A4 Write-up

Members:
Khabiir Khodabaccus
Takuya Ishii
Tyra David

1. Implementation of the model

The model we chose is the convolutional neural network as it requires less preprocessing and fits well with a classification computer vision problem. Some of the preprocessing we implemented includes background noise reduction, one-hot encoding and data augmentation.

2. Results

Our best submission yielded 0.948 accuracy. The prediction classes and Ids can be produced by using the model which can be created by running cnn_ver2.ipynb in our repository.

3. Challenges

Some team members have never implemented the CNN model so familiarizing ourselves with Keras and understanding each step of model creation were the first difficulty we faced. To overcome this challenge, this short yet informative tutorial helped us to implement the preliminary version:
https://www.sitepoint.com/keras-digit-recognition-tutorial/
Additionally, due to the vast number of hyperparameters that have to be tuned in CNNs, it is always a challenge to experiment with these hyperparameters to increase the validation accuracy of the model. Overfitting is also a common problem when fitting neural networks, and therefore, the dropout rate had to be adjusted very carefully to ensure this does not reproduce.

4. Conclusion

Contrary to some of the previous models we've seen in this bootcamp such as Naive Bayes and Random Forest, parameter tuning and having an appropriate number of layers are critical to change the accuracy of the model significantly. There is no one way to tune a neural network and research as well as trial and error are critical. Additionally, data augmentation and preprocessing is incredibly different between images and structured data and a lot was learned on how to augment them.

5. Individual Contribution

- Takuya implemented the preliminary version of the model which initially resulted in around 75% accuracy. He also organized this write up, created the repository and facilitated collaboration.

- Khabiir took responsibility in fine-tuning the model Takuya created by changing parameters, adding data augmentation, etc. The code is stored as cnn_ver1.

- Tyra later created the second cnn model which yielded the best accuracy. She tuned hyperparameters and increased the number of layers, as well as one-hot encoded the classes. Additionally, she used the data augmentation technique Khabiir implemented. The code can be found as cnn_ver2 and this model is the one we used for final submission.