CHEM E 546 FINAL PRESENTATION

ProSpecPy

BACKGROUND

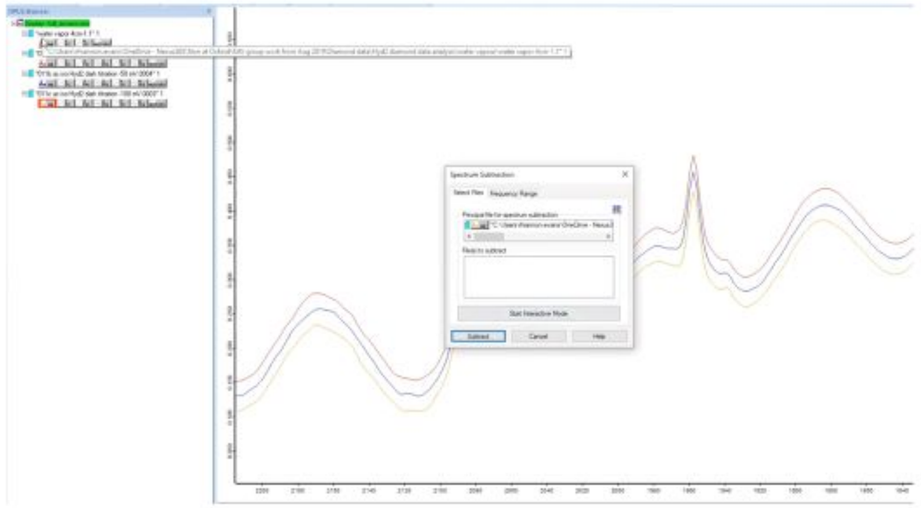
The goal of our project is to produce a python package that is able to efficiently analyze fourier-transform infrared spectroscopy (FTIR) data.

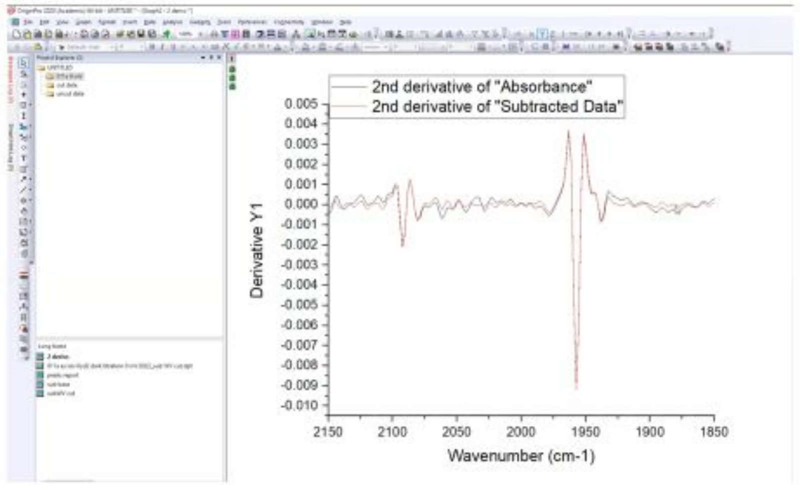
BACKGROUND

1.5 hrs per reading

~Steps

1. Subtracting water vapor
2. Cutting wavenumbers to range of interest
3. Plot data
4. Take second derivative (of absorbance) and plot
5. Add individual points to define baseline – relying on points identified from the second derivative plot
6. Fit peaks from your baseline-subtracted data
7. Plot subtracted baseline data





