The MetaRbolomics book

Jan Stanstrup, Corey D. Broeckling, Rick Helmus, Nils Hoffmann, Ewy Mathé, Thomas Naake, Luca Nicolotti, Kristian Peters, Johannes Rainer, Reza M. Salek, Tobias Schulze, Emma L. Schymanski, Michael A. Stravs, Etienne A Thévenot, Hendrik Treutler, Ralf J. M. Weber, Egon Willighagen, Michael Witting, Steffen Neumann

Contents

Preface

The term metaRbolomics has been coined for a workshop at the 2016 annual conference of the international Metabolomics society in Dublin, Ireland. On May 11, 2016, a small team of authors started to compile a review of "a few R packages" and how they can be used to do metabolomics data analysis. After some hiatus, Jan Stanstrup revived the idea and this turned into a 60+ page review (accepted in MDPI Metabolites) with more than 200 packages and more than 300 references. But the journey did not end there. In march 2018 Egon Willighagen suggested that this effort should turn into a book, and more importantly, should become a community effort, with more introduction and also include code examples. It now found a home at RforMassSpectrometry, a community website dedicated to R software for the analysis and interpretation of high throughput mass spectrometry assays, including proteomics and metabolomics experiments.

How to ...

... download data The list of packages found in the tables in this book can be downloaded from public/data/AllMetaRbolomicsTables.csv.

... add packages to the tables Go to the googlesheet and add the package. Please be careful with adding it to the right section. If it belongs in more than one table add it mulitple times as appropiate.

The package will not appear instantly in the book but only after a change is made to the book itself. You can also open an issue and request the reload such that the package shows up.

... contribute to the text There are several options. In order of convenience for the maintainer you can:

- make a pull request on the GitHub repository. You will find the text in the `rmd` folder.
- open an issue with the text you want to contribute. Clearly indicate where the text belongs.
- Send your contribution by email to jst(a t)nexs.ku.dk.

Remember to add yourself to author contributions.

1 Introduction

Metabolomics aims to measure, identify and (semi-)quantify a large number of metabolites in a biological system. The methods of choice are generally Nuclear Magnetic Resonance (NMR) spectroscopy or Mass Spectrometry (MS). The latter can be used directly (e.g. direct infusion MS), but is normally coupled to a separation system such as Gas Chromatography (GC-MS), Liquid Chromatography (LC-MS) or Capillary Electrophoresis (CE-MS). In order to increase the separation power multidimensional separation systems are becoming common, such as comprehensive two-dimensional GC or LC (GC×GC, LC×LC) or LC combined with ion mobility spectrometry (LC-IMS) before MS detection. Other detection techniques include Raman spectroscopy, UV/VIS (ultraviolet/visible absorbance spectrophotometric detection- typically with a Diode Array Detector (DAD)) and fluorescence. NMR also benefits from separation techniques, such as LC-MS-NMR or LC-SPE-NMR. Additionally, there are a wide variety of pulse programs commonly used in 1D and even bigger set of 2D pulse programs used in metabolomics and for metabolite identification, for a comprehensive review on this see [1]. A general introduction to metabolomics can be found in textbooks like [2–4] or online courses like [5,6].

All of these analytical platforms and methodologies generate large amounts of high dimensional and complex experimental raw data when used in a metabolomics context. The amount of data, the need for reproducible research, and the complexities of the biological problem under investigation necessitates a high degree of automation and standard workflows in the data analysis. Beside vendor software, which is usually not open, open source projects offer the possibility to work in community-driven teams, perform reproducible data analysis and to work with different types of raw data. Many tools and methods have been developed to facilitate the processing and analysis of metabolomics data; most seek to solve a specific challenge in the multi-step data processing and analysis workflow.

This review provides an overview of the metabolomics-related tools that are made available as packages (and a limited number of non-trivial, non-packaged scripts) for the statistics environment and programming language R [7]. We have included packages even if they are not anymore part of current CRAN or Bioconductor, i.e. as archived versions only. We have not included packages described in the literature if no longer available for download at all. We did include packages that are currently available, but not yet published in the scientific literature. The package descriptions have been grouped in sections according to the typical steps in the metabolomics data analysis pipeline for different analytical technologies, following the typical workflow steps from MS, NMR and UV data analysis, metabolite annotation, statistical analysis, molecular structure, network and pathway analysis and finally covering packages embracing large parts of the workflow.

1.1 Metabolomics data processing and analysis

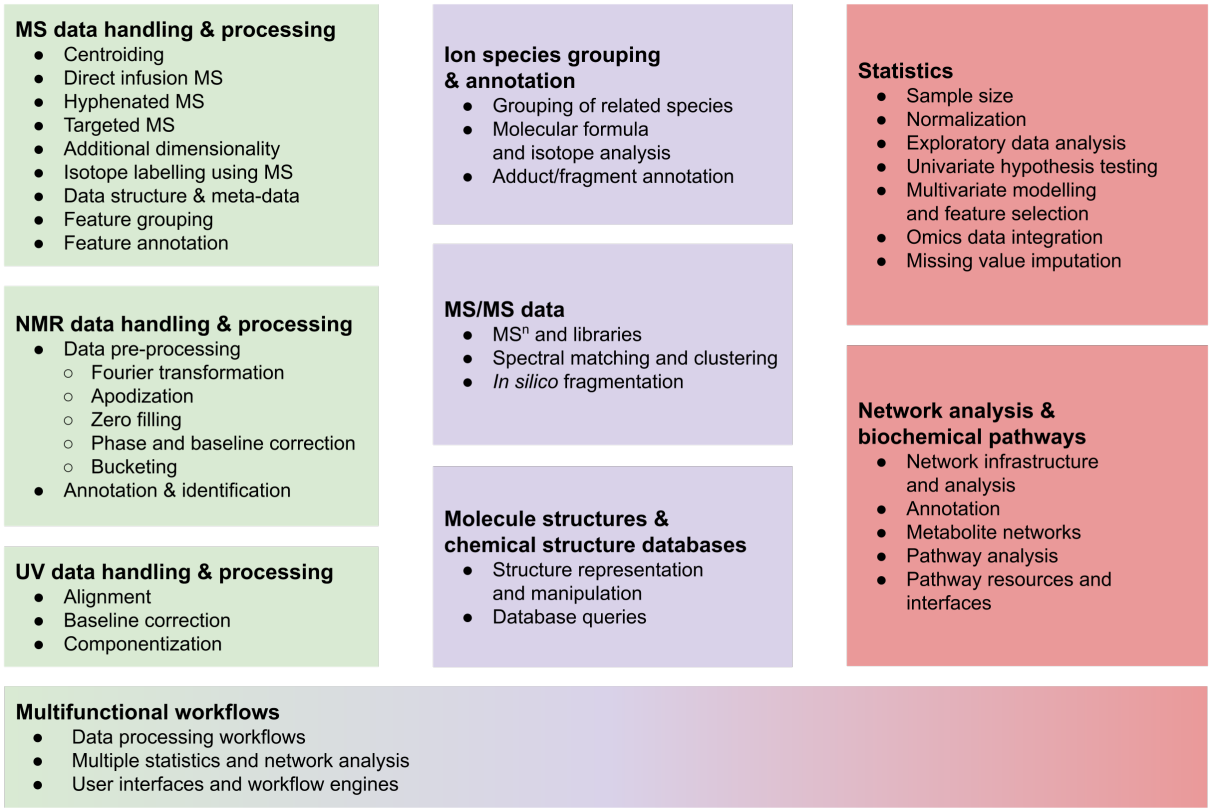


Figure 1: Overview of typical tasks in metabolomics workflows, ranging from metabolite profiling (left, green) via metabolite annotation (center, purple) to data analysis using statistics and metabolite networks (right, red).

The remainder of this section gives a broad overview and explains the typical steps, which are summarized in Figure 1, while common approaches and the available R packages are described in more detail in Section 2.

The first step for any metabolomics study is conversion from vendor formats into open data formats and pre-processing of the obtained raw data. The latter entails converting chromatographic (usually hyphenated to MS) or spectroscopic data into a data matrix suitable for data analysis. For LC-MS data this typically involves feature detection (or peak-picking) in individual samples followed by matching of features between samples. For spectroscopic data, this typically means alignment of spectra and potentially binning of the spectra into 'buckets'. The final matrix will have samples in one dimension and so-called features (unique chromatographic features or spectral bins) in the other dimension. In NMR based metabolomics, several steps are carried out to process raw time domain data to a spectrum to improve quality such as phasing and baseline correction of the spectrum. Next is alignment of peaks across spectra and samples, followed by segmenting data into bins or a peak fitting step depending on the method used.

Once the analytical data has been preprocessed, it is generally subjected to different statistical approaches to find features that are "interesting" in the context of the experimental design, e.g. differentiating diseased patients from healthy controls. In untargeted metabolomics, the selected features contain only the characteristics (e.g. m/z, retention time, chemical shift, intensity) obtained from the measurement, but not (yet) the metabolite identification or chemical structure as such. Different approaches exist for this metabolite annotation step, ranging from (usually insufficient) database lookup of exact mass (MS) or chemical shift (NMR) alone, to the use of fragmentation patterns obtained in tandem MS experiments, which can be searched against spectral databases or analysed with in silico algorithms, to spectral searching or de novo structure

elucidation using combinations of NMR experiments (often 1D and 2D).

Large parts of the metabolomics software landscape in general have been covered in reviews, recent ones include the large list of software packages [8] first described by Spicer et al. [9], and a series of annual reviews covering the list maintained by Misra and others [10–13], a review by Kannan et al. [14] and the review focussing on approaches for compound identification of LC-MS/MS data by Blaženović et al. [15]. These reviews did include software regardless of the programming environment or language used for the implementation. In section 2.9 we briefly mention how those can be accessed from within R.

This review will focus on the ecosystem of R packages for metabolomics. It provides an overview of packages to carry out one or multiple of the above mentioned steps. Some aspects are not covered in depth or not at all. For example, MS based imaging in metabolomics is an area that has unique challenges and merits its own review, and it is also beyond the scope of this review to discuss all statistical methods that could be applied in metabolomics.

1.2 The R package landscape

The core of the R language was started in 1997 and provided the basic functionality of a programming language, with some functions targeting statistics. The real power driving the popularity of R today is the huge number of contributed packages providing algorithms and data types for a myriad of application realms. Many packages have an Open Source license. This is not a phenomenon exclusive to R, but is rather a positive cultural aspect of bioinformatics software being mostly published under Open Source license terms, regardless of the implementation language. An R interpreter can be embedded in several other languages to execute R code snippets, and R code can also be executed via different workflow systems (e.g. KNIME or Galaxy, see section 2.9), which is beneficial for analysis workflows, interoperability and reuse.

These packages are typically hosted on platforms that serve as an umbrella project and are a "home" for the developer and user communities. The Comprehensive R Archive Network (CRAN) repository contains over 14,500 packages for many application areas, including some for bioinformatics and metabolomics. The "CRAN Task Views", which are manually curated resources describing available packages, books etc, help users navigate CRAN and find packages for a particular task. For metabolomics, the most relevant Task View is "*Chemometrics and Computational Physics*" [16] edited by Katharine Mullen, which includes sections on Spectroscopy, Mass Spectrometry and other tasks relevant for metabolomics applications. The Bioconductor project (BioC for short) was started by a team around Robert Gentleman in 2001 [17], and has become a vibrant community of around 1,000 contributors, working on 1,741 software, 371 data and 948 annotation packages (BioC release 3.9). In addition to a rich development infrastructure (website, developer infrastructure, version control, build farm, etc) there are regular workshops for developers and users. To enable reproducible research, BioC runs bi-annual software releases tied to a particular R release, thus ensuring and guaranteeing interoperability of packages within the same BioC release and allowing to install BioC packages from a certain release to reproduce or repeat old data analyses. On both CRAN and BioC, each package has a landing page pointing to sources, build information, binary packages and documentation. On BioC, packages are sorted (by their respective authors) into "BiocViews", where most packages are targeting genomics and gene expression analysis, and the most relevant ones for metabolomics are Cheminformatics (containing 11 packages), Lipidomics (11), SystemsBiology (66) and, of course, Metabolomics (56). Bioconductor workflows (organised as separate BioC View [18]) provide well documented examples of typical analyses. For community support, BioC maintains mailing lists, a web-based support site, slack communication channels and more. Both CRAN and BioC have a well-defined process for accepting new packages, and the respective developer guidelines (see guidelines for CRAN [19] and for BioC [20]) cover the package life-cycle from submission, updates and maintenance, to deprecation/orphaning of packages. In the case of BioC, new submissions undergo a peer review process, which also provides feedback on technical aspects and integration with the BioC landscape.

A smaller number of packages are also hosted on sites like rforge.net, r-forge.wu-wien.ac.at [21], or sourceforge.net (SF). The non-profit initiative rOpenSci [22] maintains an ecosystem around reproducible research, including staff and community-contributed R packages with additional peer review. Currently, there are no specific metabolomics related packages.

The GitHub (and also GitLab, Bitbucket) hosting services are not specific to R development, but have gained a lot of popularity due to their excellent support for participation and contribution to software projects. The maintenance of BioC packages on one of the git-based sites has become easier since the BioC team migrated to git as its version control system. A downside of these generic repository hosting sites is that there is no central point of entry, and finding packages for specific tasks is difficult compared with dedicated platforms and relies on search engines and publications. Also, while these hosting services make it easier to provide packages that do not meet BioC and CRAN requirements (e.g. rinchi due to limitations in the InChI algorithm itself), it also allows users to postpone (or circumvent entirely) the review process that helps ensure the quality of BioC contributions. In addition to generic search engines like Google.com or Bing.com, the rdrr.io is a comprehensive index of R packages and documentation from CRAN, Bioconductor, GitHub and R-Forge. Initially, its main purpose was to find R packages by name, perform full-text search in package documentation, functions and R source code. Recently, it also serves as hub to actually run R code without local installation, see Section 2.9.

1.3 Dependences and connectivity of metabolomics packages

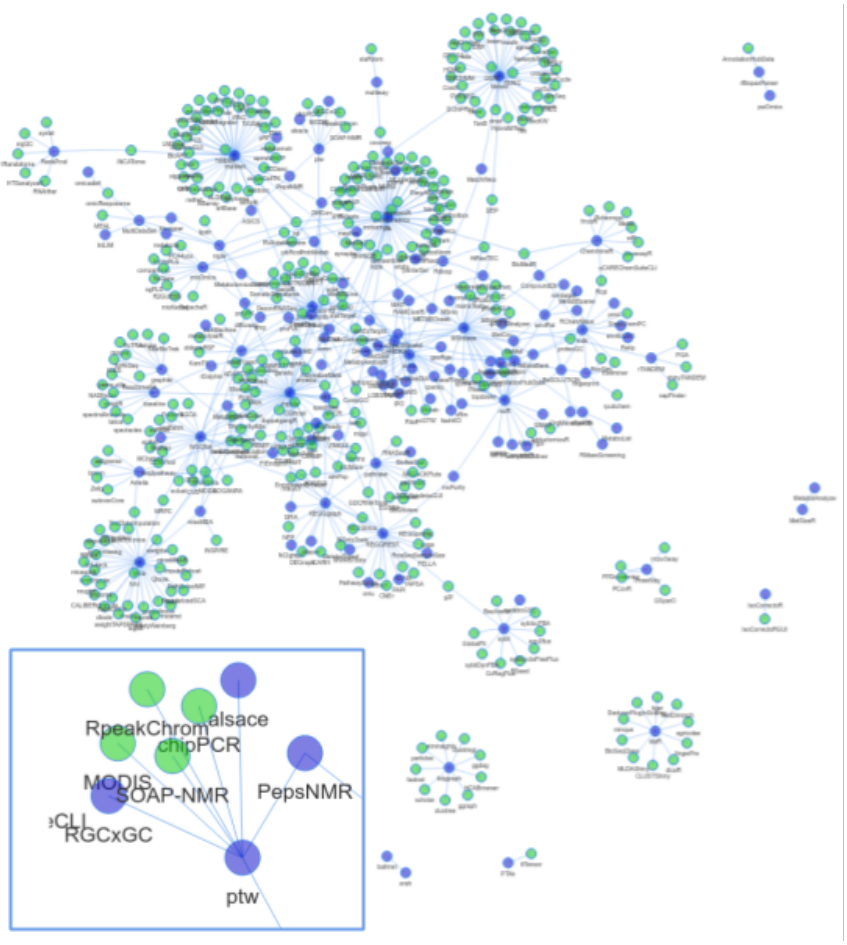


Figure 2: Dependency network of R packages. Shown in blue are packages mentioned in the review. Edges connect to packages that depend on another package, as long as they are in CRAN or BioC. Green nodes correspond to packages in CRAN or BioC not covered in the review. The inset shows the neighbourhood of the ptw package. Not shown are 1) infrastructure packages, e.g. rJava, Rcpp 2) packages from the review without reverse dependencies and 3) data packages. Some packages from the review are not in current versions of CRAN or BioC. An interactive version of this figure is also available online (rformassspectrometry.github.io/metaRbolomics-book, Appendix 2) and as supplemental file 2.

Code reuse and object inheritance can be a sign for a well-connected and interacting community. At the useR!2015 and JSM2015 conferences, A. de Vries and J. Rickert (both Microsoft, London, UK) showed the analysis of the CRAN and BioC dependency network structure [23–25]. Compared to CRAN, BioC packages had a higher connectivity: "*It seems that the Bioconductor policy encourages package authors to reuse existing material and write packages that work better together*". We repeated such an analysis [26] with the packages mentioned in this review and created a network of reverse dependencies (i.e., the set of packages that depend on these metabolomics related packages in BioC or CRAN). The resulting network is shown in Figure 2.

2 R-packages for metabolomics

This section reviews packages, relates some of those with similar functionality, and mentions how some of the packages can be used together. The sections in this review are ordered according to specific analytical approaches and the individual required steps.

2.1 Mass spectrometry data handling and (pre-)processing

For all mass spectrometers, the fundamental data generated is a mass spectrum, i.e. mass-signal intensity pairs. MS-based metabolomics data is typically acquired either as a single mass spectrum or a collection of mass spectra over time, with the time axis (retention time) defined by chromatographic (or other time domain) separation. One of the first steps in metabolomics data processing is usually the reduction of the typically large raw data produced by the instrument to a much smaller set of so called *features*, which are then subjected to downstream data analysis and interpretation. Features normally represent integrated peaks for a given mass that have been aligned across samples. Establishing these features is called *pre-processing*. The feature detection approaches and packages applicable depend on the type and characteristics of the input data. This section describes the basic data structure for some of the common analytical approaches and shows appropriate tools in R for pre-processing such data, see Table 1 for an overview of the corresponding packages.

2.1.1 Profile mode and centroided data

The mass spectra can be recorded in profile (also called continuum) mode, but are often 'centroided'. Centroiding is, in effect, a process of peak detection for a profile mode mass spectrum (hence in the m/z dimension, not in a chromatographic dimension) - a gaussian region of a continuum spectrum with a sufficiently high signal to noise ratio is integrated to give a centroided mass (a "stick" in the mass spectrum as opposed to a continuous signal) and integrated area under the curve. This results in data of reduced size - what was many m/z-intensity pairs has been reduced to a single m/z-intensity pair. Practically, this reduces the file size considerably and many data processing tools (e.g. *centWave* in xcms) require MS data that has been centroided. The centroiding can be done either during acquisition on the fly by the instrument software, or as an initial processing step. Post-acquisition centroiding can be performed during conversion of the vendor data format to open formats; typically using msconvert from ProteoWizard [27,28] which in some cases provides access to vendor centroiding algorithms or can alternatively use its own built-in centroinding method. Dedicated vendor tools can also be used, and the R packages MSnbase also provides centroiding capabilities.

2.1.2 Direct infusion mass spectrometry data

Currently, one of the highest throughput analytical approaches is direct infusion MS, where the sample is directly injected into the mass spectrometer without any chromatographic separation. This approach can be used with high mass resolution or ultra-high resolution mass spectrometers to discriminate isobaric analytes [29]. Summing or averaging these spectra generates a single mass spectrum, which is representative of that sample. Peak picking can be done using MassSpecWavelet that applies a continuous wavelet transform-based peak detection. xcms provides a wrapper for this function in the *findPeaks.MSW* function. In the Flow Injection Analysis analytical approach (FIA), the sample is transiently injected into the carrier stream flowing directly into the MS instrument. In the absence of chromatographic separation, matrix effects are a challenge for the quantification, especially in complex matrices. FIA coupled to High-Resolution Mass Spectrometry data can be processed with the proFIA workflow which provides efficient and robust peak detection and quantification.

2.1.3 Hyphenated MS and non-targeted data

Chromatographic separation before MS enables better measurement of complex samples and the ability to separate isobaric compounds. Here, the mass spectra are acquired over time as the sample components separate on the chromatography column. The mass spectrum at any given time has the same data structure as any mass spectrum - units of mass to charge ratio and time. As can be inferred from the above descriptions, chromatographically coupled mass spectrometry data is three-dimensional, with dimensions of retention time, m/z, and intensity.

