EAI Endorsed Transactions
on Collaborative Computing

Research Article **EAI.EU**

A computing method of predictive value based on fitting function in linear model

Hao Zhong[1,*], Huibing Zhang[2], Fei Jia[2]

[1]School of Computer Science, South China Normal University, Guangzhou 510631, China
[2]Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, Guilin 541004, China

Abstract

Linear models are common prediction models in collaborative computing, which mainly generates fitting function to express the relationship between feature vectors and predictive value. In the process of computing the predictive value according to the fitting function and feature vector, this paper mainly conducted the following researches. Firstly, this paper defines a change interval of predictive value according to training set. Secondly, in this paper, the change interval of predictive value corresponding to feature vector in test set is computed. Finally, according to distribution of training set in the changing interval, the predictive values corresponding to feature vectors in test set are computed. Standard data sets are used in experiment, and MAE(Mean Absolute Error) and RMSE(Root Mean Square Error) are used to evaluate the prediction results. The experimental results show that the method proposed in this paper can improve the prediction error to a certain extent.

Received on 07 June 2020; accepted on 23 September 2020; published on 02 October 2020

Keywords: linear model, linear fitting, fitting function, predictive value

doi:10.4108/eai.2-10-2020.166542

1. Introduction

Prediction models in collaborative computing include linear models and non-linear models. Non-linear models include RF(Random Forest), NN(Neural Network), GBDT(Gradient Boosting Decision Tree), XGBoost(eXtreme Gradient Boosting) and so on. Linear models include linear regression, Ridge Regression, Lasso Regression and so on. The main advantage of linear models is that they are easy to model. The most important technology of linear model is linear fitting. In the process of prediction based on linear model, the fitting function generated according to the training set is called linear fitting. The process of linear fitting is as follows: a loss function is defined by training data, and some parameters of the fitting function are obtained when the loss function takes the minimum value. The parameters characterize the quantitative relationship between the feature vector and the predictive value, so the predictive value corresponding to feature vector of test data can be computed according to these parameters. According to the error between predictive value and true value of test data, the quality of predictive result is evaluated.

The fitting function of linear model reflects the relationship between feature vector and predictive value. Because the modeling of linear models is simple and intuitive, Linear Regression, Ridge Regression, Lasso Regression and other linear models have many application in collaborative computing of social networks. For example, Wang et al. [1] applied linear regression to detection of Alzheimer's disease. Hang et al.[2] applied Ridge Regression to analysis of remote sensing data. Al-Obeidat et al. [3] applied Ridge Regression and Lasso Regression to temperature predictions in buildings. Topuz et al. [4] applied Elastic-Net to predict the number of pediatric outpatients. Li et al. [5] applied the Orthogonal Matching Pursuit to dynamic gesture recognition of radar sensors. Groen et al. [6] applied Bayesian Ridge to macroeconomic data analysis.

The above research work introduces various applications of linear models. Generally, linear models can be used in fields or directions that need data analysis, such

[*]Corresponding author. Email: scnuzhonghao@foxmail.com

as Underwater Wireless Sensors Network[7], Underwater Routing[8] and Underwater Resurrection Routing Synergy[9]. The purpose of this paper is to optimize linear models and reduce predictive errors. Based on the parameters of fitting function, this paper proposes a method for computing a new predictive value, which can improve the accuracy of predictive result. The main contributions are as follows:

- Based on fitting function in linear model, a change interval of predictive value is defined. The change interval represents the potential change range of predictive value, which can better express the relationship between feature vector and predictive value.
- Based on the change interval of predictive value, a new computing method of predictive value is proposed. Experiments show that, new predictive value is closer to the real value than original prediction value.

The remainder of the paper is organized as follows. Section 2 introduces related work. Section 3 describes computing method briefly. Section 4 describes the process of computing method in detail. Section 5 describes the experimental results on the standard dataset. Section 6 is the summary of the paper.

