



Shuzhao Li *Editor*

Computational Methods and Data Analysis for Metabolomics

METHODS IN MOLECULAR BIOLOGY

Series Editor

John M. Walker

School of Life and Medical Sciences

University of Hertfordshire

Hatfield, Hertfordshire, UK

For further volumes:
<http://www.springer.com/series/7651>

For over 35 years, biological scientists have come to rely on the research protocols and methodologies in the critically acclaimed *Methods in Molecular Biology* series. The series was the first to introduce the step-by-step protocols approach that has become the standard in all biomedical protocol publishing. Each protocol is provided in readily-reproducible step-by-step fashion, opening with an introductory overview, a list of the materials and reagents needed to complete the experiment, and followed by a detailed procedure that is supported with a helpful notes section offering tips and tricks of the trade as well as troubleshooting advice. These hallmark features were introduced by series editor Dr. John Walker and constitute the key ingredient in each and every volume of the *Methods in Molecular Biology* series. Tested and trusted, comprehensive and reliable, all protocols from the series are indexed in PubMed.

Computational Methods and Data Analysis for Metabolomics

Edited by

Shuzhao Li

Department of Medicine, Emory University School of Medicine, Atlanta, GA, USA



Editor

Shuzhao Li
Department of Medicine
Emory University School of Medicine
Atlanta, GA, USA

ISSN 1064-3745

Methods in Molecular Biology

ISBN 978-1-0716-0238-6

<https://doi.org/10.1007/978-1-0716-0239-3>

ISSN 1940-6029 (electronic)

ISBN 978-1-0716-0239-3 (eBook)

© Springer Science+Business Media, LLC, part of Springer Nature 2020

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors, and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Humana imprint is published by the registered company Springer Science+Business Media, LLC part of Springer Nature.

The registered company address is: 233 Spring Street, New York, NY 10013, U.S.A.

Preface

Metabolomics is the new biochemistry. It reinvigorates the old discipline by new data at a large scale: simultaneous measurement of thousands of chemicals in biological samples. Many of these chemicals are beyond the known metabolic intermediates. This new information fills an important gap between the interactions of genome and environment, thus conferring enormous potential for improving human health. The metabolomics data also overlap significantly with the exposome, which aims to quantify all environmental exposures. The explosive growth of metabolomics creates a large gap in training on metabolomics data analysis. This book shall provide a comprehensive guide to scientists, engineers, and students that employ metabolomics in their work, with an emphasis on the understanding and interpretation of the data.

The book is organized as follows. Chapter 1 provides an overview of the field, and the following chapters are presented in four sections: data processing for major experimental platforms (Chapters 2–7), databases and metabolite annotation (Chapters 8–13), major techniques in data analysis (Chapters 14–19), and biomedical applications (Chapters 20–23).

While it is not possible to cover all the databases and software tools, we aim to have representations of each major topic and give readers a foundation to work in this field. It is critical to note that the scientific landscape keeps evolving and tools keep changing. Therefore, it is more important to understand the rationale and principles than to replicate the protocols. This book is supplemented by example data and code at GitHub (<https://metabolomics-data.github.io>), which can be continuously updated by the community.

I would like to express my gratitude to the metabolomics group at Emory University, especially Dean Jones, Tianwei Yu, Young-Mi Go, Youngja Park, Karan Uppal, Douglas Walker, and Gary Miller. Their intellectual input and friendship made my scientific journey truly rewarding.

Atlanta, GA, USA

Shuzhao Li

Contents

<i>Preface</i>	v
<i>Contributors</i>	ix
1 Overview of Experimental Methods and Study Design in Metabolomics, and Statistical and Pathway Considerations	1 <i>Stephen Barnes</i>
2 Metabolomics Data Processing Using XCMS	11 <i>Xavier Domingo-Almenara and Gary Siuzdak</i>
3 Metabolomics Data Preprocessing Using ADAP and MZmine 2	25 <i>Xiuxia Du, Aleksandr Smirnov, Tomáš Pluskal, Wei Jia, and Susan Sumner</i>
4 Metabolomics Data Processing Using OpenMS	49 <i>Marc Rurik, Oliver Alka, Fabian Aicheler, and Oliver Kohlbacher</i>
5 Analysis of NMR Metabolomics Data	61 <i>Wimal Pathmasiri, Kristine Kay, Susan McRitchie, and Susan Sumner</i>
6 Key Concepts Surrounding Studies of Stable Isotope-Resolved Metabolomics	99 <i>Stephen F. Previs and Daniel P. Downes</i>
7 Extracting Biological Insight from Untargeted Lipidomics Data	121 <i>Jennifer E. Kyle</i>
8 Overview of Tandem Mass Spectral and Metabolite Databases for Metabolite Identification in Metabolomics	139 <i>Zhangtao Yi and Zheng-Jiang Zhu</i>
9 METLIN: A Tandem Mass Spectral Library of Standards	149 <i>J. Rafael Montenegro-Burke, Carlos Guijas, and Gary Siuzdak</i>
10 Metabolomic Data Exploration and Analysis with the Human Metabolome Database	165 <i>David S. Wishart</i>
11 De Novo Molecular Formula Annotation and Structure Elucidation Using SIRIUS 4	185 <i>Marcus Ludwig, Markus Fleischauer, Kai Dührkop, Martin A. Hoffmann, and Sebastian Böcker</i>
12 Annotation of Specialized Metabolites from High-Throughput and High-Resolution Mass Spectrometry Metabolomics	209 <i>Thomas Naake, Emmanuel Gaquerel, and Alisdair R. Fernie</i>
13 Feature-Based Molecular Networking for Metabolite Annotation	227 <i>Vanessa V. Phelan</i>
14 A Bioinformatics Primer to Data Science, with Examples for Metabolomics	245 <i>W. Stephen Pittard, Cecilia "Keeko" Villaveces, and Shuzhao Li</i>
15 The Essential Toolbox of Data Science: Python, R, Git, and Docker	265 <i>W. Stephen Pittard and Shuzhao Li</i>

16	Predictive Modeling for Metabolomics Data	313
	<i>Tusharkanti Ghosh, Weiming Zhang, Debashis Ghosh, and Katerina Kechris</i>	
17	Using MetaboAnalyst 4.0 for Metabolomics Data Analysis, Interpretation, and Integration with Other Omics Data	337
	<i>Jasmine Chong and Jianguo Xia</i>	
18	Using Genome-Scale Metabolic Networks for Analysis, Visualization, and Integration of Targeted Metabolomics Data	361
	<i>Jake P. N. Hattwell, Janna Hastings, Olivia Casanueva, Horst Joachim Schirra, and Michael Witting</i>	
19	Pathway Analysis for Targeted and Untargeted Metabolomics	387
	<i>Alla Karnovsky and Shuzhao Li</i>	
20	Application of Metabolomics to Renal and Cardiometabolic Diseases	401
	<i>Casey M. Rebholz and Eugene P. Rhee</i>	
21	Using the IDEOM Workflow for LCMS-Based Metabolomics Studies of Drug Mechanisms	419
	<i>Anubhav Srivastava and Darren J. Creek</i>	
22	Analyzing Metabolomics Data for Environmental Health and Exposome Research	447
	<i>Tuping Cai, Ana K Rosen Vollmar, and Caroline Helen Johnson</i>	
23	Network-Based Approaches for Multi-omics Integration	469
	<i>Guangyan Zhou, Shuzhao Li, and Jianguo Xia</i>	
	<i>Index</i>	489

Contributors

FABIAN AICHELER • *Applied Bioinformatics Group, University of Tübingen, Tübingen, Germany*

OLIVER ALKA • *Applied Bioinformatics Group, University of Tübingen, Tübingen, Germany*

STEPHEN BARNES • *Department of Pharmacology & Toxicology and Targeted Metabolomics and Proteomics Laboratory, University of Alabama at Birmingham, Birmingham, AL, USA*

SEBASTIAN BÖCKER • *Chair for Bioinformatics, Friedrich-Schiller University, Jena, Germany*

YUPING CAI • *Department of Environmental Health Sciences, Yale School of Public Health, New Haven, CT, USA*

OLIVIA CASANUEVA • *Department of Epigenetics, Babraham Institute, Cambridge, UK*

JASMINE CHONG • *Institute of Parasitology, McGill University, Montreal, QC, Canada*

DARREN J. CREEK • *Drug Delivery, Disposition and Dynamics, Monash Institute of Pharmaceutical Sciences, Monash University, Parkville, VIC, Australia*

XAVIER DOMINGO-ALMENARA • *Scripps Center for Metabolomics, The Scripps Research Institute, La Jolla, CA, USA*

DANIEL P. DOWNES • *Department of Chemistry, Merck & Co., Inc., Kenilworth, NJ, USA*

XIUXIA DU • *Department of Bioinformatics and Genomics, University of North Carolina at Charlotte, Charlotte, NC, USA*

KAI DÜHRKOP • *Chair for Bioinformatics, Friedrich-Schiller University, Jena, Germany*

ALISDAIR R. FERNIE • *Central Metabolism, Max Planck Institute of Molecular Plant Physiology, Potsdam-Golm, Germany*

MARKUS FLEISCHAUER • *Chair for Bioinformatics, Friedrich-Schiller University, Jena, Germany*

EMMANUEL GAQUEREL • *Institute of Plant Molecular Biology, University of Strasbourg, Strasbourg, France; Centre for Organismal Studies, University of Heidelberg, Heidelberg, Germany*

DEBASHIS GHOSH • *Department of Biostatistics and Informatics, Colorado School of Public Health, University of Colorado Anschutz Medical Campus, Aurora, CO, USA*

TUSHARKANTI GHOSH • *Department of Biostatistics and Informatics, Colorado School of Public Health, University of Colorado Anschutz Medical Campus, Aurora, CO, USA*

CARLOS GUIJAS • *Scripps Center for Metabolomics, The Scripps Research Institute, La Jolla, CA, USA*

JANNA HASTINGS • *Department of Epigenetics, Babraham Institute, Cambridge, UK*

JAKE P. N. HATTWELL • *Centre for Advanced Imaging, The University of Queensland, Brisbane, QLD, Australia*

MARTIN A. HOFFMANN • *Chair for Bioinformatics, Friedrich-Schiller University, Jena, Germany; International Max Planck Research School “Exploration of Ecological Interactions with Molecular and Chemical Techniques”, Max Planck Institute for Chemical Ecology, Jena, Germany*

WEI JIA • *University of Hawaii Cancer Center, Honolulu, HI, USA*

CAROLINE HELEN JOHNSON • *Department of Environmental Health Sciences, Yale School of Public Health, New Haven, CT, USA*

- ALLA KARNOVSKY • *Department of Computational Medicine and Bioinformatics, University of Michigan, Ann Arbor, MI, USA*
- KRISTINE KAY • *Department of Nutrition, School of Public Health, NIH Eastern Regional Comprehensive Metabolomics Resource Core (ERCMRC), Nutrition Research Institute, University of North Carolina at Chapel Hill, Kannapolis, NC, USA*
- KATERINA KECHRIS • *Department of Biostatistics and Informatics, Colorado School of Public Health, University of Colorado Anschutz Medical Campus, Aurora, CO, USA*
- OLIVER KOHLBACHER • *Applied Bioinformatics Group, University of Tübingen, Tübingen, Germany; Biomolecular Interactions, Max Planck Institute for Developmental Biology, Tübingen, Germany; Quantitative Biology Center, University of Tübingen, Tübingen, Germany; Institute for Translational Bioinformatics, University Hospital Tübingen, Tübingen, Germany; Institute for Bioinformatics and Medical Informatics, University of Tübingen, Tübingen, Germany*
- JENNIFER E. KYLE • *Biological Sciences Division, Pacific Northwest National Laboratory, Richland, WA, USA*
- SHUZHAO LI • *Department of Medicine, Emory University School of Medicine, Atlanta, GA, USA*
- MARCUS LUDWIG • *Chair for Bioinformatics, Friedrich-Schiller University, Jena, Germany*
- SUSAN MCRITCHIE • *Department of Nutrition, School of Public Health, NIH Eastern Regional Comprehensive Metabolomics Resource Core (ERCMRC), Nutrition Research Institute, University of North Carolina at Chapel Hill, Kannapolis, NC, USA*
- J. RAFAEL MONTENEGRO-BURKE • *Scripps Center for Metabolomics, The Scripps Research Institute, La Jolla, CA, USA*
- THOMAS NAAKE • *Central Metabolism, Max Planck Institute of Molecular Plant Physiology, Potsdam-Golm, Germany*
- WIMAL PATHMASIRI • *Department of Nutrition, School of Public Health, NIH Eastern Regional Comprehensive Metabolomics Resource Core (ERCMRC), Nutrition Research Institute, University of North Carolina at Chapel Hill, Kannapolis, NC, USA*
- VANESSA V. PHELAN • *Department of Pharmaceutical Sciences, Skaggs School of Pharmacy and Pharmaceutical Sciences, University of Colorado, Aurora, CO, USA*
- W. STEPHEN PITTARD • *Department of Biostatistics and Bioinformatics, Rollins School of Public Health, Emory University, Atlanta, GA, USA*
- TOMÁŠ PLUSKAL • *Whitehead Institute for Biomedical Research, Cambridge, MA, USA*
- STEPHEN F. PREVIS • *Department of Chemistry, Merck & Co., Inc., Kenilworth, NJ, USA*
- CASEY M. REBOLZ • *Welch Center for Prevention, Epidemiology and Clinical Research, Baltimore, MD, USA; Department of Epidemiology, Johns Hopkins Bloomberg School of Public Health, Baltimore, MD, USA*
- EUGENE P. RHEE • *Massachusetts General Hospital, Richard B. Simches Research Center, Boston, MA, USA*
- ANA K ROSEN VOLLMAR • *Department of Environmental Health Sciences, Yale School of Public Health, New Haven, CT, USA*
- MARC RURIK • *Applied Bioinformatics Group, University of Tübingen, Tübingen, Germany*
- HORST JOACHIM SCHIRRA • *Centre for Advanced Imaging, The University of Queensland, Brisbane, QLD, Australia*
- GARY SIUZDAK • *Scripps Center for Metabolomics, The Scripps Research Institute, La Jolla, CA, USA; Department of Molecular and Computational Biology, The Scripps Research Institute, La Jolla, CA, USA*

ALEKSANDR SMIRNOV • *Department of Bioinformatics and Genomics, University of North Carolina at Charlotte, Charlotte, NC, USA*

ANUBHAV SRIVASTAVA • *Drug Delivery, Disposition and Dynamics, Monash Institute of Pharmaceutical Sciences, Monash University, Parkville, VIC, Australia*

SUSAN SUMNER • *Department of Nutrition, School of Public Health, NIH Eastern Regional Comprehensive Metabolomics Resource Core (ERCMRC), Nutrition Research Institute, University of North Carolina at Chapel Hill, Kannapolis, NC, USA*

CECILIA “KEEKO” VILLAVECES • *Department of Mathematics, University of Georgia, Athens, GA, USA*

DAVID S. WISHART • *Department of Computing Science, University of Alberta, Edmonton, AB, Canada; Department of Biological Sciences, University of Alberta, Edmonton, AB, Canada*

MICHAEL WITTING • *Research Unit Analytical BioGeoChemistry, Helmholtz Zentrum München, Neuherberg, Germany; Analytical Food Chemistry, Technical University of Munich, Freising, Germany*

JIANGUO XIA • *Institute of Parasitology, McGill University, Montreal, QC, Canada; Department of Animal Science, McGill University, Montreal, QC, Canada; Department of Microbiology and Immunology, McGill University, Montreal, QC, Canada; Department of Human Genetics, McGill University, Montreal, QC, Canada*

ZHANGTAO YI • *Interdisciplinary Research Center on Biology and Chemistry, Shanghai Institute of Organic Chemistry, Chinese Academy of Sciences, Shanghai, People’s Republic of China*

WEIMING ZHANG • *Department of Biostatistics and Informatics, Colorado School of Public Health, University of Colorado Anschutz Medical Campus, Aurora, CO, USA*

GUANGYAN ZHOU • *Institute of Parasitology, McGill University, Montreal, QC, Canada*

ZHENG-JIANG ZHU • *Interdisciplinary Research Center on Biology and Chemistry, Shanghai Institute of Organic Chemistry, Chinese Academy of Sciences, Shanghai, People’s Republic of China*



Chapter 1

Overview of Experimental Methods and Study Design in Metabolomics, and Statistical and Pathway Considerations

Stephen Barnes

Abstract

Metabolomics has become a powerful tool in biological and clinical investigations. This chapter reviews the technological basis of metabolomics and the considerations in answering biomedical questions. The workflow of metabolomics is explained in the sequence of data processing, quality control, metabolite annotation, statistical analysis, pathway analysis, and multi-omics integration. Reproducibility in both sample analysis and data analysis is key to the scientific progress, and the recommendation is made on reporting standards in publications. This chapter explains the technical aspects of metabolomics in the context of systems biology and applications to human health.

Key words Metabolomics, GC-MS, LC-MS, NMR, Precision medicine, Systems medicine, Annotation, Recommendation

1 Introduction

1.1 *The Age of Omics and Precision Medicine*

In biomedical science, the late 1980s saw a great change in the forms and the scale of research data. Instead of cloning individual cDNAs that encoded the open reading frames of individual genes, the NIH funded sequencing of the whole human genome. The data were collected without recourse to a hypothesis; instead, the human genome project was intended to create a national (and later international) resource of genes and genomes. As a result, the opportunity arose to engineer placing representatives of the open reading frames of genes onto small glass chips (microarrays), thus allowing for the “whole” transcriptome to be interrogated in a single experiment. Since then, in studying the transcriptome, the need for selected DNA sequences has been removed, and instead, with further engineering, direct sequencing of the RNA transcripts (RNA-Seq) has occurred.

In parallel to the advances in gene sequencing, the field of protein identification was also making quick strides. Instead of chemically sequencing a protein, in the late 1980s, results from cDNA cloning of the open reading frames were used to create the entire amino acid sequence of proteins. With the introduction of matrix-assisted laser desorption ionization (MALDI) combined with time-of-flight (TOF) mass analysis, peptide mass fingerprinting allowed for even faster identification of individual proteins [1]; indeed, using bioinformatics, even proteins (with amino acid sequences generated from genome sequencing data) that had never been isolated before could be identified. The turning point regarding the generation of massive datasets came with the introduction of electrospray ionization (ESI) that coupled liquid chromatography with mass spectrometry (LC-MS) [2]. Further improvements involved scaling chromatographic flow rates down to nl/min to increase sensitivity [3]. With the use of ion traps, these assays generated large datasets consisting of information on 20,000–50,000 peptides per analysis. It was immediately obvious that it was no longer possible to interpret peptide MS/MS spectra by hand. Computer interrogation of libraries built from multiple sources, in particular from genome sequencing, began. Later, the use of reverse protein databases much improved the quality of the peptide/protein matches [4–7].

In a step relevant to metabolomics, two types of mass spectrometer, a hybrid quadrupole time-of-flight mass analyzer (QTOF) and a quadrupole-Orbitrap, came into use with the ability to collect high mass accurate (1–3 ppm or better) MS spectra of peptides eluting from LC columns and MS/MS spectra of selected peptides. This further increased the number of peptides observed in an analysis and hence the size of the collected datasets. The emphasis in proteomics until quite recently was qualitative, that is, *discovering* the presence of proteins with rudimentary quantitative analysis. This was somewhat overcome with the use of isotopically labeled reagents (iTRAQ and TMT). The introduction of Sequential Windows-MS of All the Theoretical Mass Spectra (SWATH-MS) [8] and Parallel Reaction Monitoring (PRM) [9] has led to the acquisition of complete MS¹ and MS² digital libraries and much improved quantification of detected peptides.

The rapid progress in genomics and proteomics has changed most biomedical disciplines, and translated into clinical practice. Genetic variants now guide the diagnosis and treatment of diseases. The quest of functional genomics (i.e., understanding gene functions) is, however, far from complete [10]. The contributions to human diseases by the genome and by the environment are yet to be quantified and understood, but key to the concept of precision medicine [11, 12]. Metabolomics, which shares some degree of analytical chemistry with proteomics, is critical to fill this gap between genome and environment [13].

1.2 Technologies for Metabolomics

Metabolomics is different from earlier studies of metabolism (1935–1950) in that it is a method based on chromatography, mass spectrometry, and NMR to discover *all* chemical forms in a biological sample. Initially, the study of endogenous metabolites from known metabolic pathways and of specific exogenous compounds (known drugs and natural and man-made toxins) was principally centered on gas liquid chromatography (GC) and later in combination with mass spectrometry (GC-MS). With the introduction of ESI and atmospheric pressure chemical ionization (APCI), a much wider group of metabolites and other compounds found in biological systems could now be studied.

1.2.1 GC-MS

GC-MS is a *targeted* approach to identify and quantify compounds in a biological sample. While GC-MS has extensive libraries of spectral and retention time data, it is nonetheless generally limited to compounds with masses of 400 Da or less and which once derivatized are thermally stable. Since electron impact ionization dissociates the volatile form of the metabolite to create ions, only product ion spectra are collected for each “metabolite.” This has placed an emphasis on reproducible, chromatographic separation that was made possible by the commercial introduction of open tubular, wall-coated, capillary GC in the 1970s.

1.2.2 LC-MS

The transition in metabolomics to LC-MS because of the introduction of ESI and APCI enabled a much wider group of compounds in the metabolome to be detected as positively and negatively charged molecular ions formed in each interface. The molecular ions in turn could be selectively isolated (by a quadrupole mass filter), passed into a chamber for collision with neutral gas to generate their product ion (MS/MS) spectra. Initially, the vast number of molecular ions hinted at the possibility of discovery of many new metabolic pathways. However, the soft ionization of the ESI interface leads to appearance of many adducts (Na^+ , NH_4^+ , and K^+ in positive ion spectra and formate, acetate and chloride in negative ion spectra) as well as isotopic (^{13}C) species, and multiply charged and dimeric species. Therefore, a single metabolite typically may be represented by many (5–10 or more) different ion features. The number observed is a function of the amount of a metabolite. As a further issue, although most metabolites remain as intact molecular ions, some such as β -glucuronides will undergo in-source decay giving rise to ions of glucuronic acid (m/z 176 and 113) as well as neutral loss ions (M-176). In most cases, all these multiple ion forms of a metabolites will coelute.

1.2.3 CE-MS

Significant progress has been made to capillary electrophoresis—MS (CE-MS) in the past few years, by improving the interfacing between CE and ESI. Due to the selectivity of electrophoresis,

CE-MS can be fast and work well with small volumes [14]. As is occurring in transcriptomics and proteomics, there is a strong interest in studying the metabolome of single cells. However, the fluid coming from a single cell without dilution is in the nL or even pL range. CESI-MS is suited to these volumes and since it also generates a very sharp peak for each metabolite, it has allowed for the analysis of over 100 metabolites from a frog embryo cell [15].

1.2.4 *Ion Mobility*

Besides using GC, LC or CE to chromatographically separate metabolites and MS, MS/MS and even MSⁿ to distinguish them from one another, another dimension of separation that is rapidly being introduced is ion mobility which comes in several flavors. FAIMS (high-field asymmetric waveform ion mobility spectrometry) and differential mobility operate in the ionization interface. Ions move through a small gap and are subjected to an oscillating, orthogonal and asymmetric, electric field. By applying a particular compensating voltage, individual ions can be coaxed to pass into the mass spectrometer. Further metabolite separation can be achieved by including low levels of metal ions in mobile phase or by resolvating the ions with isopropanol or other volatile solvents. The other classes of ion mobility occur inside the mass spectrometer and allow for the calculation of the collision cross section of a metabolite ion. This can be an absolute value of a metabolite ion, independent of the instrument used for its measurement or lab where it is done.

1.2.5 *NMR*

The application of NMR to metabolomics came as a consequence of two innovations—the introduction of superconducting magnets and pulse sequences. The much higher field strengths (for proton resonances from 90 MHz for iron magnets to first 400 MHz and now up to 1000 MHz) significantly improved the resolution of individual metabolite resonances, allowing direct quantitative analysis. Prior to the interest in metabolomics, samples had to be carefully dried to exclude water since it was present in many orders higher amounts to the metabolites. The introduction of WET and 1D-NOESY pulse sequences suppressed the water proton signal, thereby allowing analysis of biofluids *as is*.

1.3 *What Is the Metabolome?*

The metabolome in a human biological fluid or even in the culture of a single cell is not fully predicted by the genes in the genome of that species. Since humans can only synthesize from simpler precursors of ten of the twenty amino acids that make up human proteins, they and other forms of life therefore have to eat (or be fed) foods coming from other genomes. Some foods, such as fruits and vegetables, have genomes and hence secondary metabolic pathways, that are completely unrelated to those in humans. In addition to amino acids, these other sources may provide critical small molecules that are vital for life (vitamins), many of which are

enzyme cofactors. Furthermore, humans and other species are polygenic due to the presence of vast populations of microbial species, particularly in the gastrointestinal tract. These microbial communities carry out other pathways not represented in mammalian or plant genomes. Therefore, pathways are not two dimensional as in a single species. Instead, they are polydimensional and as such are not (yet) represented in published pathway maps or databases.

Environmental exposures and their metabolites are also present in humans. With increased sensitivity, some of these molecules can be directly detected and measured in metabolomics assays. Besides, the endogenous molecules in the metabolomics data can be used to assess biological effects from the exposures. Thus, exposome is a highly active area of metabolomics research (as discussed in Chapter 22).

As a subset of metabolomics, lipidomics is often discussed separately because lipids require their own sample preparation, analytical protocol and data processing. Within each lipid class, there are many variations of different length of carbon chains and number of double bonds. This leads to particular patterns in the mass spectrometry data, which can be leveraged in data processing. Chapter 7 by Dr. Kyle gives a practical guide on analyzing these lipidomics data.

2 Challenges in Interrogating the “Metabolome”

The diverse nature of the metabolome generates very large and complex datasets requiring statistical analysis, metabolite identification and verification, and pathway analysis procedures, as described in the chapters of this book. Although, integrated, commercial software tools exist, the metabolomics community benefit more from free and open source tools. Because commercial software is unlikely to keep up with the pace of scientific innovation, adaptability to the changing of specific needs and modifying source code is key to scientific investigations.

Before these methods can be invoked, collected data require preprocessing. For NMR (without preliminary chromatography), the chemical shifts of the NMR resonances are first referenced to 0.00 ppm with trimethylsilyl-2,2,3,3-deuterated-propionic acid added as an internal standard. The data can be analyzed by binning, dividing the NMR spectrum into small windows. However, due to pH differences in individual samples, this may lead to false metabolite comparisons. To overcome this, spectra are adjusted for the effects of pH. The methods of data preprocessing and data analysis strategies used in NMR based metabolomics analysis are described in Chapter 5 by Dr. Wimal Pathmasiri and his colleagues.

GC-MS and LC-MS data both contain information about ion features—mass-to-charge (m/z) values, retention times, and intensities. In the case of LC-MS data, besides precursor ion data (MS1), there are accompanying MS/MS (MS2) data. The latter may be on selected precursor ions, or be comprehensive, as in SWATH-MS. MS1 data can be examined in order to identify ion features. Since many of the ions are from known, abundant metabolites, their true m/z values can be used to calibrate the other observed feature masses. A further adjustment is made for variation in retention times before statistical analysis. Although this can largely be dealt with by using reproducible chromatographic methods, there are alignment tools to do this. The data preprocessing for MS metabolomics is described in relation to three popular tools, XCMS (Drs. Xavier Domingo-Almenara and Gary Siuzdak, Chapter 2), MzMine (Dr. Xiuxia Du et al., Chapter 3), and OpenMS (Dr. Kohlbacher and colleagues, Chapter 4).

The greatest challenge in metabolomics is identifying the metabolite features. Although the mass of a molecular ion can be measured with accuracies in the sub-ppm range, sufficient to obtain its empirical formula, this does not allow for its identification since there are usually multiple metabolites with identical formulae. In some cases, the metabolites can be resolved chromatographically; however, validating their identification may depend on having an authentic standard which is frequently not the case.

Chapter 11 describes the state of the art in computational prediction of chemical formula and structures by Dr. Böcker and colleagues. Initially, metabolite databases such as METLIN (Chapter 9) and the Human Metabolite Database (HMDB, Chapter 10) accumulated MS1 data from multiple sources. For some metabolites, there was accompanying MS/MS data. However, the MS/MS data were often carried out under collision conditions that were not equivalent to those used by an individual metabolomics investigator, particularly if the mass analyzer used was different (low resolution/mass accuracy ion trap or quadrupole analyzer, versus a high resolution/mass accuracy time-of-flight, orbitrap or ion cyclotron resonance analyzer). This led to the Siuzdak group to generate MS/MS data under differing voltage collisions (e.g., 0, 10, 20, or 40 V). However, creating these spectra on more than 100,000 metabolites is a substantial task. To overcome this mountain of work, there has been a strong interest in generating predicted MS/MS spectra as well as using high-resolution fragment ion libraries to suggest partial structures of a metabolite ion. An overview of tandem mass spectral and metabolite databases is given by Yi and Zhu in Chapter 8. Different strategies of metabolite annotation based on the concept of “molecular networking” are illustrated by Dr. Naake et al. in Chapter 12 on plant metabolites, and by Dr. Phelan in Chapter 13 on microbial metabolites.

The level of metabolite identification dictates the strategy of data analysis, which is explained in detail later. For the reproducibility of science, a set of standards for reporting the findings is necessary.

3 Metabolomics Reporting Recommendation

From the Metabolomics Standards Initiative (MSI) by the Metabolomics Society, a minimal reporting standard for metabolomics experiments was published to guide how a metabolomics study and its biological system should be described in a reproducible way [16, 17]. The leading data repositories (Metabolomics Workbench, <https://www.metabolomicsworkbench.org>; MetaboLights, <https://www.ebi.ac.uk/metabolights/>) now enforce a submission standard similar to those.

Part of the initiative produced the definition of four levels of reporting metabolite identification in scientific literature [18]. The four levels are level 1 for identified metabolites, level 2 for putatively annotated compounds, level 3 for putatively characterized compound classes, and level 4 for unknown compounds. Schymanski et al. [19] proposed an updated, popular level system as follows: level 1 for confirmed structure by reference standard, level 2 for probable structure, level 3 for tentative candidate, level 4 for unequivocal molecular formula, and level 5 for exact mass. The support data requirement is defined for each level (e.g., experimental MS and MS^n data for level 3). Schrimpe-Rutledge et al. ([20] further proposed that orthogonal information, e.g. IM-MS and NMR data, can be used to achieve level 2 and level 3 identification). For scientific publications, authors are recommended to share their data in a major repository, and report metabolite identification level according to one of these above references.

Besides reproducibility in sample analysis, we are reminded that reproducibility in data analysis is as important. The use of computational notebooks, version control and Docker containers, as described in Chapters 14 and 15, is among best practices. The recommendation of TRIPOD (Transparent reporting of a multi-variable prediction model for individual prognosis or diagnosis) is described in Chapter 16.

4 Data Analysis

Once detected and aligned (both NMR and GC-MS/LC-MS) processed data can undergo statistical analysis. Chapters 14, 15, and 16 introduce major skills that are employed in statistics and data science, using examples on metabolomics data analysis, including quality control and data visualization. The most commonly

used, online statistical software is MetaboAnalyst (<http://www.MetaboAnalyst.ca>) which is described in Chapter 17. MetaboAnalyst contains routines for many univariate and multivariate statistical methods, hierachal clustering, heatmaps and vector support module [21]. A strong feature of MetaboAnalyst is that it allows the investigator to download a record of all steps taken to analyze a set of data. Besides the statistical routines, MetaboAnalyst has many other components including pathway analysis.

While metabolite identification is required to perform conventional pathway and network analysis, an innovative solution for pathway without the formal identification of metabolites is Mummichog [22]. It takes all possible interpretations of an ion feature, including grouping the features with adducts and isotopic forms. The ions are separated into two groups based on their statistical importance (defined by the user). Ions from the nonsignificant group, equivalent in number to those in the statistically significant group, are selected at random and mapped to known metabolite pathways. This repeated 100 or more times to estimate nonsignificant association with each pathway. Then the significant group of metabolites are mapped to the same pathways and pathway enhancement detected. Mummichog also links individual metabolites together based on their chemistry, thereby creating chemical networks. The network overlaps with the concept of genome scale metabolic models, which are covered in depth by Dr. Witting and colleagues in Chapter 18.

5 From Metabolomics to Systems Medicine

The applications of metabolomics are broad and transformative. Both untargeted and targeted assays are used to dissect intracellular molecular mechanisms. Even more powerful is the combination with stable isotope tracing, which can be used to identify the pathway of action, and to discover new pathways (Chapter 6). Arguably, metabolomics is the most exciting innovation in epidemiology since the infusion of genomics. Hundreds of papers have been published using metabolomics in population studies, often via the approach of MWAS: metabolome-wide association studies. Chapter 20 by Drs. Rebholz and Rhee gives a practical discourse on using metabolomics for biomarkers and clinical investigations. Mass spectrometry has always been important to pharmacology. Drs. Srivastava and Creek describe a detailed example of using metabolomics to track drug mechanisms (Chapter 21). As mentioned earlier, the mass spectrometry data can capture environmental exposures and their biological responses. Chapter 22 by Caroline Johnson and colleagues describes the concepts and approaches of exposome, the ambitious effort to investigation the total environmental exposures.

In the three decades following the launch of human genome project, the technologies of -omics have become widespread and more economical. Multi-omics studies, for example, employing transcriptomics, proteomics, and metabolomics in matched samples, have become feasible. With mass spectrometry and careful sample preparation, one can even perform proteomics, metabolomics, and lipidomics from a single sample [23]. Given that each -omics technology measures only a slice of real biological complexity, the integration of multiple -omics data will produce more complete understanding of the biology. Because of the high-throughput nature of the data, a single data type is prone to false positives. The intersection of multiple -omics data is therefore effective to identify specific mechanisms. These efforts on data integration often depend on particular biological models; for example, proteins and mRNA transcripts are both linked to gene models in genome annotation. When the measurements are based on different biological matrices, data- or network-based integration is more useful, as discussed in Chapter 23.

Personal genomics is now in clinical practice. Metabolomics holds the promise of molecular phenotyping and is indispensable to our quest of precision medicine [13, 24]. As metabolite panels are routinely used in blood tests, it is possible in the foreseeable future that comprehensive metabolomics screen will become part of routine healthcare. We are optimistic that the methodologies presented in this book will help us get to this future.

References

1. Hillenkamp F, Karas M (1990) Mass spectrometry of peptides and proteins by matrix-assisted ultraviolet laser desorption/ionization. *Methods Enzymol* 193:280–295
2. Want EJ, Cravatt BF, Siuzdak G (2005) The expanding role of mass spectrometry in metabolite profiling and characterization. *Chembiochem* 6(11):1941–1951. <https://doi.org/10.1002/cbic.200500151>
3. Barnes S, Benton HP, Casazza K, Cooper SJ, Cui X, Du X, Engler J, Kabarowski JH, Li S, Pathmasiri W, Prasain JK, Renfrow MB, Tiwari HK (2016) Training in metabolomics research. I. Designing the experiment, collecting and extracting samples and generating metabolomics data. *J Mass Spectrom* 51(7):461–475. <https://doi.org/10.1002/jms.3782>
4. Cox J, Mann M (2008) MaxQuant enables high peptide identification rates, individualized p.p.b.-range mass accuracies and proteome-wide protein quantification. *Nat Biotechnol* 26(12):1367–1372. <https://doi.org/10.1038/nbt.1511>
5. Craig R, Cortens JC, Fenyo D, Beavis RC (2006) Using annotated peptide mass spectrum libraries for protein identification. *J Proteome Res* 5(8):1843–1849. <https://doi.org/10.1021/pr0602085>
6. Huttlin EL, Hegeman AD, Harms AC, Sussman MR (2007) Prediction of error associated with false-positive rate determination for peptide identification in large-scale proteomics experiments using a combined reverse and forward peptide sequence database strategy. *J Proteome Res* 6(1):392–398. <https://doi.org/10.1021/pr0603194>
7. Nesvizhskii AI (2007) Protein identification by tandem mass spectrometry and sequence database searching. *Methods Mol Biol* 367:87–119. <https://doi.org/10.1385/1-59745-275-0:87>
8. Anjo SI, Santa C, Manadas B (2017) SWATH-MS as a tool for biomarker discovery: from basic research to clinical applications. *Proteomics* 17(3–4). <https://doi.org/10.1002/pmic.201600278>

9. Peterson AC, Russell JD, Bailey DJ, Westphall MS, Coon JJ (2012) Parallel reaction monitoring for high resolution and high mass accuracy quantitative, targeted proteomics. *Mol Cell Proteomics* 11(11):1475–1488. <https://doi.org/10.1074/mcp.O112.020131>
10. Rinschen MM, Ivanisevic J, Giera M, Siuzdak G (2019) Identification of bioactive metabolites using activity metabolomics. *Nat Rev Mol Cell Biol* 20(6):353–367. <https://doi.org/10.1038/s41580-019-0108-4>
11. Collins FS (2004) The case for a US prospective cohort study of genes and environment. *Nature* 429(6990):475–477. <https://doi.org/10.1038/nature02628>
12. Manrai AK, Cui Y, Bushel PR, Hall M, Karakitsios S, Mattingly CJ, Ritchie M, Schmitt C, Sarigiannis DA, Thomas DC, Wishart D, Balschaw DM, Patel CJ (2017) Informatics and data analytics to support exposome-based discovery for public health. *Annu Rev Public Health* 38:279–294. <https://doi.org/10.1146/annurev-publhealth-082516-012737>
13. Li S, Cirillo P, Hu X, Tran V, Krigbaum N, Yu S, Jones DP, Cohn B (2019) Understanding mixed environmental exposures using metabolomics via a hierarchical community network model in a cohort of California women in 1960's. *Reprod Toxicol*. pii: S0890-6238(18)30603-8. <https://doi.org/10.1016/j.reprotox.2019.06.013>
14. Stoltz A, Jooss K, Hocker O, Romer J, Schlecht J, Neususs C (2019) Recent advances in capillary electrophoresis-mass spectrometry: instrumentation, methodology and applications. *Electrophoresis* 40(1):79–112. <https://doi.org/10.1002/elps.201800331>
15. Onjiko RM, Portero EP, Moody SA, Nemes P (2017) In situ microprobe single-cell capillary electrophoresis mass spectrometry: metabolic reorganization in single differentiating cells in the live vertebrate (*Xenopus laevis*) embryo. *Anal Chem* 89(13):7069–7076. <https://doi.org/10.1021/acs.analchem.7b00880>
16. Members MSIB, Sansone SA, Fan T, Goodacre R, Griffin JL, Hardy NW, Kaddurah-Daouk R, Kristal BS, Lindon J, Mendes P, Morrison N, Nikolau B, Robertson D, Sumner LW, Taylor C, van der Werf M, van Ommeren B, Fiehn O (2007) The metabolomics standards initiative. *Nat Biotechnol* 25(8):846–848. <https://doi.org/10.1038/nbt0807-846b>
17. Salek RM, Steinbeck C, Viant MR, Goodacre R, Dunn WB (2013) The role of reporting standards for metabolite annotation and identification in metabolomic studies. *Gigascience* 2(1):13. <https://doi.org/10.1186/2047-217X-2-13>
18. Sumner LW, Amberg A, Barrett D, Beale MH, Beger R, Daykin CA, Fan TW, Fiehn O, Goodacre R, Griffin JL, Hankemeier T, Hardy N, Harnly J, Higashi R, Kopka J, Lane AN, Lindon JC, Marriott P, Nicholls AW, Reily MD, Thaden JJ, Viant MR (2007) Proposed minimum reporting standards for chemical analysis Chemical Analysis Working Group (CAWG) Metabolomics Standards Initiative (MSI). *Metabolomics* 3(3):211–221. <https://doi.org/10.1007/s11306-007-0082-2>
19. Schymanski EL, Jeon J, Gulde R, Fenner K, Ruff M, Singer HP, Hollender J (2014) Identifying small molecules via high resolution mass spectrometry: communicating confidence. *Environ Sci Technol* 48(4):2097–2098. <https://doi.org/10.1021/es5002105>
20. Schrimpe-Rutledge AC, Codreanu SG, Sherrod SD, McLean JA (2016) Untargeted metabolomics strategies-challenges and emerging directions. *J Am Soc Mass Spectrom* 27(12):1897–1905. <https://doi.org/10.1007/s13361-016-1469-y>
21. Chong J, Soufan O, Li C, Caraus I, Li S, Bourque G, Wishart DS, Xia J (2018) MetaboAnalyst 4.0: towards more transparent and integrative metabolomics analysis. *Nucleic Acids Res* 46(W1):W486–W494. <https://doi.org/10.1093/nar/gky310>
22. Li S, Park Y, Duraisingham S, Strobel FH, Khan N, Soltow QA, Jones DP, Pulendran B (2013) Predicting network activity from high throughput metabolomics. *PLoS Comput Biol* 9(7):e1003123. <https://doi.org/10.1371/journal.pcbi.1003123>
23. Nakayasu ES, Nicora CD, Sims AC, Burnum-Johnson KE, Kim YM, Kyle JE, Matzke MM, Shukla AK, Chu RK, Schepmoes AA, Jacobs JM, Baric RS, Webb-Robertson BJ, Smith RD, Metz TO (2016) MPLEX: a robust and universal protocol for single-sample integrative proteomic, metabolomic, and lipidomic analyses. *mSystems* 1(3). <https://doi.org/10.1128/mSystems.00043-16>
24. Guo L, Milburn MV, Ryals JA, Lonergan SC, Mitchell MW, Wulff JE, Alexander DC, Evans AM, Bridgewater B, Miller L, Gonzalez-Garay ML, Caskey CT (2015) Plasma metabolomic profiles enhance precision medicine for volunteers of normal health. *Proc Natl Acad Sci U S A* 112(35):E4901–E4910. <https://doi.org/10.1073/pnas.1508425112>



Chapter 2

Metabolomics Data Processing Using XCMS

Xavier Domingo-Almenara and Gary Siuzdak

Abstract

XCMS is one of the most used software for liquid chromatography–mass spectrometry (LC-MS) data processing and it exists both as an R package and as a cloud-based platform known as XCMS Online. In this chapter, we first overview the nature of LC-MS data to contextualize the need for data processing software. Next, we describe the algorithms used by XCMS and the role that the different user-defined parameters play in the data processing. Finally, we describe the extended capabilities of XCMS Online.

Key words XCMS, Liquid chromatography, Mass spectrometry, Metabolomics, Data processing

1 Introduction

Untargeted or global metabolomics aims at quantifying as many metabolites as possible in biological samples [1]. Specifically, liquid chromatography–electrospray ionization mass spectrometry (LC-ESI-MS or simply LC-MS) is one of the most used technologies for that purpose, due to its easy sample preparation process and wide metabolite coverage. LC-MS, as we will further explain in the next sections, generates large and complex datasets comprising thousands of chromatographic peaks [2]. Computational tools are therefore needed to process this data and convert it into interpretable information.

The untargeted computational data processing workflow was popularized in 2006 by the papers describing two of the most widely used tools in untargeted metabolomics: XCMS [3] and mzMine [4, 5]. The untargeted metabolomics data processing workflow typically consists of a peak-picking process followed by a peak alignment which ultimately yields a set of features, that is, a peak or a set of peaks across samples with a unique m/z and retention time. The aim of this process is to transform raw data into a matrix containing the list of observed features with their relative peak areas or intensities for each sample in a given experiment. This list is known as the feature list, and it is used to compare

the peak area (and thus relative concentration) between the same ion peak across samples, allowing us to find statistically significant dysregulated peaks for a given phenotype.

The processes that usually follows peak-picking and alignment are statistical analysis - to find dysregulated peaks in a given phenotype-, or metabolite annotation. Computational metabolite annotation aims at providing chemical information on the observed features [6]. This annotation usually consists in determining what features stem from the same and each metabolite, determining the nature of features (e.g., determine if a feature is a protonated/ deprotonated species, an adduct, an in-source fragment, an isotope, a dimer, etc.), and providing with putative metabolite identifications. As an example, the simplest annotation process consists in using the *m/z* of the observed features and match them against the database to determine their potential identity. This is considered, however, a very weak annotation, and as we will discuss further in this chapter, more advanced techniques based on in-source fragment annotation can be used.

In this chapter, we describe XCMS, a computational workflow that integrates peak picking with alignment. It is available as an R package and also as a cloud-based resource. The latter encompasses a comprehensive suite of tools through an easy-to-use visual interface that allows for data to be shared with the community or collaborators and has a higher computational performance. It also integrates tools and modules that enable systems biology analysis and advanced metabolite annotation through its integration with the METLIN spectral library (Fig. 1). A version of XCMS Online for targeted metabolomics is also available, called XCMS-MRM.

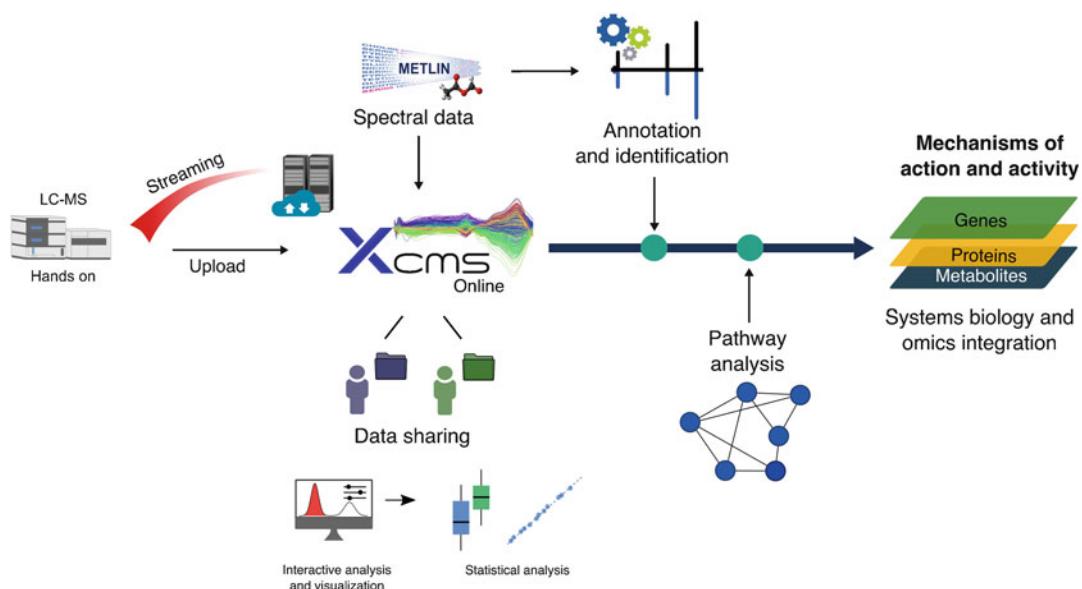


Fig. 1 XCMS Online overview. Through an interactive user interface, XCMS Online allows for data sharing, data streaming, advanced metabolite annotation, and systems biology analysis and multi-omics integration

2 Overview of LC-MS Data

When samples are injected into an LC-MS system, molecules are separated on the chromatographic column (LC). At the end of the column, the MS scans at a given scan rate (scans per second) yielding a signal that depends on the molecules being detected. This process generates a three-dimensional set of chromatographic peaks, each with specific retention time (RT), m/z , and intensity, as a result of the underlying molecules in the analyzed sample.

