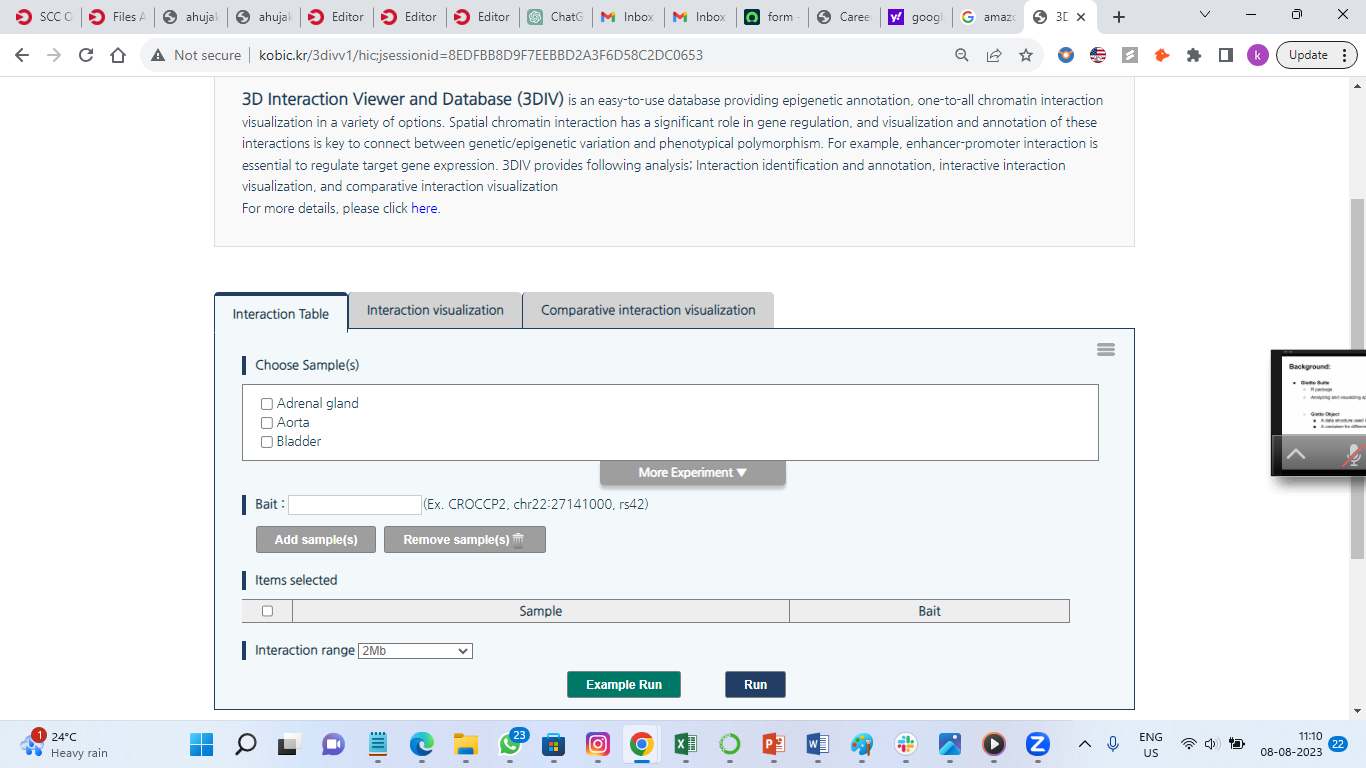
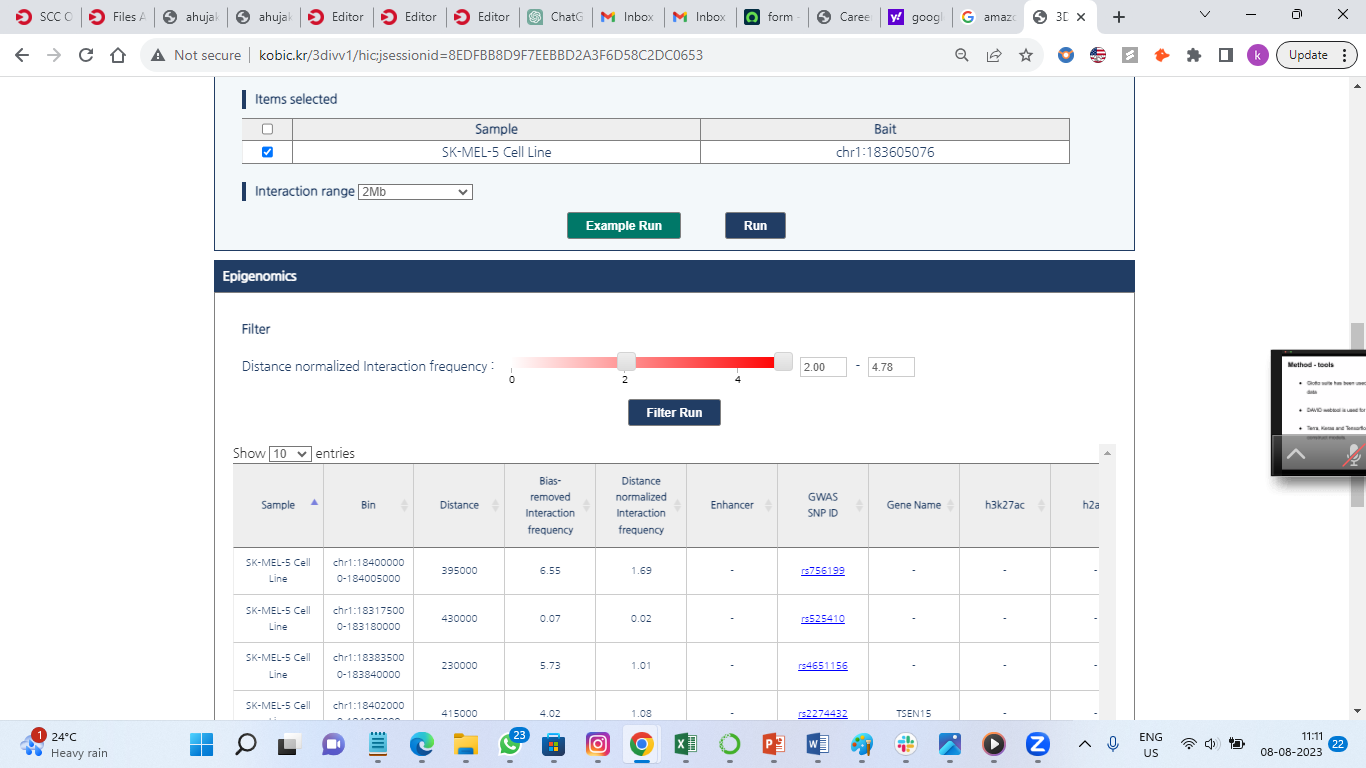
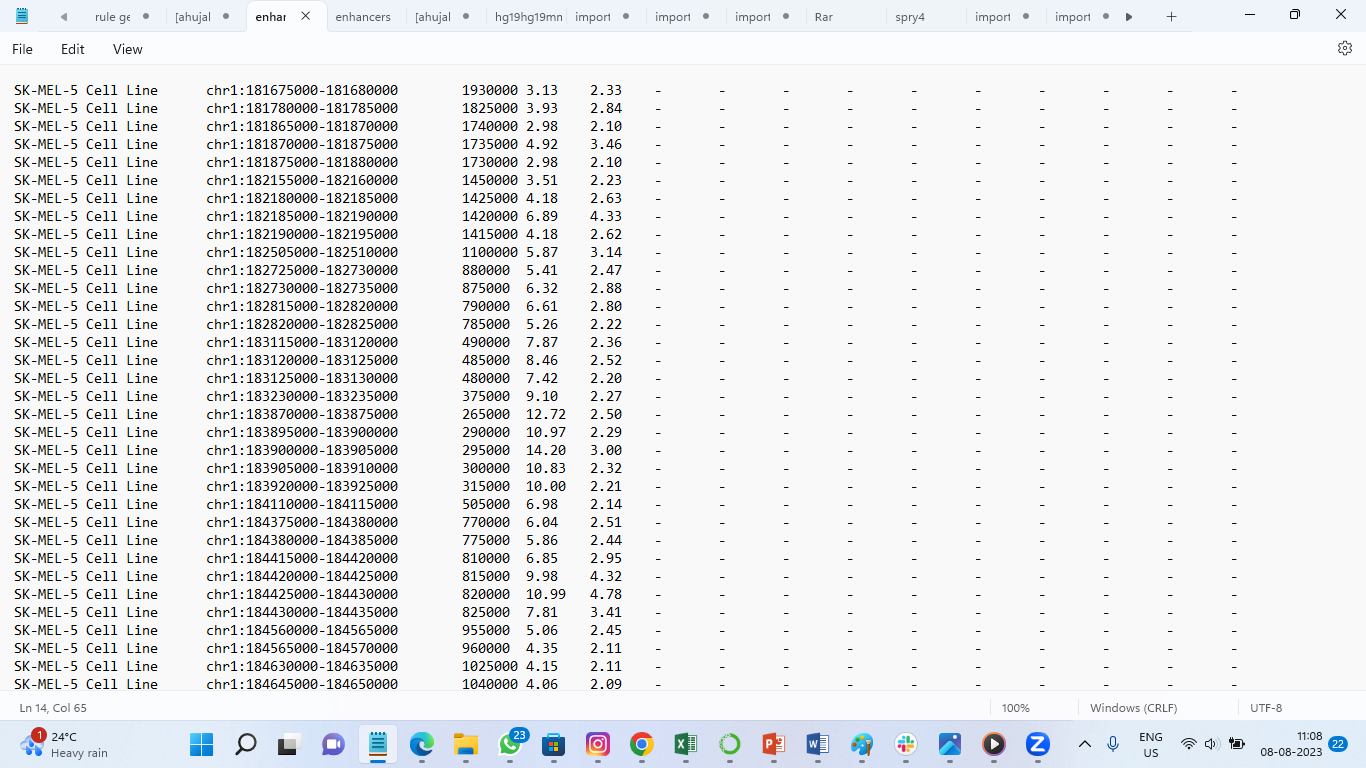
1. Open 3DIV software (http://www.3div.kr/) ,
2. Choose hg19, then HiC and then click on the interaction table
3. Select the cell line as SK MEL5Cell line (or the cell line you are interested in) by clicking on more experiment, and BAIT as whatever gene you are interested in (Example: ARPC5). Then click on Add sample and check the box and then click on the run button.



