Python Spectral Analysis Tool (PYSAT) for Point Spectra

Ryan Anderson, Nicholas Finch, Sam Clegg, Trevor Graff,
Dick Morris, Jay Laura

Spectral data are powerful, but not always easy to work with…

- Often require specialized knowledge
  - People outside the instrument team can't use data, even when released.
  - Even within the team, only a handful of people might understand the calibration process, making it hard to test new ideas
- Deriving meaningful values (e.g. mineral abundance, chemical composition, etc.) is non-trivial
  - Often complex pre-processing steps are required
  - Advanced methods (e.g. machine learning) are not familiar or accessible to all users and are under-utilized by the community
- Tools are not always shared or easy to use, leading many people to reinvent the wheel

There is a lot of point spectral data out there to work with!

- Curiosity
  - ChemCam has returned more than 400,000 LIBS spectra from Mars
  - APXS has measured many hundreds of targets
- Mars 2020
  - SuperCam will be similar to ChemCam (with added Raman and VNIR spectra)
  - PIXL will collect lots of spectra per target to map composition
- MER
  - APXS, MiniTES, Mossbauer – Lots of data thanks to mission longevity!
- Orbital data can be analyzed with many of the same methods

Need a tool to enable experimentation with these data!

- The community needs a tool that is:
  - Free – No licensing to worry about
  - Open Source – Users can add to the tool and see what it's doing
    - Make use of existing libraries!
  - Graphical Interface – Users don't have to be skilled programmers to use it
  - Flexible – The tool allows users to try different things to get the best results
  - Powerful – The tool allows users to implement everything from simple to highly complex analysis methods

Python Spectral Analysis Tool (PYSAT) for Point Spectra

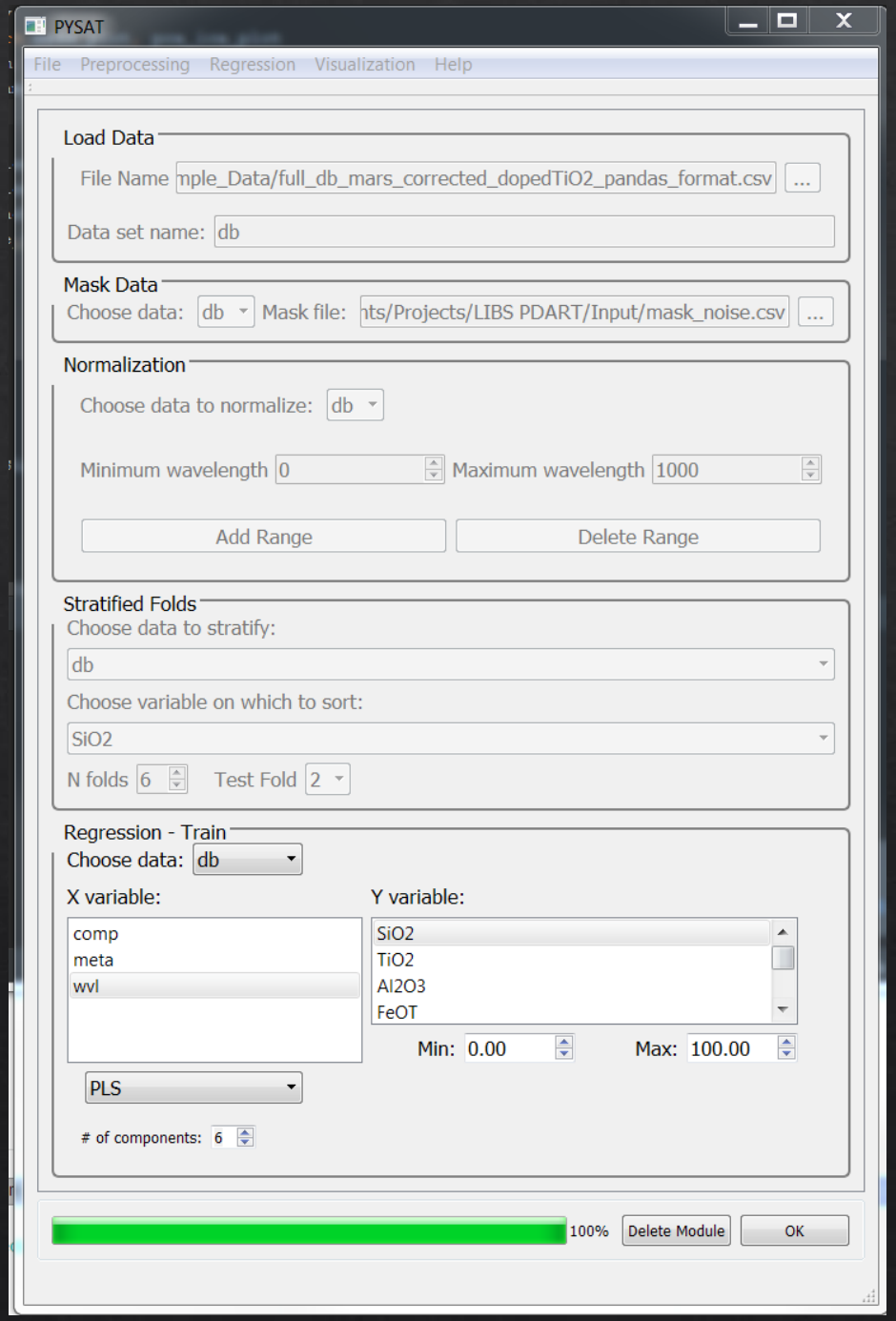
- We have developed PySAT to address the needs on the previous slide
- There are two "paths" being pursued with PySAT right now:
  - Orbital data – Gaddis et al. (#7060 this meeting)
  - Point spectra
- The focus of the point spectra tool so far has been preprocessing and multivariate regression with ChemCam/SuperCam in mind
  - But the tool doesn't care where the data came from!
    - The same methods are broadly applicable
    - For example, many of these methods were developed for VNIR spectra

