# in-situ-ftir

**# Overview**

This repository contains scripts for the fitting of gaussians to convoluted FTIR spectra. It was specifically developed to handle a time-series of in situ ATR-FTIR spectra, split the spectra into its individual gaussians, and track those peaks as they shift in intensity and energy. Along with the actual fitting script, this repository contains several support scripts for data analysis and recall of prior fit results. If you see a gap in the functionality of this suite, please contact me! I'd happy to work with you to get that functionality added.

**# Start-up Guide**

The following section is a guide for getting started. You will need the following packages: pandas, numpy, matplotlib, lmfit, natsort, and scipy. All are easily installed using pip install and have extensive documentation. **<All packages have been installed using the pip install command.>**

**## IR\_Model\_SingleSpecFit**

This script is for the fitting of gaussians to a single FTIR spectrum. It assumes the data is a two-column text file with no header. The user must define the inputs under "User Inputs". This script returns a plot of the gaussians, the fit line and the original data. Output parameters are stored as workspace variables. This script is mostly useful to get an idea of what peaks are present in your ROI before running the script that iterates over the entire in situ dataset.

![singlespec\_demo](https://user-images.githubusercontent.com/87740914/131015326-4e6e1f23-dd37-4ad4-9b34-ba37859c291c.png)

not the greatest fit, but you get the idea.

**<Got it running; had to fix a value error where there were too few outputs being asked for from a function that puts out 6 outputs. Interested in improving the fit.>**

**## IR\_Model\_BaselineTest**

This script uses a convex hull fitting algorithm to baseline a single spectrum and plots the baselined spectrum on the same plot as the raw data. The entire spectrum or a specific region of the spectrum can treated. This is useful to get an idea of how the fitting algorithm affects your data before batch processing the entire dataset. This script also allows the user to input another filepath for use as a reference (i.e. a known baselined spectrum). All spectra are normalized to 0<y<1.

**<Got it running. Did not have to make any fixes.>**

**## IR\_Model\_BaselineCorrect**

This script takes a folder with your data in it, baselines every spectrum in that folder and saves that data to a new folder in the same parent folder. For example, if the path to the folder with your data in it is 'C:/user/folder' the script will create a new folder 'C:/user/folder\_BLSub' and save the baselined spectra there. You can baseline the entire spectrum or specify a region

**<Worked well. Did not have to change anything.>**

**## IR\_Model\_MultiSpecFit**

This script is the workhorse of this repository. Feed it a folder containing all of your in situ data files, fill out the user inputs, and let it run. As the script iterates through the desired portion of the dataset, it will output a plot like the one above and at the end of the fitting, it will store the results of the fit to the user-specified folder. The annotated inputs are pictured below.

![image](<https://user-images.githubusercontent.com/87740914/131021453-f1775664-fb65-4fcc-9778-4a0f4ffe2d2e.png>)

**<Worked well. Just changed filenames.>**

**## IR\_Model\_HeatMap**

This script takes the folder of previous fit results and plots a heat map of the desired component intensities.

![heatmap\_demo](https://user-images.githubusercontent.com/87740914/131033288-878a9a86-c692-4358-beb8-ef793381c3b0.png)

**<Worked well. Only changed component name.>**

**## IR\_Model\_RawMultiPlot**

This script takes the insitu dataset from the fitting output result and plots one or many raw spectra on the same plot. This is useful to get an idea of the general progression of your peaks.

![rawmultiplot\_demo](https://user-images.githubusercontent.com/87740914/131038571-68292644-8663-43d6-ad4a-e5e824add150.png)

**<This one was a pain to edit. The inputs that determine which spectra to plot and how many on each plot, etc. are hard to understand but they are annotated now.>**

**## IR\_Model\_PlotPriorFit**

This script takes the fitting output result folder and plots all the fitting outputs in order. Careful, this outputs a lot of plots.

**<Worked well. I just used the file path for the fitting output folder.>**

**## IR\_Model\_PlotFitAreasCentersMaxima**

This script does exactly what its name says: plots area, center, maxima of prior fit result for specified components. Useful for assessing changes in component areas, centers, maxima over a run.

![PlotFitAreasCentersMaxima\_demo](https://user-images.githubusercontent.com/87740914/131039457-e2120405-e167-4922-8670-cc8728bd594a.png)

**<Worked well. I just used the file path for the fitting output folder.>**

## IR\_Model\_PlotFitAreasCentersMaximaWithVoltage

This scipt is the same as above, except voltage is overlayed onto each plot. Extra inputs are the absoulte path to the echem file (MACCOR export as tab separated .txt).

## IR\_Model\_PlotFitAreasCentersMaxima\_vs\_Voltage

This scipt is the similar to above, except voltage is plotted on the abscissa. Time is shown by the color of the data marker. Extra inputs are the absoulte path to the echem file (MACCOR export as tab separated .txt).

![PlotFitAreasCentersMaxima\_vs\_Voltage\_demo](https://user-images.githubusercontent.com/87740914/131042547-37920107-26a0-4c9a-abff-ad1823897f5a.png)

**<Have not tried these yet. Do not have voltage files from the Maccor.>**

## IR\_Model\_LoadPriorFit

A simple script to load prior fitting results as local variables. Not that useful.

## IR\_Model\_ErrorPlot

A script that plots the error of the fit versus run number. There are MSE, RMSE, and RMSE normaized by the interquartile range (nRMSE) of the raw data used for the fitting, allowing for apples-to-apples comparison among fits. The nRMSE is preferred as it can be thought of as similar to percent error.

![errorplot\_demo](https://user-images.githubusercontent.com/87740914/131384484-d9e8342b-bc97-4762-998a-e8eecd1f004f.png)

This plot shows the fit going from very bad to not-quite-so-bad-but-still-not-good

**<Worked well. Just changed the input.>**