The Effects of LP-norm Regularization on Performance

Hawoo Jung
20210054

Sangwon Kim
20232697

Janghyeok Han
20242832

Abstract

Regularization is a crucial technique in deep learning used to mitigate overfitting and enhance model generalization. This report investigates various norm-based regularization methods, specifically L1, L2, and L3 norms, and their effectiveness in classification tasks. Using the MNIST dataset for image classification and the California Housing Prices dataset for feature selection and regression analysis, we evaluate the L1, L2, and L3's convergence speed and model performance. Our research reveals that L1 regularization underperforms in model performance compared to L2 and L3, which show equivalent results. In terms of convergence speed, L1 regularization is the fastest, followed by L2 and L3. Additionally, L2 and L3 norms demonstrate superior ability to capture important features, contributing to their enhanced performance. These insights offer practical guidance for selecting appropriate regularization techniques in various deep learning applications. You can check our works at https://github.com/DaramGC/AIGS538-Deep-Learning-Final-Project/tree/main.

1 Introduction

Regularization in deep learning is a technique used to prevent overfitting, ensuring that models generalize well to unseen data.[1, 9, 5] Overfitting occurs when a model learns not only the underlying patterns in the training data but also the noise, leading to poor performance on new data. Regularization techniques introduce a penalty term to the loss function, which discourages the model from becoming too complex and overly tailored to the training data. Among these techniques, norm-based regularizations, such as L1[4] and L2[2, 6] norms are widely used. Norm-based regularizations involve calculating the norm of the model parameters and adding it as a penalty to the loss function. The LP-norm[7], defined as

$$\|\omega\|_p = (\sum_i |\omega_i|^p)^{1/p} \tag{1}$$

where P is the order of the norm, provides a measure of the magnitude of the weights. L0 and L1 norms have the ability to perform feature selection by driving certain parameters to zero, thus removing irrelevant features. The L2 norm, on the other hand, is a common regularization method that penalizes large weights uniformly, leading to smoother models. The strength of the regularization is controlled by a hyperparameter λ, with larger values of λ increasing the penalty and thus the regularization effect. This report explores the application and effectiveness of various norm-based regularization methods in classification tasks. We focus on how L1, L2, and L3 norms influence model performance, particularly in terms of feature selection and classification accuracy. Our experiments are conducted using the MNIST dataset, a benchmark for image classification, and the California Housing Prices dataset, which provides a diverse feature set for evaluating feature selection methods. By comparing these regularizations, we aim to elucidate their respective advantages and provide insights into their practical applications in deep learning models.

2 Method

We will check three things in our experiments. First, model performance according to norm-based regularizers. Second, model convergence speed among various norm-based regularizers. Lastly, visualizing regularized weights for checking how each regularizer affects model performance and convergence speed. We only give penalty on the first layers of our three layer multi-layer perceptron model. We use two different datasets for regression task and classification task. Both datasets are normalized by mean and standard deviation.

To compare model performance along with norm-based regularizers, we calculate accuracy for classification task and mean square error for regression task. We assume that higher accuracy means better performance in classification task and lower mean square error means better performance in regression task. For checking convergence speed, we use loss graph to find minimum changes of loss. However, loss graph are not good enough to show convergence of model, we additionally show visualized weight maps. Lastly for visualizing weight maps, we show weights of first layers in gray scale images. This can help understand the role of norm-based regularizers and their effects. To make weights value between 0 and 1 for gray scale image, we use following equation,

$$\omega_{rescale} = \frac{|w_{original}|}{max(|w_{original}|)} \tag{2}$$

where $w_{original}$ is weights of first multi-layer perceptron and $w_{rescale}$ is rescaled weights which satisfy $w_{rescale} \in [0, 1]$. By doing so, we can project weights to [0,1] well.

3 Experiment

We performed experiments on two datasets, MNIST[3] for classification task and California Housing Prices[8] for regression task. Three-layers MLP models were used in the experiment (128 hidden perceptrons for regression task and 100 hidden perceptrons for classification task) and we applied norm-based regularizer(L1, L2, L3) only on the first layer. Plotting of loss (for regression task) and accuracy (for classification task), and the changing process of visualized weight map for each epoch of specific interval were used to compare the performances of each regularizer. All trains are done in batch size 256 and learning rate 0.0001, 100 and 1000 iteration steps for classification and regression tasks each.