Multivariate Regression

- Goal: Use a large # of x variables (spectrum) to predict quantity of interest (e.g. chemical composition)
  - Tends to perform better than simpler methods because multivariate methods incorporate more information from the spectrum
- This is supervised learning: a regression model is "trained" based on spectra for which the composition is independently known.
- Terminology:
  - Training set: Spectra and known compositions used to train the model
  - Test set: Spectra and known compositions used to test the model on novel data
  - Overtraining: Making a model fit the training data so well that it fails on new data
  - Folds: Subsets of the data, used for cross validation
  - Cross validation: Iteratively withholding data when training a model to simulate performance on novel data

PySAT Point Spectra Tool: Interface

- Tool is designed with a modular interface
- Works with .csv files
  - Can read ChemCam "cleaned calibrated spectra" (CCS) data from PDS into the correct format
  - EDR support coming soon
- Users can design "workflows" composed of discrete data processing steps
- Workflows can be saved and restored

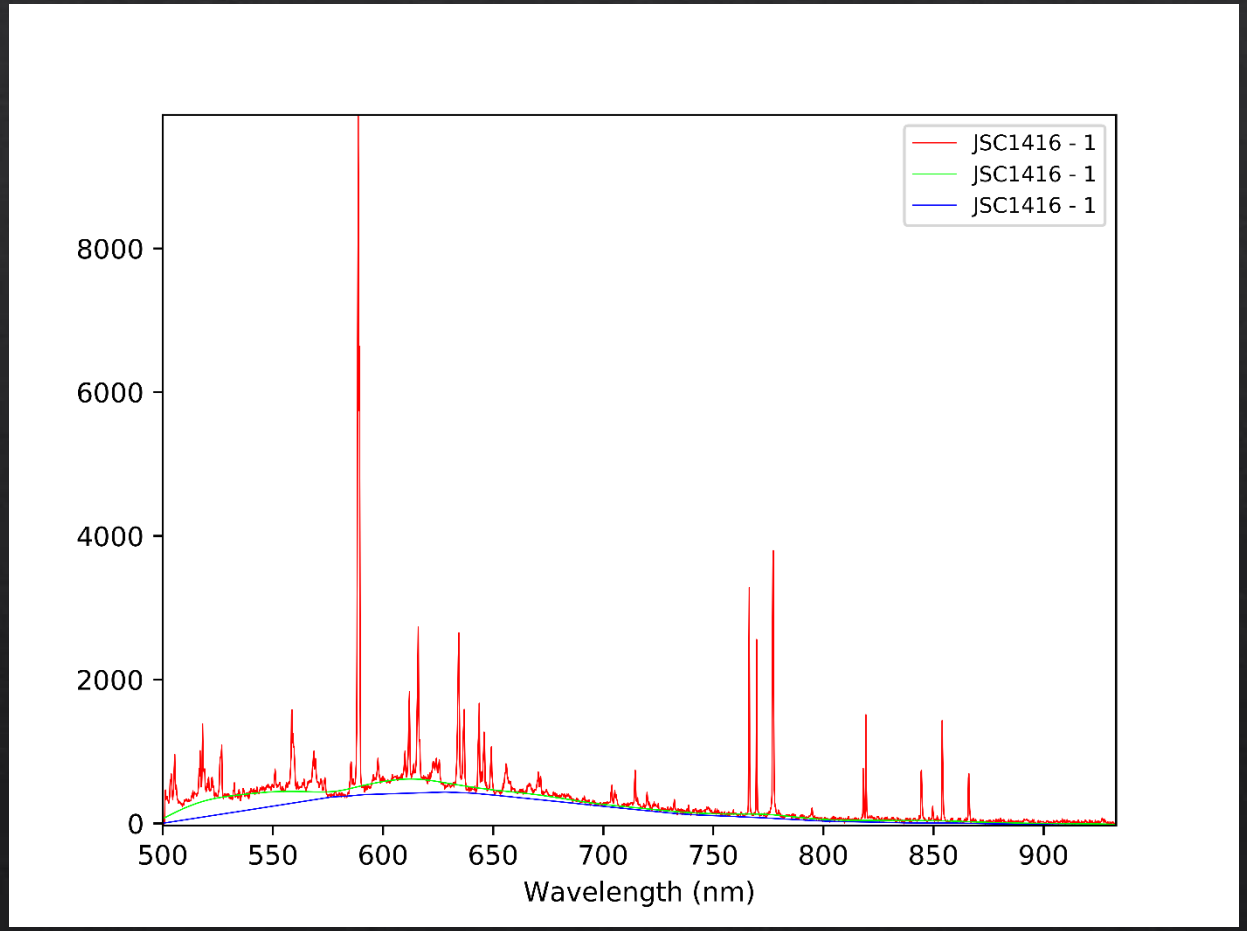


Preprocessing

- Basic capabilities:
  - Interpolation
  - Masking
  - Normalization
  - Multiply by vector
  - Divide into training and test sets
    - Stratified folds
- Advanced capabilities
  - Continuum removal
  - Dimensionality Reduction
  - Denoising (coming soon)
  - Calibration transfer (coming soon)

