APPLICATION OF MACHINE LEARNING APPROACH FOR CRIME ANALYSIS

Nitesh Ranjan Singh, Anupriya Uniyal, Aman Kumar, Amit Verma*

School of Computer Science, University of Petroleum and Energy Studies, Dehradun
{niteshranjansingh85389, anupriyauniyal21, amanroydavpublicschool,
amit.uptu2006}@gmail.com

Abstract

Today data has become the most important asset for any organization or institution for its growth and better decision making. The field of data analytics is growing day by day to train the machine for taking efficient decisions or predictions based on previous data or datasets. There are multiple techniques to train a machine for predicting results with unknown similar data. Linear Regression is one of the simple and powerful techniques to make a machine learn about the data and predict the result for new data. Linear Regression works for linear datasets only and gives the result as a continuous value. It plots a line over data points in such a way that the mean of the squared error with each data point gets minimized. Gradient descent is used to find the most appropriate coefficients of the line. In this chapter, a mathematical model using the concept of linear regression and gradient descent is implemented with python over an authentic crime dataset. In the comparative study with the existing work, it has been found that the proposed model has reduced the cost function to 0.016.

Keywords: Machine Learning, Linear Regression, Gradient Descent, Learning Rate

*Amit Verma, amit.uptu2006@gmail.com

1. Introduction

In past few decades, many state-of-art has been done in the field of data analytic for predicting various results based on the applying machine learning algorithms [1, 2, 3, 4, 5, 6, 7] on the data-sets. In this work, the focus is on comparing the accuracy of the model with the existing linear fitting function [8]. Also doing a comparative analysis with communities and crime data set which is an existing data source in the UCI machine learning repository. Thongtae et al. [9] gave a complete overview of viable strategies on information digging for crime data analysis. Knowledge discovery as the extraction of operationally actionable output from crime data for solving crimes or explaining criminality. Analyzing this data not only helps in recognizing a feature responsible for the high crime rate but also helps in taking necessary actions for the prevention of crimes. Criminal activities [10, 11, 12, 13, 14] across the globe have created a menace in society. Every year a large volume of criminal data is generated by law enforcement organizations and it is a major challenge for them to analyze this data to implement decisions for avoiding crimes in the future.

In this chapter, various Machine Learning techniques are discussed which have been applied on linear data to predict the per capita violent crimes calculated using population. Also discuss the application of simple Linear Regression method on pre-processed linear data and minimizing the cost function using gradient descent algorithm [15, 16, 17, 18, 15, 19, 20]. Pre-processing of the data is required to maintain the quality and visualization [21] of the instance. As the irrelevant and redundant data can affect accuracy. Linear Regression [22, 23, 24, 25, 26, 27, 28] is a statistical strategy for plotting the line and is used for predictive analysis. A straight line is expected between the input variables (x) and the output variables (y) showing the connection between the values. Cost Function quantifies the error between predicted values and expected values and presents it in the form of a single real number. Gradient Descent [29, 18] is the process that uses cost function on inclinations for limiting the complexity in processing means square error [30, 31, 32, 32, 33]. These strategies are carried out in this chapter utilizing a python programming [34] tool for analyzing the data sets. The focus is to improve the quality of training data by identifying the missing, mislabelled, over-scaled and inappropriate data before using it to train a machine [35]. Almost every researcher use inbuilt function in sk-learn library to trained the machine for linear data and minimize MSE. In this chapter a mathematical approach similar to gradient descent is implemented using python programming without using sk-learn library to minimize the MSE or to find most appropriate slope and y-intercept of linear regression line. Comparing the results we found that proposed work has reduced the cost function value in comparison with already existing method.

