Linear Regression: Deep Learning

Olaniyi Bayonle Alao
Summer Term, 2021
Bachelor of Electronic Engineering
Hochschule Hamm-Lippstadt
Lippstadt, Germany
olaniyi-bayonle.alao@stud.hshl.de

***Abstract*—Machine learning refers to is the ability of applications to get better at doing things without necessarily a change in the code base them is something that has really been helpful especially in this era of big data. Even though machine learning is not a new concept, it is fast gaining recognization and changing lives thanks to the increase in the processing power of computers over the years to be able to process big data at a level that has never been experienced. This paper talks about linear regression algorithm in the context of machine learning. Linear regrression is a statictical term that uses a dependent and independent variable to make predictions. In these paper, the theorectical background of linear regression was explored. Predicting target values using the Scikit-Learn scientific library for machine learning in python was made. The data used in this paper were gotten from the Scikit-Learn dataset repository library. The California housing dataset from this repository was used for training and prediction using linear regression. As a mention of how good a model as learnt, mean squared error and coefficient of determination were explained. As a conclusion, an evaulation of the trained model was discussed to understand how well it performed in predicting.**

***Index Terms*—machine learning, linear regression, deep learning, sci-kit**

Unless otherwise stated, the main reference for facts, mathematical models and terminologies used in this research paper is gotten from [6]

I. Introduction

Due to an explosive increase in the amount of data generated from different internet connected devices -Cyber-Physical-Systems (CPS)-, there has been an increase in the need to make sense of this "big data" to ensure the proper and productive functioning of businesses. By 2026, the data generation from global air fleet is projected to reach 98 billion gigabytes [1]. Thanks to the recent advancements in the information processing capabilities of computers, making sense of this huge chunk of data is now possible using data analysis techniques and machine learning algorithms to clean and make prediction from this datasets. Machine learning is the ability of machine -software application- to learn and make predictions without being programmed to do so. They are able to learn and get better with an increase in the amount of data they are being fed through through their grounded mathemical foundations.

Linear regression is a statistical test used to find out the relationship between independent and dependent variables in a data set using mathematical formular. They can also be used in projecting new relationship between the dependent and independent variables that has not been discovered. Even though linear models - linear regression - are quite simple to develop and understand, as well as good at predicting linear relationships, their approximation of nonlinear relationships have been found to be mostly unsatisfactory [5]. This paper goes more into the mathematical details this paper gives a profound, yet understandable mathematical description of Linear regression, relevant parts of the framework -Sci-Kit- used to perform predictions on the example dataset in the context of Machine learning using the Python programming language.

II. Theoretical Background

A. Machine Learning

According to [3], "Machine learning is a branch of artificial intelligence (AI) focused on building applications that learn from data and improve their accuracy over time without being programmed to do so". Examples of applications built using machine learning trained models are email filters with the ability to distinguish between desired emails and spams, auto-suggestion/auto-correct in many typing applications, self-driving vehicles, amongst others. Machine learning techniques have the advantage of automatically adapting to change detected in trends from data. Likewise, they help reduce the complexity in writing application that are somethings difficulty or even impossible to implement using traditional algorithms. The approach of solving an application complexity using machine learning is shown in figure 1.Methods used in training a Machine learning model can be grouped into three main categories.

- Supervised Learning
  This is a type of machine learning technique in which the algorithm is fed with an input and output dataset of desired solution. The desired output data is otherwise known as lables. These information help train the machine lerning model in the precise prediction of output when given a related output after being trained. Some examples of common algotithms used for supervised learnings are regression analysis - linear regression, etc.-, decision trees, amongst others. [9] The two types of supervised learning techniques used are classification and regression.
- Unsupervised Learning

