ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

# PENALIZED REGRESSION METHOD IN HIGH DIMENSIONAL DATA

**ZUHARAH JAAFAR[1], NORAZLINA ISMAIL[2]**

[1]Department of Mathematics, Universiti Teknologi Malaysia.

[2]Department of Mathematics, Universiti Teknologi Malaysia.
[1]zuharahj@gmail.com, [2]i-norazlina@utm.my

**ABSTRACT**

In huge multivariate data set with a number of variables greater than the number of samples, the standard linear model (or ordinary least squares method) performs badly. In such situations, a better option is penalized regression, which allows you to design a linear regression model that is penalized for having too many variables by adding a constraint to the equation Shrinkage or regularization procedures are other names for this. The penalty has the effect of reducing (i.e. shrinking) the coefficient values towards zero. This permits the coefficients of the less important variables to be near to or equal to zero. By decreasing the number of coefficients and maintaining those with coefficients greater than zero, penalized regression models improve prediction in new data when compared to traditional methods. We demonstrate that the proposed regularizer is capable of achieving competitive results as well as exceedingly compact networks. Extensive tests are carried out on a number of benchmark datasets to demonstrate the effectiveness of the method.

**Keywords:** *Lasso, Ridge, Adaptive Lasso, Elastic-Net, Vgg-19, MNIST, CIFAR-10, ImageNet*

## 1. INTRODUCTION

Regularization, or penalized logistic regression, is a sort of logistic model that penalizes or minimizes the impact of specific variables. When a dataset contains a large number of variables and there is no way of knowing which ones would be useful in the regression model, regularization techniques are applied. Regularization approaches impose a penalty to minimize the effect of some variables without completely deleting them from the equation in order to avoid overfitting the model to the data. This should result in a model that shows which variables have a greater impact on the predictive value than others. There are other approaches for creating a penalized regression model.

Penalized regression methods are a valuable theoretical strategy for both creating prediction models and choosing essential indicators from a pool of data that is typically much bigger. By decreasing the number of coefficients and maintaining those with coefficients greater than zero, penalized regression models improve prediction in new data when compared to traditional methods.

The performance and selection of indicators, on the other hand, are dependent on the algorithm used. The procedure of regularization used in penalized regression is another factor to consider when balancing the accuracy and generalizability of predictive models. Regularization is an automated process for reducing the strength of coefficients for predictive variables that are thought to be irrelevant in predicting the outcome to zero.

In terms of complexity, regularization also aids in balancing the accuracy and generalizability of predictive models. The number of indicators in the final model is typically referred to as complexity. The regularization process shrinks the coefficients of irrelevant variables to zero for numerous penalized regression algorithms (i.e., excluded from the model).

By balancing the bias-variance trade-off, regularizations also help to reduce overfitting **[1].** The model becomes less sensitive to the properties of the training data by reducing the magnitude of the estimated coefficients (i.e., adding bias and reducing accuracy in the training data), resulting in fewer variations in predictions when estimating the same model in the testing data (i.e., reducing variation and increasing generalizability). Various penalized regression methods are discussed and summarized in Table 1 as:

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

*Table 1: Summarized penalized regression methods*

| **Penalty regression Method** | **Description** | **Penalty** |
|---|---|---|
| **$L_0$ Regression method** | Sometimes known by the name $L_0$ penalty method. Penalizes the coefficients or parameters to some different smaller values rather than approaching them to zero values. Because of its non-convexity, it is an NP-hard problem, and finding a minimum is very difficult. | $\hat{\beta}_{best} = \arg\min_{\beta} \frac{1}{m}\sum_{j=1}^{m}\left(Z_j - \beta^T Y_j\right)^2 + \lambda\|\beta\|_0$ **[2]** |
| **Lasso ($L_1$ Regression) Method.** | This regularization method actually penalizes the absolute values of the coefficients. Makes the value of inapplicable things to zero. Also helps in removing too many redundant features in the model. | $\sum_{k=1}^{q} \lvert \beta_k \rvert < t$ **[3]** |
| **Ridge Regression ($L_2$ Regression) Method.** | Instead of penalizing the absolute coefficients, it penalizes the square of the magnitude of coefficients. It makes $\beta$ values very small but not zero. It doesn't help in removing too many redundant features in the model but to some extent can minimize their impact on the model. | $\lambda * a_1^2 + \lambda * b_1^2$ ; $\|\beta\|_2$ <br> $a_1$ and $b_1$ are regression coefficients <br> **[4]** |
| **Adaptive-Lasso Regression Method.** | It has a property of selection consistency that is it assures that once the size of the sample tends to approach infinity, the selected predictor variables approach towards true values with the probability of 1. Secondly, it also gives assurance about the asymptotical standardization of the estimators with the similar mean and covariance they already have with maximum likelihood estimation. | $\sum_{k=1}^{q}\left(\left\lvert\frac{\beta_k}{\hat{\beta}_k}\right\rvert\right) < t$ **[5]** |
| **Elastic-Net Regression Method.** | This method actually makes use of a weighted combination of both $L_1$ and $L_2$ regularization methods. It combines the advantages provided by both methods and comes to exist as the most efficient regularization/ regression technique. | $\sum_{k=1}^{q}\left(\lvert\beta_k\rvert\right) < t_1$ <br> $\sum_{k=1}^{q}\left(\lvert\beta_k^2\rvert\right) < t_2$ <br> **[6]** |

Journal of Theoretical and Applied Information Technology
30th April 2022. Vol.100. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

### 1.1 Summary of Main Contributions of the Work

In this paper, various penalized regression methods are discussed with their penalized factor. A detailed literature review of all the regression methods has been jotted down.

