On the Pulling Linear Regression and Its Applications in Digital Mammograms

NAHATAI TEPKASETKUL[1], WEENAKORN IEOSANURAK[1],
THANAPONG INTHARAH[2], WATCHARIN KLONGDEE[1]
[1]Department of Mathematics, Faculty of Science, Khon Kaen University, THAILAND
[2]Department of Statistics, Faculty of Science, Khon Kaen University, THAILAND

Abstract: - Regression analysis is a statistical approach used to investigate the correlations between variables, especially linear regression, that is a simple but effective approach for analyzing the relationship between a dependent variable and one independent variable. Since it has limitations based on the assumption that the mean of the noise should be zero, there are still some areas where it may be improved. In this article, we introduce a novel data fitting algorithm called the pulling linear regression, which is separated into two types: the line-pulling linear regression and the band-pulling linear regression. The method is developed from linear regression, which can create the regression line from the function that uses noise with various distributions. The result demonstrates that the sequence of sum square errors of the pulling linear regression is convergent. Moreover, we have a numerical example to show that the performance of the proposed algorithm is better than that of linear regression when the mean of the noise is not zero. And the last, we have an application to smooth the boundary of the pectoral muscle in digital mammograms. We found that the regression line of the proposed algorithm can do better than the linear regression when we would like to remove only the muscle part.

Key-Words: - linear regression, pulling linear regression, sum square error, root mean square error

Received: April 29, 2022. Revised: January 15, 2023. Accepted: February 9, 2023. Published: March 2, 2023.

1 Introduction

Least squares linear regression was performed by Legendre and Gauss for the prediction of planetary movement, [1]. Regression analysis is a statistical technique for examining correlations between variables used in numerous domains, including economics, engineering, physical science, biological science, social science, and medicine, among many others, [2].

Linear regression (LR) is a powerful and adaptable method for dealing with regression difficulties. The model definition, model estimation, statistical inference, model diagnosis, variable selection, and prediction are described comprehensively, [3]-[6]. Therefore, researchers are quite interested in the trend in LR models. For example, Pérez-Domínguez et al., [7], offered a contribution using linear regression and applied Dimensional Analysis (DA) to solve instability and error problems of the data transformation. Jokubaitis and Leipus, [8], studied the asymptotic normality in a high-dimensional linear regression where the covariance matrix of the regression variables has a KMS structure. Al-Kandari et al., [9], introduced a strategy for accounting for uncertainty in the residuals of the linear regression model using fuzzy statistics. Liu and Chen, [10], improved the h value for fuzzy linear regression analysis using symmetric triangular fuzzy numbers and the least fuzziness criterion. Kabán, [11], provide a new analysis of compressive least squares regression that eliminates a false $\log N$ component, where N is the total number of training points. Additionally, several researchers have developed methods related to linear regression. For example, linear mixed models are used by Yi and Tang, [12], Ahn, Zhang and Lu, [13], and multiple linear regression is used by Uyanık and Güler, [14], Liu et al., [15], Li, He and Liu, [16].

A linear regression model is defined by $y = ax + b + \eta$ where a is a scaling parameter, b is a location parameter, y is the dependent or response variable, x is the independent or predictor variable, and the random variable η is the error term in the model, [17]-[19]. The linear regression is carried out under the assumption that η has a normal distribution with a mean and variance of zero and σ^2, respectively, i.e., $E[\eta] = 0$ and $Var(\eta) = \sigma^2$.

In this article, we will consider the scenario where η has an alternative distribution or when

$E[\eta] \neq 0$, which reduces the influence of partial observations by deviating from the regression line, $y = ax + b$. The two novel algorithms are line-pulling linear regression (LPR) and band-pulling linear regression (BPR), which are presented in the next section.

The remainder of the article is structured as follows: the LPR and BPR algorithms are introduced in Section 2. Section 3 describes a mathematical proof of some property. Next, the numerical results of our algorithms are illustrated and discussed in Section 4. Section 5 shows how the application is used to remove the pectoral muscle. Finally, conclusions and some suggestions are drawn and presented in Section 6.

2 Description of LPR and BPR

We introduce two novel data-fitting algorithms: line-pulling linear regression (LPR) and band-pulling linear regression (BPR). These algorithms are defined as follows:

Let $D^{(0)} = \left\{\left(x_1, y_1^{(0)}\right), \left(x_2, y_2^{(0)}\right), \dots, \left(x_n, y_n^{(0)}\right)\right\}$ be an initial data such that x_i are distinct and $0 \leq p_1, p_2 \leq 1$. The procedure for the LPR algorithm is the following. Set $D_{LPR}^{(0)} = D^{(0)}$.

(i) Consider the k^{th} iteration, $k = 1,2, \dots$. We get the linear regression $f^{(k)}(x) = \alpha^{(k)}x + \beta^{(k)}$ of the data $D_{LPR}^{(k-1)}$, where

$$\left(\alpha^{(k)}, \beta^{(k)}\right) = \underset{(\alpha,\beta)}{\operatorname{argmin}} \sum_{i=1}^{n} \left(y_i^{(k-1)} - (\alpha x_i + \beta)\right)^2.$$

Denote $u_i^{(k)} = \alpha^{(k)}x_i + \beta^{(k)}, i = 1,2, \dots, n.$

(ii) Define a band

$$M^{(k)} = \left\{(x,y) \middle| -d_2^{(k)} \leq y - f^{(k)}(x) \leq d_1^{(k)}\right\},$$

where $d_1^{(k)} = p_1 \max_i \left\{y_i^{(k-1)} - u_i^{(k)}\right\}$, and $d_2^{(k)} = p_2 \max_i \left\{u_i^{(k)} - y_i^{(k-1)}\right\}$.

(iii) Update the data $D_{LPR}^{(k)}$ given by

$$D_{LPR}^{(k)} = \left\{\left(x_1, y_1^{(k)}\right), \left(x_2, y_2^{(k)}\right), \dots, \left(x_n, y_n^{(k)}\right)\right\},$$

such that for each $i = 1, 2, \dots, n$,

$$\left(x_i, y_i^{(k)}\right) = \begin{cases} \left(x_i, y_i^{(k-1)}\right); & \left(x_i, y_i^{(k-1)}\right) \in M^{(k)}, \\ \left(x_i, u_i^{(k)}\right); & \left(x_i, y_i^{(k-1)}\right) \notin M^{(k)}. \end{cases} \quad (1)$$

(iv) Return to step 1 to repeat until the error values

$$\sum_{i=1}^{n} \left(y_i^{(k-1)} - \left(\alpha^{(k)}x_i + \beta^{(k)}\right)\right)^2 < \varepsilon,$$

where $\varepsilon > 0$ is a fixed value.

