CCSP 2.0: An Open Source Jupyter Tool for the Prediction of Ion Mobility Collision Cross Sections in Metabolomics

Facundo Fernández: ____________________

Michael Evans: ____________________

Table of Contents

Abstract

Tandem mass spectrometric methods revolutionized the chemical identification landscape, allowing serums and molecules to be separated in two or more dimensions. Ion Mobility Mass Spectrometry workflows combined with liquid or gas chromatographic separation have continued to progress chemical identification and further increase the amount and confidence of these identities. Such advancements have also given birth to a new molecular descriptor: the Collision Cross Section, sparking heavy interest in the analytical-computational chemistry to compile these values for known molecules. The main shortcoming has been predicting the CCS value for new molecules such as Poly-Fluorinated Alkyl Sub-stances. Preliminary prediction software has revealed that predicting CCS values for this molecular class is possible, but it can prove temporally, computationally, and financially expensive between different licenses and genetic algorithm. This work combines open-source Python modules (NumPy, Mordred, Pandas, etc.) to construct an alternative workflow that is completely free and capable of running on a mid-specification laptop within a half hour. Using the M-H and combined M+H and M-H datasets taken from the McClean CCS Compendium, median prediction errors of 2.07% and 1.84%, respectively, were found using Support Vector Regression within 5 minutes on a mid-spec laptop, satisfying the 2.50% benchmark. This overall success illustrates the power and versatility of this workflow to produce low errors with datasets as large as 1300+ molecules and as few as 37. This script can be distributed on file-sharing sites like GitHub where other users may customize the free source code to fit their experimental needs.

Introduction

The advent of mass spectrometry (MS) in 1927 reinvented chemical identification, and in much the same way, ion mobility MS (IM-MS or IMS) revitalized the field in 1963 and yet again in 2016. Despite these new techniques, still only 2-10% of compounds detected in a non-targeted metabolomic study can be identified with high certainty.[1] Non-targeted metabolomic studies focus on identifying as many detected compounds, known as metabolites, as possible within a serum or solution that can contain thousands of unique metabolites. By ionizing these compounds in a magnetic field, cationic and anionic fragments can be sorted into distinct mass-to-charge ratios (*m*/*z*). The highest degree of uncertainty stems from the resolution of mass spectrometers. Thousands of molecules can have a fragment with 100.123 *m/z* value, and even more can have the same molecular formula but different construction. Without the ability to increase the accuracy by additional orders of magnitude, many mass spectrometers will struggle to separate molecules with a difference of ±0.0001 in their respective *m/*z values.[1]

In the 1950's, tandem methods were developed to make a larger distinction between compound fragments. By feeding complex samples into liquid and gas chromatography (LC and GC, respectively), individual components could be separated before being fed into a mass spectrometer to fragment separately from other ions with the same *m/z* ratio (LC-MS or GC-MS methods) facilitating even higher chemical distinction.[1-4] IM-MS uses this same logic but instead focuses on the size and charge distribution of the molecule. It measures the time required for the molecule to cross a detector in the presence of an opposing inert gas, commonly N_2 or a noble gas.[1,5-8] From this methodology, researchers discovered a new dimension of molecular annotation called Collision Cross Section (CCS). CCS values are molecular annotations directly related to the size and shape of a molecule based on the drag produced in the drift tube, producing a rotationally averaged molecular “shadow.” While powerful in tandem IM-MS, CCS shines particularly in workflows such as LC-IM-MS which plot *m/z* versus t_R versus CCS, adding a new dimension of refinement to chemical identification.[1,4]

The -omics fields have rigorously tested this annotation since the commercialization of travelling-wave IMS (TWIMS) by Waters Corp.[4] Kanu *et al* sought to standardize successful IM-MS experiments, requiring the completion of 5 basic processes: sample introduction, compound ionization, ion mobility separation, mass separation, and ion detection.[4,7] Paglia *et al.* performed an interlaboratory evaluation of the Waters Corp. Synapt G2 wherein different labs across the United States and Europe analyzed the calculated CCS values of the same 125 chemical species via TWIMS. Interlaboratory results yielded a relative standard deviation (RSD) <2% of the expected CCS value for 97% of the species, indicating a high accuracy.[5] Additionally, results across the different labs showed <5% RSD for 99% of the species and later confirmed by similar experiments by Stow *et al.*[6-7] Nye *et al.* demonstrated that inherent flaws to the construction of LC columns ultimately caused inaccuracies in LC-IM-MS-derived CCS values that could be avoided using TWIMS-based workflows.[17] The combined result cast the spotlight on the robustness and reproducibility of the CCS value but also its largest downfall, particularly in chromatography-based workflows. However, for as much attention as it had garnered, few databases existed with a comprehensive compilation and even fewer predictors.

Two schools of thought emerged for these predictors: Quantum Calculation (QC) and Machine Learning (ML). The first of the QC approaches was *Mobcal* in 1996. *Mobcal* built rotationally averaged cross sections from varying atomic coordinate files such as X-Ray Diffraction (XRD), Nuclear Magnetic Resonance (NMR), molecular dynamics, and quantum calculation, but as programming languages and computing power improved, the database was quickly outmoded [8,18] Zanotto *et al.* constructed *High Performance CCS (HPCCS)*, a modern reworking of *Mobcal* that uses a Quantum Trajectory Method to improve prediction accuracies and run times.[8-9,18] These quantum approaches are useful in the lack of available data, facilitating *ab initio* and *de novo* approaches to CCS prediction, however these workflows tend to be computationally demanding.

ML approaches offer lower computing power requirements with comparable accuracies. One of the main caveats is the requirement for a training data set, disallowing the approaches offered by the quantum workflows. Proteomic ML approaches began to dominate workflows by 2016, offering 1-3% median errors in smaller scale databases.[20] By the end of 2019, numerous metabolomics-focused databases were developed, including ISiCLE, DeepCCS, MetCCS, and AllCCS.[9-13] However, these databases fall short because they depend on tens to hundreds of thousands of predicted CCS values based on training sets limited both by numbers of molecules and even moreso by molecular classes.