In LC-ESI-MS, each detected molecule generates multiple chromatographic peaks. In most cases, a single molecule will generate a protonated/deprotonated species when using common untargeted metabolomics chromatographic methods. In addition, isotope, adducts and in-source fragments will also be observed. Particularly, in-source fragmentation, a common phenomenon in LC-ESI-MS, leads to the observation of (in-source) fragments or neutral losses. Some of these neutral losses are known as common neutral losses (e.g., $-H_2O$, $-NH_3$, $-HCOOH$), as they are observed across a wide range of biological organic molecules. Some other peaks including dimers or multimers are less frequently observed.

For the reasons mentioned above, LC-MS generates large and complex datasets comprising thousands of chromatographic peaks. Papers using untargeted metabolomics usually report the number of features observed in their experiments. However, there is still not a consensus on how many metabolites are approximately observed in routine metabolomics analyses. Through computational annotation algorithms, some studies have reported between 1000 metabolites in *Escherichia coli* [7] to 2000 metabolites in *Saccharomyces cerevisiae* [8]. How to effectively attribute all the peaks to bona fide metabolites is an active research area.

In that sense, peak or metabolite annotation in LC-MS aims at (1) grouping all features stemming from the same metabolite and (2) assigning putative metabolite identities to observed features. It is also worth mentioning that this term is also employed to refer to the annotation of fragments and metabolites in tandem MS (LC-MS/MS).

3 XCMS Workflow and Algorithms

As mentioned before, XCMS is a computational tool that integrates peak picking with alignment to generate a list of features from raw MS data. Peak picking is a computational procedure used to detect peaks in MS data and integrate their area. Historically, two types of peak picking algorithms have been the most popular: the match filter algorithm and its advanced variation for high-resolution mass spectrometry data known as the centWave algorithm. Natively,

XCMS has the option of using either the original match filter algorithm or the centWave. In this section, we will describe these two types of algorithms.

When peaks in raw MS data are detected across multiple samples, the same peak stemming from a given ion will ideally appear across the different samples with different intensities. Due to the high accuracy of modern mass spectrometers such as TOF and Orbitrap instruments, m/z values across samples tend to have variations as low as 5 and 1 ppm, respectively. On the other hand, the retention time will have larger variations across samples. Therefore, peaks from the same ion detected across different samples need to be aligned and grouped, that is, their area needs to be assigned to the same row in a data matrix so they can be quantitatively compared across different samples. Specifically, in XCMS, this peak alignment is performed by a two-step process known as peak grouping and retention time alignment. These two processes will be described in the following sections.

After peaks are detected and aligned, statistics can be used to find features showing statistically significant changes among peak areas among phenotypes. However, after peak-picking and alignment, the existence of missing values is a frequent scenario, typically affecting more than 80% of the detected features [9]. This means that around 80% of the detected features will have missing values for certain samples, that is, zero peak area or intensity. This occurs when for a given group of samples, the ion peak is below the detection limit. In other cases, the peak is observed above the detection limit but the peak picking algorithm has failed in detecting it due to low peak intensity or because the peak is masked by noise or other coeluting peaks. Having missing values reduce the power of statistical tests and analysis and can lead to biased results [9]. This is of special importance if multivariate analyses are applied, as missing values will bias the results. To tackle this problem there exist different strategies known as missing value imputation strategies [10]. Among these strategies, a commonly accepted approach is known as filling peaks, where “missing” peaks are searched again in the raw data. XCMS uses a fill peaks strategy to “fill” the “missing” peaks in the feature list and this strategy will be detailed in the following sections.

The following sections explain in detail the different steps of the XCMS workflow (Fig. 2) which consist of (1) peak picking, (2) peak grouping and retention time alignment, and (3) fill peaks.

3.1 Peak Picking: Operation and Algorithms

Peak picking aims at finding the chromatographic peaks stemming from molecules eluting from the chromatographic column. Historically, these have been based on filtering algorithms, which filter the signal to clear it from noise and thus distinguish the underlying peaks from the raw signal. There are many different peak picking algorithms that have been designed, some of which are targeted

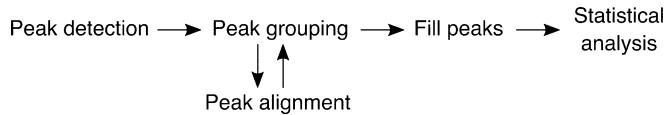


Fig. 2 XCMS general workflow. The workflow consists of peak detection, grouping of peaks into features, alignment, missing peaks filling, and the statistical analysis of features across classes

specifically at dealing with certain types of data. In LC-MS data the peak detectors need to process the complex three-dimensional LC-MS data to find these peaks. Some algorithms such as the “centWave” algorithm need the data to be reduced beforehand. This reduction consists of converting the data, which is typically acquired in profile mode, into centroid mode data [11]. This can dramatically reduce the data size and make the processing of the data far simpler. While most vendors have their own centroiding algorithms, open source alternatives can also be used. However, users should be careful if the mass accuracy is retained with these algorithms.

Specifically, in XCMS, peak picking can be performed via two algorithms: Matched filter and centWave.

3.1.1 Matched Filter

It is the original algorithm of detection of XCMS [3]. To detect peaks, the algorithm first performs a “binning” procedure consisting in slicing the data into bins of 0.1 m/z units (defined by the parameter *step*). For each bin, the algorithm then detects any peak that is above the signal-to-noise threshold as defined by the *S/N ratio cutoff* parameter (default 10). This peak detection is performed by leveraging the typical Gaussian-like shape of chromatographic peaks to detect them. That means that if these data points fit well into a second derivative Gaussian (essentially a normal distribution) they are detected as a peak. The width of this Gaussian is defined by the full width at half maximum (*FWHM*) parameter, a value in seconds. The lower the value of this parameter, the more likely is to detect false positive peaks (noise detected as a peak). To make sure that peaks are not split between two m/z bins the algorithm combines pairs of consecutive m/z bins. This algorithm was originally designed for data acquired by low resolution instruments such as single quadrupole mass spectrometers, where the highest mass accuracy was around 0.1 Da. For current high-resolution mass spectrometers (HRMS), the centWave algorithm is recommended (described below). However, if the MatchedFilter is to be applied to HRMS data, the binning value should be significantly lowered. The output of the algorithm is a peak list where each m/z , RT (and the deviations) and integrated peak intensity (peak area) are given.

3.1.2 centWave

Published in 2008 [12], this algorithm is a high resolution and consequently high mass accuracy peak detection algorithm. The “centWave” algorithm consists also of two steps. The first step consists in a dynamic binning that finds potential areas containing peaks, known as region of interest (ROI). The second step performs the peak detection within these ROIs using a highly sensitive wavelet filter (Fig. 3a). This algorithm looks for ROI consisting of data points that show low m/z deviations and increase and then decrease in intensity (Fig. 3a). The main parameters that control this behavior are the “ppm” parameter that selects the m/z span of the ROIs and the peak width which is measured in seconds and dictates how long the peak should be in chromatographic time. Once a ROI fulfilling these requirements has been discovered it is analyzed by the wavelet filter. The wavelet that is used is a Mexican hat wavelet that models the peak shape and allows for the selection of multiple closely eluting peaks within this ROI (Fig. 3b). The

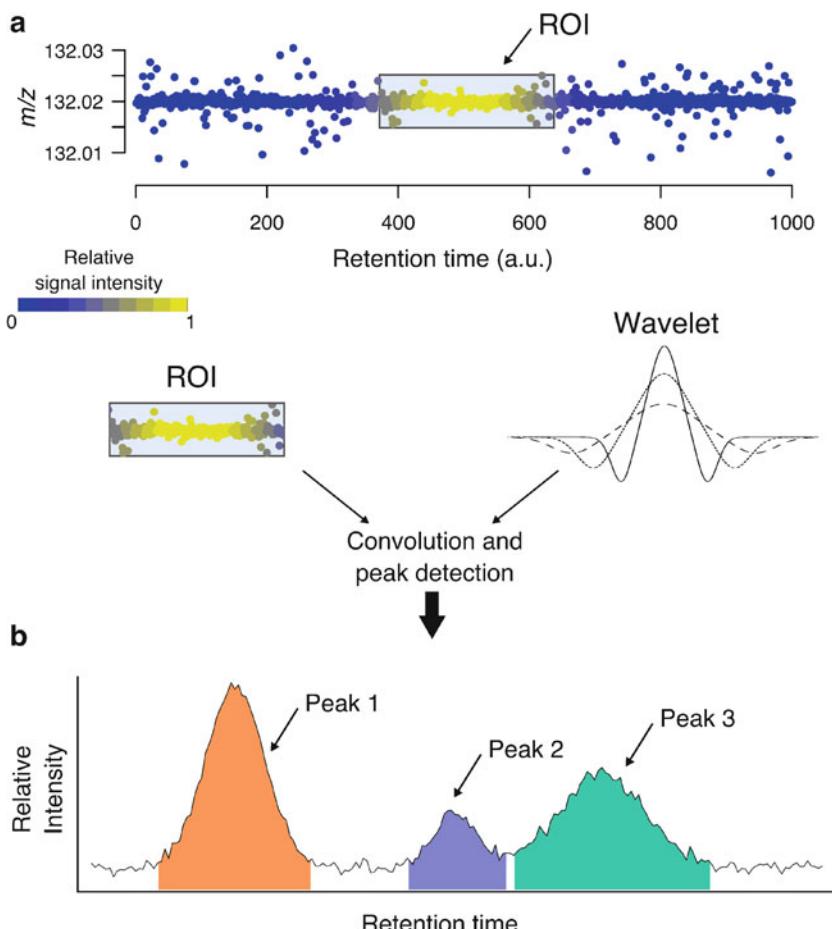


Fig. 3 The centWave peak detection overview. First, regions of interest (ROI) are detected. For each detected ROI, a wavelet filter is applied and peaks within the ROI are detected

scale or height of the peak is modulated until the best fit is achieved. If the fitting parameters are not satisfied, then the ROI is rejected as a peak. The output of this algorithm is the same as in the MatchedFilter and consists of the m/z , RT (deviations of each) and the integrated intensities.

3.2 Peak Alignment: Peak Grouping and Retention Time Alignment

In XCMS, what is considered as peak alignment is performed by a two-step approach consisting of peak grouping and alignment across samples. The order of these two steps (which one will be applied first) depends on which algorithm will be used for the alignment.

When the original “peakGroups” alignment method is used, peaks must be first grouped into features. This is done by an algorithm named “peakDensity,” which uses a similar approach as in the “matchedFilter” algorithm described above, but on the processed peak list instead of the raw data. Using thin m/z slices (defined by the “mzwid” parameter), the algorithm finds groups of peaks across samples that cluster around a certain RT. This is to find “well-behaved” peak groups (WBPG) that will be subsequently used to align the rest of the chromatogram yielding a higher accuracy in overall peak grouping. To find these WBPG, the algorithm uses a kernel density filter that pinpoints areas with groups of “well-behaved” peaks. The filter can be manipulated by the “bw” parameter or bandwidth which is defined in seconds. For HPLC data, 30 s is a common default value. WBPG need to be scattered across the retention time to allow for a good nonlinear correction. Using the median retention time of each WBPG, an alignment profile can be made by a regression technique called LOcally Estimated Scatter-plot Smoother (LOESS). This technique is a non-parametric technique that fits a smoothed line through each chromatogram and compares these smoothed lines across chromatograms to finally correct the retention time across samples.

An alternative to the nonlinear retention time alignment algorithm is the “obiwarp” algorithm. The obiwarp does not require this pregrouping stage and can directly align peaks across samples. The “obiwarp” algorithm was implemented into XCMS by Prince et al. [13]. This algorithm uses a technique called dynamic programming to find the best alignment between two chromatograms. The algorithm works in a pairwise fashion where it first finds a median retention profile across multiple chromatograms, and then finds the path to iteratively align all the chromatograms to that median profile. This algorithm can be computationally demanding and time consuming.

Regardless of the retention time alignment algorithm used, the next step is always to group the peaks. By grouping peaks, the algorithm outputs features, defined as a single or a set of peaks across samples with a unique RT and m/z . The above “peakDensity” algorithm can be used again to complete this task. Now that

the peaks have been aligned the “bw” parameter can be greatly reduced. As an additional robustness filter, there is a parameter called “minfrac.” This parameter states that for any one class (e.g., KO for knockout, WT for wild type), there must be at least a certain fraction of peaks present in that feature to be considered as a valid feature. For example, if we have 6 samples in a KO class and 6 samples in a WT class, then for a particular *m/z* and RT we need to have at least 3 peaks present in either class if “minfrac” is set at 50%. It should be noted that it is “any one class” and not both classes. Therefore, if KO has 3 peaks and WT has 0, it is still a valid feature. This parameter can be good for increasing the robustness when we need to seek peaks in all samples.

3.3 Fill Peaks

As commented before, two main causes can lead to peaks being missed. The first is that the peak has not been correctly detected or aligned by the algorithm, or the peak is under the detection limit. The fill peaks will effectively resolve and find missing peaks when the problem is due to the algorithm (the peak exists in the data but it was not correctly detected or aligned), leveraging the information from other samples where the peak has been detected (RT and *m/z*). In the second case, where the peak is under the detection limit, the fill peaks step will use the background noise in the area where the peak should be expected to determine the missing peak value. Because there may be biological reasons that a peak is missing in one class of samples not the others, the user can define the number of peaks that need to be found in a single class.

4 XCMS R Package

As commented before, XCMS comes as an open-source R package. The open source nature of XCMS allows it to be updated and improved by the community.

The R version of XCMS is a command line driven interface that lets users interact with the data directly on their systems via a collection of commands. As part of this chapter we have included workflows for both the original and alternative pipelines.

The updated XCMS (named XCMS 3 to avoid confusion from XCMS² [14]) uses new R capabilities, packages and objects to reduce memory requirements. For instance, now XCMS 3 reduces memory requirements by not loading entire raw data files into memory. Instead, it only loads selections of what is currently being handled. Additionally, the new objects also allow users to save their object files and see what methods have been called on the data, stored in the history of the object.

5 XCMS Online: Extended Capabilities

XCMS Online, which was first introduced in 2012 [15], allows users to have a cloud based and graphical system to use XCMS. The added benefits of this are that local hardware is no longer required to process data and data can be easily shared between users. Users have a selection of simple options to set themselves on the data processing path. Moreover, it allows users to perform different processing types (job types) depending on their experiment. These job types are outlined below:

Single job—Either a single file or single class of data.

Pairwise job—The classic knockout vs. wild-type experiment where a simple two class system is needed. Statistical choices can be parametric or nonparametric with both unpaired samples and paired sample types.

Multigroup job—A very dynamic job type, useful for time series where many classes are going to be compared. Statistical choices are ANOVA and Kruskal-Wallis, both with post hoc analysis.

In addition, XCMS Online allows data to be streamed directly to the cloud for processing. It allows for advanced metabolite annotation based on in-source fragment matching, and it allows for systems biology analysis and multi-omics integration (Fig. 1). More recently, XCMS-MRM Online, the targeted counterpart of XCMS, was released [16]. The following section gives an overview of these extended capabilities.

5.1 Data Streaming with XCMS

Data streaming emulates the streaming video platforms where users can directly watch movies without the need to wait until the movie has been completely downloaded. In the case of LC-MS data, we typically have to wait until all samples have been acquired and upload all the samples to the XCMS cloud-based system. However, XCMS Online allows LC-MS data to be streamed [17]. By installing a program on their instrument computer and login to XCMS Online, users can stream their data directly to the cloud and start processing the data while the data of the rest of samples are being acquired. The parameters for the job can be chosen beforehand. The streaming process dramatically reduces the overall dead time from acquisition to processing.

5.2 MISA: METLIN-Guided In-Source Annotation

XCMS Online also integrates an algorithm for advanced metabolite annotation, known as the METLIN-guided in-source annotation (MISA) algorithm [18]. This algorithm aims at using the information from in-source fragments (ISF) naturally occurring in MS^1 data to provide a robust peak and metabolite annotation. We

observed that more than 80% of molecules in METLIN readily dissociate into multiple fragments at low collision energies [6]. This implies that a considerable fraction of features usually observed in untargeted experiments are ISF. Existing tools for metabolite annotation typically take into account only common neutral losses as a result of in-source fragmentation. These common neutral losses are not specific for each metabolite. On the other hand, ISF provide more specific information on the metabolite due to low-energy fragmentation.

In-source fragments observed in LC-MS are similar to those observed in low-energy collision-induced dissociation (CID) tandem MS (MS/MS) spectra. In that sense, low-energy spectra in METLIN can be used to guide this annotation and detect the presence of specific ISF from metabolites with experimental MS/MS spectra in METLIN (Fig. 4). The current version of

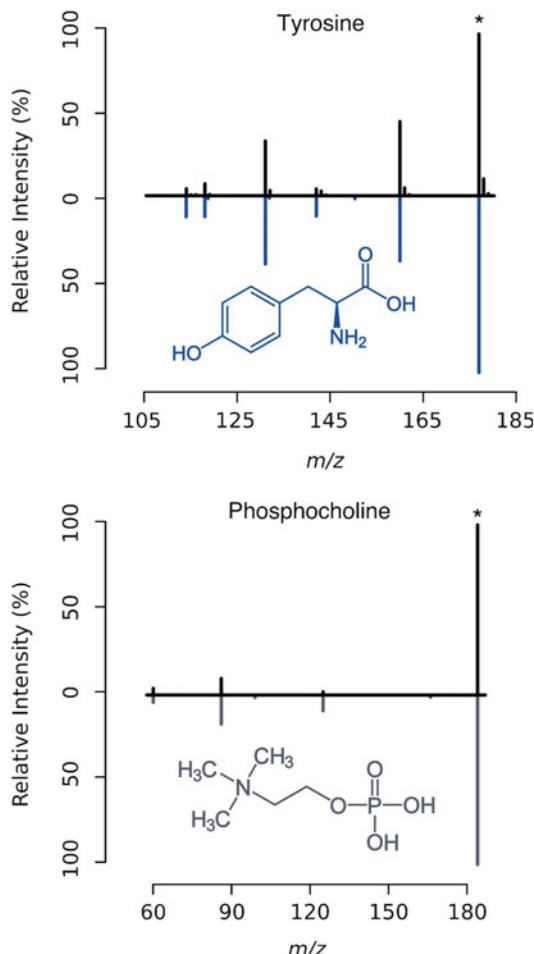


Fig. 4 In-source fragment annotation examples. LC-MS experimental peaks (black) and low energy MS/MS spectra extracted from METLIN (color, negatively rotated), for the metabolites tyrosine and phosphocholine. Protonated ion/precursor is noted (asterisk)

METLIN now contains experimental MS/MS spectra for more than 500,000 molecular standards in both positive and negative mode and at different collision energies, including experimental spectra acquired at low-energy (0 and 10 eV).

MISA is applied after data is processed by XCMS Online. MISA searches for features corresponding to protonated/deprotonated or adduct ion species from available MS/MS spectra in METLIN. Next, MISA searches for the corresponding ISF of those molecules. If both the protonated/deprotonated ion species and one or more ISF are found, MISA assigns the feature with a putative identity. In addition, it provides two scores to assess the likelihood of that annotation being correct. More information on both MISA and the scores can be found in the original study [18].

5.3 XCMS-Guided Systems Biology

A system levels analysis can provide important insights to understand biochemical mechanisms. XCMS online enables metabolomics data to be projected onto metabolic pathways and integrate it with transcriptomics and proteomics data [19].

To project quantitative metabolic data onto metabolic networks, first, metabolites need to be identified. Metabolite identification is a process that heavily relies on manual labor and expert curation. XCMS Online uses an automated predictive pathway analysis method, developed by Li et al. and known as Mumichog [20], that bypasses metabolite identification and instead uses biochemical information to annotate features and project them onto metabolic pathways. Scientist can then interpret the data and formulate biochemical mechanisms and hypothesis, and then confirm these identifications via tandem MS analysis. More details of this algorithm can be found in the original study [20] and Chapter 19 of this book.

In that sense, in XCMS Online-guided systems biology, users can directly map their results into metabolic networks, without the need to transfer data among different applications. Users can upload gene and protein data to overlay it within the pathways. Results are shown as a table and also by an interactive Pathway Cloud plot. The Pathway Cloud plot shows dysregulated pathways, ordering them by overlap percentage to other omics data and statistical significance. To the date, there are over 7600 metabolic models available for pathway analysis from BioCyc4 v19.5–20.0.

5.4 XCMS-MRM

Small molecule quantification is performed using triple quadrupole MS configured for multiple reaction monitoring (MRM) [21]. MRM is considered the gold standard for targeted quantitative analysis due to its high sensitivity and specificity. In order to process MRM data, XCMS-MRM was designed and it constitutes the XCMS Online counterpart for processing data from targeted metabolomics assays [16]. XCMS-MRM recognizes raw data files in any vendor format, providing automatic transition peak

detection, area integration and alignment, minimizing false peak integration, and reducing data analysis time. Absolute concentrations can be acquired through stable-isotope dilution, external calibration and standard addition methods. XCMS-MRM quantitative results include quality control indicators to assess accuracy and specificity, and limits of detection. It also assesses statistical significance for metabolite concentration changes. The statistical graphics and result visualization are unique features of XCMS-MRM.

6 Conclusions

LC-MS is a widely used metabolomics technology, and XCMS enables it by meeting the demands of this ever-growing scientific community. In this chapter, we have described the underlying algorithms of XCMS, and the XCMS Online platform. XCMS is a software program for LC-MS data processing, implemented as an R package, widely used by the community and actively maintained. XCMS Online is a cloud-based resource and extends its capabilities to a full suite for metabolomics data analysis. Hosted on high-performance dedicated servers at the Scripps Research Institute, XCMS Online offers outstanding processing speed that is now easily accessed by thousands of users. Together, with a freely available R package and a cloud-based data processing service, XCMS facilitates data analysis and result sharing across all vendor platforms.

Acknowledgments

This research was partially funded by the US National Institutes of Health grants R35 GM130385, P30 MH062261, P01 DA026146 and U01 CA235493; and by Ecosystems and Networks Integrated with Genes and Molecular Assemblies (ENIGMA), a Scientific Focus Area Program at Lawrence Berkeley National Laboratory for the U.S. Department of Energy, Office of Science, Office of Biological and Environmental Research, under contract number DE-AC02-05CH11231. This research benefited from the use of credits from the National Institutes of Health (NIH) Cloud Credits Model Pilot, a component of the NIH Big Data to Knowledge (BD2K) program.

References

1. Patti GJ, Yanes O, Siuzdak G (2012) Metabolomics: the apogee of the omic triology. *Nat Rev Mol Cell Biol* 13(4):263–269. <https://doi.org/10.1038/nrm3314>
2. Vinaixa M, Samino S, Saez I, Duran J, Guinovart JJ, Yanes O (2012) A guideline to univariate statistical analysis for LC/MS-based untargeted metabolomics-derived data. *Metabolite* 2(4):775–795. <https://doi.org/10.3390/metabo2040775>
3. Smith CA, Want EJ, O'Maille G, Abagyan R, Siuzdak G (2006) XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Anal Chem* 78(3):779–787. <https://doi.org/10.1021/ac051437y>
4. Mikko K, Miettinen J, Oresic M (2006) MZmine: toolbox for processing and visualization of mass spectrometry based molecular profile data. *Bioinformatics* 22(5):634–636. <https://doi.org/10.1093/bioinformatics/btk039>
5. Tomás P, Castillo S, Villar-Briones A, Oresic M (2010) MZmine 2: modular framework for processing, visualizing, and analyzing mass spectrometry-based molecular profile data. *BMC Bioinformatics* 11:395. <https://doi.org/10.1186/1471-2105-11-395>
6. Domingo-Almenara X, Montenegro-Burke JR, Benton PH, Siuzdak G (2018) Annotation: a computational solution for streamlining metabolomics analysis. *Anal Chem* 90(1):480–489. <https://doi.org/10.1021/acs.analchem.7b03929>
7. Mahieu NG, Patti GJ (2017) Systems-level annotation of a metabolomics data set reduces 25 000 features to fewer than 1000 unique metabolites. *Anal Chem* 89(19):10397–10406. <https://doi.org/10.1021/acs.analchem.7b02380>
8. Lin W, Xing X, Chen L, Yang L, Su X, Rabitz H, Lu W, Rabinowitz JD (2019) Peak annotation and verification engine for untargeted LC–MS metabolomics. *Anal Chem* 91(3):1838–1846. <https://doi.org/10.1021/acs.analchem.8b03132>
9. Trinh DK, Wahl S, Raffler J, Molnos S, Laimighofer M, Adamski J, Suhre K et al (2018) Characterization of missing values in untargeted MS-based metabolomics data and evaluation of missing data handling strategies. *Metabolomics* 14(10). <https://doi.org/10.1007/s11306-018-1420-2>
10. Runmin W, Wang J, Su M, Jia E, Chen S, Chen T, Ni Y (2018) Missing value imputation approach for mass spectrometry-based metabolomics data. *Sci Rep* 8. <https://doi.org/10.1038/s41598-017-19120-0>
11. Vereyken L, Dillen L, Vreeken RJ, Cuyckens F (2019) High-resolution mass spectrometry quantification: impact of differences in data processing of centroid and continuum data. *J Am Soc Mass Spectrom* 30(2):203–212. <https://doi.org/10.1007/s13361-018-2101-0>
12. Tautenhahn R, Böttcher C, Neumann S (2008) Highly sensitive feature detection for high resolution LC/MS. *BMC Bioinformatics* 9:504. <https://doi.org/10.1186/1471-2105-9-504>
13. Prince JT, Marcotte EM (2006) Chromatographic alignment of ESI-LC-MS proteomics data sets by ordered bijective interpolated warping. *Anal Chem* 78(17):6140–6152. <https://doi.org/10.1021/ac0605344>
14. Benton HP, Wong DM, Trauger SA, Siuzdak G (2008) XCMS2: processing tandem mass spectrometry data for metabolite identification and structural characterization. *Anal Chem* 80(16):6382–6389. <https://doi.org/10.1021/ac800795f>
15. Tautenhahn R, Patti GJ, Rinehart D, Siuzdak G (2012) XCMS online: a web-based platform to process untargeted metabolomic data. *Anal Chem* 84(11):5035–5039. <https://doi.org/10.1021/ac300698c>
16. Domingo-Almenara X, Montenegro-Burke JR, Ivanisevic J, Thomas A, Sidibé J, Teav T, Guijas C et al (2018) XCMS-MRM and METLIN-MRM: a cloud library and public resource for targeted analysis of small molecules. *Nat Methods* 15(9):681–684. <https://doi.org/10.1038/s41592-018-0110-3>
17. Montenegro-Burke JR, Aisporna AE, Benton HP, Rinehart D, Fang M, Huan T, Warth B et al (2017) Data streaming for metabolomics: accelerating data processing and analysis from days to minutes. *Anal Chem* 89(2):1254–1259. <https://doi.org/10.1021/acs.analchem.6b03890>
18. Domingo-Almenara X, Montenegro-Burke JR, Guijas C, Majumder EL-W, Benton HP, Siuzdak G (2019) Autonomous METLIN-guided in-source fragment annotation for untargeted metabolomics. *Anal Chem* 91(5):3246–3253. <https://doi.org/10.1021/acs.analchem.8b03126>

19. Tao H, Forsberg EM, Rinehart D, Johnson CH, Ivanisevic J, Paul Benton H, Fang M et al (2017) Systems biology guided by XCMS online metabolomics. *Nat Methods* 14 (5):461–462. <https://doi.org/10.1038/nmeth.4260>
20. Li S, Park Y, Duraisingham S, Strobel FH, Khan N, Soltow QA, Jones DP, Pulendran B (2013) Predicting network activity from high throughput metabolomics. *PLoS Comput Biol* 9(7). <https://doi.org/10.1371/journal.pcbi.1003123>
21. Vinzenz L, Picotti P, Domon B, Aebersold R (2008) Selected reaction monitoring for quantitative proteomics: a tutorial. *Mol Syst Biol* 4:222. <https://doi.org/10.1038/msb.2008.61>



Chapter 3

Metabolomics Data Preprocessing Using ADAP and MZmine 2

**Xiuxia Du, Aleksandr Smirnov, Tomáš Pluskal, Wei Jia,
and Susan Sumner**

Abstract

The informatics pipeline for making sense of untargeted LC–MS or GC–MS data starts with preprocessing the raw data. Results from data preprocessing undergo statistical analysis and subsequently mapped to metabolic pathways for placing untargeted metabolomics data in the biological context. ADAP is a suite of computational algorithms that has been developed specifically for preprocessing LC–MS and GC–MS data. It consists of two separate computational workflows that extract compound-relevant information from raw LC–MS and GC–MS data, respectively. Computational steps include construction of extracted ion chromatograms, detection of chromatographic peaks, spectral deconvolution, and alignment. The two workflows have been incorporated into the cross-platform and graphical MZmine 2 framework and ADAP-specific graphical user interfaces have been developed for using ADAP with ease. This chapter summarizes the algorithmic principles underlying key steps in the two workflows and illustrates how to apply ADAP to preprocess LC–MS and GC–MS data.

Key words ADAP, MZmine 2, Metabolomics, LC–MS, GC–MS, Data preprocessing, Peak picking, Alignment, Spectral deconvolution, Visualization

1 Introduction

Untargeted metabolomics, detection and relative quantitation of ideally all metabolites in a biological system, has become a powerful discovery tool in many scientific disciplines. It has benefited greatly from advances in mass spectrometry (MS) and chromatography. As a result, liquid chromatography (LC) and gas chromatography (GC) coupled to mass spectrometry (MS) have become primary analytical platforms for untargeted metabolomics.

The informatics pipeline for making sense of the resulting LC–MS and GC–MS data involves preprocessing of the raw mass spectral data to detect chemical species, assignment of specific metabolites to these species, and integration of these metabolites into a coherent and physiologically meaningful integrated multi-omics framework that can yield a holistic understanding of the biological

system (Fig. 1). As the first step of this informatics pipeline, data preprocessing is critical for the success of a metabolomics study because preprocessing errors can propagate downstream into spurious or missing compound identifications and cause misinterpretation of the metabolome. Data preprocessing workflows (Fig. 2) in open-source software tools generally consist of four sequential steps after masses have been detected from profile mass spectra (i.e., converting mass spectra from profile to centroid format). These four steps are construction of extracted ion chromatograms (EIC), detection of chromatographic peaks from EICs, peak grouping for LC–MS data or spectral deconvolution for GC–MS data, and alignment. ADAP (Automated Data Analysis Pipeline) is one such open-source workflow that has been developed for preprocessing both GC–MS and LC–MS data and incorporated into the MZmine 2 framework [1].

In sections below, we first briefly describe the evolution of ADAP and subsequently focus on describing how to carry out LC–MS and GC–MS preprocessing workflows using primarily ADAP modules in the MZmine 2 framework. These modules include EIC construction, chromatographic peak detection, spectral deconvolution, and alignment. To facilitate describing how to specify relevant parameters, we provide a brief summary of the algorithmic principles underlying these ADAP modules. To make the workflow complete and this chapter self-contained, we provide information on how to carry out other essential tasks using non-ADAP modules in MZmine 2. These tasks include: (1) import raw LC–MS and GC–MS data files into MZmine 2 and inspect them, (2) detect masses from profile mass spectra (step 1 in Fig. 2), (3) group chromatographic peaks (step 4 in Fig. 2) detected in LC–MS data, and (4) export preprocessing results for downstream statistical analysis, metabolite identification, and -omics data integration.

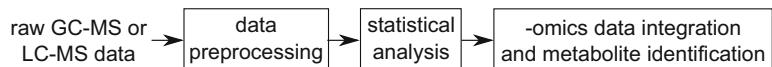


Fig. 1 Informatics pipeline for making sense of untargeted GC–MS and LC–MS metabolomics data

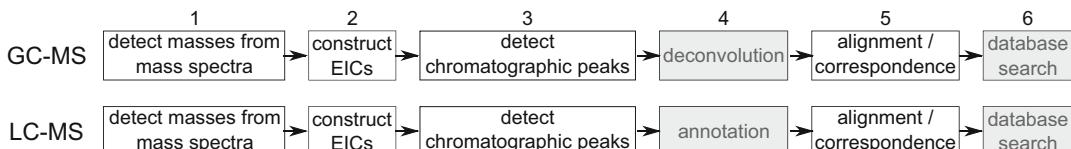


Fig. 2 General computational workflows for preprocessing GC–MS and LC–MS data in existing open-source software tools. EIC stands for extracted ion chromatogram

2 Evolution of ADAP

Development of ADAP started in 2009. The first version of ADAP was developed by Jiang et al. for fully automated preprocessing of raw GC–MS untargeted metabolomics data [2]. It was comprised of a suite of algorithms for steps 2–5 of the preprocessing workflow for GC–MS data (Fig. 2). As a critical step in this ADAP-GC workflow, spectral deconvolution has undergone significant improvements over the years made by Ni et al. [3, 4] for ADAP-GC 3.0 and ADAP-GC 3.0 and Smirnov et al. [5] for ADAP-GC 3.2.

The year of 2016 witnessed research and development efforts by Myers et al. to equip ADAP with the capability to preprocess high mass resolution LC–MS untargeted metabolomics data while addressing the high rate of false positive peaks that had been reported [6, 7]. Toward this end, ADAP algorithms were developed for constructing extracted ion chromatograms (EICs) and for detecting chromatographic peaks from EICs. Following peak detection, alignment methods in MZmine 2 can be used to correct the retention time shift from sample to sample for a complete preprocessing workflow for LC–MS data.

All of the aforementioned ADAP algorithms were written in Java and have been incorporated into the open-source and graphical MZmine 2 framework. Furthermore, specific user-friendly graphical user interfaces (GUI) have been developed to facilitate users with using the ADAP modules within MZmine 2.

3 Install MZmine 2

MZmine 2 can be downloaded at <http://mzmine.github.io/download.html>. To start MZmine 2, users should unzip the downloaded file and then open MZmine 2 by running the following script files according to the operating system of your computer.

1. startMZmine_MacOSX.command
2. startMZmine_Windows.bat
3. startMZmine_Linux.sh

4 Preprocessing Workflow for LC–MS Data

4.1 Import and Inspection of Raw Data Files

Raw data files are imported into MZmine 2 using the Raw data methods drop-down menu as shown in Fig. 3. Acceptable file formats include mzXML and netCDF. One of the greatest strengths of MZmine 2 lies in the rich built-in visualization functions that allow users to inspect the raw data, which greatly

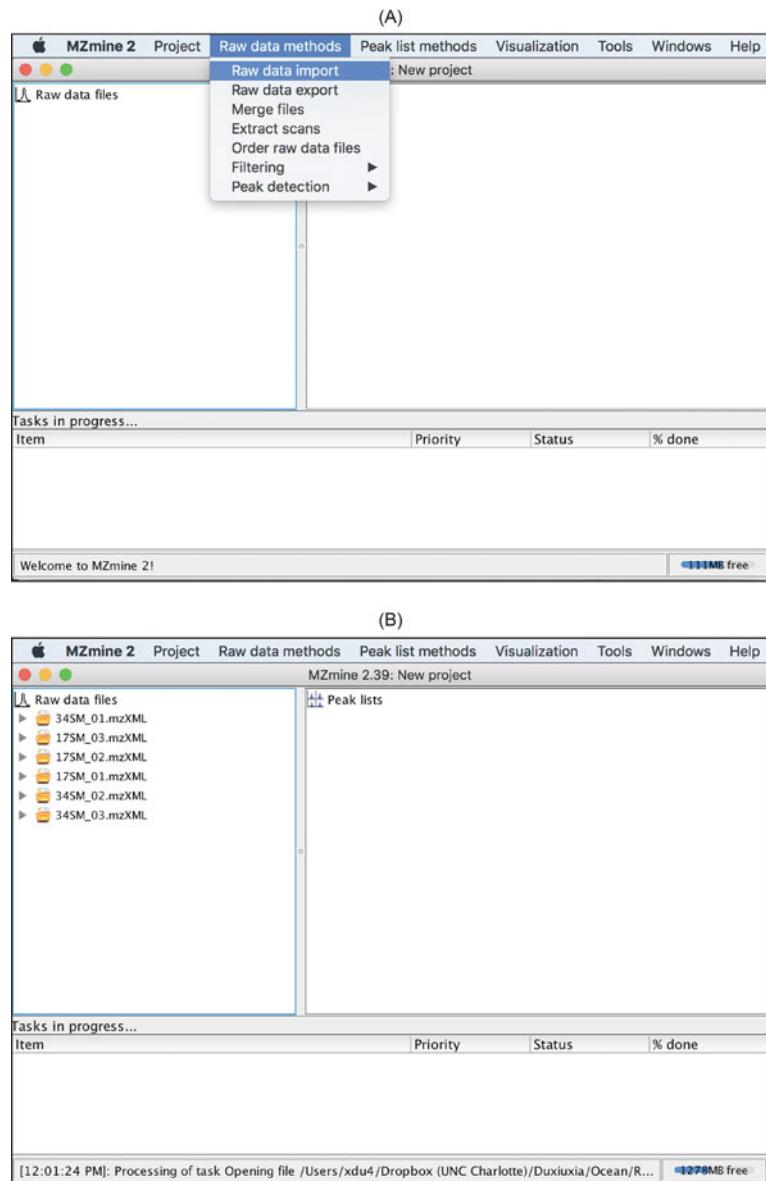


Fig. 3 Import raw data into MZmine 2. **(a)** Raw data import is in the Raw data methods drop-down menu. **(b)** Imported raw data files are listed under Raw data files on the left

facilitates users with understanding the data and making informed decisions to specify preprocessing parameters. Herein we demonstrate only the visualization capabilities that can inform data preprocessing. Readers are advised to explore the other visualization capabilities in MZmine 2.

Display of Raw Mass Spectra: Figure 4 shows the MZmine 2 capabilities to display raw spectra and the spectra meta data that includes spectra level (MS1 or MS2), acquisition time, type (p represents profile or c represents centroid), and polarity (+ represents positive and - represents negative).

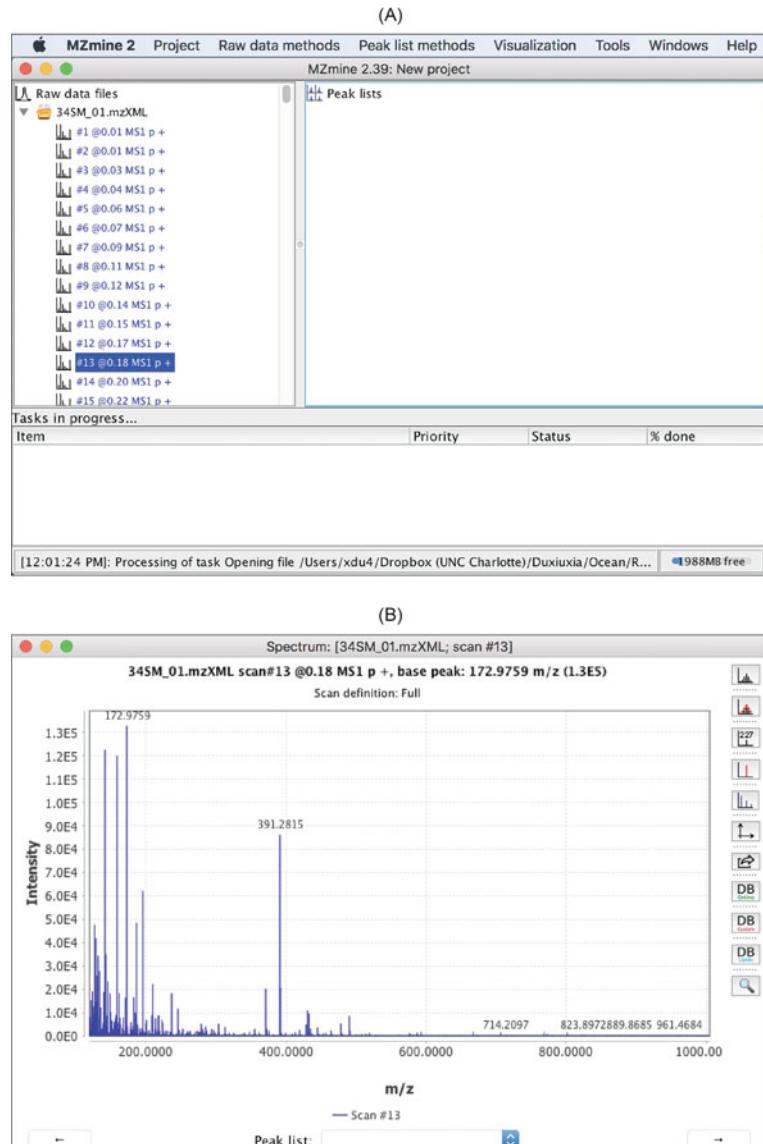


Fig. 4 Capabilities of MZmine 2 that allow users to inspect the raw spectra. (a) List of spectra in a raw data file and the spectra meta data. (b) Double click on any spectrum opens up a separate window displaying it

Display of Chromatograms: Base peak chromatograms (BPC) and total ion chromatograms (TIC) can reveal retention time shift among the data files and the approximate amount of retention time correction that is needed via alignment. Figure 5 displays the BPCs of 12 data files.

Display of m/z and Retention Time: The 2D visualizer in the Visualization drop-down menu can provide an overview of the ion species that should be detected by the peak picking algorithms from the entire data file (Fig. 6). Each ion species is characterized by a unique pair of m/z (y-axis) and retention time (x-axis).

4.2 Detect Masses from Profile Mass Spectra

The mass detection step detects mass centroids from profile mass spectra. MZmine 2 provides five centroiding methods that include Centroid, Exact mass, Local maxima, Recursive threshold, and Wavelet transform. The Centroid mass detector is for spectra that have been centroided and the other four detectors are for profile mass spectra only. The Exact mass detector is suitable for high-resolution MS data, such as provided by FTMS instruments. The Local maxima mass detector simply detects all local maxima within a spectrum, except those signals below the specified noise level. The Recursive threshold mass detector is suitable for data that has too much noise for the Exact mass detector to be used. The Wavelet transform mass detector is suitable for both high-resolution and low-resolution data. It uses the Ricker wavelet (also called Mexican Hat wavelet) and carry out a continuous wavelet transform (CWT) of the continuous profile spectra.

This Wavelet transform mass detector provides a sensitive and robust way to detect masses (Fig. 7) and we describe it in more detail herein. It requires users to set three parameters: *noise level*, *scale level*, and *wavelet window size*. *Noise level* specifies the minimum intensity level for a data point to be considered part of a spectrum. All data points below this intensity level are ignored. *Scale level* is the scale factor that either dilates or compresses the wavelet signal. When it is small (e.g., below 10), the Ricker wavelet is more contracted which in turn results in more noisy peaks being detected. *Wavelet window size (%)* is the size of the window used to calculate the wavelet signal. When the size of the window is small, more noisy peaks can be detected. Among the three parameters, *scale level*, in particular, can have a large impact on mass detection.