2. Related work

In this section, various work of multiple researchers have been discussed, few procedures have been proposed in recent years for taking care of the issue of separating information from explosive information adopting various algorithms. One of such applications is that of discovering information on criminal behavior from its recorded information by examining the recurrence of happening episodes. Thongtae et al. [9] studied and gave a complete overview of viable strategies on information digging for crime data analysis. One of such proposed data frameworks was that of the 'Regional Crime Analysis Program' that is utilized to transform data into knowledge utilizing data combination. Data combination oversees, fuses, and interprets data from various sources and conquers confusion from jumbled backgrounds. Many factors affect the success of machine learning on the given task. The quality and visualization of the instance are foremost. Irrelevant and redundant data can affect accuracy. Kotsiantis et.al [21] worked on data processing for supervised learning to remove the irrelevant and mislabelled data. The Author worked on instance selection, outliers detection, etc for appropriate data pre-processing. Brodley et.al [35] remove the data without labels before using it to train the model to improve the accuracy of the results. Their initial step is to distinguish candidate instances by utilizing machine learning calculations to label instances as accurately or mistakenly labeled. In next step all the mislabeled data is removed to clean the data for better results. Prajakta et al. [10] studied algorithms like decision tree, random forest classification, linear regression and dealt with the analysis of crime data set to lower the crime rates using NumPy, pandas, and sk-learn. In [36], various machine learning algorithms for predictive analysis for measuring the temperature dataset. The Author describes the evaluation of algorithms on Hadoop [37], an open-source for spark implementation. Chen et al. [38] proposed an overall structure for crime dataset that shows connections between data mining strategies applied in criminal and intelligence analysis and the crime types at a neighborhood, public, and global levels. They utilized element extraction, association, forecast, and pattern visualization to classify every crime type. For instance, specialists can utilize neural networks in crime substance extraction, clustering in association and visualization, etc. Right now they are related to making a cybercrime data set with the help of this system. In [29], the author has explained the gradient descent using linear regression with python. Gradient Descent [39] is the process that uses cost function on inclinations for limiting the complexity in processing mean square error. These strategies are carried out in this chapter utilizing a python programming tool for analyzing the datasets. There is at present extensive excitement around the MapReduce (MR) worldview for enormous scope data analysis. Although the essential control flow of this structure has existed in parallel SQL database administration frameworks (DBMS) for more than 20 years, some have considered MR a drastically new registering model. In [40], the author has described and compared both the parameters. Lakshmi [41] worked on various machine learning techniques and analyzed them to evaluate the timer feature on various methods independent of supervised or unsupervised techniques. Caruana et al. [42], play out an observational assessment of supervised learning on high-dimensional information. Evaluation of performance metrics on three measurements: accuracy, AUC, and squared loss and study the impact of increasing dimensionality on the presentation of the learning algorithms.

Table 1: Summary of Related Work

Author Name & Year	Approach(Feature & Classifier)	Proposed work
Thongtae et al. [9], 2008	Regional Crime Analysis Program	Comprehensive surveys of efficient and effective methods on data mining for crime data analysis
S. Kotsiantis et.al. [21], 2006	Handling missing data, normalization	Present the most well know algorithms for each step of data preprocessing to improve performance.
Brodley et.al. [35], 1999	Machine learning algorithm to for data preprocessing.	Novel approach to identify and eliminating mislabeled data for supervised learning
Prajakta Yerpude et.al. [10], 2017	Algorithms like decision tree, random forest classification, linear regression	Data mining techniques are applied to crime data.
Ananthi Sheshasaayee [36], 2017	Machine learning algorithms	paper aims at understanding the reliability of three predictive methods (SVM,KNN,Decision Tree) using fused dataset.
Chen Hsinchun et.al [38], 2004	Data mining strategies like extraction, association, forecast and pattern visualization.	Discussed general framework for crime data mining.
JVN Laxshmi [29], 2016	Gradient descent, linear regression with python.	Implemented Stochastic gradient descent using linear regression.
A. Pavlo [40], 2009	MapReduce (MR) worldview, SQL DBMS	Describe and compare MapReduce (MR) paradigm for large-scale data analysis and basic control flow of this framework has existed in parallel SQL database management systems (DBMS) for over 20 years.
JVN Lakshmi [41], 2018	Machine learning algorithms	Analyse the machine learning techniques and evaluates the timer feature on various methods

3. Methodology

In this section, for implementing the mathematical model, a simple linear regression method for linear data on a clean data-set and minimization of cost function/mean squared error using a gradient descent algorithm is used.

The steps used for implementing the mathematical model as shown in Fig 1 includes reading of the data-sets, choosing input(x) and output(y) variable, cleaning of data-set like dealing with missing values, finding unusual patterns in the data-set, finding outliers using Boxplot. After getting the clean data calculation of slope and y-intercept and y-predicted for all the values of x took place. Performance metrics for the model are calculated and after that, for minimization of mean squared error, a gradient descent algorithm to the model is applied. After applying the gradient descent algorithm, choosing the best learning rate for the model was the important step and at the end, the same algorithm is repeated until its convergence.

