

Graduate students and postdocs working in our group are expected to be—or become—familiar with the following computing systems:

Low expertise required:

[MATLAB](#) – general platform with which you can get as simple or as complicated as you like; can use for data cleaning, data analysis, plot generation, etc.

[Complex Networks toolbox](#) – **MATLAB** toolbox for complex network analysis by Lev Muchnik; contains many useful features such as k-shell decomposition

[Machine learning toolbox](#) – **MATLAB** toolbox for basic machine learning

[Community detection/modularity algorithm](#) – **MATLAB** code for finding community structure and modularity of a network

[Network components algorithm](#) – **MATLAB** code for finding components of a network, their sizes, and their member nodes

[Python](#) – general platform; again, uses can range in complexity

[NetworkX](#) – **Python** library for finding basic attributes of a network such as the degree distribution

[graph-tool](#) – **Python** library for fast component decomposition, finding modularity, large network visualization

[pandas](#) – **Python** library for data management

[NumPy](#) – **Python** library for vector and matrix operations

[SciPy](#) – **Python** library for statistics, hypothesis testing, regression, and numerical computation

[Beautiful Soup](#) – **Python** library for scraping data from websites

[Scikit-learn](#) – **Python** library for basic machine learning methods, including GLasso and stochastic gradient descent

[ImageJ](#) – **Java** image processing program used for optical CT imaging analysis

[Gephi](#) – visualization and analysis software for networks (**can be buggy and/or freeze—save work often!**)

[Pajek](#) – general network visualization software

Low/medium expertise required:

[SQLite](#) – database management system for Twitter data management and analysis

Medium expertise required:

[Graphical Lasso \(GLasso\)](#) – **MATLAB** implementation of algorithm for finding sparse inverse correlation or covariance matrix

[Collective Influence \(CI\) code](#) – **C language** implementation of algorithm to find most influential nodes in a network

[Monte Carlo for Maximum Entropy XY model](#) – **C language** method to find interaction matrix and some thermodynamical properties of a network which can be modelled via a Maximum Entropy XY model (**best for very small networks!**)

[FMRIB Software Library \(FSL\)](#) – software for model-based FMRI analysis (FEAT) and modelling the brain (BET)

[BrainNet Viewer](#) – software for brain network visualization

Medium/high expertise required:

[Medical imaging toolbox](#) – **MATLAB** toolbox specifically for medical imaging

[Natural Language Toolkit](#) – **Python**-based platform for building code used in natural language processing (e.g., on Twitter)

High expertise required:

[TensorFlow](#) – software for Deep Learning development in machine learning

For computer analysis you will need:

[Anaconda for Python 3.6](#)

[Gephi](#)

[A Twitter account](#)