** Name:Adnan Akram**

# Assignment no. 1: Course: AI( LAB)

**Instructor: Sir Aleem Raheem**

**Sap id: 25617**

**Name:Umair Ghani**

**Sapid: 8625**

1. **Obtain the dataset from [ https://**[**www.hackerearth.com/practice/machine-**](http://www.hackerearth.com/practice/machine-) **learning/machine-learning algorithms/beginners-guide-regression-analysis-plot- interpretations/practice-problems/ ].**

## Problem: Predict the employee burn out rate.

1. **Describe the problem in your own words**

The purpose of the regression problem of estimating the employee burnout rate is to determine the degree to which an employee is suffering tiredness, cynicism, and diminished professional efficacy as a result of ongoing work stress. Burnout can negatively impact an employee's health, wellbeing, and performance as well as the productivity, and reputation of the company, making it a significant issue. To address this issue, one requires a dataset with the characteristics and burnout rate of a representative sample of workers, as well as a machine learning model that can learn from the data and predict the characteristics of future workers. Some elements that might have an impact on burnout rate include:

* Gender: The employee’s gender (male or female).
* Company Type: The type of company where the employee works (service or product).
* WFH Setup Available: Whether the employee has a work from home facility available (yes or no).
* Designation: The employee’s level of work in the organization (a numerical value from 0.0 to 5.0, where higher

means higher designation).

* Resource Allocation: The amount of resource allocated to the employee to work, such as the number of working hours (a numerical value from 1.0 to 10.0, where higher means more resource).
* Mental Fatigue Score: The level of fatigue mentally the employee is facing (a numerical value from 0.0 to 10.0, where higher means more fatigue).
* Burn Rate: The target variable that we want to predict for each employee, which tells the rate of burnout while working (a numerical value from 0.0 to 1.0, where higher means more burnout).

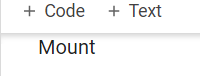
1. **Explain how linear regression can help solve this problem**.

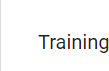
linear regression can help in finding employee burnout rate by fitting a linear model that minimizes the difference between the observed and predicted burnout rate. The model can use various features that are related to employee burnout, such as designation, resource allocation, mental fatigue score, gender, company type, etc. The model can then predict the burnout rate for new employees based on their features.

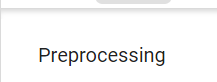
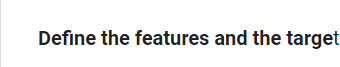
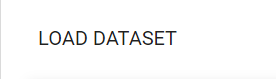
# You can solve the problem either locally using Jupyter Notebook or on Google Colab. (Hint: Utilize GPU for faster model training.)