After doing a comprehensive literature review and critical analysis of various penalized regression methods, a novel penalized regression method known as e-norm is proposed to minimize the cost function and is far more efficient as compared to the already existing start of art penalized regression methods.

The regularizer for generic neural networks based on e-norm with batch normalization is used in this work.

The proposed regularization method is embedded in the vgg-19 model in addition to batch normalization implemented on three datasets to achieve the results.

## 2. RELATED WORK

(Jian Huang, Yuling Jiao et al in 2018) The authors suggested a novel method for estimating sparse, high-dimensional linear regression models that is constructive. The method is a computer algorithm based on the KKT criteria for 0-penalized least-squares solutions. It iteratively develops a series of solutions based on support detection using primal and dual information, as well as root finding **[2].**

(Hiroaki Fukunishi, Mitsuki Nishiyama et al in 2020) Using routinely gathered claim data from health insurance and long-term care insurance databases in Japan, this study established Alzheimer's-type dementia predictions for those over 75 who do not receive long-term care services. To select influential features from a large number of feature candidates, a sparse logistic regression model with $L_0$ regularization (SLR-$L_0$) was used and compared to a sparse logistic regression model with $L_1$ regularization (SLR-$L_1$). AUC predictions for SLR-$L_0$ were 0.663 and 0.660, respectively, while the average number of selected features was 13 out of 611 for SLR-$L_0$ and 253 for SLR-$L_1$ **[7].**

(Sushrut Karmalkar, Eric Price in 2019) For the problem of sparse robust linear regression, we provide a simple and effective approach. In this task, the authors estimated a sparse vector $w^* \in \mathrm{R}^n$ using linear measurements that have been corrupted by sparse noise that can arbitrarily change a $\eta$ fraction of observed responses *y*, as well as introduce bounded norm noise to the responses. The authors observed that for Gaussian observations, a basic $L_1$ regression method can effectively predict $w^*$ for any $\eta < \eta_0 \approx 0.239$ and that this threshold is too high for the algorithm. For *k*-sparse estimation, the number of measurements required by the approach is $\mathrm{O}(k \log nk)$, which is within constant factors of the number required without any sparse noise **[8].**

(Ramy Hussein, Mohamed Elgendi et al in 2018) This research demonstrates a reliable seizure detection system that performs well in both real-world and ideal situations. To determine the most prominent features important to epileptic seizures, a feature learning method based on $L_1$-penalized robust regression is devised and applied to the EEG spectra. For seizure detection, the collected features are loaded into a random forest classifier. The performance of this seizure detection algorithm is superior to previous studies, according to results from a public benchmark dataset. Under ideal conditions, it achieves 100 percent sensitivity, 100 percent specificity, and 100 percent classification accuracy for seizure detection. The proposed technique has also been shown to be reliable in the face of white noise and EEG artifacts, particularly those caused by muscular activity and eye blinking **[9].**

(Edgar Dobriban, Stefan Wager et al in 2018) In a dense random-effects model, the authors presented a unified analysis of the predictive risk of ridge regression and regularized discriminant analysis. The authors operate in a high-dimensional asymptotic regime where $p,\ n \to \infty,\ \frac{p}{n} > 0$ and allow for arbitrary feature covariance. They gave an explicit and computationally efficient expression for the limiting prediction risk for both techniques, which is dependent only on the spectrum of the feature-covariance matrix, the signal strength, and the aspect ratio **[10].**

(Sifan Liu, Edgar Dobriban et al in 2019) The authors looked at the following three fundamental ridge regression issues:

i. What is the estimator's structure?
ii. How do you choose the regularization parameter using cross-validation correctly?
iii. How can I speed up computation without sacrificing accuracy?

In a unified large-data linear model, we consider the three difficulties. Ridge regression is represented as a covariance matrix-dependent linear combination of the real parameter and the noise in this paper. The bias of K-fold cross-validation for determining the regularization parameter is investigated, and a simple bias-correction is proposed. For ridge

Journal of Theoretical and Applied Information Technology
30th April 2022. Vol.100. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

regression, they investigated the accuracy of primal and dual sketching, and find that they are both surprisingly accurate. Simulations and empirical data analysis are used to demonstrate our findings **[11].**

(Gholamreza Hesamian, Mohammad Ghasem Akbari in 2019) The Lasso approach was expanded for multiple linear regression models with non-fuzzy explanatory variables and fuzzy answers in this study. By removing variables that are unnecessary to the fuzzy response variables, the fuzzy Lasso approach can improve the model's interpretability. A fuzzy penalized technique was used to estimate unknown fuzzy regression coefficients and tuning constants for this purpose. The proposed method's performance was also evaluated using certain common goodness-of-fit metrics. Two real-world instances and a simulation study were used to evaluate the suggested method's effectiveness **[12].**

(Nima S. Hejazi, Jeremy R. Coyle et al in 2020) The authors developed hal9001 R package implements the highly adaptive lasso (HAL), a flexible nonparametric regression and machine learning approach with various theoretically useful qualities, in a computationally efficient manner. In order to improve the algorithm's scalability, hal9001 combines an implementation of this estimator with a set of useful variable selection tools and appropriate defaults. The hal9001 R package provides a family of highly adaptive lasso estimators suitable for use in both modern large-scale data analysis and cutting-edge research efforts at the intersection of statistics and machine learning, including the emerging subfield of computational causal inference, by building on existing R packages for lasso regression and leveraging compiled code in key internal functions. 2020 (Wong) **[13].**

