Reproducible mass spectrometry data processing and compound annotation in MZmine 3

Nature protocols

Heuckeroth, Steffen; Damiani, Tito; Smirnov, Aleksandr; Mokshyna, Olena; Brungs, Corinna et al

https://doi.org/10.1038/s41596-024-00996-y https://doi.org/10.1038/s41596-024-00996-y

Protocol

Check for updates

Reproducible mass spectrometry data processing and compound annotation in MZmine 3

Steffen Heuckeroth[1,15], Tito Damiani[2,15], Aleksandr Smirnov[3], Olena Mokshyna[2], Corinna Brungs[2], Ansgar Korf[2], Joshua David Smith[2,4], Paolo Stincone[5], Nicola Dreolin[6], Louis-Félix Nothias[7,8], Tuulia Hyötyläinen[9], Matej Orešič[9,10], Uwe Karst[1], Pieter C. Dorrestein[11], Daniel Petras[5,12], Xiuxia Du[3], Justin J. J. van der Hooft[13,14], Robin Schmid[1,2,11] & Tomáš Pluskal[2]

Abstract

Untargeted mass spectrometry (MS) experiments produce complex, multidimensional data that are practically impossible to investigate manually. For this reason, computational pipelines are needed to extract relevant information from raw spectral data and convert it into a more comprehensible format. Depending on the sample type and/or goal of the study, a variety of MS platforms can be used for such analysis. MZmine is an open-source software for the processing of raw spectral data generated by different MS platforms. Examples include liquid chromatography–MS, gas chromatography–MS and MS–imaging. These data might typically be associated with various applications including metabolomics and lipidomics. Moreover, the third version of the software, described herein, supports the processing of ion mobility spectrometry (IMS) data. The present protocol provides three distinct procedures to perform feature detection and annotation of untargeted MS data produced by different instrumental setups: liquid chromatography–(IMS–)MS, gas chromatography–MS and (IMS–)MS imaging. For training purposes, example datasets are provided together with configuration batch files (i.e., list of processing steps and parameters) to allow new users to easily replicate the described workflows. Depending on the number of data files and available computing resources, we anticipate this to take between 2 and 24 h for new MZmine users and nonexperts. Within each procedure, we provide a detailed description for all processing parameters together with instructions/recommendations for their optimization. The main generated outputs are represented by aligned feature tables and fragmentation spectra lists that can be used by other third-party tools for further downstream analysis.

Key points

- MZmine is a program designed to process data from untargeted mass spectrometry (MS) experiments acquired in data-dependent acquisition mode; specifically, collision-induced dissociation and higher-energy collisional dissociation.

- This protocol provides three distinct procedures to perform feature detection and annotation of untargeted MS data produced by instrumental setups: liquid chromatography–(ion mobility spectrometry–)MS, gas chromatography–MS and (ion mobility spectrometry–)MS imaging.

Key references

Schmid, R. et al. *Nat. Biotechnol.* **41**, 447–449 (2023): https://doi.org/10.1038/s41587-023-01690-2

Schmid, R. et al. *Nat. Commun.* **12**, 3832 (2021): https://doi.org/10.1038/s41467-021-23953-9

Heuckeroth, S. et al. *Nat. Commun.* **14**, 7495 (2023): https://doi.org/10.1038/s41467-023-43298-9

A full list of affiliations appears at the end of the paper. e-mail: rschmid1789@gmail.com; tomas.pluskal@uochb.cas.cz

Introduction

Driven by rapid technological advances, the field of mass spectrometry (MS) has undergone substantial progress since the early 2000s[1]. The unprecedented sensitivity and resolving power reached by modern MS instruments enable the comprehensive characterization of both biological and nonbiological samples. Liquid chromatography (LC)–MS is currently the most popular technique for the untargeted profiling of complex mixtures, as hundreds to thousands of molecules can be detected in a single analysis. Moreover, fragmentation (MS^2, also called tandem MS or MS/MS) data can be simultaneously collected for the identification of these compounds[2]. Gas chromatography (GC)–MS offers another robust platform for global metabolite profiling of microbial[3], plant[4] and human[5] samples. Although especially suited for volatile and nonpolar compounds, GC–MS can also be used for the analysis of more polar compounds through sample-derivatization procedures[6]. The predominant ionization technique in GC–MS is electron ionization (EI), because of its universal applicability and the high reproducibility of the fragmentation spectra it produces[7].

Complementary to chromatography-based techniques is MS imaging, which enables the spatial mapping of molecules in tissue samples and is now an established tool in clinical practice[8]. Compound annotation in MS imaging often relies on the precursor m/z measurement, allowing mostly low-confidence annotations. Nevertheless, MS imaging and LC–MS data can be combined (aligned) to raise this annotation confidence[9].

Finally, ion mobility spectrometry (IMS) is being increasingly adopted in disciplines such as MS-based metabolomics and lipidomics, as it can provide an additional dimension for isobaric and isomeric metabolite separation, as well as an additional property to refine annotation (i.e., collision cross-section)[10,11].

The growing versatility and throughput of MS platforms also pose challenges in terms of volume and complexity of the produced multidimensional datasets. In fact, although manual data investigation still plays a crucial role, computational pipelines are essential to streamline the processing of untargeted MS data. General-purpose data processing tools are typically provided by MS instrument vendors. However, research applications often go beyond the scope of vendor software and demand flexible processing solutions that rely on newly published approaches[12]. MZmine is an open-source framework for processing MS data from different instrument vendors and setups. Over the years, thanks to community efforts and collaboration with other open-source projects, MZmine has become one of the most popular tools for visualizing and analyzing untargeted MS data (Box 1). The third version of the software, MZmine 3, has been released recently[9] and includes several new functionalities such as a re-designed graphical user interface (GUI), improved feature detection workflows, and support for MS imaging and IMS data. In this protocol, we provide stepwise instructions for processing untargeted MS data from several different platforms, using MZmine 3 (see 'Overview of the method').

Feature detection and annotation

The goal of MS data processing is to turn raw spectral data into a list of detected ions, to estimate their abundance and to assign chemical annotations based on multiple criteria[13]. In MZmine, this is done in a three-step approach:

- First, raw spectral data are centroided and intensity thresholds can be applied to exclude low-intensity signals (e.g., electronic noise) from further processing
- The second step is known as feature detection (also 'feature finding' or 'peak picking') and represents the cornerstone of the processing. A feature can be seen as an m/z signal (more often a group of signals) related to a single metabolite detected during MS analysis (Table 1). On the basis of the instrument setup, a feature can be characterized by additional identifiers such as retention time (RT) in chromatography–MS experiments or spatial coordinates in MS imaging data. Untargeted MS experiments typically yield hundreds to thousands of features, although a relatively small portion corresponds to meaningful metabolites detected in the sample[14]. For this reason, the goal of feature detection is to retain all relevant features in the raw spectral data while discarding 'noisy' signals. Moreover, features

detected in different samples can be aligned to enable consistent sample-to-sample comparison (e.g., statistical analysis)

- The third and last step of MS data processing in MZmine is feature annotation. Here, various chemical annotations can be assigned to each feature based on additional information retrieved from raw spectral data (e.g., isotope pattern, MS^2 spectra), using dedicated modules (e.g., lipid annotation[15]) or via leveraging the direct integration of MZmine with other popular annotation tools (e.g., SIRIUS[16], Global Natural Products Social Molecular Networking (GNPS)[17])

Overview of the method

Different data processing workflows are needed for different instrumental setups (e.g., chromatography–MS versus MS imaging) and/or acquisition settings[18]. The present protocol describes how to use MZmine 3 to perform feature detection and annotation on MS data produced by three different platforms: LC–MS, GC–EI–MS and MS imaging. Moreover, LC–MS and MS imaging data acquired with IMS devices (i.e., LC–IMS–MS and IMS–MS imaging) can also be processed.

Since many data processing parameters are dataset specific and require insights into raw spectral data (e.g., chromatographic setup, MS analyzer performance) and/or domain-specific knowledge, generally applicable values cannot be given. Similarly, showcasing this protocol on example data might generate confusion among nonexperts, as the provided parameter values cannot be readily used for processing different datasets. For this reason, we structured the present protocol as follows. First, we provide various example datasets and corresponding batch files (i.e., list of processing steps and related parameters, Table 1) to help new users getting familiar with the software (see the "Setting up MZmine to reproduce the procedures with the 'Batch mode'" section in the 'Equipment setup'). Then, we describe three distinct procedures to perform feature detection and annotation on different data types (Fig. 1): Procedure 1 for the analysis of LC–MS and LC–IMS–MS data, Procedure 2 for the analysis of GC–EI–MS data and Procedure 3 for the analysis of MS imaging and IMS–MS imaging data.

In each procedure, we give instructions for selecting the correct processing steps (i.e., MZmine modules) based on the data type and, for each module, we provide a description of the parameters with recommendations for their optimization. We encourage new MZmine users to first process the provided example datasets using the corresponding batch files as described in the "Setting up MZmine to reproduce the procedures with the 'Batch mode'" section. Thereafter, the same batch files can be used as a starting point and adjusted for processing new datasets based on the instructions given in each procedure. If needed, more detailed explanations and tutorial videos can be found in the MZmine documentation (www.mzmine.org/mzmine_documentation/).

BOX 1

Contributing to MZmine 3

Since its inception in 2004[58], the MZmine project has evolved into a collaborative, community-driven effort, and nowadays constitutes one of the most popular tools for processing untargeted MS data. Thanks to the modular architecture of MZmine, new modules can be programmed and tested independently, without the need to modify other pieces of the software. Over the years, this has greatly facilitated contributions to the project by new developers and researcher teams from all over the world[9].

To facilitate the use of the software by new users, the MZmine community creates extensive documentation materials and tutorial videos. The online documentation (https://mzmine.org/documentation) provides detailed description of each individual module and is constantly updated with the latest software releases and features. Moreover, it contains a step-by-step guide to program new processing modules and add them to the MZmine codebase.

Anyone can contribute to the MZmine community by:

1. Writing documentation: https://mzmine.org/documentation
2. Writing code and developing modules: https://github.com/mzmine/mzmine3
3. Answering questions or discussing developments on the GitHub issues page: https://github.com/mzmine/mzmine3/issues

Table 1 | Terminology for MZmine 3 data processing

Term	Explanation
Feature	In the field of MS, the term feature is used to refer to a 'meaningful' signal produced by a chemical entity detected during MS analysis. Features are characterized by a mass-to-charge ratio (m/z), intensity, and, based on the type of MS experiment, additional identifiers: In chromatography–MS experiments (e.g., LC–MS or GC–MS), features are associated with the RT of the chromatographic peak In MS imaging data, features are associated with spatial coordinates When IMS is used, features are also associated with their ion mobility value
Feature list	List of features detected in a single raw data file
Aligned feature table	List of features obtained by merging (i.e., aligning) feature lists of multiple samples
Mass list	During spectral processing (Fig. 1), each mass spectrum in the data files is processed individually and stored as a list of m/z and intensity pairs (called the mass list) that is readily usable by MZmine for the subsequent feature detection steps
EIC	Signal intensity of a specific m/z displayed at any one RT of the chromatographic run
EIM	Signal intensity of a specific m/z displayed along the mobility dimension
Chromatogram and mobilogram resolving	Splitting EIC or EIM traces that contain multiple peaks into individual features. This includes the correct splitting of partially co-eluting peaks (e.g., shoulder peaks)
RT-resolved features	Features resulting from a chromatogram resolving process (Procedure 1, Step 6), narrowing the associated RT window. See also 'Chromatogram and mobilogram resolving'
IMS-resolved features	Features resulting from a mobilogram resolving process (Procedure 1, Step 9), narrowing the associated mobility window. See also 'Chromatogram and mobilogram resolving'
Mobility scan	Mobility scans are the individual MS spectra collected during each IMS separation cycle. Each mobility scan corresponds to a data point in a mobilogram. Frame spectra are obtained by merging these mobility scans
Frame spectrum	The frame spectrum is also referred to as 'summed frame spectrum'. During one IMS separation, one ion package is accumulated and multiple scans (see mobility scan) are recorded and associated with a mobility value. The sum of these scans is called a frame spectrum. Each frame spectrum corresponds to a single data point in a chromatogram
Isotopic pattern	Distribution of m/z signals in an MS^1 spectrum that arises from isotopologues of the same molecule. Isotopologues are molecules that differ only in the isotopic composition of their atoms
Isotopologue feature	Features generated by isotopologues of the same chemical entity
Precursor ion	Ion selected and subjected to an MS/MS experiment to produce smaller fragment ions
Component	In GC–EI–MS, components refers to a group of chromatographic signals with overlapping RTs and peak shape, which includes precursor and product ions generated in the EI source
DDA	MS data acquisition mode where a certain number of precursor ions from an MS^1 scan is selected for subsequent fragmentation experiments, one by one. For example, in topN-DDA, the most *N*-abundant signals in an MS^1 survey scan are selected for fragmentation
DIA	MS data acquisition mode where all ions within a selected m/z range are selected from an MS^1 scan and subsequently fragmented. For example, in all ion fragmentation experiments (e.g., MS^E), the full m/z range undergoes subsequent fragmentation
Spectral deconvolution	A procedure to computationally reconstruct MS spectra for co-eluting analytes in GC–EI–MS data. This is needed due to the extensive in-source fragmentation caused by EI
Scan-to-scan tolerance	Maximum allowed deviation for a specific m/z over (usually) consecutive scans of the same instrument in a single acquisition. For example, the scan-to-scan m/z tolerance will depend on the mass accuracy, resolution, and precision of your instrument To optimize this parameter, we recommend manually inspecting the raw data to determine the typical mass accuracy fluctuation over consecutive scans
Feature-to-feature tolerance	Tolerance between multiple features of the same acquisition (sample). For example, a value of feature-to-feature m/z tolerance will refer to the actual versus expected difference of multiple ion adducts of the same compound (e.g., $[M+H]^+$ and $[M+Na]^+$). An RT tolerance would refer to the RT difference of the different adducts. The accuracies within a sample are usually greater than across samples
Sample-to-sample tolerance	Tolerance for the same compound between multiple instrumental acquisitions, for example, replicates or other samples. Usually described by m/z, RT and mobility tolerances. These tolerances are usually greater than the feature-to-feature tolerances within one sample
MS level	Stage of fragmentation for a given scan. 'MS level = 1' means no fragmentation, 'MS level = 2' means tandem fragmentation (i.e., MS^2), 'MS level = 3' means MS^3 fragmentation, etc
m/z peak	A peak within an MS spectrum corresponding to a detected ion
Profile data	When MS data are acquired in profile mode, an m/z peak is represented by a collection of signals over several data points. Profile raw data preserves the original information in the data, which may be lost during centroiding. However, data files in profile mode can be substantially larger in memory size. Profile data can be converted to centroid data (i.e., centroiding)
Centroid data	MS data displayed as discrete m/z signals and corresponding intensities (i.e., m/z and intensity pairs). Centroid data files are substantially smaller in memory size than profile data files
Centroiding	Converting raw mass spectra acquired in profile mode into discrete m/z values with associated intensities (centroided spectra)
Module	A module can be seen as a piece of software designed to carry out a specific task, independently of the rest of the system. A module takes some input data, performs a set of processing operations, and produces an output file, which can then be taken as input by another module. By doing so, different modules can be combined into a unique processing pipeline without affecting the entire system (modular architecture). This gives the user the flexibility to design custom workflows or even develop their own module

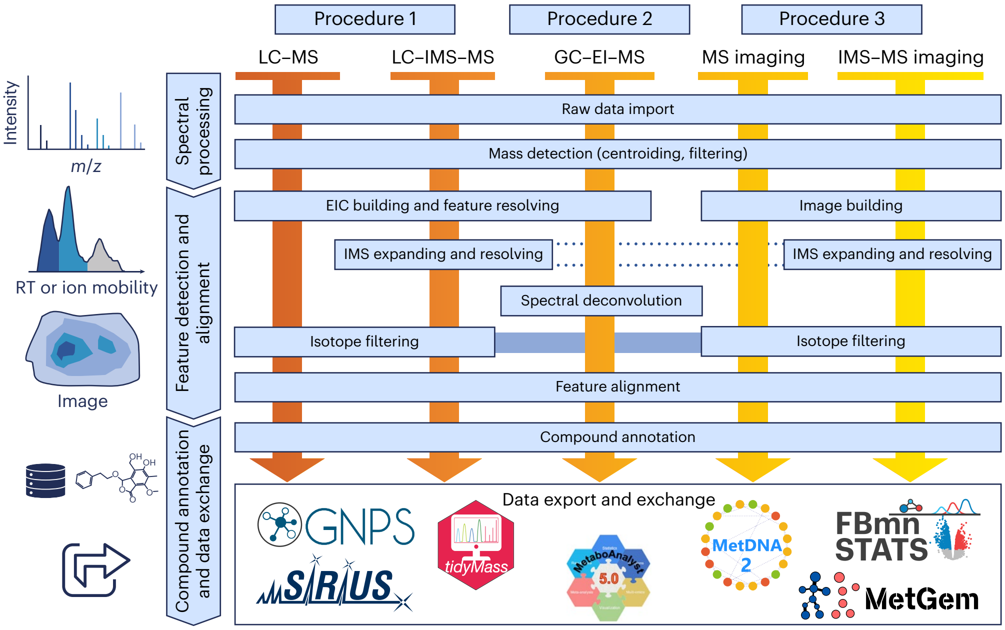


Fig. 1 | An overview of the data processing workflows in MZmine. The main data processing steps of the three procedures described in the present protocol are outlined.

Applications of the method

MZmine can process MS data for various applications including metabolomics, lipidomics, natural product research or environmental studies. Although the presented protocols use example datasets acquired from biological samples, MZmine has been used to process (IMS–)MS data from many other sample types, including food[19], dissolved organic matter[20], archaeological artifacts[21] or tattoo pigments[22,23]. Procedures 1 and 3 describe how to process LC–IMS–MS and IMS–MS imaging, respectively. As highlighted in Fig. 1, the same pipelines can be used to process non-IMS data by skipping the 'IMS expanding and resolving' step. Procedure 2 covers the processing of untargeted GC–EI–MS data, which requires a dedicated spectral deconvolution step to handle the extensive in-source fragmentation produced by EI. GC–MS data produced by 'softer' ionization techniques (e.g., chemical ionization) can be processed using the workflow described in Procedure 1.

Data formats

At the time of writing, MZmine supports the following open data formats: mzML[24], mzXML[25], imzML[26], netCDF[27] and aird[28]. Moreover, MZmine supports proprietary formats from Thermo Scientific (.raw) and Bruker Daltonics (.d and .tdf/.tsf). Raw data files from other vendors can also be processed but must first be converted into an open format using vendor-provided or third-party software.

The MSConvert[29] tool from the ProteoWizard package[30] supports the conversion of AB SCIEX, Agilent, Bruker, Shimadzu, Thermo Scientific and Waters raw data. A step-by-step guide for data conversion with MSConvert is provided at https://mzmine.org/mzmine_documentation/data_conversion.html#data-conversion-to-open-formats-mzml-imzml. Both profile and centroid data can be imported in MZmine. Centroiding of profile data can be performed during MZmine processing (Procedure 1, Step 2). However, we recommend using already centroided data because of the smaller file size and memory consumption. More information and the latest supported data formats are provided at https://mzmine.org/mzmine_documentation/data_conversion.

Comparison with other methods

Over the years, several open software tools for MS data processing have been developed and widely adopted by the scientific community. These include, among others, XCMS[31], OpenMS[32] and MS-DIAL[33]. All of these software packages are equipped with a user-friendly GUI that greatly assists researchers lacking programming skills[12]. In this regard, MZmine places great emphasis on the development and continuous improvement of highly interactive GUIs that enable the user to make informed choices on key processing parameters (for example, Boxes 2 and 3).

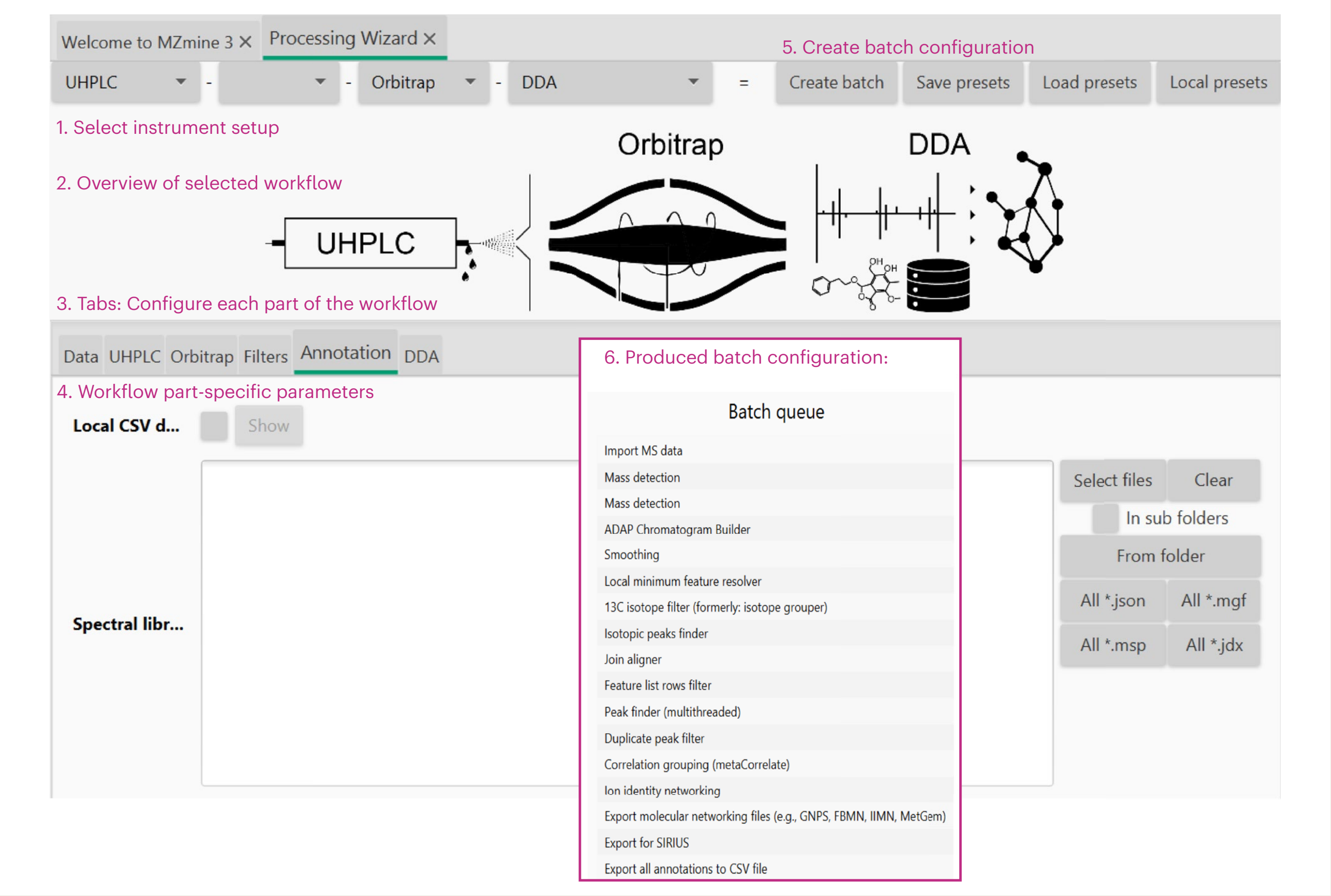
BOX 2

The processing wizard

The processing wizard is a tool for the quick and user-friendly generation of data processing workflows for different MS platforms (e.g., LC–MS, GC–MS). The goal is to make the generation of processing workflows more beginner friendly by reducing the number of parameters to set. To open the processing wizard, navigate to 'Processing wizard' in the MZmine menu. The wizard is organized in the following sections:

- Sample introduction (e.g., HPLC, MALDI)
- IMS, MS analyzer
- Data acquisition (e.g., DDA, DIA)

After selecting the desired instrumental setup, each section can be configured in the tabs shown in the bottom panel. Directories for input data import and output files export can also be specified. Although default parameters are provided, adjustments might be needed based on the specific user's application or instrument performance. After setting all the required parameters, click the 'Create batch' button to open and review the so-created batch file in the dialogue window (Extended Data Fig. 1). More information about the processing wizard can be found at https://mzmine.org/mzmine_documentation/wizard.html.



1. Select instrument setup

2. Overview of selected workflow

3. Tabs: Configure each part of the workflow

4. Workflow part-specific parameters

5. Create batch configuration

6. Produced batch configuration:

Batch queue

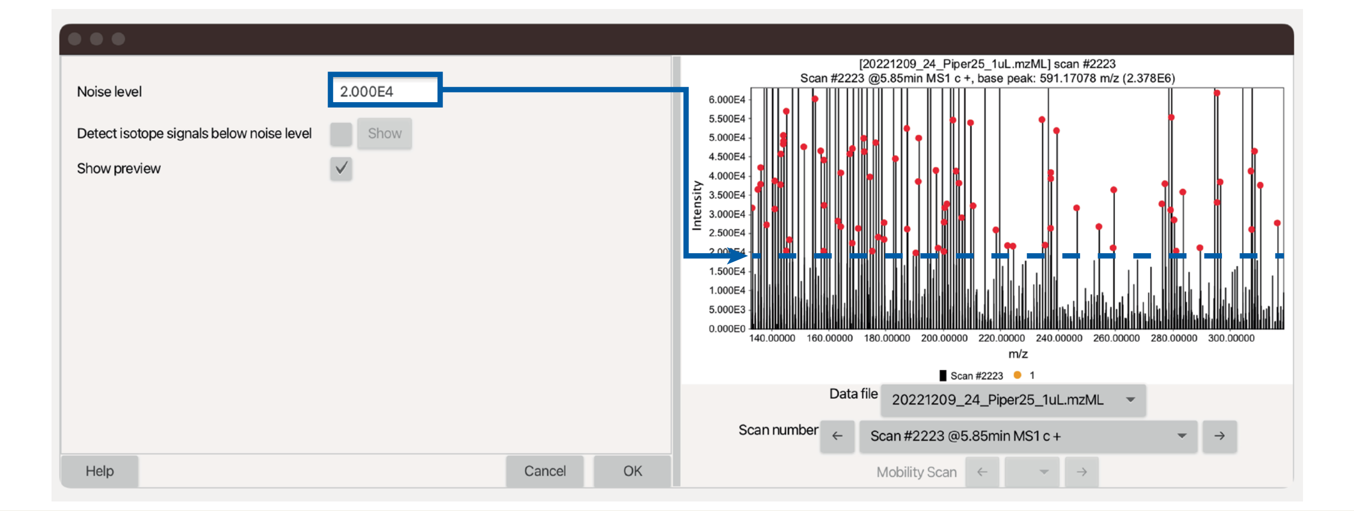
Import MS data
Mass detection
Mass detection
ADAP Chromatogram Builder
Smoothing
Local minimum feature resolver
13C isotope filter (formerly: isotope grouper)
Isotopic peaks finder
Join aligner
Feature list rows filter
Peak finder (multithreaded)
Duplicate peak filter
Correlation grouping (metaCorrelate)
Ion identity networking
Export molecular networking files (e.g., GNPS, FBMN, IIMN, MetGem)
Export for SIRIUS
Export all annotations to CSV file

BOX 3

Setting the noise level

MS background noise generally refers to nonspecific *m/z* signals detected in the absence of a specific analyte. Such noise can arise for a number of reasons (e.g., electronic noise) and is characterized by several low-intensity signals with no clear pattern among them (often referred to as 'grass' in the mass spectra). Filtering out noise from raw spectral data before feature detection prevents a large number of irrelevant background signals from being retained as false features, which may complicate downstream processing steps. This also reduces memory consumption and computing time, especially when processing large datasets. As the magnitude of background noise can vary greatly between different datasets, the best way to optimize this parameter is by visually inspecting the raw spectra. To this end, an interactive visualization panel can be opened directly from the 'Mass detection' dialogue window (Step 2). Two drop-down menus can be used to select, respectively, the data file and spectrum to display (data needs to be imported first, Step 1). Once a noise level is set, *m/z* signals above the threshold are automatically labeled with red dots. All unlabeled signals will be excluded from further processing. Ideally, the noise level should be set right above the 'grass' noise (the blue line in the figure below). Nevertheless, higher noise levels can be used to save computation time/cost when processing large datasets consisting, for example, of hundreds to thousands of data files.