1. After then you will see an option for filter for distance normalized interaction frequency , set minimum as 2.00 and press filter and then run



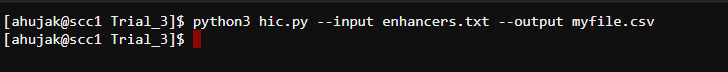
1. You will get the table. From there copy whole table (each and every column with the column headers) in one go and paste it in notepad or any text editor and save it as a text file (.txt)[ Example: enhancers.txt]



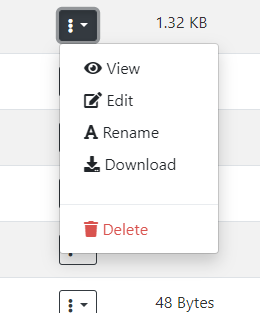
1. Login to SCC OnDemand (<https://scc-ondemand2.bu.edu/pun/sys/dashboard>)
2. Click on Files tab on top left.
3. From the dropdown list select /projectnb/langchip ( if you are not part of this project ask Dr Lang to add you in that project )

IF U WANT TO WORK IN THE ENVIRONMENT WHICH IS ALREADY STEP UP

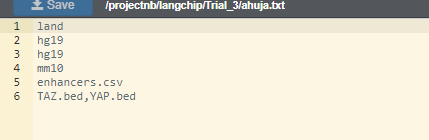
1. Go in directory Khushi in projectnb/langchip by clicking on it
2. Upload the text file that you have created from 3DIV using upload button.
3. Then click on “Open Terminal” ( choose scc1 from dropdown list)
4. Run : python3 hic.py --input <your input text file> --output <name of your output csv file> [Example: python3 hic.py --input enhancers.txt --output myfile.csv]( When you run this command , you will see nothing on the terminal but you can go back to the directory and check your file there)



1. Again, Go in directory Khushi in projectnb/langchip by clicking on it
2. Make a text file there. Click on new file which on top right and give the name say “example.txt”
3. Then you will see “example.txt” is made. Select it and click on Edit

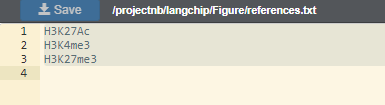


1. Write project name , hicdata( hg19/hg38), cut&run data (hg19/hg38), mouse\_genome (mm10/mm39), enhancer file name in csv format ( which you made in step 4), peak files in bed format.
2. Then click on the save button



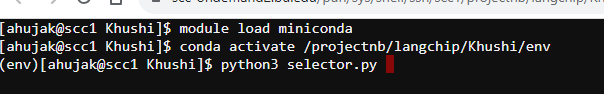
\*\*\* If you want to work with other transcriptional factor, do add them but make sure that peaks are in bed format (.bed) and you upload them in “Khushi” directory

1. Again make another text file named “references.txt” [Please note that the file name **cannot** be different here, it needs to be same]. Select it and click on Edit. As of yet it is already made if you want to make any changes you can edit by clicking on edit.
2. Type the names of the canonical histone markers. See the example below



\*\*\* If you want to add some other markers here. Add by editing “references.txt”. Again don’t change the name of the file. Also make sure that you upload the bigwig file of that histone marker (.bw) in “Khushi” directory.

1. Go again to /projectnb/langchip/Khushi and then on the top right you will see “ Open Terminal”
2. Click on “ Open Terminal”
3. You will see a linux environment being initiated
4. Type : module load miniconda and enter ( you will nothing happening on terminal)
5. Type : conda activate /projectnb/langchip/Khushi/env and enter ( This new conda environment being initiated will have both snakemake and pygenometrack activated in it)( you will nothing happening on terminal)
6. Type : Type : python3 selector.py and enter
7. It will ask you to give the path of the text file. Type : example.txt ( or the name you have given to your text file )
8. Then Enter
9. DON’T SHUTDOWN YOUR LAPTOP/COMPUTER OR DON’T LET IT GO ON SLEEP ( otherwise terminal session will end)
10. All the output files will be formed in the directory which is same as your project name

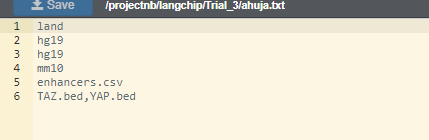


IF U WANT TO SETUP YOUR OWN ENVIRONMENT BEFORE STARTING

1. Go in directory Khushi in projectnb/langchip by clicking on it
2. From there download files named as :

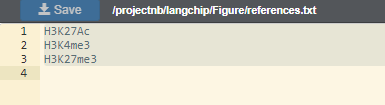
* environment.py
* hg19hg19mm10.snake
* hg19hg19mm39.snake
* hg19hg38mm10.snake
* hg19hg38mm39.snake
* hg38hg38mm10.snake
* hg38hg38mm39.snake
* hg38hg19mm10.snake
* hg38hg19mm39.snake
* khushi.py
* rty.py
* qwe.py
* selector.py
* hic.py
* TAZ.bed and YAP.bed ( if you want to use these transcription factor peaks)
* H3K27Ac.bw , H3K27me3.bw and H3K4me3.bw ( if you want to use these as histone canonical markers)