(Rahim Alhamzawi, Haithem Taha Mohammad Ali in 2018) The scale mixture of truncated normal (with exponential mixing densities) representation of the Bayesian adaptive lasso prior is used in this research to propose a novel hierarchical representation of Bayesian adaptive lasso. A full Bayesian treatment was examined, which resulted in a novel Gibbs sampler with tractable full conditional posterior distributions. We evaluate the performance of the new Gibbs sampler with some current Bayesian and non-Bayesian approaches using simulations and real data analysis. The new approach outperforms previous Bayesian and non-Bayesian approaches, according to the findings **[14].**

(Kazim Topuz, Hasmet Uner, in 2018) The elastic net (EN) variable-selection methodology is combined with probabilistic Bayesian Belief Network (BBN) in this study to create a hybrid probabilistic prediction framework. The study uses demographics, socioeconomic status, current appointment information, and the patient's and family's appointment attendance history to estimate the "no-show probability of the patient(s)". The proposed approach is supported by ten years of data from a local pediatric clinic. This EN-based BBN framework is proved to be a similar prediction methodology when compared to the top methodologies in the literature **[15].**

(Achim Ahrens, Christian B. Hansen et al in 2020) Developed LASSOPACK, which is a collection of programmers for penalized regression methods in high-dimensional settings, such as when the number of predictors p is big and possibly exceeds the number of data. Both lasso and logistic lasso regression are supported by LASSOPACK. There are six primary programmers in the package: Lasko2, square-root lasso, elastic net, ridge regression, adaptive lasso, and post-estimation OLS are all implemented in lasso2. For cross-section, panel, and time-series data, cv lasso provides K-fold cross-validation and rolling cross-validation. For cross-section and panel data, r lasso implements theory-driven penalization for the lasso and square-root lasso. The related programs for logistic lasso regression are lasso logit, cv lasso logit, and r lasso logit **[16].**

(Sara van Erp, Daniel L. Oberski et al in 2019) To make comparisons easier, we present a theoretical and conceptual comparison of nine distinct shrinkage priors and, if possible, parametrize the priors in terms of scale mixture of normal distributions. In simulation research, we show the distinct properties and behaviour of shrinkage priors and compare their performance in terms of prediction and variable selection. We also present two empirical examples to demonstrate how Bayesian penalization can be used. Finally, researchers can use the R package bayesreg (https://github.com/sara-vanerp/bayesreg) to do Bayesian penalized regression with innovative shrinkage priors in a simple way **[17].**

## 3. MAIN CONTRIBUTION OF THE WORK

In this work, various penalized regression methods are discussed with their penalized factor. A detailed literature review of all the regression methods has been jotted down in this paper. After doing a comprehensive literature review and critical analysis of various penalized regression methods.

Journal of Theoretical and Applied Information Technology
30th April 2022. Vol.100. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A novel penalized regression method known as e-norm is proposed to minimize the cost function and is far more efficient as compared to the already existing start of art penalized regression methods.

## 4. PROPOSED WORK

The most frequent choice for using the regularization method in deep learning models is the l2 norm which is also known by the name weight decay or ridge regression. The mathematical formulation of $L_2$ norm is basically a vector form and for complex vector, it is depicted as:

$y = [y_1 \quad y_2 \quad \cdots \quad y_n]^T$ and is given by

$$|y| = \sqrt{\sum_{j=1}^{m} |y_j|^2}$$

For real vectors the $L_1$ norm for vector $y = [y_1 \quad y_2 \quad \cdots \quad y_n]^T$ is given as :

$$|y| = |y_1| + |y_2| + \cdots + |y_n|$$

Penalizing property of weight decay varies directly with the magnitude of weights i.e more penalty for greater weights and less penalty for smaller weights. This method is not able to shrink the values of weights to zero. Thus is not able to introduce any kind of sparsity required for dense neural networks. This lack of introducing sparsity makes this regularization unfit for current dense neural networks. Though $L_1$ regularizer is best choice to introduce the sparsity, but the major drawback of the regression method is that it makes choice for only non-zero parameters and excludes others. The $L_0$ is the norm is given as:

$$\|y\|_0 = \lim_{p \to 0} \sum_{j=1}^{n} |y_j|^p$$

Finding the solution of this l0 regularization method is an NP-hard problem so we relax this solution from $L_0$ to novel e-norm which is given as

We define the new norm $\|.\|_e$ as

$$\|y\|_e = \left( \sum_{j=1}^{n} \left( \frac{1}{2^j} |y_j| \right)^p \right)^{\frac{1}{p}}$$

where $y = [y_1 \quad y_2 \quad \cdots \quad y_n]^T \in \mathrm{R}^n$

The function $\|.\|_e$ defines a norm on $\mathrm{R}^n$, which can be verified easily.

The LeCP estimator is given by

$$\hat{\beta}(\alpha_1, \alpha_2) = \arg\min_{\beta} \left\{ \|y - \mathbf{x\beta}\|_\infty + \alpha_1 \|\boldsymbol{\beta}\|_e^p + \alpha_2 P_c(\boldsymbol{\beta}) \right\} \ldots(1)$$

Where

$$P_c(\boldsymbol{\beta}) = \sum_{j=1}^{n-1} \sum_{i>j} \frac{1}{2^{ij}} \left\{ \frac{(\beta_i - \beta_j)^2}{(1-\rho_{ij})} + \frac{(\beta_i + \beta_j)^2}{(1+\rho_{ij})} \right\} \ldots(2)$$

$\|\beta\|_\infty = \max_j \{|\beta_j|\}$, $\alpha_1$ and $\alpha_2$ are positive tuning parameters and $\rho_{ij}$ denotes the correlation coefficient between the $i$th and $j$th predictors.