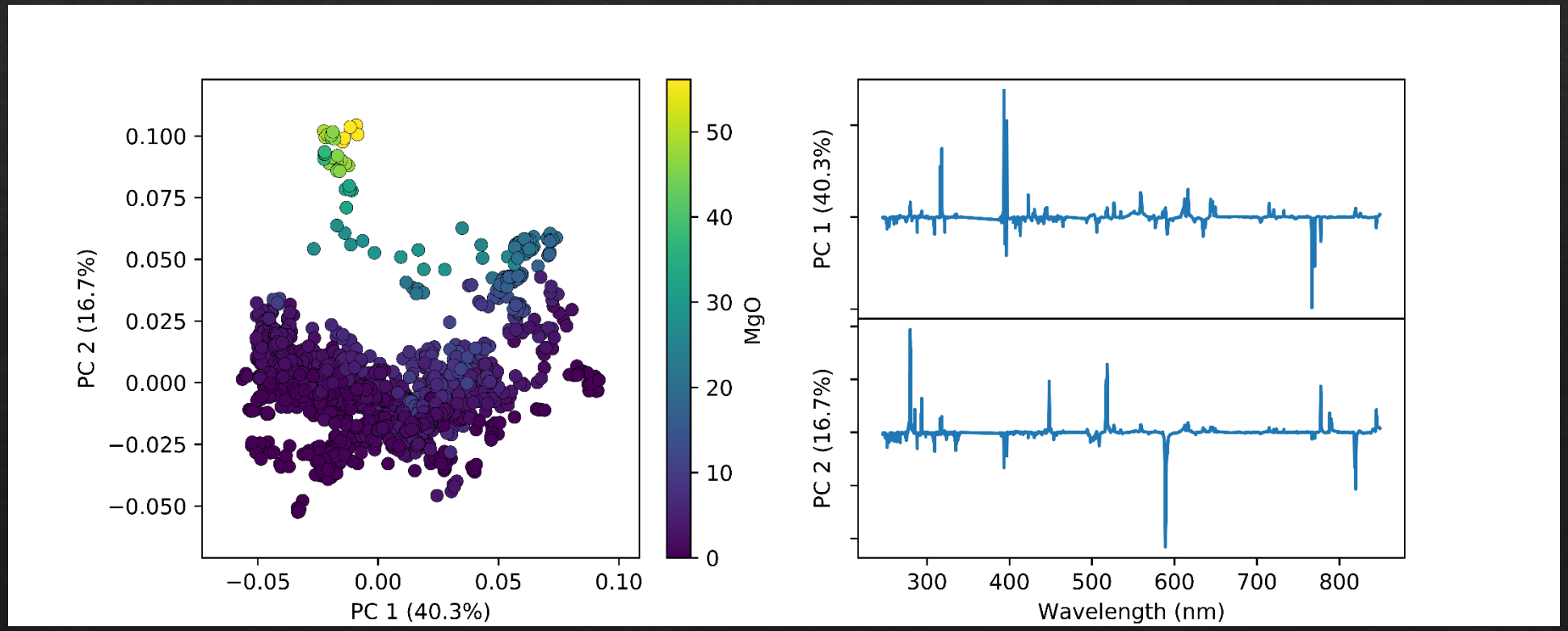
Continuum Removal

- Multiple continuum-removal options are available, courtesy of Tommy Boucher:
  - Asymmetric Least Squares (ALS)
  - Dietrich
  - Iterative Polynomial Fit (PolyFit)
  - Adaptive Iteratively Reweighted Penalized Least Squares (AirPLS)
  - Fully Automatic Baseline Correction (FABC)
  - Kajfosz-Kwiatek (KK)
  - Mario
  - Median
  - Undecimated Stationary Wavelet (used by ChemCam)



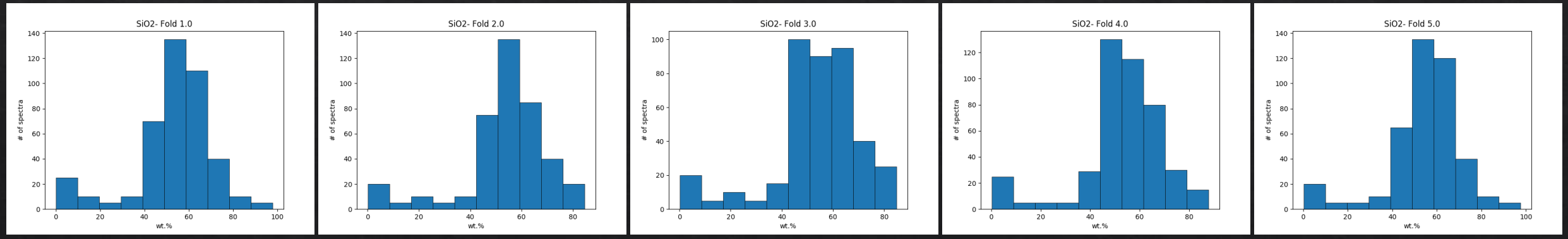
Dimensionality Reduction

- Data can be transformed using Principal Component Analysis (PCA) or two different Independent Component Analysis (ICA) algorithms (FastICA and JADE)
- Scores and loadings can be plotted, and scores can be color-coded based on the value of a metadata or composition column in the data.

