Introduction to Machine Learning - Practicum 3 - Linear Regression

Topics covered: Linear Regression, Gradient Descent Algorithm, RMSE

Deliverables:

- Your submission for this practicum should be an archive of two files, named ***lin_reg.py*** and ***<your_name>_report.pdf***.
- ***lin_reg.py*** should contain all the codes necessary with proper comments for the practicum.
- ***your_name_report.pdf*** should follow the outline described in Section 4 and you should put in all the required materials including the answers to all the questions.
- You should zip both files named ***<your_name>_ML_P03.zip***.
- In addition to the two files, your zip file should contain:
    - a folder named ***output***, where your output files are stored.
    - a folder named ***data*** where you will store your training datasets that you have split using the methods stated in Section 5.

  This will help me to run your code with the exact split of train and test dataset that you have made in your implementation. Failure to follow conventions will result in penalties in marks.

Objectives:

- To get familiarized implementing your second machine learning algorithm, which is Linear Regression from the scratch (i.e., without using any machine learning API).
- To implement Gradient Descent algorithm to learn the parameters of linear regression function from the training data.
- To apply linear regression algorithm to predict the housing prices in Boston city.
- To get familiarized on tuning the performance of gradient descent algorithm.
- To demonstrate the capability to write simple technical report on describing a machine learning algorithm and its performance.

1. Linear Regression

In statistics, linear regression is a linear approach for modeling the relationship between a scalar dependent variable y and one or more explanatory variables (or independent variables) denoted X. The case of one explanatory variable is called simple linear regression. For more than one explanatory variable, the process is called multiple linear regressions. Linear regression was the first type of regression analysis to be studied rigorously, and to be used extensively in many practical applications. Linear regression has been used in prediction, forecasting, and error reduction by fitting a predictive model to an observed data set of y and X values. After developing such a model, if an additional value of X is then given without its accompanying value of y, the fitted model can be used to make a prediction of the value of y.

In this practicum, you will design the linear regression algorithm to predict the housing prices in Boston using the dataset provided by the UCI Machine Learning repository https://archive.ics.uci.edu/ml/machine-learning-databases/housing/.

2. Gradient Descent

When there are one or more inputs variables, you can use a process of optimizing the values of the coefficients by iteratively minimizing the error of the model on your training data. This operation is called Gradient Descent and works by starting with random values for each coefficient. The sum of the squared errors is calculated for each pair of input and output values. A learning rate is used as a scale factor and the coefficients are updated in the direction towards minimizing the error. The process is repeated until a minimum sum squared error is achieved or no further improvement is possible.

When using this method, you must select a learning rate (α) parameter that determines the size of the improvement step to take on each iteration of the procedure. Gradient descent is often taught using a linear regression model because it is relatively straightforward to understand. In practice, it is useful when you have a very large dataset either in the number of rows or the number of columns that may not fit into memory.

In this practicum, you will implement the gradient descent algorithm from the scratch to learn the parameters for your linear regression model.

3. Dataset

The Boston house price dataset contains information about the housing values in suburbs of Boston. This dataset was originally taken from the *StatLib* library which is maintained at Carnegie Mellon University and is now available on the UCI Machine Learning Repository. Each instance describes the properties of a Boston suburb and the task is to predict the house prices in thousands of dollars. There are 13 numerical input variables with varying scales describing the properties of suburbs. The readme file in the dataset provides details on the dataset. Read this file before implementing your algorithm to check whether you need to do any data preprocessing steps. To give a brief introduction, the Boston housing data consists of 506 entries that represent the aggregated data about 14 features for homes from various suburbs in Boston. The input attributes and the output attribute are as follows:

1. *CRIM*: per capita crime rate by town
2. *ZN*: proportion of residential land zoned for lots over 25000 sq. ft.
3. *INDUS*: proportion of non-retail business acres per town
4. *CHAS*: Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
5. *NOX*: nitric oxides concentration (parts per 10 million)
6. *RM*: average number of rooms per dwelling
7. *AGE*: proportion of owner-occupied units built prior to 1940
8. *DIS*: weighted distances to five Boston employment centers
9. *RAD*: index of accessibility to radial highways
10. *TAX*: full-value property-tax rate per $10,000
11. *PTRATIO*: pupil-teacher ratio by town
12. *B*: 1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town
13. *LSTAT*: % lower status of the population
14. *MEDV* (output): Median value of owner-occupied homes in $1000's

4. Documentation

For documentation, you can use LATEX or Word format that has been provided in your first practicum. You need to write a detailed technical report on this practicum by explaining the algorithm, the techniques you have used to implement it, your dataset, evaluation metrics and how did you split your training and test data, etc. The results should be plotted using appropriate graphs and if needed use tables to report the results. More importantly, you need to acknowledge the learning materials in the reference section that you have used for this practicum. You are highly encouraged to use equations, tables, figures, charts, etc. wherever appropriate. Failing to do so will result in deduction of points for documentation. You should submit ONLY the “pdf” of the document and NO other file formats will be accepted.

5. Your tasks

1. State whether you need to perform any data preprocessing on this dataset and do those preprocessing steps if needed.
2. Construct the linear regression equation to predict the housing prices in Boston. Explain this equation briefly and the number of parameters that need to be estimated.
3. Plot the input attributes (pick any 5 relevant attributes) and the output attribute in the given dataset to visualize how the output variable varies with respect to each of your input attribute.
4. Construct the error function using the linear regression equation that you constructed in task 2. Again, briefly explain this equation.
5. Describe briefly on the gradient descent learning algorithm for learning the parameters of your linear regression function. Implement it. The major part of this task is to find the optimal value of the learning rate (α) and the convergence criterion. How will you do this? Explain in this section. (Also note, you may need to use this implementation in your future practicums)
6. Split the dataset into training and test set using k-fold cross-validation method. Choose the k value to be 5, 10 and 15. (You can use your code from your previous practicum)
7. For each of the folds in task 6, run the gradient descend algorithm to compute the parameters. Using this parameters, plot the values of predicted_y and the actual y values from the training data. What do you observe? In addition, you need to evaluate on how accurate is your model predicts the output. In order to do this, you need to measure RMSE (root mean squared error). Explain briefly on RMSE and compute the RMSE for each of the fold in task 6.
8. Write down the other metrics similar to RMSE, to evaluate the model fit?
9. Discuss the pros & cons of the Linear Regression Algorithm.
10. Discuss different ways that the Linear Regression algorithm can be improved.

6. Rubrics

This practicum is graded over total of 70 points. The breakdown is as follows:

- Correct answers to task 1 (and implementations, if any) (5 points)
- Proper construction of linear regression equation and sufficient explanation (5 points)
- Proper plot and discussions of task 3 (5 points)
- Error function equation and descriptions (5 points)
- Correct implementation of Gradient Descent algorithm and answers to task 5 (10 points)
- Discussion on performance and computation measurements for each experimental setup presented in detail (10 points)
- Plots of predicted_y and actual_y values from dataset in task 7 (5 points)
- Provide a table of RMSE values for each value of k and explain (7 points)
- Correct answers to task 8 (3 points)
- Correct answers to task 9 and 10 (5 points)

Note: Submission requirements must be met i.e., reasonable comments, proper format of the report (Not copy and pasted from the practicum specifications!), detailed explanation (using necessary equations, tables, figures, charts, etc.), correct documentation and file formats for the submission, etc. If any of these requirements are not satisfactory, then zero marks for this practicum.