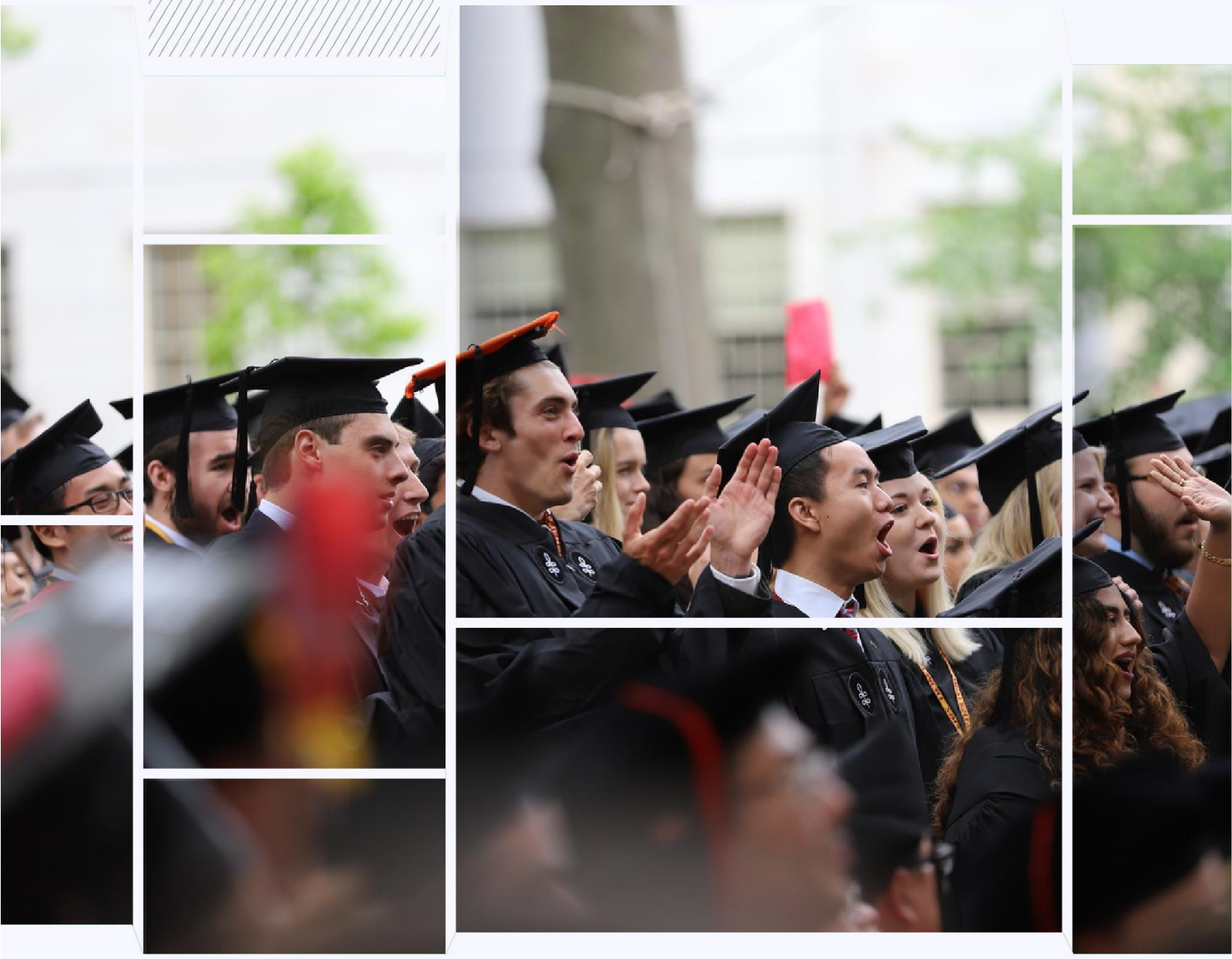
HARVARD EDUCATIONAL AND SCIENTIFIC REVIEW

International Agency for Development of Culture, Education and Science



Vol.2. Issue 1 Pages 181-187.
10.5281/zenodo.7010047

Harvard Educational and Scientific Review
International Agency for Development of Culture, Education and Science
United Kingdom
Street: 2 High Street City: Ashby Phone number 079 6425 7122
Zip code DN16 8UZ Country United Kingdom
USA
Soldiers Field Boston, MA 02163 +1.800.427.5577

