Manual for CCS Prediction Using Polarizability and Mass

This Python script was written by Dr. Pattipong Wisanpitayakorn of Mahidol University, Thailand, to assist users in CCS prediction using our approach, as described in our manuscript [I will provide the link after manuscript acceptance].

This script, freely available at https://github.com/MSBSiriraj/SVR_CCSPrediction, trains a model and predict CCS values based on the polarizabilities and m/z of a small set of experimentally measured CCS values of a reference chemical standard. First, the script will read the "Training.xlsx" file and train the support vector regression model based on the adducts' polarizabilities and m/z values. Then, the script will read the "Predicting.xlsx" file and predict the CCS values based on the polarizabilities and m/z values given in the file. Finally, the script will create an Excel file called "Results.xlsx", which will include the content from "Predicting.xlsx" along with an additional column for the predicted CCS values. Please note that this script currently only supports singly-charged adduct molecules.

Here is the step-by-step for CCS prediction

1. Experimentally measure the CCS values and m/z of the adducts of the reference standards.
2. In MarvinSketch software (downloadable at https://docs.chemaxon.com/display/lts-europium/marvinsketch-downloads.md), use the Polarizability Plugin to calculate the polarizability of the adducts as follows:
    a. Build the molecule by hand or import the molecules by name or images (File >> Import Name).
    b. Modify the molecule into the adduct of interest.
    c. Navigate to the Polarizability Plugin by going to "Calculations" >> "Charge" >> "Polarizability".
    d. In the Polarizability Plugin, select the "Take 3D geometry (Thole)" box and then click "OK".
3. Make an Excel file and name it "Training.xlsx", containing 5 columns: "Name", "Adduct", "m/z", "Polarizability", and "Exp CCS". This file should contain the calculated and experimentally measured values of the reference chemical standards.
4. Calculate the polarizabilities of the adducts of interest.
5. Make an Excel file and name it "Predicting.xlsx", containing 4 columns: "Name", "Adduct", "m/z", and "Polarizability". This file should contain the polarizabilities and m/z of the adducts for which we want to predict CCS values.
6. **(If testing the code with the provided Excel files, start here) Run the Python code, "SVRModel_Polarizabilityandmz.py," in the same folder as the "Training.xlsx" and "Predicting.xlsx" Excel files**. **The code will create an Excel file called "Results.xlsx", containing the predicted CCS values.** Make sure to install all related Python packages: pandas, numpy, and sklearn.