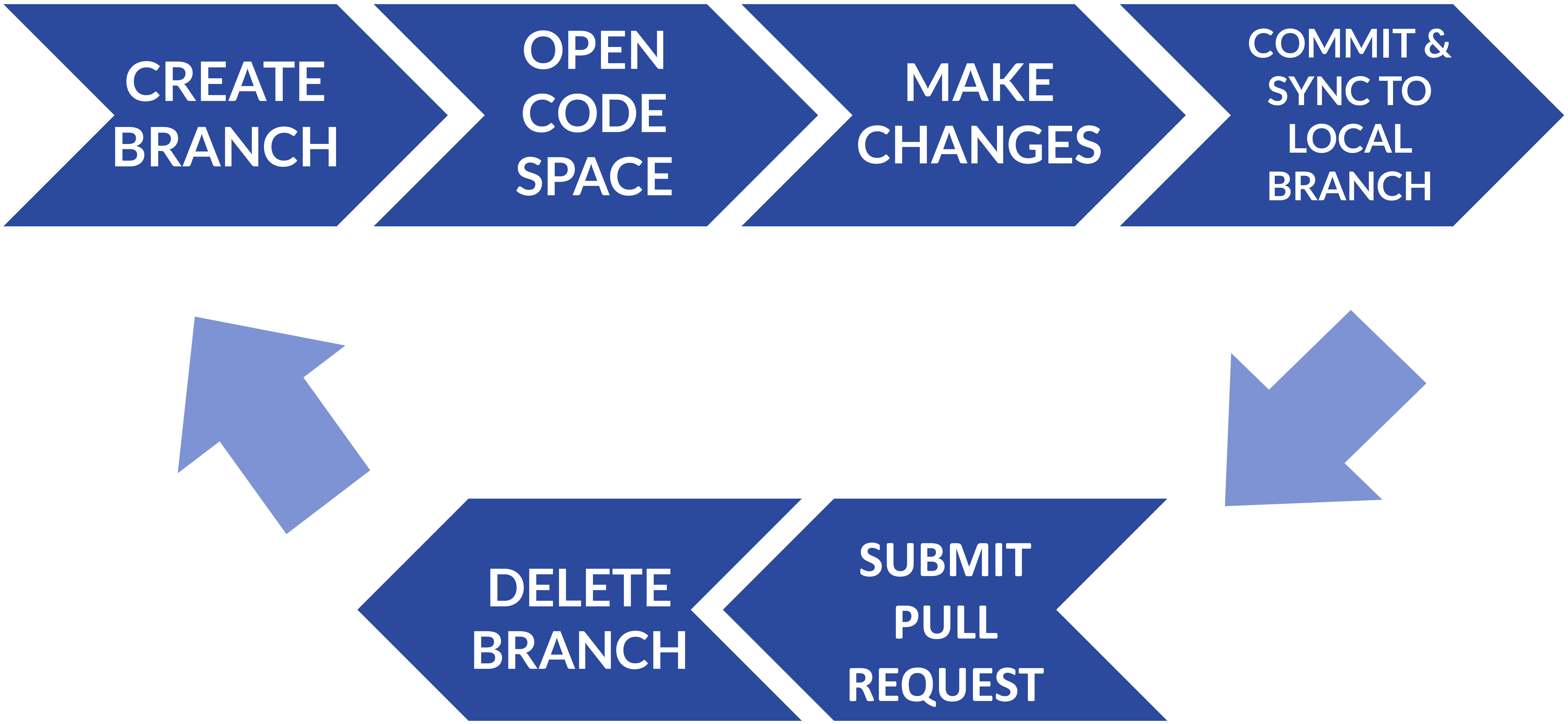
OBJECTIVES:

- ☒ SUBTRACT WATER VAPOR.
- ☒ CUT WAVENUMBERS TO ROI.

