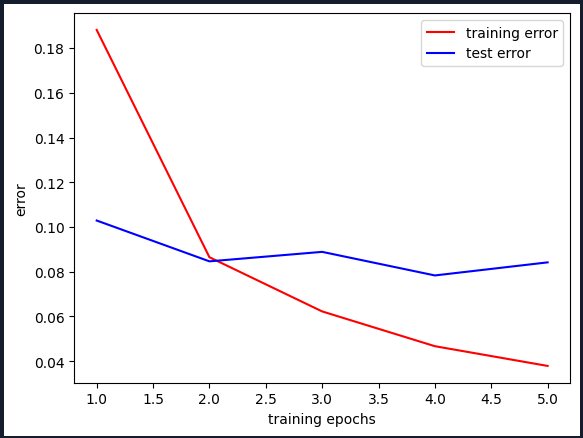
**ReLU with 5 epochs results**:

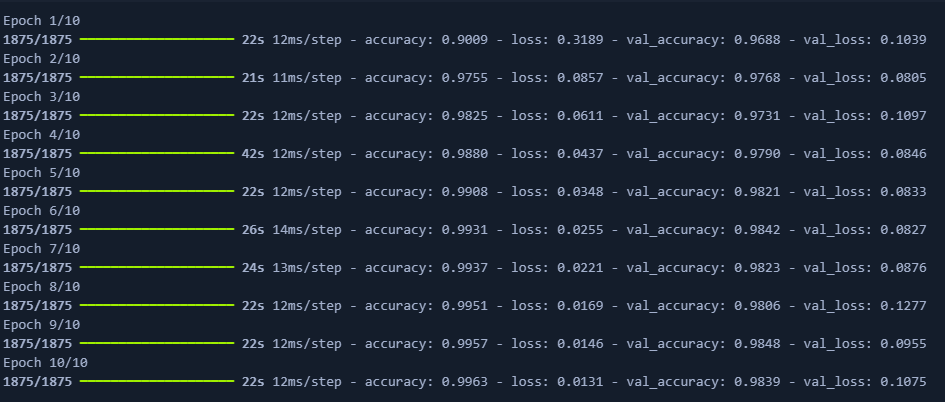


**ReLu with 5 epochs plot:**

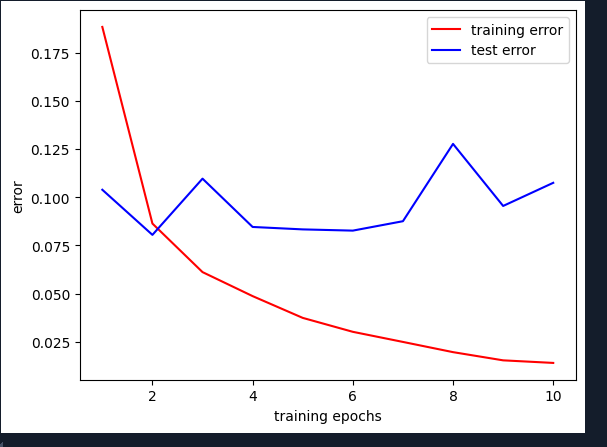


* We can see that the training accuracy hits 98%, and the model did good for 5 epochs.

**ReLU with 10 epoch:**



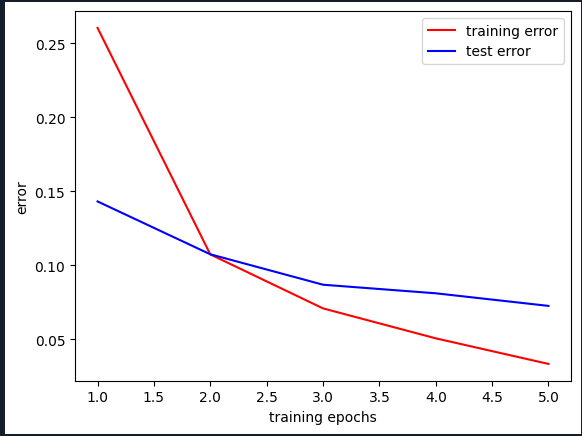
**ReLU with 10 epochs:**



* Although it looks like its performing well but there might be signs of overfitting after epoch, so next I used early stopping criteria to solve the problem. (see the ipynb file)