2. Related Works

Linear Regression is the basis of all linear models. For the optimization of Linear Regression, the main direction is to optimize the solution process of loss function and improve convergence speed. The methods used mainly include Least Square Method, Batch Gradient Descent[10], Random Gradient Descent[11] or Mini-Batch Gradient Descent[12]. In addition, Zhang et al. [13] proposed a method of feature weighting, by giving important features a greater weight and minor features a relatively smaller weight, the fitted curve is closer to data distribution. Wang et al. [14] considered that high-dimensional features are more suitable for regression models, and proposed combining GBDT with regression models. He used GBDT to improve the dimensionality of input features and prediction accuracy.

By transforming loss function of Linear Regression, various linear models are generated and applied in different scenarios. For example, for the over-fitting problem in Linear Regression, Ridge Regression [15] added regularization term of L2 norm to loss function, which reduces regression coefficient of sparse features and improves stability of Linear Regression. Aiming at the problem these are large number of sparse features in high-dimensional data, Lasso Regression [16] added regularization term of L1 norm to loss function. It can delete some features which has small regression coefficient, reduce the dimension of features, and improve generalization ability of Linear Regression. Elastic-Net [17] combines the advantages of Ridge Regression and Lasso Regression, and adds a regularization term to loss function. By setting parameters to balance the proportion of the L1 regularization term and the L2 regularization term, the Elastic-Net results tend to Ridge Regression or Lasso Regression as the parameters change. Elastic-Net improves the stability of Linear Regression while improving the generalization ability of Linear Regression to some extent. Orthogonal Matching Pursuit [18] adds a restriction term to loss function, which can limit the maximum number of non-zero elements in the regression coefficient. Under a specified number of non-zero elements, it tends to get optimal regression coefficient to improve prediction accuracy of Linear Regression. Bayesian Ridge [19] assumes that the distribution of regression coefficients is spherical Gauss distribution, and estimates the regression coefficients by maximizing the marginal likelihood function. ARD Regression [20] assumes that the distribution of regression coefficients is elliptical Gauss distribution parallel to coordinate axis, and estimates the regression coefficients by maximizing the marginal likelihood function. When using loss function to calculate errors in samples, Huber Regressor [21] sets a threshold. When the error value of sample exceeds the threshold, sample is divided into abnormal data, and the abnormal data is given a small weight. To some extent, the stability of Linear Regression has been improved. In the process of finding the minimum value of loss function, Least Angle Regression [22] solves angle of current loss residual, and takes feature vector with smallest angle as a direction to find the minimum value. It can obtain ideal results after a limited number of iterations, increase speed at which loss function converges. Lasso Lars [23] combines advantages of Least Angle Regression and Lasso Regression. The regularization term of the L1 norm is added to loss function, and angle for computing current loss residual is found in the training set, and the feature vector with the smallest angle is taken as the direction for finding the minimum value. Improve the generalization ability of Linear Regression, and also increase the speed of the loss function convergence. Passive Aggressive Regressor [24] is a linear model proposed for big data, which incorporates the idea of perceptron into minimization process of loss function. However, the regularization parameter is added to perceptron, which limits parameter size of the perceptron, and avoids the over-fitting problem caused by rough classification of perceptron.

In view of loss function of linear regression, a series of modifications and improvements have been made in the current research work. The stability of the model, the generalization ability of the model, and the convergence speed of loss function have been improved. Based on the fitting functions generated by various regressions, this paper redefines a computing method of prediction value without changing the algorithm complexity of linear model, in order to reduce predictive error.

3. Introduction of computing method

The definitions of symbols in this paper are shown in Table 1.

As shown in Algorithm 1, the process of the computing method is divided into the following four parts. The first step is training process of linear model. According to feature vectors $TRx = \{TRx_1, TRx_2, ..., TRx_n\}$ and corresponding true values $TRy = \{TRy_1, TRy_2, ..., TRy_n\}$, fitting function $Y = W^T X + \omega$ is generated. According to fitting function, the predictive value of feature vector is calculated and a set $\{TRy\_P_1, TRy\_P_2, ..., TRy\_P_n\}$ is obtained. According to parameter α, a potential change interval $[TRy\_P_n*\alpha,\ TRy\_P_n*(2\text{-}\alpha)]$ is defined, where original predictive value TRy_P_n is the center of the interval.

