**Predict Covid Cases for 5 US States (CA, FL, GA, HI, NY)**

**The below instructions are based on the Johns Hopkins IDD Lab model, a minimal working example of which is documented here:** [**https://github.com/HopkinsIDD/COVID19\_Minimal/wiki/Getting-Started-in-a-US-Location**](https://github.com/HopkinsIDD/COVID19_Minimal/wiki/Getting-Started-in-a-US-Location)**.**

**However, there are some changes that have been made to get the model to work. Those are highlighted in yellow.**

**Note that the modified changes are intended to generate COVID19 case projections for the desired time range. Further analysis in the original JHU documentation are not tested/modified.**

**Checkout projects from GitHub**

The COVIDScenarioPipeline repo will be put inside the COVID19\_ repo. They are treated as independent, so push and pull code to them independently, and make sure they are both at the specific commit that you need.

1. Create a spatial repo from the COVID19\_Minimal template by navigating to [COVID19\_Minimal](https://github.com/HopkinsIDD/COVID19_Minimal), creating a fork, and then from inside that fork, clicking "Use this template". For this example, we'll do CA, FL, GA, HI, NY. We will name it COVID19\_5\_states and create the repository as vineetmp.
2. Checkout the spatial repo you just created. The following command can be entered in Git Bash installed on computer. In Git Bash, navigate to a preferred directory before running the below code. I used ‘cd "C:\Users\vpayyapp1\OneDrive - WestRock Co\projects\Covid Prediction\5\_states\_test"’ to naviagate to “…\5\_states\_test” directory. The double quotes are needed due to the space in the path. This directory must be already created before running the below line.

git clone https://github.com/vineetmp/COVID19\_5\_states.git

1. Checkout COVIDScenarioPipeline repo within the spatial repo. This might take a couple of minutes.

cd COVID19\_5\_states

git clone https://github.com/HopkinsIDD/COVIDScenarioPipeline.git

1. Git Bash may now be closed.

**Run Docker Image**

While working with this pipeline, we recommend editing files from your local machine and running scripts from the provided Docker container. The advantage of this container is that it already has all the packages installed, which takes some time.

If you want to install all the packages on your local computer, in COVIDScenarioPipline, see the R requirements in packages.R and local\_install.R, OS requirements in Dockerfile, and Python requirements in requirements.txt. Good luck.

1. Go to the Docker Hub website ([https://hub.docker.com](https://hub.docker.com/)) and create an account.
2. Start the Docker service on your computer. One way is to download, install, and run Docker Desktop. The Google can help you with this, potentially more than our support team can, but feel free to ask questions.
3. Open a terminal. First navigate to the “COVID\_5\_states” directory, so that docker image can be pulled to this directory. For example, I used: cd C:\Users\vpayyapp1\OneDrive - WestRock Co\projects\Covid Prediction\5\_states\_test\COVID19\_5\_states. For the docker commands in this section, if you run into permissions problems, you will need to put sudo in front.
4. Pull the docker image from hub.docker.com. You'll only have to do this the first time.

docker pull hopkinsidd/covidscenariopipeline:latest-dev

1. Run the docker container with your current directory mounted as /home/app/covidspcd

On Linux or Mac:

docker run -it --rm -v "$(pwd)":/home/app/covidsp hopkinsidd/covidscenariopipeline:latest-dev

On Windows:

docker run -it --rm -v %CD%:/home/app/covidsp hopkinsidd/covidscenariopipeline:latest-dev

\*\*You may need to change "%CD%" to your explicit directory\*\*

For example, I used: docker run -it --rm -v “C:\Users\vpayyapp1\OneDrive - WestRock Co\projects\Covid Prediction\5\_states\_test\COVID19\_5\_states”:/home/app/covidsp hopkinsidd/covidscenariopipeline:latest-dev

The quotation marks before and after the dir name are required if there are spaces in the dir name.

1. You are now in the docker container in /home/app. You may use the command “ls” or “dir” to print the list of items in the current directory. The directory you ran step #3 from is mapped to /home/app/covidsp in the container.

cd covidsp

You may use the command “ls” or “dir” to print the list of items in the current directory.

1. The Docker container needs some local R packages installed. Run this:

Rscript COVIDScenarioPipeline/local\_install.R

If there's a prompt Enter one or more numbers, or an empty line to skip updates:, just hit <Enter>.

1. The Docker container requires local Python packages to be installed. Run:

pip install -r COVIDScenarioPipeline/requirements.txt

The documentation suggests to use “python COVIDScenarioPipeline/setup.py install” – this won’t work.