Following that, we shall introduce the BPR algorithm. Set $D_{BPR}^{(0)} = D^{(0)}$. The steps of the algorithm are defined the same as the LPR algorithm, except for step (iii), replaced by step (iii*) as follows.

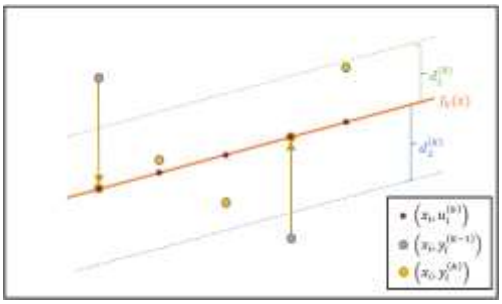
(iii*) Update the data $D_{BPR}^{(k)}$ given by

$$D_{BPR}^{(k)} = \left\{\left(x_1, y_1^{(k)}\right), \left(x_2, y_2^{(k)}\right), \dots, \left(x_n, y_n^{(k)}\right)\right\},$$

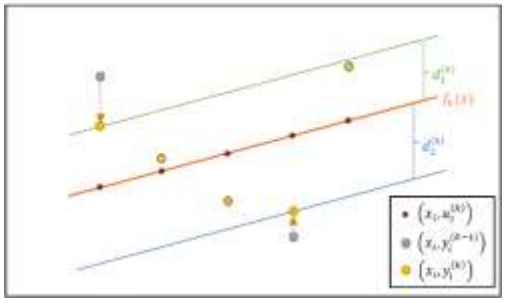
such that for each $i = 1, 2, \dots, n$,

$$\left(x_i, y_i^{(k)}\right) = \begin{cases} \left(x_i, u_i^{(k)} + d_1^{(k)}\right); & \left(x_i, y_i^{(k-1)}\right) \notin M^{(k)}, \\ & y_i^{(k-1)} > u_i^{(k)} + d_1^{(k)}, \\ \left(x_i, y_i^{(k-1)}\right); & \left(x_i, y_i^{(k-1)}\right) \in M^{(k)}, \\ \left(x_i, u_i^{(k)} - d_2^{(k)}\right); & \left(x_i, y_i^{(k-1)}\right) \notin M^{(k)}, \\ & y_i^{(k-1)} < u_i^{(k)} - d_2^{(k)}. \end{cases} \quad (2)$$

The two above algorithms are different for updating data. The LPR algorithm is used to pull the points outside the band toward the regression line, but the BPR algorithm is used to pull those points toward the boundary of the band as illustrated in Fig. 1.



(a) LPR



(b) BPR

Fig. 1: The proposed algorithm.

The following is an example of how to understand our algorithm.

Example 1. Let $p_1 = p_2 = 0.5$. We take $y_i = x_i + 1 + \eta_i$ where $x_i = i$ and $\eta_i \in (-1,1)$ is a uniform noise.

We obtain the initial data for our example, $D^{(0)} = \{(1,1.88), (2,2.76), (3,4.53), (4,5.59), (5,5.37)\}$, (see, Fig. 2).

In the first iteration, we obtain the linear least square regression:

$$f^{(1)}(x) = 0.98x + 1.08.$$

The points $(1,2.06), (2,3.05), (3,4.03), (4,5.01), (5,5.99)$ that lie on the regression $y = f^{(1)}(x)$ and

$$d_1^{(1)} = 0.5 \max\{(1.88 - 2.06), (2.76 - 3.05), (4.53 - 4.03), (5.59 - 5.01), (5.37 - 5.99)\} = 0.29$$

$$d_2^{(1)} = 0.5 \max\{(2.06 - 1.88), (3.05 - 2.76), (4.03 - 4.53), (5.01 - 5.59), (5.99 - 5.37)\} = 0.31$$

Then, $M^{(1)} = \{(x,y) | -0.31 \le y - f^{(1)}(x) \le 0.29\}$ is shown in Fig. 3.

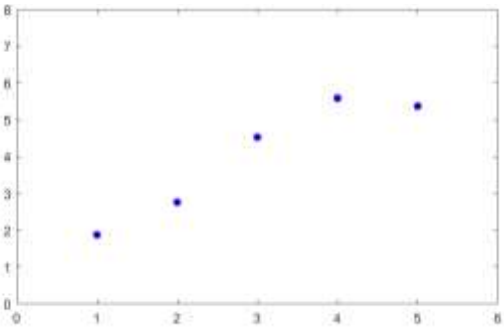


Fig. 2: Initial data.

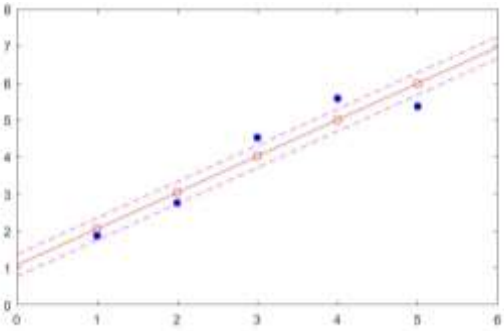


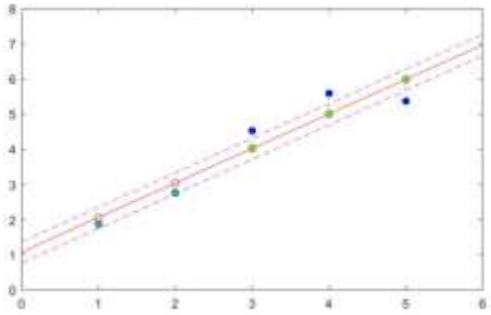
Fig. 3: The band $M^{(1)}$ for the example.

