**SpectroView User Guide**

Welcome to SpectroView! This is an application that is built for finding patterns and potential species in an electromagnetic spectrum in a number of ways, packaged in a GUI to make this process quick and easy. There are a number of features in this program that can be used to help.

If you have any additional questions about the program, feel free to contact the owner at [kylegilbert@csus.edu](mailto:kylegilbert@csus.edu) with the subject “SpectroView”.

**Table of Contents**

Titles:

* [Installing and Running](#lj8mcvd0jd4b)
* [Important Notes](#7n1sh3l3h5v3)
* [Basic Usage](#s09s285h4ygz)
* [Step-By-Step Guides](#ufpnto7w7rve)

Subjects:

* [Loading a File](#o7ge7usao8be)
* [Initializing a Dataset](#fr9t5537ih7)
* [Peak Picking](#xxzjcvwlet49)
* [Exporting](#ho7h3pt8lj9c)
* [Merging Datasets](#llol7pwf68ox)
* [I2 Plot](#mop7041y55k1)
* [Zooming](#z7k613palkwj)

**Installing and Running**

<https://github.com/ArchonX64/spectro-view>

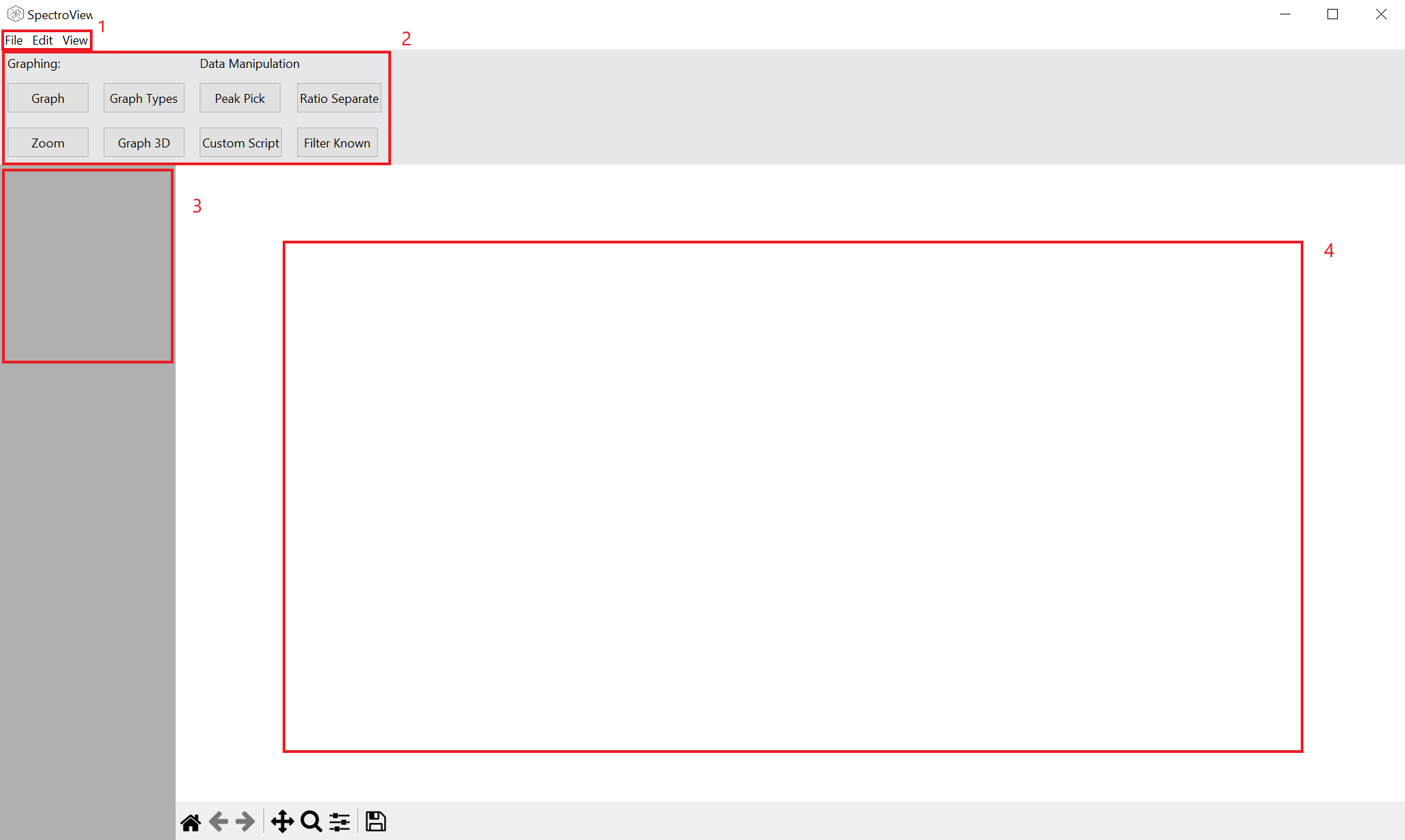
1. SpectroView is built entirely in Python 3.9, so make sure that you have that version of Python installed on your system.
2. Download the entire contents of the GitHub repository listed above and extract it into the destination of your choice. These instructions will refer to this location as the “SpectroView folder”.
   1. This is a private repository, so you may need to request access. Contact the owner at [kylegilbert@csus.edu](mailto:kylegilbert@csus.edu) or someone who can ask the owner to submit your GitHub username.
3. SpectroView requires a number of Python dependencies that will need to be downloaded:
   1. tkinter (included with most Python interpreters)
   2. numpy (included with install of pandas)
   3. matplotlib
   4. pandas
   5. openpyxl
   6. xlsx2csv
   7. scipy
4. You have two options of running this program:
   1. Python Interpreter: This way will be simplest to set up and is recommended if you plan on modifying the code, however it may take longer to load the application each time.
