Pacific Northwest
NATIONAL LABORATORY

PNNL-33377

Development of a hybrid neural network and transfer learning model for optimized ICP-MS/MS operation

September 2023

Khadouja Harouaka
Rachel E Richardson
Evan C Glasscock
Amanda French
Isaac Arnquist
Eric Hoppe
Sarah Akers
Kelly Stratton
Draguna Vrabie

U.S. DEPARTMENT OF ENERGY

Prepared for the U.S. Department of Energy
under Contract DE-AC05-76RL01830

DISCLAIMER

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC05-76RL01830

Printed in the United States of America

Available to DOE and DOE contractors from
the Office of Scientific and Technical Information,
P.O. Box 62, Oak Ridge, TN 37831-0062
www.osti.gov
ph: (865) 576-8401
fox: (865) 576-5728
email: reports@osti.gov

Available to the public from the National Technical Information Service
5301 Shawnee Rd., Alexandria, VA 22312
ph: (800) 553-NTIS (6847)
or (703) 605-6000
email: info@ntis.gov
Online ordering: http://www.ntis.gov

PNNL-33377

Development of a hybrid neural network and transfer learning model for optimized ICP-MS/MS operation

September 2023

Khadouja Harouaka
Rachel E Richardson
Evan C Glasscock
Amanda French
Isaac Arnquist
Eric Hoppe
Sarah Akers
Kelly Stratton
Draguna Vrabie

Prepared for
the U.S. Department of Energy
under Contract DE-AC05-76RL01830

Pacific Northwest National Laboratory
Richland, Washington 99354

Abstract

Correct function and calibration of instrumentation is a crucial assumption for any scientific experiment. One such instrument, tandem inductively coupled plasma mass spectrometer (ICP-MS/MS), has in-depth calibration settings that range across 30+ different parameters, making it difficult to determine optimal conditions without expertise and some degree of trial and error. Often, these settings are hand-tuned, a time-intensive process prone to local maxima and human error. While some automation is available, the automation also may favor local optimizations over a global optimum. In addition to these difficulties, day to day instrument variability can further complicate the calibration process. We propose a solution to this problem as a machine learning (ML) algorithm that learns how each parameter helps determine the calibration sensitivity across several elements, and re-weights parameters over time as instrument variability changes (e.g., a global neural network (NN) with a time-dependent transfer learning (TL) component). This model would be able to generate a surface of predicted calibration sensitivities and their respective parameters, and a simple multivariate algorithm would be able to pull out the optimum results with the settings associated with them. Here-in, we describe our initial findings in working towards this goal, including data extraction from historical files, exploratory data analysis, and some initial model building to better describe the data and the feasibility of our goal.

Acknowledgments

This research was supported by the **Mathematics of Artificial Reasoning for Science (MARS) Initiative**, under the Laboratory Directed Research and Development (LDRD) Program at Pacific Northwest National Laboratory (PNNL). The historical data acquired for this project was facilitated by the Environmental Molecular Sciences Laboratory (EMSL), a national scientific user facility sponsored by the DOE's Office of Biological and Environmental Research and located at PNNL. PNNL is a multi-program national laboratory operated for the U.S. Department of Energy (DOE) by Battelle Memorial Institute under Contract No. DE-AC05-76RL01830.

Acronyms and Abbreviations

ICP-MS/MS – Tandem inductively coupled plasma mass spectrometry

NN – Neural Network

TL – Transfer Learning

ML – Machine Learning

m/z – mass to charge ratio

MARS – Mathematics of Artificial Reasoning for Science

PNNL – Pacific Northwest National Laboratory

LDRD – Laboratory Directed Research and Development

DOE – U.S. Department of Energy

EMSL – Environmental Molecular Sciences Laboratory

PDF – Portable Document Format

XML – Extensible Markup Language

EDA – Exploratory Data Analysis

CPS – counts per second

Contents

Abstract ii
Acknowledgments iii
Acronyms and Abbreviations iv
1.0 Introduction 1
1.1 Chemistry of the ICP-MS/MS 1
1.2 Selection of Machine Learning (ML) models 3
2.0 Data Extraction 5
2.1 Optical Character Recognition (OCR) methods 5
2.2 Common errors and error removal 6
2.3 Comparison to XML files and manual extractions 6
3.0 Exploratory data analysis (EDA) 7
3.1 Distribution of response and predictor variables 7
3.2 Relationships between predictor variables 9
3.3 Principal Component Analysis (PCA) with metadata 10
3.3.1 Full dataset 10
3.3.2 Thallium dataset 11
3.4 Correlation of observations through time 13
4.0 Initial models 15
4.1 Linear modeling approach 15
4.1.1 Linear modeling performance 15
4.1.2 Strongest predictor variables 16
4.1.3 Range Subset 16
4.2 Random forest approach 17
4.2.1 Random forest performance 17
4.2.2 Strongest predictor variables 18
4.2.3 Model tuning ..... **Error! Bookmark not defined.**
4.2.4 Range Subset 19
Discussion and Future Directions 21
5.0 References 22
Appendix A – Link to Agilent reference A.1

Figures

Figure 1. ICP-MS tuning requires adjustment to optimize sensitivity for different analytes and experimental goals; a) Instrument sensitivity across analytes when tuned for mid-range masses; b) instrument sensitivity across analytes when optimizing oxidized products. 1

Figure 2. Framework for localizing a neural network model with transfer learning. Figure originally printed in Puneet Mishra, Dário Passos, Deep calibration transfer: Transferring deep learning models between infrared spectroscopy instruments (2). 4
Figure 3. Mean-subtracted predictor values. 8
Figure 4. Trelliscopejs depiction of predictor values density. 8
Figure 5. Hierarchical clustering of predictors via complete Euclidean distances. 9
Figure 6. Pearson's correlation between predictors used in our initial models. 10
Figure 7. PCA of all varying numeric dataset predictors, distinguishing (left) tandem vs. single MS used in a run as well as (right) different targeted masses in a run. 11
Figure 8. PCA of all varying numeric dataset predictors for thallium, colored by continuous variables of interest including (top-left) acquisition time, (top-right) our response variable Average count, and range (bottom-left). 12
Figure 9. PCA loadings of all varying numeric dataset predictors for thallium. 13
Figure 10. Decomposition of response values for thallium with detection set at range 20,000. Dataset was divided into periods of 64 observations. 14
Figure 11. (Left) Auto-correlation and (Right) partial autocorrelation on values for thallium with detection set at range 20,000. 14
Figure 12. Predicted response generated by linear model plotted on top of observed responses. 15
Figure 13. Range's effect on the response variable, where increased ranges report a larger signals. 16
Figure 14. Difference between robust linear model predictions and observed values as a percentage. 17
Figure 13. Predicted response generated by average-performance random forest model plotted on top of observed responses. Parameters for the random forest were set to 2000 total trees with 17 random variables used in each tree. 18
Figure 14. Node purity and MSE improvements with the inclusion of predictors in random forest trees. 19

Tables

Table 1. All predictor and response variables collected for use in predictive modeling. 2
Table 2. Investigated OCR methods. 5
Table 3. Commonly analyzed standard elements in historical dataset. 7

1.0 Introduction

Use Body Text for paragraphs in this section. PNNL reports use http://www.chicagomanualofstyle.org/home.html for document style. Right-click and choose open hyperlink to view the style guide.

Atomic ICP-MS/MS is an effective method for measuring elemental concentrations of materials and is widely used across various industries and academic research. While commercial ICP-MS/MS software come equipped with 'autotune' features, they are not iterative, can be inflexible in optimizing multiple mass ranges, and are ineffectual at tuning collision cell chemistry (**Figure 1**). We propose to replace the expertise of an experienced user with a hybrid neural network (NN) and transfer learning (TL) machine learning model that would interface with ICP-MS/MS software to effectively automate instrument tuning.

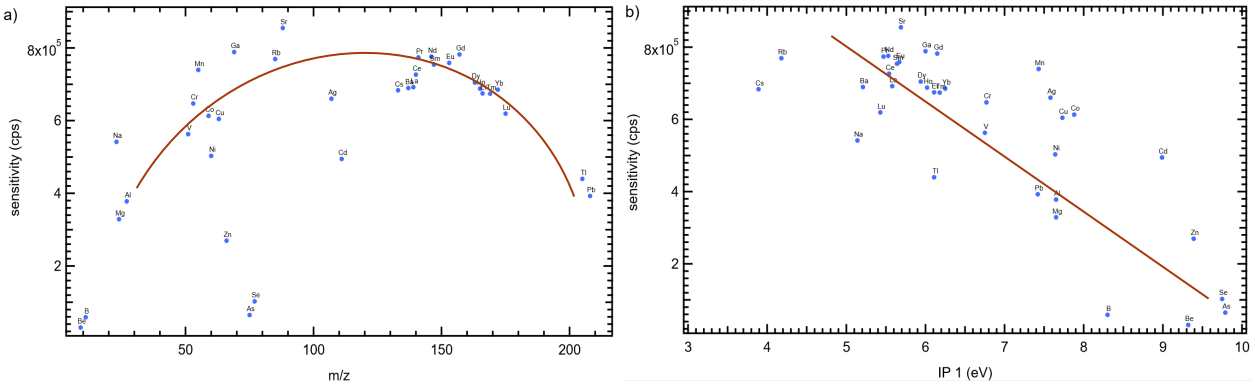

Figure 1. ICP-MS tuning requires adjustment to optimize sensitivity for different analytes and experimental goals; a) Instrument sensitivity across analytes when tuned for mid-range masses; b) instrument sensitivity across analytes when optimizing oxidized products.

1.1 Chemistry of the ICP-MS/MS

Use Body Text for paragraphs in this section. PNNL reports use http://www.chicagomanualofstyle.org/home.html for document style. Right-click and choose open hyperlink to view the style guide.