In chromatography-based experiments, the noise level in the spectra can vary across the same chromatographic run. For instance, greater noise is often observed toward the end of GC–MS runs due to increased column bleeding. In these situations, different noise levels can be applied to different RT ranges using the 'Scans' filters. Note that the mass detection can be run on one RT segment at a time, therefore multiple 'Mass detection' calls are needed, one for each RT segment (Procedure 2, Step 2).



Noise level: 2.000E4

Detect isotope signals below noise level □ Show

Show preview ☑

Data file: 20221209_24_Piper25_1uL.mzML

Scan number: Scan #2223 @5.85min MS1 c +

Furthermore, MZmine can save results from each individual processing step, which can be manually (re)inspected by the user. This simplifies workflow optimization and backtracking of potential errors during the setting of parameters.

At the time of writing, MZmine is among relatively few software packages that support the full processing of IMS data (both LC–IMS–MS and IMS–MS–imaging)[9]. Moreover, one unique function of MZmine is the possibility to combine MS data from various instrumental setups, for example, LC–MS and MS-imaging. To do so, users are normally required to master different data processing software, for instance, specific to chromatography–MS or MS–imaging, and to use a third, external tool to integrate the results. The alignment and annotation of LC–IMS–MS and IMS–MS–imaging data are showcased in Procedure 3, Step 10.

Limitations of this protocol and software

Untargeted feature detection workflows in MZmine are primarily designed for data acquired in data-dependent acquisition (DDA) mode, in particular collision-induced dissociation and higher-energy collisional dissociation. Rule-based annotation approaches (e.g., see lipid annotation modules, Procedure 1, Step 21) may be limited when used on data produced with different ionization techniques, such as electron impact excitation of ions from organics (EIEIO). Nonetheless, annotation of EIEIO spectra can still be performed via spectral library search against reference libraries produced using EIEIO.

At the time of writing, MZmine 3 is not optimized for processing MS data acquired in data-independent acquisition (DIA) mode. We recommend users to explore other more established tools for DIA data processing, such as MS-DIAL[33] and Skyline[34].

The present protocol does not illustrate all MZmine 3 features that may be useful for untargeted MS data processing (e.g., blank sample subtraction). Information and tutorials on how to use additional software functions can be found at https://mzmine.org/mzmine_documentation/index.html.

Processing large datasets

In MZmine 3, special attention was directed toward scalability owing to the ever-increasing study sizes and availability of public data. Nevertheless, when processing very large datasets (e.g., hundreds to thousands of data files), we recommend applying the following measures to minimize the memory consumption and reduce the chance of software crashes during processing. All points are covered in more detail in the online documentation (https://mzmine.org/mzmine_documentation/index.html).

- Set the temporary file directory to a fast local drive (e.g., solid-state drive) with enough free space (see the 'Materials' section)
- Use the recommended file formats (e.g., native .tdf and .tsf formats for Bruker data, .mzML format for all other vendors, .imzML format for imaging data)
- Optimize the batch file on a subset of representative samples (e.g., pooled quality control samples, randomly selected samples) before proceeding to the full processing
- Run MZmine through the command line interface. This will avoid memory usage by the GUI
- Use the advanced data import (Procedure 1, Step 1) to perform the mass detection during the data import. By doing so, all signals below the specified noise level are not imported, thus saving memory and processing time
- In the batch file, set the 'Original feature list' parameter to either 'IN PLACE' or 'REMOVE' (Procedure 1, Step 5) in all steps
- Adjust processing parameters. In general, increasing the noise level and other feature constraints (e.g., minimum feature height) will reduce the amount of data to be processed. Missed features will be recovered during gap filling
- Use the 'Join aligner' module for the feature alignment (Procedure 1, Step 13) and the 'Peak finder' module for the gap filling (Procedure 1, Step 14), as they are optimized for large data volumes
- If the experimental design includes replicate samples and features are expected to be detected in a minimum number of replicates, we recommend applying the 'Min aligned features (samples)' filter (Procedure 1, Step 22) before the gap-filling step. This will remove the 'nonreproducible' features and reduce the volume of data being processed

If software crashes still occur after implementing these measures, we recommend upgrading the random-access memory (RAM) of your processing PC, or consider using a computer cluster.

Transparent and FAIR data processing

Modern scientific research is required to adhere to the FAIR principles (findable, accessible, interoperable and reusable) to ensure transparency, reproducibility and reusability of the produced results[35]. Every scientific publication should provide clear instructions on how and where to access the experimental data and any digital object used, for instance, software tools, algorithms, and workflows[36]. MZmine is often part of larger computational workflows and even minor differences in the produced output files can impair the reproducibility of downstream data analysis steps. Therefore, to guarantee full reproducibility of the processing output in MZmine, the following elements are necessary:

- Raw data files. If raw files were converted to open formats (e.g., .mzML), ideally, also the original files in the vendor format should be shared. The vendor formats might contain additional metadata that is lost during conversion. Also, profile mode raw data is usually converted to centroid .mzML files, which reduces the volume of data
- MZmine batch file. A batch file contains the complete configuration of a list of processing steps, specifically modules and their related parameters. Virtually any processing pipeline can be saved as a batch file and executed in the MZmine GUI or in the command-line mode. Loading batch files into the GUI allows to visualize the structure and review settings. Batch files offer a convenient way to share a precise description of the data processing and help others reproduce it
- MZmine version used to perform the processing. MZmine uses semantic versioning—major.minor.patch version. Stable releases are permanently stored and available in the MZmine GitHub repository, different from the development release that is only meant for testing. For this reason, it is strongly recommended to use stable MZmine releases to process data for scientific publications. Starting with MZmine 3.4.0, information about the MZmine version is also included in the batch file. Furthermore, batch files also contain versions for each step that are updated if any user parameter changes
- Output files produced by MZmine and exported for downstream data analysis, for example, feature lists and MS spectra files stored in .csv and .mgf, respectively
- Metadata file that contains the list of input raw data files and the corresponding sample information based on the experimental design of the study

A way to ensure open and long-term access to the above-listed files is to upload them into public MS data repositories such as GNPS/MassIVE[17], MetaboLights[37] or MetabolomicsWorkbench[38]. By doing so, the uploaded files are assigned an accession number that can be easily referenced in scientific publications, databases and other resources.

Materials

Software

- MZmine 3 (latest release)
- (Optional) MSConvert from the ProteoWizard package (latest release)[29,30]
- (Optional) FTP client (e.g., WinSCP)

Equipment

Hardware

▲**CRITICAL** The procedure requires a personal computer or other computing resources with an internet connection and either a Windows, Linux or macOS platform. The exact requirements depend on the type of experiment and/or the size of the dataset.

- Small datasets (<100 files): 4+ central processing unit (CPU) cores, 8+ GB RAM
- Medium datasets (100–5,000 files): 8+ CPU cores, 24+ GB RAM
- Large datasets (>5,000 files): 8+ CPU cores, 64+ GB RAM
- MS imaging datasets: 8+ CPU cores, 64+ GB RAM

Datasets

▲**CRITICAL** All datasets used in this protocol are publicly available in open data repositories and listed under the accession numbers provided below. The MassIVE datasets can be downloaded using an FTP client by following this step-by-step guide. If you are processing your own dataset(s), make sure data are converted to recommended file formats (see the 'Processing large datasets' section). Each of these datasets has a corresponding 'batch file' to reproduce the analysis.

- LC–IMS–MS data from plant extracts (Procedure 1): MSV000091634. This dataset was acquired from the LC–IMS–MS analysis of hydroalcoholic extracts of Piperaceae plants (nine data files). MS data were acquired with a quadrupole time-of-flight (QTOF) mass

spectrometer equipped with a trapped-IMS (TIMS) device. Fragmentation MS2 spectra were collected in parallel accumulation-serial fragmentation (PASEF) mode[39]
- GC–EI–MS from clinical trial (Procedure 2): ST000981. This dataset was acquired from a study on healthy research cats receiving clindamycin administered with a synbiotic or a placebo. More information about the dataset can be found in the original publication[40]
- MS imaging data from sheep brain samples (Procedure 3): MSV000090328. This dataset was acquired from the LC–IMS–MS and IMS–MS imaging analysis of sheep brain samples. Hydrophilic interaction chromatography (HILIC) chromatography was used. MS data were acquired with a QTOF equipped with a TIMS device. A matrix-assisted laser desorption ionization (MALDI) source was used for the IMS–MS imaging analysis
- (Optional) LC–IMS–MS data for lipid annotation (Procedure 1, Step 21): MSV000091642. This dataset was acquired from the LC–IMS–MS analysis of sheep brain samples[41]. HILIC chromatography was used. MS data were acquired with a QTOF equipped with a TIMS device. Fragmentation MS2 spectra were collected in PASEF mode
- (Optional) LC–MS data for statistical analysis (Procedure 1, Step 25): MTBLS265. LC–MS analysis of blood samples from 30 patients. The dataset includes three replicates per sample required by MetaboAnalyst for multivariate statistics. MS data were acquired with an Orbitrap MS instrument. More information about the dataset can be found in the original publication[42]

Batch files

▲ **CRITICAL** We provide configuration batch files for each example dataset to easily replicate all three procedures described in this protocol. The provided batch files are optimized for the corresponding example dataset. The same batch files should not be used to process different data without adaptation, as this would probably produce unreliable results; rather, they represent a good reference and starting point for parameter optimization.

- Batch file for Procedure 1: batch_procedure-1.xml. Batch file for processing LC–IMS–MS data in the native Bruker format. A spectral library search step for feature annotation is included. Export steps for feature-based molecular networking (FBMN) and the SIRIUS software are included
- Batch file for Procedure 2: batch_procedure-2.xml. Batch file for processing GC–EI–MS data (centroided). A spectral library search step for feature annotation is included. An export step for FBMN is included
- Batch file for Procedure 3: batch_procedure-3.xml. Batch file for processing IMS–MS data in the native Bruker format
- (Optional) Batch file for Procedure 1, Step 21: batch_lipid_annotation.xml. Batch file for processing LC–IMS–MS lipidomics data in the native Bruker format. Feature annotation is done using the lipid annotation module
- (Optional) Batch file for Procedure 1, Step 25: batch_metaboanalyst.xml. Batch file for processing LC–MS data (centroided). An export step for statistical analysis in MetaboAnalyst is included

Spectral libraries

▲ **CRITICAL** The provided batch files include a step of spectral library search for feature annotation (Procedure 1, Step 19). This requires a spectral library file to be imported into MZmine (Procedure 1, Step 18). The following public spectral libraries can be freely downloaded from the MassBank of North America (MoNA, https://mona.fiehnlab.ucdavis.edu/downloads).

- Spectral library for Procedure 1: 'LC–MS/MS Positive Mode' library from MoNA
- Spectral library for Procedure 2: 'GC–MS Spectra' library from MoNA

Equipment setup

MZmine 3 installation

1. Download and install the latest stable release of MZmine from https://github.com/mzmine/mzmine3/releases/latest.

2. Open MZmine and set a temporary file directory to a local drive with enough free space (preferably a solid-state drive). To do so, navigate to 'Project → Set preferences → General → Temporary file directory' and browse the desired directory. Changes in the 'Temporary file directory' require a restart of the software to take effect.
3. (Optional) Set additional memory-usage options as described at https://mzmine.org/mzmine_documentation/performance.

(Optional) Setting up MZmine to reproduce the procedures with the 'Batch mode'

In MZmine, several parameters have to be set, but only a few are crucial to tune processing for specific datasets. This generally requires insights in the spectral raw data and domain-specific knowledge depending on the application. We encourage first-time users to download the example datasets and use the corresponding batch files to run the processing pipeline as described below.

▲ **CRITICAL** The batch files provided in the present protocol are optimized for each corresponding example dataset. The same batch files should not be used to process different data without adaptation, as this would probably produce unreliable results. Nonetheless, they can be used as a reference and as a starting point for parameter optimization.

1. Open MZmine and navigate to 'Project → Batch mode'. This will open the dialogue shown in Extended Data Fig. 1, which can be used to load, inspect, edit and run batch files.
2. Click the 'Load' button and import the batch file corresponding to the dataset and procedure (e.g., batch_procedure-1.xml for Procedure 1).
3. Select 'Replace' and click 'OK' to load the batch file. All batch processing steps are now displayed in the 'Batch queue' panel. Values for the individual parameters are already set. Double-click on any step in the Batch queue to open the corresponding dialogue box to review and/or modify the parameters. Some of these dialogue boxes offer a 'Show preview' option for interactive parameter optimization. For the preview to work, data must be already imported in MZmine.
4. Double-click on the 'Import MS data' step to open the window for selecting the data files to import.
5. Browse the MS data files to process (Procedure 1, Step 1) using the corresponding buttons. Either Individual files (i.e., 'Select files' button) or all files in a directory (i.e., 'From folder' button) can be imported. Alternatively, data files can be drag and dropped in the 'File names' panel.
6. Browse the spectral libraries to use for feature annotation (Procedure 1, Step 18). Alternatively, spectral libraries can be drag-and-dropped in the 'Spectral library files' panel. If spectral matching is not used for feature annotation and no spectral library is imported, make sure to also remove the 'Spectral library search' step from the Batch queue.
7. On the basis of the batch file being used, double-click on the export steps and select a directory in your filesystem for the results export. One or more of the following export steps can be present:

'Export/Submit to GNPS-FBMN'	Set the 'Filename' to a suitable file path (e.g., 'C:\Data\project_gnps' on Windows)
'Export for SIRIUS'	Set the 'Filename' to a suitable file path (e.g., 'C:\Data\project_sirius')
'Export for statistics (MetaboAnalyst)'	Set the 'Filename' to a suitable file path (e.g., 'C:\Data\project_metaboanalyst')

8. Click the 'OK' button in the dialogue window to start the batch processing.

Procedure 1: LC–MS and LC–IMS–MS

▲ **CRITICAL** Refer to Fig. 2 for an overview of Procedure 1. In addition, more information is available on the online documentation (LC–MS workflow, LC–IMS–MS workflow (https://mzmine.org/mzmine_documentation/workflows/lcmsworkflow/lcms-workflow.html)).

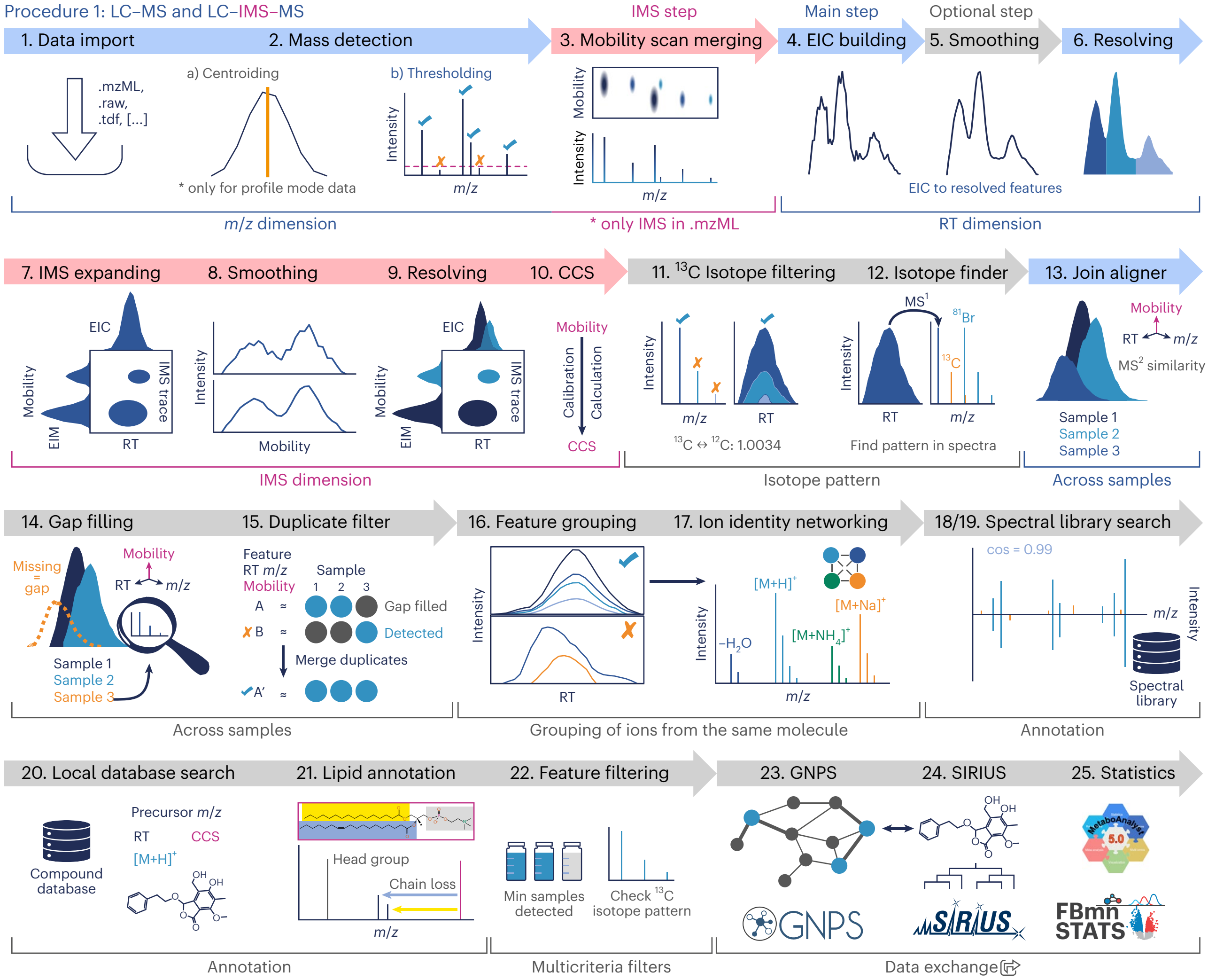


Fig. 2 | A schematic representation of the LC(−IMS)−MS workflow described in Procedure 1. The steps strictly required for processing LC−MS data are colored in blue. Additional steps required for LC−IMS−MS data are coloured in magenta. Optional steps for both workflows (in gray) can be applied to further refine or annotate feature lists. The steps are numbered according to the described procedure.

Data import and mass detection

1. Import MS data. Raw data files can be imported into MZmine 3 by drag and dropping directly in the ‘MS data files’ tab in the main window. Another option is to use the ‘Import MS data’ module described step-by-step below.
   - Navigate to ‘Raw data methods → Raw data import → MS data (advanced)’
   - Click on ‘Select file’ to directly browse the data files in your filesystem. Alternatively, all data files of a specific format can be imported from a directory using the corresponding button (e.g., ‘All *.mzML’, ‘All *.raw’)
   - Disable the ‘Advanced import’ checkbox. This option is only needed for very large datasets (see the ‘Processing large datasets’ section)
   - (Optional) ‘Spectral library files’ can be imported in a way similar to that described in step 1b. Spectral libraries are needed to perform spectral library search (Procedure 1, Step 19). If this is not the case, clear any text in the ‘Spectral library files’ panel
2. Mass detection. Each mass spectrum in the data files is processed individually and stored as a list of *m/z* and intensity pairs (called mass list), which is readily usable by the software for the subsequent feature detection steps. An intensity threshold (i.e., ‘Noise level’) can also

be set to exclude low-intensity signals (e.g., electronic and/or chemical noise) from further processing.

- Navigate to 'Raw data methods → Mass detection → Mass detection'
- Specify the 'Raw data files' to process (e.g., 'All raw data files' or 'As selected in main window')
- Set the 'Scans' filters. Use the 'Set filters' button to run the mass detection on separate MS levels (Table 1). When MS^2 spectra are collected (e.g., LC–MS/MS), the mass detection should be run twice: once for 'MS level = 1' and once for 'MS level = 2' (for example, batch file 'batch_procedure-1.xml')
- Select the 'Mass detector' algorithm from the drop-down menu. Five mass detection algorithms are available. The choice of the algorithm depends on the raw data characteristics (e.g., profile versus centroid data, low versus high MS resolution). A detailed description of each mass detector option can be found at https://mzmine.org/mzmine_documentation/module_docs/featdet_mass_detection/mass-detection-algorithms.html. We recommend the 'Centroid' mass detector for centroid data and the 'Exact mass' algorithm for profile high-resolution MS data
- Click the 'Setup' to set the 'Noise level'. This will exclude the *m*/*z* signals below the specified intensity threshold from further processing. An interactive visualization panel can be opened by ticking the 'Show preview' checkbox (Box 3)
- (Only IMS data) Specify the 'Scan types (IMS)' to be processed. This parameter only applies to IMS data and determines whether mobility scans, frame spectra, or both should be processed. Since frame spectra are obtained by merging multiple mobility scans (Table 1), we recommend applying two different noise levels. This can be done by running the mass detection on 'Mobility scans only' and 'Frames only' separately (e.g., batch file 'batch_procedure-1.xml')

  ▲**CRITICAL STEP** The noise level in the spectra can vary greatly due to a number of reasons, such as the type of the mass analyzer and specific acquisition settings. For this reason, MZmine provides an interactive visualization panel to help the user optimize this step (Box 3).

3. (Only IMS data in .mzML format) Mobility scan merging. In this step, mobility scans are merged to create frame spectra (Table 1). This is required only when processing IMS data in .mzML format. When processing data files in the native Bruker format (.d and .tdf), mobility scans are merged in the background during the data import. If you are using the example dataset provided for this procedure, you can skip this step and go to Step 4.
   - Navigate to 'Raw data methods → File merging → Mobility scan merging'
   - Specify the 'Raw data files' to process
   - Select the 'Show preview' checkbox and choose a frame to preview
   - Set the 'Noise level' to 0 to deactivate thresholding at this step. As with Step 2, a noise level can be set and will be applied to the merged frame spectra
   - Select the 'Merging type'. This determines how the signal intensities are calculated in the merged frame spectra. The 'Summed' option (recommended), sums the intensities for the same *m*/*z* value detected across the mobility scans
   - Click the 'Clear filters' button for the 'Scans' filters, since the noise level was set for MS level 1 and MS level 2 separately
   - Choose the '*m*/*z* weighting' method. This parameter determines how the *m*/*z* values are calculated in the merged frame spectrum. The 'Linear' method (recommended) attributes larger weight to more intense signals
   - Set the scan-to-scan '*m*/*z* tolerance'. This is the maximum allowed deviation for an *m*/*z* value detected across consecutive mobility scans to be considered the same. We recommend 0.005 *m*/*z* or 15 ppm as a starting point for most time-of-flight (TOF) instruments

Chromatogram building and resolving

▲**CRITICAL** The following steps of chromatogram building and resolving require insight into the width, height and number of data points of chromatographic peaks, which depend on the instrument and the LC–MS method. Raw data can be explored using the 'Raw data overview' module (Extended Data Fig. 2). See the online documentation at https://mzmine.org/mzmine_documentation/module_docs/lc-ms_featdet/featdet_adap_chromatogram_builder/adap-chromatogram-builder.html. for more details.

4. Extracted ion chromatogram (EIC) building with 'ADAP chromatogram builder'. This step builds an EIC for each *m*/*z* value detected over a minimum number of consecutive MS^1 scans in the LC–MS run. EICs matching a set of user-defined requirements (e.g., minimum number of data points and intensity) are stored as features in a feature list. Although various EIC building algorithms are available, we recommend using the Automated Data Analysis Pipeline (ADAP) chromatogram builder module. A detailed description of this module is provided at https://mzmine.org/mzmine_documentation/module_docs/lc-ms_featdet/featdet_adap_chromatogram_builder/adap-chromatogram-builder.html.
   - Navigate to 'Feature detection → LC-MS → ADAP chromatogram builder'
   - Specify the 'Raw data files' to process
   - Set the 'Scan filters'. Enable the checkbox, click the 'Show' button, and set the 'MS level filter' as 'MS1, level = 1'
   - Set the 'Minimum consecutive scans' as the minimum number of consecutive MS^1 scans where an *m*/*z* must be detected above a certain intensity (see the next parameter) for the corresponding EIC to be considered valid. This parameter largely depends on the MS acquisition settings used during the analysis. Usually, no less than three to five should be used, as lower values would produce false features
   - Set the 'Minimum intensity for consecutive scans' as the minimum intensity an *m*/*z* must exceed in consecutive MS^1 scans (see the previous parameter) for the corresponding EIC to be considered valid. A good starting point is one to three times the 'Noise level' used for the MS^1 level in the Mass detection (Step 2). If LC–IMS–MS data are being processed, consider the noise level applied to frame scans
   - Set the 'Minimum absolute height' as the minimum intensity the highest data point in the EIC must exceed for the corresponding EIC to be considered valid. A good starting point is three to ten times the 'Noise level' used for the MS^1 level in the Mass detection (Step 2). If LC–IMS–MS data are being processed, consider the noise level applied to frame scans
   - Set the '*m*/*z* tolerance (scan-to-scan)'. This is the maximum allowed *m*/*z* deviation between consecutive scans in the EIC. This parameter largely depends on the MS analyzer type and performance. A good starting point is '0.003 *m*/*z* or 5 ppm' for Orbitrap instruments and '0.005 *m*/*z* or 15 ppm' for TOF devices
   - Provide a 'Suffix' (e.g., '_eic') to name the newly created feature lists. This option is present in most of the modules described below. We recommend using a different suffix for each module to easily recognize the features lists produced by each processing step

     ▲**CRITICAL STEP** The *m*/*z* tolerances must be specified as both an absolute value (in *m*/*z*) and a relative value (in ppm). MZmine automatically calculates the tolerance for each *m*/*z* value using the maximum of the absolute and relative tolerances.