For the pre-processing of LC-MS and GC-MS data, xcms is widely used. A recent paper reviewed some of the "xcms family" packages [30] though many more packages exist that build on xcms by providing tools for specialised analyses while others provide improvements of some of the xcms processing steps such as improved peak picking (xMSanalyzer, warpgroup, cosmiq). xcms itself provides a number of different algorithms for peak picking such as *matchedFilter* [31], *centWave* [32] and *massifquant* [33]. apLCMS, yamss, KPIC2 and enviPick also provide peak picking for LC-MS data independently of xcms. In cases where the alignment of the peak data of different samples is considered (e.g. in cohort studies), xcms and apLCMS include methods to group the peaks by their m/z and retention times within tolerance levels. The groups are split into sub-groups using density functions and the consensus m/z and retention time is assigned to each bin.

2.1.4 Targeted data and alternative representations of data

In addition to the most standard "spectra over time" representation of chromatographically separated MS data, there are a number of alternative ways to represent the data or simplify the data. The signal intensity for a given mass (or mass range) over chromatographic time can be represented as two equal length vectors, with retention time and intensity as units for the values of those vectors. Examples of these vector pairs include the extracted ion chromatogram (EIC, sometimes also referred to as selected or eXtracted ion chromatogram SIC, XIC), where these chromatograms represent the intensity of a given mass over (retention) time. The data thus contains no spectra but a number of SICs. Frequently this is accomplished by summarizing the raw data in a two dimensional matrix consisting of m/z and time dimensions, with each cell holding the signal intensity for that m/z and retention time range (or bin). Low mass resolution mass spectrometers often represent the data natively as a SIC and targeted data are also usually represented this way. Recent versions of xcms are also able to process such data, and additional xcms-based functionalities for analysis of targeted data can be found in the packages TargetSearch and SWATHtoMRM, while analysis of isotope labeled data can be found in the packages X13CMS, geoRge, and IsotopicLabelling. SIMAT also provides processing for targeted data and does not rely on xcms.

2.1.5 Additional dimensionality

The vast majority of data collected for metabolomics comprises of three dimensions: retention time, m/z, and intensity. However, there are more complicated analytical approaches that add additional dimensionality to the data. Two dimensional chromatography offer two separations in the chromatographic (retention time) domain. The eluent from one column is captured by retention time range and transferred to a second column where a fast orthogonal separation occurs. When coupled to a mass spectrometer, this generates four-dimensional data (m/z, first retention time, second retention time, intensity).

Ion mobility separation (IMS) is a gas phase separation method offering resolution of ions based on molecular shape. This separation occurs on timescales of tens of microseconds, which generates a nested data structure in which there are dozens to hundreds of mass spectra collected across the IMS separation time scale. One can envision this as an ion mobility 'chromatogram' - however, this chromatogram is nested within the actual chromatographic separation, thus LC-IMS-MS data is also four dimensional.

Most MS instruments offer the capability to perform selection (or filtering) of ions for fragmentation. The precursor selection can be performed through a quadrupole or ion trap, and fragmentation is often induced by collisions with an inert collision gas. Because this adds a level of mass spectrometry, it is called tandem MS, MS2 or MS/MS. Ion trap instruments can further select fragment ions and acquire MSn spectra.

There are several data independent MS/MS approaches, whereby MS/MS precursor selection is done, typically, on a scanning basis. These approaches perform precursor selection in a manner which does not depend on any feedback from the instrument control software or the MS level data. In practice, this precursor window can be either m/z or ion mobility based. The processing tools within the R universe (discussed below) are so far underdeveloped for these approaches. With the increased popularity of multidimensional separation, the need for algorithms that can fully utilize the increased separation power is also increasing.

Currently, osd provide peak picking for unit resolution GC×GC-MS. While the msPeak package provides peak picking for GC×GC-MS data, the peak picking is done on the total ion chromatogram, thus not taking advantage of the mass selectivity provided by the MS detector. It does not appear that any package for R exists that provides peak picking for GC×GC-MS, LC×LC-MS or LC-IMS-MS, similar to (or even better than) commercial tools (e.g. ChromaTOF, GC Image, ChromSquare). Also, at least in the case of GC×GC-MS, unit mass resolution still seems to be the most common use-case, even though high-resolution MS could further improve signal deconvolution and ultimately, analyte identification. Such capabilities are crucial for moving these new powerful analytical approaches into mainstream metabolomics analysis.

2.1.6 Structuring data and metadata

The result from the pre-processing is usually a matrix of abundances, rows being features (or features grouped into compounds/molecules) and columns being the samples. Within the statistical community, it is common nowadays to manipulate data matrices with rows as observations and columns as features, this difference stems from the early days, when spreadsheet programs could only handle a limited number of columns smaller than the number of *e.g.* genes. Such matrices can be easily encapsulated into an *ExpressionSet* class from Bioconductor's Biobase package [34], the more recent *SummarizedExperiment* defined in the SummarizedExperiment [35] package or the *mSet* class from the metabolomics focussed Metabase [36] package. The main advantage of such objects is their inherent support to align quantitative data along with related metadata (i.e. feature definitions/annotations as row - and sample annotations as column metadata). As an example, a *SummarizedExperiment* can be generated from xcms pre-processing results by adding the output from the *featureValues* function on the xcms result object as quantitative assay and the outputs of the *featureDefinitions* and *pData* functions as row and column annotations, respectively. Many Bioconductor packages for omics data analysis have native support for such objects (e.g., pcaMethods, STATegRa, ropls, biosigner, omicade4).

For the downstream export of mass spectrometry data from metabolomics or lipidomics experiments, the package rmzTab-M provides support for exporting quantitative and identification results backed by analytical and mass spectrometric evidence into the mzTab-M metabolomics file format [37].

Table 1: R packages for mass spectrometry data handling and (pre-)processing.

Functionalities	Package	Repo
MS data handling		
Parser for common file formats: mzXML, mzData, mzML and netCDF. Usually not used directly by the end user, but provides functions to read raw data for other packages.	mzR	BioC
Infrastructure to manipulate, process and visualise MS and proteomics data, ranging from raw to quantitative and annotated data.	MSnbase	BioC
Export and import of processed metabolomics MS results to and from the mzTab-M for metabolomics data format.	rmzTab-M	GitHub
Converts MRM-MS (.mzML) files to LC-MS style .mzML.	MRMConverteR	GitHub
Infrastructure for import, handling, representation and analysis of chromatographic MS data.	Chromatograms	GitHub
Infrastructure for import, handling, representation and analysis of MS spectra.	Spectra	GitHub
Peak picking, grouping and alignment (LC-MS focussed or general)		
Pre-processing and visualization for (LC/GC-)MS data. Includes visualization and simple statistics.	xcms	BioC
Automatic optimization of XCMS parameters based on isotopes.	IPO	BioC
Parameter tuning algorithm forXCMS, MZmine2, and other metabolomics data processingsoftware.	Autotuner	BioC

Table 1: R packages for mass spectrometry data handling and (pre-)processing. *(continued)*

Functionalities	Package	Repo
Pre-processing and visualization for (LC/GC-)MS data. Includes visualization and simple statistics.	yamss	BioC
Peak picking with XCMS and apLCMS, low intensity peak detection via replicate analyses. Multi-parameter feature extraction and data merging, sample quality and feature consistency evaluation. Annotation with METLIN and KEGG.	xMSanalyzer	SF
Pre-processing and alignment of LC-MS data without assuming a parametric peak shape model allowing maximum flexibility. It utilizes the knowledge of known metabolites, as well as robust machine learning.	apLCMS	SF
Peak detection using chromatogram subregion detection, consensus integration bound determination and Accurate missing value integration. Outputs in XCMS-compatible format.	warpgroup	GitHub
Peak picking for (LC/GC-)MS data, improving the detection of low abundance signals via a master map of m/z/RT space before peak detection. Results are XCMS-compatible.	cosmiq	BioC
m/z detection (i.e. peak-picking) for accurate mass data, collecting all data points above an intensity threshold, grouping them by m/z values and estimating representative m/z values for the clusters; extracting EICs.	AMDORAP	SF
(GC/LC)-MS data analysis for environmental science, including raw data processing, analysis of molecular isotope ratios, matrix effects, and short-chain chlorinated paraffins.	enviGCMS	CRAN
Sequential partitioning, clustering and peak detection of centroided LC-MS mass spectrometry data (.mzXML), with Interactive result and raw data plot.	enviPick	CRAN
PeakpickingwithXCMS. Groups chemically related features beforealignmentacross samples. Additional processing after alignment includes feature validation, re-integration and annotation based on custom database.	massFlowR	GitHub
KPIC2 extracts pure ion chromatograms (PIC) via K-means clustering of ions in region of interest, performs grouping and alignment, grouping of isotopic and adduct features, peak filling and Random Forest classification.	KPIC2	GitHub
Isotope labeling using MS		
Analysis of untargeted LC/MS data from stable isotope-labeling experiments. Also uses XCMS for feature detection.	geoRge	GitHub
Correction of MS and MS/MS data from stable isotope labeling (any tracer isotope) experiments for natural isotope abundance and tracer impurity. Separate GUI available in IsoCorrectoRGUI.	IsoCorrectoR	BioC
Extension of XCMS that provides support for isotopic labeling. Detection of metabolites that have been enriched with isotopic labeling.	X13CMS	NA
Analysis of isotopic patterns in isotopically-labeled MS data. Estimates the isotopic abundance of the stable isotope (either 2H or 13C) within specified compounds.	IsotopicLabelling	GitHub
Finding the dual (or multiple) isotope labeled analytes using dual labeling of metabolites for metabolome analysis (DLEMMA) approach, described in Liron [42].	Miso	CRAN

Table 1: R packages for mass spectrometry data handling and (pre-)processing. *(continued)*

Functionalities	Package	Repo
Targeted MS		
Peak picking using peak apex intensities for selected masses. Reference library matching, RT/RI conversion plus metabolite identification using multiple correlated masses. Includes GUI.	TargetSearch	BioC
Pre-processing for targeted (SIM) GC-MS data. Guided selection of appropriate fragments for the targets of interest by using an optimization algorithm based on user provided library.	SIMAT	BioC
Deconvolution of MS2 spectra obtained with wide isolation windows.	decoMS2	NA
Deconvolution of SWATH-MS experiments to MRM transitions.	SWATHtoMRM	NA
Automatic analysis of large scale MRM experiments.	MRMAnalyzer	NA
Tailors peak detection for targeted metabolites through iterative user interface. It automatically integrates peak areas for all isotopologues and outputs extracted ion chromatograms (EICs).	AssayR	GitHub
Targeted peak picking and annotation. Includes Shiny GUI.	peakPantheR	GitHub
Toolkit for working with Selective Reaction Monitoring (SRM) MS data and other variants of targeted LC-MS data.	sRm	GitHub
Deconvolution of SWATH-MS data.	DecoMetDIA	GitHub
Targeted peak picking and annotation. All functions through Shiny GUI.	TarMet	GitHub
GC-MS and GC×GC-MS		
Unsupervised data mining on GC-MS. Clustering of mass spectra to detect compound spectra. The output can be searched in NIST and ARISTO [50].	MSeasy	CRAN
Pre-processing for GC/MS, MassBank search, NIST format export.	erah	CRAN
Pre-processing using AMDIS [53, 54] for untargeted GC-MS analysis. Feature grouping across samples, improved quantification, removal of false positives, normalisation via internal standard or biomass; basic statistics.	Metab	BioC
Deconvolution of GC-MS and GC×GC-MS unit resolution data using orthogonal signal deconvolution (OSD), independent component regression (ICR) and multivariate curve resolution (MCR-ALS).	osd	CRAN
Corrects overloaded signals directly in raw data (from GC-APCI-MS) automatically by using a Gaussian or isotopic-ratio approach.	CorrectOverloadedPeaks	CRAN
Alignment of GC data. Also GC-FID or any single channel data since it works directly on peak lists.	GCalignR	CRAN
GC-MS data processing and compound annotation pipeline. Includes the building, validating, and query of in-house databases.	metaMS	BioC
Peak picking for GC×GC-MS using bayes factor and mixture probability models.	msPeak	SF
Peak alignment for GC×GC-MS data with homogeneous peaks based on mixture similarity measures.	mSPA	SF
Peak alignment for GC×GC-MS data with homogeneous and/or heterogenous peaks based on mixture similarity measures.	SWPA	SF
Chemometrics analysis GC×GC-MS: baseline correction, smoothing, COW peak alignment, multiway PCA is incorporated.	RGCxGC	CRAN

Table 1: R packages for mass spectrometry data handling and (pre-)processing. *(continued)*

Functionalities	Package	Repo
Retention time and mass spectra similarity threshold-free alignments, seamlessly integrates retention time standards for universally reproducible alignments, performs common ion filtering, and provides compatibility with multiple peak quantification methods.	R2DGC	GitHub
Flow injection / direct infusion analysis		
Pre-processing of data from Flow Injection Analysis (FIA) coupled to High-Resolution Mass Spectrometry (HRMS).	proFIA	BioC
Flow In-jection Electrospray Mass Spectrometry Processing: data processing, classification modelling and variable selection in metabolite fingerprinting	FIEmspro	GitHub
Processing Mass Spectrometry spectrum by using wavelet based algorithm. Can be used for direct infusion experiments.	MassSpecWavelet	BioC
Other		
Filtering of features originating from artifactual interference. Based on the analysis of an extract of E. coli grown in 13C-enriched media.	credential	GitHub
Wrappers for XCMS and CAMERA. Also includes matching to a spectral library and a GUI.	metaMS	BioC
Processing of peaktables from AMDIS, XCMS or ChromaTOF. Functions for plotting also provided.	flagme	BioC
Parametric Time Warping (RT correction) for both DAD and LC-MS.	ptw	CRAN
R wrapper for X!Tandem software for protein identification.	rTANDEM	BioC
Building, validation, and statistical analysis of extended assay libraries for SWATH proteomics data.	SwathXtend	BioC
Split a data set into a set of likely true metabolites and likely measurement artifacts by comparing missing rates of pooled plasma samples and biological samples.	MetProc	CRAN
Quality of LC-MS and direct infusion MS data. Generates a report that contains a comprehensive set of quality control metrics and charts.	qcrms	GitHub

2.1.7 Ion species grouping and annotation

In MS-based metabolomics, the characterisation and identification of metabolites involves several steps and approaches. After peak (feature) table generation, several tools can be used for grouping features that are postulated to originate from the same molecule. These include the widely-used CAMERA for MS1 data, as well as RAMClustR (particularly for DIA data), MetTailor, nontarget, CliqueMS and peakANOVA. Packages that support interpretation of the relationship between the ion species, including adducts, isotopes and in-source fragmentation, are InterpretMSSpectrum, CAMERA, nontarget and mzMatch [38]. See Table 2 for a summary of these packages.

Detailed reconstructed isotope patterns can be used to determine the molecular formula of potential candidates. In the case of molecular formula and isotope analysis, the m/z and intensities for a given (set of) features can be used to calculate a ranked list of possible molecular formulas, based on the accurate mass and relative isotope abundances. The Rdisop, GenFormR and enviPat packages are able to simulate and decompose isotopic patterns into molecular formula candidates. Some post processing can calculate e.g. the double bond equivalents (DBE) and similar characteristics to reduce the number of false positive assignments. Another additional source of information to improve molecular formula estimation is to include

MS/MS spectra, as used in MFAssignR, InterpretMSSpectrum or GenFormR.

A typical next step is the annotation of *m/z* with putative metabolites using accurate mass lookup, or if the molecular formula was calculated, lookup of the formula in metabolite databases. It has to be noted that annotation with accurate mass search by no means is equivalent to identification. Under the assumption that all the metabolites measured in a sample have some biochemical relation, a global annotation strategy as used in ProbMetab can help as well. Here, the individual ranked lists of formulae are re-evaluated to also maximize the number of pairs with (potential) biochemical substrate-product pairs. The masstrixR package contains several utility functions for accurate mass lookup. It enables matching of measured *m/z* values against a given database or library and can additionally perform matching based on retention times (RT) and/or collisional cross sections (CCS) if available.

Table 2: R packages for ion species grouping, annotation, molecular formula generation and accurate mass lookup.

Functionalities	Package	Repo
Molecular formula and isotope analysis		
Simulation of and decomposition of Isotopic Patterns.	Rdisop	BioC
Calculation of isotope fine patterns. Also adduct calculations and molecular formula parsing. Web version available at www.envipat.eawag.ch.	enviPat	CRAN
Molecular formula assignment, mass recalibration, signal-to-noise evaluation, and unambiguous formula selections are provided.	MFAssignR	GitHub
Uses GenForm for molecular formula generation on mass accuracy, isotope and/or MS/MS fragments, as well as performing MS/MS subformula annotation.	GenFormR	GitHub
Checking element isotopes, calculating (isotope labelled) exact monoisotopic mass,m/zvalues, mass accuracy, and inspecting possible contaminant mass peaks, examining possible adducts in ESI and MALDI.	MSbox	CRAN
MS feature grouping		
Grouping of correlated features into pseudo compound spectra using correlation across samples and similarity of peak shape. Annotation of isotopes and adducts. Works as an add-on to XCMS.	CAMERA	BioC
Grouping of features based on similarity between coelution profiles.	CliqueMS	CRAN
Cluster based feature grouping for non-targeted GC or LC-MS data.	RAMClustR	CRAN
Uses dynamic block summarisation to group features belong to the same compound. Correction for peak misalignments and isotopic pattern validation.	MetTailor	SF
Isotope & adduct peak grouping, homologous series detection.	nontarget	CRAN
Bayesian approach for grouping peaks originating from the same compound.	peakANOVA	NA
Combination of data from positive and negative ionization mode finding common molecular entities.	MScombine	CRAN
Grouping of correlated features into pseudo compound spectra using correlation across sample. Annotation of isotopes and adducts. Can work directly with the XCMS output.	Astream	NA
Navigation of high-resolution MS/MS data in a GUI based on mass spectral similarity.	MetCirc	BioC
Deconvolution of MS/MS spectra obtained with wide isolation windows.	decoMS2	NA
Ion/adduct/fragment annotation		
Bayesian probabilistic annotation.	ProbMetab	GitHub

Table 2: R packages for ion species grouping, annotation, molecular formula generation and accurate mass lookup. *(continued)*

Functionalities	Package	Repo
Isotope & adduct peak grouping, unsupervised homologous series detection.	nontarget	CRAN
Automatic interpretation of fragments and adducts in MS spectra. Molecular formula prediction based on fragmentation.	InterpretMSSpectrum	CRAN
Automated annotation using MS2 data or databases and retention time. Calculation of spectral and chemical networks.	compMS2Miner	GitHub
Screening, annotation, and putative identification of mass spectral features in lipidomics. Default databases contain ~25,000 compounds.	LOBSTAHS	BioC
Automated annotation of fragments from MS and MS2 and putative identification against simulated library fragments of ~500,000 lipid species across ~60 lipid types.	LipidMatch	GitHub
Annotation of lipid type and acyl groups on independent acquisition-mass spectrometry lipidomics based on fragmentation and intensity rules.	LipidMS	CRAN
Accurate mass and/or retention time and/or collisional cross section matching.	masstrixR	GitHub
Downloads KEGG compounds orthology data and wraps the KEGGREST package to extractgene data.	omu	CRAN
Paired mass distance analysis to find independent peaks inm/z-retention time profiles based on retention time hierarchical cluster analysis and frequency analysis of paired mass distances within retention time groups. Structure directed analysis to find potential relationship among those independent peaks. Shiny GUI included.	pmd	CRAN
Preprocessing (xcms), replicate merging, noise, blank and missingness filtering, feature grouping, annotation of known compounds, isotopic labeling analysis, annotation from KEGG or HMDB, common biotransformations and probabilistic putative metabolite annotation using MetAssign.	mzMatch	GitHub
Putative annotation of unknowns in MS1 data.	xMSAnnotator	SF

2.2 Metabolite identification with MS/MS data

The annotation of features from MS^1 experiments alone has limited specificity. Additional structural information for metabolite identification is available from tandem MS and higher-order MS^n experiments. There are different approaches, ranging from targeted MS/MS experiments and DDA to DIA (e.g. MS^E, all-ion, broad-band CID, SWATH and other vendor terms). Table 3 provides a summarized overview of R packages for these types of experiments.

2.2.1 MS/MS data handling, spectral matching and clustering

Generation of high-quality MS/MS spectral libraries and MS/MS data can be a tedious task. It involves wet lab steps of preparing solutions of reference standards as well as creating MS machine-specific acquisition methods. Several steps can be automated using different R packages presented here.