A desire to better control these calibration and validation sets arose within the field that could theoretically allow more accurate descriptor calculation and thus more powerful CCS prediction. Soper-Hopper *et al.* developed a PLS-based workflow that combined with Dragon 7.0 molecular descriptor software with genetic algorithm variable selection. Calculated CCS values were within <2% median error for 500 molecules of varying class. The primary issue stems from the time and computing power required by the genetic algorithm, however there are also financial barriers to such analysis as Dragon 7.0 (now alvaDesc) license cost upwards of $1,300.[15] Genetic algorithms procedurally iterate on a base model that optimize around a set of parameters. Specific time measurements for the genetic algorithm were not available, but the evaluation of thousands of individual descriptors would theoretically require significant permutations and time commitment on the order of hours to even days.

This proposed work aims to recreate the results from Soper-Hopper *et al.* with minimal effort and time expended on the end-user's behalf. The proposed workflow eliminates the genetic algorithm, and replaces PLS regression with support vector regression (SVR). This work also adds the ability to automatically randomize the calibration and validation data sets which makes resulting prediction less reliant on a particular data set that introduces bias error. This work would ultimately unlock real time CCS prediction for raw data. By packing the script using a Jupyter Notebook and distributing via GitHub, this work will allow any user direct access to the source code and the ability to customize the script to whatever the study requires, improving end-user accessibility.

Literature Review

Metabolomics—the study of small, molecular products of metabolism—is one of the younger -omics fields, established in 1998. Perhaps the greatest hindrance to the field as a whole is the "Dark Metabolome" as non-targeted studies seldom identify more than 2-10% of detected compounds with high certainty. This problem arises from many molecules producing mass fragments with similar *m/z* values in the ten thousandths place (0.0001), making resolution a primary limiting factor.[1] To combat this, two hallmark methods in analytical chemistry were developed: orthogonal analytical workflows such as Liquid Chromatography linked to Mass Spectrometry (LC-MS) and tandem mass spectrometry (MS-MS).[1-4] The advent of ion-mobility mass spectrometry (IM-MS) and its variants (travelling-wave, drift tube, etc.) added a third dimension of coupled analysis by introducing a powerful, new molecular descriptor: Collision Cross-Sections (CCS).[1,5-7] This new angle was seized by computational laboratories, developing new workflows and databases to accurately predict, compile, and transmit these values to other groups.[9-16] Establishing a reliable means of parsing through the remaining 90% of compounds remains an unresolved issue, particularly as new metabolites like poly-fluorinated alkyl substances (PFAS) emerge almost daily.

The modern interpretation of IM-MS has changed quite drastically since its inception by Dr. Earl McDaniel in 1963 from a method to facilitate mass spectrometry to an analytical branch within MS workflows with its own variances.[1-2,4] To focus this rapid broadening, Kanu *et al.* stated successful IM-MS experiments must accomplish 5 basic processes: sample introduction, compound ionization, ion mobility separation, mass separation, and ion detection.[4] Ion mobility and mass separations are of particular note as they represent the groundwork for CCS values derivation; they originate from molecular drag against opposing neutral gases such as N_2. Waters

Corp. commercially introduced Synapt HDMS in 2006 and began to popularize travelling wave ion-mobility MS (TWIMS), instrumentation that has since been redesigned into the TWIMS gold standard, the Synapt G2.[4,7] A 2014 experiment by Paglia *et al.* and a similar 2017 experiment by Stow *et al.* demonstrated the accuracy and reproducibility of CCS values by performing interlaboratory evaluations across different labs in the United States and Europe. These studies produced <2% RSD to expected CCS values for 97% of the involved species, indicating high accuracy, and <5% RSD for 99% of species when comparing the results between labs, illustrating high precision.[4-7] These studies served as a launching point for computational CCS prediction for molecules. However, the lack of compatible CCS calculation software stymied this interest.

One of the first of these predictors, *Mobcal,* was developed Dr. M.F. Melseh and coworkers in 1996, computing rotationally average molecular cross sections based on experimentally determined atomic coordinate files such as NMR data, X-ray scattering, and quantum calculations over 15 years.[8,11,13] Since then, numerous CCS prediction software packages have been developed using a variety of techniques that eventually outmoded *Mobcal*, leading Zanotto and co-workers to construct a modern rendition of *Mobcal—High Performance CCS (HPCCS)—* which produced more accurate predictions at a faster rate.[11] By 2016, machine learning approaches began to dominate workflows, boasting median relative errors between 1-3% in smaller scale databases.[8-13,16]

Zhiwei Zhou and co-workers realized this trend and called for further development into large-scale metabolite CCS predictors with a focus on machine learning techniques.[10] METLIN—a database created in 2005—was one of the first published metabolomics predictor packages, adding the CCS prediction functionality.[12] ISiCLE, DeepCCS, and MetCCS, and

AllCCS followed suit, decreasing relative errors for more of the included molecules. Most notably, the latter contains <4% relative error for 84% of the predicted molecules.[10-14] One of the major shortcomings with all of these predictors is their lack of flexibility which plays a role in the higher errors for the other predicted molecules. One of the key components of ML workflows is an available training dataset to run the predictions. MetCCS used just under 400 training molecules to predict 35,000 CCS values. AllCCS used similar tactics in positive and negative ion modes (n = 1851 for positive; n = 795 for negative) alongside an 80% calibration-validation split to predict over 2 million molecules. These algorithms calculate the CCS values of new molecules using the same algorithm each time. This repetition may not account for the structures and tendencies of new classes of molecules, leading to increased errors that lower the effectiveness of these ML workflows as new metabolites are reported.[10-14]

Soper-Hopper *et al.* sought to change this trend. Rather than using one fixed dataset to build a model and predict CCS values for novel molecules, a workflow capable of downloading a custom calibration and validation dataset and built a model was constructed. Two-dimensional molecular descriptors for 500 molecules were calculated via Dragon 7.0 software by Kode Solutions. Genetic algorithm variable selection and PLS regression predicted molecular CCS values within 2% median error.[10] This study demonstrated that accurate CCS prediction is possible from curated data sets, but the set-up is not seamless. Data set construction and curation must be completed outside of MatLab and Dragon 7.0, adding to analysis time. Additionally, genetic algorithms are computationally expensive.

As such, this proposed work aims to develop a streamlined CCS prediction workflow that is highly customizable, approachable at any programming skill level, and free of charge to the anyone. Python would be the programming language of choice as it is one of the most flexible and current languages available with a broad selection of open-source software packages. In particular, the Mordred package is capable of calculating 1800+ 2-dimensional descriptors which could be used to train the support vector regression algorithm. Data sets will be constructed from entries in the McLean CCS Compendium with randomized calibration and validation sets based on a ratio the end user can set.[16] After verifying this workflow's accuracy, CCS values will be predicted for newer classes of molecules not found in the compendiums such as poly-fluorinated alkyl substances (PFAS) and polycyclic aromatic hydrocarbons (PAH). From start to finish, we aim to complete the prediction process in less than an hour on a mid-specification laptop.

