To run the code, please follow the following steps:

1. Choose the dataset that you are going to use it.
2. Use the appropriate python file to read dataset
3. Read dataset
   1. If you are using Davis or KIBA datasets use read\_dataset.py and set the dataset path and dataset name properly and then run the code. The output of the code is three matrix include XD, XT and y. XD is a N\*100 matrix where N is the number protein-ligand pairs and 100 is the maximum length of the ligand SMILES sequence. XT is a N\*100 matrix where N is the number protein-ligand pairs and 1000 is the maximum length of the protein sequence. Also, y is a N\*1 vector which shows the corresponding affinity value for each protein-ligand pair.
   2. If you are using PDBbind dataset use read\_pdb\_dataset.py and set the dataset path (path\_pdb) properly and then run the code. It should be noted that Y\_PDB.txt and name\_PDB.txt should be placed in the current directory. The output of the code is three matrix include XD, XT and y. XD is a N\*100 matrix where N is the number protein-ligand pairs and 100 is the maximum length of the ligand SMILES sequence. XT is a N\*100 matrix where N is the number protein-ligand pairs and 1000 is the maximum length of the protein sequence. Also, y is a N\*1 vector which shows the corresponding affinity value for each protein-ligand pair.
4. Split the dataset into train and test set and then create train\_drugs, train\_prots, and train\_Y, val\_drugs, val\_prots, and val\_Y = davis\_data. Then, store them in mat file.

In a simple scenario, you can utilize the following codes (in the paper, we have utilized k-fold cross validation):

from sklearn.model\_selection import train\_test\_split

train, test, train\_Y, val\_Y = train\_test\_split(np.hstack((XD, XT)), y, test\_size=0.3, random\_state=42)

train\_prots = train[:, 100:]

val\_prots = test[:, 100:]

train\_drugs = train[:, 0:100]

val\_drugs = test[:, 0:100]

1. Set the data\_path (path of mat file that you have stored in the previous step) properly and run sgcn.py. It should be noted that there are some hyper parameters (see config.py) in this python file that you can set them properly based on the paper. All of hyperparamter values have set in config.py