

# Mega Tiler v1.0

Geospatial Laser Applications and Measurements  
Applied Research Laboratories  
University of Texas at Austin

# ICESat-2 Mega Tiler

- The ICESat-2 Mega Tiler is a geospatial analysis tool that allows users to load, process, analyze, plot, and export ICESat-2 ATL03 data that has been stored in LAZ 1.4 mega tiles
- Users can analyze a region of the world by specifying a latitude/longitude bounding box
- Regions can be analyzed by individual 1° x 1° tiles (up to a maximum of 5° x 5° tiles)
- All ICESat-2 data within the specified latitude/longitude bounding box can be exported to:
  - .gzip file
  - .csv file
  - .laz 1.4 file
  - .kml file
  - .tif file
- The ICESat-2 Mega Tiler is a command line tool built in Python (it requires no .exe or .dll files)

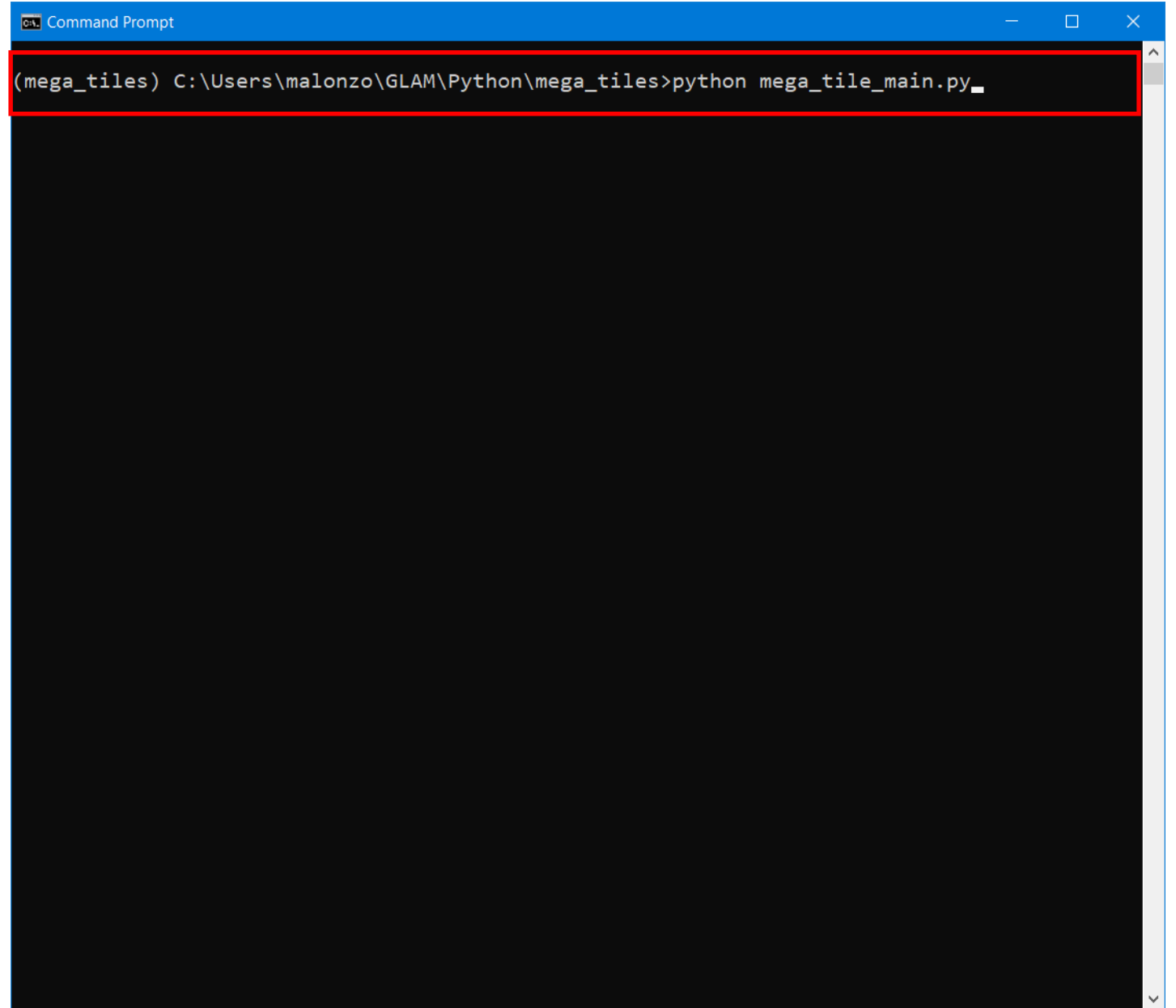
# ICESat-2 Mega Tiler

- The code can be run in the Windows Command shell by first activating the Python Conda environment:

```
>> activate mega_tiles
```

and then typing:

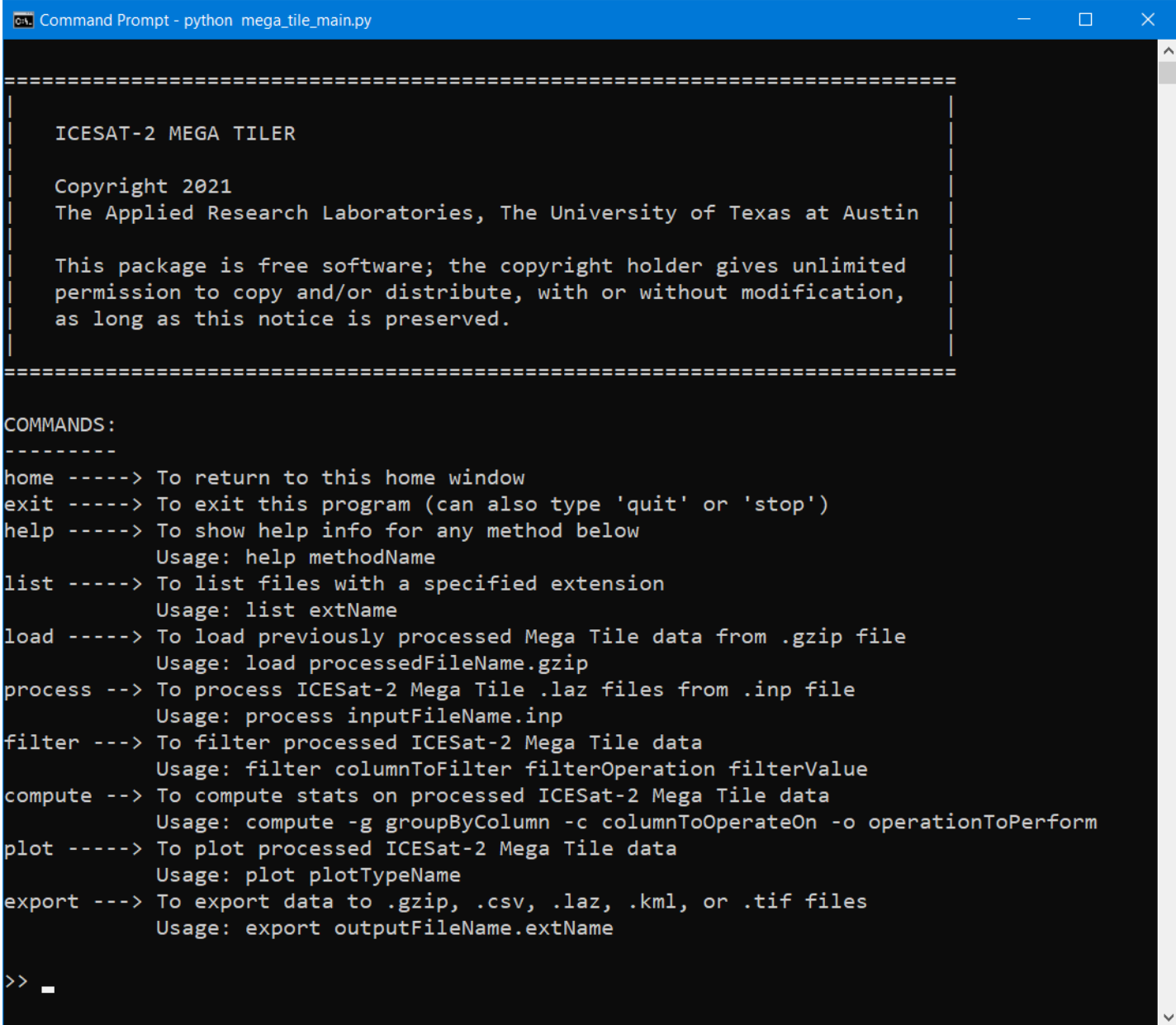
```
>> python mega_tile_main.py
```

A screenshot of a Windows Command Prompt window. The title bar is blue and says "Command Prompt". The command prompt shows the current directory as "C:\Users\malonzo\GLAM\Python\mega\_tiles" and the command "python mega\_tile\_main.py" is being entered. The prompt is "(mega\_tiles)". The command prompt is highlighted with a red border.

```
Command Prompt
(mega_tiles) C:\Users\malonzo\GLAM\Python\mega_tiles>python mega_tile_main.py
```