Methodology

The script described in this work was written using Python 3 in Visual Studio Code (Microsoft). Due to speed limitations during later stages of development, the script was transferred to a Jupyter notebook to process the script more quickly. CCSP 2.0 downloads one of two types of molecular data via Excel Spreadsheet (.xlsx; Microsoft): PubChem CID or InChi. In CID mode, these codes are translated into InChi codes using the RDKit Python module. Files are selected using the Tkinter. InChi codes are then coverted to SMILES codes and then mol objects by RDKit. Mordred reads the mol objects and calculates 1613 2-D molecular descriptors for each and arranges them in a Pandas dataframe. Descriptors containing any errors or constant values were removed. The descriptor matrix is then split randomly into calibration and validation data sets set by a percentage (34-66% respectively, for this work). PLSToolbox is then used to standardize the values via Z-transform and construct an algorithm using support vector regression. This regression pairs penalty (C) and tolerance (ε) parameters to find the best fit. This fit can vary between datasets. A k-fold cross-validation was run on the calibration set, and then the validation set was run. The primary statistics for consideration are median absolute error (MAE) and root mean square error (RMSE), but the script also calculates average percent error, mean bias error, average absolute error, median absolute percent error, and R^2 for the fit.

Initial testing was completed using the McLean Unified CCS Compendium using M+H (n = 703) and M-H (n = 599) subsets. The Compendium was selected as it was a publicly downloadable resource with diverse sets of molecules that could theoretically generate a more robust and flexible algorithm. This was further tested using PAH (n = 61), halogenated metabolites (Halogens; n = 102) and PFAS (n = 37) molecules provided by Baker and Foster at North Carolina State University. These tests were run 1001 times with the median graphs shown in the *Results* section of this work. All sets were divided into a 50-50 Calibration-Validation splits.

A second version of CCSP 2.0 exists where calibration and validation sets are manually loaded, allowing for custom datasets to be uploaded and analyzed. This version of the script functions identically, lacking only the function that splits a unified dataset into two portions. This version was used to run the PFAS set. The PFAS data set was provided by Baker and Foster at North Carolina State University, containing 37 molecules tested in their laboratory. Plots for both versions are constructed using Pyplot from Matplotlib. The below *Figure 1* visualizes the overall workflow for CCSP 2.0.

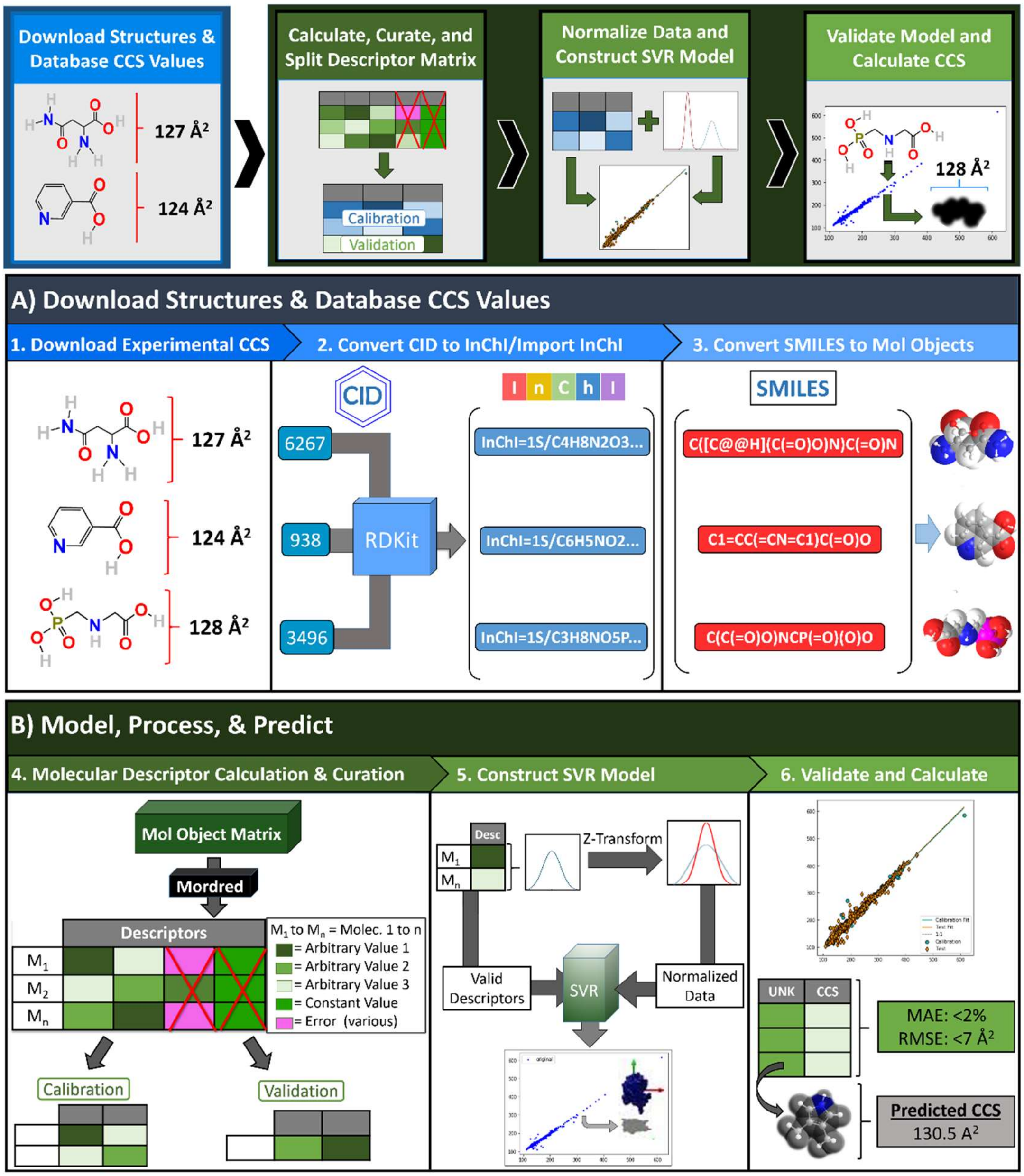


Figure 1: CCSP 2.0 Workflow

The above figure visualizes the approach used in CCSP 2.0. Molecular Identifiers are loaded in and converted to SMILES codes. The Mordred Python package calculates 1613 descriptors, and invalid descriptors are cleared away. Calibration and Validation sets are formed. The Calibration set is used to construct and cross-validate the model, and the Validation set is used as a blank test of the model, producing predicted CCS values.

