Appendix A

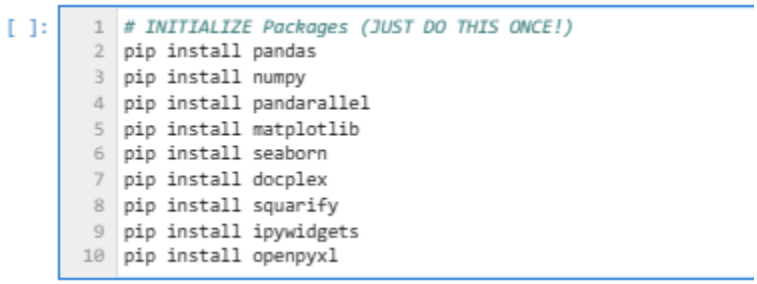
Download the Jupyter Notebook file called "DynamicPredictedMineralogyForDistribution.ipynb" here: https://github.com/charraden/DynamicPredictedMineralogy.git

Python package prerequisites (user to install): pandas, numpy, pandarallel, matplotlib, seaborn, docplex, squarify, ipywidgets, and openpyxl. The first block of the Jypyter notebook contains the pip install code for these packages (just run this cell). NOTE: DOCPLEX can be a bit tricky, but there is good documentation online to help troubleshoot.

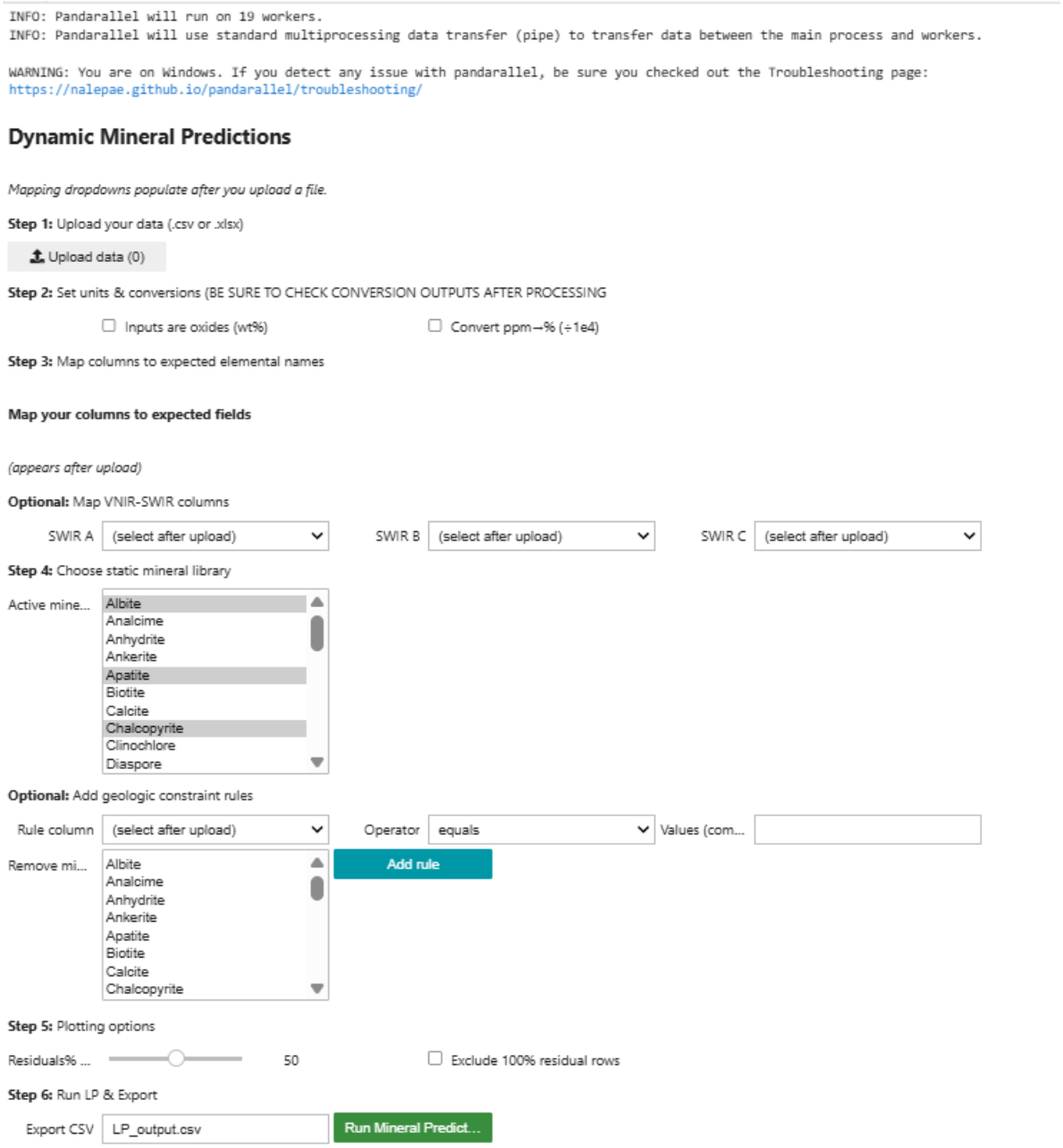
This Jupyter notebook contains Python code to execute the predcited mineralogy using VNIR-SWIR and user-defined geological constraints to dynamically change the mineral library for the linear programming. The incoming data should have all of the bulk chemistry (+/- pXRF Si), and VNIR-SWIR mineralogy integrated for each sample into a single CSV. While this can be automated, variations in data exports and formats make universally applicable code challenging to provide. In this study, data integration was done manually, combining geochemistry and VNIR-SWIR data into a single CSV file used as the input for LP-based mineral predictions (this integrated datasheet can be downloaded in the supplemental materials).