In case of targeted MS/MS, the instrument isolates specific (specified via method files) masses and fragments them is one possibility. Manually writing targeted MS/MS methods from metabolomics data can be tedious if several tens to hundreds of ions need to be fragmented. The MetShot package supports creating targeted method files for some Bruker and Waters instruments. For all other vendors optimized lists of non-overlapping peaks (RT-m/z pairs) can be generated to optimize acquisition in the lowest possible number of methods.

In Data dependent acquisition (DDA) the instrument is configured to apply a set of rules which determine which precursor ions are fragmented and MS/MS spectra acquired. DDA approaches also produce a lot of spectra for background peaks or contaminants, which are often of limited use for the purpose of metabolomics studies. Using the RMassBank package, MS1 and MS/MS data can be recalibrated and spectra cleaned of artifacts generated. After database lookup of corresponding identifiers, MassBank records are generated.

In data independent acquisition mode (DIA), the isolation windows are broader, or in some cases, all ions are fragmented, e.g. the Weizmass library [39] is based on MSE. The computational challenge for DIA data is to deconvolute the MS/MS data and assign the correct precursor ion. DIA data analysis support is currently being implemented in several R packages.

MS/MS spectra can be further processed for example by selecting a representative MS/MS spectrum among all spectra associated with a chromatographic peak or by fusing them into a *consensus* spectrum. Subsequently, spectra can be used in downstream analyses such as spectral matching or clustering. Due to the re-use of infrastructure from the MSnbase package, xcms has recently gained native support for MS/MS data handling and hence allows to extract all MS/MS spectra associated with a feature or chromatographic peak for further processing.

While DDA and DIA are convenient methods, users might miss the accuracy and full control on what is fragmented in the targeted approach. The packages rcdk, MetShot and RMassBank can be combined into a workflow (see [40]) for the generation of records to be uploaded to MS/MS spectral databases (e.g. MassBank [41]) or to be used off-line. MetShot allows the user to specify an arbitrary number RT-m/z pairs and first sorts them into non-overlapping subsets for which in a second step MS/MS methods (Bruker) or target lists (Agilent, Waters) are generated. It is possible to allow multiple collision energies in a single or separate experiment methods. rcdk was used for calculation of exact masses of adducts. MS/MS data were then acquired on a Bruker maXis plus UHR-Q-ToF-MS. After data collection each run was manually checked for data quality and processed with RMassBank.

Spectral matching of measured MS/MS data with spectral libraries is an important step in metabolite identification. Different possibilities for matching of two spectra exist, ranging from simple cosine similarity and the normalized dot product to X-Rank and proprietary algorithms. In MSnbase, different spectra can be compared. Functions for comparison include the number of common peaks, their correlation, their dot product or alternatively a custom comparison function can be supplied. In addition, it will be possible to import spectra from different file formats such as NIST msp, mgf, and Bruker library to MSnbase objects using the MSnio package. MSnbase therefore seems to be the most flexible R package for the computation of spectral similarities. Spectra are binned before comparison. The OrgMassSpecR package contains a simple

cosine spectral matching between two spectra. The two spectra are aligned with each other within a defined *m/z* error window using one spectrum as the reference. The feature-rich compMS2Miner can import msp files and uses the dot product to calculate the spectral similarity, the msPurity package can perform spectral matching using different similarity functions, and MatchWeiz implements the probabilistic X-Rank algorithm [42].

A growing number of packages, e.g. LOBSTAHS [43], LipidMatch [44] and LipidMS [45], support the annotation of lipids, see Table 2. They use a combination of lipid database lookup, spectral or selected fragment mass matching and *in silico* spectra prediction. To improve disambiguation between lipids of the same species that may only differ in their fatty acid chain composition, they usually rely on identifying specific MS/MS feature masses that are indicative of substructure fragments, such as the lipid headgroup, the headgroup with a certain fatty acid attached, or losses of fatty acid(s), and other modifications, such as oxidation. Additionally, they require certain intensity ratios between characteristic fragments of a lipid in order to identify the lipid species or subspecies.

Table 3: R packages for MS/MS data.

Functionalities	Package	Repo
MS2 and libraries		
Tools for processing raw data to database ready cleaned spectra with metadata.	RMassBank	BioC
From RT-m/z pairs (or m/z alone) creates MS2 experiment files with non-overlapping subsets of the targets. Bruker, Agilent and Waters supported.	MetShot	GitHub
Creating MS libraries from LC-MS data using XCMS/CAMERA packages. A multi-modular annotation function including X-Rank spectral scoring matches experimental data against the generated MS library.	MatchWeiz	GitHub
Assess precursor contribution to fragment spectrum acquired or anticipated isolation windows using "precursor purity" for both LC-MS(/MS) and DI-MS(/MS) data. Spectral matching against a SQLite database of library spectra.	msPurity	BioC
Automated quantification of metabolites by targeting mass spectral/retention time libraries into full scan-acquired GC-MS chromatograms.	baitmet	CRAN
MS2 spectra similarity and unsupervised statistical methods. Workflow from raw data to visualisations and is interfaceable with XCMS.	CluMSID	BioC
Import of spectra from different file formats such as NIST msp, mgf (mascot generic format), and library (Bruker) to MSnbase objects.	MSnio	GitHub
Multi-purpose mass spectrometry package. Contains many different functions .e.g. isotope pattern calculation, spectrum similarity, chromatogram plotting, reading of msp files and peptide related functions.	OrgMassSpecR	CRAN
Annotation of LC-MS data based on a database of fragments.	MetaboList	CRAN
In silico fragmentation		
In silico fragmentation of candidate structures.	MetFragR	GitHub
SOLUTIONS for High ReSOLUTION Mass Spectrometry including several functions to interact with MetFrag, developed during the SOLUTIONS project (www.solutions-project.eu).	ReSOLUTION	GitHub
Uses MetFrag and adds substructure prediction using the isotopic pattern. Can be trained on a custom dataset.	CCC	GitHub

Table 3: R packages for MS/MS data. *(continued)*

Functionalities	Package	Repo
Retention time prediction based on compound structuredescriptors. Five different machine learning algorithms are available to build models. Plotting available to explore chemical space and model quality assessment.	Retip	GitHub

2.2.2 Reading of spectral databases

NIST msp files and derived msp-like dialects are a commonly used plain text format for the representation of mass spectra. The msp format described by NIST as part of their Library Conversion Tool [46] documentation, but has many different dialects due to rather loose format definitions. R packages which support the import and export of this file format are able to both use spectral libraries for identification, as well as to create and enrich spectral libraries with new data.

There are various R packages which support the import of NIST msp files (see Table 3), but the support of different dialects varies, e.g., the NIST-like spectral libraries from RIKEN PRIME [47] cannot be parsed by some readers. In addition, none of these packages currently supports the import of additional attributes such as 'InChIKey:' or 'Collision_energy:' as used in the export of MoNA libraries [48]. In essence, most of the packages support the format shown in Listing S1 (see Supplementary File 1, 'basic NIST' in Table S1). The metaMS package supports NIST msp files as shown in Listing S2 (see Supplementary File 1, termed 'canonical NIST') and RIKEN PRIME provides a similar format with different attributes as shown in Listing S3 (see Supplementary File 1). The packages metaMS, OrgMassSpecR, enviGCMS, and TargetSearch support the export of NIST msp files. The remaining packages partially support the export of results to NIST msp files (see Table S1).

One of the most flexible packages for the handling of NIST msp files is metaMS. This package imports and exports the most attributes, although it does not entirely support generic attributes, and the export is very slow (we observed 20 min for an 8 MB file). In addition, a good library reader should also support mgf (mascot generic format) as available for download from GNPS [49] as well as other common formats such as the MassBank record format and different vendor library formats such as Bruker (.library, another msp flavour) and Agilent (.cef).

2.3 NMR data handling and (pre-)processing

NMR is another analytical technique commonly used in metabolomics research. The pre-processing steps for NMR data normally include Fourier transformation, apodization, zero filling, phase and baseline correction and finally referencing and alignment of spectra. Other steps commonly used are removing the areas without any metabolites such as the water region (from 4.7 to 4.9 ppm), as they generally contain no useful information. There are several R packages that can carry out the above tasks (see Table 4). The PepsNMR and speaq are two examples of such R-based packages. The 1D NMR spectra can then be segmented into spectral regions (also known as bins or buckets) subjected directly to statistical data analysis after a normalisation step. The size of the bins could be fixed or variable (adopted or intelligent binning) based on NMR peaks or even each data point from each peak (full data point resolution) used for data analysis. The NMRProcFlow [50] package provides a graphical and interactive interface for 1D NMR spectral processing and analysis. Additionally, it provides various spectral alignment methods with the ability to use the corresponding experimental-factor levels in a visual and interactive environment, bridging the gap between experimental design and subsequent statistical analyses. Alternatively peak picking (based on the regions of interest, ROI) can be performed and individual compounds can be identified and integrated prior to statistical analysis. Targeted profiling aims to identify and quantify specific compounds in a sample. The packages that use such approach (ROI) are rDolphin, and rNMR. The bucketed/integrated spectra are normalised to minimise the biological and technical variation. The most common methods are normalisation to a constant sum (e.g. total sum of integral/bin intensities), probabilistic quotient normalisation [51] and dry weight tissue or protein content.

NMR metabolite annotation uses either chemical shifts and multiplicity matching from existing database, such as Human Metabolome Database [52–55] (HMDB), literature experimental search or uses simulated reference library compounds [56] to match or to fit the existing biological spectra. 1D NMR data often is not sufficient for a confident assignment of the metabolite peaks [57] therefore complementary 2D spectral data acquisition are often required to confirm the assignment [58]. The only package that explicitly deals with 2D NMR is rNMR that takes a targeted approach where the user defines regions of interest to be quantified and compared. DOLPHIN, originally written in MATLAB [59], uses both 1D and 2D NMR data for targeted profiling that is also available as an R version called rDolphin. We are not aware of other R packages that handle 2D NMR data processing. Several general multiway statistical tools such as PARAFAC [60], Tucker3 [61] and MCR have been described [62] that are able to analyse 1D and 2D NMR data, see the section on statistical analysis for a list of packages available for these techniques. BATMAN uses a Bayesian model and some template information such as chemical shifts, J-couplings, multiplicity and intensity ratios derived from spectral database to automatically quantify metabolites in a targeted manner [63].

Table 4: R packages for NMR data handling, (pre-)processing and analysis.

Functionalities	Package	Repo
Data processing and Analysis		
A tool for processing of 1H NMR data including: Apodization, baseline correction, bucketing, Fourier transformation, warping and phase correction. Bruker FID can be directly imported.	PepsNMR	GitHub
Spectra alignment, peak picking based processing, Quantitative analysis and visualizations for 1D NMR.	speaq	CRAN
Interactive environment based on R-Shiny that includes a complete set of tools to process and visualize 1D NMR spectral data. Processing includes baseline correction, ppm calibration, removal of solvents and contaminants and re-alignment of chemical shifts.	NMRProcFlow	Bitbucket

Table 4: R packages for NMR data handling, (pre-)processing and analysis. *(continued)*

Functionalities	Package	Repo
TheMetaboMateR toolbox covers basic processing andstatisticalanalysis steps including; several spectral quality assessment (such as dealing with baseline distortions, water suppression to quality assessment of shimming and line width) as well as pre-processing (referencing,baselinecorrection, ...) to multivariate analysis statistics functions.	MetaboMate	GitHub
Data Analysis and Identification		
Analysis of 1D and 2D NMR spectra using a ROIs based approach. Export to MMCD or uploaded to BMRB for identification.	rNMR	NA
Pre-processing and identification in an R-based GUI for 1D NMR.	rDolphin	GitHub
Bayesian automated metabolite analyser for 1D NMR spectra. Deconvolution of NMR spectra and automate metabolite quantification. Also identification based on chemical shift lists.	BATMAN	RF
"ASICS: an automatic method for identification and quantification of metabolites in complex 1D 1H NMR spectra."	ASICS	BioC
ASICSdata: 1D NMR spectra for ASICS.	ASICSdata	BioC
shiny-based interactive NMR data import and Statistical TOtal Correlation SpectroscopY (STOCSY) analyses.	iSTATS	CRAN
NMR and integration with Genomics		
MWASTools: an integrated pipeline to perform NMR based metabolome-wide association studies (MWAS). Quality control analysis; MWAS using various models (partial correlations, generalized linear models); visualization of statistical outcomes; metabolite assignment using STOCSY; and biological interpretation of MWAS results.	MWASTools	BioC
An Integrated Suite for Genetic Mapping of Quantitative Variations of 1H NMR-Based Metabolic Profiles. mQTL-NMR provides a complete metabotype quantitative trait locus (mQTL) mapping analysis pipeline for metabolomic data.	mQTL.NMR	BioC
Handles hyperspectral data, i.e. spectra plus further information such as spatial information, time, concentrations, etc. Such data are frequently encountered in Raman, IR, NIR, UV/VIS, NMR, MS, etc.	hyperSpec	CRAN

2.4 UV data handling and (pre-)processing

Another, in metabolomics sometimes under-appreciated, analytical approach is UV absorption detection, usually coupled to an HPLC or UHPLC system. In some cases, the photo-diode array detector (DAD or PDA) is part of an LC-MS system, actually an LC-UV-MS setup. There are other detectors (e.g. fluorescence) with a different principle, but similar characteristics when it comes to the acquired data. Alignment and baseline correction are typically the first steps of preprocessing LC-UV data. Alignment can be achieved for example with the alsace or the ptw package while baseline correction can be achieved using the hyperSpec, ChemoSpec, mdatools (or the baseline packages). The alsace package provides an alternative to using all channels (wavelengths) by first finding unique components (i.e. "pure" spectra) and then performing peak-picking in these components. After alignment general multiway statistical methods like PARAFAC, simultaneous component analysis (SCA), and Tucker Factor Analysis can be applied in the same matter as feature tables would be handled. Table 5 provides an overview of the available R packages for UV data.

Table 5: R Packages for UV data handling and (pre-)processing.

Functionalities	Package	Repo
DAD		
Multivariate Curve Resolution (Alternating Least Squares) for DAD data.	alsace	GitHub
Parametric Time Warping (RT correction) for both DAD and LC-MS.	ptw	CRAN
Handles hyperspectral data, i.e. spectra plus further information such as spatial information, time, concentrations, etc. Such data are frequently encountered in Raman, IR, NIR, UV/VIS, NMR, MS, etc.	hyperSpec	CRAN
Projection based methods for preprocessing, exploring and analysis of multivariate data.	mdatools	CRAN
Collection of baseline correction algorithms, along with a GUI for optimising baseline algorithm parameters.	baseline	CRAN

2.5 Statistical analysis of metabolomics data

Following the feature detection and grouping steps outlined in the sections above, different paths to statistical analysis are available in R and Bioconductor. Once the "sample versus variable" feature matrix of molecule intensities or abundances has been generated, comprehensive statistical analyzes can be performed by using the vast range of packages provided by the R statistical software and the Bioconductor project (see Table 6), see for instance StatisticalMethod biocViews [64] and the ExperimentalDesign [65], Cluster [66], Multivariate [67], MachineLearning [68] CRAN Task Views [69]. As mentioned in the introduction we will only cover common statistical approaches used in metabolomics. Areas such as time-series analysis, clustering methods, machine learning and visualisation of high-dimensional data were dealt with in various books and literature reviews [70–78].

With regard to statistical analyses in untargeted metabolomics, two strategies can be differentiated that necessitate the use of different methods. The first strategy "metabolite profiling" is performed by most untargeted metabolomics studies. Here, a bottom-up approach is taken where sets or classes of pre-defined metabolites are studied usually in different phenotypes of the same biological species and differences in metabolites are usually related to more coarse functional or biological levels (e.g. to phenotype or to control vs. treatment in biomedical studies) [79]. Exploratory data analysis, univariate methods, hierarchical clustering (HCA), Principal Component Analysis (PCA) and Multi-Dimensional Scaling (MDS) like methods are very common in metabolite profiling approaches. Feature/variable selection is performed to find only the most significant metabolite candidates that explain the underlying research question, usually using univariate methods to target only specific metabolites that are interesting to the research question of the study [80–83].

The second strategy "metabolite fingerprinting" is commonly used in biomedicine, environmental metabolomics and eco-metabolomics to find metabolite patterns across metabolite profiles. Here, metabolites are characterised without necessarily identifying them and characterisation usually occurs from spatiotemporally coarser scales to intrinsic scales within biological species [84]. Multivariate statistical methods are used that require reduction of high-dimensional data and, thus, ordination methods are commonly applied like (Orthogonal) Partial Least Squares regression (sometimes also coupled to Discriminant Analysis) ((O)PLS(-DA)), (Linear) Discriminant Analysis ((L)DA), and (Canonical) Correspondence Analysis ((C)CA) that allow to relate sets of explanatory variables containing species traits or environmental properties (such as soil type, plant height, smoker/non-smoker, gender, etc.) to the metabolite feature matrix [77,85,86]. Other machine learning methods like Random Forests (RF), Support Vector Machines (SVM) and Neural Networks (NN or ANN) are also applicable [87]. Lately, untargeted metabolomics data is related to other 'omics using network analysis or Procrustes analysis to visualise (dis)similarities between two or more 'omics data sets [88–91].

Extracting a restricted list of features which still provide a high prediction performance (i.e., a molecular signature) is critical for biomarker validation and clinical diagnostic. Several strategies have been described for feature selection [92,93] (e.g., wrapper approaches such as Recursive Feature Elimination, Genetic Algorithms, or sparse models such as Lasso, Elastic Net, or sparse PLS). Such techniques are implemented in R packages, which also provide detailed comparisons on real datasets in terms of the stability and the size of the selected signature, the prediction performance of the final model, and the computation time [94–97].

A great number of packages is available to perform statistics on metabolomics datasets. Some of them focus on performing a number of specific tasks, such as sample size estimation, batch normalization, exploratory data analysis, univariate hypothesis testing, multivariate modeling and Omics data integration. Others, listed in the section 'Multi workflow steps' in Table 9, adopt a more comprehensive approach, providing statistics toolbox that cover different methods and functionalities.

muma is a package designed to be compatible with MS and NMR generated data. The package mainly focuses on performing statistics. It does not contain functions for data extraction and the user has to provide values arranged in a *data.frame* format. The pre-processing is limited to missing value imputation, noise filtering, variable scaling and normalization. The package also provides tools for outlier detection, univariate and multivariate analysis. Notably, the package offers a script for Statistical TOtal Correlation

SpetroscopY (STOCSY) on NMR data.

MOFA proposes tools for the integration of data coming from different omics disciplines (multi-omics). Using factor analysis it allows to calculate hidden factors that capture the biological sample variation across multi-omics datasets, thus allowing marker discovery. MOFA also provide various tools for the visualization of results. IntLIM also supports integration of other omics datasets with metabolomics data by leveraging linear modeling to identify gene-metabolite pairs whose relationship differs from one phenotype to another (e.g. positive correlation in one phenotype, negative or no correlation in another). IntLIM includes a user-friendly web interface to perform data quality control of input data, identification of phenotype-dependent gene-metabolite pairs, and interactive visualization of results. This tool is particularly useful for integrating transcriptomic and metabolomic or other omics data by generating novel hypothesis in a data-driven manner.

MetaboDiff is presented as an entry level, user friendly package for differential metabolomics analysis. The information contained in the input data (metabolomics measurements and metadata) are stored in S4 objects which are used for the downstream processing. The pre-processing consists of missing value imputation, outlier removal and data normalization, while the data analysis part offers a variety of statistical methods including tools to explore how metabolites relate to each other in sub-pathways.

MetaboAnalystR is a toolbox built over several R packages and contains more than 500 functions organised in eleven modules. The package was created to overcome the limitations of the homonymous web application, such as the possibility of creating flexible customized workflows (including xcms interoperability) and the capacity of dealing with large data sets. MetaboAnalystR functionalities cover a wide range of tools: exploratory statistical analysis, biomarker analysis, power analysis, biomarker meta-analysis, functional enrichment analysis, pathway and joint pathway analysis. Through an implementation of the *mummichog* algorithm [98], MetaboAnalystR also allows to infer pathways for from user-generated *m/z* peak-lists. Using the MetaboAnalyst knowledgebase, MetaboAnalystR provides access to metabolite set libraries, compound libraries and pathway libraries.

Table 6: R packages for statistical analysis of metabolomics data.