3.1. Understand the Data-sets

Communities and crime dataset [43] used for conducting experiments. Dataset comprises of socio-economic data from the 1990 US Census, law enforcement data from the 1990 US LEMAS survey, and crime data from 1995 FBI UCR. Having 128 attributes and 1994 instances. From the large list of attributes, only two attributes are chosen for data analysis. The chosen attributes are population and violent crimes per pop. The variable violent crime per pop was calculated using the population values mentioned in the 1995 FBI UCR data. The variables included in the dataset involves community such as population which is the combination of different population of a community like urban, rural, income, etc.

3.2. Choose input and target variable

Population(x) as input variable and ViolentCrimesPerPop(y) as target variable represented in Fig 3 is chosen. Goal of this model is to reduce the cost

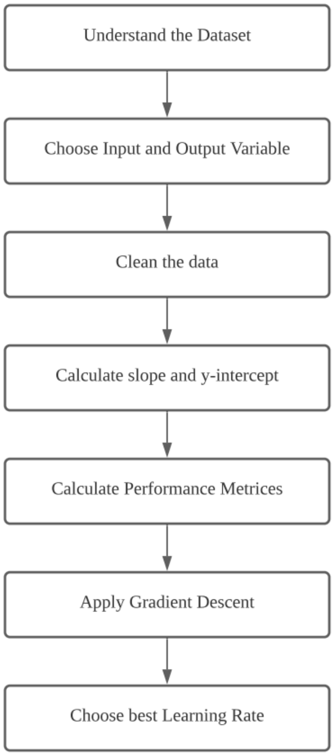

Understand the Dataset
Choose Input and Output Variable
Clean the data
Calculate slope and y-intercept
Calculate Performance Metrices
Apply Gradient Descent
Choose best Learning Rate

Figure 1: Work Flow representing the basic steps involved the proposed in Model

Table 2: Overview of dataset

Dataset Characteristics	**Number of Instances**	**Area**
Multivariate	1994	Social
Attribute Characteristics	**Number of Attributes**	**Date Donated**
Real	128	2009-07-13
Associate Tasks	**Missing Value?**	**Number of web Hits**
Regression	yes	314143

function (less than 0.0170.)

Table 3: Sample dataset

population(x)	**ViolentCrimesPerPop(y)**
0.19	0.2
0	0.67
0	0.43
0.04	0.12
0.01	0.03
0.02	0.14
0.01	0.03
0.01	0.55
0.03	0.53
0.01	0.15

3.3. Cleaning the data

At First, the cleaning of the dataset is done for better results then rows are dropped where the population(x) = 0. Boxplot as shown in Fig 2, for dropping the row containing outliers.

In this case, outliers are 0.15, 0.19, 0.25, 0.29, and 1, removed from the data sets.

3.4. Calculate slope θ_1 and y-intercept θ_0

Linear Regression Model with one variable is used.

Hypothesis: $h_\theta(x) = \theta_0 + \theta_1 x$, where $\theta_{i's}$ are the parameters.

Choose θ_0, θ_1 so that $h_\theta(x)$ is close to y for the training example(x,y). So firstly θ_1 is calculated which is slope of the regression line where r is pearson's correlation coefficient, S_y is Standard deviation of y and S_x is standard deviation

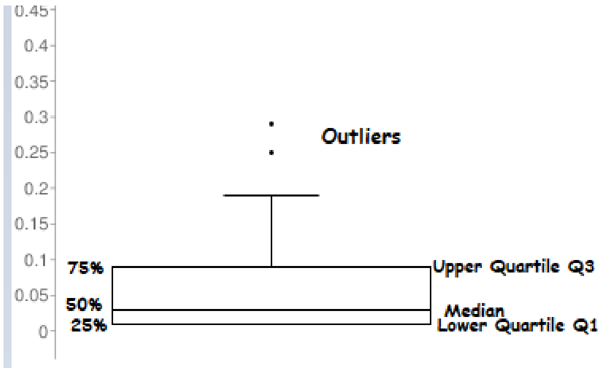


Figure 2: Boxplot

of x as shown in Eq 1, 2, 3 & 4 respectively. Where n is the number of data points, $\bar{x}$ & $\bar{y}$ representing the mean of x_i and y_i.

