Assignment 3

Anabelle Amiot 261054336, Aryan Chaturvedi 260976059,
Shahin Jowkar Dris 260989348

April 7, 2024

1 Abstract

We aimed to implement a multilayer perceptron and use it to classify images representing various letters signed in sign language. We wanted to test how its accuracy varied by changing the number of hidden layers as well as the number of units in each. Overall, our findings were that the highest accuracy on the dataset was found with a 2 hidden layer MLP with 128 units per hidden layer and a learning rate of 0.0005, with a classification accuracy of 99.65%. When comparing the most optimal performance of all three models we worked with, the lowest accuracy was obtained with the MLP with no hidden layer (66.60 %). Overall, these results indicate that increasing the number of hidden layers in the MLP leads to a higher accuracy.

2 Introduction

Multilayer Perceptron is a type of neural network composed of multiple layers. All consecutive ones are fully connected amongst themselves [1]. They can have different numbers of hidden layer, containing different numbers of units and use different activation functions, among other modifiable elements. The goal of this project is to explore these different hyperparameters and how they impact the accuracy of correctly classifying the letter shown by sign language from the MNIST sign language dataset. Studying these types of datasets also has a huge impact on increasing accessibility for people who have a form of deafness or speech impairment [2]. That dataset is a standard for machine learning, which means the prediction accuracies using different techniques on this model are very well documented. High levels of accuracy (99%) can be obtained on this set using Convolutional Neural Network [2].

3 Datasets

The dataset we were working with consisted of 27 455 images, each depicting hands singing letters in sign language. Each image has a label which corresponds to the letter being represented. Two of the letters in the sign language alphabet require hand-motion (A and J) and are therefore not included, since we are only considering letters represented by a single image frame. Therefore, we have 24 labels that will be one hot encoded from 1 - 25, in alphabetical order. The 784 pixels that make up each image are stored as the data we will be using to classify. We also normalized the data using the method of mean subtraction. After this, we split the data into training, validation and testing sets, with the following ratio : [0.7, 0.15, 0.15].

4 Results

We implemented an MLP with no hidden layer, an MLP with one hidden layer that used ReLu activations and an MLP with two hidden layers also with ReLu activations. For our hyperparameter testing, we started by trying out various learning rates (0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05)for each of these three models. For the MLP with no hidden layer, as shown in Fig. 1 we obtained the highest accuracy (66.60%) with a learning rate of 0.001. For the MLP with 1 hidden layer, using a width of 32, we obtained an accuracy of 91.33% using a learning rate of 0.005. For the one with 2 hidden layers, we managed to get an accuracy of 91.24% when the learning rate was set to 0.001, as shown in Fig. 3.

Once we had established the most optimal learning rates , we proceeded to test different number of units per layer (32, 64, 128 , 256) for 1 hidden layer MLP and 2 hidden layer MLP. For the 1 hidden layer MLP, the number of hidden units with the highest accuracy was 64 units (98.57%), see Fig. 4. For the 2 hidden layer MLP, the same test was done, but using all possible combinations of these number of units. This resulted in a maximal accuracy of 99.52% when both hidden layers contained 64 units 5.

We expected no hidden layer MLP to have the worst performance, followed by 1 hidden layer MLP and 2 hidden layer MLP to have the best performance. This is the result we observed through our experiments. There was a significant gap in performance between the MLP with no hidden layer and the one with one (31.97%) compared to the difference between the MLP with one hidden layer and the one with two, which was only 0.95% greater

Even though the increase in accuracy is slight, the difference in cross-entropy loss convergence is not. When comparing Fig. 6 and Fig. 7, we can observe that the 2 hidden layer MLP converges faster than the 1 hidden layer MLP. This highlights the fact that increasing the network depth can allow for a more complex understanding of the features used. Using the models we had, increasing the depth of the MLP did lead to an increase in accuracy, as expected.