1. After downloading these files go back to projectnb/langchip
2. Click on New Dir somewhere on top right
3. Give some name to the directory ( Example : Summer)
4. This will make a new directory. Click on the new directory formed.
5. Using Upload button on right top corner, upload all the files that we have downloaded.
6. Upload the text file that you have created from 3DIV using upload button.
7. Then click on “Open Terminal”
8. Run : python3 hic.py --input <your input text file> --output <name of your output csv file> [Example: python3 hic.py --input enhancers.txt --output myfile.csv]
9. Then go again to your directory on demand user interface
10. Make a text file there. Click on new file which on top right and give the name say “example.txt”
11. Then you can see “example.txt” is formed. Select it and click on Edit.
12. Write project name , hicdata( hg19/hg38), cut&run data (hg19/hg38), mouse\_genome (mm10/mm39), enhancer file name in csv format (which you made in step 10), peak files in bed format.
13. Then click on the save button



\*\*\* If you want to work with other transcriptional factor, do add them but make sure that peaks are in bed format (.bed) and you upload them in new directory you created.

1. Again make another text file named “references.txt” [Please note that the file name **cannot** be different here, it needs to be same]. Select it and click on Edit.
2. Type the names of the canonical histone markers. See the example below



\*\*\* If you want to add some other markers here. Add by editing “references.txt”. Again don’t change the name of the file. Also make sure that you upload the bigwig file of that histone marker (.bw) in the new directory you made.

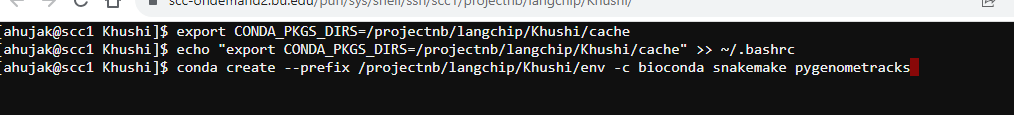
1. Go again to /projectnb/langchip/Your Directory Name( example Summer) and then on the top right you will see “ Open Terminal”
2. Click on “ Open Terminal”
3. You will see a linux environment being initiated
4. Type : module load miniconda and enter
5. Type : export CONDA\_PKGS\_DIRS=/projectnb/langchip/<Your directory name>/cache and enter

[Example: export CONDA\_PKGS\_DIRS=/projectnb/langchip/Summer/cache]

1. Type : echo "export CONDA\_PKGS\_DIRS=/projectnb/langchip/<your directory name>/cache" >> ~/.bashrc and enter

[Example: echo "export CONDA\_PKGS\_DIRS=/projectnb/langchip/Summer /cache" >> ~/.bashrc]

1. Type : conda create --prefix /projectnb/langchip/<Your directory name>/env -c bioconda snakemake pygenometracks and enter [ This step takes a while as you are creating a new environment with snakemake and pygenometrack, while running in between it will ask for proceed type y and enter ]
2. Type : conda activate /projectnb/langchip/Your directory name/env and enter
3. Type : python3 environment.py and enter
4. Type : python3 selector.py and enter
5. It will ask you to give the path of the text file. Type : example.txt ( or the name you have given to your text file )
6. Then Enter
7. All the output files will be formed in the directory which is same as your project name



The Main output files:

* enhancer\_sequences.csv - File containing the sequences of the enhancer coordinates
* output\_blast\_with\_headers.csv – File containg the blast output
* updated\_updated\_filtered\_output\_blast.csv – Updated and filtered blast output
* blast.bed – coordinates of enhancers that showed BLAST positive results
* peakname\_intersected\_peaks.csv- Mapping of TF with enhancer coordinates
* peakname\_annotated\_intersected\_peaks.bed – Annotation of mapping peaks
* combined\_enhancers.csv – gives the all possible mapped peaks of enhancer in a sophisticated manners
* figures ( filename will be coordinates of the enhancer)
* hg38/hg19\_coordinates.bed ( only formed if the reference genome of hic and cut&run is different)-changed coordinates depending upon the reference genome