**Materials and methods**

**MCnebula algorithm**

**Overview**. The analysis of untargeted LC-MS/MS datasets typically starts with feature detection. Annotating these features is time-consuming and extracting useful information for further biological research can also be challenging. To address these issues, we present MCnebula, with an abundance-based classes (ABC) selection algorithm, to boost mass spectrometry data analysis by focusing on critical chemical classes and visualization in multiple dimensions.

**Molecular formula and chemical structure prediction**. The analysis of MS/MS spectra involve a process of inference and prediction. We deduce the molecular formula based on the molecular weight of elements composition and possible fragmentation pattern of the MS/MS spectrum by SIRIUS; We then search for the exact chemical structure from compound structure databases. However, this process is often uncertain as several factors can affect the reliability of MS/MS data and the correctness of inference. SIRIUS generates a list of candidates for the potential molecular formula, chemical structure, and chemical classification behind each MS/MS spectral feature. In such cases, MCnebula extract these candidates and determine the unique molecular formula and chemical structure for each MS/MS spectrum based on the highest score of prediction.

**Top candidate selection by multiple score systems**. After the process of compound prediction, we get candidates for molecular formula, structure, and chemical classes. Some candidates are correct while others are not. While the correct predictions for molecular formula and chemical structure are unique, the chemical classification may have multiple correct predictions belonging to different classes. The scoring method depends on personalized research purpose. We use scores based on various criteria such as isotopes, mass error, structural similarity, and chemical classes to rank and filter the candidates. With numerous score systems available, we filter out low-scoring candidates and focus on those with higher scores, which are more likely to be the correct compound. However, in most cases, the top candidates from the three scoring systems are not always consistent. So, we choose a ‘specific candidate’ as a reference in the user-selected scoring system and retrieve its chemical information from the other systems for data integration. We obtain unique molecular formulas and chemical structure for the reference by scoring and ranking, but for chemical classes, more work is needed.

**Chemical classification**. Compounds with MS/MS spectrum can be classified based on their overall structure and local structure, which we refer to as the dominant structure and substructure, respectively. Therefore, in the chemical classification system, we can classify compounds not only based on their dominant structure, but also based on their substructure. When the dominant structure is unavailable, or the MS/MS spectral fragment is insufficient, we classify the compounds using substructure information to gain knowledge of the compound.

**ABC selection**. The ABC selection algorithm evaluates all features collectively in an untargeted LC-MS/MS dataset, by examining the number and abundance of features for each chemical classification at different levels with both substructure and dominant structure. Based on this analysis, representative classes are selected for subsequent analysis (as shown in Figure 6).

* Create Stardust Classes (Inner filter). By using the posterior probability of classification prediction (PPCP) data for each feature, simple threshold or absolute conditions are set to filter the chemical classes, and refer to as ‘inner’ filtering.
* Cross filter Stardust Classes. This involves combining the data of the chemical classes and their classified features (i.e. Stardust Classes data), grouping them based on the chemical classes, and then performing statistics on the features within each group. Statistics may also be performed on these data in conjunction with features annotation data, or to compare groups with each other. This method involves crossover of attributes for filtering, hence it is referred to as ‘cross’ filtering. (See details in the next subsection about Cross filter Stardust Classes.)

The resulting dataset is called Nebula-Index, which records multiple chemical classes and their associated features. Each chemical class is considered as a ‘Nebula’ and its classified ‘features’ are the components of the ‘Nebula’. These Nebulae are visualized as networks, with Parent-Nebula representing all features combined and Child-Nebulae representing individual classes.

**Details of Cross filter Stardust Classes.** This method involves integrating three modules, as shown in Figure 6:

*Cross-filtering by ‘quantity’* (abundance selection): The ‘features’ quantity limitation is set for each group (a chemical class with its classified ‘features’). The minimum quantity of ‘features’ within the group and the maximum proportion of ‘features’ quantity within the group versus all ‘features’ (unique) quantity of all groups are used as thresholds. Chemical classes with too many or too few ‘features’ are filtered out.