When the *scale level* is small, a significant number of very narrow noise peaks can be detected. They are passed to the subsequent EIC construction and can form false EIC peaks. As the *scale level* increases, the number of detected noise peaks decreases. However, a larger *scale level* could cause a noticeable shift in the centroid m/z values. Figure 8c, d depicts the m/z values

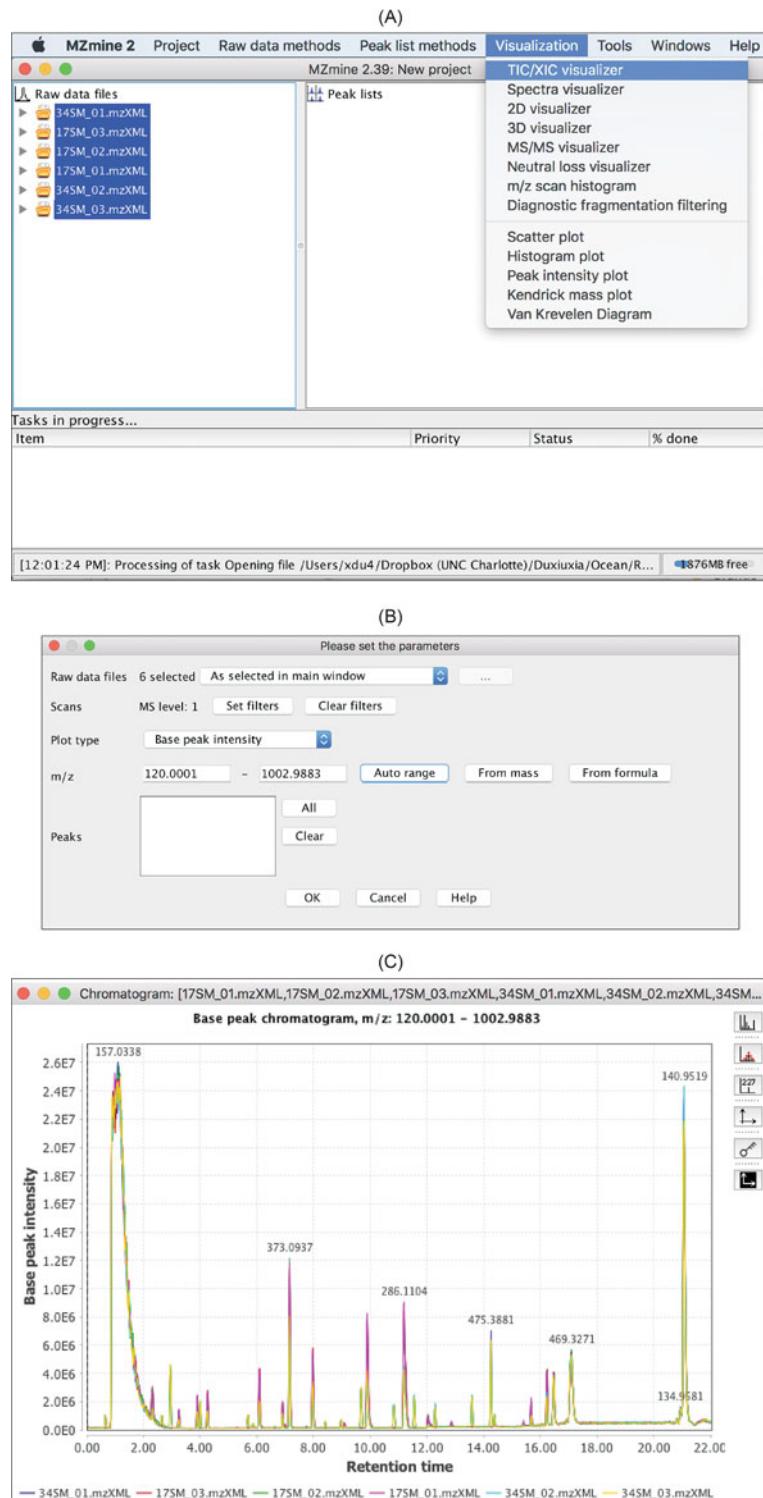


Fig. 5 Inspect BPCs. (a, b) Display BPCs by using the TIC/XIC visualizer in the visualization drop-down menu. (c) BPCs of 12 data files

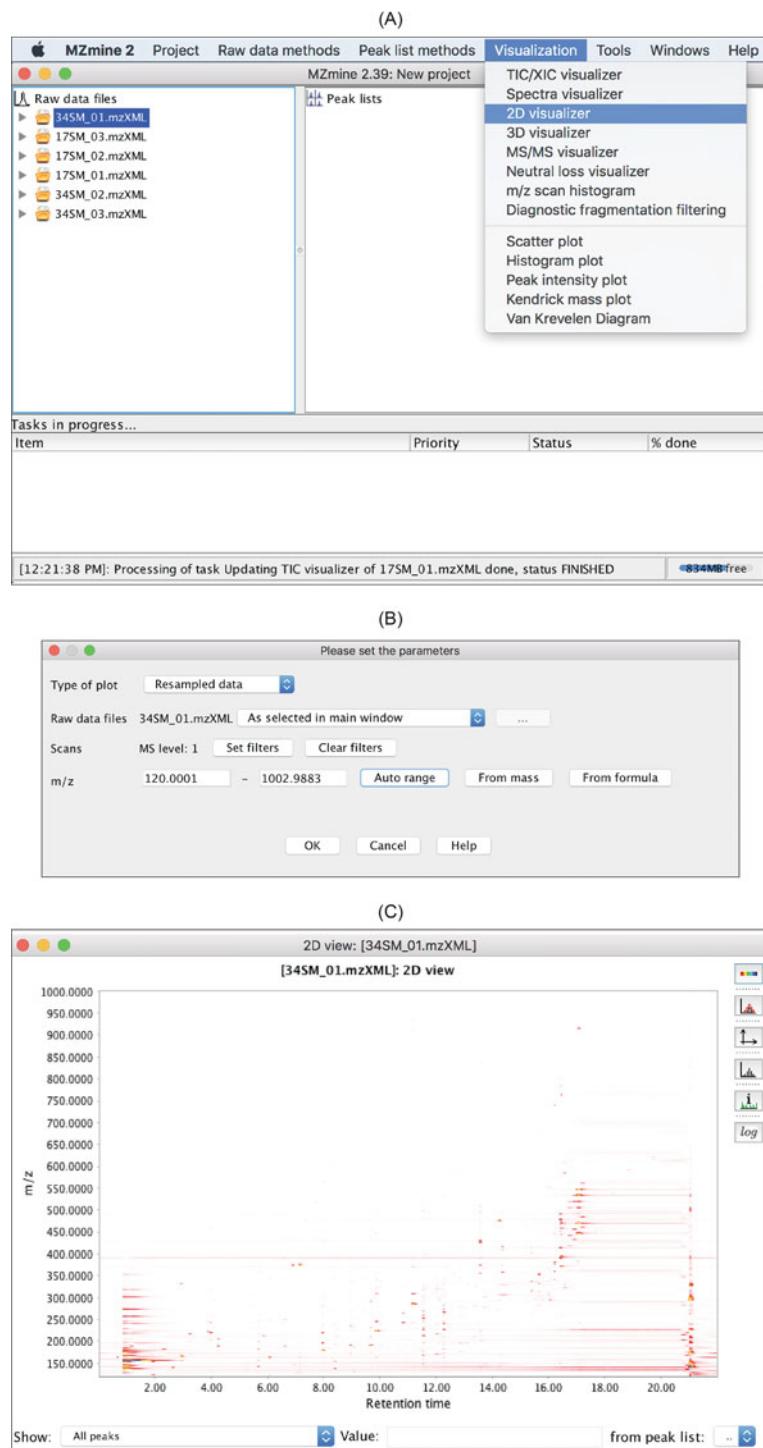


Fig. 6 2D visualization of a raw data file. (a, b) 2D visualizer can be accessed via the Visualization drop-down menu. (c) Ion species are displayed via the 2D visualization

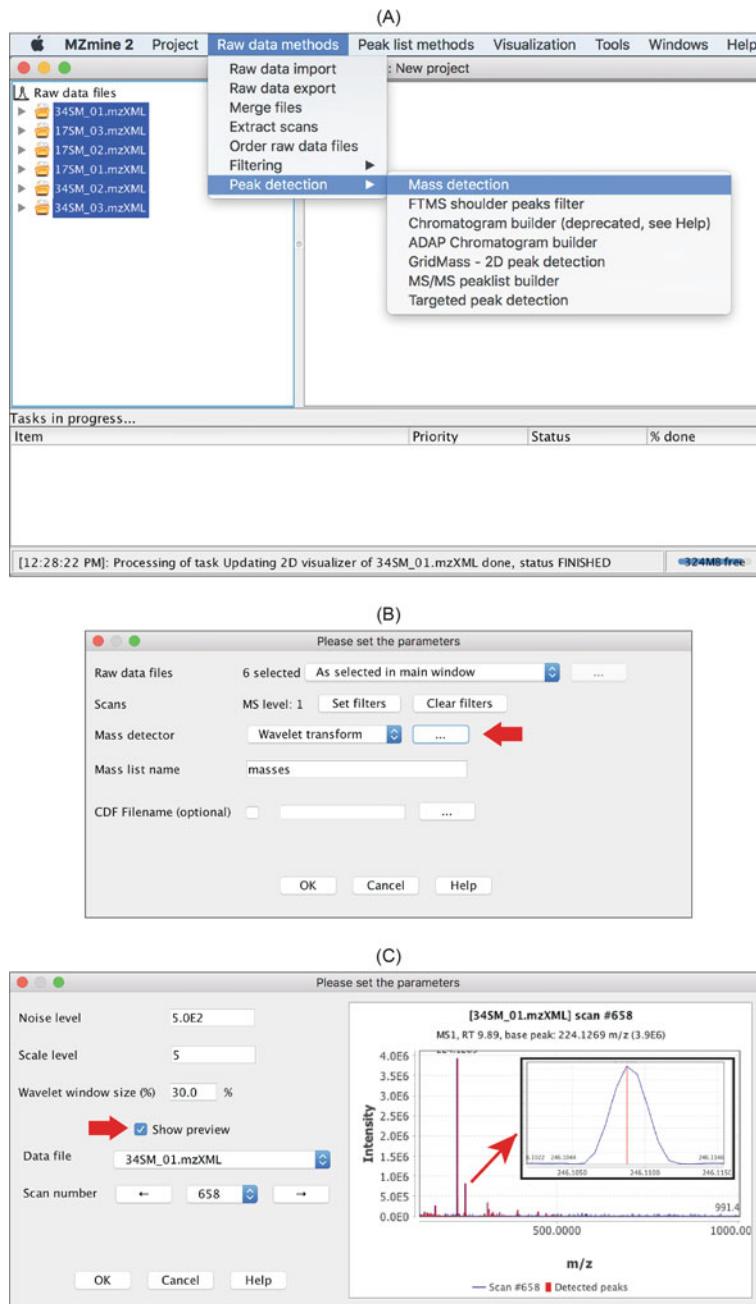


Fig. 7 Mass detection in MZmine 2. (a) The Mass detection method can be accessed via the Raw data methods draw-down menu. (b) Wavelet transform is one of the mass detection methods. Click the button pointed to by the red arrow would open the window in (c) for specifying parameters (c) User-defined parameters for the wavelet transform method. Check Show preview opens up the preview window. Effect of parameter changes is displayed almost immediately, which greatly facilitates specifying parameters. The inset shows the profile mass peak in blue and the detected mass centroid in red

detected from consecutive scans when *scale levels* are set at 5 and 15, respectively. Compared to the m/z values detected at *scale level* equal to 5, most of the m/z values detected at *scale level* 15 are larger. When the final representative m/z for a chromatographic peak is calculated as the weighted average of all of the centroid m/z values along the EIC as shown in Fig. 8b, the difference in the final representative m/z values between using *scale level*=5 and *scale level*=15 is \sim 19 ppm. This difference in the mass values is big enough to cause different compounds to be eventually identified.

Regardless of which of the mass detectors is used, the results of mass detection for a particular profile mass spectrum can be accessed by clicking masses under the profile mass spectrum (Fig. 9). It is relevant to note that mass detection can also be carried out by using *msConvert* that is part of *ProteoWizard* [8]. *msConvert* detects masses by either using a CWT-based method or calling

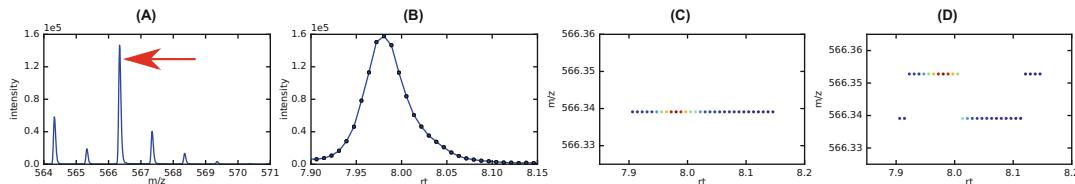


Fig. 8 Differences in the resulting mass values caused by different *scale levels* when using the wavelet transform-based mass detection in MZmine 2. (a) One of the consecutive mass spectra from which the mass indicated by red arrow is to be detected. (b) The EIC of the mass. The mass values of the blue dots along the elution profile are depicted in (c) and (d) with *scale level* being 5 and 15, respectively

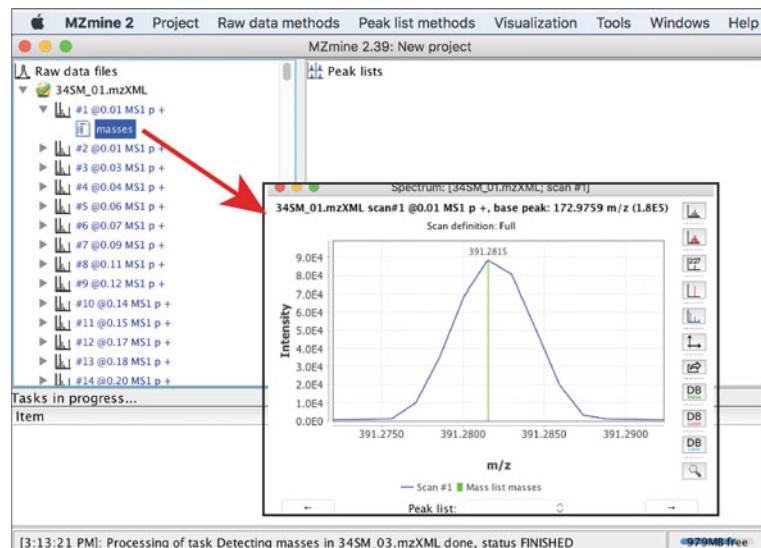


Fig. 9 Examine mass detection results for a particular profile mass spectrum. Vertical lines in green indicate mass values that have been detected

functions provided by vendors of mass spectrometers. The resulting centroid data can be imported into MZmine 2 for data preprocessing.

4.3 Construct EICs by ADAP

In untargeted metabolomics, the masses of ion species that have been detected by a mass analyzer are unknown prior to data preprocessing. It is up to the step of EIC construction to determine. With mass centroids detected from profile mass spectra, construction of EICs can begin. Figure 10 shows how to carry out this step using ADAP. ADAP examines all of the data points in the entire data file and works from the largest intensity data point down to the smallest. As a result, a list of ions is produced that have been detected by the mass analyzer over a continuous retention time period. This approach in constructing EICs is in contrast to the EIC construction process in other open-source software tools such as XCMS where EICs are built chronologically in retention time. The advantage of starting an EIC from the highest intensity point among all of the data points belonging to this EIC is that the reference mass for the EIC has the highest possible mass measurement accuracy. This is particularly important for TOF-type mass analyzers whose mass measurement accuracy tends to be higher for more intense signals.

Construction of EICs by ADAP requires that the following four parameters be specified:

- (a) *Min group size in number of scans*. In the entire chromatogram there must be at least this number of sequential scans having points above the *Group intensity threshold* set by the user.
- (b) *Group intensity threshold*. See above
- (c) *Min highest intensity*. There must be at least one point in the chromatogram that has an intensity greater than or equal to this value.
- (d) *m/z tolerance*. Maximum *m/z* difference of data points in consecutive scans in order to be connected to the same chromatogram.

As a result of the EIC construction, a list of EICs is produced for each data file (Fig. 10c). Each EIC can be examined by double clicking it and opening up a window as shown in Fig. 11.

4.4 Detect Chromatographic Peaks by ADAP

After EICs have been constructed, ADAP detects chromatographic peaks from each of these EICs using the continuous wavelet transform (CWT) that is similar to what the wavelet transform mass detector uses. Specifically, wavelet coefficients are first calculated as the inner product between the EIC and the Ricker wavelets at different wavelet scales and locations. Subsequently, peak location

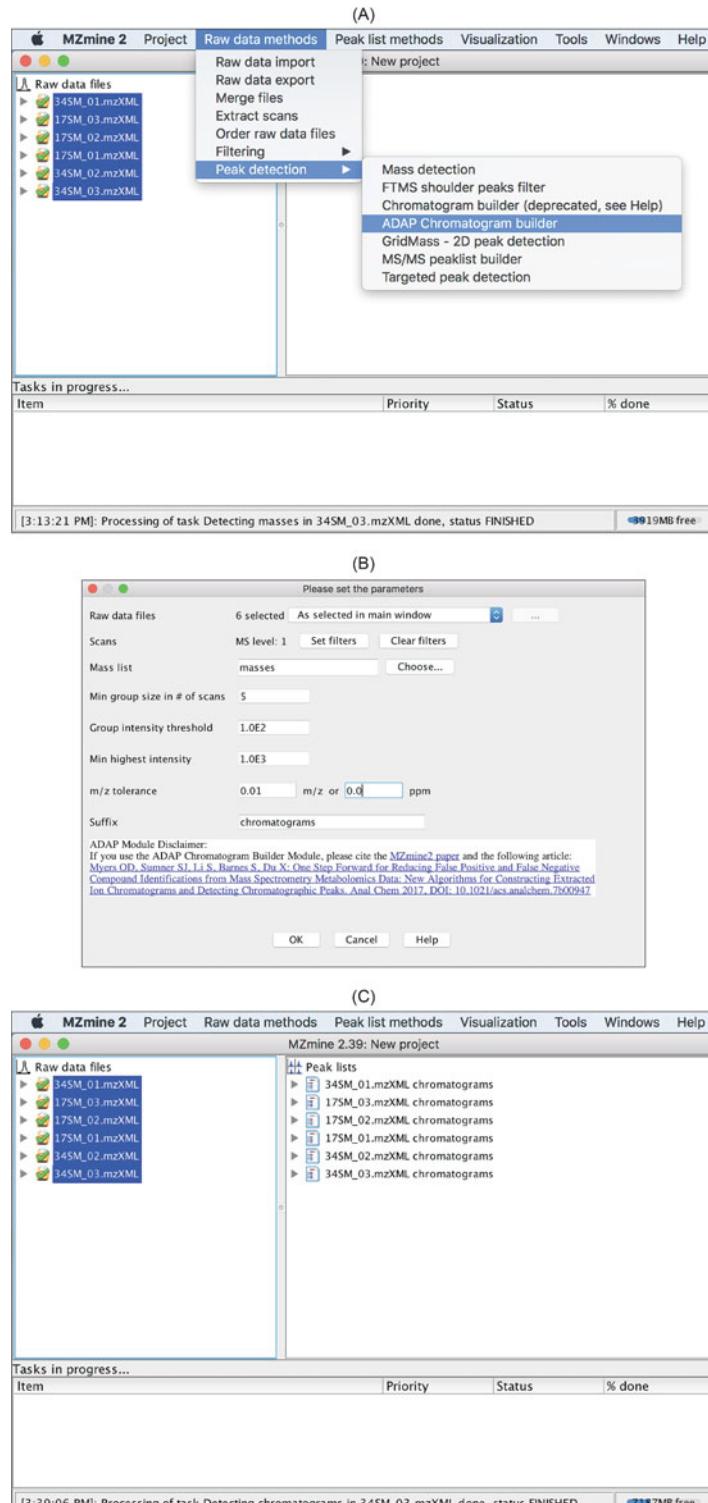


Fig. 10 Construction of EICs in ADAP. (a) EIC construction is achieved by using the ADAP Chromatogram builder method. (b) Parameters relevant to this method. (c) A list of EICs is produced for each data file

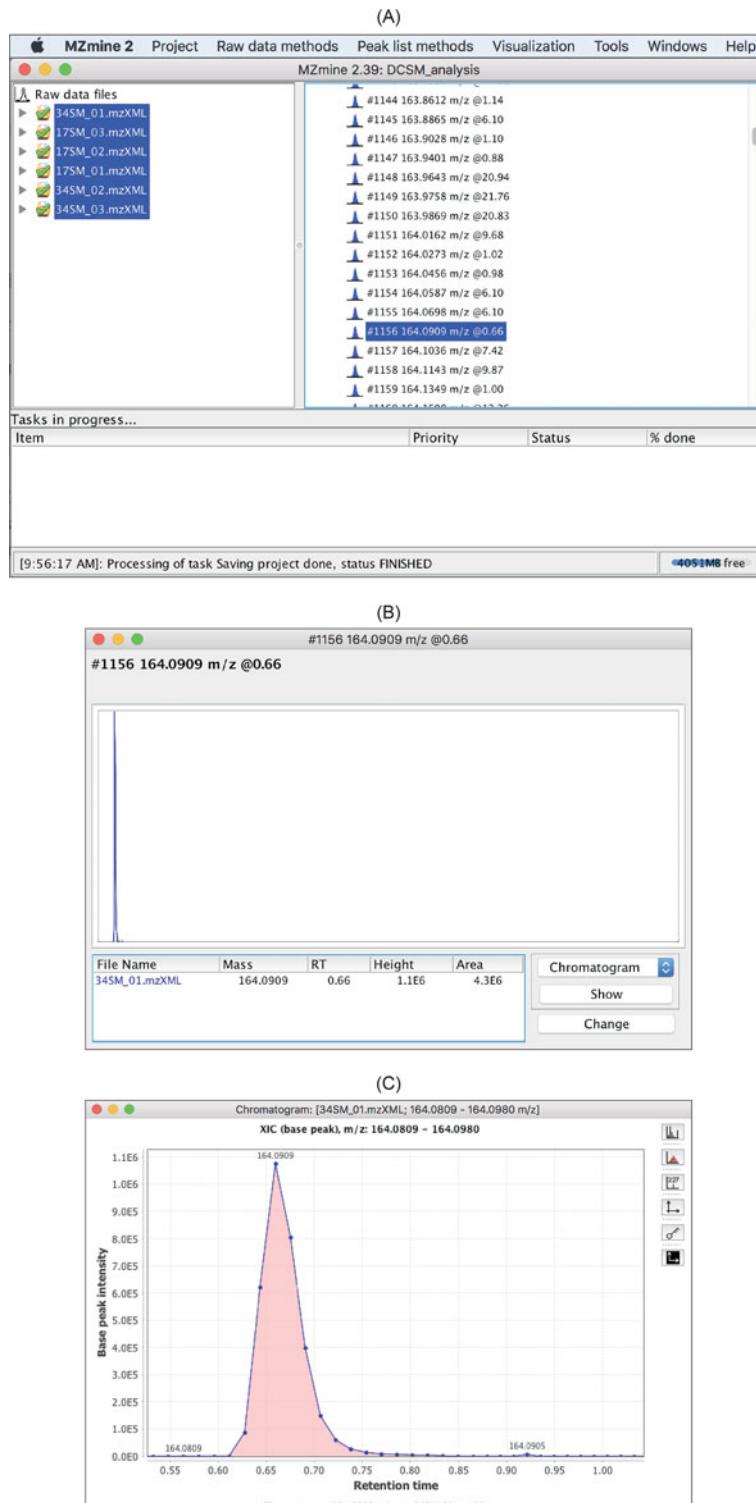


Fig. 11 Examine EICs. (a) Select a particular EIC. (b) Double click the selected EIC opens this window. Select Chromatogram and click Show opens the EIC in (c) for visual examination

and boundaries are determined through ridgeline detection and simple local minima search. Finally, peak boundaries are adjusted using a local minima search. This boundary adjustment is necessary because the rough estimates for the left and right boundary based on ridgeline detection are symmetric, i.e., having the same distance from the peak location.

This ADAP peak detection method is accessed via Chromatogram deconvolution in the Peak list methods drop-down menu (Fig. 12a). To choose the parameters appropriately, we strongly recommend that users check the Show preview box. The preview function allows a user to see the effect of parameter changes immediately on peak detection for a chosen EIC. Any EIC from any of the data files can be chosen using the Peak list and Chromatogram drop-down menu (Fig. 12b). The following six parameters need to be specified:

- (a) SNR Threshold. Signal-to-noise threshold to filter out noise peaks. For details about how SNR is calculated, we refer readers to the publication by Myers et al. [9].
- (b) Min feature height. The smallest intensity a peak can have and be considered a real feature.
- (c) Coefficient/area threshold. The best coefficient (largest inner product of wavelet with peak in ridgeline) divided by the area under the curve of the feature.
- (d) Peak duration range. The acceptable range of peak widths. Peaks with widths outside this range will be rejected.
- (e) RT wavelet range. The range of wavelet scales used to build matrix of coefficients. Scales are expressed as RT values (minutes) and correspond to the range of wavelet scales that will be applied to the chromatogram. Choose a range that is similar to the range of peak widths expected to be found from the data.

4.5 Alignment

Alignment intends to identify corresponding peaks across samples. MZmine 2 provides four alignment algorithms: Join aligner, RANSAC aligner, Hierarchical aligner (GC), and ADAP Aligner. The first two algorithms, Join aligner and RANSAC aligner, are for aligning LC-MS data and the latter two, Hierarchical aligner (GC) and ADAP Aligner, are for aligning GC-MS data. Both of the two algorithms for aligning LC-MS data achieve alignment by finding chromatographic peaks that have similar m/z and retention time. Figure 13 shows how to perform alignment using the RANSAC aligner in MZmine 2. Aligned peaks can be examined and exported (Fig. 14). The exported peak list can be used for univariate and multivariate statistical analysis for determining the significant metabolites between phenotypes and training a predictive model for predicting phenotypes.



Fig. 12 Detection of chromatographic peaks using ADAP. **(a)** Detection of chromatographic peaks from EICs by ADAP can be accessed via Chromatogram deconvolution in the Peak list methods drop-down menu. **(b)** Specification of parameters is facilitated by the Show preview function

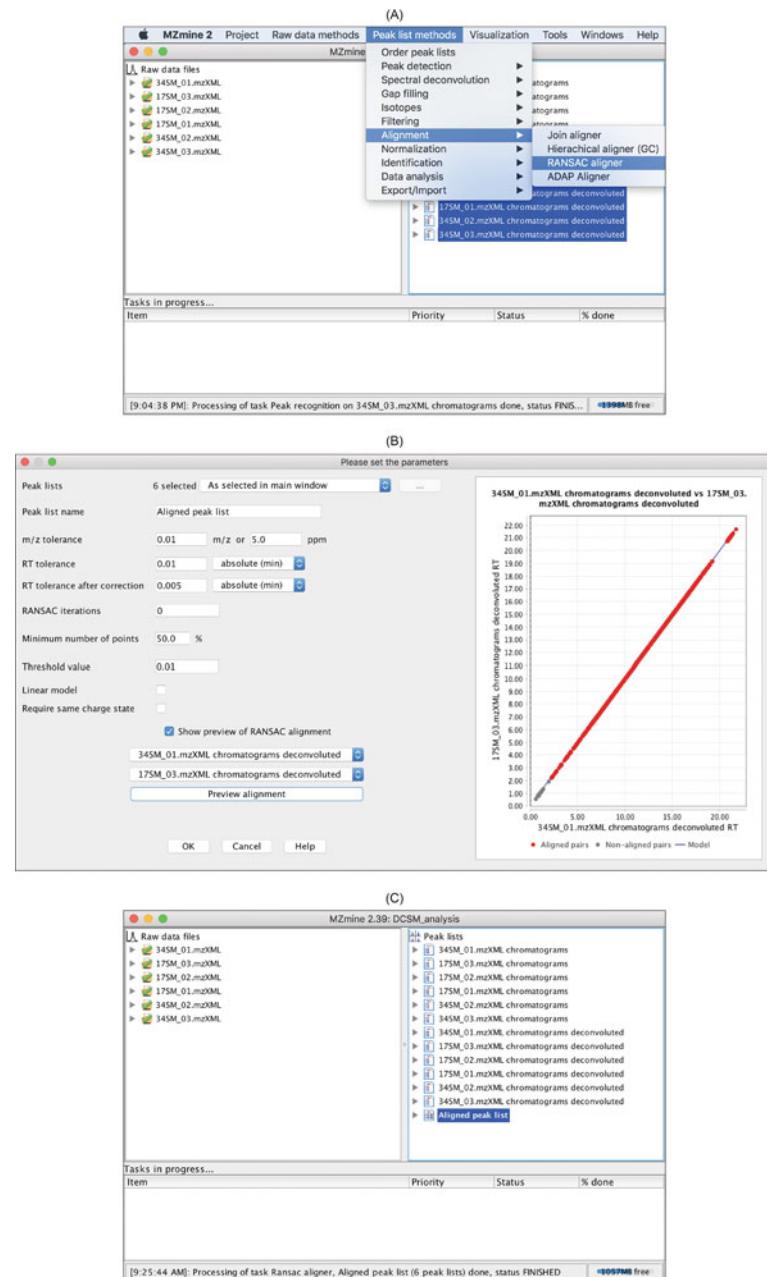


Fig. 13 Alignment. (a) Alignment methods can be accessed through the Peak list methods drop-down menu. (b) It is strongly recommended to use the preview function for specifying parameters. (c) After alignment, an Aligned peak list is produced and can be exported

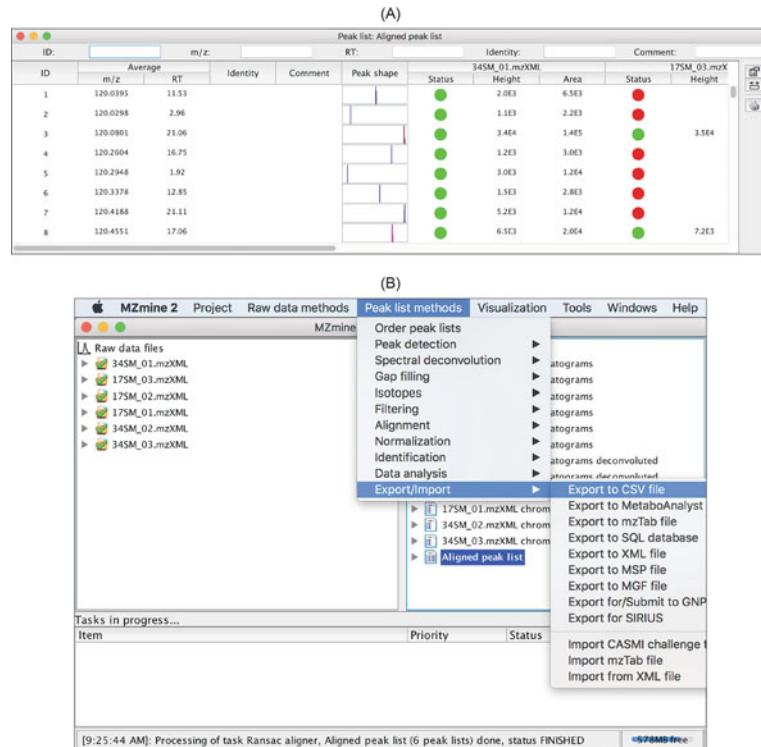


Fig. 14 Visualization and export of the aligned peak list. (a) Double clicking the Aligned peak list opens up the list of peaks for visual examination. (b) The aligned peak list can be exported in .csv, MetaboAnalyst, or other format

5 Preprocessing Workflow for GC–MS Data

As shown in Fig. 2, the preprocessing workflows for both LC–MS and GC–MS data contain the steps of mass detection, EIC construction, and detection of EIC peaks. The corresponding methods and the procedures that have been described above for LC–MS data preprocessing can be used for GC–MS data preprocessing as well. However, the GC–MS workflow contains a step called spectral deconvolution that is unique. This stems from the fact that the commonly used electron ionization used in GC–MS analysis fragments molecular ions into product ions in the ionization source. When compounds are not resolved chromatographically, product ions from different molecular ions co-exist in the same mass spectrum. In order to eventually identify/annotate the compounds that correspond to each molecular ion, spectral deconvolution needs to be performed to produce a pure mass spectrum of product ions and the molecular ion for the compound. Spectral deconvolution is especially necessary for low mass resolution GC–MS data that is still commonly acquired.

In addition to the unique spectral deconvolution in GC–MS preprocessing, the ADAP-GC preprocessing workflow features an alignment algorithm that is compound-based, rather than peak-based. Specifically, the ADAP-GC alignment algorithm looks for similar compounds across samples based on spectral similarity and proximity in retention time. This is very different from the RANSAC alignment algorithm and other peak-based algorithms that align chromatographic peaks only. If *n*-alkanes was added into the samples and therefore retention index of compounds can be calculated, alignment of compounds should take advantage of the retention index information, but ADAP-GC is currently not equipped with this capability yet.

5.1 Spectral Deconvolution

The most recent version of the ADAP-GC spectral deconvolution algorithm is 3.2 [5]. The algorithm starts with automated determination of deconvolution windows. For each deconvolution window, a sequence of four computational steps is carried out including: (1) two rounds of hierarchical clustering for estimating the number of compounds in the window, (2) selection of the sharpest and unique chromatographic peaks as the model peaks, (3) construction of pure mass spectrum for each compound, and (4) correction of splitting issues. Figure 15 shows how to access ADAP-GC 3.2 in MZmine 2 and lists the user-defined parameters. Similar to ADAP peak detection described earlier, it is strongly recommended that users use the Show preview function to make informed decisions about the parameters (Fig. 15b). After spectral deconvolution completes, a list of pure mass spectra is produced for each data file (Fig. 16).

5.2 Alignment

GC–MS samples are aligned by finding the same compounds across the data files based on spectral similarity and retention time proximity. Specifically, a score is calculated as follows to measure the likelihood that two spectra, c_1 and c_2 , correspond to the same compound:

$$\text{Score}(c_1, c_2) = wS_{time}(c_1, c_2) + (1 - w)S_{spec}(c_1, c_2), \quad (1)$$

where S_{time} is the retention time proximity between c_1 and c_2 and S_{spec} is the spectrum similarity between c_1 and c_2 . w is a weighting factor specifying the relative importance of S_{time} and S_{spec} . S_{spec} is calculated as the normalized dot product between c_1 and c_2 . Figure 17 shows how to use the alignment method. The following parameters need to be specified.

1. *Min confidence*: minimum fraction of samples where aligned components must be present. It takes values between 0.0 and 1.0.

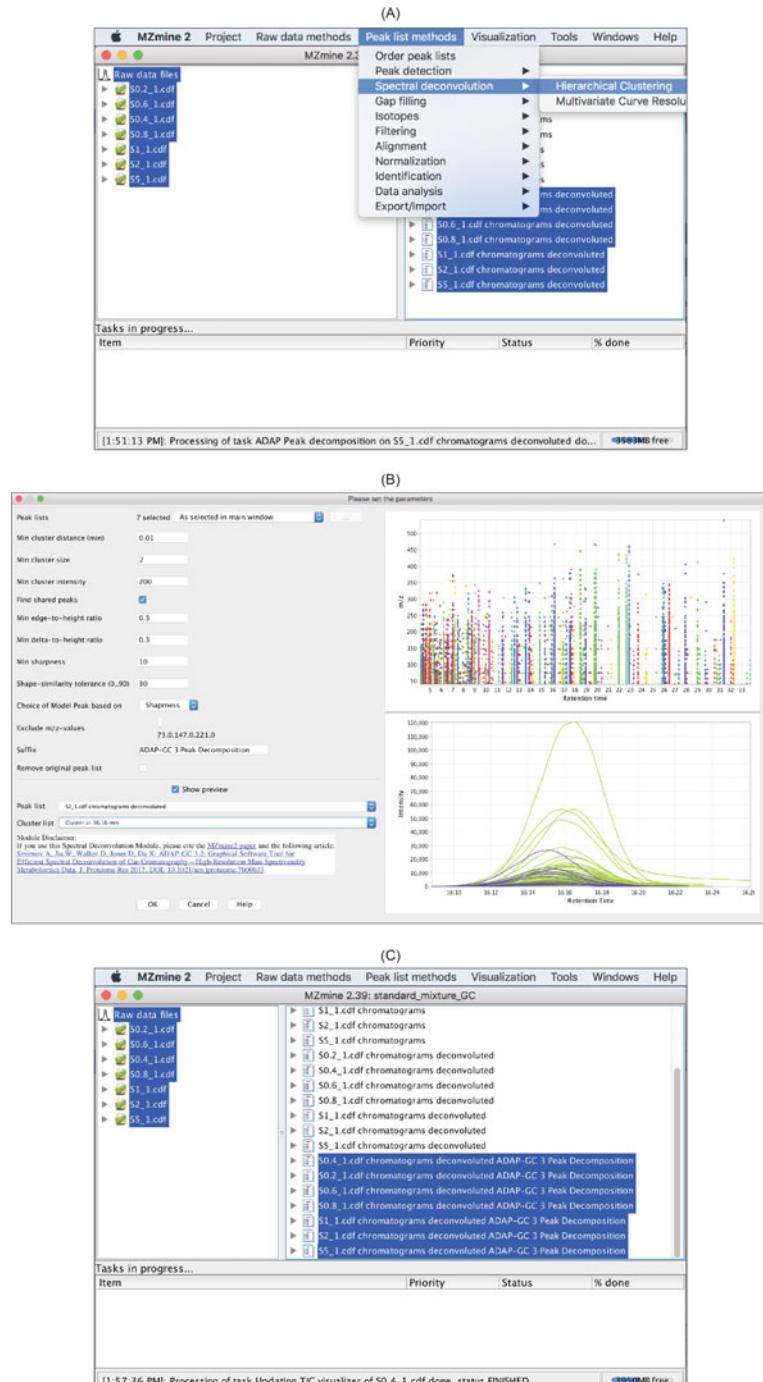


Fig. 15 Spectral deconvolution. (a) ADAP-GC 3.2 spectral deconvolution can be accessed via Peak list methods → Spectral deconvolution → Hierarchical Clustering. (b) User-defined parameters and the preview function that helps with specifying these parameters. (c) After spectral deconvolution, a list of pure mass spectra is produced for each data file

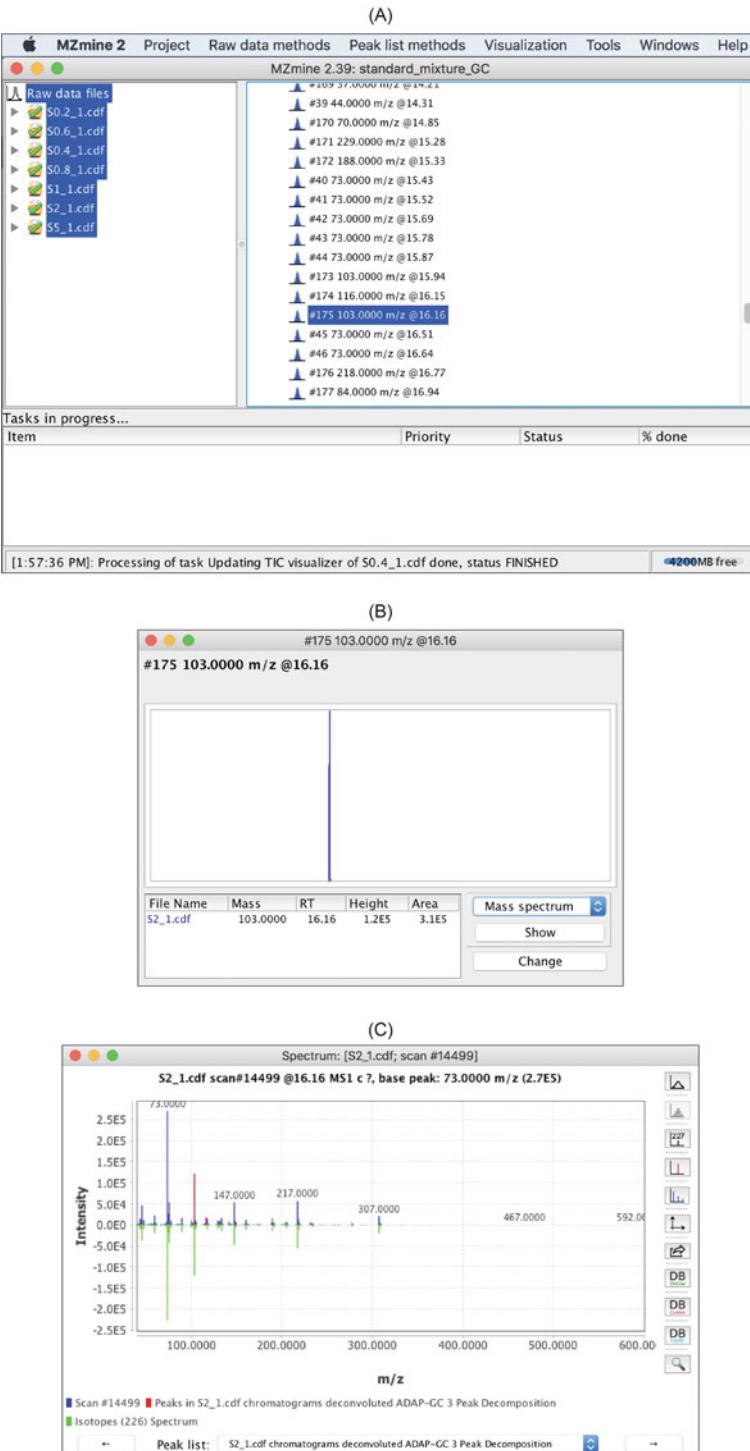


Fig. 16 Examine spectral deconvolution results. (a) Expand the list of mass spectra that has been produced for each data file. (b) Each mass spectrum can be examined by double clicking it to open up a window. Select the data file and Mass spectrum and click Show to open the window in (c). The constructed pure spectrum is shown in green in the context of the raw spectrum shown in blue

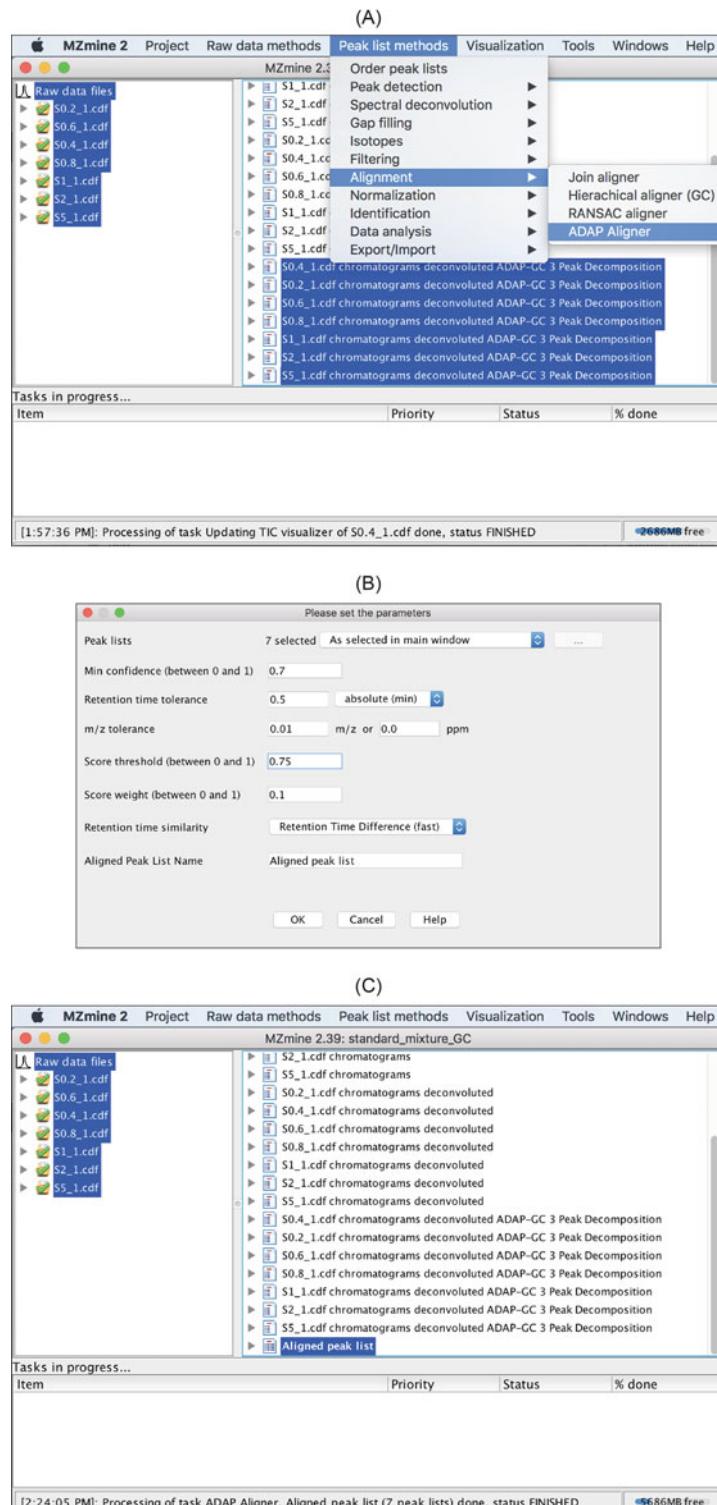


Fig. 17 GC alignment. (a) The Alignment method can be accessed in the drop-down menu of Peak list methods → Alignment → ADAP Aligner. (b) Specify parameters whose meaning can be found in the Help file. (c) The aligner produces the Aligned peak list

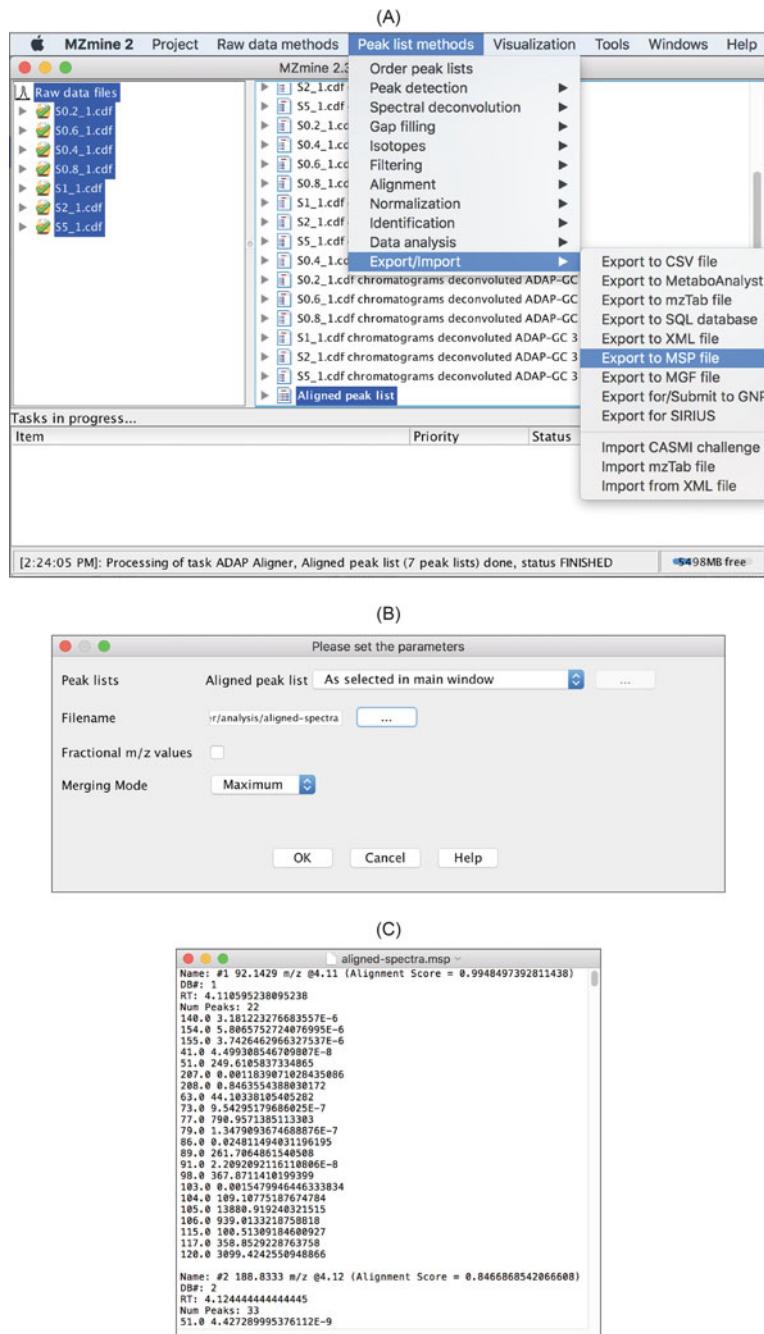


Fig. 18 Export GC-MS spectra. (a) Select the Aligned peak list and specify the export file name. (b) An example of the exported spectra produced by the spectral deconvolution. (c) Exported GC-MS spectra are stored in an .msp file

2. *Retention time tolerance*: maximum retention time difference between aligned compounds in different samples.
3. *m/z tolerance*: maximum *m/z* difference to consider two *m/z* values in two spectra as the same. This is used for determining the quantitation mass for a particular compound. This mass is defined as the most frequent mass across all of the spectra for this compound.
4. *Score threshold*: minimum score as calculated in Eq. 1 to consider c_1 and c_2 to correspond to the same compound. It takes values between 0.0 and 1.0. The default value is 0.75.
5. *Score weight*: w in Eq. 1 and takes values between 0.0 and 1.0. The default value is 0.1.
6. *Retention time similarity*: S_{time} in Eq. 1 as the difference in retention time.

5.3 Export of GC–MS Preprocessing Results

The pure mass spectra that the spectral deconvolution step has constructed can be exported in .msp or .mgf format for matching the spectra against spectral libraries for compound identification or annotation. Figure 18 shows the procedure. The resulting .msp file can be directly imported to the *NIST MS Search* software tool for compound identification or annotation.

6 Conclusions

ADAP is a suite of computational algorithms and the associated graphical user interface for preprocessing untargeted LC–MS and GC–MS metabolomics data. Incorporation of these algorithms into the prevalent MZmine 2 takes advantage of the rich visualization capabilities in MZmine and benefits users of MZmine 2.

Acknowledgements

We thank the USA National Science Foundation award 1262416 and National Institutes of Health/National Cancer Institute grant U01CA235507 for funding the research and development of ADAP.