5. (Optional) Chromatogram smoothing. We recommend applying smoothing to EICs only if they exhibit a 'jagged' profile (i.e., large intensity fluctuations of consecutive data points). Jagged EICs may cause inaccurate peak integration and erroneous splitting of peaks into multiple features during the EIC resolving step (Step 6). On the other hand, excessive smoothing can lead to peak shape distortion and artifacts. For this reason, we recommend using the 'Show preview' option to evaluate the effect of the chosen smoothing parameters.
   - Navigate to 'Feature detection → Smoothing'
   - Specify the 'Feature lists' to process. When running modules individually, various options are available (e.g., 'As selected in the main window', 'Feature list name pattern'). When using the batch mode (see "Setting up MZmine to reproduce the procedures with the 'Batch mode'" section), the option 'Those created by previous batch step' must be selected
   - Choose the 'Smoothing algorithm'. We recommend using 'Savitzky Golay'

- Click the 'Setup' button and tick the 'Retention time smoothing' checkbox. Set the number of data points to use for smoothing. We recommend using half the number of data points of a chromatographic peak
- Tick the 'Show preview' checkbox to open an interactive visualization panel to help adjust the smoothing parameters. Use the drop-down menus to select, respectively, the feature list and feature to display. We recommend choosing a medium-intensity EIC trace that well represents the 'jaggedness' in the data. Importantly, when changing the smoothing parameters, the preview does not automatically update. It is necessary to select a new feature from the drop-down menu to visualize the newly set parameters
- Specify how to handle the 'Original feature list'. This option determines whether to 'KEEP' in memory or 'REMOVE' the input feature list(s) once the processing is completed. We recommend using the 'KEEP' option during parameter optimization
- Provide a 'Suffix' (e.g., '_RT-smooth') to name the newly created feature lists

6. EIC resolving with the 'Local minimum resolver'. The EIC traces built in the previous steps are stored in a feature list per sample. EICs might contain multiple chromatographically separated peaks that need to be resolved into individual features. Although various EIC resolving algorithms are available, we recommend using the local minimum resolver module when processing LC data. Refer to Box 4 for a more detailed description of the optimization of the local minimum resolver. A detailed description of all feature resolvers is provided at https://mzmine.org/mzmine_documentation/module_docs/featdet_resolver_local_minimum/local-minimum-resolver.html.
    - Navigate to 'Feature detection → Chromatogram resolving → Local minimum resolver'
    - Specify the 'Feature lists' to process
    - Specify how to handle the 'Original feature list'
    - Enable the 'MS/MS scan pairing' option. This will pair each resolved feature to the corresponding MS^2 fragmentation spectrum (collected in DDA mode), based on the RT offset between the chromatographic peak and the moment the MS^2 was triggered during the run

    Click the 'Show' button and set the following parameters:
    - Set the 'MS1 to MS2 precursor tolerance (*m*/*z*)' as the maximum allowed deviation between the *m*/*z* associated with the feature, and the precursor *m*/*z* the MS^2 was triggered for. As a starting point, the same *m*/*z* tolerance set in the chromatogram building step (Step 4) can be used
    - Set the 'Retention time filter' as 'Use feature edges'. This option pairs a feature with the corresponding MS^2 spectrum only if the latter was triggered within the feature's RT range. The 'Use tolerance' option uses a fixed tolerance between the feature's peak apex and the RT of the MS^2 scan
    - (Optional) Enable and set a 'Minimum relative feature height' to limit the pairing of an MS^2 scan with multiple features. When an MS^2 scan can be paired with multiple features within the specified tolerances, only those with intensity above *x*% of the most intense feature will be considered. When enabled, the default value (25%) should provide good results for most applications
    - Disable the 'Minimum required signals'. This parameter is designed to remove empty MS^2 scans in spectral library building workflows
    - Ignore all the remaining parameters at this stage. These are related to IMS and have no effect on the 'Retention time' dimension. They will be discussed in the 'IMS expanding and resolving' section

    In the main dialogue window continue setting the following parameters:
    - Select 'Dimension → Retention time'
    - Enable the 'Show preview' option to open an interactive visualization panel to help adjust the resolving parameters (Box 4). Once opened, select the feature list and EIC traces to display from the corresponding drop-down menus

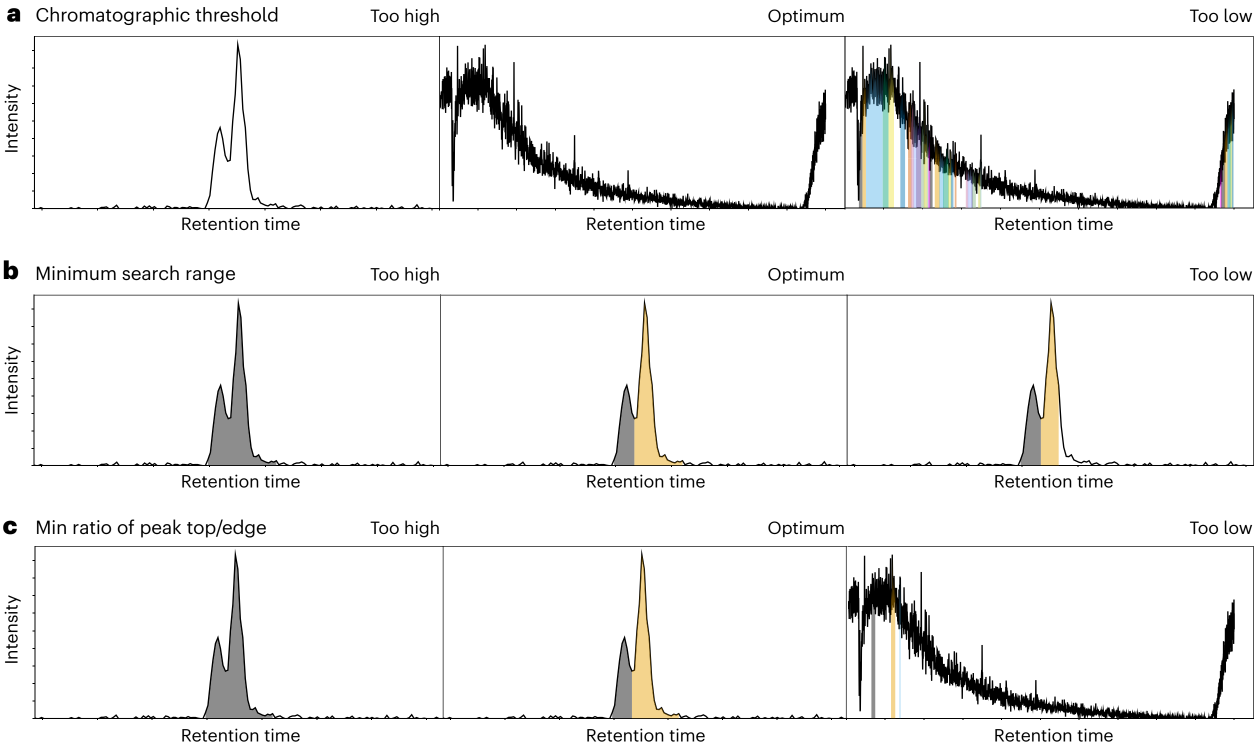
BOX 4

Optimize feature resolving

The EIC resolving step plays a crucial role in the feature detection of chromatography–MS data. The goal is to split multiple peaks that are resolved or partially co-eluting in the EIC traces into individual features (i.e., chromatographic resolving). The local minimum resolver module (recommended for LC–MS) assumes that a local minimum in an EIC trace corresponds to the valley between two adjacent peaks and uses it to split fully resolved or 'shoulder' peaks. Thorough optimization of the algorithm parameters is crucial to ensure reproducible detection of true features across all samples while minimizing 'noisy' peaks to be retained as features. Particular attention should be given to the 'Chromatographic threshold', 'Minimum search range RT' and 'Min ratio of peak top/edge' parameters. Their effect on the EIC resolving results is illustrated in the figure below, and a more detailed description can be found at https://mzmine.org/mzmine_documentation/module_docs/featdet_resolver_local_minimum/local-minimum-resolver.html.

MZmine provides a preview panel (Step 6) to interactively display the effect of the selected parameters on the detected features (Extended Data Fig. 3). We recommend using this option for understanding and fine-tuning the algorithm, especially when working with a new dataset.

When the feature list has been selected, two EIC traces are displayed in the two subpanels. MZmine automatically tries to select a 'noisy' and 'good' EIC trace in the upper and lower panel, respectively, based on the height-to-area ratio. The detected features are color filled. Resolved peaks are shown in different colors. During the optimization, all clear chromatographic peaks in the lower panel should be fully retained, while as few features as possible should be detected in the upper panel.



- Set the 'Chromatographic threshold'. This parameter represents an important filter for chromatographic noise (e.g., solvent background contaminants). Briefly, the *x*% least-intense data points from the whole chromatogram are removed before the resolving. For LC–(IMS–)MS data, we recommend using a value between 50% and 90%
- Set the 'Minimum search range RT/Mobility'. This is the RT window used for local minimum search. A good starting point is the full-width at half maximum (FWHM,

expressed in minutes) of a typical chromatographic peak in the data. The optimal RT search range mainly depends on the chromatographic system setup and MS acquisition settings. An overly narrow RT search range can cause peak edges to be truncated. Conversely, an overly wide search range might lead to the incomplete resolution of closely eluting peaks

- Set the 'Minimum relative height' to 0 to ignore this parameter. We recommend not to use this parameter, because modern MS analyzers offer dynamic ranges spanning several orders of magnitude
- Set the 'Minimum absolute height' a peak needs to reach to be retained as a feature after the resolving procedure. We recommend using the same value as used in Step 4
- Set the 'Min ratio of peak top/edge'. This is the minimum ratio between the intensities of the highest point (apex) and edges of a peak to be retained as a feature after the resolving step (i.e., the peak apex must be *x* times more intense than the peak edges). We recommend setting 2 as a starting value
- Set the 'Minimum scans (data points)'. This parameter is very similar to the 'Minimum consecutive scans' settings in the ADAP chromatogram builder module (Step 4); therefore, the same recommendations can be followed
- Define the allowed 'Peak duration range'. This is the acceptable duration of a chromatographic peak to be retained as a feature after the resolving. This parameter can be used to filter out noisy features based on their overly short, or long, duration. We recommend setting the minimum duration to 0 (the previous parameter already defines a minimum peak duration in terms of data points) and not be too strict when setting the maximum duration, because tailing chromatographic peaks can also be discarded
- Provide a 'Suffix' (e.g., '_RT-res') to name the newly created feature lists

IMS expanding and resolving (only for IMS data)

▲ **CRITICAL** At this stage, only the merged frame spectra have been examined for the detection of RT-resolved features, while the IMS dimension has not yet been considered. In the next three steps (i.e., Steps 7–9), the individual mobility scans are inspected to create IMS-resolved features. Steps 7–9 are required only when processing LC–IMS–MS data. If you are processing LC–MS data, skip these steps and go to Step 11 Similar to chromatogram building and resolving, we encourage the user to explore the raw data and gain the necessary insights to choose the optimal processing parameters. This can be done using the 'Ion mobility raw data overview' module (Extended Data Fig. 4).

7. (Only IMS data) Expanding features to the IMS dimension. In this step, MZmine takes the *m*/*z* associated to every RT-resolved feature and searches the individual mobility scans for signals to build the corresponding extracted ion mobilogram (EIM). A detailed description of the 'IMS expander module' is provided at https://mzmine.org/mzmine_documentation/module_docs/lc-ims-ms_featdet/featdet_ims_expander/ims-expander.html.
    - Navigate to 'Feature detection → LC–IMS–MS → IMS Expander'
    - Enable and set the '*m*/*z* tolerance'. This is the maximum allowed deviation between the *m*/*z* of the RT-resolved features and the *m*/*z* signals in the individual mobility scans. In contrast to the chromatogram building and resolving steps, a higher *m*/*z* tolerance might be needed; this is because the mass accuracy in individual mobility scans tends to be lower compared with the frame spectra. We recommend 0.005 *m*/*z* or 15 ppm as a starting point for most TOF–MS instruments
    - (Optional) 'Raw data instead of thresholded'. When enabled, this option replaces the noise level used in the mass detection with the provided intensity threshold
    - Disable 'Override default mobility bin width (scans)' to use MZmine's default binning of mobility scans. This option is usually not required for general untargeted analysis
    - Disable 'Maximum features per thread'. When enabled, this option controls thread parallelization, which affects RAM consumption and processing time. It is usually not needed for processing LC–IMS–MS data

    ◆ **TROUBLESHOOTING**

8. (Only IMS data, optional) Mobilogram smoothing. Similar to Step 5, we recommend applying smoothing to EIMs only if they exhibit a jagged profile, as this may cause inaccurate resolving of mobility features (Step 9). Since this is often the case for LC–IMS–MS data, we generally recommend performing this step.
   - Navigate to 'Feature detection → Smoothing'
   - Specify the 'Feature lists' to process (Step 5)
   - Choose the 'Smoothing algorithm'. We recommend using 'Savitzky Golay'
   - Click the 'Setup' button, and tick the 'Mobility smoothing' checkbox
   - Within Setup, set the number of data points to use for smoothing. We recommend using half the number of data points of a mobility peak
   - Tick the 'Show preview' checkbox and select 'Mobility' as preview dimension to open an interactive visualization panel. Use the drop-down menus to select, respectively, the feature list and feature to display. We recommend choosing a medium-intensity EIC trace that well represents the 'jaggedness' in the data
   - Specify how to handle the 'Original feature list' (Step 5)
   - Provide a 'Suffix' to name the newly created feature lists (e.g., '_IMS-smooth')

   ▲ **CRITICAL STEP** When changing the smoothing parameters, the preview does not update automatically. It is necessary to select a new feature from the drop-down menu to visualize the newly set parameters.
9. (Only IMS data) Mobilogram resolving. Similar to what was described in 'Chromatogram building and resolving', the EIM traces built in Step 7 have to be split into individual mobility peaks using a resolving algorithm. We recommend using the local minimum resolver module. Since the parameters of this module are already described in Step 6, we focus on the differences between EIMs and EICs resolving here.
   - Navigate to 'Feature detection → Chromatogram resolving → Local minimum resolver'
   - Select 'Dimension → Mobility'
   - Enable the 'MS/MS scan pairing' option and proceed as described in Step 6

   At this point, consider the following IMS-related parameters:
   - Enable the 'Limit by ion mobility edges' option. This option pairs a mobility-resolved feature with the corresponding MS^2 spectrum only if the latter was triggered within the feature's'mobility range (more details are available at https://mzmine.org/mzmine_documentation/module_docs/featdet_resolver_local_minimum/local-minimum-resolver.html)
   - (Optional) 'Merge MS/MS spectra (TIMS)'. This option only applies to fragmentation MS^2 spectra acquired in PASEF mode. When enabled, multiple MS^2 spectra acquired for the same precursor *m*/*z* and associated to the same feature are merged into a single spectrum. We recommend enabling this option when low-abundant compounds are of interest
   - (Optional) Enable and set the 'Minimum signal intensity (absolute and relative)'

   When the 'Merge MS/MS spectra (TIMS)' option is enabled, these two thresholds can be used to remove low-intensity signals from the merged MS^2 spectra (see previous parameter).
   - Set the 'Chromatographic threshold'. A lower value compared with the EIC resolving step should be used because mobility peaks are generally wider and less resolved than LC peaks. We recommend using a value between 35% and 70%
   - Adjust the 'Minimum search range RT/Mobility' parameter. This is the mobility window used for the local minimum search. We recommend starting the optimization at 0.005 for TIMS, 0.5 for travelling wave-IMS (TWIMS), and 1 for drift time-IMS (DTIMS) devices
   - Provide a 'Suffix' (e.g., '_IM-res') to name the newly created feature lists
   - All other parameters can be optimized as described in Step 6

   ▲ **CRITICAL STEP** The optimal mobility search range mainly depends on the IMS unit and scale employed by the instrument used for the analysis. For example, TIMS devices measure the ion mobility in Vs/cm^2 (typically between 0.5 and 2.0), whereas time-dispersive instruments (TWIMS and DTIMS) use milliseconds (typically between 0 and 90).

10. (Only IMS data, optional) CCS calibration and calculation. MZmine supports two methods for collision cross-section (CCS) calibration: either an external calibration file is imported or a calibration can be calculated using a list of reference compounds. The external calibration is described below, whereas the calibration method based on reference compounds is covered at https://mzmine.org/mzmine_documentation/index.html. After applying the vendor calibration software, the Agilent raw data folder contains a 'OverrideImsCal.xml', whilst Waters raw data contains a 'mob_cal.csv' file. Bruker raw data is automatically calibrated during the import from .tdf raw files and does not require this step.
    - Navigate to 'Feature list methods → Processing → External CCS calibration'
    - Specify the 'Raw files' to process
    - Select the external 'Calibration file'
    - Click the 'OK' button. After applying the calibration, the CCS values are automatically calculated in the 'Isotope pattern finder' step

Isotope filtering

▲**CRITICAL** During the EIC building, EIC traces are constructed for all the *m*/*z* signals detected during the mass detection. As a consequence, signals generated by isotopologues of the same chemical entity produce multiple features in the feature list, which constitutes redundant information for downstream data analysis. This is a common issue for C-containing molecules, where the ^{13}C isotope signal is easily detected. At the same time, the isotopic pattern holds essential information for the purpose of feature annotation.

11. ^{13}C isotope filter. This module removes ^{13}C-related features from the processed feature lists and assigns the retrieved ^{13}C isotopic pattern to the monoisotopic peak.
    - Navigate to 'Feature list methods → Isotopes → 13C isotope filter'
    - Specify the 'Feature lists' to process (Step 5)
    - Specify how to handle the 'Original feature list' (Step 5)
    - Set the '*m*/*z* tolerance (intra-sample)'. This is the maximum allowed *m*/*z* difference between the examined feature and its potential ^{13}C-isotopologues in the feature list. We recommend using a fairly strict tolerance, based on the MS analyzer performance
    - Set the 'Retention time tolerance'. This is the maximum allowed RT deviation between potential ^{13}C-related features. As isotopologues should produce fully overlapping chromatographic peaks, a strict tolerance can be used
    - (Only IMS data) Enable and set the 'Mobility tolerance'. This is the maximum allowed IMS deviation between potential ^{13}C-related features. Here, a strict mobility tolerance can be used since isotopologues should undergo identical IMS separation. We recommend 0.008 for TIMS and 0.5 for TWIMS and DTIMS
    - Tick the 'Monotonic shape' checkbox to filter ^{13}C-related features only when the retrieved isotopic pattern has a monotonically decreasing trend (typical for small molecules). For small molecule applications, we recommend enabling this option
    - Set the 'Maximum charge' state to be considered when calculating the *m*/*z* of ^{13}C isotopes. For small molecules applications, we recommend to use 1 or 2
    - Set the 'Representative isotope' as 'most intense'
    - Tick the 'Never remove features with MS2' checkbox to avoid filtering ^{13}C-related features for which an MS^2 scan has been acquired (even though they match the filtering criteria)
    - Provide a 'Suffix' (e.g., '_deiso') to name the newly created feature lists

    ▲**CRITICAL STEP** This module removes features matching the filtering criteria from the feature lists being processed. This also means that false ^{13}C-related features can be erroneously discarded. Therefore, we recommend using fairly strict tolerances, based on the instrument performance, to reduce such a risk.

12. Isotope pattern finder. This module searches and annotates potential isotope patterns for each feature based on its *m*/*z* and a list of chemical elements specified by the users.

According to the retrieved isotope pattern, a charge state is also assigned. Unlike the 'Isotope filter', this module does not remove isotopic features, but annotates them as part of an isotopic pattern.

- Navigate to 'Feature list methods → Isotopes → Isotope pattern finder'
- Click the 'Setup' button and select the 'Chemical elements' to consider for the isotope search
- Set the '*m*/*z* tolerance (feature-to-scan)'. This is the maximum allowed *m*/*z* difference between the examined feature and its potential isotopologues. Since this module uses the raw data to find potential isotope signals, a slightly wider *m*/*z* tolerance may be appropriate compared with the ^{13}C isotope filter
- Set the 'Maximum charge of isotope *m*/*z*' to be considered when calculating the isotopes' *m*/*z*. For small molecules applications, we recommend using 1 or 2
- Set the 'Search in scans' as 'Single most intense'

Alignment and gap-filling across samples

▲ **CRITICAL** Any untargeted MS experiment performed on multiple samples aims at comparing, qualitatively or quantitatively, the analytes detected across the set of analyzed samples. However, chromatography–MS experiments are subject to instrumental drift that produces fluctuations in RT, ion mobility and mass accuracy over the course of the analysis. As a consequence, the same analyte is almost never detected with the same RT, *m*/*z* and mobility over consecutive LC–(IMS)–MS runs. The goal of feature alignment is to account for such variations and align the features corresponding to the same molecular entity across different instrument runs[43,44]. By doing so, feature lists from multiple samples can be merged into a single, aligned feature table.

13. Join aligner. In MZmine, the feature alignment is based on alignment scores calculated using a combination of user-defined tolerances and weights for each available analysis dimensions (i.e., *m*/*z*, RT and ion mobility; for more details, see Box 5). Although various alignment algorithms are available, we recommend using the 'Join aligner' module for LC–(IMS–)MS data. A more detailed description of this module is provided at https://mzmine.org/mzmine_documentation/module_docs/align_join_aligner/join_aligner.html.
    - Navigate to 'Feature list methods → Alignment → Join aligner'
    - Specify the 'Feature lists' to process (Step 5)
    - Set an '*m*/*z* tolerance (sample-to-sample)'. This is the maximum allowed *m*/*z* deviation between the samples for feature alignment. This is a sample-to-sample tolerance and largely depends on the performance (stability) of the MS analyzer over time. We recommend 0.005 or 15 ppm as a starting point for most Orbitrap and TOF instruments
    - Set a 'Retention time tolerance'. This is the maximum allowed RT deviation between the features being aligned. This is a sample-to-sample tolerance and largely depends on the reproducibility of your chromatographic system
    - (Only IMS data) Enable and set a 'mobility tolerance'. This is the maximum allowed mobility deviation between the features being aligned
    - Set the 'Weight for *m*/*z*', 'Weight for RT' and 'Mobility weight'. These weights define the importance given to *m*/*z*, RT and mobility, respectively, when multiple features fall within the tolerance (Box 5). We recommend assigning equal weight (e.g., 1) to all the dimensions as a starting point
    - Disable all other remaining options. They can be used in particular applications that require higher confidence in the alignment (Box 5)
    - Provide a 'Feature list name' to name the newly created aligned feature lists
14. (Optional) Gap filling. One of the main challenges in untargeted feature detection is reproducible detection of low-intensity features. In fact, these can be erroneously filtered out at various stages of the workflow, producing false missing values (i.e., gaps) in the aligned feature table. MZmine offers the possibility to re-inspect such gaps by checking for the presence of omitted signals in the original raw data. If a relevant signal is found, it is integrated and reincluded in the feature table, thus filling the gap. In MZmine, this process is called 'gap filling' and can be performed using the Peak finder module. During

BOX 5

Feature alignment

During the alignment, multiple features can fall within one, or more, tolerance windows set for each analysis dimension (i.e., RT, *m/z* and mobility). The best alignment match is chosen using a weighted scoring system that considers all the available analysis dimensions to assign a global alignment score. Alignment scores are calculated using the following equation:

$$\mathrm{score}_p = \sum\nolimits_{\mathrm{dim}} \left(1 - \frac{\Delta_{\mathrm{dim}}}{\mathrm{tolerance}_{\mathrm{dim}}}\right) \cdot \mathrm{weight}_{\mathrm{dim}}$$

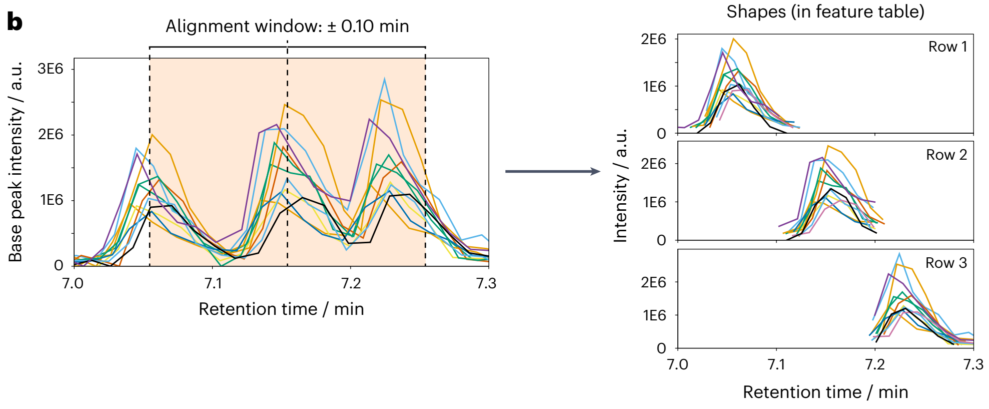
where:

- Δ_{dim} is the difference in the value of the considered dimension (e.g., dim = RT) between the features being aligned. Perfectly matching RTs will produce a $\Delta_{\mathrm{RT}} = 0$
- $\mathrm{tolerance}_{\mathrm{dim}}$ is the maximum sample-to-sample allowed deviation for the considered dimension
- $\mathrm{weight}_{\mathrm{dim}}$ is the importance given to each dimension in the calculation of the global alignment score
- score_p is the global alignment score for the features being aligned. It is obtained by summing the weighted score of each dimension

Unlike other steps (e.g., 'Chromatogram building and resolving'), no 'Show preview' option is available to interactively assess the alignment quality. Therefore, the alignment results can only be evaluated a posteriori. To do so, a set of metrics can be displayed for each feature in the aligned feature list (see orange columns in panel **a** of the box figure). These include the 'Aligned features' (i.e., number of aligned samples), 'Rate' (i.e., ratio between the number of aligned and total samples), 'Extra features' (i.e., number of other possible alignment matches within the defined tolerances), 'weighted distance score' (i.e., reflects the alignment score_p) and the average difference between the value of each dimension pre- and post-alignment (i.e., 'Δ *m/z*', 'Δ RT', and 'Δ Mobility' columns).

a

					Alignment (13 samples)							
ID	RT	m/z	Height	Area	Rate	Aligned features	Σ Extra features	Weighted distance score	Δ m/z ppm	Δ m/z	Δ RT	Δ Mobility
1	7.05	293.1860	2.0E6	9.1E4	0.85	11	9	0.96	0.23	0.0001	0.008	
2	7.15	293.1860	2.5E6	1.3E5	0.85	11	17	0.97	0.26	0.0001	0.005	
3	7.23	293.1860	2.8E6	1.5E5	0.85	11	11	0.96	0.34	0.0001	0.007	



the gap filling, artifacts (duplicate features) can be introduced in the feature table in case of misaligned features (for more information, see the online documentation (https://mzmine.org/mzmine_documentation/module_docs/gapfill_peak_finder/gap-filling.html)). Such artifacts can be removed later using the 'Duplicate filter' module (Step 15).