*Cross-filtering by ‘score’* (Goodness assessment): This step associates the Stardust Classes data with ‘features’ annotation data. For each group, the Goodness assessment is performed for each target attribute (continuous attribute, generally a scoring attribute of compound identification, such as ‘Tanimoto similarity’). If the group satisfies all the expected Goodness, the chemical class is retained. The Goodness () is calculated as follows: , where is the quantity of ‘features’ of which the target attributes satisfy the cut-off, and is the quantity of all ‘features’.

The assessment of Goodness is related to the parameters of ‘tolerance’ and ‘cutoff’: the expected Goodness value of ‘tolerance’ and the actual Goodness, which is related to the parameter ‘cutoff’.

Goodness assessment can be given to multiple target attributes. Note that the chemical class is retained only if it passes the Goodness assessment of all target attributes. The main purpose of this step is to filter out chemical classes with too many ‘features’ of low structural identification.

*Cross-filtering by ‘identical’* (identicality assessment): This step involves a similarity assessment of chemical classes. A hierarchical range is set for chemical classification, and groups within this range are compared for similarity. If the classified ‘features’ of two groups are almost identical to each other, the chemical class represented by one of the groups is excluded. The degree of identicality between two groups (A and B) is assessed, as well as the value of the parameter ‘identical\_factor’ () :

: ratio of the classified ‘features’ of A belonging to B

: ratio of the classified ‘features’ of B belonging to A

: value of parameter ‘identical\_factor’

If and , the two groups are considered identical, and the group with fewer ‘features’ is discarded. The purpose of this step is to filter out classes that may incorporate each other and are similar in scope. The *in silico* prediction approach may not be able to distinguish which class the potential compound belongs to from the LC-MS/MS spectra.

**Network graph presentation**. The features and their annotations are integrated as Nebulae based on the Nebula-Index. These Nebulae are represented as network-type graph data. The feature annotation data includes top candidates for chemical formula and structure. The MS/MS spectral similarity of the features is calculated and used to generate the edge data for the network graph.

**Visualization system**. MCnebula integrates various R packages to format data, including the ‘ggplot2’ package. To make visualization easier for users, we developed the ‘ggset’ data class, which stores pre-defined ggplot2 plotting functions and parameters for visualizing Nebulae. Users can customize the visualization according to their specific needs or the requirements of the publisher.

**Statistical analysis**. MCnebula integrates the functions of the ‘limma’ package for differential expression analysis of RNA-sequence and microarray data[40], and package them for differential analysis of metabolomics data. The gene expression matrix and feature quantification matrix of LC-MS are similar, both have phenotypic variables (sample information) and dependent variables (gene expression or feature quantification values). Our method is only appropriate for experimental designs in which explanatory variables are factorial variables and the design matrix is without an intercept [41].

**Data structure**. MCnebula was primarily developed using the R S4 system of object-oriented programming. All data including ‘features’ annotation data and visualization data is stored in a single object (class ‘mcnebula’), which simplifies the application, makes data management and analysis easier to perform and repeat.

**Reporting system**. MCnebula includes a reporting system that enables the analysis process to be output as a PDF document or in other formats. The reporting system is based on the ‘report’ data class, which stores each step of the analysis as a section and can be easily modified according to the user’s requirements. Furthermore, the ‘rmarkdown’ R package [43] is incorporated in the reporting system to generate reports.

**Code Compatibility**. MCnebula performs downstream analysis by extracting data from the pre-computed SIRIUS project, which is the primary data source for MCnebula 2. The SIRIUS software is continually updated and enhanced. From SIRIUS version 4 to version 5 (https://github.com/boecker-lab/sirius), the data structure and attribute names in the project directory have been modified. To ensure that MCnebula is not affected by version problems, we have designed its application programming interface (API) for various SIRIUS versions.

**MCnebula evaluation**

**Spectra dataset for evaluation**. To evaluate the performance of MCnebula, the spectra from the GNPS MS/MS library was used (http://prime.psc.riken.jp/compms/msdial/main.html#MSP). To prevent overfitting during library match evaluation, ‘noise’ was added to the MS/MS spectra[44]. Two models of noise were simulated: medium noise and high noise. The simulation involved a global mass shift, individual mass deviations, intensity variations, and additional ‘noise peaks.’ Isotope patterns were also simulated using the ‘get.isotopes.pattern’ function within the ‘rcdk’ R package[45]. The mass and abundance of isotopes were considered for the adduct type to increase or decrease exact mass. The ‘isotope peaks’ were merged into the MS1 list of compounds, and all spectra collections were formatted into mgf or csv file for the continuing MCnebula and benchmark analysis.