References

1. Pluskal T, Castillo S, Villar-Briones A, Oresic M (2010) MZmine 2: modular framework for processing, visualizing, and analyzing mass spectrometry-based molecular profile data. *BMC Bioinf* 11:395
2. Jiang W, Qiu Y, Ni Y, Su M, Jia W, Du X (2010) An automated data analysis pipeline for GC-TOF-MS metabonomics studies. *J Proteome Res* 9(11):5974–5981
3. Ni Y, Qiu Y, Jiang W, Suttlemyre K, Su M, Zhang W, Jia W, Du X (2012) ADAP-GC 2.0: deconvolution of coeluting metabolites from GC/TOF-MS data for metabolomics studies. *Anal Chem* 84(15):6619–6629

4. Ni Y, Su M, Qiu Y, Jia W, Du X (2016) ADAP-GC 3.0: improved peak detection and deconvolution of co-eluting metabolites from GC/TOF-MS data for metabolomics studies. *Anal Chem* 88(17):8802–8811
5. Smirnov A, Jia W, Walker DI, Jones DP, Du X (2018) ADAP-GC 3.2: graphical software tool for efficient spectral deconvolution of gas chromatography-high-resolution mass spectrometry metabolomics data. *J Proteome Res* 17(1):470–478
6. Coble JB, Fraga CG (2014) Comparative evaluation of preprocessing freeware on chromatography/mass spectrometry data for signature discovery. *J Chromatogr A* 1358:155–164
7. Rafiei A, Sleno L (2015) Comparison of peak-picking workflows for untargeted liquid chromatography/high-resolution mass spectrometry metabolomics data analysis. *Rapid Commun Mass Spectrom* 29(1):119–127
8. Chambers MC, Maclean B, Burke R, Amodei D, Ruderman DL, Neumann S, Gatto L, Fischer B, Pratt B, Egertson J, Hoff K, Kessner D, Tasman N, Shulman N, Frewen B, Baker TA, Brusniak MY, Paulse C, Creasy D, Flashner L, Kani K, Moulding C, Seymour SL, Nuwaysir LM, Lefebvre B, Kuhlmann F, Roark J, Rainer P, Detlev S, Hemenway T, Huhmer A, Langridge J, Connolly B, Chadick T, Holly K, Eckels J, Deutsch EW, Moritz RL, Katz JE, Agus DB, MacCoss M, Tabb DL, Mallick P (2012) A cross-platform toolkit for mass spectrometry and proteomics. *Nat Biotechnol* 30(10):918–920
9. Myers OD, Sumner SJ, Li S, Barnes S, Du X (2017) One step forward for reducing false positive and false negative compound identifications from mass spectrometry metabolomics data: new algorithms for constructing extracted ion chromatograms and detecting chromatographic peaks. *Anal Chem* 89(17):8696–8703



Chapter 4

Metabolomics Data Processing Using OpenMS

Marc Rurik, Oliver Alka, Fabian Aicheler, and Oliver Kohlbacher

Abstract

This chapter describes the open-source tool suite OpenMS. OpenMS contains more than 180 tools which can be combined to build complex and flexible data-processing workflows. The broad range of functionality and the interoperability of these tools enable complex, complete, and reproducible data analysis workflows in computational proteomics and metabolomics. We introduce the key concepts of OpenMS and illustrate its capabilities with a complete workflow for the analysis of untargeted metabolomics data, including metabolite quantification and identification.

Key words OpenMS, Data analysis, Workflows, Reproducible science, Metabolomics

1 Introduction

1.1 OpenMS

Mass spectrometry workflows can be highly complex, due to many different separation techniques, acquisition strategies, quantification, and identification methods. As a solution, we present OpenMS, an open-source C++ library for LC-MS data management that offers a wide range of algorithms for proteomics and metabolomics data analysis. It is available under the permissive 3-clause BSD license for Windows, macOS, and Linux (www.openms.de) [1, 2]. OpenMS is composed of a set of individual tools that can be combined in a highly flexible manner allowing its application from standard operations to newly developed methods. The software is interesting for both users and developers since the library can be used as a foundation for tool and algorithm development (*see Note 1*). Fast scripting and prototyping is supported via Python bindings (pyOpenMS) [3]. Command line tools are available for high-throughput processing, which is especially useful on cluster or cloud infrastructures. Additionally, OpenMS is integrated into the workflow engine Konstanz Information Miner (KNIME), allowing for the visual construction of data analysis workflows [4, 5] (*see Note 2*). In the following, we will present the application of OpenMS and KNIME for the quantification and identification of metabolomics data.

1.2 Data Preparation

A multitude of vendor-specific, proprietary file formats exist for LC-MS data. In order to process this data with OpenMS it needs to be converted to the open HUPO-PSI standard format mzML [6]. Data conversion from the vendor-specific formats to mzML can be performed using the free software MSConvertGUI included in ProteoWizard [7]. Additionally, if the data was acquired in profile mode, centroiding is necessary as almost all OpenMS algorithms require data to be in centroid mode. This step can be performed during the data conversion by using the peakPicking filter provided by ProteoWizard. Alternatively, the OpenMS tool PeakPickerHiRes can be used as the first step of the analysis pipeline.

1.3 Data Visualization Using TOPPView

Data visualization is a common starting point for many studies. It allows additional insight into the data, can act as a quality control step and help with the optimization of the tools' parameter values. In addition, visualization can be used to inspect intermediate or final results throughout the data analysis process. To this end, OpenMS provides TOPPView [8], a graphical application for the interactive visualization of raw data and analysis results. It supports the visualization of individual spectra and entire LC-MS maps in 2D and 3D representations. Results of different OpenMS tools such as the location and extent of detected features (see Subheading 2.1) can be inspected in conjunction with the raw data. An example for this is shown in Fig. 1.

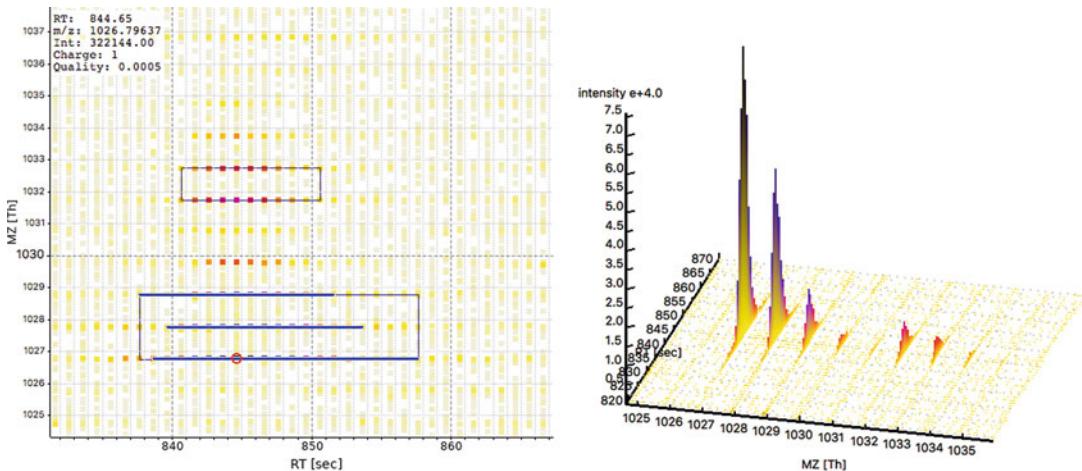


Fig. 1 LC-MS sample visualized in TOPPView in 2D and 3D with color-coded intensities. Two features that were detected with FeatureFinderMetabo (see Subheading 2.1) are shown as rectangles representing their location and extent

During the development of a new analysis pipeline it is often useful to assess and tune various parameters to best fit the experimental and instrument setup. To assist with this, many OpenMS tools can be executed directly from within TOPPView allowing for the immediate visualization of their results with different parameter choices.

1.4 Workflow Construction Using KNIME

KNIME (<https://www.knime.com>) is an open-source graphical workflow engine that allows for the user-friendly construction of complex data analysis workflows. A workflow is a series of computational steps that describe exactly how the input data is processed. Due to its comprehensive set of data analysis tools and its openness to extensions, KNIME is well-suited to construct complete computational workflows. These workflows describe all tools that make up the data analysis and, most importantly, also all parameters required. Such workflows are an essential step towards reproducible analysis of metabolomics data. Finalized workflows can easily be shared with others, e.g., in the supplemental material of a publication, allowing them to reproduce the results.

In KNIME, nodes represent the individual data-processing steps of the workflow. A node uses ports to read input data and write output data. The data is transported along the edges that connect the input and output ports of a series of nodes. An example is shown in Fig. 2.

Another advantage of KNIME is that its community of users provide a large collection of plugins which can be integrated into the workflow. This makes it possible to add all commonly performed downstream analysis steps including statistical hypothesis testing, data visualization, machine learning, and cheminformatics methods to the same workflow. In addition, scripting nodes for R, Python and other languages have proven to be useful. Example workflows using OpenMS in conjunction with these plugins can be downloaded from <https://www.openms.de/workflows> (see Note 3).

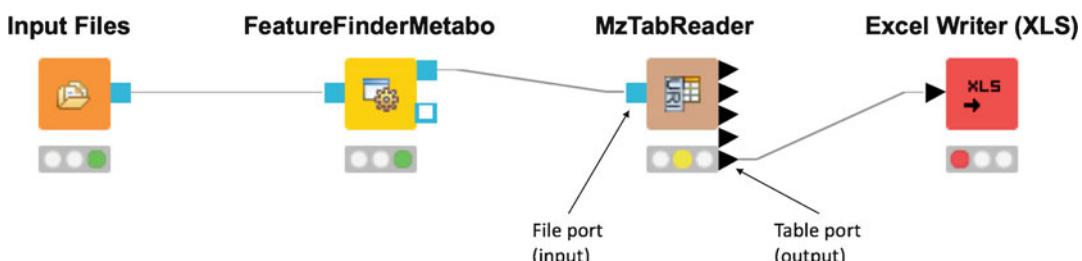


Fig. 2 A short KNIME workflow that detects and quantifies metabolites in a single LC-MS sample and exports the results to an Excel file. Note that some nodes process entire files (square ports) while other nodes process Excel-like tables (triangle ports)

2 Untargeted Quantification of Metabolites

In this section, we will demonstrate how to construct an automated workflow for the analysis of untargeted LC-MS metabolomics data. The first step is the quantification of putative metabolites in each input file. Afterwards, metabolites that are observed with different adducts are grouped together. Finally, we correct for potential retention time shifts and link the information across all input files. This results in a list of, at this point, unidentified metabolites that are characterized by their m/z , charge, retention time, and intensity across all input files. In Subheading 3, different tools for compound identification are added to the workflow.

2.1 Feature Detection

FeatureFinderMetabo [9] is a robust algorithm to detect and quantify metabolite features in an LC-MS sample. The term feature refers here to all signals caused by an analyte in a specific charge state and adduct situation. The signals corresponding to a feature do contain the full chromatographic profile in the retention time dimension and all isotopic traces in the m/z dimension. However, due to different adduct species, multiple features can belong to the same metabolite (this is addressed in Subheading 2.2).

Feature detection can be divided into four steps: (a) mass trace detection, (b) peak separation, (c) hypothesis generation, and (d) feature assembly.

Mass trace detection starts at the peaks with the highest intensity which serve as potential seeds. It extends them in both directions along the retention time dimension. The mass error is expected to be heteroscedastic, which means that it will be larger at lower intensities. To account for this, the estimations that decide whether additional peaks are added to a mass trace are iteratively refined to make sure that the low intensity ends are properly detected (Fig. 3a).

Metabolites that have a sufficiently similar m/z and elute in close proximity can be located on a single uninterrupted mass trace. FeatureFinderMetabo detects these cases based on the elution profile and splits the feature at the local minimum between both elution profiles (Fig. 3b).

Next, mass traces that likely originate from the same metabolite are assembled into a feature. To decide whether an additional mass trace should be added to a feature, it has to satisfy three criteria: (1) the mass trace co-elutes within a user-defined RT range, (2) the mass trace has the correct m/z distance within a user-defined mass accuracy and range of possible charge states, and (3) the observed isotope pattern is likely to be caused by a metabolite, i.e., only certain intensity ratios between the isotopic traces are plausible. In proteomics, the averagine model is widely used to identify plausible isotopic traces, but this model is not applicable here since metabolites differ widely in their elemental composition. To

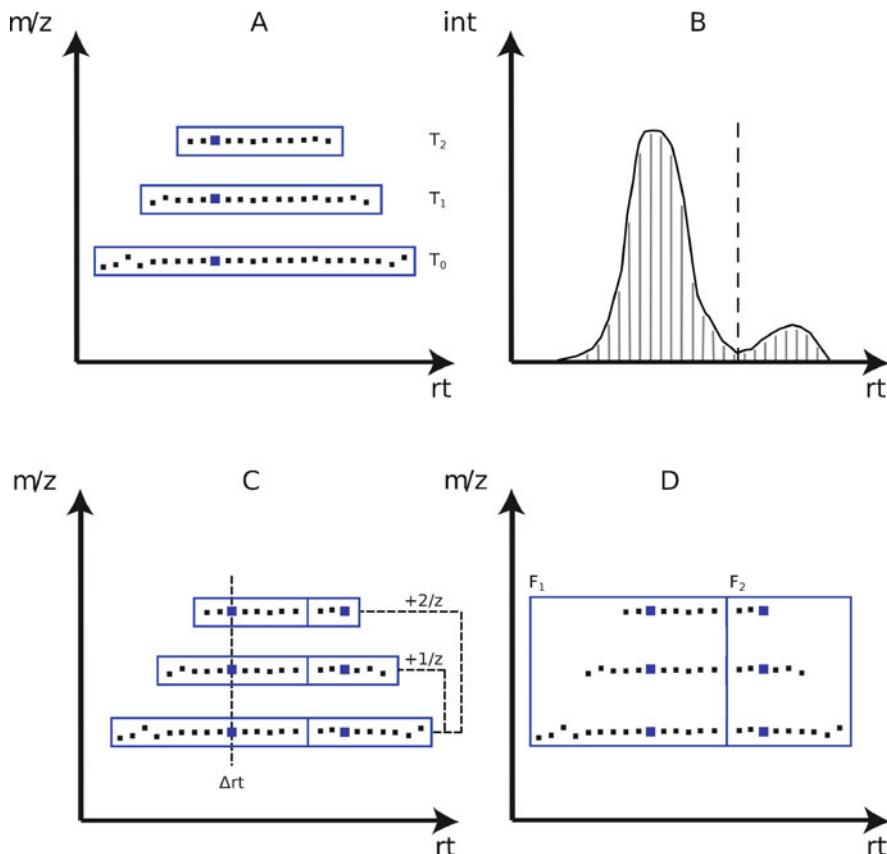


Fig. 3 FeatureFinderMetabo feature detection process: **(a)** Mass trace detection starts at the peaks of highest intensity (marked by a blue square) and extends in both directions. **(b)** Split the mass trace at the local minimum if it contains more than one compound. **(c)** Test whether co-eluting mass traces are isotopic traces of the same compound. **(d)** Assemble highest scoring feature hypotheses. The final features are characterized by centroid m/z , retention time, charge, and intensity. A visualization of a feature in TOPPView is shown in Fig. 1

address this, distinct sum formulas between 1 and 1000 Da were used to train a support vector machine (SVM) to distinguish between likely and unlikely isotope abundance ratios (Fig. 3c).

All possible feature hypotheses are scored based on their agreement with the model and the best-scoring hypotheses are assembled into features. The resulting features are characterized by their centroid retention time, m/z , charge, and intensity, where the feature intensity corresponds to the area under the monoisotopic trace (Fig. 3d).

2.2 Adduct Grouping

The previous feature finding step aims to detect and group all isotopic traces of a compound ion into one feature represented by centroid m/z , retention time, and intensity. However, a compound might be detected multiple times in the same sample, with different adduct species or common charge neutral modifications.

Successfully grouping different ion species of a compound allows to leverage knowledge across ions and infer information missing from some of the features. Besides determining the neutral mass, ion adducts and charges can be annotated to individual features, facilitating computations in subsequent sample alignment and linking steps.

For metabolomics, the MetaboliteAdductDecharger provides this functionality. Inspired by similar algorithms used in peptide decharging [10], this tool aggregates features inside an elution window mainly by mass shifts and feature charges that are consistent with assumed adduct hypotheses.

Given user-defined adduct probabilities and charge ranges, a combinatorial table of pairwise adduct complexes and resulting mass shifts is considered to create connected graphs of co-eluting features. For each such connected group, the feature adducts corresponding to the observed mass shifts are then assigned via an integer linear programming (ILP) approach. That is, we seek to resolve all conflicts arising from multiple explanations for observed mass shifts. The optimal solution is chosen by maximizing overall ion probabilities.

2.3 Map Alignment

After feature detection, an optional but oftentimes appropriate processing step is to perform chromatographic alignment of the samples with each other. Here, the goal is to compensate for commonly occurring constant or linear chromatography-related distortions between samples.

This can be done with the MapAlignerPoseClustering tool, which reduces feature elution time differences between sample maps. To achieve this, the method aims to find affine retention time transformations between samples that minimize the overall time shifts of corresponding features. As a consequence, subsequent aggregation of respective compound ion features found in multiple measurements is more straightforward. To search the solution space of possible time shifts and scaling distortions (i.e., compression or extension of a samples chromatography) efficiently, pose clustering is performed [11]. The method aligns the features of multiple maps by using one as the reference to which the other maps are aligned, akin to matching star charts in astronomy. It determines the optimal linear transformation for each pair of maps and transforms the feature retention times accordingly (see Fig. 4a, b).

2.4 Feature Linking

The FeatureLinkerUnlabeledQT algorithm is one of the OpenMS methods to group features belonging to the same compound ion across multiple samples. Using a QT clustering approach, features are matched based on m/z and retention time [12]. By considering all possible pairs inside user-defined m/z and retention difference

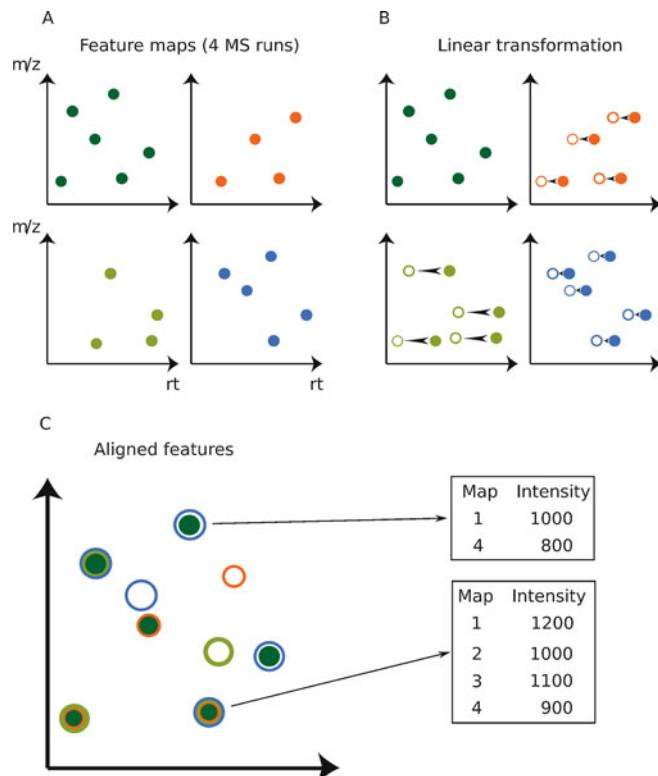


Fig. 4 Map alignment and feature linking are used to combine information across multiple samples. **(a)** Features have been quantified in all samples individually using FeatureFinderMetabo. **(b)** MapAlignerPoseClustering applies a linear transformation to address potential retention time shifts. **(c)** The retention time-aligned features can then be grouped using FeatureLinkerUnlabeledQT (see Subheading 2.4)

tolerances, the tool can resolve variations in the local chromatography not addressed by the previous affine retention time transformations, like differing elution orders.

In case a previous alignment step was performed, retention time tolerances of the search window used for linking can be tightened. As a consequence, the number of pairs of features across samples having to be considered for correspondence is reduced. Besides computational advantages, this also reduces the likelihood of spurious matches.

Resulting linked features are collected as so-called consensus features. Each consensus feature is represented by a centroid which integrates the m/z , retention time, and charge of its member features. A consensus feature further allows the comparison of ion intensities across maps (see Fig. 5).

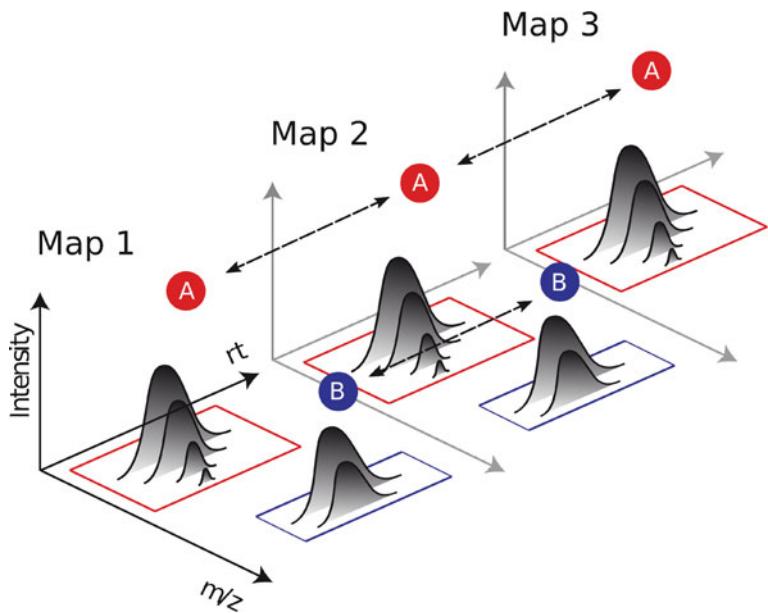


Fig. 5 FeatureLinkerUnlabeledQT takes several feature maps and stores the corresponding features in a consensus map. Here, feature A and feature B are linked across three different maps

3 Basic Metabolite Identification

The previous section discussed how to detect and quantify metabolite features characterized by retention time, *m/z*, charge, and intensity. Identifying the underlying compounds remains one of the most challenging problems in metabolomics. This section introduces three computational approaches implemented in OpenMS to arrive at putative metabolite identifications.

3.1 Accurate Mass Search

The foundation of the AccurateMassSearch (AMS) tool is a database of known metabolites and their exact neutral masses. OpenMS provides access to the Human Metabolome Database (HMDB) [13], but generally any compound database can be used. In addition to such a database, a list of potential adducts has to be considered to arrive at the neutral masses of the compounds. OpenMS provides two adduct lists for positive and negative polarity, which can be edited depending on the experimental setup, e.g., which buffers were used. Ideally this step should use the same set of adducts as during adduct grouping.

3.2 Spectral Library Search

While accurate mass search can be used to obtain a general idea which compounds may be present in the sample, it will often provide multiple putative identifications. To arrive at more confident identifications, it is necessary to fragment the compounds and

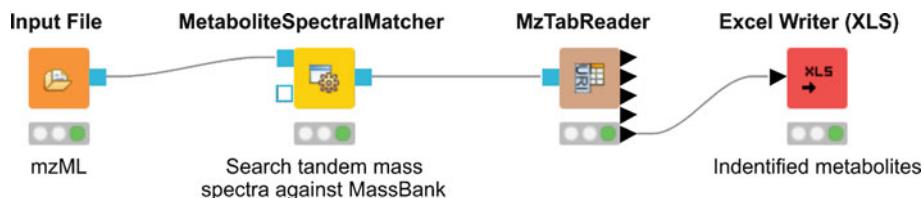


Fig. 6 KNIME workflow to identify metabolites by searching all tandem mass spectra against a spectral library, in this case MassBank

compare them with reference spectra. Spectral libraries are a fast and reliable way to refine compound identifications, but are limited by the number of available compounds in the database.

OpenMS provides the tool MetaboliteSpectralMatcher to search all tandem mass spectra against a spectral library (*see* Fig. 6). The candidate spectra are scored against the database spectra using a modified version of the Hyperscore described by Fenyö and Beavis [14]

$$\text{HS}(S_{\text{query}}, S_{\text{database}}) = \log \left(\sum_{i=0}^n I_i \cdot I'_i \right) + \log n! \quad (1)$$

where I_i and I'_i are the intensities of all n matched peaks. MetaboliteSpectralMatcher can use any public or in-house spectral library in the mzML format. By default, OpenMS provides MassBank, the most comprehensive publicly available spectral library [15].

3.3 De Novo Identification

For de novo identification, SIRIUS [16, 17] and CSI:FingerID [18] are integrated in the OpenMS tool SiriusAdapter (*see* Fig. 7). A preprocessing step is performed using the OpenMS framework to optimize and convert the input data to a SIRIUS compatible format. SiriusAdapter can then perform identification of features at MS2 level, i.e., molecular formulas are assigned de novo. Subsequently, the feature can be searched in a molecular structure database. SiriusAdapter uses information from both MS and MS/MS spectra (mzML) and an optional featureXML. The tool provides different modes, depending on the input and output configuration.

Preprocessing can be applied if additional feature information is present, which is used to map all MS2 spectra to their corresponding features. SIRIUS will internally merge and jointly process all MS2 spectra allocated to the same feature. To reduce the feature space further, a mass trace filter (number of isotopic traces) can be applied. Additionally, adduct information can be provided using a featureXML processed by the MetaboliteAdductDecharger or AccurateMassSearch. Depending on the workflow, SiriusAdapter can be used either for preprocessing only (output port SIRIUS.ms) or full data processing (output port SIRIUS.mzTab), *see* Fig. 7.

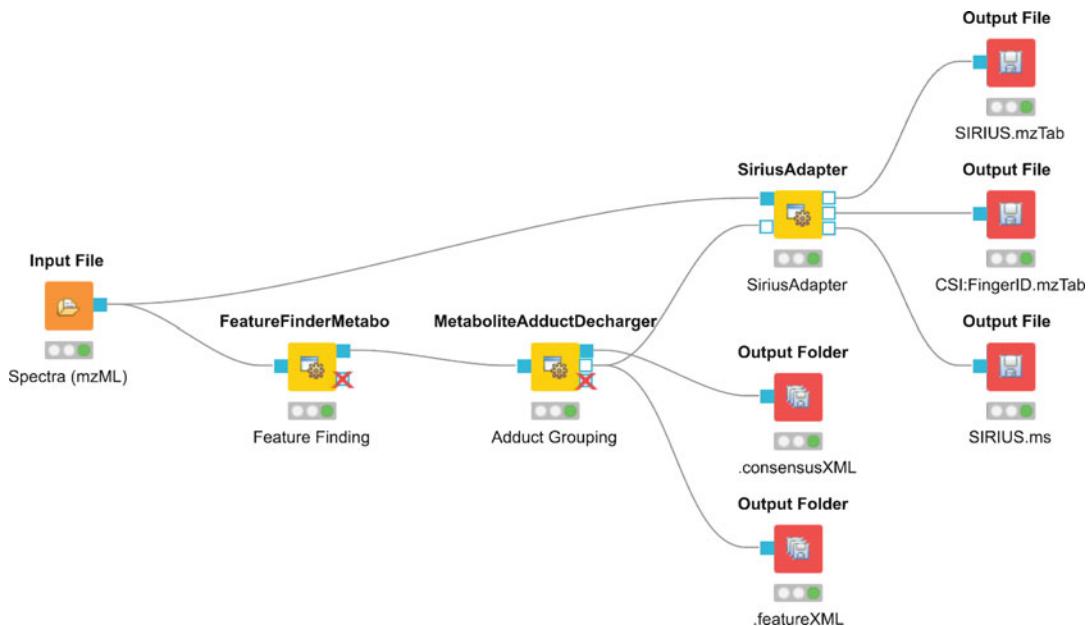


Fig. 7 Workflow for metabolite quantification using FeatureFinderMetabo and de novo identification using SiriusAdapter

4 Quantification and Identification Workflow

An example OpenMS workflow for the analysis of an untargeted metabolomics study consisting of any number of samples is shown in Fig. 8. Note that metabolite detection and adduct grouping are performed separately for each file. For this reason, these nodes are enclosed by the nodes ZipLoopStart and ZipLoopEnd, which will automatically iterate over all input files, process them and collect the results. The alignment and aggregation of all detected features across all samples are then performed by MapAlignerPoseClustering and FeatureLinkerUnlabeledQT as described in Subheadings 2.3 and 2.4. In this workflow, metabolite identification is performed by querying HMDB using AccurateMassSearch. Putative identifications can be refined with spectral library search or de novo identifications (*see* Subheadings 3.2 and 3.3). Finally, MzTabReader converts the data to a table-based file format that can be read using Excel and similar software.

5 Notes

OpenMS is in constant development to improve its algorithm library, tools, and usability. For further information about metabolomics data analysis with OpenMS and KNIME, we recommend the following sources:

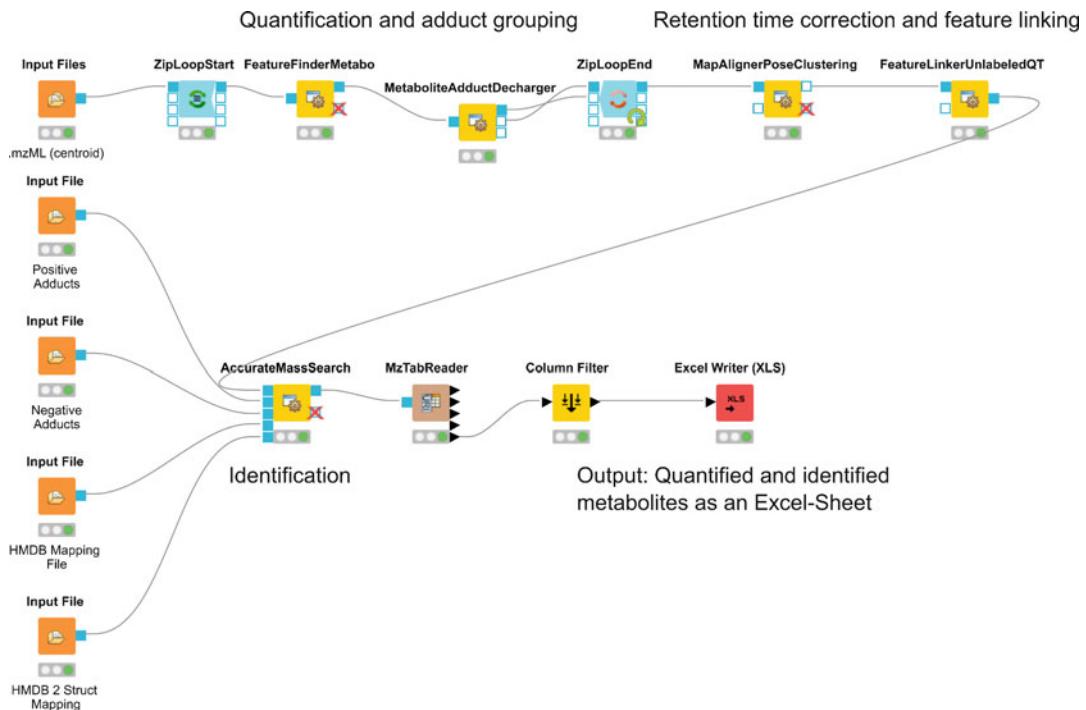


Fig. 8 Basic OpenMS workflow for metabolite quantification and identification in KNIME

1. The full OpenMS source code is available in our GitHub repository <https://github.com/OpenMS/OpenMS>. We are always open to contributions, feel free to file issues on GitHub or use the chat system Gitter at <https://gitter.im/OpenMS/OpenMS> to get in contact with the developers.
2. Installers for OpenMS and the workflow engine KNIME are available at
 - OpenMS: <https://www.openms.de/download/openms-binaries>
 - KNIME: <https://www.knime.org/downloads/overview>
 We provide detailed instructions on how to install the OpenMS plugin in KNIME at <https://www.openms.de/getting-started/creating-workflows>.
3. Visit <https://www.openms.de> for general information and news about OpenMS. A detailed step-by-step guide can be obtained from <https://www.openms.de/tutorials>. In addition, we provide ready-to-use example workflows for common tasks at <https://www.openms.de/workflows>. Please note the importance to adapt the parameters of the algorithms to your experimental and instrument setup. You can use the TOPPView to assess the impact of different parameter choices.

References

1. Röst HL, Sachsenberg T, Aiche S, Bielow C, Weisser H, Aicheler F, Andreotti S, Ehrlich H-c, Gutenbrunner P, Kenar E, Liang X, Nahnse S, Nilse L, Pfeuffer J, Rosenberger G, Rurik M, Schmitt U, Veit J, Walzer M, Wojnar D, Wolski WE, Schilling O, Choudhary JS, Malmström L, Aebersold R, Reinert K, Kohlbacher O (2016) OpenMS: a flexible open-source software platform for mass spectrometry data analysis. *Nat Methods* 13:741–748
2. Pfeuffer J, Sachsenberg T, Alka O, Walzer M, Fillbrunn A, Nilse L, Schilling O, Reinert K, Kohlbacher O (2017) OpenMS - a platform for reproducible analysis of mass spectrometry data. *J Biotechnol* 261(February):142–148
3. Röst HL, Schmitt U, Aebersold R, Malmström L (2014) pyOpenMS: a Python-based interface to the OpenMS mass-spectrometry algorithm library. *Proteomics* 14(1):74–77
4. Berthold MR, Cebron N, Dill F, Gabriel TR, Kötter T, Meini T, Ohl P, Sieb C, Thiel K, Wiswedel B (2007) KNIME: the Konstanz Information Miner. In: Studies in classification, data analysis, and knowledge organization (GfKL 2007). Springer, Berlin
5. Fillbrunn A, Dietz C, Pfeuffer J, Rahn R, Landrum GA, Berthold MR (2017) KNIME for reproducible cross-domain analysis of life science data. *J Biotechnol* 261 (February):149–156
6. Martens L, Chambers M, Sturm M, Kessner D, Levander F, Shofstahl J, Tang WH, Römpf A, Neumann S, Pizarro AD *et al* (2011) mzML—a community standard for mass spectrometry data. *Mol Cell Proteomics* 10(1): R110.000133
7. Kessner D, Chambers M, Burke R, Agus D, Mallick P (2008) ProteoWizard: open source software for rapid proteomics tools development. *Bioinformatics* 24(21):2534–2536
8. Sturm M, Kohlbacher O (2009) TOPPView: an open-source viewer for mass spectrometry data. *J Proteome Res* 8(7):3760–3763
9. Kenar E, Franken H, Forcisi S, Wörmann K, Häring H-U, Lehmann R, Schmitt-Kopplin P, Zell A, Kohlbacher O (2014) Automated label-free quantification of metabolites from liquid chromatography–mass spectrometry data. *Mol Cell Proteomics* 13:348–359
10. Bielow C, Ruzeck S, Huber CG, Reinert K (2010) Optimal decharging and clustering of charge ladders generated in ESI-MS. *J Proteome Res* 9(5):2688–2695
11. Lange E, Gropl C, Schulz-Trieglaff O, Leinenbach A, Huber C, Reinert K (2007) A geometric approach for the alignment of liquid chromatography mass spectrometry data. *Bioinformatics* 23(13):i273–i281
12. Weisser H, Nahnse S, Grossmann J, Nilse L, Quandt A, Brauer H, Sturm M, Kenar E, Kohlbacher O, Aebersold R, Malmström L (2013) An automated pipeline for high-throughput label-free quantitative proteomics. *J Proteome Res* 12:1628–1644
13. Wishart DS, Feunang YD, Marcu A, Guo AC, Liang K, Vázquez-Fresno R, Sajed T, Johnson D, Li C, Karu N, Sayeeda Z, Lo E, Assempon N, Berjanskii M, Singhal S, Arndt D, Liang Y, Badran H, Grant J, Serra-Cayuela A, Liu Y, Mandal R, Neveu V, Pon A, Knox C, Wilson M, Manach C, Scalbert A (2018) HMDB 4.0: the human metabolome database for 2018. *Nucleic Acids Res* 46(D1): D608–D617
14. Fenyo D, Beavis RC (2003) A method for assessing the statistical significance of mass spectrometry-based protein identifications using general scoring schemes. *Anal Chem* 75 (4):768–774
15. Horai H, Arita M, Kanaya S, Nihei Y, Ikeda T, Suwa K, Ojima Y, Tanaka K, Tanaka S, Aoshima K, Oda Y, Kakazu Y, Kusano M, Tohge T, Matsuda F, Sawada Y, Hirai MY, Nakanishi H, Ikeda K, Akimoto N, Maoka T, Takahashi H, Ara T, Sakurai N, Suzuki H, Shibata D, Neumann S, Iida T, Tanaka K, Funatsu K, Matsuura F, Soga T, Taguchi R, Saito K, Nishioka T (2010) MassBank: a public repository for sharing mass spectral data for life sciences. *J Mass Spectrom* 45(7):703–714
16. Böcker S, Letzel MC, Lipták Z, Pervukhin A (2009) SIRIUS: decomposing isotope patterns for metabolite identification. *Bioinformatics* 25 (2):218–224
17. Böcker S, Dührkop K (2016) Fragmentation trees reloaded. *J Cheminform* 8(1):1–26
18. Dührkop K, Shen H, Meusel M, Rousu J, Böcker S (2015) Searching molecular structure databases with tandem mass spectra using CSI: FingerID. *Proc Natl Acad Sci* 112:12580–12585



Chapter 5

Analysis of NMR Metabolomics Data

Wimal Pathmasiri, Kristine Kay, Susan McRitchie, and Susan Sumner

Abstract

In this chapter, we summarize data preprocessing and data analysis strategies used for analysis of NMR data for metabolomics studies. Metabolomics consists of the analysis of the low molecular weight compounds in cells, tissues, or biological fluids, and has been used to reveal biomarkers for early disease detection and diagnosis, to monitor interventions, and to provide information on pathway perturbations to inform mechanisms and identifying targets. Metabolic profiling (also termed metabotyping) involves the analysis of hundreds to thousands of molecules using mainly state-of-the-art mass spectrometry (MS) and nuclear magnetic resonance (NMR) spectroscopy technologies. While NMR is less sensitive than mass spectrometry, NMR does provide a wealth of complex and information rich metabolite data. NMR data together with the use of conventional statistics, modeling methods, and bioinformatics tools reveals biomarker and mechanistic information. A typical NMR spectrum, with up to 64k data points, of a complex biological fluid or an extract of cells and tissues consists of thousands of sharp signals that are mainly derived from small molecules. In addition, a number of advanced NMR spectroscopic methods are available for extracting information on high molecular weight compounds such as lipids or lipoproteins. There are numerous data preprocessing, data reduction, and analysis methods developed and evolving in the field of NMR metabolomics. Our goal is to provide an extensive summary of NMR data preprocessing and analysis strategies by providing examples and open source and commercially available analysis software and bioinformatics tools.

Key words NMR, Metabolomics, Metabotyping, Quality control, Preprocessing, Multivariate data analysis

1 Introduction

1.1 Definitions

Nuclear magnetic resonance (NMR) spectroscopy is a quantitative, robust, highly reproducible, and nondestructive analytical technique, and it is widely used in metabolomics analyses. NMR spectroscopy relies on the magnetic properties of the nucleus of an atom [1]. When placed in a strong magnetic field, these nuclei resonate at characteristic frequencies in the radio frequency range (e.g., 500 MHz for ^1H at 11.7 T magnetic field, Table 1). NMR spectroscopy has long been used as an analytical tool in studying the chemical structure, conformation, and molecular dynamics of both small molecules and macromolecules and more details about NMR

Table 1

NMR frequencies and natural abundance of some useful nuclei in an 11.7 T magnetic field strength (values adopted from Bruker NMR frequency table)

Nucleus	Basic frequency (MHz)	Natural abundance (%)
¹ H	500	100
¹³ C	125.7	1.07
¹⁵ N	50	0.36
³¹ P	202.5	100

spectroscopy are described in detail in textbooks [1] and literature. For easy understanding, some important features in the NMR spectroscopy and NMR metabolomics analysis are defined below.

NMR chemical shift and J-coupling. Variations in the NMR frequency of an atom occur from neighboring atoms or attached functional groups within the molecular structure (chemical microenvironment), and these provide detailed information about the structure and conformation of the molecule. The variation in resonance is called the chemical shift and is typically reported in parts per million (ppm) to enable comparison of results between spectrometers of different frequency. Parts per million is the frequency of the resonance in Hertz divided by the spectrometry frequency. Splitting patterns that arise in the NMR spectrum (also referred to as *J*-coupling) are used together with chemical shift for identification of chemical structure. A variety of NMR pulse sequences are used to reveal the chemical shift and *J*-coupling information and extract structural information through detection of various nuclei (¹H, ¹³C, ¹⁵N, ³¹P).

Metabolome, metabolites, and metabotype. The metabolome consists of low molecular weight chemical compounds called metabolites that are present in biological specimens. These metabolites can either be produced as a result of endogenous cellular metabolism (endogenous metabolites) or as a result of exposure (environmental chemicals, food ingestion, drug intake), or derived from microbial metabolism.

Stable isotope resolved metabolomics. Some atoms can exist in multiple forms called isotopes. These isotopes are used for investigating (tracing) the fate of endogenous metabolites between metabolic pathways. Carbon 13 (¹³C) is an example of one stable isotope used in NMR metabolomics studies. Naturally occurring compounds contain only 1.1% ¹³C, and thus, high concentrations of these compounds or long NMR run times are needed for signal detection. Using compounds that are

enriched in ^{13}C enables the performance of experiments that trace the ^{13}C , to reveal the metabolism of the enriched molecules which have signal intensities above those of the endogenous background.

1.2 Applications of Metabolomics Analysis Methods

Cells, tissues, and biological fluids are rich in low molecular weight metabolites, and metabolomics involves the analysis of these low molecular weight metabolites [2–5]. Common biological fluids used in metabolomics include serum, plasma, saliva, urine, cerebrospinal fluid, feces, and exhaled breath. Unlike other omics (genomics, transcriptomics, and proteomics) technologies, metabolomics gives the most functional information about the system since metabolites are the final end products of genomic, transcriptomic, and/or proteomic perturbations [6] within a complex network of biological pathways. Metabolomics studies are designed to reveal the pattern of signals that increase or decrease as a function of health and wellness, or response to treatment. Through determining signals that associate with health and disease states, metabolites can be identified and mapped to biochemical pathways to provide insights into biological mechanisms.

In the past decade, advances in technology have enabled the application of metabolomics in a variety of diverse research areas that cover basic, biomedical, and clinical sciences to measure these metabolites in readily accessible biological fluids, cells, and tissues to correlate back with the responses at the cellular or organ level. The terms metabolomics, metabonomics, metabolic profiling, metabolic phenotyping, and metabotyping are widely used interchangeably by the research community to basically the same technique and procedures [7, 8]. The technologies that can be used to detect and measure these metabolites are very diverse. NMR spectroscopy and mass spectrometry (often coupled with chromatography) are the two main analytical platforms in metabotyping whereas other analytical techniques such as liquid chromatography coupled electrochemical detection (for detecting neuro transmitters [9] or capillary electrophoresis methods [10]) can be used in the targeted analysis. Advances in high-throughput, high-resolution analytical technologies and data analysis platforms have allowed for metabotyping in large-scale population-based Metabolome Wide Association Studies (MWAS) and metabotype quantitative trait locus (mQTL) mapping studies by using both NMR Spectroscopy [11–13] and mass spectrometry [14–16]. These methods are more useful than Genome Wide Association Studies (GWAS) and QTL mapping for discovering biomarkers for disease risk because metabolic perturbations are the most downstream products of all of the other omics activities. Furthermore, integrating of MWAS and mQTL with GWAS and QTL mapping studies improves biomarker identification.

A number of metabolome/phenome centers [17–21] have been established to provide services to the growing demand for metabotyping in large scale population-based studies. A metabotyping study can be conducted in the form of an untargeted or a targeted analysis method. Untargeted methods are typically used for discovery and hypothesis generation. In the untargeted workflow, signals that correspond to the metabolome (endogenous and exogenous molecules) are detected, and the data is subjected to subsequent statistical analysis, modeling, and bioinformatics analyses. Signals that are responsible for the differentiation of study phenotypes (e.g., difference in case vs control, difference in high vs. low dose) are then identified. On the other hand, targeted methods are more often used to test hypotheses. In this workflow, a set of analytes are selected to test a hypothesis, for example, about the disease state or response to treatment. The targeted metabolomics analysis involves the use of metabolite standards (including isotopically labeled compounds) and calibration curves. Statistical analysis and other modeling approaches are performed on the analytes once these metabolites are quantified. The findings from any untargeted study must be confirmed or validated by using an appropriate targeted metabolomics analysis method.

2 NMR Metabolomics

2.1 NMR Spectroscopy

In a typical ^1H NMR experiment, a radiofrequency pulse in a kHz range is applied using a transmitter coil in the NMR probe to the sample (in a tube) placed in the external magnetic field. Then the individual nuclei are in Boltzmann equilibrium aligned (low-energy state) or antialigned (high-energy state) with respect to the applied magnetic field. Once the RF pulse is applied, the Boltzmann equilibrium is perturbed and the nuclei in the low energy state become excited into the high-energy state. The nuclei are then relaxed back to the original equilibrium status by a mechanism called relaxation (nuclei–nuclei and nuclei–matrix) and a decaying signal is recorded in the receiver in the form of a free induction decay (FID), which is a time domain spectrum. The recorded NMR signal is a collection of decaying sine waves (sinusoids) and this analog electronic signal is digitized to create a free induction decay (FID). The FID is Fourier transformed to convert the time domain spectrum into a frequency domain spectrum (Fig. 1). A window function is also applied (apodization) during the Fourier Transform process to improve the NMR spectrum. It has been an active area of NMR research to study the chemical shift and J -coupling patterns of molecules with varying functional moieties and molecular conformation and there is a vast number of literature data (including NMR tables from instrument vendors) available that assists in structure elucidation purposes. A number of databases [22–26]

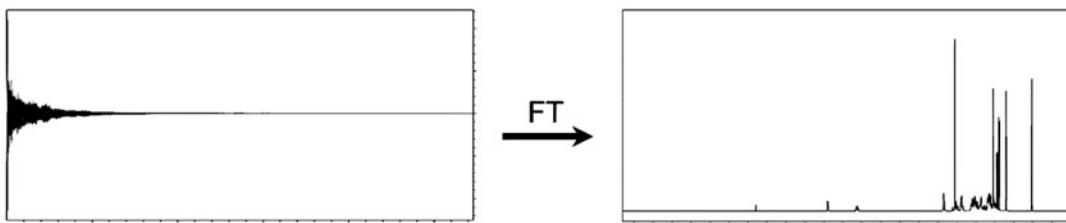


Fig. 1 NMR signal is measured as a FID and then Fourier transformed to obtain the frequency domain spectrum (example shown here is for a ^1H NMR spectrum of a cholesterol derivative)

are available for searching NMR spectra of molecules (HMDB, BMRB, Birmingham Metabolite Database, NMRShift DB2) and many software tools have been developed to predict NMR spectra based on NMR properties (ChemDraw, ACD, MNova). A variety of instrumental methods called pulse programs have been developed (and are still being developed) to collect one-dimensional (1D), two-dimensional (2D), and multidimensional NMR spectra, and these experiments assist in elucidating structure of the molecules. NMR spectroscopy is a powerful tool for elucidating structural details of both small molecules (e.g., identification of natural products and pharmaceutical compounds) and macromolecules (e.g., proteins and nucleic acids), and their molecular complexes and their dynamics. The resonance frequency is proportional to the applied external magnetic field so that nuclei resonate at higher radio frequencies on instruments with higher magnetic fields (e.g., 500 MHz at 11.7 T and 950 MHz at 22.3 T for ^1H). A higher resolution is created by a larger dispersion of the peaks in the NMR spectrum. Most NMR metabolomics studies have been conducted with 600 and 700 MHz field strengths, while 950 MHz have also been employed [27, 28].

