

Fall 2021 CDS Graduate Course

CSI 500

Computational Science Tools

Online Class, no scheduled meeting times



Syllabus

Dr. Stephen L. Scott

sscotta@gmu.edu

College of Science
Department of Computational and Data Science
Computational Social Science Program

Office hours: by appointment

Overview

This course introduces computer skills and packages commonly used in quantitative scientific research. The course includes materials from earlier versions of CSI 500, CSI 601, and CSI 602. This course is offered by Computational & Data Sciences and may not be repeated for credit.

Recommended Prerequisites: There are no formal prerequisites for this course. Students are expected to have some basic familiarity with computing and mathematics at the level of college algebra. However, online video discussions may include references to topics from calculus, vector and matrix algebra, descriptive statistics, and will include examples of elementary computer programming. Students without these prerequisites are absolutely welcome to enroll, as all the necessary mathematics and computer science topics will be introduced as needed and should not pose an impediment to completing the assignments successfully.

Registration Restrictions:

Enrollment in the course is limited to students with a class of Advanced to Candidacy, Graduate, Non-Degree or Senior Plus. The course is available to Graduate, Non-Degree or Undergraduate level students, however, students in a Non-Degree Undergraduate degree may not enroll.

Schedule Type: Online course for Spring 2020

Objective

This course is designed to provide students with a basic facility in four popular software tools used in data analytics and academic research. The tools are as follows:

- **LaTeX:** a computer language used for typesetting journal articles, technical reports, books, theses, and dissertations.
- **Python:** an object-oriented programming language widely used in industry and academia. Python offers full-featured scientific computing environment using a clean, elegant syntax that is easy to learn.
- **R:** a computer language and statistical analysis environment based on the S-Plus language originally developed by AT&T. R is widely used for statistical analysis, and provides high quality mathematical graphing capabilities.

- **NetLogo**: an agent-based modeling environment used for simulation of complex systems ranging from biology, chemistry, mathematics, economics, network science, anthropology, and other disciplines.

Course Organization

The course will be presented as a series of online learning modules. The course is very much "hands-on", and does not delve deeply into mathematical or computational theories. Each module will cover some aspect of one of the four tools, and lectures will include several short hands-on demonstrations or exercises. Students are strongly urged to complete the online exercises while viewing the video lectures so as to apply their knowledge of the tools. In addition, students will be assigned a weekly lab containing problem sets derived from the lecture slides.

The course also includes a "capstone" module, in which students will combine results obtained from the four learning modules into a short 3-5 page technical article demonstrating proficiency in preparing a manuscript in a form appropriate for submission to a peer-reviewed professional journal.

Course Grading

The course has no midterm exam, no final exam, and no "pop-quiz" exams. The course grade will be determined entirely based cumulative scores received on weekly lab assignments, each of which will contain approximately 3 to 5 problems derived from the video lectures for that week. All assignments are equally weighted, however, some assignments may require more extensive analysis or software coding, or combining interim results from previous lab worksheets. Students should expect to spend about 1-2 hours to complete each worksheet.

Assignments will be graded using the following criteria.

- Accuracy: is the assignment technically accurate? Does it generate the correct results?
- Skill proficiency: did the assignment demonstrate an understanding of the techniques referenced in the homework?
- Style Guide conformance: for software code, does it express the solution clearly and with enough internal documentation or comments? For PDF documents, do they have the proper formatting (margins, fonts, use of captions, etc.)
- Timeliness: Was the assignment submitted on time?
- Content: does the content of the submitted assignment address the problem or exercise posed in the homework?
- Completeness: did the assignment address all the problems / exercises posed in the homework?

There are 15 lab worksheet assignments in the course. Each assignment is weighted at 100 points for a maximum of 1500 points. Semester grades will be determined as follows.

Cumulative Score from Weekly Lab Assignments	Semester Grade
1350 and above	A
1200 to 1349	B
1050 to 1199	C
900 to 1049	D
Below 900	F

Late homework policy

Students are expected to complete assignments in a timely manner. Worksheets should be uploaded to the BlackBoard course site by 11:59:59 pm Eastern time on the due date, typically late Sunday evening. Late homework will be considered on a case-by-case basis the event that you are unable to complete an assignment by the assigned due date. Please email me to advise if this situation arises.

Homework grading policy

I will generally grade assignments within 24 to 48 hours of posting. Students should expect to receive feedback on their homework assignments using the BlackBoard system. Assignments will be graded on the basis of successful demonstration of proficiency in the assigned topics rather than computational efficiency or stylistic elegance.