- Navigate to 'Feature list methods → Gap filling → Peak finder'
- Specify the 'Feature lists' to process (Step 5)
- Set the 'Intensity tolerance'. Maximum allowed intensity deviation between consecutive scans when building the EIC for gap-filled features. A higher tolerance will retain more 'jagged' EICs during gap-filling. We recommend 20% as a starting point

- Set the '*m*/*z* tolerance sample-to-sample'. Maximum allowed *m*/*z* deviation between gap-filled signal in the raw data and the feature's *m*/*z* in the feature table. It is a sample-to-sample tolerance and the same recommendation provided in Step 13 can be followed
- Set the 'Retention time tolerance (sample-to-sample)'. This is the RT window (around the feature's RT) used to examine the raw data for gap filling. It is a sample-to-sample tolerance; therefore, the same recommendation provided in Step 13 can be followed
- Set the 'Minimum scans (data points)'. This parameter is very similar to the 'Minimum consecutive scans' settings in the ADAP chromatogram builder module (Step 4); therefore, the same recommendations can be followed
- Provide a 'Suffix' (e.g., '_gap-filled') to name the newly created feature lists

15. (Optional) Duplicate filter. This module is intended for removing artifacts (duplicate features) that can originate from the gap filling of misaligned features. In fact, when a misaligned feature undergoes gap filling, all the correctly aligned signals are retrieved, thus creating a 'duplicate feature' (for more information, see the online documentation at https://mzmine.org/mzmine_documentation/module_docs/filter_duplicate_features/duplicate_feature_filter.html). The 'Duplicate filter' module removes such duplicates by merging them into one consensus feature.
    - Navigate to 'Feature list methods → Feature list filtering → Duplicate feature filter'
    - Select the 'Filter mode'. This parameter determines how the RT and *m*/*z* of the consensus feature are calculated after the merging. The 'NEW AVERAGE' option (recommended) recalculates the features RT and *m*/*z* as average between the duplicate features
    - Set the '*m*/*z* tolerance', 'RT tolerance' and, for mobility data, enable and set a 'Mobility tolerance'. Features falling within these tolerances will be considered duplicates and thus removed. Therefore, we recommend using strict tolerances to avoid removing false duplicates
    - Disable the 'Require same identification' checkbox
    - Provide a 'Suffix' (e.g., '_dup-filt') to name the newly created feature lists

    ▲ **CRITICAL STEP** Similar to the '^{13}C isotope filter' (Step 11), this module removes features matching the filtering criteria. Therefore, we recommend using strict tolerances to avoid removing false duplicates. As a rule of thumb, stricter tolerances than those used in the alignment step (Step 13) must be used.

Feature annotation

▲ **CRITICAL** MZmine offers various feature annotation modules to assign ion adducts, molecular formulas, and chemical structures to the detected features. Furthermore, harmonized data exchange formats enable direct interface of MZmine with other annotation tools. Figure 3 provides an overview of the most popular modules and third-party tools for feature annotation integrated with MZmine. The full list of available feature annotation tools is provided at https://mzmine.org/mzmine_documentation/tool_integration.html.

16. (Optional) Feature grouping: connecting ions of the same molecule. MZmine annotates features originating from the same chemical entity (e.g., multiple adducts) in two subsequent steps: Step 16 (feature grouping) and Step 17 (ion identity networking (IIN). The metaCorrelate module searches for features with overlapping RT and chromatographic profiles to annotate them as 'correlated features'. Besides checking if features fall within the same RT window, the chromatographic profile can be considered to distinguish between ions actually originating from the same molecule and features that are just co-eluting (Fig. 4).
    - Navigate to 'Feature list methods → Feature grouping → Correlation grouping (metaCorrelate)'
    - Specify the 'Feature lists' to process (Step 5)
    - Set the 'RT tolerance'. This is the maximum allowed RT deviation between features to be grouped together. We recommend using a strict tolerance (e.g., ~FWHM / 3) when the 'Feature shape correlation' option is disabled. Otherwise, a wider tolerance

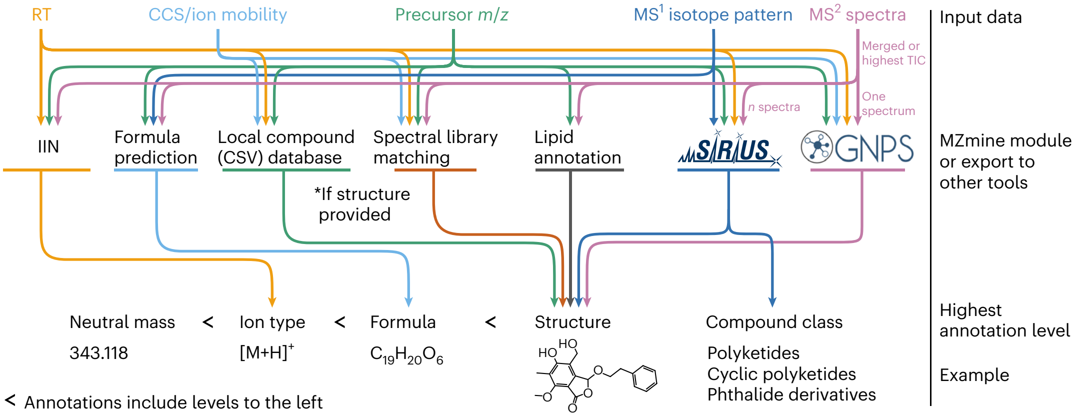


Fig. 3 | An overview of popular modules and third-party tools for feature annotation integrated with MZmine. The various modules and third-party tools use different information retrieved during the MZmine preprocessing (e.g., precursor *m*/*z*, isotope pattern, MS2 spectra) to assign annotation.

(~FWHM × 2) can be used because the 'Feature shape correlation' option will provide a stringent filter for grouping

- Set the 'Minimum feature height' for a feature to be considered for the grouping. Features with intensity below this threshold will be disregarded. Set it to 0 to ignore this parameter
- Set an 'Intensity threshold for correlation'. This threshold is used by the 'Feature shape correlation' option, when enabled. Data points with intensity below this threshold will be disregarded. Set it to 0 to ignore this parameter
- (Optional) Click the 'Show' button and set the 'Min samples filter'

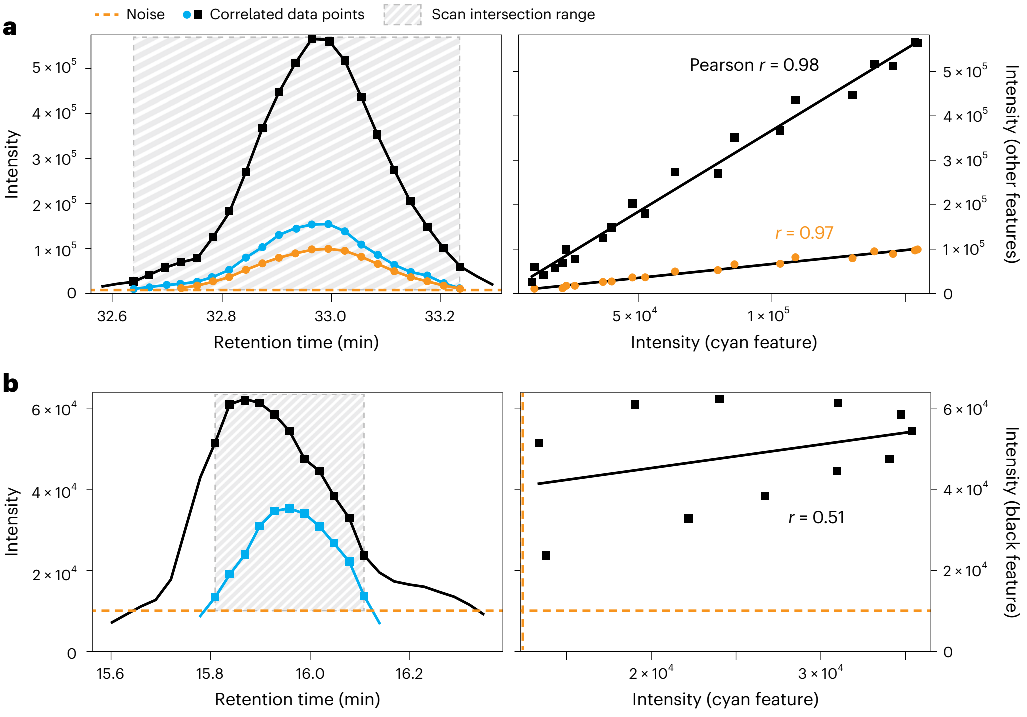


Fig. 4 | Feature grouping by feature shape correlation. a, Features that originate from the same molecule exhibit highly correlated feature shapes at the same RT. **b**, The Pearson correlation drops substantially for features with different shapes or slight RT shifts.

- Set the 'Min samples in all' as the minimum number of samples (absolute or relative) in which two features must be detected together to be grouped. This may exclude or include gap-filled features, see below
- Set 'Min samples in group' to 0 to ignore this parameter. This can be used when sample groups are included in the experimental design and the information is provided using the 'Sample metadata' module (more information is available at https://mzmine.org/mzmine_documentation/index.html)
- Set the 'Min %-intensity overlap'. This is the minimum intensity overlap between the smaller feature and the rest of the features being grouped. The intensity overlap is calculated considering the sum of all data point intensities within the RT range of the features being grouped. The default value (60%) should provide good results for most applications
- Enable the 'Exclude estimated features (gap-filled)' option to ignore gap-filled features during the grouping. This also applies to the minimum samples filters. We recommend using this option when smoothing was applied during the processing (Steps 5 and 8)

In the main dialogue window continue setting the following parameters:

- Enable the 'Feature shape correlation' option and set the corresponding parameters by clicking the 'Show' button. When enabled, the features' chromatographic profile is taken into account for the grouping. We recommend using this option if most features have at least five MS^1 data points (i.e., points-per-peak), two on each side of the apex. The following default parameters should provide good results for most applications:

'Min data points = 5' and 'Min data points on edge = 2'	These are the minimum numbers of total data points and data points per peak side a feature must exhibit to be considered for grouping. According to the typical number of points-per-peak in the data, these values can be increased to make the feature grouping more strict
'Measure = PEARSON'	Although other correlation measures are available, we recommend using the Pearson correlation as a starting point
Set the 'Min feature shape correlation'	This is the minimum level of correlation between the chromatographic profiles of the feature being grouped. A 85% Pearson correlation threshold (default) should provide good results for most applications
Disable the 'Min total correlation' option	This option represents an additional constraint that considers all data points from all the features being grouped (more information is available at https://mzmine.org/mzmine_documentation/module_docs/id_ion_networking/metacorr/metacorr.html)

- (Optional) Enable the 'Feature height correlation' option and set the corresponding parameters by clicking the 'Show' button. This represents an additional constraint for the grouping based on the heights of feature pairs across samples (see the online documentation at https://mzmine.org/mzmine_documentation/module_docs/id_spectral_library_search/spectral_library_search for more information). When enabled, the default parameters should provide good results for most applications
- Provide a 'Suffix' (e.g., '_corr') to name the newly created feature lists

17. (Optional) IIN: identification of ions This step examines correlated features (i.e., features with same RT and chromatographic profile) to annotate those generated by multiple adducts (e.g., $[M+H]^+$ and $[M+Na]^+$) or in-source modifications (e.g., $[M-H_2O]$) of the same chemical entity. To do so, grouped features (Step 16) are compared pairwise against a user-defined list of adduct ions (e.g., [M+H]+), modifications (e.g., $[M-H_2O]$), and multimeters. When the *m/z* difference between two features corresponds to a possible adduct or in-source modification, they are annotated (Fig. 5a). Multiple ion features that describe the same neutral molecule (neutral mass) are then represented by an ion identity network, which is refined in a final step to only retain the most confident networks (i.e., the largest number of ions pointing to the same neutral mass; Fig. 5c). To perform this step, the Feature grouping module has to be run first (Step 16).
    - Navigate to 'Feature list methods → Feature grouping → Ion identity networking'
    - Set the '*m/z* tolerance (intra-sample)'. This is the maximum allowed *m/z* deviation when annotating two features as adducts pair or modification. This tolerance is very similar

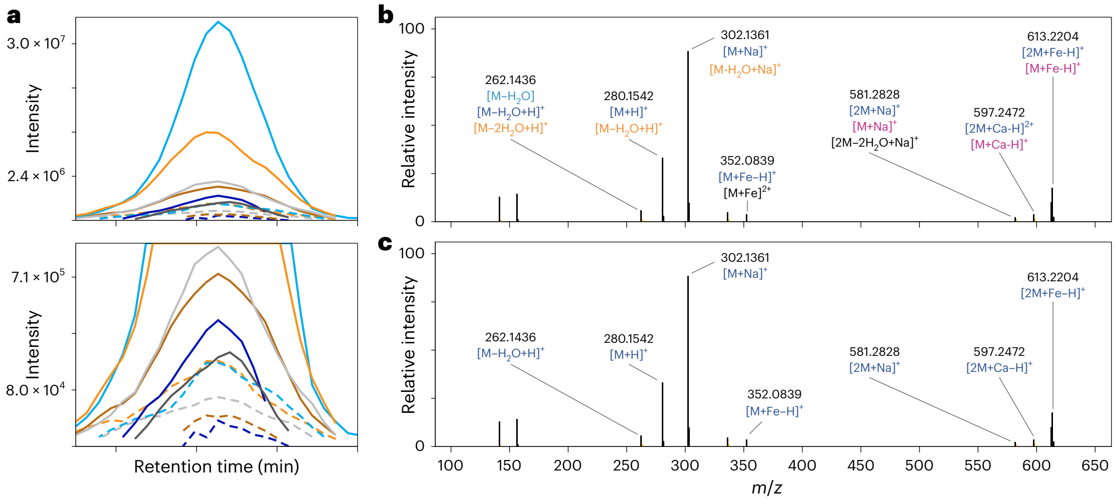


Fig. 5 | IIN annotation refinement. a, Grouped features are searched pairwise against a user library of ion adducts, in-source fragments and multimers. **b**, Annotated IIN results for an example molecule in the MS1 spectrum. Each signal might be explained by different ions where annotations are linked in ion identity networks (colored labels). **c**, After IIN refinement, only the best annotation that is supported by the largest network is retained. This means that [M−H_2O] that was only defined by a single difference of *m/z* −18 can be defined as [M−H_2O+H]+ through connections to other ion adducts, such as [M+Na]+.

to the one used in the ^{13}C isotope filter (Step 11); therefore, the same recommendations can be followed

- Select ‘Check = ONE FEATURE’ to annotate two features if their *m/z* difference matches a possible adduct in at least one sample where the features were detected. The ‘ALL FEATURES’ option is more stringent and requires the *m/z* difference to match in all samples the features were detected
- Set the ‘Min height’. This is the minimum height for a feature to be considered for the adduct annotation. Set it to 0 to ignore this parameter and consider all features regardless of their intensity
- Define the ‘Ion identity library’ to use for the annotation by clicking the ‘Setup’ button and setting the following parameters:

‘MS mode’	Ionization polarity
‘Adducts’ and ‘Modifications’	Adducts and modifications’ to consider for the annotation. Adducts and modifications are combined together (e.g., [M−H_2O+Na]+) to create the final list of ions to search for. Use the ‘Reset positive’ and ‘Reset negative’ buttons to show/restore the default adduct lists
‘Maximum charge’ and ‘Maximum molecules/ cluster’	Maximum charge and multimers to consider for the annotation. We recommend setting both these parameters to 2 as a starting point for applications involving small molecules

- (Optional) Enable and set the ‘Annotation refinement’ by clicking the ‘Show’ button. These are additional constraints to consider for the adduct annotation and retain only the most confident annotation (more information is available at https://mzmine.org/mzmine_documentation/module_docs/id_ion_networking/metacorr/metacorr.html). When enabled, the default parameters should provide good results for most applications

18. (Optional) Import spectral libraries. To perform feature annotation based on spectral matching (Step 19), spectral library files first have to be imported into MZmine. The following file formats are supported: .json (e.g., the MoNA and GNPS libraries), .mgf, .msp (e.g., the National Institute of Standards and Technology library) and .jdx. Library files can be imported in MZmine by drag and drop directly in the ‘Libraries’ tab in the main window. Alternative

ways of importing files are described in the online documentation (https://mzmine.org/mzmine_documentation/module_docs/id_spectral_library_search/spectral_library_search.html). Some of the most popular public spectral libraries can be freely downloaded using the links provided below. We recommend using the .json format when available:

- MassBank of North America (MoNA): https://mona.fiehnlab.ucdavis.edu/downloads. Several libraries are available. If you are using the example dataset provided for this procedure, we recommend downloading the 'LC-MS/MS Positive Mode' library
- GNPS: https://gnps-external.ucsd.edu/gnpslibrary. Several different libraries are available. If you are using the example dataset provided for this procedure, we recommend downloading the 'GNPS-LIBRARY' library
- MassBank: https://github.com/MassBank/MassBank-data/releases/tag/2022.12.1. Download the 'MassBank_NIST.msp' file

19. (Optional) Spectral library search. Spectral library matching is the main approach for metabolite annotation in MS-based experiments. The collected fragmentation spectra are matched against reference spectral libraries to assign putative chemical structures to features matching a set of user-defined criteria, for example, *m/z* tolerance and similarity score. In addition, RT- and CCS-based constraints can be used for chromatography–MS and ion mobility data. To perform this step, spectral library files first have to be imported into MZmine (Step 18). In-house created, commercial (e.g., NIST) and open spectral libraries (e.g., MoNA, GNPS) can be used. The spectral library search can be performed on entire feature lists or individual features selected from a feature list. To process the entire feature list(s), navigate to 'Feature list methods → Annotation → Search spectra → Spectral library search'. In the dialogue box, specify the 'Feature lists' to process. For individual features open the feature list → Select one or multiple features → Right click → 'Search → Spectral library search'.
    - Specify the 'Spectral libraries' to use for the spectral matching. Libraries must be already imported (Step 18)
    - Set the 'Scans for matching'. For LC–MS/MS experiments, use the 'MS2 ≥ (merged)' option
    - Set the 'Precursor *m/z* tolerance'. This tolerance applies to the precursor *m/z* (not to the fragment *m/z*) and serves as a filter to reduce the number of library entries to match. A higher *m/z* tolerance can be set when public spectral libraries are used, as they are generally acquired with different instrument types, resulting in varying mass accuracy levels. We recommend 0.01 *m/z* or 20 ppm as a starting point
    - Set the 'Spectral *m/z* tolerance' to pair the fragment *m/z* signals of the experimental and library spectra. As for the 'Precursor *m/z* tolerance', strict tolerances should be avoided. We recommend setting this tolerance higher than the 'Precursor *m/z* tolerance'. A good starting point is 0.01 *m/z* and 25 ppm
    - Enable 'Remove precursor' to exclude the precursor *m/z* signal (± 4 Da) from the matching. This option can be useful when comparing experimental and library spectra acquired under different fragmentation methods (e.g., fragmentation mode or collision energies). This can result in varying intensities of the precursor ion, which decreases the overall spectral similarity. On the other hand, this option can reduce false library matches due to high abundant precursor ions in both the experimental and library spectra that match and account for most of the similarity
    - Set the number of 'Minimum matched signals' that needs to be paired between the experimental and library spectra. A higher number (e.g., 6) results in increased confidence in the match. However, overly high values can impair the spectral matching for molecules with poor fragmentation patterns. We recommend 4 as a starting point for small molecules. Lower values will increase the probability of false library matches
    - Select the 'Similarity' calculation algorithm to use for the spectral matching. This is the algorithm used to calculate the similarity between the experimental and library spectra. Choose the 'Weighted cosine similarity' for MS^2 data and the 'Composite cosine identity' for MS^1 and GC–EI–MS data, as this algorithm considers the relative intensity of neighboring signals (more detailed information about the cosine similarity

calculation can be found at https://mzmine.org/mzmine_documentation/module_docs/id_spectral_library_search/spectral-similarity-measures.html). With the 'Setup' button, further parameters are available:

'Weights'	Weights for calculating the cosine similarity between the experimental and library spectra based on the *m*/*z* and signal intensity
'Minimum cos similarity'	Only the library matches with cosine similarity above this threshold will be considered. A minimum similarity of 0.7, although the threshold to use largely depends on the next parameter
'Handle unmatched signals'	Determines how nonmatching signals (i.e., *m*/*z* signals that occur only in the experimental or library spectrum) affect the cosine similarity. We recommend the setting 'KEEP ALL AND MATCH TO ZERO' for GC–EI–MS and MS2 spectra

- Disable all the 'Advanced' options by unticking the corresponding checkbox. These options can be used to add further constraints to consider in the library search. A detailed description of these options can be found at https://mzmine.org/mzmine_documentation/module_docs/id_spectral_library_search/spectral_library_search

▲**CRITICAL STEP** Public spectral libraries contain mass spectra acquired under a wide range of instrumental conditions and using a wide range of sample preparation and data curation protocols[45]. As a consequence, spectra can vary greatly in terms of observed mass fragments, intensity ratios, and spectral quality[46]. For this reason, we recommend the users to treat annotation results with caution when public data repositories are used for automated spectral matching.

◆ **TROUBLESHOOTING**

20. (Optional) Local compound database search. The local compound database search requires a compound table (i.e., a text file) with at least one of the following pieces of information: precursor *m*/*z*, neutral mass, chemical formula or chemical structure (i.e., SMILES). Additional details, such as RT, mobility and CCS, can also be used as further annotation constraints. When the neutral mass, chemical formula or SMILES is provided, the *m*/*z* of the corresponding adducts can be automatically calculated and used for the annotation.
    - Navigate to 'Feature list methods → Annotation → Search precursor mass → Local compound database (CSV) search'
    - Specify the 'Feature lists' to process (Step 5)
    - Click the 'Select' button and browse the database file in your filesystem
    - Specify the 'Field separator' of your database file (e.g., ',' for CSV files)
    - In the 'Columns' table, select the columns of the database file to import by ticking the corresponding checkbox. The names of columns to import are specified under 'Column name (csv)'. To edit them, double-click on the name, type the new column header and press enter. The attribute each database column corresponds to is specified under 'Data type (MZmine)'
    - (Optional) Enable the 'Use adducts' option to calculate and use the *m*/*z* of the adduct for the annotation, instead of the precursor *m*/*z*. When this option is enabled, MZmine automatically calculates the *m*/*z* of the specified adducts and/or in-source modifications based on the compound neutral mass. Therefore, to use this option, one among neutral mass, chemical formula or SMILES information must be provided in the database file. The list of adducts to search can be specified by clicking the 'Setup' button

'MS mode'	Ionization polarity
'Adducts' and 'Modifications'	Adducts and modifications to consider for the annotation. Adducts and modifications are combined together (e.g., $[M{-}H_2O{+}Na]+$) to create the final list of ions to search for. Use the 'Reset positive' and 'Reset negative' buttons to show/restore the default adduct lists
'Maximum charge' and 'Maximum molecules/cluster'	Maximum charge and multimers to consider for the annotation. We recommend setting both these parameters to 2 as a starting point for applications involving small molecules

- Set the 'm/z tolerance' as the maximum allowed deviation between the experimental mass and the exact mass provided in the database file. This tolerance depends mainly on the mass accuracy offered by the mass spectrometer used for the measurement
- (Optional) Disable the 'Retention time tolerance', 'Mobility time tolerance' and 'CCS tolerance (%)' options. When enabled, they are included as further annotation constraints. To do so, RT, mobility time and CCS value must be provided in the database file, respectively (zero and/or empty entries are ignored)
- (Optional) Disable the 'Filter filename header' option. This option is intended for library building workflows and restricts the matching to a specific sample. If this is not the case, ignore this option
- (Optional) Leave the 'Append comment fields' field empty

  ▲**CRITICAL STEP** For the import to be successful, the column headers in the database file must match exactly the names specified under 'Column name (csv)'.