**Edit config file**

**This section is not specified in the documentation, yet is required to get the next sections run without errors.**

There are several edits that need to be first made to “home/app/covidsp/config.yml”. Some changes are suggested in <https://github.com/HopkinsIDD/COVID19_Minimal/wiki/Config-File-Options>, however, not all of these changes will provide the required outcomes. Based on trial and error, the following changes are needed to the config file obtained from GitHub. Those that mandatorily need to be changed are marked in red. Others can be left at the value provided in the aforementioned documentation link.

this\_file\_is\_unedited: FALSE

Section spatial\_setup:

name: state5

end\_date: 2020-08-31 ## (to get August projections)

nsimulations: 2 ## (reduce from 15 to 2, for faster and more concise output)

base\_path: data/state5

modeled\_states:

- CA

- FL

- GA

- HI

- NY

setup\_name: state5

shapefile: data/state5/shp/counties\_2010\_state5.shp

shapefile\_name: data/state5/shp/counties\_2010\_state5.shp

Section importation:

census\_api\_key: <your census api key> ## replace “<your census api key>” with the census API key. To get the key, place a request here: <https://api.census.gov/data/key_signup.html>. The key will be emailed.

Additionally, manually create "data/state5/" subdirectory to store data files specific to this trial run for this test run (5 US states). Files such as "geodata.csv", "mobility.txt" will go into this directory. Note that “data/state5/shp/” is automatically created when the script runs.

Section seeding:

Comment the entire seeding section that uses “method: FolderDraw”, and uncomment the seeding section that uses “method: PoissonDistributed”. The documentation, however, instructs to use the latter only when the entire importation section is not needed/unused/commented.

lambda\_file: data/state5/seeding.csv

Section interventions:

Comment “-Scenario1” from settings. That is, use

scenarios:

- None

# - Scenario1

Instead of

scenarios:

- None

- Scenario1

This implies that no interventions are used. This can be edited later as needed. There is an option in this section to define time periods when intervention was/will be implemented.

As we are interested only in predicting case counts, no need to change anything in the “hospitalization” or “report” sections.

Note that the changes made so far are not final. In some of the sections below, further changes are needed.

**Generate geodata.csv and mobility.csv setup files**

For US locations:

1. Pull large data in repo. This may involve installing the git LFS tool and adding your Census API key.

cd COVIDScenarioPipeline

git lfs install

git lfs pull

cd ..

export CENSUS\_API\_KEY=<copy your key here>

1. From the spatial scenario directory, create long-form geodata and mobility files with:

Rscript COVIDScenarioPipeline/R/scripts/build\_US\_setup.R -c config.yml -p COVIDScenarioPipeline -w TRUE

1. Check the contents of data/state5/geodata.csv and data/state5/mobility.csv.

**Create seeding file**

Comment the whole importation section in config.yml. This is because the “FolderDraw” method doesn’t work (was tested), and the documentation instructs to comment the importation section if “PossonDistributed” is used. Note that this is a temporary change, and this will be reverted after seeding file has been generated. This temporary change is to prevent an error thrown when the below line is run.

1. Once the geodata.csv file exists, create a seeding file by running:

Rscript COVIDScenarioPipeline/R/scripts/create\_seeding.R -c config.yml

This step might take a few minutes to complete.

1. Check the contents of data/minimal/seeding.csv.

**Get shapefile data**

For this part to work, the "config$importation" section needs to be uncommented. Or else, will need to define census API key as a new section such as "config$api$census\_api\_key" and use this in the 2nd line ("tidycensus::census\_api\_key(key = config$api$census\_api\_key" instead of "tidycensus::census\_api\_key(key = config$importation$census\_api\_key")

This shapefile must have a GEOID column and be somewhere in the data directory.

To retrieve shapefile data for US states from the US census, run these commands in R:

To run in R, first open R prompt by typing “R” in the docker prompt.

config <- covidcommon::load\_config("config.yml")

tidycensus::census\_api\_key(key = config$importation$census\_api\_key)

covidImportation::get\_county\_pops(c('CA', 'FL', 'GA', 'HI', 'NY'), 'state5')

We may not be using the shapefiles, however, need to test if providing ‘state5’ as the region name for all 5 states together in the above line of code is okay or not.

**Edit the config**

Change on the following item in the config file.

|  |  |  |
| --- | --- | --- |
| spatial\_setup::popnodes | The name of the column in spatial\_setup::geodata file that specifies population | pop2010 |

All other items in this section may be skipped except for as we have already edited the config file for the previous sections to work.

The config file config.yml controls all of the options currently available. (See [this page for more details](https://github.com/HopkinsIDD/COVID19_Minimal/wiki/Config-File-Options).) This file has a tabbed outline structure. We will refer to keys using their full position in the outline. For example, we denote

spatial\_setup:

...

geodata: minimal

as spatial\_setup::geodata having a value of minimal

Some parts of the pipeline are only for US states. If modeling one or more US states, see Case 1. Otherwise, see Case 2.

