GEMSpa

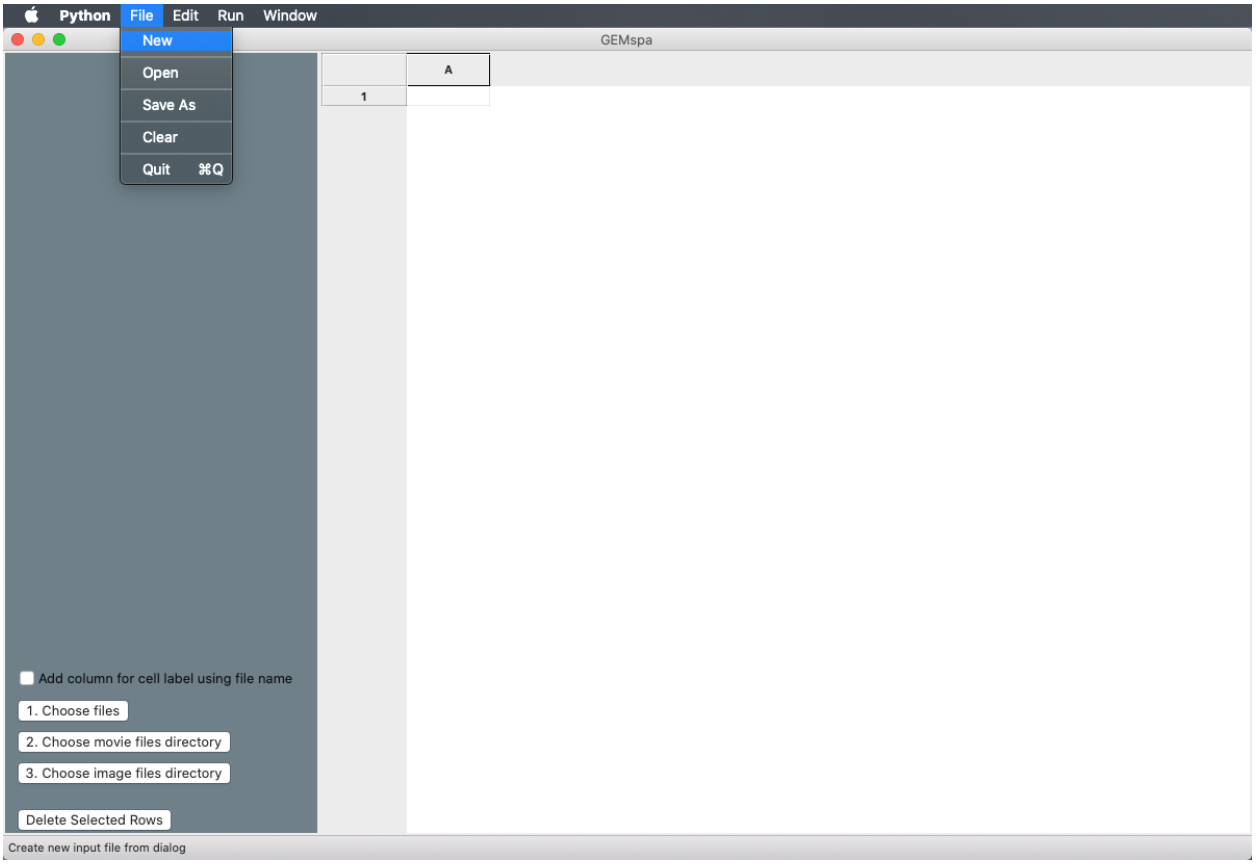
USER MANUAL

5/17/2022

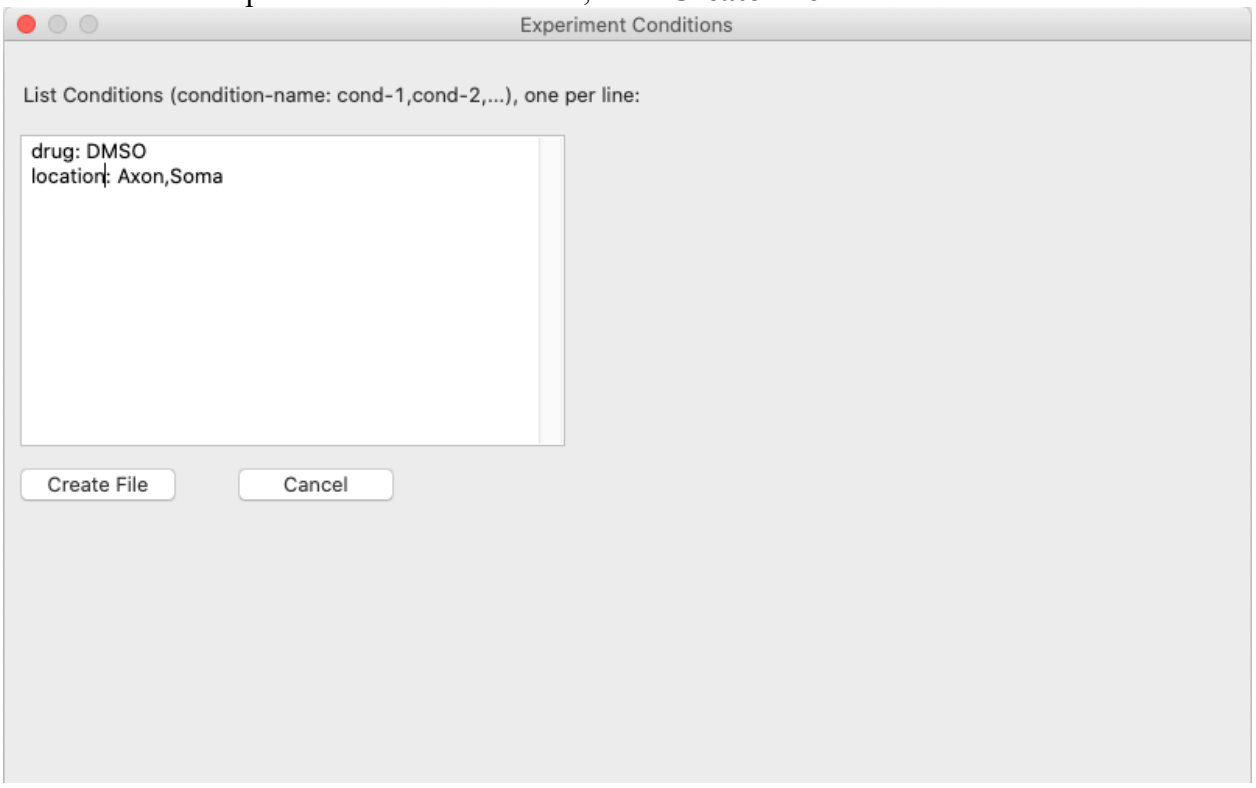
SECTION 1: CREATING THE DATA SUBMISSION FILE

The first step for running GEMspa is to create a data submission file. This file is a tab-delimited text file with pre-defined headers that contains the list of CSV files (tracking output from MOSAIC), along with the experimental conditions for each group. This file can be created using the GEMspa GUI interface:

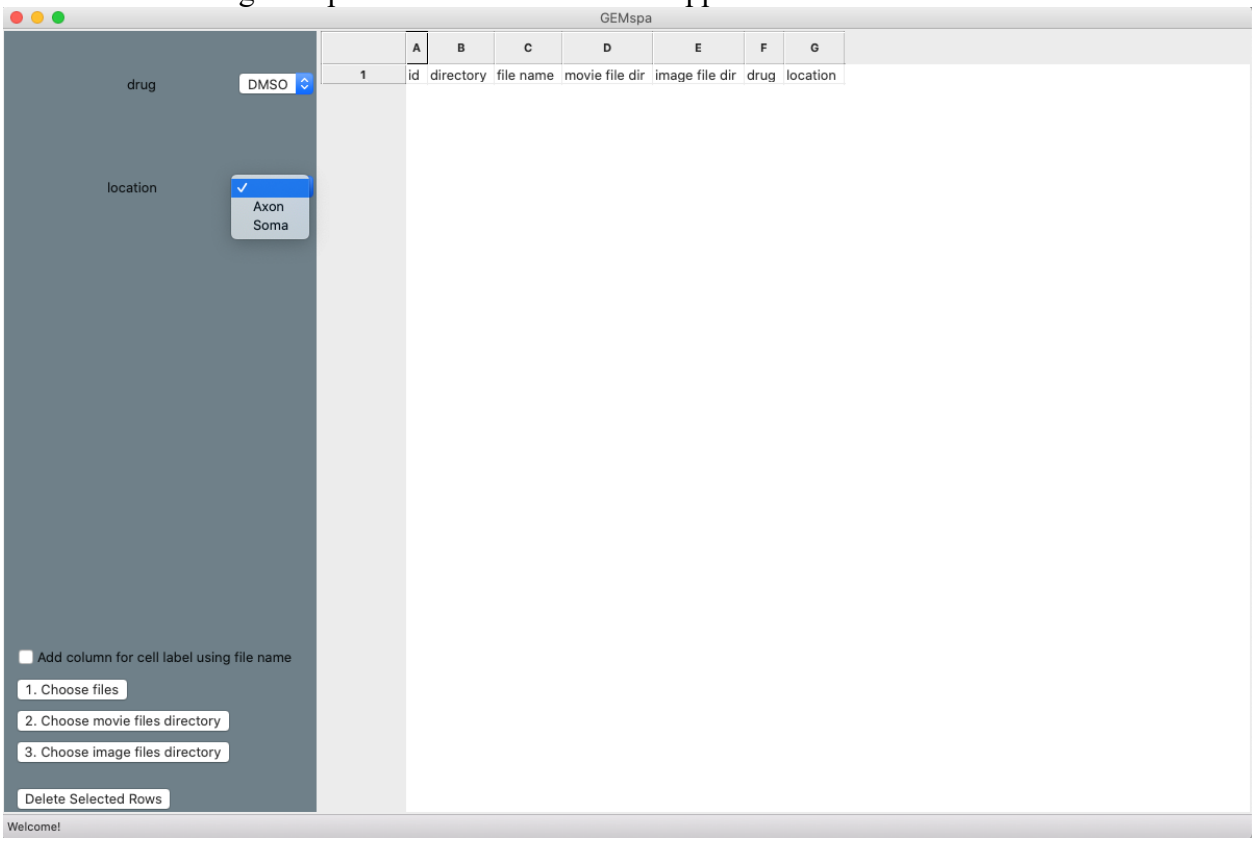
Go To **File -> New**



The dialog box for experimental conditions will open. Enter the conditions for your experiment. Follow the format in the example below. When finished, click **Create File**:

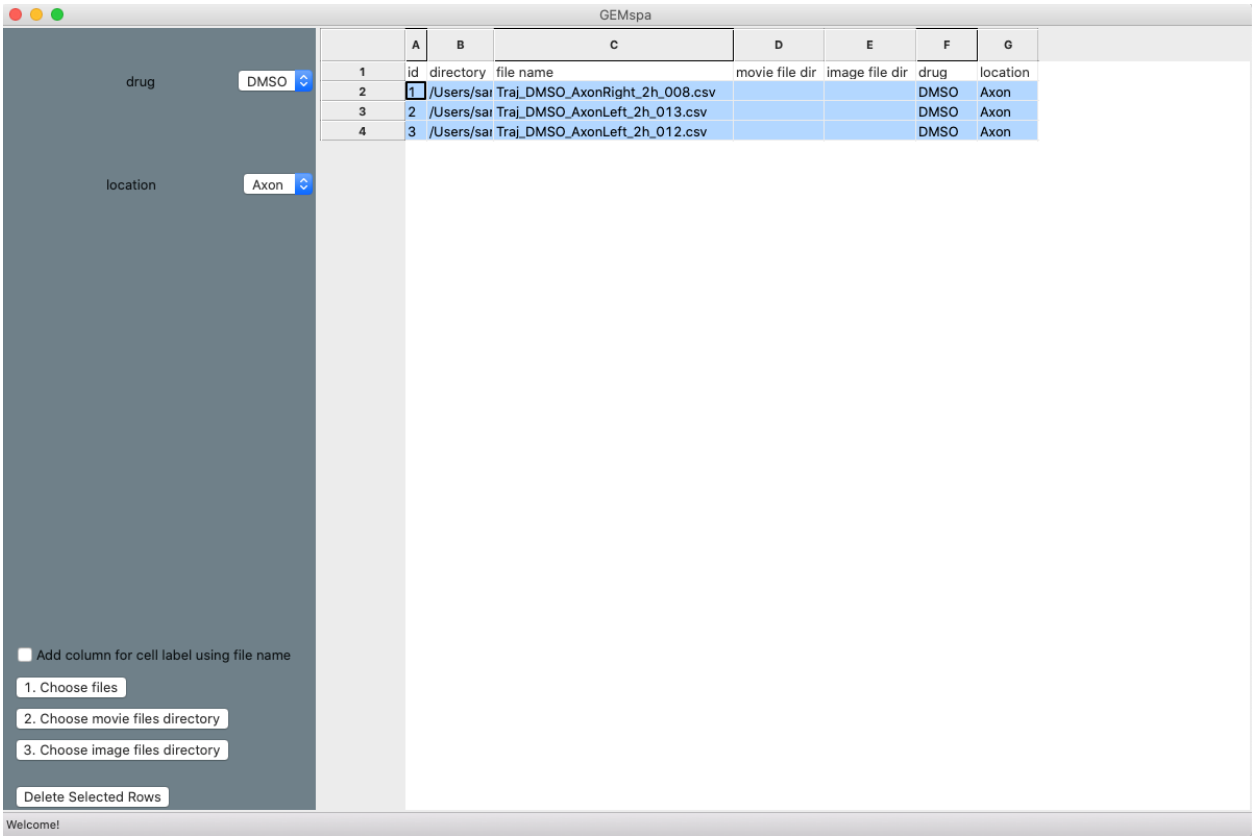


The column names for the data sheet in the main window will be populated, and drop-down boxes on the left side with the listing of experimental conditions will appear:



The next step is to add your list of CSV files (MOSAIC output files) for each group. Use the drop-down boxes to choose the conditions and click on the button **1. Choose files**. Then, select a set of CSV files and they will be added to the data sheet.

Grouping data by cell ID:
If you would like to group your data by cell (for example, if you have imaged the same cells at multiple time points and want to see results for each cell over time), GEMspa provides a mechanism to include that information in the output table and visualize results organized in this way. Check the **Add column for cell label using file name** check box *before beginning to add your CSV files to the table*, and see SECTION 4 for instructions regarding the required naming convention for the files.



Continue to add all CSV files for your experiment by selecting from the drop-down boxes to choose the conditions and clicking on **1. Choose files**.

Optional Extra Features

At any point while adding your CSV files, you may select a set of files from the list and associate a directory with additional files by clicking one of the following buttons (see SECTION 4 for more details on these features and the required file naming conventions):

- **2. Choose movie files directory** – a directory containing nd2 or tiff GEM movie files for reading time step and scale directly from the meta data
- **3. Choose image files directory** – a directory containing image files for making rainbow tracks and/or ImageJ ROI files/MASK images for filtering MOSAIC tracks by region.

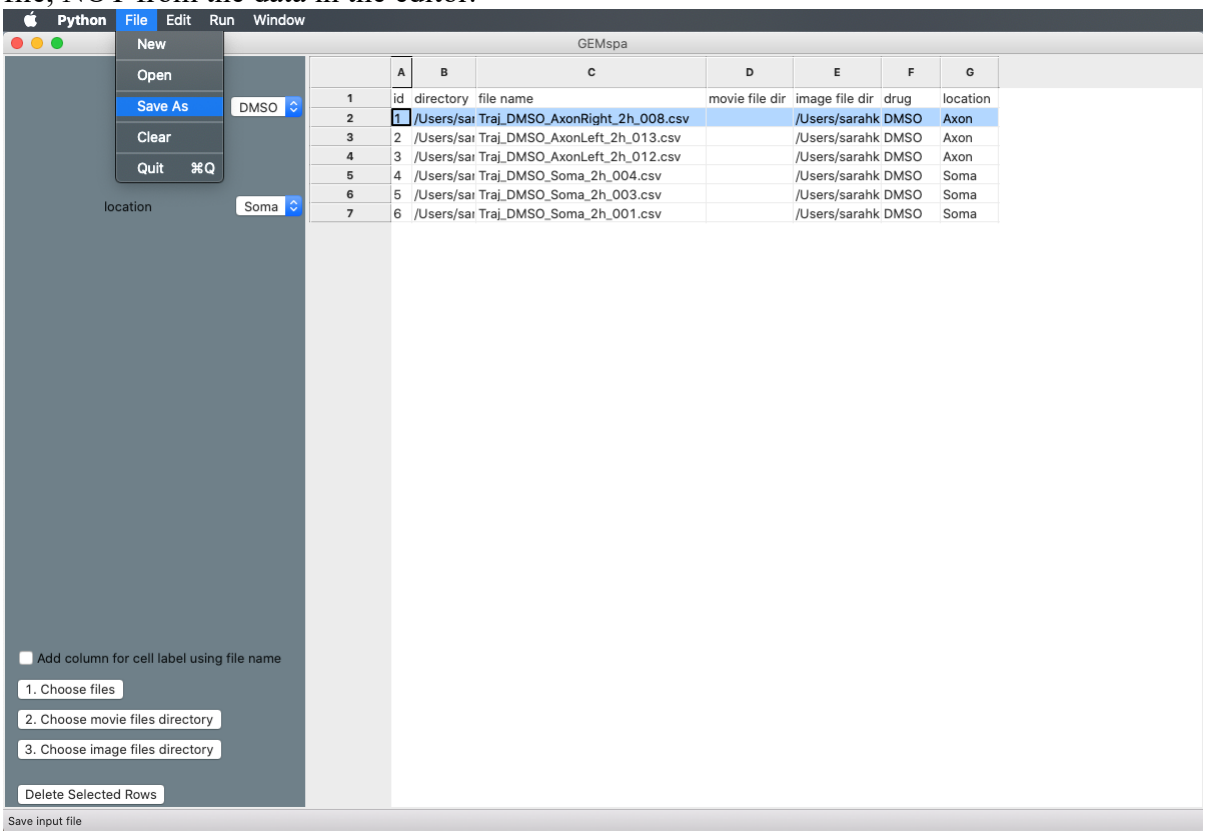
Note that you must select the set of CSV files in the matrix for which you'd like to add these directories. The program will not automatically add them for all CSV files in the list. (in case files are split between multiple directories). Each time a list of CSV files are added using the **1. Choose files** *button, that set will be automatically selected for you.*

Deleting Rows

You may delete row(s) added by mistake with the **Delete Selected Rows** button. *Be careful not to delete the header row!*