International Agency for Development of Culture, Education and Science United Kingdom
USA Soldiers Field Boston

Har. Edu.a.sci.rev. 0362-8027 181
Vol.2. Issue 1 Pages 181-187.
10.5281/zenodo.7010047

THE APPLICATION OF MULTIPLE LINEAR REGRESSION ALGORITHM AND PYTHON FOR CROP YIELD PREDICTION IN AGRICULTURE

Nodir Rahimov
Tashkent University of Information Technologies named after Muhammad al-Khwarizmi, Tashkent, Uzbekistan, E-mail: r_nodir@mail.ru
Khasanov Dilmurod
Tashkent University of information technologies named after Muhammad al-Khwarizmi, Tashkent, Uzbekistan, E-mail: tatusf2015@gmail.com

INTRODUCTION

The difference between Linear regression and Multiple Linear regression methods is at number of independent variables (parameters). Not always every result depends on only one thing. Therefore, Multiple Linear regression method is more effective than Linear regression. For example, in automobile industry, each details of a car can be made different technology and company. As well as, each detail can have various quality and material. That is why, every car obviously has different price. And this can bring the issue that calculate the price of a car not easily. Not only in automobile industry but also any manufacturer company has this kind of problem in today`s world. Agriculture also has such kind of problems. One of them is prediction crop yield for next year or next seasons. Especially, it is the most important thing in countries that rely on agriculture. Because in agriculture major there are so many different parameters that impact on crop yield. The weather, rainfall, amount of minerals that is given by farmers can be example for it. Due to the process of calculating is complicated we have clear and real dataset about crop yield of the last years. There are several steps from data collection to prediction. These steps are illustrated in this diagram [2]:

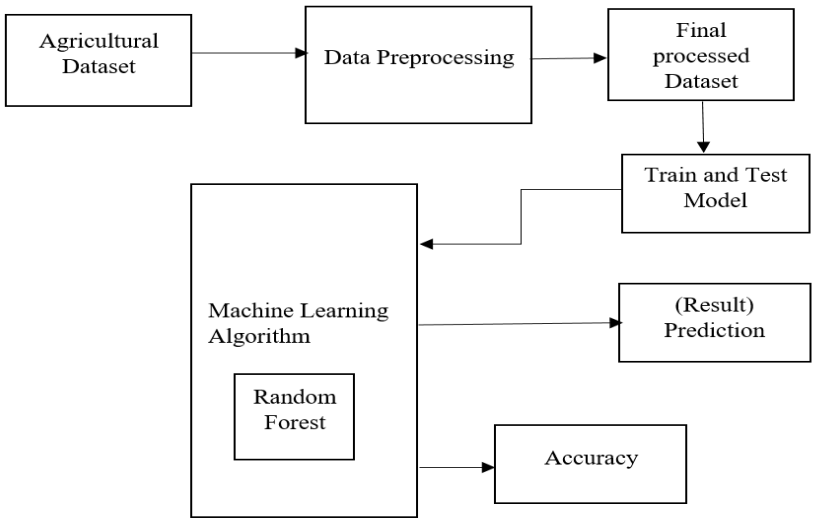


Figure 1. Steps from data collection to prediction.

In the first step, Agricultural Dataset is collected by domain scientists. Crop prediction with this proposed system developed with only by using soil properties such as micro; macro nutrients Ec (electrical conductivity), pH (acidity) values as input or independent features and suggested crop as

output or dependent features. The abovementioned soil properties is collected from a soil lab. Dataset consists of nearly 1600 samples. The dataset is analyzed before generating a model.

Next step consists of Data Preprocessing. Finding correlation among features: The study of data reveals the nature of data as numerical data. The correlation values ranges from -1 to +1; the correlation value of a feature which is near to -0.01 to +0.01 can be dropped since it denotes the value is equal to zero which mean there is no correlation. In this dataset the features such as N, P, Na, Zn and B were removed for crop prediction.