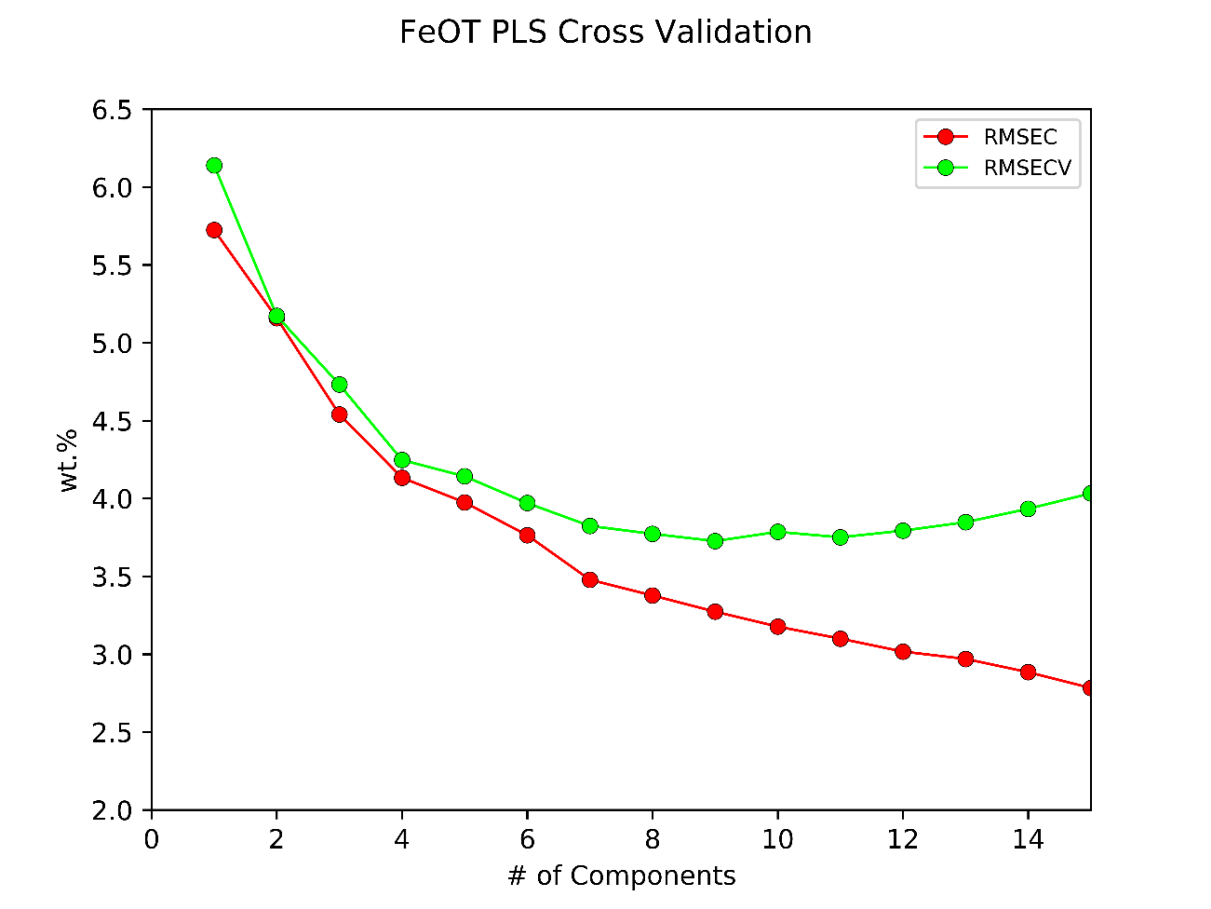


Stratified Folds

- Dividing data into folds is required to do k-fold cross validation
- One fold is specified as a test set to evaluate regression results
- The remaining folds are assigned to the training set
- “Stratified” means the data is sorted on the variable of interest and assigned sequentially to each fold. This leads to a relatively consistent distribution among the folds:



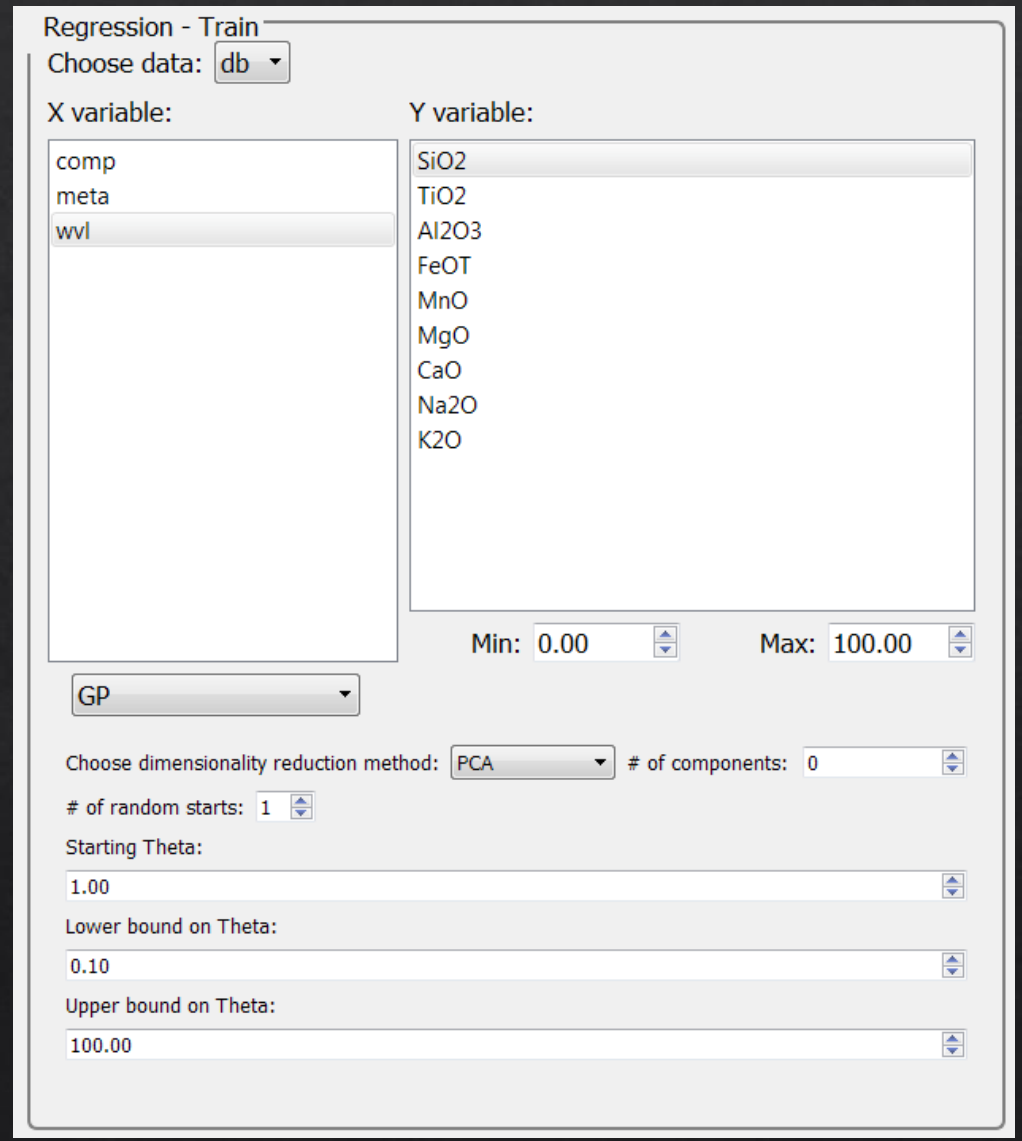
Cross Validation



- Uses cross-validation capabilities in the scikit-learn library to help choose best settings.
- Cross-validation is flexible enough to handle variation of multiple parameters