The ICP-MS/MS instrument functions by ionizing the sample in an argon plasma, then creating and focusing an ion beam through a series of electrical lenses. The ions can be selectively sampled by their mass to charge ratio (m/z) by passing the beam through quadrupole mass filters. In the instance that the analyte is interfered by a species of the same m/z (e.g., ^{87}Sr and ^{87}Rb), gas phase ion-molecule reactions can be employed in a collision reaction cell to affect a separation during the measurement.

There are several tunable parameters that control the creation, focusing and sampling of the ion beam to optimize the transmission of the ion beam through the system to the detector. Additional tuning parameters also control the gas phase reaction chemistry in the cell, which need to be optimized to maximize the production of a desired product or removal of an interference species. Typically, the instrument user will manually adjust ~ 30 software

parameters (Table 1) in an iterative process to maximize the analyte signal intensity. As the tuning parameters are congruent with ion m/z, the user will need different settings for analytes across the mass range. Further complications arise when reaction chemistry is part of the analysis. As such, instrument tuning can be time sensitive and require a fair amount of expertise to leverage the full potential of the ICP-MS/MS that can otherwise measure virtually every element in the periodic table. Refer to Appendix A for more detailed information about the settings used during calibration.

Table 1. All predictor and response variables collected for use in predictive modeling.

Variable	Related Feature	Adjustable	Model Role	Range
Ave. Count	Response detection	No	Response	0 – 20,000+*
Range	Response detection	Yes	Predictor	0-50,000+*
Concentration	Response detection	Yes	Predictor	NA*
RF power	Operation settings	Yes	Predictor	500 - 1600 W
RF matching	Operation settings	Yes	Predictor	0.20 – 3.00 V
Sample depth	Operation settings	Yes	Predictor	3.0 – 28.0 mm
Nebulizer gas	Operation settings	Yes	Predictor	0.00 – 2.00 L/min
Option gas	Operation settings	Yes	Predictor	0.0 – 100.0 %
Nebulizer pump	Operation settings	Yes	Predictor	0.00 – 0.50 rps
S/C temp	Operation settings	Yes	Predictor	-5 – 20 °C
Makeup gas	Operation settings	Yes	Predictor	0.00 – 2.00 L/min
Plasma gas	Operation settings	Yes	Predictor	15.00 – 23.00 L/min
Auxiliary gas	Operation settings	Yes	Predictor	0.90 – 1.20 L/min
Extract 1	Lenses	Yes	Predictor	-200.0 – 10.0 V
Extract 2	Lenses	Yes	Predictor	-250 – 10.0 V
Omega bias	Lenses	Yes	Predictor	-200 – 10 V
Omega lens	Lenses	Yes	Predictor	-50.0 – 50.0 V
Q1 entrance	Lenses	Yes	Predictor	-100 – 20.0 V
Q1 exit	Lenses	Yes	Predictor	-50 – 20 V
Cell focus	Lenses	Yes	Predictor	-50 – 20.0 V
Cell entrance	Lenses	Yes	Predictor	-150 – 10 V
Cell exit	Lenses	Yes	Predictor	-150 – 10 V
Deflect	Lenses	Yes	Predictor	-150.0 – 20.0 V
Plate bias	Lenses	Yes	Predictor	-150 – 10 V
Q1 mass gain	Q1	Yes	Predictor	0 – 255
Q1 mass offset	Q1	Yes	Predictor	0 – 511
Q1 axis gain	Q1	Yes	Predictor	0.9800 – 1.0200
Q1 axis offset	Q1	Yes	Predictor	-0.50 – 0.50
Q1 bias	Q1	Yes	Predictor	-100.0 – 20.0 V

Q1 prefilter bias	Q1	Yes	Predictor	-50.0 – 20.0 V
Q1 postfilter bias	Q1	Yes	Predictor	-50.0 – 20.0 V
SLS factor	Q1	Yes	Predictor	0.00 – 1.00
SLG factor	Q1	Yes	Predictor	0.20 – 1.00
Use Gas	Q1	Yes	Predictor	Yes/No
He flow	Q1	Yes	Predictor	0.0 – 12.0 mL/min
H2 flow	Q1	Yes	Predictor	0.0 10.0 mL/min
3rd gas flow	Q1	Yes	Predictor	0 – 100 %
4th gas flow	Q1	Yes	Predictor	0 – 100 %
OctP Bias	Q1	Yes	Predictor	-150.0 – 20.0 V
Axial Acceleration	Q1	Yes	Predictor	-2.0 – 2.0 V
OctP RF	Q1	Yes	Predictor	30 – 180 V
Energy Discrimination	Q1	Yes	Predictor	-20.0 – 150.0 V
Q2 mass gain	Q2	Yes	Predictor	0 – 255
Q2 mass offset	Q2	Yes	Predictor	0 – 511
Q2 axis gain	Q2	Yes	Predictor	0.9800 – 1.0200
Q2 axis offset	Q2	Yes	Predictor	-0.50 – 0.50
Q2 bias	Q2	Yes	Predictor	-100.0 – 0.0 V
Torch H	Torch Axis	Yes	Predictor	-2.0 – 2.0 mm
Torch V	Torch Axis	Yes	Predictor	-2.0 – 2.0 mm
Discriminator	EM (Hardware Settings)	Yes	Predictor	0.0 – 200.0 mV
Analog HV	EM (Hardware Settings)	Yes	Predictor	0 – 3500 V
Pulse HV	EM (Hardware Settings)	Yes	Predictor	0 – 2000 V

*Dependent on element used

1.2 Selection of Machine Learning (ML) models

Use Body Text for paragraphs in this section. PNNL reports use http://www.chicagomanualofstyle.org/home.html for document style. Right-click and choose open hyperlink to view the style guide.

Neural network (NN) models are used to predict a certain outcome given a series of inputs, often used in a categorical or probability-based context. Built over several layers of weighted nodes based on input parameters and known outcomes, NN have been increasingly more popular for modeling and predicting outcomes in complex systems. As with most models, NNs tend to perform best with large amounts of data. Several years' worth of tuning results are available to us for this purpose and additional data points can be easily collected as necessary. After being shown to efficiently predict instrument sensitivity given a set of input parameters, our proposed model would be able to predict outcomes in a confined parameter space using chemical principals and properties of the analyte. A multivariate optimization algorithm can then be run over the parameter space and return the input settings to maximize instrument sensitivity. While the results from this workflow will establish a baseline for optimized inputs, the variance in instrument operation requires additional fine-tuning (*i.e.,* localizing a generalized

model (3)). TL has previously been shown to be an effective tool for localizing general models and re-mapping lengthy calibration processes across instruments. The combination of a NN with a TL framework outperformed PLS, PFCE, and a deep learning method (Figure 1, 2) (1, 2).

While we have less complex predicted variables than used for previously published calibration transfer data, we note that the context is similar enough to provide a framework to structure the NN and transfer learning models (Fig 2). Within the provided framework and the ability to access a live signal for accumulating data, we believe a similar solution would be possible for tuning both across instruments as well as within day-to-day operations of a single instrument.

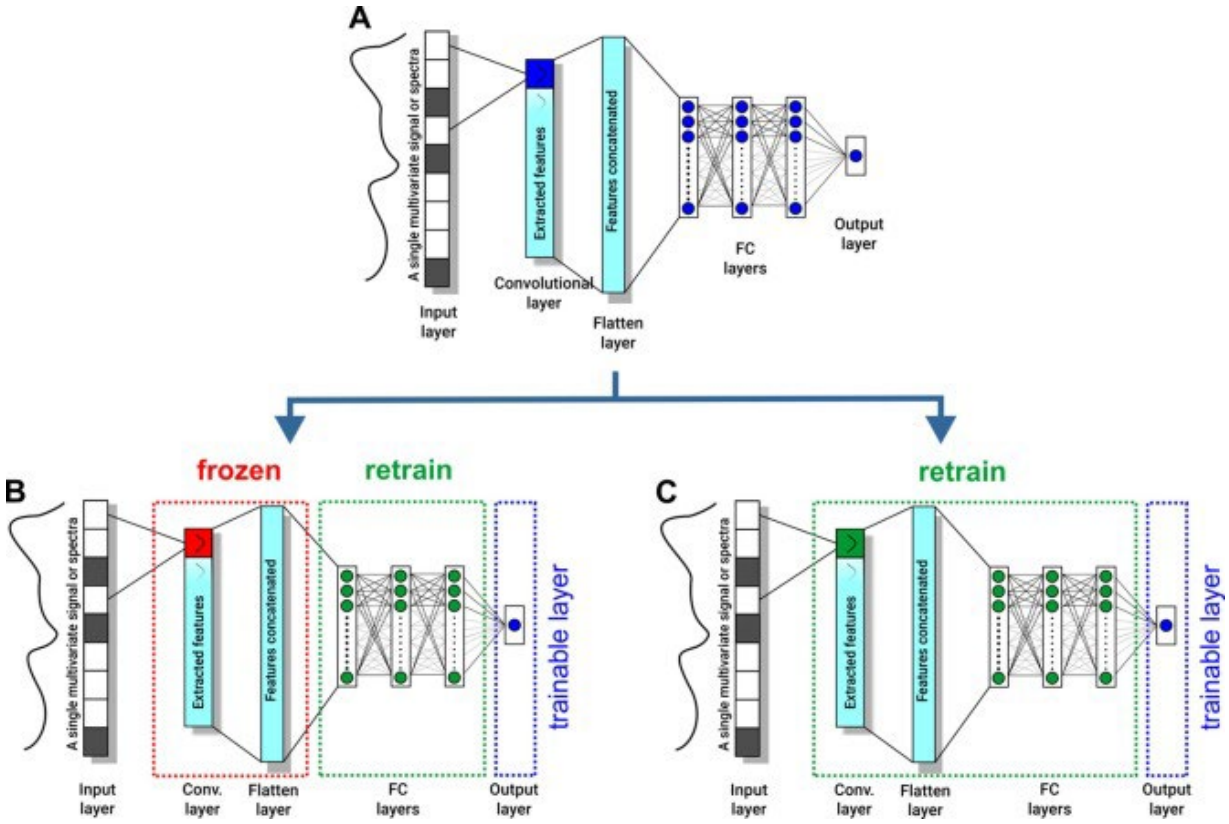

Figure 2. Framework for localizing a neural network model with transfer learning. Figure originally printed in Puneet Mishra, Dário Passos, Deep calibration transfer: Transferring deep learning models between infrared spectroscopy instruments (2).