The previous implementation of 2 hidden layer MLP we were working with used ReLu activations. Using the optimal learning rate we have already found, we reran the tests with two different activation functions (sigmoid and Leaky-Relu). The results showcased in Fig. 8 show that the highest accuracy was found with the ReLu activation. We had expected Leaky ReLu to be the activation function with the highest accuracy, due to its role in preventing the vanishing gradient problem. However, the hyperparemeters used for all three of these models had been tuned by testing done using ReLu as the activation functions, which could explain why ReLu has a higher accuracy in this case.

When using MLP, overfitting can occur and L2 regularization is a well-know technique that can help prevent this by adding a penalty to weights.We implemented this techniques in our models by adding $\lambda||W||^2$ to the values that were being subtracted from the weights on all layers. Different values of λ were tested (0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1, 2), and as Fig.9 shows, the one that resulted in the highest accuracy for MLP was 0.0001. Overall, we expected the use of L2 regularization to improve the accuracy since it would contribute to the reduction of potential overfitting happening in our models. However, this was not the case with the highest accuracy we got being 88.87%, compared to the 99.52% we had previously got. Further testing of different lambda values would be required to see if L2 regularization could be used to improve accuracy.

We also implemented a Convolutional Neural Network with 3 convolutional layers as well as 2 fully connected ones. This was done using the Keras library. We performed a similar analysis to what had been done for previous MLP models we tested, and found that the highest accuracy on this implementation was with 256 hidden units in both ReLU layers. We also found that the ideal learning rate was 0.001. Through the selection of the most optimal hyperparameters, we were able to obtain an accuracy of (94.26%). We also tested different values for strides used in both the convulational and fully connected layers, see Fig. **??**

Through the various hyperparameters we tested on different models, we concluded that the MLP architecture with the highest classification accuracy on the MNIST sign language dataset was (99.96%) using the 2 Layer MLP with 128 and 128 hidden units , see Fig. 12.

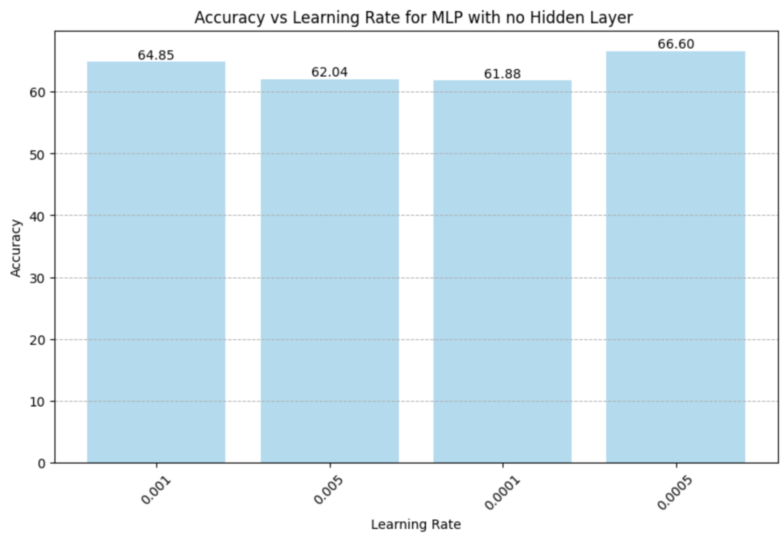

Figure 1: Accuracy (%) at various learning rates for MLP with no hidden layer.