2.2 NMR Metabolomics Analysis

NMR spectroscopy and chromatography coupled mass spectrometry are the two main spectroscopic methods used in the metabolomics investigations. Both platforms are highly suitable for metabolomics analysis, are complimentary, and have their own analytical strengths and weaknesses. This chapter focuses on NMR spectroscopy which can detect a wide variety of chemical classes (alcohols, amines, amino acids, aromatics, bile acids, carboxylic acids (including fatty acids), ketones, lipids, nucleosides, nucleotides, phenols, sulfur-containing compounds, and sugars) including compounds which are difficult to ionize on mass spectrometry systems.

The metabolites in a complex biological specimen (urine, serum, plasma, saliva, feces, etc.) can be measured simultaneously and provide a metabolic fingerprint of the system. We recover information on metabolic changes related to the metabotype by applying standard statistical techniques, chemometrics,

bioinformatics, and other data analysis and mining methods on spectral data. In addition, stable isotope resolved metabolomics [29] is used to extract metabolic flux information [30] in specific metabolic pathways. Cellular metabolism is dynamic and is a complex system of metabolic networks. Therefore, metabolic flux analysis [31] using stable isotopes facilitates determining the metabolic reaction rates (fluxes) for better understanding of the cellular mechanisms at systems biology level. In addition, stable isotope-based tracer experiments can be used to analyze the fate of certain metabolite(s) in metabolic pathways (e.g., metabolites of drug molecules or an external environmental compound).

The high-resolution NMR spectroscopy (500 MHz and above) technique is a quantitative, robust, highly reproducible, and non-destructive technique [32] for analysis of biological fluids, as well as extracts of tissues and cells. NMR typically does not require chromatographic separation of metabolites from the complex mixture (biological fluid or tissue/cell extract), while mass spectrometry methods typically require chromatographic liquid—or gas—separation and is destructive to the sample.

Sample contamination and carryover is a complex issue in mass spectrometry studies but is not problematic in NMR studies since there is no physical contact between the sample and the spectrometer. Furthermore, routine automatic data acquisition methods coupled with automatic sample changers are available which accommodate samples in 96 tube format in NMR tube racks. Here samples are stored under refrigerated conditions and equilibrated to the desired temperature before and after inserting the tube into the magnet. This is especially helpful in acquiring data sequentially for batches of samples in large-scale metabolomics studies. Typically, it requires a few minutes to acquire a ^1H NMR spectrum, and about 15–20 min to acquire a routinely selected set of NMR spectra (1D ^1H and 2D-JRES) for a single sample.

In longitudinal larger-scale epidemiological or clinical metabotyping studies in particular, it is not possible to acquire data for all samples simultaneously. Such analyses require multiday and multi-batch sample preparation and data acquisitions. Therefore, the longitudinal stability of NMR spectroscopy makes it ideal for high-throughput data acquisition needs for such studies [33]. The data acquired in multiple batches (with extensive quality control procedures described in Subheading 3) can be combined for analysis, including workflows for processing multicohort untargeted metabolomics [34].

For NMR metabolomics analysis, samples are typically prepared and transferred to a 5 mm NMR tube, while lower diameter (1.7–3 mm) tubes are available if limited in sample volume, and NMR flow cells or microprobes are also available to accommodate samples with very low volumes (e.g., 10 μl). There are techniques such as hyperpolarized NMR methods [22, 35, 36] that can

increase the sensitivity to detect atoms of low natural abundance (such as ^{13}C and ^{15}N) thus enabling the use of ^{13}C and ^{15}N NMR (1D and 2D, for example) spectroscopy for detection of metabolites in NMR metabolomics analysis in a short period of time. Additionally, there are hyphenated NMR systems that are coupled with liquid chromatography (LC-NMR) or solid phase extraction technique (SPE-NMR) in cases where a reduced complexity of the spectra is needed. Combined LC-NMR-MS or LC-SPE-NMR- MS systems have also been developed, thus enabling recovery of exact molecular weight information of metabolites in addition to the NMR spectra for improved identification and quantification of metabolites in the profiling of complex biological samples [22].

2.3 NMR Metabolomics Workflow

2.3.1 Study Design

An example workflow for broad spectrum (or untargeted) metabolomics data using NMR is shown in Fig. 2. This workflow includes study design, sample collection, sample selection, sample randomization, sample preparation, data acquisition, data preprocessing, and data analysis.

Metabolomics investigations have been conducted that have revealed that genetics (e.g., race, gender, polymorphisms), nutrition (e.g., diet, nutrients, weight, gut microbiome), mental health (e.g., stress, depression, cognition, behaviors), and exposures (e.g., tobacco use, pollution, drugs, medications, personal care products)

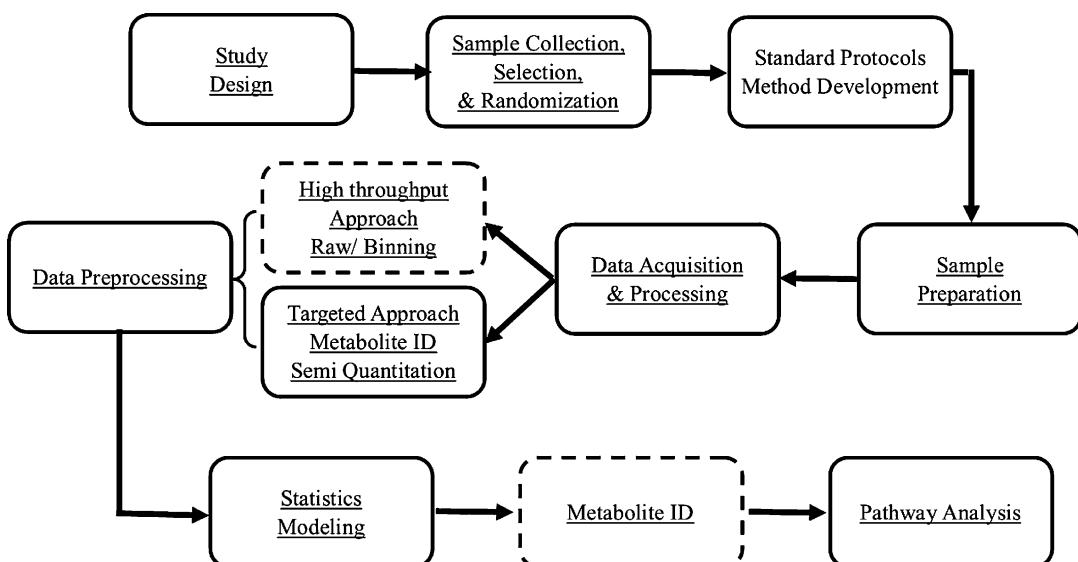


Fig. 2 Schematic diagram explaining the NMR metabolomics workflow. One advantage of NMR metabolomics is that both untargeted and targeted data analysis can be performed using the same NMR spectrum. In this workflow, metabolites are identified after using statistics and modeling approaches to reveal signals that are important to defining the study phenotype. Quality Controls (underlined) are included at each step of the pipeline

all influence the metabotype [37]. Therefore, it is critical to take these factors into consideration when designing the sample collection for the study to ensure that the appropriate metadata is collected for all subjects, and that the study is properly powered. In addition, when selecting samples from a repository, these factors should be considered in the power calculations in order to select samples for subjects for each of the phenotypic groups, to match the subjects between phenotypic groups for these factors, and to determine other uncontrollable confounding factors for consideration. Otherwise, subsequent statistical analysis methods will fail to achieve the desired goals of the study.

2.3.2 Sample Selection and Randomization

The number of samples needed for the study will depend on the variables mentioned above, as well as the strength of the phenotype. More samples are needed when the study involves weaker phenotypes (e.g., control vs. mild anxiety), whereas smaller number of samples are sufficient for situations with stronger phenotypes (e.g., healthy vs. cancer cases) to extract desirable information. For a validation or a confirmation study, a sufficient number of samples that can be determined by power calculations such as multivariate power calculations [38] must be used to achieve statistically significant results. Selection of biospecimens derived from a different cohort is preferable for such validation studies. Samples must be randomized for both the sample preparation and analysis sequence in the data acquisition in order to avoid any analytical bias.

2.3.3 Biospecimen Availability

It is also necessary to consider other factors such as the type of anticoagulant (or preservative, if any), storage condition, and number of freeze–thaw cycles. Biospecimens that have undergone different collection procedure, a different number of freeze–thaw cycles, or different storage temperatures ($-20\text{ }^{\circ}\text{C}$, $-80\text{ }^{\circ}\text{C}$) can give rise to different profiles. Ideally, samples with no freeze–thaw cycles and stored at $-80\text{ }^{\circ}\text{C}$ are best suited for metabolomics analysis. This can be achieved by preparing multiple aliquots at the sample collection stage, prior to freezing. The most important thing is to keep all conditions of the samples the same within a particular metabolomics study. The type of anticoagulant used for samples can affect the NMR spectrum. Serum is best suited for NMR metabolomics, while heparinized plasma (preferable over EDTA or citrated plasma) can also be used. EDTA (and its Ca^{2+} , and Mg^{2+} complexes) or citrate in the biospecimens results in signals that hinder or obscure the signals of some endogenous metabolite signals and therefore such peaks must be removed from further analysis. Therefore, valuable information is lost if EDTA or citrate is used as an anticoagulant in samples. However, it was found that useful biochemical could still be effectively recovered using samples collected with these anticoagulants [39]. Once a

biomarker metabolite or a set of metabolites is discovered using an untargeted metabolomics analysis method, it is important to confirm or validate the findings by designing and applying an appropriate targeted analysis method by even using samples generated in a different cohort. Subsequently, blinded studies can also be designed for biomarker validation purposes.

2.3.4 Biological Sample Types

Human subject metabolomics investigations using NMR have generally used urine, serum, or plasma, while saliva and stool samples are becoming more commonly utilized. In addition to these available biological matrices, a wide variety of other sample types such as cerebrospinal fluid, seminal fluid, bronchoalveolar lavage fluid, amniotic fluid, seminal fluid, and extracts of organ tissue have been used in NMR metabolomics. On the other hand, any available biological matrices (e.g., urine, serum or plasma, feces, tissue, organ, and other biological samples) can be used for studies involving animal models.

2.3.5 Sample Collection and Storage

Standard established protocols are needed for the sample collection and storage [40] of biological samples. Consistency in sample collection and storage is an important consideration [41]. For biomarker discovery and validation, metabolomics data analysis using combined sample sets derived from longitudinal or multicenter multicohort population studies is important for increasing the statistical power and, also, is challenging due to inconsistencies in the sample collection in the cohorts and storage conditions in the biobanks. There are ongoing efforts to integrate metabolomics data obtained from samples in multiple biobanks [42]. While it is not always possible to obtain biospecimens from different cohorts that have been collected and stored in identical manner, it is important to have access to the protocols regarding the collection and handling of the samples in order to understand differences that may arise in the profiles between cohorts.

2.3.6 Sample Preparation

NMR metabolomics is typically performed on samples (biofluids, extracts of biofluids, cells, or tissue) in solution. There are a number of standard protocols designed for metabotyping in both small and large-scale studies and published in literature for preparing biological samples for NMR spectroscopy [32, 43]. There are other methods available for analyzing metabolites in solid samples such as intact tissue biopsy samples using high-resolution magic angle spinning NMR [44, 45] or using in vivo magnetic resonance spectroscopy [46] to record metabolite profiles of organs or tissues (in magnetic resonance imaging). Regardless of the method utilized, standard established protocols for the sample preparation should be used and documented, and method development work should be undertaken for new biological matrices using representative samples before handling the real study samples.

1. Minimal sample preparation method

Biological samples such as blood (serum and plasma), urine, cerebrospinal fluid (CSF), and saliva can be prepared by mixing an NMR buffer into each sample aliquot. This method has the advantage of preserving the sample so that it can be analyzed by other assays following the NMR metabolomics analysis. NMR buffer typically contains a phosphate buffer ($\text{pH} = 7.4$), an internal standard (trimethyl-silyl-[2,2,3,3-d₄] propionic acid (TSP) or 4,4-dimethyl-4-silapentane-1-sulfonic acid (DSS-d₆), chemical shift indicator), sodium azide (NaN₃, to prevent bacterial growth), and D₂O (for deuterium locking of the spectrometer).

2. Extraction sample preparation method

Samples generated from tissues, feces, or blood (serum or plasma) can be prepared by homogenization or extracting with a solvent (organic or aqueous or mixtures), drying, and reconstitution in an NMR buffer [27, 47–49]. Ultrafiltration techniques using molecular weight cut-off filters can also be employed to remove macromolecules from biofluid samples such as serum, plasma, cerebrospinal fluid, and breast milk before mixing with NMR buffer solution [50, 51].

2.3.7 Data Acquisition

The data acquisition methods selected depend on the study question. Typically, NMR metabolomics uses 1D ¹H NMR spectroscopic methods because of the high sensitivity of ¹H (and simplicity) in contrast to very low sensitivity of ¹³C or ¹⁵N because of the very low natural abundance. Serum and plasma samples contain macromolecules such as proteins, lipids, and lipoproteins. Atoms in these macromolecules relax faster and therefore the signals deriving from these molecules become broadened making it harder to analyze small molecule metabolites.

A number of different NMR pulse sequences are commonly used in ¹H NMR metabotyping of biological samples to address these issues to improve the spectral quality of the small molecules. The four most commonly used sequences in routine metabolomics data acquisition from blood samples are NOESY presaturation (presat), Carr-Purcell-Meiboom-Gill (CPMG), J-Resolved, and Diffusion-edited method [43].

1. NOESY presat: The NOESY presat pulse sequence is the first increment of a NOESY pulse sequence [24] (RD-gz(1)-90°-t-90°-tm-gz(2)-90°-ACQ, where RD is the relaxation delay, t is a short delay between pulses, tm is the mixing time, 90° is the duration of the radio frequency pulse, gz(1) and gz(2) are the pulse field gradients, and ACQ is the data acquisition time).
2. Carr-Purcell-Meiboom-Gill (CPMG) spin-echo sequence: The CPMG spin-echo sequence uses a CPMG presat pulse

sequence which is a relaxation editing method applied to attenuate signals derived from macromolecules (faster relaxing molecules) in order to analyze small molecule metabolites. The CPMG pulse program (RD-90°-(t-180°-t)_n-ACQ, where in addition to definitions above, 180° is a 180° RF pulse, t is spin-echo delay, n is the number of loops) is a relaxation-editing pulse sequence.

3. 2D J-resolved sequence: ¹H J-resolved experiment provides J-coupling information and aids in metabolite identification purposes. In addition, the use of skyline or sum projections of 2D J-resolved spectra [52, 53] have shown to be useful in high throughput binning approaches to reduce spectral overlaps. The J-Resolved pulse program [36] has the sequence RD-90°-t1-180°-t1-ACQ, where t1 is an incremented time period.
4. 1D diffusion edited (DOSY): 1D DOSY is used to attenuate signals of faster moving small molecules in order to observe slower moving macromolecules such as lipoproteins and lipids. DOSY is useful in characterizing lipoprotein fractions in blood based on diffusion coefficients [54]. The Diffusion-edited NMR spectra use a pulse sequence with bipolar gradients (RD-90°-G1-180°-G1-90°-G2-T-90°-G1-180°-G1-90°-G2-t-90°-ACQ, where RD is a relaxation delay, 90° is a 90° RF pulse, G1 is the pulsed-field gradient that is applied to allow editing, 180° is a 180° RF pulse, G2 is a spoil gradient applied to remove unwanted magnetization components).

The diffusion delay D is the time during which the molecules are allowed to diffuse, the period (90°-G1-180°-G1-90°-G2-T-); t is a delay to allow the longitudinal eddy currents caused within the sample to decay [36].

For urine, NOESY-presat, and J-resolved spectra are used frequently for data acquisition. Additional selective 1D (e.g., selective 1D-TOCSY), 2D and multidimensional spectra (TOCSY, HSQC, HMBC, HSQC-TOCSY) that are needed for metabolite identification can be acquired using a subset of selected samples, or a pooled sample. The 2D ¹H-¹³C HSQC experiment is particularly useful for identification and quantification of metabolites [55]. Other experiments such as ³¹P NMR can be used to study molecules with phosphate groups such as energy metabolites (ADP, AMP, and ATP). ¹³C or ¹⁵N enrichment is used in stable isotope resolved metabolomics studies to investigate metabolites derived from the enriched isotopic compound over that of background from endogenous sources.

3 Quality Control (QC) Methods

Quality Control (QC) starts with the development of an analytical and data analysis plan that is consistent with the metabolomics workflow and the goals of the investigation. This plan is reviewed by all investigators involved in handling samples, data acquisition, and data analysis. In the metabolomics laboratory, standard operating procedures and quality control checklists are utilized for each process: (a) documentation and recording, (b) biospecimen receipt, inventory, and storage, (c) biospecimen preparation, (d) instrument tuning and calibration, and adequacy of standards pre- and poststudy sample analysis, (e) data acquisition of study samples, and (f) assessment of quality control standards during study sample data acquisition. A standard laboratory procedure for following, reporting, and documenting all steps in the workflow should be established and followed by all laboratory personnel.

Types of quality control standards can vary between laboratories based on the methods being employed. The use of reference material and the use of quality control pools are optimal.

1. Reference material

For large scale studies, longitudinal studies, or studies that will involve multiple sites conducting metabolites analysis, it is important to establish a reference material (Fig. 3a). For example, creating a large pool of human urine, or plasma/serum, that is aliquoted and provided to all participating laboratories can be an ideal way to monitor the metabolomics signatures longitudinally and between sites. These reference materials are interspersed with the randomized study samples in each batch, and the signals/peaks assessed for variability within batch runs, comparisons between batches, or in longitudinal analysis. Such reference material facilitates data filtration and merging of data acquired from multiple batches or between laboratories. This external pooled sample can be prepared in-house or purchased. For example, a large number of volunteers contribute biospecimens to a bulk pool, and aliquots are made and stored frozen for subsequent use. The samples must have the same freeze-thaw cycles throughout the entire studies in order to use in the subsequent QC and data analysis procedures. Standard reference material (e.g., NIST SRM1950 [56]) can also be used as an external reference standard. Typically, external reference material samples are prepared identical to the randomized study samples and interspersed in each batch [43].

2. Study pools

Study samples can also be pooled to create a quality control within each batch (Fig. 3b, c). The use of QC study pools [57, 58] not only enables assessment of data quality but also assists in metabolite identification. In studies where sufficient

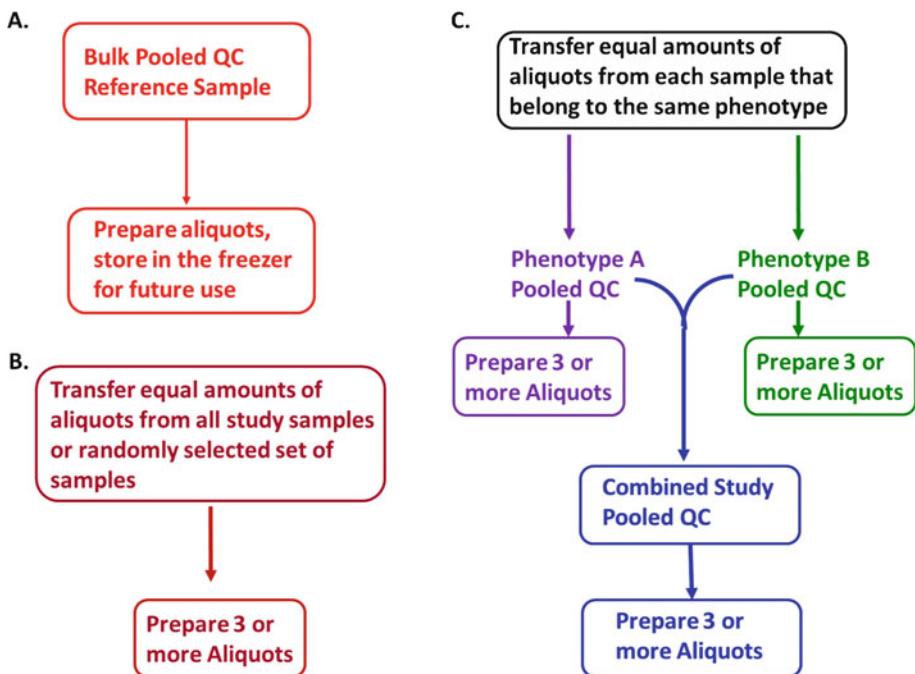


Fig. 3 Schematic diagram showing preparation of QC samples. **(a)** Preparation of external pooled QC reference samples. Biological fluids (serum, plasma, or urine) can be collected from volunteers to make a bulk pooled QC reference material, which is aliquoted and stored in desirable volumes at -80°C . **(b)** Preparation of total study pooled QC samples. For small studies, an equal amount from each sample in a study or a randomly selected subset of samples is transferred into a container for pooling. Contents are thoroughly mixed and aliquots of the pooled QC are prepared. For large studies, study pooled QC samples can be prepared by randomly selecting a subset of samples for pooling and aliquots of these study pooled samples are used for each of the batch of samples. In addition, batch pooled samples can be prepared by pooling equal amounts from all samples within the batch. **(c)** Preparation of both phenotypic and combined pooled QC samples using the study biospecimens

amount of each biospecimen is available, phenotypic pools and a total study pool can be prepared. For large-scale metabotyping studies that involve multiple batches of samples, it is necessary to include QC pools to assess both intra- and inter-batch variation within a specific study. In studies such as longitudinal studies, it is also impractical to prepare a pooled QC samples by using all samples in the study. In such cases, a random selection of samples representing the whole study can be used. For example, the entire study sample set can be randomized into batches, and the first batch can be used to prepare a pooled QC sample at the beginning of the study, which is aliquoted in sufficient amounts, and used across the whole study.

3.1 QC of Data Acquisition

NMR spectrometer performance and the field homogeneity is evaluated routinely, using NMR reference standards (e.g., by optimizing for best line shape and line width), to ensure high quality of data acquisition. In addition, NMR spectra are visually inspected for

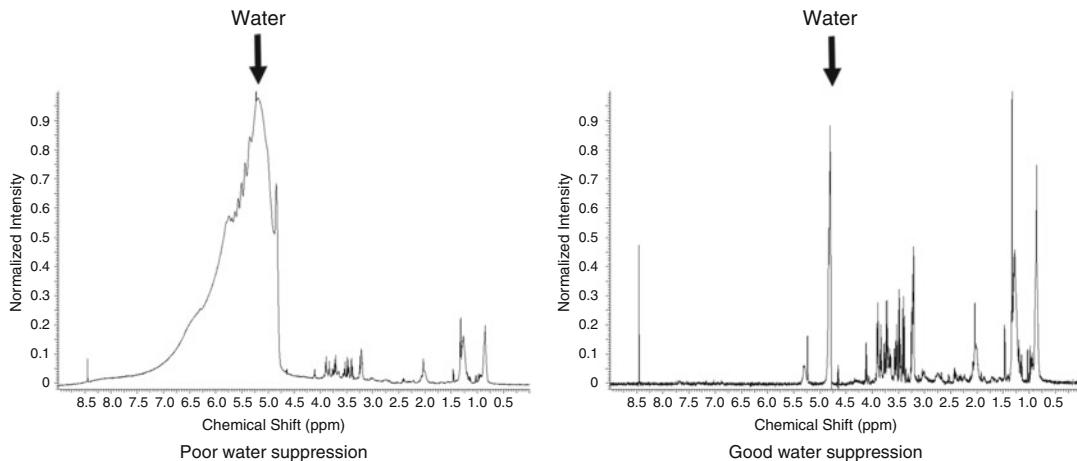


Fig. 4 Effect of water suppression on the quality of the NMR spectrum of serum. Signal intensities of metabolites are very low in the spectrum with the poor water suppression

water suppression, line shape, phase, and baseline. Water signal in aqueous samples needs to be suppressed as metabolites are in micromolar (μM) to millimolar (mM) concentration whereas the concentration of water is $\sim 55 \text{ M}$ (i.e., $110 \text{ M}^1\text{H}$). Otherwise HDO signal will dominate the spectrum making it difficult to analyze or identify comparatively small metabolite peaks (Fig. 4). Narrow linewidths of NMR signals increases the peak resolution, especially in complex spectra with overlapped peak areas. Typically, a linewidth of $1.0 \pm 0.5 \text{ Hz}$ (without line broadening factor) at half height of the singlet peak of the chemical shift indicator is desirable. In addition, the spectral lines should be Lorentzian in shape. If the water suppression is not sufficient or the line shape and width are not to the specifications, then the spectra needs to be rerun or discarded from the analysis.

3.2 Analytical Reproducibility

The quality of normalized spectral data is evaluated using the pooled QC samples. The QC pools are technical replicates and there should be minimal variation among these samples. An initial evaluation of data can be carried out using a nonsupervised analysis method such as principal component analysis (PCA) and other unsupervised methods such as intraclass correlation [59], or hierarchical clustering trees [33] can also be used to assess the analytical reproducibility. In PCA, a tight clustering of pooled QC samples and dispersion of study samples [60, 61] indicates that the analytical variation is minimal compared to biological variation between study samples. If the QC samples were created from the study samples, then the QC samples in the PCA plot should be centered among the sample group from which the QC was created. If external pools were used then the location of the QC pools in relation to the study samples does not provide any additional

information. QC samples can also be used for applying data filtration methods to obtain the overall high quality of data. Typically, coefficient of variation (determined as %CV) within a variable is assessed using QC samples, and the data with higher %CV (>30%, for example) are excluded from further data analysis of the entire dataset.

4 NMR Metabolomics Data Analysis

4.1 NMR Data

The position of NMR signals of observed atoms (e.g., ^1H) within a particular molecule are measured as chemical shifts in parts per million (ppm). The ^1H NMR spectrum of alanine is shown in Fig. 5. The alanine molecule has three carbon atoms with attached protons with different functional groups (Fig. 5). The ^1H NMR spectrum of alanine in the solvent D_2O has two distinct signals: one for the methyl group (signal A), and one for the H attached to the C-2 carbon (signal B). ^1H atoms in the NH_2 group and OH group usually exchange with deuterium in the solvent and are not observable on the NMR time scale (millisecond). The ^1H signal splits into coupling patterns that are dependent on the neighboring atoms

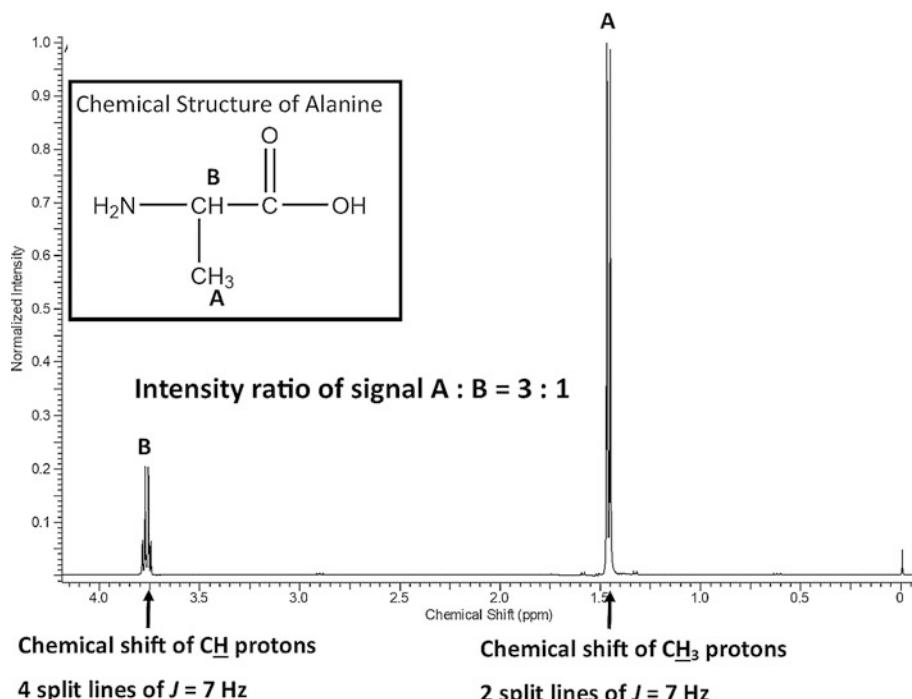


Fig. 5 NMR Spectrum of alanine showing chemical shifts and J -coupling of CH_3 and CH proton groups. The chemical structure of alanine is shown (in the box) above the NMR spectrum. Protons in the methyl group (A) are split into 2 lines by a neighboring proton (B) and the CH is split into 4 lines by neighboring methyl protons (A). Proportion of peak intensities of signal A: B is 3: 1

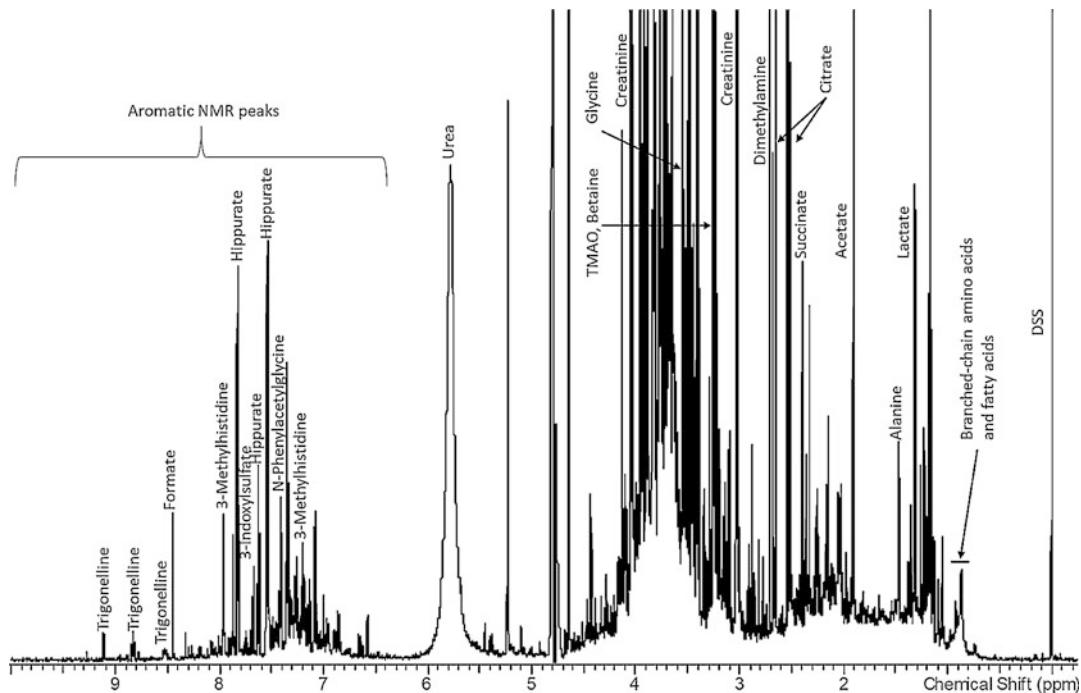


Fig. 6 A 700 MHz ^1H NMR spectrum of a pooled urine sample with example identified metabolites. 4,4-dimethyl-4-silapentane-1-sulfonic acid (DSS), is used as an internal standard for chemical shift referencing and relative concentration determination

and their conformations within the molecule (J -coupling). In the spectrum of alanine, the methyl signal is split into a doublet (two lines, J coupling = ~7 Hz) by the neighboring CH proton, and the CH signal is split into a quartet (four lines, J coupling = ~7 Hz) by the neighboring CH_3 protons. The intensities of each signal are proportional to the number of ^1H atoms underlying each signal ($\text{CH}_3:\text{CH}$ is 3:1). The chemical shifts and splitting pattern (J -coupling) enables the deconvolution of peaks in the NMR spectrum and is used in metabolite identification. Biological samples are complex mixtures and may contain hundreds of molecules detectable by NMR belonging to various compound classes with varying concentrations that results in a large dynamic range. Some of these signals are characteristic metabolite peaks, while others are indistinguishable due to overlap, and others represent unknowns. A typical ^1H spectrum of human urine with some metabolite annotations is shown in Fig. 6. To assist with identification of metabolites in biological samples, multidimensional methods or peak deconvolution can be incorporated; some of these techniques are described later in the chapter.

4.2 Data Preprocessing

As described earlier, NMR data are recorded as FIDs and are then Fourier transformed into spectra and this is performed at the instrument level. Before the Fourier transformation step of the

FID, zero filling and apodization is applied to increase the quality of the spectrum. In the zero filling step, the size of the dataset is artificially increased by adding zeros to the FID. It increases the digital resolution of the peaks without affecting the chemical shift or J -coupling. During apodization, a window function is applied to the FID (after zero filling) in the form of an exponential multiplier with a line broadening factor. Exponential multiplication increases the signal-to-noise ratio of the spectrum and the line broadening factor reduces the resolution. Therefore, selecting the optimum parameters for apodization is a tradeoff between the resolution and the signal-to-noise ratio. The resultant spectral signals are Lorentzian in shape for a spectrum recorded in a homogeneous magnetic field. There can be phase errors in the spectrum due to the time delay between the NMR excitation pulse and the switching on the receiver. Phase correction is applied to obtain a spectrum in the absorption mode. The next step is to correct baseline offsets by subtracting a low order polynomial fit using software. Finally, the origin of the chemical shift scale is calibrated to the position of the internal standard signal or other standard peak. One of the complications in using NMR-based metabolic profiles is the presence of very small but significant variations in the chemical shift of metabolite peaks between spectra. NMR chemical shift depends on the microenvironment within the molecule and subtle changes in pH or ionic strength among samples can result in variations in peak positions. A peak alignment algorithm must be used in such cases. A number of algorithms have been reported in literature, for example, recursive segment-wise peak alignment (RSPA) [62], interval-correlation-shifting (icoshift) [63], hierarchical cluster-based peak alignment (CluPA), [64], correlation optimized warping (COW) [65], dynamic time warping (DTW) [65], and recursive unreferenced alignment of spectra (RUNAS) [66]. An example of peak alignment using CluPA algorithm applied in NMRProcFlow using urine samples is shown in Fig. 7. Although these methods improve peak alignment, they each suffer slight limitations because of challenges associated with complexity and signal overlap in NMR spectra. However, it is useful in situations where certain select peaks need to be aligned.

4.2.1 Phenotypic Anchoring

After phase and baseline correction, and chemical shift referencing, NMR data is transferred to a data format accessible for further statistical analysis and modeling. An NMR spectrum is a data matrix consisting of chemical shift and peak intensity information. A single spectrum usually contains up to 65,536 data points. There are two methods for handling NMR data: (1) Using raw data (full resolution) or binned data (data reduction); (2) Peak deconvolution, identification, and relative quantification of metabolites. Full-resolution data analysis is computationally intensive and generally requires high-performance computing clusters. In such cases, the

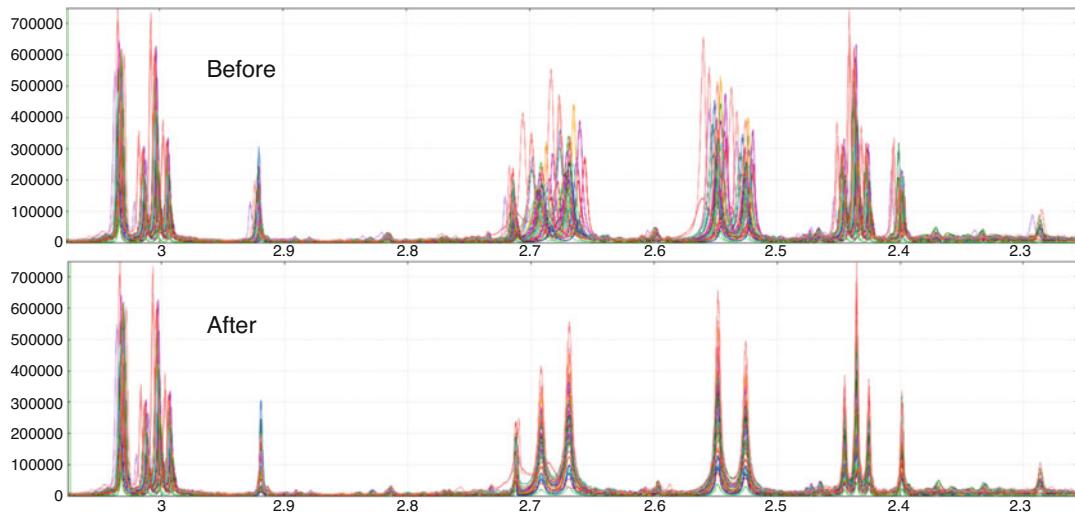


Fig. 7 NMR spectra of urine samples aligned by CluPA algorithm using NMRProcFlow software

raw data can directly be used for modeling. Otherwise, data must be preprocessed using a process called binning. Both of these processes offer high-throughput data analysis. Some artifacts and unwanted spectral regions must be removed prior to further analysis of either raw or processed spectral data. These include residual water signal, internal standards, and applied chemicals or medication and their metabolites used as treatments in the study. The NMR peaks in the spectra can be deconvoluted into metabolites and quantified. Despite efforts toward automation of metabolite identification, fitting, and quantification of metabolite levels in 1D ^1H NMR spectra, this process is somewhat hampered by signal overlap, high dynamic range of metabolites, small shifts in chemical shift between spectra for the same metabolites, and complex splitting patterns (multiplicity) of the peaks of a metabolite, and level of noise. Another disadvantage of automation is that unknown metabolites cannot be identified, since only known metabolite peaks can be modeled in automating.

4.2.2 Data Analysis Using Raw NMR Data

In order to recover meaningful information on metabolic biomarkers using NMR data, data preprocessing steps and application of a variety of bioinformatics or statistical analysis methods are necessary. NMR data can be analyzed by using full resolution spectra or binning approach with reduced dimensionality. In either of the methods, NMR data is typically organized in a matrix in such a way that the samples (observations) are in rows and variables are in columns. In the full resolution data analysis, data are directly imported into analysis software of choice. Either FIDs or raw Fourier transformed data can be imported into software and an array of data can be prepared using software codes. The RBNMR [67] code

written in Matlab is a useful tool that can prepare a data structure that includes metadata. Appropriate Matlab codes must be used (in-house or already coded) to extract the data from such data structures into further analysis. In-house or publicly available codes written in other languages (R, Python, Java, C++) can also be used for this purpose. It is also possible to import raw FIDs (before Fourier transformation) directly into software and automate data processing and preprocessing steps by using in-house or a variety of publicly available tools or algorithms. These steps include Fourier transformation, phase and baseline correction, chemical shift referencing, peak alignment, removal of unwanted regions, and normalization and scaling. Once generated, a variety of data analytic methods can be applied on the resultant data matrix to recover biomarker information from full resolution NMR data (methods outlined or described in other sections).

In a Bayesian-based method called CRAFT [68] (Complete Reduction to Amplitude Frequency Table), which is a time domain approach, FIDs are directly converted into a frequency-amplitude table. The workflow in the CRAFT consists of a two-step process [68]. The first step involves digitally filtering and down sampling of data to obtain several sub-FIDs. During the second part of the CRAFT process, these sub-FIDs are modeled as sums of decaying sinusoids using the Bayesian probability theory. Authors report that this approach is better than the analysis of Fourier transformed frequency domain spectra for identifying changes in metabolite signals in complex spectra with signal overlaps, such as biological samples [68].

4.2.3 *NMR Binning*

Binning or bucketing implements data reduction of NMR metabolomics data. In the conventional binning approach, the spectrum is divided into evenly spaced windows called bins or buckets and the peak area in each bin is integrated. The bucket width is typically 0.01 or 0.04 ppm (some uses 0.001 ppm bin width). Binning can also reduce the effect of variations in peak positions in spectra to some extent. The main disadvantage of conventional binning is that parts of metabolite peaks can fall into the neighboring bins or buckets. Several methods have been proposed to overcome this: intelligent bucketing [69, 70], adaptive binning [71], and optimized bucketing [72]. In the intelligent binning, the bin boundaries are loosened by a specified percentage (e.g., 50% looseness) allowing peaks to fall within the same bin. Adaptive binning corrects for variation in chemical shifts through utilization of a wavelet transform. The optimized bucketing algorithm generates an average spectrum of all data and defines bin boundaries using that spectrum. The NMR binning of projections of 2D J-resolved spectra can be performed by using the JBA algorithm [73] employed in the R-based MWASTools [74] software.

4.2.4 NMR Data Normalization

Data normalization is one of the most important steps in metabolomics analysis, and the goal is to remove any variation in total amount of material between samples. Specifically, it is important for analyzing urine samples where concentration can have changes in orders of magnitude. Similarly, it is important to normalize data when analyzing cell or tissue extracts where it can be difficult to control analyte levels due to factors such as sampling or extraction efficiency. These changes would be disproportionate to the relevant biological or biochemical variations attributed to metabotype, thus hindering extracting valuable biomarker information. Despite many methods in the literature for data normalization, no single approach is optimum for any given study. The data must be processed appropriately according to the study and type of samples and data. The normalization of data is typically performed row-wise (on observations). Commonly used normalization methods include constant sum normalization, probabilistic normalization, cubic spline normalization, normalizing to a standard peak, and normalizing to the total volume collected, specific gravity, or to the total mass of sample used. In the constant sum normalization, each integral (or intensity) is divided by the sum of all integrals or intensities and typically multiplied by a constant number such as 1000. Probabilistic quotient normalization (PQN) [75] is based on the calculation of a most probable dilution factor by looking at the distribution of the quotients of the amplitudes of a test spectrum by those of a reference spectrum. It was shown that the PQN method is more robust and more accurate than the widespread total sum normalization using experimental spectra of a complete metabolomics study as well as simulated spectra. It should be noted that the choice of a standard metabolite should be carefully selected and it should not have biological variation due to metabotype. For example, though normalization to the creatinine peak in urine samples is common in clinical studies, creatinine would not be a suitable choice where it is expected to change in patients (e.g., in kidney disease).

4.2.5 Centering, Scaling and Transformation of Data

Appropriate data pretreatment could include centering, scaling or transformation, methods which are performed column-wise (on variables) in a table of data [76]. Centering (mean centering) helps to adjust for the skew between high and low abundance metabolites in order to emphasize the relevant variation (covariance) between biological samples. Centering is done by centering the mean of each column (variable) to zero and subtracting the value of the variable from the mean of each column. Scaling of data is a method that adjusts for the differences in magnitude between metabolites by converting the data into concentration change relative to a scaling factor [76].

Data analysis without this preprocessing step could lead to loss of information arising from only slight changes in metabolite levels

in the metabolic pathways. Metabolites with higher concentration levels could become dominant and hinder small but significant biological variations in the metabotype. Scaling methods fall under one of two subclasses where the scaling factor is either (1) a data dispersion measure (e.g., standard deviation) or (2) a size measure (e.g., mean, median) [76]. Examples of the data dispersion method which use spread as the scaling factor are: auto scaling; Pareto Scaling; Variable stability (VAST) scaling [77]; range scaling. In auto scaling (also called unit variance (UV)), the standard deviation is used as the scaling factor. Because in this method all metabolites become equally important, one potential problem is the artificial inflation of small values. VAST scaling builds upon auto-scaling but aims to focus on stable metabolites by using both standard deviation and the coefficient of variation as scaling factors. Pareto scaling uses the square root of standard deviation as scaling factors, and can reduce the dominant effect of high metabolite concentrations while minimizing potential inflation of low concentration metabolite data. In the range scaling method, the range between minimum and maximum value is used [76]. Examples of size measure scaling (with a magnitude-based scaling factor) are: level scaling and median scaling. Level scaling uses mean concentration as the scaling factor, and converts metabolite concentration changes into changes in percent concentration. Level scaling is an effective approach when large relative changes are expected but typically median scaling is a the more robust size measure scaling approach. Transformations are used to make skewed distributions of data more symmetric. Log and power transformation are some examples. Since normalization and scaling have different purposes, it is possible to use a combination of these methods.

4.3 Data Analysis Methods

As outlined earlier, the NMR metabolomics data can be analyzed in either raw or preprocessed forms. A vast number of statistical analysis and modeling approaches are available in the literature and the number is rapidly increasing. Most of these analyses are performed by using Matlab (or Mathematica), R, Java, or Python codes. Some methods have focused on integrating the whole NMR metabolomics pipeline (from importing, preprocessing, analyzing, and graphical visualization to identifying biomarker metabolites) while others have been developed for addressing specific data processing, modeling, or identifying needs.

4.3.1 Multivariate Data Analysis

Metabolomics analysis involves data in high dimensional data matrices: typically, samples in rows and variables in columns. The use of multivariate data analysis methods [78] is common in analyzing metabolomics data [79]. Multivariate data analysis is a projection-based method that uses eigenvalues and a number of variations have been reported in literature [78, 80, 81]. The non-supervised multivariate method, principal component analysis

(PCA), is a technique that can be used to observe pattern, trends, or outliers in a data set. Quality control of data at the sample level can also be assessed using PCA (QC pooled samples must be tightly clustered in PCA [60, 61]). Supervised analysis methods such as PLS-DA and OPLS-DA can be used to identify discriminatory metabolites or features that are perturbed in the biological system. The OPLS-DA method is used to filter out unwanted variation within data that are not due to biochemical/biological changes in the metabotype. The scores plot typically depicts the subjects and the trends in PCA or group separation in PLS-DA and OPLS-DA and the loading plot shows the variables that are responsible for the trend or group separation. The models must be cross-validated in order to avoid overfitting. Another important feature in the supervised multivariate data analysis (PLS-DA and OPLS-DA) is the Variable Influence on Projections (VIP) plot, which generates a list of variables (metabolites or features) that are responsible for the discrimination of the study phenotypes. The coefficient plots also provide biomarker information in the OPLS-DA which can be used to generate statistical correlation spectroscopic plots such as STOCSY (described later). Most of the methods in literature have used codes developed in Matlab or other similar programming languages or commercially available software packages such as SIMCA. Since metabotype is dependent on many factors such as genetics, diet, disease status, and environment, there can be situations where confounding factors can affect the outcome of modeling using multivariate data. Posma and coworkers reported that Covariate-Adjusted Projection to Latent Structures (CA-PLS) [82] approach can be used in such situations where confounding is an issue.

4.3.2 Statistical Correlation Spectroscopy

The development of statistical correlation spectroscopic tools was based on the covariance in peak intensities across multiple independent samples to generate associations between signals that arise from the same molecular structure [8]. Statistical total correlation spectroscopy (STOCSY) and an extended collection of software tools based on STOCSY using statistical correlation spectroscopy has been introduced and some select tools are described below. Statistical total correlation spectroscopy (**STOCSY**) [83] is a method that assists biomarker identification in NMR metabolomics and displays the correlating peaks in the NMR spectra. It differs from most standard 2D correlation spectroscopy (such as TOCSY) with the advantage that it correlates intramolecular peaks and also can show correlations of peaks that belong to other metabolites in the same pathway (positively or negatively) [8, 83] that are perturbed by biological processes. A combination of OPLS-DA (supervised multivariate data analysis) and STOCSY can provide information about discriminatory metabolite peaks and their

identification as potential biomarkers in metabolic pathways. In this method, a driver peak is used to identify and assign other correlated peaks into metabolites. Subset optimization by reference matching (**STORM**) [84] is a tool that uses multivariate data analysis for information recovery. It is capable of selecting subsets of spectra that have information about discrimination between the study groups. Cluster analysis statistical spectroscopy (**CLASSY**) is another method that combines qualitative metabolic profiling and quantitative changes in the biological matrix using a local-global correlation clustering Scheme [84]. It can visualize graphically in a high throughput way. Blaise and coworkers have reported an algorithm called statistical recoupling of variables (**SRV**) [85] and it utilizes covariance/correlation ratios of consecutive variables in the NMR spectra before significant testing. It can extract information about statistically significant metabolites in the perturbed biological systems. The recoupled-statistical total correlation spectroscopy (**RSTOCSY**) is another useful tool that combines SRV with STOCSY [86]. A further development with orthogonal signal filtering method has been applied in orthogonal filtered recoupled-STOCSY (**OR-STOCSY**) [86] to find a list of pairs of metabolites that experience correlated perturbations. Statistical homogeneous cluster spectroscopy (**SHOCSY**) [87] is another statistical approach that was developed to remove unwanted variation within a biological class in the metabolomics analysis. The algorithm in SHOCSY is shown to be capable of identifying a subset of spectra by categorizing into clusters of spectra that have similar spectral features (similar biochemical composition) improving the predictability in modeling.