2.0 Data Extraction

The historical data provided by our chemical analysts included 1166 portable document format (PDF) files reporting calibration results from an Agilent 8900 triple quadrupole ICP-MS/MS used in the lab at PNNL. Additional files from a second instrument were not obtained at the time of writing this report. As the acquired PDFs did not contain machine-readable text, resources were dedicated data extraction from these files in the most accurate and efficient manner possible to ensure data integrity. To such ends, we investigated and employed several open-source optical character recognition (OCR) methods as well as manual data extraction to pull information about our predictor and response variables. In addition, during the last weeks of the project received 28,916 batch and sample extensible markup language (XML) files that were used to approximate truth in calibration runs. We were not able to acquire calibration XML files at the time of writing (16Sep2022).

2.1 Optical Character Recognition (OCR) methods

Open-source OCR methods were explored in R, python, and command line executable environments (Table 2). Python implemented Tesseract and Doctr engines showed the best performance of investigated methods by visual evaluation of extraction completeness.

Table 2. Investigated OCR methods.

Package/Method	Coding Environment	Engine	Able to run?	Speed	Accuracy	Extract Use
tesseract	R	Tesseract	Yes	Slow	Medium-High	No
tabulizer	R	Tabula	Yes	Slow	Medium	No
TesserOCR	Python	Tesseract	Yes	Fast	High*	Yes
Doctr	Python	Doctr	Yes	Slow	Medium-High	Yes
OCRmyPDF	Command line executable	Tesseract	Yes	Slow	Medium	No
EasyOCR	Command line executable	EasyOCR	Online only	-	-	-

* Evaluated by line-by-line settings

For TesserOCR, several methods were investigated to ensure best performance, including whole page and line by line recognition. The most complete results were obtained by line-by-line recognition with limited characters defined for recognition. Line-by-line recognition was performed by manually entering the approximate location of the desired text, then passing the area to OpenCV's edge detection function to minimize the whitespace surrounding the text.

The Doctr implementation conducted a full-page extraction to detect characters and was significantly slower than the TesserOCR implementation. However, unlike other tested methods, Doctr relies on a slightly different engine that appeared to make-up for errors observed in the TesserOCR method.

2.2 Common errors and error removal

Like any image-reading implementation, OCR methods like TesserOCR and Doctr have weaknesses in character recognition that are non-trivial and widely occurring. TesserOCR results contained many observations where negative signs (-), decimal points, and cross-page observations were misread. Doctr, on the other hand, showed difficulties in correctly reading number sequences. However, between the two methods and our knowledge of viable values for each of the parameters, we were able to establish filters for common errors and consensus algorithms for weighting one result over another.

Comparing the results of the two extractions, 7768 discrepancies were observed and resolved out of 61503 comparisons in 1079 documents (~13%). The resolved results yielded notable improvements in initial models, increasing accuracy by ~5% and ~10% for linear and random forest models, respectively.

2.3 Comparison to XML files and manual extractions

The XML batch files that were provided for use as ground truth had particular strengths and weaknesses for our analyses. The strength of these files is the manner of extraction – for each parameter, the numeric settings could be located within the file without the issue of misreads. However, several samples are sun in a single batch and batches are not calibration-run specific and did not contain our response variable, making it tricky to apply this data in the right context. We utilized the creation and modification times of each XML file to attempt to achieve the closest reasonable matches to calibration files from using those batch methods. For the response variables, we used comparisons to hand-extracted values from the PDF files to verify our result. Using extractions from the batch XML files and manual extractions as ground truth, we fully verified the accuracy of our auto-extraction methods on 40 batches. The methods achieved 100.0% accuracy over these files, with 0 files highlighted for spot-checking and further manual correction.

3.0 Exploratory data analysis (EDA)

During our iterative process of data extraction and implementing initial ML models, we relied heavily on exploratory data analysis for outlier detection, assessing variable relationships, and understanding the similarity and differences across calibration runs. We are especially cognizant of differences between commonly measured elements in the historical dataset (Table 3).

Table 3. Commonly analyzed standard elements in historical dataset.

Element	Mass indicators	Occurrence
Lithium	7	48
Potassium	39, 41	272
Cobalt	59	99
Yttrium	89	141
Cerium	156, 140	596
Thallium	205	659
Associated observations	Many	5021

3.1 Distribution of response and predictor variables

Distribution of predictor values plays a strong role in choosing appropriate models and methods to implement. Outliers in distributions of mean subtracted predictors also provided our initial targets to spot-check our results (Figure 3). Some predictors, including Option Gas, Omega Lens, and Torch HV, rarely varied in the data and were removed in subsequent models.

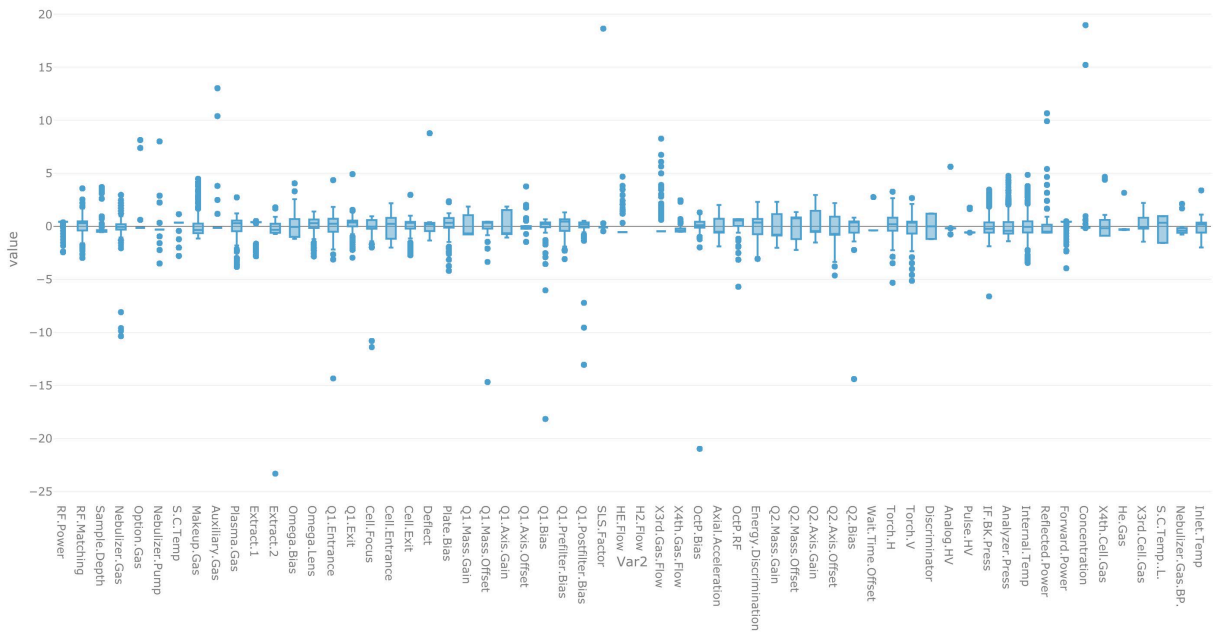

Figure 3. Mean-subtracted predictor values.

In addition, we utilized the R package trelliscopejs to observe the density of the observations for each of predictor variables across each type of run (Figure 4). This methodology allowed us to manage many plots in an efficient fashion and also assisted in outlier detection.

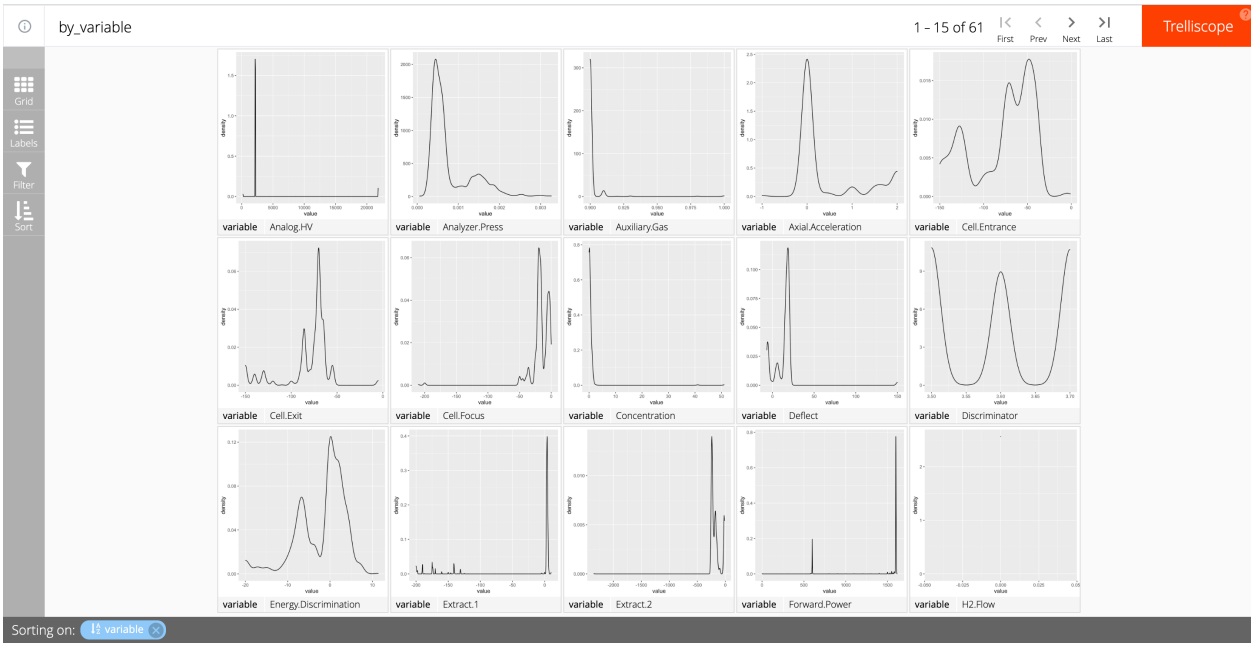

Figure 4. Trelliscopejs depiction of predictor values density.

