CGCNN Procedure

1. Extract the feature vectors for nodes and edges.
   NODES-number of connections of a node: found by calculating the number of times a node is listed in the bar connectivity columns
   Bar connectivity for the first entry is -
   Node: connectivity(number of times it appears)
   1: 3
   2: 3
   3: 3
   4: 3
   5: 3
   6: 3
   7: 3
   8: 3 all the nodes from 1 to 8 appear 3 times

   EDGE- two features need to be extracted for edges
   Naming of edges- in the columns of bar connectivity named as bar_connectivity_i_1 and bar_connectivity_i_2 i represents the edge number and 1 or 2 represents which node it connects
Eg. In the first entry the column bar_connectivity_12_1 signifies edge number 12 and 1st node of the edge is 7
column bar_connectivity_12_2 signifies edge number 12 and 2nd node of the edge is 8

- length of edge: obtained via nodal positions (distance formula)
  To calculate the length of the edge apply distance formula between the two nodes of the edge
  Eg. edge 12 find distance between node 7 and node 8
  Nodal coordinates are given as nodal_positions_i_j
  i represents the node number
  j =1 represents x coordinate value, j =2 represents x coordinate value
  j =3 represents x coordinate value
  Eg. Nodal_positions_7_2 represents the y coordinate value of node 7
  To calculate distance between node 7 and node 8: use formula

  $$\sqrt{(x_8 - x_7)^2 + (y_8 - y_7)^2 + (z_8 - z_7)^2} = D$$

  x_8 is given by Nodal_positions_8_1
  y_8 is given by Nodal_positions_8_2
  z_8 is given by Nodal_positions_8_3 and so on the other values

- angle each edge makes with the x,y, and z-axis: direction cosines formula implemented on the nodal points (angle with x axis= x2-x1/ distance between the points)

$$l = \frac{x_8 - x_7}{D}, \; m = \frac{y_8 - y_7}{D}, \; n = \frac{z_8 - z_7}{D}$$

4 features of each edge would be extracted - length of edge, l, m, n

Contruct a CGCNN model with the above given information.

Form a graph using node and edge parameters as given above by applying CNN with R1 layers and further create the entire CGCNN model with L1,L2 Hidden layers and pooling layers as mentioned on the page 1,2,3 of the pdf named “cgcnn” attached. Do hyperparameter tuning in the hidden layer changing various functions so as to increase accuracy of model atleast above 90 percent.

2. The feature vectors extracted above i.e. node connectivity, edge length, l, m ,n and a,b,c,alpha,beta,gamma from the spreadsheet are the input feature vector for the neural network. In the dataset 2000 points data is given. Out of 2000 datapoints, 1600 points are used for training, 200 for validation and 200 for testing.
3. Y(values to be predicted via neural network) vector is Cx, Cy, Cz and nx, ny, nz. [given in spreadsheet]

Final output required-

1. Accuracies/errors of the model in predicting the given properties.
2. Function that takes input of nodal positions, bar connectivity, a,b,c,alpha,beta,gamma as input and predits the properties Cx, Cy, Cz and nx, ny, nz with the accuracy more than 90 percent.
3. Plot of Y predicted vs Y true
4. Error plot
5. Summary/ documentation of the model