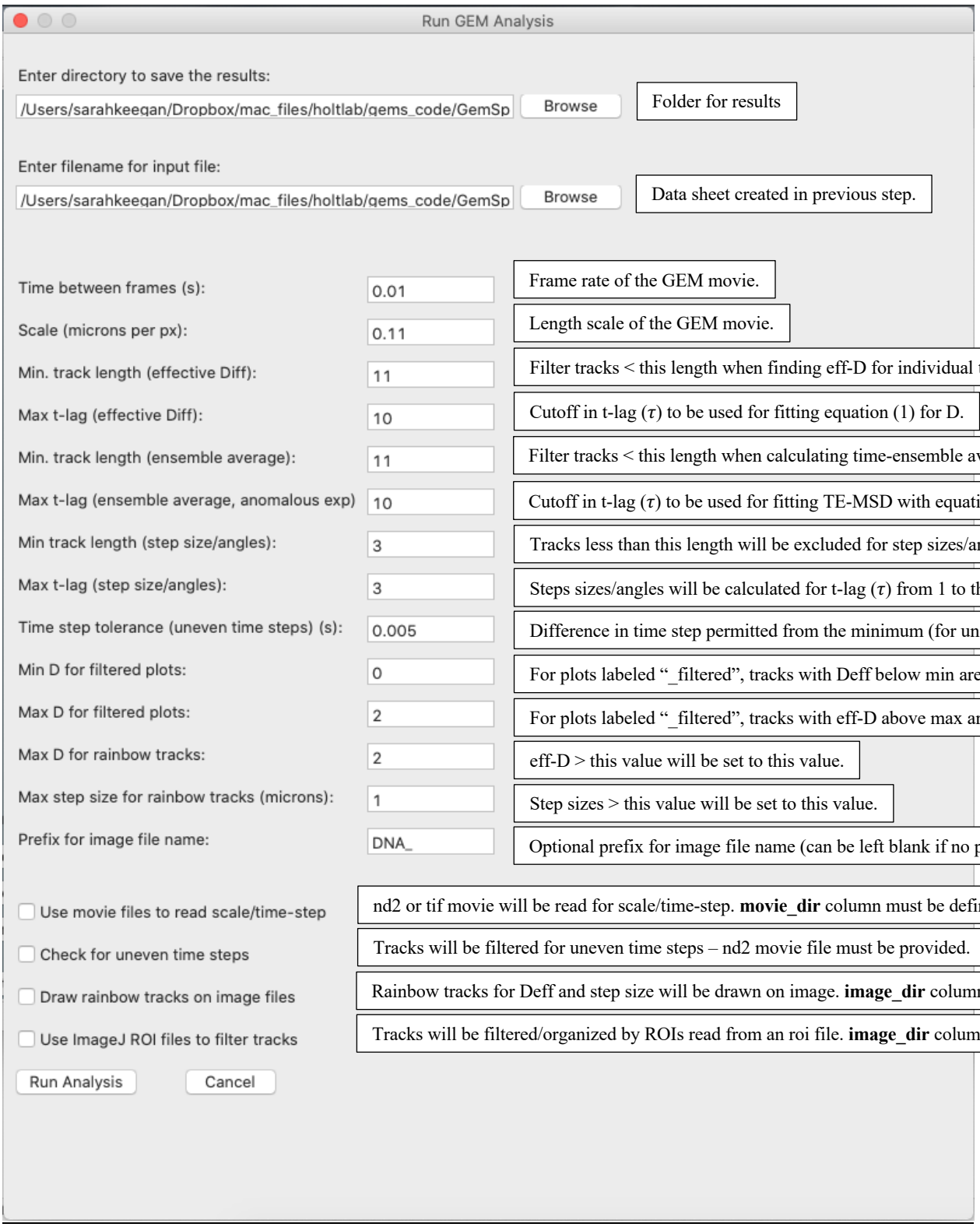
Saving

When finished adding all of your data, it is important to SAVE this file as a **.txt** file (**File->Save As**). You will need to select the saved version of this file when running the analysis: GEMspa reads directly from the saved file, NOT from the data in the editor.



SECTION 2: RUNNING THE ANALYSIS

After creating and saving the data sheet for your experiment, you are ready to run the analysis. In the GEMspa interface, click on **Run -> GO.** The Run Dialog Box will open. Here it has been annotated with brief descriptions for the parameters. Section 4 provides further details for the parameters associated with the optional features: movies with uneven time-steps, rainbow tracks, ROIs for filtering tracks



MSD = (2d) mean square displacement, τ = time displacement (t-lag) for calculating MSD value

$MSD(\tau) = 4D\tau$ **eq. 1**, D = effective diffusion coefficient (um^2/s), fitted up to max τ, as specified above

$MSD(\tau) = 4K\tau^{\alpha}$ **eq. 2**, α = anomalous exponent, K = generalized Diffusion coefficient (um^2/s^{α}), fitted up to max τ, as specified above

Once you have chosen your parameters, click "Run Analysis." GEMspa will begin to run. A log file will be output in the results directory where you can view the status of the run and any errors encountered. Eventually, the full output will be saved to the results directory. See SECTION 3 for details on all output.

SECTION 3: OUTPUT

GEMspa will output the following data files in tab-delimited format (.txt):

summary.txt – results summarized for each input csv file or ROI. Includes columns in the input data sheet and additional columns as follows:
roi: if filtering by ROIs, the ROI name is listed here. If a mask image was used, then the label of the region is listed here. A labeled mask image is output to the <**image file dir**>
D_median: median effective diffusion coefficient over all tracks in the file/ROI; for each track D is calculated by fitting eq. 1 to the MSD values
D_median_filtered: median effective diffusion coefficient over all tracks where min <= D <= max from Run parameters ("Max/Min D for filtered plots")
D_mean: mean effective diffusion coefficient (as above)
D_mean_filtered: mean filtered effective diffusion coefficient (as above)
num_tracks: number of tracks for this file/ROI
num_tracks_D: count of tracks with track length >= min from Run parameters ("Min. track length (effective Diffusion)")
area: if filtering by ROIs, the ROI area (in microns)
ensemble_D: effective diffusion coefficient calculated by fitting eq. 1 to the time-ensemble average MSD values
ensemble_r_sq: r^2 for the fit of eq. 1
ensemble_loglog_K: generalized diffusion coefficient calculated from fitting eq. 2 to the time-ensemble average MSD values*
ensemble_aexp: anomalous exponent calculated from fitting eq. 2 to the time-ensemble average MSD values*
ensemble_loglog_r_sq: r^2 for the fit of eq. 2*
group: concatenation of the experimental conditions for this row (used for making/labeling plots)

group_summary.txt – results summarized by group (each combination of experimental conditions from the input data sheet). Includes the following columns:
ensemble_D: effective diffusion coefficient calculated by fitting eq. 1 to the time-ensemble average MSD values
ensemble_r_sq: r^2 for the fit of eq. 1
ensemble_loglog_K: generalized diffusion coefficient calculated from fitting eq. 2 to the time-ensemble average MSD values*
ensemble_aexp: anomalous exponent calculated from fitting eq. 2 to the time-ensemble average MSD values*
ensemble_loglog_r_sq: r^2 for the fit of eq. 2*
D_group_median: median effective diffusion coefficient over all tracks in the group; for each track the D is calculated by fitting eq. 1 to the MSD values
D_group_mean: mean effective diffusion coefficient (D calculated as above)