**Evaluation method.** MCnebula and benchmark workflow was conducted for all the three simulated datasets. SIRIUS 4 command-line interface (CLI) (version 4.9.12) was applied for computation, and filtered out MS/MS spectra with empty fragmentation peaks. In total 7524 out of 8782 compounds were left for evaluation. ClassyFire was used to assess the classification accuracy[24]. After *in-silico* annotation, we obtained structure annotation, International Chemical Identifier Key (InChIKey), and other metadata of these compounds. Considering that ClassyFire only supports chemical identities those structures have been classified in its server previously, we used the first hash block of InChIKeys (InChIKey planar, which represents the molecular skeleton) to query the PubChem application programming interface (API) (https://pubchemdocs.ncbi.nlm.nih.gov/pug-rest) [46]. This provided us with all the possible InChIKeys of isomerism (stereo, isotopic substitution) [47]. Classification of small molecules depends on their molecular skeleton, so chemical identities that share the same InChIKey planar are identical in classification. The InChIKey list was imported into ClassyFire to obtain chemical classification. In our R script, once any InChIKey of isomerism matched the classified data in the database, we turned off the acquisition status for this molecular skeleton. In the end, we collated, integrated, and assigned all these chemical annotations as standard reference.

Due to differences in algorithms and classified results, we evaluated the MCnebula and benchmark methods separately. Since sub-structural classification was not available for the benchmark method, we excluded these classes during the evaluation analysis. Nevertheless, some compounds within the remaining classes may still be categorized as sub-structural. We assigned three levels for evaluation: ‘True,’ ‘Latent,’ and ‘False.’ ‘True’ indicated that the classified classes were consistent with those of ClassyFire. ‘Latent’ indicated that the classified classes were not consistent with ClassyFire, but their parent classes at the ‘class’ level were consistent. ‘False’ indicated that the classified classes were completely inconsistent with ClassyFire.

To evaluate the identification of classes or structures, we merged the results with a standard reference by InChIKey planar. For the evaluation of chemical structure identification, we considered a structure as ‘true’ if it matched the chemical structure identified by InChIKey planar. However, this evaluation neglected stereochemistry due to the limit of LC-MS/MS detection.

**Serum dataset**

We re-analyzed 245 LC-MS/MS runs from MASSIVE (id no. MSV000083593) (blanks, controls, and samples) [42]. MZmine2 (version 2.53) was applied for feature detection []. MS/MS spectra were exported in mgf format for SIRIUS 4 (version 4.9.12) computation[22], and merged across samples into one fragmentation list with a 30% peak count threshold filtering. The feature detection workflow was based on FBMN preprocessing and SIRIUS computational prerequisites. ZODIAC [39], CSI:fingerID [20], CANOPUS [34] were involved in SIRIUS. Specifically, we customized SIRIUS to detect the Iodine element while predicting the formula. We used MCnebula for subsequent data analysis and exported them as reports (see the section on Data and code availability).

Kyoto Encyclopedia of Genes and Genomes (KEGG) metabolic pathway enrichment analysis was conducted with the identified InChIKey planar structures to search for compounds in the metabolic pathway. To account for the possibility of identified structural deviations due to stereoisomerism, we used the InChIKey planar to obtain all possible InChIKeys and PubChem CID via PubChem API. Pubchem CID was converted to KEGG ID by the R package MetaboAnalystR. KEGG enrichment with the ‘pagerank’ algorithm was performed via the R package FELLA. We integrated these methods into functions within MCnebula workflow, which are freely available as the ‘exMCnebula2’ package.

