Course Syllabus: MSDS 7333
Quantifying the World

Course Designer:	Associate Professor Monnie McGee Department of Statistical Science Southern Methodist University, Dallas, Texas
Course Instructors: Office Hours:	Alan Elliott, Daniel Engels, Eric Larson, Monnie McGee and Melvin Greer (email: mbgreer@smu.edu)
Required Texts:	Nolan, D. and Temple Lang, D. (2015). *Data Science in R: A Case Studies Approach to Computational Reasoning and Problem Solving*. Boca Raton, FL: CRC Press (NTL). McKinley, Wes (2014, 2017). Python for Data Analysis. O'Reilly Publishers, available in PDF download (McKinley).
Prerequisites:	Last semester of MSDS Program

Learning Objectives: The student will

- Be able to form a testable hypothesis from an unstructured problem.
- Create an analysis plan to test afore-mentioned hypothesis
- Use code in R, SAS, and Python to perform various advanced analyses
- Learn about some advanced statistical and computational methods to analyze data
- Communicate the findings of a research project in a clear, concise, and scientific manner.

Course Overview

Quantifying the World (QTW) is built in two-week chunks of varying subject matter. The first week of each chunk consists of seventy – ninety minutes of theoretical and methodological background that will be necessary to solve the case study for the second week. Topics covered include multiple imputation, branching processes, parallel processing, management of very large data sets, time series, machine learning, and predication of location for indoor positioning systems.

Course Expectations (i.e. You have been warned …)

The Nolan and Temple-Lang text (NTL) contains example R code for 12 different case studies. We present five of them and the statistical/computational methods to fully understand them in this course. However, due to the ever-changing nature of R, this code doesn't always work as stated in the text. The same is true of the Python code in the McKinley text. You are expected to work out these issues as much as possible.

Grading: Case Studies (50%), Participation (30%), Asynchronous material (20%)

Case Studies (50%): Every other week in class will be spent working through a Case Study using R, SAS, or Python. These case studies will be written up and turned in for a grade. Case study write ups are to be "technical report ready", which basically means that it is suitable for public viewing.

Participation (30 %): Weekly postings to discussion board on various topics. During case study weeks, the discussion board will consist of hints and help with the case study. Please do not post the code that you have used to solve a problem. You may post hints and pseudo code as needed.

Asynchronous material (20%): Watching asynchronous material and answering questions in BLTs as required.

Course Coverage

Unit	Lecture Topic/ Case Study	Reading	Software	Presenter
1	Multiple Imputation	Notes on Imputation	SAS	Elliott
2	Replacing missing data in a real study	Case Study Coding and Write-Up	SAS	Elliott
3	Analysis of Time Series	Chapters 5, 10, and 11 of McKinley	Python	McGee
4	Financial Data Modeling	Case Study Coding and Write Up	Python	McGee
5	Modeling signal attenuation, cross-validation, densities and distributions	Chapter 1 of NTL and associated readings	R	Engels
6	Predicting Location via Indoor Positioning Systems	Case Study Coding and write up	R	Engels
7	Piecewise Linear Regression, Lowess, Longitudinal Data	Chapter 2 of text of NTL and associated readings	R	McGee
8	Modeling Runners' Times in the Cherry Blossom Race	Case Study Coding and Write-Up	R	McGee
9	Naïve Bayes, Recursive Trees, CART	Chapter 3 of NTL and associated readings	R	McGee
10	Using Statistics to Identify Spam	Case Study Coding and Write-Up	R	McGee
11	Branching Processes and Monte Carlo Simulation	Chapter 7 of NTL and associated readings	R	McGee
12	Simulation Study of a Branching Process	Case Study Coding and Write-Up	R	McGee
13	Parallel Computing, Split-Apply-Combine, SQL	Chapter 5 of NTL and associated readings	Python/R	Larson
14	Analyzing Airline Flight Delays – A 12GB data set	Case Study Coding and Write-Up	Python/R	Larson
15	Rewrite one of the case studies that you have done previously for the final exam.			

Rubric
Grading for assignments will total 100% based on the percentage of each component of the assignment. Weekly Assignments are due by 6:30pm CT on Tuesday each week. No assignments turned in after 6:30pm CT will be accepted or graded.