In the second step, for training data, true values of the feature vectors are known. When TRy_n is greater than $TRy\_P_n*(2\text{-}\alpha)$, $TRy\_P_n*(2\text{-}\alpha)$ is new predictive value corresponding to feature vector TRx_n. When TRy_n is in interval $[TRy\_P_n*\alpha,\ TRy\_P_n*(2\text{-}\alpha)]$, TRy_P_n is new predictive value corresponding to feature vector TRx_n. when TRy_n is less than $TRy\_P_n*\alpha$, $TRy\_P_n*\alpha$ is new predictive value corresponding to feature vector TRx_n. When parameter α changes in interval (0,1), the MAE and RMSE values of training data change accordingly. When MAE and RMSE take the minimum value, parameter α is determined.

In the third step, predictive values $\{TEy\_P_1, TEy\_P_2, ..., TEy\_P_m\}$ of test set $TEx = \{TEx_1, TEx_2, ..., TEx_m\}$ is computed according to fitting function, and change interval $[TEy\_D_m,\ TEy\_U_m]$ of each predictive value TEy_P_m is obtained from parameter α. For each change interval $[TEy\_D_m,\ TEy\_U_m]$, according to original predictive value TRy_P_n and true value TRy_n of each training data, the distribution $TRx' = \{TRx'_1, TRx'_2, ..., TRx'_l\}$ of training data in the interval is generated.

In the fourth step, for test data TEx_m, the following known conditions include original predictive value TEy_P_m, change interval $[TEy\_D_m,\ TEy\_U_m]$ of predictive value TEy_P_m, and training data distribution $TRx' = \{TRx'_1, TRx'_2, ..., TRx'_l\}$ in interval $[TEy\_D_m,\ TEy\_U_m]$. According to these three known conditions, the final predictive value TEy_F_m of TEx_m is computed.

4. Computing method and Complexity Analysis

Algorithm 1 The computing process of new prediction value in linear model

```
Input: The feature vector of training set $TRx = \{TRx_1, TRx_2, ..., TRx_n\}$, the feature vector of test set $TEx = \{TEx_1, TEx_2, ..., TEx_m\}$, and the true value of training set $TRy = \{TRy_1, TRy_2, ..., TRy_n\}$
Output: The new predictive value $TEy\_F_m$ of $TEx_m$
Getlossfunction($TRx$, $TRy$); //By the training set
Get $W^T$, $\omega$; //When the loss function takes the minimum value
Define the potential change interval [$TRy\_P_n*\alpha$, $TRy\_P_n*(2\text{-}\alpha)$]; //By $W^T$ and $\omega$
$\alpha \leftarrow$ Getparameter($TRy\_P_n*\alpha$, $TRy\_P_n*(2\text{-}\alpha)$, $TRy_n$)
$TRx' \leftarrow$ Getlabels($TEy\_P_m*\alpha$, $TEy\_P_m*(2\text{-}\alpha)$, $TRy\_P$, $TRy$)
$TEy\_F_m \leftarrow$ Getprevalue($TEy\_P_m*\alpha$, $TEy\_P_m*(2\text{-}\alpha)$, $TRx'$)
```

4.1. Change interval of predictive value

In training process of linear model, according to feature vectors of training sets $TRx = \{TRx_1, TRx_2, ..., TRx_n\}$, parameters W^T and ω are set, and loss function is defined. By solving the minimum of loss function, parameters W^T and ω are obtained, the fitting function $Y = W^T X + \omega$ is generated. According to fitting function, the predictive values $\{TRy\_P_1, TRy\_P_2, ..., TRy\_P_n\}$ of all training data are computed. Taking each predictive value TRy_P_n as center of interval $[TRy\_P_n*\alpha,\ TRy\_P_n*(2\text{-}\alpha)]$. At this time, the paper has the following definitions:

- If true value TRy_n of TRx_n is greater than $TRy\_P_n*(2\text{-}\alpha)$, set $TRy\_P_n*(2\text{-}\alpha)$ as predictive value of TRx_n.
- If true value of TRy_n lies in interval $[TRy\_P_n*\alpha,\ TRy\_P_n*(2\text{-}\alpha)]$, set TRy_P_n as predictive value of TRx_n.
- If true value TRy_n of TRx_n is less than $TRy\_P_n*\alpha$, set $TRy\_P_n*(2\text{-}\alpha)$ as predictive value of TRx_n.

