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

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.

Contributing 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

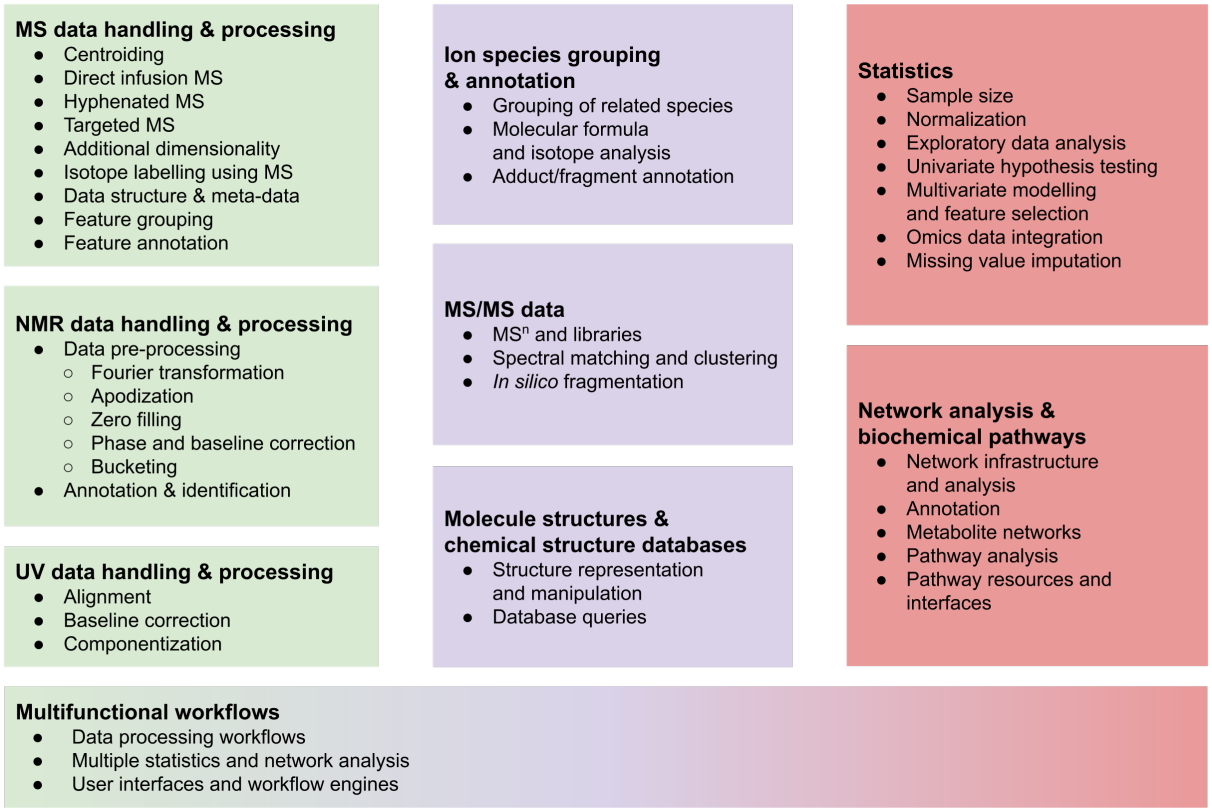


Figure 1: Overview of typical tasks in metabolomics workflows, ranging from metabolite profiling (left, green) via metabolite annotation (center, purple) to data analysis using statistics and metabolite networks (right, red).