21. (Optional) Lipid annotation. MZmine offers a dedicated module for lipid annotation, which comes integrated with a set of predefined fragmentation rules for several glycerolipid and glycerophospholipid classes and subclasses. Furthermore, custom rulesets can be defined by the user and used for the annotation of derivatization products, oxidized forms, etc. The module first generates a database of lipid species based on a list of selected lipid classes/subclasses, the number of possible carbon atoms and double bond equivalents (DBE). From this database, theoretical precursor m/z are calculated and searched within the feature list. Moreover, in silico fragmentation spectra can be predicted (using both predefined and custom fragmentation rules) and matched against the experimental MS^2 data. Lipid annotation results can be visualized through dedicated visualization modules (e.g., Box 6).
    - Navigate to 'Feature list methods → Annotation → Search spectra → Lipid annotation'
    - Specify the 'Feature lists' to process (Step 5)
    - Select the 'Lipid classes' to consider for the database generation

BOX 6

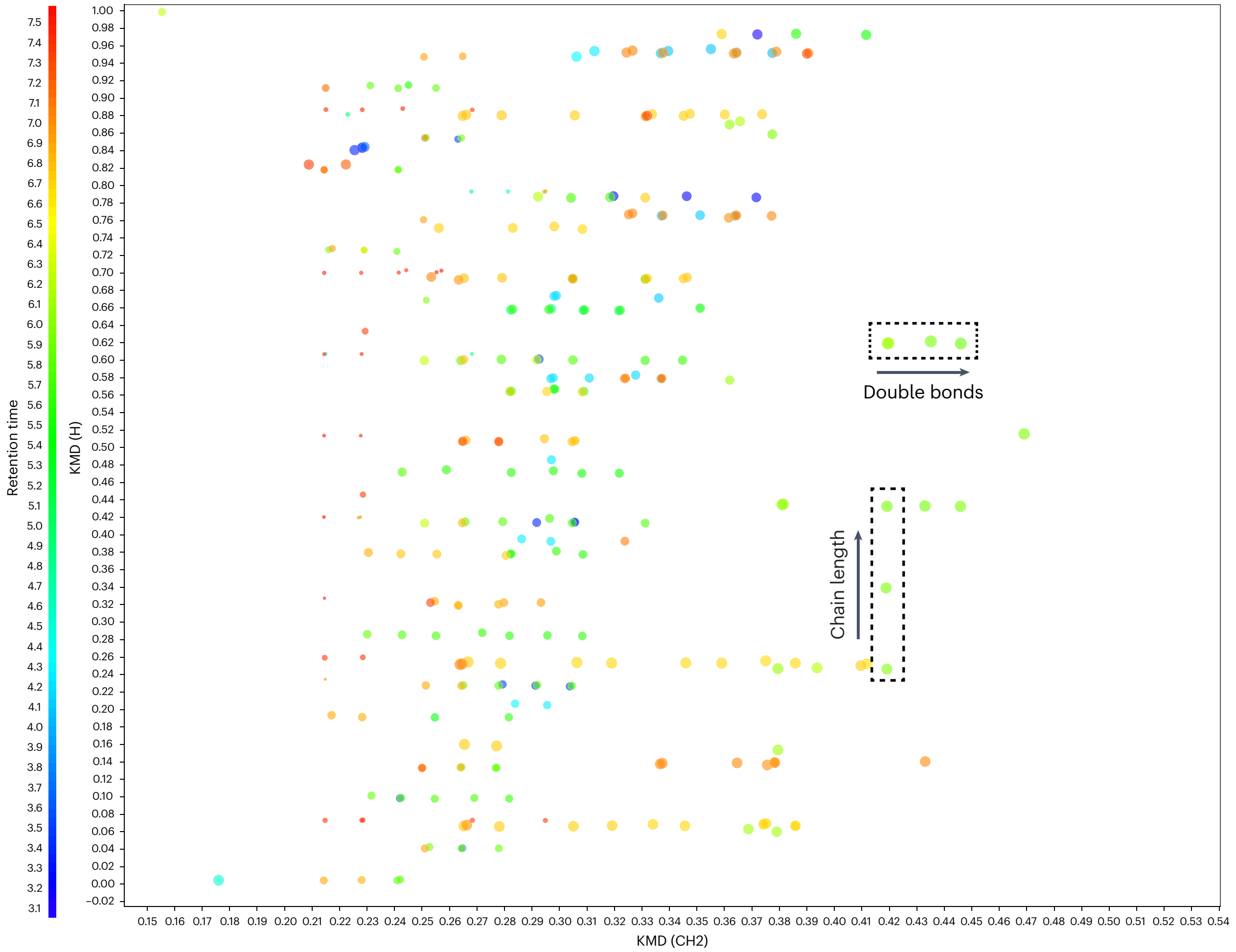
Visualization tools for lipidomics

The Kendrick mass defect (KMD) plot is a graphical tool to assist the identification of compounds that include repeating units in their chemical structures[59,60]. Although originally developed for petroleum analysis, the KMD plot can be used to highlight differences in the acyl chain length and saturation of homologous lipid species[15,59,61,62]. MZmine provides a module for the generation of KMD plots based on the repeating unit of interest (e.g., H_2, CH_2)[63]. Moreover, the chromatographic information (i.e., RT) can be included in the plot in the form of a colour-coded scale[64], thus enabling further visualization possibilities (see box figure). The KMD plot shown below can be generated as follows:

1. Navigate to 'Visualization → Kendrick mass plot'
2. Select the 'Feature list' to process
3. Select the 'Peaks' (i.e., features) to be used in the generation of the plot. Click the 'Set to all' button to use all the features in the feature list. Specific m/z and RT ranges can be set using the 'Add' button
4. Select the 'Kendrick mass base for y-Axis'. This is the structural unit of interest (e.g., CH_2, H) used to calculate the KMD displayed on the y axis
5. Choose the variable displayed on the 'X-axis' from the drop-down menu. The 'm/z' and 'KM' options are available. However, a second Kendrick mass base can be displayed on the x axis by enabling the 'Kendrick mass base for x-Axis' checkbox
6. Choose the variable displayed on the 'Z-axis' (i.e., in the form of a color-coded scale). Multiple options are available and, here too, another Kendrick mass base can be displayed on the z axis by enabling the 'Kendrick mass base for z-Axis' checkbox
7. Choose the 'Bubble size' (i.e., data point size). The size of data points displayed in the KMD plot can be scaled to a variable chosen from the drop-down menu
8. Select the 'Z-axis scale' and specify a 'Range for z-Axis scale' to display. This can be used, for example, to display specific RT ranges on the z axis
9. Select the 'Heatmap style'. Various color palettes are available

The plot displays the features annotated by the lipid annotation module (Step 21 in the main Procedure) using two KMDs against each other (i.e., H on the y axis and CH_2 on the x axis). By doing so, homologous lipid species form easily recognizable series based on the varying chain length (vertical) and level of unsaturation (horizontal). The size of data points and the color scale represent the m/z and RT of each feature, respectively.

(continued from previous page)



- Specify the ranges of 'Number of carbon atoms in chains' and 'Number of double bonds in chains' in all side chains combined. The selected ranges should be set in accordance with the selected lipid classes, as well as sample preparation and analysis methods, to minimize false positive annotations. For example, we recommend 14–26 carbons and 0–6 DBE for a lipid class with a single side chain, and 56–86 carbons and 0–18 DBE for cardiolipins
- Set the '*m*/*z* tolerance MS1 level'. This is the maximum allowed difference between the experimental and theoretical *m*/*z* values. This parameter mainly depends on the accuracy of the MS measurement
- (Optional) The 'Show database' button opens a separate window to visualize the lipid species database generated with the current parameters. Various info are displayed for each lipid species (e.g., exact mass, implemented fragmentation rule. Moreover, the 'Info' column highlights whether multiple annotations might occur if only the MS^1 information is considered (due to isomeric/isobaric overlap)

Activate the 'Search for lipid class specific fragments in MS/MS spectra' option if MS^2 data were acquired. When deactivated, annotations are assigned based on MS^1 data only. Click the 'Show' button and set the following parameters:

- Set the '*m*/*z* tolerance MS2 level'. This is the maximum *m*/*z* allowed difference between the experimental and theoretical fragment signals
- Set the 'Minimum MS/MS score' between 0 and 100. This is the portion of intensity of the theoretical MS^2 spectrum explained by the experimental MS^2 spectrum
- (Optional) Enable the 'Keep unconfirmed annotations' checkbox to annotations based on MS^1 data only (these will be labelled in the feature list)

Enable the 'Search for custom lipid class' option to use custom lipid classes for the annotation. To add a custom lipid class, click the 'Add' button and set the following parameters:

- Define the 'Custom lipid class name' (e.g., 'oxidised PC'), a 'Custom lipid class abbreviation' (e.g., 'PC+O'), and a 'Lipid Backbone Molecular Formula' (e.g., '$C_8H_{20}O_6PN$' for oxidized phosphatidylcholine). Multiple customized lipid classes can be defined and stored as a .json file
- Click the 'Add' button to 'Add Lipid Chains' to the backbone. At the time of writing, acyl and alkyl side chains are supported
- Enable the 'Add fragmentation rules' option to add fragmentation rules for the custom lipid class being created. Multiple fragmentation rules can be set and stored as a .json file. Click the 'Add' button and set the following parameters:

'Polarity'	Ionization polarity
'Ionization method'	Expected adduct type
'Lipid fragmentation rule type'	Type of fragmentation rule. Several different options are available (e.g., HEADGROUP_FRAGMENT)
'Lipid fragment information level'	Level of structural information the fragment can provide for the annotation (e.g., 'MOLECULAR_SPECIES_LEVEL')
'Molecular formula'	If a certain formula is needed for a fragmentation rule (e.g., headgroup fragment or headgroup neutral loss), specify the fragment's molecular formula (e.g., 'C5H15NO4P+' for the typical PC headgroup fragment)

▲ **CRITICAL STEP** Selecting many lipid classes with wide ranges of number of carbon atoms and DBE increases the size of the database exponentially and, thus, the computation time. For this reason, we recommend running the lipid annotation module multiple times and using specific ranges of carbon atoms and DBE for selected lipid classes (see the provided batch_lipid_annotation.xml).

Feature filtering

22. (Optional) Feature list rows filter. This module allows the user to remove unwanted entries in a feature list using different filters. All features matching the filtering criteria will be either kept or removed from the feature list. As several different filters are available and can be useful in specific applications, a few examples are explained below, and a detailed description for each filter is provided at https://mzmine.org/mzmine_documentation/module_docs/feature_list_row_filter/feature_list_rows_filter.
    - Navigate to 'Feature list methods → Feature list filtering → Feature list rows filter'
    - Enable the 'Min aligned features (samples)' filter and set the minimum number of samples (absolute or relative) in which a feature needs to be detected. This filter is commonly used to keep only features that were 'reproducibly' detected in analysis replicates or pooled quality control samples
    - Enable the 'Minimum features in an isotope pattern' filter and set the minimum number of isotope signals to be detected in a feature. This filter is commonly used to remove all those features for which an isotopic pattern was not detected
    - Enable the 'Never remove feature with MS2' option to always retain features associated with an MS^2 spectrum, regardless of the filters used. This option is commonly used when processing data for applications where the MS^2 data is the focus (e.g., molecular networking)
    - Set 'Keep or remove rows' to 'Keep rows that match all criteria'. The alternative option removes all features matching the selected criteria

Data export

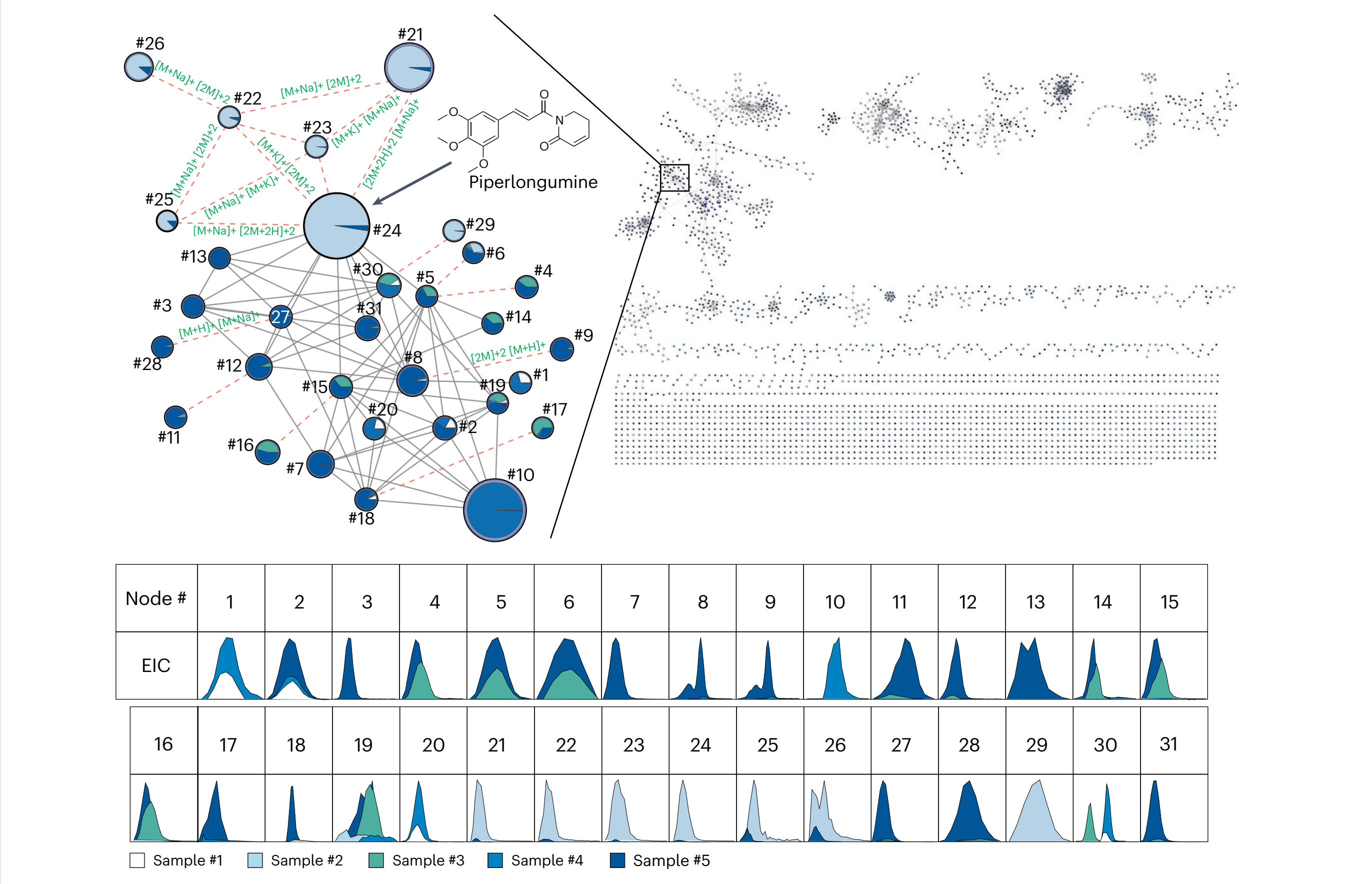
▲ **CRITICAL** MZmine enables the export of both quantitative (feature table) and qualitative (fragmentation spectra list) summaries of the feature detection and annotation workflow. Such outputs constitute the basis for a wide range of downstream data analysis such as feature-based molecular networking[47] (Step 23), software packages for compound structure prediction (e.g., SIRIUS, Step 24), and statistical and pathway analysis (e.g., MetaboAnalyst, Step 25). Over the years, a number of other tools have integrated the output from MZmine into their pipelines; a list of such tools is available at https://mzmine.org/mzmine_documentation/tool_integration. Besides export modules designed for specific third-party tools, export of feature lists is also possible via more general export modules available in MZmine and covered in the online documentation (https://mzmine.org/mzmine_documentation/index.html).

23. (Optional) Export for FBMN and ion identity molecular networking (IIMN). FBMN is a computational approach to increase metabolite annotation rates based on MS^2 spectral similarity (Box 7). As described below, MZmine 3 allows the direct export of quantification table and MS^2 spectral list files necessary to perform FBMN using third-party tools such as GNPS, MetGem or FERMO[47–50]. Moreover, feature correlation information from the 'Ion identity networking' module (Step 17) can also be exported to perform IIMN[51] on the GNPS platform.
    - Navigate to 'Feature list methods → Export feature list → Molecular networking (GNPS, FBMN, IIMN, MetGem)'
    - Specify the 'Feature lists' to export (Step 5)
    - Click the 'Select' button and set the 'Filename' to a suitable file path in your computer filesystem (e.g., 'C:\Data\project_gnps' on Windows) for the export of the results. Two separate files are exported: a feature quantification table (CSV format) and an MS^2 spectral list (MGF format). If the IIN module was run (Step 17), MZmine exports an additional file (edges_msannotation.csv) containing edges connecting features annotated as multiple adducts of the same molecule (Box 7)
    - Disable the 'Merge MS/MS (experimental)' option
    - Select 'Filter rows = ONLY WITH MS2' to export only features associated with an MS2 spectrum (Step 5). If 'Ion identity networking' was performed in Step 16, select 'MS2 OR ION IDENTITY'
    - Set the 'Feature intensity' measure (i.e., peak area or height) to use in the quantification table being exported
    - Set the 'CSV export = SIMPLE'
    - Disable the 'Submit to GNPS' checkbox. This option enables the direct submission of the files to GNPS for launching a FBMN job. However, the launched job cannot be saved to your GNPS account
24. (Optional) SIRIUS export. SIRIUS is a software suite that combines automated interpretation of MS^2 spectra and machine learning to annotate each detected feature with an elemental formula, chemical structure and compound class. As described below, MZmine 3 enables the export of MS^2 spectral lists for the direct input into SIRIUS.
    - Navigate to 'Feature list methods → Export feature list → SIRIUS / CSI-FingerID'
    - Specify the 'Feature lists' to export (Step 5)
    - Click the 'Select' button and set the 'Filename' to a suitable file path in your computer filesystem (e.g., 'C:\Data\project_sirius' on Windows) for the export of the results. A MS^2 spectral list file (MGF format) will be exported
    - Disable the 'Merge MS/MS (experimental)' option
    - Set the 'm/z tolerance'. This tolerance is used to remove duplicate signals that were detected, for example, as an isotope and as a correlated feature at the same time
    - Disable all the remaining options
25. (Optional) Export for statistical analysis. Multiple downstream statistical workflows exist to further analyze results from MZmine and FBMN results. Although spreadsheet tools (e.g., Microsoft Excel) can be used for statistical analysis and data visualization, these tools normally offer only a set of basic univariate tests and suffer limitations in terms of reproducibility and scalability. Alternatively, widely used programming languages

BOX 7

Feature-based molecular networking

Molecular networking is strategy for untargeted MS data clustering, annotation propagation and visualization. Briefly, it organizes untargeted MS data into networks where ions sharing similar MS^2 spectra appear as connected nodes. FBMN expands the concept of classical molecular networking[65] by including the feature detection information (e.g., RT, peak area) in the molecular network construction[47]. Moreover, ion species of the same compound that do not connect in the network due to different fragmentation behaviour (e.g., [M+H]$^+$ versus [M+Na]$^+$) can be highlighted using IIMN, which includes the IIN information (Step 17) in the FBMN workflow[51]. Although various solutions exist[48,50,66,67], the most widely used platform to perform FBMN is the GNPS ecosystem[17]. The main advantage of using the GNPS platform is the possibility to perform spectral library search against the GNPS spectral data repositories and to use a range of other computational tools for feature annotation integrated in the GNPS ecosystem (e.g., MASST[68], network annotation propagation[69]). An example of a molecular network generated using the example dataset provided for this procedure is shown in the figure below. The piperlongumine subnetwork is highlighted. MS2 similarity edges are shown as solid gray lines and IIN edges are shown as dashed red lines. Nodes are shown as pie charts representing the intensity of each feature in the different samples (node size is proportional to the summed signal intensity). Each node is numbered and the corresponding aligned EICs are shown in the table below.



such as R, Python or MATLAB offer more extensive analysis capabilities. Here, the aligned feature table from MZmine is typically imported and formatted for the desired statistical tests to perform. For efficient statistical analysis, various scripted pipelines and GUI-based web tools are available, such as the FBMN-STATS pipeline[52]. A widely used web platform that provides a user-friendly interface for statistical analysis of metabolomics data is MetaboAnalyst[53]. As described below, MZmine 3 enables the export of aligned feature tables in the format required for uploading to MetaboAnalyst[54]. Since MetaboAnalyst requires sample information (i.e., metadata) to be included in the exported feature table, a metadata file has to be either imported or created in MZmine first (Box 8). We provide an example dataset (see the 'Required data' section) together with sample metadata (metadata_metaboanalyst.tsv) and batch (batch_metaboanalyst.xml) files to perform the untargeted feature detection and export the aligned feature table in a MetaboAnalyst-compatible format.

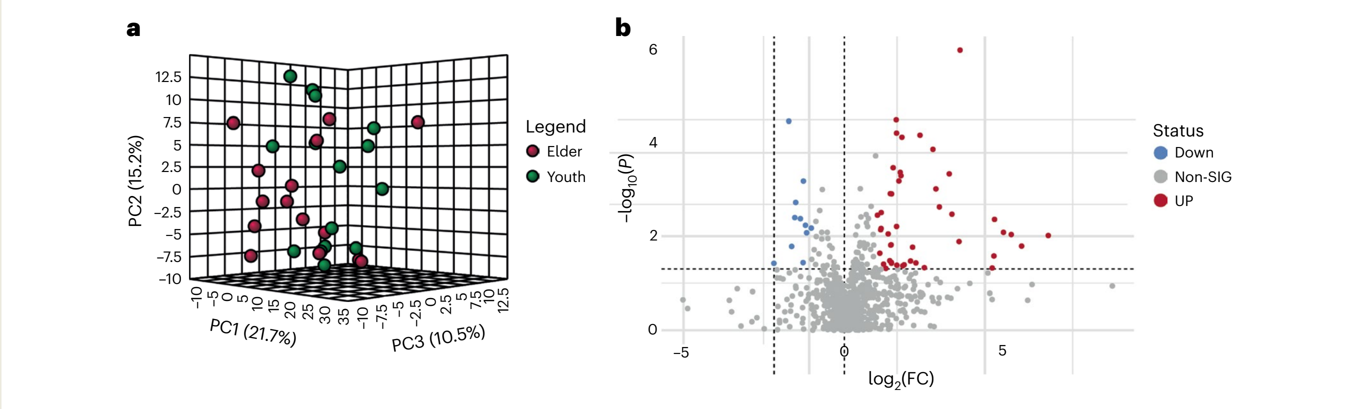
- Navigate to 'Project → Sample Metadata'. This will open a tab to add, edit or import metadata from an external file (TXT or TSV format)
- Click 'Import parameters' and select the metadata file to import. For the provided example dataset, use the metadata file 'metadata_metaboanalyst.tsv' (provided)
- Navigate to 'Feature list methods → Export feature list → Statistics Export (MetaboAnalyst)'
- Select the 'Feature lists' to export (Step 5)
- Choose the metadata grouping (e.g., 'Age')
- Click the 'Select' button and set the 'Filename' to a suitable file path in your computer filesystem (e.g., 'C:\Data\project_metaboanalyst' on Windows) for the results export. A quantification table (CSV format) including metadata information will be exported ▲ **CRITICAL STEP** For 'Import parameters' to work, raw data files must be already imported in MZmine as the software will automatically try to match the metadata file with the raw file names.

BOX 8

Downstream statistical analysis

As mentioned in Step 25, different tools and strategies can be used for downstream statistical analysis of MZmine feature tables and FBMN results; these include spreadsheet tools and programming scripts. A recent pipeline for downstream statistical analysis of MZmine and, in particular, FBMN results is 'The Hitchhiker's Guide for Statistical Analysis for Feature-Based Molecular Network' (FBMN-STATS), which contains modules for data cleanup, batch correction, as well as multivariate and univariate analysis. The code is available in multiple scripting languages (R, Python and Qiime2) as Jupyter and Google Colab notebooks. In addition to the scripting version, a GUI (FBMN-STATS-GUIde) is available[52]. Another widely used web-based platform is MetaboAnalyst, a popular tool for the postprocessing of metabolomics data, including enrichment analysis, biomarker analysis and statistical analysis. MZmine enables the export of feature lists integrated with sample metadata needed for comparative statistical analysis in MetaboAnalyst. Metadata can be either imported in MZmine from an external file (TXT or TSV format) or created directly through the GUI. Importing metadata files requires a specific template, which can be obtained by exporting a blank metadata table directly from the MZmine 'Sample metadata' module. For further information on metadata creation and exporting, see Step 25.

Here, we showcase the capabilities of MetaboAnalyst when combined with MZmine. Example plots generated with MetaboAnalyst 5.0 using the provided dataset and metadata file are shown in the figure below. Samples were split into two equal groups (i.e., youth versus elder) and the MZmine quantification table was filtered and transformed in MetaboAnalyst. More information about MetaboAnalyst 5.0 can be found in the corresponding publication[54].



Procedure 2: GC–EI–MS

▲ **CRITICAL** Refer to Fig. 6 for an overview of Procedure 2.

1. Import MS data Import the data as described in Procedure 1, Step 1.
2. Mass detection. Perform mass detection as described in Procedure 1, Step 2. A more pronounced background noise is often observed at higher GC temperature due to increased column bleeding. For this reason, we recommend applying a higher noise level towards the end of the GC run. This can be done by running the mass detection on two different RT range separately: one for the first and the other for the second part of the GC run (see, e.g., batch file 'batch_procedure-2.xml')[21].
3. EIC building with 'ADAP chromatogram builder'. Perform EIC building step as described in Procedure 1, Step 4.
4. (Optional) Chromatogram smoothing. Perform the EIC smoothing step as described in Procedure 1, Step 5.
5. EIC resolving with the ADAP resolver. As explained in Procedure 1, Step 6, the EIC traces built in the previous steps might contain multiple chromatographic peaks that need to be resolved into individual features. Although various EIC resolving algorithms are available, we recommend using the ADAP resolver module when processing GC–EI–MS data. In contrast to the local minimum resolver (described in Procedure 1, Step 6), which resolves EIC traces based on local minima, the ADAP resolver uses the continuous wavelet transform algorithm to detect peaks in EIC traces[55]. A detailed description of this module is provided at https://mzmine.org/mzmine_documentation/module_docs/featdet_resolver_adap/adap-resolver.html.
   - Navigate to 'Feature detection → Chromatogram resolving → ADAP resolver'
   - Specify the 'Feature lists' to process (see Procedure 1, Step 5)
   - Specify how to handle the 'Original feature list'. This option determines whether to 'KEEP' in memory or 'REMOVE' the input feature lists once the processing is completed. We recommend using the 'KEEP' option during parameter optimization (see the 'Processing large datasets' section)
   - Disable the 'MS/MS scan pairing' option when processing GC–EI–MS data
   - Select 'Dimension → Retention time'
   - Set the '(S/N) threshold'. This is the minimum signal-to-noise (S/N) a feature has to exhibit after resolving to be considered valid. The S/N ratio is the ratio between the

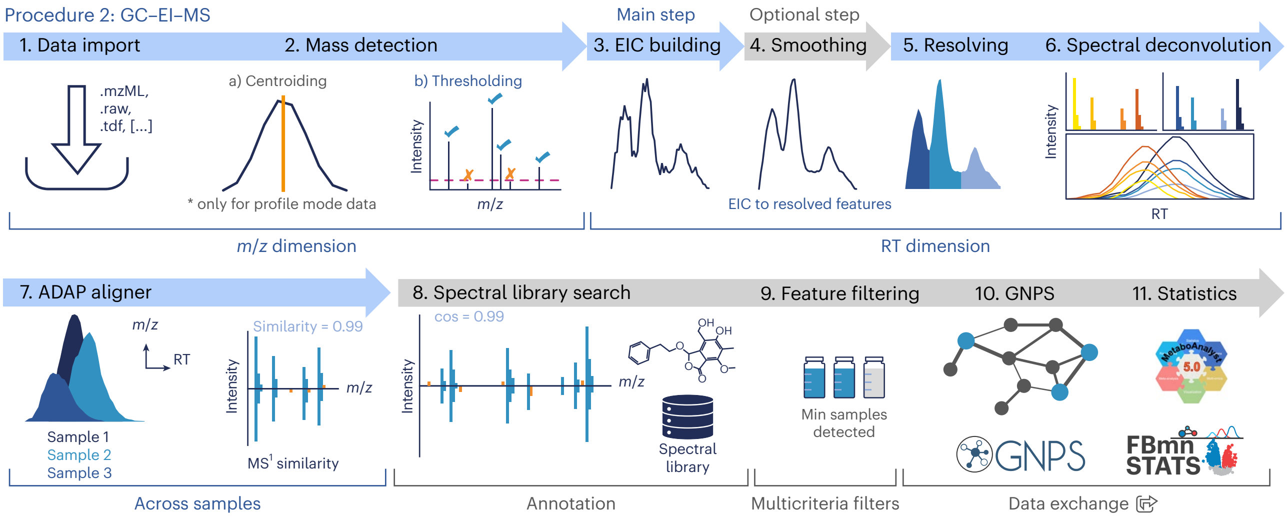


Fig. 6 | A schematic representation of the GC–EI–MS workflow described in Procedure 2. A graphical reference for each step (see numbers) summarizes steps required for the GC–EI–MS workflow in blue. Additional optional steps (in gray) may be applied to improve the input into the next steps or to provide additional annotations and results.