Suppose the value of $TRy\_P_n*\alpha$ is represented by a, and value of $TRy\_P_n*(2\text{-}\alpha)$ is represented by b. The specific method is shown in formula (1).

$$TRy\_F_n = \begin{cases} b, & TRy_n \geq b. \\ \frac{a+b}{2}, & TRy_n \in (a, b). \\ a, & TRy_n \leq a. \end{cases} \tag{1}$$

As shown in formula (1), since true value $TRy = \{TRy_1, TRy_2, ..., TRy_n\}$ of training set is known, according to computing of MAE and RMSE, the objective functions are defined as shown in formula (2) and formula (3).

Table 1. Symbols used in the paper

Feature vectors of training set	$TRx = \{TRx_1, TRx_2, ..., TRx_n\}$
Feature vectors of test set	$TEx = \{TEx_1, TEx_2, ..., TEx_m\}$
True values of training set	$TRy = \{TRy_1, TRy_2, ..., TRy_n\}$
True values of test set	$TEy = \{TEy_1, TEy_2, ..., TEy_m\}$
Fitting function	$Y = W^T X + \omega$
Original predictive value of TRx_n	TRy_P_n
Change interval value of TRy_P_n	$[TRy\_D_n, TRy\_U_n]$
New predictive value of TRx_n	TRy_F_n
Original predictive value of TEx_m	TEy_P_m
Change interval value of TEy_P_m	$[TEy\_D_m, TEy\_U_m]$
Distribution in $[TEy\_D_m, TEy\_U_m]$	$TRx' = \{TRx'_1, TRx'_2, ..., TRx'_d\}$
Location label of TRx'	$L = \{L_1, L_2, ..., L_d\}$
New predictive value of TEx_m	TEy_F_m

$$MAE\_TR = \frac{1}{n} * \sum_{i=1}^{n} \left| TRy_i - TRy\_F_n \right| \tag{2}$$

$$RMSE\_TR = \sqrt{\frac{1}{n} * \sum_{i=1}^{n} (TRy_i - TRy\_F_n)^2} \tag{3}$$

The range of parameter α is $(0, 1)$. When both MAE_TR and $RMSE\_TR$ are minimum, parameter α is obtained, that is, each test data TEy_P_m corresponds to a change interval $[TEy\_D_m, TEy\_U_m]$. The significance of defining the change interval is that, if it can correctly estimate the relationship between true value TEy_m and change interval $[TEy\_D_m, TEy\_U_m]$ (three relationships: $TEy_m \geq TEy\_U_m$, $TEy_m \leq TEy\_D_m$, $TEy_m \in (TEy\_D_m, TEy\_U_m)$), it can get the minimum prediction error.

4.2. Computing method for new predictive values

After input test data TEx_m, original predictive value TEy_m is computed according to fitting function $Y = W^T X + \omega$, and change interval $[TEy\_D_m, TEy\_U_m]$ is obtained according to parameter α. In training set, data TRx_n satisfying $TEy\_D_m \leq TRy\_P_n \leq TEy\_U_m$ is added to set $TRx' = \{TRx'_1, TRx'_2, ..., TRx'_d\}$, and corresponding location label L_d of TRx'_d is shown in formula (4):

$$L_d = \begin{cases} 2, & TRy_n \geq TEy\_U_m. \\ 1, & TRy_n \in (TEy\_D_m, TEy\_U_m). \\ 0, & TRy_n \leq TEy\_D_m. \end{cases} \tag{4}$$

According to formula (4), the number of 0, 1 and 2 in location label set $L = \{L_1, L_2, ..., L_d\}$ is expressed by $S0$, $S1$ and $S2$ respectively. The larger value of $S2$, the greater probability that true value TEy_m of TEx_m satisfies $TEy_m \geq TEy\_U_m$. The larger value of $S1$, the greater probability that true value TEy_m of TEx_m satisfies $TEy_m \in (TEy\_D_m, TEy\_U_m)$. The larger value of $S0$, the greater probability that true value TEy_m of TEx_m satisfies $TEy_m \leq TEy\_D_m$. In practice, the relationship between true value TEy_m and change interval $[TEy\_D_m, TEy\_U_m]$ can not be accurately judged. With three values of $S0$, $S1$ and $S2$ as probabilistic values, the final predictive values TEy_F_m corresponding to TEx_m is defined as shown in formula (5):

$$TEy\_F_m = \frac{S2 * TEy\_U_m + S1 * TEy\_P_m + S0 * TEy\_D_m}{S2 + S1 + S0} \tag{5}$$