Functionalities	Package	Repo
Sample size		
Estimate sample sizes for metabolomics experiments, (NMR and targeted approaches supported).	MetSizeR	CRAN
Normalization		
Cross-contribution robust multiple standard normalization. Normalization using internal standards.	crmn	CRAN
Within and between batch correction of LC-MS metabolomics data using either QC samples or all samples.	batchCorr	GitLab
Normalisation for low concentration metabolites. Mixed model with simultaneous estimation of a correlation matrix.	Metabnorm	SF
A collection of data distribution normalization methods.	Normalizer	NA
Functions for drift removal and data normalisation based on: component correction, median fold change, ComBat or common PCA (CPCA).	intCor	NA
Normalisation using a singular value decomposition.	EigenMS	SF
Normalization based on RUV-random [164].	MetNorm	CRAN
SVR-based normalization and integration for large-scale metabolomics data.	MetNormalizer	GitHub
Drift correction using QC samples or all study samples.	BatchCorrMetabolomics	GitHub
Signal and Batch Correction for Mass Spectrometry	SBCMS	GitHub
Multiple fitting models to correct intra- and inter-batch effects.	MetaboQC	CRAN
Collection of functions designed to implement, assess, and choose a suitable normalization method for a given metabolomics study.	NormalizeMets	CRAN

Table 6: R packages for statistical analysis of metabolomics data. *(continued)*

Functionalities	Package	Repo
Exploratory Data Analysis		
A large number of methods available for PCA.	pcaMethods	BioC
Chemometric analysis of NMR, IR or Raman spectroscopy data. It includes functions for spectral visualisation, peak alignment, HCA, PCA and model-based clustering.	ChemoSpec	BioC
Joint analysis of MS and MS2 data, where hierarchical cluster analysis is applied to MS2 data to annotate metabolite families and principal component analysis is applied to MS data to discover regulated metabolite families.	MetFamily	GitHub
Univariate hypothesis testing		
Many methods for corrections for multiple testing.	multtest	BioC
Estimate tail area-based false discovery rates (FDR) as well as local false discovery rates (fdr) for a variety of null models (p-values, z-scores, correlation coefficients, t-scores).	fdrtool	CRAN
GUI for statistical analysis using linear mixed models to normalize data and ANOVA to test for treatment effects.	MetabR	RF
Derives stable estimates of the metabolome-wide significance level within a univariate approach based on a permutation procedure which effectively controls the maximum overall type I error rate at the level.	MWSL	GitHub
Multivariate modeling and feature selection		
Various multivariate methods to analyze metabolomics datasets. Main methods include PCA, Partial Least Squares regression (PLS), and extensions to the PLS like PLS Discriminant Analysis PLS-DA and the orthogonal variants OPLS(-DA).	ropls	BioC
Package for performing Partial Least Squares regression (PLS).	pls	CRAN
PPCA, PPCCA, MPPCA.	MetabolAnalyze	CRAN
General framework for building regression and classification models.	caret	CRAN
ASCA, figure of merit (FoM), PCA, Goeman's global test for metabolomic pathways (Q-stat), Penalized Jacobian method (for calculating network connections), Time-lagged correlation method and zero slopes method. It also includes centering and scaling functions.	MetStaT	CRAN
RF for the construction, optimization and validation of classification models with the aim of identifying biomarkers. Also normalization, scaling, PCA, MDS.	RFmarkerDetector	CRAN
PLS-DA, RF, SVM, GBM, GLMNET, PAM.	OmicsMarkeR	BioC
Recursive feature elimination approach that selects features which significantly contribute to the performance of PLS-DA, Random Forest or SVM classifiers.	biosigner	BioC
Find Biomarkers in two class discrimination problems with variable selection methods provided for several classification metods (lasso/elastic net, PC-LDA, PLS-DA, and several t-tests).	BioMark	CRAN
Unsupervised feature extraction specifically designed for analysing noisy and high-dimensional datasets.	KODAMA	CRAN
Non-parametric method for identifying differentially expressed features based on the estimated percentage of false predictions.	RankProd	CRAN

Table 6: R packages for statistical analysis of metabolomics data. *(continued)*

Functionalities	Package	Repo
Fits multi-way component models via alternating least squares algorithms with optional constraints: orthogonal, non-negative, unimodal, monotonic, periodic, smooth, or structure. Fit models include InDScal, PARAFAC, PARAFAC2, SCA, Tucker.	multiway	CRAN
Decompose a tensor of any order, as a generalisation of SVD also supporting non-identity metrics and penalisations. 2-way SVD is also available. Also includes PCAn (Tucker-n) and PARAFAC/CANDECOMP.	PTAk	CRAN
Fits multi-way component models via alternating least squares algorithms with optional constraints. Fit models include Individual Differences Scaling, Multiway Covariates Regression, PARAFAC (1 and 2), SCA, and Tucker Factor Analysis.	ThreeWay	CRAN
Performs variable selection in a multivariate linear model by estimating the covariance matrix of the residuals then use it to remove the dependence that may exist among the responses and eventually performs variable selection by using the Lasso criterion.	MultiVarSel	CRAN
Performs the O2PLS data integration method for two datasets yielding joint and data-specific parts for each dataset.	OmicsPLS	CRAN
Contains ordination methods such as ReDundancy Analysis (RDA), (Canonical or Detrended) Correspondence Analysis (CCA, DCA for binary explanatory variables), (Non-metric) Multi-Dimensional Scaling ((N)MDS) and other univariate and multivariate methods. Originally developed for vegetation ecologists, many functions are also applicable to metabolomics.	vegan	CRAN
Linear and non-linear Discriminant Analysis methods (e.g. LDA), stepwise selection and classification methods useful for feature selection.	klaR	CRAN
Variable selection methods for PLS, including significance multivariate correlation (SMC), selectivity ratio (SR), variable importance in projections (VIP), loading weights (LW), and regression coefficients (RC). It contains also some other modelling methods.	plsVarSel	CRAN
Predictive multivariate modelling using PLS and Random Forest Data. Repeated double cross unbiased validation and variable selection.	MUVR	GitLab
Biomarker validation for predicting survival. Cross validation methods to validate and select biomarkers when the outcome of interest is survival.	MetabolicSurv	CRAN
Pre-treatment, classification, feature selection and correlation analyses of metabolomics data.	metabolyseR	GitHub
Components search, optimal model components number search, optimal model validity test by permutation tests, observed values evaluation of optimal model parameters and predicted categories, bootstrap values evaluation of optimal model parameters and predicted cross-validated categories.	packMBPLSDA	CRAN
Robust identification of time intervals are significantly different between groups.	OmicsLonDA	BioC
Omics Data integration		

Table 6: R packages for statistical analysis of metabolomics data. *(continued)*

Functionalities	Package	Repo
Multiple co-inertia analysis of omics datasets (MCIA) is a multivariate approach for visualization and integration of multi-omics datasets. The MCIA method is not dependent on feature annotation therefore can extract important features even when there are not present across all datasets.	omicade4	BioC
STATegRa combines information in multiple omics datasets to evaluate the reproducibility among samples and across experimental condition using component analysis (omicsNPC implements the NonParametric Combination) and clustering.	STATegRa	BioC
Statistical framework supporting many different types of multivariate analyses (e.g. PCA, PLS, CCA, PLS-DA, etc.).	mixOmics	CRAN
STatistics in R Using Class Templates - Classes for building statistical workflows using methods, models and validation objects. Provides mechanism for STATO integration.	STRUCT	GitHub
Multi-omics base clases integrable with commonly used R Bioconductor objects for omics data; container that holds omics results.	MultiDataSet	BioC
Identifies analyte-analyte (e.g. gene-metabolite) pairs whose relationship differs by phenotype (e.g. positive correlation in one phenotype, negative or no correlation in another). The software is also accessible as a user-friendly interface at intlim.bmi.osumc.edu.	IntLIM	GitHub
Missing value imputation		
Mixture-model for accounting for data missingness'.	metabomxtr	BioC
Kernel-Based Metabolite Differential Analysis provides a kernel-based score test to cluster metabolites between treatment groups, in order to handle missing values.	KMDA	CRAN
Visualization and imputation of missing values. VIM provides methods for the evaluation and visualization of the type and patterns of missing data. The included imputation approaches are kNN, Hot-Deck, iterative robust model-based imputation (IRMI), fast matching/imputation based on categorical variables and regression imputation.	VIM	CRAN
Graphical user interface for VIM.	VIMGUI	CRAN
kNN based imputation for microarray data.	impute	BioC
Bootstrap based algorithm and diagnostics for fast and robust multiple imputation for cross sectional, time series or combined cross sectional and time series data.	Amelia	CRAN
Algorithms and diagnostics for the univariate imputation of time series data.	imputeTS	CRAN
Methods for the Imputation of incomplete continuous or categorical datasets. missMDA allows missing data imputation using in categorial, continuous or mixed-type datasets using PCA, CA, a multiple correspondence analysis (MCA) model, a multiple factor analysis (MFA) model or factorial analysis for mixed data (FAMD).	missMDA	CRAN
Random forest based missing data imputation for mixed-type, nonparametric data. An out-of-bag (OOB) error estimate is used for model optimization.	missForest	CRAN

Table 6: R packages for statistical analysis of metabolomics data. *(continued)*

Functionalities	Package	Repo
Multivariate imputation by chained equations using fully conditional specifications (FCS) for categorical, continuous and binary datasets. It includes various diagnostic plots for the evaluation of the imputation quality.	mice	CRAN
Missing data imputation using an approximate Bayesian framework. Diagnostic algorithms are included to analyze the models, the assumptions of the imputation algorithm and the multiply imputed datasets.	mi	CRAN
Iterative Gibbs sampler based left-censored missing value imputation.	GSimp	GitHub
Multiple workflow steps		
Missing value imputation, filtering, normalisation and averaging of technical replications.	MSPrep	SF
HCA, Fold change analysis, heat maps, linear models (ordinary and empirical Bayes), PCA and volcano plots. Also log transformation, missing value replacement and methods for normalisation. Cross-contribution compensating multiple internal standard normalisation (ccmn) and remove unwanted variation (ruv2).	metabolomics	CRAN
Data processing, normalization, statistical analysis, metabolite set enrichment analysis, metabolic pathway analysis, and biomarker analysis.	MetaboAnalystR	GitHub
Pipeline for metabolomics data pre-processing, with particular focus on data representation using univariate and multivariate statistics. Built on already published functions.	muma	GitHub
Framework for multiomics experiments. Identifies sources of variability in the experiment and performs additional analysis (identification of subgroups, data imputation, outlier detection).	MOFA	BioC
Performs entry-level differential analysis on metabolomics data.	MetaboDiff	GitHub
STRUCT wrappers for filtering, normalisation, missing value imputation, glog transform, HCA, PCA, PLSDA, PLSR, t-test, fold-change, ANOVA, Mixed Effects, post-hoc tests	STRUCTToolbox	GitHub
Data transformation, filtering of feature and/or samples and data normalization. Quality control processing, statistical analysis and visualization of MS data.	pmartR	GitHub
Quality control, signal drift and batch correction, transformation, univariate hypothesis testing.	metabolis	GitHub
Missing value filtering and imputation, zero value filtering, data normalization, data integration, data quality assessment, univariate statistical analysis, multivariate statistical analysis such as PCA and PLS-D and potential marker selection	MetCleaning	GitHub
Univariate analysis (linear model), PCA, clustered heatmap, and partial correlation network analysis. Based on classes from the Metabase package(Zhu 2019).	ShinyMetabase	GitHub
Outlier detection, PCA, drift correction,visualization, missing value imputation,classification.	MetabolomicsBasics	CRAN
Pre-processing, differential compound identification and grouping, traditional PK parameters calculation, multivariate statistical analysis, correlations, cluster analyses and resulting visualization.	polyPK	CRAN

Table 6: R packages for statistical analysis of metabolomics data. *(continued)*

Functionalities	Package	Repo
Integration of omics data using multivariate methods such as PLS. Performs community detection and network analysis to allow visualization of positive or negative associations between different datasets generated using samples from the same individuals. Also available as ashinyapp (https://kuppal.shinyapps.io/xmwas).	xMWAS	GitHub
Joint metabolic model-based analysis of metabolomics measurements and taxonomic composition from microbial communities.	MIMOSA	GitHub

2.6 Handling of molecule structures and chemical structure databases

Several packages that can deal with cheminformatics tasks, property calculations, metabolite lookup in (web) databases or mapping between databases or structure format conversions (see Table 7).

A well-established package is rcdk which provide a comprehensive subset of functions from the Chemistry Development Kit [99]. rcdk provides a computer readable representation of molecular structures and provide a wealth of functions to import structures from different molecule structure description formats, manipulate structures, visualize structures and calculate properties and molecular fingerprints. The package fingerprint can then be used to compare fingerprints. rinchi provides reading and writing of InChI and InChIKeys [100]. ChemmineR is an alternative to rcdk, providing many similar functions, with more tools for fingerprints, clustering and others through querying the ChemMine Tools web service [101]. ChemmineR also has significantly faster parsing of SDF files, which can be an advantage when reading large databases. A large number of additional descriptors are available in the package camb which focuses on quantitative predictive models. ChemmineOB provides conversion between a large number of chemical structure formats using OpenBabel [102]. A notable exception is InChI/InChIKey, which is not directly supported by ChemmineOB or ChemmineR and one would thus have to go through rinchi and rcdk for offline import from InChI to ChemmineR or ChemmineOB. RChemMass is a package that combines the functionality of the rcdk with that of RMassBank, and enviPat. The package RRDKit makes (part of) the functionality of the RDKit [103] toolkit available from within R.

A number of existing compound databases are useful for metabolomics. These can supply metadata such as common names and synonyms, database identifiers and experimental or predicted properties. The Rpubchem package provides lookup of information available in PubChem [104,105], while the webchem package provide query of a large number of databases including PubChem, ChemSpider [106], Wikidata [107], Chemical Translation Service [108], PHYSPROP [109], Chemical Identifier Resolver [110] and others. BridgeDbR can be used to map identifiers (metabolites, but also genes and proteins, and interactions) between databases, e.g. PubChem to ChemSpider identifiers; RMassBank and RChemMass also provide some useful web-retrieval functions.

The analysis of identified compounds on the level of substance classes can give biochemical insights which are not obvious from the individual structures, or in case the structures are not fully elucidated. The web tool ClassyFire is able to annotate a given structure with compound classes from their ChemOnt taxonomy as well as different substituents [111]. The classyfireR package supports the retrieval of substance classes using the RESTful API of the ClassyFire tool based on InChIKeys.

Table 7: R Packages for molecule structures and chemical structure databases.

Functionalities	Package	Repo
Structure representation and manipulation		
Subset of functions from the Chemistry Development Kit. Provide a computer readable representation of molecular structures and provide functions to import structures from different molecule structure description formats, manipulate structures, visualize structures and calculate properties and molecular fingerprints.	rcdk	CRAN
Similar torcdkin functionality and provides more fingerprints and clustering methods and provides additional tools through querying the ChemMine Tools web service.	ChemmineR	BioC
Provides conversion of structure representation through OpenBabel.	ChemmineOB	BioC
Exposes functionalities of the RDKit library, including reading and writing of SF files and calculating a few physicochemical properties.	RRDKit	GitHub
Read and write InChI and InChIKey from and torcdk.	rinchi	GitHub
Maximum Common Substructure Searching using ChemmineR structures.	FmcsR	BioC

Table 7: R Packages for molecule structures and chemical structure databases. *(continued)*

Functionalities	Package	Repo
Basic cheminformatics functions tailored for mass spectrometry applications, enhancing functionality available in other packages likercdk, enviPat, RMassBank etc.	RChemMass	GitHub
Provides fingerprinting methods forrcdk.	fingerprint	CRAN
Database queries		
Calculation of molecular properties.	camb	GitHub
Querying information from PubChem.	Rpubchem	CRAN
Querying information from various web services (CACTUS, CTS, PubChem, ChemSpider) as part of compound list generation.	RMassBank	BioC
Querying information from a large number of databases.	webchem	CRAN
R Interface to the ClassyFire REST API.	classyfireR	CRAN
Allows mapping of identifiers from one database to another, for metabolites, genes, proteins, and interactions.	BridgeDbR	BioC
Define utilities for exploration of human metabolome database, including functions to retrieve specific metabolite entries and data snapshots with pairwise associations.	hmdbQuery	BioC
Parsers for many compound databases including HMDB, MetaCyc, ChEBI, FooDB, Wikidata, WikiPathways, RIKEN respect, MaConDa, T3DB, KEGG, Drugbank, LipidMaps, MetaboLights, Phenol-Explorer, MassBank.	MetaDBparse	GitHub
Functionality to create and use compound databases generated from (mostly publicly) available resources such as HMDB, ChEBI and PubChem.	CompoundDb	GitHub
Standardized and extensible framework to query chemical and biological databases.	biodb	GitHub

2.7 Network analysis and biochemical pathways

The R environment offers packages to analyse networks of metabolomics data and metabolic pathways (see Table 8). Within this section, we refer to 'pathway' as a linked series of chemical reactions between molecules, conveyed by enzymes that lead to a product or change in a cell. These molecules are also known as metabolites and transformations occur in the same cellular compartment or in close vicinity. The term 'network' refers to the entity of metabolites that are connected biologically, chemically or structurally (e.g. similarity between MS/MS spectra of two metabolites), functionally or by any other measure (e.g. statistically correlated).

2.7.1 Network infrastructure and analysis

The R environment offers general infrastructure for network analysis. Functionality is implemented in a plethora of software packages, amongst others igraph, tidygraph or the statnet suite. These packages offer functions to generate networks from respective data input (e.g. adjacency matrices), to analyse networks, calculate network properties and to visualize networks. Generally, any kind of metabolomics data that can be converted to an interpretable format for one of these packages can be analyzed by generic network analysis tools. For example, MSnbase offers functionality to calculate similarity scores between MS/MS spectral data that can be readily interpreted as a spectral similarity network (see [112] for the pioneering work of mass spectral molecular networking for biological systems). Such networks can be analysed by the functions provided by the above-mentioned packages or by packages tailored more towards the analysis of biological data (e.g. RedeR). Specifically interesting for metabolomics applications is DiffCorr, an R package to compare correlation networks from two different experimental conditions, that builds on an association measure such as Pearson's correlation coefficient to identify distinctive properties. DiffCorr enables testing of differential correlation of high-dimensional data sets by identifying the first principal component-based 'eigen-molecules' in the correlation networks. DiffCorr then tests these differential correlation values based on Fisher's z-transformation to identify discriminating metabolite pairs that show different response to conditions. Another R package, more tailored towards the analysis of metabolomics data, is BioNetStat, which creates correlation-based networks from metabolite concentration data and analyses the networks based on graph spectra (group of eigenvalues in an adjacency matrix), spectral entropy, degree distribution and node centralities. BioNetStat also allows for KEGG pathway visualization of metabolite data.

2.7.2 Metabolite annotation

As mentioned above in section 2.2, a major challenge in metabolomics is metabolite annotation, spanning the annotation of known compounds (dereplication) or annotation of unknown metabolites and proposing hypotheses of their structures. Network and pathway analysis can be employed to putatively annotate metabolites in metabolomics data sets. The Bioconductor package MetNet aims at facilitating detection and putative annotation of unknown MS1 features in untargeted metabolomic studies. MetNet infers networks by using an ensemble of statistical associations between intensity values across samples and structural information (mass difference matching between features to a list of enzymatic transformation, retention time adjustment) to infer metabolic networks and guide the annotation of especially specialized metabolites of plant, fungi or bacteria samples. Another package to improve annotation is the package xMSAnnotator that incorporates a multi-criteria scoring algorithm to annotate mass features into different confidence levels. xMSAnnotator uses coelution, pathway level correlations, correlation and KEGG [113–115], HMDB, Toxin and Toxin Target Database (T3DB) [116,117], LipidMaps [118] and ChemSpider [106] for annotation and incorporates several filter steps, e.g. by defining modules of co-expressing m/z features using WGCNA and a topological overlap-based dissimilarity matrix and thereby categorizing related metabolites into the same network modules.

Molecular networking starting from MS/MS data can enhance the annotation of metabolites. MetDNA, implemented in R, JavaScript and Python (available via a web interface on http://metdna.zhulab.cn), combines MS1 and MS/MS data to putatively annotate features in metabolomics data sets [119]. MetDNA uses

a metabolic reaction network based recursive algorithm for metabolite annotation employing spectral matching of MS/MS spectra in an automatic fashion. The iterated application of similarity matching between reaction pairs, a substrate metabolite with its product metabolite displaying similar chemical structures, allows the expansion of annotation using seed metabolites or previously annotated metabolites.

MetCirc, designed for the annotation of MS/MS features in untargeted metabolomics data, visualizes the spectral similarity matrix (e.g. the normalized dot product) between MS/MS spectra in a Circos-like interactive shiny application. Within the shiny application, similarity scores can be thresholded, MS/MS spectra can be interactively explored and annotated based on expert knowledge given the similarity score and displayed spectral features. MetCirc relies on the MSnbase framework to store MS/MS spectral data and to calculate similarities between spectra. Similarly, CluMSID employs spectral similarity matching to guide annotation of MS/MS spectra, incorporates functionality to calculate a correlation networks and for hierarchical and density-based clustering. compMS2Miner is another R package for MS/MS feature annotation and offers functionality for noise filtering, MS/MS substructure annotation, calculation of correlation- and spectral similarity-based networks and interactive visualization.