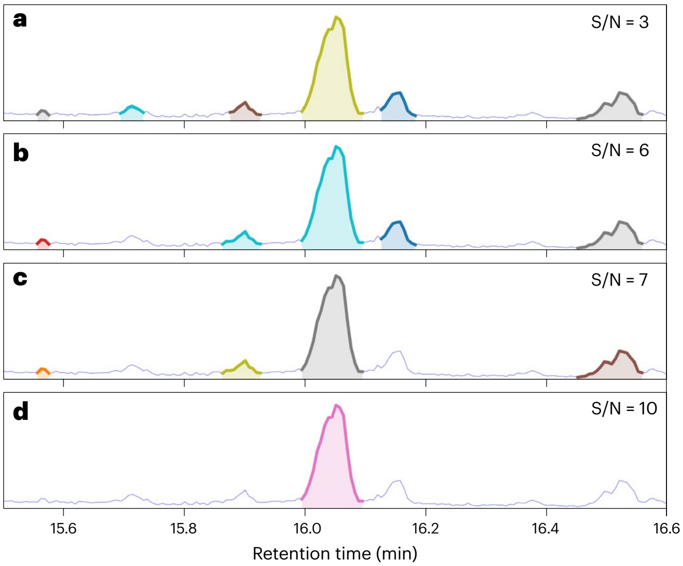


Fig. 7 | The S/N threshold parameter of ADAP resolver. Peaks produced by ADAP resolver with the parameter 'S/N estimator' set to 'Intensity Window S/N' and various values of the parameter 'S/N threshold'. **a**, S/N = 3: too many peaks are detected. **b**, S/N = 7: most peaks are detected except some peaks in close proximity to large peaks. **c**, S/N = 6: an optimal number of peaks is detected. **d**, S/N = 10: too few peaks are detected.

signal of the peak and the nearby background. High S/N ratios (e.g., ≥10) are normally associated with 'real' features whereas 'noisy' features (i.e., hardly distinguishable from the nearby background; Fig. 7) normally exhibit low S/N ratios (e.g., ≤3). We recommend a starting value of 6

- Specify the 'S/N estimator'. This is the algorithm used to estimate the S/N ratio of each peak detected during the resolving. Two options are available: the 'Intensity Window S/N' (preferred in most cases), uses the feature height as the signal level and the standard deviation of the data immediately before and after the feature as the noise level. Due to this, the S/N estimation might not be accurate for closely eluting features (Fig. 7b,c)
- Set the 'Coefficient/area threshold'. This is the minimum ratio between the highest wavelet coefficient of a peak and its area. The parameter is designed to assist the detection of low-intensity peaks when the noise level is high (e.g., as high as the real signal). This can be done by setting a lower 'S/N threshold' (e.g., 3–5) and a high 'Coefficient/area threshold' (e.g., ≥300). We recommend disabling this parameter (i.e., set it to 0) when the noise level is low
- Specify the 'min feature height'. This is the minimum signal intensity a peak needs to reach to be retained as a feature after the resolving. We recommend using the same value set for the 'Minimum absolute height' in the ADAP chromatogram building module (Step 3)
- Define the allowed 'Peak duration range'. This is the acceptable duration of a chromatographic peak to be retained as a feature after the resolving. This parameter can be used to filter out noisy features based on their overly short, or long, duration
- Set the 'RT wavelet range'. This is the range of wavelets RT width used for detecting peaks. It must be noted that this parameter is used to detect peaks, while the 'Peak duration range' is used to filter out noisy peaks based on their overly short, or long, duration
- Provide a 'Suffix' (e.g., '_ADAP-res') to name the newly created feature lists (Procedure 1, Step 5)

  ▲**CRITICAL STEP** The ADAP Resolver algorithm is very sensitive to the upper limit of this parameter, which we recommend to set to approximately half a typical peak width. The lower limit can be set to 0.

6. Spectral deconvolution. Spectral deconvolution is a crucial step in the feature detection of GC–EI–MS due to the extensive in-source fragmentation caused by EI. In fact,

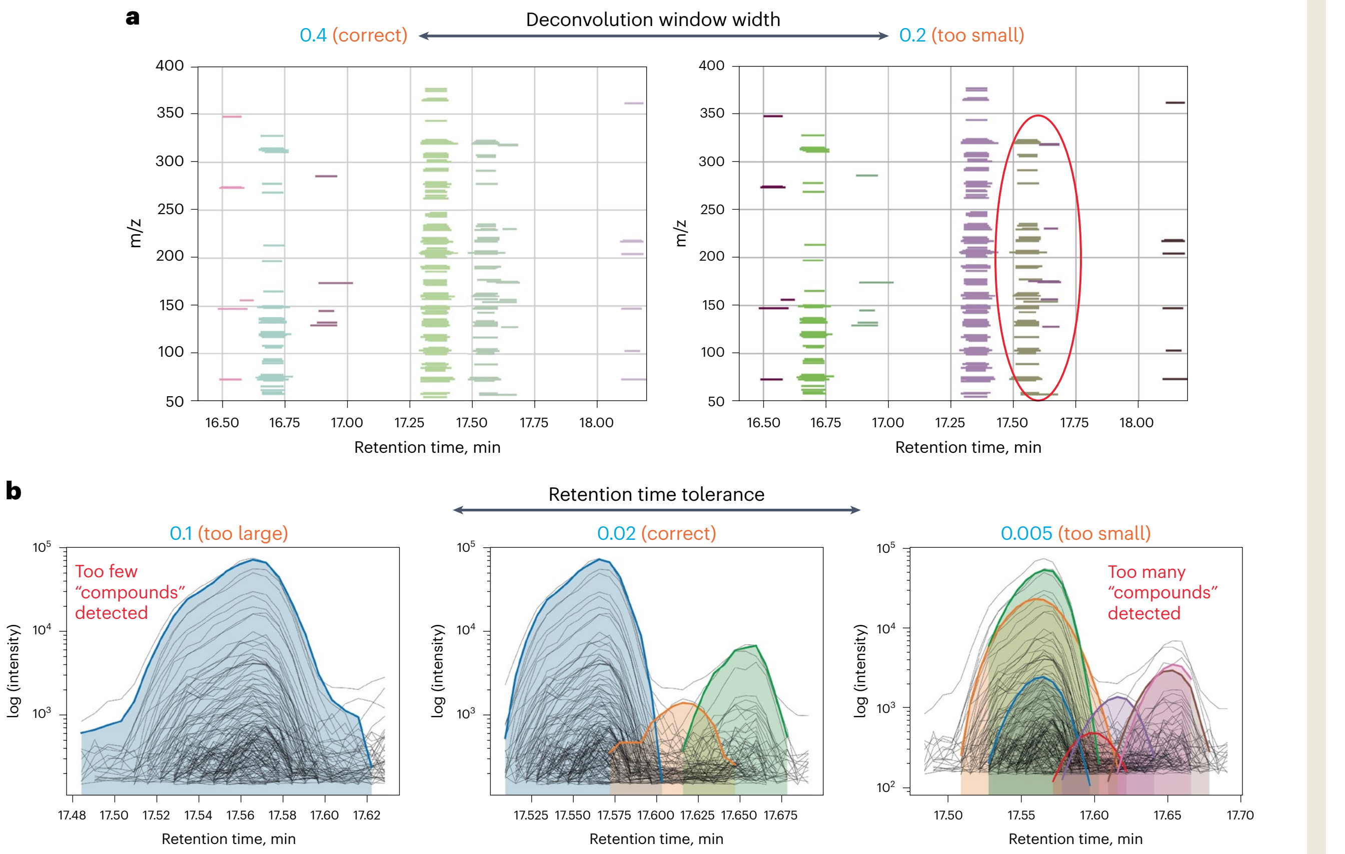
EI-produced spectra can contain fragment ions originating from different co-eluting metabolites[56]. Therefore, spectral deconvolution is necessary to computationally reconstruct fragmentation mass spectra for features not fully resolved by chromatography. The so-reconstructed spectra are then used during the feature alignment and, most importantly, in the feature annotation step. For this reason, the fine-tuning of the deconvolution parameters is crucial (Box 9).

- Navigate to 'Feature list methods → Spectral deconvolution (GC) → Multivariate curve resolution'. Although two algorithms are available, we recommend the 'Multivariate curve resolution' for its simplicity

BOX 9

Spectral deconvolution

The spectral deconvolution constructs fragmentation mass spectra of GC–EI–MS features in three steps. In the first step, the entire RT range is split into disjoint deconvolution windows, which separate the detected GC–EI–MS peaks into nonoverlapping interval (see panel a in the box figure). Each interval represents a detected GC–EI–MS peak, and the different colors denote the produced clusters. The fine-tuning of the 'deconvolution window width' parameter is crucial to achieve optimal clustering In the second step, peaks within each deconvolution window are clustered based on their chromatographic shapes to infer the number of GC–EI–MS features within each window (see colored peaks in panel b of the box figure). Their number can be adjusted by changing the 'retention time tolerance' parameter. When the RT tolerance is too large, low-intensity features can be missed. On the other hand, when the RT tolerance is too small, the algorithm will produce false features with inaccurate fragmentation mass spectra. The third step in the spectral deconvolution is building fragmentation mass spectra of the GC–EI–MS features by decomposing every EIC into a linear combination of the shapes of the features inferred by the second step (colored peaks in panel b of the box figure).



- Specify the 'Features' and 'Chromatograms' list to process. This algorithm requires both EICs constructed in Step 3 (i.e., 'Chromatograms') and peaks detected in Step 5 (i.e., 'Features'). To do so, enable the 'Specific feature lists' option from the drop-down menu, click the 'Select' button and manually select the feature lists produced by the 'ADAP Chromatogram Builder' and the 'ADAP Resolver', respectively. As an alternative, name patterns in the feature lists can be used to automatize the selection (e.g., batch mode). To do so, choose the 'Feature list name pattern' option from the drop-down menu, click the select button and type a suitable name pattern. For example, 'Chromatograms' and 'Features' can be selected by typing the '*' character, followed by the suffix used to name the feature lists created in the respective step (e.g., '*_eic' for 'Chromatograms' and '*_ADAP-res' for 'Features')
- Set the 'Deconvolution window width'. This is the maximum width of a deconvolution window. Overall, the optimal deconvolution window should be wide enough to contain co-eluting peaks within the same window, and small enough to allow a fast execution of the algorithm (Box 9)
- Specify 'Retention time tolerance' for the peak grouping. This is the maximum allowed RT deviation between any two peaks being grouped
- Specify the 'Minimum number of peaks' in a group to be considered valid. Groups with fewer peaks are discarder. This parameter is dataset-specific and mainly depends on the number of peaks detected by the *ADAP resolver* module (Step 5). Typically, values between 1 and 10 (or more) are used. By setting this parameter to 1, all groups are allowed
- (Optional) Enable the parameter 'Adjust apex retention time' if most peaks consist of few data points (e.g., four to eight). When this option is enabled, the algorithm fits a parabola into each EIC peak to determine its apex and calculate the RT[57]

  ▲**CRITICAL STEP** The 'Deconvolution window width' parameter directly affects the spectral deconvolution performance. Overly narrow deconvolution windows can cause suboptimal feature detection and/or inaccurate reconstruction of the fragmentation spectra. Overly wide deconvolution windows can substantially increase the computing time.

7. ADAP feature alignment Although two alignment algorithms are available for GC–EI–MS data, we recommend using the ADAP aligner module. This module aligns deconvoluted features across multiple samples based on the RT proximity and similarity of their reconstructed fragmentation mass spectra.
   - Navigate to 'Feature list methods → Alignment → ADAP aligner (GC)'
   - Specify the 'Feature lists' to process
   - Specify the 'Min confidence'. This is the minimum fraction of samples a feature should be detected in to be retained during the alignment. For example, if a feature is expected in at least *N* out of *M* samples, set this parameter to *N*/*M*. Set it to 0 to ignore this parameter
   - Set the 'Retention time tolerance'. This is the maximum allowed RT deviation between features being aligned
   - Specify '*m*/*z* tolerance (sample-to-sample)'. This is the maximum allowed *m*/*z* deviation between the samples for feature alignment. This is a sample-to-sample tolerance and largely depends on the performance (stability) of the MS analyzer over time
   - Set the 'Score threshold'. This is the minimum spectral similarity for features being aligned
   - Specify 'Score weight'. When multiple features fall within the defined tolerances, this parameter defines the contribution of the RT proximity and spectral similarity in calculating the total alignment score. When set to 0, only the spectrum similarity is considered. When set to 1, only the RT difference is considered. When a value between 0 and 1 is set, a weighted combination of spectral similarity and the RT difference is used. The default value of this parameter is 0.1
   - Set the 'Retention time similarity = Retention time difference (fast)'. This is the algorithm for calculating the RT similarity for the alignment. Although two options are available, we recommend using the 'Retention time difference' option

Feature annotation and data export

8. (Optional) Spectral library search. Feature annotation based on spectral matching can be performed as described in Procedure 1, Steps 18 and 19, with the following adjustments:
   - Import a library of GC–EI–MS spectra. If you are using the example dataset provided for this procedure, we recommend downloading the 'GC-MS spectra' library from the MoNA website (https://mona.fiehnlab.ucdavis.edu/downloads)
   - In the 'Spectral library search' module, set the 'Scans for matching = MS1'
   - Select 'Similarity = Composite cosine identity' algorithm. This algorithm considers the relative intensity of neighboring signals in the similarity calculation and is recommended for GC–EI–MS data. It is used to calculate the similarity between experimental and library spectra
9. (Optional) Feature list rows filter. Perform feature filtering as described in Procedure 1, Step 22. A detailed description for each filter is provided at https://mzmine.org/mzmine_documentation/module_docs/feature_list_row_filter/feature_list_rows_filter.
10. (Optional) Export for feature-based molecular networking. To export the feature quantification table and MS^2 spectral list for FBMN, a different module from the one described in Procedure 1, Step 22, should be used when dealing with GC–EI–MS data.
    - Navigate to 'Feature list methods → Export feature list → GNPS–GC–MS (with ADAP)'
    - Specify the 'Feature lists' to process
    - Click the 'Select' button and set the 'Filename' to a suitable file path in your computer filesystem for the export of the results (e.g., 'C:\Data\project_gnps)'. Two separate files are exported: a feature quantification table (CSV format) and an MS^2 spectral list (MGF format)
    - Select the 'Representative *m*/*z* = As in feature table'. This is the *m*/*z* assigned to each feature in the MGF file
    - Set the 'Feature intensity' measure (i.e., peak area or height) to use in the quantification table being exported
11. (Optional) Export for statistics (MetaboAnalyst) Export the aligned feature table for statistical analysis in MetaboAnalyst as described in Procedure 1, Step 25.

Procedure 3: MS imaging and IMS–MS imaging

▲**CRITICAL** Refer to Fig. 8 for an overview of Procedure 3.

1. Import MS data Import data as described in Procedure 1, Step 1.
   ▲**CRITICAL** Although we recommend converting MS imaging data to the .imzML format before processing, at the time of writing this format does not support IMS. Therefore, IMS–MS imaging data can only be imported into MZmine using the native Bruker format (.tdf).
2. Mass detection. Perform mass detection step as described in Procedure 1, Step 2. When processing IMS–MS imaging data, mass detection has to be run on both mobility scans and frames spectra. Two different noise levels can be applied (Procedure 1, Step 2).

Feature image detection and resolving

3. Image builder. This step builds an image for each *m*/*z* value detected over a minimum number of adjacent MS^1 scans (pixel) in the MS imaging analysis. Images matching a set of user-defined requirements (e.g., minimum number of data points and intensity) are stored as features in a feature list.
   - Navigate to 'Feature detection → Imaging → Image builder'
   - Specify the 'Raw data files' to process
   - Set the 'Scan filters'. Enable the checkbox, click the 'Show' button and set the 'MS level filter' as 'MS1, level = 1'. When fragmentation data are acquired, images can be built also for the MS^2 level

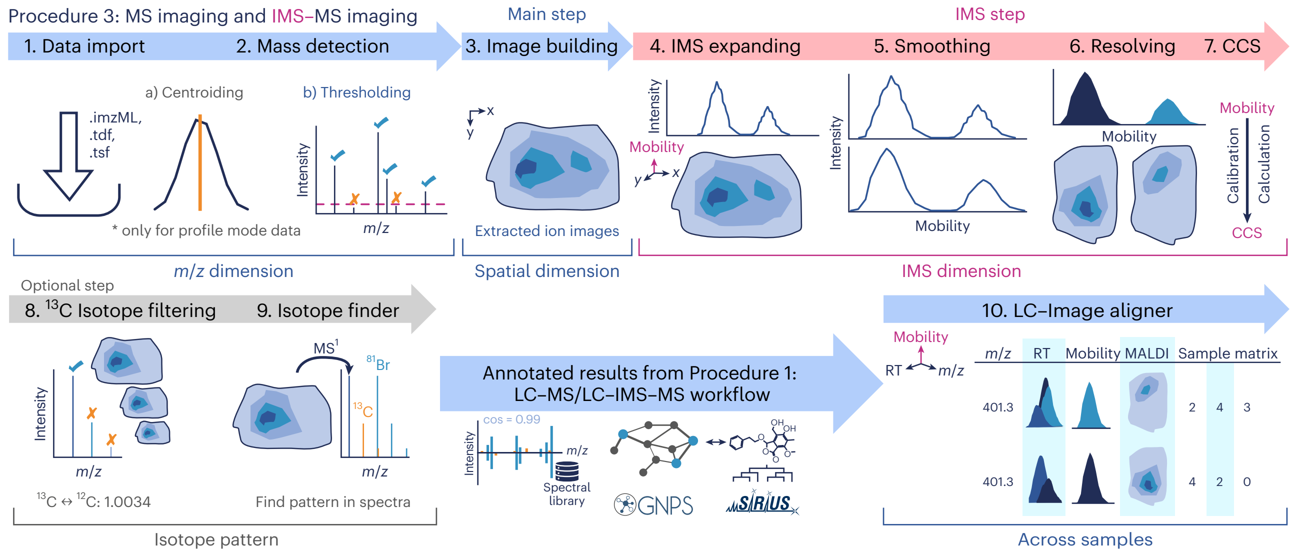


Fig. 8 | A schematic representation of the MS imaging and IMS–MS imaging workflows described in Procedure 3. A graphical reference for each step (see numbers), summarizing the steps required for the MS imaging workflow in blue. Additional steps required for IMS–MS imaging data are highlighted in magenta and optional steps for both workflows, in gray, may be applied to improve the input into the next steps or to provide additional annotations and results.

- Set the '*m*/*z* tolerance (scan-to-scan)'. As imaging experiments typically require longer analysis time than LC–MS, mass accuracy drift may occur during the measurement (especially when using TOF instruments). For this reason, we recommend using larger tolerances compared to LC–MS data processing (e.g., 0.005 *m*/*z* or 20 ppm)
- Set the 'Min consecutive scans' as the minimum number of consecutive pixels where an *m*/*z* must be detected for the corresponding image to be considered valid. This parameter mainly depends on the spatial resolution of your instrument and the size of the sample
- Set the 'Minimum absolute height' as the minimum intensity the most intense pixel in the image must exceed for the corresponding image to be considered valid
- Set the 'Minimum total signals'. This is the minimum number of pixels an ion image must contain to be considered valid
- Provide a 'Suffix' to name the newly created feature lists (e.g., '_img')

4. (Only IMS data) IMS expander. As explained in Procedure 1, Step 7, in this step the individual mobility scans are inspected to create IMS-resolved features.
   - Navigate to 'Feature detection → LC-IMS-MS → Ims expander'
   - Enable and set the '*m*/*z* tolerance'. This is the maximum allowed deviation between the *m*/*z* of the image feature (frame scans only) and the *m*/*z* signals in the individual mobility scans
   - Disable the 'Raw data instead of thresholded' parameter unless low-intensity compounds are of interest. Enabling this option increases computation cost
   - Disable 'Override default mobility bin width (scans)' to use MZmine's default binning of mobility scans
   - Enable and set the 'Maximum features per thread'. This parameter controls thread parallelization (i.e., number of images processed at the same time), which affects RAM consumption and processing time

     ▲ **CRITICAL STEP** Processing MS imaging data is much more computational demanding than LC–MS data processing. For this reason, we recommend setting a small number of 'Maximum features per thread' (e.g., 5–10). If the software crashes at this step, lower the value.
5. (Only IMS data, optional) EIM smoothing. Perform mobilogram smoothing step as described in Procedure 1, Step 8.

6. (Only IMS data) EIM resolving. Resolve the mobilogram as described in Procedure 1, Step 9.
7. (Only IMS data, optional) CCS calibration and calculation. Calibrate and calculated the CCS values as described in Procedure 1, Step 10.
8. (Optional) ^{13}C isotope filter. While applying the ^{13}C isotope filter is recommended in most cases when processing LC–MS data, extra attention should be paid when using this module on MS imaging data. This is because isobaric overlap of ^{13}C isotopic signals is much more frequent in MS imaging data. Consider applying this filter based on the resolving power of your MS instruments. The module can be used as described in Procedure 1, Step 11 with the following adjustment:
   - Set the 'RT tolerance' to a high value (e.g., 1.0×10^4). This is needed to ignore the acquisition time associated with each feature. In fact, although there is no chromatographic separation, a total ion current (TIC) is still acquired over time in MS imaging experiments
9. Isotope pattern finder. Annotate the isotopes with the isotope pattern finder module as described in Procedure 1, Step 12.

Alignment with LC–MS data

10. (Optional) LC-Image-Aligner. If the same samples were analyzed by means of MS–imaging and LC–MS, the feature detection results from both datasets can be aligned into a single feature list to increase annotation confidence[9]. This can be done using the 'LC-Image-Aligner' module, which uses an alignment scoring system similar to the Join aligner algorithm (Box 10).
    - Navigate to 'Feature list methods → Alignment → LC-Image-Aligner'
    - Specify the 'Feature lists' to process
    - Set the '*m*/*z* tolerance'. This is the maximum allowed *m*/*z* deviation between the imaging experiment and the aligned LC–(IMS–)MS feature list. This tolerance should take into account potential mass accuracy drifts occurred during the measurements and largely depends on the stability of the MS analyzer
    - Set the '*m*/*z* weight' and 'Mobility weight' (Procedure 1, Step 13). If the 'mobility tolerance' option is disabled, the 'mobility weight' will be ignored
    - (Only IMS data) Enable and set a 'mobility tolerance' if IMS–MS imaging and LC–IMS–MS data are being aligned
    - Provide a 'Feature list name' to name the newly created aligned feature list. Use '{lc}' to insert the name of the feature LC feature list

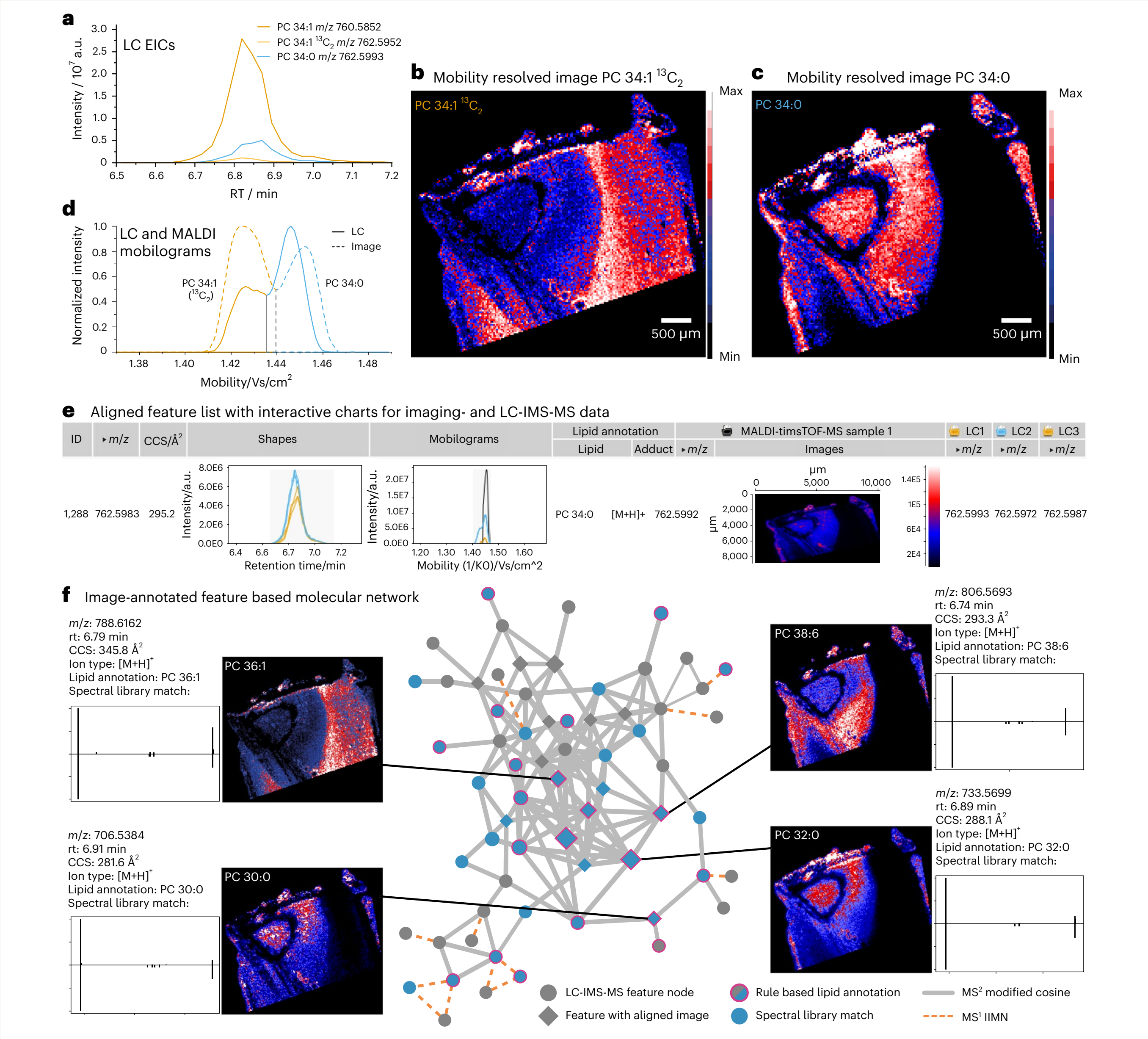
      ▲**CRITICAL STEP** The LC feature list is obtained by processing the LC–(IMS–)MS data as described in Procedure 1. The final aligned feature list should be used here.