4.3. Complexity analysis

The most important part of the linear model is to find the minimum of the loss function. Several main methods are Least Squares Method, Random Gradient Descent, Batch Gradient Descent and Mini-Batch Gradient Descent. The complexity of the linear model is the same as method for finding the minimum of the loss function. Assuming that data quantity is N_s and dimension of feature vector is N_f, where $N_s \gg N_f$. The complexity of least square method is $O(N_s * N_f^2) = O(N_s)$. The complexity of Random Gradient Descent is $O(N_s * N_f) = O(N_s)$. The complexity of Batch Gradient Descent and Mini-Batch Gradient Descent are $O(N_s^2)$.

According to the above analysis, the minimum complexity of linear model is $O(N_s)$. Based on fitting function of linear model, a method for calculating the new predictive value is proposed in this paper. The complexity of computing the change interval is $O(N_s)$. The complexity of obtaining data distribution is $O(N_s)$. The complexity of computing predictive value is $O(1)$. Generally speaking, the total complexity of the proposed method is $O(N_s) + O(N_s) + O(N_s) + O(1) = O(N_s)$, which improves the prediction effect of the model without changing complexity of the linear model.

5. Data sets and Experiments

5.1. Data Sets

In the sklearn database[25], three standard data sets are selected to test the effectiveness of the proposed method. The data set is suitable for regression tasks. The selected data set and invocation method are shown in Table 2.

Table 2. Datasets

Invoke Method	Representation
datasets.load-boston()	boston
datasets.fetch-california-housing()	california
datasets.load-diabetes()	diabetes

5.2. Experimental result of data sets

In the first experiment of the paper, Linear Regression (represented by LR) was used to predict and analyze the three data sets. At the same time, according to the method proposed in this paper, a new predictive value of Linear Regression (expressed by LR+) is computed. The predictive results were evaluated by using MAE and RMSE. The experimental results are shown in Table 3:

Table 3. Prediction results of three data sets

data sets	α	LR (MAE)	LR+ (MAE)	LR (RMSE)	LR+ (RMSE)
california	0.747	0.54	0.52	0.73	0.72
boston	0.865	3.84	3.66	5.78	5.72
diabetes	0.724	46.17	45.18	58.52	58.07

As shown in Table 3, this paper presents a new method to computer the predictive value, which can reduce the predictive error to a certain extent without changing the complexity. This is because when test data are input, corresponding original predictive value is computed on fitting function. In practice, the true value of test data is located above or below original predictive value. If the position of true value can be correctly judged, prediction error can be reduced. In this paper, we take the original predictive value as a center, and define change interval of predictive value. According to data distribution in this interval, the position relationship between true value of test data and original predictive value is judged. New predictive values are computed according to the change interval and position relationship. The experimental results show that the new predictive value is closer to the true value.

5.3. Experimental result of linear models

In the second experiment of this paper, the data set boston was analyzed by linear models such as Ridge Regression and Lasso Regression. At the same time, according to the method proposed in the paper, computer new predictive values for linear models such as Ridge Regression and Lasso Regression(expressed by Ridge Regression+, Lasso Regression+). MAE and RMSE are used to evaluate the predictive effect. The experimental results are shown in Table 4.