In Fig. 3, we found that points $(x_3, y_3), (x_4, y_4)$ and (x_5, y_5) are outside of the band $M^{(1)}$. From equations (1) and (2), we will update the data as follows:

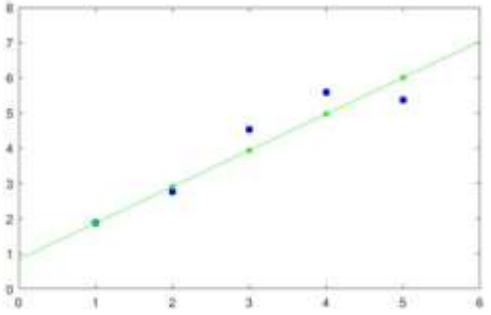
LPR: We get the range of $D_{LPR}^{(1)}$ as $\left\{y_1^{(0)}, y_2^{(0)}, u_3^{(1)}, u_4^{(1)}, u_5^{(1)}\right\}$ and obtain the updated data, $D_{LPR}^{(1)} = \{(1,1.88), (2,2.76), (3,4.03), (4,5.01), (5,5.99)\}$, (see, Fig. 4(a)).

BPR: We get the range of $D_{BPR}^{(1)}$ as $\left\{y_1^{(0)}, y_2^{(0)}, u_3^{(1)} + d_1^{(1)}, u_4^{(1)} + d_1^{(1)}, u_5^{(1)} - d_2^{(1)}\right\}$ and obtain the updated data, $D_{BPR}^{(1)} = \{(1,1.88), (2,2.76), (3,4.32), (4,5.30), (5,5.68)\}$, (see, Fig. 5(a)).

The results of the LPR and BPR algorithms are shown in Tables 1 and 2, respectively. In the 6^{th} iteration for LPR and the 11^{th} iteration for BPR, we obtain the regressions $f(x) = 1.03x + 0.84$ and $f(x) = 1.04x + 0.84$, respectively, for $\varepsilon = 0.5 \times 10^{-5}$, shown in Fig. 4(b) and Fig. 5(b).

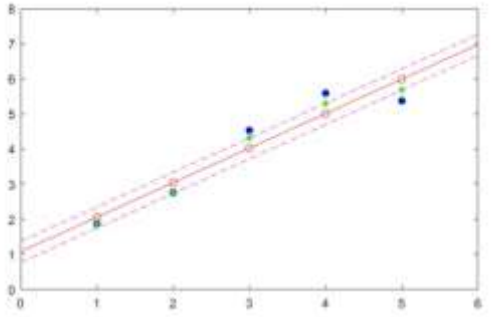


(a) the updated data for the 1^{st} iteration

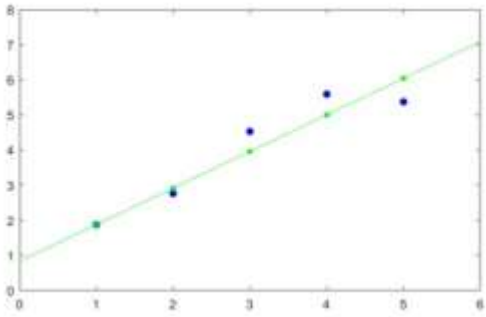


(b) the data fitting for 6^{th} iteration

Fig. 4: The result for the LPR algorithm.



(a) the updated data for the 1^{st} iteration



(b) the data fitting for 11^{th} iteration

Fig. 5: The result for the BPR algorithm.

Table 1. The results of the LPR algorithm for $D^{(0)} = \{(1,1.88), (2,2.76), (3,4.53), (4,5.59), (5,5.37)\}$.

k	$u_i^{(k)} = a^{(k)}x_i + b^{(k)}$		Data					Error
	$a^{(k)}$	$b^{(k)}$	$y_1^{(k)}$	$y_2^{(k)}$	$y_3^{(k)}$	$y_4^{(k)}$	$y_5^{(k)}$	
0			1.880	2.760	4.530	5.590	5.370	
1	0.981	1.083	1.880	2.760	4.026	5.007	5.988	0.109×10
2	1.046	0.793	1.880	2.886	3.932	5.007	5.988	0.284×10^{-1}
3	1.034	0.838	1.880	2.905	3.932	4.972	6.006	0.200×10^{-2}
4	1.032	0.843	1.875	2.905	3.939	4.972	6.003	0.865×10^{-4}
5	1.032	0.842	1.874	2.907	3.939	4.971	6.003	0.523×10^{-5}
6	1.032	0.842						0.174×10^{-6}

Table 2. The results of the BPR algorithm for $D^{(0)} = \{(1,1.88), (2,2.76), (3,4.53), (4,5.59), (5,5.37)\}$.

k	$u_i^{(k)} = a^{(k)}x_i + b^{(k)}$		Data					Error
	$a^{(k)}$	$b^{(k)}$	$y_1^{(k)}$	$y_2^{(k)}$	$y_3^{(k)}$	$y_4^{(k)}$	$y_5^{(k)}$	
0			1.880	2.760	4.530	5.590	5.370	
1	0.981	1.083	1.880	2.760	4.318	5.299	5.679	0.109×10
2	1.014	0.946	1.880	2.806	4.152	5.166	5.847	0.362×10^{0}
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
10	1.398	0.840	1.880	2.919	3.961	5.000	6.038	0.128×10^{-4}
11	1.398	0.840						0.328×10^{-5}

3 Main Results

Definition 1. Let $p_1, p_2 \in [0,1]$, $D^{(0)} = \left\{\left(x_1, y_1^{(0)}\right), \left(x_2, y_2^{(0)}\right), \ldots, \left(x_n, y_n^{(0)}\right)\right\}$ be an initial data such that x_i are distinct. The sum square error (SSE) of k^{th} iteration for LPR with respect to $\left(D^{(0)}; p_1, p_2\right)$ is given by

$$SSE_{LPR}^{(k)}\left(D^{(0)}; p_1, p_2\right) = \sum_{i=1}^{n}\left(y_i^{(k-1)} - \left(\alpha^{(k)}x_i + \beta^{(k)}\right)\right)^2, \tag{3}$$

where

$$\left(\alpha^{(k)}, \beta^{(k)}\right) = \underset{(\alpha,\beta)}{\operatorname{argmin}} \sum_{i=1}^{n}\left(y_i^{(k-1)} - (\alpha x_i + \beta)\right)^2,$$

and $y_i^{(k)}$ is an updated value as mentioned in LPR.