BOX 10

Alignment of LC–IMS–MS and IMS–MS imaging data

MZmine 3 enables the alignment of LC–(IMS–)MS and (IMS–)MS imaging data into a single, aligned feature list. By doing so, MS^2-based identifications from the LC dataset can be used to increase the confidence in the annotation of imaging features. Such a workflow requires LC–MS and MS imaging data to be acquired from the same sample and processed according to Procedure 1: LC–(IMS–)MS and Procedure 3: (IMS–)MS–imaging, respectively. The obtained feature lists are then aligned using the *m*/*z* and mobility (if available) dimension. The figure below illustrates an example of alignment of features detected in LC–IMS–MS and IMS–MS imaging data. In particular, a shows EICs of PC 34:1 and isobarically overlapping PC 34:1 13C2 and PC 34:0 (mobility-resolved). b and c show mobility-resolved images of PC 34:1 13C2 and PC 34:0. d displays Overlaid mobilograms of LC–IMS–MS and MALDI–IMS–MS imaging data, solid and dashed lines, respectively, of *m*/*z* 762.5945 ± 0.01, which includes both lipids in the isobaric overlap. The solid or dashed gray line indicates where the EIMs were split in the LC or MALDI data to create the mobility-resolved features. Finally, aligned results can be visualized in a molecular network after performing FBMN (Procedure 1, Step 23) on the LC–IMS–MS features as described by Schmid et al.[9]. Figure adapted with permission from ref. 9, Springer Nature America.

(continued from previous page)



ID	▸*m/z*	CCS/Å^2	Shapes	Mobilograms	Lipid annotation: Lipid	Lipid annotation: Adduct	MALDI-timsTOF-MS sample 1: ▸*m/z*	MALDI-timsTOF-MS sample 1: Images	LC1: ▸*m/z*	LC2: ▸*m/z*	LC3: ▸*m/z*
1,288	762.5983	295.2			PC 34:0	[M+H]+	762.5992		762.5993	762.5972	762.5987

Troubleshooting

Troubleshooting advice can be found in Table 2. We also recommend checking the MZmine website, where the latest news is posted. Common issues and solutions are also described at https://mzmine.org/mzmine_documentation/troubleshooting.html.

Table 2 | Troubleshooting table

Step	Problem	Possible reason and/or solution
All	User faces an issue not described in this protocol	Go to the MZmine GitHub page and open an issue (https://github.com/mzmine/mzmine3/issues) that describes the problem. We recommend also including the MZmine log file
	The processing results obtained with the latest version of MZmine are inconsistent with those described in this protocol	This could be due to changes/updates in newer MZmine versions. To fully reproduce the results described in this protocol, download and use MZmine 3.5.0 (which was used in the preparation of this protocol)
	The GUI freezes during remote desktop connection sessions	This is a known bug of the Java GUI and will be resolved in future versions. Close and re-open MZmine solves the issue. If no errors are encountered, the processing is typically completed in the background
	The software uses all PC memory and crashes at various stages of the pipeline when processing large datasets	When processing large datasets, we recommend applying a few measures (see the 'Processing large datasets' section) to minimize the memory consumption during the most computational-demanding steps
Procedures 1–3, Step 1	Data import fails	Make sure the file format being imported is supported by MZmine (see online documentation at https://mzmine.org/mzmine_documentation/data_conversion.html). If so, ensure the original data files are not corrupted and/or no error is introduced during the file conversion
Procedure 1, Step 7	The software crashes at this step	Enable the 'Maximum features per thread' option and set a small number (e.g., 10). Gradually increase the value, if needed
Procedure 1, Step 18	Library matches are expected, but none are retrieved	Ensure the correct MS level is used for the library search (e.g., GC–EI–MS data are generally stored as 'MS level = 1'). To do so, double-click on the data files to open the 'Raw data overview'. The table in the bottom panel contains information for every scan in the data file (Extended Data Fig. 2). The MS level is displayed in the 'MSn' column
Procedure 3, Step 1	Impossible to import .imzML files	The .imzML file import used by MZmine requires an internet connection. In case the internet connection is working, ensure there is no proxy and that you are running MZmine with the administrative rights

Timing

The time required to perform MS data processing can be divided into:

- Pipeline design and optimization of the processing parameters, which may take from a few minutes to several hours, based on the user's expertise, prior experience with the software, etc.
- Actual computing time, which depends mainly on the chosen pipeline, processing parameters, and hardware resources (e.g., the number of cores and RAM memory available)

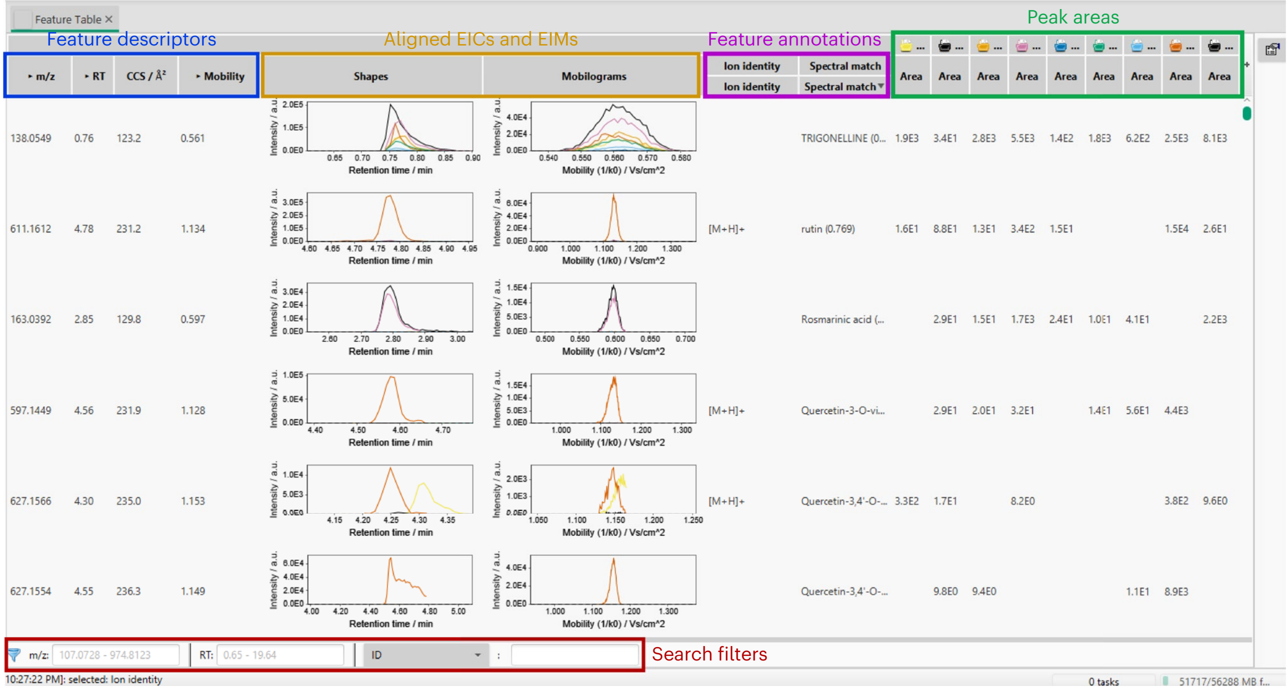
Therefore, on MS data cannot, in general, be estimated. In this protocol, we provide example datasets and corresponding batch files to help nonexperts replicating the described procedures (see "Setting up MZmine to reproduce the procedures with the 'Batch mode'" section).

We anticipate this to take up to 1 h for new MZmine users.

Anticipated results

In this protocol, we describe how to use MZmine 3 to perform untargeted feature detection and annotation on example datasets from three different MS platforms (i.e., LC–IMS–MS, GC–EI–MS and IMS–MS imaging). A batch file optimized for each example dataset is provided to reproduce the data processing described in each procedure (see the 'Required data' section). Although the same batch files cannot be used to process different datasets without adaptation, they represent a good reference for new users and a starting point for parameter optimization.

The main outputs generated during feature detection and annotation in MZmine are represented by aligned feature intensity tables (CSV format) and MS^2 spectral lists (MGF format). Aligned feature tables contain information about the abundance of each feature across the different samples, as well as other chemical annotations (e.g., isotopic pattern, adduct type, spectral library match). Feature list can be visualized and explored in MZmine (Fig. 9).



▸ m/z	▸ RT	CCS / Å²	▸ Mobility	Shapes	Mobilograms	Ion identity	Spectral match	Area	Area	Area	Area	Area	Area	Area	Area	Area
138.0549	0.76	123.2	0.561				TRIGONELLINE (0...	1.9E3	3.4E1	2.8E3	5.5E3	1.4E2	1.8E3	6.2E2	2.5E3	8.1E3
611.1612	4.78	231.2	1.134			[M+H]+	rutin (0.769)	1.6E1	8.8E1	1.3E1	3.4E2	1.5E1			1.5E4	2.6E1
163.0392	2.85	129.8	0.597				Rosmarinic acid (...		2.9E1	1.5E1	1.7E3	2.4E1	1.0E1	4.1E1		2.2E3
597.1449	4.56	231.9	1.128			[M+H]+	Quercetin-3-O-vi...		2.9E1	2.0E1	3.2E1		1.4E1	5.6E1	4.4E3	
627.1566	4.30	235.0	1.153			[M+H]+	Quercetin-3,4'-O-...	3.3E2	1.7E1		8.2E0				3.8E2	9.6E0
627.1554	4.55	236.3	1.149				Quercetin-3,4'-O-...		9.8E0	9.4E0				1.1E1	8.9E3	

m/z: 107.0728 - 974.8123 RT: 0.65 - 19.64 ID :

10:27:22 PM]: selected: Ion identity 0 tasks 51717/56288 MB f...

Fig. 9 | A screenshot of a feature list visualized in MZmine. The displayed columns can be changed by clicking the button in the top-right corner, and the search filters can be used to control the displayed features.

MS^2 spectral lists represent a summary of the fragmentation spectra associated with each feature. Both these outputs are used by other third-party tools for further downstream analysis (see the 'Data export' section and Procedure 1, Steps 23–25).

All batch files and corresponding output files (feature lists and MS^2 spectral lists) produced by processing the example datasets are available in the Supplementary Information.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

All example datasets used in this protocol are publicly available through the GNPS-MassIVE, MetaboLights and Metabolomics Workbench repositories under the following accession numbers: MSV000091634, Procedure 1, LC–IMS–MS; ST000981, Procedure 2, GC–EI–MS; MSV000090328, Procedure 3, IMS–MS imaging; MSV000091642, lipid annotation (Procedure 1, Step 21), LC–IMS–MS; MTBLS265, export for statistics (Procedure 1, Step 25), LC–MS. The FBMN results can be accessed on GNPS at https://gnps.ucsd.edu/ProteoSAFe/status.jsp?task=ffd5aee568b54d9da1f3b771c459ebe5.

Code availability

The latest release of MZmine can be downloaded from https://www.mzmine.org. The complete source code is available at https://github.com/mzmine/mzmine3/ under the MIT licence. The MZmine documentation is hosted on GitHub and available at https://www.mzmine.org/documentation.

Received: 4 July 2023; Accepted: 26 February 2024;
Published online: 20 May 2024

References

1. Alseekh, S. et al. Mass spectrometry-based metabolomics: a guide for annotation, quantification and best reporting practices. *Nat. Methods* **18**, 747–756 (2021).
2. Da Silva, R. R., Dorrestein, P. C. & Quinn, R. A. Illuminating the dark matter in metabolomics. *Proc. Natl Acad. Sci. USA* **112**, 12549–12550 (2015).
3. Müller, C., Binder, U., Bracher, F. & Giera, M. Antifungal drug testing by combining minimal inhibitory concentration testing with target identification by gas chromatography–mass spectrometry. *Nat. Protoc.* **12**, 947–963 (2017).
4. Lisec, J., Schauer, N., Kopka, J., Willmitzer, L. & Fernie, A. R. Gas chromatography mass spectrometry-based metabolite profiling in plants. *Nat. Protoc.* **1**, 387–396 (2006).
5. Chan, E. C. Y., Pasikanti, K. K. & Nicholson, J. K. Global urinary metabolic profiling procedures using gas chromatography–mass spectrometry. *Nat. Protoc.* **6**, 1483–1499 (2011).
6. Goodacre, R., Vaidyanathan, S., Dunn, W. B., Harrigan, G. G. & Kell, D. B. Metabolomics by numbers: acquiring and understanding global metabolite data. *Trends Biotechnol.* **22**, 245–252 (2004).
7. Aksenov, A. A., da Silva, R., Knight, R., Lopes, N. P. & Dorrestein, P. C. Global chemical analysis of biology by mass spectrometry. *Nat. Rev. Chem.* **1**, 1–20 (2017).
8. Kompauer, M., Heiles, S. & Spengler, B. Atmospheric pressure MALDI mass spectrometry imaging of tissues and cells at 1.4-µm lateral resolution. *Nat. Methods* **14**, 90–96 (2017).
9. Schmid, R. et al. Integrative analysis of multimodal mass spectrometry data in MZmine 3. *Nat. Biotechnol.* https://doi.org/10.1038/s41587-023-01690-2 (2023).
10. Paglia, G., Smith, A. J. & Astarita, G. Ion mobility mass spectrometry in the omics era: challenges and opportunities for metabolomics and lipidomics. *Mass Spectrom. Rev.* https://doi.org/10.1002/mas.21686 (2021).
11. Vasilopoulou, C. G. et al. Trapped ion mobility spectrometry and PASEF enable in-depth lipidomics from minimal sample amounts. *Nat. Commun.* **11**, 331 (2020).
12. Chang, H.-Y. et al. A practical guide to metabolomics software development. *Anal. Chem.* **93**, 1912–1923 (2021).
13. Tautenhahn, R., Böttcher, C. & Neumann, S. Highly sensitive feature detection for high resolution LC/MS. *BMC Bioinforma.* **9**, 504 (2008).
14. De Vijlder, T. et al. A tutorial in small molecule identification via electrospray ionization-mass spectrometry: the practical art of structural elucidation. *Mass Spectrom. Rev.* **37**, 607–629 (2018).
15. Korf, A., Jeck, V., Schmid, R., Helmer, P. O. & Hayen, H. Lipid species annotation at double bond position level with custom databases by extension of the MZmine 2 open-source software package. *Anal. Chem.* **91**, 5098–5105 (2019).
16. Dührkop, K. et al. SIRIUS 4: a rapid tool for turning tandem mass spectra into metabolite structure information. *Nat. Methods* **16**, 299–302 (2019).
17. Wang, M. et al. Sharing and community curation of mass spectrometry data with Global Natural Products Social Molecular Networking. *Nat. Biotechnol.* **34**, 828–837 (2016).
18. Pluskal, T. et al. in *Processing Metabolomics and Proteomics Data with Open Software* 232–254 (Royal Society of Chemistry, 2020).
19. Hammann, S., Korf, A., Bull, I. D., Hayen, H. & Cramp, L. J. E. Lipid profiling and analytical discrimination of seven cereals using high temperature gas chromatography coupled to high resolution quadrupole time-of-flight mass spectrometry. *Food Chem.* **282**, 27–35 (2019).
20. Simon, C. et al. Mass difference matching unfolds hidden molecular structures of dissolved organic matter. *Environ. Sci. Technol.* **56**, 11027–11040 (2022).
21. Korf, A. et al. Digging deeper—a new data mining workflow for improved processing and interpretation of high resolution GC–Q-TOF MS data in archaeological research. *Sci. Rep.* **10**, 767 (2020).
22. Brungs, C. et al. Tattoo pigment identification in inks and skin biopsies of adverse reactions by complementary elemental and molecular bioimaging with mass spectral library matching. *Anal. Chem.* **94**, 3581–3589 (2022).
23. Wolf, C. et al. Mobility-resolved broadband dissociation and parallel reaction monitoring for laser desorption/ionization–mass spectrometry—tattoo pigment identification supported by trapped ion mobility spectrometry. *Anal. Chim. Acta* **1242**, 340796 (2023).
24. Deutsch, E. W. Mass spectrometer output file format mzML. *Methods Mol. Biol.* **604**, 319–331 (2010).
25. Pedrioli, P. G. A. et al. A common open representation of mass spectrometry data and its application to proteomics research. *Nat. Biotechnol.* **22**, 1459–1466 (2004).
26. Römpp, A. et al. imzML: imaging mass spectrometry markup language: a common data format for mass spectrometry imaging. *Methods Mol. Biol.* **696**, 205–224 (2011).
27. Rew, R. & Davis, G. NetCDF: an interface for scientific data access. *IEEE Comput. Graph. Appl.* **10**, 76–82 (1990).
28. Lu, M., An, S., Wang, R., Wang, J. & Yu, C. Aird: a computation-oriented mass spectrometry data format enables a higher compression ratio and less decoding time. *BMC Bioinforma.* **23**, 35 (2022).
29. Adusumilli, R. & Mallick, P. Data conversion with ProteoWizard msConvert. *Methods Mol. Biol.* **1550**, 339–368 (2017).
30. Chambers, M. C. et al. A cross-platform toolkit for mass spectrometry and proteomics. *Nat. Biotechnol.* **30**, 918–920 (2012).
31. Smith, C. A., Want, E. J., O'Maille, G., Abagyan, R. & Siuzdak, G. XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Anal. Chem.* **78**, 779–787 (2006).
32. Röst, H. L. et al. OpenMS: a flexible open-source software platform for mass spectrometry data analysis. *Nat. Methods* **13**, 741–748 (2016).
33. Tsugawa, H. et al. MS-DIAL: data-independent MS/MS deconvolution for comprehensive metabolome analysis. *Nat. Methods* **12**, 523–526 (2015).
34. Kirkwood, K. I. et al. Utilizing Skyline to analyze lipidomics data containing liquid chromatography, ion mobility spectrometry and mass spectrometry dimensions. *Nat. Protoc.* **17**, 2415–2430 (2022).
35. Wilkinson, M. D. et al. The FAIR guiding principles for scientific data management and stewardship. *Sci. Data* **3**, 160018 (2016).
36. Barker, M. et al. Introducing the FAIR principles for research software. *Sci. Data* **9**, 622 (2022).
37. Haug, K. et al. MetaboLights—an open-access general-purpose repository for metabolomics studies and associated meta-data. *Nucleic Acids Res* **41**, D781–D786 (2013).
38. Sud, M. et al. Metabolomics Workbench: an international repository for metabolomics data and metadata, metabolite standards, protocols, tutorials and training, and analysis tools. *Nucleic Acids Res* **44**, D463–D470 (2016).
39. Meier, F. et al. Online parallel accumulation–serial fragmentation (PASEF) with a novel trapped ion mobility mass spectrometer. *Mol. Cell. Proteom.* **17**, 2534–2545 (2018).
40. Whittemore, J. C., Stokes, J. E., Laia, N. L., Price, J. M. & Suchodolski, J. S. Short and long-term effects of a synbiotic on clinical signs, the fecal microbiome, and metabolomic profiles in healthy research cats receiving clindamycin: a randomized, controlled trial. *PeerJ* **6**, e5130 (2018).
41. Matyash, V., Liebisch, G., Kurzchalia, T. V., Shevchenko, A. & Schwudke, D. Lipid extraction by methyl-tert-butyl ether for high-throughput lipidomics. *J. Lipid Res.* **49**, 1137–1146 (2008).
42. Chaleckis, R., Murakami, I., Takada, J., Kondoh, H. & Yanagida, M. Individual variability in human blood metabolites identifies age-related differences. *Proc. Natl Acad. Sci. USA* **113**, 4252–4259 (2016).
43. Smith, R., Ventura, D. & Prince, J. T. LC–MS alignment in theory and practice: a comprehensive algorithmic review. *Brief. Bioinform.* **16**, 104–117 (2015).
44. Pluskal, T., Uehara, T. & Yanagida, M. Highly accurate chemical formula prediction tool utilizing high-resolution mass spectra, MS/MS fragmentation, heuristic rules, and isotope pattern matching. *Anal. Chem.* **84**, 4396–4403 (2012).
45. Renai, L. et al. Combining feature-based molecular networking and contextual mass spectral libraries to decipher nutrimetabolomics profiles. *Metabolites* **12**, 1005 (2022).
46. Bazsó, F. L. et al. Quantitative comparison of tandem mass spectra obtained on various instruments. *J. Am. Soc. Mass Spectrom.* **27**, 1357–1365 (2016).
47. Nothias, L. F. et al. Feature-based molecular networking in the GNPS analysis environment. *Nat. Methods* **17**, 905–908 (2020).
48. Olivon, F. et al. MetGem software for the generation of molecular networks based on the *t*-SNE algorithm. *Anal. Chem.* **90**, 13900–13908 (2018).
49. Elie, N., Santerre, C. & Touboul, D. Generation of a molecular network from electron ionization mass spectrometry data by combining MZmine2 and MetGem software. *Anal. Chem.* **91**, 11489–11492 (2019).
50. Zdouc, M. M. et al. FERMO: a dashboard for streamlined rationalized prioritization of molecular features from mass spectrometry data. Preprint at *bioRxiv* https://doi.org/10.1101/2022.12.21.521422 (2022).
51. Schmid, R. et al. Ion identity molecular networking for mass spectrometry-based metabolomics in the GNPS environment. *Nat. Commun.* **12**, 3832 (2021).
52. Pakkir Shah, A. K. The hitchhiker's guide to statistical analysis of feature-based molecular networks from non-targeted metabolomics data. Preprint at *ChemRxiv* https://doi.org/10.26434/chemrxiv-2023-wwbt0 (2023).
53. Pang, Z. et al. MetaboAnalyst 5.0: narrowing the gap between raw spectra and functional insights. *Nucleic Acids Res* **49**, W388–W396 (2021).
54. Pang, Z. et al. Using MetaboAnalyst 5.0 for LC–HRMS spectra processing, multi-omics integration and covariate adjustment of global metabolomics data. *Nat. Protoc.* **17**, 1735–1761 (2022).
55. Myers, O. D., Sumner, S. J., Li, S., Barnes, S. & Du, X. One step forward for reducing false positive and false negative compound identifications from mass spectrometry metabolomics data: new algorithms for constructing extracted ion chromatograms and detecting chromatographic peaks. *Anal. Chem.* **89**, 8696–8703 (2017).
56. Du, X., Smirnov, A., Pluskal, T., Jia, W. & Sumner, S. in *Computational Methods and Data Analysis for Metabolomics* (ed. Li, S.) 25–48 (Springer, 2020).
57. Smirnov, A. et al. adap-gc 4.0: application of clustering-assisted multivariate curve resolution to spectral deconvolution of gas chromatography–mass spectrometry metabolomics data. *Anal. Chem.* **91**, 9069–9077 (2019).
58. Katajamaa, M., Miettinen, J. & Oresic, M. MZmine: toolbox for processing and visualization of mass spectrometry based molecular profile data. *Bioinformatics* **22**, 634–636 (2006).

59. Lerno, L. A. Jr, German, J. B. & Lebrilla, C. B. Method for the identification of lipid classes based on referenced Kendrick mass analysis. *Anal. Chem.* **82**, 4236–4245 (2010).
60. Sleno, L. The use of mass defect in modern mass spectrometry. *J. Mass Spectrom.* **47**, 226–236 (2012).
61. Helmer, P. O., Korf, A. & Hayen, H. Analysis of artificially oxidized cardiolipins and monolyso-cardiolipins via liquid chromatography/high-resolution mass spectrometry and Kendrick mass defect plots after hydrophilic interaction liquid chromatography based sample preparation. *Rapid Commun. Mass Spectrom.* **34**, e8566 (2020).
62. Müller, W. H. et al. Dual-polarity SALDI FT–ICR MS imaging and Kendrick mass defect data filtering for lipid analysis. *Anal. Bioanal. Chem.* **413**, 2821–2830 (2021).
63. Korf, A. et al. Three-dimensional Kendrick mass plots as a tool for graphical lipid identification. *Rapid Commun. Mass Spectrom.* **32**, 981–991 (2018).
64. Korf, A., Fouquet, T., Schmid, R., Hayen, H. & Hagenhoff, S. Expanding the Kendrick mass plot toolbox in MZmine 2 to enable rapid polymer characterization in liquid chromatography–mass spectrometry data sets. *Anal. Chem.* **92**, 628–633 (2020).
65. Aron, A. T. et al. Reproducible molecular networking of untargeted mass spectrometry data using GNPS. *Nat. Protoc.* **15**, 1954–1991 (2020).
66. Beniddir, M. A. et al. Advances in decomposing complex metabolite mixtures using substructure- and network-based computational metabolomics approaches. *Nat. Prod. Rep.* **38**, 1967–1993 (2021).
67. Dührkop, K. et al. Systematic classification of unknown metabolites using high-resolution fragmentation mass spectra. *Nat. Biotechnol.* https://doi.org/10.1038/s41587-020-0740-8 (2020).
68. Wang, M. et al. Mass spectrometry searches using MASST. *Nat. Biotechnol.* **38**, 23–26 (2020).
69. da Silva, R. R. et al. Propagating annotations of molecular networks using in silico fragmentation. *PLoS Comput. Biol.* **14**, e1006089 (2018).

Acknowledgements

T.P. is supported by the Czech Science Foundation (GA CR) grant 21-11563M and by the European Union's Horizon 2020 research and innovation programme under Marie Skłodowska-Curie grant agreement no. 891397. T.D. is supported by the European Regional Development Fund, Programme Johannes Amos Comenius project 'IOCB MSCA PF Mobility' no. CZ.02.01.01/00/22_010/0002733. C.B. is supported by the Czech Academy of Sciences Program to Support Prospective Human Resources. A.S. and X.D. are supported by the National Institutes of Health grant U01CA235507. P.C.D. is supported by R01GM107550, R03OD034493, R01DK136117 and NSF 2152526. We thank F. Rooks and M. FitzGerald for editing the manuscript.

Author contributions

Design and oversight of the project performed by S.H., T.D., A.S., R.S. and T.P. Online MZmine documentation performed by S.H., T.D., O.M., C.B., A.K., R.S. Testing the workflows described and improving the descriptions performed by S.H., T.D., O.M., C.B., A.K., J.D.S., P.S., N.D., L.-F.N., T.H., M.O., U.K., P.C.D., D.P., X.D., J.J.J.vdH. and R.S.

Competing interests

P.C.D. served as a consultant for DSM Animal Health in 2023, is an advisor and has equity in Cybele and is a co-founder, advisor and holds equity in Ometa, Enveda and Arome with prior approval by UC San Diego. J.J.J.vdH. is a member of the Scientific Advisory Board of NAICONS Srl., Milan, Italy, and consults for Corteva Agriscience, Indianapolis, IN, USA.

Additional information

Extended data is available for this paper at https://doi.org/10.1038/s41596-024-00996-y.