3.2 Relationships between predictor variables

Relationships between each predictor used in the model was also investigated to give context to results from initial ML models – ultimately, we will require all tunable parameters as predictors since we hope to return those parameters to the user in the final models, but interpretation of our initial models would benefit greatly from this information. Specifically, when determining variable importance in our later models, we consider that highly correlated variables might undergo masking during model training.

Hierarchical clustering of the predictor variables using Euclidian distances offered both expected and unusual associations (Figure 5). Mass gain and offset are expectedly close due to relationships described in Appendix A, however RF Power and Pulse associations are not clearly explained by our current knowledge. Strong correlations were observed between these predictors as well (Figure 6).

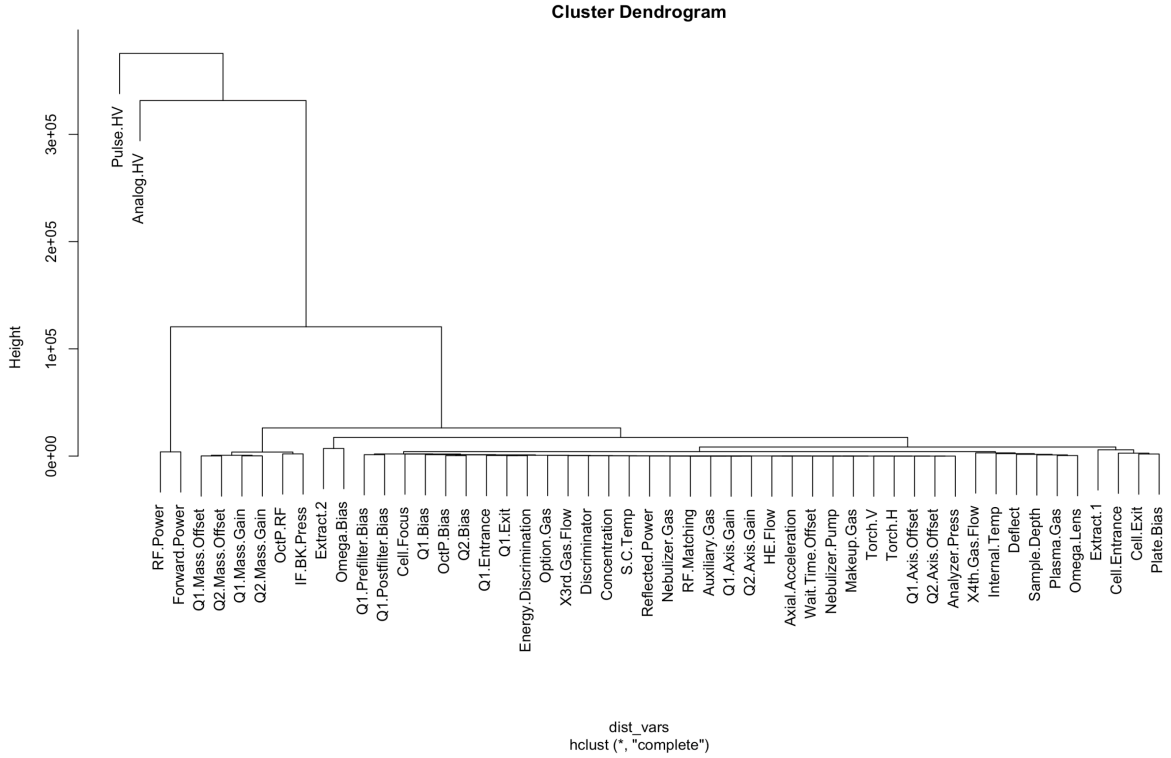

Figure 5. Hierarchical clustering of predictors via complete Euclidean distances.

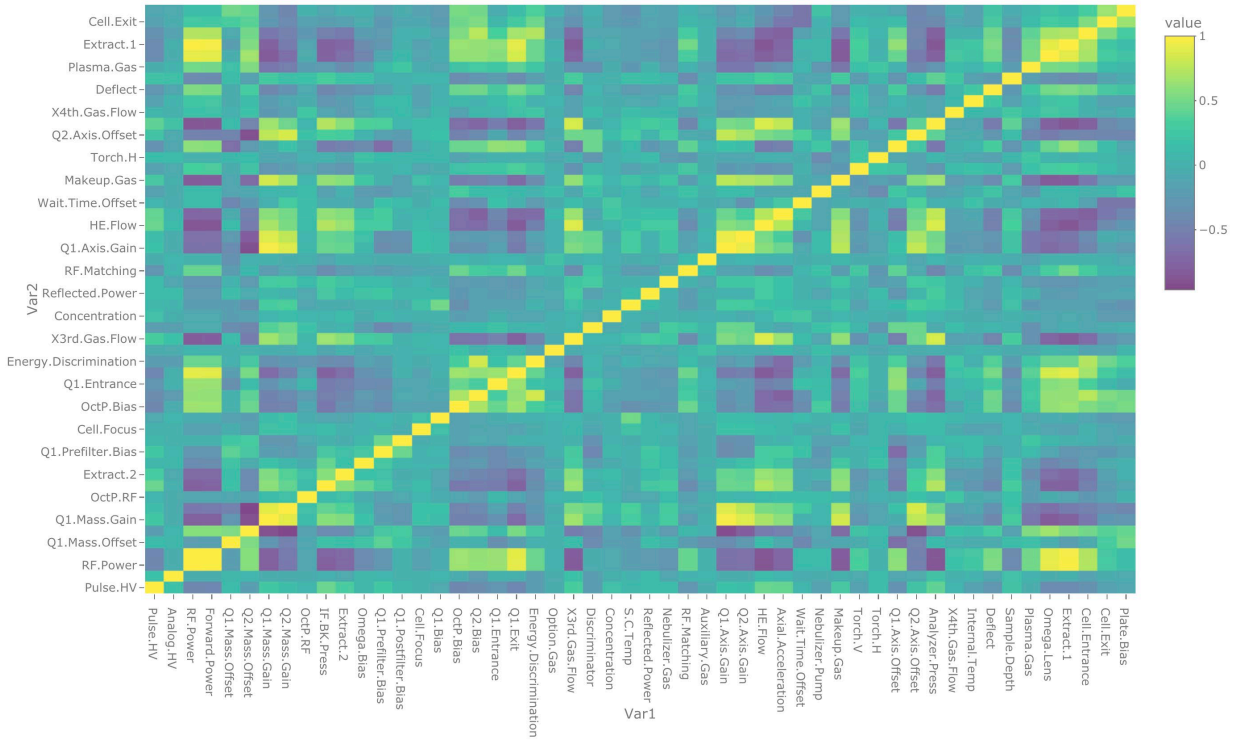

Figure 6. Pearson's correlation between predictors used in our initial models.

3.3 Principal Component Analysis (PCA) with metadata

3.3.1 Full dataset

We also assessed the some of our expected assumptions via principal component analysis. Our assumptions asserted that 1) analysts use measurably different settings for different methodologies and 2) analysts use differing settings for different molecules.

As expected, we were able to observe differences in clustering based on method, plama tempurature, additional elements in the solution, and the primary element of observation.

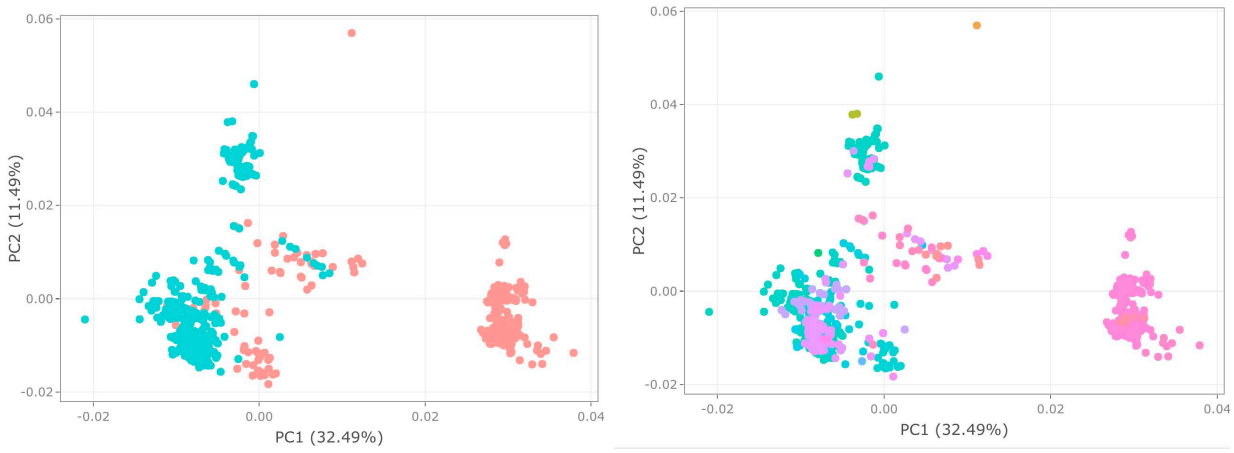

Figure 7. PCA of all varying numeric dataset predictors, distinguishing (left) tandem vs. single MS used in a run as well as (right) different targeted masses in a run.

