Finding Charged Pions and Calculating Their Cross Section

Editing the XML file:

ssh into ACCRE
shell into singularity
cd into myX-SCAPE/X-SCAPE/build
emacs ../config/jetscape_user.xml

Edit to have this info:
<nEvents> 100000 </nEvents>
Before <JetScapeWriterAscii> on </JetScapeWriterAscii> write:
<outputFilename> pion_pp_pTHat5_10_out </outputFilename>
<JetScapeWriterFinalStateHadronsAscii> on </JetScapeWriterFinalStateHadronsAscii>

Then, in <pTHatMin> change the value to 5, and in <pTHatMax> change value to 10, and in <eCM> change the value to 200

(Make sure that you change your output filename to be descriptive enough so that you can understand what exactly is in the file after it is output)

Then Save and exit emacs (Control-x Control-c)

Running the XML file:

Then (in same directory), run:
./runJetscape ../config/jetscape_user.xml

It will run for 100000 events, so it may take a while
When it's done, make an analysis directory
mkdir Analysis

Moving to Analysis:

Move all of the output files to the Analysis directory by typing the command
mv pion_pp_pTHat5_15_out* Analysis/
cd Analysis
emacs pionpluscross_secv4.cpp (creates a new file)

(This file will be used to find negative and positive pions and combine their cross section)

On my GitHub, search for a file of the same name and download the file or copy the contents of that file onto your new file.
Look through the file for int main(){ and look for std::string filename
Edit the filename to be the same as the one that you had just moved into your analysis directory
Save and exit emacs (Control-x Control-c)

Make the cpp file using this command:
g++ -o pionpluscross_secv4 pionpluscross_secv4.cpp -lgsl -lgslcblas -lm
(make sure you have GSL installed, if you don't, then just run the command:
apt-get install libgsl-dev
And it should download and install, then try to make the cpp file again)

Once it makes, and there are no errors, run this command:
./pionpluscross_secv4

And it should print out values in these columns:
pT Cross Section Cross Section Error

Then copy and paste the values into a text file on your laptop

If any of them have
pT 0 -nan

That is fine, still copy and paste the entire thing, but change the -nan values to 0

Plot Using Jupyter Notebook:

Now, go into my GitHub again, and download or copy the Jupyter Notebook file

Once you have that text file saved somewhere, preferably in the same directory as the Jupyter Notebook, then you can open Jupyter and edit the python code to load your text file by changing the filename in np.loadtxt, and then run the cell under Charged Pion Cross Section

Finding Neutral Pions and Calculating Their Cross Section

Editing the XML file:

cd back into the build directory (if you are in Analysis, then the command is cd ../)
emacs ../config/jetscape_user.xml

Rename the output filename to include that these are neutral pions, so it could be pi0_pp_pTHat…

In the <JetHadronization> section, add these lines below <name> colorless </name>:
<LinesToRead>
    111:mayDecay = off
</LinesToRead>
(This ensures that the pi0 particle will not decay into two photons, and therefore we can calculate its cross section)
Save and exit emacs

Running the XML file:

Following the same steps as before:
./runJetscape ../config/jetscape_user.xml

Moving to Analysis:

Wait for the run to finish, and then move all output files to Analysis directory
cd Analysis

emacs pioncross_secv1.cpp (creates new file, or you can download mine from the GitHub)
Copy the information from the code with the same name on my GitHub
Edit the filename under the int main(){ function again to match yours

Make the file using this command:
g++ -o pionpluscross_secv4 pionpluscross_secv4.cpp -lgsl -lgslcblas -lm

Then run the file:
./pioncross_secv1

The output information will be the same as before, with columns of
pT Cross Section Cross Section Error

Copy and Paste these values down on a text file on your local machine (laptop), and save in the same directory that you have your Jupyter Notebook in

Plot Using Jupyter Notebook:

Open the Jupyter Notebook and edit the cell under Neutral Pion Cross Section to contain the name of your file after the np.loadtxt, then run the cell

PYTHIA Analysis:

Getting Started:
PYTHIA is already downloaded and installed in the singularity shell (located in myX-SCAPE/X-SCAPE/external_packages/pythia8309)
So, go to the PYTHIA directory and cd into examples
Make a new file:

emacs main999.cc

Then copy the contents of the file I have under the name main100.cc
Then save and exit emacs

Then, use the make command:
```
 g++ -o main999 main999.cc -lgsl -lgslcblas -lm -I../include -O2 -std=c++11 -pedantic -W -Wall -Wshadow -fPIC -pthread  -L../lib -Wl,-rpath,../lib -lpythia8
```

Then run the file:
```
./main999
```

It will run for a while, and then you will get output that is structured the same as the XSCAPE output, so it can be copied and put into a text file that can then be used to plot in python