# ICESat-2 Mega Tiler

- This is the main window which contains all of the user commands



```
Command Prompt - python mega_tile_main.py

=====

ICESAT-2 MEGA TILER

Copyright 2021
The Applied Research Laboratories, The University of Texas at Austin

This package is free software; the copyright holder gives unlimited
permission to copy and/or distribute, with or without modification,
as long as this notice is preserved.

=====

COMMANDS:
-----
home -----> To return to this home window
exit -----> To exit this program (can also type 'quit' or 'stop')
help -----> To show help info for any method below
               Usage: help methodName
list -----> To list files with a specified extension
               Usage: list extName
load -----> To load previously processed Mega Tile data from .gzip file
               Usage: load processedFileName.gzip
process --> To process ICESat-2 Mega Tile .laz files from .inp file
               Usage: process inputFileName.inp
filter ---> To filter processed ICESat-2 Mega Tile data
               Usage: filter columnToFilter filterOperation filterValue
compute --> To compute stats on processed ICESat-2 Mega Tile data
               Usage: compute -g groupByColumn -c columnToOperateOn -o operationToPerform
plot -----> To plot processed ICESat-2 Mega Tile data
               Usage: plot plotTypeName
export ---> To export data to .gzip, .csv, .laz, .kml, or .tif files
               Usage: export outputFileName.extName

>> _
```

# ICESat-2 Mega Tiler

- Use the `process` command to process mega tile data
- Type `help process` (or any other command) for more info on how to use that command

```
Command Prompt - python mega_tile_main.py

Copyright 2021
The Applied Research Laboratories, The University of Texas at Austin

This package is free software; the copyright holder gives unlimited
permission to copy and/or distribute, with or without modification,
as long as this notice is preserved.

=====

COMMANDS:
-----
home -----> To return to this home window
exit -----> To exit this program (can also type 'quit' or 'stop')
help -----> To show help info for any method below
               Usage: help methodName
list -----> To list files with a specified extension
               Usage: list extName
load -----> To load previously processed Mega Tile data from .gzip file
               Usage: load processedFileName.gzip
process --> To process ICESat-2 Mega Tile .laz files from .inp file
               Usage: process inputFileName.inp
filter ---> To filter processed ICESat-2 Mega Tile data
               Usage: filter columnToFilter filterOperation filterValue
compute --> To compute stats on processed ICESat-2 Mega Tile data
               Usage: compute -g groupByColumn -c columnToOperateOn -o operationToPerform
plot -----> To plot processed ICESat-2 Mega Tile data
               Usage: plot plotTypeName
export ---> To export data to .gzip, .csv, .laz, .kml, or .tif files
               Usage: export outputFileName.extName

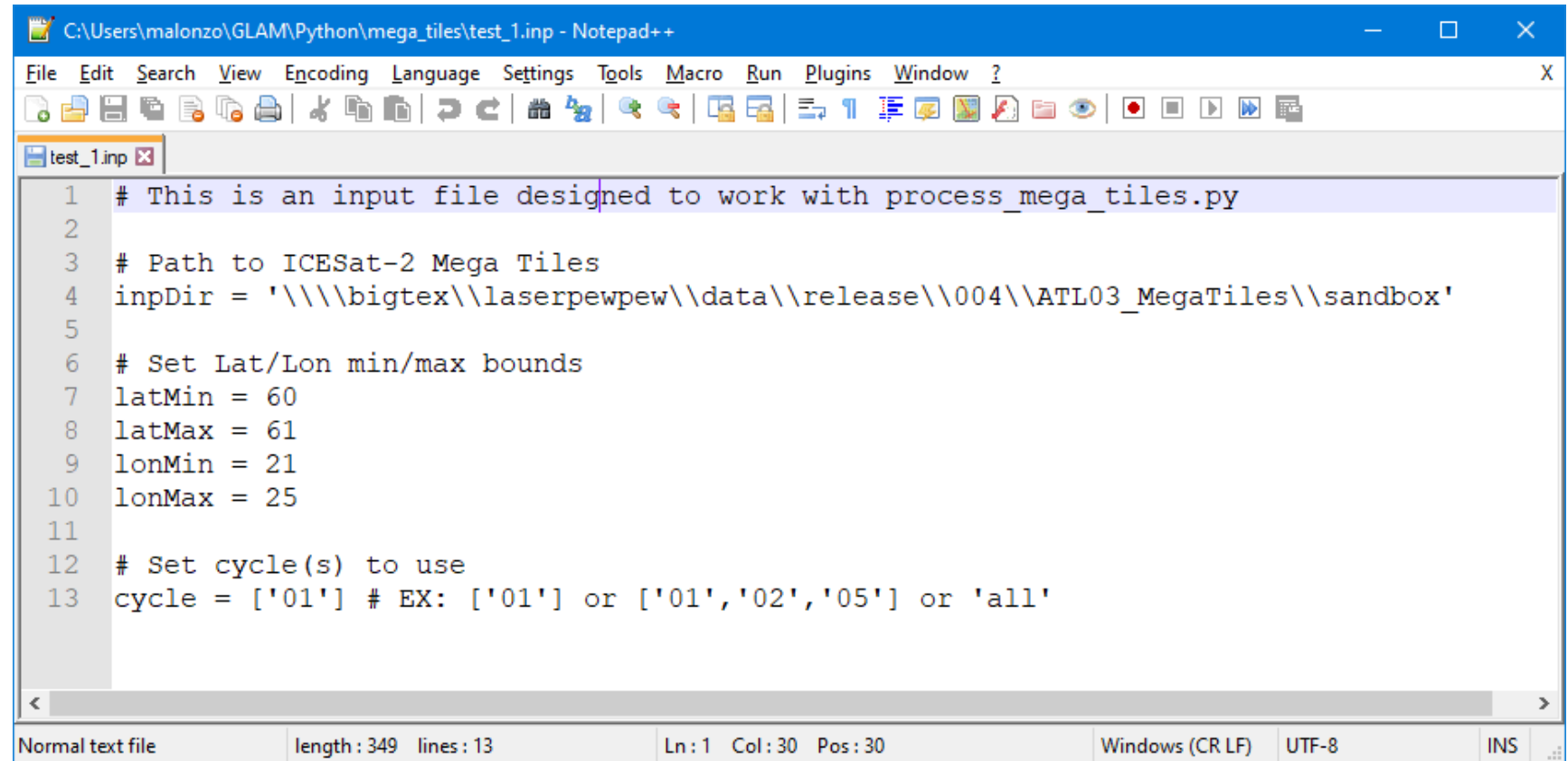
>> help process

process --> To process ICESat-2 Mega Tile .laz files from .inp file
               Usage: process inputFileName.inp
               Example: process test.inp

>> _
```

# ICESat-2 Mega Tiler

- The process command reads an input (.inp) file with lat/lon bounds and cycle number for the mega tile data

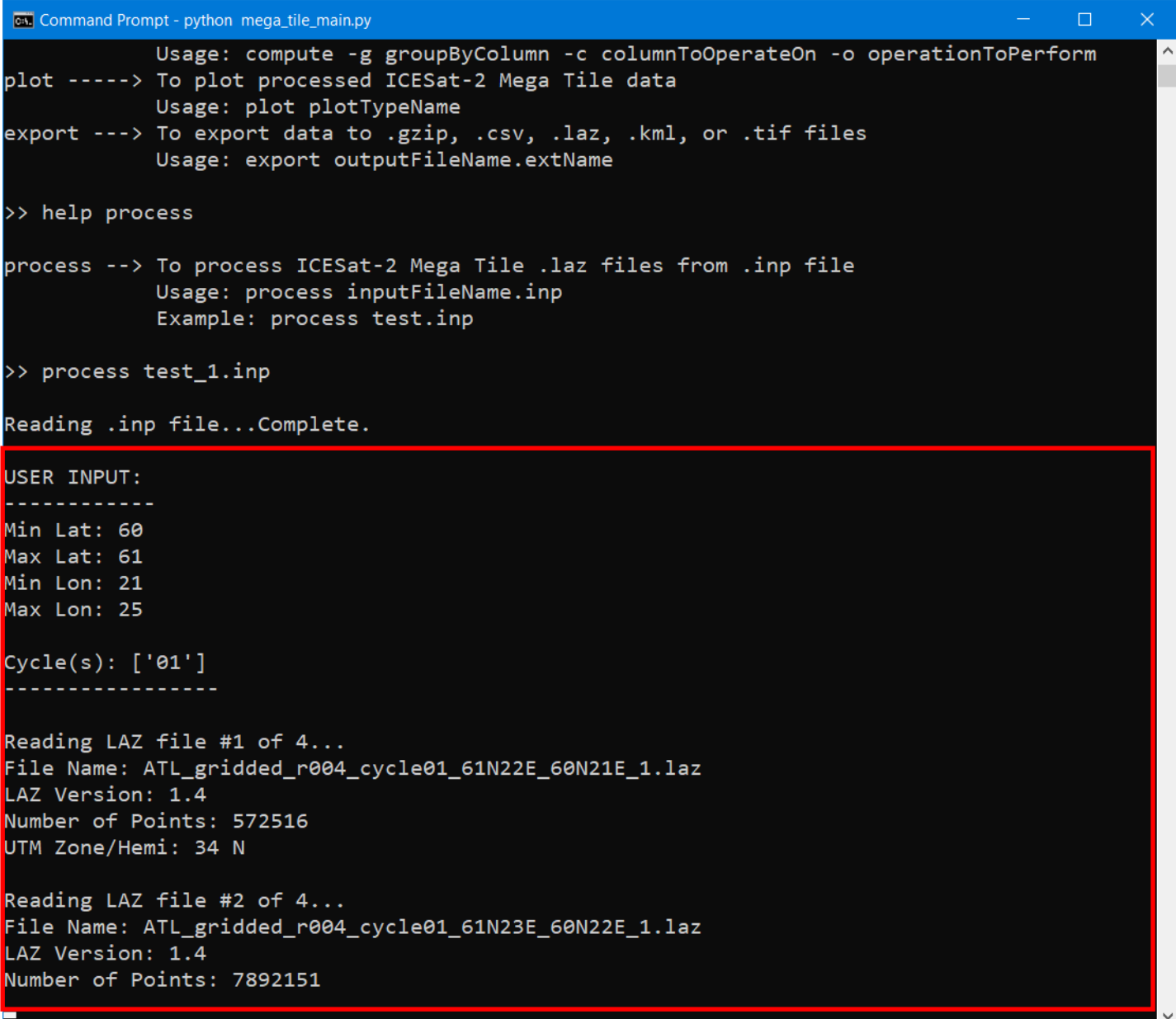


```
C:\Users\malonzo\GLAM\Python\mega_tiles\test_1.inp - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
test_1.inp
1 # This is an input file designed to work with process_mega_tiles.py
2
3 # Path to ICESat-2 Mega Tiles
4 inpDir = '\\\\bigtex\\laserpewpew\\data\\release\\004\\ATL03_MegaTiles\\sandbox'
5
6 # Set Lat/Lon min/max bounds
7 latMin = 60
8 latMax = 61
9 lonMin = 21
10 lonMax = 25
11
12 # Set cycle(s) to use
13 cycle = ['01'] # EX: ['01'] or ['01','02','05'] or 'all'
Normal text file length: 349 lines: 13 Ln: 1 Col: 30 Pos: 30 Windows (CR LF) UTF-8 INS
```