If $\rho_{ij}^2 \neq 1$ for $i \neq j$, the penalty (2) can be written in quadratic form as $P_c(\beta) = \boldsymbol{\beta}^T \mathbf{V} \boldsymbol{\beta}$

Where, $\mathbf{V} = (v_{ij})$ is a $n \times n$ positive definite matrix where the $(i, j)$th term is given by

$$v_{ij} = \begin{cases} 2\sum_{r \neq i} \frac{1}{1-\rho_{ir}^2} & \text{if } i = j \\ -2\frac{\rho_{ij}}{1-\rho_{ij}^2} & \text{if } i \neq j \end{cases}$$

Putting, $\lambda = \alpha_2 / (\alpha_1 + \alpha_2)$, the optimization problem (1) is equivalent to

$$\hat{\beta}(\alpha_1, \alpha_2) = \arg\min_{\beta} \left\{ \|y - \mathbf{x\beta}\|_\infty \right\} \ldots(3)$$ such that $(1-\lambda)\|\boldsymbol{\beta}\|_e^p + \lambda P_c(\boldsymbol{\beta}) \leq \nu$ for $\nu \geq 0$.

The penalty $J(\beta) = \alpha_1 \|\boldsymbol{\beta}\|_e^p + \alpha_2 P_c(\boldsymbol{\beta})$ can be written as

$$\begin{aligned} J(\beta) &= \alpha_1 \|\boldsymbol{\beta}\|_e^p + \alpha_2 P_c(\boldsymbol{\beta}) \\ &= (\alpha_1 + \alpha_2) \left\{ \frac{\alpha_1}{\alpha_1 + \alpha_2} \|\boldsymbol{\beta}\|_e^p + \frac{\alpha_2}{\alpha_1 + \alpha_2} P_c(\boldsymbol{\beta}) \right\} \\ &= \alpha \left[ (1-\lambda) \|\boldsymbol{\beta}\|_e^p + \lambda P_c(\boldsymbol{\beta}) \right] \end{aligned}$$

Where, $\alpha = \alpha_1 + \alpha_2$. Hence, the problem (1) is equivalent to finding

$$\hat{\beta}(\alpha, \lambda) = \arg\min_{\beta} \left\{ \|y - \mathbf{x\beta}\|_\infty + \alpha \left[ (1-\lambda) \|\boldsymbol{\beta}\|_e^p + \lambda P_c(\boldsymbol{\beta}) \right] \right\} \ldots(4)$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Where, $\lambda$ is the positive hyper parameter responsible for controlling the effect of regularization term and the loss term $\hat{\beta}(\alpha,\lambda) = \arg\min_{\beta}\left\{\|y - \mathbf{x\beta}\|_{\infty}\right\}$ given in equation (4). This loss function is meant to make the comparison between the predicted value and ground truth output. The loss function depends upon the task performed by us. If our task is a classification problem, then the default loss function used is cross-entropy and for regression problems, it is either Mean square error loss function or Mean absolute error loss function. Which are given in the equation below. And $\hat{\beta}(\alpha,\lambda) = \alpha\left[(1-\lambda)\|\boldsymbol{\beta}\|_e^p + \lambda P_c(\boldsymbol{\beta})\right]$ is the regularizer on the set of weight parameter's $\beta$ .

$$\text{Loss}(.) = \begin{cases} \dfrac{1}{2n}\left(X - \sum_{k=1}^{n}\beta^T{}_k x_k\right) & \text{for clasification} \\ \dfrac{-1}{L}\sum_{l\in L} y_l \log \hat{y}_l & \text{MSE for regression} \\ \dfrac{1}{m}\sum_{k=1}^{m}\left|y_k - \hat{y}_k\right| & \text{MAE for regression} \end{cases}$$

To enforce sparsity in the dense neural networks, our main motive was to construct the best fit regularization method so that proper balance is maintained between negative as well as non-negative para meters This novel e-norm enforces sparsity in the neural network by doing the shrinkage of weight values to zero. This e-norm has unique property as compared to l0 norm as it takes into consideration non –negative real numbers and does not exclude any other numbers. Thus becomes more efficient and feasible solution for introducing sparsity in the dense neural networks in the current era of IOT constrained technology, whose requirements are constrained like power constraints, memory constraint's, and resource constraints.

## 5. NUMERICAL EXPERIMENTS AND SIMULATIONS

Baseline Networks and Datasets used for simulation: The basic vgg-19 Network is trained with the l1, l2 and the proposed novel e-norm regular subjected to following datasets.

**MNIST [18]:** This dataset consists of seventy thousand hand written digit images each of dimensionality $28 \times 28$ belonging to 10 classes. Total number of training samples images are 60,000 and total number of test images are 10,000.

**CIFAR-10[19]:** The dataset consists of 60,000 images belonging to 10 different classes, each of dimensionality $32 \times 32$. The dataset consists of 5,000 training images and 1000 testing images.

**ImageNet20]:** A very large scale database consisting of 1,386,167 image samples belonging to 1000 different categories. Every image sample of dimensionality $256 \times 256$. It consists of 1,281,167 training sample images, 100,000 testing sample images, and 50,000 validation image samples.