3.3.2 Thallium dataset

In addition to confirming our assumptions for the overall dataset, we also looked within element parameters to consider possible “groupings” of settings that analysts perfer across elements.

While we note distance in these assessments, these do not indicate necessarily batch effects, but different settings used over time. By comparing the sensitivity observed within each of these clusters, however, we can better determine if we are seeing variation of the instrument performance that we would hope to capture in localizing models. For example, the settings driving the PC1 distance in for the most measured element thallium (element corresponding to 205) are depicted in Fig. The measurements in each of these groups seem to have mostly similar responses in terms of sensitivity but the settings themselves are varying over time. Range especially correlates with cases of low sensitivity.

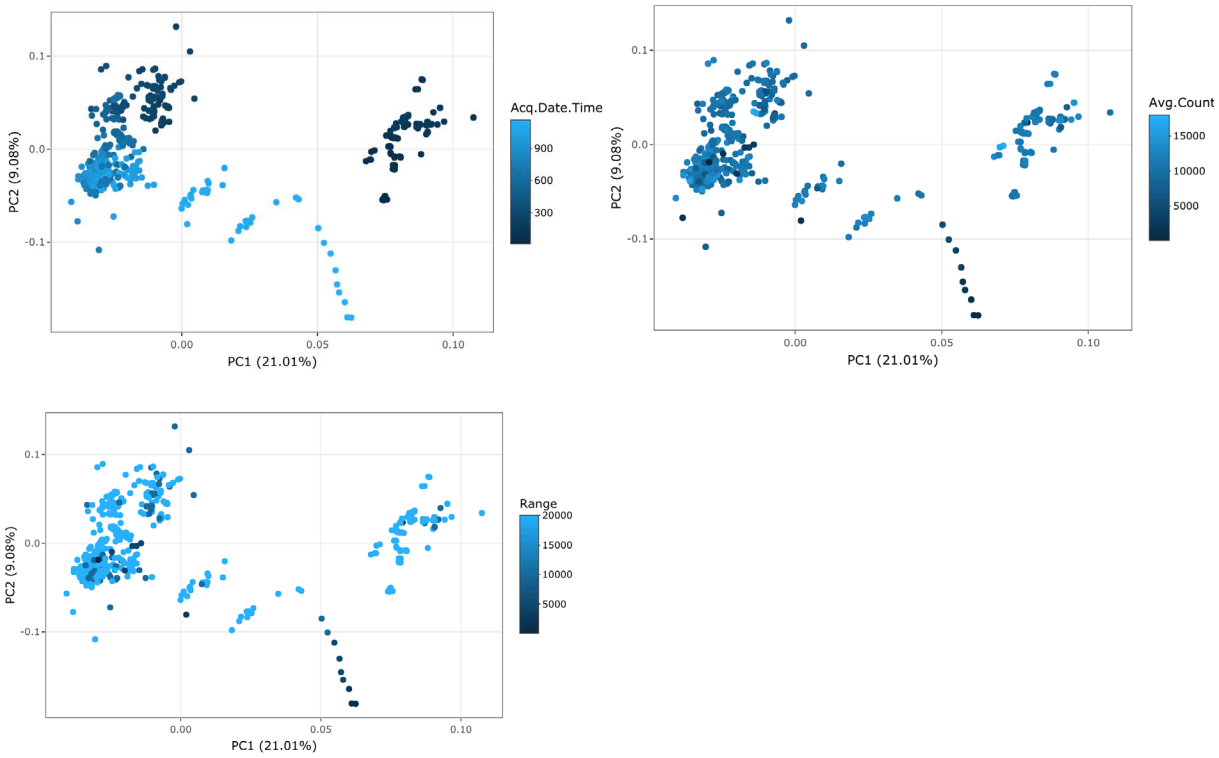

Figure 8. PCA of all varying numeric dataset predictors for thallium, colored by continuous variables of interest including (top-left) acquisition time, (top-right) our response variable Average count, and range (bottom-left).

For the thallium dataset, loadings along the x-axis (PC1) where highly influenced by Q1 and Q2 mass gains and offset values (Figure 8). Discussing with our analysts, the most likely reason for these variables as driving forces is that they are rarely changed unless a full tune is run on the instruments, tweaking all settings at those times. Therefore, we suspect the primary differences between clusters on the x-axis relates to those times of fully re-tuned settings. For PC2, the strongest drivers also include some of these features that are adjusted during full-tunes as well as more regularly changed features like Plate bias and Q1 bias.

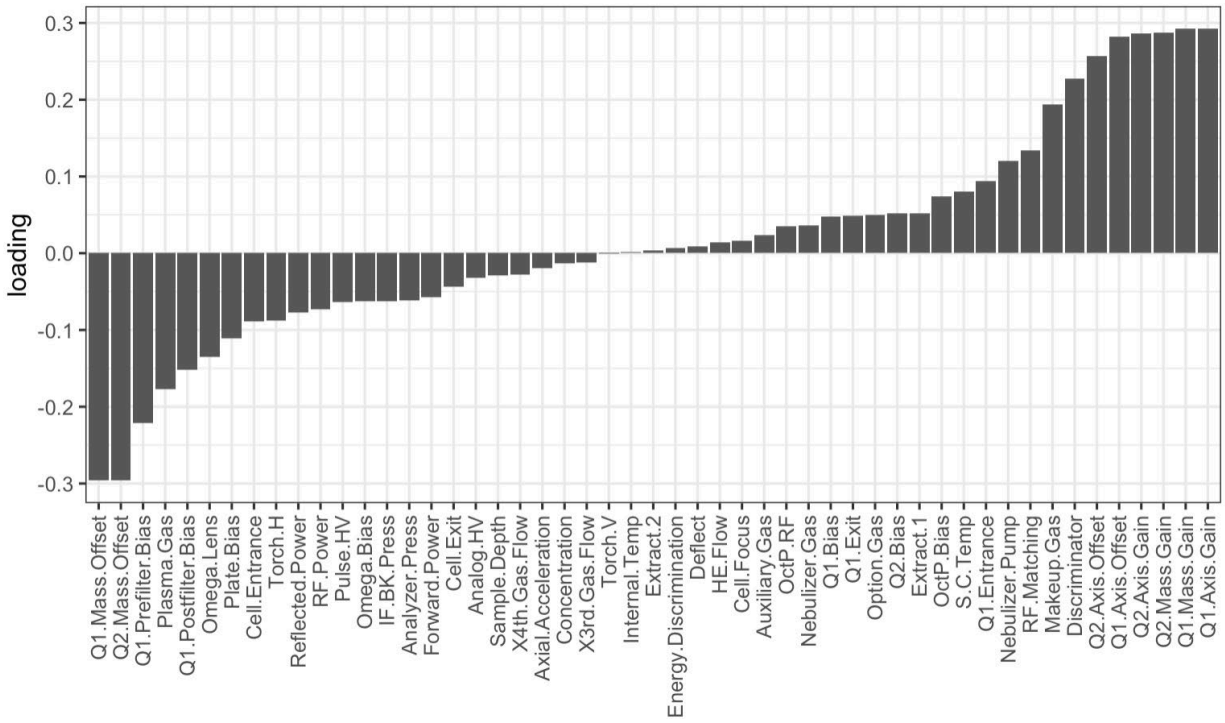

Figure 9. PCA loadings of all varying numeric dataset predictors for thallium.

3.4 Correlation of observations through time

One of the trickiest and most integral parts of our data the variation and dependence of sensitivity over time. Independence of observations is not something we can assume, which can lead to biased estimators in models like linear regression. To get a clear picture of the relation of time to our response observations, we consider the lagged correlation for our data specifically for thallium (205) with a range of 20,000.

Breaking the data into relational components, we broke the response variable into 8 groups of 64 observations to assess the trends over time. The overall trend appears to be non-linear, but we notice possible outliers even in the observations restricted to range 20,000. We were not able to discern the validity of these observations at the time of writing (16Sep2022).

