

# Learning journal & AI Usage

## **Learning progress 1:**

Over the first weeks of this project, we decided to go about learning individually. Each member of the group studied basics of neural network by himself. We use class materials, videos and articles to inform ourselves about the basic structure of a neural network and how it works.

This knowledge was deepened in small conversations with our teammates. If we came to disagreements in our understanding of the matter, we used ChatGPT to clean up any confusion and explain what error were made in our thinking.

Through this process we could create a strong fundamental knowledge of what we are going to do.

## **Learning progress 2:**

This was the first time we actually sat together and started working on the project. There were initial hiccups in trying to work with google collab, as it didn't synchronize correctly for us and did not have the option to properly merge our desynced code.

We also tried opening our google collab notebook in a visual studio code environment as well as a pycharm one. Both were unsuccessful. Pycharm seems to be able to do it, but we decided that this was too much of a hassle to get working and we instead opted to work through a github repository, using usual collaboration matters we were already familiar with.

In a next step, we could finally start working on the project. Our first goal was to import the MNIST Dataset and visualize it using matplotlib.

We were unaware that the entire dataset was available through torchvision, and we wasted some time trying to figure out where to get it. We eventually found code snippets online that put us on the right path.

We played around with the library a bit, trying to figure out how we could properly extract data useful to us. After a while we still sat in front of a blank canvas.

We used ChatGPT to avoid wasting hours of time reading library documentation and example code. This gave us a rough outline as to what we would need to do to get the data we were looking for.

After some time we finally accessed the desired data. Now it was time to put the log outputs we had into a first visualisation using matplotlib.

Fortunately, our team member Benaja had some experience with matplotlib, and we did not have to spend too much time visualizing example data as well as the

distribution of labels in both test and train set.

ChatGPT was used to help fix errors during this process, again, to avoid reading too much documentation.

This sets us up with some knowledge over the datasets as well as the usage of torchvision and allows us to properly work on the neural network.

### Learning progress 3:

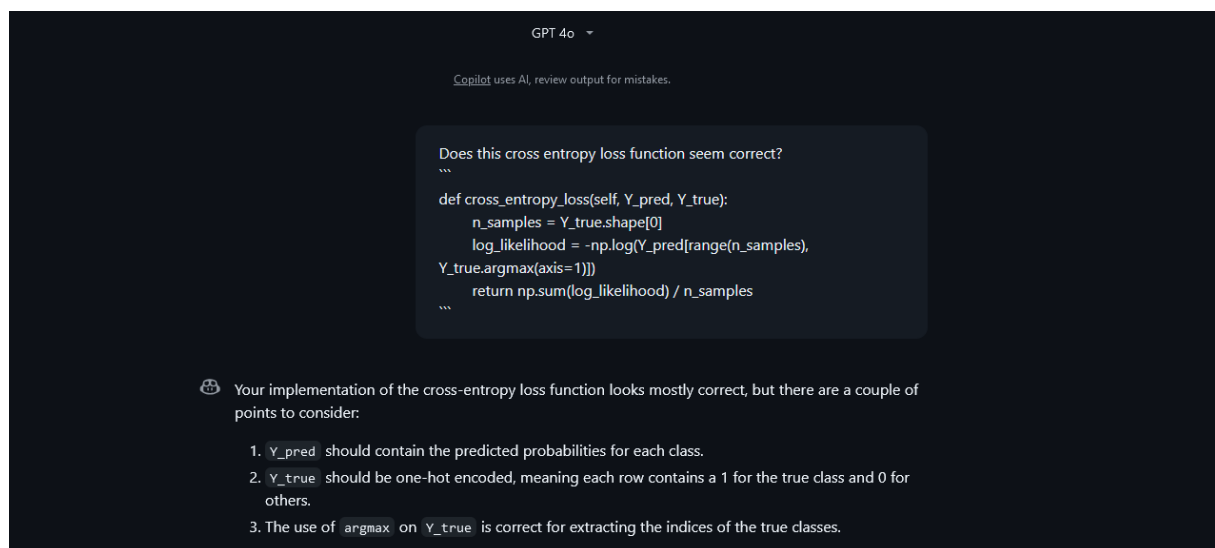
The second time we came together we sat down for a full work day (and a bit more) to really push the projects' progress. We used ChatGPT to build a skeleton of a neural network class that we could fill with our actual code.

There was a lot of trial and error going on. We all mostly sat on one problem until one of us had figured it out and moved on from there.

Even when we had the full implementation, it was far from perfect. There were numerous instances where we got outputs we knew not to be correct.

Some were easy logical error we found ourselves, mostly by checking each others' code. Things like X and Y values being switched, train and test set, etc.

Some others were more complicated, and required us to check with outside help, mainly ChatGPT. We use dit to find isolated errors in our code and explain what we were doing wrong.



After a lot of work we ended up with a model that seemed to still be incorrect. We found our model to have 100% accuracy at times, which we believed to be wrong. After a while, we agreed to let it go for now and consult Mr. Hackstein about it.

### **Learning Progress 4:**

We successfully consulted Mr. Hackstein and our implementation seems to be correct. He told us that, with a small task like this one, a model can easily be overfitted in a single epoch, and that that is what is happening to us.

This is very good news to us, as that means that we are done with the bulk of the work. The next steps consist of cleaning up, documenting, and making sure our project fits the criteria.

For this, we split our remaining work between our members for them to work on it individually. This is very handy as we are all short on time up until the hand-in of the project.

### **Learning progress 5:**

All group members successfully finished their assigned tasks. Those tasks included documentation, making the workbook actually pleasing to look at, clean up old code, describe the neural network in text, correct the matplotlib visualisation, create a unit test with manual calculations to prove our models' correctness.

Those tasks took up more time than we had initially hoped or liked, but ended up going down surprisingly smoothly (unlike the neural network) and without much help.

## **ChatGPT Usage**

*What is referred to here as «ChatGPT» may also be Copilot or in some cases Claude.*

ChatGPT was used as described in the journal for the following tasks:

- Verify knowledge and aid in the learning process of abstract concepts
- Provide example code on how to use specific libraries or how to extract desired values as a shortcut to extensively reading documentation
- Build skeletons of classes to provide a structure we can work with
- Verify the correctness of written code
- Find and fix errors, both of technical and logical nature
- Give a rough outline and inspiration for texts and explanations to guide and support in the creative progress of writing easy to read and conclusive texts
- Format markdown professionally