Course Topics

Topics to be addressed in weekly lab assignments include (but are not limited to) the following.

LaTeX Weekly Lab Assignments

- Install the MiKTeX/TexStudio LaTeX environment
- Use built-in LaTeX document formatting templates
- Create document title, abstract, and hierarchical sections
- Create tables and figures
- Create references
- Create citations and bibliography

Python Weekly Lab Assignments

- Install the Anaconda Python IDE
- Learn Python data types and built-in data structures
- Learn to write Python functions
- Learn basic Object-Oriented features of Python
- Learn to read and write data files and CSV files
- Learn how to use Python plots for data visualization
- Learn how to use Jupyter Notebooks for documenting and sharing analysis products

R Weekly Lab Assignments

- Install R and RStudio environments
- Learn R data types and built-in data structures
- Learn to write R functions
- Learn R graphics for data visualization
- Learn R ggplot graphics for advanced data visualization

NetLogo Weekly Lab Assignments

- Install the NetLogo environment
- Learn basic agent-based modeling (ABM) concepts
- Learn to create your own functions and reporters
- Learn to use ABMs to analyze complex systems

Capstone Assignment

- Combine results from the four learning modules to prepare a technical paper in LaTeX including title, abstract, sections, tables, figures, references, citations, and bibliography
- Install a LaTeX template from a professional journal (Association for Computing Machinery) using the OverLeaf online LaTeX publishing system
- Combine results from the four learning modules to populate a LaTeX paper using the Association for Computing Machinery (ACM) publication template format

COVID-19 Statement

For Fall 2021, CSI 500 is planned to be taught as an entirely online asynchronous class. Students are not required to be on any George Mason University campus. In the event that you are affected by COVID-19 and unable to complete your assignments, please contact me so that we can discuss alternative arrangements. The class will follow GMU guidelines regarding COVID-19 protocols; if for example the University is required to suspend operations due to the public health emergency, we will follow the University policies and guidelines.

Academic Honesty and Collaboration

The integrity of the University community is affected by the individual choices made by each of us. GMU has an Honor Code with clear guidelines regarding academic integrity. Three principles to follow at all times are that: (1) all work submitted be your own; (2) when using the work or ideas of others, including fellow students, give full credit through accurate citations; and (3) if you are uncertain about the ground rules on a particular assignment, ask for clarification. No grade is important enough to justify academic misconduct.

Disability Statement

If you have a documented learning disability or other condition that may affect academic performance you should: 1) make sure this documentation is on file with the Office of Disability Services (SUB I, Rm. 222; 703-993-2474; <http://www.gmu.edu/student/drc>) to determine the accommodations you need; and 2) talk with me to discuss your accommodation needs.

Title IX Statement

As a faculty member, I am designated as a "Responsible Employee", and must report all disclosures of sexual assault, interpersonal violence, and stalking to Mason's Title IX Coordinator per University Policy 1202. If you wish to speak with someone confidentially, please contact one of Mason's confidential resources such as Student Support and Advocacy Center (SSAC) at 703-380-1434, or Counseling and Psychological Services (CAPS) at 703-993-2380. You may also seek assistance from Mason's Title IX Coordinator by calling 703-993-8730 or emailing titleix@gmu.edu .

GMU Library Resources

The GMU library includes a number of helpful resources. Here are some examples and points of contact. Feel free to make use of these GMU resources if and as needed during the semester.

Technical Librarians

Margaret Lam, Physical Sciences & STEM Data Librarian

Debby Kermer, Data Services Research Consultant

They can assist with the following topics.

- Learn R: https://infoguides.gmu.edu/learn_r
- Python for Data: https://infoguides.gmu.edu/learn_python
- Working with Data: <https://infoguides.gmu.edu/data-work>

Theresa Calcagno, IT & Engineering Librarian, can provide guidance on the following topics:

- identify and select appropriate publication/presentation venues
- prepare a publication proposal
- prepare a paper for publication/presentation
- survive the peer review process
- identify and avoid ethical issues related to publication and copyright

Textbooks and Resources

The course has no required textbooks. However, there are several recommended references listed below and students are highly encouraged to obtain these resources for the course. In addition, we will be reading several journal papers from the literature on the socioeconomics of diffusion of innovation.

Diffusion of Innovation Resources

Our semester project will involve an extended study of E. M. Rogers' Diffusion of Innovation theory. Here are some references on this domain; selected journal papers will be assigned as required readings in the first 2 weeks of class, although interested students may optionally read the book for additional depth. Note that Rogers' book has gone through five editions since its original publication in 1962; the citation below is for the 5th edition. However, the key concepts have remained consistent so any edition will be satisfactory.