2.7.3 Generation of metabolic networks

Several R packages implement the functionality to generate metabolic networks. These networks can afterwards be analysed by their topological properties, be used to identify motifs that differ between experimental conditions or queried to find associations between metabolic features. MetaMapR generates metabolic networks by integrating enzymatic transformation, structural similarity between metabolites, mass spectral similarity and empirical correlation information. Hereby, MetaMapR queries biochemical reactions in KEGG and molecular fingerprints for structural similarities in PubChem. Furthermore, MetaMapR aims at incorporating metabolites with unknown biochemistry and unknown structures, and integrates other data sources (genomic, proteomic, clinical data). The package Metabox offers a pipeline for metabolomics data analysis, including functionality for data-driven network construction using correlation, estimation of chemical structure similarity networks using substructure fingerprints. Its statistical analysis highlights metabolites that are altered based on the experimental design group, which can be further interrogated by network and pathway analysis tools. Furthermore, the package MetabNet includes functionality to perform targeted metabolome-wide association studies (MWAS) and to guide the association of unknowns to a specific metabolic pathway, followed by mapping a target metabolite to the metabolic network structure.

2.7.4 Pathway analysis

Several R packages enable pathway analysis that uses quantitative data of metabolites and maps these to biological pathways. The Bioconductor package pwOmics analyses proteomics, transcriptomics and other -omics data in combination to highlight molecular mechanisms for single-point and time-series experiments. In downstream analyses, pwOmics allows for pathway, transcription factor and target gene identification.

Another important aspect commonly executed is enrichment analysis to identify pathways that are up- or downregulated given an experimental condition. The R environment offers a whole range of enrichment analysis packages (e.g. tmod for metabolite data). Targeted more towards pathway analysis, FELLA is a Bioconductor package for enrichment analysis. FELLA detects discriminative metabolic features, maps these to known biological pathways of the KEGG database and detects enriched terms by a diffusion algorithm. CePa offers enrichment analysis tools extending conventional gene set enrichment methods by incorporating pathway topologies. CePa takes nodes rather than terms for analysis and uses network centralities as weight of nodes incorporating pathways from the Pathway Interaction Database (PID, [120]), including NCI/Nature Pathway Interaction , BioCarta [121], Reactome [122] and KEGG [113–115].

MetaboDiff offers functionality to pinpoint to metabolome-wide differences using PCA and t-distributed stochastic neighbor embedding (tSNE) building on the MultiAssayExperiment S4 class. Using t-test or ANOVA, MetaboDiff identified metabolites that differ in their abundance between groups and identifies

modules/sub-pathways by using WGCNA that indicate changes in biological pathways. SDAMS (Semi-parametric differential abundance analysis method for proteomics and metabolomics data from mass spectrometry) building upon the SummarizedExperiment S4 class, performs differential abundance analysis on metabolomics data by linking (non-normally distributed) metabolite levels to phenotypic data, containing zero and possibly non-normally distributed non-zero intensity values.

Many R packages guide the discovery of biomarkers for specific phenotypes. Among these is lilikoi, that maps features to pathways by using standardized HMDB IDs, transforms metabolomic profiles to pathway-based profiles using pathway deregulation scores, a measure how much a sample deviates from a normal level, followed by feature selection, classification and prediction. INDEED (INtegrated DiffErential Expression and Differential network analysis) aims to detect biomarkers by performing a differential expression analysis, which is combined with a differential network analysis based on partial correlation and followed by a network topology analysis. Subsequently, activity scores are calculated based on differences detected in the differential expression and the topology of the differential network that will guide the selection of biomarkers. Another R package for biomarker and feature selection is MoDentify which finds regulated modules, groups of correlating molecules that can span from few metabolites to entire pathways, to a given phenotype. These groups are possibly functionally coordinated, coregulated or driven by a similar or same biological process. Score maximization using a multivariable linear regression model with the candidate module as dependent and the phenotype and optional covariates as independent variables identifies the modules. Furthermore, MoDentify implements Gaussian graphical models, where depending on the resolution nodes reflect metabolites or entire pathways.

PAPi (Pathway activity profiling) assigns pathway activity scores to samples to represent the potential pathway activity and statistically detects affected pathways by applying t-test or ANOVA. PAPi uses KEGG pathway identifiers. pathwayPCA, with gene selection in mind, offers multi-omics data analysis by estimating sample-specific pathway activities, e.g. taken from the rWikiPathways interface. pathwayPCA takes continuous, binary or survival outcomes as input and estimates contributions of individual genes towards pathway significance.

R offers packages to analyze metabolic systems and to estimate biochemical reaction rates in metabolic networks using flux balance analysis, e.g. BiGGR, abcdeFBA, sybil, and fbar. For example, BiGGR interfaces with the BiGG databases that contains reconstructions of metabolic networks. After importing pathways from the database, flux balance and downstream routines can be performed, e.g. linear optimization routines or likelihood-based ensembles of calculated flux distributions fitting experimental data.

The package MetaboLouise simulates longitudinal metabolomics data. The simulation builds on a mathematical representation that is parameterized according to underlying biological networks, i.e. by defining metabolites and relation between them by initializing enzyme rates. Optionally, the package implements functionality to vary the rates depending on the network state, to add external fluxes and to analyze results based on different parameters.

2.7.5 Pathway resources and interfaces

A plethora of pathway resources exist, aptly aggregated by Pathguide.org. A number of these resources can be accessed by R packages, which were partly reviewed in [123]: rBiopaxParser, graphite, NCIgraph, pathview, KEGGgraph, SBMLR, rsbml, gaggle, and PSICQUIC. Of these, graphite stores pathway information for proteins and metabolites of currently fourteen species (version 1.28.0). Available databases are KEGG, Biocarta, Reactome, NCI/Nature Pathway Interaction Database, HumanCyc, Panther, SMPDB and PharmGKB. graphite offers in addition topological and statistical pathway analysis tools for metabolomics data by interfaces with the Bioconductor packages SPIA and clipper and supports functionality to build own pathways. Furthermore, RPathVisio enables creating and editing biological pathways. RPathVisio enables to visualise data on pathways, to perform statistics on pathway data, and provides an interface to WikiPathways. KEGGREST allows to access the KEGG REST API via a client interface. The package provides utility to search keywords, convert identifiers and link across databases. The package also allows to return amino acid sequences as AAStringSet or nucleotide sequences as DNAStringSet objects (from the Biostrings [124]

package). Another package, paxtoolsr, provides literature-curated pathway using the Biological Pathway Exchange (BioPAX) format by providing an interface to the Pathway Commons database (including data from the NCI Pathway Interaction Database (PID), PantherDB, HumanCyc, Reactome, PhosphoSitePlus and HPRD). rWikiPathways is an interface between R and WikiPathways.org. Pathways can be queried, interrogated and downloaded to the R session. Furthermore, rWikiPathways associates metabolite information to pathways when providing the system code of a chemical database (e.g. from HMDB, ChEBI, or ChemSpider). RaMP provides a relational database of Metabolomics Pathways, integrates pathway, gene, and metabolite annotations from KEGG, HMDB, Reactome, and WikiPathways. The database is downloadable as a standalone MySQL dump, for integration with other software, and is also accessible through an R package, and includes a shiny [125] web interface that supports four basic queries: 1) retrieve analytes (genes of metabolites) given a pathway name; 2) retrieve a pathways for one or more analytes; 3) retrieve analytes involved in the same reaction; 4) retrieve ontologies (cellular location, biofluid locations, etc.) from metabolites. The web interface also supports pathway overrepresentation analysis on genes, metabolites, or genes and metabolites combined (query 3) and includes clustering of significantly enriched pathways according to the percent of overlapping analytes between pathways. Further, the web interface provides network visualization of gene-metabolites relationships (query 4).

Table 8: R packages for network analysis and Biochemical pathways.

Functionalities	Package	Repo
Network infrastructure and analysis		
Infrastructure for representation of networks, analysis and visualization.	igraph	CRAN
Infrastructure for representation of networks, analysis and visualization.	tidygraph	CRAN
Infrastructure for representation of networks, analysis and visualization.	statnet	CRAN
Interactive visualization and manipulation of networks.	RedeR	BioC
Comparison of correlation networks from two experiments.	DiffCorr	CRAN
Correlation-based networks from metabolomics data and analysis tools.	BioNetStat	BioC
Annotation		
Putative annotation of unknowns in MS1 data.	MetNet	BioC
Putative annotation of unknowns in MS1 data.	xMSAnnotator	SF
Putative annotation of unknowns using MS1 and MS2 data.	MetDNA	GitHub
Visualization of spectral similarity networks, putative annotation of unknowns using MS2 data.	MetCirc	BioC
Putative annotation of unknowns using MS2 data, clustering of MS2 data.	CluMSID	BioC
Putative annotation of unknowns using MS2 data.	compMS2Miner	GitHub
Generation of metabolite networks		
Biochemical reaction networks, spectral and structural similarity networks.	MetaMapR	GitHub
Correlation-based networks, structural similarity networks.	Metabox	GitHub
Targeted metabolome-wide association studies.	MetabNet	SF
Generation of scale-free correlation-based networks.	WGCNA	CRAN
Pathway analysis		
Analysis of -omics data, pathway, transcription factor and target gene identification.	pwOmics	BioC
MSEA a metabolite set enrichment analysis with factor loading in principal component analysis.	mseapca	CRAN
Enrichment analysis of a list of affected metabolites.	tmod	CRAN

Table 8: R packages for network analysis and Biochemical pathways. *(continued)*

Functionalities	Package	Repo
Network-based enrichment analysis of a list of affected metabolites.	FELLA	BioC
Pathway-based enrichment analysis of a list of affected metabolites.	CePa	CRAN
Differential analysis, modules/sub-pathway identification using networks.	MetaboDiff	GitHub
Integrates metabolic networks and RNA-seq data to construct condition-specific series of metabolic sub-networks and applies to gene set enrichment analysis	metaboGSE	CRAN
Differential analysis.	SDAMS	BioC
Biomarker identification.	lilikoi	CRAN
Biomarker identification.	INDEED	BioC
Biomarker identification.	MoDentify	GitHub
Pathway activity profiling.	PAPi	BioC
Pathway activity profiling.	pathwayPCA	BioC
Flux balance analysis.	BiGGR	BioC
Flux balance analysis.	abcdeFBA	CRAN
Flux balance analysis.	sybil	CRAN
Flux balance analysis.	fbar	CRAN
Identification of affected pathway from phenotype data (interface with graphite).	SPIA	BioC
Identification of affected pathway from phenotype data (interface with graphite).	clipper	BioC
Interface to PathVisio and WikiPathways and pathway analysis and enrichment.	RPathVisio	GitHub
Enrichment analysis of a list of genes and metabolites.	RaMP	GitHub
Simulation of longitudinal metabolomics data based on an underlying biological network	MetaboLouise	CRAN
Pathway resources and interfaces		
BioPax parser and representation in R.	rBiopaxParser	BioC
Interface to KEGG, Biocarta, Reactome, NCI/Nature Pathway Interaction Database, HumanCyc, Panther, SMPDB and PharmGKB.	graphite	BioC
Interface to NCI Pathways Database.	NCIgraph	BioC
Interface to KEGG.	pathview	BioC
Interface to KEGG.	KEGGgraph	BioC
Interface to systems biology markup language (SBML).	SBMLR	BioC
Interface to systems biology markup language (SBML).	rsbml	BioC
Interface to Gaggle-enabled software (Cytoscape, Firegoose, Gaggle Genome browser).	gaggle	BioC
Interface to molecular interaction databases.	PSICQUIC	BioC
Interface to KEGG REST server.	KEGGREST	BioC
Interface to BioPAX OWL files and the Pathway Commons (PW) molecular interaction database.	paxtoolsr	BioC
Interface to WikiPathways.	rWikiPathways	BioC
Database that integrates metabolite and gene biological pathways from HMDB, KEGG, Reactome, and WikiPathways. Includes user-friendly R Shiny web application for queries and pathway enrichment analysis.	RaMP-DB	GitHub

2.8 Multifunctional workflows

When dealing with non-targeted metabolomics data sets, data processing represents a key step for obtaining meaningful and consistent results. While the type and number of data processing methods may vary according to the experimental design and aim of the study, some key steps can be identified that are common for most metabolomics experiments. For this reason, a number of multifunctional R based workflows have been developed over the years. A key advantage of using multifunctional workflows is that most of the functions the user needs are available within the same "environment", so that the data does not have to be formatted to comply with functions in other packages. In this respect, a quite common backbone of R workflows consists in performing a pre-processing step that generates an R object that can be used as argument for different functions. Another advantage is that, in most cases, workflows allow a certain degree of flexibility so that functionalities can be used as standalone functions (modular workflows) to better comply with the user's needs. The packages covering larger parts of metabolomics workflows available in R are listed in Table 9.

These multifunctional packages include comprehensive workflows that focus on multiple aspects, such as: data pre-processing, data validation, preliminary statistical analysis and data visualisation of large metabolomics datasets. The considered workflows support both MS based data (LC-MS and GC-MS) and data generated by different analytical platforms. MAIT (Metabolite Automatic Identification Toolkit) offers pre-processing, annotation, statistical analysis and data visualization. It relies on xcms for peak picking and on CAMERA for the preliminary annotation. In addition to CAMERA, the peak annotation process is implemented by including a functionality that allows relating in-source mass losses to specific biotransformations. Human biotransformations are already included, additional biotransformation criteria can be added by the end user. MAIT also provides a number of statistical tools and visual representations (e.g. PCA, boxplot, PLS) as well as a function to perform identifications using accurate mass search in HMDB. MetMSLine shows some similarities with MAIT in terms of processing stages (xcms-based pre-processing, multivariate statistics, metabolite identifications). Functionalities characterizing MetMSLine include: normalisation, signal drift correction using a smoothing method, noise transformation and outlier removal. SimExTargId is a wrapper of different software and R packages for LC-MS data. It includes tools for data conversion (Proteowizard), peak picking and annotation (xcms and CAMERA), outlier detection and data correction (MetMSLine), and basic statistical analysis. A special feature of SimeExTargId is the real time monitoring of the different workflow stages aimed at metabolomics core facilities; users are notified by email in case of processing errors (e.g. outlier detection, signal drift). mzMatch is slightly different from the above mentioned workflows and is designed to fit in a broader processing pipeline itself. The project also includes a dedicated file format (peakML) and a Java environment. The different modules can still be used independently. mzMatch supports peak picking and grouping using xcms, reproducibility calculation, data normalization. The peakMonitor app identifies peaks using the local database. The identification is performed on the basis of m/z and retention time values with user-defined mass accuracy and retention time deviation values.

MetaDB is built by integrating the metaMS R package into a web application written in Grails. It has also been designed to be integrated with the MetaboLights database. MetaDB supports both LC-MS and GC-MS datasets and offers a wide range of functionalities, including: data storage and metadata management (using the ISA-Tab format and ISACreator tool [126,127]), peak picking and annotation (via metaMS, an xcms and CAMERA add-on) and QC plots.

MStractor is designed for non-expert users to carry out non-targeted data processing on LC-MS experiments. It gathers xcms and CAMERA functions in an user-friendly pipeline, requiring minimal input and providing graphical QC outputs throughout the workflow. It also includes a manual peak curation step and the possibility of calculating descriptive statistics for each sample class.

patRoon is an interface for different MS-based open source software for non-targeted data processing. patRoon covers different aspects of metabolomics workflows, such as: file conversion to open data formats (mzXML and mzML), feature extraction and grouping (using a number of open software and R packages: xcms, OpenMS, enviPick), extraction of MS and MS/MS data (mzR), component generation (RAMClustR, CAMERA, nontarget), formula calculation (GenForm) and compound identification through automatic annotation of MS/MS spectra (MetFrag and SIRIUS with CSI:FingerID). Other functionalities include (in-

teractive) visualization and reporting of workflow data, comparison and combining results from different workflow algorithms and several data reduction and selection strategies.

Specmine provides a general framework that addresses a variety of different analytical platforms, such as LC-MS, GC-MS, NMR, IR and UV-Vis. The package supports many data formats and includes the possibility of adding metadata in a tabular format. It relies on xcms for LC-MS and GC-MS data pre-processing, on hyperSpec for NMR, IR and UV/VIS data processing and on MAIT for metabolite identification. Specmine provides scripts for missing values imputation, univariate and multivariate statistics and machine learning methods. A number of case studies are available for testing purposes.

mQTL.NMR is a package specific for the systematic analysis of 1H NMR metabolomics in quantitative genetics. The package mainly focuses on NMR spectral data pre-processing (normalization, scaling and peak alignment), mQTL mapping in different model organisms, structural assignment of marker metabolites, and result visualization.

enviMass is a comprehensive workflow for the data-mining of LC-MS and GC-MS datasets, which also supports MS/MS experiments. It provides the user with a graphic interface and a flexible workflow structure covering common processing steps such as data conversion, peak picking, noise removal, peak picking, mass re-calibration, data normalisation, and blank subtraction. It also offers a number of more specific and advanced functionalities including: isotopologue and adduct grouping, homologous series detection and visualization, estimation of atom counts for nontarget components, temporal sequences, profile trend detection and processing of both data dependent and data independent acquisition of MS/MS experiments. RMassScreening is a workflow for batch processing of LC-HRMS datasets using a script interface, YAML-based setting configuration and visual interactive data evaluation. It provides wrappers for script-based usage of enviPick and basic enviMass components, and implements suspect screening and combinatorial prediction of possible metabolites (transformation products) from parent compounds. A GUI provides facilities to analyze the results, grouped by sample groups and experimental timepoints, by applying freely adjustable filters.

MetaboNexus is an interactive data analysis platform for metabolomics experiments, which provides a user friendly R shiny-based GUI designed to work without the need of web server connections. It allows preprocessing (using xcms and MZmine), data scaling, univariate and multivariate statistics (t-test, ANOVA, PCA, PLS-DA, Random Forest, Heatmap), putative metabolite identification (library matching of MS and MS/MS adduct with METLIN, HMDB and MassBank databases), and a number of functions for data visualization.

Table 9: R packages with multifunctional workflows.

Functionalities	Package	Repo
Convenience wrapper for pre-processing tools (XCMS, CAMERA) and a number of statistical analyses.	MAIT	BioC
Preprocessing (xcms), replicate merging, noise, blank and missingness filtering, feature grouping, annotation of known compounds, isotopic labeling analysis, annotation from KEGG or HMDB, common biotransformations and probabilistic putative metabolite annotation using MetAssign.	mzMatch	GitHub
XCMS and CAMERA based workflow for non-targeted processing of LC-MS datasets, It includes pre-processing, peak picking,peak filtering, data normalization and descriptive statistics calculation.	MStractor	GitHub
Performs simultaneous raw data to mzXML conversion (MSConvert), peak-picking, automatic PCA outlier detection and statistical analysis, visualization and possible MS2 target list determination during an MS1 metabolomic profiling experiment.	simExTargId	GitHub
Pre-processing of large LC-MS datasets. Performs automatic PCA with iterative automatic outlier removal and, clustering analysis and biomarker discovery.	MetMSLine	GitHub

Table 9: R packages with multifunctional workflows. *(continued)*

Functionalities	Package	Repo
Workflow for the systematic analysis of 1H NMR metabolomics dataset in quantitative genetics. Performs pre-processing, mQTL mapping, metabolites structural assignment and offers data visualisation tools.	mQTL.NMR	BioC
Workflow for pre-processing, qc, annotation and statistical data analysis of LC-MS and GC-MS based metabolomics data to be submitted to public repositories.	MetaDB	GitHub
Specmine is a framework mainly built on a number of already published packages. It supports data processing form different analytical platforms (LC-MS, GC-MS, NMR, IR, UV-Vis).	specmine	GitHub
Common interface for a number of different MS based data processing software. It covers various aspects, such as data preparation and data extraction, formula calculation, compound identification and reporting.	patRoon	GitHub
Processing of high resolution of LC-MS data for environmental trend analysis.	enviMass	Zenodo
Workflow for preprocessing of LC-HRMS data, suspect screening, screening for transformation products using combinatorial prediction, and interactive filtering based on ratios between sample groups.	RMassScreening	GitHub
Workflow to perform pre-processing, statistical analysis and metabolite identifications based on database search of detected spectra.	MetaboNexus	GitHub
Shiny-based platform to extract differential features from LC-MS data, includes XCMS-based feature detection, statistical analysis, prediction of molecular formulas, annotation of MS2 spectra, MS2 molecular networking and chemical compound database search.	METABOseek	GitHub
RShiny interface to Metabolomics packages & MetaboAnalyst scripts.	MetaboShiny	GitHub
Preprocessing and visualizing for LC-MS data, as well as statistical analyses, mainly based on univariate linear models.	amp	GitHub

2.9 User interfaces and workflow management systems

Visualisation is an important part of data analysis. Traditionally graphics in R have been focussed on creating static plots, while typical explorative studies generally require interactive visualisation to fully investigate the data. User interactions could range from simply zooming in chromatographic or spectroscopic data through to temporarily excluding data from a complex plot for clarity. Several packages in R are available for making interactive plots, e.g. the Plotly library [128] to create interactive graphics from the static plots generated by the popular plotting framework ggplot2 [129]. The use of interactive plots in R is growing, and is helped by an increasing number of code examples available.