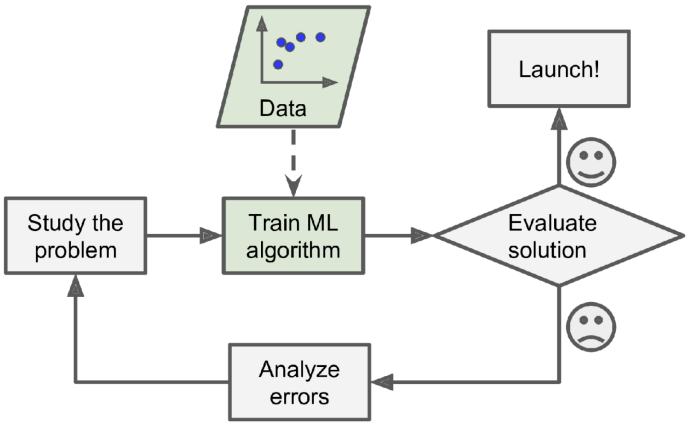

Fig. 1. Machine Learning approach

This refers to the machine learning technique in which the algorithm is only fed with input data for the training process. The model learns "unsupervised" by finding out and grouping patterns in inputted datasets. Unlike supervised learning, they require large amount of unlabeled datasets to train -properly find patterns in datasets.

- Semisupervised Learning
  this is a learning technique that uses both labeled -input and desired output- and unlabled -only input- datasets for training machine lerning models.

The other methods used in training a machine learning model are reinforcement learning and deep learning. Deep learning is a subset of machine learning who's algorithm defines an artificial neural network -ANN- that is designed to emulate the way an human brain learns, and unsupervised or semisupervised learning technique to train [3]. On other hand, reinforcement learning is a reward based learning in the sense that the machine leaning is trained by being given a point for reaction to certain events.

B. Linear Regression Model

Regression is a predictive analysis with a long but glorious history from its successful applications to problems in the statistics and economics domain. Regression is a kind of supervised learning technique for determining the best fit line to describe patterns in data: linear regression uses a straight line to describe these patterns. [9]. The best fit line is the line that reduces the summed squared difference between the value of the line of a certain value ***x*** and its corresponding ***y*** values [**?**]. The mathematical expression for linear regression is:

$$y = \beta X + \beta_0 \tag{1}$$

Where y is the dependent variable, X is the independent variable of the equation, β is a cofficient which reperesnts the slope of the regression line, β_0 is a constant value called the **bias**. Equation (1) is the same as the equation of a straight line in linear algebra.

The independent variable X in (1) is calculated using the mathematical expression below:

$$X = \frac{\sum (x - \overline{x})(y - \overline{y})}{\sum (x - \overline{x})^2} \tag{2}$$

Where $\overline{x}$ and $\overline{y}$ are the mean values of all the respective x and y variables.

Linear regression analysis can be sub-divided into simple and multiple linear regression depending on the number of their independent variables. If the independent variable is one, the regression is said to be a simple linear regression, but if the independent variable is more than one, it is a multiple linear regression. In this paper, we will make analysis using the simple linear regression model.

C. Model Evaluation

The prediction made by the model using equation (1) can be evaluated using the statistical methods below to find out how close the predicted value is to the the actual value.

1) R^2: this is otherwise known as the coefficient of determination or the coefficient of multiple determination is a measure of how close the best fit line is to the original data using a simple mean. The value of the output of this calculation ranges from 0 to 1 [8]. A value of 1 refers to the fact that all the points fits 100% to the the regression line. The lesser the value, the farther the points are to the regression line. The coefficient of determination is defined by the following equation:

$$R^2 = \frac{\sum_{i=1}^{n} (\hat{y}_i - \overline{y})^2}{\sum_{i=1}^{n} (y_i - \overline{y})^2} \tag{3}$$

Where n is the number of observations i.e. total number of variables, $\hat{y}_i$ is the estimated value of the dependent variable for the i^{th} observation computed by the regression equation, y_i is the observed value of the dependent variable for the i^{th} observation and $\overline{y}$ is the mean of all n observations of the dependent variable [8]

2) Mean Squared Error: Mean Squared Error (MSE) is a performance measure that helps determine how much error is made in the predicted value in relation to the the actual value of the ouput given the same input. The equation of the calculation is denoted by:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{f}(x_i))^2 \tag{4}$$

where x_i is a vector of all i^{ith} observation, $\hat{f}$ is the prediction function. The lower the value of MSE, the more accurate the prediction is.

III. Algorithm Implementation

All the algorithm needed for prediction using the linear regression model have been implemented in machine learning libraries like Scikit learn which we will be using in this paper.