# ICESat-2 Mega Tiler

- Messages are sent to the command window while data is processing
- An example of processing an input file named test\_1.inp:

```
>> process test_1.inp
```



```
Command Prompt - python mega_tile_main.py

Usage: compute -g groupByColumn -c columnToOperateOn -o operationToPerform
plot ----> To plot processed ICESat-2 Mega Tile data
Usage: plot plotTypeName
export ---> To export data to .gzip, .csv, .laz, .kml, or .tif files
Usage: export outputFileName.extName

>> help process

process --> To process ICESat-2 Mega Tile .laz files from .inp file
Usage: process inputFileName.inp
Example: process test.inp

>> process test_1.inp

Reading .inp file...Complete.

USER INPUT:
-----
Min Lat: 60
Max Lat: 61
Min Lon: 21
Max Lon: 25

Cycle(s): ['01']
-----

Reading LAZ file #1 of 4...
File Name: ATL_gridded_r004_cycle01_61N22E_60N21E_1.laz
LAZ Version: 1.4
Number of Points: 572516
UTM Zone/Hemi: 34 N

Reading LAZ file #2 of 4...
File Name: ATL_gridded_r004_cycle01_61N23E_60N22E_1.laz
LAZ Version: 1.4
Number of Points: 7892151
```

# ICESat-2 Mega Tiler

- A snapshot of the processed data after the input file has been processed
- Some default stats are displayed for each field in the .laz file

```
Command Prompt - python mega_tile_main.py

[9930596 rows x 32 columns]

Dataframe Stats:
-----

```

	min	max	mean	std
lon	21.81876	25.00000	22.55297	0.70362
lat	60.00000	61.00000	60.49191	0.28730
utme	545237.65777	722614.65850	585400.89254	38985.34767
utm_n	6651846.21172	6765973.97678	6707411.32824	31709.99371
z_hae	-64.46783	257.60364	55.62802	33.15887
z_msl	-64.46783	257.60364	55.19500	33.57765
intensity	0.00000	0.00000	0.00000	0.00000
return_number	0.00000	0.00000	0.00000	0.00000
number_of_returns	0.00000	0.00000	0.00000	0.00000
scanner_channel	0.00000	0.00000	0.00000	0.00000
classification	1.00000	3.00000	1.32758	0.58482
user_data	0.00000	0.00000	0.00000	0.00000
scan_angle_rank	0.00000	0.00000	0.00000	0.00000
point_source_id	0.00000	0.00000	0.00000	0.00000
gps_time	0.00000	0.00000	0.00000	0.00000
geoid	0.00000	29.00000	0.43302	3.06021
beam_number	1.00000	6.00000	3.21321	1.72287
snow_flag	0.00000	2.00000	1.36881	0.59238
land_flag	0.00000	16.00000	3.30888	3.62065
yr doy	18287.00000	18360.00000	18326.15015	20.31193
rel_canopy_height	0.00000	255.00000	11.78776	47.76650
solar_elev	0.00000	235.00000	115.14034	101.97889
signal_conf	0.00000	255.00000	3.72202	1.18929

```
Processing complete.

>>
```



# ICESat-2 Mega Tiler

- Users can filter data

- For instance:

```
>> filter classification == 1
```

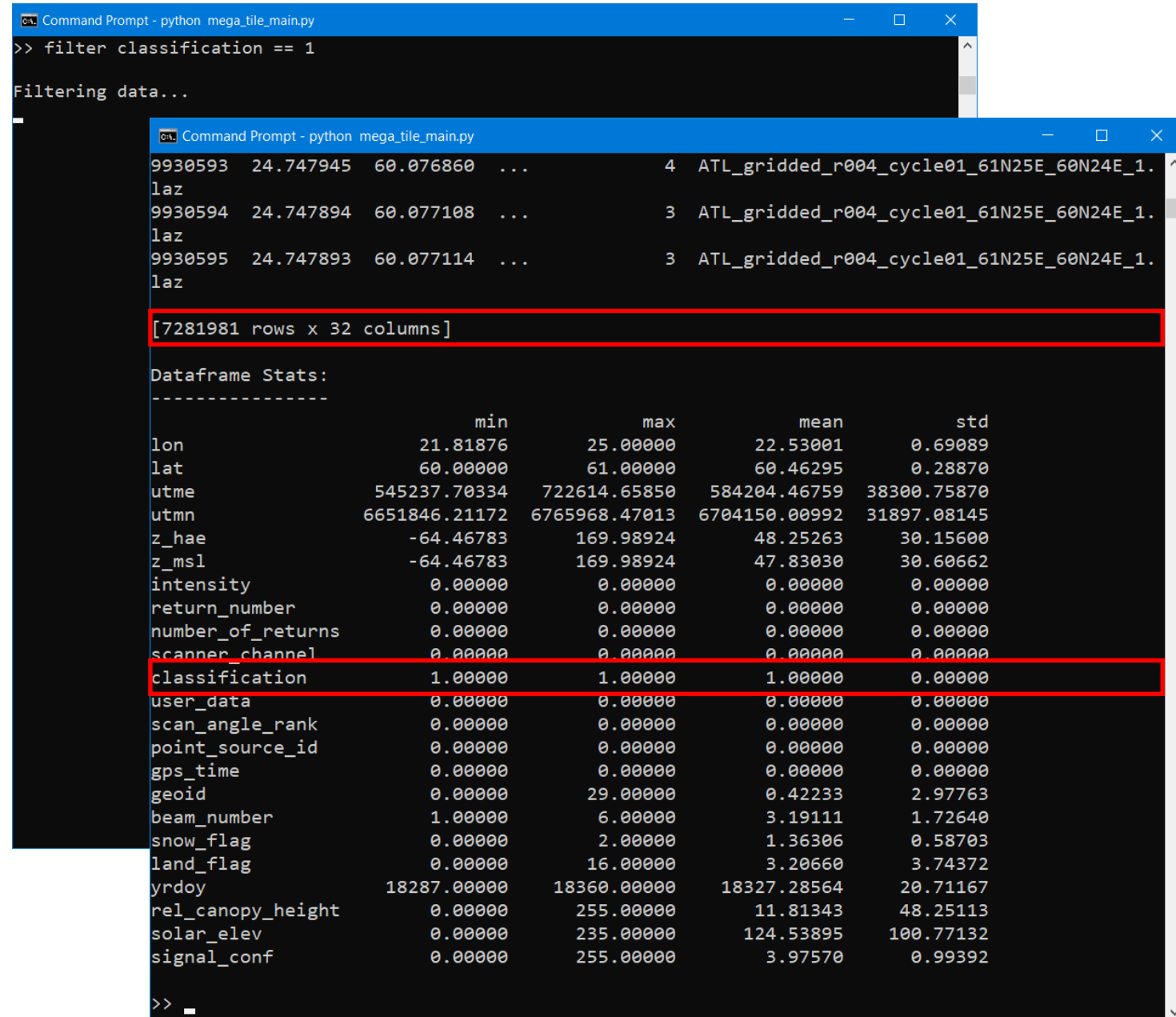
- Unfiltered dataframe shape was:

- 9,930,596 x 32

- Filtered dataframe shape is:

- 7,281,981, x 32

- Dataframes cannot be unfiltered after issuing the `filter` command



```
Command Prompt - python mega_tile_main.py
>> filter classification == 1
Filtering data...

9930593  24.747945  60.076860  ...      4  ATL_gridded_r004_cycle01_61N25E_60N24E_1.
laz
9930594  24.747894  60.077108  ...      3  ATL_gridded_r004_cycle01_61N25E_60N24E_1.
laz
9930595  24.747893  60.077114  ...      3  ATL_gridded_r004_cycle01_61N25E_60N24E_1.
laz

[7281981 rows x 32 columns]

Dataframe Stats:
-----

```

	min	max	mean	std
lon	21.81876	25.00000	22.53001	0.69089
lat	60.00000	61.00000	60.46295	0.28870
utme	545237.70334	722614.65850	584204.46759	38300.75870
utm_n	6651846.21172	6765968.47013	6704150.00992	31897.08145
z_hae	-64.46783	169.98924	48.25263	30.15600
z_msl	-64.46783	169.98924	47.83030	30.60662
intensity	0.00000	0.00000	0.00000	0.00000
return_number	0.00000	0.00000	0.00000	0.00000
number_of_returns	0.00000	0.00000	0.00000	0.00000
scanner_channel	0.00000	0.00000	0.00000	0.00000
classification	1.00000	1.00000	1.00000	0.00000
user_data	0.00000	0.00000	0.00000	0.00000
scan_angle_rank	0.00000	0.00000	0.00000	0.00000
point_source_id	0.00000	0.00000	0.00000	0.00000
gps_time	0.00000	0.00000	0.00000	0.00000
geoid	0.00000	29.00000	0.42233	2.97763
beam_number	1.00000	6.00000	3.19111	1.72640
snow_flag	0.00000	2.00000	1.36306	0.58703
land_flag	0.00000	16.00000	3.20660	3.74372
yrday	18287.00000	18360.00000	18327.28564	20.71167
rel_canopy_height	0.00000	255.00000	11.81343	48.25113
solar_elev	0.00000	235.00000	124.53895	100.77132
signal_conf	0.00000	255.00000	3.97570	0.99392

```
>> _
```

# ICESat-2 Mega Tiler

- Users can compute stats on the data

- For instance:

```
>> compute -g classification
        -c ['z_hae']
        -o mean
```

- This command will group all data by classification and compute the mean Z HAE (height above ellipsoid) for each classification

```
Command Prompt - python mega_tile_main.py
intensity          0.00000      0.00000      0.00000      0.00000
return_number      0.00000      0.00000      0.00000      0.00000
number_of_returns  0.00000      0.00000      0.00000      0.00000
scanner_channel    0.00000      0.00000      0.00000      0.00000
classification      1.00000      3.00000      1.32758      0.58482
user_data          0.00000      0.00000      0.00000      0.00000
scan_angle_rank    0.00000      0.00000      0.00000      0.00000
point_source_id    0.00000      0.00000      0.00000      0.00000
gps_time           0.00000      0.00000      0.00000      0.00000
geoid              0.00000      29.00000     0.43302      3.06021
beam_number        1.00000      6.00000      3.21321      1.72287
snow_flag          0.00000      2.00000      1.36881      0.59238
land_flag          0.00000      16.00000     3.30888      3.62065
yrday              18287.00000    18360.00000   18326.15015    20.31193
rel_canopy_height  0.00000      255.00000    11.78776      47.76650
solar_elev         0.00000      235.00000    115.14034     101.97889
signal_conf        0.00000      255.00000     3.72202      1.18929

Processing complete.

>> compute -g classification -c ['z_hae'] -o mean

Group By: classification
Column to Operate on: ['z_hae']
Operation: mean

Computed Output:
-----

          z_hae
classification
1          48.252631
2          74.488727
3          80.696774

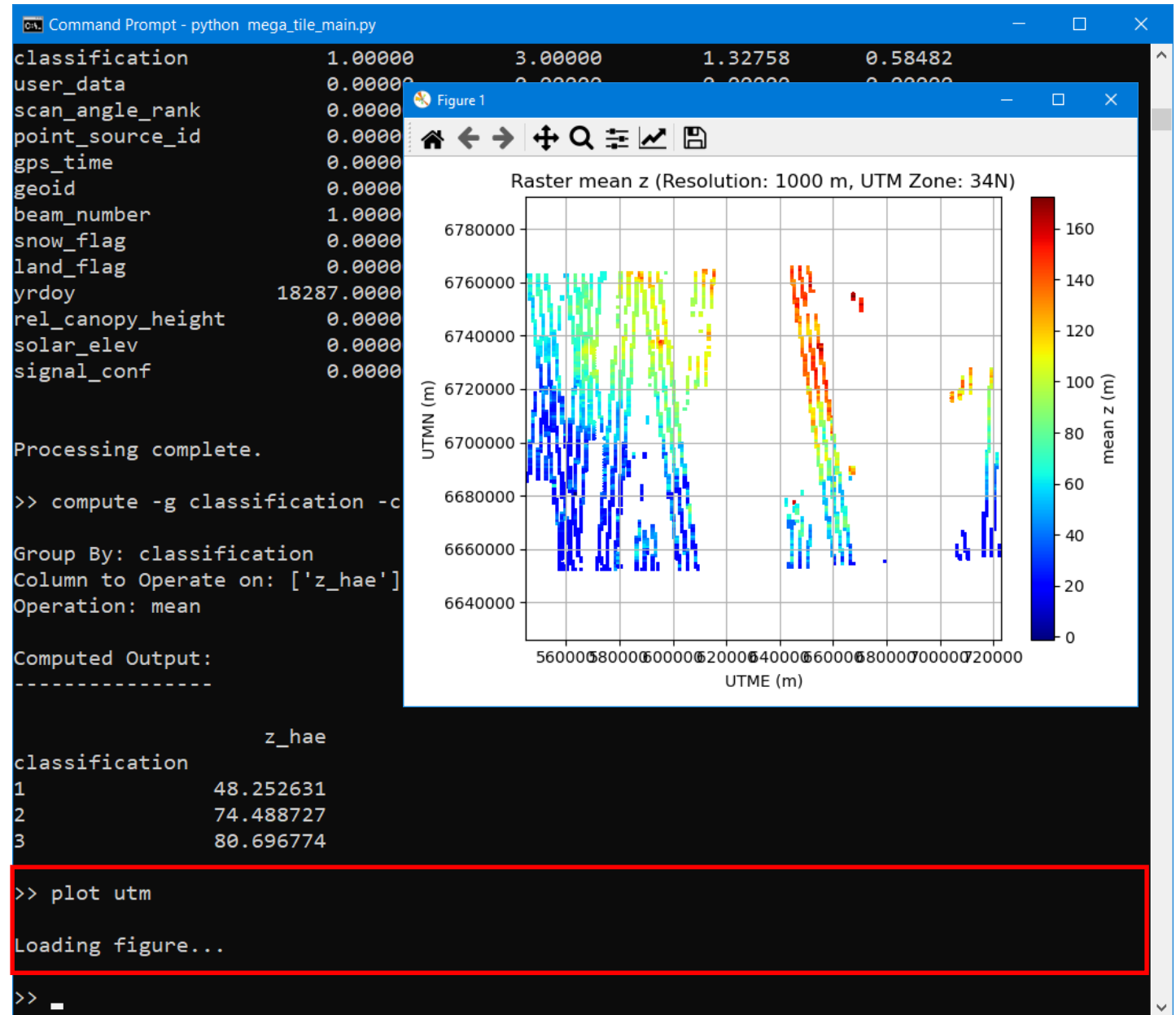
>> _
```