Another way interactive plots, and even full GUI tools, are being introduced into R is through the shiny framework, which can create web apps using the full power of R packages as the backend. Many such tools related to metabolomics data analysis are also becoming available, which decreases the learning curve considerably for the typical metabolomics scientist without a computational background. A current gap in the shiny metabolomics landscape are powerful and re-usable widget collections for e.g. spectra viewers, molecular structures or metabolic networks.

There are several approaches to create, share and use data analysis in R for developers and users, with different strengths and weaknesses. Table 10 summarizes several ways to create and run a data analysis with some interpretation and comparative comments. Note that in some cases it is difficult to quantify "implementation simplicity", e.g. in the case of shiny apps, which can range from rather straightforward to highly complex.

***Table 10**: Categorization of creating and sharing R code and data analysis functionality. Symbols indicate strengths (+, ++) or weaknesses (-, –) or neutral (o) assessment.*

Framework	Implementation simplicity low to high	User-friendliness low to high	Interactivity	Example URLs
R script	++	–	-	write.mzTab
R Markdown vignette	o	o	–	xcms, patRoon
Jupyter Notebook	o	+	+	MSEAp
LearnR (CRAN)	-	++	+	LearnR Examples
shiny app	–	++	++	MetFamily and apps in e.g. RaMP-DB, IntLIM

All of these environments can be run locally, or installed on a (local or cloud-based) server. Recently, several initiatives have started to provide publicly available computing resources. Examples are e.g. the previously mentioned rdrr.io, which offers to paste R code into an online console for execution. The console can also be embedded into individual websites. The same project also hosts rnotebook.io, which allows to create and run R notebooks. The shinyapps.io platform operated by RStudio Inc has free and paid options to host shiny apps. The binder project (involving members from large academic institutions and companies (like UC Berkeley, Cal Poly San Luis Obispo, Wild Tree Tech Switzerland, Netflix or Simula Research Lab) is an infrastructure to create and use shareable, interactive and reproducible data analysis (not only) with R [130] by taking any GitHub repository, turning it into a Docker image and launching it on a cloud service. The package holepunch [131] simplifies preparing an R project for launching on binder. A public instance is the mybinder.org service providing (limited) resources to execute R based scripts in a hosted Rstudio, Jupyter

notebook or applications written with e.g. shiny. The binder infrastructure code is available on GitHub, so that the service can be offered by universities and research groups to its users, lifting the resource limitations of the public instance.

In some cases an R package can provide bindings to existing tools and libraries written in other languages (see Table 11). This is for example in the case for the packages rcdk or MetFragR using the rJava bindings, or mzR which is a wrapper around the Proteowizard C++ library using the Rcpp package. The fairly new reticulate package provides the corresponding infrastructure to execute Python from R code.

Several workflow systems support workflow nodes and tools that can wrap and execute R code, and in turn build on the huge number of R packages (not only) for metabolomics. In this way, systems like KNIME [132,133] and Galaxy [134,135] also provide a graphical user interface and visual programming using the wrapped R functionality, and possibly combine with tools developed in other programming frameworks.

Galaxy is a web-based environment for omics data analysis [136]. The Workflow4metabolomics.org online Galaxy infrastructure dedicated to metabolomics [135] includes wrappers of xcms, CAMERA, metaMS, proFIA, ropls, biosigner and is open to new contributions. W4M is supported by two national infrastructures: the French Institute of Bioinformatics (www.france-bioinformatique.fr) and the Infrastructure for Metabolomics and Fluxomics (www.metabohub.fr) [137]. Wrapping R code into a Galaxy module is quite straightforward: examples can be found on the toolshed central repository (toolshed.g2.bx.psu.edu) and in the RGalaxy bioconductor package. An additional benefit is that the workflow developers need to ensure seamless data flow through the workflow steps, and often contribute the glue code to bridge the gap between objects and data structures that are not always directly compatible across different packages and softwares, thus also improving interoperability beyond the use in workflow systems.

Workflows and input/output data can be publicly referenced [138,139] on the Workflow4metabolomics platform, thus enabling fully reproducible research. By using workflow systems the reuse and reprocessing of data sets is greatly encouraged, as well as the tracking of data provenance [140]. This way, workflows help to boost the FAIR principles that were shaped for data [141].

Table 11: Packages to interface R with other languages and workflow environments

Functionalities	Package	Repo
Given an R function and its manual page, make the documented function available in Galaxy.	RGalaxy	BioC
Integration of R and C++. Many R data types and objects can be mapped back and forth to C++ equivalents.	Rcpp	CRAN
Low-Level R to Java Interface.	rJava	CRAN
Interface to 'Python' modules, classes, and functions and translation between R and Python objects.	reticulate	CRAN

2.10 Metabolomics data sets

Sharing of data has become increasingly common, and metabolomics data are available from MetaboLights [142] in the EU, GNPS [49] and Metabolomics Workbench [143] in the US. In the context of this review, we focus instead on data in R packages, which is important for development, unit testing, documentation and user training (see Table 12). While there is no difference in R between software and data packages per se, they are handled differently in the Bioconductor infrastructure and separate views exist.

There are several data sets with raw data from LC-MS and flow injection analysis, which can be used by the data pre-processing packages in the previous sections. Other packages contain pre-processed data from GC-MS, LC-MS or NMR in the form of peak tables, which are then typically used in statistics packages, network analysis and other downstream analyses.

Table 12: Metabolomics data sets packaged as R packages.

Functionalities	Package	Repo
LC-MS		
12 HPLC-MS NetCDF files (Agilent 1100 LC-MSD SL).	faahKO	BioC
16 UPLC-MS mzData files (Bruker microTOFq).	mtbls2	BioC
12 UPLC-MS mzML files (AB Sciex TripleTOF 5600, SWATH mode).	mtbls297	GitHub
Different raw MS files (LTQ, TripleQ, FTICR, Orbitrap, QTOF) some in different formats (mzML, mzXML, mzData, mzData.gz, NetCDF, mz5). Also mzid format from proteomics.	msdata	BioC
Metadata and DDA MS/MS spectra of 15 narcotics standards (LTQ Orbitrap XL).	RMassBankData	BioC
183 x 109 peak table.	ropls	BioC
69 x 5,501 peak table.	biosigner	BioC
40 x 1,632 peak table.	BioMark	CRAN
Raw MS files from a set of blanks and standards that contain common environmental contaminants (acquired with Bruker maXis 4G).	patRoonData	GitHub
Proteomics, metabolomics GC-MS and Lipidomics data from Calu-3 cell culture; 3 mockulum treated and 9 MERS-CoV treated; Time point, 18 hour from MassIVE dataset ids MSV000079152, MSV000079153, MSV000079154.	pmartRdata	GitHub
FIA-MS		
6 mzML files (human plasma spiked with 40 compounds acquired in positive mode on an orbitrap fusion).	plasFIA	BioC
mzML files (Thermo Exactive) from comparison of leaf tissue from 4 B. distachyon ecotypes with Flow-infusion electrospray ionisation-high resolution mass spectrometry (FIE-HRMS). Also includes data sets with 10 technical injections of human urine and another 10 injections from leaf tissue (ecotype ABR1).	metaboData	GitHub
GC-MS		
52 x 154 peak table.	pcaMethods	BioC
NMR		
18 x 189 peak table.	MetabolAnalyze	CRAN
33 x 164 peak table.	MetabolAnalyze	CRAN
ASICSdata: 1D NMR spectra for ASICS.	ASICSdata	BioC

3 Conclusions

This review surveyed both the scientific literature and the R landscape for packages relevant to metabolomics research. While it was very easy to find relevant packages in CRAN and even more so in BioC, many packages are scattered across other source code hosting platforms. While GitHub has a concept of topics (see github.com/search?q=topic:metabolomics+topic:r), and crawlers like rdrr.io can find R packages across several platforms, the best findability can be achieved through well-integrated umbrella projects like Bioconductor, which provide additional infrastructure and also improve the community interaction through conferences and workshops.

This also shows the need for more detailed metadata of the R packages allowing easier mixing and matching of packages, noting that Bioconductor already does a very good job. R packages already have a long standing history of metadata annotation via their DESCRIPTION and CITATION files. These provide links to other packages (e.g. dependencies and suggestions) and literature describing the package. Exposing package and vignette metasdata with semantic approaches will support the community in developing further, more advanced multi-functional workflows for metabolomics. Authors have recently adopted Bioschemas [144] to make metadata easier findable. For example, efforts to start annotation in vignettes allows the ELIXIR Training eSupport System TeSS (tess.oerc.ox.ac.uk) to pick up newer versions (see this git commit [@"attempt to add bioschemas.org json-ld to the vignette html · bridgedb/bridgedbr@40e741a · github"_n.d.]), and efforts are underway to expose content from the DESCRIPTION file as Bioschemas annotation on Bioconductor (see this pull request [@"added template for bioschemas tool annotation by egonw · pull request #25 · bioconductor/bioconductor.org · github"_n.d.]). These actions greatly contribute to community adoption and encourage the reuse of R-based computational workflows in different use cases [140].

In some cases, software described in the literature was only available "on request", which in practice often turns out to be not available anymore. This review also did not assess whether the R packages (and their dependencies) can be installed on a current R installation. A recent survey [145] showed how the repeatability of papers using scientific software drops when software is not available or does not install. Issues/bug reports were filed for packages that were found that were not able to be tested on contemporary operating systems. The way out of the (un-)repeatability trap can be expressed in very few, seemingly trivial, rules [146] and hosting the code in the open repositories, if possible with regular builds or even Continuous Integration. As discussed earlier, the metabolomics packages have tighter connections in an established community such as Bioconductor, rather than in other package repositories. In the last few years, Bioconductor packages for metabolomics and proteomics data analysis started converging towards a common mass spectrometry infrastructure, which simplifies interoperability between these packages. Based on experiences from these efforts, the RforMassSpectrometry (RforMassSpectrometry.org) initiative was recently started aiming at providing efficient, thoroughly documented, tested and flexible R software for MS data import, handling and analysis. Significant improvements can thus be expected in the future, simplifying and unifying MS data handling for the benefit of the end users. RforMassSpectrometry also contains the metaRbolomics-book [147], which will be a continuously developed resource with additional examples beyond this review.

The authors expect that the metaRbolomics landscape will continue its steady growth rate and keep track of the evolving metabolomics experiments to come.

References

1. Emwas, A.-H.; Roy, R.; McKay, R.T.; Tenori, L.; Saccenti, E.; Gowda, G.A.N.; Raftery, D.; Alahmari, F.; Jaremko, L.; Jaremko, M. et al. NMR spectroscopy for metabolomics research. *Metabolites* **2019**, *9*.

2. *Metabolomics in practice: Successful strategies to generate and analyze metabolic data*; Lämmerhofer, M., Weckwerth, W., Eds.; Wiley-VCH Verlag GmbH & Co. KGaA: Weinheim, Germany, 2013; ISBN 9783527330898.

3. Villas-Boas, S.G.; Nielsen, J.; Smedsgaard, J.; Hansen, M.A.E.; Roessner-Tunali, U. *Metabolome analysis: An introduction*; 1st ed.; Wiley, John & Sons, Incorporated, 2007; p. 319; ISBN 978-0-471-74344-6.

4. *Metabolomics: Practical guide to design and analysis*; Wehrens, R., Salek, R., Eds.; Chapman & hall/CRC mathematical and computational biology; Chapman; Hall/CRC, 2019; ISBN 1498725260.

5. International Metabolomics Society Free Tools & Learning Resources - Metabolomics Society Wiki.

6. Salek, R.; Emery, L.; Beisken, S. Metabolomics: An introduction EMBL-EBI train online.

7. R Core Development Team R: A language and environment for statistical computing 2018.

8. Spicer, R. GitHub - RASpicer/MetabolomicsTools 2018.

9. Spicer, R.; Salek, R.M.; Moreno, P.; Cañueto, D.; Steinbeck, C. Navigating freely-available software tools for metabolomics analysis. *Metabolomics : Official journal of the Metabolomic Society* **2017**, *13*, 106.

10. Misra, B.B.; Hooft, J.J.J. van der Updates in metabolomics tools and resources: 2014-2015. *Electrophoresis* **2016**, *37*, 86–110.

11. Misra, B.B.; Fahrmann, J.F.; Grapov, D. Review of emerging metabolomic tools and resources: 2015-2016. *Electrophoresis* **2017**, *38*, 2257–2274.

12. Misra, B.B. New tools and resources in metabolomics: 2016-2017. *Electrophoresis* **2018**, *39*, 909–923.

13. Misra, B. GitHub - biswapriyamisra/metabolomics: Tools databases resources in metabolomics & integrated omics in 2015-2016 2017.

14. Kannan, L.; Ramos, M.; Re, A.; El-Hachem, N.; Safikhani, Z.; Gendoo, D.M.A.; Davis, S.; Gomez-Cabrero, D.; Castelo, R.; Hansen, K.D. et al. Public data and open source tools for multi-assay genomic investigation of disease. *Briefings in Bioinformatics* **2016**, *17*, 603–615.

15. Blaženović, I.; Kind, T.; Ji, J.; Fiehn, O. Software tools and approaches for compound identification of LC-MS/MS data in metabolomics. *Metabolites* **2018**, *8*.

16. Mullen, K. CRAN task view: Chemometrics and computational physics 2019.

17. Gentleman, R.C.; Carey, V.J.; Bates, D.M.; Bolstad, B.; Dettling, M.; Dudoit, S.; Ellis, B.; Gautier, L.; Ge, Y.; Gentry, J. et al. Bioconductor: Open software development for computational biology and bioinformatics. *Genome Biology* **2004**, *5*, R80.

18. Bioconductor Bioconductor - BiocViews.

19. The Comprehensive R Archive Network CRAN repository policy.

20. Bioconductor Bioconductor - developers.

21. Theußl, S.; Zeileis, A. Collaborative software development using r-forge. *The R journal* **2009**, *1*, 9.

22. Boettiger, C.; Chamberlain, S.; Hart, E.; Ram, K. Building software, building community: Lessons from the rOpenSci project. *Journal of open research software* **2015**, *3*.

23. Vries, A. de; Rickert, J. The network structure of r packages on CRAN & BioConductor 2015.

24. Vries, A. de Differences in the network structure of CRAN and BioConductor (revolutions) 2015.

25. Vries, A. de GitHub - andrie/cran-network-structure: Scripts used for my UseR!2015 presentation on the network structure of CRAN 2015.

26. Neumann, S. GitHub - sneumann/metaRbolomics: Metabolomics in r and bioconductor 2019.

27. Chambers, M.C.; Maclean, B.; Burke, R.; Amodei, D.; Ruderman, D.L.; Neumann, S.; Gatto, L.; Fischer, B.; Pratt, B.; Egertson, J. et al. A cross-platform toolkit for mass spectrometry and proteomics. *Nature Biotechnology* **2012**, *30*, 918–920.

28. Kessner, D.; Chambers, M.; Burke, R.; Agus, D.; Mallick, P. ProteoWizard: Open source software for rapid proteomics tools development. *Bioinformatics* **2008**, *24*, 2534–2536.

29. Fuhrer, T.; Heer, D.; Begemann, B.; Zamboni, N. High-throughput, accurate mass metabolome profiling of cellular extracts by flow injection-time-of-flight mass spectrometry. *Analytical Chemistry* **2011**, *83*, 7074–7080.

30. Mahieu, N.G.; Genenbacher, J.L.; Patti, G.J. A roadmap for the XCMS family of software solutions in metabolomics. *Current Opinion in Chemical Biology* **2016**, *30*, 87–93.

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. *Analytical Chemistry* **2006**, *78*, 779–787.

32. Tautenhahn, R.; Böttcher, C.; Neumann, S. Highly sensitive feature detection for high resolution LC/MS. *BMC Bioinformatics* **2008**, *9*, 504.

33. Conley, C.J.; Smith, R.; Torgrip, R.J.O.; Taylor, R.M.; Tautenhahn, R.; Prince, J.T. Massifquant: Open-source kalman filter-based XC-MS isotope trace feature detection. *Bioinformatics* **2014**, *30*, 2636–2643.

34. Huber, W.; Carey, V.J.; Gentleman, R.; Anders, S.; Carlson, M.; Carvalho, B.S.; Bravo, H.C.; Davis, S.; Gatto, L.; Girke, T. et al. Orchestrating high-throughput genomic analysis with bioconductor. *Nature Methods* **2015**, *12*, 115–121.

35. Martin Morgan, V.O. SummarizedExperiment. *Bioconductor* **2017**.

36. Zhu, C. Zhuchcn/metabase: A r package to store, manipulate, analyze, and visualize metabolomics data 2019.

37. Hoffmann, N.; Rein, J.; Sachsenberg, T.; Hartler, J.; Haug, K.; Mayer, G.; Alka, O.; Dayalan, S.; Pearce, J.T.M.; Rocca-Serra, P. et al. mzTab-m: A data standard for sharing quantitative results in mass spectrometry metabolomics. *Analytical Chemistry* **2019**, *91*, 3302–3310.

38. Scheltema, R.A.; Jankevics, A.; Jansen, R.C.; Swertz, M.A.; Breitling, R. PeakML/mzMatch: A file format, java library, r library, and tool-chain for mass spectrometry data analysis. *Analytical Chemistry* **2011**, *83*, 2786–2793.

39. Shahaf, N.; Rogachev, I.; Heinig, U.; Meir, S.; Malitsky, S.; Battat, M.; Wyner, H.; Zheng, S.; Wehrens, R.; Aharoni, A. The WEIZMA spectral library for high-confidence metabolite identification. *Nature Communications* **2016**, *7*, 12423.

40. Witting, M. GitHub - michaelwitting/ms2dbworkflow.

41. Horai, H.; Arita, M.; Kanaya, S.; Nihei, Y.; Ikeda, T.; Suwa, K.; Ojima, Y.; Tanaka, K.; Tanaka, S.; Aoshima, K. et al. MassBank: A public repository for sharing mass spectral data for life sciences. *Journal of Mass Spectrometry* **2010**, *45*, 703–714.

42. Mylonas, R.; Mauron, Y.; Masselot, A.; Binz, P.-A.; Budin, N.; Fathi, M.; Viette, V.; Hochstrasser, D.F.; Lisacek, F. X-rank: A robust algorithm for small molecule identification using tandem mass spectrometry. *Analytical Chemistry* **2009**, *81*, 7604–7610.

43. Collins, J.R.; Edwards, B.R.; Fredricks, H.F.; Van Mooy, B.A.S. LOBSTAHS: An adduct-based lipidomics strategy for discovery and identification of oxidative stress biomarkers. *Analytical Chemistry* **2016**, *88*, 7154–7162.

44. Koelmel, J.P.; Kroeger, N.M.; Ulmer, C.Z.; Bowden, J.A.; Patterson, R.E.; Cochran, J.A.; Beecher, C.W.W.; Garrett, T.J.; Yost, R.A. LipidMatch: An automated workflow for rule-based lipid identification using untargeted high-resolution tandem mass spectrometry data. *BMC Bioinformatics* **2017**, *18*, 331.

45. Alcoriza-Balaguer, M.I.; García-Cañaveras, J.C.; Lopez, A.; Conde, I.; Juan, O.; Carretero, J.; Lahoz, A. LipidMS: An r package for lipid annotation in untargeted liquid chromatography-data independent acquisition-mass spectrometry lipidomics. *Analytical Chemistry* **2018**, *91*, 836–845.

46. Standards, T.N.I. of; Technology Library conversion tool 2012.

47. Tsugawa, H.; Cajka, T.; Kind, T.; Ma, Y.; Higgins, B.; Ikeda, K.; Kanazawa, M.; VanderGheynst, J.; Fiehn, O.; Arita, M. MS-DIAL: Data-independent MS/MS deconvolution for comprehensive metabolome analysis. *Nature Methods* **2015**, *12*, 523–526.

48. MassBank of North America MoNA downloads.

49. Wang, M.; Carver, J.J.; Phelan, V.V.; Sanchez, L.M.; Garg, N.; Peng, Y.; Nguyen, D.D.; Watrous, J.; Kapono, C.A.; Luzzatto-Knaan, T. et al. Sharing and community curation of mass spectrometry data with global natural products social molecular networking. *Nature Biotechnology* **2016**, *34*, 828–837.

50. Jacob, D.; Deborde, C.; Lefebvre, M.; Maucourt, M.; Moing, A. NMRProcFlow: A graphical and interactive tool dedicated to 1D spectra processing for NMR-based metabolomics. *Metabolomics : Official journal of the Metabolomic Society* **2017**, *13*, 36.