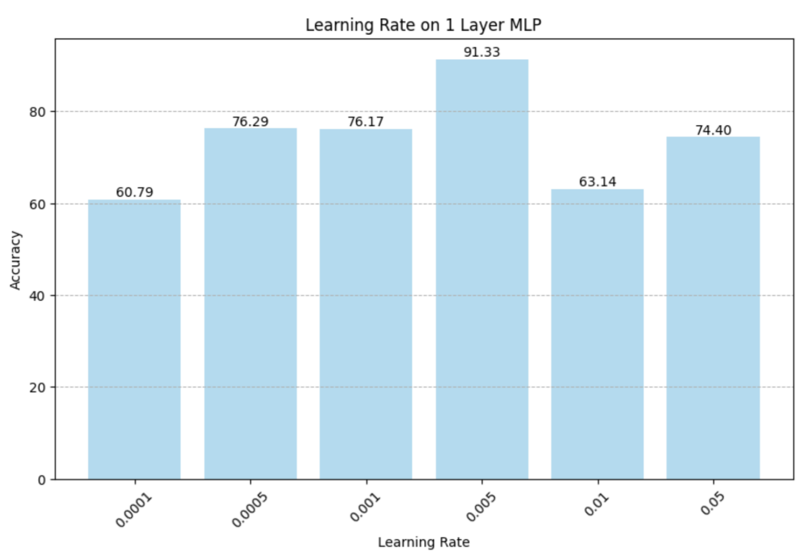

Figure 2: Accuracy (%) at various width for MLP with 1 hidden layer.

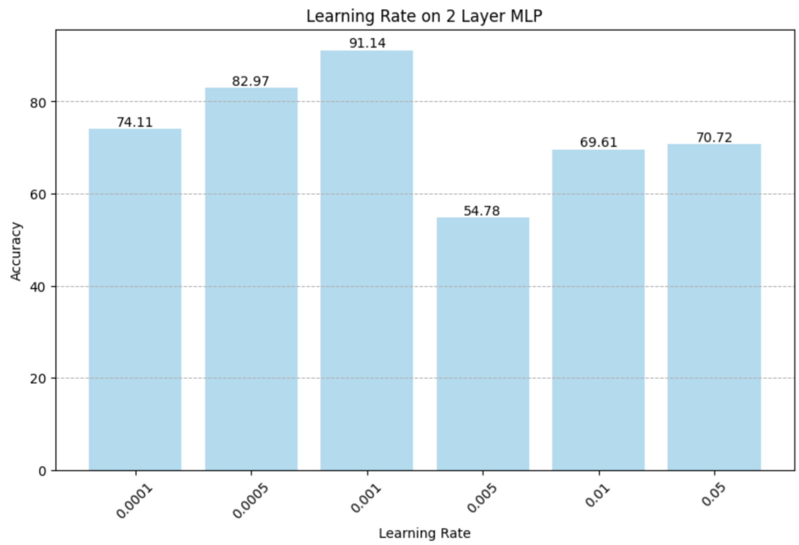

Figure 3: Accuracy (%) at various learning rates for MLP with 2 hidden layers containing 128 units.

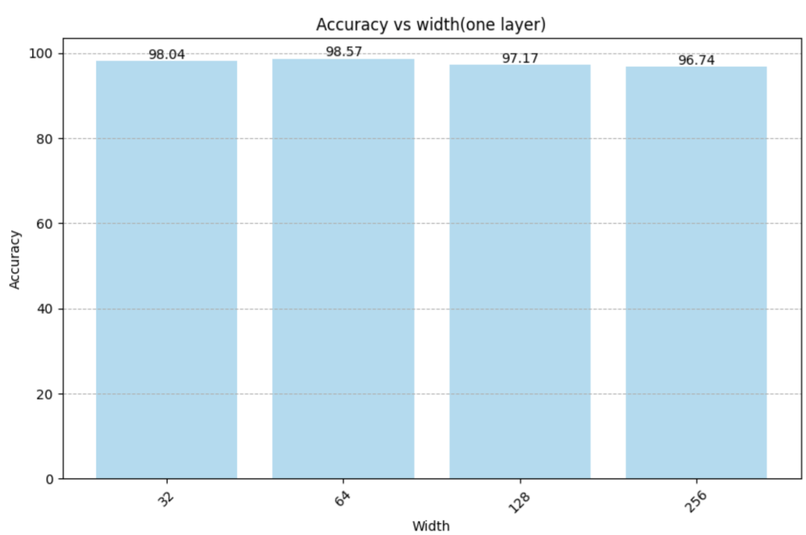

Figure 4: Accuracy (%) at various width for MLP with 1 hidden layer.