      1. Find the location of your python interpreter and the “main.py” file from the SpectroView folder
      2. Run the python interpreter with the location of “main.py” as the only argument in your command prompt.
   2. Creating an executable: This will allow you to bundle up everything relating to this program into one executable. This takes longer to set up, but is easy to distribute and easy to load.
      1. Find and install PyInstaller (<https://pypi.org/project/pyinstaller/>)
      2. Locate the executable “pyinstaller” in the pyinstaller folder (NOT the .py file!).
      3. Run the executable in a command prompt with the location of the “main.spec” file from the SpectroView folder. This process will take a minute or two.
      4. Once completed, a new folder named “dist” will have been created in the SpectroView folder. Your executable will be located inside, which can be used in any location.
      5. This can be opened like any other executable.
5. If completed correctly, a GUI should open. If one does not or an error message appears, retry the steps or feel free to contact the owner about a problem.

**Important Notes**

* SpectroView is not designed to create scientifically accurate datasets. It uses a number of shortcuts in order to make the data easier to analyze. While the graphs can be used to show accurate patterns, be wary when using data outputted from this program as experimental data.
* SpectroView does not modify the data files it opens, it creates a copy of whatever it reads. There is no need to worry about the original data being modified in a way you do not know about. However, be wary of your file names to prevent accidental overwriting.
* SpectroView is very much a work-in-progress program and there will be bugs and unfinished features.

**Basic Usage**

Now that you are able to run SpectroView, you should have a GUI that looks like this.