Third stage includes finishing processed dataset. It is called also dataset scaling. The dataset is examined to find out the range of the feature it is found that the values differ their exits no uniformity; with this; it is not possible to generate a correct model. A solution is to scale all the values in a predefined range which is nothing but -1 to +1. The above step called scaling and it is implemented with the help of Standard Scalar a pre processing function in sklearn. The code is as follows. For this process Python has some specific libraries (figure 2).

```
In [ ]: from sklearn.preprocessing import StandardScaler
        scaled_data = StandardScaler().fit(X)
        X_scal = scaled_data.transform(X)
```

Figure 2. Import and usage Data scaling library in Python.

Next stage in this process is training machine and testing accuracy prediction with dependent variables . First of all predicting was had to observe that program based on algorithm analyses the given data (dependent and independent variables) and tries to predict new value of function for next season. In order to do it we need Gradient Descent to approach to result. And for calculating Gradient Descent also need Cost function. As a Cost function we chose MSE (Mean Square Error) function for Multiple Linear Regression algorithm [3].

$$MSE = \frac{1}{2m}\sum_{i=1}^{m}(f(x_i) - y_i)^2 \qquad (1)$$

m – number of data in dataset,
i – index of data in dataset,
f(x) – prediction value of function in i-index,
y – dependent variable in i-index.

The purpose of Gradient Descent is minimizing Cost function. Because of this, we work on MSE. As we know, in order to minimize the function we have to find the coefficients that approaching minimum value of this function.

As well as, the theory of Regression method is had to learnt before the main process. First and foremost method in statistics is linear regression. This method is very useful for one parameter problems such as tuition fee in education, taxes in economics, price of raw materials and so on. The mathematical equation representation for the same is:

$$y = a + bx \qquad (2)$$

where y is the predicted output, x is the input variable, b is the slope and a is the bias. The above idea can be extended to multiple linear regression where more than one input features which produces single

output feature. This method is usually used for complicated problems in marketing, economics, tax system.

The mathematical representation of multiple linear regression is:

$$y = b_1x_1 + b_2x_2 + b_3x_3 + \cdots + b_nx_n + a \quad (3)$$

A neural network model can be created by calculating Weights and bias value at each and every node . The layer consists of various nodes, layers are classified in to input, hidden and output layers. Inputs are multiplied with weights of the node to form a summation of the activation function . The activation is a transformation function that may be a linear or non-linear, applied to every input before it gets transferred to the next layer or to the output layer [1,12]. After that we can define the Cost function with coefficients:

$$MSE = \frac{1}{2m}\sum_{i=1}^{m}((b_1x_1^{(i)} + b_2x_2^{(i)} + \cdots + b_nx_n^{(i)}) - y^{(i)})^2 \quad (4)$$

n – number of columns. Data in dataset generate vectors m x n (m – rows number, n – columns number).

In (4) formula there are have n coefficients and they are:

B = [$b_1, b_2, \dots, b_n$] – saved as a vector

Based on the rule of Gradient Descent we take a one order derivative and generate these formulas:

$$\begin{cases} \frac{\delta MSE}{\delta b_1} = \frac{1}{m}\sum_{i=1}^{m}\left(\left(b_1x_1^{(i)} + b_2 2 + \cdots + b_nx_n^{(i)}\right) - y^{(i)}\right)x_1^{(i)} \\ \frac{\delta MSE}{\delta b_2} = \frac{1}{m}\sum_{i=1}^{m}\left(\left(b_1x_1^{(i)} + b_2 2 + \cdots + b_nx_n^{(i)}\right) - y^{(i)}\right)x_2^{(i)} \\ \frac{\delta MSE}{\delta b_3} = \frac{1}{m}\sum_{i=1}^{m}\left(\left(b_1x_1^{(i)} + b_2 2 + \cdots + b_nx_n^{(i)}\right) - y^{(i)}\right)x_3^{(i)} \\ \qquad \dots\dots\dots\dots \\ \qquad \dots\dots\dots\dots \\ \frac{\delta MSE}{\delta b_n} = \frac{1}{m}\sum_{i=1}^{m}\left(\left(b_1x_1^{(i)} + b_2 2 + \cdots + b_nx_n^{(i)}\right) - y^{(i)}\right)x_n^{(i)} \end{cases} \quad (5)$$