51. Dieterle, F.; Ross, A.; Schlotterbeck, G.; Senn, H. Probabilistic quotient normalization as robust method to account for dilution of complex biological mixtures. Application in 1H NMR metabonomics. *Analytical Chemistry* **2006**, *78*, 4281–4290.

52. Wishart, D.S.; Knox, C.; Guo, A.C.; Eisner, R.; Young, N.; Gautam, B.; Hau, D.D.; Psychogios, N.; Dong, E.; Bouatra, S. et al. HMDB: A knowledgebase for the human metabolome. *Nucleic Acids Research* **2009**, *37*, D603–10.

53. Wishart, D.S.; Jewison, T.; Guo, A.C.; Wilson, M.; Knox, C.; Liu, Y.; Djoumbou, Y.; Mandal, R.; Aziat, F.; Dong, E. et al. HMDB 3.0–the human metabolome database in 2013. *Nucleic Acids Research* **2013**, *41*, D801–7.

54. Wishart, D.S.; Tzur, D.; Knox, C.; Eisner, R.; Guo, A.C.; Young, N.; Cheng, D.; Jewell, K.; Arndt, D.; Sawhney, S. et al. HMDB: The human metabolome database. *Nucleic Acids Research* **2007**, *35*, D521–6.

55. Wishart, D.S.; Feunang, Y.D.; Marcu, A.; Guo, A.C.; Liang, K.; Vázquez-Fresno, R.; Sajed, T.; Johnson, D.; Li, C.; Karu, N. et al. HMDB 4.0: The human metabolome database for 2018. *Nucleic Acids Research* **2018**, *46*, D608–D617.

56. Beckonert, O.; Keun, H.C.; Ebbels, T.M.D.; Bundy, J.; Holmes, E.; Lindon, J.C.; Nicholson, J.K. Metabolic profiling, metabolomic and metabonomic procedures for NMR spectroscopy of urine, plasma, serum and tissue extracts. *Nature Protocols* **2007**, *2*, 2692–2703.

57. Pudakalakatti, S.M.; Dubey, A.; Jaipuria, G.; Shubhashree, U.; Adiga, S.K.; Moskau, D.; Atreya, H.S. A fast NMR method for resonance assignments: Application to metabolomics. *Journal of Biomolecular NMR* **2014**, *58*, 165–173.

58. Ludwig, C.; Viant, M.R. Two-dimensional j-resolved NMR spectroscopy: Review of a key methodology in the metabolomics toolbox. *Phytochemical Analysis* **2010**, *21*, 22–32.

59. Gómez, J.; Brezmes, J.; Mallol, R.; Rodríguez, M.A.; Vinaixa, M.; Salek, R.M.; Correig, X.; Cañellas, N. Dolphin: A tool for automatic targeted metabolite profiling using 1D and 2D (1)H-NMR data. *Analytical and Bioanalytical Chemistry* **2014**, *406*, 7967–7976.

60. Shinzawa, H.; Nishida, M.; Kanematsu, W.; Tanaka, T.; Suzuki, K.; Noda, I. Parallel factor (PARAFAC) kernel analysis of temperature- and composition-dependent NMR spectra of poly(lactic acid) nanocomposites. *The Analyst* **2012**, *137*, 1913–1921.

61. Chen, K.; Park, J.; Li, F.; Patil, S.M.; Keire, D.A. Chemometric methods to quantify 1D and 2D NMR spectral differences among similar protein therapeutics. *AAPS PharmSciTech* **2018**, *19*, 1011–1019.

62. Pedersen, H.T.; Dyrby, M.; Engelsen, S.; Bro, R. Application of multi-way analysis to 2D NMR data. In; Annual reports on NMR spectroscopy; Elsevier, 2006; Vol. 59, pp. 207–233 ISBN 9780125054591.

63. Hao, J.; Astle, W.; De Iorio, M.; Ebbels, T.M.D. BATMAN–an r package for the automated quantification of metabolites from nuclear magnetic resonance spectra using a bayesian model. *Bioinformatics* **2012**, *28*, 2088–2090.

64. Bioconductor Bioconductor - BiocViews: packages found under StatisticalMethod.

65. Groemping, U. CRAN task view: Design of experiments (DoE) & analysis of experimental data 2019.

66. Leisch, F.; Gruen, B. CRAN task view: Cluster analysis & finite mixture models 2019.

67. Hewson, P. CRAN task view: Multivariate statistics 2018.

68. Hothorn, T. CRAN task view: Machine learning & statistical learning 2019.

69. The Comprehensive R Archive Network CRAN task views.

70. Bishop, C.M. *Pattern recognition and machine learning (information science and statistics)*; Springer: New York, 2006; p. 738; ISBN 978-0387310732.

71. Brusco, M.J.; Shireman, E.; Steinley, D. A comparison of latent class, k-means, and k-median methods for clustering dichotomous data. *Psychological methods* **2017**, *22*, 563–580.

72. Felici, G. *Mathematical methods for knowledge discovery and data mining*; Idea Group Reference: Hershey, 2007; p. 371; ISBN 978-1599045283.

73. *Introduction to multivariate analysis*; Routledge, 2018; ISBN 9780203749999.

74. Manly, B.F. *Multivariate statistical methods*; 4th ed.; Routledge: Boca Raton, 2017; p. 270; ISBN 9781498728966.

75. Müllner, D. Modern hierarchical, agglomerative clustering algorithms. *arXiv* **2011**, *abs/1109.2378*.

76. Murtagh, F.; Contreras, P. Algorithms for hierarchical clustering: An overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* **2012**, *2*, 86–97.

77. Shaw, P.J.A. *Multivariate statistics for the environmental sciences (mathematics)*; 1st ed.; Hodder Education Publishers: London, 2003; p. 233; ISBN 0-340-80763-6.

78. Zaslavsky, L.; Ciufo, S.; Fedorov, B.; Tatusova, T. Clustering analysis of proteins from microbial genomes at multiple levels of resolution. *BMC Bioinformatics* **2016**, *17 Suppl 8*, 276.

79. Hall; Robert, D. *Annual plant reviews, biology of plant metabolomics*; 1st ed.; Wiley, John & Sons, Incorporated, 2011; p. 448; ISBN 978-1-4051-9954-4.

80. Cai, Q.; Alvarez, J.A.; Kang, J.; Yu, T. Network marker selection for untargeted LC-MS metabolomics data. *Journal of Proteome Research* **2017**, *16*, 1261–1269.

81. Christin, C.; Hoefsloot, H.C.J.; Smilde, A.K.; Hoekman, B.; Suits, F.; Bischoff, R.; Horvatovich, P. A critical assessment of feature selection methods for biomarker discovery in clinical proteomics. *Molecular & Cellular Proteomics* **2013**, *12*, 263–276.

82. Rohart, F.; Gautier, B.; Singh, A.; Lê Cao, K.-A. mixOmics: An r package for 'omics feature selection and multiple data integration. *PLoS Computational Biology* **2017**, *13*, e1005752.

83. Wen, B.; Mei, Z.; Zeng, C.; Liu, S. metaX: A flexible and comprehensive software for processing metabolomics data. *BMC Bioinformatics* **2017**, *18*, 183.

84. Peters, K.; Worrich, A.; Weinhold, A.; Alka, O.; Balcke, G.; Birkemeyer, C.; Bruelheide, H.; Calf, O.W.; Dietz, S.; Dührkop, K. et al. Current challenges in plant eco-metabolomics. *International Journal of Molecular Sciences* **2018**, *19*.

85. Gromski, P.S.; Xu, Y.; Correa, E.; Ellis, D.I.; Turner, M.L.; Goodacre, R. A comparative investigation of modern feature selection and classification approaches for the analysis of mass spectrometry data. *Analytica Chimica Acta* **2014**, *829*, 1–8.

86. Legendre, P.; Legendre, L.F.J. *Numerical ecology, volume 24 (developments in environmental modelling)*; 3rd ed.; Elsevier, 2012; p. 1006; ISBN 978-0-444-53868-0.

87. Clarke, B.; Fokoue, E.; Zhang, H.H. *Principles and theory for data mining and machine learning*; Springer series in statistics; Springer New York: New York, NY, 2009; ISBN 978-0-387-98134-5.

88. Feng, J.; Li, B.; Jiang, X.; Yang, Y.; Wells, G.F.; Zhang, T.; Li, X. Antibiotic resistome in a large-scale healthy human gut microbiota deciphered by metagenomic and network analyses. *Environmental Microbiology* **2018**, *20*, 355–368.

89. Fukushima, A.; Kusano, M.; Redestig, H.; Arita, M.; Saito, K. Integrated omics approaches in plant systems biology. *Current Opinion in Chemical Biology* **2009**, *13*, 532–538.

90. Vaughan, A.A.; Dunn, W.B.; Allwood, J.W.; Wedge, D.C.; Blackhall, F.H.; Whetton, A.D.; Dive, C.; Goodacre, R. Liquid chromatography-mass spectrometry calibration transfer and metabolomics data fusion. *Analytical Chemistry* **2012**, *84*, 9848–9857.

91. Wei, R.; Wang, J.; Su, M.; Jia, E.; Chen, S.; Chen, T.; Ni, Y. Missing value imputation approach for mass spectrometry-based metabolomics data. *Scientific reports* **2018**, *8*, 663.

92. Degenhardt, F.; Seifert, S.; Szymczak, S. Evaluation of variable selection methods for random forests and omics data sets. *Briefings in Bioinformatics* **2019**, *20*, 492–503.

93. Saeys, Y.; Inza, I.; Larrañaga, P. A review of feature selection techniques in bioinformatics. *Bioinformatics* **2007**, *23*, 2507–2517.

94. Determan Jr, C.E. Optimal algorithm for metabolomics classification and feature selection varies by dataset. *International journal of biology* **2014**, *7*.

95. Rinaudo, P.; Boudah, S.; Junot, C.; Thévenot, E.A. Biosigner: A new method for the discovery of significant molecular signatures from omics data. *Frontiers in molecular biosciences* **2016**, *3*, 26.

96. Shi, L.; Westerhuis, J.A.; Rosén, J.; Landberg, R.; Brunius, C. Variable selection and validation in multivariate modelling. *Bioinformatics* **2019**, *35*, 972–980.

97. Wehrens, R.; Franceschi, P. Meta-statistics for variable selection: TheR PackageBioMark. *Journal of statistical software* **2012**, *51*.

98. Li, S.; Park, Y.; Duraisingham, S.; Strobel, F.H.; Khan, N.; Soltow, Q.A.; Jones, D.P.; Pulendran, B. Predicting network activity from high throughput metabolomics. *PLoS Computational Biology* **2013**, *9*, e1003123.

99. Willighagen, E.L.; Mayfield, J.W.; Alvarsson, J.; Berg, A.; Carlsson, L.; Jeliazkova, N.; Kuhn, S.; Pluskal, T.; Rojas-Chertó, M.; Spjuth, O. et al. The chemistry development kit (CDK) v2.0: Atom typing, depiction, molecular formulas, and substructure searching. *Journal of cheminformatics* **2017**, *9*, 33.

100. Heller, S.R.; McNaught, A.; Pletnev, I.; Stein, S.; Tchekhovskoi, D. Inchi, the IUPAC international chemical identifier. *Journal of cheminformatics* **2015**, *7*, 23.

101. Backman, T.W.H.; Cao, Y.; Girke, T. ChemMine tools: An online service for analyzing and clustering small molecules. *Nucleic Acids Research* **2011**, *39*, W486–91.

102. O'Boyle, N.M.; Banck, M.; James, C.A.; Morley, C.; Vandermeersch, T.; Hutchison, G.R. Open babel: An open chemical toolbox. *Journal of cheminformatics* **2011**, *3*, 33.

103. Landrum, G. RDKit: Open-source cheminformatics software. **2016**.

104. Kim, S.; Thiessen, P.A.; Bolton, E.E.; Chen, J.; Fu, G.; Gindulyte, A.; Han, L.; He, J.; He, S.; Shoemaker, B.A. et al. PubChem substance and compound databases. *Nucleic Acids Research* **2016**, *44*, D1202–13.

105. Wang, Y.; Xiao, J.; Suzek, T.O.; Zhang, J.; Wang, J.; Bryant, S.H. PubChem: A public information system for analyzing bioactivities of small molecules. *Nucleic Acids Research* **2009**, *37*, W623–33.

106. Pence, H.E.; Williams, A. Chemspider: An online chemical information resource. *Journal of chemical education* **2010**, *87*, 1123–1124.

107. Erxleben, F.; Günther, M.; Krötzsch, M.; Mendez, J.; Vrandečić, D. Introducing wikidata to the linked data web. In *The semantic web – ISWC 2014*; Mika, P., Tudorache, T., Bernstein, A., Welty, C., Knoblock, C., Vrandečić, D., Groth, P., Noy, N., Janowicz, K., Goble, C., Eds.; Lecture notes in computer science; Springer International Publishing: Cham, 2014; Vol. 8796, pp. 50–65 ISBN 978-3-319-11963-2.

108. Wohlgemuth, G.; Haldiya, P.K.; Willighagen, E.; Kind, T.; Fiehn, O. The chemical translation service–a web-based tool to improve standardization of metabolomic reports. *Bioinformatics* **2010**, *26*, 2647–2648.

109. SRC, Inc. Scientific databases.

110. Group NCI/CADD chemical identifier resolver.

111. Djoumbou Feunang, Y.; Eisner, R.; Knox, C.; Chepelev, L.; Hastings, J.; Owen, G.; Fahy, E.; Steinbeck, C.; Subramanian, S.; Bolton, E. et al. ClassyFire: Automated chemical classification with a comprehensive, computable taxonomy. *Journal of cheminformatics* **2016**, *8*, 61.

112. Watrous, J.; Roach, P.; Alexandrov, T.; Heath, B.S.; Yang, J.Y.; Kersten, R.D.; Voort, M. van der; Pogliano, K.; Gross, H.; Raaijmakers, J.M. et al. Mass spectral molecular networking of living microbial colonies. *Proceedings of the National Academy of Sciences of the United States of America* **2012**, *109*, E1743–52.

113. Kanehisa, M.; Furumichi, M.; Tanabe, M.; Sato, Y.; Morishima, K. KEGG: New perspectives on genomes, pathways, diseases and drugs. *Nucleic Acids Research* **2017**, *45*, D353–D361.

114. Kanehisa, M.; Sato, Y.; Furumichi, M.; Morishima, K.; Tanabe, M. New approach for understanding genome variations in KEGG. *Nucleic Acids Research* **2019**, *47*, D590–D595.

115. Kanehisa, M.; Goto, S. KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Research* **2000**, *28*, 27–30.

116. Lim, E.; Pon, A.; Djoumbou, Y.; Knox, C.; Shrivastava, S.; Guo, A.C.; Neveu, V.; Wishart, D.S. T3DB: A comprehensively annotated database of common toxins and their targets. *Nucleic Acids Research* **2010**, *38*, D781–6.

117. Wishart, D.; Arndt, D.; Pon, A.; Sajed, T.; Guo, A.C.; Djoumbou, Y.; Knox, C.; Wilson, M.; Liang, Y.; Grant, J. et al. T3DB: The toxic exposome database. *Nucleic Acids Research* **2015**, *43*, D928–34.

118. Fahy, E.; Sud, M.; Cotter, D.; Subramaniam, S. LIPID MAPS online tools for lipid research. *Nucleic Acids Research* **2007**, *35*, W606–12.

119. Shen, X.; Wang, R.; Xiong, X.; Yin, Y.; Cai, Y.; Ma, Z.; Liu, N.; Zhu, Z.-J. Metabolic reaction network-based recursive metabolite annotation for untargeted metabolomics. *Nature Communications* **2019**, *10*, 1516.

120. Schaefer, C.F.; Anthony, K.; Krupa, S.; Buchoff, J.; Day, M.; Hannay, T.; Buetow, K.H. PID: The pathway interaction database. *Nucleic Acids Research* **2009**, *37*, D674–9.

121. Nishimura, D. BioCarta. *Biotech Software & Internet Report* **2001**, *2*, 117–120.

122. Fabregat, A.; Jupe, S.; Matthews, L.; Sidiropoulos, K.; Gillespie, M.; Garapati, P.; Haw, R.; Jassal, B.; Korninger, F.; May, B. et al. The reactome pathway knowledgebase. *Nucleic Acids Research* **2018**, *46*, D649–D655.

123. Kramer, F.; Bayerlová, M.; Beißbarth, T. R-based software for the integration of pathway data into bioinformatic algorithms. *Biology* **2014**, *3*, 85–100.

124. Tenenbaum, D. Bioconductor - KEGGREST 2019.

125. Chang, W.; Cheng, J.; Allaire, J.; Xie, Y.; McPherson, J. Shiny: Web application framework for r 2012.

126. Rocca-Serra, P.; Brandizi, M.; Maguire, E.; Sklyar, N.; Taylor, C.; Begley, K.; Field, D.; Harris, S.; Hide, W.; Hofmann, O. et al. ISA software suite: Supporting standards-compliant experimental annotation and enabling curation at the community level. *Bioinformatics* **2010**, *26*, 2354–2356.

127. Sansone, S.-A.; Rocca-Serra, P.; Field, D.; Maguire, E.; Taylor, C.; Hofmann, O.; Fang, H.; Neumann, S.; Tong, W.; Amaral-Zettler, L. et al. Toward interoperable bioscience data. *Nature Genetics* **2012**, *44*, 121–126.

128. Plotly Technologies Inc Collaborative data science 2015.

129. Wickham, H. *Ggplot2: Elegant graphics for data analysis*; Springer International Publishing : 2016; ISBN 978-3-319-24277-4.

130. Jupyter, P.; Bussonnier, M.; Forde, J.; Freeman, J.; Granger, B.; Head, T.; Holdgraf, C.; Kelley, K.; Nalvarte, G.; Osheroff, A. et al. Binder 2.0 - reproducible, interactive, sharable environments for science at scale. In Proceedings of the Proceedings of the 17th python in science conference; SciPy, 2018; pp. 113–120.

131. Ram, K. Configure your r project for binderhub • hole punch.

132. Liggi, S.; Hinz, C.; Hall, Z.; Santoru, M.L.; Poddighe, S.; Fjeldsted, J.; Atzori, L.; Griffin, J.L. KniMet: A pipeline for the processing of chromatography-mass spectrometry metabolomics data. *Metabolomics : Official journal of the Metabolomic Society* **2018**, *14*, 52.

133. Verhoeven, A.; Giera, M.; Mayboroda, O.A. KIMBLE: A versatile visual NMR metabolomics workbench in KNIME. *Analytica Chimica Acta* **2018**, *1044*, 66–76.

134. Davidson, R.L.; Weber, R.J.M.; Liu, H.; Sharma-Oates, A.; Viant, M.R. Galaxy-m: A galaxy workflow for processing and analyzing direct infusion and liquid chromatography mass spectrometry-based metabolomics data. *GigaScience* **2016**, *5*, 10.

135. Giacomoni, F.; Le Corguillé, G.; Monsoor, M.; Landi, M.; Pericard, P.; Pétéra, M.; Duperier, C.; Tremblay-Franco, M.; Martin, J.-F.; Jacob, D. et al. Workflow4Metabolomics: A collaborative research infrastructure for computational metabolomics. *Bioinformatics* **2015**, *31*, 1493–1495.

136. Goecks, J.; Nekrutenko, A.; Taylor, J.; Team, G. Galaxy: A comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome Biology* **2010**, *11*, R86.

137. Metabohub National infrastructure in metabolomics and fluxomics 2019.

138. Guitton, Y.; Tremblay-Franco, M.; Le Corguillé, G.; Martin, J.-F.; Pétéra, M.; Roger-Mele, P.; Delabrière, A.; Goulitquer, S.; Monsoor, M.; Duperier, C. et al. Create, run, share, publish, and reference your LC-MS, FIA-MS, GC-MS, and NMR data analysis workflows with the Workflow4Metabolomics 3.0 galaxy online infrastructure for metabolomics. *The International Journal of Biochemistry & Cell Biology* **2017**, *93*, 89–101.

139. Workflow4metabolomics Referenced W4M histories workflow4metabolomics.org.

140. Goble, C.; Cohen-Boulakia, S.; Soiland-Reyes, S.; Garijo, D.; Gil, Y.; Crusoe, M.R.; Peters, K.; Schober, D. FAIR computational workflows. *Zenodo* **2019**.

141. Wilkinson, M.D.; Dumontier, M.; Aalbersberg, I.J.J.; Appleton, G.; Axton, M.; Baak, A.; Blomberg, N.; Boiten, J.-W.; Silva Santos, L.B. da; Bourne, P.E. et al. The FAIR guiding principles for scientific data management and stewardship. *Scientific data* **2016**, *3*, 160018.

142. Haug, K.; Salek, R.M.; Conesa, P.; Hastings, J.; Matos, P. de; Rijnbeek, M.; Mahendraker, T.; Williams, M.; Neumann, S.; Rocca-Serra, P. et al. MetaboLights–an open-access general-purpose repository for metabolomics studies and associated meta-data. *Nucleic Acids Research* **2013**, *41*, D781–6.