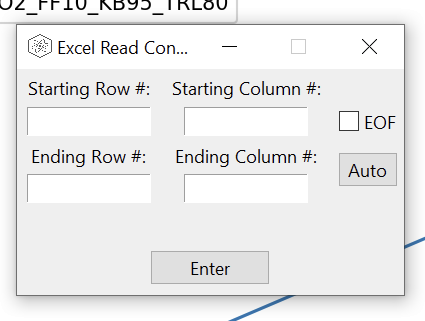


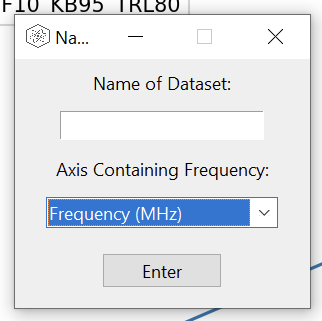
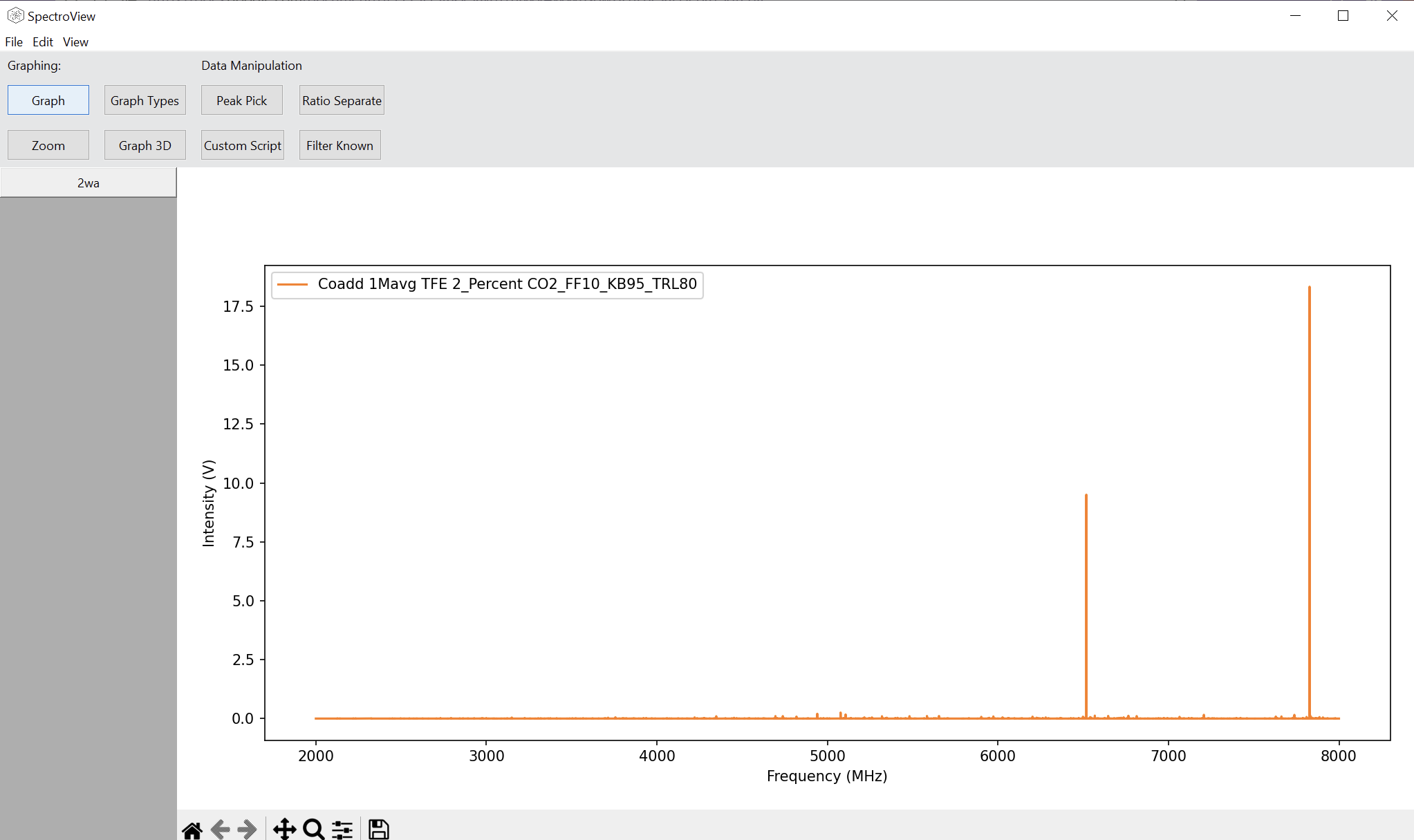
Here is a basic overview of all of the visible features, which will be gone over in more detail later.