$$\theta_1 = r\frac{Sy}{Sx} \tag{1}$$

$$r = \frac{\sum(x-\bar{x})(y-\bar{y})}{\sqrt[2]{\sum(x-\bar{x})(y-\bar{y})}} \tag{2}$$

$$S_x = \sqrt{\frac{\sum_{i=1}^{n}(x_i-\bar{x})^2}{n-1}} \tag{3}$$

$$S_y = \sqrt{\frac{\sum_{i=1}^{n}(y_i-\bar{y})^2}{n-1}} \tag{4}$$

Using clean data, $r = 0.37383$, $S_y = 0.037383$, $S_x = 0.182116671$. Based on above results the slope θ_1**=1.830238727** then θ_1 is calculated which is slope of the regression line. $\theta_0 = \bar{y} - \theta_0\bar{x}$, where $\bar{y}$ is the mean of y and $\bar{x}$ is the mean of x. using above results the y-intercept is θ_0 **= 0.147877984**.
If x=0.13, then $h_\theta(x)$=0.3858090185

3.5. Performance Measure of Model

Here, the MSE (Mean Squared Error) = 0.028531314, But the goal is to get MSE below 0.0170. So, now using Gradient Descent for minimizing the MSE (Cost Function) as shown in Table 5

Table 4: Predicted value, Error, Squared Error

x	**y**	$h_\theta(x)$	$(y - h_\theta(x))$	$(y - h_\theta(x))^2$
0.04	0.12	0.22108753	0.101088	0.01021869
0.01	0.03	0.16618037	-0.13618	0.01854509
0.02	0.14	0.18448276	0.044483	0.00197872
0.01	0.03	0.16618037	-0.13618	0.01854509
0.01	0.55	0.16618037	0.38382	0.14731751
0.03	0.53	0.20278515	0.327215	0.10706956
0.01	0.15	0.16618037	-0.01618	0.0002618
0.13	0.24	0.38580902	0.145809	0.02126027
0.02	0.08	0.18448276	0.104483	0.01091665
0.03	0.06	0.20278515	0.142785	0.0203876

Table 5: Performance Metrics

MSE (Mean Squared Error)	0.028531314
RMSE (Root Mean Squared Error)	0.168911324
RSE (Relative Squared Error)	0.860245299
R^2	0.139754701

3.6. To minimize the MSE apply Gradient Descent Algorithm

Gradient Descent 5 is an efficient optimization algorithm that attempts to find a local or global minimum of a Cost Function(MSE) 6. It is a technique to use the derivative of the cost function to change the parameter values, to minimize the cost.

The main goal of the learning procedure is to optimize the objective Cost Function as shown in Eq. 7. Gradient Descent is one of the supervised machine learning approaches which optimizes the cost function in the learning process. The aim is to minimize the cost function and is defined as MSE to understand the weights by using the sum of squared errors amid trained set and real outcomes. Further trying to minimize the MSE using Gradient Descent. In Gradient Descent, assume that there is a straight line called Hypothesis 5 which is the best fit for all input variable (x) and output variable (y) as shown in Fig 3.

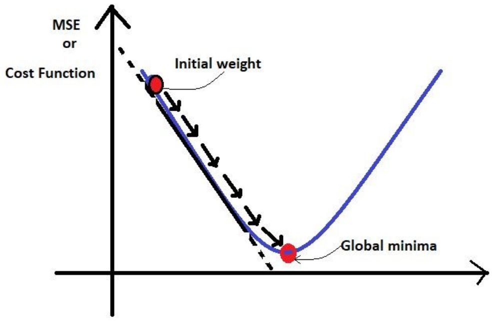


Figure 3: Gradient Descent

$$h_\theta(x) = \theta_0 + \theta_1 x \; where \; \theta_{i's} : parameters \tag{5}$$

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x_i) - y_i) \tag{6}$$

$$min_{\theta_0, \theta_1} J(\theta_0, \theta_1) \tag{7}$$

Following are the steps involved in finding the appropriate value of the coefficients θ_0 & θ_1 for the best fit as shown in Fig. 4 [44]

- Start with some θ_0, θ_1
- keep changing θ_0, θ_1 to reduce $J(\theta_0, \theta_1)$ until end up at a minimum.
- Run the Gradient Descent Minimization algorithms until convergence.

