



Project 2

One of the ways that computer scientists push themselves to new levels of innovation is through challenges to solve important, "wicked" problems. Kaggle is a web-based company that hosts such challenges. It has recently published two challenges related to COVID-19. For this project you will participate in one of them (you do not have to actually submit an entry to the challenge---only to me---but you may enter the challenge if you wish).

Grading

- 10% Proper set up of Jupyter notebook that identifies the task you are working on.
- 20% Explanation of problem or subproblem you are solving
- 20% Explanation of your strategy: why do you think it will work.
- 20% Explanation of your code.
- 30% Quality of code / results

We encourage you to use the structure of the notebook to your advantage. I.e., write it so that explanations use snippets of your working code as examples.

Rules

1. You may work in groups of up to four people;
2. I'm OK with some communication between groups, but keep it public, say, on the mycourses discussion board or discord.
3. Submit to the Project 2 dropbox the same Jupyter notebook (.pynb file) you would submit to Kaggle. (You do not actually have to submit your solution to Kaggle, but you can, or you can even keep working on it after the the CS 635 deadline and submit it later; I'm happy to advise you on any such endeavors).

Procedure

1. Create a [Kaggle account here](#), if you don't have one yet.
2. Read the [basic instructions about kaggle](#).
3. [Go to this competition](#). Carefully read the instructions.
4. Choose a task. Make it clear in your notebook which one you choose.

5. Download the data, if you wish to work on your home machine.

CS Resources

1. weasley, granger, and lovegood are still at your service.
2. I do not believe there is much support for Jupyter. So if you want to develop here, you are probably best off developing in normal python3, then transferring the program to a notebook later (and you should plan this process out so that you don't introduce bugs during the process).
3. I have downloaded the data and made it available to you at
`/home/fac/cmh/covid`
4. [My kaggle notebook](#) is a good place to start if you are stuck.

Notes on department machines

You have been given access to the GPU machines weasley, granger, and lovegood, which should run your code much faster than non-GPU-based machines. One important concern is that tensorflow by default tries to reserve all GPUs it can find. The easiest way to ensure this doesn't happen is to set an environment variable. You can do this, e.g., in your python code via

```
import os
os.environ["CUDA_VISIBLE_DEVICES"]="0"
```

Where "0" is the name of a free GPU. Your program will then use only this GPU. You can also do this directly in your shell, of course. Use the shell command

```
nvidia-smi
```

to see a list of GPUs.