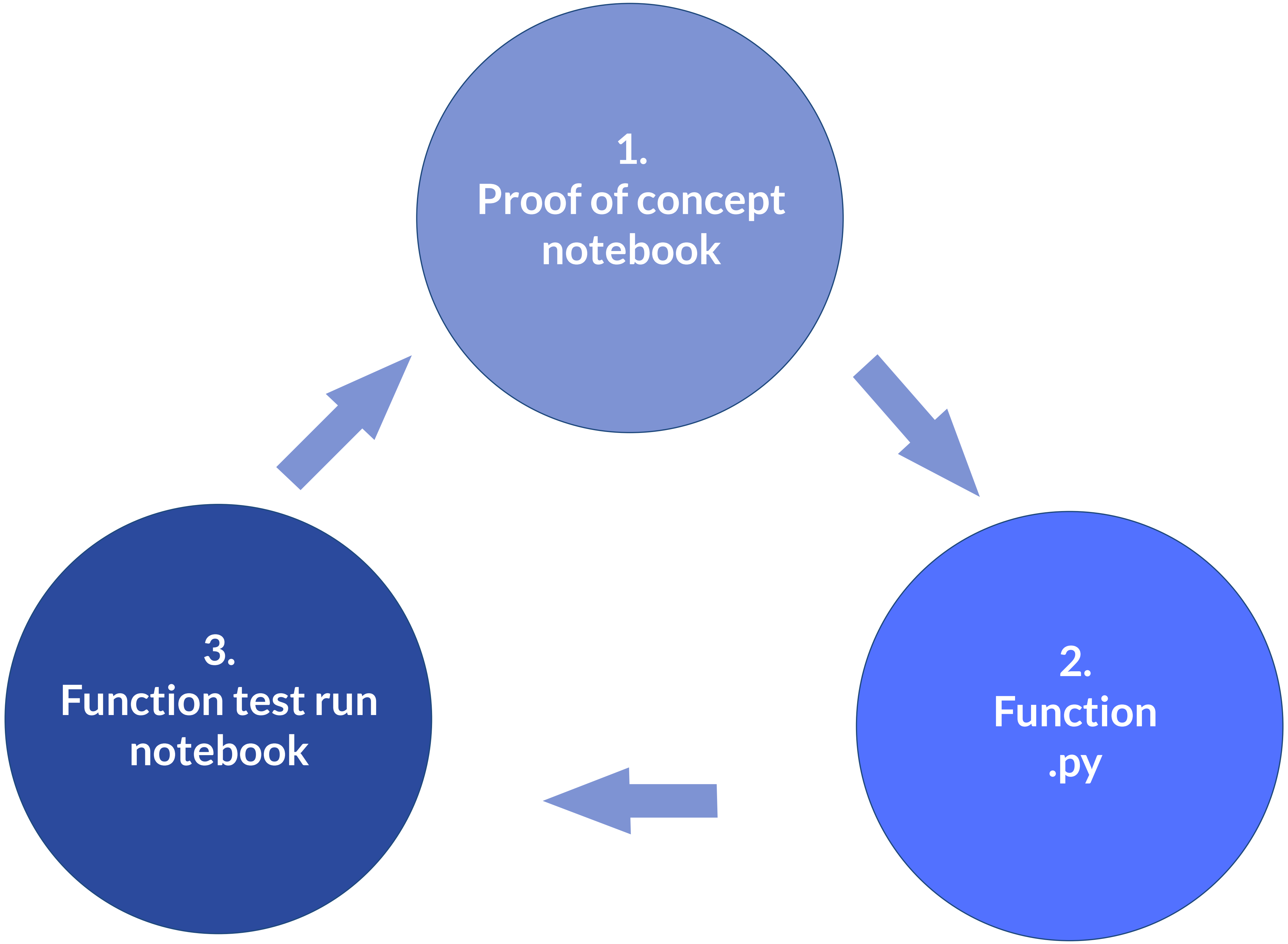
- ☒ PLOT DATA
- ☒ TAKE 2ND DERIVATIVE AND PLOT
- ❑ ADD INDIVID. PTS TO DEFINE BASELINE.

