CDS 102 Syllabus

Introduction to Computational and Data Sciences

The One-Page Summary

What: This lab course will supplement and extend the material from CDS 101. This lab is designed to be taken alongside CDS 101 (or in a later semester *after* having already taken CDS 101).

When and Where: Section 203 of CDS 102 will meet Thursdays at 4:30-5:45 pm in Research Hall 249.

Who: This section of CDS 102 is taught by Abdullah Almalki, Graduate Lecturer in the Department of Computational and Data Sciences (email: aalmalki@gmu.edu). Your section also has the following STARs (teaching assistants): TBD. Office hours are posted on Blackboard.

How: This course is organized into labs

There are three policies that are particularly important to remember:

- *The "Hogwarts" Policy*: Help will always be given in CDS 102 to those who ask for it. If you are stuck, please reach out to your instructor and/or a STAR (TA), and we will guide you through the material.
- *Academic Honesty*: Plagiarism, copying, cheating, sharing answers, etc. are strictly prohibited. You will be caught and reported to the University, which can result in failing the course or expulsion. Limited use of AI tools is permitted (see the AI tools section of the syllabus for more guidance).
- *Communication*: All class-related communication should be through Slack (not email).

The Fine Print

- Classroom: Research Hall 249.
- Meeting times: Thursdays at 4:30-5:45 pm.
- University holidays: Spring Recess (Mon. Mar 4 - Sun. Mar 10).
- Credit hours: 1.0 credit hour.
- Prerequisites: None, but a background in algebra is assumed.
- Mason Core: This course counts towards the Mason Core Natural Science + lab when taken with the CDS 101 lecture course.

Course Description

During this course, students will develop basic skills for loading, cleaning, transforming, and visualizing real-world datasets using the R programming language and the RStudio integrated development environment. Statistical methods for analyzing, interpreting, and predicting dataset trends are then introduced and approached from a computational point of view using randomization and simulation. An emphasis is placed on documenting one's scientific work using the R Markdown format to fulfill the principles of reproducible research. Connections are highlighted between statistical inference and the scientific method and how this is related to both the scientific method's power and its limitations. These tools will also be used to critically examine statistical claims reported in mass media, demonstrating how scientific literacy and a basic knowledge in statistics are indispensable tools to making sense of our modern world. Special topics like machine learning and dashboards may also be covered as time allows. The labs complement and allow further practice for the topics covered in CDS 101.

Objectives

By the end of the course, students will be able to:

- Obtain, clean, transform, and visualize a dataset using the R programming language.
- Interpret, and predict dataset trends using statistical inference and models.
- Document their work using R Markdown, a reproducible research format.
- Manage files and source code using GitHub.

Course Materials

Textbooks

This course utilizes two textbooks that are freely available online:

- R for Data Science: Import, Tidy, Transform, Visualize, and Model Data by H. Wickham and G. Grolemund (O'Reilly Media, Sebastopol, CA, 2017). http://r4ds.had.co.nz/
- Statistical Inference via Data Science: A ModernDive into R and the Tidyverse by C. Ismay and A.Y. Kim (Chapman and Hall/CRC, 2019). https://moderndive.com

Software

During the course we will use various software for writing code in the R programming language. This software is all free and installation instructions will be provided.

Platforms

The course will be administered through the following online platforms:

- GitHub: https://github.com
- Slack: Link on Blackboard
- Blackboard: https://mymasonportal.gmu.edu

The course website operates as the central repository for course materials, copies of the lecture slides and handouts, homework instructions, and links to the lecture videos hosted on YouTube. Slack is the primary communication medium, replacing email (see the Contact policy below) while also serving as a discussion board. GitHub is used for connecting your class files to RStudio, tracking changes, distributing starter files for homework assignments and certain module exercises, and for project collaborations. Blackboard is used for module exercises, part of the midterm exam, for submitting assignments, and returning grades.

Policies

Contact policy

All correspondence is to be done using the private, invite-only Slack workspace for the course. Direct messages on Slack are to be used for contacting me instead of emails. My ground rules for direct messages are as follows:

- I check and respond to messages periodically (weekdays 11am-10pm).
- Allow up to 24 hours for a response during normal hours.
- Questions about homework problems or troubleshooting a technical issue should be posted to the relevant Slack channels so one of the team can help you.
- Just because I view a message does not mean I will respond right away.
- I generally don't respond to messages over weekends and school holidays.
- If your questions are involved enough, I will ask you to schedule an appointment with me.
- Email policy: Emails sent during the first week of classes will be responded to, but I will respond to you using Slack. Emails sent to me after the first week will be ignored. (Obviously, if your issue is about not being able to access Slack, then you should send me an email!)

Tech support: R, RStudio, GitHub, and your computer

Post all technical issues or error messages for R, RStudio, GitHub, and your computer in the designated Slack channel #rstudio-github-help. This is so that other students can either help out or see how to resolve what is likely a common problem. If it becomes clear that the error or issue is highly specific, then discussion can be moved to Direct Message or handled via a screen sharing session.

