Data Science Engineering Methods and Tools – INFO 6105

You will learn the math foundations and basic tools of Data Science while mastering Python. You will learn how to use multi-dimensional arrays and think in vectors. You will learn how to operate on tables, time series, and manipulate large spreadsheets. You will learn basic theories in probability, Bayesian statistics, and linear algebra by leveraging the 4 basic Data Science libraries written for Python: NumPy, Pandas, SciPy, and Scikit-learn. This class gives you the fundamental knowledge to be able to advance to advanced topics in Machine Learning (ML), and for applying for jobs that involve data analysis, such as jobs in the life sciences, financial, advertising, and social Web industries.

Numpy adds Python support for large multi-dimensional arrays and matrices, along with a library of high-level mathematical functions to operate on these arrays. It focuses on fast number calculations, reads in fixed datatypes, improves RAM efficiency, and teaches you to think in Vectors. Pandas adds support for more refined data manipulation and analysis. It adds support for data structures and operations for manipulating tables and time series. SciPy is a collection of classic math and science algorithms and helper functions built on top of Numpy, such as linear and nonlinear regression, numerical optimization, etc. If you know the rules for dealing with your data, SciPy is the library for you. If you want the computer to learn the rules instead, and give you probabilistic answers, then Scikit-learn, built on top of SciPy, is what you need. It is a Python module for machine learning at a basic level. If you can solve a problem with the methods in SciPy, it's more straightforward. If you can't, there's a good chance your problem is solvable using methods from Scikit-learn. We finish with an introduction to famous Machine Learning frameworks like TensorFlow, Torch, and Keras, and use cases like Natural Language Understanding and Vision.

The basic languages of Data Science are R and Python. We will start with a one-class introduction of R to get used to manipulating spreadsheets instead of single-entity variables, and then we switch to Python for the rest of the semester. A programming background in one managed language (C#, Java, or Python) is required, otherwise this class will be extremely hard for non-programmers. Your knowledge of Python will improve to black-belt level.

Grade is based on homework (30%), 3 exams spread out across the semester (10% each), a final project (30%), and class participation (10%).