3.1 MNIST experiment

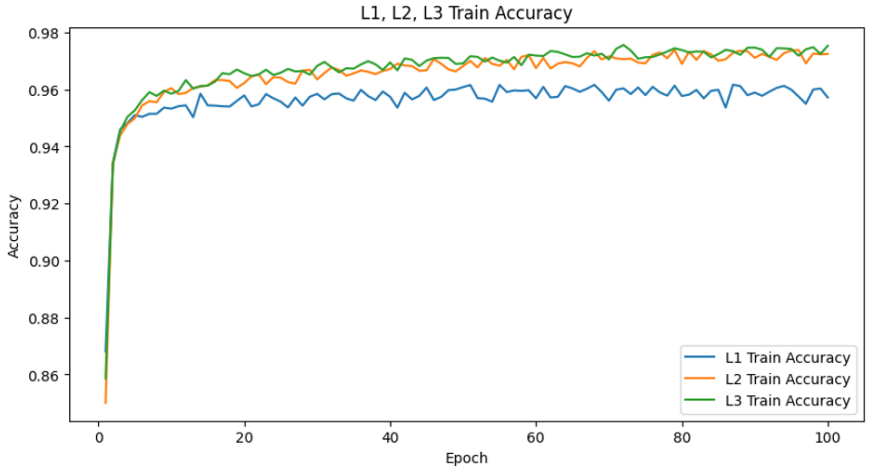

Figure 1: Train accuracy plots of model L1, L2, L3.

In MNIST set, Figure 1 shows that L2 and L3 have similar performance, and performance of L1 was worse than the others. Also, we could see convergence speed was fastest in the following order: L1,

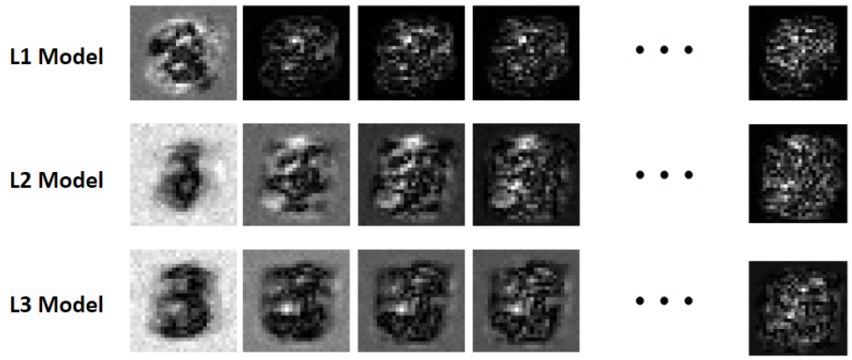

Figure 2: Weight map changes in model L1, L2, L3

L2, and L3. (Graph of comparing convergence speed is contained in Appendix, Figure 5.) Likewise, in the weight map in Figure 2, it is visually confirmed that the convergence speed orders are the same with the result above. Additionally, we found that the final weight maps look different from each other due to distinct regularizing ability.

3.2 California Housing Prices experiment

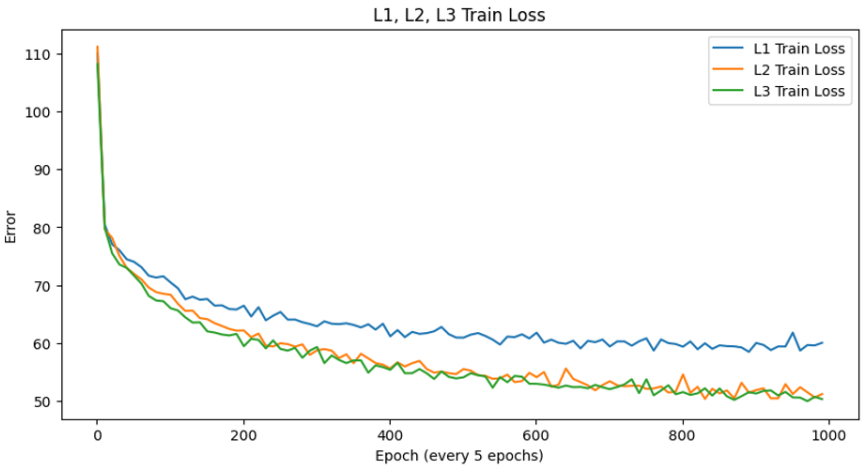

Figure 3: Train loss plots of model L1, L2, L3.

