Steps to create method 4:

**Java Steps:**

Compile and Run “ParseConnData\_FollowHopsV3.java”

* “Edge.java”, “Vertice.java”, and “GraphCycle.java” are used by this program so have them in the same folder/project when compiling.
* Make sure full file is in the DATA\_FOLDER path and is named “full\_dataset.csv”
* If you are running windows, make sure you add the “windows” param when running.

When processing is complete:

* Output from running will be placed in the OUTPUT\_FOLDER path and will be named “FinalGraph\_V4.csv”
* Copy this output into the Data folder as it will be used in the next step

Compile and Run “ParseConnData\_Create\_ANN\_DataV2.java”

* Converts the “FinalGraph\_V4.csv” into:
  + Output\_NN\_In\_V7.csv (the input X matrix for python)
  + Output\_OneHot\_V7.csv (the input y matrix for python)

**Python Steps:**

Convert the csv files containing the matrix data into a numpy matrix.

Using the code in “Code\_py/V7” folder:

* Open the “NN\_SaveArraysV7.py” file.
* You will notice at the top “X = np.array([[“ followed by data. This data is directly from the “Output\_NN\_In\_V7.csv”
* When you want to use new data, the data should be replaced.
* You will surround the contents of the csv file (removing the header line) with:
  + Before the data: “X = np.array(“
  + After the data: “)”
* You will also notice after this area another bit of data before the “OneHot” data that begins with “y = np.array(“ followed by the “Output\_OneHot\_V7.csv” data.
  + Replace everything between the “y = np.array(“ and “)” which is near the bottom.
* Everything else should remain the same, except the output path in the bottom two lines.
  + The bottom two lines indicate where the output numpy matrix data should be saved.
* After updating, run the “NN\_SaveArraysV7.py” file and two files should be created:
  + x\_V7.npy and y\_V7.npy
  + These will be used in the next step
* Run “NN\_backProp\_opt3\_V7.py” to teach the ANN.
* Output will go to the screen. Capture it if you wish to used the output to create any of the graphs in the paper. Typically I only needed the first 5-10 lines of data.

**Next steps:**

1. After obtaining the same results contained in the paper, add lines of code to the ANN logic to store the ANN’s learned state (saving the weights and bias values).
2. Obtain new data and process it as far as the “NN\_SaveArraysV7.py” step and use the output of that step to confirm that the ANN has learned correctly.
3. Will need to clone and update the NN\_backProp\_opt3\_V7.py” to NOT learn but instead load the results saved in #1 above and then run the new data through to see if it has been taught correctly.
4. I would continue to read and apply chapters 10+ from the textbook that I was using for the basis of the code as that was where I left off (teaching the ANN).

Hope this helps.