Build regression models using NumPy with Python

Business Objective

Regression is a supervised learning algorithm. Regression analysis is used to establish a relationship between one or more independent variables and a dependent variable. There are several variations in regression analysis like linear, multiple linear, and nonlinear.

One can create regression models with the help of the ‘Scikit-learn’ library, the most valuable and robust library for machine learning in Python. But, in this project, we will be building our models from scratch using NumPy. Building your model allows for more flexibility during the training process, and one can tweak the model to make it more robust and responsive to real-world data as required in the future during re-training or in production.

This project explains how linear regression works and how to build various regression models such as linear regression, ridge regression, lasso regression, and decision tree from scratch using the NumPy module.

Data Description

The dataset provides information about the players of a particular sport, and the target is the predict the scores. The dataset consists of around 200 rows and 13 columns.

Aim

To build multiple regression models from scratch using the NumPy module.

Tech stack

- Language - Python
- Libraries - Pandas, NumPy

Approach

1. Importing the required libraries and reading the dataset.
2. Data pre-processing
    - Removing the missing data points
    - Dropping categorical variables
    - Checking for multi-collinearity and removal of highly correlated features
3. Creating train and test data by randomly shuffling data
4. Performing train test split
5. Model building using NumPy
    - Linear Regression Model
    - Ridge Regression
    - Lasso Regressor
    - Decision Tree Regressor
6. Model Validation
    - Mean Absolute Error
    - R2 squared

Modular code overview

```
InputFiles
  |_EPL_Soccer_MLR_LR.csv

SourceFolder
  |_Engine.py
  |_ML_Pipeline
            |_DataPreparation.py
            |_metrics.py
            |_label_encoding.py
            |_LinearRegression.py
            |_LassoRegression.py
            |_RidgeRegression.py
            |_RegressionTree.py
  |_README
  |_requirements.txt

Lib
  |_Data_Exploration.ipynb
  |_Regression_Models.ipynb
  |_Regression_Tree_numpy.ipynb


Output
  |_linear_model.pkl
  |_lasso_model.pkl
  |_ridge_model.pkl
  |_Reg_tree_model.pkl
```

Once you unzip the modular_code.zip file you can find the following folders within it.

1. input
2. src
3. output
4. lib

1. Input folder **-** It contains all the data that we have for analysis. There are two csv files in our case,
    - EPL_Soccer_MLR_LR

2. Src folder - This is the most important folder of the project. This folder contains all the modularized code for all the above steps in a modularized manner. This folder consists of:
    - Engine.py
    - ML_Pipeline
      The ML_pipeline is a folder that contains all the functions put into different python files which are appropriately named. These python functions are then called inside the engine.py file.

3. Output folder **-** The output folder contains the four models that we trained for this data. These models can be easily loaded and used for future use and the user need not have to train all the models from the beginning.
   Note: This model is built over a chunk of data. One can obtain the model for the entire data by running engine.py by taking the entire data to train the models.

4. Lib folder **-** This is a reference folder. It contains the original ipython notebooks that we saw in the videos. There is a reference folder that consists of the presentations used during the explanation.

Project Takeaways

1. What is Regression?
2. What are the applications of regression?
3. Different types of regression
4. Differentiation between regression and classification
5. What is linear regression
6. What is loss function?
7. What is gradient descent?
8. Drawbacks of linear regression
9. Understanding bias and variance
10. What is ridge and lasso regression?
11. What is a decision tree?
12. Understanding the different terminologies in the decision tree
13. Advantages and disadvantages of decision trees
14. Importing the dataset and required libraries.
15. Missing data handling using appropriate methods.
16. Finding a correlation between the features.
17. Building various regression models from scratch using NumPy module
18. Gaining confidence in the model using metrics such as MSE, and R squared.