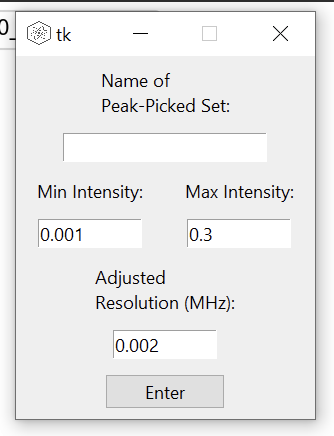
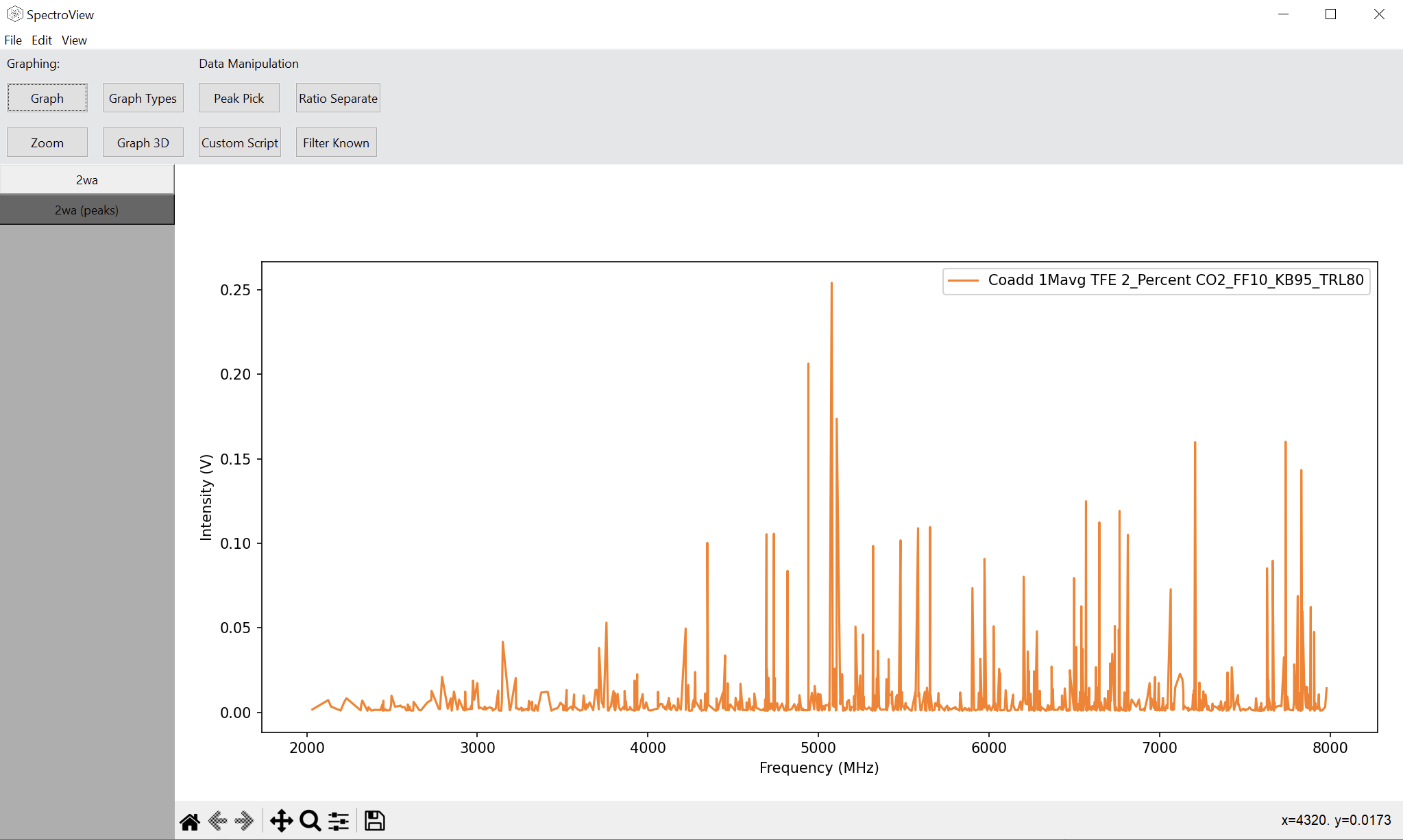
1. This is the primary menubar, which contains most of the file modification and data configuration for the program.
   1. **File:**
      1. **Open:** Allows you to select a data file to open, which can be from a number of sources. Some file types will have additional windows pop up in order to properly extract data from.
      2. **Export Dataset:** Allows you to export a selected dataset into a file, such as a CSV.
      3. **Export Graph**: Allows you to export the current graph shown as a .png file
      4. **New File Association:** Allows you to run a custom Python script to load data from unsupported file types.
      5. **Exit:** Quits the program.
   2. **Edit**
      1. **Info:** Allows you to modify the way SpectroView interacts with the data, such as the name, x-axis, and frequency axis
      2. **Data:** Allows you to modify the actual data, such as removing target columns, removing values with a condition and modifying the values.
   3. **View**
      1. **Graph Size:** Allows you to manipulate the size of the graph on the screen. The graph should normally scale with the size of the window, but this can work to do it manually if it stops working.
2. This area contains the main buttons that will be most commonly used to shape the data and graphs to help identify patterns
   1. **Graph:** Graphs the currently selected dataset.
   2. **Zoom:** Allows you to change how much the graph is zoomed in. You can either input certain values to graph to, or have the program automatically fit the scale.
      1. Note: Leaving a box empty will tell the program to leave that border at the same value it is currently.
   3. **Graph Types:** Allows you to change how data points are displayed on the graph.
      1. Line: Each value is connected to one another through a line. Useful for full spectra. Least graphically intensive
      2. Scatter: Each value is displayed as a single point. May be slow to load on large datasets
      3. Stem: Displays every data point as a line going from the x-axis to the location of the point. Useful for theoretical spectra or spectra that contain only peaks.
      4. None: Tells the program not to graph that axis.
   4. **Graph 3D:** Allows you to plot on a three-dimensional graph. Three different axes must be chosen in order to plot.
      1. Note: Is very graphically intensive, not recommended with plots with > 10,000 data points.