D_group_std: standard deviation of effective diffusion coefficients (D calculated as above)
D_group_sem: SEM of effective diffusion coefficients (D calculated as above)
aexp_group_median: median of the anomalous exponents obtained by fitting eq. 2 to the MSD values for each track
aexp_group_mean: mean of the anomalous exponents obtained by fitting eq. 2 to the MSD values for each track
aexp_group_std: standard deviation of the anomalous exponents obtained by fitting eq. 2 to the MSD values for each track
aexp_group_sem: SEM of the anomalous exponents obtained by fitting eq. 2 to the MSD values for each track
group_num_tracks: number of tracks for this group (based on min. track length setting in Run parameters)
ensemble_num_tracks: number of tracks included in ensemble averaging (based on min. track length for ensemble average in Run parameters)
MSD_ave_τ (min < τ < max): ensemble averaged MSD values for given τ, average is over all tracks in the group
MSD_std_τ (min < τ < max): standard deviation of MSD values for given τ

all_data.txt – results for each track in each file or ROI. Includes the columns in the input data sheet, with additional columns as follows:
roi (see above)
D_median (see above)
D_median_filtered (see above)
D_mean (see above)
D_mean_filtered (see above)
avg_velocity: mean step size/time-step for the current track
int_mean: mean of spot intensities for the current track
int_std: standard deviation of spot intensities for the current track
Trajectory: trajectory ID from input file
D: effective diffusion coefficient calculated by fitting eq. 1 to the MSD values of the current track
err: one standard deviation error on the estimated parameter (D)
r_sq: r^2 for the fit
rmse: root mean square error (residuals)
track_len: the length of the current track
D_max_tlag: max τ considered for fitting eq. 1
K: generalized diffusion coefficient calculated by fitting eq. 2 to the MSD values for the current track*
aexp: anomalous exponent (α), calculated by fitting eq. 2 to the MSD values for the current track*
aexp_r_sq: r^2 for the fit of eq. 2
aexp_rmse: root mean square error (residuals) for α
group (see above)

** Note: fitting of equation 2 for α, K is performed on a log-log scale:*
$\mathrm{MSD}(\tau) = 4K\tau^{\alpha}$
$\log \mathrm{MSD}(\tau) = \alpha \log \tau + \log 4K$

all_data_angles.txt – all angles listed by t-lag, for each file/ROI. Relative angle defined as in Burov, et al. PNAS 2013.

all_data_step_sizes.txt – all steps sizes (distance between successive points in each trajectory) listed by t-lag, for each file/ROI. In microns.

summary_step_sizes.txt – mean/median of step sizes for each t-lag value, for each file/ROI. In microns.

cos_theta_by_group.txt – average cosine of the relative angles (same as in all_data_angles.txt), where average is taken over all angles in each group. cos $\theta = 0$ indicates no correlation, cos $\theta > 0$ indicates correlation, and cos $\theta < 0$ indicates anti-correlation. (average is calculated for each t-lag.)

GEMspa will also output the following plots in pdf format:

summary_D_median.pdf – median effective diffusion coefficient is shown per file/ROI for each sample group as a box plot (each data point is a file/ROI from the sample group)
summary_D_median_filtered.pdf – same as above, but filtered based on min/max effective diffusion coefficient from Run parameters
summary_combined_D.pdf – median effective diffusion coefficient for all tracks from each sample group combined. Error bar is 95% confidence interval.
MSD_ensemble_by_group.pdf – ensemble average MSD plots
cos_theta_by_group.pdf – average cosine θ (θ=relative angles), plotted at each τ, shown for each group
combined_allgroups_Deff_gkde.pdf – the distribution of the effective diffusion coefficients (D from fitting eq. 1 to each track) plotted as a gaussian kernel density estimation (smooth curve), for each group (bin size=0.01)
combined_allgroups_Deff.pdf - the distribution of the effective diffusion coefficients (D from fitting eq. 1 to each track) plotted as a normalized histogram (bar heights sum to 1), for each group (bin size = 0.1)
combined_allgroups_alpha_gkde.pdf – the distribution of the anomalous exponent values (α from fitting eq. 2 to each track) plotted as a gaussian kernel density estimation (smooth curve), for each group (bin size=0.01)
combined_allgroups_alpha.pdf - the distribution of the anomalous exponent values (α from fitting eq. 2 to each track) plotted as a normalized histogram (bar heights sum to 1), for each group (bin size = 0.1)
alpha_D_heatmaps.pdf – bivariate distribution of log(D) vs. anomalous exponent (α) for each track, for each group

If Rainbow tracks option is selected, GEMspa will output the following images:

<image_name>_tracks_Deff.tif – "rainbow tracks" drawn on the designated image file. Tracks are colored by effective diffusion coefficient.
<image_name>_tracks_ss.tif – "rainbow tracks" drawn on the designated image file. Tracks are colored by step size.
<image_name>_tracks_time.tif – "rainbow tracks" drawn on the designated image file. Tracks are colored by time.

SECTION 4: INSTRUCTIONS FOR OPTIONAL FEATURES

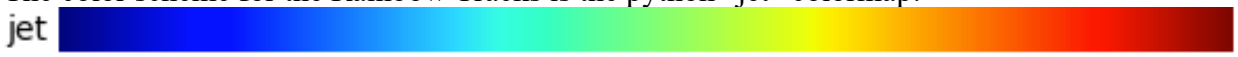
Grouping by cell ID
If you would like to group your data by cell (for example, if you have imaged the same cells at multiple time points and want to track each cell over time), GEMspa provides a mechanism to include that information in the output table and visualize results organized in this way. While creating the data submission file (SECTION 1), check the **Add column for cell label using file name** check box *before beginning to add your CSV files to the table*. **File naming**: your CSV files must contain a suffix indicating cell ID, separated by an underscore ('_') from the rest of the file name, e.g. '<filename>_A.csv' for cell A, '<filename>_B.csv' for cell B, etc. GEMspa will extract the cell ID from the file name and create a new column in the data sheet named "cell."