A. Libraries and Tools used

1) Scikit Learn: Scikit learn is an open-source python library that provides algorithms that are used in machine learning. This library provide functionalities for solving machine learning jobs like regression, classification, clustering, model selection, pre-processing - like spliting data sets into test and train subsets -amongst others [7]. The scikit-learn API are designed around this main design principles which are consistency - all objects (basic or composite) share a consistent interface composed of a limited set of methods -, inspection - parameters are exposed as public attributes -, composition, sensible defaults - provides understandable default parameters which gives baseline solution for tasks at hand -and nonproliferation of classes - datasets are represented as NumPy arrays or SciPy sparse matrices [2].

2) Numpy: NumPy is an open-source Python library that that provides routines that allows for fast operations on multidimensional arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation[1].

3) Pandas: Pandas is an open-source data analysis and manipulation python library that allows high-performance easy exploration, cleaning and processing of tabular data structures in Python[2].

4) Matplotlib: Matplotlib is a Python library used along with Numpy for creating static, animated and interactive visualization of data[3].

5) Seaborn: Seaborn is a Python library which builds upon the functionalities of matplotlib and integrates closely with pandas data structures for making statistical graphics to datasets[4].

Figure 2 shows the import statments for the functions and libraries mentioned above in python.

```
from sklearn.datasets import fetch_california_housing
import pandas as pd
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import matplotlib.pyplot as plt
%matplotlib inline
```

Fig. 2. Import statements

IV. Practical Example

In this paper, the california housing datasets from [4] included in the scikit sklearn datasets library is used to train and predict the average prices of housing in california using linear regression.

[1]https://numpy.org/doc/stable/user/whatisnumpy.html
[2]https://pandas.pydata.org/docs/
[3]https://matplotlib.org/
[4]https://seaborn.pydata.org/introduction.html

A. Inspecting Dataset

The dataset used in this paper is gotten from scikit-learn dataset repository using. The dataset has 20,640 instance number, 8 numeric predictive atrributes and 1 target attribute which is expected output of the prediction, which is the median house value for California districts. A block group is the smallest geographical unit for which the U.S. Census Bureau publishes sample data (a block group typically has a population of 600 to 3,000 people), the datasets were obtained in a 1990 census [4]. The predictive attributes have the following information:

- MedInc: represents the median income in a block.
- HouseAge: represents the median house age in block.
- AveRooms: represents the average number of rooms.
- AveBedrms: represents the average number of bedrooms.
- Population: represents the population in the block.
- AveOccup: represents the average house occupancy.
- Latitude: represents the house block latitude.
- Longitude: represents the house block longitude.

A summary of these attributes and the corresponding target values as pandas dataframe is depicted in figure 3. The target attribute represents the median house value in unit of 100,000.

	MedInc	HouseAge	AveRooms	AveBedrms	Population	AveOccup	Latitude	Longitude	target
0	8.3252	41.0	6.984127	1.023810	322.0	2.555556	37.88	-122.23	4.526
1	8.3014	21.0	6.238137	0.971880	2401.0	2.109842	37.86	-122.22	3.585
2	7.2574	52.0	8.288136	1.073446	496.0	2.802260	37.85	-122.24	3.521
3	5.6431	52.0	5.817352	1.073059	558.0	2.547945	37.85	-122.25	3.413
4	3.8462	52.0	6.281853	1.081081	565.0	2.181467	37.85	-122.25	3.422
...	...	...	...	...	...	...	...	...	...
20635	1.5603	25.0	5.045455	1.133333	845.0	2.560606	39.48	-121.09	0.781
20636	2.5568	18.0	6.114035	1.315789	356.0	3.122807	39.49	-121.21	0.771
20637	1.7000	17.0	5.205543	1.120092	1007.0	2.325635	39.43	-121.22	0.923
20638	1.8672	18.0	5.329513	1.171920	741.0	2.123209	39.43	-121.32	0.847
20639	2.3886	16.0	5.254717	1.162264	1387.0	2.616981	39.37	-121.24	0.894

20640 rows × 9 columns

Fig. 3. Dataset overview including the target column

The correlation between the attributes is shown in figure 4. Figure shows the pearson correlation between the attributes as a heatmap. The darker the blue colour and the closer the value is to 1, the higher the correlation of the attribure to each other. It can be seen from figure 4 that the median income income in a block has the highest correlation to the target value followed by house age and average rooms. The other values show little or no correlation to the target value because of their negative values.

