Right now, two main type of data are used: sqlite3 database-dumped data and plain text input file. The database will be mostly fixed, or updated for every several month.  I have python functions to parse those files. For user uploaded data, it will be plain text file. We could create  temporary folders to store those data (~1-5 gb?), after the analysis, the folder will be deleted. The output visualization, we can look at those packages, heatmaply, plotly, ggnetwork, shiny DT table.

Liuyang

**From:** Tom Balmat <[thomas.balmat@duke.edu](mailto:thomas.balmat@duke.edu)>  
**Date:** Thursday, August 13, 2020 at 7:55 PM  
**To:** Liuyang Wang <[liuyang.wang@duke.edu](mailto:liuyang.wang@duke.edu)>  
**Subject:** RE: Python in R

What are the data elements that the python script uses?  Are they all in the H2P2 database?  The interactive component of our system runs on a Linux server and we can execute the python script there.  Existing H2P2 data and CPAG results for NHGRI and other sets will be queried from the database server.  User uploaded data will also be stored in the database for at least long enough to run the CPAG script.  One option is to query the database from within python for all data needed.  Another (less desirable) approach would be to query the database from within R, save the output in a format consistent with what the python script currently uses, then execute the script.  The output would be read by the R script for user review.

Tom

**From:** Liuyang Wang <[liuyang.wang@duke.edu](mailto:liuyang.wang@duke.edu)>   
**Sent:** Thursday, August 13, 2020 4:17 PM  
**To:** Tom Balmat <[thomas.balmat@duke.edu](mailto:thomas.balmat@duke.edu)>  
**Subject:** Re: Python in R

Hi Tom,

     The current version has lots of redundant data, it is 7.67 Gb.  The Dropbox link: <https://www.dropbox.com/sh/01qlr9q75bl0ppx/AAChLazvi0TvpVf0e_0Yjtw7a?dl=0> . If you have a Dropbox account, I can share the whole folder with you.

      I have a quick run shell file in the folder called “run\_cpag.sh”.

      I strong recommend to call shell command in R with system command, such as system(“python main.py cpagdb --threads 30 --subtype H2P2 --H2P2-Pcut 1e-6”).

    I did not have any test on other computer and platform, such as windows, please let me know if it works on your computer.

Liuyang

**From:** Tom Balmat <[thomas.balmat@duke.edu](mailto:thomas.balmat@duke.edu)>  
**Date:** Thursday, August 13, 2020 at 4:03 PM  
**To:** Liuyang Wang <[liuyang.wang@duke.edu](mailto:liuyang.wang@duke.edu)>  
**Subject:** RE: Python in R

Hi Liuyang,

Will you send me a sample of your command line syntax for generating a csv file?

Thanks,

Tom

**From:** Liuyang Wang <[liuyang.wang@duke.edu](mailto:liuyang.wang@duke.edu)>   
**Sent:** Tuesday, August 11, 2020 1:33 PM  
**To:** Tom Balmat <[thomas.balmat@duke.edu](mailto:thomas.balmat@duke.edu)>  
**Subject:** Re: Python in R

Hi Tom,

     I used this to write python code in Rstudio. I will try to use this to call python module.  If this does not work, we can R “system(cmd)” which I used a lot.

Liuyang

**From:** Tom Balmat <[thomas.balmat@duke.edu](mailto:thomas.balmat@duke.edu)>  
**Date:** Tuesday, August 11, 2020 at 12:32 PM  
**To:** Liuyang Wang <[liuyang.wang@duke.edu](mailto:liuyang.wang@duke.edu)>  
**Subject:** Python in R

Hi Liuyang,

There is a py\_run\_file() function in the reticulate package.  Will this work for executing the script from R?

<https://cran.r-project.org/web/packages/reticulate/vignettes/calling_python.html>

Tom