Results

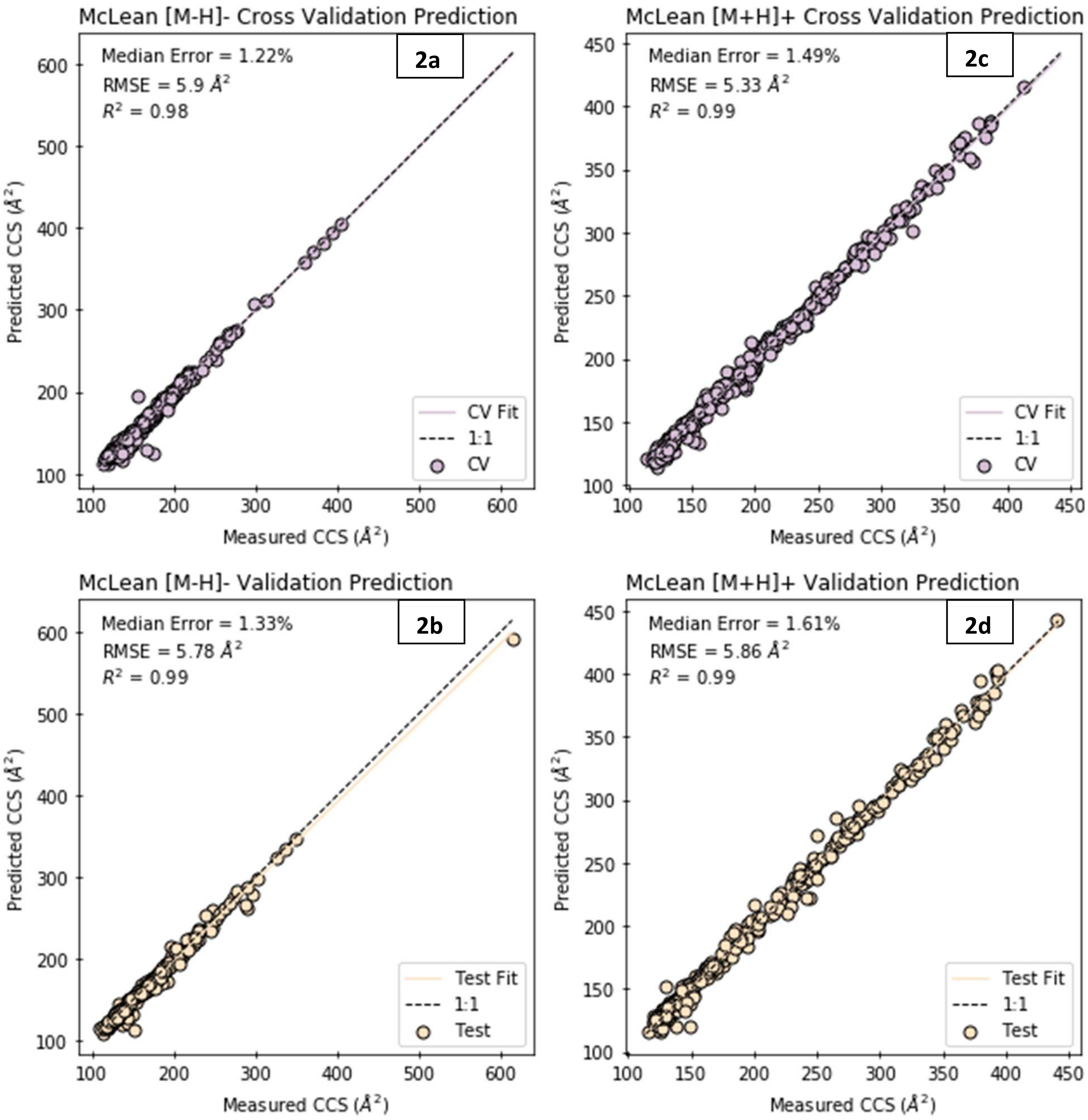


Figure 2: Compendium Prediction Plots (M+H and M-H)

These graphs plot the cross validated (top) and validation (bottom) fits for the McLean M-H (left) and M+H (right). The primary discussion focusses on the validation sets for these measurements as the model did not previously analyze those samples as is not the case with the *k-fold cross validation sets. Validation errors were <1.7% MAE and <6.0 Å^2. These charts were taken from the medians of 1001 runs.*

Accuracy of the model can be visually assessed by the above and following figures where the CCS value predicted by CCSP 2.0 is plotted against experimentally measured/provided CCS value. A perfect match results in a 1:1 line across the graph as highlighted by the dashed line. The combined M-H and M+H Compendium sets provided the most chemically diverse data set for testing and were used as benchmarks. Cross Validation for the M-H set (*Figure 2a*) yielded an R^2 of 0.98 with 1.22% median absolute error (MAE) and 5.90 Å^2 root mean squared error (RMSE). The pure Validation set (*Figure 2b*)—previously unseen by the model—generally reported improved results with a 0.99 R^2 and 5.78 Å^2 RMSE but a slight increase in MAE to 1.33%. The M+H set yielded opposite results. MAE increased from 1.49% to 1.61%, and RMSE rose from 5.33 Å^2 to 5.86 Å^2.

Smaller data sets reflected the M+H results, having increased errors from the Cross Validation to the pure Validation sets. Of these, the PFAS (*Figures 3a and 3b*) reported the highest errors of the applied datasets with 1.36% MAE and 5.48 Å^2 RMSE which are still well-below the target threshold of 2% and 7.00 Å^2, respectively.

The Halogens and PAH sets were two newer data sets originally not included at the start of this work but produced relevant results that warranted their inclusion. As with the PFAS and M+H data sets, the Cross Validation data yielded lower errors, but both were still well within the expected parameters. For the Validation sets, MAE was <1% and RMSE <2.50 Å^2. The immediate reaction tends toward a lower analyte count as each were <100 total while the McLean sets had >400 in both Calibration and Validation sets. However, this claim is not fully supported when looking at the Halogens (*Figure 3c and 3d*) and PAH (*Figure 3e and 3f*) results.