# ICESat-2 Mega Tiler

- Users can plot the mega tile data in UTM coordinates by altitude

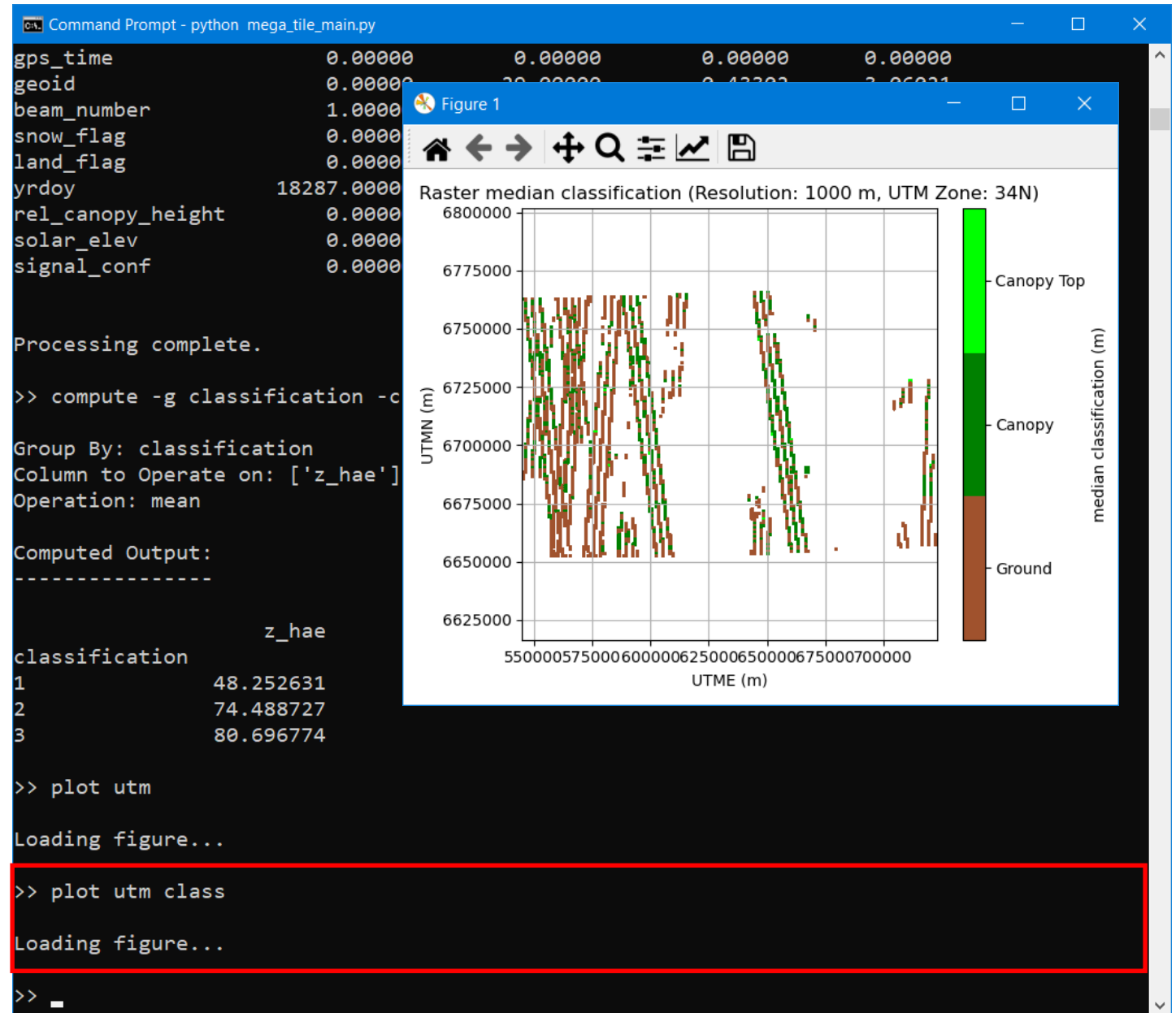
\* Any text with “utm” was replaced with “xy” in v1.0 to be open to UTM or Polar Stereographic coordinates

