Problem Formulation

Th structure of chemical compounds can offer insight into not only the nature of the molecule, but also its properties and relationships internally and within molecules. Studying such compounds could benefit from the application of graph neural networks (GNN). This is a powerful tool for modeling and capturing dependencies and relationships between compounds. "conventional" data more resembles a feature-value vector format, whereas graphs exhibit a node-edge format with no vector representation. This creates a challenge for which GNNs were developed where given a set of training graphs associated with a label, the network will learn a model from these training graphs and tries to predict the unseen graph.

In this project, the goal is to detect anticancer activity using GNNs where each chemical compound has a graph representation (atoms being nodes and bonds as edges), a compound is considered positive if it is active against cancer and negative otherwise. The GNNs employs graphs convolutions (similar to matrix convolutions in the image classification assignment), and similarly the GNN propagates the information of a node in a graph to its neighboring nodes (just like propagating information of a pixel to its neighboring pixel). The input variables for this problem are nodes and edges (atoms and bonds), the process involves first a transition function that takes as input the features of each node, the edge features of each node, the neighboring nodes' state, and the neighboring nodes' features and outputting the nodes' new state. One of the defining features of a GNN is that it uses a form of neural message passing in which vector messages are exchanged between nodes in the graph and updated using neural networks. The output of this problem is detection of chemical compounds or drugs that are positive towards cancer cells showing their effectivity. Data mining functions requires will include the Graph Neural Networks library to be able to use GNN layers. Some of the challenges include dealing with imbalanced data which would require functions needed to balance the data, and also managing various message passing mechanism within the network.

Graph Convolutional Network Aggregation Mechanisms

GNNs are composed of message passing mechanisms which are basically nodes updating their states and exchanging information by passing "messages" to their neighboring nodes until they reach a stable equilibrium. First, I ran the model with the parameters set in the template and achieved accuracies of around 0.63-0.65. Then I added the aggregate_function and set it to mean, the input to the aggregate_function is a set of embedding of the nodes in graph neighborhood and generates a message based on this aggregated neighborhood information. The default for this parameter is set to sum which gave best performance out the rest (mean, max, sqrt_n). Then I decided to play around with the message_activation_function which helps the network learn complex patterns in the data, first decided to use than and Relu because these are pointwise operations, in the GNN they can be computed at each node individually, without looking at the rest of the graph. Relu which directly outputs the input if poaitive and zero otherwise, seemed to have performed better and it makes sense since that is nature of our data. However, one of the major factors that improved model performance was increasing the number of GNN layers "num_layers" to 9, this means in the training the algorithm looks at 9 neighboring points to learn.