- Rogers, E. M. (2010). *Diffusion of Innovations*. Simon and Schuster.
- Robertson, T. S. (1967). The process of innovation and the diffusion of innovation. *The Journal of Marketing*, 14-19.
- Rogers, E. M. (1976). New product adoption and diffusion. *Journal of Consumer Research*, 2(4), 290-301.
- Rogers, E. M., Medina, U. E., Rivera, M. A., & Wiley, C. J. (2005). Complex adaptive systems and the diffusion of innovations. *The Innovation Journal: The Public Sector Innovation Journal*, 10(3), 1-26.

LaTeX Resources

Quick Reference

https://www.nyu.edu/projects/beber/files/Chang_LaTeX_sheet.pdf

Online References

Oetiker, T., Partl, H., Hyna, I., & Schlegl, E. (2016). The not so short introduction to LATEX 2 ϵ .

<https://tobi.oetiker.ch/lshort/lshort.pdf>

Krishnan, E., Ed. (2003). LATEX Tutorials: A Primer.

<https://www.tug.org/twg/mactex/tutorials/ltxprimer-1.0.pdf>

Trade Books

Lamport, L. (1994). LATEX: a document preparation system: user's guide and reference manual. Addison-Wesley.

Kopka, H., & Daly, P. W. (2003). Guide to LATEX. Pearson Education.

Abrahams, P.W., Berry, K. and Hargreaves, K.A. (1992). TeX for the Impatient. Addison-Wesley.

Python Resources

Quick Reference

https://s3.amazonaws.com/assets.datacamp.com/blog_assets/PythonForDataScience.pdf

http://www.cogsci.rpi.edu/~destem/igd/python_cheat_sheet.pdf

Online References

Downey, A. (2012). *Think Python*. O'Reilly Media, Inc.

Available as a free PDF download at <http://greenteapress.com/wp/think-python/> . Make sure you obtain the 2nd edition for Python 3.x, not the 1st edition for Python 2.x.

<http://docs.python.org> contains additional documentation at the Python website.

Trade Publications

Lutz, M. and Asher, D. (2007) Learning Python, 3rd Edition. O'Reilly & Associates.

Zelle, J. (2016). Python Programming: Introduction to Computer Science. Franklin, Beedle and Associates Inc.

R Resources

Quick Reference

<https://cran.r-project.org/doc/contrib/Short-refcard.pdf>

<https://www.rstudio.com/wp-content/uploads/2016/10/r-cheat-sheet-3.pdf>

<https://www.rstudio.com/wp-content/uploads/2015/03/ggplot2-cheatsheet.pdf>

Online References

An Introduction to R. Notes on R: A Programming Environment for Data Analysis and Graphics Version 3.4.3 (2017-11-30) <https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf>

Haschke, P. (2013). Introduction to R. University of Rochester.

<http://www.sas.rochester.edu/psc/thestarlab/help/rcourse/R-Course.pdf>

Trade Books

Dalgaard, P. (2008). *Introductory statistics with R*. Springer Science & Business Media.
Kabacoff, R. (2011). *R in Action*. Manning Press.

NetLogo Resources

Quick Reference

<http://luis.izqui.org/resources/NetLogo-6-0-QuickGuide.pdf>

Online References

Scott, S., and Koehler, M. (2013) A Field Guide to NetLogo.

https://s3.amazonaws.com/complexityexplorer/ABMwithNetLogo/Field+Guide+to+NetLogo+v14-netlogoExtension-index_02.pdf (Note: covers versions up to 5.x - minor compatibility issues for 6.x)

OpenABM tutorials and references. <https://www.openabm.org/page/modeling-tutorials>

Trade Books

Wilensky, U., & Rand, W. (2015). An introduction to agent-based modeling: modeling natural, social, and engineered complex systems with NetLogo. MIT Press.

Railsback, S. F., & Grimm, V. (2011). *Agent-based and individual-based modeling: a practical introduction*. Princeton University Press.

Required Software

Students are expected to download and install the following tools and packages. Version numbers are the most recent versions available and the ones we'll be using in class. Although not recommended, you may install other versions at your own discretion.