B. Split Dataset and Train Model

The dataset is split into training and testing data using the $train\_test\_split$ function provided by the $sklearn.model\_selection$ class. The dataset was split into 20% - 4,218 - for testing purposes and 80% -16, 512 - for training purposes. Split the dataset gives the opportunity of setting aside some dataset for testing purposes to efficiently evaluate the accuracy of the trained model. If the dataset were not split, there is probability of the model predicting the target value through memorising and we want to avoid this.

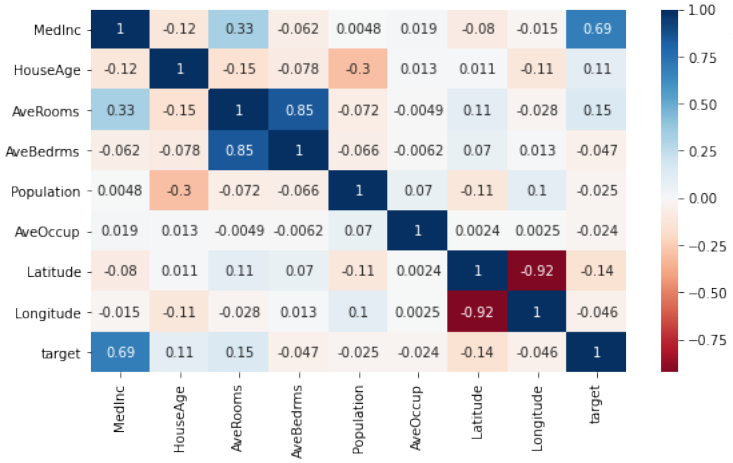

Fig. 4. Pearson correlation heatmap of the dataset attributes

An instance of the linear regression model class was instanced, and the training datasets were fed into the fit function which is an implementation of the linear regression formula. The value of the coefficients after training the model is [4.48674910e-01 9.72425752e-03 -1.23323343e-01 7.83144907e-01 -2.02962058e-06 -3.52631849e-03 -4.19792487e-01 -4.33708065e-01]. The value of the intercept was found to be -37.023277706063894. Ideally, these are the values that are needed if manual calculation of the target value y is to be made.

C. Result and Discussion

With the regression model successfully trained, the testing datasets can now be fed to the model to predict the target values. The coefficient of determination R^2 of training datasets for the model was found to be 0.6125511913966952. When predictions were made using the testing dataset, there were some offsets in the predicted values to the target value as shown in figure thatttttt which shows a graph of the first 20 values. The Mean squared error of the predicted value to the target value was found to be 0.56 and coefficient of determination to be 0.58. These values explains the reason why there were some offsets in the predictions.

With the result of the predictions from the model, it can be said that the model has underfitted the training data because the features used did not have an accurate linear corellation. This can be seen in figure 4 which showed some negative values. We can also understand that a lot of not so known factors influence house prices which were not included in the datasets.

V. Conclusion

As can be highlighted from the linear regression equation, it is a fairly straightforward regression model to get started with machine learning. However, there are quite poor at predicting with datasets that do not have strong linear correlation as depicted in the trained model in this paper. As a suggestion, different regression algorithms should be used to train datasets to be able to find the best fit for the desired prediction job.

References

[1] Mro survey 2016.

[2] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122, 2013.

[3] IBM Cloud Education. What is machine learning?, Jul 2020.

[4] R. Kelley Pace and Ronald Barry. Sparse spatial autoregressions. *Statistics & Probability Letters*, 33(3):291–297, 1997.

[5] K.-Y Lee, K.-H Kim, J.-J Kang, S.-J Choi, Y.-S Im, Y.-D Lee, and Y.-S Lim. Comparison and analysis of linear regression & artificial neural network. *International Journal of Applied Engineering Research*, 12:9820–9825, 01 2017.

[6] Luca Massaron and Alberto Boschetti. *Regression Analysis with Python*. Packt Publishing Ltd, 2016.

[7] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[8] Astrid Schneider, Gerhard Hommel, and Maria Blettner. Linear Regression Analysis. *Dtsch Arztebl International*, 107(44):776–782, 2010.

[9] Oliver Theobald. *Machine learning for absolute beginners: a plain English introduction*. Scatterplot press, 2017.