   5. **Peak Pick:** Allows you to eliminate all points except for the peaks of lines that fall within a certain range.
   6. **Custom Script:** Allows you to run a custom python script on a dataset that will create a new dataset with modifications. Must contain a function that returns a pandas DataFrame, and takes a pandas DataFrame, name of the x-axis, and specified column as parameters.
   7. **Ratio Separate:** Allows you to generate ratios between two different axes in a new axes, and then to optionally remove all ratios between a certain range.
   8. **Filter Known:** Allows you to remove all parts of a certain dataset that are within a certain frequency range of another dataset
      1. Note: Both datasets must only contain peaks for this to work correctly.
3. This area will contain all of the datasets that you have currently loaded, along with their names. Selected datasets will be darker gray.
   1. **Right Clicking on a Dataset:**
      1. **Replicate:** Allows you to copy the dataset into a new one. Will require a new name for the dataset.
      2. **Merge:** Allows you to merge with another dataset along the frequency axis. Also includes an option to combine values that are within a certain frequency.
      3. **Split:** Allows you to take certain axes from a dataset and move them into a new dataset. The frequency values will stay the same for each.
      4. **Delete:** Deletes the dataset from the program.
4. This is the area where datasets will be graphed. You have a few options for interacting with this graph.
   1. **Key Commands**: This program takes some keyboard inputs that are similar to those found in the AABS family of programs. These include:
      1. **A / S:** Moving left and right
      2. **K / L:** Moving left and right slowly
      3. **2 / 3:** Moving up and down slowly
      4. **Q / E:** Zooming out / in, horizontally
      5. **W / Z:** Zooming in / out, vertically
   2. **Matplotlib Toolbar:** This toolbar give you the options (in order) to:
      1. Reset Original View
      2. Cycle Between Views
      3. ^^
      4. Pan in the graph
      5. Zoom into a selected rectangle
      6. Configure how the graph is viewed
      7. Save an image of the figure
      8. IMPORTANT:
         1. This toolbar is independent of the rest of the program. Any changes made with this toolbar will be ignored when doing other operations in SpectroView