For instance, `plot utm` has been replaced with `plot xy`



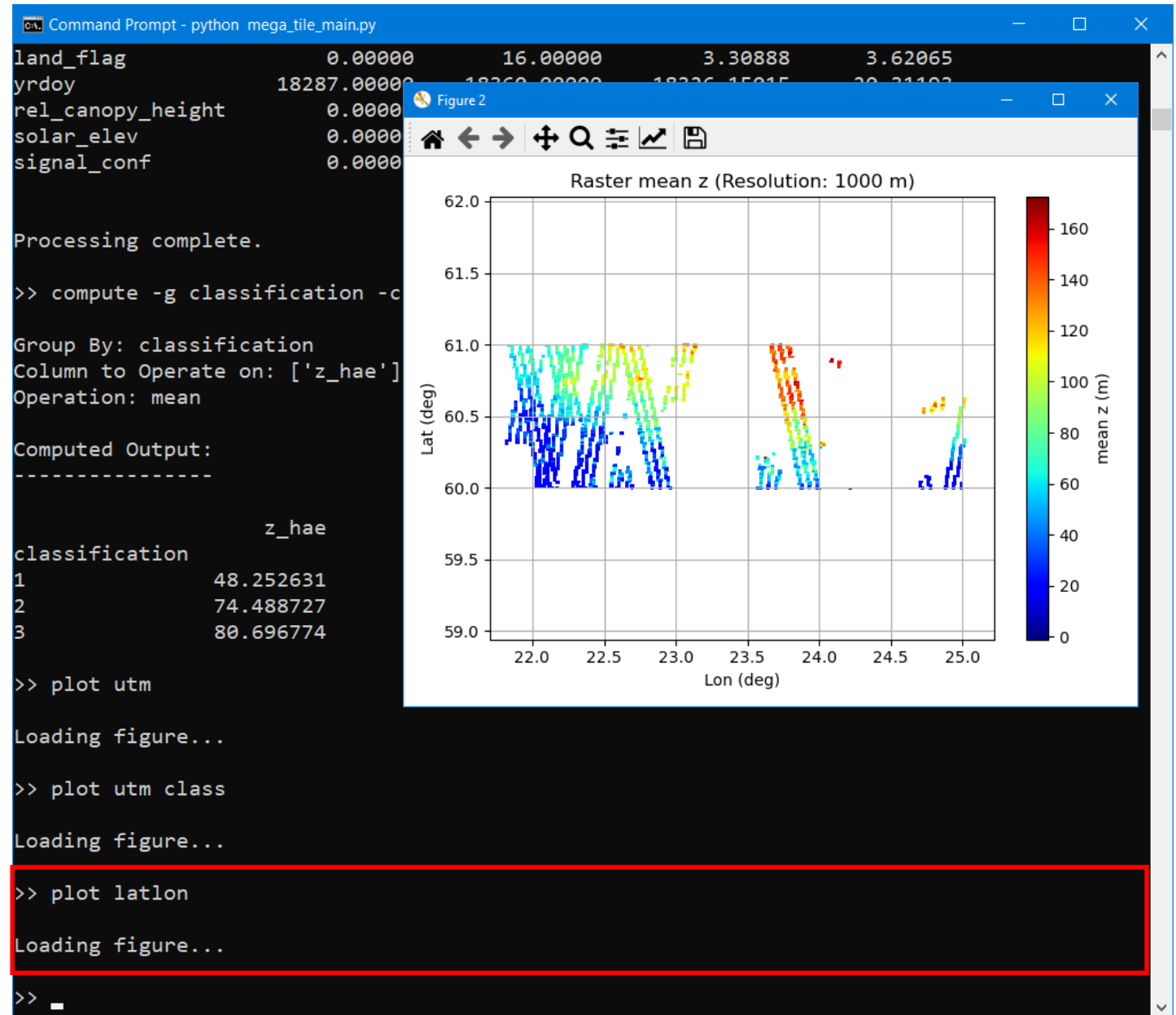
# ICESat-2 Mega Tiler

- Users can plot the mega tile data in UTM coordinates by classification



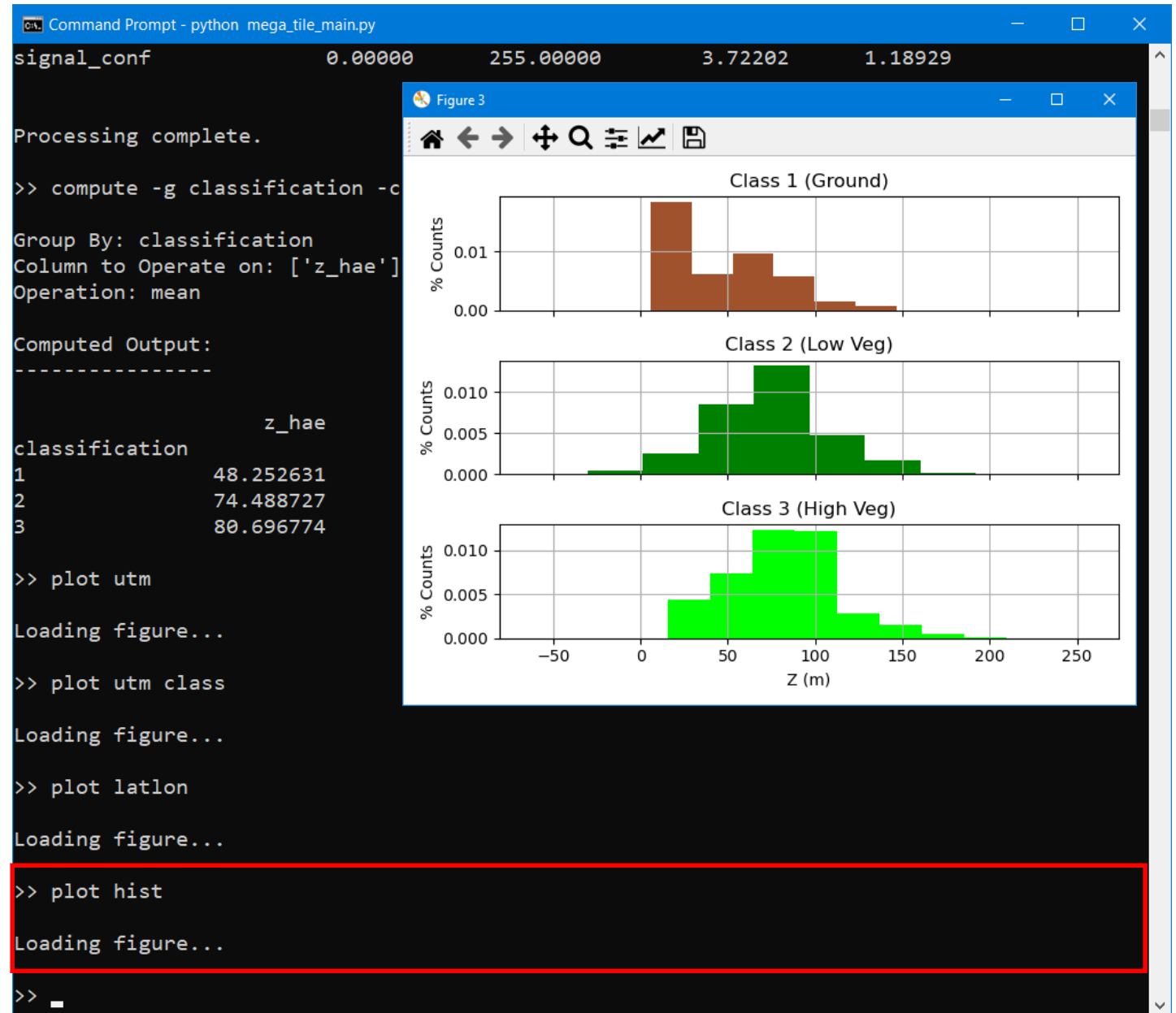
# ICESat-2 Mega Tiler

- Users can plot the mega tile data in lat/lon coordinates



# ICESat-2 Mega Tiler

- Users can plot a histogram of the data



# ICESat-2 Mega Tiler

- Users can plot the mega tile data as a 3D point cloud to pan/zoom around
- Data can be colored by:
  - Altitude (default)
  - Classification
- Users can rotate/zoom the 3D point cloud with the mouse and pan around the point cloud by holding Shift + Mouse Button

```
Command Prompt - python mega_tile_main.py
Column to Operate on: ['z_hae']
Operation: mean

Computed Output:
-----
                                z_hae
classification
1                                48.252631
2                                74.488727
3                                80.696774

>> plot utm
Loading figure...

>> plot utm class
Loading figure...


>> plot latlon
Loading figure...

>> plot hist
Loading figure...

>> plot hist all
Loading figure...

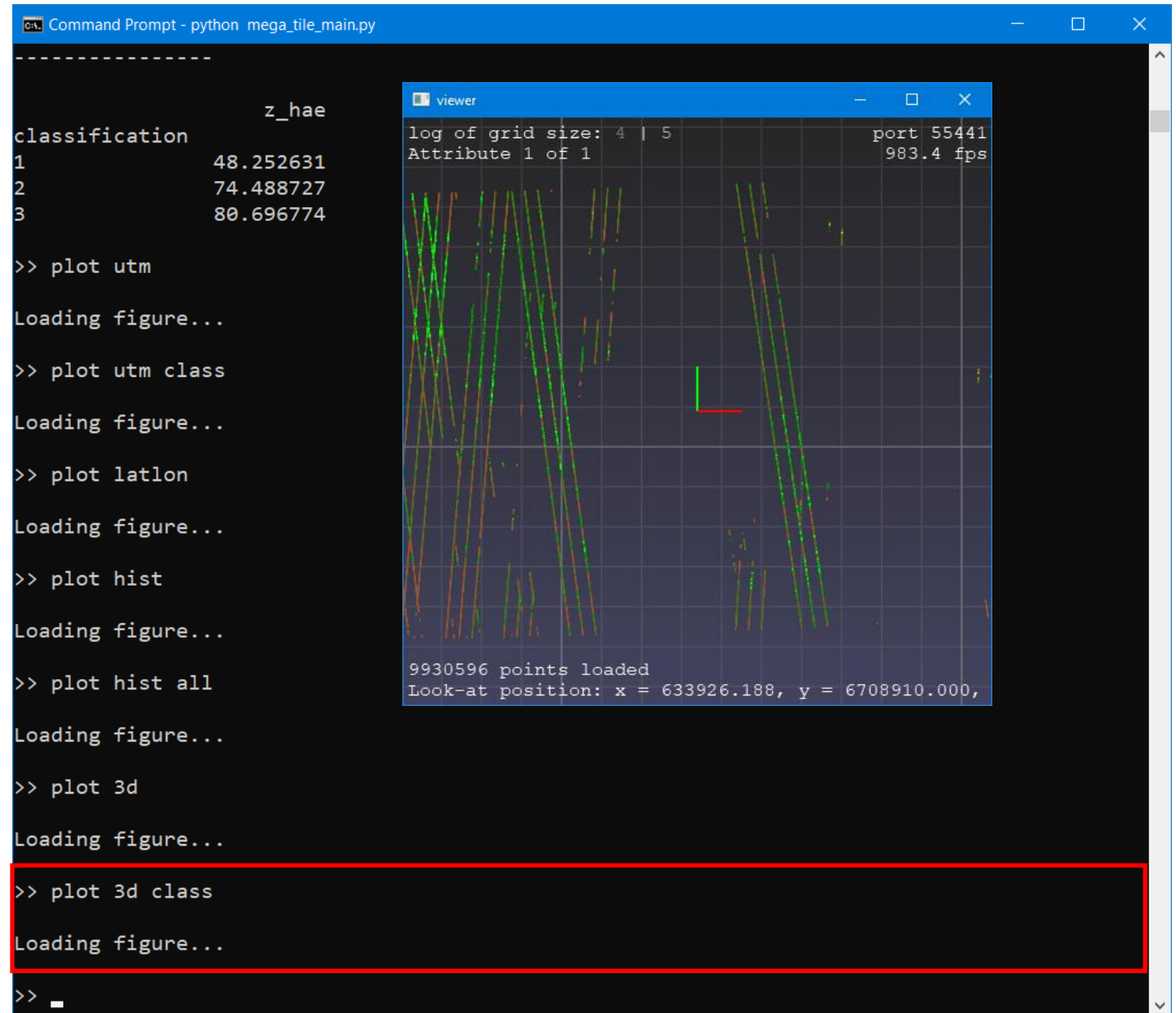
>> plot 3d
Loading figure...

>> _
```



# ICESat-2 Mega Tiler

- The point cloud can be colored by classification
  - Ground = Brown
  - Canopy = Dark Green
  - Top of Canopy = Light Green





# ICESat-2 Mega Tiler

- Data can be exported to:

- .gzip
- .csv
- .laz 1.4
- .kml
- .tif

\* Any text with “utm” was replaced with “xy” in v1.0 to be open to UTM or Polar Stereographic coordinates

```
Command Prompt - python mega_tile_main.py
exit -----> To exit this program (can also type 'quit' or 'stop')
help -----> To show help info for any method below
                Usage: help methodName
list -----> To list files with a specified extension
                Usage: list extName
load -----> To load previously processed Mega Tile data from .gzip file
                Usage: load processedFileName.gzip
process --> To process ICESat-2 Mega Tile .laz files from .inp file
                Usage: process inputFileName.inp
filter ---> To filter processed ICESat-2 Mega Tile data
                Usage: filter columnToFilter filterOperation filterValue
compute --> To compute stats on processed ICESat-2 Mega Tile data
                Usage: compute -g groupByColumn -c columnToOperateOn -o operationToPerform
plot -----> To plot processed ICESat-2 Mega Tile data
                Usage: plot plotTypeName
export ---> To export data to .gzip, .csv, .laz, .kml, or .tif files
                Usage: export outputFileName.extName

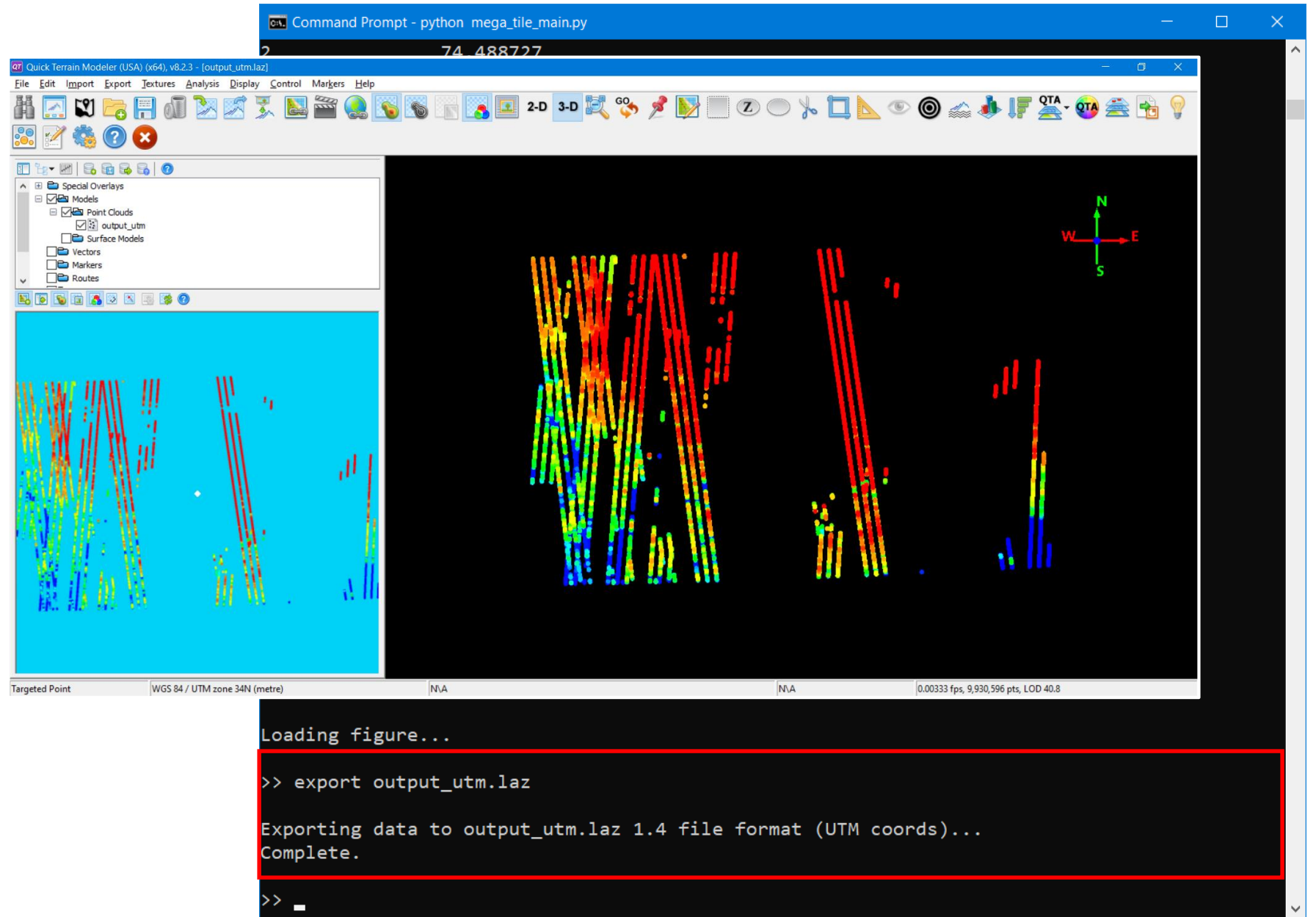
>> help export

export --> To export data to .gzip, .csv, .laz, .kml, or .tif files
                Usage: export outputFileName.extName
                Example: export fileName.csv OR export fileName.laz latlon
                Methods:
                    gzip --> export to .gzip file
                    csv ---> export to .csv file
                    laz ---> export to .laz 1.4 file
                Options:
                    utm ----> use UTM coordinates in .laz file (default)
                    latlon -> use Lat/Lon coordinates in .laz file
                    kml ---> export to .kml file
                    tif ---> export to .tif file
                Options:
                    utm ----> use UTM coordinates in .tif file (default)
                    latlon -> use Lat/Lon coordinates in .tif file

>> _
```