4.3.3 Metabolite Identification

A ^1H NMR spectrum of a biological matrix is complex and consists of thousands of signals deriving from both endogenous and exogenous metabolites. Therefore, a successful combination of multivariate and univariate data analysis approaches can provide biomarker spectral features in the ^1H NMR spectra. The statistical correlation spectroscopy allows for recovering important information on marker metabolites by reducing the complexity in the spectra. In addition to the 1D NMR spectra, two or multidimensional NMR methods can provide valuable information for the identification of these features as metabolites. The application of a combination of 1D, 2D (^1H J-resolved, 2D DQF-COSY, TOCSY, HSQC, and HMBC), and multidimensional NMR spectroscopic methods [88–90] to obtain spectra on selected samples (pooled QC sample, for example) aids in identification of metabolites in biological samples. In addition, phenotype-specific metabolite information (abundant in one phenotype while minimally present or absent in another) may also be recovered if spectral data are available for representative phenotypic pooled samples. The metabolite peaks

in 1D NMR or 2D NMR can be used to search online databases such as Human Metabolome Database (HMDB), Biological Magnetic Resonance Data Bank (BMRB), and Birmingham Metabolite Database. The DQF-COSY (^1H - ^1H) and TOCSY (^1H - ^1H or ^{13}C - ^{13}C) experiments are designed to obtain homonuclear coupling information within a molecule. The TOCSY experiment provide information about the entire ^1H - ^1H spin system that are coupled in a molecule. The 2D HSQC method gives direct connectivity of atoms in the molecule (e.g., ^1H - ^{13}C or ^1H - ^{15}N) whereas HMBC can provide information about long-range bonds between atoms within the molecule. Another useful NMR experiment is a hybrid HSQC and TOCSY experiment, 2D ^1H - ^{13}C HSQC-TOCSY, for example. The 2D HSQC-TOCSY experiment can provide two types of information: direct ^1H - ^{13}C correlations and relayed correlations connecting each protonated carbon with all ^1H nuclei belonging to the same spin system. 2D INADEQUATE experiment provides ^{13}C - ^{13}C direct bond information enhancing the elucidation of carbon backbone structure of compounds. 3D HCCH-TOCSY and 3D HCCH-COSY experiments are some examples for multidimensional experiments that aid metabolite identification. Use of 2D, hybrid, and multidimensional NMR spectra (alone or combined) facilitate identification of known and unknown metabolites found in complex mixtures without the need of any physical separation of metabolites [90]. A number of software-based approaches for metabolite identification is available in literature for metabolite identification. Complex Mixture Analysis by NMR (**COLMAR**) [91] is a publicly available web server, which has a collection of tools deployed in web servers [89] for identification of metabolites using spectra collected as 1D (^1H , ^{13}C), 2D (TOCSY, HSQC), and multidimensional (HSQC-TOCSY) spectra and by incorporating searches in databases such as BMRB and the Carbon TOCSY NMR Metabolomics Database (TOCCATA) [92].

The ratio analysis nuclear magnetic resonance spectroscopy (**RANSY**) [93], a covariance-based method reported in literature, and uses ratios of peak height or integrals to identify all the peaks of a specific metabolite in NMR spectra of a biological samples. It is based on the principle that all the peak ratios of a molecule are fixed and proportional to the number of protons underlying each peak. The peak ratios are divided by covariance in this method to generate an individual spectrum of a metabolite.

Use of smart isotope tags in NMR spectroscopy is another method that enables identification of metabolites in complex biological samples. In this approach, ^{15}N -labeled ethanolamine or cholamine [94, 95] is used in a derivatization reaction to tag the ^{15}N label into carboxyl-containing metabolites. A ^{15}N and ^1H chemical shift library of metabolites has been developed using 2D

^1H - ^{15}N HSQC spectra. The smart tagging is performed on biological sample, ^1H - ^{15}N HSQC spectra are recorded on the samples, and ^{15}N and ^1H chemical shifts observed for cross peaks in the spectra are compared to those in the library.

1D ^{13}C spectroscopy (proton decoupled) has many advantages for a metabolomics study, including a large spectral dispersion, narrow singlets at natural abundance, and a direct measure of the backbone structures of metabolites compared to ^1H NMR spectra. However, it suffers from the very low sensitivity due to low natural abundance of ^{13}C isotope (~1.1%) combined with a decreased gyromagnetic ratio γ (one quarter of that of ^1H ; energy transition is proportional to γ^3 , and sensitivity is decreased by 64 times) unless the metabolomics experiments are performed by using ^{13}C enrichment experiments. Another disadvantage of this experiment is the inability to detect quaternary carbon atoms (carbon atoms with no protons attached). On the other hand, an INADEQUATE (incredible natural abundance double quantum transfer experiment) experiment can provide the complete carbon skeleton structure of a metabolite. This type of experiment is very helpful when ^{13}C isotope enrichment is performed prior to analysis. A semiautomated algorithm was developed by Clendinen and coworkers [96], INADEQUATE Network Analysis (INETA), where untargeted analysis can be performed and metabolites can be identified by using INADEQUATE spectral data and an *in silico* database constructed using BMRB database.

There are efforts to deconvolute the signals in NMR spectra into metabolites for the identification of metabolites. Another advantage of NMR spectroscopy is that the same spectrum can be used for both untargeted and semitargeted analysis. Since NMR spectroscopy is a quantitative technique, the use of an internal standard with known concentration allows for quantitation of metabolites [97]. This is a relative or semiquantitative method because there is no calibration curve employed to obtain absolute concentration. For example, Chenomx is a commercial package that contains a library of metabolites that has been modeled to account for chemical shift variation due to pH. It should also be noted that the spectra should be acquired to account for the total relaxation of ^1H atoms of the metabolites in the biological sample, in order to allow for identification and semiquantification of metabolites in the sample. It is possible to apply this method to profile a selected set of metabolites in a batch profiling mode. The experience of the analyst or prior knowledge of metabolites signals plays a critical role in this approach to a successful completion of identification and semiquantification using this method. On the other hand, Röhnisch et al. [98] have recently reported automated quantification algorithm (AQuA) for targeted quantification of metabolites that can account for signal overlaps of metabolites in complex

spectra in a high throughput fashion. It utilizes spectral data extracted from a library consisting of one standard calibration spectrum for each metabolite. The authors have tested this algorithm, compared with manual fitting using Chenomx software, and concluded that this approach produced accurate results with high efficiency. There are open source methods such as BATMAN and Bayesil that has been reported in literature for automation efforts for the identification and quantification of metabolites. BATMAN [99, 100] is an R- package that models NMR resonances on the basis of a user-controllable set of templates, each of which specifies the chemical shifts, J-couplings and relative peak intensities for a single metabolite. It produces outputs to include relative concentration for named metabolites together with associated Bayesian uncertainty estimates, and the fit of the remainder of the spectrum using wavelets. An improvement to this approach has been reported [101], where spectral ordering (sorting) is combined with peak deconvolution using BATMAN software. BAYESIL [102] (<http://bayesil.ca/>) performs several NMR processing steps and then matches spectra against a reference metabolite library using a probabilistic graphical model that approximates the most probable metabolic profile. It should be kept in mind that metabolites in signal overlapped regions of an ^1H NMR spectra make deconvolution and quantification difficult for complex biological samples. The use of 2D ^1H - ^{13}C HSQC is helpful since it enables signal dispersion. A method has been developed by using the fast maximum likelihood reconstruction (FMLR) algorithm [55] to identify and quantify metabolites using 2D ^1H - ^{13}C HSQC spectra acquired in about 15 min. The software package rNMR [103] provides a simple GUI-based method for visualizing, identifying, and quantifying metabolites across multiple one- or two-dimensional NMR spectra.

4.3.4 Workflows and Web-Based Analysis Programs

There are a number of open source workflows, and web-based programs available for NMR metabolomics analysis. In addition, standalone programs and web-based servers have been proposed in literature for metabolomics data analysis. Some of these methods are briefly described in this section with relevant literature and links. **Automics** [104] is an integrated metabolomics analysis platform that has functionality to each step in a NMR metabolomics workflow including processing spectra, nonsupervised and supervised multivariate data analysis and graphical visualization. **KIMBLE** (KNIME-based Integrated MetaBoLomics Environment [105]) is a KNIME workflow management system based NMR metabolomics workflow. It is a platform that is integrated with algorithms and software tools necessary for untargeted and targeted analysis. It is self-documenting and combines data, algorithms, and software in one file [105], allowing for reuse of the exact workflow parameters

in future. Even the workflow for a single project can be saved with all tools and parameters as a single virtual machine. These capabilities make KIMBLE a more robust platform for reproducibly using the NMR metabolomics workflow. **Metaboanalyst** is a web-based metabolomics platform [106–109] for uploading metabolomics data (both NMR and MS) and data analysis including pathway analysis. It has a number of modules with various preprocessing, statistical, multivariate analysis, identification, pathway analysis, and integrated omics data analysis tools. A standalone package for local installation is available on Metaboanalyst web page. Metabolomics Univariate and Multivariate Statistical Analysis (**MUMA**) [110] is an R-based pipeline that guides the user in the data analysis (univariate and multivariate data analysis) and interpretation. It also provides additional tools specifically designed to help the user in the interpretation of NMR data, such as STOCSY and RANSY. **MVAPACK** [111] is an open source package written in GNU Octave programming language to process metabolomics data from FIDs to modeling. It has a collection of tools including those needed for traditional NMR processing (apodization, zero-filling, Fourier transformation, manual and automatic phase correction, region of interest selection, and peak picking, integration, and referencing), data preprocessing (alignment, binning, normalization, and scaling), and multivariate data analysis (nonsupervised and supervised), and model validation tools. **NMRProcFlow** [112] (<https://nmrprocflow.org/>) is a web-based work flow to process 1D NMR spectra interactively with visualization tools. The workflow can be accessed through the web or can be installed locally as a virtual machine. It allows the user to import spectra and perform baseline and phase correction, chemical shift calibration, peak alignment, NMR binning (after removing unwanted regions including baseline noise), and merging of metadata. The workflow process can be saved and used again similarly for other datasets. A cloud-based computing framework for NMR metabolomics analysis using Hadoop streaming with Matlab [113] has been reported indicating the possibility of using the cloud-based computing power for NMR metabolomics applications. **Pathomx** [114] is another workflow-based tool developed using Python for processing, analyzing, visualizing, and exporting metabolomics data. The R-based package **speaq 2.0** [115] contains a workflow that includes peak alignment using CluPA algorithm, peak picking, data imputation, and peak table generation. The output that speaq produces is compatible for using other R-based statistical and multivariate data analysis packages. **Workflow4Metabolomics (W4M)** [116] is an open source and Galaxy-based web platform (<https://workflow4metabolomics.org/>) that has algorithms for data preprocessing, statistical analysis, and annotation. It is a virtual research environment based on high performance computing

environment (660 cores and 100 TB). It has modules for analyzing both MS and NMR data. Users can access the web version or the whole framework and computing tools can be installed locally as a virtual machine. **ASClS** (Automatic Statistical Identification in Complex Spectra) [117], is another R-based software package that comprises of a workflow with algorithms developed for automated analyzing of NMR spectra. The workflow includes raw NMR data import, baseline processing, peak alignment, NMR binning, metabolite identification and quantification. In addition, ASICS workflow includes tools for multivariate data analysis (PCA and OPLS-DA), identification of NMR bins or metabolites that discriminate phenotypes, and some other statistical analysis.

4.3.5 Tools for Metabotyping in Population-Based Association Studies

In molecular epidemiological studies, genome-wide association studies (GWAS) are conducted to discover the genetic associations with the disease risk in the populations. With the technical advances in spectroscopy, Metabolome Wide Association Studies (MWAS) can be conducted to reveal metabolite associations with the disease risk. The use of metabotyping in epidemiologic studies can facilitate the identification of disease-risk biomarker metabolites that can be used for developing diagnostic tools or tools for monitoring treatment interventions. The potential application of NMR metabotyping in MWAS studies was demonstrated in literature [52, 118, 119]. The 1D ^1H NMR spectra [118] or 1D projections (skyline or sum) of 2D ^1H J-resolved spectra (pJRES) [52] can also be utilized in the MWAS approach. An R-based MWASTools (<https://rdrr.io/bioc/MWASTools/>) [74] package was introduced by Rodriguez-Martinez and coworkers and it provides a complete pipeline to perform metabolome-wide association studies. It has many tools such as quality control analysis, data filtering, different MWAS association models, model validation, visualization, NMR metabolite identification using STOCSY algorithm, and biological interpretation using pathway mapping. Similarly, metabotype quantitative trait locus (mQTL) mapping has emerged as a new tool that uses quantitative variation of ^1H NMR spectra [12, 13] between biological samples. The 1D projections of J-resolved spectra (pJRES) have been used [52] in this approach as pJRES spectra have been shown to demonstrate enhanced peak dispersion, efficient attenuation of lipid resonances, better metabolite identification capacities, and analytical reproducibility.

4.3.6 Metabolic Pathway Analysis

Metabotyping studies can result in identification of certain metabolites, which have undergone changes in levels in the biological samples. Then the next step is to interpret these findings to answer the biological questions mechanistically by exploring the metabolic pathways, which have been perturbed due to the disease

phenotype, or treatment intervention, for example. Metabolic pathways are complex and inter-connected system of genes, transcripts, proteins, and metabolites. In order to query pathways, software tools are needed and a number of pathway analysis tools have been developed and curated by researchers and commercial vendors. A few useful software tools are discussed here. **Metaboanalyst** has tools for metabolic pathway analysis and modules for integrative analysis for metabolomics, metagenomics, and transcriptomics data. It has algorithms for over-representation analysis, metabolite set enrichment analysis, and pathway topology analysis. A built-in reference metabolome is included in the pathway analysis and the user has the option using own in-house reference metabolome if available. **Metscape** [120] is a tool for interactive exploration and visualization of experimental metabolomics and gene expression data in the context of metabolic networks. A prominent feature is the ability to enter gene expression data and to examine them in the context of metabolic networks. Metscape plug-in can be used using Cytoscape software (<https://cytoscape.org/>). Integrated pathway-level analysis (**IMPaLA**) [121], a web-based tool, allows for joint pathway analysis using transcriptomics or proteomics and metabolomics data. Overrepresentation or enrichment analysis can be performed with user-specified lists of metabolites and genes using over 3000 preannotated pathways from 11 databases. **MetaCore** (<https://clarivate.com/products/meta-core/>, Clarivate Analytics, PA, USA) is a commercially available web-based software, and it has many features including pathway enrichment analysis, network building, and integrated pathway analysis.

4.3.7 Metabolomics Data Repositories

It is important to make data and metadata including experimental data of metabolomics studies available to the research community for various reasons. It helps a wider community of researchers and bioinformaticians to obtain data for testing their software tools using publicly available data. It also facilitates reproducibility of conducting metabolomics studies by the availability of standardized experimental protocols. Therefore, establishing metabolomics data repositories and data coordinating centers are important and the Metabolomics Workbench [21] in the USA and MetaboLights in the Europe are two such data repositories. The Metabolomics Workbench serves as a national and international repository for metabolomics data and metadata and provides analysis tools and access to metabolite standards, protocols, tutorials, training, and more functionalities. On the other hand, Metabolights [122] is a database for metabolomics experiments, experimental data, and metadata for a variety of species, techniques and it covers metabolite structures and their reference spectra as well as information on their biological roles, locations, and concentrations.

5 Summary

In this chapter, we have described the major steps in a typical NMR metabolomics workflow with the focus on the available data analysis (computational and bioinformatics) tools and key important points to consider when designing an analysis method. The high-resolution NMR spectroscopy is a powerful analytical technique with higher reproducibility for metabolic phenotyping studies. Because of its robustness, NMR based metabotyping is particularly suitable for large-scale population-based metabolomics studies that involves thousands of samples and data acquisition using multiple batches of samples. Along with the advances in the NMR technology, the computational power (hardware, clusters, and cloud computing) has grown and the development of statistical and bioinformatics tools is also on the rise. Hence, the potential use of NMR spectroscopy for large-scale metabotyping studies enabling discovery of biomarkers of disease risk in populations in parallel to the genome wide association studies (GWAS) has also become increased. The methods described in this chapter can equally be applicable for metabolomics studies involving plants, microbes, and other model organisms.

References

1. Jacobsen NE (2007) NMR spectroscopy explained : simplified theory, applications and examples for organic chemistry and structural biology. Wiley, Hoboken, NJ
2. Nicholson JK, Connelly J, Lindon JC, Holmes E (2002) Metabonomics: a platform for studying drug toxicity and gene function. *Nat Rev Drug Discov* 1:153–161
3. Kaddurah-Daouk R, Kristal BS, Weinshilboum RM (2008) Metabolomics: a global biochemical approach to drug response and disease. *Annu Rev Pharmacol Toxicol* 48:653–683. <https://doi.org/10.1146/annurev.pharmtox.48.113006.094715>
4. Stewart DA, Dhungana S, Clark RF, Pathmasiri WW, McRitchie SL, Sumner SJ (2015) Omics technologies used in systems biology. In: Fry R (ed) Systems biology in toxicology and environmental health, 1st edn. Academic, Waltham, MA, pp 57–84
5. Sumner SCJ, Pathmasiri W, Carlson JE, McRitchie SL, Fennell TR (2018) Metabolomics. In: Smart R, Hodgeson E (eds) Molecular and biochemical toxicology. Wiley, Hoboken, NJ
6. Johnson CH, Gonzalez FJ (2012) Challenges and opportunities of metabolomics. *J Cell Physiol* 227(8):2975–2981. <https://doi.org/10.1002/jcp.24002>
7. Ryan D, Robards K (2005) Metabolomics: the greatest omics of them all? *Anal Chem* 24:285–293
8. Robinette SL, Lindon JC, Nicholson JK (2013) Statistical spectroscopic tools for biomarker discovery and systems medicine. *Anal Chem* 85(11):5297–5303. <https://doi.org/10.1021/ac4007254>
9. Bird SS, Sheldon DP, Gathungu RM, Vourous P, Kautz R, Matson WR, Kristal BS (2012) Structural characterization of plasma metabolites detected via LC-electrochemical coulometric array using LC-UV fractionation, MS, and NMR. *Anal Chem* 84 (22):9889–9898. <https://doi.org/10.1021/acs.analchem.8b02278u>
10. Sasaki K, Sagawa H, Suzuki M, Yamamoto H, Tomita M, Soga T, Ohashi Y (2018) A metabolomics platform by capillary electrophoresis coupled with a high-resolution mass spectrometry for plasma analysis. *Anal Chem* 91 (2):1295–1301. <https://doi.org/10.1021/acs.analchem.8b02994>
11. Bictash M, Ebbels TM, Chan Q, Loo RL, Yap IK, Brown IJ, de Iorio M, Daviglus ML, Holmes E, Stamler J, Nicholson JK, Elliott P

- (2010) Opening up the "Black Box": metabolic phenotyping and metabolome-wide association studies in epidemiology. *J Clin Epidemiol* 63(9):970–979. <https://doi.org/10.1016/j.jclinepi.2009.10.001>
12. Hedjazi L, Gauguier D, Zalloua PA, Nicholson JK, Dumas ME, Cazier JB (2015) mQTL: NMR: an integrated suite for genetic mapping of quantitative variations of (¹H) NMR-based metabolic profiles. *Anal Chem* 87(8):4377–4384. <https://doi.org/10.1021/acs.analchem.5b00145>
13. Cazier JB, Kaisaki PJ, Argoud K, Blaise BJ, Veselkov K, Ebbels TM, Tsang T, Wang Y, Bishoreau MT, Mitchell SC, Holmes EC, Lindon JC, Scott J, Nicholson JK, Dumas ME, Gauguier D (2012) Untargeted metabolome quantitative trait locus mapping associates variation in urine glycerate to mutant glycerate kinase. *J Proteome Res* 11(2):631–642. <https://doi.org/10.1021/pr200566t>
14. Gibson G, Gieger C, Geistlinger L, Altmaier E, Hrabé de Angelis M, Kronenberg F, Meitinger T, Mewes H-W, Wichmann HE, Weinberger KM, Adamski J, Illig T, Suhre K (2008) Genetics meets metabolomics: a genome-wide association study of metabolite profiles in human serum. *PLoS Genet* 4(11):e1000282. <https://doi.org/10.1371/journal.pgen.1000282>
15. Sekula P, Goeck ON, Quaye L, Barrios C, Levey AS, Romisch-Margl W, Menni C, Yet I, Gieger C, Inker LA, Adamski J, Gronwald W, Illig T, Dettmer K, Krumsiek J, Oefner PJ, Valdes AM, Meisinger C, Coresh J, Spector TD, Mohnay RP, Suhre K, Kastenmuller G, Kottgen A (2016) A metabolome-wide association study of kidney function and disease in the general population. *J Am Soc Nephrol* 27(4):1175–1188. <https://doi.org/10.1681/ASN.2014111099>
16. Kraus WE, Muoio DM, Stevens R, Craig D, Bain JR, Grass E, Haynes C, Kwee L, Qin X, Slentz DH, Krupp D, Muehlbauer M, Hauser ER, Gregory SG, Newgard CB, Shah SH (2015) Metabolomic quantitative trait loci (mQTL) mapping implicates the ubiquitin proteasome system in cardiovascular disease pathogenesis. *PLoS Genet* 11(11):e1005553. <https://doi.org/10.1371/journal.pgen.1005553>
17. MRC-NIHR National Phenome Center. <https://www.imperial.ac.uk/phenome-centre>. Accessed February 2019
18. Clinical Phenotyping Centre. <http://www.imperial.ac.uk/clinical-phenotyping-centre/>. Accessed February 2019
19. Phenome Center Birmingham. <https://www.birmingham.ac.uk/research/activity/phenome-centre/index.aspx>. Accessed February 2019
20. Australian National Phenome Center. <https://www.wahtn.org/enabling-platforms/australian-national-phenome-centre/>. Accessed February 2019
21. Sud M, Fahy E, Cotter D, Azam K, Vadivelu I, Burant C, Edison A, Fiehn O, Higashi R, Nair KS, Sumner S, Subramaniam S (2016) Metabolomics Workbench: an international repository for metabolomics data and metadata, metabolite standards, protocols, tutorials and training, and analysis tools. *Nucleic Acids Res* 44(D1):D463–D470. <https://doi.org/10.1093/nar/gkv1042>
22. Markley JL, Bruschweiler R, Edison AS, Eghbalnia HR, Powers R, Raftery D, Wishart DS (2017) The future of NMR-based metabolomics. *Curr Opin Biotechnol* 43:34–40. <https://doi.org/10.1016/j.copbio.2016.08.001>
23. Ludwig C, Easton JM, Lodi A, Tiziani S, Manzoor SE, Southam AD, Byrne JJ, Bishop LM, He S, Arvanitis TN, Günther UL, Viant MR (2011) Birmingham metabolite library: a publicly accessible database of 1-D ¹H and 2-D ¹H J-resolved NMR spectra of authentic metabolite standards (BML-NMR). *Metabolomics* 8(1):8–18. <https://doi.org/10.1007/s11306-011-0347-7>
24. Wishart DS (2008) Quantitative metabolomics using NMR. *TrAC Trends Anal Chem* 27(3):228–237. <https://doi.org/10.1016/j.trac.2007.12.001>
25. Wishart DS, Tzur D, Knox C, Eisner R, Guo AC, Young N, Cheng D, Jewell K, Arndt D, Sawhney S, Fung C, Nikolai L, Lewis M, Coutouly MA, Forsythe I, Tang P, Shrivastava S, Jeroncic K, Stothard P, Amegbey G, Block D, Hau DD, Wagner J, Miniaci J, Clements M, Gebremedhin M, Guo N, Zhang Y, Duggan GE, Macinnis GD, Weljie AM, Dowlatabadi R, Bamforth F, Clive D, Greiner R, Li L, Marrie T, Sykes BD, Vogel HJ, Querengesser L (2007) HMDB: the human metabolome database. *Nucleic Acids Res* 35(Database issue):D521–D526. <https://doi.org/10.1093/nar/gkl923>
26. Kuhn S, Schlorer NE (2015) Facilitating quality control for spectra assignments of small organic molecules: nmrshiftdb2—a free in-house NMR database with integrated LIMS for academic service laboratories. *Magn Reson Chem* 53(8):582–589. <https://doi.org/10.1002/mrc.4263>

27. Laine JE, Bailey KA, Olshan AF, Smeester L, Drobna Z, Styblo M, Douillet C, Garcia-Vargas G, Rubio-Andrade M, Pathmasiri W, McRitchie S, Sumner SJ, Fry RC (2017) Neonatal metabolomic profiles related to prenatal arsenic exposure. *Environ Sci Technol* 51(1):625–633. <https://doi.org/10.1021/acs.est.6b04374>
28. Szabo DT, Pathmasiri W, Sumner S, Birnbaum LS (2017) Serum metabolomic profiles in neonatal mice following oral brominated flame retardant exposures to hexabromocyclododecane (HBCD) alpha, gamma, and commercial mixture. *Environ Health Perspect* 125(4):651–659. <https://doi.org/10.1289/EHP242>
29. Fan TW, Lane AN (2011) NMR-based stable isotope resolved metabolomics in systems biochemistry. *J Biomol NMR* 49(3–4):267–280. <https://doi.org/10.1007/s10858-011-9484-6>
30. Creek DJ, Chokkathukalam A, Jankevics A, Burgess KE, Breitling R, Barrett MP (2012) Stable isotope-assisted metabolomics for network-wide metabolic pathway elucidation. *Anal Chem* 84(20):8442–8447. <https://doi.org/10.1021/ac3018795>
31. Zamboni N, Fendt S-M, Rühl M, Sauer U (2009) ¹³C-based metabolic flux analysis. *Nat Protoc* 4(6):878–892. <https://doi.org/10.1038/nprot.2009.58>
32. Beckonert O, Keun HC, Ebbels TMD, Bundy J, Holmes E, Lindon JC, Nicholson JK (2007) Metabolic profiling, metabolomic and metabonomic procedures for NMR spectroscopy of urine, plasma, serum and tissue extracts. *Nat Protoc* 2(11):2692–2703. <https://doi.org/10.1038/nprot.2007.376>
33. Dumas M-E, Maibaum EC, Teague C, Ueshima H, Zhou B, Lindon JC, Nicholson JK, Stamler J, Elliott P, Queenie HE (2006) Assessment of analytical reproducibility of ¹H NMR spectroscopy based metabonomics for large-scale epidemiological research: the INTERMAP study. *Anal Chem* 78:2199–2208
34. Karaman I, Ferreira DL, Boulange CL, Kaluarachchi MR, Herrington D, Dona AC, Castagne R, Moayyeri A, Lehne B, Loh M, de Vries PS, Dehghan A, Franco OH, Hofman A, Evangelou E, Tzoulaki I, Elliott P, Lindon JC, Ebbels TM (2016) Workflow for integrated processing of multi-cohort untargeted (¹H) NMR metabolomics data in large-scale metabolic epidemiology. *J Proteome Res* 15(12):4188–4194. <https://doi.org/10.1021/acs.jproteome.6b00125>
35. Bornet A, Maucourt M, Deborde C, Jacob D, Milani J, Vuichoud B, Ji X, Dumez JN, Moing A, Bodenhausen G, Jannin S, Giraudieu P (2016) Highly repeatable dissolution dynamic nuclear polarization for heteronuclear NMR metabolomics. *Anal Chem* 88(12):6179–6183. <https://doi.org/10.1021/acs.analchem.6b01094>
36. Dumez JN, Milani J, Vuichoud B, Bornet A, Lalande-Martin J, Tea I, Yon M, Maucourt M, Deborde C, Moing A, Frydman L, Bodenhausen G, Jannin S, Giraudieu P (2015) Hyperpolarized NMR of plant and cancer cell extracts at natural abundance. *Analyst* 140(17):5860–5863. <https://doi.org/10.1039/c5an01203a>
37. Johnson CH, Patterson AD, Idle JR, Gonzalez FJ (2012) Xenobiotic metabolomics: major impact on the metabolome. *Annu Rev Pharmacol Toxicol* 52:37–56. <https://doi.org/10.1146/annurev-pharmtox-010611-134748>
38. Blaise BJ, Correia G, Tin A, Young JH, Vergnaud AC, Lewis M, Pearce JT, Elliott P, Nicholson JK, Holmes E, Ebbels TM (2016) Power analysis and sample size determination in metabolic phenotyping. *Anal Chem* 88(10):5179–5188. <https://doi.org/10.1021/acs.analchem.6b00188>
39. Barton RH, Waterman D, Bonner FW, Holmes E, Clarke R, Procardis C, Nicholson JK, Lindon JC (2010) The influence of EDTA and citrate anticoagulant addition to human plasma on information recovery from NMR-based metabolic profiling studies. *Mol BioSyst* 6(1):215–224. <https://doi.org/10.1039/b907021d>
40. Bernini P, Bertini I, Luchinat C, Nincheri P, Staderini S, Turano P (2011) Standard operating procedures for pre-analytical handling of blood and urine for metabolomic studies and biobanks. *J Biomol NMR* 49(3–4):231–243. <https://doi.org/10.1007/s10858-011-9489-1>
41. Haid M, Muschett C, Wahl S, Romisch-Margl-W, Prehn C, Moller G, Adamski J (2018) Long-term stability of human plasma metabolites during storage at –80 degrees C. *J Proteome Res* 17(1):203–211. <https://doi.org/10.1021/acs.jproteome.7b00518>
42. Dane AD, Hendriks MM, Reijmers TH, Harms AC, Troost J, Vreeken RJ, Boomsma DI, van Duijn CM, Slagboom EP, Hankemeier T (2014) Integrating metabolomics profiling measurements across multiple biobanks. *Anal Chem* 86(9):4110–4114. <https://doi.org/10.1021/ac404191a>

43. Dona AC, Jimenez B, Schafer H, Humpfer E, Spraul M, Lewis MR, Pearce JT, Holmes E, Lindon JC, Nicholson JK (2014) Precision high-throughput proton NMR spectroscopy of human urine, serum, and plasma for large-scale metabolic phenotyping. *Anal Chem* 86 (19):9887–9894. <https://doi.org/10.1021/ac5025039>
44. Beckonert O, Coen M, Keun HC, Wang Y, Ebbels TMD, Holmes E, Lindon JC, Nicholson JK (2010) High-resolution magic-angle-spinning NMR spectroscopy for metabolic profiling of intact tissues. *Nat Protoc* 5 (6):1019–1032. <https://doi.org/10.1038/nprot.2010.45>
45. Wong A, Jimenez B, Li X, Holmes E, Nicholson JK, Lindon JC, Sakellariou D (2012) Evaluation of high resolution magic-angle coil spinning NMR spectroscopy for metabolic profiling of nanoliter tissue biopsies. *Anal Chem* 84(8):3843–3848. <https://doi.org/10.1021/ac300153k>
46. Gillies RJ, Morse DL (2005) In vivo magnetic resonance spectroscopy in cancer. *Annu Rev Biomed Eng* 7:287–326. <https://doi.org/10.1146/annurev.bioeng.7.060804.100411>
47. Stewart DA, Winnike JH, McRitchie SL, Clark RF, Pathmasiri WW, Sumner SJ (2016) Metabolomics analysis of hormone-responsive and triple-negative breast cancer cell responses to paclitaxel identify key metabolic differences. *J Proteome Res* 15 (9):3225–3240. <https://doi.org/10.1021/acs.jproteome.6b00430>
48. Livanos AE, Greiner TU, Vangay P, Pathmasiri W, Stewart D, McRitchie S, Li H, Chung J, Sohn J, Kim S, Gao Z, Barber C, Kim J, Ng S, Rogers AB, Sumner S, Zhang XS, Cadwell K, Knights D, Alekseyenko A, Backhed F, Blaser MJ (2016) Antibiotic-mediated gut microbiome perturbation accelerates development of type 1 diabetes in mice. *Nat Microbiol* 1(11):16140. <https://doi.org/10.1038/nmicrobiol.2016.140>
49. Loeser RF, Pathmasiri W, Sumner SJ, McRitchie S, Beavers D, Saxena P, Nicklas BJ, Jordan J, Guermazi A, Hunter DJ, Messier SP (2016) Association of urinary metabolites with radiographic progression of knee osteoarthritis in overweight and obese adults: an exploratory study. *Osteoarthr Cartil* 24 (8):1479–1486. <https://doi.org/10.1016/j.joca.2016.03.011>
50. Psychogios N, Hau DD, Peng J, Guo AC, Mandal R, Bouatra S, Sinelnikov I, Krishnamurthy R, Eisner R, Gautam B, Young N, Xia J, Knox C, Dong E, Huang P, Hollander Z, Pedersen TL, Smith SR, Bamforth F, Greiner R, McManus B, Newman JW, Goodfriend T, Wishart DS (2011) The human serum metabolome. *PLoS One* 6 (2):e16957. <https://doi.org/10.1371/journal.pone.0016957>
51. Smilowitz JT, O'Sullivan A, Barile D, German JB, Lonnerdal B, Slupsky CM (2013) The human milk metabolome reveals diverse oligosaccharide profiles. *J Nutr* 143 (11):1709–1718. <https://doi.org/10.3945/jn.113.178772>
52. Rodriguez-Martinez A, Posma JM, Ayala R, Harvey N, Jimenez B, Neves AL, Lindon JC, Sonomura K, Sato TA, Matsuda F, Zalloua P, Gauguier D, Nicholson JK, Dumas ME (2017) J-resolved ¹H NMR 1D-projections for large-scale metabolic phenotyping studies: application to blood plasma analysis. *Anal Chem* 89(21):11405–11412. <https://doi.org/10.1021/acs.analchem.7b02374>
53. Fonville JM, Maher AD, Coen M, Holmes E, Lindon oC, Nicholson JK (2010) Evaluation of full-resolution J-resolved ¹H NMR projections of biofluids for metabonomics information retrieval and biomarker identification. *Anal Chem* 82:1811–1821
54. Liu M, Tang H, Nicholson JK, Lindon JC (2002) Use of ¹H NMR-determined diffusion coefficients to characterize lipoprotein fractions in human blood plasma. *Magn Reson Chem* 40(13):S83–S88. <https://doi.org/10.1002/mrc.1121>
55. Chylla RA, Hu K, Ellinger JJ, Markley JL (2011) Deconvolution of two-dimensional NMR spectra by fast maximum likelihood reconstruction: application to quantitative metabolomics. *Anal Chem* 83 (12):4871–4880. <https://doi.org/10.1021/ac200536b>
56. Phinney KW, Ballihaut G, Bedner M, Benford BS, Camara JE, Christopher SJ, Davis WC, Dodder NG, Eppe G, Lang BE, Long SE, Lowenthal MS, McGaw EA, Murphy KE, Nelson BC, Prendergast JL, Reiner JL, Rimmer CA, Sander LC, Schantz MM, Sharpless KE, Sniegowski LT, Tai SS, Thomas JB, Vetter TW, Welch MJ, Wise SA, Wood LJ, Guthrie WF, Hagwood CR, Leigh SD, Yen JH, Zhang NF, Chaudhary-Webb M, Chen H, Fazili Z, LaVoie DJ, McCoy LF, Momin SS, Paladugula N, Pendergrast EC, Pfeiffer CM, Powers CD, Rabinowitz D, Rybak ME, Schleicher RL, Toombs BM, Xu M, Zhang M, Castle AL (2013) Development of a Standard Reference Material for metabolomics research. *Anal Chem* 85

- (24):11732–11738. <https://doi.org/10.1021/ac402689t>
57. Dunn WB, Broadhurst D, Begley P, Zelena E, Francis-McIntyre S, Anderson N, Brown M, Knowles JD, Halsall A, Haselden JN, Nicholls AW, Wilson ID, Kell DB, Goodacre R, Human Serum Metabolome C (2011) Procedures for large-scale metabolic profiling of serum and plasma using gas chromatography and liquid chromatography coupled to mass spectrometry. *Nat Protoc* 6(7):1060–1083. <https://doi.org/10.1038/nprot.2011.335>
58. Gika HG, A G, Theodoridis EM, Wilson ID (2012) A QC approach to the determination of day-to-day reproducibility and robustness of LC-MS methods for global metabolite profiling in metabolomics/metabolomics. *Bioanalysis* 4(18):2239–2247
59. Townsend MK, Clish CB, Kraft P, Wu C, Souza AL, Deik AA, Tworoger SS, Wolpin BM (2013) Reproducibility of metabolomic profiles among men and women in 2 large cohort studies. *Clin Chem* 59(11):1657–1667. <https://doi.org/10.1373/clinchem.2012.199133>
60. Masson P, Spagou K, Nicholson JK, Want EJ (2011) Technical and biological variation in UPLC-MS-based untargeted metabolic profiling of liver extracts: application in an experimental toxicity study on galactosamine. *Anal Chem* 83(3):1116–1123. <https://doi.org/10.1021/ac103011b>
61. Chan EC, Pasikanti KK, Nicholson JK (2011) Global urinary metabolic profiling procedures using gas chromatography-mass spectrometry. *Nat Protoc* 6(10):1483–1499. <https://doi.org/10.1038/nprot.2011.375>
62. Veselkov KA, Lindon JC, Ebbels TMD, Crockford D, Volynkin VV, Holmes E, Davies DB, Nicholson JK (2009) Recursive segment-wise peak alignment of biological ^1H NMR spectra for improved metabolic biomarker recovery. *Anal Chem* 81:56–66
63. Savorani F, Tomasi G, Engelsen SB (2010) icoshift: a versatile tool for the rapid alignment of 1D NMR spectra. *J Magn Reson* 202(2):190–202. <https://doi.org/10.1016/j.jmr.2009.11.012>
64. Vu TN, Valkenborg D, Smets K, Verwaest KA, Dommissie R, Lemière F, Verschoren A, Goethals B, Laukens K (2011) An integrated workflow for robust alignment and simplified quantitative analysis of NMR spectrometry data. *BMC Bioinformatics* 12:405
65. Larsen FH, van den Berg F, Engelsen SB (2006) An exploratory chemometric study of ^1H NMR spectra of table wines. *J Chemom* 20(5):198–208. <https://doi.org/10.1002/cem.991>
66. Alonso A, Rodriguez MA, Vinaixa M, Tortosa R, Correig X, Julia A, Marsal S (2014) Focus: a robust workflow for one-dimensional NMR spectral analysis. *Anal Chem* 86(2):1160–1169. <https://doi.org/10.1021/ac403110u>
67. RBNMR. <https://www.mathworks.com/matlabcentral/fileexchange/40332-rbnmr>. Accessed February 2019
68. Krishnamurthy K (2013) CRAFT (complete reduction to amplitude frequency table)—robust and time-efficient Bayesian approach for quantitative mixture analysis by NMR. *Magn Reson Chem* 51(12):821–829. <https://doi.org/10.1002/mrc.4022>
69. Intellegent bucketing: Part 1. <https://www.acdlabs.com/download/publ/2004/enc04/intelbucket.pdf>. Accessed February 2019
70. Intellegent bucketing: Part 2. <https://www.acdlabs.com/download/publ/2004/intelbucket2.pdf>. Accessed February 2019
71. Davis RA, Charlton AJ, Godward J, Jones SA, Harrison M, Wilson JC (2007) Adaptive binning: an improved binning method for metabolomics data using the undecimated wavelet transform. *Chemom Intell Lab Syst* 85(1):144–154. <https://doi.org/10.1016/j.chemolab.2006.08.014>
72. Sousa SAA, Magalhães A, Ferreira MMC (2013) Optimized bucketing for NMR spectra: three case studies. *Chemom Intell Lab Syst* 122:93–102. <https://doi.org/10.1016/j.chemolab.2013.01.006>
73. Rodriguez-Martinez A, Ayala R, Posma JM, Harvey N, Jimenez B, Sonomura K, Sato TA, Matsuda F, Zalloua P, Gauguier D, Nicholson JK, Dumas ME (2018) pJRES Binning Algorithm (JBA): a new method to facilitate the recovery of metabolic information from pJRES ^1H NMR spectra. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/bty837>
74. Rodriguez-Martinez A, Posma JM, Ayala R, Neves AL, Anwar M, Petretto E, Emanueli C, Gauguier D, Nicholson JK, Dumas ME (2018) MWASTools: an R/bioconductor package for metabolome-wide association studies. *Bioinformatics* 34(5):890–892. <https://doi.org/10.1093/bioinformatics/btx477>
75. Dieterle F, Ross A, Schlotterbeck G, Senn H (2006) Probabilistic quotient normalization as robust method to account for dilution of complex biological mixtures. Application in

- 1H NMR metabonomics. *Anal Chem* 78:4281–4290
76. van den Berg RA, Hoefsloot HC, Westerhuis JA, Smilde AK, van der Werf MJ (2006) Centering, scaling, and transformations: improving the biological information content of metabolomics data. *BMC Genomics* 7:142. <https://doi.org/10.1186/1471-2164-7-142>
77. Keun HC, Ebbels TMD, Antti H, Bolland ME, Beckonert O, Holmes E, Lindon JC, Nicholson JK (2003) Improved analysis of multivariate data by variable stability scaling: application to NMR-based metabolic profiling. *Anal Chim Acta* 490 (1–2):265–276. [https://doi.org/10.1016/s0003-2670\(03\)00094-1](https://doi.org/10.1016/s0003-2670(03)00094-1)
78. Eriksson L, Byrne T, Johansson E, Trygg J, Vikström C (2013) Multi-and megavariate data analysis basic principles and applications. Umetrics Academy, Umeå
79. Johan T, Holmes E, Lundstedt T (2007) Chemometrics in metabolomics. *J Proteome Res* 6:469–479
80. Bylesjö M, Rantalainen M, Cloarec O, Nicholson JK, Holmes E, Trygg J (2006) OPLS discriminant analysis: combining the strengths of PLS-DA and SIMCA classification. *J Chemom* 20(8–10):341–351. <https://doi.org/10.1002/cem.1006>
81. Bylesjö M, Rantalainen M, Nicholson JK, Holmes E, Trygg J (2008) K-OPLS package: kernel-based orthogonal projections to latent structures for prediction and interpretation in feature space. *BMC Bioinformatics* 9:106. <https://doi.org/10.1186/1471-2105-9-106>
82. Posma JM, Garcia-Perez I, Ebbels TMD, Lindon JC, Stamler J, Elliott P, Holmes E, Nicholson JK (2018) Optimized phenotypic biomarker discovery and confounder elimination via covariate-adjusted projection to latent structures from metabolic spectroscopy data. *J Proteome Res* 17(4):1586–1595. <https://doi.org/10.1021/acs.jproteome.7b00879>
83. Cloarec O, Dumas ME, Craig A, Barton RH, Trygg J, Hudson J, Blancher C, Gauguier D, Lindon JC, Holmes E, Nicholson J (2005) Statistical total correlation spectroscopy: an exploratory approach for latent biomarker identification from metabolic 1H NMR data sets. *Anal Chem* 77(5):1282–1289. <https://doi.org/10.1021/ac048630x>
84. Posma JM, Garcia-Perez I, De Iorio M, Lindon JC, Elliott P, Holmes E, Ebbels TM, Nicholson JK (2012) Subset optimization by reference matching (STORM): an optimized statistical approach for recovery of metabolic biomarker structural information from 1H NMR spectra of biofluids. *Anal Chem* 84 (24):10694–10701. <https://doi.org/10.1021/ac302360v>
85. Blaise BJ, Shintu L, Bnd E, Emsley L, Dumas M-E, Toulhoat P (2009) Statistical recoupling prior to significance testing in nuclear magnetic resonance based metabolomics. *Anal Chem* 81:6242–6251
86. Blaise BJ, Navratil V, Emsley L, Toulhoat P (2011) Orthogonal filtered recoupled-STOCSY to extract metabolic networks associated with minor perturbations from NMR spectroscopy. *J Proteome Res* 10 (9):4342–4348. <https://doi.org/10.1021/pr200489n>
87. Zou X, Holmes E, Nicholson JK, Loo RL (2014) Statistical HOmogeneous Cluster SpectroscopY (SHOCSY): an optimized statistical approach for clustering of (1)H NMR spectral data to reduce interference and enhance robust biomarkers selection. *Anal Chem* 86(11):5308–5315. <https://doi.org/10.1021/ac500161k>
88. Dona AC, Kyriakides M, Scott F, Shephard EA, Varshavi D, Veselkov K, Everett JR (2016) A guide to the identification of metabolites in NMR-based metabolomics/metabolomics experiments. *Comput Struct Biotechnol J* 14:135–153. <https://doi.org/10.1016/j.csbj.2016.02.005>
89. Bingol K (2018) Recent advances in targeted and untargeted metabolomics by NMR and MS/NMR methods. *High Throughput* 7(2). <https://doi.org/10.3390/ht7020009>
90. Bingol K, Bruschweiler R (2017) Knowns and unknowns in metabolomics identified by multidimensional NMR and hybrid MS/NMR methods. *Curr Opin Biotechnol* 43:17–24. <https://doi.org/10.1016/j.copbio.2016.07.006>
91. Robinette SL, Zhang F, Brüschweiler-Li L, Brüschweiler R (2008) R web server based complex mixture analysis by NMR. *Anal Chem* 80:3606–3611
92. Bingol K, Zhang F, Bruschweiler-Li L, Bruschweiler R (2012) TOCCATA: a customized carbon total correlation spectroscopy NMR metabolomics database. *Anal Chem* 84(21):9395–9401. <https://doi.org/10.1021/ac302197e>
93. Wei S, Zhang J, Liu L, Ye T, Gowda GA, Tayyari F, Raftery D (2011) Ratio analysis nuclear magnetic resonance spectroscopy for selective metabolite identification in complex samples. *Anal Chem* 83(20):7616–7623. <https://doi.org/10.1021/ac201625f>

94. Ye T, Mo H, Shanaiah N, Nagana Gowda GA, Zhang S, Raftery D (2009) Chemoselective ¹⁵N tag for sensitive and high-resolution nuclear magnetic resonance profiling of the carboxyl-containing metabolome. *Anal Chem* 81:4882–4888
95. Tayyari F, Gowda GA, Gu H, Raftery D (2013) ¹⁵N-cholamine—a smart isotope tag for combining NMR- and MS-based metabolic profiling. *Anal Chem* 85(18):8715–8721. <https://doi.org/10.1021/ac401712a>
96. Clendinen CS, Pasquel C, Ajredini R, Edison AS (2015) ^{(13)C} NMR metabolomics: INADEQUATE network analysis. *Anal Chem* 87(11):5698–5706. <https://doi.org/10.1021/acs.analchem.5b00867>
97. Weljie AM, Newton J, Mercier P, Carlson E, Slupsky CM (2006) Targeted profiling: quantitative analysis of ^{1H} NMR metabolomics data. *Anal Chem* 78(13):4430–4442. <https://doi.org/10.1021/ac060209g>
98. Rohnisch HE, Eriksson J, Mullner E, Agback P, Sandstrom C, Moazzami AA (2018) AQuA: an automated quantification algorithm for high-throughput NMR-based metabolomics and its application in human plasma. *Anal Chem* 90(3):2095–2102. <https://doi.org/10.1021/acs.analchem.7b04324>
99. Hao J, Astle W, De Iorio M, Ebbels TM (2012) BATMAN—an R package for the automated quantification of metabolites from nuclear magnetic resonance spectra using a Bayesian model. *Bioinformatics* 28 (15):2088–2090. <https://doi.org/10.1093/bioinformatics/bts308>
100. Hao J, Liebeke M, Astle W, De Iorio M, Bundy JG, Ebbels TM (2014) Bayesian deconvolution and quantification of metabolites in complex 1D NMR spectra using BATMAN. *Nat Protoc* 9(6):1416–1427. <https://doi.org/10.1038/nprot.2014.090>
101. Liebeke M, Hao J, Ebbels TM, Bundy JG (2013) Combining spectral ordering with peak fitting for one-dimensional NMR quantitative metabolomics. *Anal Chem* 85 (9):4605–4612. <https://doi.org/10.1021/ac400237w>
102. Ravanbakhsh S, Liu P, Bjorndahl TC, Mandal R, Grant JR, Wilson M, Eisner R, Sinelnikov I, Hu X, Luchinat C, Greiner R, Wishart DS (2015) Accurate, fully-automated NMR spectral profiling for metabolomics. *PLoS One* 10(5):e0124219. <https://doi.org/10.1371/journal.pone.0124219>
103. Lewis IA, Schommer SC, Markley JL (2009) rNMR: open source software for identifying and quantifying metabolites in NMR spectra. *Magn Reson Chem* 47(Suppl 1):S123–S126. <https://doi.org/10.1002/mrc.2526>
104. Wang T, Shao K, Chu Q, Ren Y, Mu Y, Qu L, He J, Jin C, Xia B (2009) Automics: an integrated platform for NMR-based metabolomics spectral processing and data analysis. *BMC Bioinformatics* 10:83. <https://doi.org/10.1186/1471-2105-10-83>
105. Verhoeven A, Giera M, Mayboroda OA (2018) KIMBLE: a versatile visual NMR metabolomics workbench in KNIME. *Anal Chim Acta* 1044:66–76
106. Chong J, Soufan O, Li C, Caraus I, Li S, Bourque G, Wishart DS, Xia J (2018) MetaboAnalyst 4.0: towards more transparent and integrative metabolomics analysis. *Nucleic Acids Res* 46(W1):W486–W494. <https://doi.org/10.1093/nar/gky310>
107. Xia J, Psychogios N, Young N, Wishart DS (2009) MetaboAnalyst: a web server for metabolomic data analysis and interpretation. *Nucleic Acids Res* 37(Web Server issue): W652–W660. <https://doi.org/10.1093/nar/gkp356>
108. Xia J, Wishart DS (2011) Web-based inference of biological patterns, functions and pathways from metabolomic data using MetaboAnalyst. *Nat Protoc* 6(6):743–760. <https://doi.org/10.1038/nprot.2011.319>
109. Metaboanalyst. <https://www.metaboanalyst.ca/MetaboAnalyst/faces/home.xhtml>. Accessed February 2019
110. Gaude E, Chignola F, Spiliotopoulos D, Spitaleri A, Ghitti M, Garcia-Manteiga M, Mari S, Musco G (2013) mumu, An R package for metabolomics univariate and multivariate statistical analysis. *Curr Metabolomics* 1 (2):180–189. <https://doi.org/10.2174/2213235x11301020005>
111. Worley B, Powers R (2014) MVAPACK: a complete data handling package for NMR metabolomics. *ACS Chem Biol* 9 (5):1138–1144. <https://doi.org/10.1021/cb4008937>
112. Jacob D, Deborde C, Lefebvre M, Maucourt M, Moing A (2017) NMRProcFlow: a graphical and interactive tool dedicated to 1D spectra processing for NMR-based metabolomics. *Metabolomics* 13(4):36. <https://doi.org/10.1007/s11306-017-1178-y>
113. Gunaratna K, Anderson P, Ranabahu A, Sheth A (2010) A study in hadoop streaming with matlab for NMR data processing. Paper presented at the 2010 IEEE second international conference on cloud computing technology and science.