The network used for numerical experiments is given as in [21] and the graphical visualization of the network is depicted in figure 1 as:

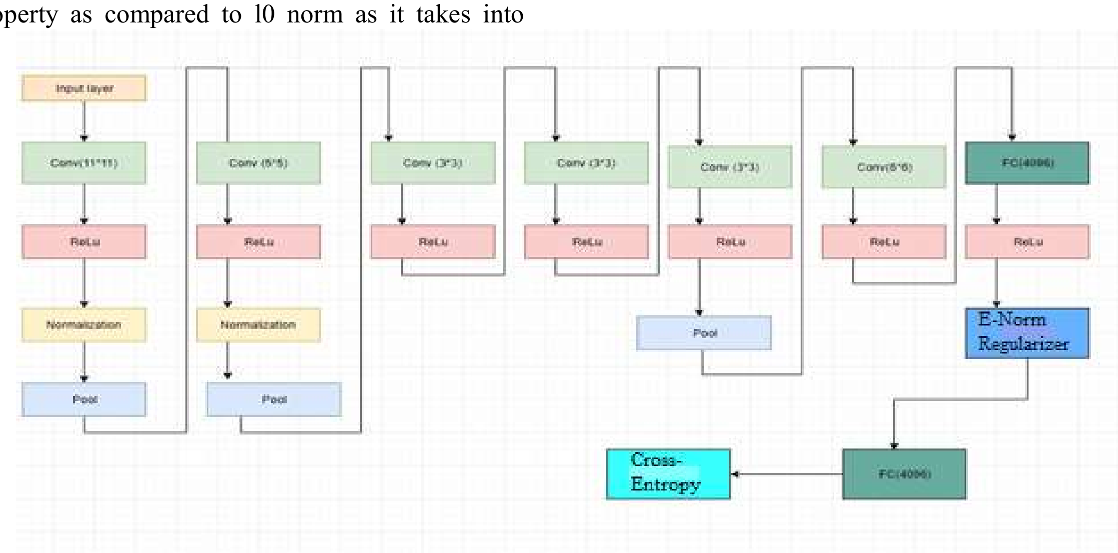



*Figure 1: Graphical description of VGG-19 model for numerical evaluation*

Journal of Theoretical and Applied Information Technology
30th April 2022. Vol.100. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The VGG-19 model was implemented on all three datasets and the results are observed for the initial learning rate of 0.001. For every 60 epochs, and learning rate decays by a factor of 0.1. No other regularization method like dropout is used. Except the network implements batch normalization regularization method in combination with the proposed e-norm regularizer for varying values of $\alpha = (0.1, 0.2, 0.3, 0.4, 0.5)$. The observed results are recorded in Table 2 as below: This table summarizes the results for testing error rate, sparsity of weights, and neuron sparsity across 10 runs of vgg-19 model after every 250 epochs.

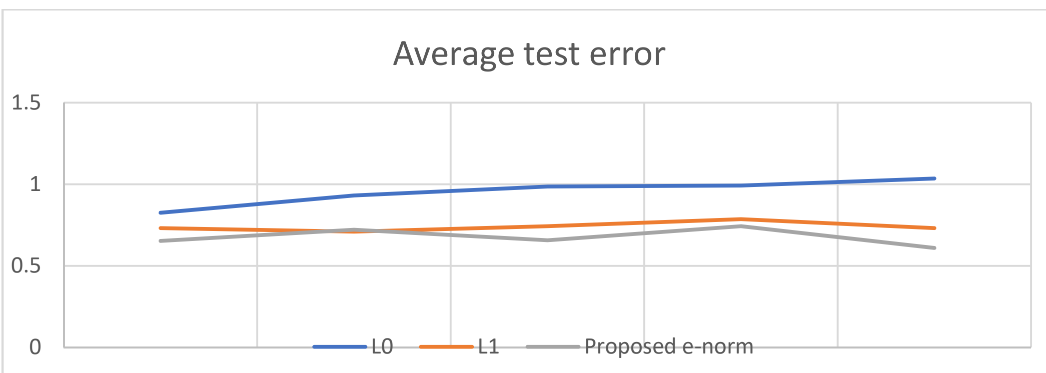
*Table 2: Average test error, weight sparsity and Neuron sparsity for MNIST dataset*

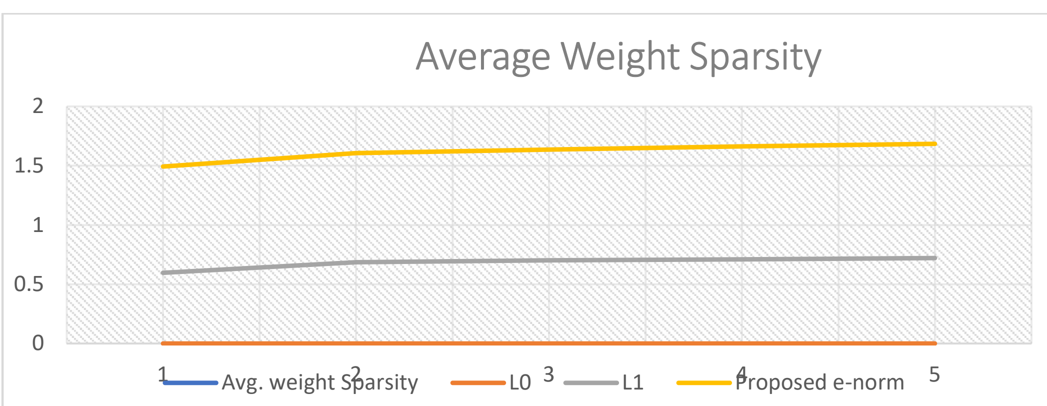
| **Avg. test error(%)** | **L0** | **L1** | **Proposed e-norm** |
|---|---|---|---|
| $\alpha = 0.1$ | 0.8261 | 0.7321 | 0.6521 |
| $\alpha = 0.2$ | 0.9321 | 0.7104 | 0.722 |
| $\alpha = 0.3$ | 0.9856 | 0.7421 | 0.656 |
| $\alpha = 0.4$ | 0.9923 | 0.7856 | 0.742 |
| $\alpha = 0.5$ | 1.0345 | 0.7320 | **0.6102** |