**Herb dataset**

**Materials and processing.** The dried bark of E. ulmoides was obtained from the ZheJiang ZuoLi Chinese Medical Pieces LTD. Two types of preparations, Raw-Eucommia and Pro-Eucommia, were made as follows: (1) Raw-Eucommia: The shreds or blocks of E. ulmoides dried bark were taken, powdered, and passed through 80-mesh sieves for further processing. (2) Pro-Eucommia: The shreds or blocks of E. ulmoides dried bark were fried with saline water (using an amount of salt equal to 2% of the weight of E. ulmoides, with 10 times the volume of water to dissolve), and smothered in airtight conditions for 30 minutes. Then, the barks were dried in an oven at 60 °C, followed by baking at 140 °C for 60 minutes. Finally, the baked barks were powdered and passed through 80-mesh sieves for further processing.

**Sample preparation.** Weighing 2 g of Raw-Eucommia powder and Pro-Eucommia powder, adding 50 ml of methanol/water (1:1, v/v), and subjecting it to ultrasonication (20 kHz for 40 min). The mixture was then filtered to separate the filtrate and residue. The residue was further extracted with 50 ml of methanol/water (1:1, v/v) under ultrasonication (40 kHz, 250 W for 20 min) and filtered again. The filtrate of the two extracts was combined and the solvent was evaporated. Methanol/water (1:1, v/v) was added to redissolve the extract and the volume was adjusted to 5 ml. Finally, the supernatant was obtained by centrifugation (12,000 r.p.m. for 10 min) for subsequent LC-MS analysis.

**LC-MS experiments**. LC-MS analysis was performed using a Dionex Ultimate 3000 UHPLC system coupled with a high-resolution Fourier-transform mass spectrometer. The mobile phase consisted of solvent A (formic acid/water) and solvent B (formic acid/acetonitrile). The Waters Acquity HSS T3 column was used for separation. The gradient profile for separation was as follows: 0-2 min, 2% - 5% B; 2-10 min, 5%-15%, B; 10-15 min, 15%-25% B; 15-18 min, 25%-50% B; 18-23 min, 50-100% of solvent B at 23 min. The flow rate was 0.3 mL/min and the column temperature was set at 40°C. Mass spectrometric analysis was performed using an Orbitrap Elite instrument equipped with an ESI source that operated in the negative ionization mode. The survey scans were conducted in the Orbitrap mass analyzer operating at a 120,000 resolving power. A mass range of 100-1500 m/z and a normalized collision energy of 30 eV were used for survey scans. The analysis method was set to analyze the top 10 most intense ions from the survey scan, and a dynamic exclusion was enabled for 15 s.

**MCnebula Workflow**. MZmine2 is a software for mass spectrometry data processing and was used in this study for feature detection in the E.ulmoides dataset. The resulting features were then processed using the SIRIUS software. The subsequent analysis of the E.ulmoides dataset was similar to the analysis of the serum metabolic dataset and a report was generated for this analysis. The details of the analysis, as well as the code and data, are available in the Data and Code Availability section of the study.

**Data processing**

   Raw data (.raw) were converted into mzML format in centroid mode via MSConvert in ProteoWizard[[5](file:///C:\l)]. The .mzML files were processed with MZmine2 (v.2.53) and followed by SIRIUS 4 in Pop!-OS (Ubuntu) 22.04 LTS 64-bits workstation (Intel Core i9-10900X, 3.70GHz 20, 125.5 Gb of RAM) [[8](file:///C:\l)]. MCnebula (MCnebula2) and other R packages were executed in Pop!\_OS (Ubuntu) 22.04 LTS 64-bits PC (Intel Core i7-1065G7, 1.3 GHz 8, 16 Gb of RAM).

Discussion materials

Classifying compounds based on their dominant structure is straightforward. For example, we classify Taxifolin as a flavone, not a phenol, even though its local structure contains a phenol substructure. We prefer to classify compounds based on their dominant structure because it is more concise and provides more information. However, during the MS/MS spectral analysis, we sometimes can only classify compounds based on their substructure.

It is essential to understand the hierarchy of chemical classification to identify and organize compounds effectively. Flavones belong to a superior classification of flavonoids, which in turn is part of the higher-level classification of phenylpropanoids and polyketides. These classifications are part of the broader classification of organic compounds.

For example, the class of ‘Organic compounds,’ which covers almost all compounds that can be detected in metabolomics data, is too broad to be of any help to biological research. The parameter settings are not absolute, and the thresholds can be adjusted according to the needs of the study.