Using formulas above all coefficients are updated like:

$$\begin{cases} b_1 = b_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(\left(b_1 x_1^{(i)} + b_2 2 + \cdots + b_n x_n^{(i)} \right) - y^{(i)} \right) x_1^{(i)} \\ b_2 = b_2 - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(\left(b_1 x_1^{(i)} + b_2 2 + \cdots + b_n x_n^{(i)} \right) - y^{(i)} \right) x_2^{(i)} \\ \cdots\cdots\cdots \\ \cdots\cdots\cdots \\ b_n = b_n - \alpha \frac{1}{m} \sum_{i=1}^{m} \left(\left(b_1 x_1^{(i)} + b_2 2 + \cdots + b_n x_n^{(i)} \right) - y^{(i)} \right) x_n^{(i)} \end{cases} \quad (6)$$

In each epoch all coefficients are updated and calculate predict. After that program calculates the Cost function and check the accuracy, if MSE value is small number to dependent value in dataset, then predict the result. Now we can pass back the Agriculture major for application of Multiple Linear Regression in this sphere.

For the beginning training all libraries are imported:

```
In [65]: %matplotlib notebook
         import matplotlib.pyplot as plt
         import pandas as pd
         import numpy as np
         from sklearn import linear_model, datasets
         from sklearn.metrics import mean_absolute_error, mean_squared_error
         from sklearn.model_selection import train_test_split
```

Then dataset is uploaded:

```
In [16]: boston =datasets.load_boston()
```

```
In [18]: boston_X = pd.DataFrame(boston.data,columns =boston.feature_names)
         Y = boston.target
```

Next stage includes visualization:

```
In [20]: plt.scatter(boston_X['LSTAT'], Y, color = 'blue')
         plt.show()
```

Above code provide us coordinate of data (figure 3).

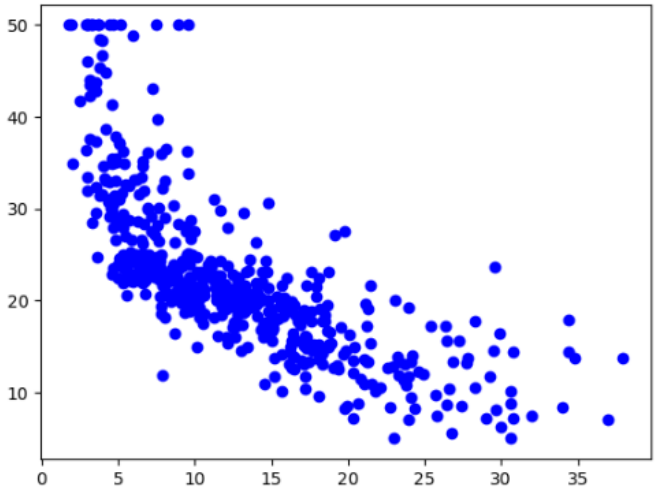


Figure 3. The location of each data.

All vector's columns and rows are defined in dataset. And also we need to train machine by data through Python code:

```
In [33]: X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size = 0.2, random_state =1)

In [34]: reg = linear_model.LinearRegression()

In [36]: reg.fit(X_train, Y_train)

Out[36]: LinearRegression()
```

After training all coefficient take value and we can predict now.

```
In [38]: Y_predict = reg.predict(X_test)

In [66]: fig = plt.figure()

         ax =fig.add_subplot(111,projection='3d')
         ax.scatter(X_test['LSTAT'],X_test['RM'], Y_test,c ='r', marker = 'o' )
         ax.scatter(X_test['LSTAT'],X_test['RM'], Y_predict,c ='g', marker = 'o' )
         ax.set_xlabel('LSTAT')
         ax.set_xlabel('RM')
         ax.set_xlabel('MEDV')
         plt.show()
```