| **Avg. weight Sparsity** | **L0** | **L1** | **Proposed e-norm** |
|---|---|---|---|
| $\alpha = 0.1$ | $2.11 \times 10^{-4}$ | 0.5962 | 0.896 |
| $\alpha = 0.2$ | $2.15 \times 10^{-4}$ | 0.6861 | 0.921 |
| $\alpha = 0.3$ | $2.19 \times 10^{-4}$ | 0.7032 | 0.932 |
| $\alpha = 0.4$ | $2.05 \times 10^{-4}$ | 0.7102 | 0.953 |
| $\alpha = 0.5$ | $2.28 \times 10^{-4}$ | 0.7201 | **0.964** |

| **Avg.Neuron Sparsity** | **L0** | **L1** | **Proposed e-norm** |
|---|---|---|---|
| $\alpha = 0.1$ | 0.5201 | 0.6896 | 0.6923 |
| $\alpha = 0.2$ | 0.5628 | 0.6923 | 0.7102 |
| $\alpha = 0.3$ | 0.6102 | 0.7123 | 0.7532 |
| $\alpha = 0.4$ | 0.6143 | 0.7256 | 0.7821 |
| $\alpha = 0.5$ | 0.6326 | 0.7899 | **0.8123** |

**a) Graphical simulated results on MNIST Dataset depicted in fig 2. a) test error rate b) weight sparsity c) neuron sparsity**



Journal of Theoretical and Applied Information Technology
30th April 2022. Vol.100. No 8

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

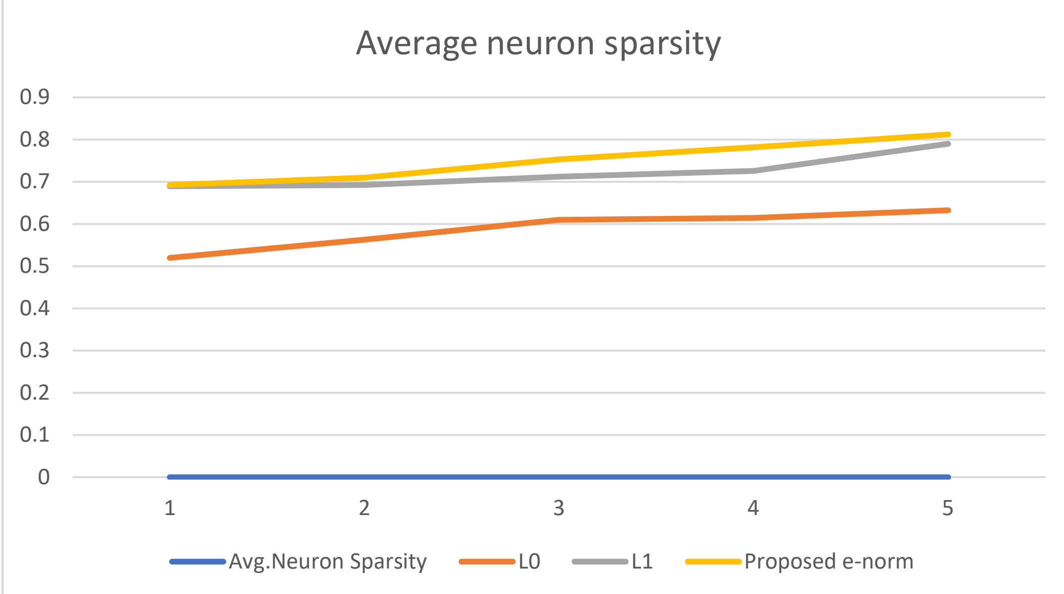



*Figure 2: Graphical visualization of Table 1. On MNIST dataset*

For MNIST dataset vgg-19 across all the parameters and methods achieves the highest neuron weight sparsity of 81%, weight sparsity of 96%, and least error rate of 61% at $\alpha = 0.5$ as is clear from the above graphical observations and tabular results recorded.

b) **Average error rate of CIFAR-10 dataset**.

