There are 3 main functions of the code:

**The dataset:**

Is in all\_data.csv

**Generating training/test splits:**

First, make a csv file in Data/Multitask\_data/All\_datasets/crossval\_split\_specs. It will look something like this:

|  |  |  |  |
| --- | --- | --- | --- |
| Data\_types\_for\_component | Values | Data\_type\_for\_split | Train\_or\_split |
| Experiment\_ID | Whitehead\_siRNA | Amine | split |
| Experiment\_ID | LM\_3CR | Amine | split |
| Library\_ID,Delivery\_target | RM\_carbonate, generic\_cell | Amine | split |
| Experiment\_ID,Library\_ID | A549\_form\_screen,IR\_Reductive\_amination | Amine | split |
| Experiment\_ID,Library\_ID | A549\_form\_screen,other | smiles | train |
| Experiment\_ID | Luke\_Raj\_133\_3\_form | Random\_id | train |
| Library\_ID | RM\_Michael\_addition\_branched | Amine | split |
| Experiment\_ID | IR\_BL\_AG\_4CR | Amine | split |
| Experiment\_ID | IR\_BL\_AG\_3CR\_sting | Amine | split |
| Experiment\_ID | Akinc\_Michael\_addition | Amine | split |
| Experiment\_ID | Li\_Thiolyne | Amine | split |
| Experiment\_ID | Liu\_Phospholipids | Amine | split |
| Experiment\_ID | Miller\_zwitter\_lipids | Amine | split |
| Experiment\_ID | Zhou\_dendrimer | Amine | split |
| Experiment\_ID | Lee\_unsat\_dendrimer | Amine | split |
| Library\_ID | 4CR\_Ketone | smiles | train |

This is a way to do specify the training-test split. The columns hopefully make sense but one key column is “train\_or\_split”. If it’s “split”, then the data will be split into training and validation. If it’s “train”, then the data will all be put in the training set.

To do a split, run the command:

python3 Merge\_datasets\_v2.py specified\_cv\_split (split filename) ultra-held-out-fraction (either nothing or “morgan”)

The “ultra-held-out-fraction” is the fraction of data that you want held out of all of the cross-validation sets. This is useful for reporting error with predictions based on an ensemble average of all the cross-validation splits.

If you add “morgan” then binary Morgan fingerprints (radius 2, 2048 bits) will be included.

The output will be 5 folders cv\_0, …, cv\_4. Also an “ultra\_held\_out” folder if that option was specified.

**Training a model:**

To run a model, run the command:

python3 Merge\_datasets\_v2.py train\_optimized\_from\_to\_cv\_already\_split (location of optimized hyperparameter configs) (location of data for training)

The location of the optimized hyperparameter configs is the name of a folder in Data/Multitask\_data/All\_datasets/Splits/folder. Inside that folder there should be a path hyperopt/optimized\_configs.json. These hyperparameters will be used for model training.

The location of data for training is the name of a folder in Data/Multitask\_data/All\_datasets/Splits/crossval\_splits.

*If the folder name ends in Morgan*: the model will include binary Morgan fingerprints, radius 2, 2048 bits.

**Testing model performance:**

Run the command:

python3 Merge\_datasets\_v2.py analyze\_cv

Analyze\_cv

**Making predictions with the model:**

**Analyzing predictions of the model:**