To use this Jupyter Notebook:

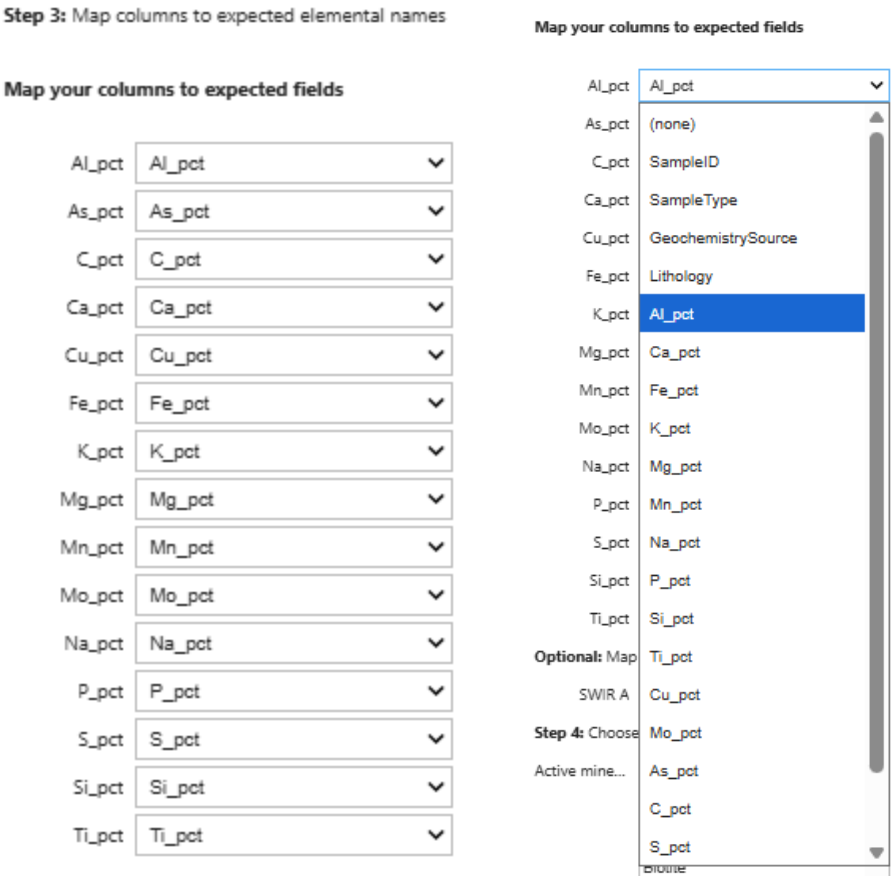
1) Install the required packages by clicking inside the top code block, holding down the SHIFT key and hitting the ENTER key (this will execute the package installation). This only needs to be done once!