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

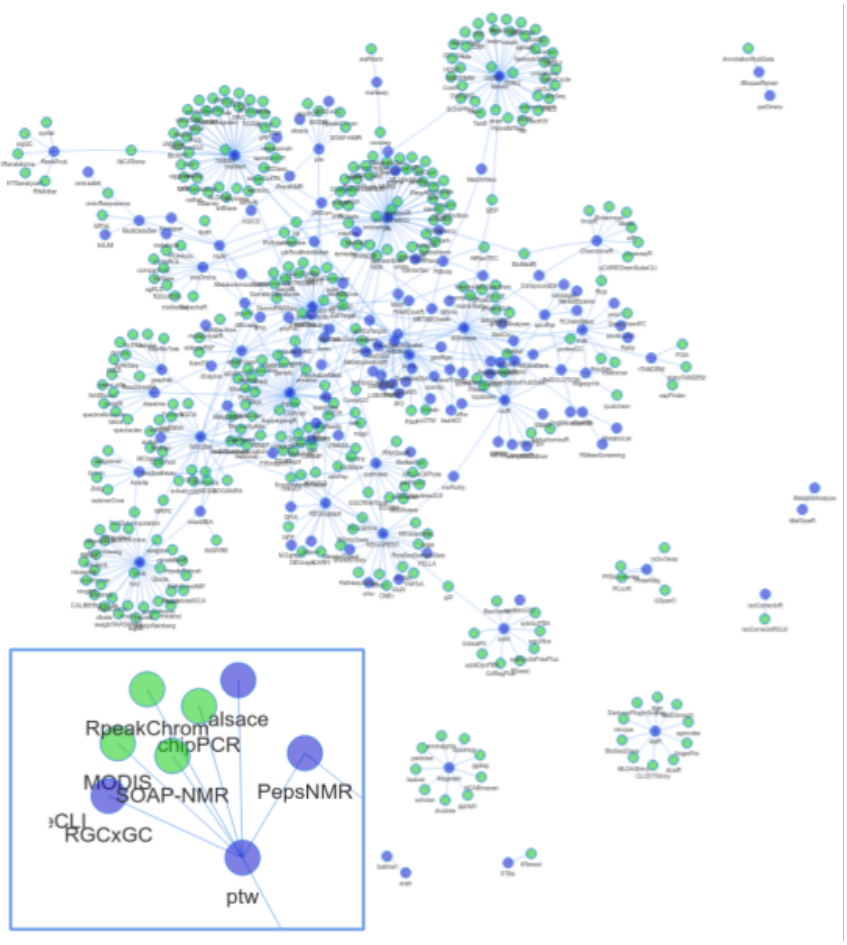


Figure 2: Dependency network of R packages. Shown in blue are packages mentioned in the review. Edges connect to packages that depend on another package, as long as they are in CRAN or BioC. Green nodes correspond to packages in CRAN or BioC not covered in the review. The inset shows the neighbourhood 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

2.1 Mass spectrometry data handling and (pre-)processing

2.1.1 Profile mode and centroided data

2.1.2 Direct infusion mass spectrometry data

2.1.3 Hyphenated MS and non-targeted data

2.1.4 Targeted data and alternative representations of data

2.1.5 Additional dimensionality

2.1.6 Structuring data and metadata

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
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

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

Functionalities	Package	Repo
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
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

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

Functionalities	Package	Repo
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
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		

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

Functionalities	Package	Repo
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

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

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

Functionalities	Package	Repo
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
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

2.2 Metabolite identification with MS/MS data

2.2.1 MS/MS data handling, spectral matching and clustering

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
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

2.3 NMR data handling and (pre-)processing

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
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

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

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
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		

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

Functionalities	Package	Repo
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
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

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

Functionalities	Package	Repo
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		
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

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

Functionalities	Package	Repo
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
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

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

Functionalities	Package	Repo
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

2.6 Handling of molecule structures and chemical structure databases

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
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

2.7.1 Network infrastructure and analysis

2.7.2 Metabolite annotation

2.7.3 Generation of metabolic networks

2.7.4 Pathway analysis

2.7.5 Pathway resources and interfaces

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
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

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

Functionalities	Package	Repo
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

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, probabilistic putative metabolite annotation.	mzMatch	SF
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
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

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

Functionalities	Package	Repo
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

Table 10: 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

Table 11: 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

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.

4 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

4.0.0.1 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)
```

4.0.0.2 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/
## RtmpKYNMBZ/file3bd639cc31cb'
```

```
## Warning in readLines(file): incomplete final line found on '/tmp/
## RtmpKYNMBZ/file3bd6398fe9f5'
```

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

Building dependency network

In total, we were analysing 292 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.