For both cases, delete the line this\_file\_is\_unedited, or set it's value to FALSE. This is just to make sure people edit the config.yml.

There are several tags in the config like <Your State Postal Code>. Fill these tags in with state-specific information.

| **Config Item** | **Explanation of Value** | **Example** |
| --- | --- | --- |
| name | Give it a name | blue\_hawaii |
| spatial\_setup::modeled\_states | This should be a list of the states you want to simulate, with each state on it's own line preceded by - | modeled\_states: - MD - VA |
| spatial\_setup::popnodes | The name of the column in spatial\_setup::geodata file that specifies population | pop2010 |
| spatial\_setup::shapefile | A path to a shapefile relative to spatial\_setup::base\_path with a GEOID column. | The shapefile generated in the section above: shp/counties\_2010\_HI.shp |
| spatial\_setup::shapefile\_name | same as spatial\_setup::shapefile | shp/counties\_2010\_HI.shp |
| importation::census\_api\_key | An api\_key to use [tidycensus](https://walkerke.github.io/tidycensus/reference/census_api_key.html). | Please protect this key |

**Build and run**

Quit R by running “quit()” in the R prompt. No need to save the workspace image, as the required files have been saved already. Now we should be back on the docker prompt (covidsp$).

From covidsp$, run “python COVIDScenarioPipeline/simulate.py -c config.yml -s None -n 2”. This is described in “COVID19\_5\_states/COVIDScenarioPipeline/vignettes/how\_to.html”

The above step will save COVID case predictions as “.parquet” files in “COVID19\_5\_states\model\_output\” directory.

To convert these files into CSV, create “parquet\_to\_csv.py” script in “COVID19\_5\_states” directory. The contents of “parquet\_to\_csv.py” are:

### script to convert parquet files in model\_output directory into CSV  
  
import os  
import pandas as pd  
  
cwd = os.getcwd()  
model\_output\_path = os.path.join(cwd, 'model\_output', 'state5\_None')  
  
for filename in os.listdir(model\_output\_path):  
 if filename.endswith('.parquet'):  
 df = pd.read\_parquet(os.path.join(model\_output\_path, filename))  
 df.to\_csv(os.path.join(model\_output\_path, filename.replace('parquet', 'csv')))

Now from docker prompt covidsp$, run “python parquet\_to\_csv.py”. This will convert .parquet into .csv, and save it in the same respective directory.

That’s it, we now have COVID case projected counts of susceptible (S), exposed (E), infected (I), and recovered (R) for each county in the 5 selected states, for every day in the time period selected.

**None of the below steps are required, as we only want Covid case projections and those have been obtained in CSV files by the previous steps in this section (by running “simulate.py” and “parquet\_to\_csv.py”).**

**##################################################################################################################################**

1. From the spatial scenario directory, create the Makefile using the R script.

Rscript COVIDScenarioPipeline/R/scripts/make\_makefile.R -c config.yml

1. There is a bug in the Makefile that causes the report generation done by the make command in step 5 to fail. To get around this, steps 2-4.

Make a report directory and sub-directory because that is the .Rmd expects to be two directories below right now.

mkdir notebooks

cd notebooks

mkdir HI\_today

cd ../..

1. Make the initial R markdown (.Rmd) by running the following command.

Rscript -e 'rmarkdown::draft("notebooks/HI\_today/HI\_report.Rmd",template="state\_report",package="report.generation",edit=FALSE)'

1. Write the render line into compile\_Rmd.R. (compile\_Rmd.R is later used by fancy automation.)

echo 'rmarkdown::render("notebooks/HI\_today/HI\_report.Rmd", params=list(state\_usps="HI"))' >compile\_Rmd.R

1. Build and run.

If you see the following error during the make command, this is a known bug in the report generation step, which Rscript compile\_Rmd.R bypasses.

Execution halted

Makefile:11: recipe for target 'notebooks/blue\_hawaii\_20200520/blue\_hawaii\_20200520\_report.html' failed

make: \*\*\* [notebooks/blue\_hawaii\_20200520/blue\_hawaii\_20200520\_report.html] Error 1

Commands:

make clean # Removes all the generated files so they can be re-generated

make

Rscript compile\_Rmd.R # Getting around report generation bug

Tada! The data that you are looking for is in csv's in the folders model\_output and hospitalization.

The report you are looking for is in notebooks/HI\_today/HI\_report.html. View with a web browser.

**Exit the Docker container**

To exit without stopping the container so you can attach later, type Ctrl-p; Ctrl-q.

Otherwise, exit and remove the container with Ctrl-c.