Similarly, the SSE of k^{th} iteration for BPR with respect to $\left(D^{(0)}; p_1, p_2\right)$ is given by

$$SSE_{BPR}^{(k)}\left(D^{(0)}; p_1, p_2\right) = \sum_{i=1}^{n}\left(y_i^{(k-1)} - \left(\alpha^{(k)}x_i + \beta^{(k)}\right)\right)^2, \tag{4}$$

where

$$\left(\alpha^{(k)}, \beta^{(k)}\right) = \underset{(\alpha,\beta)}{\operatorname{argmin}} \sum_{i=1}^{n}\left(y_i^{(k-1)} - (\alpha x_i + \beta)\right)^2,$$

and $y_i^{(k)}$ is an updated value as mentioned in BPR.

In specific case, $p_1 = p_2 = 0$. We observe that $d_1^{(1)} = d_2^{(1)} = 0$, i.e., $M^{(1)} = \{(x,y)|y = f^{(1)}(x)\}$. That is, each point in $D^{(0)}$ is pulled into the line $y = f^{(1)}(x)$. Therefore, $SSE_{LPR}^{(k)}\left(D^{(0)}; p_1, p_2\right) = 0$ and $SSE_{BPR}^{(k)}\left(D^{(0)}; p_1, p_2\right) = 0$, $k \geq 2$.

In the opposite case, $p_1 = p_2 = 1$. We then get $d_1^{(k)} = \max_i \left\{ y_i^{(k-1)} - u_i^{(k)} \right\}$ and $d_2^{(k)} = \max_i \left\{ u_i^{(k)} - y_i^{(k-1)} \right\}$. It means that all points are in $M^{(k)}$, i.e., $D^{(k)} = D^{(0)}$. Therefore, $SSE_{LPR}^{(k)}\left(D^{(0)}; p_1, p_2\right) = SSE_{LPR}^{(1)}\left(D^{(0)}; p_1, p_2\right)$ and $SSE_{BPR}^{(k)}\left(D^{(0)}; p_1, p_2\right) = SSE_{BPR}^{(1)}\left(D^{(0)}; p_1, p_2\right)$, $k \geq 2$.

From Tables 1 and 2, we observe that the sum square errors are decreasing for both algorithms. This leads to the following property.

Lemma 1. Let $D^{(0)} = \left\{ \left(x_1, y_1^{(0)}\right), \left(x_2, y_2^{(0)}\right), \ldots, \left(x_n, y_n^{(0)}\right) \right\}$ be an initial data such that x_i are distinct. If $p_1, p_2 \in [0,1]$, the sequences $SSE_{LPR}^{(k)}\left(D^{(0)}; p_1, p_2\right)$ and $SSE_{BPR}^{(k)}\left(D^{(0)}; p_1, p_2\right)$ are decreasing on k.

Proof. For the k^{th} iteration where $k = 1, 2, 3, \ldots$
Let $u_i^{(k)} = \alpha^{(k)} x_i + \beta^{(k)}, i = 1,2, \ldots, n$, such that,

$$\left(\alpha^{(k)}, \beta^{(k)}\right) = \underset{(\alpha,\beta)}{\operatorname{argmin}} \sum_{i=1}^{n} \left(y_i^{(k-1)} - (\alpha x_i + \beta) \right)^2$$

Case *LPR:* From equation (1), we get

$$y_i^{(k)} = \begin{cases} y_i^{(k-1)}, & \left(x_i, y_i^{(k-1)}\right) \in M^{(k)}, \\ u_i^{(k)}, & \left(x_i, y_i^{(k-1)}\right) \notin M^{(k)}. \end{cases}$$

Thus,

$$\begin{aligned} &\left(y_i^{(k)} - u_i^{(k)}\right)^2 \\ &= \begin{cases} \left(y_i^{(k-1)} - u_i^{(k)}\right)^2, & \left(x_i, y_i^{(k-1)}\right) \in M^{(k)}, \\ 0, & \left(x_i, y_i^{(k-1)}\right) \notin M^{(k)}. \end{cases} \\ &\leq \left(y_i^{(k-1)} - u_i^{(k)}\right)^2. \end{aligned}$$

Next, consider equation (3),

$$\begin{aligned} &SSE_{LPR}^{(k+1)}\left(D^{(0)}; p_1, p_2\right) \\ &= \sum_{i=1}^{n} \left(y_i^{(k)} - \left(\alpha^{(k+1)} x_i + \beta^{(k+1)}\right) \right)^2 \\ &= \min_{(\alpha,\beta)} \sum_{i=1}^{n} \left(y_i^{(k)} - (\alpha x_i + \beta) \right)^2 \\ &\leq \sum_{i=1}^{n} \left(y_i^{(k)} - \left(\alpha^{(k)} x_i + \beta^{(k)}\right) \right)^2 \\ &= \sum_{i=1}^{n} \left(y_i^{(k)} - u_i^{(k)} \right)^2 \\ &\leq \sum_{i=1}^{n} \left(y_i^{(k-1)} - u_i^{(k)} \right)^2 \\ &= E_{LPR}^{(k)}\left(D^{(0)}; p_1, p_2\right). \end{aligned}$$

Case *BPR:* From equation (2), we get

$$y_i^{(k)} = \begin{cases} u_i^{(k)} + d_1^{(k)}, & \left(x_i, y_i^{(k-1)}\right) \notin M^{(k)}, \\ & \quad y_i^{(k-1)} > u_i^{(k)} + d_1^{(k)}, \\ y_i^{(k-1)}, & \left(x_i, y_i^{(k-1)}\right) \in M^{(k)}, \\ u_i^{(k)} - d_2^{(k)}, & \left(x_i, y_i^{(k-1)}\right) \notin M^{(k)}, \\ & \quad y_i^{(k-1)} < u_i^{(k)} - d_2^{(k)}. \end{cases}$$

Thus,

$$\begin{aligned} &\left(y_i^{(k)} - u_i^{(k)}\right)^2 \\ &= \begin{cases} \left(d_1^{(k)}\right)^2, & 0 \leq d_1^{(k)} < y_i^{(k-1)} - u_i^{(k)}, \\ \left(y_i^{(k-1)} - u_i^{(k)}\right)^2, & \left(x_i, y_i^{(k-1)}\right) \in M^{(k)}, \\ \left(-d_2^{(k)}\right)^2, & 0 \leq d_2^{(k)} < u_i^{(k)} - y_i^{(k-1)}, \end{cases} \\ &\leq \left(y_i^{(k-1)} - u_i^{(k)}\right)^2. \end{aligned}$$