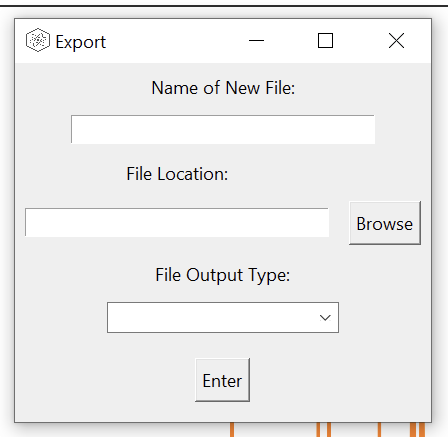
**Step-By-Step Guides**

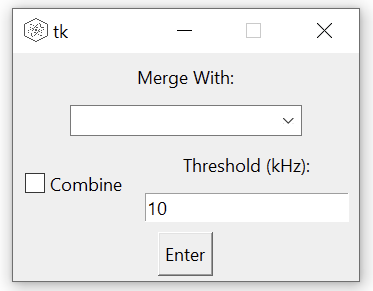
1. Loading a CSV file and exporting the peaks of it.

* 1. In order to start, we will have to load the file. Go to the menu bar at the top of the screen and go to File > Open, and select the .csv file you wish to open.
     1. Note: This process is the same for any file type, with the exception of steps b and c being CSV file specific.
  2. This screen will appear when a file is selected:  
     
  3. You have three options with this window:
     1. Take only certain values from file: input the starting row / column for all four boxes and click enter:
     2. Only certain columns, but everything in those columns: Input all boxes except for the ending row, select EOF, and click enter. This will allow you to select an infinite amount of data for some columns while avoiding others
     3. All of the file: Simply click Auto. This will be appropriate for most situations.

* 1. After this, a new window will pop up:  
     
  2. The two options:
     1. Name of Dataset: The name as it will appear in the program
        1. If left blank, the name of the file will be used.
     2. Axis Containing Frequency: The column label that indicates the frequency column.
        1. This is autogenerated for most inputs.
        2. IMPORTANT: Make sure that this option is correct before anything else. Many operations rely on the frequency axis.
  3. After this, the dataset will appear in the left hand bar. By selecting the dataset, you can click “Graph” in the header to graph the dataset. Your screen should look similar to this (the graph will look different):  
     

* 1. In order to make a peak pick, you will need to select the dataset again, so that it is dark gray. Then, click the “Peak Pick” button from the header. A screen like this should appear:  
     
  2. The options:
     1. Name of Peak-Picked Set: If left blank, the name of the original dataset will be used with “(peaks)” added onto the end.
     2. Min / Max Intensity: The intensity ranges that the program will look for peaks in.
        1. NOTE: The values automatically generated may not be accurate for your data. Make sure to analyze your data to determine your own min / max in order to create an accurate peak pick
        2. COMING SOON: A data file that will save user selections
     3. Adjusted Resolution: The peak picking program uses a cubic spline to adjust the data to a certain resolution for more accurate peak selection. The default, 2 kHz, should be adequate for most cases.
  3. Click “Enter” to generate a new peak-picked spectrum. You can graph it to make sure that is correct:  
     