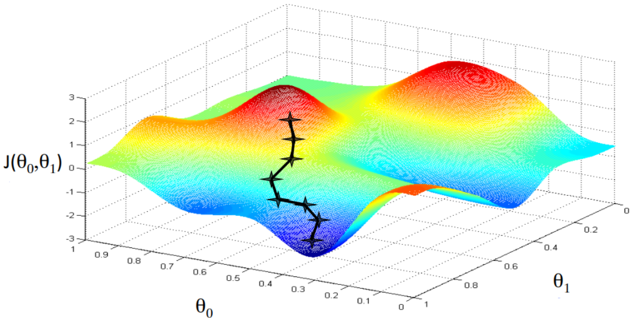


Figure 4: Figure showing θ_0, θ_1 gradually converges towards a minimum value

Gradient Descent runs iteratively to find the optimal values of the parameters corresponding to the minimum value of the given cost function, the mathematical procedure is shown below.

repeat until convergence {

$\theta_0 := \theta_0 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x_i) - y_i)$

$\theta_1 := \theta_1 - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x_i) - y_i).(x_i)$

}

update θ_0 and θ_1 simultaneously

above equation got from by doing partial derivative of $J(\theta_0, \theta_1)$

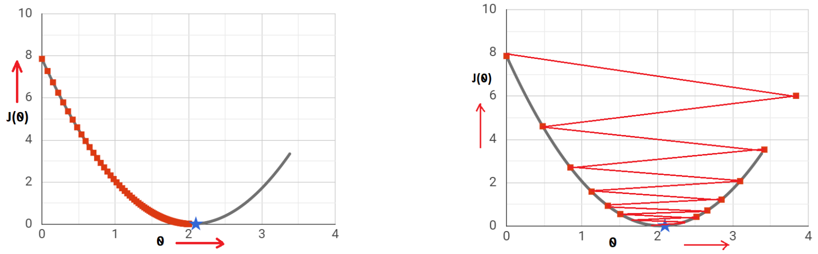
$j = 0 : \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x_i) - y_i)$

$j = 1 : \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1) = \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x_i) - y_i).(x_i)$

3.7. Choose the best Learning Rate (α) and Repeat Gradient Descent Algorithm until convergence

To solve for the gradient, iterate over the data-sets points using the new cost 'θ_0' and bias 'θ_1' values and compute the partial derivation. This new gradient tells us the slope of the cost function (MSE) at the current position and the direction. Should move to update the parameters. The size of the update is controlled by the learning rate(α).

- If 'α' is too small, 'gradient'descent' can be slow as shown in Fig. 5a.
- If 'α' is too large, 'gradient descent' can overshoot the minimum and it may fail to 'converge', or even' diverge' as shown in Fig. 5b.



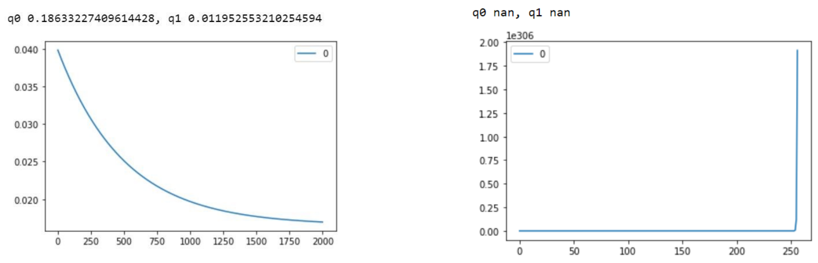
(a) Learning rate α is small

(b) Learning rate α is large

Figure 5: Variation in cost function with the learning rate α

It is very important to choose the correct learning rate α for the best fit. In mathematical model, tried different learning rates [45, 46] like small to large. As shown in Fig. 6a, a small learning rate $\alpha = 0.001$ is considered, and after 2000 iterations the minimum cost is around 0.020 till now goal is not achieved. And Fig.6b, large value of learning rate $\alpha = 5$ is considered and the cost shoots up after 250 steps. This shows that its is very important to chose the correct value of learning rate α to achieve the goal of minimum cost.