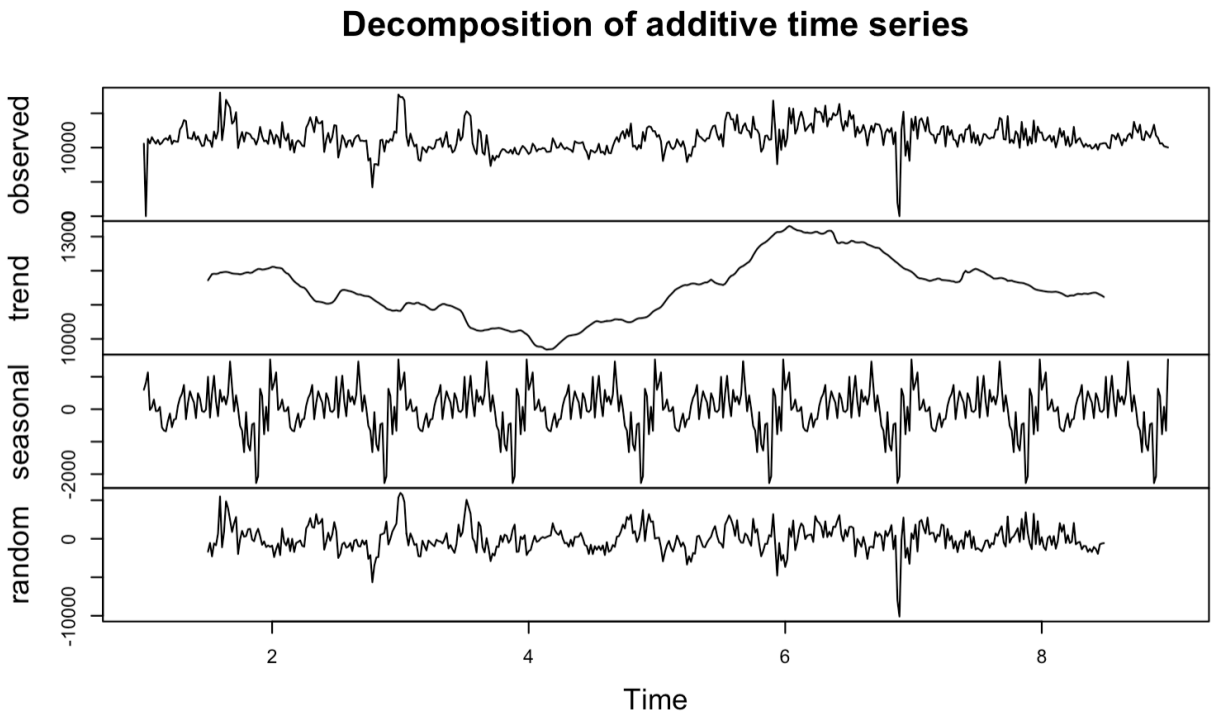

Figure 10. Decomposition of response values for thallium with detection set at range 20,000. Dataset was divided into periods of 64 observations.

Utilizing auto-correlation functions in R, nearby timepoints appear to more strongly correlate than distant ones, as would be expected with instrumentation variability and repetitive use of the same settings. We notice that when the linear dependence is removed from previous lag periods, the autocorrelation diminishes more rapidly as lag increases.

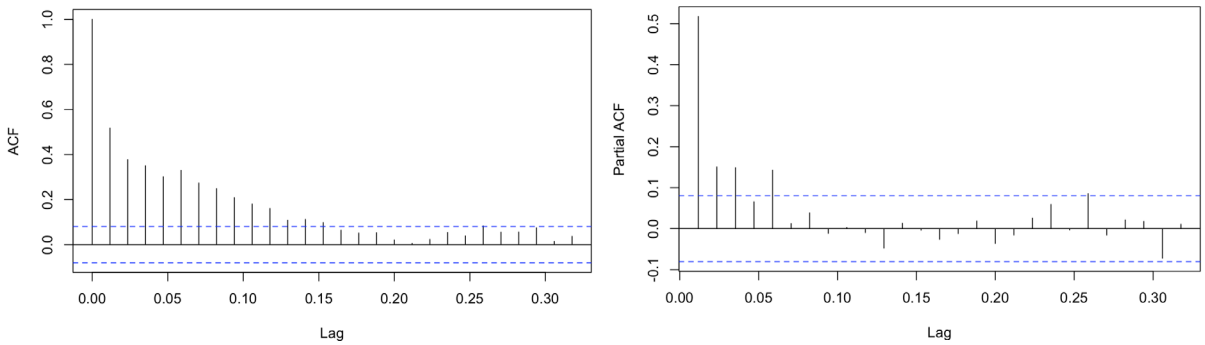

Figure 11. (Left) Auto-correlation and (Right) partial autocorrelation on values for thallium with detection set at range 20,000.

4.0 Initial models

We assume predictive power of the instrument settings to be able to give us accurate estimations of sensitivity, however are not yet ready to compare across molecules due to lack of historical data. Due to the most common occurrence of thallium in our historical data, we model the predictiveness of our variables using only thallium measurements in the following models.

4.1 Linear modeling approach

4.1.1 Linear modeling performance

Considering the possibility of a simple linear combination between all of our predictors, a straightforward linear model is able to moderately capture much of the trend of our dataset. Our R-squared value resulted at around roughly 70% in this model (Figure 12).

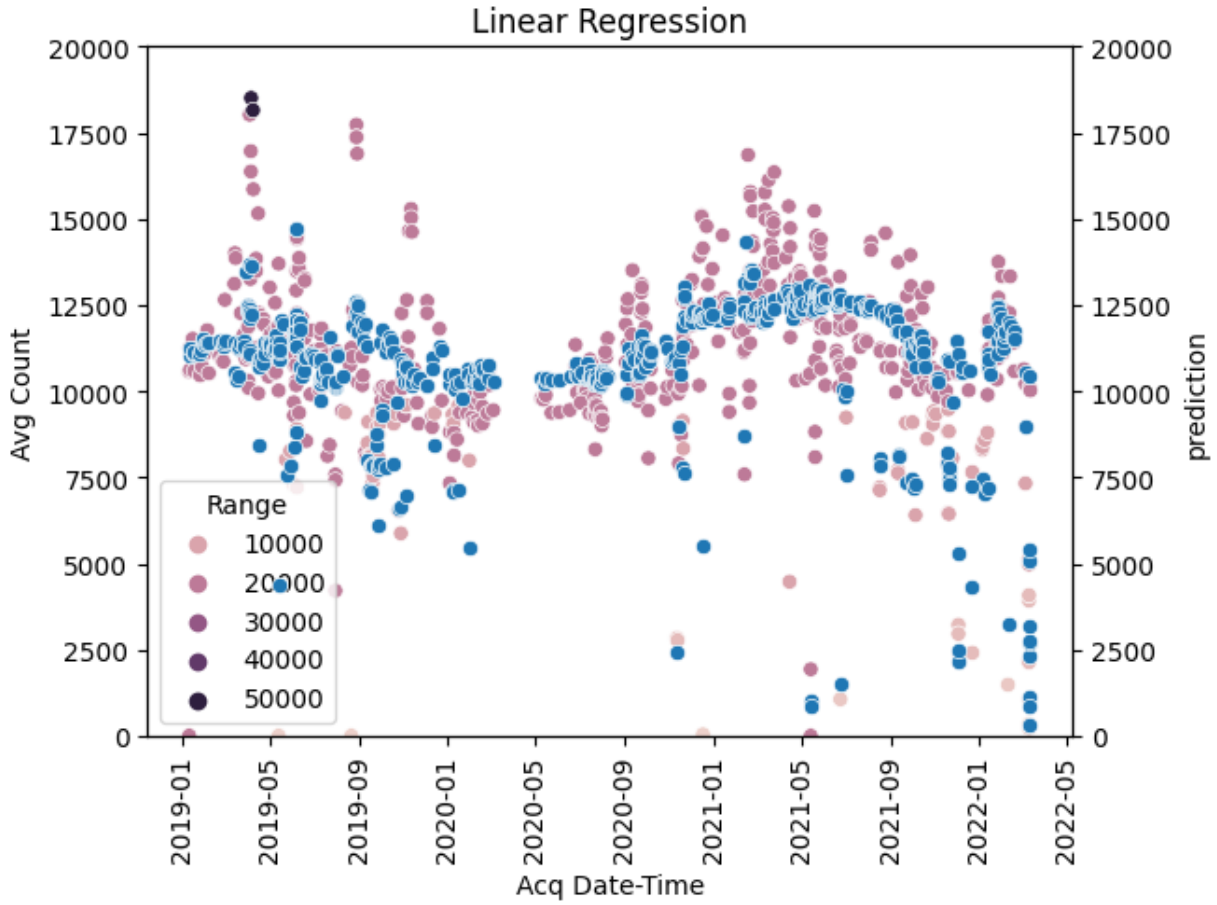

Figure 12. Predicted response generated by linear model plotted on top of observed responses.

4.1.2 Strongest predictor variables

In the linear model, the significant predictors include Range, Use Gas, and Plate Bias, (p-value < 0.001), followed by Acq Date Time, Auxiliary Gas, Extract 1, 4th gas flow, Axial Acceleration, and Analog HV (p-value < 0.01). However, the factor levels of some of the strongest predictions are extremely disproportionate, and in a similar linear model of residuals, Range appeared to be significant for explaining where these higher levels of error occurred. This is a somewhat intuitive result, as Range directly determines what proportion of the gaussian signal from the element is reported (Figure 13).

