We started by plotting a couple of histograms of some of our features within X which include several 100 cell types. The following two charts show the data variability of two of the cells. We used this information and decided to normalize our training outputs between -1 and 1. We also applied a one hot encoder to the cell type. We implemented a NN with one hidden layer so we can test the effectiveness of our feature encoding as well as our feature selection. There is a meta data file that includes information about each test given in the training data that could be used for modifying our features to specific domain characteristics. Included below is a histogram that shows the availability of our training data. We have all of the all for NK cells, T cells and Regulatory cells and we are trying to predict, B cells and Mylnoid cells. We think that the data distribution will impact what model we end up choosing to implement this with. A neural network might perform better because the data is not necessarily reliant on the calculated value of the training data. We also have to be careful with overfitting as we could definitely create a model that does not necessarily generalize well to the increase in data for other cell types. Our plan for the rest of the semester is to migrate this code to Google Colab so we can run more than one hidden layer on the neural network. Depending on how the NN responds, we might look into alternative NN strategies that can optimize the performance of our model. We also want to look more into feature engineering and add some content to the Jupyter notebook so users who view our model can understand design decisions. A blue graph with black lines

Description automatically generatedA graph of cell type histogram

Description automatically generatedA blue graph with black text

Description automatically generated