Reading Time step and scale directly from GEM movie meta-data
If you have GEM movie files (nd2/oem-tif) for reading the time step and scale information from the meta-data, you can select rows from the data sheet and then click on the button **2. Choose movie files directory** to choose the directory where the movie files are located. *Note that for uneven time step filtering, only nd2 movie files are acceptable. OEM TIF files may not have complete information on individual time steps.*

Time step filtering of tracks for uneven time steps
If your GEM movie may contain time steps that are unequal, GEMspa will filter to include only tracks (or portions of tracks) where the time between frames is within a certain range relative to the minimum time between frames for the movie. This range is defined in the Run parameter called "Time step resolution." For example, if time step resolution is 0.005s and the min. time step is 0.012s, then any frames with time step > 0.017s will be considered as invalid and these frames will be removed from a track. The largest continuous portion of a track not containing any invalid frames will be set as the new, fixed track.

Rainbow Tracks and filtering by ROI/MASK
If you have an additional image (e.g. a DNA stain), GEMspa can draw 'rainbow tracks' on this image for visualizing both the effective Diffusion coefficient (Deff) of each track and the step-sizes. Select rows from the data sheet and then click the button **3. Choose image files directory** to choose the directory where the image files are located. If you want to limit the tracks that are analyzed from the CSV file to only tracks within an ROI, the ROI files should be placed in this directory also. Note: ROI files can be accepted as .roi/zip or as a mask (tif). If the ROI file is a mask, it is not necessary for an image file to be present. If the ROI file is .zip/roi, then the corresponding image file must be present.

The color scheme for the Rainbow Tracks is the python "jet" colormap:



The Run parameters "Max D for rainbow tracks" and "Max step size for rainbow tracks" allow you to set the top value for the dark red color. Dark blue corresponds to Deff or step size = 0.

File naming conventions
In order for GEMspa to correctly match the movie/image/ROI files with the corresponding CSV files, use the following naming conventions:

CSV file: Traj_<filename>.csv

Movie file (nd2 or OEM TIF): <filename>.tif or <filename>.nd2

Image file (2D greyscale TIF image): PRE_<filename>.tif

ROI file (ImageJ): (PRE_)<filename>.zip or (PRE)<filename>.roi
or
Image Mask: (PRE_)<filename>_mask.tif

PRE_: PRE_ is an optional prefix that can be part of the image file name. This prefix is entered in the Run dialog box. It can also be left blank. If ROI/Mask files exist, GEMspa will search for them with and without the prefix. Either naming style is acceptable.

ROI Types: These ROI types can be read by GEMspa: Rectangle, Rotated Rectangle, Oval, Polygon, Freehand. (Rounded Rectangle, Elliptical and Selection Brush Tool are not supported).

Mask images: A mask image may also be used instead of an ROI file from ImageJ. The image mask should be a 2-valued image with 0 for background and 1 or 255 as foreground.

Output Naming: In ImageJ, each ROI in the ROI Manager has a name, and it is possible to edit this name. These ROI names will be included in the output data of GEMspa for linking results back to the corresponding ROIs. If an image mask is used, GEMspa will output a labeled image where each region is labeled with an ID number. These ID numbers will be included in the GEMspa output tables.

SECTION 5: INSTALLATION

1) Download and install python version 3.7 from https://www.python.org/downloads/release/python-379/ - scroll to the bottom of the page under Files – select the appropriate installer. Here are direct links:
   macOS 64-bit installer (Mac OS X 10.9 and later)
   Windows x86 executable installer

   Note: Python version 3.10 may give errors when installing the GEMSpa requirements

2) Clone (download) the github repository at https://github.com/liamholtlab/GEMspa

3) Create a virtual environment – an isolated python environment - for running GEMSpa.

   Mac Instructions

   i) Open Terminal by going to Launchpad and typing "Terminal" in the Search Box.
   ii) Since Mac OS comes pre-installed with python, we can check to ensure the python that was installed in Step 1 is being used for our GEMSpa installation.

   At the Terminal prompt ($), type:

   ```
   which python3
   ```

This command should print the location of python 3.7 that was just installed (usually it is installed at: `/Library/Frameworks/Python.framework/Versions/3.7/bin/python3.7`). You can check the timestamp on the file by typing:

```
ls –l /Library/Frameworks/Python.framework/Versions/3.7/bin/python3.7
```

If this does not appear to be the correct installation, you can try:

```
which python3.7
```

In this case, use the name `python3.7` in step b. below.

iii) Install and activate a virtual environment (an isolated installation of python) and install the python packages needed for GEMSpa. In the Terminal, type:

```
python3 –m venv gemspa–python
source gemspa–python/bin/activate
python3 –m pip install –r <path–to–GEMSpa>/requirements.txt
```

`gemspa–python` can be any name you want to call the virtual environment – just don't use any special characters or spaces in the name. `<path–to–GEMSpa>` is the full path to the location where you downloaded GEMSpa.

iv) Now you are ready to run GEMSpa. GEMSpa can be run from the Terminal by typing:

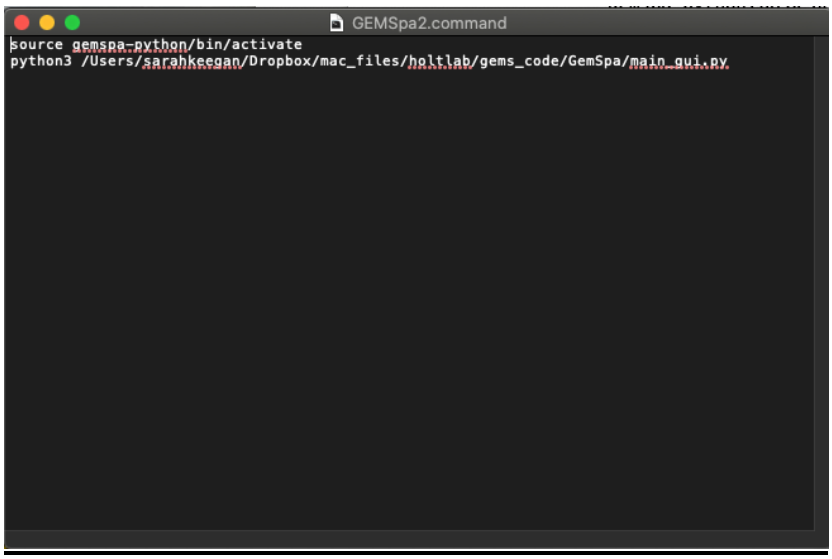
```
gemspa–python/bin/python3 <path–to–GEMSpa>/main_gui.py
```

v) You can also create a Desktop Shortcut so you don't have to open Terminal and type in the commands manually:

Open a blank text file with TextEdit. Change the type to "Plain Text" (Format -> "Make Plain Text"). Enter the following commands:

```
source gemspa–python/bin/activate
gemspa–python/bin/python3 <path–to–GEMSpa>/main_gui.py
```