When posting about an issue, here are some basic questions to answer that will help with troubleshooting:

- What did you expect to happen when you ran your code?
- What is actually happening when you run your code?
- If there's an error message, tell us what it is. A screenshot works, provided you a) don't crop the image as that can remove useful information by accident, and b) take a real screenshot, not a photograph of your screen using your phone.
- Is there any other context we should know? For example, if a file won't load, did you check that you are in the correct project or that the file actually exists? Did your issue appear only after you worked on a different assignment? Did you recently install a package not used in class?

Illness and emergencies

It is a student's responsibility to inform me about illnesses or personal/family emergencies that will interfere with submitting work on time. This must be done as soon as possible. In case of illness, you may be asked to provide a doctor's note before being granted an assignment extension or exemption.

I understand that certain emotional or physical situations can impact a student's willingness and ability to communicate what is going on and that it can take a few days to inform me about a personal emergency or severe illness. At the same time, all students are expected to exercise personal responsibility. It is not acceptable to wait to tell me about the impacts of a personal illness or emergency until you're about to fail the course due to missing multiple submission deadlines. Exemptions may be granted at my discretion.

Accommodations policy

Students with disabilities who need academic accommodations, please see me and contact the Office of Disability Services (ODS) at (703) 993-2474. All academic accommodations must be arranged through the ODS: http://ods.gmu.edu/.

Campus closures

If the campus closes, or if a class meeting needs to be canceled or adjusted due to weather or other concern, students should check Blackboard and Slack for updates on how to continue learning and for information about any changes to events or assignments.

Grading

110 points will be available to earn over the semester divided over the following activities:

Category	Weight
Labs:	100%

You will submit twelve graded lab reports over the course of the semester, each worth 10 pts. The best eleven of those lab reports will be counted towards your grade (i.e. the lab report with the lowest score will not count towards your final grade).

Based on the final total score, your final grade will be determined as follows: A+ [98-100], A [94-97.99], A- [90-93.99], B+ [88-89.99], B [84-87.99], B- [80-83.99], C+ [78-79.99], C [74-76.99], C- [70-72.99], D [65-69.99], F [<65].

Individual requests for extra credit or a grading curve will not be granted, no exceptions. Any opportunities to earn extra points will be offered to the entire class. Grading curves are handled on a per-assignment basis and are applied to all students equally.

Late work policy

Unless otherwise noted, assignments are to be submitted by 11:59pm on the due date. The following penalties apply for most assignments (please note that weekends count as days):

- First day late, by 11:59pm: -15%
- Second day late, by 11:59pm: -30%
- Third day late or later: no credit

Please ask for an extension if you need it - reasonable extension requests are usually granted. However, extensions need to be requested more than 24 hours *before the due date* **in writing** (preferably via Slack). Extensions are to be completed within the time-frame I set forth.

Students are responsible for informing me *in advance* about any religious holidays, scheduled varsity sports trips, or other school-sponsored activities that will interfere with submitting an assignment on time.

Students with 4 or more labs that are unsubmitted or 3+ days late will receive an automatic F for the course, no exceptions.

Regrading appeals policy

Regrade appeals need to clearly state what you want regraded and justify the request by citing evidence. Appeals are only to be used for correct answers being marked as incorrect, misapplication of the grading rubric, or incorrectly tallied points. The number of points a question, exercise, or rubric category is worth or that were deducted for an incorrect answer or mistake cannot be appealed and are not up for debate or negotiation.

Regrade appeals must be sent in writing (preferably via Slack message).

Expectations

Labs: At the beginning of each lab you will be provided with a set of instructions and a link to obtain your lab repository on GitHub. The lab reports will be completed and submitted using the starter files provided to you in the lab repository. To fill out the lab reports, you will need to read and follow the instructions. The labs are similar to interactive online tutorials, where you will read some instructions, see an example, and then complete an exercise to demonstrate that you understand a concept. The exercises consist of many kinds of activities, such as creating visualizations, manipulating datasets, performing statistical analysis, and building models. Many of the labs then conclude with a set of questions where you will perform additional analysis and interpret the meaning of your results. When you have completed writing up your report in the RMarkdown file format, you will push the final result to GitHub.

You will be provided with lab report templates each class, and a complete lab report should have the following:

- Your name
- A title
- The date you are submitting the lab report
- A sub header for each exercise and each question, with your answers and work going in the space in between the headers
- Your responses will document your procedure for completing each exercise and question. This involves explaining each code block that you write. Be precise. Write how you would write a tutorial for a new user. Do not assume that the reader will fill in any missing steps.
- The finished lab report should have proper grammar and spelling.
- Follow the rule that anytime you need to write a response or explain a code block, you should write in complete sentences. If you are interpreting something like a visualization or a table, it is likely that you will need to write more than one sentence.

If you are in an in-person lab section, you may need to finish writing up the lab report outside of class. See each lab report for the due date.

Final exam: There is no final exam for the lab section.

Conduct

Academic integrity

> "Student members of the George Mason University community pledge not to cheat, plagiarize, steal, or lie in matters related to academic work."

