MLNS - DL Assignment Report (Final)

Kapil Rajesh Kavitha - 2021101028

Baseline Approach

- Try to identify regions around the digits in the image
- Pad these regions and resize the images (these were left incomplete due to time constraints)
- Ultimately, the base images were used.
- Using these images, train a CNN with 50 classes (Arbitrary assumption for a classification-based approach, regression can be explored too)

Results:

Final Results:

- Training Accuracy: 16.56%
- Validation Accuracy: 17.65%

Test: 5.77%

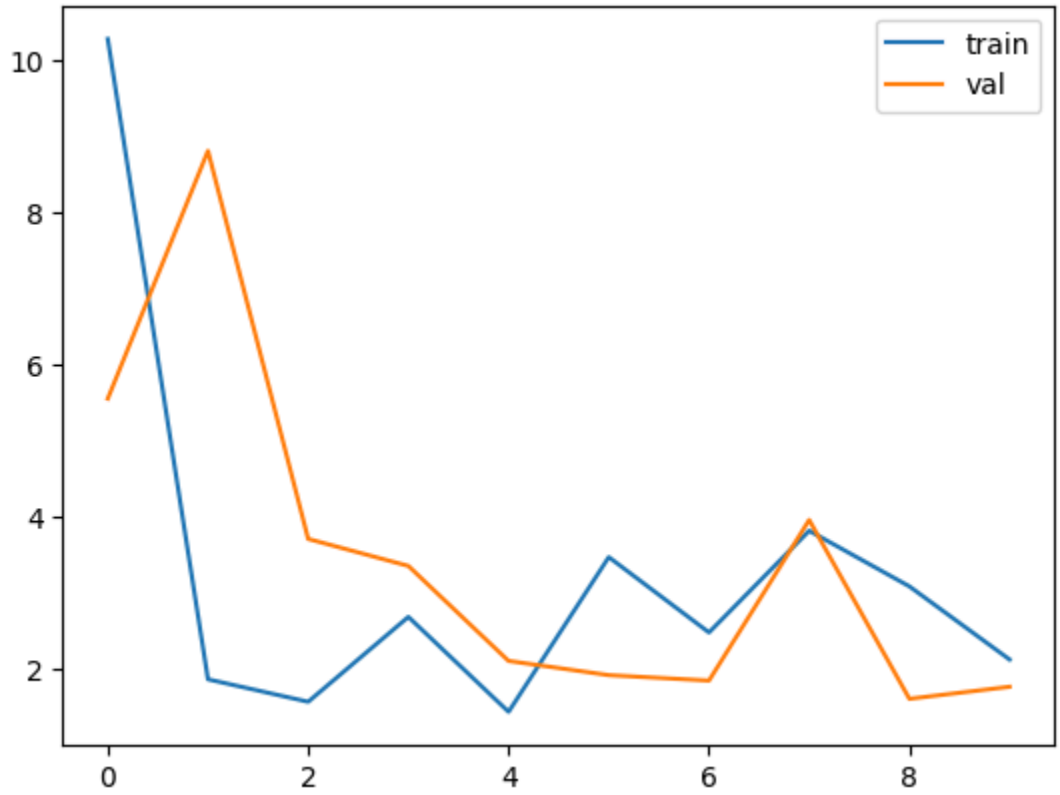
Intermediate Approaches

- Initially, I attempted to complete the approach I started as part of the baseline, by identifying regions around digits.
- The identified digits would be stored in an array called ‘resized_images’.
- An **MNIST** model was trained to identify each individual digit identified from a particular image, and the sum of the identified digits was used as the prediction for the image.
- This approach, although it showed promise, yielded unsatisfactory results with a test accuracy of **8%**

Final Approach

- The final approach involved the use of a **modified ResNet-50** model. This model was used to try and utilize the power of the feature extraction capabilities from ImageNet.
- ‘Data0.npy’ was used as the train set, ‘data1.npy’ as the validation set, and ‘data2.npy’ as the test set.
- This model was modified to perform the given task using **regression.**
    - `self.fc = nn.Sequential( torch.nn.Linear(128, 1)`
    - The “classification” model is modified to output a single scalar value which represents the sum of digits in the image.
- The first convolution of the model is also modified to accept a single channel, so that it can handle grayscale images.

Loss curves from final approach



Results obtained

- Train accuracy: **45.62%**
- Validation accuracy: **41.41%**
- Test accuracy: **41.77%**

These results represent a big step up from the other attempts. But there is greater scope for improvement in scores for all the approaches which were tested out.