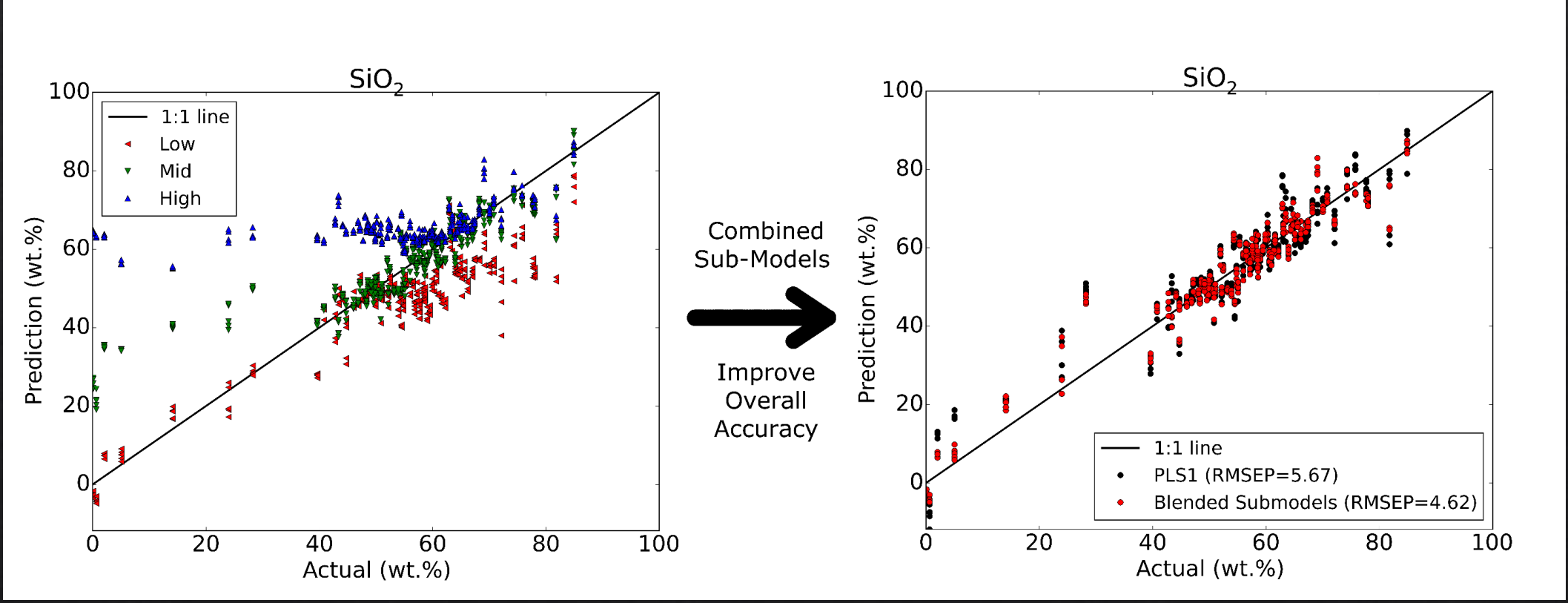
Regression

- Currently implemented:
  - Ordinary Least Squares (OLS)
  - Partial Least Squares (PLS)
  - Gaussian Process Regression (GP)
  - Orthogonal Matching Pursuit (OMP)
  - LASSO
- In development:
  - Elastic Net
  - Bayesian Ridge Regression
  - ARD
  - Least Angle Regression (LARS)
  - Lasso LARS
  - Support Vector Regression (SVR)
  - Kernel Ridge Regression (KRR)



Submodel Regression

- For diverse data sets, a single model sometimes can't capture all variations
- Smaller models trained on a restricted range can perform better in that range
- Blending these "submodels" can give overall better results.



Future Work

- Development
  - Streamlining code and interface
  - Simple function fitting
  - Clustering and classification
    - Leverage scikit-learn
  - Improve PCA/ICA so that new data can be projected
  - Outlier removal
  - Calibration transfer methods
- Application
  - Improve ChemCam calibration
  - Develop SuperCam calibration
  - Experiment with other data sets (lab data, APXS, MiniTES, Mossbauer, etc.)

https://github.com/USGS-Astrogeology/PySAT

https://github.com/USGS-Astrogeology/PySAT_Point_Spectra_GUI