In California Housing Prices set, Figure 3 shows that the model performance order is L1 < L2 = L3, same as MNIST set. And the convergence speed order was L1 > L2 = L3, as can be seen in the loss graph and weight map in Figure 4. (Graph of comparing convergence speed is contained in Appendix, Figure 7.) In the weight map, Brighter pixel means it has a bigger relation. According to the experiment, we can conclude that 'latitude' has the biggest relation with price when we use L1. However, in common sense, latitude does not seem to have a significant effect on price. So, we assumed that the performance of L1 was low compare that L2 and L3 which weighted less to the feature 'latitude'.

4 Conclusion

In the paper, we proposed an experiment of comparing the performance of norm-based regularization methods on regression and classification task. We chose using a method to visually represent

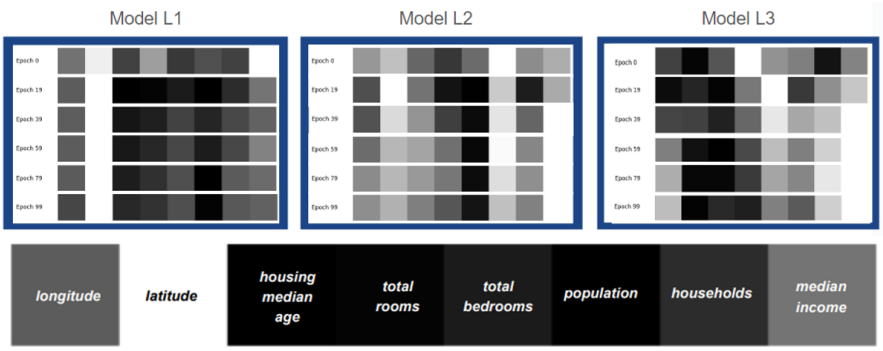

Figure 4: Weight map changes in model L1, L2, L3 and final map result in L1.

changes in the weight map, not only loss and accuracy graph. This helps us to understand effects of performance and convergence of norm-based regularizers. However, we cannot find outstanding difference between L2 and L3 regularization in our experiments. We leave finding new methods to compare L2 and L3 regularization that work well for future works.

References

[1] Bishop, C.M., Nasrabadi, N.M.: Linear models for regression. In: Pattern Recognition and Machine Learning. Vol. 4. Springer. pp. 137–178 (2006)

[2] Corinna Cortes, Mehryar Mohri, A.R.: L2 regularization for learning kernels. In: arXiv preprint arXiv:1205.2653, 2012

[3] Deng, L.: Ieee signal processing magazine, 29(6). In: The MNIST database of handwritten digit images for machine learning research. pp. 141—142 (2012)

[4] Gen Li, Yuantao Gu, J.D.: The efficacy of l1 regularization in two-layer neural networks. In: arXiv preprint arXiv:2010.01048, 2020

[5] Goodfellow, I., Bengio, Y., Courville, A.: Regularization for deep learning. In: Deep Learning. The MIT Press. pp. 230–237 (2016)

[6] van Laarhoven, T.: L2 regularization versus batch and weight normalization. In: arXiv preprint arXiv:1706.05350v1, 2017

[7] Moravec, J.: A comparative study: L1-norm vs. l2-norm; point-topoint vs. point-to-line metric; evolutionary computation vs. gradient search. In: Applied Artificial Intelligence, 29(2). pp. 164—-210 (2015)

[8] Pace, R.K., Barry, R.: Statistics and probability letters. volume 33. number 3. may 5. In: Sparse Spatial Autoregressions. pp. 291–297 (1997)

[9] Simon, J.D.P.: Regularization. In: Understanding Deep Learning. The MIT Press. pp. 140–141 (2023)

A Appendix

Figure 5: Train and test accuracy graph in model L1, L2, L3 (MNIST).