Next, consider equation (4),

$$\begin{aligned} &SSE_{BPR}^{(k+1)}\left(D^{(0)}; p_1, p_2\right) \\ &= \sum_{i=1}^{n} \left(y_i^{(k)} - \left(\alpha^{(k+1)} x_i + \beta^{(k+1)}\right) \right)^2 \\ &= \min_{(\alpha,\beta)} \sum_{i=1}^{n} \left(y_i^{(k)} - (\alpha x_i + \beta) \right)^2 \\ &\leq \sum_{i=1}^{n} \left(y_i^{(k)} - \left(\alpha^{(k)} x_i + \beta^{(k)}\right) \right)^2 \\ &= \sum_{i=1}^{n} \left(y_i^{(k)} - u_i^{(k)} \right)^2 \\ &\leq \sum_{i=1}^{n} \left(y_i^{(k-1)} - u_i^{(k)} \right)^2 \\ &= E_{BPR}^{(k)}\left(D^{(0)}; p_1, p_2\right). \end{aligned}$$

This completes the proof. ■

From equations (3) and (4), it is obvious that the sequences $SSE_{LPR}^{(k)}\left(D^{(0)}; p_1, p_2\right)$ and $SSE_{BPR}^{(k)}\left(D^{(0)}; p_1, p_2\right)$ are nonnegative sequences, that is, they have the lower bound to be zero, and, by Lemma 1, they are monotone decreasing, this leads to the following theorem.

Theorem 2. Let $D^{(0)}$ be an initial data. If $p_1, p_2 \in [0,1]$, the sequences $SSE_{LPR}^{(k)}\left(D^{(0)}; p_1, p_2\right)$ and $SSE_{BPR}^{(k)}\left(D^{(0)}; p_1, p_2\right)$ are convergent on k.

4 Numerical Examples

This section shows the numerical example using the initial data generated from the linear function $y = x + 1$ and noises (uniform distribution $U(a, b)$, normal distribution $N(a, b)$, and gamma distribution $G(a, b)$ *). We also use various values of p_1 and p_2 to compare the root mean square error and the number of iterations of the proposed algorithm.

In Table 3, we generate y_i by the equation

$$y_i = (i - 11) + 1 + \eta_i,$$

where η_i is a generated noise, and $i = 1,2, \dots ,21$. Thus, the root mean square error (RMSE) is given by

$$\text{RMSE}^{(k)} = \sqrt{\frac{1}{21}\sum_{i=1}^{21}\left(u_i^{(k)} - y_i\right)^2},$$

where $u_i^{(k)}$ is obtained from the linear least square of k^{th} iteration of the proposed algorithms. We summarise the result in Table 3 as follows.

- $\eta_i \sim U(-10,0)$, the LPR algorithm with $p_1 = 1$ and $p_2 = 0.25$ has the minimum value of RMSE with 59^{th} iteration.
- $\eta_i \sim U(-7.5,2.5)$, the BPR algorithm with $p_1 = 0.75$ and $p_2 = 0.25$ has the minimum value of RMSE with 15^{th} iteration.
- $\eta_i \sim U(-5,5)$, the LPR algorithm with $p_1 = 0.75$ and $p_2 = 0.75$ has the minimum value of RMSE with 11^{th} iteration.
- $\eta_i \sim U(-2.5,7.5)$, the BPR algorithm with $p_1 = 0.25$ and $p_2 = 0.75$ has the minimum value of RMSE with 17^{th} iteration.
- $\eta_i \sim U(0,10)$, the LPR algorithm with $p_1 = 0.25$ and $p_2 = 1$ has the minimum value of RMSE with 71^{th} iteration.
- $\eta_i \sim N(-4,1)$, the LPR and BPR algorithms with $p_1 = 1$ and $p_2 = 0$ have the minimum value of RMSE with 52^{th} iteration.

Table 3. The root mean square error and the number of iterations of the LPR and BPR.

