EAS 560 – Data Science Challenge

(Due: Friday, 14 August 2020, 5:00pm)

The goal of this data science challenge is to **practice the development, implementation, and application of a real-world data mining and modeling code**. Your challenge is to take one of the following COVID-19 modeling codes, which are all used in the actual modeling of the current disease spread and impact predictions, and continue its development on your own. You can also choose to start this development from scratch, in which case you are free to reuse parts of these codes as you see fit.

https://medium.com/data-for-science/epidemic-modeling-101-or-why-your-covid19-exponential-fits-are-wrong-97aa50c55f8

https://www.kaggle.com/lisphilar/covid-19-data-with-sir-model

https://towardsdatascience.com/building-your-own-covid-19-epidemic-simple-model-using-python-e39788fbda55

https://towardsdatascience.com/visualise-covid-19-case-data-using-python-dash-and-plotly-e58feb34f70f

Your tasks in detail are to:

1) understand the modeling code you have chosen;
2) add new features and functionality to the code or the underlying mathematical models, thus increasing their scope and level of detail, as well as improving the results/insights they yield;
3) improve and expand the provided code as needed consistent with best practices for producing a deliverable software product. That means, your code should have adequate documentation and commenting; if you implement plotting functionality, the resulting images should be *meaningful*, *attractive*, and *"publication quality"*; if you produce text outputs, they should be nicely formatted; you may want to employ an integrated development environment (e.g., Eclipse, Spyder, PyCharm) and version control (Git and your own Github repository);
4) apply your code and the new models you have created to test different hypotheses and scenarios that may be of interest and attempt to validate these using available data (e.g., from Johns Hopkins University or the WHO);
5) plan and document all your development activities and modeling "experiments" using the Redmine project management software (I will make a dedicated Redmine instance accessible to you shortly);
6) every Friday, 5pm, (starting on 06/19) you have to submit a 1-page summary (pdf, 1 inch margins, 11pt Times New Roman, 1.5x line spacing) of your past week's activities and your plans for the next week to me; you should base your weekly report on your Redmine activities and the two should be consistent; at the last day of the challenge, you will submit a 10-page summary report of your project (which you may base on your weekly reports) as well as your code and documentation. The latter should highlight the changes you made and the new features/functionality you implemented.

You have about 9 weeks to work on this open-ended, free-form assignment, and **you are expected to deliver a corresponding quantity and quality of work.**

Your submission will be graded based on:

- the quantity and utility of new features/functionality, the quality of your implementation (including efficiency, documentation, readability, clarity, design, extensibility), and whether your program works correctly or not; as part of the evaluation, we will try to run your code, so please provide sufficient instructions for how to do so;

- the quality of the modeling "experiments" you have conducted as well as their results, derived insights, and discussion you provide;
- the quality of your reports;
- the way you utilized tools such as Redmine or Github.

Failure to follow all assignment instructions or work that is obviously nonsensical may result in penalties. Particularly original solutions may be rewarded. Precision is a virtue, so please avoid fluffed out and waffling reports. **Submissions that are disorganized, illegible, or otherwise unprofessionally presented may have receive penalties** at my discretion.

This assignment comes with an additional perk: At the end of this challenge, you can make your code publically available (e.g., on GitHub, Sourceforge, BitBucket) and use it as a work sample or part of your portfolio. This will come in handy should you ever decide to apply for a job that requires coding skills and proof thereof.

Please submit your weekly reports, as well as your final report and zipped software package back to me *via* email. Please use the following email subject line "EAS 560 DSC submission by <your name>" so that I can keep track of everyone's work. Please use your **@buffalo.edu account** when you communicate *via* email.

Tip 1: If you have trouble getting your code to run, try to google the error message you get. They will likely lead you to a useful stackoverflow page that will help you solve the problem.
Note: **You may collaborate, exchange ideas, and interact on technical questions with your classmates, but every student has to submit an individual solution that has been independently written up. No two submissions can be alike.** Collaborators should be listed and contributions should be credited. Failure to do so constitutes academic dishonesty and carries penalties as discussed below.

Breaches of academic integrity (e.g., plagiarism, cheating) are unacceptable and will result in a failing grade for the entire course. It's not right and it's not worth it! It is expected that you behave in an honorable and respectful way as you learn and share ideas. To summarize UB's policy on dishonesty: A student will not present, as his or her own, the work of another, or any work that has not been honestly performed; will not take any examination by improper means, and will not aid and abet another in any dishonesty. Please consult UB's Academic Integrity Policies at: https://catalog.buffalo.edu/policies/integrity.html