2) Then, check the mineral library (starting on line 46) to ensure that all minerals expected in the samples are included, all of the minerals reported in the VNIR-SWIR data are included, and that the chemistries for each mineral are fit-for-purpose. Minerals can be added (following the code conventions in this block), and chemistry of existing minerals can be modified here as well.
   NOTE: if you do update the mineral library, go up to "Kernel" at the top and select "Restart Kernel". It will you ask you if you are sure (you are). This ensures the mineral library changes you have made are recognized
3) Click inside the third block, holding down the SHIFT key and hitting the ENTER key (this will execute the code and a series of interactive widgets will appear at the bottom, below the block of code).



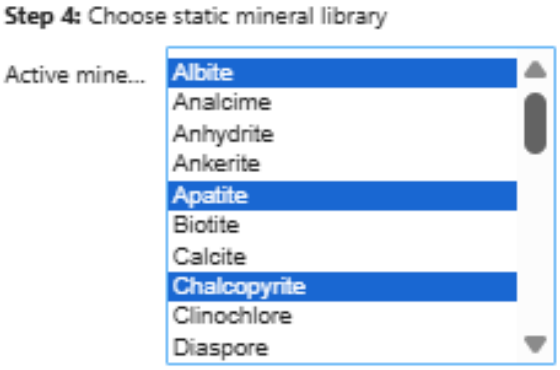
4) Click the "Upload data" button, and select the integrated CSV which contains chemistry, VNIR-SWIR, and any other columns required to define geologically-significant rules. Click OK.
5) Set the units (if required). The code expects the values to be coming in as elemental wt.% - use these check boxes to convert oxides or ppm to elemental wt.%.
6) Map the columns from your CSV (available in the drop-down menus) to the expected elemental columns (these are populated from the mineral library, so if you add a new mineral with a new element, other elements will appear here.



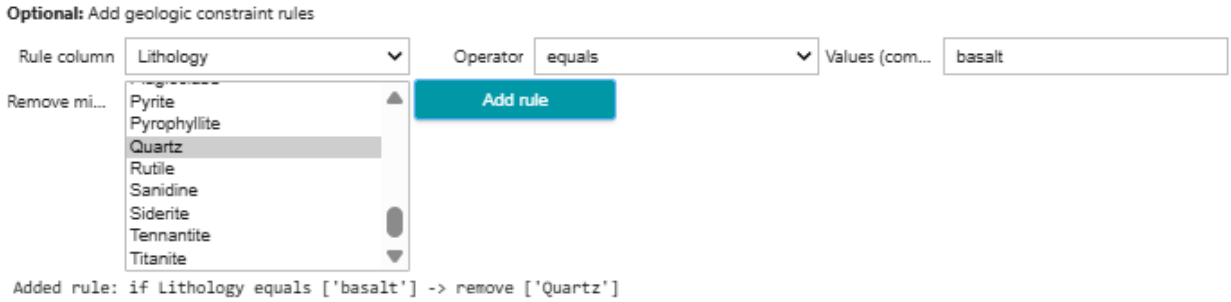
7) Map the columns containing the VNIR-SWIR data as well.

NOTE: if you don't link the SWIR columns, these minerals will not be considered dynamically (on a row-by-row basis).

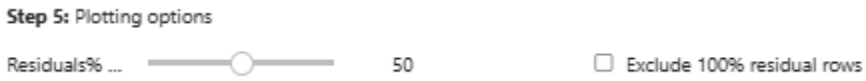
8) Choose the static mineral library by highlighting them (you can select multiple minerals by holding down the CTRL key). These are the minerals that will be considered for **every sample/row** in the incoming data. DO NOT include the VNIR-SWIR active minerals you want to consider dynamically.



9) Optional: add geologic constraints by selecting the rule column (perhaps lithology or magnetic susceptibility response), then use the operators to define the rule. When finished, click "Add rule" and the new rule will show below.