Save this file as "GEMSpa.command" on your Desktop. Edit the extension so that there is no ".txt" extension. It must end in ".command." Here is an example:



In order to "execute" this file, you will need to grant it execute permissions. Open Terminal, and type:

```
chmod a+x ~/Desktop/GEMSpa.command
```

Windows Instructions

i) Open Command Prompt by clicking Start and then typing "cmd" into the Search Box.

ii) Install a virtual environment (an isolated installation of python) and install the python packages needed for GEMSpa. In the Command Prompt, type:

```
python -m venv gemspa-python
C:/<current-dir>/gemspa-python/Scripts/python -m pip install -r <path-to-GEMSpa>/requirements.txt
```

This will create a folder called `gemspa-python` with a python installation in the current directory. `gemspa-python` can be any name you want to call the virtual environment – just don't use any special characters or spaces in the name. `<path-to-GEMSpa>` is the full path to the location where you downloaded GEMSpa. <current-dir> is the current directory (where the gemspa-python folder was created).

iii) Now you are ready to run GEMSpa. GEMSpa can be run from the Command Prompt by typing:

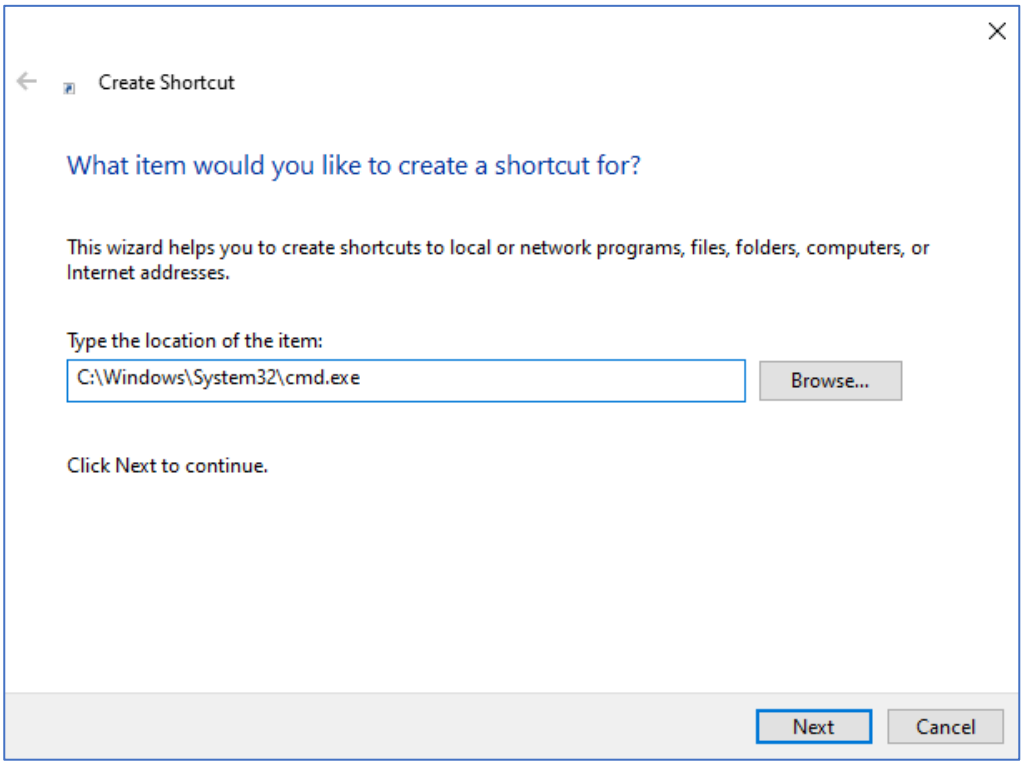
```
C:/<current-dir>/gemspa-python/Scripts/python <path-to-GEMSpa>/main_gui.py
```

NOTE: If you encounter errors when trying to run these commands, try enclosing the command in quotation marks. For example:

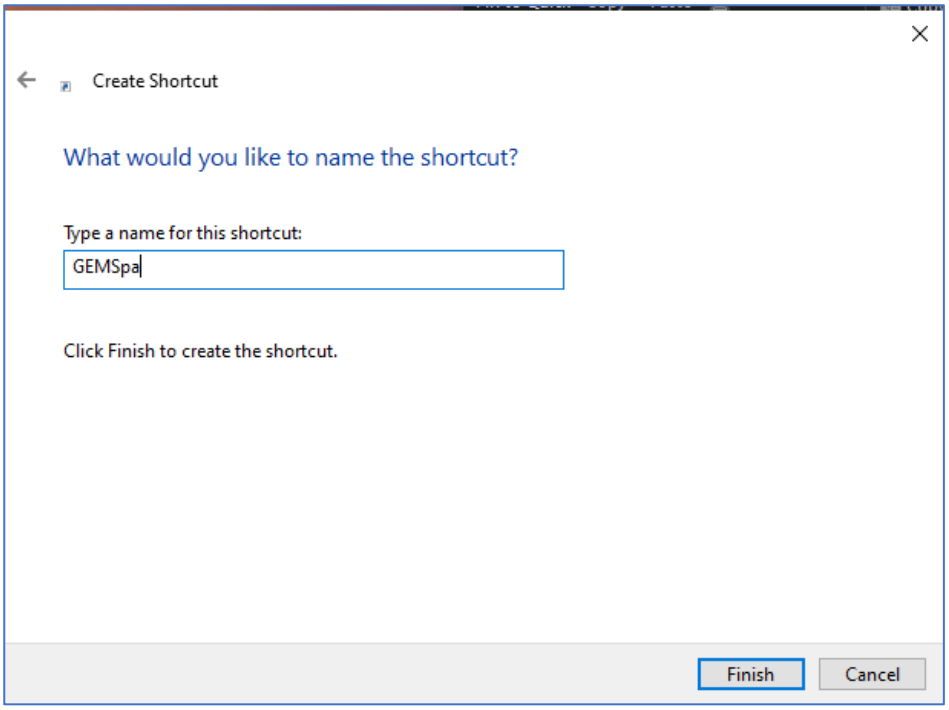
```
"gemspa-python/Scripts/python" <path-to-GEMSpa>/main_gui.py
```

iv) You can also create a Desktop Shortcut so you don't have to open Terminal and type in the commands manually:

Right-click on the Desktop and Select New->Shortcut. In the Text Box, enter the location of the Command Prompt: C:\Windows\System32\cmd.exe



Give the shortcut a name, e.g. GEMSpa, and then click Finish.