* 1. Exporting: Make sure you have the peak picked set chosen and navigate to the menu bar to File > Export Dataset. Selecting this will bring up this window:  
     
  2. The Options:
     1. Name of New File: What you want the file to be called
     2. File Location: Where you want the file to be saved to. You can either type in a directory or select one from “Browse”
     3. File Output Type: Select what type of file should be created from the dataset.
  3. Click “Enter” and you now should have a file complete with all of the peaks from your dataset!

1. Merging two datasets
   1. In order to be able to access this function, you will need two separate datasets loaded into SpectroView.
   2. Once loaded, right click one of the datasets and go to “Merge”
   3. This screen should appear:  
      
   4. Under the “Merge With” column, select the additional dataset you would like to merge with.
   5. “Combine” allows you to have points that are close enough in frequency to a certain threshold to be treated as the same frequency. This is important for when you need operations that depend on correlating the data of different graphs. Combine should ONLY be used on datasets containing only peaks.
   6. For example:

| **Frequency** | **Data 1** | **Data 2** |
| --- | --- | --- |
| 15678.2345 | 0.000051443 |  |
| 15679.2394 |  | 0.0009273 |
| 15679.353 | 0.0001332 |  |

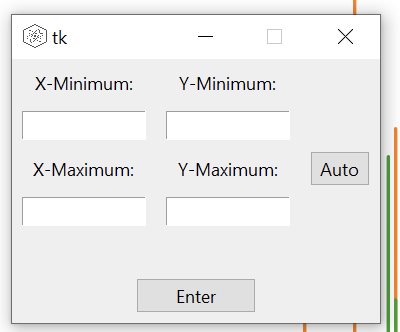
Would be converted to this:

| **Frequency (MHz)** | **Data 1** | **Data 2** |
| --- | --- | --- |
| 15678.2345 | 0.000051443 | 0.0009273 |
| 15679.353 | 0.0001332 |  |

As you can see, some frequency data is lost so make sure not to treat this as experimental data.

* 1. Once you have your options selected, press “Enter” and a new dataset should be formed, that includes the name of the dataset that was right clicked with a “+” followed by the name of the dataset selected in the menu.

1. Making an I2 plot
   1. The first thing that you will need is a singular dataset that has two different columns. Having a CSV file with two columns can be loaded, or you can follow the steps above to merge two individual datasets.
   2. It is highly recommended to do a “Combine” enabled merge, as this process relies on having points with the same frequency.
   3. In order to make an I2 plot, you must make the x-axis the intensities of one of the datasets, while the y-axis contains the intensities of the other (this one is default when a dataset is loaded in).
   4. In order to change the x-axis:
      1. Select the dataset
      2. At the top of your screen go to Edit > Info
      3. Under x-axis, change it to the first dataset, assuming you are using the second dataset as the y-axis.
   5. Then, simply click plot to see the I2 plot.
   6. You may see an empty screen. This is because it is still on the same scale as it was for plotting frequency vs. intensity, so you will have to adjust the zoom:
      1. Select the dataset
      2. In the header containing the main buttons, click “Zoom”

* + 1. A screen like this should pop up:  
       
    2. There are two options:
       1. Manual
          1. If you know the exact numbers that you want to zoom into, simply put them into the boxes and click “Enter”
          2. NOTE: Any boxes left blank will cause those limits to not change.
       2. Automatic:
          1. By clicking “Auto”, the graph will automatically size the graph to include every point. This is usually the best option.
  1. You should now be able to see your I2 plot.

1. Saving all points that are within a certain ratio range
   1. To start off with, you will need a dataset that has at least two sets of intensities.
      1. It is recommended to use the “Combine” button if you are meringing, as this process will require