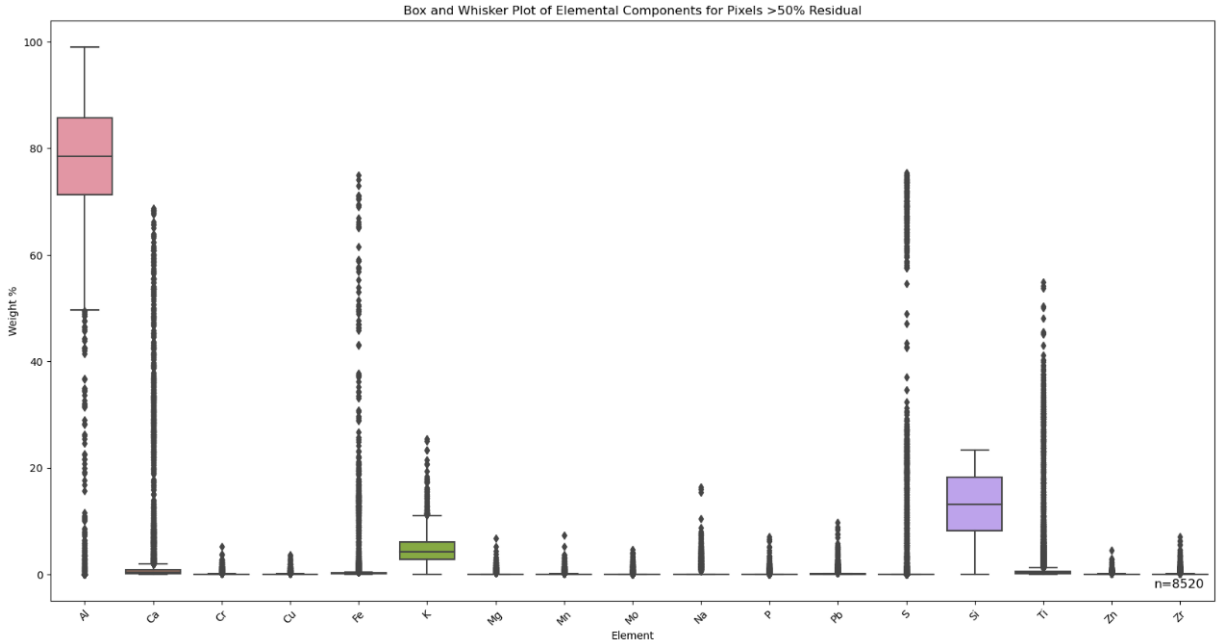
10) Set the plotting options to show the chemistry of high residual samples. Default is to show chemistry for any row reporting >50% residual mineralogy.



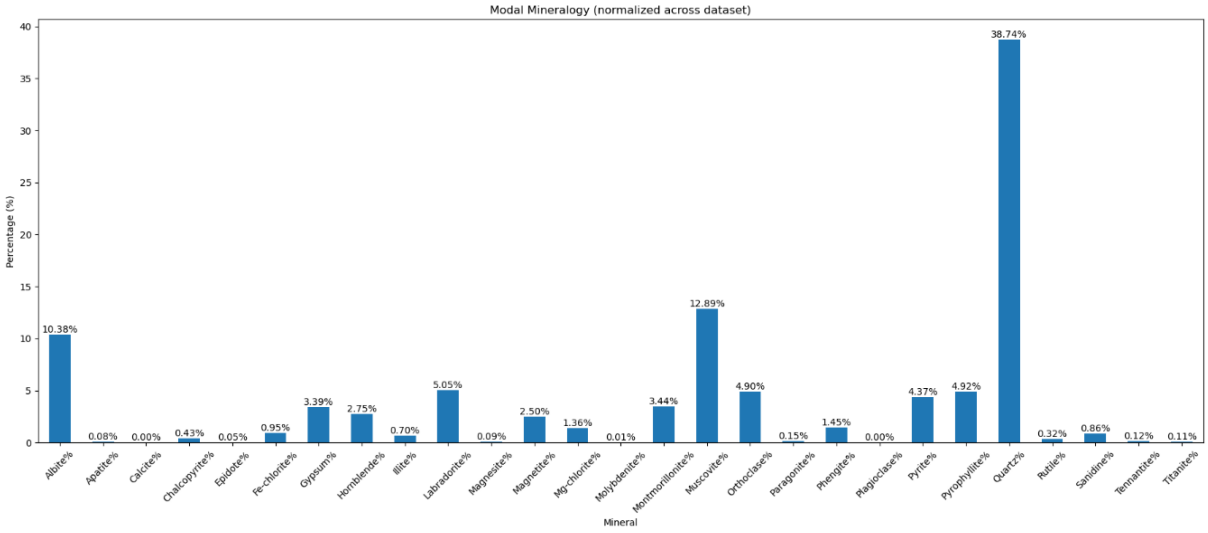
11) Create a filename for the output CSV (it will save to the same folder where the Jupyter Notebook is saved on your computer). Then click "Run LP & Export".

12) The code will execute, and print updates as it works through the predicted mineralogy. When it is finished, a message will appear that the new CSV was exported (with a file path).

