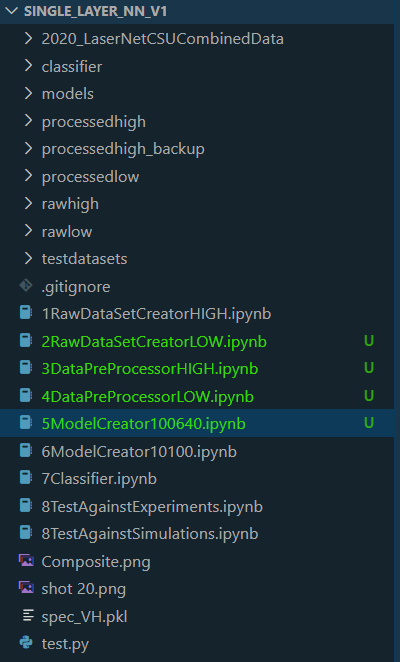
How to use the Single-layer NN prediction model.

**Downloading the Folder**

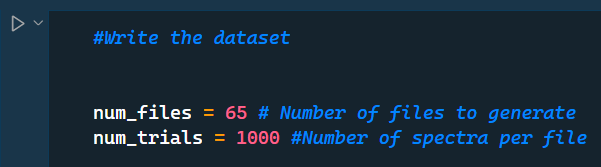
1. Download the ‘Single\_Layer\_NN\_V1’ folder from the Hussein Lab/Experimental Data/LaserNet and External Experiments/2020 LaserNet CSU Combined Data/NN Model/Single\_Layer\_NN\_V1.zip
2. Extract the zip file and open the Single\_Layer\_NN\_V1 folder in VSCode. If done correctly, file structure should look like this:



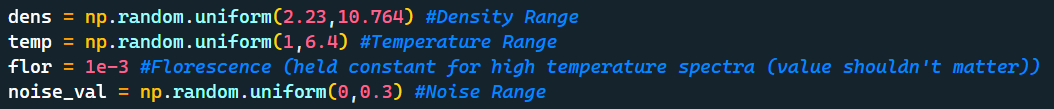
i) Note, this folder exists on the lab computer at C:\Users\HusseinLab\Documents\Single\_Layer\_NN\_V1

**Generating Dataset**

1. Datasets Generation is split into 2 Jupyter Notebooks ‘1RawDataSetCreatorHigh.ipynb’ and ‘2RawDataSetCreatorLow.ipynb’ these are meant to generate the datasets for the 1-6.4keV range and the 0.1-1keV range respectively. The difference between each the two notebooks is that ‘1RawDataSetCreatorHigh’ only saves density and temperature values for each spectra generated, where along with the density and temperature values, whereas ‘2RawDataSetCreatorHigh’ saves fluorescence values as well.
2. To create a new dataset, open the notebook for the temperature range a new dataset is needed for, and run each cell sequentially (ensure an Anaconda environment is used for all notebooks).
3. Once cell 6 is reached use the ‘num\_files’ and ‘num\_trials’ variables to control the size of the dataset generated.



The total number of spectra in the dataset is equal to num\_files \* num\_trials. The reason for saving the dataset across multiple different files is for easier control of separating the training dataset from the testing dataset, as well as to avoid creating large Arrays and/or DataFrames which use excessive amounts of RAM and slow down the dataset writing process.



Above are the input parameters for the high temperature dataset creator in cell 6.

For finer control of the dataset parameters, adjust these numbers. For the rationality of the values used in the example, see note 1.



1. Once this cell is run, it will create 2 \* num\_files number of DataFrames and export them to the rawhigh/rawlow folders in .pkl format.

The DataFrames are generated and saved as pairs. EID\_n is setup such that each row in the DataFrame contains the 2995 intensity values created by the simulate SCRAM code for each spectra, and DTF\_n is setup so that each row contains the density, temperature, and fluorescence (if applicable) values which were used to generate the Spectra in the corresponding row of the EID\_n DataFrame. When the dataset is created, the file structure should look like this ————————————>

1. The next cells in the dataset creator note are a check to ensure that the Spectra were generated correctly. It pulls the test\_VH+noise\_VH values from the last spectra generated, and compares that to a new spectra generated with the density, temperature, and fluorescence values which were saved when it was generated. If the dataset creation process worked properly, the spectra should line up with each other.

**Dataset Processing**

**Notes**

Note 1:

i) Density was chosen to range from 2.23 - 10.764g/cc, as that is the range from

[the minimum density value the SCRAM tables allow for]

to

[120% the density of solid copper]

These choices were kept the same for the low and high temperature dataset creators. As density as temperature shouldn’t be correlated with each other. 120% the density of solid copper was chosen, as densities above that should not appear in experimental data. Making them largely extraneous.

ii) Temperature range for the high temperature model is from 1 - 6.4keV, as that is the range from

[the minimum temperature where fluorescence doesn’t influence the spectra drastically]

to

[maximum temperature SCRAM tables allow]

iii) Temperature range for the low temperature model is from 0.1 - 1keV, as that is the range from

[Minimum temperature SCRAM tables allow]

to

[the maximum temperature where fluorescence influences the spectra drastically]

Subnote: The simulate\_scram code has a hard cutoff on accounting for fluorescence emission at 1keV, beyond that temperature, the flor value is ignored by the code, and the spectra is generated using self emission only. The 1keV point was chosen somewhat arbitrarily, and could be modified in the future for further testing. Theoretically, the cutoff could be set at 6.4keV, where fluorescence would still influence the spectra, but only slightly, negatively impacting the models performance at guessing the fluorescence parameter, but allowing for one model to be used across the whole temperature range.

iv) Fluorescence range for the high temperature model is constant at 1e-3, in theory, this value could be set to anything without impacting the spectra.

v) Fluorescence range for the low temperature model is from 1e-2 to 1e-4, as this is a safe range which stops the code from generating spectras with intensities so low the model will have a difficult time picking up on patterns, or intensities so high that they are unrealistic for the temperature range.

vi) noise\_val controls the intensity of the noise present in the spectra. Noise of 0 was chosen as the minimum as that represents a perfectly clean spectra, and 0.3 was chosen as the maximum as that is about the maximum of noise level present in most of the experimental spectra. This was done using the eye test and could be pinned down better in the future.