- ❑ FIT PEAKS FROM BASELINE-SUBTRACTED DATA
- ❑ PLOT SUBTRACTED BASELINE DATA

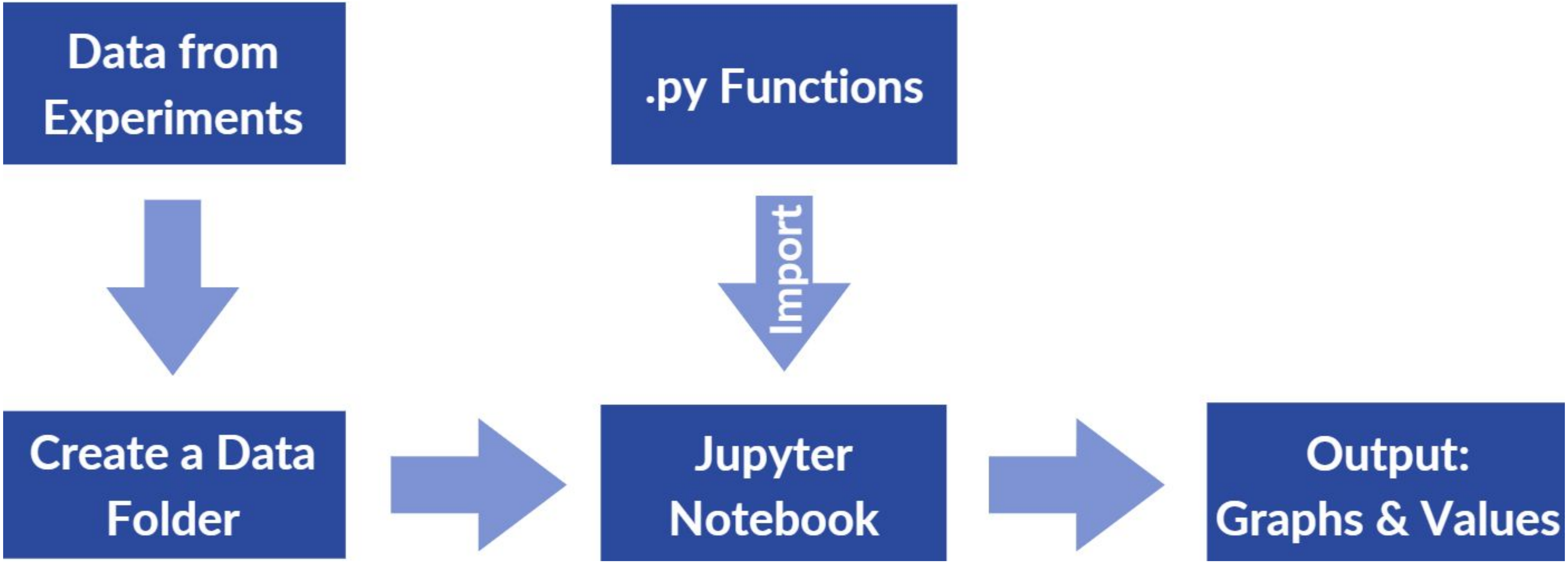
TEAM WORKFLOW



DEVELOPMENT WORKFLOW



ANALYSIS WORKFLOW



1

OBJECTIVE

SUBTRACTING WATER VAPOR

OBJECTIVE 1

CURRENT STATUS:

Working function which subtracts the water vapor spectrum.

CHALLENGES:

Robustness is insufficient due to limited water vapor data.

NEXT STEPS:

Gain more water vapor data to improve accuracy.

FUNCTION 1

Remove_wv.py performs water vapor subtraction on raw spectral data. It interpolates raw spectral data and water vapor spectrum to a common grid. Water vapor is subtracted to account for atmospheric noise.

2

OBJECTIVE

CUTTING WAVENUMBERS TO RANGE OF INTEREST

OBJECTIVE 2

CURRENT STATUS: Complete with no bugs.

FUNCTION 2

Cut_range.py cuts the specified range from raw spectra data and performs atmospheric subtraction.