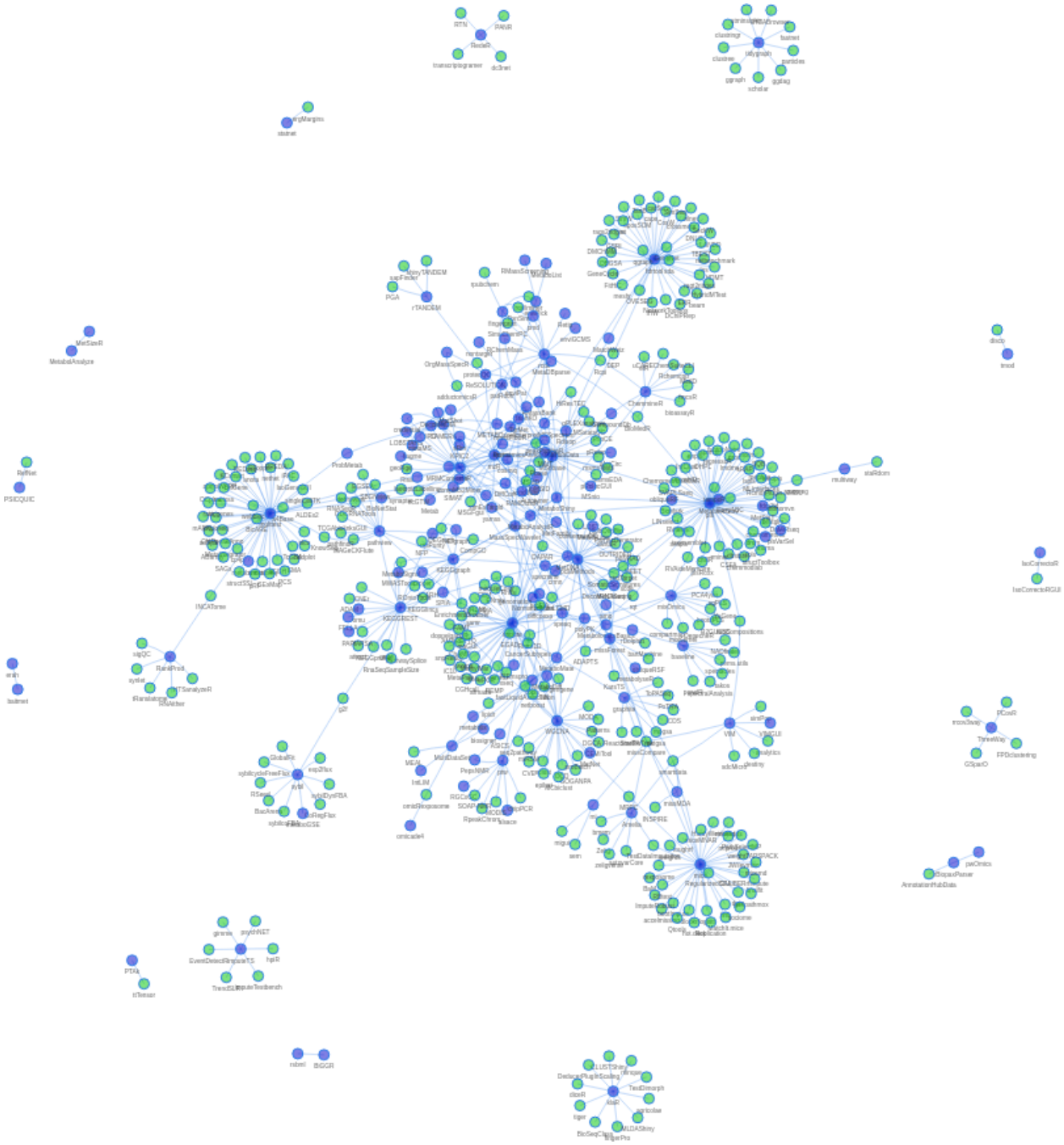
From the total, 68 packages had at least one such reverse dependency.

```
## 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.