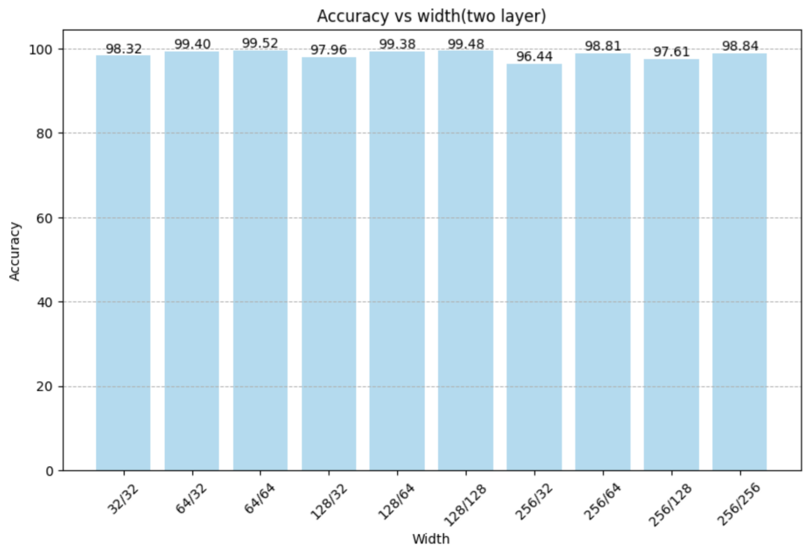

Figure 5: Accuracy (%) at various learning rates for MLP with 2 hidden layers containing 128 units.

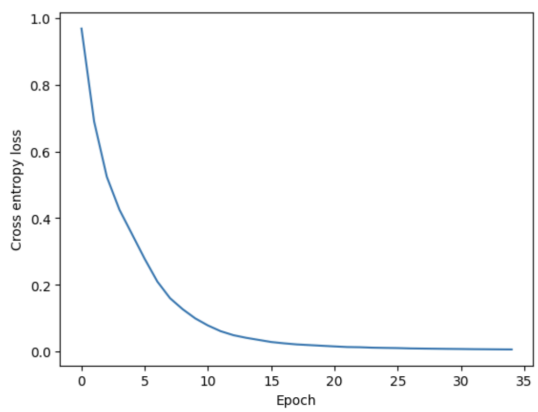

Figure 6: Cross entropy loss vs epochs for 1 hidden layer MLP with 64 units.

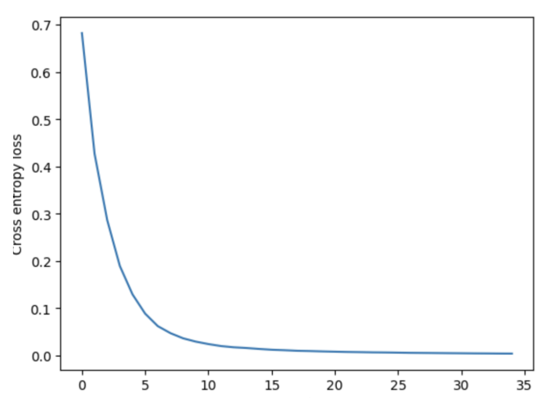

Figure 7: Cross entropy loss vs epochs for 2 hidden layer MLP with 64 units.