# ICESat-2 Mega Tiler

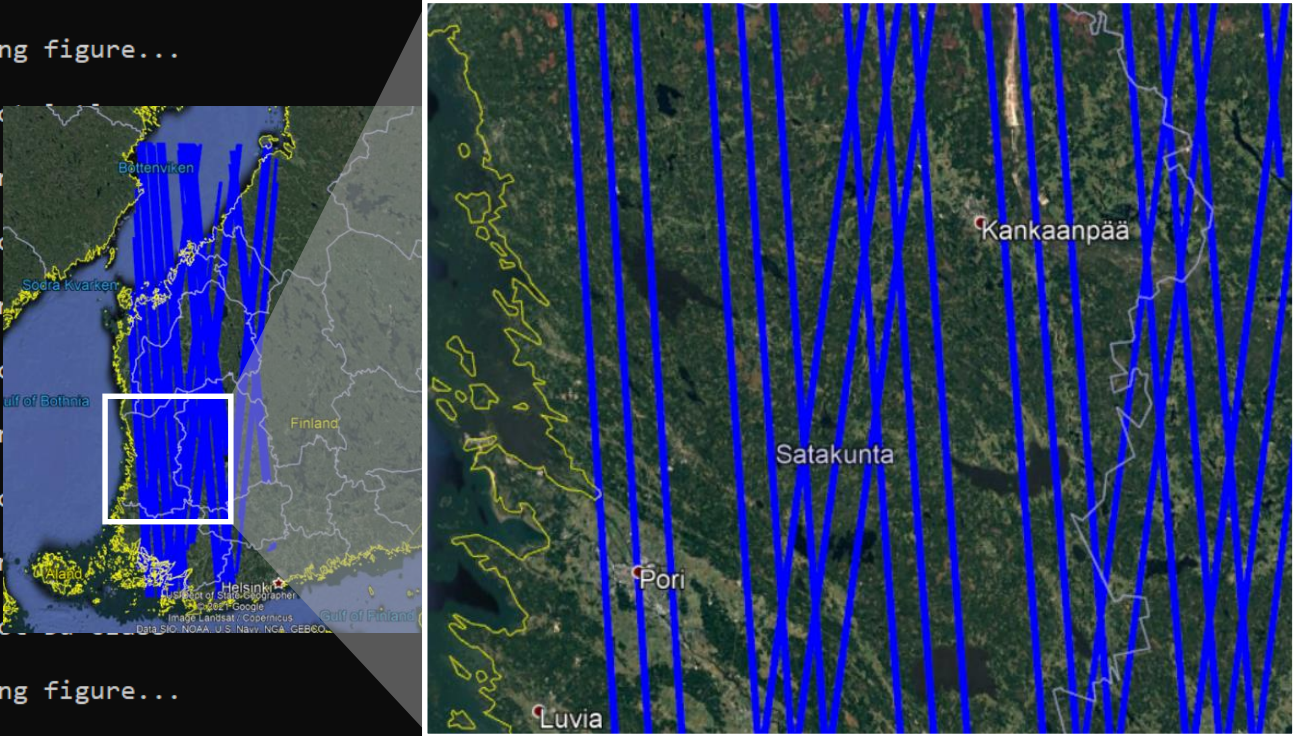
- Export to .laz file



# ICESat-2 Mega Tiler

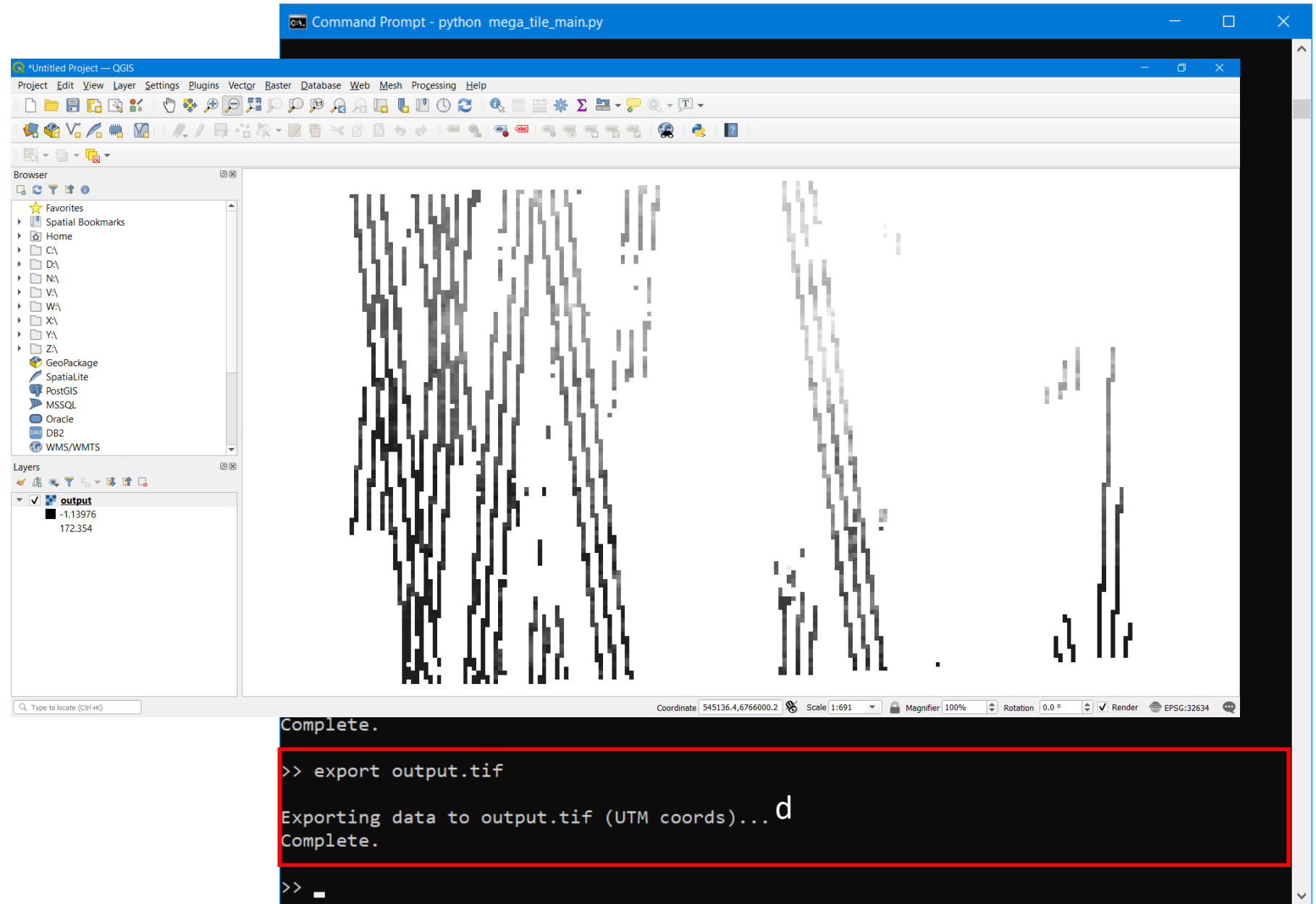
- Export to .kml file
- Each individual ICESat-2 track within the specified lat/lon bounds is shown as a separate line segment in the KML file

```
Command Prompt - python mega_tile_main.py
Loading figure...
>> plot utm class
Loading figure...
>> plot
Loading figure...
>> plot
Loading figure...
>> plot
Loading figure...
>> plot
Loading figure...
>> plot
Loading figure...
>> plot
Loading figure...
>> export output_utm.laz
Exporting data to output_utm.laz 1.4 file format (UTM coords)...
Complete.
>> export output.kml
Exporting data to output.kml...
Complete.
>>
```