PAH sets had fewer associated molecules but had higher errors in both validation RMSE and MAE.

Error	**Soper-Hopper Workflow[15]**	**CCSP 2.0: M-H**	**CCSP 2.0; M+H**	**CCSP 2.0: PFAS**	**CCSP 2.0: Halogens**	**CCSP 2.0: PAH**
MAE	2.00%	1.33%	1.61%	1.36%	0.49%	0.78%
RMSE	7.00 Å^2	5.78 Å^2	5.86 Å^2	5.48 Å^2	1.46 Å^2	2.43 Å^2

Table 2: CCSP 2.0 Error versus Soper-Hopper Workflow (Validation)

This figure tabulates the MAE and RMSE data from each of the CCSP 2.0 workflows as compared to the literature Soper-Hopper workflow.

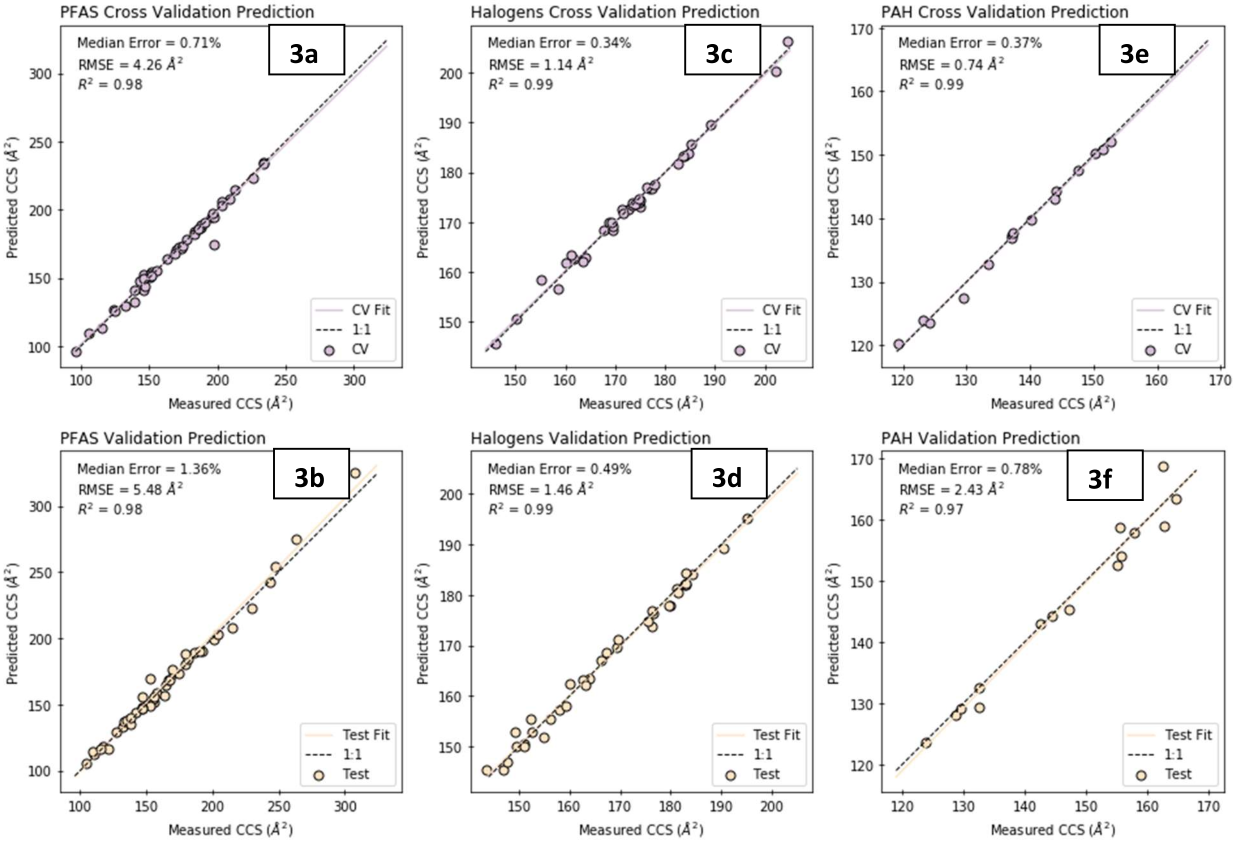
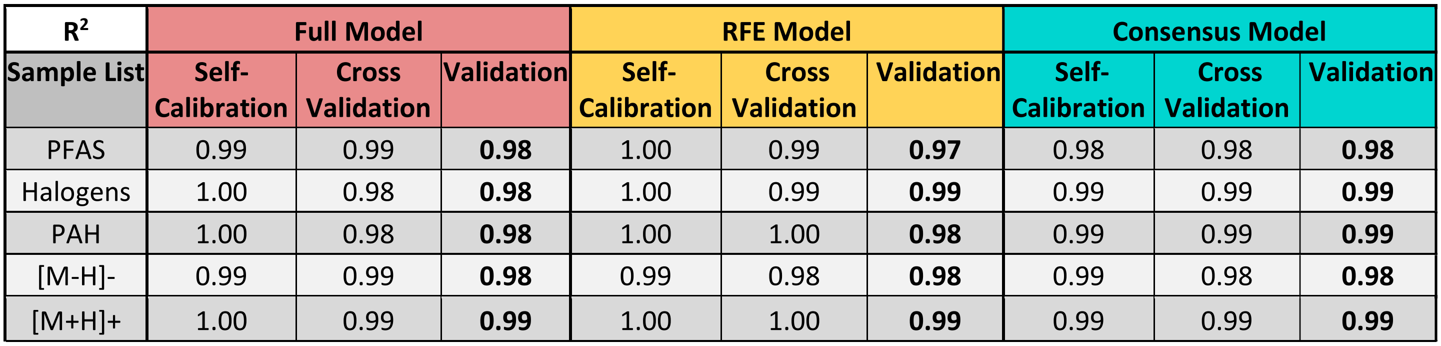