Noises	Algorithms	Proportions											
		$p_1 = 0$ $p_2 = 0$	$p_1 = 0.25$ $p_2 = 0.25$	$p_1 = 0.50$ $p_2 = 0.50$	$p_1 = 0.75$ $p_2 = 0.75$	$p_1 = 0$ $p_2 = 1$	$p_1 = 0.25$ $p_2 = 1$	$p_1 = 0.25$ $p_2 = 0.75$	$p_1 = 0.50$ $p_2 = 0.75$	$p_1 = 1$ $p_2 = 0$	$p_1 = 1$ $p_2 = 0.25$	$p_1 = 0.75$ $p_2 = 0.25$	$p_1 = 0.75$ $p_2 = 0.50$
$U(-10,0)$	LPR	3.555 (2)	3.623 (5)	3.627 (10)	3.663 (17)	8.414 (64)	8.414 (64)	4.259 (7)	3.966 (11)	0.144 (58)	0.143 (59)	2.860 (9)	3.198 (11)
	BPR	3.555 (2)	3.649 (8)	3.651 (14)	3.644 (31)	8.414 (64)	8.410 (88)	5.022 (20)	4.409 (25)	0.144 (58)	0.146 (77)	2.133 (19)	2.758 (23)
$U(-7.5,2.5)$	LPR	1.563 (2)	1.562 (5)	1.509 (9)	1.269 (14)	5.819 (74)	5.819 (74)	2.202 (7)	1.676 (10)	2.357 (88)	2.357 (88)	0.401 (9)	0.889 (9)
	BPR	1.563 (2)	1.511 (8)	1.464 (14)	1.425 (34)	5.819 (74)	5.815 (93)	3.266 (16)	2.606 (22)	2.357 (88)	2.330 (122)	0.174 (15)	0.337 (19)
$U(-5,5)$	LPR	0.372 (2)	0.352 (5)	0.370 (7)	0.323 (11)	3.268 (65)	3.264 (67)	0.906 (8)	0.592 (12)	4.367 (88)	4.367 (88)	0.880 (10)	0.482 (11)
	BPR	0.372 (2)	0.374 (8)	0.378 (14)	0.362 (29)	3.268 (65)	3.271 (96)	1.545 (17)	0.886 (20)	4.367 (88)	4.351 (117)	2.170 (19)	1.419 (21)
$U(-2.5,7.5)$	LPR	1.698 (2)	1.698 (5)	1.769 (8)	1.656 (15)	1.694 (55)	1.694 (55)	1.270 (9)	1.584 (11)	6.304 (63)	6.304 (64)	2.007 (9)	1.829 (11)
	BPR	1.698 (2)	1.721 (8)	1.701 (13)	1.673 (29)	1.694 (55)	1.692 (72)	0.203 (17)	0.718 (23)	6.304 (63)	6.299 (87)	3.346 (18)	2.729 (21)
$U(0,10)$	LPR	4.813 (2)	4.804 (5)	4.825 (9)	4.940 (15)	0.027 (71)	0.026 (71)	4.293 (9)	4.383 (9)	9.082 (98)	9.082 (98)	5.738 (9)	5.290 (13)
	BPR	4.813 (2)	4.803 (8)	4.839 (14)	4.914 (32)	0.027 (71)	0.075 (103)	3.110 (19)	3.765 (21)	9.082 (98)	9.067 (132)	6.477 (18)	5.890 (21)
$N(-4,1)$	LPR	4.073 (2)	4.074 (5)	4.107 (8)	4.053 (14)	5.670 (52)	5.670 (52)	4.356 (5)	4.183 (10)	2.360 (52)	2.360 (52)	3.724 (8)	3.954 (10)
	BPR	4.073 (2)	4.074 (7)	4.069 (12)	4.077 (26)	5.670 (52)	5.668 (70)	4.708 (15)	4.471 (19)	2.360 (52)	2.362 (69)	3.340 (16)	3.652 (19)
$N(-1,1)$	LPR	1.235 (2)	1.186 (5)	1.058 (6)	1.040 (12)	2.459 (44)	2.460 (45)	1.295 (9)	1.207 (9)	0.877 (88)	0.874 (88)	0.863 (7)	0.903 (10)
	BPR	1.235 (2)	1.279 (7)	1.172 (13)	1.176 (27)	2.459 (44)	2.456 (58)	1.741 (17)	1.469 (19)	0.877 (88)	0.937 (123)	0.621 (16)	0.839 (20)
$N(1,1)$	LPR	1.082 (2)	1.078 (5)	1.108 (8)	1.079 (13)	1.200 (57)	1.201 (58)	0.702 (9)	0.915 (10)	3.013 (53)	3.014 (54)	1.396 (7)	1.317 (9)
	BPR	1.082 (2)	1.061 (7)	1.065 (12)	1.067 (26)	1.200 (57)	1.198 (76)	0.362 (18)	0.598 (21)	3.013 (53)	3.011 (69)	1.767 (17)	1.492 (19)
$N(4,1)$	LPR	3.899 (2)	3.906 (6)	3.927 (8)	4.095 (14)	2.007 (65)	2.009 (66)	3.640 (9)	3.793 (10)	6.256 (57)	6.255 (57)	4.422 (7)	4.274 (12)
	BPR	3.899 (2)	3.946 (7)	3.972 (13)	3.990 (27)	2.007 (65)	2.014 (87)	3.185 (17)	3.520 (22)	6.256 (57)	6.253 (79)	4.704 (19)	4.476 (21)
$G(1,1)$	LPR	0.788 (2)	0.788 (4)	0.788 (6)	0.739 (10)	0.023 (33)	0.022 (34)	0.637 (7)	0.693 (9)	1.662 (41)	1.663 (42)	0.873 (6)	0.828 (10)
	BPR	0.788 (2)	0.782 (7)	0.776 (13)	0.777 (26)	0.023 (33)	0.025 (44)	0.389 (13)	0.528 (19)	1.662 (41)	1.659 (54)	1.175 (13)	1.022 (19)
$G(2,2)$	LPR	2.827 (2)	2.824 (5)	2.877 (10)	2.965 (13)	0.299 (72)	0.306 (72)	2.346 (7)	2.568 (9)	5.697 (59)	5.697 (59)	3.465 (8)	3.186 (12)
	BPR	2.827 (2)	2.863 (8)	2.875 (14)	2.865 (32)	0.299 (72)	0.332 (97)	1.809 (17)	2.172 (20)	5.697 (59)	5.693 (80)	4.176 (20)	3.683 (23)

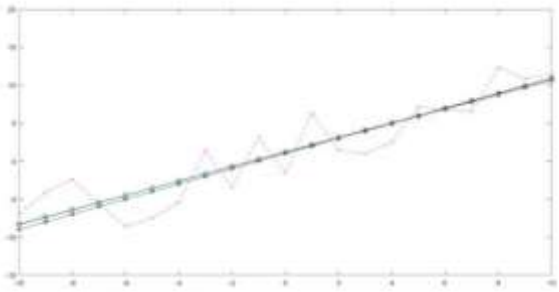
* We random noises by using MATLAB program version R2017a, the codes for $U(a,b)$, $N(a,b)$, and $G(a,b)$ are random('unif',a,b,m,n), random('norm',a,b,m,n), and random('gam',a,b,m,n), respectively, where $m \times n$ is a size of random noises.

- $\eta_i \sim N(-1,1)$, the BPR algorithm with $p_1 = 0.75$ and $p_2 = 0.25$ has the minimum value of RMSE with 16^{th} iteration.
- $\eta_i \sim N(1,1)$, the BPR algorithm with $p_1 = 0.25$ and $p_2 = 0.75$ has the minimum value of RMSE with 18^{th} iteration.
- $\eta_i \sim N(4,1)$, the LPR and BPR algorithms with $p_1 = 0$ and $p_2 = 1$ have the minimum value of RMSE with 65^{th} iteration.
- $\eta_i \sim G(1,1)$, the LPR algorithm with $p_1 = 0.25$ and $p_2 = 1$ has the minimum value of RMSE with 34^{th} iteration.
- $\eta_i \sim G(2,2)$, the LPR and BPR algorithms with $p_1 = 0$ and $p_2 = 1$ have the minimum value of RMSE with 72^{th} iteration.