114. Fitzpatrick MA, McGrath CM, Young SP (2014) Pathomx: an interactive workflow-based tool for the analysis of metabolomic data. *BMC Bioinformatics* 15(1):396
115. Beirnaert C, Meysman P, Vu TN, Hermans N, Apers S, Pieters L, Covaci A, Laukens K (2018) speaq 2.0: a complete workflow for high-throughput 1D NMR spectra processing and quantification. *PLoS Comput Biol* 14(3):e1006018. <https://doi.org/10.1371/journal.pcbi.1006018>
116. Giacomoni F, Le Corguille G, Monsoor M, Landi M, Pericard P, Petera M, Duperier C, Tremblay-Franco M, Martin JF, Jacob D, Goulitquer S, Thevenot EA, Caron C (2015) Workflow4Metabolomics: a collaborative research infrastructure for computational metabolomics. *Bioinformatics* 31(9):1493–1495. <https://doi.org/10.1093/bioinformatics/btu813>
117. Lefort G, Liaubet L, Canlet C, Tardivel P, Pere MC, Quesnel H, Paris A, Iannuccelli N, Vialaneix N, Servien R (2019) ASICS: an R package for a whole analysis workflow of 1D ¹H NMR spectra. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/btz248>
118. Chadeau-Hyam M, Ebbels TMD, Brown IJ, Chan Q, Stamler J, Huang CC, Daviglus ML, Ueshima H, Zhao L, Holmes E, Nicholson JK, Elliott P, Iorio MD (2010) Metabolic profiling and the metabolome-wide association study: significance level for biomarker identification. *J Proteome Res* 9:4620–4627
119. Castagne R, Boulange CL, Karaman I, Campanella G, Santos Ferreira DL, Kaluarachchi MR, Lehne B, Moayyeri A, Lewis MR, Spagou K, Dona AC, Evangelos V, Tracy R, Greenland P, Lindon JC, Herrington D, Ebbels TMD, Elliott P, Tzoulaki I, Chadeau-Hyam M (2017) Improving visualization and interpretation of metabolome-wide association studies: an application in a population-based cohort using untargeted ¹H NMR metabolic profiling. *J Proteome Res* 16(10):3623–3633. <https://doi.org/10.1021/acs.jproteome.7b00344>
120. Karnovsky A, Weymouth T, Hull T, Tarcea VG, Scardoni G, Laudanna C, Sartor MA, Stringer KA, Jagadish HV, Burant C, Athey B, Omenn GS (2012) Metscape 2 bioinformatics tool for the analysis and visualization of metabolomics and gene expression data. *Bioinformatics* 28(3):373–380. <https://doi.org/10.1093/bioinformatics/btr661>
121. Kamburov A, Cavill R, Ebbels TMD, Herwig R, Keun HC (2011) Integrated pathway-level analysis of transcriptomics and metabolomics data with IMPaLA. *Bioinformatics* 27(20):2917–2918. <https://doi.org/10.1093/bioinformatics/btr499>
122. Haug K, Salek RM, Conesa P, Hastings J, de Matos P, Rijnbeek M, Mahendraker T, Williams M, Neumann S, Rocca-Serra P, Maguire E, Gonzalez-Beltran A, Sansone SA, Griffin JL, Steinbeck C (2013) MetaboLights—an open-access general-purpose repository for metabolomics studies and associated meta-data. *Nucleic Acids Res* 41(Database issue):D781–D786. <https://doi.org/10.1093/nar/gks1004>



Chapter 6

Key Concepts Surrounding Studies of Stable Isotope-Resolved Metabolomics

Stephen F. Previs and Daniel P. Downes

Abstract

“Omics”-based analyses are widely used in numerous areas of research, advances in instrumentation (both hardware and software) allow investigators to collect a wealth of data and therein characterize metabolic systems. Although analyses generally examine differences in absolute or relative (fold-) changes in concentrations, the ability to extract mechanistic insight would benefit from the use of isotopic tracers. Herein, we discuss important concepts that should be considered when stable isotope tracers are used to capture biochemical flux. Special attention is placed on *in vivo* systems, however, many of the general ideas have immediate impact on studies in cellular models or isolated-perfused tissues. While it is somewhat trivial to administer labeled precursor molecules and measure the enrichment of downstream products, the ability to make correct interpretations can be challenging. We will outline several critical factors that may influence choices when developing and/or applying a stable isotope tracer method. For example, is there a “best” tracer for a given study? How do I administer a tracer? When do I collect my sample(s)? While these questions may seem straightforward, we will present scenarios that can have dramatic effects on conclusions surrounding apparent rates of metabolic activity.

Key words Isotope, Metabolomics, Flux, Pathway, Metabolic activity

1 Introduction: Tracer Studies Can Yield an Unambiguous View of Pathway Activity

The levels of a given metabolite are often a critical factor for differentiating healthy and disease states. For example, the concentration of circulating glucose in a fasted subject serves as a key diagnostic of diabetes. In other cases, intracellular metabolites (e.g., acyl-CoA, diacylglycerol or ceramide concentrations) could play a pivotal role in driving a disease phenotype [1–3]. Therefore, one could argue that attention should be focused on developing methods to readily quantify the concentration of any/all analytes. While this is certainly a valid perspective, the ability to measure pathway activity should also be closely positioned, especially since a biochemical flux rate(s) must change before a pool can change [4]. In fact, measurements of flux become more critical when one

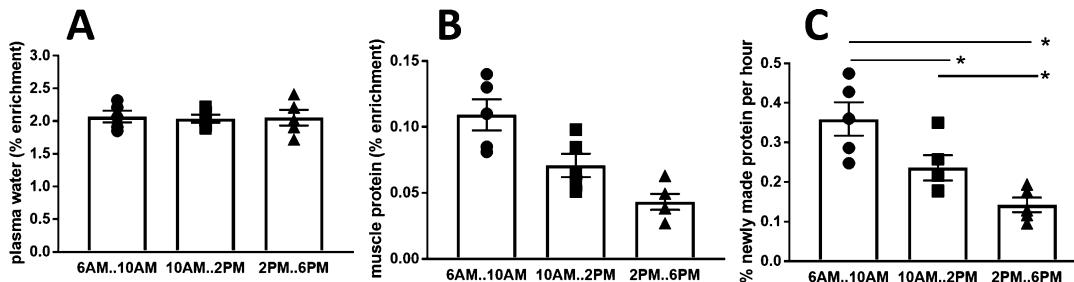


Fig. 1 Metabolic flux can change independent of pool size. Sprague-Dawley rats ($n = 15$, 253 ± 12 g, mean \pm sem) were fed a standard rodent diet on a 12-h light-dark cycle (dark between 6 PM and 6 AM). An intraperitoneal bolus of ^2H -water (15 μL 99% ^2H -water per gram body weight) was given to a subgroup of 5 rats at 6 AM and samples of plasma and mixed leg muscle were collected at 10 AM, a second group was then given a bolus of ^2H -water and samples collected at 2 PM, a final group was given a bolus of ^2H -water at 2 PM and samples were collected at 6 PM. Protein synthesis was estimated from measurements of the ^2H -labeling of plasma water and alanine (isolated from total muscle protein, $n = 5$ per group, Eq. 5) [7]. Although muscle mass did not change, different metabolic flux (protein synthesis) was observed at different times after the feeding cycle ended (Asterisk represents $p < 0.05$ using a 2-tailed t-test, assuming equal variance; this study was completed using an IACUC approved protocol)

aims to examine large pools that have seemingly low metabolic activity and/or when more than one pathway may contribute to the appearance of a product [5, 6].

We consider the following example to emphasize the novelty of studying pathway flux in a case where there is a large pool with a relatively low metabolic activity (Fig. 1). Sprague-Dawley rats weighing ~250 g were maintained on a 12-h feeding schedule (6 PM to 6 AM), following the end of a feeding cycle animals were subdivided into three groups. The first group was given an intraperitoneal bolus of ^2H -water at 6 AM and samples of mixed skeletal muscle were collected at 10 AM, at that time a second group was given a bolus of ^2H -water and samples were collected at 2 PM, the final group was given a bolus of ^2H -water at 2 PM and the samples were collected at 6 PM. Although changes in muscle mass could not be observed over these intervals, the incorporation of ^2H -alanine into total mixed muscle protein was dramatically different (Fig. 1b). The inability to detect differences in the pool size reflect the fact that the overall traffic flow (i.e., protein synthesis) only represents a small fraction of the total amount of protein, that is, less than 1% is made per hour (Fig. 1c); however, the tracer analyses clearly reveal differences in biochemical flux at different periods after the feeding. The literature contains the necessary details for readers who may be interested in using ^2H -water to study protein synthesis [7–12].

The fact that tracers can detect changes in metabolic activity before one can discern any change in concentration or pool size is novel but should be interpreted with some degree of caution too [13]. For example, the study outlined above only considered the

input of amino acids, we do not have a measure of protein degradation. Consequently, it is not possible to draw definitive conclusions regarding true long-term outcomes surrounding muscle protein homeostasis if we only examine one side of the biology. In cases where one can observe a change in the pool size while using tracers to probe the synthesis of a product, it should be possible to calculate the degradation (i.e., change in pool size = synthesis – degradation) [5, 14].

A second area where tracers can make an impact concerns examples where a product may be derived from multiple sources. Namely, although increased glucose production has been recognized as an important factor in diabetes, the contribution of specific organs (i.e., liver, kidney, and intestine) has been debated for many years [15–17]. Likewise, the importance of targeting glycogenolysis vs gluconeogenesis has generated considerable discussion [18–20]. Fortunately, tracers can be used to gain novel insight regarding integrative, whole-body metabolic biochemistry, including the pathophysiology surrounding disease states. We will further develop these types of scenarios by using triglyceride synthesis as a model problem. This example will draw out lesser appreciated factors that one must contend with when developing and/or applying a stable isotope tracer protocol.

2 Strategies to Enable Study Designs

We will consider several key questions related to the experimental design. In our opinion it is not possible to give specific protocol recommendations since conditions and models will vary, however, there are a few critical decision points that influence virtually all tracer studies. Our goal is to outline these steps using practical examples that underscore the impact of various choices.

2.1 What Is the “Best” Tracer?

This question is obviously broad, so we will try to provide an answer in the context of studying lipid flux, more specifically triglyceride synthesis. Perhaps a first line of thinking is to use a labeled fatty acid, one is then immediately faced with subsequent questions around which fatty acid and how to formulate the tracer? When thinking about “which fatty acid” one may consider ^{13}C - vs ^2H -labeled forms, as this could certainly be important from an analytical perspective, but the question is more complicated in the eyes of a biologist since palmitate and oleate may not be handled in the same way [21, 22]. Therefore, choosing one fatty acid over another could lead to different outcomes in the flux calculations under conditions where fatty acids are not utilized in proportion to their availability [6, 21, 22].

A less biased tracer might be labeled glycerol or glucose, these precursors are expected to provide the backbone on which fatty acids are esterified [23]. Again, the choice of how to label the

precursor and what product to measure could influence the data interpretation. Patterson and colleagues clearly demonstrated the formation of multiple glycerol species following the acute administration of [²H₅]glycerol. Although the injected glycerol tracer was 5 amu greater than that of the endogenous glycerol (M5 vs M0, respectively), the M5 triglyceride-glycerol only represented ~30% of the total labeled triglycerides that were synthesized [23]. In fact, they observed the appearance of triglyceride-glycerol species with 1, 2, 3 and 4 ²H (M1, M2, M3 and M4, respectively). While this type of data is consistent with known biochemical pathways, the implications of “isotope scrambling” needs to be considered when one thinks about modeling the metabolic flux (discussed later). For example, if we expected that since the precursor was M5 we only need to measure products that contain 5 ²H, could there be consequences if we ignore the M1, M2, M3, and M4 mass isotopomers?

Indeed, other tracer approaches could also be considered. For example, one could administer a tracer that labels fatty acids during de novo synthesis (i.e., ¹³C-acetate or ²H-water) or during (re) esterification (i.e., ¹⁸O-water) [24]. Consequently, the ability to define a truly “best” tracer can be difficult. Perhaps the most important factor here has little to do with the tracer protocol per se but rather the biological question that is to be addressed.

Finally, investigators should consider whether the tracer could perturb the metabolic flux that is under investigation [25]. In cases where ¹³C-glucose is used to probe lipid flux, one might hypothesize that there would be little metabolic alteration if we increased the pool size ~10% (which is a reasonable expectation if the goal is to reach ~10% enrichment in the glucose precursor). Presumably a relatively minor perturbation like this would have minimal impact on downstream processes. In contrast, if we chose to administer ¹³C-acetate we would have to contend with the delivery of extra Na⁺ (to avoid the acid load) and we would have to make assumptions about flux via a “new” pathway. For example, under most conditions one expects negligible flux of acetate [25, 26]. Therefore, even though ¹³C-acetate can be incorporated into fatty acids [27, 28], one may question whether the physiological state approximates true endogenous conditions.

2.2 How Do I Administer the Tracer?

To build on the examples that are outlined above we need to appreciate other challenges of the experimental design, namely, the tracer needs to enter the system that is under investigation. If we imagine *in vivo* studies, we are immediately presented with several options and challenges. Although most of the tracers noted earlier are water soluble, it is not easy to simply dissolve a labeled fatty acid in saline prior to administration. Fatty acid tracers are typically bound to albumin or solubilized with Intralipid [29], when an intra-vascular route of administration is used, or they can

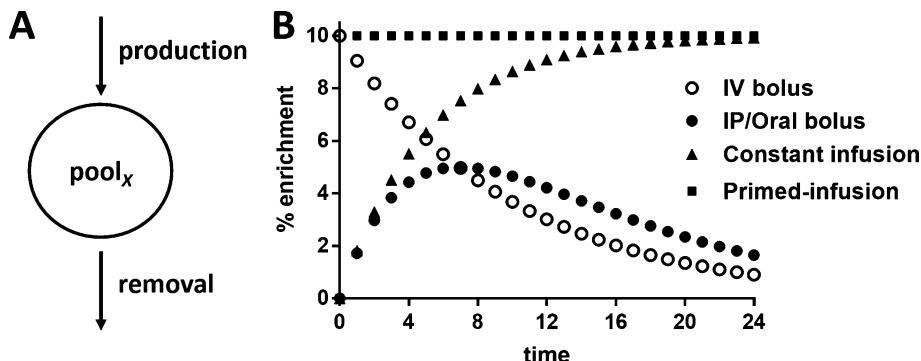


Fig. 2 Theoretical labeling profiles for different routes of tracer administration in vivo. If we consider a simple one compartment model (Panel a), we can expect distinct labeling profiles depending on how a tracer is administered (Panel b). For simplicity, we have assumed that one would target ~10% enrichment of a circulating pool. In the case where one contrasts the intravenous (IV) vs intraperitoneal (IP) or oral bolus methods, we have assumed that the same dose of tracer is administered. For purposes of illustration, we expect that the maximum labeling achieved with the IP or oral bolus will never reach that when the same dose of tracer is given as an IV bolus; when using the IP or oral bolus the temporal rise to maximum enrichment is impacted by the balance between absorption and metabolism kinetics. A constant infusion will be associated with a rise to steady-state labeling, whereas a primed-infusion could lead to an “immediate” steady state if the system is well described

be mixed with food and then administered (consumed) orally [22, 30, 31].

The route of administration is associated with more caveats, namely, if we obtain intravenous access should we use a bolus injection or an infusion [32, 33]? Likewise, if we choose an oral route of administration is it better to mix the tracer with food or can we add isotopes to the drinking water [34]? Again, there is not really a right or wrong answer, the approach will likely be driven by the question. We should also note that many tracer studies use a hybrid technique, that is, the primed-infusion (Fig. 2). Although intuition may lead one to think that primed-infusions imply that we are talking about intravenous studies, in fact, the use of labeled-water also falls under this category; one can give an intraperitoneal bolus followed by addition of the tracer to the drinking water, therein achieving a “square-wave” profile of the precursor. The ability to deliver tracers for long periods of time allows for studies of analytes with slow rates of turnover [6, 11].

2.3 How Can I Get a Metabolic Rate from My Tracer Data?

An important factor to consider when deciding on the approach for administering a tracer is the type of kinetic data that one intends to obtain. For example, studying the overall kinetics of a product may require a different strategy as compared to dissecting the source(s) of that product. We will consider the kinetics of triglyceride in adipose tissue as a model problem, however, we will first review some general concepts concerning a simpler system and then

introduce the key ideas surrounding the use of precursor → product labeling profiles in studies of more complicated problems.

Figure 2a contains an example of a single pool, production (input, or rate of appearance) is usually thought to follow zero-order kinetics and removal (outflow or rate of disappearance) is generally assumed to be first-order [35]. Figure 2b demonstrates four scenarios that can be encountered when tracers are used to estimate a metabolic rate, for example, glucose turnover in the plasma [35]. The plots represent the outcomes that one expects if labeled glucose (e.g., [U^{-13}C_6]) is given as an intravenous bolus (open circles), an oral or intraperitoneal bolus (solid circles), a constant infusion (diamonds) or a primed-infusion (squares). We will briefly review how each approach can be used to estimate the kinetics of a given analyte.

First, if we assume there is rapid mixing of the tracer in a single pool, then, following the administration of an intravenous bolus of the tracer (open circles, Fig. 2b), we can determine the glucose flux from the pool size and the fractional turnover. The former is determined from the initial tracer dilution and the later from the tracer dilution measured over time. Specifically, one can estimate the pool size using the equation:

$$\text{pool size} = \frac{\text{tracer dose}}{\text{initial enrichment}} \quad (1)$$

where the units of tracer dose (mg, μmol , etc.) reflect the units that define the pool size. The term “initial enrichment” can be estimated by extrapolating the tracer dilution to the time = 0 point or, if the system mixes very rapidly, one may be able to simply measure the enrichment shortly after injecting the tracer. If we consider the problem of glucose kinetics, one typically assumes a single (well-mixed) compartment and therefore the decrease in enrichment (or tracer dilution) can be described using the equation:

$$\text{enrichment}(t) = \text{initial enrichment} \times e^{-kt} \quad (2)$$

where “ k ” represents the fractional turnover. In this scenario there are different approaches for determining the tracer dilution. When several samples are collected one can fit the decrease in enrichment using nonlinear regression, or, in cases where two samples are collected, one can transform the y -axis to “ln scale” and determine the slope using linear regression [35, 36].

Second, in cases where an intraperitoneal (or oral) bolus is used one can estimate the kinetics by modeling the entire curve or by using a limited region of the data (i.e., after the peak has been reached). Again, if we assume that the analyte exists in a single, well-mixed, pool we can determine the flux using the equation:

$$\text{rate of appearance} = \frac{FD_0[\text{glucose}]k^{\text{abs}}k^{\text{dil}}}{k^{\text{abs}}c_0^{\text{dil}} - c_0^{\text{abs}}k^{\text{dil}}} \quad (3)$$

where F is the bioavailability of the tracer, D_0 is the dose of the tracer administered at time 0, and $[\text{glucose}]$ is the mean blood glucose pool throughout the experiment. The terms k^{abs} and k^{dil} are the kinetic turnover rates for absorbance and tracer dilution (commonly referred to as elimination), respectively, and c_0^{abs} and c_0^{dil} are the back calculated concentrations of the tracer at time 0 for the absorbance and dilution (elimination) curves, respectively [33]. When tracers are given via an IP or oral bolus one can estimate the metabolic rate using different approaches. For example, van Dijk et al. proposed an elegant application of this logic in studies of glucose kinetics in rodent models, in which they considered the entire labeling profile [33]. We performed comparable studies but only modeled the tail of the tracer dilution [32, 37].

When either a constant infusion (triangles) or a primed-infusion (squares) is used, we can estimate the turnover from the steady-state enrichment (i.e., the dilution rate of the tracer) according to the equation:

$$\begin{aligned} \text{rate of appearance} &= \text{tracer infusion rate} \\ &\times \left(\frac{\text{enrichment of the infusate}}{\text{enrichment of the analyte pool}} - 1 \right) \end{aligned} \quad (4)$$

where the units that define the “rate of appearance” are the same as those used to define the “tracer infusion rate.” We should note that this approach is often used in studies of circulating analytes; therefore, the “enrichment of the analyte pool” represents the enrichment of some endogenous molecule in the plasma. In theory, the use of a primed-infusion allows one to reach a steady-state enrichment of the analyte pool in a shorter amount of time as compared to when a constant infusion is used. That said, there are examples where achieving the correct balance between priming doses and infusion rates can require some development, otherwise investigators run a risk of making misleading conclusions [35, 38, 39].

The types of scenarios that were just described have been widely used to quantify the kinetics in cases where a pool turns over with considerable frequency over the duration of an experiment (circulating glucose is an excellent model problem). However, there are other cases where a pool may experience very limited synthesis (e.g., mixed muscle protein, Fig. 1). We will consider how tracers can be used to estimate kinetics in these cases as well. This next scenario emphasizes a key point, in that, we may not be able to directly administer a labeled form of the product. Consequently, one will typically rely on the use of a precursor-product labeling

ratio. The general rationale is that if a precursor can be maintained at a stable enrichment then the enrichment of the product will increase and reach a steady-state enrichment that is equal to that of the precursor [13]; although this may seem straightforward there are some important caveats to this logic which will be highlighted [13, 40].

In cases where a precursor-product labeling ratio is used to quantify the kinetics one can estimate the fractional synthesis of a product using the equation:

$$\text{fraction newly made product} = \frac{\text{initial change in the enrichment of the product}}{\text{enrichment of the precursor} \times \text{time}} \quad (5)$$

This equation applies when the precursor rapidly (“instantaneously”) reaches a steady-state enrichment and one can measure a pseudolinear increase in the enrichment of the product [41, 42]. In cases where one collects multiple samples it is possible to model the change in enrichment of the product using the equation:

$$\begin{aligned} \text{product enrichment}(t) &= \text{steady-state product enrichment} \\ &\times (1 - e^{-kt}) \end{aligned} \quad (6)$$

As with Eq. 5, Eq. 6 requires that the precursor enrichment is stable, preferably reflecting a “square wave” over the period in which the product enrichment is being sampled [41]. In cases where the precursor enrichment may vary, the same logic will apply but adjustments need to be made in the calculations [43, 44].

Now let us consider how we might estimate triglyceride kinetics, for example, in adipose tissue. In this scenario it is not possible to directly administer a labeled triglyceride and measure its enrichment over time in adipose tissue (triglycerides are normally hydrolyzed by lipoprotein lipase and not directly removed by adipocytes) [45]. Therefore, we need to consider how to measure the conversion of a labeled precursor into a labeled product, (e.g., [¹³C₆] glucose → [¹³C₃]glycerol-3-phosphate → [¹³C₃]triglyceride-glycerol). For simplicity we will assume what is expected when either a single IV bolus or a primed-infusion is used (Fig. 3), as well, we will assume that one would measure the enrichment of the triglyceride-glycerol backbone. It is certainly possible that fatty acids will be labeled too, in fact, one can find major discrepancies between the kinetics of triglyceride-glycerol vs triglyceride-fatty acids. Although these discrepancies are especially obvious when the structural composition of the pool is being remodeled we will ignore fatty acid labeling for now [6].

When the precursor is given as a bolus, the product enrichment will rise and then fall over time (Fig. 3a). Obviously, the magnitude

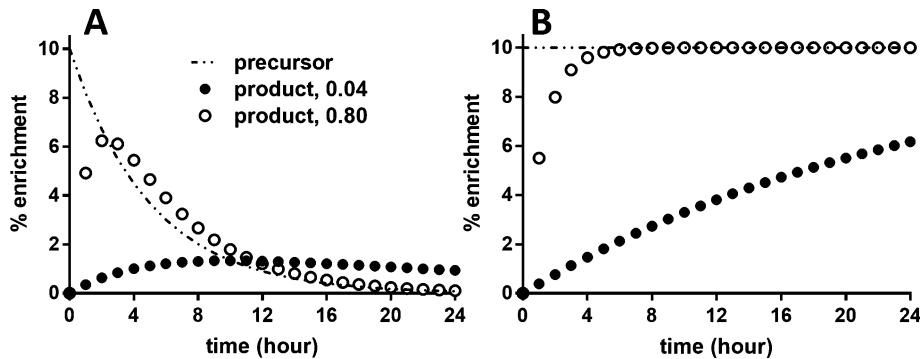


Fig. 3 Theoretical precursor and product labeling profiles for different routes of tracer administration in vivo. Panel **a** considers a case where two subjects are given the same IV bolus of a labeled precursor (e.g., [$\text{U}-^{13}\text{C}_6$] glucose) and the temporal dilution is identical between the subjects (dashed line), however, the product (e.g., triglyceride) is turning over at ~ 0.04 or 0.80 pool/h, solid or open symbols, respectively. Panel **b** considers the same scenario, however, the precursor is given as a primed-infusion (dashed line); again, the product is turning over at ~ 0.04 or 0.80 pool/h, solid or open symbols, respectively

and duration of the change in product enrichment will be driven by how much precursor is administered, the half-life of the precursor and the turnover of the product [44]. In contrast, if we give a primed-infusion of the precursor we expect to see a steady rise in the enrichment of the product (Fig. 3b). Note that in both Fig. 3a, b, we assume that the product represents a single pool modeled by a single exponential. We will now compare these modes of tracer administration and include the labeling profiles in cases where products may turnover at ~ 0.04 or ~ 0.8 pools/h (we have arbitrarily set the glucose precursor to be $\sim 10\%$ enriched). When a bolus is administered it is possible to draw different conclusions regarding the relative activity of the respective pools. For example, although there is a 20-fold difference between the turnover of the respective product pools, if we collected a sample at a later point in time (e.g. at ~ 12 hours), we might erroneously conclude that the pools have comparable activity, or that the slower pool is turned over more rapidly than the faster pool (Fig. 3a). In contrast, the use of a primed-infusion would allow us to ascribe the correct relative activities of the pools regardless of when we collected a sample, however, a drawback is that the true difference between the activities of the respective pools is grossly underestimated if we assumed that a single sample could provide a definitive answer regarding the kinetics (Fig. 3b). Namely, using a sample collected at 8, 16 or 24 h (open circles, Fig. 3b) and applying Eq. 5 we would conclude that the fractional synthesis is ~ 0.13 , 0.06 , or 0.04 pools/h, respectively [41]; none of these values reflect the true turnover (~ 0.80 pool/h), therein underscoring the impact of indiscriminate sampling.

2.4 How Many Samples Do I Need and When Should They Be Collected?

Building on the example shown in Fig. 3 raises obvious questions around how the number of samples and the timing of their collection will impact our conclusions [46]. In our experience, even if we could collect multiple samples the use of a single tracer bolus is not practical for studying a problem such as triglyceride flux in adipose tissue. The pool size is generally quite large, and the turnover is relatively slow. We should note that the time scale used here is exaggerated even for rodent studies [5, 6]. This does not mean that the logic surrounding a bolus administration of a tracer is flawed, in fact, this approach is used in studies of plasma lipoprotein kinetics [23, 47]. However, if a single bolus is administered we are somewhat obligated to collect multiple samples to determine if we are on the up-slope or the down-slope of a product labeling curve. Note that for the rapidly turning over product in Fig. 3a, the enrichment is comparable at ~1 h and ~5 h. If we had only collected a sample at ~5 h, we could draw appropriate conclusions regarding the relative differences between the kinetics of the two product pools. The open circles clearly demonstrate substantially more enrichment as compared to the closed circles, consistent with the faster relative turnover (Fig. 3a). Unfortunately, we would be far from an accurate estimate of the true turnover, for example, the open circles are on the descending phase of the enrichment curve by 5 h and therefore we have grossly underestimated the synthesis in that condition.

Those conclusions are in strong contrast with the example in Fig. 3b, where the enrichment continues to increase until it reaches a plateau, making it possible to consider using a single time point to estimate the kinetics [42]. Such a statement would be appealing to the analytical lab since it would mean fewer samples to process and less data to acquire, which implies a level of resource sparing. However, it also has the potential drawback that the longer we wait to collect our samples the less accurate will be our estimation of the true differences between the two product pools [42]. Although it is tempting to think that a single data point can be used to infer the kinetics (e.g., Eq. 5), several factors must align to ensure the validity of that logic. Given the complexity of some biological problems it is perhaps better to consider more extensive sampling schemes. At the very least, pilot studies should evaluate critical assumptions noted here. Those preliminary data may suggest reasonable options to simplify the experimental design in follow up studies.

These questions are of central importance in virtually all studies where tracers are administered in the context of studying precursor → product relationships, as such, they deserve special attention. In addition, addressing these questions should underscore the importance of tracer administration modes and sampling schemes on the data interpretation. A discussion of the topic is best

2.5 How Do We Know Whether the Intracellular Dilution of a Precursor Is Creating a Problem and How Can We Adjust the Calculations of Metabolic Flux?

exemplified if we build off Fig. 3b. We will add some complexity to reflect true events surrounding metabolic biochemistry and integrative physiology.

The example in Fig. 3b shows a case where our substrate (e.g., ^{13}C -glucose) is the only precursor for the glycerol backbone in triglyceride; therefore, we expect that we will eventually reach a steady-state enrichment of triglyceride-glycerol and that its enrichment will equal that of the precursor. In the example, the labeled glucose precursor pool is ~10% enriched, consequently, the product should reach the same level if there are no other sources of the glycerol backbone. However, several studies have demonstrated the existence of alternative sources of triglyceride-glycerol [5, 48–51]. For example, although the adipose tissue is thought to have little or no glycerokinase, it has been hypothesized that glyceroneogenesis could occur (i.e., the conversion of lactate, pyruvate, etc. to triose-phosphates) [52]. Let us further evolve the reaction sequence in our metabolic pathway to see what happens if glucose is not the only source of triglyceride-glycerol.

Figure 4a outlines a scenario that is supported by experimental data. If we assume that ^{13}C -glucose is given as a primed-infusion, and that the precursor is ~10% enriched we expect that the enrichment of triglyceride-glycerol will reach that of the precursor (as shown in Fig. 4c, open triangles). However, if we found comparable precursor enrichment in two conditions but different product enrichment (i.e., we collected samples from the respective groups at any of the time points noted in Fig. 4c) can we confidently state that the metabolic flux of the product is different between the respective groups? The answer is “no.” Although the examples represented in Fig. 4a, b demonstrate a case where two groups are maintained at the same glucose enrichment (~10%) and the product enrichment is approximately twofold different for the respective curves. We do not know whether some factor other than triglyceride synthesis is altered. In fact, the simulation shown here reflects differences in the precursor dilution and not in the triglyceride kinetics. The triangles (Fig. 4c) represent the enrichment of triglyceride-glycerol, in each case the glucose is ~10% enriched and the half-life of triglyceride is ~ 3.4 h (or ~0.2 pool/h). However, the open triangles reflect the outcome if ^{13}C -glucose was the sole precursor (Fig. 4a), whereas the solid triangles reflect the outcome if there was another source of triglyceride-glycerol (lactate, pyruvate, etc.), that enters the cell “cold” (Fig. 4b). Since this second source of the glycerol backbone is not labeled the pathway is invisible, this underscores a key assumption, in that, the labeled precursor that we administer may not be the true precursor. For example, if we collected a single sample at ~6 h and we applied Eq. 5 we would conclude that the turnover was ~0.11 vs ~0.06 pools/h, open vs. solid triangles, respectively. This is in strong contrast to the true value of ~0.20 pools/h in both cases.

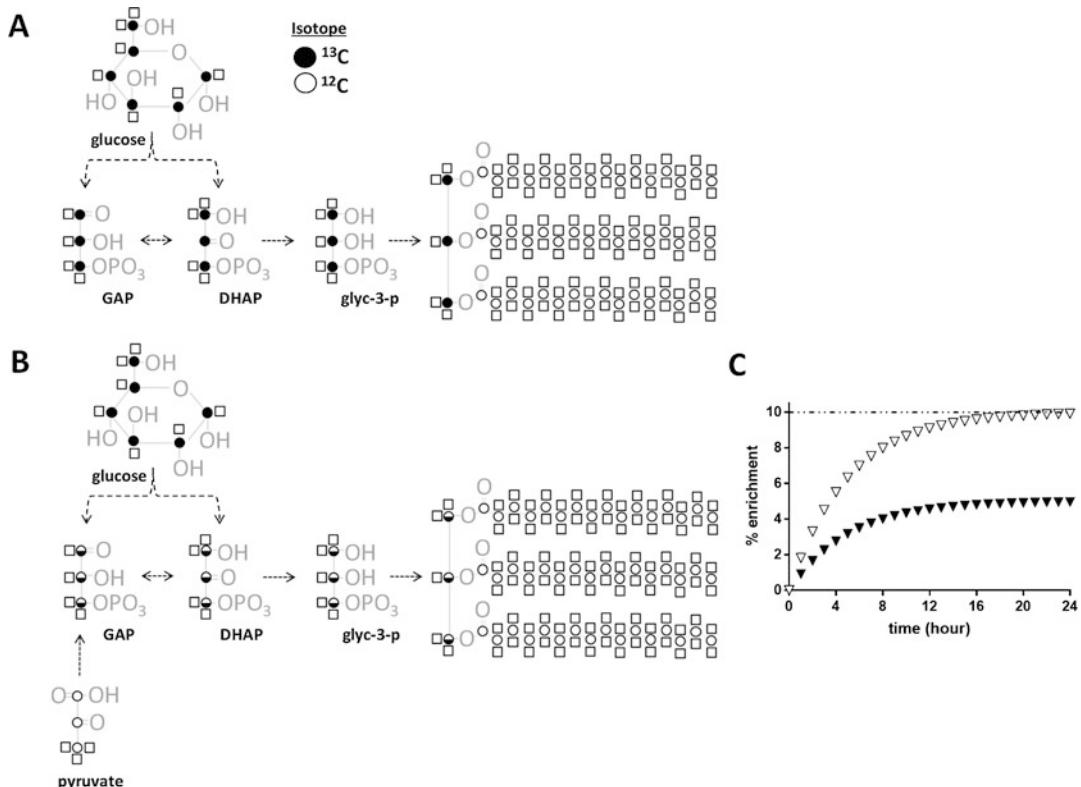


Fig. 4 Product labeling can be impacted by biochemical rates and sources of substrates. One can consider using a primed-infusion of [U^{13}C_6]glucose in studies of triglyceride kinetics. For simplicity, we assume that ^{13}C - will only be incorporated into the glycerol-backbone (in reality, ^{13}C - can be incorporated into the fatty acids as well). If glucose is the only carbon source (Panel a), the ^{13}C - enrichment of triglyceride-glycerol will reach a steady-state (Panel c, open triangles), which is equal to that of the ^{13}C -glucose being infused (Panel c, dashed line). In theory, one could estimate triglyceride turnover by collecting a single sample during the pseudolinear phase of the triglyceride labeling and comparing that against the ^{13}C -enrichment of the infused glucose (see Eq. 5). In contrast, if glucose and pyruvate, for example, each made similar contributions to the triose-phosphate (and glycerol-phosphate) pool (Panel b), we could observe very different absolute changes in the triglyceride enrichment (Panel c, solid triangles). In this case, we would draw erroneous conclusions by comparing the enrichment of the product with that of the infused glucose (see Eq. 5). The true precursor (i.e., glycerol-3-phosphate) receives equal input of carbon from labeled (e.g., glucose) and unlabeled (e.g., pyruvate) sources. If we sampled at several time points (e.g., 2, 6, 12, and 24 h) and applied Eq. 6 we would determine that the triglyceride turnover is the same between the groups (i.e., the correct conclusion). Note that the correct stoichiometry to generate the curve represented by the closed triangles in Panel c is 1 glucose and 2 pyruvate

It is important to note that the sampling scheme can help dissect these types of problems. For example, if we ran the experiment described above, and we determined that the glucose enrichment was the same between two groups, but the enrichment of triglyceride-glycerol was different (i.e., we collected a sample at ~6 h) we might be inclined to conclude that triglyceride synthesis

was different between the groups (as we just noted). However, this would assume that the ^{13}C -enrichment of glucose reflects the true precursor labeling for both groups. On the other hand, if we were able to collect several samples across a reasonable time range (e.g., ~2, 6, 12, and 24 h) we would immediately recognize that the fractional turnover of the triglyceride is comparable between the two groups, that is, if we fit the enrichment curves using Eq. 6 we would see that it takes the same amount of time to reach a plateau in both groups (Fig. 4c). We would also note that the plateau labeling is different between the groups, which could be explained by the entry of “cold” carbon; the dilution of our labeled glucose precursor could be “proved” by measuring the enrichment of an intermediate at or beyond the mixing point. That is, if we sampled the triose-phosphates or glycerol-3-phosphate we would see that the enrichment is 50% lower than that of the glucose. Future studies could then be designed with this new knowledge in mind. For example, we could sample the triose-phosphates or glycerol-3-phosphate enrichment and set that value as the true precursor labeling, we could then consider using a single time point (e.g., measure triglyceride-glycerol at a 6 h time point and Eq. 5) and therein draw reliable conclusions regarding triglyceride kinetics [42]. This could make some aspects of our studies simpler since we could perhaps then collect and measure fewer samples, and the experimental time might be shortened (e.g., from 24 h to 6 h).

A critical take-home message is that although we may observe the incorporation of a labeled precursor into a product we should question the potential of any dilution in the steps between the precursor and the product. The example shown here might seem extreme since there are several reactions between glucose and triglyceride-glycerol and readers may think that if we use a more direct precursor that this problem would not arise. That is not entirely the case – studies of protein synthesis often administer a labeled amino acid and measure its incorporation into a protein (s) [35, 53]. Readers may recognize that there are fewer steps in that precursor → product relationship, that is, free amino acids → tRNA-bound amino acids → protein-bound amino acids. However, another problem is observed: the enrichment of the amino acid that is administered can undergo substantial intracellular dilution since cells continuously degrade proteins and recycle amino acids. In fact, the enrichment of free amino acids has been shown to experience ~40% dilution upon entry into a cell [54].

This raises a question regarding the novelty of using ^2H - or ^{18}O -water to study metabolic flux [24]. Briefly, water readily moves across membranes and its enrichment is virtually identical in all body compartments shortly after it is administered [55, 56]. For example, we have given ~5–20 μL of ^2H -water per gram body weight to rodents (via an intraperitoneal bolus) and observed ≥90% distribution within ~20 min (unpublished observations). Since labeled water rapidly enters cells we should expect a homogeneous labeling. However, we then need to assume that the entry of

labeled water into intermediates that are used in the synthesis of end-products is faster than the synthesis of the end-products [57], that is, the generation of ${}^2\text{H}$ -labeled triose-phosphates should be much faster than their incorporation into triglyceride-glycerol. A critical caveat centers on whether it is acceptable to measure the enrichment of the entire triglyceride-glycerol or whether we should devise a strategy to measure the enrichment of specific hydrogens [5, 6, 43, 48, 58].

3 Impact of “Secondary” Tracers

Our discussion has considered somewhat clean systems, in which the precursor has only a single route through which it can be incorporated into a product. This is rarely the case when we conduct *in vivo* experiments, as one might expect since labeled precursors can traverse pathways in multiple tissues and then mix again in the plasma before undergoing incorporation into a product of interest [59]. We will now outline some of the problems that can arise around data interpretation in cases where secondary tracers are produced, this can impact seemingly “clean” *in vitro* models just as severely as it can impact more complex *in vivo* models. There are good examples where investigators have recognized how tracer recycling could impact the interpretation of stable isotope-resolved studies in cases where acute tracer perturbations are used to probe tissue- and pathway-specific activity *in vivo* [60, 61].

3.1 Our Primary Tracer Undergoes Rearrangement Before Reaching Its Final Destination

We previously incubated hepatocytes with $\sim 150 \mu\text{M} [{}^2\text{H}_5]\text{glycerol}$ and measured the ${}^2\text{H}$ -enrichment of glyceride-glycerol at various times. Figure 5 demonstrates the generation of different labeled species which can be explained if $[{}^2\text{H}_5]\text{glycerol}$ undergoes conversion to triose-phosphates faster than incorporation into triglyceride-glycerol. These data are consistent with the literature [23] and imply that we could miss aspects of triglyceride-flux by ignoring the generation of alternate precursors. For example, in this case the primary glycerol tracer was M5 but the product triglyceride-glycerol contained all of the mass isotopomers. As eluded to earlier, the observation is not an artifact of using an *in vitro* model, since this occurs in humans [23] and is consistent with known biochemical reactions (Fig. 5a). Therefore, one should be aware of the potential for isotopic rearrangements and/or scrambling of the labeling localization.