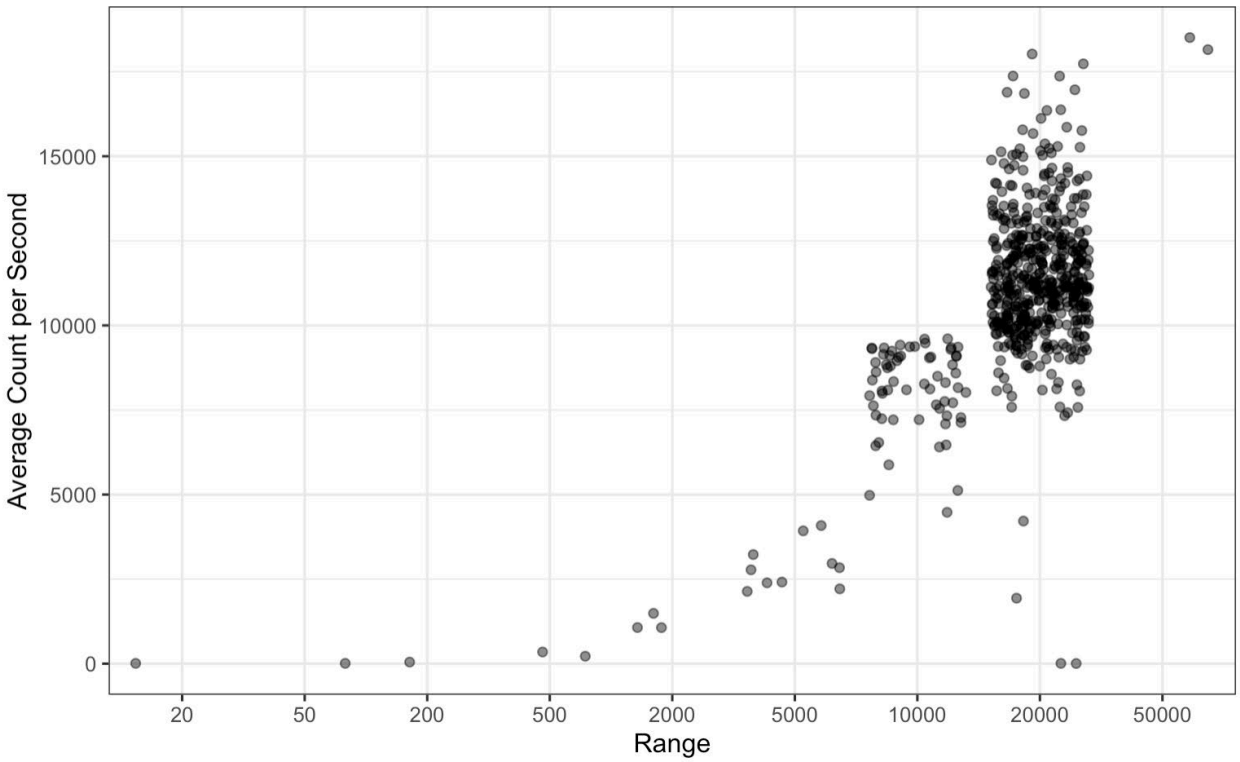

Figure 13. Range's effect on the response variable, where increased ranges report a larger signals.

4.1.3 Range Subset

Referencing the figure above, we recalculated our explained variability in further refined data, where Range is equal to 20,000. Under this new dataset, The Adjusted R-squared value dropped to 0.3328. Based on the outliers in the residual, we also removed two outliers that had response measured well below the other points in the data (< 5000). Despite, the low variance explained, we next considered how close our predictions are to the original values. With a range set at 20,000 for thallium, we were able to achieve a majority within 10% of the predicted value (65.8%, ~1,000 cps) and most observations within 20% of the predicted value (89.1%, ~2,000 cps). Using a modified linear model robust to outliers from the MASS package in R, we

observed modest improvements under the same circumstances (69.4% and 91.2%, respectively) (Figure 14).

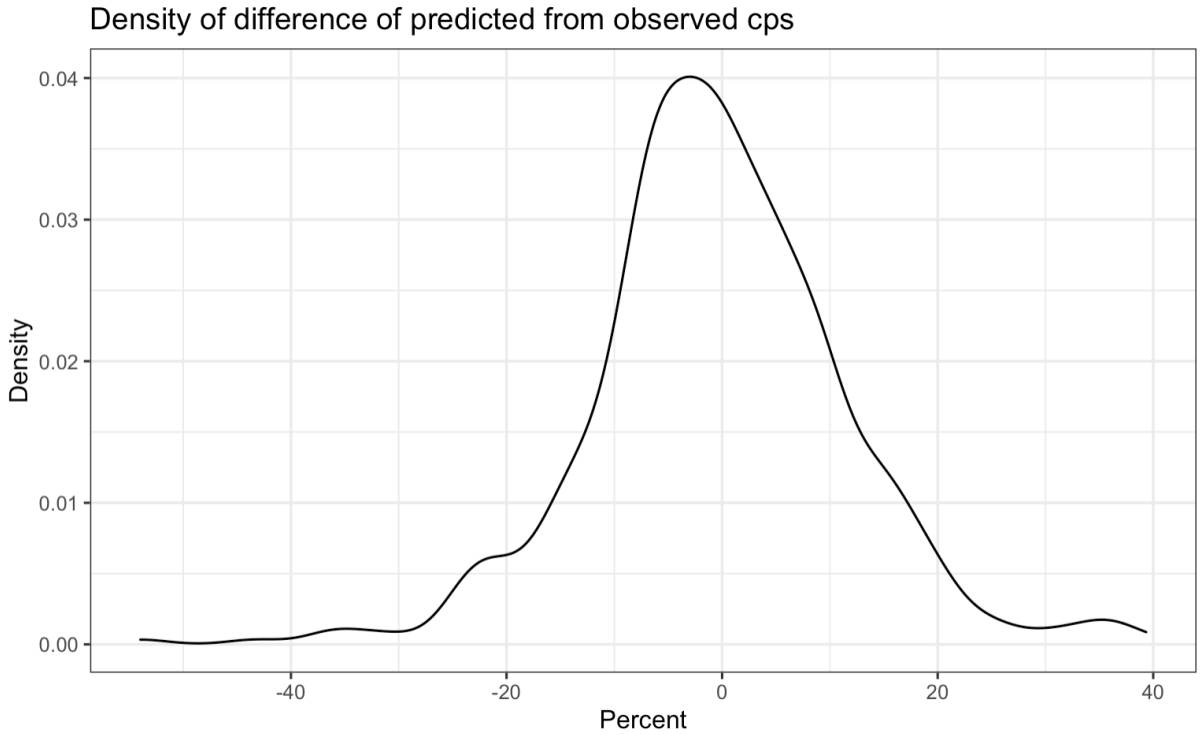

Figure 14. Difference between robust linear model predictions and observed values as a percentage.

4.2 Random forest approach

4.2.1 Random forest performance

Random forests are often considered because of a relative lack of assumptions made about the data used for modeling as well as the innate interaction properties picked up by the model. Using a regression-based random forest with 70% of our data used for training with 30% hold-out, the average performance across 100 randomly generated subsets of our data captured ~75% of the variance in our hold-out data and ~73% of the variance in our training data (Figure 13). This approach appears to capture much more of the variability present in the historical observations for thallium, which we suspect to result from innate interaction effects picked up in the random forest model.

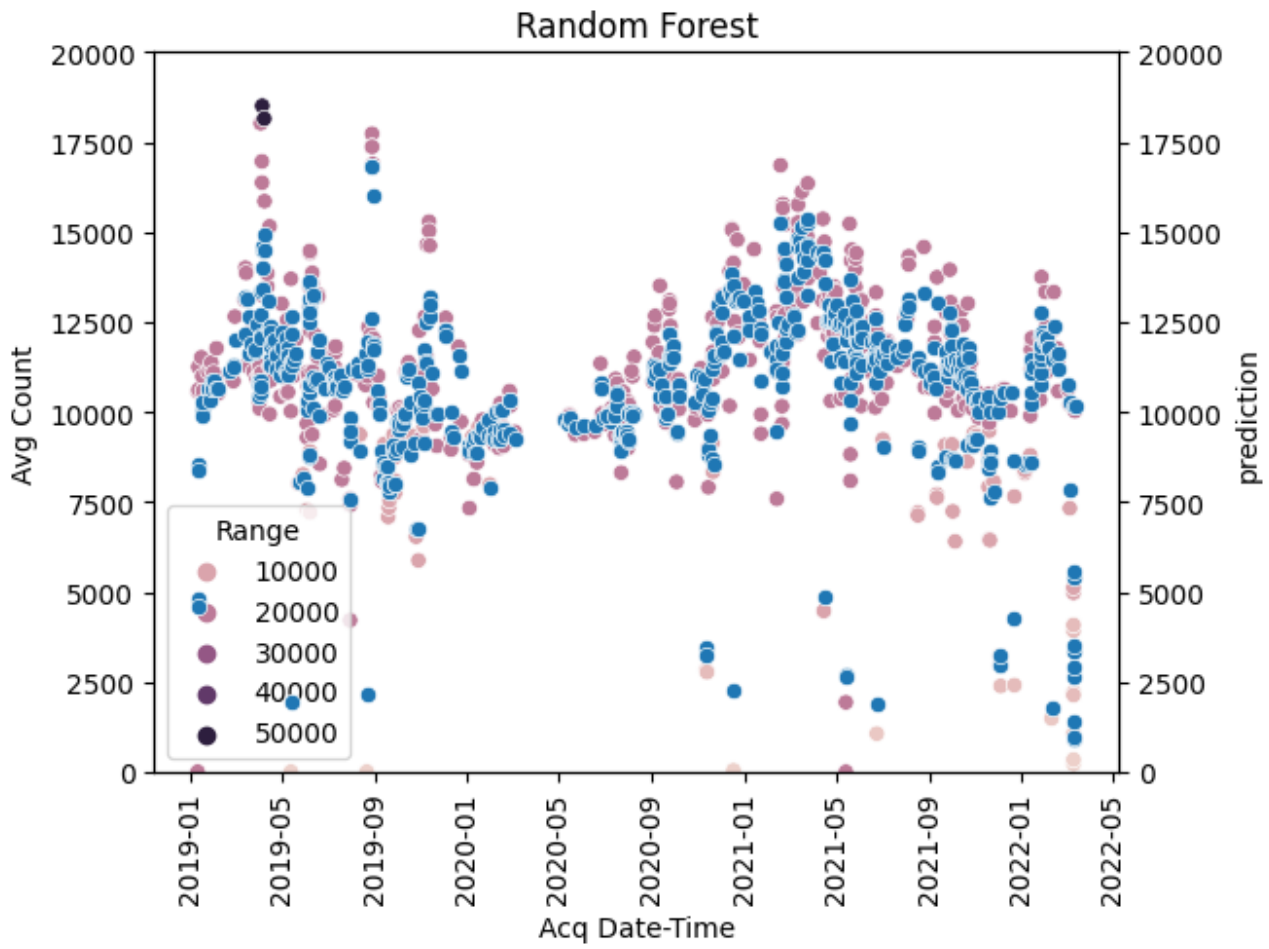

Figure 15. Predicted response generated by average-performance random forest model plotted on top of observed responses. Parameters for the random forest were set to 2000 total trees with 17 random variables used in each tree.