After getting the appropriate learning rate α, Gradient Descent can converge to a local minima even with the fixed learning rate. In this case, gradient descent



(a) α = 0.001

(b) α = 5

Figure 6: Cost function with α = 0.001 & α = 5

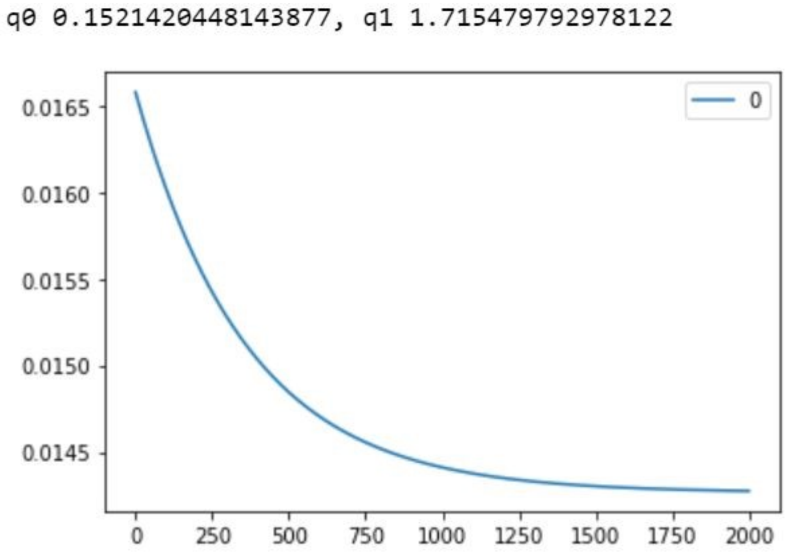
will automatically take smaller steps. So no need to decrease the learning rate over time.

4. Result and Comparative Analysis

In this section, firstly applied linear regression to the dataset. One variable denoted by x, is regarded as a predictor-features variable. The other variable denoted by y is regarded as the response-crime variable.

Target = a + b(Features) For example: High crime = a + b(Population)

Linear Regression minimizes the sum of squares of the variables predicted by linear approximation. The calculated mean squared error is 0.02853 using inbuilt function of linear regression. The calculated result is greater which can be further minimized. Therefore, proceeding with the gradient descent. Following are the steps to be implemented while using a Linear Regression predictive model in Python. We implemented the model for minimization of cost function. Firstly we took a very small learning rate i.e α = 0.001, MSE calculated after 2000 minimization steps is around 0.020 which can be further improved. Now we took high value of learning rate as α = 5 and after 2000 minimizations steps the cost shoots up to high value after 250 attempts. Experimenting with learning rate, now we took α=0.12, and gradient descent converged after 100-150 attempts. After doing multiple attempts we finally get the value of learning rate as but it can be α=1 with which the gradient descent converged after 350 iterations. And cost function is minimized upto 0.0160 which is much lesser than 0.0179 as mention in [10]



```
q0 0.1521420448143877, q1 1.715479792978122
```

Figure 7: Final Graph when α=1

4.1. Comparative Analysis

In this section, comparison of proposed work is done with existing paper [10] as shown in Table. 6. The main objective is to reduce MSE without using builtin fit function from scikit-learn library. The gradient descent in programmed using python to reduce cost function. It has been observed that our proposed work has significantly reduce the value of cost function to 0.0160 as compare to 0.0179 in [10]

Table 6: Comparative Analysis.

S. No.	Methods	Approach	Cost Function
1.	Prajakta Yerpude et.al. [10]	Python , scikit-learn and fit() function	0.0179
2.	Proposed Work	Mathematical Model implemented in python	**0.0160**

5. Conclusion and Future Scope

In this chapter, a comparative analysis of existing linear crime data using a simple mathematical approach is performed. Gradient descent algorithm is used for minimizing the cost function. The methodology is based on first cleaning data using various techniques like Boxplot thus finding outliers and then calculating the cost function and hence, minimizing the cost function of the data. Basic Python programming tools such as NumPy and pandas are used focusing on a single input variable(x) i.e. population target variable and output variable(y) i.e. ViolentCrimePerPop(y). Different values of learning rate α are put and the graph is predicted until its convergence. This is an optimization algorithm used to find the global minimum of a function and updating the parameters of the model. In the future, work can be extended for multiple regression, polynomial regression (quadratic and cubic equation), and Support Vector Machine (SVM) algorithm for the same mathematical model.