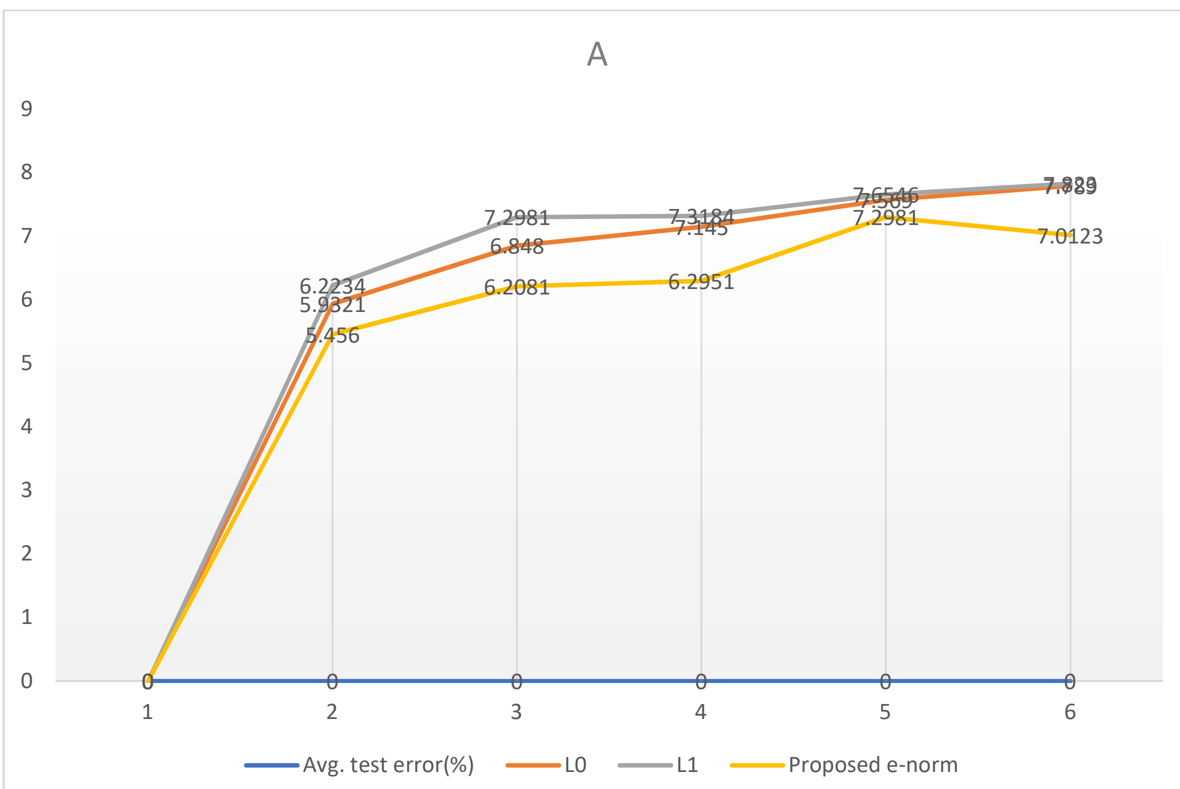



*Figure 3: Graphical visualization of Avg. test error. On CIFAR-10 dataset*

c) **Average test error rate on ImageNet dataset**.

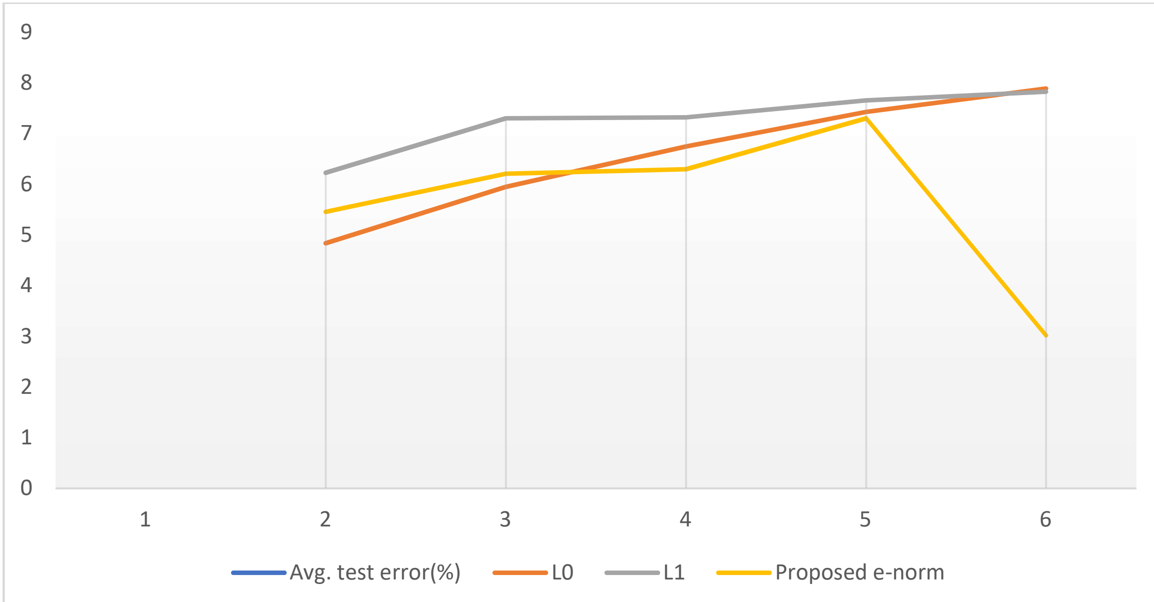



*Figure 3:. Graphical visualization of Avg. test error. On ImageNet dataset*

From the above graphical results obtained it is clear that our proposed method achieves the least error rate on testing all the three databases across all parameters and methods. On varying values of $\alpha$ the proposed network with the novel achieves weight sparsity of 42% and average neuron sparsity of 54% on Cifar-10 dataset, and achieves 45% of weight sparsity and 79% of neuron sparsity on ImageNet database. Thus outperforming all the existing regularization method by gaining remarkable results.

