Journal Report 4
9/23/19-9/29/19
Emily Ye
Computer Systems Research Lab
Period 2, White

Daily Log

Monday September 23

Ran Spektral example code for a GCN in graph batch mode and got test losses of about 0.69 (training on sample size of 1000) and about 0.90 (training on sample size of 2000).

Tuesday September 24

Read Spektral documentation for single mode, batch mode, graph batch mode, and their differences.

Went through graph batch GCN code and marked down lines that I did not fully understand. Reviewed TensorFlow and Spektral documentation for all of the parts that I marked.

Compared graph batch GCN code with edge-conditioned GCN code and noted down differences between them.

Thursday September 26

Wrote method to calculate R^2 value for accuracy graph of edge-conditioned GCN to use for comparison with accuracy of graph batch GCN.

Began working on accuracy graph to show difference between predicted values and actual values (like the one made for the edge-conditioned GCN) for graph batch GCN.

Timeline

Date	Goal	Met
September 16	Create a basic GCN using Spektral	Began creating GCN using Spektral with edge-conditioned convolutions
September 23	Complete GCN with edge-conditioned convolutions	Completed edge-conditioned GCN with test loss of about 0.03
September 30	Complete GCN in graph batch mode for comparison with edge-conditioned GCN	Completed basic graph batch GCN, but did not finish coding accuracy graph for comparison with edge-conditioned GCN
October 7	Code accuracy graph (predictions v. true values) for graph batch GCN	
October 14	Research and begin implementing other kinds of GCNs (such as adaptive graph convolutional network presented by Li, et al. [2018])	

Reflection

This week, my main goal was to implement a GCN in graph batch mode based on Spektral example code. Since Spektral provided example code for both an edge-conditioned GCN and a GCN in graph batch mode, and I was not sure which one would be better for our final project, I tried to implement both and compare them to each other. Originally, because I had not read much about Spektral's graph batch mode, I naively planned to just compare their losses. However, after running the graph batch GCN, I noticed that the initial loss was far higher than the initial loss for the edge-conditioned GCN, so using the loss as a way to compare both types of GCNs would not be logical.

I spent most of Tuesday reading more about Spektral's graph batch mode and decided that the easiest way to compare GCNs would be to create accuracy graphs for each kind, plotting predicted values against actual values like the graphs attached in my last journal report. Thus, I coded a method to find the R^2 value for the edge-conditioned GCN's accuracy graph. Unfortunately, since the graph batch GCN has a completely different structure from the edge-conditioned GCN, I could not reuse the code I had already written to plot a graph and calculate the R^2 value, so that is my goal for next week.

Furthermore, I found some other kinds of GCNs while doing research this week. Originally, I had planned to just implement two kinds and compare them, but since I now have more options, I hope to implement some other kinds of GCNs to use for comparison with the graph batch and edge-conditioned GCNs that I already have. Being able to plot graphs and calculate R^2 values will be useful for these new GCNs as well. Thus, I have amended my timeline so that I begin researching and implementing other GCNs after finishing the accuracy graph for the graph batch GCN.