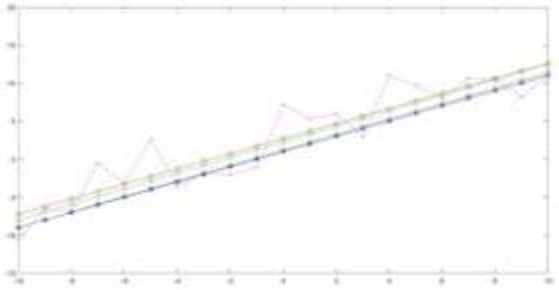
We observe that the number of iterations of the BPR algorithm is always greater than or equal to that of the LPR algorithm. Next, we discuss the values of p_1 and p_2. In the case $p_1 = p_2$, the regression line located in the centre of the point. This designation is suitable for noise having a mean of zero. When $p_1 < p_2$, the bandwidth above the regression line is narrower than that below. The points that are pulled down are more than those that are pulled up. As a result, the regression line will gradually drop in the next iteration. This designation is suitable for noise with a mean of more than zero. When $p_1 > p_2$, the result is the opposite of the previous case. As a result, the regression line will gradually rise in the next iteration. This designation is appropriate for noise having a mean less than zero.

Furthermore, when every noise is positive, such as when η_i is derived from $U(0,10)$, $G(1,1)$, or $G(2,2)$, the appropriate p is $p_2 = 1$. On the other hand, if every noise is negative, the appropriate p is $p_1 = 1$.

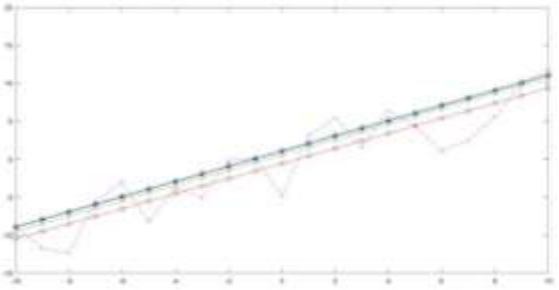
The following figures are some examples from Table 3. Fig. 6 shows the regression of the original function with $\eta_i \sim U(-5,5)$, $\eta_i \sim U(-2.5,7.5)$ and $\eta_i \sim U(-7.5,2.5)$, respectively. The black asterisks on the black line are the points (x,y) of the original function, the pink points are in $D^{(0)}$, and the red, green, and blue lines are the regression lines of linear regression, LPR, and BPR, respectively.



(a) $\eta_i \sim U(-5,5)$ and $p_1 = 0.75, p_2 = 0.75$



(b) $\eta_i \sim U(-2.5,7.5)$ and $p_1 = 0.25, p_2 = 0.75$



(c) $\eta_i \sim U(-7.5,2.5)$ and $p_1 = 0.75, p_2 = 0.25$

Fig. 6: The regression line with noise.

5 Applications

In general, the data points used to create the regression line have noise from the beginning. As a result, we cannot know whether those noises are positive or negative, making it impossible to choose a suitable p-value. Therefore, the selection of p must be determined depending on the desired outcome. For example, if the mean is to be used to represent the data, p_1 and p_2 should be equal. If the regression line is lower than the total data, $p_1 < 1$ and $p_2 = 1$ should be set. Similarly, set $p_1 = 1$ and $p_2 < 1$ when we need a regression line over all data points.

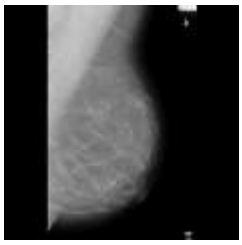
This section will demonstrate how the proposed algorithm can be used to analyse mammograms. In the mediolateral-oblique (MLO) view, the existence

of the pectoral muscle may mislead the diagnosis of cancer due to its high-level similarity to the breast body. Therefore, we cut the pectoral muscle part and then employ LPR or BPR to smooth the boundary.

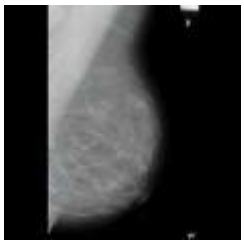
We separate the area of the pectoral muscle and breast using the difference in intensity of a mammogram. We have its border and transform to the Cartesian coordinate, which are referred to as the connection points and defined as $D^{(0)}$. Fig. 7 depicts an example of a mammogram and shows the connection points $D^{(0)}$ of the mammogram, with the area above the points representing the pectoral part and the area below representing the muscle part. Since we want to remove only the muscle part, the regression shall be below all points. We then define $p_2 = 1$.

The value of p_1 should be in the range $[0,1)$; we specify $p_1 = 0.75$ for this example. Fig. 8 shows the regression lines derived from linear regression, LPR, and BPR using red, green, and blue, respectively. We get the regression $y = -0.3383x + 464.8160$ from linear regression, $y = -0.4670 + 480.6891$ from LPR in 2107^{th} iteration, and $y = -0.4752 + 484.2343$ from BPR in 2705^{th} iteration. Moreover, the mammograms after removing the pectoral muscle using linear regression, LPR, and BPR, respectively, are shown in Fig. 9.

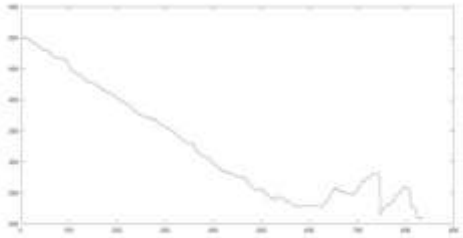
The images obtained by finding the regression line by the LPR and BPR algorithms are comparable to and better than those obtained by the linear regression.



(a) the original mammogram



(b) the connection points plot on the mammogram



(c) the connection points plot on a graph

Fig. 7: An example of a mammogram.