Code on Google colab

# Ensure that your code is properly commented and use headings to separate different sections



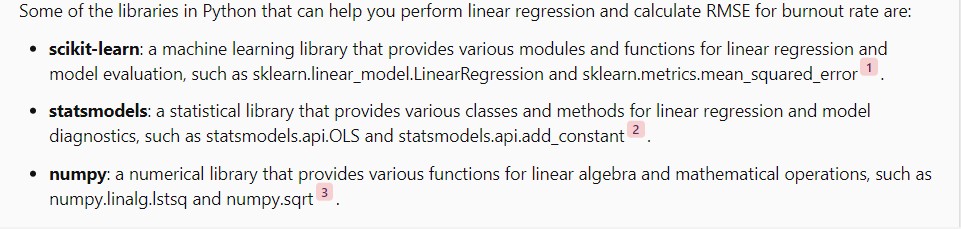
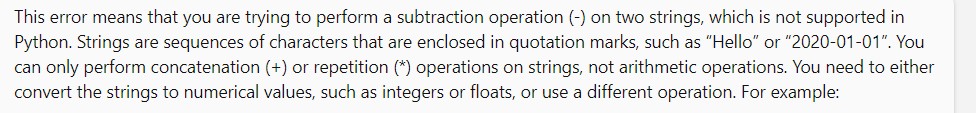
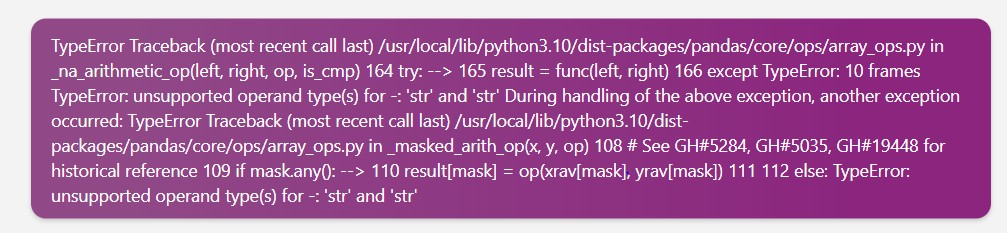
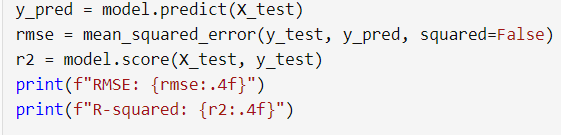
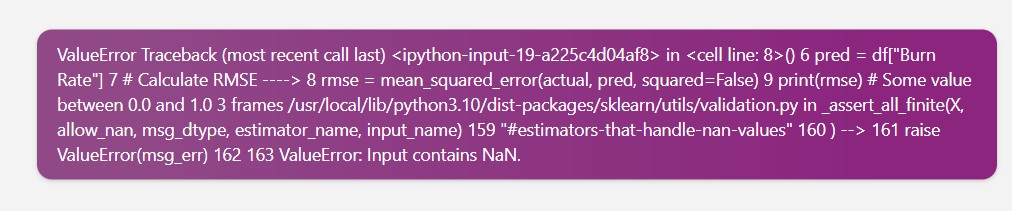








1. **Follow the ML lifecycle to solve the problem and provide separate code sections for each stage**





Splitting data into features (X) and target {y) is a common step in machine learning, where you separate the input variables (features) and the output variable (target) that you want to predict. Feats res are the characteristics or attributes of the data that can influence the target, such as gender, company type WFH setup availability, designation, resource allocation, and mental fatigue score in the burnout rate dataset. Target is the variable that you want to model

forecast, suchi as be rnout rate in the burnout rate dataset.

to split data into features (X) and target (yJ in Python, you can use pandas to select the columns oT the data frame that correspond to the features and the target. For example, iT you have a data frame named df that contains the burnout rate dataset, you can do:

X df . d top ( "Burn Rate™ , axi s=1) # Drop the named "Burn Rate" and as sign the test to y = df[ " Burn Rate" ] # Select the column named "Burn Rate" and as sign it to y

TS of 20 @

--- ValueError Traceback (most recent call last)

<ipython-input-70-e19d1f330d1d> in <ceII line: 2›0 1 4 Plot the testing data points in preen ----> 2 pit.scatter(X test y train, coIor='green', IabeI='Testing data') 3 4 # Plot the regression line in red 5 pit.plot(X test, y pred, coIor='red', labeI='Regression line') 2 frames /usr/local/lib/python3.10/dist-packages/matpIotIib/axes/ axes.py in scatter(self, x, y, s, c, marker, cmap, norm, vmin, vmax, alpha, linewidths, edpecolors, plotnonfinite, \*\*kwargs) 4582 y = np.ma.raveI(y) 4583 if x.size != y.size- > 4584 raise VaIueError("x and y must oe the same size") 4585 4586 if s is None: ValueError: x and y