Table 4. Prediction results of three data sets

line models	α	MAE	RMSE
Ridge Regression	0.864	3.84	5.79
Ridge Regression+	0.864	3.65	5.72
Lasso Regression	0.836	4.20	6.32
Lasso Regression+	0.836	3.99	6.25
ElasticNet	0.833	4.34	6.53
ElasticNet+	0.833	4.15	6.51
OrthogonalMatchingPursuit	0.816	4.40	6.46
OrthogonalMatchingPursuit+	0.816	4.17	6.31
BayesianRidge	0.874	3.93	5.97
BayesianRidge+	0.874	3.73	5.87
ARDRegression	0.847	4.05	6.01
ARDRegression+	0.847	3.86	5.90
HuberRegressor	0.805	4.35	6.90
HuberRegressor+	0.805	4.14	6.78
Lars	0.861	3.85	5.86
Lars+	0.861	3.66	5.76
LassoLars	0.856	3.84	5.81
LassoLars+	0.856	3.66	5.73
PassiveAggressiveRegressor	0.529	7.20	9.35
PassiveAggressiveRegressor+	0.529	6.38	8.69

As shown in Table 4, based on fitting function generated by linear models such as Ridge Regression and Lasso Regression, this paper compares the errors between original predictive value and new predictive value. The experimental results show that, new predictive values proposed in this paper can reduce predictive error of all linear models to a certain extent. This is because the purpose of all linear models is to generate a fitting function, that represents the relationship between feature vector and predictive value. However, there is always an error between predictive value and true value. Based on fitting function of the linear model, this paper proposes a method for computing predictive value. The new predictive value can be closer to the true value than original predictive value, so predictive errors of all linear models can be improved to a certain extent.

6. Conclusion

The work of this paper includes following two parts. Firstly, based on fitting function of linear model, a change interval of predictive value is defined. If the relationship between true value and interval can be correctly judged, the prediction error can be effectively reduced. Secondly, when predicting the test data, the relationship between true value and interval can not be accurately judged. Therefore, according to distribution of training data in change interval, the relationship between true value and change interval of the predictive value is estimated. According to the relationship, new predictive value is computed to replace the original predictive value. Tested on standard data sets, new predictive values can reduce the predictive error to a certain extent.

The paper estimates the relationship between true value and change interval based on probability, so it can only improve predictive error to a certain extent. If the relationship between true value and change interval can be computed concretely, the improvement of predictive error will be more obvious. In the next step, this paper will continue to study how to analyze relationship the relationship between true value and change interval concretely based on classification algorithm.

References

[1] Shui-Hua Wang, Sidan Du, Yin Zhang, Preetha Phillips, Le-Nan Wu, Xian-Qing Chen, and Yu-Dong Zhang. Alzheimer's disease detection by pseudo zernike moment and linear regression classification. *CNS & Neurological Disorders-Drug Targets (Formerly Current Drug Targets-CNS & Neurological Disorders)*, 16(1):11–15, 2017.

[2] Renlong Hang, Qingshan Liu, Huihui Song, Yubao Sun, Fuping Zhu, and Hucheng Pei. Graph regularized nonlinear ridge regression for remote sensing data analysis. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(1):277–285, 2016.

[3] Feras Al-Obeidat, Bruce Spencer, and Omar Alfandi. Consistently accurate forecasts of temperature within buildings from sensor data using ridge and lasso regression. *Future Generation Computer Systems*, 2018.

[4] Kazim Topuz, Hasmet Uner, Asil Oztekin, and Mehmet Bayram Yildirim. Predicting pediatric clinic no-shows: a decision analytic framework using elastic net and bayesian belief network. *Annals of Operations Research*, 263(1-2):479–499, 2018.

[5] Gang Li, Rui Zhang, Matthew Ritchie, and Hugh Griffiths. Sparsity-based dynamic hand gesture recognition using micro-doppler signatures. In *2017 IEEE Radar Conference (RadarConf)*, pages 0928–0931. IEEE, 2017.

[6] Jan JJ Groen and George Kapetanios. Revisiting useful approaches to data-rich macroeconomic forecasting. *Computational Statistics & Data Analysis*, 100:221–239, 2016.

[7] Shahzad Ashraf, Sehrish Saleem, and Tauqeer Ahmed. Sagacious communication link selection mechanism for underwater wireless sensors network. *International Journal of Wireless and Microwave Technologies*, 10(4):22–33, 2020.

[8] Shahzad Ashraf, Zeeshan Aslam, Adnan Yahya, and Adnan Tahir. Underwater routing protocols: Analysis of intrepid link selection mechanism, challenges and strategies. *International Journal of Scientific Research in Computer Science and Engineering*, 8:0–00, 04 2020.