## 6. CONCLUSION

The regularizer for generic neural network based on e-norm with batch normalization is used in this work. The proposed regularization method is embedded in vgg-19 model in addition with batch normalization implemented on three datasets to achieve the results. The experimental results obtained demonstrate that in general our proposed novel regularizer achieves least test error rate and achieves greater sparsity thus greatly helps in compressing dense neural networks.

The experimental results obtained demonstrate that in general our proposed novel regularizer achieves least test error rate and achieves greater sparsity thus greatly helps in compressing dense neural networks.

## 7. CRITICS OF THE RESEARCH WORK

According to the numerical results, on any CNN trained on a given dataset, no single sparse regularizer outperforms all others. One regularizer may be effective in one scenario while it may perform worse on a another case. Due to the plethora of sparse regularizer available and the numerous parameters to fine-tune, particularly for one CNN that has been trained on. One approach is to create an automatic classification system for a given dataset. A framework for machine learning that efficiently identifies the best candidates regularizer as well as parameters Automatic systems have been used in recent works. Matrix completion can be used to depict machine learning. a statistical learning issue and a problem. These Frameworks can be tweaked to find the best sparse data.

## 8. FUTURE SCOPE

In Future the novel regularizer will be implemented on other types of networks such as LSTM and RNN's across other benchmark datasets. we will also focus on constructing even deeper convolutional neural networks using transfer learning and data augmentation on all the layers of the network for biomedical applications with even lesser training time and improved accuracy rate.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

**REFERENCES:**


[1]. James G, Witten D, Hastie T, Tibshirani R. An Introduction to Statistical Learning [Internet]. Springer Texts in Statistics. 2013. 618 p. Available from: http://books.google.com/books?id = 9tv0taI8l6YC.

[2]. Huang, J., Jiao, Y., Liu, Y., & Lu, X. (2018). A constructive approach to L0 penalized regression.

[3]. Ranstam, J., & Cook, J. A. (2018). LASSO regression. Journal of British Surgery, 105(10), 1348-1348.

[4]. Tsigler, A., & Bartlett, P. L. (2020). Benign overfitting in ridge regression. arXiv preprint arXiv:2009.14286.

[5]. Hejazi, N. S., Coyle, J. R., & van der Laan, M. J. (2020). hal9001: Scalable highly adaptive lasso regression inR. Journal of Open Source Software, 5(53), 2526.

[6]. Amini, F., & Hu, G. (2021). A two-layer feature selection method using genetic algorithm and elastic net. Expert Systems with Applications, 166, 114072.

[7]. Fukunishi, H., Nishiyama, M., Luo, Y., Kubo, M., & Kobayashi, Y. (2020). Alzheimer-type dementia prediction by sparse logistic regression using claim data. Computer Methods and Programs in Biomedicine, 196, 105582.

[8]. Karmalkar, S., & Price, E. (2018). Compressed sensing with adversarial sparse noise via $L_1$ regression. arXiv preprint arXiv:1809.08055.

[9]. Hussein, R., Elgendi, M., Wang, Z. J., & Ward, R. K. (2018). Robust detection of epileptic seizures based on L1-penalized robust regression of EEG signals. Expert Systems with Applications, 104, 153-167.

[10]. Dobriban, E., & Wager, S. (2018). High-dimensional asymptotic of prediction: Ridge regression and classification. The Annals of Statistics, 46(1), 247-279.

[11]. Liu, S., & Dobriban, E. (2019). Ridge regression: Structure, cross-validation, and sketching. arXiv preprint arXiv:1910.02373.

[12]. Hesamian, G., & Akbari, M. G. (2019). Fuzzy lasso regression model with exact explanatory variables and fuzzy responses. International Journal of Approximate Reasoning, 115, 290-300.

[13]. Hejazi, N. S., Coyle, J. R., & van der Laan, M. J. (2020). hal9001: Scalable highly adaptive lasso regression inR. Journal of Open Source Software, 5(53), 2526.

[14]. Alhamzawi, R., & Ali, H. T. M. (2018). The Bayesian adaptive lasso regression. Mathematical biosciences, 303, 75-82.

[15]. Topuz, K., Uner, H., Oztekin, A., & Yildirim, M. B. (2018). Predicting pediatric clinic no-shows: a decision analytic framework using elastic net and Bayesian belief network. Annals of Operations Research, 263(1), 479-499.

[16]. Ahrens, A., Hansen, C. B., & Schaffer, M. (2020). LASSOPACK: Stata module for lasso, square-root lasso, elastic net, ridge, adaptive lasso estimation and cross-validation.

[17]. Van Erp, S., Oberski, D. L., & Mulder, J. (2019). Shrinkage priors for Bayesian penalized regression. Journal of Mathematical Psychology, 89, 31-50.

[18]. Y. LeCun, C. Cortes, and C. J. Burges, "The mnist database of handwritten digits, 1998," 1998. [Online]. Available: http://yann. lecun. com/exdb/mnist.

[19] A. Krizhevsky, V. Nair, and G. Hinton, "The cifar-10 Dataset," 2014 [Online]. Available: http://www.cs.toronto.edu/kriz/cifar.html.

[20] O. Russakovsky et al., "ImageNet large scale visual recognition challenge," Int. J. Comput. Vis., vol. 115, no. 3, pp. 211–252, 2015.

[21]. Habib, G., & Qureshi, S. (2020). Biomedical Image Classification using CNN by Exploiting Deep Domain Transfer Learning. International Journal of Computing and Digital Systems, 10, 2-11.