- In order to use LaTeX, students will need to download an implementation of the Tex libraries for your workstation operating system (Windows, MacOS, Linux). There are several free implementations available; you may download the appropriate installer at <https://miktex.org/download>.
- In addition, we will be using the TexStudio IDE for LaTeX, which is available as a free download for a variety of platforms. See the "Downloads" tab at the website at <https://www.texstudio.org/>.
- We will be using Python version 3.8, which is available as a free download for a variety of platforms (Windows, MAC OS, Linux). However, students are strongly encouraged to use the Anaconda scientific distribution, which contains a set of pre-installed scientific and engineering software libraries. Anaconda downloads for various platforms are available at <https://www.anaconda.com/products/individual>.
 - Be sure to download the free edition as there is no need to pay for a commercial license for this course. The course will be taught using Anaconda's built-in Spyder editor and

Integrated Development Environment (IDE), so students using alternative Python installations or code editors will be responsible for resolving any library compatibility issues.

- **CAUTION:** the most recent builds of the Anaconda individual edition of the Python/Spyder IDE have a known bug that limits the IDE to editing only, and it won't support compiling or running Python. If you experience a "unable to start kernel" error using the default Anaconda Individual edition, uninstall said installation, and instead install a legacy release from the archives site at <https://repo.anaconda.com/archive/>. I have successfully used the older release Anaconda3-5.3.1 on a Windows machine - https://repo.anaconda.com/archive/Anaconda3-5.3.1-Windows-x86_64.exe
-
- We will be using NetLogo 6.2.0, available at <https://ccl.northwestern.edu/netlogo/download.shtml>
- We will be using R 4.1.1 - available at for Windows at <https://cran.r-project.org/bin/windows/base/>, for MacOS at <https://cran.r-project.org/bin/macosx/> and for Linux at <https://cran.r-project.org/bin/linux/> .
- We will also be using RStudio Desktop 1.4.1717 (Note: download RStudio Desktop free open source edition, NOT the desktop commercial, commercial server, or commercial server PRO editions) available at <https://www.rstudio.com/products/rstudio/download/>

Students are expected to provide their own computing resources to complete assignments. GMU has several computing labs available if you need access to computing platforms.

Inclement Weather Policy

As this is an online class and there are no scheduled meeting times, students are expected to complete assignments and keep up with the course content even in the event of weather emergency conditions resulting in the closure of the GMU Fairfax campus. In the event of extraordinary circumstances, other class options may be provided; check the website at www.gmu.edu for updates and also check the class Blackboard site for announcements and updates.

Office Hours

This course is offered entirely as an online course, so we will not be meeting physically in a classroom setting. In order to provide an option for one-on-one interaction or small group discussions, I have a George Mason University Zoom account that can be used as a virtual office.

If you would like to set up an office hour discussion, send an email to my GMU faculty account (sscotta@gmu.edu) by Thursday, and I will set up the virtual office for Friday between the hours of 1pm and 2pm. Note that student participation is welcomed, but attendance is not required and is not part of the grading expectations.

Zoom Meeting Dial In Information

Topic: Stephen L Scott's Personal Meeting Room

Join Zoom Meeting

<https://gmu.zoom.us/j/2881593065>

Meeting ID: 288 159 3065

One tap mobile

+13017158592,,2881593065# US (Washington DC)

+12678310333,,2881593065# US (Philadelphia)

Dial by your location

+1 301 715 8592 US (Washington DC)

+1 267 831 0333 US (Philadelphia)

Meeting ID: 288 159 3065

Find your local number: <https://gmu.zoom.us/j/2881593065>

Discussion Boards

If you would like to set up a discussion board to exchange ideas with fellow students, send an email to my GMU faculty account (sscotta@gmu.edu) and I will enable write-access for the topic. Participation in discussion boards is entirely voluntary and does not affect course grading.

Emails

I will be checking my GMU email regularly as well, so if you have questions or comments, send email to my faculty email account (sscotta@gmu.edu). Note that per GMU university policy and compliance with federal FERPA regulations, I can only respond to student emails from a GMU email address, i.e., an address ending in @gmu.edu.

Course Outline

The following schedule shows the topics to be presented. Some variation in presentation schedule may occur, however. Students are encouraged to visit the class Blackboard website frequently for the latest updates and announcements. Note the due date for weekly lab assignments: these should be submitted to the online BlackBoard system by the date and time specified, using the file format(s) specified in the worksheet problem sets. Worksheets that require software source code should be formatted according to the guidelines listed at the end of this syllabus.