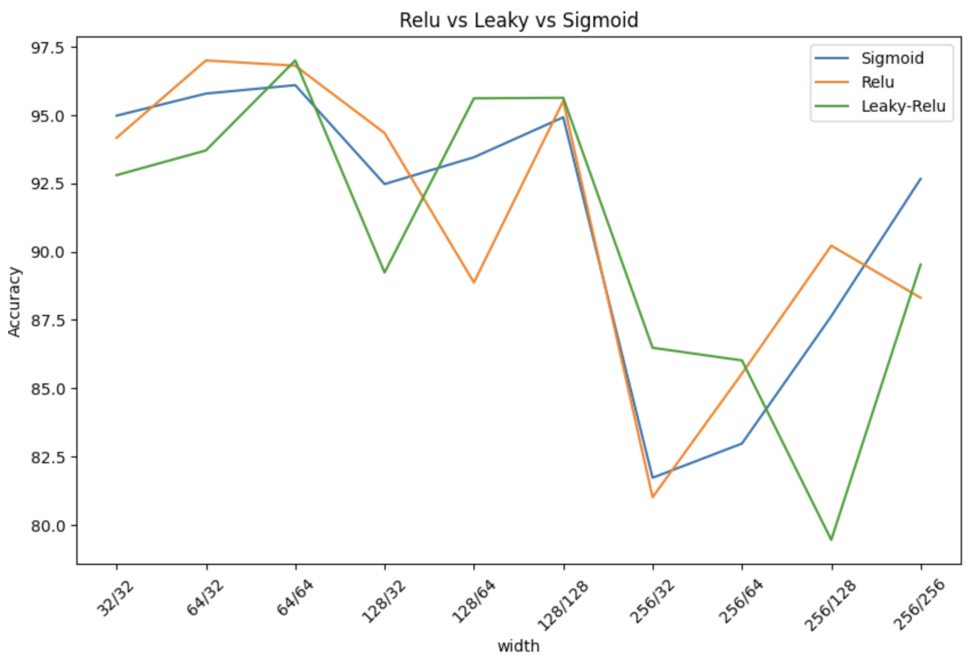

Figure 8: Accuracy (%) at different width for 2 hidden layer MLP using different activation functions.

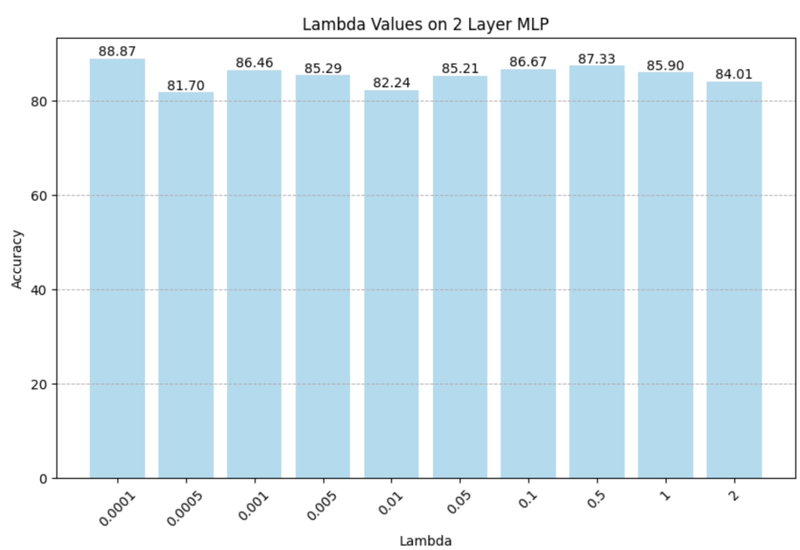

Figure 9: Accuracy (%) of 2 layer MLP using L2 Regularization for different λ values.

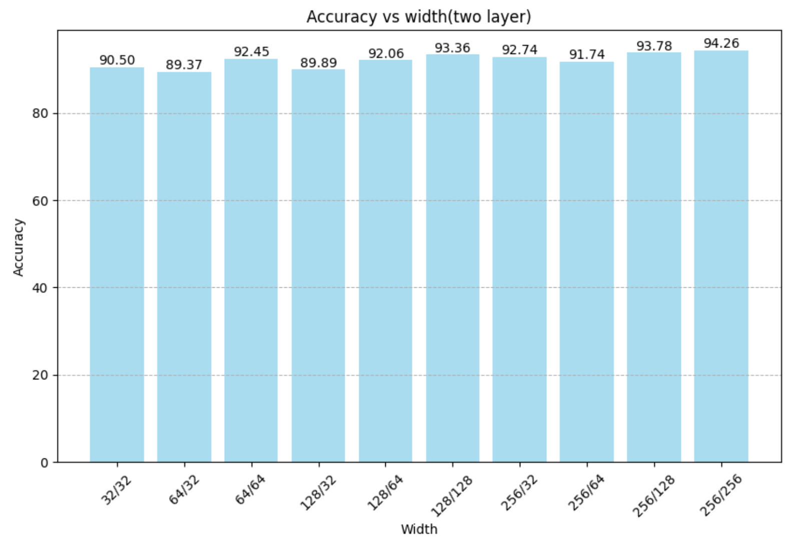

Figure 10: Accuracy (%) for various number of hidden units used in each layer of the CNN.