References

[1] H. Tamano, S. Nakadai, T. Araki, Optimizing multiple machine learning jobs on mapreduce, in: 2011 IEEE Third International Conference on Cloud Computing Technology and Science, IEEE, 2011, pp. 59–66.

[2] A. Manar, S. Ploix, Machine learning with python/scikit-learn-application to the estimation of occupancy and human activities, SIMUREX, 2015.

[3] W. Romsaiyud, W. Premchaiswadi, An adaptive machine learning on mapreduce framework for improving performance of large-scale data analysis

on ec2, in: 2013 Eleventh International Conference on ICT and Knowledge Engineering, IEEE, 2013, pp. 1–7.

[4] G. Bonaccorso, Machine learning algorithms, Packt Publishing Ltd, 2017.

[5] T. O. Ayodele, Types of machine learning algorithms, New advances in machine learning 3 (2010) 19–48.

[6] B. Mahesh, Machine learning algorithms-a review, International Journal of Science and Research (IJSR).[Internet] 9 (2020) 381–386.

[7] M. Mohammed, M. B. Khan, E. B. M. Bashier, Machine learning: algorithms and applications, Crc Press, 2016.

[8] L. Fahrmeir, T. Kneib, S. Lang, B. Marx, Regression, Springer, 2007.

[9] P. Thongtae, S. Srisuk, An analysis of data mining applications in crime domain, in: 2008 IEEE 8th International Conference on Computer and Information Technology Workshops, IEEE, 2008, pp. 122–126.

[10] P. Yerpude, Predictive modelling of crime data set using data mining, International Journal of Data Mining & Knowledge Management Process (IJDKP) Vol 7.

[11] L. McClendon, N. Meghanathan, Using machine learning algorithms to analyze crime data, Machine Learning and Applications: An International Journal (MLAIJ) 2 (1) (2015) 1–12.

[12] X. Zhang, L. Liu, L. Xiao, J. Ji, Comparison of machine learning algorithms for predicting crime hotspots, IEEE Access 8 (2020) 181302–181310.

[13] M. L. Rich, Machine learning, automated suspicion algorithms, and the fourth amendment, University of Pennsylvania Law Review (2016) 871–929.

[14] R. Ch, T. R. Gadekallu, M. H. Abidi, A. Al-Ahmari, Computational system to classify cyber crime offenses using machine learning, Sustainability 12 (10) (2020) 4087.

[15] S. Hochreiter, A. S. Younger, P. R. Conwell, Learning to learn using gradient descent, in: International Conference on Artificial Neural Networks, Springer, 2001, pp. 87–94.

[16] S. Ray, A quick review of machine learning algorithms, in: 2019 International conference on machine learning, big data, cloud and parallel computing (COMITCon), IEEE, 2019, pp. 35–39.

[17] J. Keuper, F.-J. Pfreundt, Asynchronous parallel stochastic gradient descent: A numeric core for scalable distributed machine learning algorithms, in: Proceedings of the Workshop on Machine Learning in High-Performance Computing Environments, 2015, pp. 1–11.

[18] L. Bottou, Large-scale machine learning with stochastic gradient descent, in: Proceedings of COMPSTAT'2010, Springer, 2010, pp. 177–186.

[19] D. Steinkraus, I. Buck, P. Simard, Using gpus for machine learning algorithms, in: Eighth International Conference on Document Analysis and Recognition (ICDAR'05), IEEE, 2005, pp. 1115–1120.

[20] Y. Ying, M. Pontil, Online gradient descent learning algorithms, Foundations of Computational Mathematics 8 (5) (2008) 561–596.

[21] S. B. Kotsiantis, D. Kanellopoulos, P. E. Pintelas, Data preprocessing for supervised leaning, International journal of computer science 1 (2) (2006) 111–117.

[22] O. O. Aalen, A linear regression model for the analysis of life times, Statistics in medicine 8 (8) (1989) 907–925.

[23] I. Naseem, R. Togneri, M. Bennamoun, Linear regression for face recognition, IEEE transactions on pattern analysis and machine intelligence 32 (11) (2010) 2106–2112.