[9] Shahzad Ashraf, Asif Raza, Zeeshan Aslam, Hamad Naeem, and Tauqeer Ahmed. Underwater resurrection routing synergy using astucious energy pods. *Journal of Robotics and Control (JRC)*, Volume 1:173–184, 04 2020.

[10] Maximilian Schleich, Dan Olteanu, and Radu Ciucanu. Learning linear regression models over factorized joins. In *Proceedings of the 2016 International Conference on Management of Data*, pages 3–18. ACM, 2016.

[11] Prateek Jain, Sham M. Kakade, Rahul Kidambi, Praneeth Netrapalli, and Aaron Sidford. Parallelizing stochastic gradient descent for least squares regression: Mini-batching, averaging, and model misspecification. *J. Mach. Learn. Res.*, 18:223:1–223:42, 2017.

[12] Lala Septem Riza, Imam Fachmi Nasrulloh, Enjun Junaeti, Rena Zain, and Asep Bayu Dani Nandiyanto. graddescentr: An r package implementing gradient descent and its variants for regression tasks. In *2016 1st International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, pages 125–129. IEEE, 2016.

[13] Junbo Zhang, CY Chung, and Yingduo Han. Online damping ratio prediction using locally weighted linear regression. *IEEE Transactions on Power Systems*, 31(3):1954–1962, 2015.

[14] Yaozheng Wang, Dawei Feng, Dongsheng Li, Xinyuan Chen, Yunxiang Zhao, and Xin Niu. A mobile recommendation system based on logistic regression and gradient boosting decision trees. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 1896–1902. IEEE, 2016.

[15] Le Zhang and Ponnuthurai Nagaratnam Suganthan. Benchmarking ensemble classifiers with novel co-trained kernel ridge regression and random vector functional link ensembles [research frontier]. *IEEE Computational Intelligence Magazine*, 12(4):61–72, 2017.

[16] Wei Li, Jianxing Feng, and Tao Jiang. Isolasso: a lasso regression approach to rna-seq based transcriptome assembly. *Journal of Computational Biology*, 18(11):1693–1707, 2011.

[17] Pin-Yu Chen, Yash Sharma, Huan Zhang, Jinfeng Yi, and Cho-Jui Hsieh. EAD: elastic-net attacks to deep neural networks via adversarial examples. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 10–17, 2018.

[18] Changchen Zhao, Wen-Liang Hwang, Chun-Liang Lin, and Weihai Chen. Greedy orthogonal matching pursuit for subspace clustering to improve graph connectivity. *Inf. Sci.*, 459:135–148, 2018.

[19] Qi Shi, Mohamed Abdel-Aty, and Jaeyoung Lee. A bayesian ridge regression analysis of congestion's impact on urban expressway safety. *Accident Analysis & Prevention*, 88:124–137, 2016.

[20] Momoko Muto and Osamu Ezaki. High dietary saturated fat is associated with a low risk of intracerebral hemorrhage and ischemic stroke in japanese but not in non-japanese: a review and meta-analysis of prospective cohort studies. *Journal of atherosclerosis and thrombosis*, 25(5):375–392, 2018.

[21] Song Jiang and Xiaotian Han. Dmgroup at emoint-2017: Emotion intensity using ensemble method. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, WASSA@EMNLP 2017, Copenhagen, Denmark, September 8, 2017*, pages 243–248, 2017.

[22] Huiqin Jiang and Rung-Ching Chen. The selection and reduction of fuzzy rules based on the least angle regression algorithm. In *2018 International Symposium on Computer, Consumer and Control (IS3C)*, pages 432–435. IEEE, 2018.

[23] Theodore Panagiotidis, Thanasis Stengos, and Orestis Vravosinos. On the determinants of bitcoin returns: A lasso approach. *Finance Research Letters*, 27:235–240, 2018.

[24] Arnold Salas, Stephen J Roberts, and Michael A Osborne. A variational bayesian state-space approach to online passive-aggressive regression. *arXiv preprint arXiv:1509.02438*, 2015.

[25] 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.