Final Portfolio

Due: December 11, 2017 @ 11:59pm

Overview

In lieu of a written final exam, you will construct a personal course portfolio containing three elements:

1. An annotated list of R functions

2. Evidence of learning and growth during the semester

3. Comparative discussion of related simulations

You will be provided with a starter repository on Github that must be used for putting together your individual portfolio, following the instructions provided below. The portfolio must be submitted by the end of the day on **December 11th, 2017** by issuing a *Pull Request* in the usual way. Students will then present and discuss their portfolio with the course instructor as part of the course's 5–10 minute final interview on **December 18, 2017** during the scheduled final exam period.

Grading

The assessment of your portfolio and final interview will be combined into a single grade, with the portfolio worth 70% and the final interview worth 30%. As per the syllabus, the combined grade is worth 30% of your overall course grade.

Portfolio guidelines

Consider the following guidelines as you get started with putting together your portfolio, **but please note that this list is not exhausitive**. It is important that you fulfill the basic requirements for each element, but organization, thoroughness, and creativity are encouraged. Grades will reflect the quality of the portfolio, such as whether the portfolio simply contains the minimal work needed to "check the boxes" or if the student has gone above and beyond to demonstrate growth in and mastery of the course topics.

Annotated list of R functions

Purpose — A significant portion of the course is dedicated to learning how to use R and the `tidyverse` ecosystem to wrangle and explore data, and then analyze it using statistics. Remembering the different functions, how they are used, and what problems they solve requires consistent practice and review. This is why it's important to have personal notes that explain how a function works in your own words, which can be referenced long after you've completed this course.

For your portfolio — Put together an annotated list of R functions for your portfolio by adding information to the template file in your starter repository. The template file comes with a pre-filled example for the `ggplot2` syntax and how to use `geom_histogram()`, as well as listing other sections that are expected to be part of your notes. Use these examples as inspiration for how to put together notes on the other functions in R.

When writing notes for a function, please include the following:

- The function's name

- The important inputs

    - If the input was used in class or for a homework assignment, then it's important

    - For `ggplot2` functions, include a separate subsection for important aesthetic mappings (see example)

- A summary — 1 or 2 sentences — of what the function does

- An example showing how to use the function

At a minimum, your notes should discuss the following packages and functions:

- base R

    - `c()`

    - `length()`

    - `seq()`

    - `sqrt()`

- The `ggplot2` package

    - Anything we've used in class or on a homework assignment

- The `readr` package

    - `read_csv()`

- The `dplyr` package

    - `select()`

    - `filter()`

    - `mutate()`

    - `group_by()`

    - `summarize()`

- The `tidyr` package

    - `gather()`

    - `separate()`

- `stats` functions

    - All summary statistics discussed in class, such as `mean()` and `quantile()`

    - The normal distribution functions, such as `dnorm()`, `rnorm()`, `pnorm()`, and `qnorm()`

    - Modeling functions: `lm()`

- The `infer` package

    - Hypothesis test: point estimate comparing a numeric or categorical variable from a dataset with an ideal normal distribution

    - Confidence interval: using bootstrapping to compute the confidence interval

Remember, these notes are ultimately for you, so think about what you would find helpful when you refer back to these.

Evidence of learning and growth

Purpose — You've practiced and reviewed a lot of different skills this semester during in-class demos, group work, message board posts about the readings, homework assignments, and the midterm project. You've learned a lot! Now it's time to reflect on what you've accomplished and learned about computational and data science during the course. This section of your portfolio should be a combination of story telling and scientific argument, meaning that you need to support your claims by presenting *evidence* of learning. Your evidence should be in the form of digital

artifacts, which in most cases will be your RMarkdown files. Take care with what you include, as each digital artifact in your portfolio must be mentioned and should not contradict what you write about your learning experience.

For your portfolio — Consider the following guidelines and suggestions when compiling your digital artifacts and writing about your learning experience:

- The digital artifacts **must** be copied into the `artifacts` folder in your Github repository. Once copied, you can stage, commit, and push the files as usual. If you do not understand how to do this, it is important that you talk with the instructor about it as soon as possible.

- The following items are things you can include as digital artifacts: any RMarkdown files you created during class, **your** contributions during the midterm project, your homework assignments, and your posts on Github about the readings.

- Think about chronology: consider the dates you created or submitted the digital artifacts you are presenting as evidence. Your artifacts should be reasonably spaced out date-wise when visualized as part of a timeline.

- At the very minimum, there should be at least **four** major pieces of evidence, with the first piece corresponding to weeks 1–3 of the course, the second piece to weeks 4–6, the third piece to weeks 7–9, and the fourth piece to weeks 10–12.

- Use an *evidence-driven* approach for your writeup, meaning that you first select your evidence and then put together your write-up.

- There are two approaches you can take for presenting each piece of evidence of learning (you can make use of both kinds in your portfolio)

  - Fix and clean up your earlier work. When doing this, make sure to include **before** and **after** versions of the files to allow for a direct comparison. Your evidence of learning is discussing and showing how you are now better positioned to identify mistakes and improve the presentation of your earlier work.

  - Present your favorite writeups from during the semester and trace how we can see an "arc" of learning and improvement through them. Note that using this approach requires that the original work was mostly free of errors the first time around.

- If applicable, present how you've started applying things we've learned during this class to *other* classes at GMU, at your job, or to your personal hobby.

Comparative discussion of simulations

Purpose — The field of computational and data sciences extends beyond the topics focused on in this course. From the CDS-101 perspective, we've used the terms **model** and **simulation** as shorthand for *data-driven* models and simulations. Yet, there is an alternative approach to model and simulation building that works in the opposite direction and is an indispensible tool in the natural sciences and engineering. This class of models and simulations *generate* predictions and data wihout using an underlying dataset as input. To distinguish these from their *data-driven* counterparts, we will refer to them as follows:

- A *microscopic* or *mechanism-driven* model or simulation is based on the known laws of nature. An example is deriving equations of motion for the planets in our solar system using Newton's law of universal gravitation.

After building this kind of model or simulation, the researcher will scan the model's parameter space and look for trends in the predictions and outputs, which are then compared against experimental data (if available). If the model or simulation generates predictions or outputs that accord with the experimental data, then the proposed mechanism can be regarded as a plausible explanation for observed trends. However, if the predictions or outputs fail to agree with the experimental data, then the model or simulation is falsified and the proposed mechanism is rejected.

For your portfolio — In the near future you will be provided with a short summary of two or three simulations that share a common lineage that you can easily run on your computer. The simulations will be visual and interactive, allowing you to change a small set of parameters using a simple dashboard. After becoming familiar with the different simulations and developing a basic intuition for how each one behaves, you will then compare and contrast them in a

short write-up. Your comparative discussion must be at least 2–3 paragraphs in length (a minimum of one paragraph per simulation) and include the following:

- At least three ways in which the simulations are similar to one another

- For each simulation, at least one way it is *different* from the others

- Pick one of the simulations and suggest a feature or rule that you could add to it that would change its outputs and predictions. You only need to do this using plain language, you are not expected to write any code for this. Be sure to hypothesize what you think the changes will do and what they might mean. For example, how do you anticipate that your proposed change will simulate different physical mechanisms or human behavior?