must oe the same size

trainning in linear regression to find employe burnout rate

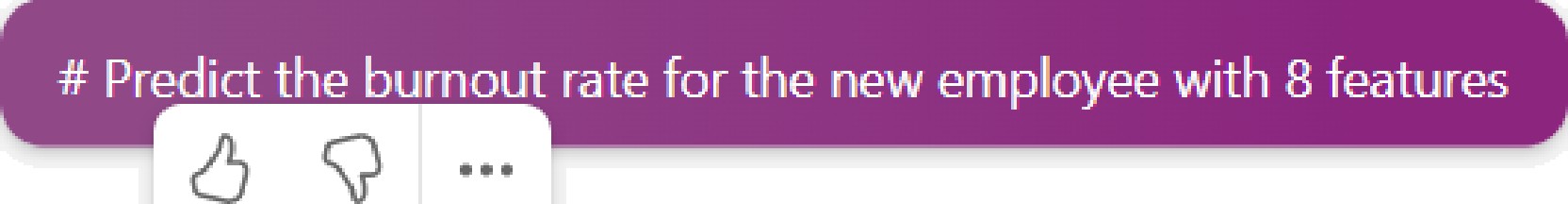
### It seems like you are trying to plot a scatter plot and a regression line using matplotlib, but you are getting an error. The error message says that x and y must be the same size, which means that the arrays X test and y train have different lengths. This is probably because you are using the wrong array for the y values. you should u3e y test instead of y train, since they match the X test Try changing your code to this:

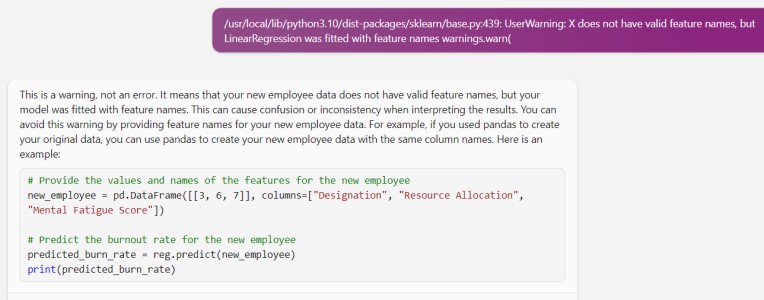
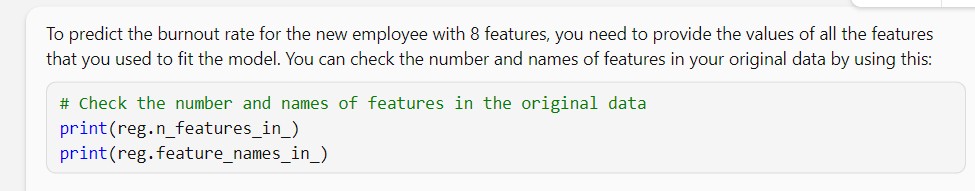
-# Plot the testing data points in green

p1t. scatter(X\_test, y\_test, color= ' green' , Gabe1= ' Testing data ' )

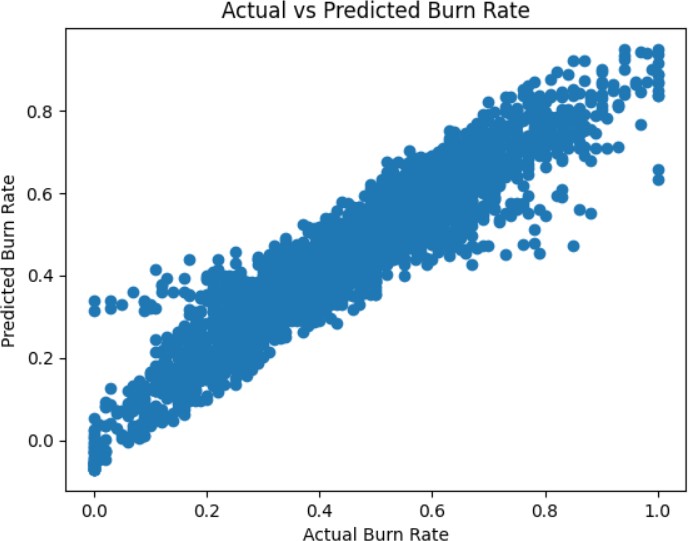
4 P1ot the regression 11ne in red

pit.plot(X test, y pred, color='red’, labe1=’Regression line’)





1. **Add more graphs in code to visualize specially train\_test graph**



1. **Use RMSE as accuracy measure try to minimize this as you can.**



**GITHUBLINK:**