Figure 3: Specialized Data Set plots

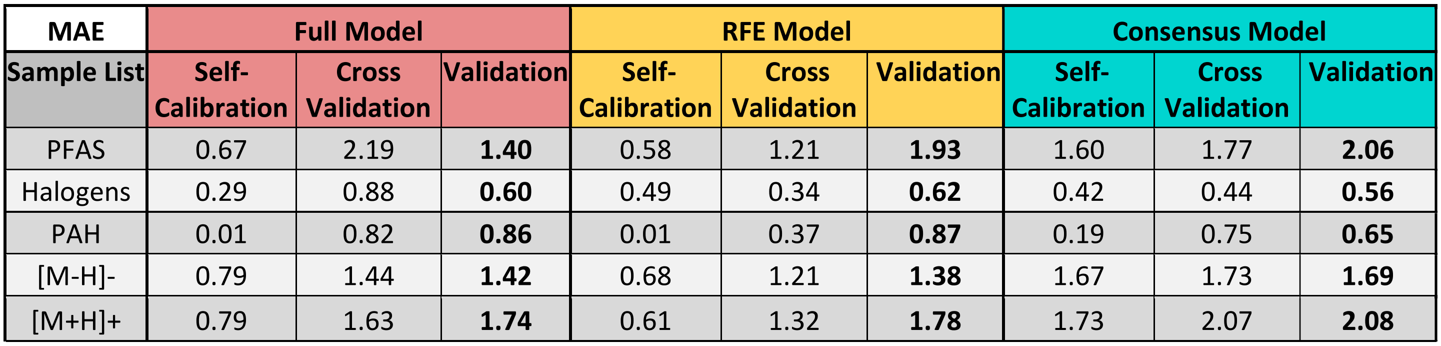
These graphs plot the cross validated (top) and validation (bottom) fits for the PFAS (left), Halogens (center), and PAH (right) data sets. The PFAS data set appears as a sort of bridging *observation between the specialized and the diverse datasets. The much smaller Halogens and PAH sets produced RMSE < 2.5 Å^2 with median errors < 1%. This charts were taken from the medians of 1001 runs.*

.

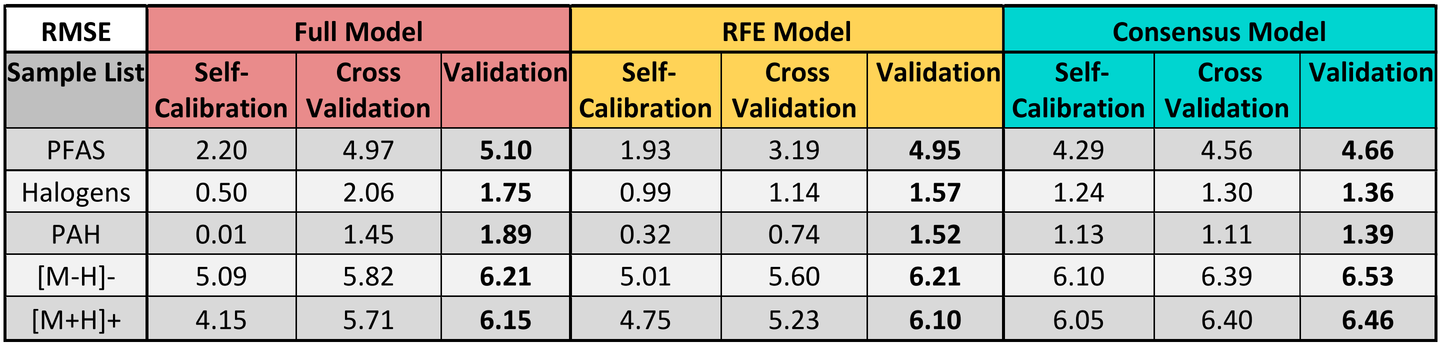
The PFAS dataset appears very similar to the Combined Compendium even with substantially fewer datapoints. This run features the lowest MAE and RMSE values displayed in this work at 1.04% and 3.02 Å^2 respectively.



R^2	Full Model			RFE Model			Consensus Model		
Sample List	Self-Calibration	Cross Validation	Validation	Self-Calibration	Cross Validation	Validation	Self-Calibration	Cross Validation	Validation
PFAS	0.99	0.99	**0.98**	1.00	0.99	**0.97**	0.98	0.98	**0.98**
Halogens	1.00	0.98	**0.98**	1.00	0.99	**0.99**	0.99	0.99	**0.99**
PAH	1.00	0.98	**0.98**	1.00	1.00	**0.98**	0.99	0.99	**0.99**
[M-H]-	0.99	0.99	**0.98**	0.99	0.98	**0.98**	0.99	0.98	**0.98**
[M+H]+	1.00	0.99	**0.99**	1.00	1.00	**0.99**	0.99	0.99	**0.99**



MAE	Full Model			RFE Model			Consensus Model		
Sample List	Self-Calibration	Cross Validation	Validation	Self-Calibration	Cross Validation	Validation	Self-Calibration	Cross Validation	Validation
PFAS	0.67	2.19	**1.40**	0.58	1.21	**1.93**	1.60	1.77	**2.06**
Halogens	0.29	0.88	**0.60**	0.49	0.34	**0.62**	0.42	0.44	**0.56**
PAH	0.01	0.82	**0.86**	0.01	0.37	**0.87**	0.19	0.75	**0.65**
[M-H]-	0.79	1.44	**1.42**	0.68	1.21	**1.38**	1.67	1.73	**1.69**
[M+H]+	0.79	1.63	**1.74**	0.61	1.32	**1.78**	1.73	2.07	**2.08**



RMSE	Full Model			RFE Model			Consensus Model		
Sample List	Self-Calibration	Cross Validation	Validation	Self-Calibration	Cross Validation	Validation	Self-Calibration	Cross Validation	Validation
PFAS	2.20	4.97	**5.10**	1.93	3.19	**4.95**	4.29	4.56	**4.66**
Halogens	0.50	2.06	**1.75**	0.99	1.14	**1.57**	1.24	1.30	**1.36**
PAH	0.01	1.45	**1.89**	0.32	0.74	**1.52**	1.13	1.11	**1.39**
[M-H]-	5.09	5.82	**6.21**	5.01	5.60	**6.21**	6.10	6.39	**6.53**
[M+H]+	4.15	5.71	**6.15**	4.75	5.23	**6.10**	6.05	6.40	**6.46**

Table 3: CCSP 2.0 Optimization using RFE and Consensus Features

The above table set displays the median of 1001 runs for each of data set. The Full Model tables indicate the raw result of CCSP 2.0 using 2-parameter linear SVR without the use of Recursive Feature Elimination (RFE). The RFE Model columns indicate measurements taken on reruns of the dataset using Recursive Feature Selection to narrow down the number of molecular descriptors from >1200 to <400. The Consensus Model takes the most popular of these remaining descriptors (all that appeared ≥950 times in the 1001 runs and analyzes the dataset yet again.