3

OBJECTIVE

PLOT DATA

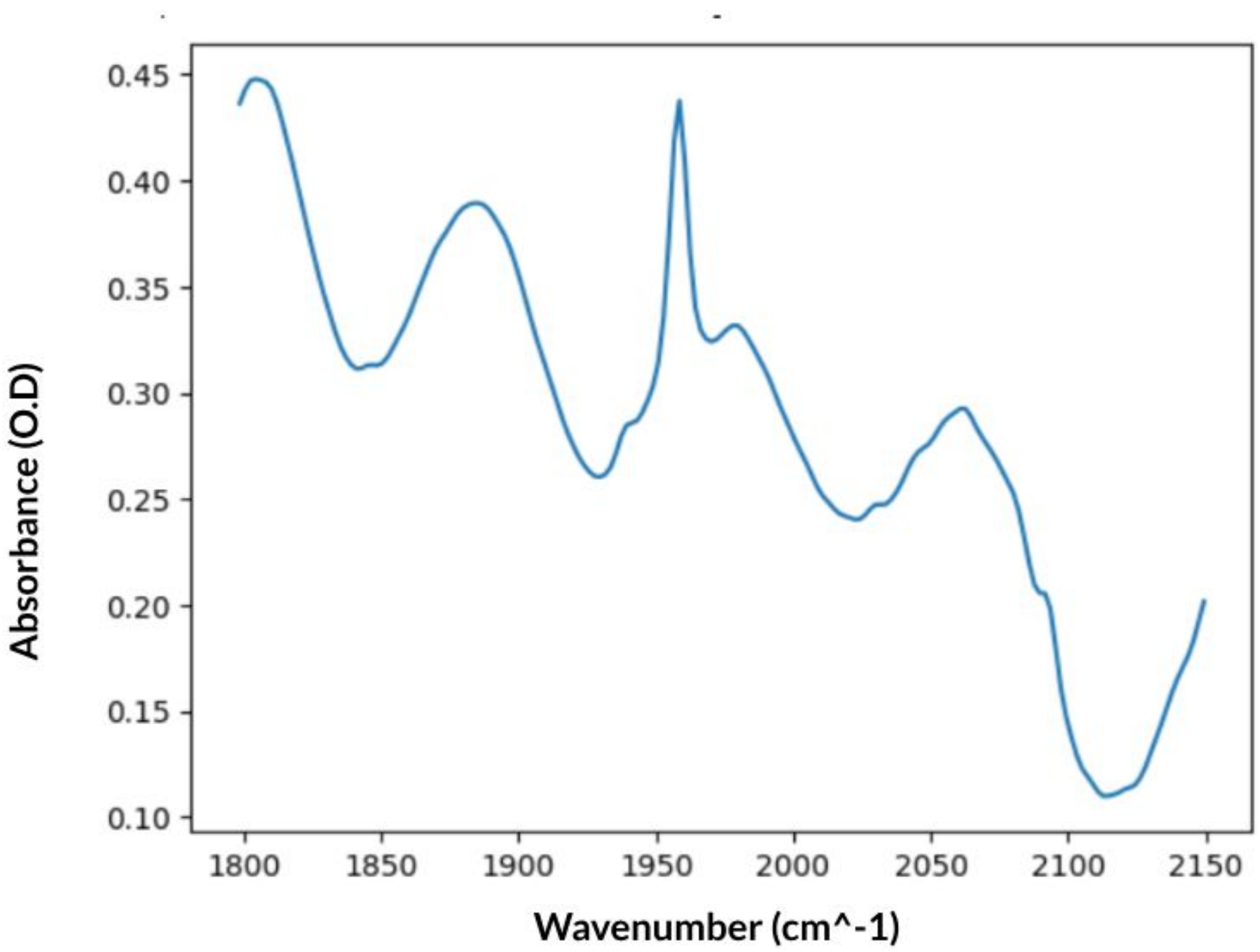
OBJECTIVE 3

CURRENT STATUS:

Complete with no bugs.

NEXT STEPS:

Get user input on plot visuals and change accordingly.



4

OBJECTIVE

TAKE 2ND DERIVATIVE & PLOT

OBJECTIVE 4

CURRENT STATUS:

Second derivative plotted, visually identical to the example results shown in methodology video, second derivative peak identified and plotted.

CHALLENGES:

When to bring in the spline function

NEXT STEPS:

Running more data through it to see how the function responds to various sets of spectra data.

FUNCTION 3

Second_deriv.py takes the data set, gives spline function of 2nd derivative and returns two relevant variables to allow for future peak finding and plotting of the second derivative.

REMAINING OBJECTIVES/ NEXT STEPS

Feature details:
- Add individual points to address baseline
- Fit peaks from baseline-subtracted data
- Plot subtracted baseline data

General:
- More user stories to improve the applicability of package
- Add test for peak x values
- Make our own variable class to minimized package complexity

Thank you!

Questions?