Blue plot is train accuracy, and the orange plot is test accuracy. Red horizontal line means the point where the models converge. We can see the converge speed order is L1 > L2 > L3.

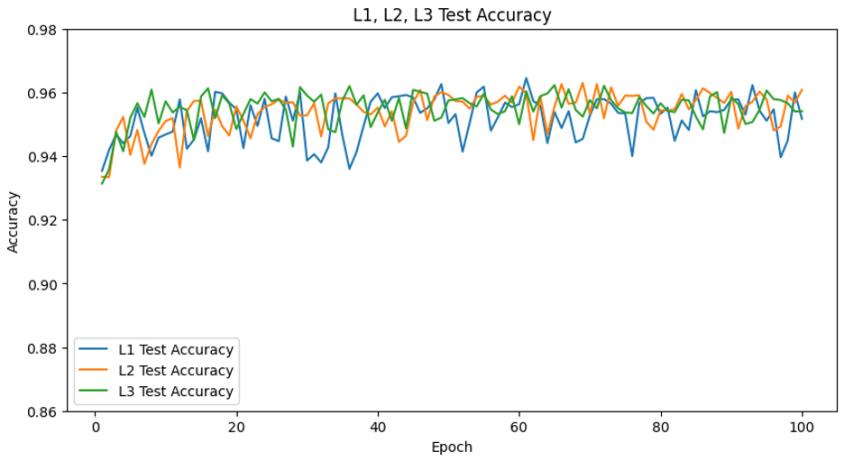

Figure 6: Test accuracy plots of model L1, L2, L3. (MNIST).

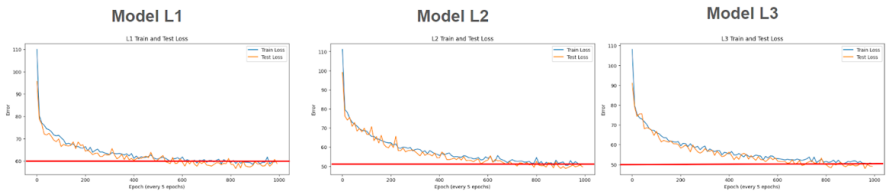

Figure 7: Train and test accuracy graph in model L1, L2, L3 (California Housing Prices).

Blue plot is train loss, and the orange plot is test loss. Red horizontal line means the point where the models converge. We can see the converge speed order is L1 > L2 = L3.

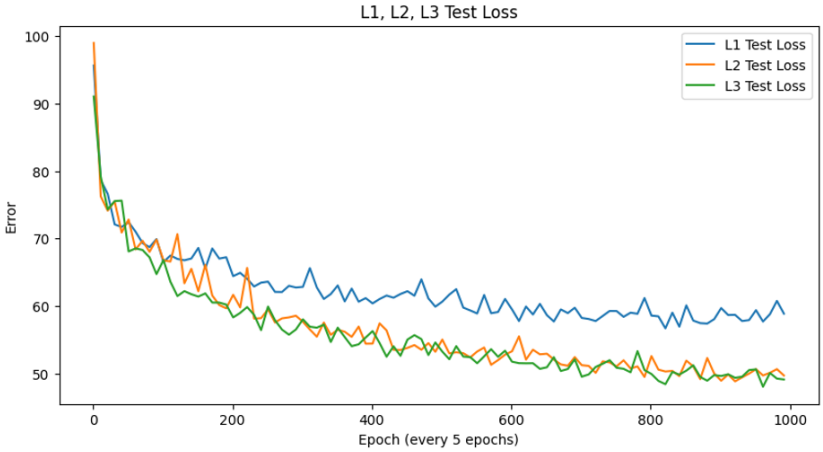

Figure 8: Test loss plots of model L1, L2, L3. (California Housing Prices).