The results for which of the three models (Full, RFE, and Consensus) depended on the size of the dataset. Initial measurements of the Consensus Model yielded an increase in MAE and RMSE for the M+H and M-H models, but the opposite was the case for the smaller, specialized molecule sets like PFAS, Halogens, and PAH. In almost all cases, MAE and RMSE are equivalent or improved from the Soper-Hopper results.[15]

Code Section	**Soper-Hopper[15]**	**CCSP 2.0: M+H**
Descriptor Calculation	<600 s	60 s
Regression	UNK	139 s
Total	<600 s	199 s

Table 4: CCSP 2.0 Timing versus Soper-Hopper Workflow

*This figure tabulated the time data from each of the CCSP 2.0 workflows as compared to the literature Soper-Hopper workflow. The comparison workflow is denoted with * as not all time data was available. This is inconsequential as CCSP 2.0 completed all tasks before the previous workflow was finished with the Descriptor Calculation step. All data shown is for the combined Compendium dataset.*

The timing of this script is also an important component to this work. Soper-Hopper *et al* set a relative benchmark, reporting that descriptor calculation took just under 600s (regression time was not reported). Of the 5 data sets, the largest was the M+H set, meaning that it would take the longest to analyze. The median observed time for an M+H round of analysis was 60 s for descriptor calculation and then 139 s to complete the regression section of the code, totaling 199 s. Soper-Hopper *et al* did not publish their entire list of timings but did mention that descriptor calculation took <600s, nearly 3x as long as the entire analysis time for CCSP 2.0.

Discussion

Overall, these results are encouraging for mobile CCS prediction on mid-spec laptops. At its longest, 709 molecules were converted to final mol objects and analyzed with 1.14 x 10^6 total entries (709 molecules with 1613 descriptors each) in just over 3 min. The Dragon 7.0-based approach conducted by Soper-Hopper *et al*. completed the calculation of 3608 descriptors for approximately 500 molecules (1.8 x 10^6 total entries) in less than 10 min, indicating a significant comparative increase in computation efficiency for CCSP 2.0.[15] The original set-ups of this work involved directly mirroring the partial least-squares approach used by Soper-Hopper which constructed, validated, and graphed the data in 20 s for a combined M+H and M-H dataset. While the errors were comparable to Soper-Hopper, the PLS approach was quickly outmoded by the SVR method. As spectrometers do not run in both positive and negative ion modes, such a combined dataset is not practical. Thus, neither the PLS or combined dataset are described in detail for this work.

The SVR model is built on a Grid Search optimization of two parameters: C and ε. For a particular data point on a fit, there is a certain threshold where the model views the point as "correct" even if it does not completely align with the expected 1:1 line, a noted tolerance value which was decided from a list of 0.01, 0.05, 0.1, 0.5, and 1. If a point dips outside of this range, then an error penalty is applied to the model, the C-parameter, which used values between 2^{-6} and 2^3 progressing via $2^{n\pm1}$. While the C-parameter set the penalty associated for values outside of a particular range, epsilon set the threshold for error before the penalty is applied (0.01, 0.05, 0.1, 0.5, 1. Initial load tests using the combined dataset yielded significant improvement in both error quantifications, decreasing to 8.67 Å^2 RMSE while MAE also decreased to 1.84%. However further improvement was observed when using the more analytically useful individual

Compendium sets as MAE decreased to 1.74% and 1.42% for M+H and M-H, respectively. Additionally, the RMSE decreased to 6.15 Å^2 and 6.21 Å^2 in the same respect. These decreases were not without the traditional tradeoff however. By expanding the tolerance of the model, the increased accepted error in the residuals can decrease the MAE but ultimately lead to an increase in the RMSE, whereas increasing the penalty limits the fit of the model to producing lower RMSE values but generally increased MAE. As such, this is more of an optimization process rather than a minimization.

Thus far the discussion has focused on the Compendium datasets, and this stems from their increased chance for error due to the diversity of these data sets. As mentioned earlier, the Compendium was used as a benchmark for a broad prediction. Once we had a proof of concept with the Compendium benchmark, we then implemented the PFAS, Halogens, and PAH into our regular testing which was after SVR was fully implemented. This was an important decision as the introduction of RFE and the Consensus Model revealed noteworthy results between the large, diverse sets and the small, specialized sets.

RFE reconstructs the model using an ever-shrinking number of descriptors based on their weights (importance) within the model which further decreased errors by limiting overfitting. The Consensus Model took in these remaining descriptors (<400 in many cases) and trimmed the number to those that appeared in 950 of the 1001 runs. The data is then run through the model again. The expectation was this would further limit overfitting and extraneous descriptor inclusion, but a divergence between the large and smaller data sets was observed. For the smaller sets, the expected decrease in MAE and RMSE were observed, but the error for the Compendium sets increased by 0.3-0.4% and 0.3-0.4 Å^2. This stems from the nature of each set. The Compendium sets specialize in having a large, diverse set of molecules that to make a broader model. This translates to a “jack of all trades, master of none” situation where there is no consistent dependence on just a few descriptors. Instead, a variety of descriptors are used to better predict all of these which means their relative importance can vary depending on the distribution of molecule classifications between the Calibration-produced model and the blind test of the Validation set. By eliminating some descriptors to establish the Consensus Model, there may be molecules left under-represented in their predictions, leading to the slightly increased inaccuracy observed. Despite this, these errors remain on or under par with the Soper-Hopper benchmark in the RFE and Consensus Models, lending credence to CCSP 2.0’s ability to accurately predict with either. This result simply implies that specialized datasets are more likely to benefit from the extra Consensus Model as opposed to diverse sets.

Future Directions

Current efforts are focused on introducing new feature selections methods to investigate if such selection has a significant decrease in error to redeem a further increase in analysis time. This will be accomplished by randomizing the McLean datasets 1001 times with each feature selection technique. Early trials take 15+ hours to complete on modest hardware and is not meant to be representative of the laboratory applications. Purely, this portion of the experiment is meant to test the viability of these feature selection routes in CCSP 2.0. Higher power computing is currently required to complete this task in a timely fashion to make a final recommendation for the inclusion of feature selection in the current script and the most analytically useful method.