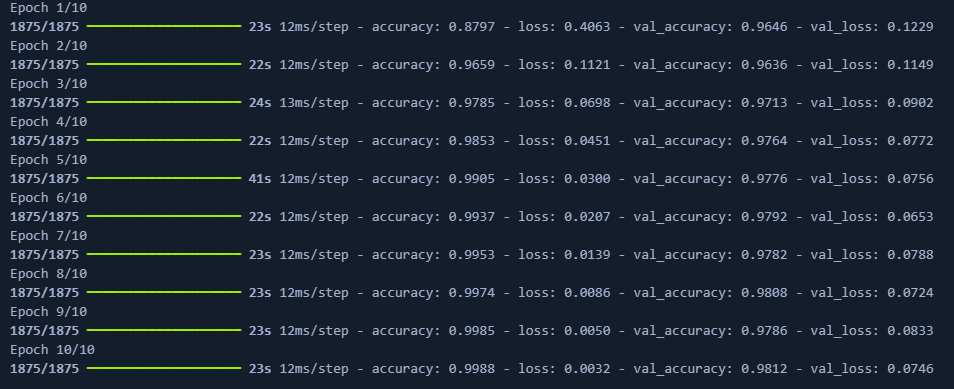
**tanh with 5 epochs:**

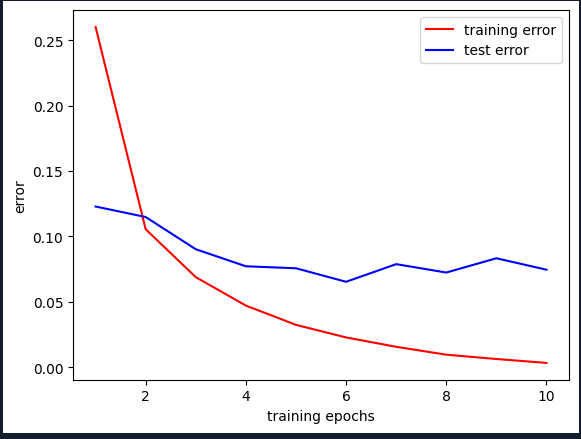
****

****

* Tanh showed a well performance but no the best ( acc = 97%), so I tested it on 10 epochs next.

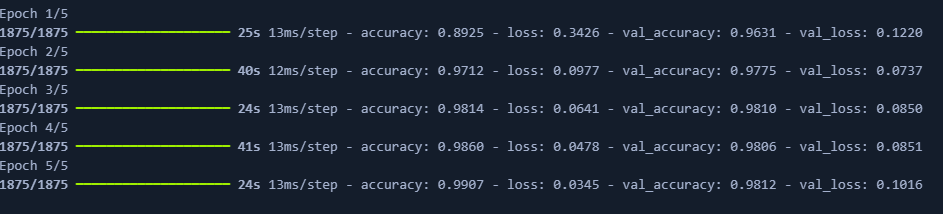
**tanh with 10 epochs:**

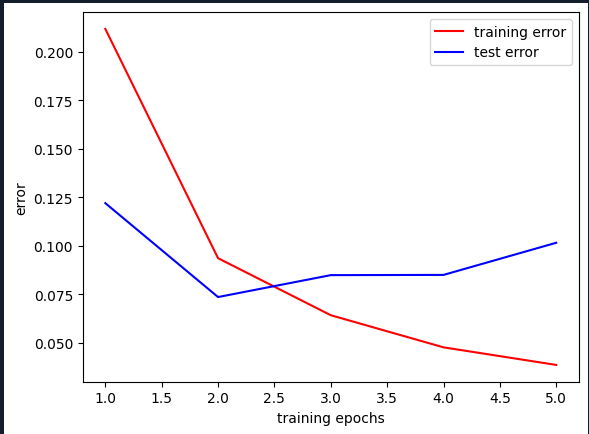




* Although the accuracy increased to 98%, but it may lead to saturation.