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41596-024-00996-y.

Correspondence and requests for materials should be addressed to Robin Schmid or Tomáš Pluskal.

Peer review information *Nature Protocols* thanks Guowang Xu and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permissions information is available at www.nature.com/reprints.

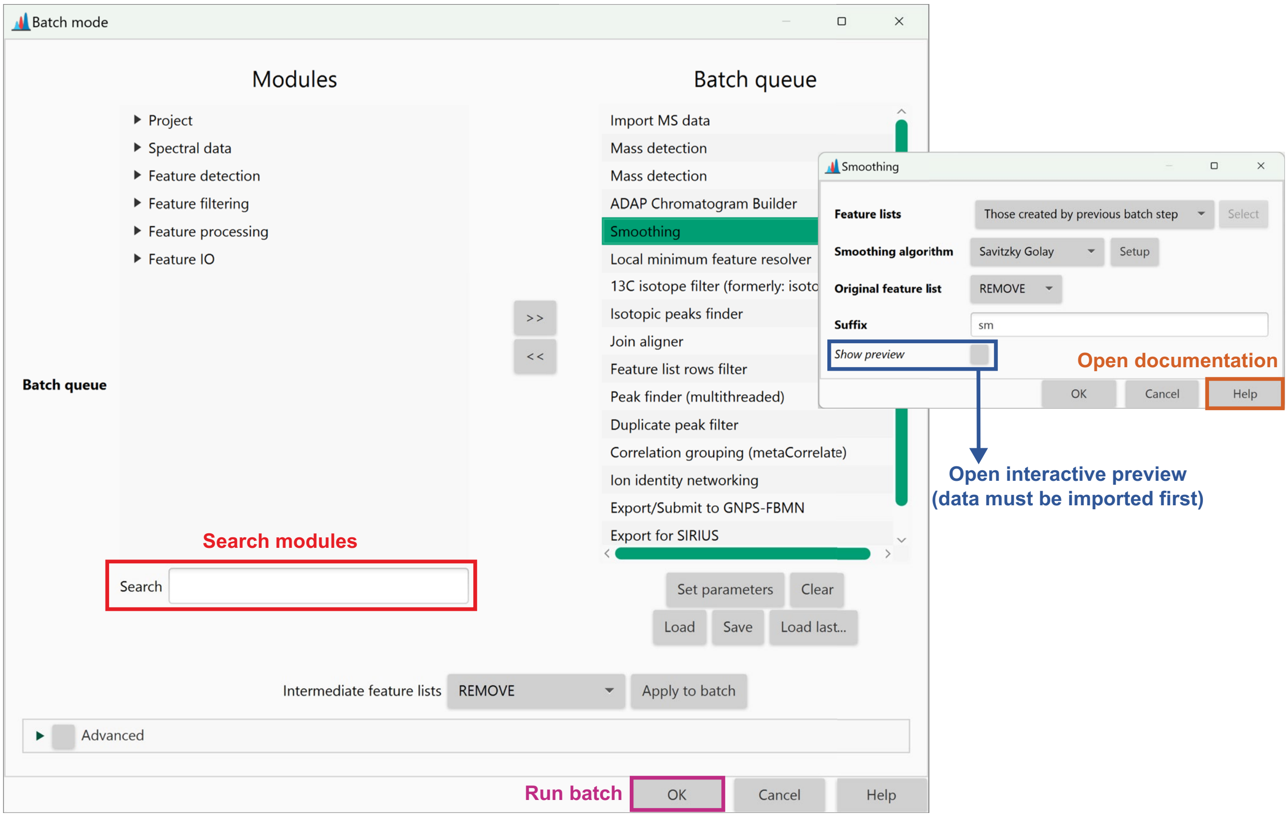
Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Related links

Key references using this protocol

Schmid, R. et al. *Nat. Biotechnol.* **41**, 447–449 (2023): https://doi.org/10.1038/s41587-023-01690-2
Schmid, R. et al. *Nat. Commun.* **12**, 3832 (2021): https://doi.org/10.1038/s41467-021-23953-9
Heuckeroth, S. et al. *Nat. Commun.* **14**, 7495 (2023): https://doi.org/10.1038/s41467-023-43298-9

[1]University of Münster, Münster, Germany. [2]Institute of Organic Chemistry and Biochemistry of the Czech Academy of Sciences, Prague, Czech Republic. [3]University of North Carolina at Charlotte, Charlotte, NC, USA. [4]First Faculty of Medicine, Charles University, Prague, Czech Republic. [5]University of Tuebingen, Tuebingen, Germany. [6]Waters Corporation, Wilmslow, UK. [7]University of Geneva, Geneva, Switzerland. [8]Université Côte d'Azur, CNRS, ICN, Nice, France. [9]Örebro University, Örebro, Sweden. [10]University of Turku and Åbo Akademi University, Turku, Finland. [11]Skaggs School of Pharmacy and Pharmaceutical Sciences, University of California San Diego, La Jolla, CA, USA. [12]University of California Riverside, Riverside, CA, USA. [13]Wageningen University & Research, Wageningen, the Netherlands. [14]University of Johannesburg, Johannesburg, South Africa. [15]These authors contributed equally: Steffen Heuckeroth, Tito Damiani.



Batch mode

Modules

- Project
- Spectral data
- Feature detection
- Feature filtering
- Feature processing
- Feature IO

Batch queue

1. Import MS data
2. Mass detection
3. Mass detection
4. ADAP Chromatogram Builder
5. Smoothing
6. Local minimum feature resolver
7. 13C isotope filter (formerly: isoto
8. Isotopic peaks finder
9. Join aligner
10. Feature list rows filter
11. Peak finder (multithreaded)
12. Duplicate peak filter
13. Correlation grouping (metaCorrelate)
14. Ion identity networking
15. Export/Submit to GNPS-FBMN
16. Export for SIRIUS

Search modules

Search

Intermediate feature lists: REMOVE

Smoothing

Feature lists: Those created by previous batch step

Smoothing algorithm: Savitzky Golay

Original feature list: REMOVE

Suffix: sm

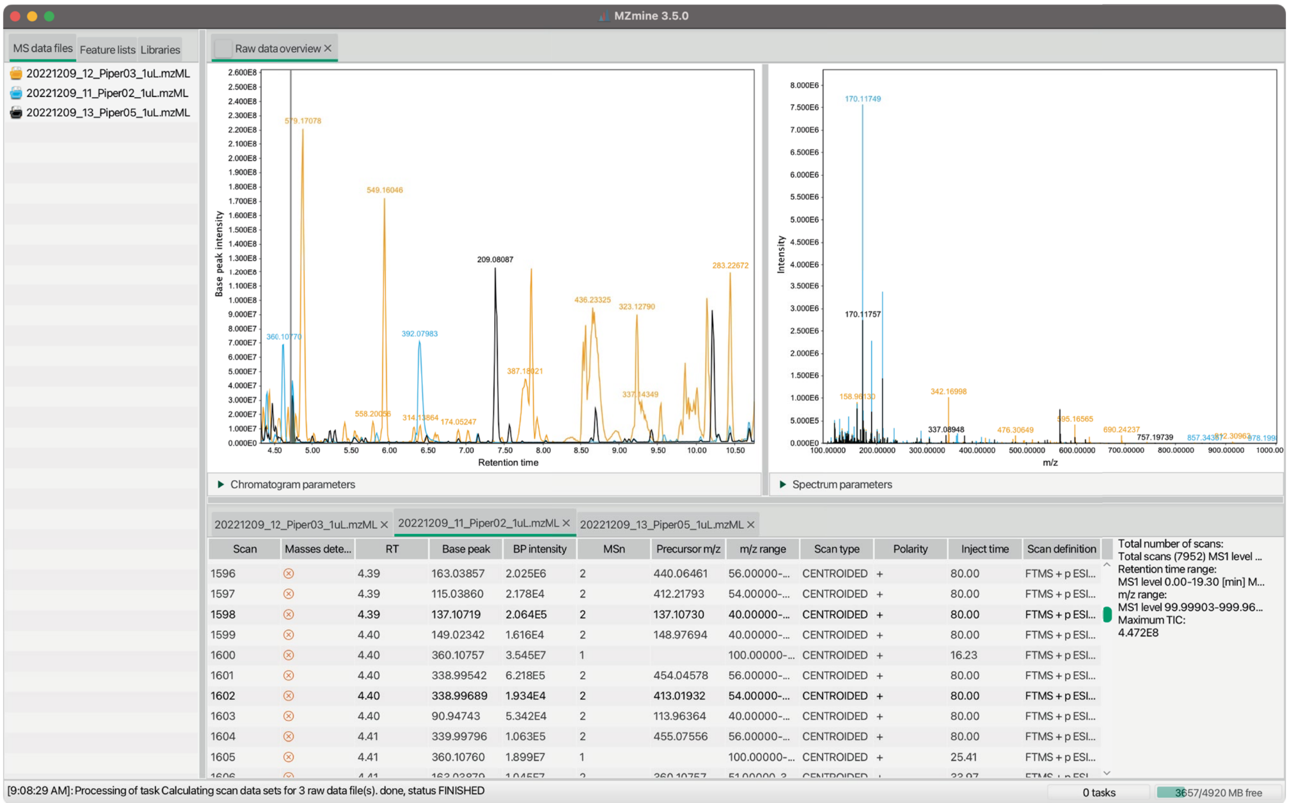
Show preview □

Open documentation

Open interactive preview (data must be imported first)

Run batch

Extended Data Fig. 1 | Screenshot of the batch mode dialogue box. The current processing steps are displayed in the 'Batch queue' panel. Additional steps can be selected from the 'Modules' panel and included using the double-arrows buttons. The current batch file can be saved using the 'Save' button whereas other batch files can be imported using the 'Load' button. Some modules offer a 'Show preview' option that can be opened by ticking the corresponding checkbox. For the preview to work, data must be already imported in MZmine. The online documentation for each processing module can be opened using the 'Help' button.

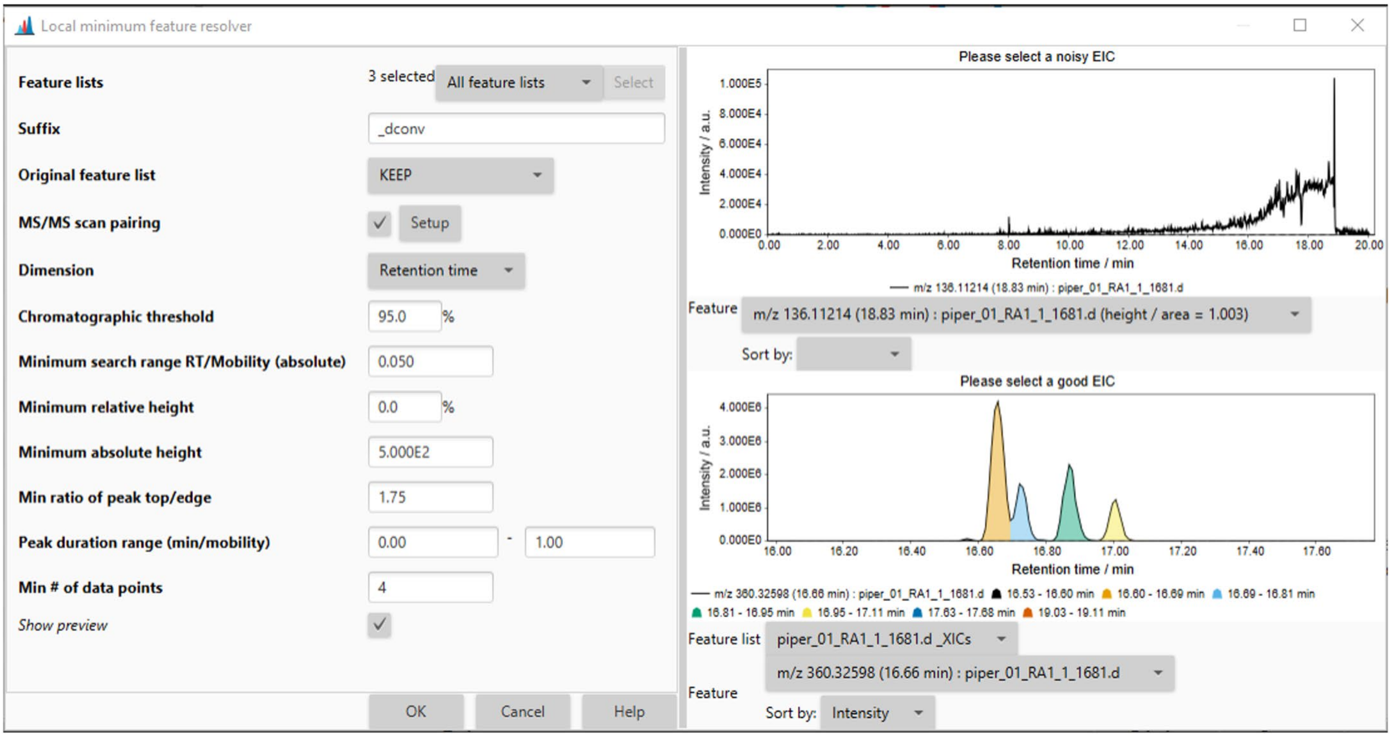


Scan	Masses dete...	RT	Base peak	BP intensity	MSn	Precursor m/z	m/z range	Scan type	Polarity	Inject time	Scan definition
1596	⊗	4.39	163.03857	2.025E6	2	440.06461	56.00000-...	CENTROIDED	+	80.00	FTMS + p ESI...
1597	⊗	4.39	115.03860	2.178E4	2	412.21793	54.00000-...	CENTROIDED	+	80.00	FTMS + p ESI...
1598	⊗	4.39	137.10719	2.064E5	2	137.10730	40.00000-...	CENTROIDED	+	80.00	FTMS + p ESI...
1599	⊗	4.40	149.02342	1.616E4	2	148.97694	40.00000-...	CENTROIDED	+	80.00	FTMS + p ESI...
1600	⊗	4.40	360.10757	3.545E7	1		100.00000-...	CENTROIDED	+	16.23	FTMS + p ESI...
1601	⊗	4.40	338.99542	6.218E5	2	454.04578	56.00000-...	CENTROIDED	+	80.00	FTMS + p ESI...
1602	⊗	4.40	338.99689	1.934E4	2	413.01932	54.00000-...	CENTROIDED	+	80.00	FTMS + p ESI...
1603	⊗	4.40	90.94743	5.342E4	2	113.96364	40.00000-...	CENTROIDED	+	80.00	FTMS + p ESI...
1604	⊗	4.41	339.99796	1.063E5	2	455.07556	56.00000-...	CENTROIDED	+	80.00	FTMS + p ESI...
1605	⊗	4.41	360.10760	1.899E7	1		100.00000-...	CENTROIDED	+	25.41	FTMS + p ESI...
1606	⊗	4.41	163.03870	1.045E7	2	360.10757	51.00000-3...	CENTROIDED	+	23.97	FTMS + p ESI

Total number of scans:
Total scans (7952) MS1 level ...
Retention time range:
MS1 level 0.00-19.30 [min] M...
m/z range:
MS1 level 99.99903-999.96...
Maximum TIC:
4.472E8

[9:08:29 AM]: Processing of task Calculating scan data sets for 3 raw data file(s). done, status FINISHED

Extended Data Fig. 2 | Screenshot of the 'Raw data overview' module. The module displays three panels: chromatogram panel (left), mass spectrum panel (right) and scan information panel (bottom panel), which contains information for every scan in the data file.



Local minimum feature resolver

Feature lists: 3 selected, All feature lists

Suffix: _dconv

Original feature list: KEEP

MS/MS scan pairing: ☑

Dimension: Retention time

Chromatographic threshold: 95.0 %

Minimum search range RT/Mobility (absolute): 0.050

Minimum relative height: 0.0 %

Minimum absolute height: 5.000E2

Min ratio of peak top/edge: 1.75

Peak duration range (min/mobility): 0.00 - 1.00

Min # of data points: 4

Show preview: ☑

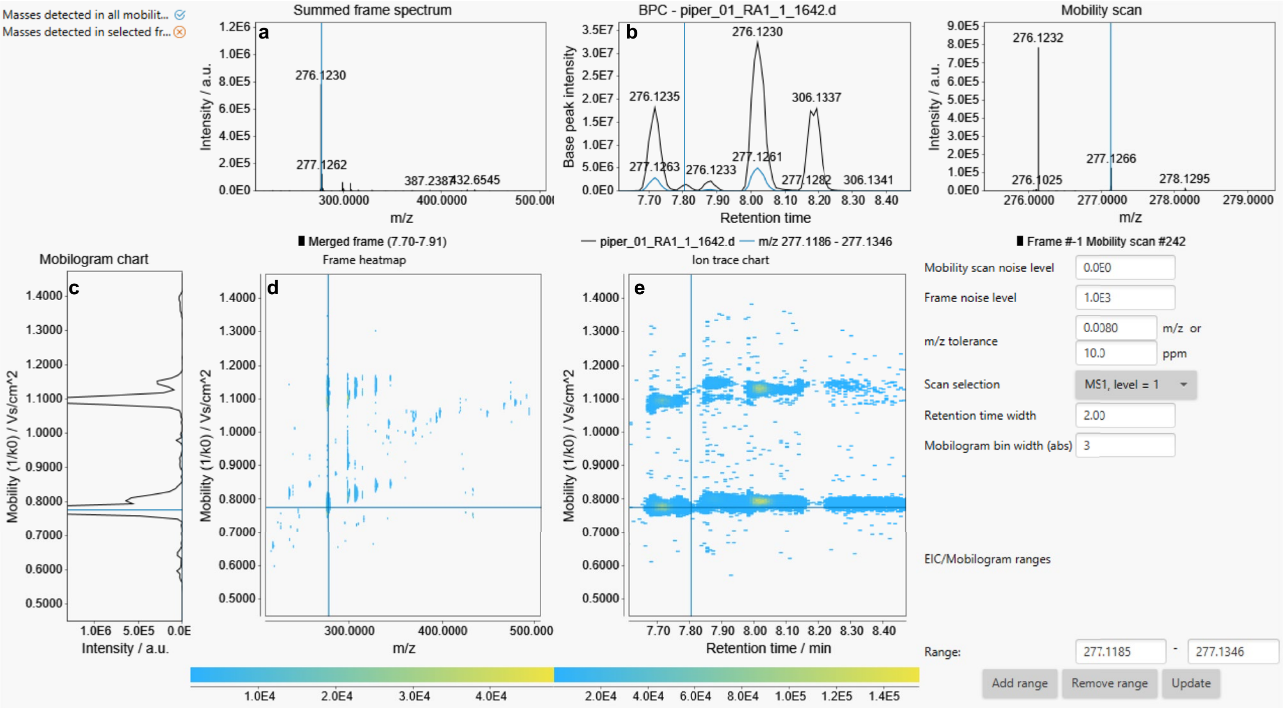
Feature: m/z 136.11214 (18.83 min) : piper_01_RA1_1_1681.d (height / area = 1.003)

Feature list: piper_01_RA1_1_1681.d _XICs

Feature: m/z 360.32598 (16.66 min) : piper_01_RA1_1_1681.d

Sort by: Intensity

Extended Data Fig. 3 | Screenshot of the interactive visualisation panel in the Local minimum resolver module. Two subpanels are present: one for ‘noisy’ and one for ‘good’ EIC traces. The goal of the parameters optimization is to ensure detection of true features while minimising ‘noisy’ peaks to be retained as features. Feature lists and EIC traces to display can be chosen from the corresponding drop-down menus. Detected features are colour-filled and resolved peaks are shown in different colours.



Mobility scan noise level 0.0E0

Frame noise level 1.0E3

m/z tolerance 0.0080 m/z or 10.0 ppm

Scan selection MS1, level = 1

Retention time width 2.00

Mobilogram bin width (abs) 3

EIC/Mobilogram ranges

Range: 277.1185 - 277.1346

Extended Data Fig. 4 | Screenshot of the 'Ion mobility raw data overview' module. a, A summed frame spectrum with a blue indicator at the selected m/z. **b**, A chromatogram plot showing the BPC (black) and EIC (blue) of the selected m/z. The blue indicator shows the RT of the selected frame. **c**, A total ion mobilogram of the selected frame. **d**, A mobility vs. m/z heatmap of the selected frame. **e**, An ion mobility trace of the selected m/z in RT and mobility dimensions.

nature portfolio

Corresponding author(s): Robin Schmid, Tomáš Pluskal

Last updated by author(s): 2024-01-30

Reporting Summary

Nature Portfolio wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Portfolio policies, see our Editorial Policies and the Editorial Policy Checklist.

Please do not complete any field with "not applicable" or n/a. Refer to the help text for what text to use if an item is not relevant to your study.
For final submission: please carefully check your responses for accuracy; you will not be able to make changes later.

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a Confirmed

● ○ The exact sample size (*n*) for each experimental group/condition, given as a discrete number and unit of measurement

● ○ A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly

● ○ The statistical test(s) used AND whether they are one- or two-sided
Only common tests should be described solely by name; describe more complex techniques in the Methods section.

● ○ A description of all covariates tested

● ○ A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons

●

○ A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)

● ○ For null hypothesis testing, the test statistic (e.g. *F*, *t*, *r*) with confidence intervals, effect sizes, degrees of freedom and *P* value noted
Give P values as exact values whenever suitable.

● ○ For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings

● ○ For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes

● ○ Estimates of effect sizes (e.g. Cohen's *d*, Pearson's *r*), indicating how they were calculated

Our web collection on statistics for biologists contains articles on many of the points above.

Software and code

Policy information about availability of computer code

Data collection	MSV000091634, ST000981, MSV000090328, MSV000091642, MTBLS265
Data analysis	MZmine 3 (v3.5.0)

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Portfolio guidelines for submitting code & software for further information.

Data

Policy information about availability of data

All manuscripts must include a data availability statement. This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A description of any restrictions on data availability
- For clinical datasets or third party data, please ensure that the statement adheres to our policy

All data used in this protocol are publicly available in open data repositories: MassIVE (MSV000091634, MSV000090328, MSV000091642), Metabolomics Workbench (ST000981), MetaboLights (MTBLS265)

Human research participants

Policy information about studies involving human research participants and Sex and Gender in Research.

Reporting on sex and gender

Population characteristics

Recruitment

Ethics oversight

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

● Life sciences ○ Behavioural & social sciences ○ Ecological, evolutionary & environmental sciences

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size

Data exclusions

Replication

Randomization

Blinding

Behavioural & social sciences study design

All studies must disclose on these points even when the disclosure is negative.

Study description

Research sample

Sampling strategy

Data collection

Timing

Data exclusions

Non-participation

Randomization

Ecological, evolutionary & environmental sciences study design

All studies must disclose on these points even when the disclosure is negative.

Study description

Research sample

Sampling strategy

Data collection

Timing and spatial scale

Data exclusions

Reproducibility

Randomization

Blinding

Did the study involve field work? ○ Yes ○ No

Field work, collection and transport

Field conditions

Location

Access & import/export

Disturbance

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

n/a	Involved in the study
●	○ Antibodies
●	○ Eukaryotic cell lines
●	○ Palaeontology and archaeology
●	○ Animals and other organisms
●	○ Clinical data
●	○ Dual use research of concern

Methods

n/a	Involved in the study
●	○ ChIP-seq
●	○ Flow cytometry
●	○ MRI-based neuroimaging

Antibodies

Antibodies used

Validation

Eukaryotic cell lines

Policy information about cell lines and Sex and Gender in Research

Cell line source(s)

Authentication

Mycoplasma contamination

Commonly misidentified lines
(See ICLAC register)

Palaeontology and Archaeology

Specimen provenance

Specimen deposition

Dating methods

□ Tick this box to confirm that the raw and calibrated dates are available in the paper or in Supplementary Information.

Ethics oversight

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Animals and other research organisms

Policy information about studies involving animals; ARRIVE guidelines recommended for reporting animal research, and Sex and Gender in Research

Laboratory animals

Wild animals

Reporting on sex

Field-collected samples

Ethics oversight

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Clinical data

Policy information about clinical studies

All manuscripts should comply with the ICMJE guidelines for publication of clinical research and a completed CONSORT checklist must be included with all submissions.

Clinical trial registration

Study protocol

Data collection

Outcomes

Dual use research of concern

Policy information about dual use research of concern

Hazards

Could the accidental, deliberate or reckless misuse of agents or technologies generated in the work, or the application of information presented in the manuscript, pose a threat to:

No Yes

○ ○ Public health

○ ○ National security

○ ○ Crops and/or livestock

○ ○ Ecosystems

○ ○ Any other significant area

Experiments of concern

Does the work involve any of these experiments of concern:

No Yes

○ ○ Demonstrate how to render a vaccine ineffective

○ ○ Confer resistance to therapeutically useful antibiotics or antiviral agents

○ ○ Enhance the virulence of a pathogen or render a nonpathogen virulent

○ ○ Increase transmissibility of a pathogen

○ ○ Alter the host range of a pathogen

○ ○ Enable evasion of diagnostic/detection modalities

○ ○ Enable the weaponization of a biological agent or toxin

○ ○ Any other potentially harmful combination of experiments and agents

ChIP-seq

Data deposition

□ Confirm that both raw and final processed data have been deposited in a public database such as GEO.

□ Confirm that you have deposited or provided access to graph files (e.g. BED files) for the called peaks.

Data access links
May remain private before publication.

Files in database submission

Genome browser session
(e.g. UCSC)

Methodology

Replicates

Sequencing depth

Antibodies

Peak calling parameters

Data quality

Software

Flow Cytometry

Plots

Confirm that:

□ The axis labels state the marker and fluorochrome used (e.g. CD4-FITC).

□ The axis scales are clearly visible. Include numbers along axes only for bottom left plot of group (a 'group' is an analysis of identical markers).

□ All plots are contour plots with outliers or pseudocolor plots.

□ A numerical value for number of cells or percentage (with statistics) is provided.

Methodology

Sample preparation

Instrument

Software

Cell population abundance

Gating strategy

□ Tick this box to confirm that a figure exemplifying the gating strategy is provided in the Supplementary Information.

Magnetic resonance imaging

Experimental design

Design type

Design specifications

Behavioral performance measures

Acquisition

Imaging type(s)

Field strength

Sequence & imaging parameters

Area of acquisition

Diffusion MRI ○ Used ○ Not used

Preprocessing

Preprocessing software

Normalization

Normalization template

Noise and artifact removal

Volume censoring

Statistical modeling & inference

Model type and settings

Effect(s) tested

Specify type of analysis: ○ Whole brain ○ ROI-based ○ Both

Statistic type for inference
(See Eklund et al. 2016)

Correction

Models & analysis

n/a	Involved in the study
○	○ Functional and/or effective connectivity
○	○ Graph analysis
○	○ Multivariate modeling or predictive analysis

Functional and/or effective connectivity

Graph analysis

Multivariate modeling and predictive analysis