Students in this class are permitted to ask questions about the assignments on Slack and discuss assignments in private communications, however it is important to make sure that you write your assignments by yourself and in your own words, meaning that students are not permitted to collaborate on write-ups for assignments and projects. In the same vein, do not duplicate or paraphrase another person's material or ideas and represent them as your own. "Individual assignment" is the default classification for all assignments, exams, and projects in the course; any exceptions to the rule will be noted in the instructions. Content that comes from a resource (including an AI tool) or another student should be properly cited.

A note on sharing or reusing code found on other GitHub repos or on websites like Wikipedia or Stack Overflow. I am aware that there are solution sets, sample snippets of code, etc. that can be of use while working on your assignments, projects and exercises during the course. It's common knowledge that researchers in both industry and academia will use search engines while writing code. Being able to search for existing solutions so that you don't "reinvent the wheel" is a useful skill. Therefore, unless I specify otherwise, you are permitted to use these resources **as long as you provide a citation**.

Exceptions to this rule are:

- For individual assignments, you cannot reuse anything from another student's work (past or present), including but not limited to R Markdown documents, code, plain text explanations, etc.
- You are not permitted to consult or use solution sets for any of the assignments, activities, and projects for the course.
- You are not permitted to ask other students from this or previous semesters for copies of their assignments or projects, even to use for reference.
- ANY MATERIAL THAT IS TAKEN IN WHOLE OR IN PART FROM ANOTHER SOURCE AND NOT PROPERLY CITED WILL BE TREATED AS A VIOLATION OF MASON'S ACADEMIC HONOR CODE.

Other violations of Mason's Honor Code will be treated similarly. Suspected violations will be reported to the Office of Academic Integrity. The minimum sanctions I will recommend are:

- Minor infraction (e.g. improper citation):
  - First offense (university-wide): 0% on the assignment.
  - Second offense: F for the course.
  - Third offense: expulsion from the university.
- Major infraction (e.g. cheating in an exam, or copying from another student's lab/project, or submitting an AI generated lab):
  - First offense: F for the course.
  - Second offense: expulsion from the university.

AI policy (e.g. ChatGPT)

Recent developments in "AI" tools such as ChatGPT mean that it is possible to use programs to generate plausible looking code and interpretations to some of the assignments in this course.

(Sometimes, these are even correct!)

However, the goal of this course is for you to learn how to write computer code and understand the output of that code. The reason that this is important is because you cannot assess whether AI generated code is correct without understanding how to write the code without AI.

In this course, the work that you submit must be in your own words. I.e. it is not ok to get ChatGPT (or any other AI tool) to complete your assignment for you. Submitting answers that you have obtained from ChatGPT is like to asking another person to complete your assignment for you, and will be treated as a case of cheating and referred to the Office of Academic Integrity as an Honor Code violation (resulting in an automatic F for the entire course).

However, you are welcome to use AI tools to help you better understand parts of the course work, as long as you are not relying on it to complete the exercises.

AI tools should help you learn more, not replace your learning.

As a rule of thumb, no more than 25% of the code you write (or blanks that you fill in) should be coming from external sources such as websites or AI. In both cases, you should cite where you got this code from, so that you are not flagged for cheating.

If you use an AI tool, you need to submit your interaction with it as a supplementary submission along with the assignment.

Written answers should **not** be generated using AI.

Verbal examination: If you use excessive amounts of AI generated code in a piece of coursework, then you may be required to meet with your instructor for a *viva* (a verbal examination) in which you will be required to explain how this code works. If you cannot explain what a piece of code is doing and why you included it then you will get a 0 for that exercise. If you cannot explain multiple pieces of code then you will be referred to the Office of Academic Integrity for an Honor Code violation (see previous section).

Decorum/discourse

Students are expected to be civil in their conduct and respectful of their fellow classmates and the instructor for the duration of the course on all discussion platforms. Students are expected to follow proper grammar and punctuation in their posted messages and to refrain from using internet slang, abbreviations, and sarcasm.

I will address violations of classroom decorum on a case-by-case basis and reserve the right to enact grade-based penalties for disruptive or repeat violations. Penalties for decorum violations cannot be negotiated or appealed.

Mason diversity statement

George Mason University promotes a living and learning environment for outstanding growth and productivity among its students, faculty and staff. An emphasis upon diversity and inclusion throughout the campus community is essential to achieve these goals. Diversity is broadly defined to include such characteristics as, but not limited to, race, ethnicity, gender, religion, age, disability, and sexual orientation. Diversity also entails different viewpoints, philosophies, and perspectives. Attention to these aspects of diversity will help promote a culture of inclusion and belonging, and an environment where diverse opinions, backgrounds and practices have the opportunity to be voiced, heard and respected.

Disclaimer

The instructor reserves the right to modify this syllabus at any time during the course to improve the learning experience and classroom environment. The pacing of the course and the list of covered topics may also be adjusted in response to student progress. The course objectives reflect what a student is expected to understand by the end of the course after putting in the necessary time and effort both inside and outside the classroom and completing all assignments. These outcomes are not a guarantee, and students will get more out of the course the more they put into it. Any acquired skills and knowledge can fade over time if not reviewed or practiced after the course concludes.

Schedule

A provisional calendar for the labs will be made available on Blackboard. Timings of different labs may be changed subject to our progress throughout the semester.