Vol.2. Issue 1 Pages 181-187.
10.5281/zenodo.7010047

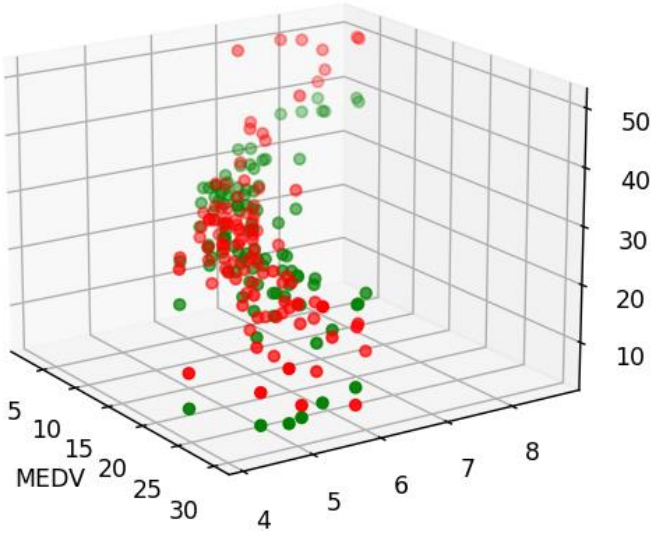


Figure 4. 3D visualization about the difference between prediction for existing value and dataset value.

However, we have to know accuracy level, and try to find it though MSE and MAE (Mean Absolute Error).

```
In [64]: print("Ortacha Kvadratik xatolik: ", mean_squared_error(Y_test,Y_predict))
         Ortacha Kvadratik xatolik:  32.08375607139396

In [59]: print("Ortacha Absolut xatolik: ", mean_absolute_error(Y_test,Y_predict))
         Ortacha Absolut xatolik:  4.50521181335465
```

CONCLUSION

Multiple Linear Regression method gives us efficiency in so many majors in order to predict the value in future. Especially, in agriculture we can use this method as an algorithm to find the result for next season. Of course, for it we need clear dataset and optimal program. There are so many programming languages in today`s technology-focused world but Python is so beneficial for Data Science and Artificial Intelligence. I have to say there are effective libraries for data analysis to develop the prediction program. And I used such kind of libraries to construct my project. I illustrate the theory of Linear Regression, Multi Linear Regression and also its implementation in agriculture. For instance, In Asian countries rely on Agriculture to survive. Because of this, I think this kind of articles and researches are more and more important.

Vol.2. Issue 1 Pages 181-187.
10.5281/zenodo.7010047

REFERENCES

1. M.Lavanya, R.Paramaswari. A Multiple Linear Regressions Model for Crop Prediction with Adam Optimizer and Neural Network. International Journal of Advanced Computer Science and Applications, Vol. 11, No. 4, 2020.
2. P.Kamath,P.Patil, S.Srilatha, Sushma, S.Sowmya. Crop yield forecasting using data mining. Global transitions proceedings. https://doi.org/10.1016/j.gltp.2021.08.008.
3. N.Raximov, O.Primqulov, B.Daminova,"Basic concepts and stages of research development on artificial intelligence", International Conference on Information Science and Communications Technologies (ICISCT), www.ieeexplore.ieee.org/document/9670085/metrics#metrics
4. Andriy Burkov, The Hundred-Page Machine Learning. 2019.
5. Khasanov Dilmurod, Tojiyev Ma'ruf,Primqulov Oybek., "Gradient Descent In Machine". International Conference on Information Science and Communications Technologies (ICISCT), https://ieeexplore.ieee.org/document/9670169.
6. M.Tojiyev,O.Primqulov,D.Xasanov, "Image segmentation in OpenCV and Python, DOI:10.5958/2249-7137.2020.01735.8.
7. Shai Shalev-Shwartz, Shai Ben-David**.** Understanding Machine Learning **–** Cambridge university press. 2014. pg. 46-85.
8. Oliver Theobald. Machine Learning for Absolute Beginners**. –** Scatterplot Press. 2017. pg.72-109.
9. John D. Kelleher, Brian Mac Namee, and Aoife D'Arcy. Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies**. –** The MIT Press. 2015. pg. 16-102.
10. Daniel Nelson. What is Gradient Descent. (internet source). 2020.