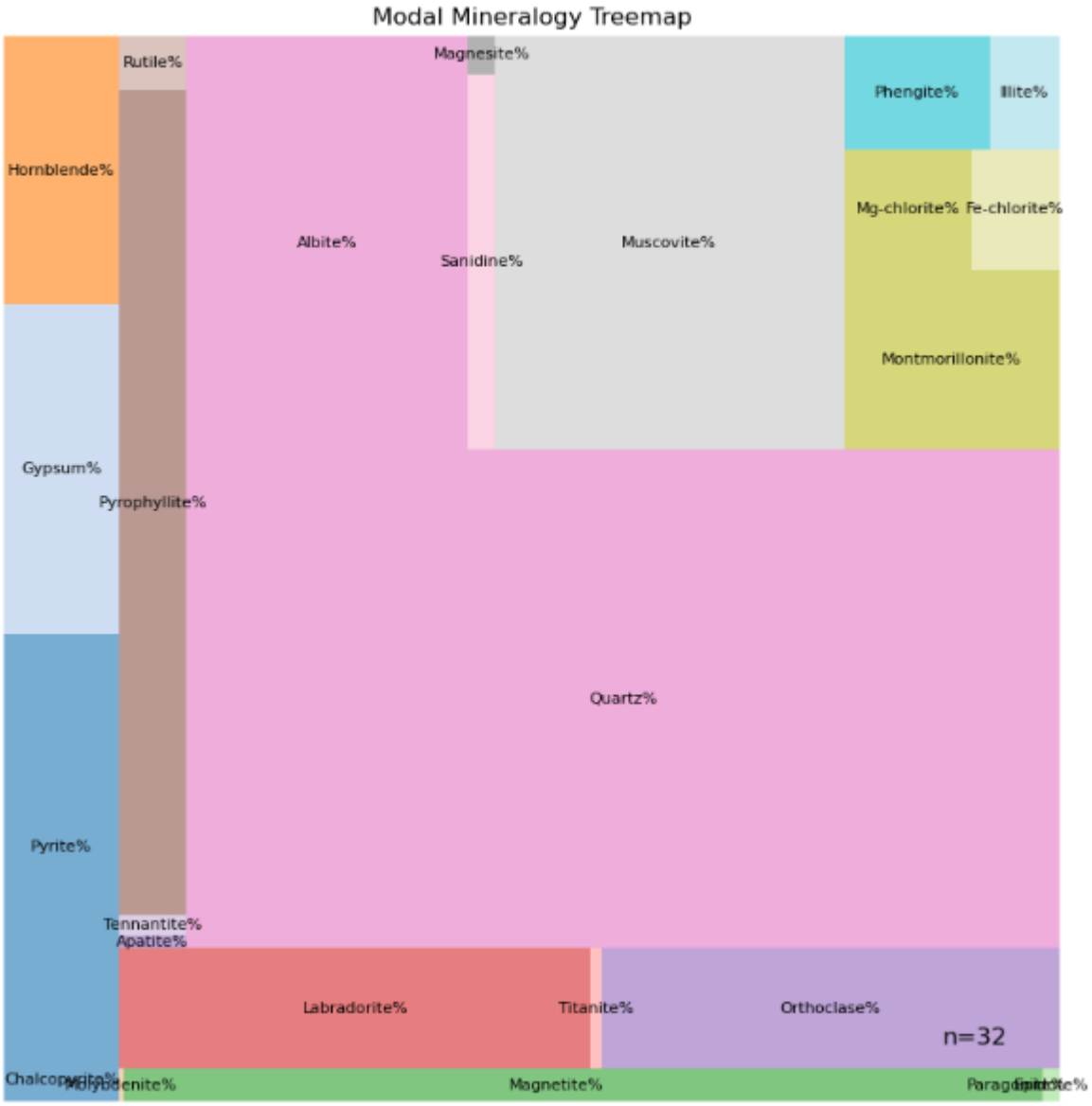
13) Output plots will also appear, showing:
    a. A box and whisker plot showing the chemical distribution for any rows/samples with >50 (or whatever you set the threshold to) residual mineralogy. This is useful to determine if a mineral might be missing from the library.



b. Modal mineralogy across the entire incoming dataset as a bar plot



c. Modal mineralogy across the entire incoming dataset as a treemap



14) The output CSV now contains predicted modal mineral percentages reported in columns with the mineral names. The column "HighestProp" shows the highest abundance mineral reported for that row/sample.