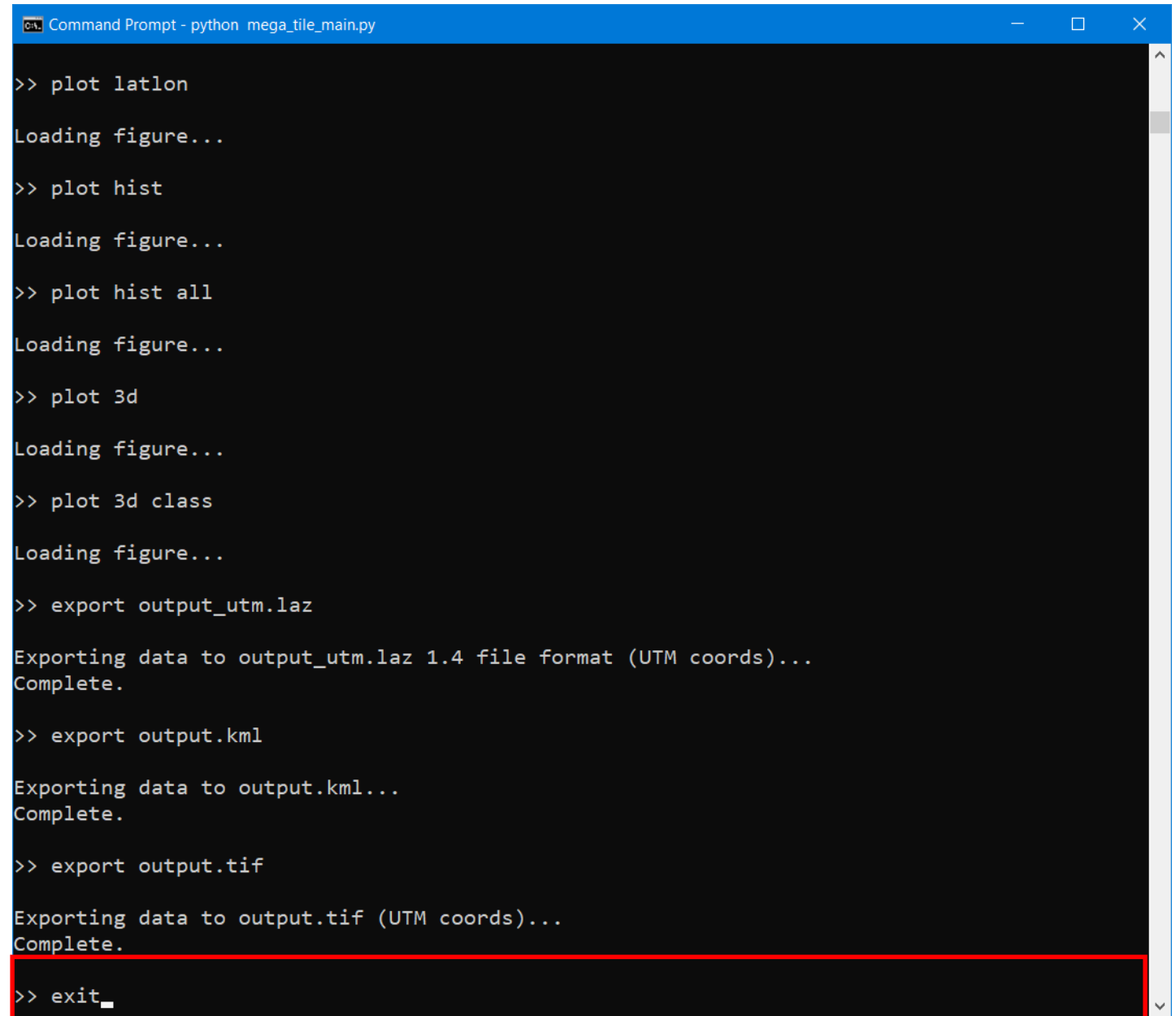
# ICESat-2 Mega Tiler

- Export to .tif file



# ICESat-2 Mega Tiler

- Type `exit` to exit from the tool and return to Windows Command shell



```
Command Prompt - python mega_tile_main.py

>> plot latlon
Loading figure...

>> plot hist
Loading figure...

>> plot hist all
Loading figure...

>> plot 3d
Loading figure...

>> plot 3d class
Loading figure...

>> export output_utm.laz
Exporting data to output_utm.laz 1.4 file format (UTM coords)...
Complete.

>> export output.kml
Exporting data to output.kml...
Complete.

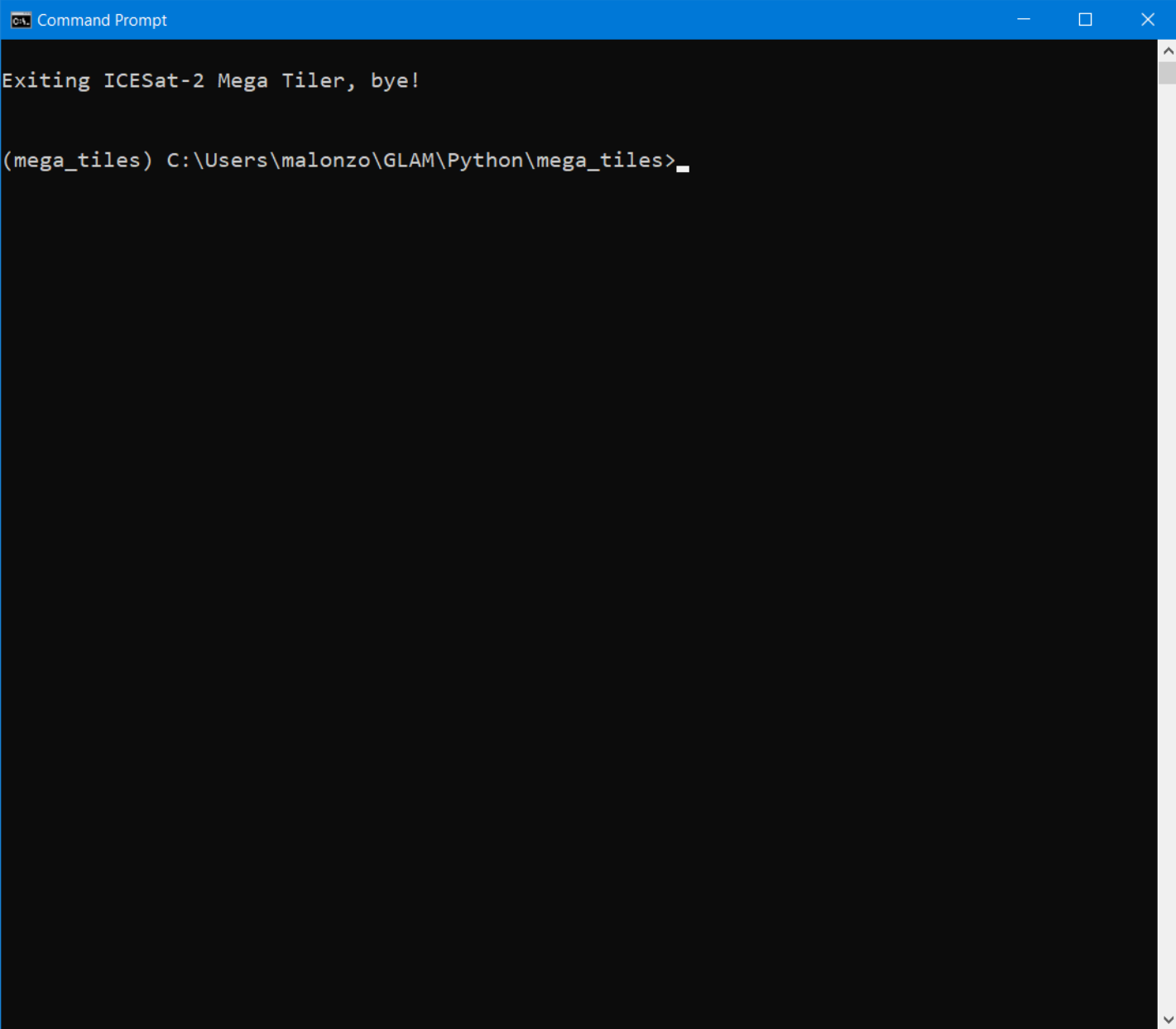
>> export output.tif
Exporting data to output.tif (UTM coords)...
Complete.

>> exit_
```

# ICESat-2 Mega Tiler

- Back to the Windows command shell
- Still in the Python Conda environment
- To exit Python Conda environment, type:

```
>> deactivate
```



```
Command Prompt

Exiting ICESat-2 Mega Tiler, bye!

(mega_tiles) C:\Users\malonzo\GLAM\Python\mega_tiles>
```