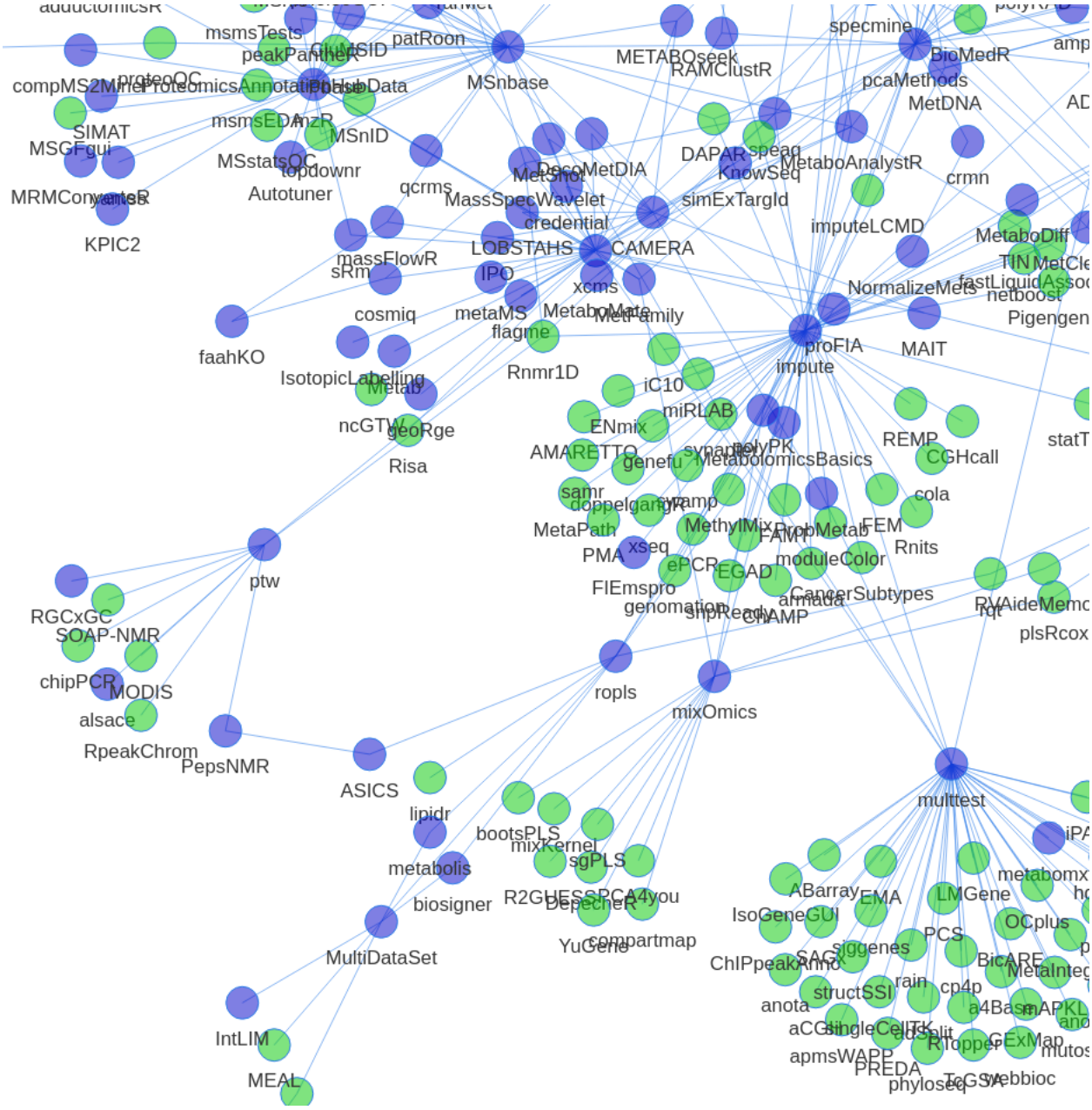
Save network plot

```
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_0.8              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
```