School of Computer Science Engineering and Technology

Course- BTech	Type- Core
Course Code- 301	Course Name-AIML
Year- 2022	Semester- Even
Date- 26-01-2022	Batch- IV Sem Spl

3 - Lab Assignment # No. (3.1)

Objective: The task is to implement multiple linear regression regression.

1. As you saw in the previous simple linear regression task that previous year grades (G2) have significant correlation with third year grades (G3). But G2 is not direct causation of G3, there are many factors which determine G3. Let's add few more variables which may help to determine G3. Download the dataset 'Student Performance' provided by UCI Machine Learning repository.
Dataset link: https://archive.ics.uci.edu/ml/datasets/student+performance (5)

2. Read the data and store the features in X and output variable in Y. Consider the following features only in X from the downloaded dataset: (10)
Features (X)

1) age - student's age (numeric: from 15 to 22)
2) address - student's home address type (binary: 'U' - urban or 'R' - rural)
3) famsize - family size (binary: 'LE3' - less or equal to 3 or 'GT3' - greater than 3)
4) reason - reason to choose this school (nominal: close to 'home', school 'reputation', 'course' preference or 'other')
5) studytime - weekly study time (numeric: 1 -<2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)
6) failures - number of past class failures (numeric: n if 1<=n<3, else 4)
7) schoolsup - extra educational support (binary: yes or no)
8) famsup - family educational support (binary: yes or no)
9) paid - extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)
10) activities - extra-curricular activities (binary: yes or no)
11) higher - wants to take higher education (binary: yes or no)
12) internet - Internet access at home (binary: yes or no)
13) romantic - with a romantic relationship (binary: yes or no)
14) freetime - free time after school (numeric: from 1 - very low to 5 - very high)
15) goout - going out with friends (numeric: from 1 - very low to 5 - very high)
16) health - current health status (numeric: from 1 - very bad to 5 - very good)
17) absences - number of school absences (numeric: from 0 to 93)
18) G1 - first year math grades (numeric: from 0 to 100)
19) G2 - second year math grades (numeric: from 0 to 100)

Output target (Y)

20) G3 - final year math grades (numeric: from 0 to 100, output target)

3. **Data Pre-processing step:** Transform categorical features into numerical features. Use either one hot encoding, label encoding or any other suitable pre-processing technique. Also scale the numerical columns value using minmax_scale() or any other scaling function. (10)

4. Train Linear Regression Model (sklearn.linear_model.LinearRegression class) (10)

5. Print 'Mean Squared Error' (MSE) obtained on the same dataset i.e. same X and y (5)
(sklearn.metrics.mean_squared_error function)

Further fun (will not be evaluated)

- Train LassoRegression and RidgeRegression as well. Read about them from scikit-learn user guide.
- *Step-up challenge*: Get down the MSE (mean squared error) below 3.25 using linear models
- Implement multiple linear regression from scratch
- Plot loss curve (Loss vs number of iterations)

Helpful links

- Scikit-learn documentation for linear regression: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html
- Read till where you feel comfortable: https://jakevdp.github.io/PythonDataScienceHandbook/05.06-linear-regression.html