: 0.71
Process finished with exit code 0

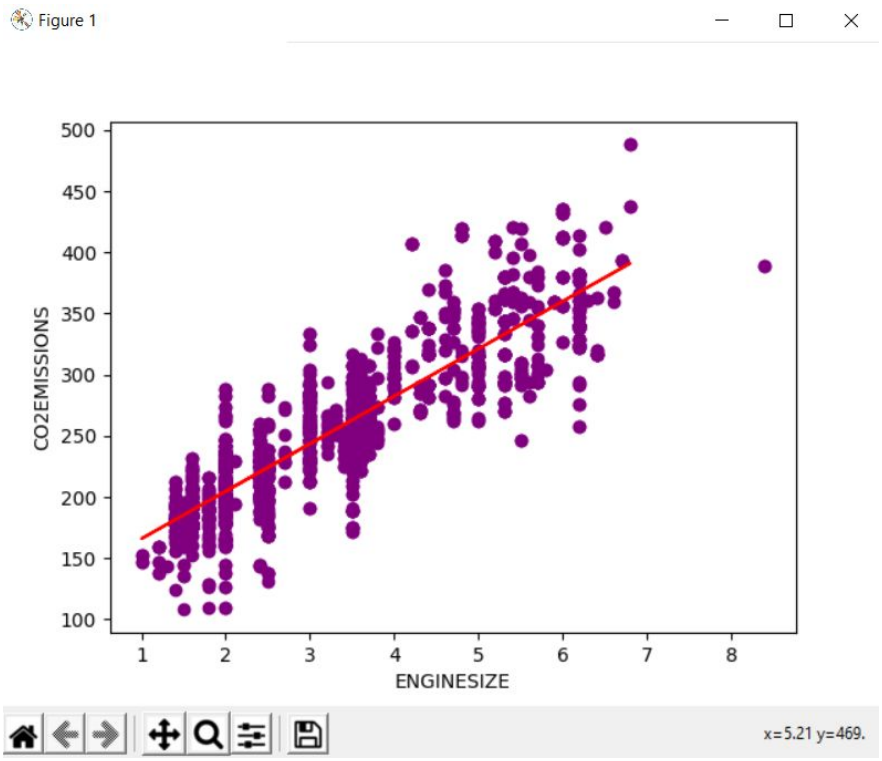


Figure 1
CO2EMISSIONS
ENGINESIZE
x=5.21 y=469.

Section 2: "Target Audience."

The target audience for my program would definitely be high school and college students enrolled in a calculus, linear algebra, and statistics course. Most students in high school would definitely find a linear regression model helpful when plotting data points for a science course and needing to find the accuracy of their data. In my physics and chemistry course, we dealt a lot with given data points and viewing the line of best fit in Excel. This program would have been extremely useful in plotting the line of best fit and identifying the correlation between the independent and dependent variables, along with the distance from the line to the points.

Section 3: "Specific Programming Techniques Used."

In the program, I utilized four different libraries: pandas, numpy, matplotlib.pyplot, and sklearn. Pandas was necessary in order to read the csv file; numpy was necessary in order to turn the axes into arrays, matplotlib.pyplot was necessary in order to how the visualization of the plotted data, and sklearn was used to import the linear regression in order to display it on the screen. The main file that the data deals with is the file that the user inputs it, as long as it is a csv file and the data is properly aligned. When using numpy, I decided to use an array to make sure that the trained data would be ordered properly and assigned to a new variable. This helped keep the program organized and helped ensure that I wouldn't use the untrained data set. One function I defined was the prediction function, which allowed me to use the equation $y = mx + b$ in order to calculate the future dependent variable based on the user's input of the independent variable. The data was collected based on the user's input of the csv file, and the x and y axes. Based on this given information, the program was able to generate a graph with the line of best fit, along with the individual points.

Section 4: “Challenges.”

There were several issues that I encountered when running the program. First of all, I had trouble figuring out the step between entering the axes and generating a dataset from the information inputted by the user. In order to solve this challenge, I had to create a new variable called data, which would allow the current axes to be transformed into a dataset. The next obstacle I encountered related to training and testing the data. I was not certain on what percent of the data should be trained and tested, therefore I utilized the internet in order to find where the most significant statistical evidence appears, which is 80-20. I used the len function in order to take the length of the data and trained 80% of it and leave the rest for testing. My final challenge was dealing with sets of arrays. I had previous knowledge of arrays from past languages, therefore incorporating them was straightforward. I had difficulty figuring out exactly where the arrays should be placed and the exact function for them. Once I figured out that I could easily assign a variable to each different array, in order to preserve the previous variable, I solved the issue. These were the only key challenges that I was faced with throughout the duration of the project.

Section 5: “Future Extensions.”

If there was more time to work on this project, I would try to implement different regressions, such as polynomial, exponential, and logarithm. At the start of the project, I intended to allow the user to choose which regression they wanted to see, however, I felt that I was overwhelmed with school work, therefore I decided to keep the project short and simple. I would also allow the user to compare different data sets, along with the ability for the user to input their own data points. There are also settings where a program has access to its memory, where the user could access previous entries. This would allow the user to compare past data

with new data. All of these additional features would have been a nice touch to the project if I had more time to work on the program.