4.2.2 Strongest predictor variables

The strongest variable in our average-performing random forest model is unsurprisingly detection range given its direct influence on the response variable and our precious results from our linear model (Figure 14). Plate bias, analyzer press, sample depth and omega lens also rank highly with a few other predictors, but diminishing returns are seen past the first 10 predictors. As expected from the trend noted in the decomposition matrix, acquisition time also ranks highly for prediction in the dataset. Strongly reported variables have been confirmed by our MS analysts as commonly tuned features.

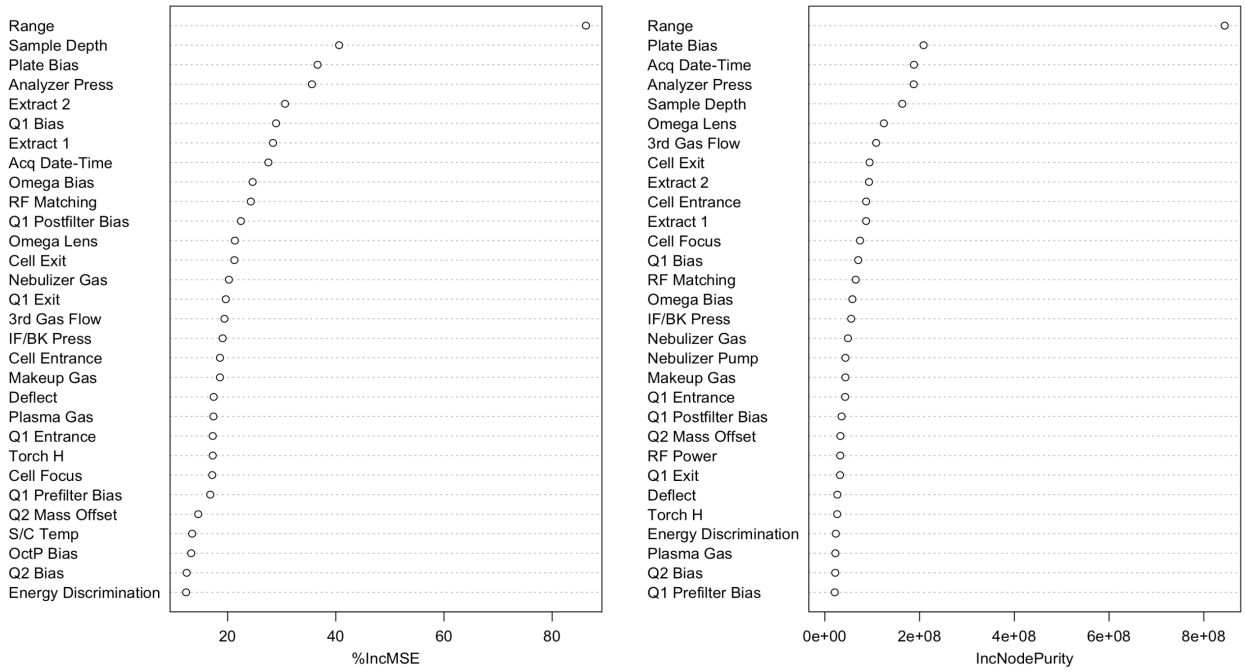

Figure 16. Node purity and MSE improvements with the inclusion of predictors in random forest trees.

4.2.3 Range Subset

As with our linear forest model, we also consider a restricted range data performance. Of 10 different 70% training and 30% test subsets of the data, average performance ranks at ~50% variability explained in both test and training results. Considering the distance of predicted response from the observed values, 97.4% of the predicted values fall within 20% of the observed value, and 88.9% of predicted values fall within 10% of the observed values.

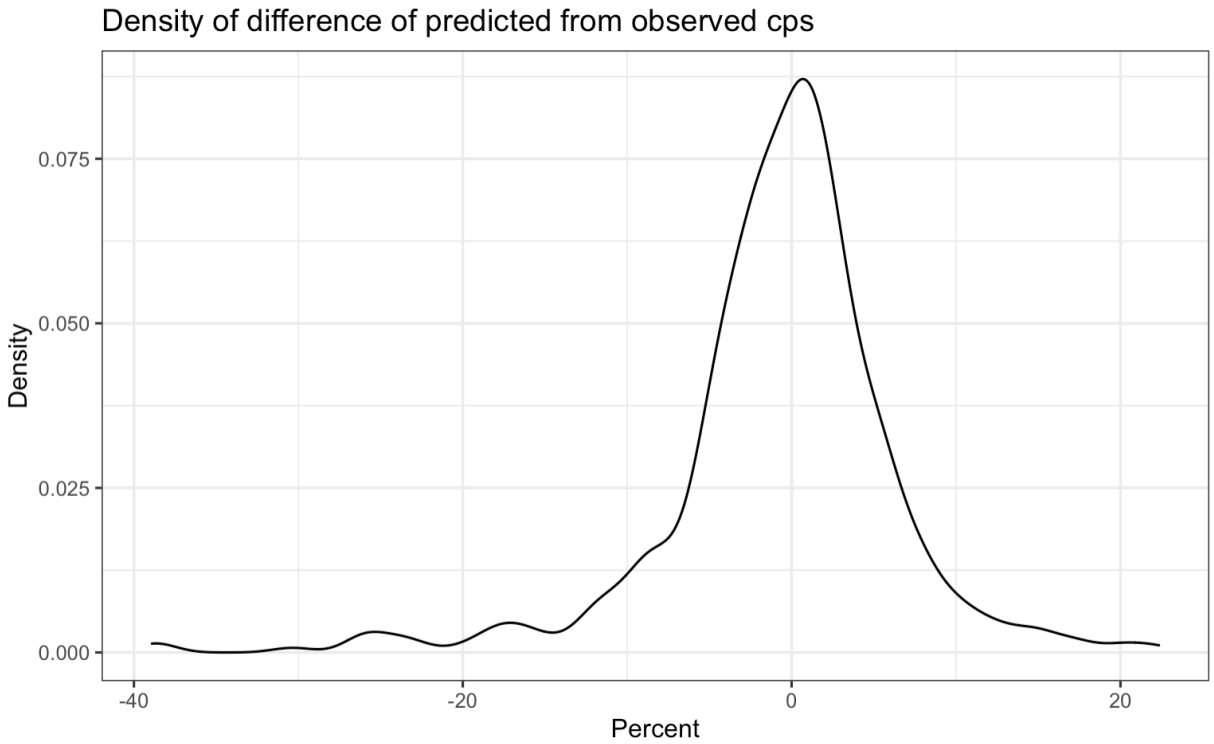

Figure 5. Difference between random forest predictions and observed values as a percentage.

Discussion and Future Directions

We have completed much of the initial investigation of our data, with more questions in mind to investigate in future discussion. From our results, we can see that our Random forest fits our data with much better trending than simple and robust linear models. We attribute this to interaction effects innately considered in the random forest, but in future work we would like to verify this by extracting possible interaction effects indicated by the random forest and adding those into our linear models. We suspect this also may make-up for the differences in the most strongly weighted variables between the two models.

Investigation of the time dependence of our data also leads us to consider how our test and training datasets may be flawed. During our current work, we sampled randomly for each of these, but it may be more appropriate to divide and holdout specific blocks of time and evaluate the differences in these time blocks further.

As current work stands, we are particularly interested in determining what is causal to the remaining unexplained variance in our models. We suspect that another variable we ought to consider in our analyses is proximity to other sample measurements – i.e., we expect that running the instrument with certain elements may leave carry-over effects for subsequent runs. The greatest barrier to understanding this relationship is our current relative lack of data in general and across multiple measured elements. In addition, additional data is required for a more robust model and future development of the neural network and transfer learning models we hope to achieve. We hope that in future investigation, we might have dedicated lab time planned out to generate custom data for the project or pull in data from other interested collaborators such as commercial instrument manufacturers and commercial labs where these instruments are also in regular use.

5.0 References

1. Puneet Mishra, Dário Passos, Realizing transfer learning for updating deep learning models of spectral data to be used in new scenarios, Chemometrics and Intelligent Laboratory Systems, Volume 212, 2021, 104283, ISSN 0169-7439, https://doi.org/10.1016/j.chemolab.2021.104283.

2. Puneet Mishra, Dário Passos, Deep calibration transfer: Transferring deep learning models between infrared spectroscopy instruments, Infrared Physics & Technology, Volume 117, 2021, 103863, ISSN 1350-4495, https://doi.org/10.1016/j.infrared.2021.103863.

3. J. Padarian, B. Minasny, A.B. McBratney, Transfer learning to localise a continental soil vis-NIR calibration model, Geoderma, Volume 340, 2019, Pages 279-288, ISSN 0016-7061, https://doi.org/10.1016/j.geoderma.2019.01.009.

4. Ahmad Alwosheel, Sander van Cranenburgh, Caspar G. Chorus, Is your dataset big enough? Sample size requirements when using artificial neural networks for discrete choice analysis, Journal of Choice Modelling, Volume 28, 2018, Pages 167-182, ISSN 1755-5345, https://doi.org/10.1016/j.jocm.2018.07.002.

Appendix A – Link to Agilent reference

Calibration settings are best described by publicly available technical documentation of the Agilent 8900, which can be located at the following URL:

https://www.agilent.com/cs/library/applications/5991-6943EN.pdf

Pacific Northwest National Laboratory

902 Battelle Boulevard
P.O. Box 999
Richland, WA 99354

1-888-375-PNNL (7665)

www.pnnl.gov