The script in its current form requires a more streamlined approach between its different forms. Pending the results of the feature selection efforts, allowing the user to run datasets with and without that functionality (either simultaneously or separately depending on computing power) could prove a useful addition. Additionally, a streamlined approach to inserting unknown data for calculation has been noted. Currently, unknown data must be loaded separately into the validation set, requiring a slightly more user-intensive version of the script. Run times are still comparative, but this also negates the automatic and random separation of molecules into calibration and validation data sets. As such, incorporating a separate block of script to take in that unknown set and run the model through it separately would be the ideal approach for this issue.

CCSP 2.0 currently requires previously gathered data in order to run. A more advanced goal would be a direct connection to an IM-MS apparatus such that raw data could be fed into the script. Depending on compatibility with open-source Python software, this could be a direct connection or via IM-MS software that can export raw data via spreadsheet. Predicted chromatographic retention times were considered in the early stages of this work but ultimately did not progress as expected. We believe that CCSP 2.0 builds a general outline for such predictions, but the actual execution requires heavy modification of the inner workings of the script as well as complex considerations such as temperature, mobile phase composition, etc.

Conclusion

This work has shown the capabilities of an in-house CCS prediction software built entirely on open-source Python 3 Modules. While the current workflow calculates less than half of the molecular descriptors of paid software like Dragon 7.0, each iteration of this workflow from PLS to SVR to SVR with tolerance showed marked increases that were comparable to the established Soper-Hopper et al workflow.[15] MAE measurements at or below 2% were observed, particularly in the SVR-based methods. While more testing is needed, it appears that adding tolerances to the SVR model may decrease MAE in larger data sets at the cost of increasing RMSE and vice versa. This pattern is consistent with many analytical processes, resulting in an efforts geared towards error optimization such that both values decrease with successive iterations.

Furthermore, this workflow has been shown to complete this entire process from structure conversion to descriptor calculation to data curation and finally regression in less than half of the time the established workflow takes to calculate its set of molecular descriptors before the genetic algorithm is even employed, all conducted on a medium specification laptop. Feature selection methods are currently being explored which risk longer analysis times but further optimized errors.

The overarching picture is that CCSP 2.0 is an efficient and cost-effective alternative to licensed CCS pre-diction software, offering unparalleled access to the source code and thus end user customization with potential to further develop. This translates to CCSP 2.0 being fully capable of aiding metabolite identification and database creation to assist future novel compound screening and beyond.

References

1. Monge, M. E., et al. (2019). "Challenges in Identifying the Dark Molecules of Life." Annual Review of Analytical Chemistry **12**(1): 177–199.

2. Bouwmeester, R., et al. (2019). "Comprehensive and Empirical Evaluation of Machine Learning Algorithms for Small Molecule LC Retention Time Prediction." Analytical Chemistry **91**(5): 3694-3703.

3. Dodds, J. N., et al. (2020). "Rapid Characterization of Per- and Polyfluoroalkyl Substances (PFAS) by Ion Mobility Spectrometry-Mass Spectrometry (IMS-MS)." Analytical Chemistry **92**(6): 4427-4435.

4. Kanu, A. B., et al. (2008). "Ion mobility–mass spectrometry." Journal of Mass Spectrometry **43**(1): 1-22.

5. Paglia, G. and G. Astarita (2017). "Metabolomics and lipidomics using traveling-wave ion mobility mass spectrometry." Nature Protocols **12**(4): 797-813.

6. Stow, S. M., et al. (2017). "An Interlaboratory Evaluation of Drift Tube Ion Mobility–Mass Spectrometry Collision Cross Section Measurements." Analytical Chemistry **89**(17): 9048-9055.

7. Paglia, G., et al. (2014). "Ion Mobility Derived Collision Cross Sections to Support

Metabolomics Applications." Analytical Chemistry **86**(8): 3985-3993.

8. Mesleh, M. F., et al. (1997). "Structural Information from Ion Mobility Measurements: Effects of the Long-Range Potential." The Journal of Physical Chemistry A **101**(5): 968-968.

9. Zanotto, L., et al. (2018). "High performance collision cross section calculation—HPCCS." Journal of Computational Chemistry **39**(21): 1675-1681.

10. Zhou, Z., et al. (2018). "Advancing the large-scale CCS database for metabolomics and lipidomics at the machine-learning era." Current Opinion in Chemical Biology **42**: 34-41.

11. Guijas, C., et al. (2018). "METLIN: A Technology Platform for Identifying Knowns and Unknowns." Analytical Chemistry **90**(5): 3156-3164.

12. Zhou, Z., et al. (2020). "Ion mobility collision cross-section atlas for known and unknown metabolite annotation in untargeted metabolomics." Nature Communications **11**(1): 4334.

13. Plante, P.-L., et al. (2019). "Predicting Ion Mobility Collision Cross-Sections Using a Deep Neural Network: DeepCCS." Analytical Chemistry **91**(8): 5191-5199.

14. Rahman, M. F., et al. (2014). "Behaviour and fate of perfluoroalkyl and polyfluoroalkyl

substances (PFASs) in drinking water treatment: A review." Water Research **50**: 318-340.

15. Soper-Hopper, M. T., et al. (2020). "Metabolite collision cross section prediction without energy-minimized structures." Analyst **145**(16): 5414-5418.

16. Picache, J. A., et al. (2019). "Collision cross section compendium to annotate and predict multi-omic compound identities." Chemical Science **10**(4): 983-993.

17. Nye, L. C., et al. (2019). "A comparison of collision cross section values obtained via travelling wave ion mobility-mass spectrometry and ultra high performance liquid chromatography-ion mobility-mass spectrometry: Application to the characterisation of metabolites in rat urine." Journal of Chromatography A **1602**: 386-396.

18. Shvartsburg, A. A. and M. F. Jarrold (1996). "An exact hard-spheres scattering model for the mobilities of polyatomic ions." Chemical Physics Letters **261**(1): 86-91.

19. Zhou, Z., et al. (2017). "MetCCS predictor: a web server for predicting collision cross-section values of metabolites in ion mobility-mass spectrometry based metabolomics." Bioinformatics **33**(14): 2235-2237.