Week	Module	Topics	Lab Worksheet Assignment	Due Date (posted to BlackBoard by 11:59:59 EST)	Number points
1	LaTeX	LaTeX Basics, Articles, Reports	LaTeX Worksheet #1	8/29/2021	100
2	LaTeX	Tables, Equations, Bibliographies	LaTeX Worksheet #2	9/5/2021	100
3	LaTeX	Table on Contents, Table of Tables, Table of Figures, Fonts	LaTeX Worksheet #3	9/12/2021	100
4	Python	Python basics, variables, expressions, strings, lists, formatted printing	Python Worksheet #1	9/19/2021	100
5	Python	tuples, dictionaries, functions, recursion	Python Worksheet #2	9/26/2021	100
6	Python	read/write data and CSV files, Objects and classes, Objects and inheritance	Python Worksheet #3	10/3/2021	100
7	Python	Matplotlib graphing, NetworkX graphs, Python applications, Jupyter Notebooks	Python Worksheet #4	10/10/2021	100
8	NA	Intro to R, Intro to RStudio, variables, expressions, numbers, vectors, Objects, modes, attributes	R Worksheet #1	10/17/2021	100

Week	Module	Topics	Lab Worksheet Assignment	Due Date (posted to BlackBoard by 11:59:59 EST)	Number points
9	R	Factors, Arrays and matrixes Lists and DataFrames, Read/write data files,	R Worksheet #2	10/24/2021	100
10	R	grouping, loops, conditional execution Functions, standard R graphics,	R Worksheet #3	10/31/2021	100
11	R	enhanced ggplot R graphics, R applications	R Worksheet #4	11/7/2021	100
12	NetLogo	Intro to NetLogo, agent-based modeling	NetLogo Worksheet #1	11/14/2021	100
13	NetLogo	agents and attributes functions and reporters applying a simple social network model	NetLogo Worksheet #2	11/21/2021	100
14	NA	Thanksgiving	NO CLASS	11/28/2021	0
15	Capstone	Journal paper formatting Background and bibliographic citations Methods, Discussion	Capstone Worksheet #1 (will mostly include working on your final document)	12/5/2021	100
16	Capstone	Findings, graphics and tables, conclusions, generating the bibliography, final PDF production	Capstone Worksheet #2 (finalize your camera-ready article using professional journal formatting standards)	12/12/2021	100

Source Code Formatting Conventions

For assignments involving creation of computer source code, the following conventions will be expected.

LaTeX conventions

LaTeX source code should include the following comment block at the beginning of each source code file.

```
% filename: name_of_file.tex
%
% description: description of what this file is supposed to do
%
% your name
% your class (CSI 500)
% date
% assignment number
%
% anything else important for me know when grading your assignment
%
% any use of 3rd party code or libraries should be properly cited
%
```

You are encouraged to add comments to your LaTeX source code to document key sections or features, such as the following example shows.

```
% =====
%
% this is the section where we provide a review of the literature
%
% =====
/section{Background and Literature Review}

% =====
%
% this is the section where we talk about methods
%
% =====
/section{Methods}
```

Python conventions

Python source code should include the following comment block at the beginning of each source code file.

```

# filename: file_name.py
#
# description: what is this code supposed to do
#
# your name
# your class: CSI 500
# date
# assignment number
#
# anything else important for me know when grading your assignment
#
# any use of 3rd party code or libraries must be properly cited

```

Python classes and functions (which we'll cover in lectures) should be documented as such.

```

#
# function name
# function description (what is this function supposed to do)
#
def myfunc():
    code here
    code here

#
# class name
# class description (what is this class supposed to do)
#
class myclass():
    code here
    code here

```

R conventions

R source code should include the following comment block at the beginning of each source code file.

```

# filename: file_name.R
#
# description: what is this code supposed to do
#
# your name
# your class: CSI 500
# date
# assignment number
#
# anything else important for me know when grading your assignment
#
# any use of 3rd party code or libraries must be properly cited

```

R functions should be documented as follows.

```

#
# function name
# function description (what is this function supposed to do)
#
function myfunc( params ) {
    code here
    code here
}

```

R data structures important to your code should be similarly documented.

```

#
# sale.data
# this is the data frame for simulated random sales data
#
sales.data = data.frame(yr=2000:2014,qty=rnorm(15,1000,100))

```

NetLogo conventions

NetLogo source code should include the following comment block at the beginning of each source code file. Note that in NetLogo, you will be submitting a complete model package file (*.nlogo), and not just the source code - we'll cover this later in class lectures.

```

; filename: file_name.nlogo
;
; description: what is this code supposed to do
;
; your name
; your class: CSI 500
; date
; assignment number
;
; anything else important for me know when grading your assignment
;
; any use of 3rd party code or libraries must be properly cited

```

NetLogo procedures and reporters should be documented as follows.

```

; procedure name
;
; description of what the procedure does
;
to procedure-name[ optional-param ]
    code here
end

; reporter name
;

```



```
; description of what the reporter does
;
to reporter-name [ optional-param ]
  code here
  report return-value
end
```