Brian Hogan, Master's in Applied Data Science
Teaching Philosophy – 2020

Prior to training as a scientist I worked as a process engineering professional using discrete event modeling and simulation technology to design, augment, and optimize pharmaceutical development, manufacturing, packaging, and resource requirements from medical doctors to floor technicians. Work as a symbolic analyst focused on training leaders and managers in statistics and supervised methods. Consulting challenging me to quickly assess a business owner's experience, training, and build manager confidence in deliverables. I found great satisfaction in making connections and expanded this skill by running training courses teaching scientists, such as NASA engineers, simulation programming. I thrived on a constant idea flow and overcoming challenges to ensure a successful outcome. I knew I wanted to teach in academic and choose data science to augment my consulting technical skills with solid academic training for work post-secondary education. Curiously, of my professors, Dr. Ami Gates @ Georgetown, commented "the world needs programmers" and I knew teaching would be the best avenue for me to generate societal equity by helping thriving mindsets with programming techniques and statistical reasoning skills.

Interestingly before I started my master's in 2018, I was advising individuals and small groups of graduate students in academic research, writing, statistics, and programming. During my master's in sociology at Boston College I became very committed to academic writing and began advising peer students. This experience taught me how to be a creative advisor who is able to connect to others, find the right message, and encourage others to advance work.

An important component of an instructor's role is to foster student motivation, strategize challenging assignments, and build confidence so students trust their abilities. When I tutor students in programming I begin with an overview of systems theory and how to abstract a set of interrelated components defining a structure or how something operates. I find when students learn to abstract by drawing a box around an area of interest they can readily assess system relationships inputs, outputs, resources, and constraints. Abstraction also helps a student learn a languages API, input and output storage containers, and how to classify objects, such as Python's lists, dictionaries, and strings. My goal is help students simplify large abstract universes into manageable chunks practice coding exercises such as reading, cleaning and looping, and learn how use program library resources to build environments.

Recognizing people work, study, and grok at different rates has led me to strategize exercises and anecdotes to help students see the benefits of their endeavors and fill their *idea hopper*. To ensure students feel supported, outside the classroom I am available to discuss challenges critical to learning. Sometimes a quick response or short discussion maintains motivation to solve challenges and discover new approaches. In difficult assignments I stress the importance of documenting thinking approaches for questions not completed to show class commitment when everything else required to be human disrupts focus. Most times, I find students are motivated to catch up and benefit from course material as often work performed builds a gateway to new occupations and increases pillars of self-worth.

I am a firm believer in having high quality, researched, and tested materials to advance course objectives. While considered a standard for any accredited program, I highlight to stress to academic search committees my commitment to a student's education experience with an eye towards ensuring retention rate reduction. If materials need improvement, additional reference materials, or helpful examples I ensure they are generated. I was fortunate to have many amazing professors in my

education experience and I endeavor to match the quality of their lecture delivery styles. Teaching builds various mental pillars encouraging students to grow and consider new means to approach challenges.

One material set includes syllabus, reading materials, and asynchronous video lecture with supporting Microsoft PowerPoint training slides. When materials are carefully staged, students are ensure weekly hours required to advance studies. A second material set is lecture handouts. Providing detailed materials in class helps students build confidence in lecture theory, algorithms, and examples to support program learning. For example, when I was tutoring on Federalist Paper classification for Hamilton, Jay, or Madison authorship I discovered several students struggling to build the corpus in the statistics program **R**. By providing a process flow-chart and code component examples one student dramatically expanded the approach for predicting characters in Shakespeare. As a research writer, I am committed to supporting science's body of knowledge and regularly provide students with relevant articles to further interest and have them build develop and appreciation of the research community. Finally, I work hard to balance homework, labs, and group projects to ensure course standards and student participation.

Every student of any group, race, and orientation has an opportunity to learn and be successful. From the noted Roman Marcus Aurelius, "nothing has such power to broaden the mind as the ability to investigate systematically and truly all that comes under thy observation in life." Computer science provides a platform to promote equity, build effective thinking, and the data economy facilitates students being able to actively incorporate scientific methods into their daily work and activities.