Rubric	**Percentage**
Assessment of submission adherence to specific weekly instructions as detailed in 2ds Unit Case Study. The key to a passing submission lies in following the specific directions found on the assignment page.	60%
Content of submission sections. Ability to write a report that explains work completed and demonstrates your ability to apply what you have learned	20%
Code appendix, section labeling, interpretation of results, appropriate use of graphics, spelling and grammar	20%
Total	**100%**

Letter Grade	**Percentage**
A	94% to 100%
A-	90% and less than 94%
B+	88% and less than 90%
B	84% and less than 88%
B-	80% and less than 84%
C	70% and less than 80%
F	0% and less than 70%

Rescheduling Assignments: Weekly Assignments are due by 6:30pm CT on Tuesday each week. No assignments turned in after 6:30pm CT will be accepted or graded. **Note: Issues with technology is NOT an excuse for late work.** You have been warned to start your assignments early enough so that you can resolve such issues before they affect your ability to turn in the work.
Submission guidelines for assignments

- Your name must be at the top of the first page and on each successive page.
- Submit case studies as a formal written paper using Microsoft Word only. The case study should have an abstract, an introduction, a literature review, a methods section, a results section, and a future work/discussion/conclusion section. Code should be included in an appendix to the document. Spelling and grammar count!
- Use an easy-to-read variable-width font (I like Ariel, Helvetica, and Geneva fonts—this document is in Helvetica 11 point) with a minimum of 11 point font.
- Relevant code and output must be included in-line at the appropriate point using Courier New (or other fixed width) font, in 10 point size. **Inclusion of irrelevant code or output, even in an appendix, will be penalized**. All software output must be given in the text or as a table created in Word (or the software you are using).

- Any graphics must be electronically cut and pasted in-line at the appropriate point of the write-up. You can use Word to resize the graphics appropriately. **Screen shots from SAS, R, or Python tabular output are not allowed in the document text or in the appendix.** All tables and figures should have descriptive titles and captions. In short, the reader should be able to understand the content of the figure or table without reading the associated text.
- Any mathematical notation must be provided with appropriate use of subscripts, superscripts, and symbols. Use MS Equation or another equation editor if you submit your work in Word.

Best Practices for Success in MSDS 73?? (and other courses also)

Attendance. Take responsibility for your commitment. Attendance means not only being there for synchronous sessions but also participating in asynchronous work.

Citizenship. You need to be actively engaged to succeed in this class. Talking on cell phones, texting, "facebooking," tweeting, or leisure web browsing are prohibited in class. I consider these to be a disruption (not to mention rude).

Integrity. A lot of the graded work occurs outside of class, so I expect honesty and integrity in what you submit for evaluation. Evidence of academic dishonesty will minimally result in zeros for all involved parties, and perhaps University-level disciplinary action. Don't risk your academic career.

Humility. Don't get lost! Ask questions in class. If something isn't clear to you, it probably isn't clear to others either. Questions may arise because I haven't made a connection clear or have inadvertently left out an important point. Your question gives me a chance to explain more clearly. Don't be proud or shy.

Organization. Don't procrastinate! This is a technology-driven course. Count on your computer failing or your wireless connection breaking the night before a due date. Start early and give yourself a chance to succeed.

Deadlines. You will generally have a week to complete an assignment. Due dates and times will be clearly indicated. Weekly Assignments are due by 6:30pm CT on Tuesday each week. No assignments turned in after 6:30pm CT will be accepted or graded.

Getting help. If questions arise while doing assignments/exams, do your best to resolve these questions before the assignment is due, first by taking time to seek answers yourself, and then via e-mail to your instructor or other students. **I encourage you and expect you to seek help.**

Collaboration. I encourage the formation of study groups and collaboration with your fellow students in tackling the assignments. Working together in groups on homework is permitted, even encouraged. **However, every student should write up and complete his or her homework independently. Students who chose to turn in exactly the same work will share the grade assigned.** Talking about problems with other people does help in learning, but just copying the solutions from one another doesn't help!

Looks do matter! All assignments must be NEATLY executed and organized. You risk a zero on any assignment submitted in a sloppy manner. See submission guidelines for more detail.

University Policies

Grading Policy: Graduate Students must receive a C or better in a course in order to pass the course. If a student must retake a course, then the second grade and the first grade are averaged for the purposes of the overall GPA. Failure to maintain a GPA of 3.0 or better will result in dismissal from the program.

Incompletes will be given only in the case of extraordinary circumstances that prevent you from finishing the semester. You must have completed at least 50% of the course with a passing grade to be eligible for an incomplete.

Religious Observance: Religiously observant students wishing to be absent on holidays that require missing class should notify the live session instructor via e-mail, and should discuss with the instructor, in advance, acceptable ways of making up any work missed because of the absence. (See University Policy No. 1.9.)

Excused Absences for University Extracurricular Activities: Students participating in an officially sanctioned, scheduled University extracurricular activity will be given the opportunity to make up class assignments or other graded assignments missed as a result of their participation. It is the responsibility of the student to make arrangements with the instructor prior to any missed scheduled examination or other missed assignment for making up the work (University Undergraduate Catalogue).