[24] M. Zinkevich, M. Weimer, A. J. Smola, L. Li, Parallelized stochastic gradient descent., in: NIPS, Vol. 4, Citeseer, 2010, p. 4.

[25] D. Maulud, A. M. Abdulazeez, A review on linear regression comprehensive in machine learning, Journal of Applied Science and Technology Trends 1 (4) (2020) 140–147.

[26] J. Chen, K. de Hoogh, J. Gulliver, B. Hoffmann, O. Hertel, M. Ketzel, M. Bauwelinck, A. Van Donkelaar, U. A. Hvidtfeldt, K. Katsouyanni, et al., A comparison of linear regression, regularization, and machine learning algorithms to develop europe-wide spatial models of fine particles and nitrogen dioxide, Environment international 130 (2019) 104934.

[27] S. Kavitha, S. Varuna, R. Ramya, A comparative analysis on linear regression and support vector regression, in: 2016 online international conference on green engineering and technologies (IC-GET), IEEE, 2016, pp. 1–5.

[28] T. Doan, J. Kalita, Selecting machine learning algorithms using regression models, in: 2015 IEEE International Conference on Data Mining Workshop (ICDMW), IEEE, 2015, pp. 1498–1505.

[29] J. Lakshmi, Stochastic gradient descent using linear regression with python, International Journal on Advanced Engineering Research and Applications 2 (7) (2016) 519–524.

[30] D. M. Allen, Mean square error of prediction as a criterion for selecting variables, Technometrics 13 (3) (1971) 469–475.

[31] C. J. Willmott, K. Matsuura, Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance, Climate research 30 (1) (2005) 79–82.

[32] T. Chai, R. R. Draxler, Root mean square error (rmse) or mean absolute error (mae)?–arguments against avoiding rmse in the literature, Geoscientific model development 7 (3) (2014) 1247–1250.

[33] C. M. Theobald, Generalizations of mean square error applied to ridge regression, Journal of the Royal Statistical Society: Series B (Methodological) 36 (1) (1974) 103–106.

[34] S. Robinson, H. Baayen, Beginning python programming for language research.

[35] C. E. Brodley, M. A. Friedl, Identifying mislabeled training data, Journal of artificial intelligence research 11 (1999) 131–167.

[36] H. Naganathan, S. P. Seshasayee, J. Kim, W. K. Chong, J.-S. Chou, Wildfire predictions: Determining reliable models using fused dataset, Global Journal of Computer Science and Technology.

[37] T. Asha, U. Shravanthi, N. Nagashree, M. Monika, Building machine learning algorithms on hadoop for bigdata, International Journal of Engineering and Technology 3 (2) (2013) 143–147.

[38] H. Chen, W. Chung, J. J. Xu, G. Wang, Y. Qin, M. Chau, Crime data mining: a general framework and some examples, computer 37 (4) (2004) 50–56.

[39] S. Ruder, An overview of gradient descent optimization algorithms, arXiv preprint arXiv:1609.04747.

[40] A. Pavlo, E. Paulson, A. Rasin, D. J. Abadi, D. J. DeWitt, S. Madden, M. Stonebraker, A comparison of approaches to large-scale data analysis, in: Proceedings of the 2009 ACM SIGMOD International Conference on Management of data, 2009, pp. 165–178.

[41] J. Lakshmi, Machine learning techniques using python for data analysis in performance evaluation, International Journal of Intelligent Systems Technologies and Applications 17 (1-2) (2018) 3–18.

[42] R. Caruana, N. Karampatziakis, A. Yessenalina, An empirical evaluation of supervised learning in high dimensions, in: Proceedings of the 25th international conference on Machine learning, 2008, pp. 96–103.

[43] https://archive.ics.uci.edu/ml/datasets/communities+and+crime.

[44] https://medium.com/@dbcerigo/on-why-gradient-descent-is-even-needed-25160197a635.

[45] V. Plagianakos, G. Magoulas, M. Vrahatis, Learning rate adaptation in stochastic gradient descent, in: Advances in convex analysis and global optimization, Springer, 2001, pp. 433–444.

[46] M. D. Zeiler, Adadelta: an adaptive learning rate method, arXiv preprint arXiv:1212.5701.