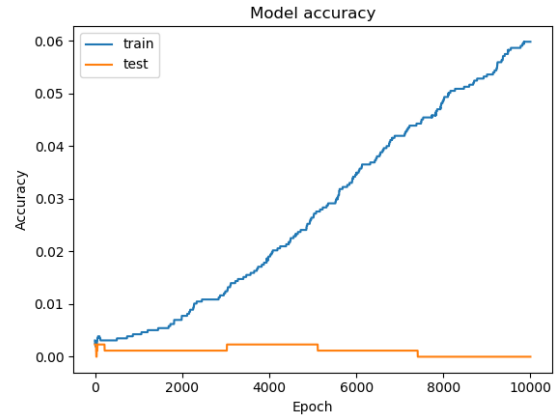
Fig A16

Validation and training accuracy and loss graphs from the second 10k-epoch training session



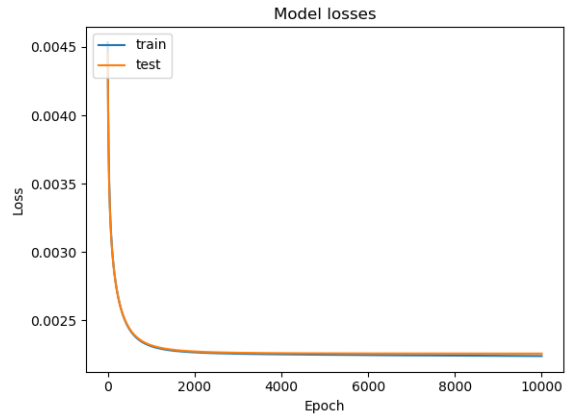


Fig A17

Beginning of the data of NN training accuracy and losses from the second 10k-epoch training session

TRA 10K2, ReLU/Orthogonal	VAL 10K2, ReLU/Orthogonal
0.00310559	0.002328289
0.00310559	0.002328289
0.00310559	0.002328289
0.00310559	0.002328289
0.00310559	0.002328289
0.002717391	0.002328289
0.002717391	0.002328289
0.002329193	0.002328289
0.002329193	0.002328289
0.002329193	0.002328289
0.002717391	0.002328289
0.002329193	0.002328289

Fig	C14
Email to Ma'am Kiel Granada	

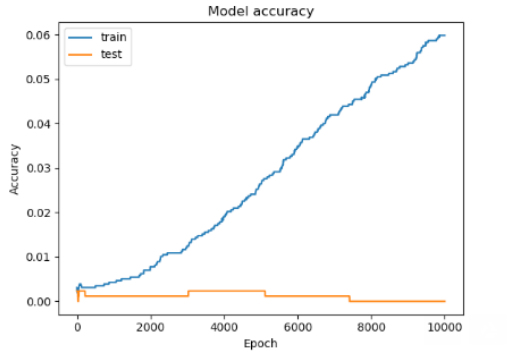
R3-A09 Progress Report (Nov 26)

Joachim Alfonso Navarro 8:19 PM (0 minutes ago)
to Kiel

Good evening ma'am!

We tried training our neural network further (10,000 more epochs), but we realized that we got the same data as last time. The neural network did not save its progress. I looked into the problem and found out that the NN saves when it encounters an increase in *validation* accuracy and not *training* accuracy, the latter of which is increasing.

Below is the accuracy graph from this 10K run. We can, at least, confide in the fact that the neural network's learning is shaping up to be replicable - the NN hit the same 6% peak within 10K epochs as the last run.



Thank you for your time.

Joachim Navarro
R3-A09

Fig	A18
New NN training plan	

R3-A09 Revised Training Plan

Make the checkpointing fix, then do the number of training epochs as outlined in the calculation below. Let $K = 1\,000$ and $M = 1\,000\,000$.
Using a linearly proportional fit we can estimate the number of epochs needed to train the NN to 100% accuracy:

$$\begin{aligned}
\frac{5.97\%}{10K\text{ epochs}} &= \frac{100\%}{t\text{ epochs}} \\
5.97t &= 100(10K)\text{ epochs} \\
5.97t &= 1M\text{ epochs} \\
t &= \frac{1M\text{ epochs}}{5.97} \\
t &\approx \boxed{167\,505\text{ epochs}}
\end{aligned}$$