**Reg’d packages:**

Pandas, numpy, openpyxl, pyqt6, scipy, matplotlib, seaborn, csv,

**Plot TD-DFT workflow**

Before any functionality happens, read in user defined data from variable names, specifically  
  
directory = r’sample/directory’  
input\_unit = wavelength # can be wavelength (nm), wavenumber (cm-1), or eV  
output\_unit = wavelength # can be wavelength (nm), wavenumber (cm-1), or eV  
shift = 20 #must be a number or float   
shift\_unit = wavelength # can be wavelength (nm), wavenumber (cm-1), or eV  
output\_lowervalue: 200  
output\_uppervalue: 500  
output\_spacing: 1

1. **Functionality 1**: For each .spectrum.rootN file in a directory:
   * Extract Energy and TotalSpectrum data.
   * Interpolate this data onto a common grid (range of min(Energy) – max(Energy), increment of 1).
   * Convert the Energy column from the input\_unit to the shift\_unit using a conversion factor defined by an imported dictionary called constants:  
     from constants import constants as c  
       
     for wavelength to wavelength: energy \* c[‘wl2wl’]  
     for wavelength to wavenumber: energy \* c[‘wl2cm-1’]  
     for wavelength to eV: energy \* c[‘wl2eV’]  
       
     for wavenumber to wavelength: energy \* c[‘cm-12wl’]  
     for wavenumber to wavenumber: energy \* c[‘cm-12cm-1’]  
     for wavenumber to eV: energy \* c[‘cm-12eV’]  
       
     for eV to wavelength: energy \* c[‘eV2wl’]  
     for eV to wavenumber: energy \* c[‘eV2cm-1’]  
     for eV to eV: energy \* c[‘eV2eV’]
   * Add the shift to each interpolated energy
   * Convert the shifted interpolated energies to the output\_unit using the same conversion workflow from the imported dictionary called constants
   * Interpolate the shifted, interpolated data onto a new x grid defined by min, max, spacing with the format: (output\_lowervalue, output\_uppervalue, output\_spacing)
   * Write this final interpolated data for each root file into an Excel sheet with headers energy, root1, root2, root3, etc.
   * Each unique filename forms a new sheet in the Excel file.
2. **Functionality 2**: For each .spectrum file in the same directory:
   * Extract Energy and TotalSpectrum data as above.
   * Interpolate this data onto a common grid (range of min(Energy) – max(Energy), increment of 1).
   * Convert the Energy column from the input\_unit to the shift\_unit using a conversion factor defined by an imported dictionary called constants:  
     from constants import constants as c  
       
     for wavelength to wavelength: energy \* c[‘wl2wl’]  
     for wavelength to wavenumber: energy \* c[‘wl2cm-1’]  
     for wavelength to eV: energy \* c[‘wl2eV’]  
       
     for wavenumber to wavelength: energy \* c[‘cm-12wl’]  
     for wavenumber to wavenumber: energy \* c[‘cm-12cm-1’]  
     for wavenumber to eV: energy \* c[‘cm-12eV’]  
       
     for eV to wavelength: energy \* c[‘eV2wl’]  
     for eV to wavenumber: energy \* c[‘eV2cm-1’]  
     for eV to eV: energy \* c[‘eV2eV’]
   * Add the shift to each interpolated energy
   * Convert the shifted interpolated energies to the output\_unit using the same conversion workflow from the imported dictionary called constants
   * Interpolate the shifted, interpolated data onto a new x grid defined by min, max, spacing with the format: (output\_lowervalue, output\_uppervalue, output\_spacing)
   * Write interpolated data into a new Excel file summary.xlsx.
   * Headers are energy, molecule\_a spectrum, molecule\_b spectrum, etc, where each column is a different .spectrum file.