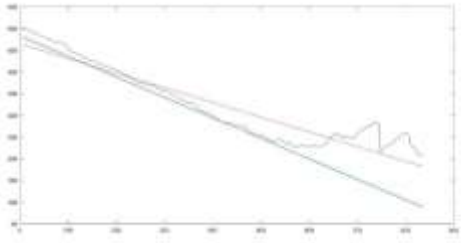
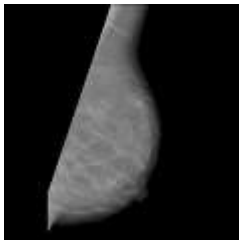
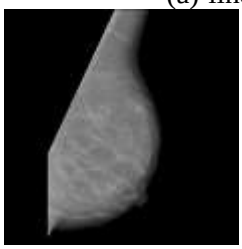


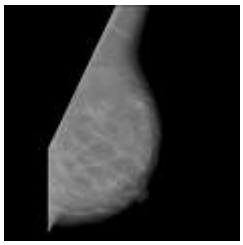
Fig. 8: The regression lines.



(a) linear regression



(b) LPR



(c) BPR

Fig. 9: The mammograms after removing the muscle part by different algorithms.

6 Conclusions and Discussions

In this paper, we proposed the algorithm to create a regression line from an original function with noise η, where η is not necessarily a normal distribution with a mean of zero, called the line-pulling linear regression (LPR) and the band-pulling linear regression (BPR). These algorithms can set the regression line to the centre, top, or bottom of data points by assigning values p_1 and p_2. If $p_1 = p_2 = 0$, the LPR and BPR algorithms provide the same regression lines as linear regression. When $p_1 < p_2$, the resulting line is below the linear regression. And when $p_1 > p_2$, the resulting line is above the linear regression. However, since we do not know the noise distribution in the data, we determine the value of p_1 and p_2 based on user requirements.

The numerical examples show that the results of the LPR and BPR algorithms are similar. The noticeable difference is the number of iterations for which the LPR algorithm converges faster than the BPR algorithm.

The application of these algorithms is the smoothing of the pectoral muscle's boundary. We use $p_1 = 0.75$ and $p_2 = 1$ to create the regression line at the bottom of all data points, ensuring we remove only the muscle part.

In addition, the LPR and BPR algorithms can be extended to more complicated models, such as using quadratic or cubic polynomial equations rather than a linear equation, which is expected to bring greater application benefits.

Acknowledgement:
The first author would like to express gratitude to the Science Achievement Scholarship of Thailand (SAST) for financial assistance for this paper. This research is supported by Department of Mathematics, Faculty of Science, Khon Kaen University, Fiscal Year 2022.

References:

[1] S. M. Stigler, *The history of statistics: The measurement of uncertainty before 1900*, Harvard University Press, 1986.

[2] D. C. Montgomery, E. A. Peck, G. G. Vining, *Introduction to linear regression analysis*, John Wiley & Sons, 2021.

[3] X. Su, X. Yan, C. Tsai, Linear regression, *WIREs Computational Statistics*, Vol.4, No.3, 2012, pp. 275-294.

[4] W. Yao, L. Li, A New Regression Model: Modal Linear Regression, *Scandinavian Journal of Statistics*, Vol.41, 2014, pp. 656-671.

[5] K. H. Zou, K. Tuncali, S. G. Silverman, Correlation and simple linear regression, *Radiology*, Vol.227, No.3, 2003, pp. 617-628.

[6] D. Maulud, A. M. Abdulazeez, A review on linear regression comprehensive in machine learning, *Journal of Applied Science and Technology Trends*, Vol.1, No.4, 2020, pp.140-147.

[7] L. Pérez-Domínguez, H. Garg, D. Luviano-Cruz, J.L. García Alcaraz, Estimation of Linear Regression with the Dimensional Analysis Method, *Mathematics*, Vol.10, No.10, 2022, pp. 1645.

[8] S. Jokubaitis, R. Leipus, Asymptotic normality in linear regression with approximately sparse structure, *Mathematics*, Vol.10, No.10, 2022, pp. 1657.

[9] M. Al-Kandari, K. Adjenughwure, K. Papadopoulos, A Fuzzy-Statistical Tolerance Interval from Residuals of Crisp Linear Regression Models, *Mathematics*, Vol.8, No.9, 2020, pp. 1422.

[10] X. Liu, Y. Chen, A systematic approach to optimizing h value for fuzzy linear regression with symmetric triangular fuzzy numbers, *Mathematical Problems in Engineering*, Vol.2013, 2013.

[11] A. Kabán, New bounds on compressive linear least squares regression, *Artificial intelligence and statistics*, 2014, pp. 448-456.

[12] J. Yi, N. Tang, Variational Bayesian inference in high-dimensional linear mixed models, *Mathematics*, Vol.10, No.3, 2022, pp. 463.

[13] M. Ahn, H. H. Zhang, W. Lu, Moment-based method for random effects selection in linear mixed models, *Statistica Sinica*, Vol.22, No.4, 2012, pp. 1539.

[14] G. K. Uyanık, N. Güler, A Study on Multiple Linear Regression Analysis, *Procedia - Social and Behavioral Sciences*, Vol.106, 2013, pp. 234-240.

[15] M. Liu, S. Hu, Y. Ge, G. B. Heuvelink, Z. Ren, X. Huang, Using multiple linear regression and random forests to identify spatial poverty determinants in rural China, *Spatial Statistics*, Vol.42, 2021, pp. 100461.

[16] Y. Li, X. He, X. Liu, Fuzzy multiple linear least squares regression analysis, *Fuzzy Sets and Systems*, 2022.

[17] S. Weisberg, *Applied Linear Regression*, 4th editio, 2014.

[18] M. S. Paolella, *Linear models and time-series analysis: regression, ANOVA, ARMA and GARCH*, John Wiley & Sons, 2018.
[19] A. C. Rencher, G.B. Schaalje, *Linear models in statistics*, John Wiley & Sons, 2008.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

-Nahatai Tepkasetkul carried out the conceptualization, investigation, methodology, software, validation, visualization, writing - original draft, and writing - review & editing.
-Weenakorn Ieosanurak carried out the supervision, validation, and writing - review & editing.
-Thanapong Intharah carried out the conceptualization, supervision, validation, and writing - review & editing.
-Watcharin Klongdee carried out the conceptualization, investigation, methodology, supervision, validation, visualization, writing - original draft, and writing - review & editing.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

The first author would like to express gratitude to the Science Achievement Scholarship of Thailand (SAST) for financial assistance for this paper. This research is supported by Department of Mathematics, Faculty of Science, Khon Kaen University, Fiscal Year 2022.

Conflict of Interest

The authors have no conflicts of interest to declare that are relevant to the content of this article.