143. Sud, M.; Fahy, E.; Cotter, D.; Azam, K.; Vadivelu, I.; Burant, C.; Edison, A.; Fiehn, O.; Higashi, R.; Nair, K.S. et al. Metabolomics workbench: An international repository for metabolomics data and metadata, metabolite standards, protocols, tutorials and training, and analysis tools. *Nucleic Acids Research* **2016**, *44*, D463–70.

144. Gray, A.J.G.; Goble, C.A.; Jimenez, R. Bioschemas: From potato salad to protein annotation. In Proceedings of the Proceedings of the ISWC 2017 posters & demonstrations and industry tracks co-located with 16th international semantic web conference (ISWC 2017), vienna, austria, october 23rd - to - 25th, 2017.; Nikitina, N., Song, D., Fokoue, A., Haase, P., Eds.; CEUR-WS.org, 2017; Vol. 1963.

145. Collberg, C.; Proebsting, T.A. Repeatability in computer systems research. *Communications of the ACM* **2016**, *59*, 62–69.

146. Taschuk, M.; Wilson, G. Ten simple rules for making research software more robust. *PLoS Computational Biology* **2017**, *13*, e1005412.

147. Stanstrup, J.; Broeckling, C.D.; Helmus, R.; Hoffmann, N.; Mathé, E.; Naake, T.; Nicolotti, L.; Peters, K.; Rainer, J.; Salek, R. et al. The MetaRbolomics book.

Appendices

Appendix 1: The MSP File Format and package support

```
Name: unknown
Num Peaks: 2
85.345 100; 76.321 50;
```

Listing S1: Example for the basic NIST format.

```
Name: 1-Methylhistidine
Synon: (2S)-2-amino-3-(1-methyl-1H-imidazol-4-
yl)propanoic acid
SYNON: $:00in-source
DB#: HMDB0000001_c_ms_1469
InChIKey:
BRMWTNUJHUMWMS-LURJTMIESA-N
Instrument_type: GC-MS
Retention_index: 1807.71
Formula: C7H11N3O2
MW: 169
ExactMass: 169.0851
Comments:
"column=5%-phenyl-95%-dimethylpolysiloxane
capillary column" "derivatization type=2 TMS"
"derivatization formula=C13H27N3O2Si2"
"derivative mw=313.544" "retention
index=1807.71" "retention index type=based on 9
n-alkanes (C10–C36)" "instrument type=GC-MS"
"chromatography type=GC" "cas
number=332-80-9" "molecular
formula=C7H11N3O2" "total exact
mass=169.085126592"
"InChIKey=BRMWTNUJHUMWMS-
LURJTMIESA-N"
Num Peaks: 10
70 0.014; 71 0.007; 72 0.02; 76 0.008; 77 0.008;
78 0.002; 79 0.003; 80 0.005; 81 0.108; 82 0.017;
```

Listing S2: Example for the canonical NIST format.

```
NAME: Aspartame; LC-ESI-ITFT; MS2; CE
PRECURSORMZ: 295.128848
PRECURSORTYPE: [M+H]+
INSTRUMENTTYPE: LC-ESI-ITFT
SMILES:
COC(=O)C(CC1=CC=CC=C1)N=C(O)C(N)CC(O)=O
INCHIKEY:
IAOZJIPTCAWIRG-UHFFFAOYNA-N
Ontology: Peptides
COLLISIONENERGY: 35
FORMULA: C14H18N2O5
RETENTIONTIME:
IONMODE: Positive
Comment: registered in MassBank
Num Peaks: 9
120.0804 13
180.10201 138
217.0968 14
235.10789 390
245.0921 274
260.09171 132
263.1026 286
277.11859 1000
278.1022 28
```

Listing S3: RIKEN PRIME msp format example.

Table S1: Overview of MS/MS handling in different R packages. '-' means not available, for the remaining entries see the text above.

package	read msp	write msp	spectral matching and additional information
baitmet			N vs DB; cosine, Stein & Scott composite similarity product
compMS2Miner	NIST, RIKEN PRIME msp	RIKEN PRIME msp	N vs DB; dot product
enviGCMS		basic NIST	
erah	NIST	only result export	N vs DB; cosine
flagme		only result export	
metaMS	NIST	NIST; slow	1 vs DB, N vs DB; proprietary
MatchWeiz			N vs DB; X-Rank
MetCirc			N vs N; normalized dot product; will switch to MSnbase functions soon
MSeasy		only result export	N vs DB; Queries the NIST mass spectral search tool
MSnbase	**	**	1 vs 1, N vs N; dot product and more, user def.
msPurity			N vs DB; dot product
OrgMassSpecR	basic NIST	basic NIST	1 vs 1; normalized dot product
RAMClustR			RAMClustR can import and utilize spectrum similarities from MS-FINDER;
rTANDEM			N vs DB; dot product; R wrapper for X!Tandem software
SwathXtend	- (PeakView / OpenSWATH)	- (PeakView / OpenSWATH)	
TargetSearch	NIST (with error)	NIST	N vs DB; RI-based

Appendix 2: metaRbolomics dependencies network

Libraries and settings

```
options("repos" = list(CRAN="http://cran.rstudio.com/"))

library(devtools)   # for revdp()
library(igraph)     # for graph_from_edgelist/( and simplify() )
library(visNetwork) # for visNetwork() and friends
library(networkD3)  # for saveNetwork()
library(chromote)   # for default_chromote_object()
library(webshot2)   # for webshot()
library(png)        # For displaying an image
library(dplyr)
library(purrr)

source("scripts/revDepNetHelper.R")

set_default_chromote_object(Chromote$new(browser = Chrome$new(args = "--no-sandbox")))
```

Read package names from our table

```
reviewTables <- read.delim("public/data/AllMetaRbolomicsTables.csv", stringsAsFactors = FALSE)
reviewPkgs <- reviewTables[,"Package"]

pkgs <- reviewPkgs
```

Get reverse dependencies

3.0.0.1 For CRAN and BioC packages

```
el <- sapply(pkgs, function(pkg) {
  rd <- revdep(pkg, dependencies = c("Depends", "Imports", "LinkingTo"),
               recursive = FALSE, ignore = NULL, bioconductor = TRUE)
  as.matrix(cbind(Package=rep(pkg, length.out=length(rd)), ReverseDep=rd))
})
el <- do.call(rbind, el)
```

3.0.0.2 For GitHub and GitLab

The above `devtools::revdep` cannot read from GitHub/GitLab repositories. We have a helper function that downloads and parses the DESCRIPTION file from GitHub/GitLab. Since we cannot get reverse dependencies directly for GitHub/GitLab packages, those packages they are only used as additional reverse depedencies for the CRAN/BioC packages.

```
gitdeps_reverse <- reviewTables %>%
                      mutate(dep_tree = map(Code_link, get_git_deps)) %>%
                      pull(dep_tree) %>%
                      bind_rows() %>%
                      filter(Dep %in% el[,"Package"]) %>%
                      rename(Package = Dep, ReverseDep = Package) %>%
                      as.matrix()
```

```
## Warning in readLines(file): incomplete final line found on '/tmp/
## RtmpQa3zna/file3aef6af8ed62'

## Warning in readLines(file): incomplete final line found on '/tmp/
## RtmpQa3zna/file3aef57897b0e'
```

```
el <- rbind(el, gitdeps_reverse)
```

Building dependency network

In total, we were analysing 296 packages. For each package, this returns the set of packages in CRAN or BioC that depend on, import from or link to the package (i.e., its direct reverse dependencies) using the `devtools::revdep()` function. A few packages with the highest number of reverse dependencies have been excluded, as they would dominate the visualisation. It was not possible to detect reverse dependencies from other hosting places such as GitHub or GitLab.

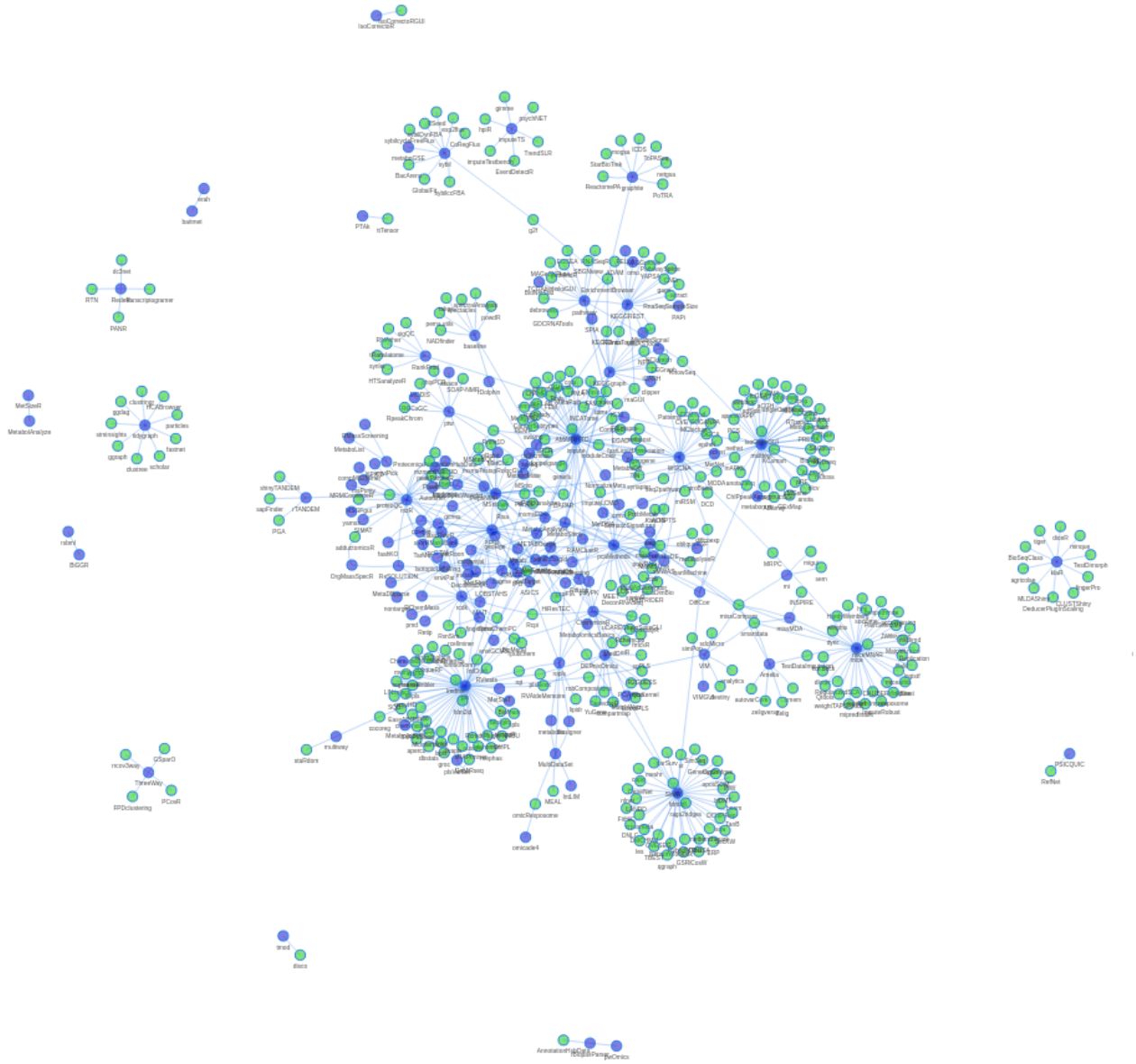
From the total, 68 packages had at least one such reverse dependency.

```
## Remove packages with most reverse dependencies
## which would dominate the network

el <- el[! el[,"Package"] %in% c("Rcpp", "igraph", "vegan", "caret", "rJava", "reticulate"), ]

## Create graph, and simplify redundancy
g <- graph_from_edgelist(el, directed = TRUE)
g <- igraph::simplify(g, remove.multiple = TRUE, remove.loops = TRUE)

# get data and plot :
data <- toVisNetworkData(g)

data$nodes <- cbind(data$nodes,
                    font.size=30,
                    color.background = ifelse(data$nodes[,"id"] %in% pkgs ,
                           rgb(0, 0, 200, 128, max = 255),
                           rgb(0, 200, 0, 128, max = 255)))

vn <- visNetwork(nodes = data$nodes,
                 edges = data$edges,
                 width=1000, height=1000) %>%
  visPhysics(timestep = 0.3,
             barnesHut = list(centralGravity=0.35,
                              springLength = 95)) %>%
  visOptions(highlightNearest = TRUE)

vn
```

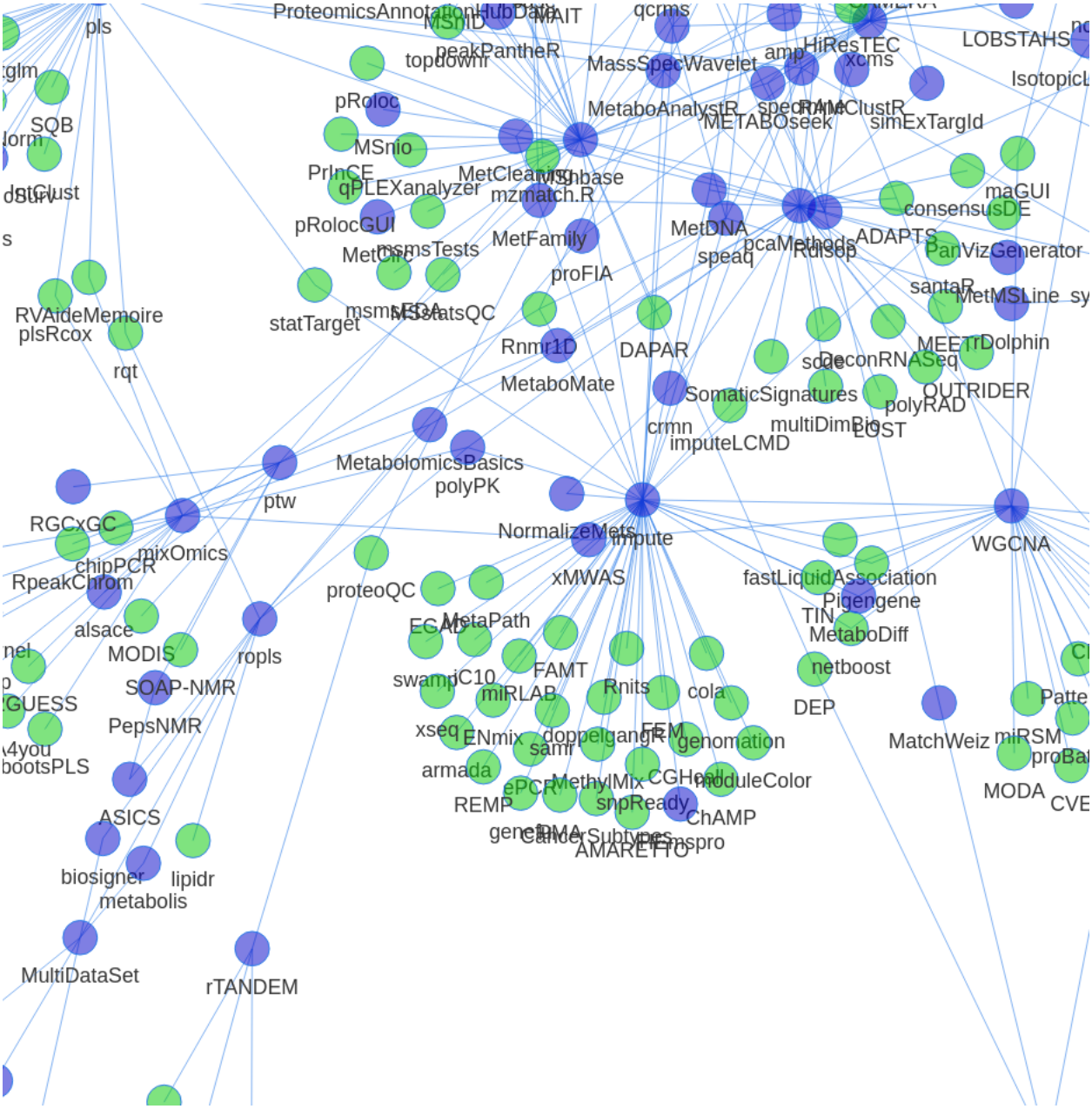
Figure S1: Dependency network of R packages. Shown in blue are packages mentioned in the review. Edges connect to packages that depend on another package, as long as that is in CRAN or BioC. Green nodes correspond to packages in CRAN or BioC not covered in the review. Not shown are 1) infrastructure packages e.g. rJava, Rcpp 2) packages from the review without reverse dependencies and 3) data packages. Some packages from the review are not in current versions of CRAN or BioC. An interactive version of this figure is available from https://stanstrup.gitlab.io/metaRbolomics-book/appendix-2-metarbolomics-dependencies-network.html.

Save network plot

```
saveNetwork(vn, "vn.html")
webshot("vn.html", "revDepNet-60.png", delay = 60)
```



```
vnZoom <- visNetwork(nodes = data$nodes,
                     edges = data$edges,
                     width=1000, height=1000) %>%
    visIgraphLayout()%>%
    visEvents(type="once", startStabilizing = 'function() {
              this.fit({nodes:["ptw", "Rnmr1D", "RpeakChrom", "alsace",
                               "PepsNMR", "ASICS", "MODIS", "RGCxGC"]})
            }')
saveNetwork(vnZoom, "vnZoom.html")
webshot("vnZoom.html", "revDepNet-zoom.png", delay = 5)
```



You ca access the files at:

- vn.html
- revDepNet-60.png
- vnZoom.html
- revDepNet-zoom.png

Notes

The source code for this page is on GitHub at gitlab.com/stanstrup/metaRbolomics-book

The HTML output is shown at https://stanstrup.gitlab.io/metaRbolomics-book/appendix-2-metarbolomics-dependencies-network.html

and https://stanstrup.gitlab.io/metaRbolomics-book/vn.html (Caveat: long rendering time, blank page without any visible progress)

This page was created with the following packages:

```
sessionInfo()
```

```
## R version 3.6.1 (2017-01-27)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: Ubuntu 16.04.6 LTS
##
## Matrix products: default
## BLAS:   /home/travis/R-bin/lib/R/lib/libRblas.so
## LAPACK: /home/travis/R-bin/lib/R/lib/libRlapack.so
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] desc_1.2.0          png_0.1-7           webshot2_0.0.0.9000
##  [4] chromote_0.0.0.9001 networkD3_0.4       visNetwork_2.0.8
##  [7] igraph_1.2.4.1      devtools_2.2.0      usethis_1.5.1
## [10] tikzDevice_0.12.3   purrr_0.3.2         kableExtra_1.1.0
## [13] DT_0.9              dplyr_0.8.3         googlesheets_0.3.0
## [16] readr_1.3.1         knitr_1.24
##
## loaded via a namespace (and not attached):
##  [1] httr_1.4.1          pkgload_1.0.2       jsonlite_1.6
##  [4] viridisLite_0.3.0   shiny_1.3.2         assertthat_0.2.1
##  [7] BiocManager_1.30.4  cellranger_1.1.0    yaml_2.2.0
## [10] remotes_2.1.0       sessioninfo_1.1.1   pillar_1.4.2
## [13] backports_1.1.4     glue_1.3.1          digest_0.6.20
## [16] promises_1.0.1.9002 rvest_0.3.4         colorspace_1.4-1
## [19] websocket_1.1.0     htmltools_0.3.6     httpuv_1.5.2
## [22] pkgconfig_2.0.2     bookdown_0.13.2     xtable_1.8-4
## [25] scales_1.0.0        webshot_0.5.1       processx_3.4.1
## [28] later_0.8.0.9004    tibble_2.1.3        ellipsis_0.2.0.1
## [31] withr_2.1.2         cli_1.1.0           magrittr_1.5
## [34] crayon_1.3.4        mime_0.7            memoise_1.1.0
## [37] evaluate_0.14       ps_1.3.0            fs_1.3.1
## [40] xml2_1.2.2          pkgbuild_1.0.5      tools_3.6.1
## [43] prettyunits_1.0.2   hms_0.5.1           stringr_1.4.0
## [46] munsell_0.5.0       callr_3.3.1         compiler_3.6.1
## [49] rlang_0.4.0         grid_3.6.1          rstudioapi_0.10
## [52] htmlwidgets_1.3     filehash_2.4-2      crosstalk_1.0.0
## [55] rmarkdown_1.15      testthat_2.2.1      codetools_0.2-16
## [58] curl_4.0            R6_2.4.0            fastmap_1.0.0
```

```
## [61] zeallot_0.1.0       rprojroot_1.3-2     stringi_1.4.3
## [64] Rcpp_1.0.2          vctrs_0.2.0         tidyselect_0.2.5
## [67] xfun_0.9
```