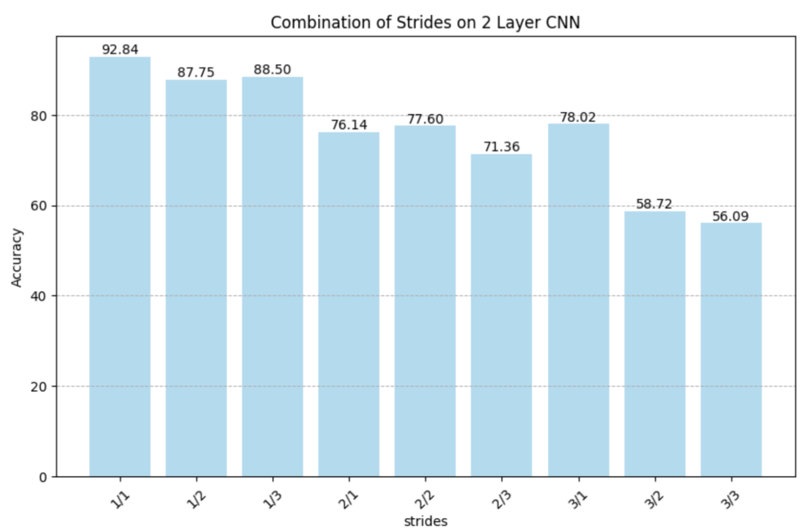

Figure 11: Accuracy (%) for various strides used in convulational and fully connected layers of the CNN.

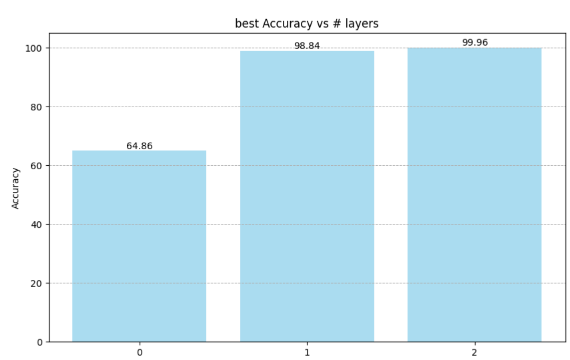

Figure 12: Comparisons of the best performances of all 3 models.

5 Discussion and Conclusion

Overall, our results showed that increasing the network depth leads to a higher accuracy and to a faster convergence of the model, which is what we expected. The activation function that resulted in the highest accuracy was ReLu, however, we have reason to believe Leaky ReLu would perform better if we had chosen all hyperparameters while using that function. We compared our results to that of the Convolutional Network created using Keras and found that our accuracy was comparable. Our results also showed that an increase of the number of layers per hidden layer did not necessarily lead to an increase of accuracy. Furthermore, through our testing, we really noted the importance of using a validation set, which resulted in an improvement of over 20% compared to the results we were getting prior without using one.

For future analysis, we would be interested in testing different batch sizes. We also would like to analyze the impact that the numbers of filter, their size as well as the padding has on the accuracy of the models since they are not apart of the hyperparameters we analyzed in this project.

6 Statement of Contribution

Anabelle, Aryan and Shahin all worked on all tasks as well as the report.

References

[1] Marius-Constantin Popescu, Valentina E Balas, Liliana Perescu-Popescu, and Nikos Mastorakis. Multilayer perceptron and neural networks. *WSEAS Transactions on Circuits and Systems.*

[2] RS Sabeenian, S Sai Bharathwaj, and M Mohamed Aadhil. Sign language recognition using deep learning and computer vision. *J Adv Res Dyn Control Syst*, 12(5 Special Issue):964–968, 2020.