## **Background on Data**

The data provided for the purpose of this assignment came from the Kannada MNIST dataset, provide by Kaggle for competition purposes. It is a dataset comprised of images of number for which the goal is to be able to accurately classify them. This is a large dataset comprised of 784 features to be used, with each feature containing 6,000 instances.

# **Data Preparation**

For this assignment, data preparation was very minimal as this assignment entailed the creation of a neural network for classification purposes. However, with neural networks, it is necessary to create a training and validation set. I used the MNIST training set to create the initial X and Y training sets. The validation sets were then comprised of the first and last 5000 instances of the aforementioned training set.

## **Results/Evaluation**

## **MNIST Neural Networks: Optimizer 1**

Number of	Number of	<u>Total</u>	<u>Optimizer</u>	Training	<u>Validation</u>	Test Accuracy
<u>Hidden</u>	<u>Hidden Nodes</u>	Training		<u>Accuracy</u>	Accuracy	
<u>Layers</u>		<u>Time</u>				
		(epochs = 10)				
1	100	48 s	ReLu	0.9513	0.9290	0.8644
1	25	50.9 s	ReLu	0.9462	0.9240	0.8584
3	100	50.9 s	ReLu	0.9569	0.9300	0.8770
3	25	52.1 s	ReLu	0.9293	0.9134	0.8572

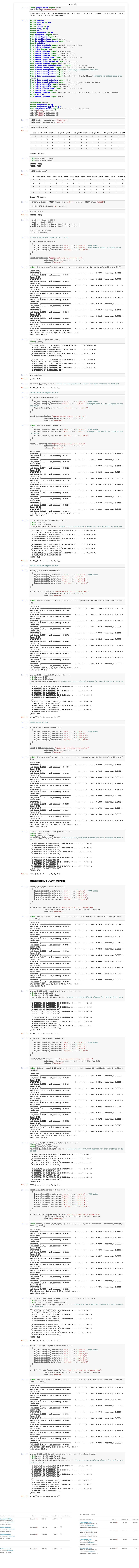
## **MNIST Neural Networks: Optimizer 2**

Number of	Number of	<u>Total</u>	<u>Optimizer</u>	<u>Training</u>	<u>Validation</u>	Test Accuracy
<u>Hidden</u>	Hidden Nodes	Training		<u>Accuracy</u>	Accuracy	
<u>Layers</u>		<u>Time</u>				
		(epochs = 10)				
3	100	1.1 min	RMSprop	0.9936	0.9486	0.9248
3	25	1 min	RMSprop	0.9969	0.9548	0.9354
5	100	1.6 min	RMSprop	0.9908	0.9512	0.9258
5	25	1.6 min	RMSprop	0.9956	0.9586	0.9324

I produced two tables, one of which used the ReLu optimizer, while the other used the RMSprop optimizer. I did this to increase the accuracy of my model while also wanting to compare the differences between the 2 optimizers. I also increased the number of hidden layers in the latter table to see if that also made a difference.

## **Model Performance and Recommendation**

As it pertains to the performance of the model clearly the neural networks built in the second model proved to be more successful as can be seen by test accuracy. Even when comparing the performance of both models, that used 3 hidden layers, the change in optimizer made a significant difference. From a performance perspective, I would highly suggest adjusting the optimizers (including the optimizer parameter of learning rate), number of hidden layers and number nodes per layer. I believe these can all contribute to the accuracy of a model on unseen data.



Use for Final Score