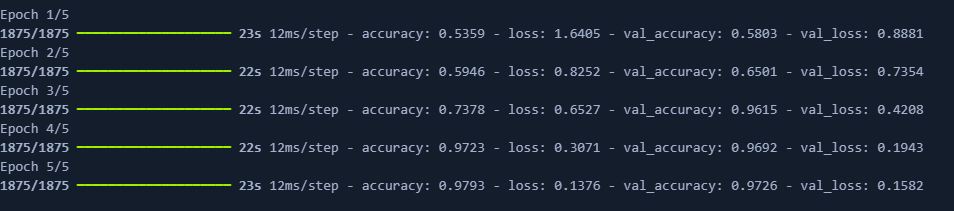
**swish with 5 epochs:**

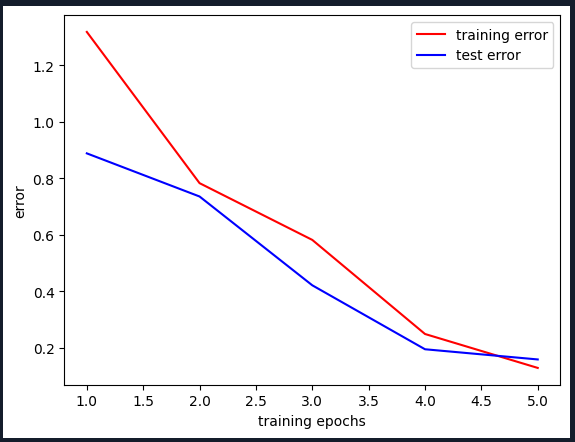
****



* Here loss value decreases at the begging then it starts to increase, which may indicate overfitting.

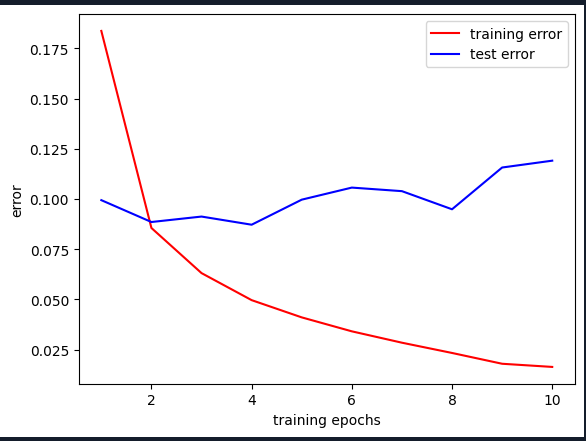
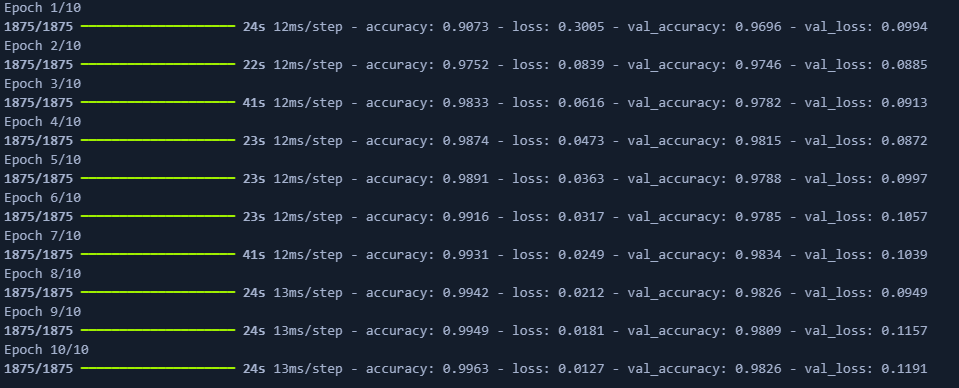
**leakyReLU with 5 epochs:**

****



* The results of the leakyReLU where good, accuracy increased to 97% and loss function decreased in a good way. I decided to try it on 10 epochs to see if it gets better.

**LeakyReLU with 10 epochs:**

****

* for 10 epochs it shows signs for overfitting.