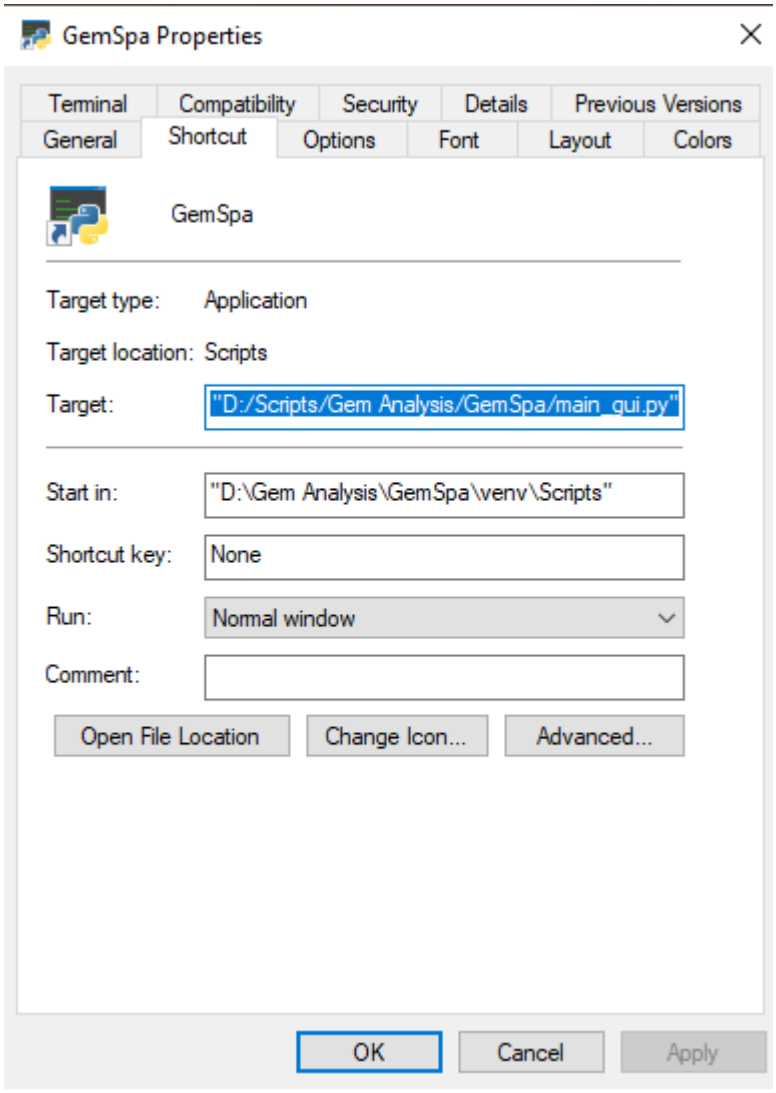


The Shortcut will be created on the Desktop. Right click on it and select Properties.

In the **Target** text box, enter `<path-to-GEMSpa>/main_gui.py`
(`<path-to-GEMSpa>` is the full path to the location where you downloaded GEMSpa.)

In the **Start In** text box, enter the path to the python executable that you created above:
`C:/<current-dir>/gemspa-python/Scripts`

See below for an example:

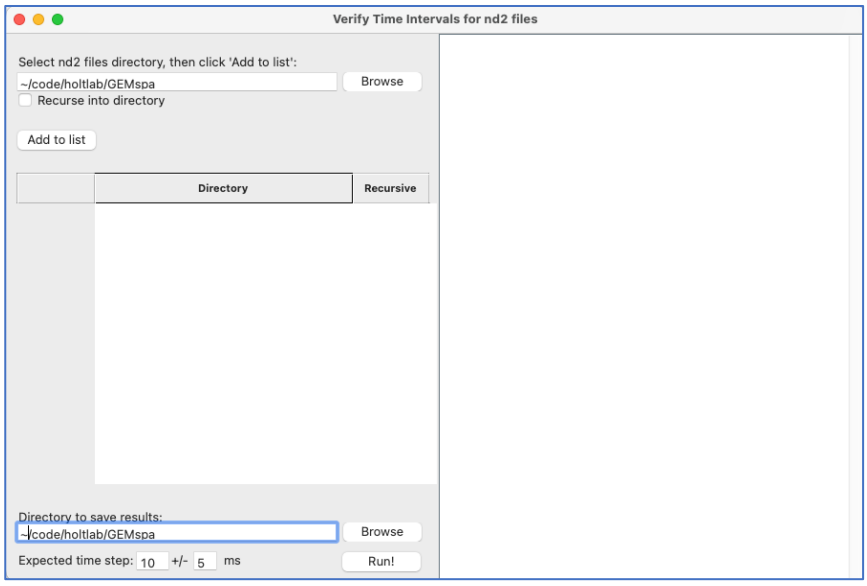

GemSpa Properties
Terminal
Compatibility
Security
Details
Previous Versions
General
Shortcut
Options
Font
Layout
Colors
GemSpa
Target type: Application
Target location: Scripts
Target: "D:/Scripts/Gem Analysis/GemSpa/main_gui.py"
Start in: "D:\Gem Analysis\GemSpa\venv\Scripts"
Shortcut key: None
Run: Normal window
Comment:
Open File Location
Change Icon...
Advanced...
OK
Cancel
Apply

SECTION 6: TIME INTERVAL CHECKING FOR ND2 MOVIE FILES

The GEMspa repository includes a utility program for verifying the time step (frame rate) of nd2 movie files. This script will use the python package nd2reader (https://github.com/Open-Science-Tools/nd2reader) to pull the metadata and read the value "sampling interval" (metadata->experiment->loops[0]->sampling interval), as well as the time information for each frame of the movie (the "timesteps" array). The script will check that the "sampling interval" matches the expected time step (user input) and also, that the time interval between each frame is consistent and matches this value. If any errors are found, output is printed both on screen and to a text file. The program input allows multiple directories of nd2 files to be verified all in one run.

This program may be run under the GEMspa environment (see GEMspa installation instructions), e.g.:

```
python <path-to-GEMSpa>/test_time_interval.py
```



1) Choose each of the directories where your nd2 files are located and click "Add to list."
2) Enter the expected time step (e.g., 10ms). (+/- 5ms refers to the resolution, e.g., as shown above, any deviation > 5ms from the expected time step of 10ms will result in an error message printed to the output.)

All values are rounded to the nearest millisecond (ms) (i.e., lowest resolution is 1ms)