3.2 Cross Talk of Tracers Between Tissues

If we aim to study metabolic activity *in vivo* we must recognize that our primary tracer can generate other labeled precursors. Not only can this occur in the form of different mass or positional isomers of the original precursor (shown in Fig. 5) but one can also expect new sources of label altogether. An example of secondary tracers can be

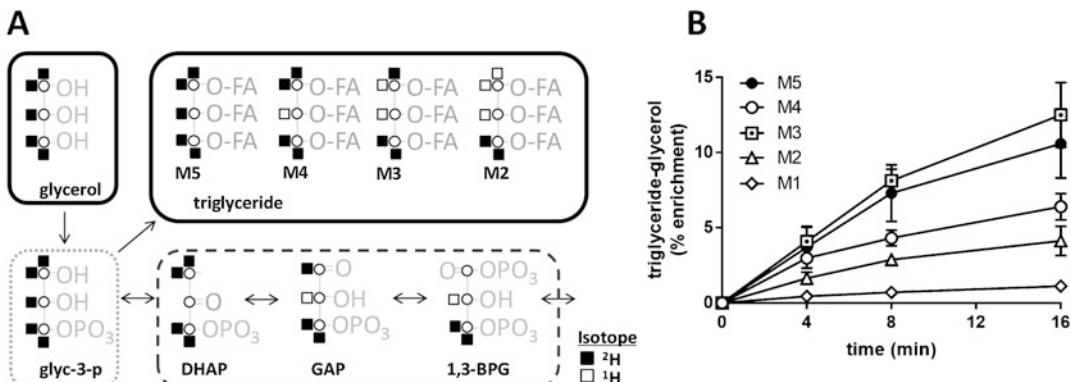


Fig. 5 Impact of side compartments on isotope scrambling and potential errors estimating biochemical flux. The biochemical model outlines potential reactions in the conversion of [²H₅]glycerol to triglyceride-glyceride (Panel a). Although there are two steps in the pathway (i.e., glycerol → glycerol-3-phosphate → triglyceride), the pool of glycerol-3-phosphate is also linked to the pool of triose-phosphates. Consequently, the relative rates of flux (i.e., glycerol-3-phosphate → triglyceride vs glycerol-3-phosphate ↔ triose-phosphates) will impact the mass isotopomer distribution profile of triglyceride-glycerol (Panel a). Hepatocytes were isolated from rats and incubated immediately with ~150 μM [²H₅]glycerol (M5, >95% enriched), triglycerides were isolated at different times and the appearance of various isotopically labeled species (M1 → M5) was measured (Panel b, unpublished data from previous studies [76]). Note that solid and open squares represent ²H and ¹H, respectively

seen in studies of triglyceride flux in adipose tissue, where investigators used labeled water to capture total glyceride flux and carbon-labeled glucose to differentiate the sources of triglyceride-glycerol (although in some studies the glucose tracer contained radioactive ¹⁴C- atoms the impact would be the same if it had contained ¹³C-glucose) [49, 50]. In these types of studies, we should appreciate the fact that carbon-labeled glucose is consumed by virtually all cells and we should expect the generation of carbon-labeled by-products (e.g., carbon-labeled lactate and pyruvate) (Fig. 6a) [62]. Therefore, it is not advisable to conclude that the carbon labeling of triglyceride-glycerol can differentiate the contribution of glucose vs. nonglucose sources of triglyceride-glycerol. Although one may consider that 3-carbon intermediates will enter and traverse the TCA cycle (shown as red symbols, Fig. 6a) and therefore one could make corrections for tracer recycling by measuring the M1 → M3 species, the scheme has oversimplified the physiological problem. In fact, the carbon-labeled lactate generated by the red blood cells, for example, could be removed by the liver or kidney, enter/traverse the TCA cycle, and appear again in the blood as carbon-labeled glucose [59], in which case, one would still observe a scrambled isotope pattern in the triglyceride-glycerol; however, the M1 → M3 species could now also come from glucose. This type of tissue cross talk can dramatically confound the

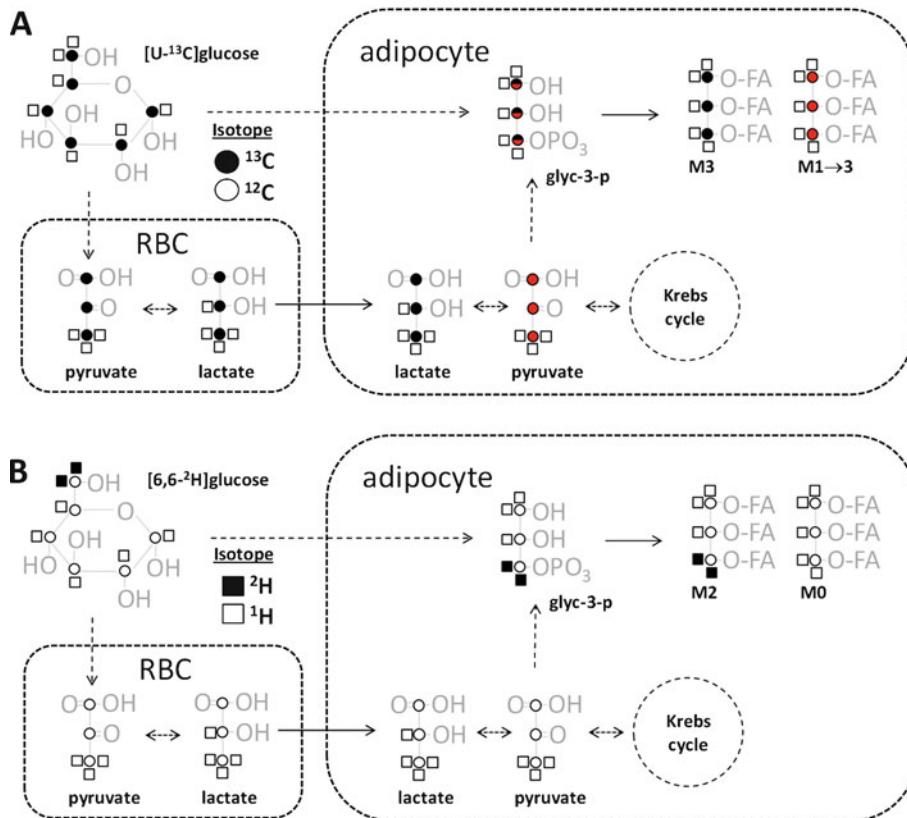


Fig. 6 Integrative physiology and tissue-induced tracer cross talk. The interpretation of tracer studies requires special attention when there is tissue cross talk. Although labeled-glucose can be used to quantify triglyceride synthesis, one should exercise caution when choosing the tracers if the goal is to sort out the contribution of specific pathways. As shown in Fig. 4, sampling of glycerol-phosphate (or triose-phosphates) may be necessary to study true rates of triglyceride turnover. One could imply that comparing the enrichment of glycerol-phosphate with that of glucose could also yield information on the source(s) of triglyceride-glycerol. Panel A demonstrates that it is possible to “recycle” label from one tissue to another. In this case, the infusion of [$^{13}\text{C}_6$]glucose can lead to the generation of ^{13}C -pyruvate in an adipocyte (via glycolytic flux/transfer of ^{13}C - from the red blood cell, RBC). The solid red circles represent ^{13}C -lactate/pyruvate that traverses the Krebs cycle prior to conversion into triglyceride-glycerol; those reactions would lead to the generation of different mass isotopomers. In contrast, Panel B demonstrates that [$6,6-^2\text{H}_2$]glucose should not be impacted by this type of tissue cross talk, ^2H should only appear in triglyceride-glycerol if there is direct conversion of glucose. The ^2H -labeling of pyruvate could be affected by keto-enol tautomerism, transamination and/or by traversing the TCA cycle before it moves to glycerol-3-phosphate

interpretation in cases where one aimed to differentiate the sources of an end-product.

In contrast, the outcome is expected to be different if one uses [$6,6-^2\text{H}_2$]glucose (Fig. 6b). This tracer will undergo the same peripheral metabolism as carbon-labeled glucose however the isotope is lost during those reactions. Consequently, we should expect

that the appearance of labeled triglyceride-glycerol from [6,6-²H] glucose will accurately reflect the direct contribution of glucose [5].

The generation of secondary tracers via tissue cross talk has been known for many years and can present problems for studies that rely on *in vivo* models [25, 59, 63]. Not only can this be seen in examples where we look for precursor → product conversions, as shown above, but we can also see an inverse type of problem when we look at substrate oxidation [64]. For example, suppose we aimed to measure the conversion of glucose to CO₂, that is, the movement of a key metabolic substrate to a waste product. Again, intuition may lead us to think that administering a ¹³C-glucose tracer would be desirable and that all we would need to do is measure the ¹³C-enrichment of CO₂. In fact, this is not so simple since ¹³C- can escape the pathway via exchange reactions at key nodes. This was clearly demonstrated by Krebs and Katz in classical studies that used radiolabeled tracers [59, 63] and elegantly demonstrated in more recent studies by Wolfe and colleagues using stable isotopes in humans [64–66]. Just as we might expect, correction strategies have been proposed to circumvent some of the problems that are associated with the oxidation of ¹³C-tracers [65–67]. As an alternative, investigators have proposed the use of ²H-labeled tracers, where measures of substrate oxidation via the production of ²H-water would be less influenced by exchange reactions [68–70].

4 Notes

Presumably readers may now start to develop a better appreciation of key areas on which to focus attention when designing and executing tracer-based studies of metabolic flux. Although there have been many advances in the hardware and software as related to mass spectrometry and nuclear magnetic resonance spectroscopy, problems surrounding the physiology of integrative systems still require attention. It is our experience that investigators can now readily collect and process large sets of data, however, we should recognize the many caveats and pitfalls can influence our ability to reliably interpret the biochemical flux or metabolic activity. Fortunately, there is a wealth of information, which can help us avoid some of the problems outlined here [35, 44, 53, 71].

We should also note that our discussion has been biased toward a consideration of enrichment, that is, the ratio of labeled to total species of a given analyte. Although there are different views on how to express stable isotope labeling [72–75] it is necessary to emphasize a key aspect of tracer studies when the goal is to measure a synthetic process vs a degradative process [13]. The examples outlined here have focused attention on events surrounding the synthesis of new material. For example, we gave ¹³C-glucose that

was ~10% enriched and we noted that our product, that is, triglyceride-glycerol, could reach 10% enrichment if glucose was the sole substrate being used in the synthesis. In cases where we aim to measure the breakdown can we extract information from the decrease in enrichment? For example, if one thinks about classical pulse-chase experiments, intuition might lead one to consider the decrease in enrichment being caused by a clearance or degradation process. In fact, this is not true [13]. The decrease in enrichment that one observes during a “chase,” that is, once the precursor input has been stopped, will reflect the dilution of a tracer (or, stated another way, the synthesis of that product from a cold source). There seems to be some confusion in the literature regarding these matters, and this is certainly one area where stable isotope-based experiments can deviate from classical radiolabeled tracer studies [13].

In our experience it is not possible to give strict conditions for designing a tracer study, as the exact details will depend on many variables. For example, we would likely design different protocols if we were interested in studying triglyceride synthesis in adipose tissue vs liver in a mouse model or if we were interested in triglyceride synthesis and secretion in plasma in a rodent model vs a human subject. This is in strong contrast to purely analytical methods, that is, lipidomic analyses that might be used to measure the abundance of a triglyceride. In those cases, the same (or similar enough) approach can be used to quantify the abundance and isotopic labeling of triglycerides regardless of their source. Classical methods (e.g., the Folch or Bligh and Dyer extraction) can be used to isolate triglycerides from plasma, liver or adipose tissue samples (regardless of the model). Therefore, when one aims to quantify the abundance and/or isotope labeling of an analyte it is possible to devise more concrete steps or rules to ensure consistency of the data.

Hopefully we have provided some general guidelines on which to begin designing tracer experiments. Investigators should appreciate the fact that protocols which are valid in one type of study may require substantial editing for use in another type of study, even if it appears that we have the same goal in the respective studies. In our experience tracer studies require that one respect a handful of rules but there is also an opportunity for creativity. Everyone will recognize a basic rule learned in elementary school, that is, mixing blue and yellow yields green; however, artists regularly adjust the mixture to achieve the “best” green for a given painting. In our opinion, the effective application of tracer methods requires that one balance some strict principles against some imagination since biological problems vary in their complexity, including the constraints that are imposed when examining different models.

Acknowledgments

SF Previs thanks Dr Richard Higashi (University of Kentucky) for interesting discussions which influenced this work, the central ideas and flow of this manuscript evolved as we shared notes regarding our respective talks for a joint presentation at a conference.

References

- Samuel VT, Liu ZX, Qu X, Elder BD, Bilz S, Befroy D, Romanelli AJ, Shulman GI (2004) Mechanism of hepatic insulin resistance in non-alcoholic fatty liver disease. *J Biol Chem* 279(31):32345–32353
- Erion DM, Shulman GI (2010) Diacylglycerol-mediated insulin resistance. *Nat Med* 16(4):400–402
- Coen PM, Goodpaster BH (2012) Role of intramyocellular lipids in human health. *Trends Endocrinol Metab* 23(8):391–398
- Turner SM, Hellerstein MK (2005) Emerging applications of kinetic biomarkers in preclinical and clinical drug development. *Curr Opin Drug Discov Devel* 8(1):115–126
- Bederman IR, Foy S, Chandramouli V, Alexander JC, Previs SF (2009) Triglyceride synthesis in epididymal adipose tissue contribution of glucose and non-glucose carbon sources. *J Biol Chem* 284(10):6101–6108
- Brunengraber DZ, McCabe BJ, Kasumov T, Alexander JC, Chandramouli V, Previs SF (2003) Influence of diet on the modeling of adipose tissue triglycerides during growth. *Am J Physiol Endocrinol Metab* 285(4):E917–E925
- Gasier HG, Fluckey JD, Previs SF (2010) The application of (H_2O)-H-2 to measure skeletal muscle protein synthesis. *Nutr Metab* 7:31
- Previs SF, Fatica R, Chandramouli V, Alexander JC, Brunengraber H, Landau BR (2004) Quantifying rates of protein synthesis in humans by use of (H_2O)-H-2: application to patients with end-stage renal disease. *Am J Physiol Endocrinol Metab* 286(4):E665–E672
- Wilkinson DJ, Franchi MV, Brook MS, Narici MV, Williams JP, Mitchell WK, Szewczyk NJ, Greenhaff PL, Atherton PJ, Smith K (2014) A validation of the application of D2O stable isotope tracer techniques for monitoring day-to-day changes in muscle protein subfraction synthesis in humans. *Am J Physiol Endocrinol Metab* 306(5):E571–E579
- Miller BF, Wolff CA, Peelor FF III, Shipman PD, Hamilton KL (2015) Modeling the contribution of individual proteins to mixed skeletal muscle protein synthetic rates over increasing periods of label incorporation. *J Appl Physiol* 118(6):655–661
- Rachdaoui N, Austin L, Kramer E, Previs MJ, Anderson VE, Kasumov T, Previs SF (2009) Measuring proteome dynamics *in vivo*. *Mol Cell Proteomics* 8(12):2653–2663
- Busch R, Kim YK, Neese RA, Schade-Serin V, Collins M, Awada M, Gardner JL, Beyens C, Marino ME, Misell LM et al (2006) Measurement of protein turnover rates by heavy water labeling of nonessential amino acids. *Biochim Biophys Acta* 1760(5):730–744
- Daurio NA, Wang SP, Chen Y, Zhou H, McLaren DG, Roddy TP, Johns DG, Milot D, Kasumov T, Erion MD et al (2017) Enhancing studies of pharmacodynamic mechanisms via measurements of metabolic flux: fundamental concepts and guiding principles for using stable isotope tracers. *J Pharmacol Exp Ther* 363(1):80–91
- Samarel AM (1991) In vivo measurements of protein turnover during muscle growth and atrophy. *FASEB J* 5(7):2020–2028
- DeFronzo RA, Ferrannini E (1987) Regulation of hepatic glucose metabolism in humans. *Diabetes Metab Rev* 3(2):415–459
- DeFronzo RA, Ferrannini E, Hendler R, Wahren J, Felig P (1978) Influence of hyperinsulinemia, hyperglycemia, and the route of glucose administration on splanchnic glucose exchange. *Proc Natl Acad Sci U S A* 75(10):5173–5177
- Previs SF, Brunengraber DZ, Brunengraber H (2009) Is there glucose production outside of the liver and kidney? *Annu Rev Nutr* 29:43–57
- Hundal RS, Krssak M, Dufour S, Laurent D, Lebon V, Chandramouli V, Inzucchi SE, Schumann WC, Petersen KF, Landau BR et al (2000) Mechanism by which metformin reduces glucose production in type 2 diabetes. *Diabetes* 49(12):2063–2069
- Shulman GI, Landau BR (1992) Pathways of glycogen repletion. *Physiol Rev* 72(4):1019–1035

20. Chung ST, Chacko SK, Sunehag AL, Haymond MW (2015) Measurements of gluconeogenesis and glycogenolysis: a methodological review. *Diabetes* 64(12):3996–4010
21. Bessesen DH, Venson SH, Jackman MR (2000) Trafficking of dietary oleic, linolenic, and stearic acids in fasted or fed lean rats. *Am J Physiol Endocrinol Metab* 278(6):E1124–E1132
22. Romanski SA, Nelson RM, Jensen MD (2000) Meal fatty acid uptake in human adipose tissue: technical and experimental design issues. *Am J Physiol Endocrinol Metab* 279(2):E447–E454
23. Patterson BW, Mittendorfer B, Elias N, Satyanarayana R, Klein S (2002) Use of stable isotopically labeled tracers to measure very low density lipoprotein-triglyceride turnover. *J Lipid Res* 43(2):223–233
24. Previs SF, McLaren DG, Wang SP, Stout SJ, Zhou H, Herath K, Shah V, Miller PL, Wilsie L, Castro-Perez J et al (2014) New methodologies for studying lipid synthesis and turnover: looking backwards to enable moving forwards. *Biochim Biophys Acta* 1842(3):402–413
25. Previs SF, Kelley DE (2015) Tracer-based assessments of hepatic anaplerotic and TCA cycle flux: practicality, stoichiometry, and hidden assumptions. *Am J Physiol Endocrinol Metab* 309(8):E727–E735
26. Befroy DE, Perry RJ, Jain N, Dufour S, Cline GW, Trimmer JK, Brosnan J, Rothman DL, Petersen KF, Shulman GI (2014) Direct assessment of hepatic mitochondrial oxidative and anaplerotic fluxes in humans using dynamic (¹³C) magnetic resonance spectroscopy. *Nat Med* 20(1):98–102
27. Hellerstein MK, Christiansen M, Kaempfer S, Kletke C, Wu K, Reid JS, Mulligan K, Hellerstein NS, Shackleton CHL (1991) Measurement of de novo hepatic lipogenesis in humans using stable isotopes. *J Clin Investig* 87(5):1841–1852
28. Beysen C, Ruddy M, Stoch A, Mixson L, Rosko K, Riiff T, Turner SM, Hellerstein MK, Murphy EJ (2018) Dose-dependent quantitative effects of acute fructose administration on hepatic de novo lipogenesis in healthy humans. *Am J Physiol Endocrinol Metab* 315(1):E126–E132
29. McLaren DG, He T, Wang SP, Mendoza V, Rosa R, Gagen K, Bhat G, Herath K, Miller PL, Stribling S et al (2011) The use of stable-isotopically labeled oleic acid to interrogate lipid assembly in vivo: assessing pharmacological effects in preclinical species. *J Lipid Res* 52(6):1150–1161
30. Barrows BR, Timlin MT, Parks EJ (2005) Spillover of dietary fatty acids and use of serum nonesterified fatty acids for the synthesis of VLDL-triacylglycerol under two different feeding regimens. *Diabetes* 54(9):2668–2673
31. Verhoeven NM, Schor DSM, Previs SF, Brunengraber H, Jakobs C (1997) Stable isotope studies of phytanic acid alpha-oxidation: in vivo production of formic acid. *Eur J Pediatr* 156:S83–S87
32. Wang SP, Zhou D, Yao Z, Satapati S, Chen Y, Daurio NA, Petrov A, Shen X, Metzger D, Yin W et al (2016) Quantifying rates of glucose production in vivo following an intraperitoneal tracer bolus. *Am J Physiol Endocrinol Metab* 311(6):E911–E921
33. van Dijk TH, Laskowitz AJ, Grefhorst A, Boer TS, Bloks VW, Kuipers F, Groen AK, Reijngoud DJ (2013) A novel approach to monitor glucose metabolism using stable isotopically labelled glucose in longitudinal studies in mice. *Lab Anim* 47(2):79–88
34. Sun RC, Fan TW, Deng P, Higashi RM, Lane AN, Le AT, Scott TL, Sun Q, Warmoes MO, Yang Y (2017) Noninvasive liquid diet delivery of stable isotopes into mouse models for deep metabolic network tracing. *Nat Commun* 8(1):1646
35. Wolfe RR, Chinkes DL (2005) Isotope tracers in metabolic research: principles and practice of kinetic analyses. Wiley-Liss, Hoboken, NJ
36. Shipley RA, Clark RE (1972) Tracer methods for in vivo kinetics. Theory and applications. Academic, New York
37. Wang SP, Satapati S, Daurio NA, Kelley DE, Previs SF (2017) Reply to letter to the editor: “The art of quantifying glucose metabolism”. *Am J Physiol Endocrinol Metab* 313(2):E259–E261
38. Matthews DE, Downey RS (1984) Measurement of urea kinetics in humans: a validation of stable isotope tracer methods. *Am J Phys* 246(6 Pt 1):E519–E527
39. Ostlund RE Jr, Matthews DE (1993) [¹³C] cholesterol as a tracer for studies of cholesterol metabolism in humans. *J Lipid Res* 34(10):1825–1831
40. Zhou H, Wang SP, Herath K, Kasumov T, Sadygov RG, Previs SF, Kelley DE (2015) Tracer-based estimates of protein flux in cases of incomplete product renewal: evidence and implications of heterogeneity in collagen turnover. *Am J Physiol Endocrinol Metab* 309(2):E115–E121
41. Foster DM, Barrett PH, Toffolo G, Beltz WF, Cobelli C (1993) Estimating the fractional synthetic rate of plasma apolipoproteins and lipids

- from stable isotope data. *J Lipid Res* 34(12):2193–2205
42. Daurio NA, Wang Y, Chen Y, Zhou H, Carballo-Jane E, Mane J, Rodriguez CG, Zafian P, Houghton A, Addona G et al (2019) Spatial and temporal studies of metabolic activity: contrasting biochemical kinetics in tissues and pathways during fasted and fed states. *Am J Physiol Endocrinol Metab* 316(6):E1105–E1117
43. Bederman IR, Dufner DA, Alexander JC, Previs SF (2006) Novel application of the “doubly labeled” water method: measuring CO₂ production and the tissue-specific dynamics of lipid and protein in vivo. *Am J Physiol Endocrinol Metab* 290(5):E1048–E1056
44. Steele R (1971) Tracer probes in steady-state systems. Springfield. Charles C Thomas, Illinois
45. Frayn KN, Coppock SW, Fielding BA, Humphreys SM (1995) Coordinated regulation of hormone-sensitive lipase and lipoprotein lipase in human adipose tissue in vivo: implications for the control of fat storage and fat mobilization. *Adv Enzym Regul* 35:163–178
46. Previs SF, Herath K, Castro-Perez J, Mahsut A, Zhou H, McLaren DG, Shah V, Rohm RJ, Stout SJ, Zhong W et al (2015) Effect of error propagation in stable isotope tracer studies: an approach for estimating impact on apparent biochemical flux. *Methods Enzymol* 561:331–358
47. Melish J, Le NA, Ginsberg H, Steinberg D, Brown WV (1980) Dissociation of apoprotein B and triglyceride production in very-low-density lipoproteins. *Am J Phys* 239(5):E354–E362
48. Chen JL, Peacock E, Samady W, Turner SM, Neese RA, Hellerstein MK, Murphy EJ (2005) Physiologic and pharmacologic factors influencing glyceroneogenic contribution to triacylglyceride glycerol measured by mass isotopomer distribution analysis. *J Biol Chem* 280(27):25396–25402
49. Nye CK, Hanson RW, Kalhan SC (2008) Glyceroneogenesis is the dominant pathway for triglyceride glycerol synthesis in vivo in the rat. *J Biol Chem* 283(41):27565–27574
50. Botion LM, Brito MN, Brito NA, Brito SRC, Kettelhut IC, Migliorini RH (1998) Glucose contribution to in vivo synthesis of glyceride-glycerol and fatty acids in rats adapted to a high-protein, carbohydrate-free diet. *Metabolism* 47(10):1217–1221
51. Botion LM, Kettelhut IC, Migliorini RH (1995) Increased adipose-tissue glyceroneogenesis in rats adapted to a high-protein, carbohydrate-free diet. *Horm Metab Res* 27(7):310–313
52. Ballard FJ, Hanson RW, Leveille GA (1967) Phosphoenolpyruvate carboxykinase and the synthesis of glyceride-glycerol from pyruvate in adipose tissue. *J Biol Chem* 242(11):2746–2750
53. Waterlow JC (2006) Protein turnover. CABI, Oxfordshire
54. Lichtenstein AH, Cohn JS, Hachey DL, Millar JS, Ordovas JM, Schaefer EJ (1990) Comparison of deuterated leucine, valine, and lysine in the measurement of human apolipoprotein A-I and B-100 kinetics. *J Lipid Res* 31(9):1693–1701
55. McCabe BJ, Bederman IR, Croniger CM, Millward CA, Norment CJ, Previs SF (2006) Reproducibility of gas chromatography-niass spectrometry measurements of H-2 labeling of water: application for measuring body composition in mice. *Anal Biochem* 350(2):171–176
56. Annegers J (1954) Total body water in rats and in mice. *Proc Soc Exp Biol Med* 87(2):454–456
57. Brook MS, Wilkinson DJ, Atherton PJ, Smith K (2017) Recent developments in deuterium oxide tracer approaches to measure rates of substrate turnover: implications for protein, lipid, and nucleic acid research. *Curr Opin Clin Nutr Metab Care* 20(5):375–381
58. Strawford A, Antelo F, Christiansen M, Hellerstein MK (2004) Adipose tissue triglyceride turnover, de novo lipogenesis, and cell proliferation in humans measured with 2H₂O. *Am J Physiol Endocrinol Metab* 286(4):E577–E588
59. Krebs HA, Hems R, Weidemann MJ, Speake RN (1966) The fate of isotopic carbon in kidney cortex synthesizing glucose from lactate. *Biochem J* 101(1):242–249
60. Kowalski GM, De Souza DP, Burch ML, Hamley S, Kloehn J, Selathurai A, Tull D, O’Callaghan S, McConville MJ, Bruce CR (2015) Application of dynamic metabolomics to examine in vivo skeletal muscle glucose metabolism in the chronically high-fat fed mouse. *Biochem Biophys Res Commun* 462(1):27–32
61. Kowalski GM, De Souza DP, Risis S, Burch ML, Hamley S, Kloehn J, Selathurai A, Lee-Young RS, Tull D, O’Callaghan S et al (2015) In vivo cardiac glucose metabolism in the high-fat fed mouse: comparison of euglycemic-hyperinsulinemic clamp derived measures of glucose uptake with a dynamic metabolomic flux profiling approach. *Biochem Biophys Res Commun* 463(4):818–824

62. Landau BR, Wahren J, Ekberg K, Previs SF, Yang DW, Brunengraber H (1998) Limitations in estimating gluconeogenesis and Cori cycling from mass isotopomer distributions using [U-C-13(6)]glucose. *Am J Physiol* 37(5):E954–E961
63. Katz J, Chaikoff IL (1955) Synthesis via the Krebs' cycle in the utilization of acetate by rat liver slices. *Biochim Biophys Acta* 18(1):87–101
64. Sidossis LS, Coggan AR, Gastaldelli A, Wolfe RR (1995) Pathway of free fatty acid oxidation in human subjects. Implications for tracer studies. *J Clin Invest* 95(1):278–284
65. Sidossis LS, Coggan AR, Gastaldelli A, Wolfe RR (1995) A new correction factor for use in tracer estimations of plasma fatty acid oxidation. *Am J Phys* 269(4 Pt 1):E649–E656
66. Wolfe RR, Jahoor F (1990) Recovery of labeled CO_2 during the infusion of C-1- vs C-2-labeled acetate: implications for tracer studies of substrate oxidation. *Am J Clin Nutr* 51(2):248–252
67. Toth MJ, MacCoss MJ, Poehlman ET, Matthews DE (2001) Recovery of (13)CO(2) from infused [1-(13)C]leucine and [1,2-(13)C(2)]leucine in healthy humans. *Am J Physiol Endocrinol Metab* 281(2):E233–E241
68. Beysen C, Murphy EJ, McLaughlin T, Rüff T, Lamendola C, Turner HC, Awada M, Turner SM, Reaven G, Hellerstein MK (2007) Whole-body glycolysis measured by the deuterated-glucose disposal test correlates highly with insulin resistance in vivo. *Diabetes Care* 30(5):1143–1149
69. Raman A, Blanc S, Adams A, Schoeller DA (2004) Validation of deuterium-labeled fatty acids for the measurement of dietary fat oxidation during physical activity. *J Lipid Res* 45(12):2339–2344
70. Votruba SB, Zeddun SM, Schoeller DA (2001) Validation of deuterium labeled fatty acids for the measurement of dietary fat oxidation: a method for measuring fat-oxidation in free-living subjects. *Int J Obes Relat Metab Disord* 25(8):1240–1245
71. Landau BR, Wahren J (1992) Nonproductive exchanges: the use of isotopes gone astray. *Metabolism* 41(5):457–459
72. Ramakrishnan R (2006) Studying apolipoprotein turnover with stable isotope tracers: correct analysis is by modeling enrichments. *J Lipid Res* 47(12):2738–2753
73. Cobelli C, Toffolo G, Foster DM (1992) Tracer-to-tracee ratio for analysis of stable isotope tracer data: link with radioactive kinetic formalism. *Am J Phys* 262(6 Pt 1):E968–E975
74. Chinkes DL, Aarsland A, Rosenblatt J, Wolfe RR (1996) Comparison of mass isotopomer dilution methods used to compute VLDL production in vivo. *Am J Phys* 271(2 Pt 1):E373–E383
75. Kharroubi AT, Masterson TM, Aldaghlas TA, Kennedy KA, Kelleher JK (1992) Isotopomer spectral analysis of triglyceride fatty acid synthesis in 3T3-L1 cells. *Am J Phys* 263(4 Pt 1):E667–E675
76. Previs SF, Hallowell PT, Neimanis KD, David F, Brunengraber H (1998) Limitations of the mass isotopomer distribution analysis of glucose to study gluconeogenesis—heterogeneity of glucose labeling in incubated hepatocytes. *J Biol Chem* 273(27):16853–16859



Chapter 7

Extracting Biological Insight from Untargeted Lipidomics Data

Jennifer E. Kyle

Abstract

Lipidomics data generated using untargeted mass spectrometry techniques can offer great biological insight to metabolic status and disease diagnoses. As the community's ability to conduct large-scale studies with deep coverage of the lipidome expands, approaches to analyzing untargeted data and extracting biological insight are needed. Currently, the function of most individual lipids are not known; however, meaningful biological information can be extracted. Here, I will describe a step-by-step approach to identify patterns and trends in untargeted mass spectrometry lipidomics data to assist users in extracting information leading to a greater understanding of biological systems.

Key words Lipidomics, Lipidome, Untargeted, Mass spectrometry, LipidMaps, Blood plasma

1 Introduction

Lipidomics is the study of lipids within a cell or organisms to understand the function and structure of biological systems. Lipids are highly diverse biomolecules that have many roles critical to living systems, including acting as membrane structural components, signaling molecules, and energy sources [1]. The field of lipidomics has been increasingly recognized in the past few years [2] due to the advances of mass spectrometry techniques, and biological insight gained through lipidomics measurements. Lipids have been shown to be important in cellular homeostasis [3, 4], signatures of disease [5–7], and affected by inborn errors in metabolism [8]. Even with these advancements the exact function of most individual lipids is not known. For untargeted lipidomics analyses, a few hundred individual lipids may be identified within a biological matrix [9, 10]. This leads to many questions such as how the cell utilizes lipids for their critical functions [11] and how cellular homeostasis may be affected by alterations in the lipid profile. The interpretation of lipids is currently focused on known

roles of the lipid components (e.g., fatty acids) or the type of lipids (e.g., triglyceride) identified.

Multiple review papers on sample collection to storage procedures, lipid extraction from biological matrices, mass spectrometry technologies and methods, appropriate QCs as well as software are available [2, 12–16] and will not be covered here. In this chapter, I will provide an approach to exploring lipidomics data at the global level (i.e., all lipids identified in the biological study) to help guide the understanding and interpretation of the lipidome. The integration of other “omics” data, such as proteomics and transcriptomics, can aid in interpretation of lipidomics data as it enables mechanistic support for lipidomics observations.

2 Lipid Classification and Annotation

Lipids are highly structural diverse biomolecules mainly due to the various combinations of fatty acids, fatty acid linkages, and head groups. Despite this great diversity, lipids have two basic biochemical building blocks, ketoacyl or isoprene [17]. LipidMaps has played an important role in the field of lipidomics through standardizing the classification and annotation scheme of lipid nomenclature. LipidMaps has categorized lipids into eight main categories [18], three of which (sphingolipids (SL), glycerophospholipids (GP), and glycerolipids (GL)) are routinely identified and characterized using total lipid extraction methods [19–21] and untargeted mass spectrometry for human health studies. Each lipid category is further divided using a subclassification hierarchy including lipid main classes and lipid subclasses [18]. This chapter will focus on the lipid categories SL, GP, and GL.

Most of the lipids that are SL, GP, and GL contain a common name that utilizes a specific annotation Scheme ZZ(X1:Y1/X2:Y2):

ZZ(X1:Y1/X2:Y2) → example PC(16:0/20:4)

ZZ = lipid class (e.g., PC)

X1 = number of carbons in chain 1 (e.g., 16)

Y1 = number of double bonds in chain 1 (e.g., 0)

X2 = number of carbons in chain 2 (e.g., 20)

Y2 = number of double bonds in chain 2 (e.g., 4)

This scheme enables the lipid common name to be used for identifying patterns and trends in untargeted lipidomics data. Additional annotations within the parentheses can provide further subclass differentiation; for example, a “P” in PE(P-16:0/20:4) reveals the PE lipid is plasmalogen (or alkenyl chain), and absence of a chain as 0:0 in PC(0:0_16:0) reveals the lipid is a monoacylglycerophosphocholine (LPC) or lysoPC.

Recently, the annotation scheme has become more refined in that the “/” now designates the sn position of the chains is known, whereas “_” has been suggested to indicate the sn position is not

known [22]. This change is slowly becoming incorporated in the lipidomics literature.

The LipidMaps annotation scheme has not been universally adopted as lipid annotation prior to LipidMaps continues to be used (e.g., PtdPC vs. PC). However, the abbreviations of lipids are similar and therefore can still be parsed as detailed below.

3 Parsing the Lipid Common Name for Lipidomics Pattern Discovery

One of the difficulties with interpreting lipidomics data is that the specific functions for most lipids are not known nor is there an inclusive database for those with known roles. Biological knowledge of lipids exists at the subclass level [23] and the fatty acid level (e.g., arachidonic acid (AA), saturated fatty acids, etc.), but knowledge of the intact or complete lipid itself is sparse [24, 25] compared to the number of unique lipids in biological systems [17]. Since the role of individual components of lipids is better understood, the common name of most SLs, GPs, and GLs can be used to gather information to begin initial steps toward biological interpretation of lipidomics data (Fig. 1).

Recently, a tool of Lipid Mini-On (*Mining and Ontology*) was developed to parse the lipid common names into ontology bins corresponding to its components (e.g., classification, chain composition, chain characteristics) [26]. The tool further performs enrichment analysis on those components and assigns a significance value. More details are provided in Subheading 4.2 step 4.

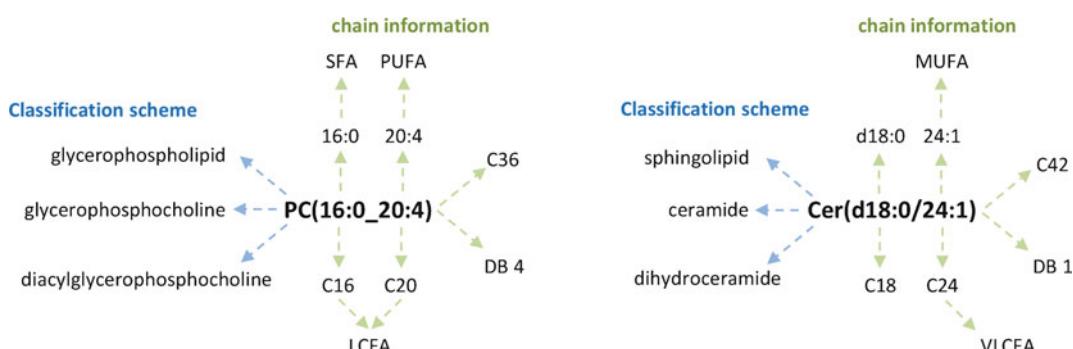


Fig. 1 Classification information and specific chain details can be gathered from the lipid common name. For example, PC(16:0_20:4), the PC reveals the type of lipid (i.e., classification scheme) while the fatty acids (FAs) can be parsed at the fatty level (16:0 and 20:4), the FA characteristics (e.g., 16 carbons long (C16)), and also by the total number of carbons and double bonds of the chains (i.e., 36 carbons (C36) and 4 double bonds (DB 4)). The FAs themselves include a saturated FA (SFA) and polyunsaturated FA (PUFA) and the chain lengths show they are both long chain FAs (LCFAs). A second example using Cer(d18:0/24:1) can be similarly parsed. MUFA monounsaturated FA, VLCFA very long chain FA

4 Approach to Pattern Discovery and Understanding the Lipidome

Using the following approach, you will see that lipids tend to have patterns based on their classification, both at the inter- and intra-subclass level as well as chain composition or chain characteristics. Here I present my method using two lipidomics studies, cell-sorted human lung tissue [27] and human plasma from donors with Ebola virus disease [28]. This approach applies to lipidomics studies where confident identifications have been made and statistical analysis has provided results on differences (e.g., log₂ fold change or Zscore) between groups (e.g., controls and diseased samples). This approach involves:

1. Understanding the lipids within the sample biological matrix (Subheading 4.1).
2. Identifying signatures and patterns in lipidomics data (Subheading 4.2).
3. Integrating ‘omics data (Subheading 4.3).

4.1 Understanding the Lipids Within the Sample Biological Matrix

Understanding how lipids influence membranes and where lipids are localized at the cellular level or within biofluids will aid in the interpretation of the lipidome. Several excellent reviews on the cellular lipid subclass distribution and fatty acid characteristics are available [4, 11, 29, 30]. These reviews detail the influence of the phospholipid head groups and fatty acid composition on membrane composition, fluidity, width, and charge. For example, the small head group of PE lipids induces a negative membrane curvature, and unsaturated lipids lead to more fluid membranes. Very few lipid classes in human or mammalian systems are organelle specific. Exceptions include cardiolipin which is located mostly in the inner mitochondrial membrane and bis(monoacylglycerophosphomonoacylglycerol) (BMP; also known as monoacylglycerophosphomonoacylglycerol (LBPA)) located in late endosome and lysosomes.

Less information is available on the localization of lipids in biological fluids. Biofluids are particularly complicated due to both endogenous and exogenous sources of lipids. Lipids have been identified in many biofluids [31]; however, given their hydrophobic nature they are, for the most part, present in low abundance. In plasma and CSF, lipids are primarily associated with lipoproteins [32–34]. In CSF, lipids are also associated with brain-derived nanoparticles [35]. In urine, the presence of lipids has been suggested to be due to urea facilitating the dissolution of the lipids in urine [36].

Most lipidomics studies using biofluids are conducted using blood plasma where lipids are abundant. Lipids in blood plasma are associated with lipoprotein membranes (e.g., PC and sphingomyelin (SM)), lipoprotein cargo (e.g., triglycerides (TGs) and cholesterol ester), and proteins (e.g., lyso-PC). A handful of publications

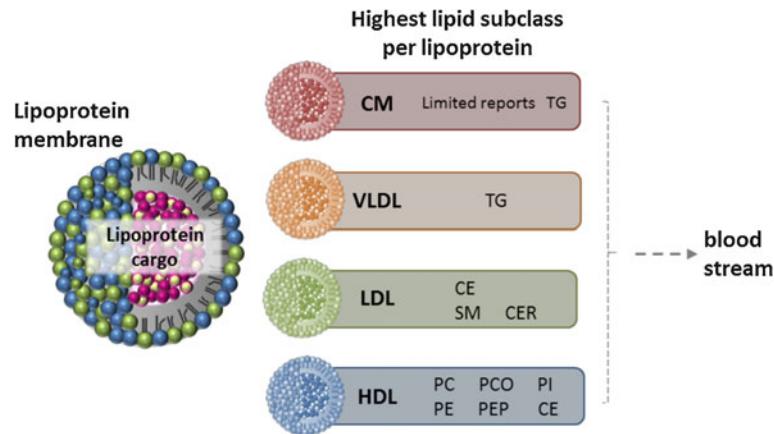


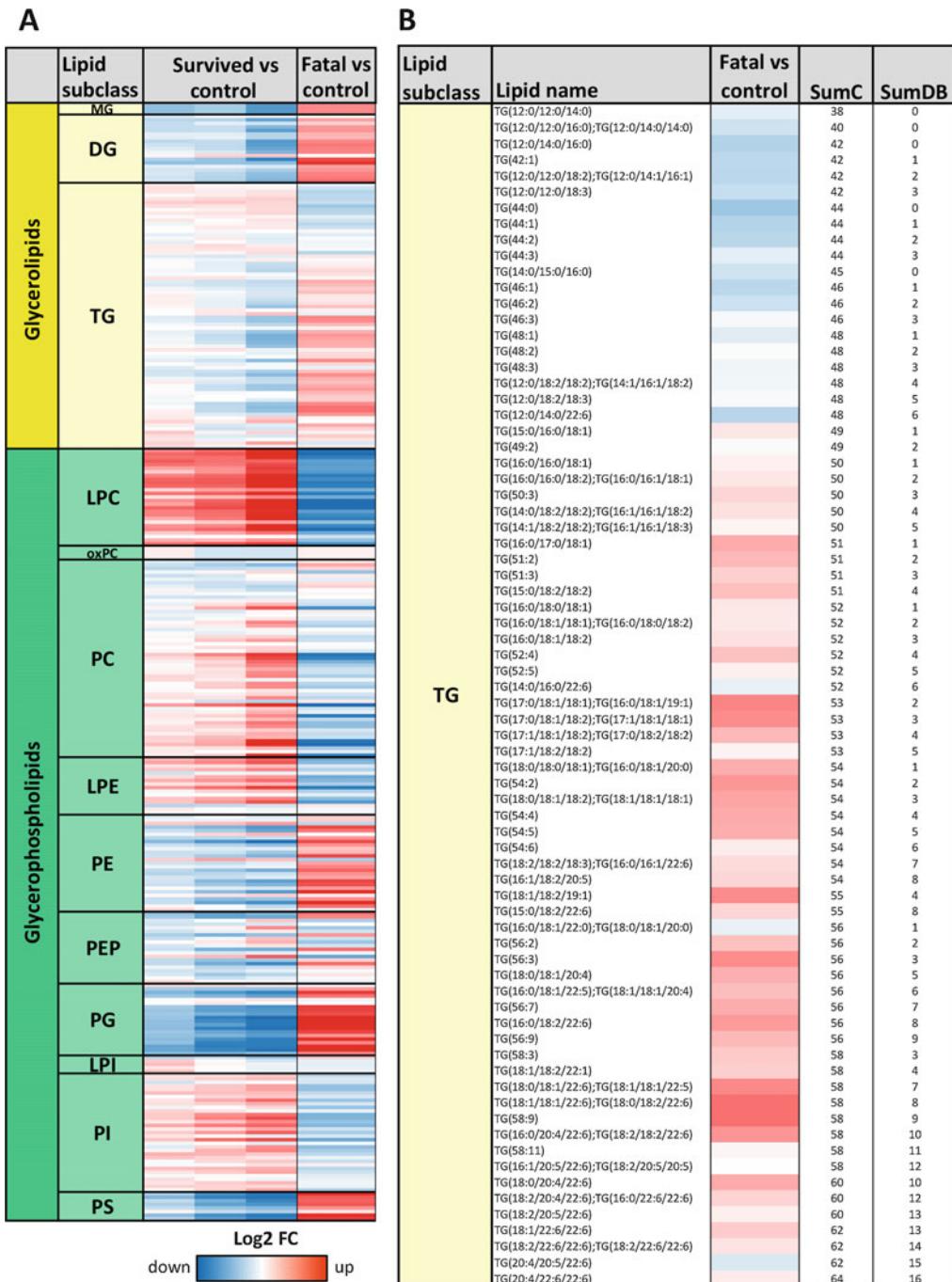
Fig. 2 Summary of lipid subclass distribution per lipoprotein type in human plasma. Lipoproteins contain an outer lipid membrane and transport lipids (lipoprotein cargo), which make up most of the lipids identified in blood plasma. These subclass associations may be helpful in interpreting the source of lipid subclass trends in lipidomics data for human plasma

have characterized the lipid profile of the different types of lipoproteins [37–41] revealing which lipid subclass is highest in a particular lipoprotein (Fig. 2). For example, ceramides (Cer) and sphingomyelins (SM) are most abundant in low-density lipoproteins (LDL) and diacylglycerophosphoethanolamine (PE) in high-density lipoproteins (HDL) [37, 40, 41]. These same studies also reveal that phospholipids and sphingolipids are present in all lipoproteins, but the abundance of the lipids varies. Associating this knowledge with a plasma can be difficult, in particular due to the multiple factors that can influence the lipid abundance including diet [42], gender [43], and circadian rhythms [44]. In addition to lipoproteins, human bodily fluids also contain extracellular vesicles, which are thought to contribute to physiology and pathology, increasing or decreasing in abundance in diseased states [45] and possibly contributing to lipidome signature.

4.2 Identifying Signatures and Patterns in Lipidomics Data

To understand the lipidome, one must identify the trends, patterns, and unique signatures. Organizing the lipid identifications with associated abundance values in the following manner will allow for observations to be gathered at the global level as well as inter- and intrasubclass levels. Perform the steps 1–4 with both the global results (i.e., results from all of the identified lipids) as well as the results from your query of interest (e.g., statistically significant lipids)

Step 1. Organize the statistical results file based on the LipidMaps classification scheme. This will highlight global trends as well as trends across lipid categories and subclasses (Fig. 3a).



Increasing number of total chain carbons and double bonds

Fig. 3 Organization of lipidome of human Ebola virus disease. **(a)** The lipids are organized by category (glycerolipids and glycerophospholipids), subclass, and then intrasubclass by the total number of chain carbons and double bonds. **(b)** Detailed intrasubclass organization of TGs. MG monoacylglyceride, DG diacylglyceride, TG triacylglyceride, LPC monoacylglycerophosphocholine, oxPC oxidized diacylglycerophosphocholine, PC diacylglycerophosphocholine, LPE monoacylglycerophosphoethanolamine, PE diacylglycerophosphoethanolamine, PEP plasmalogen PE, PG diacylglycerophosphoglycerol OR bis(monoacylglycerol)phosphate, LPI PI monoacylglycerophosphoinositol, PI diacylglycerophosphoinositol, PS diacylglycerophosphoserine. Data used to generate the figure is from [28] (Fig. 2) from the original Supplemental Table S2 of publication [28]

- (a) Lipid category
e.g., Glycerophospholipids,
- (b) Lipid main class
e.g., Glycerophosphocholines,
- (c) Lipid subclass
e.g., monoacylglycerophosphocholine (LPC), diacylglycerophosphocholine (PC), alkylacylglycerophosphocholine (PCO).

Step 2. Organize within each subclass by the sum of the total number of chain carbons (SumC) and then the sum of the total number of chain double bonds (SumDB). This will highlight intrasubclass trends (Fig. 3b).

- (a) This organization largely results in shorter chained carbons with no or low number of double bonds at the top and longer chained polyunsaturated chains at the bottom.
- (b) LipidSplit can calculate SumC and SumDB values and is freely available (<https://github.com/PNNL-Comp-Mass-Spec/LIPID-Split>)

Note: For sphingolipids and ether linked phospholipids also sort alphabetically as intrasubclass trends are not as common as with phospholipids and glycerolipids or based on the long chain base or ether lipid chain. For example, organizing ceramides alphabetically groups the long chain base by sphinganine (d18:0) then sphingosine (d18:1) and plasmalogens will organize by P-16:0 then P-18:0.

Step 3. Examine the fatty acid composition at both the intersubclass and intrasubclass level. Lipids with particular fatty acids, commonly polyunsaturated fatty acids (PUFA), may have a different pattern than the other lipids. In the example presented in Fig. 4, all PI lipids containing 20:4 tend to increase, whereas other lipids in that subclass tend to decrease.

Step 4. Perform enrichment analysis. Lipid Mini-On performs enrichment analyses using the lipid common name annotation and associated classification to highlight ontology terms within a given list of lipids [26]. The tool examines the data as outlined above such that it determines if lipids are enriched in a statistically significant way based on their classification (e.g., lipid category, main class, and subclass) and also inter- and intraclassification by the chain characteristics (e.g., fatty acids themselves and fatty acid length and number of double bonds) (see Fig. 1). The query list is compared to the list of all lipids identified in

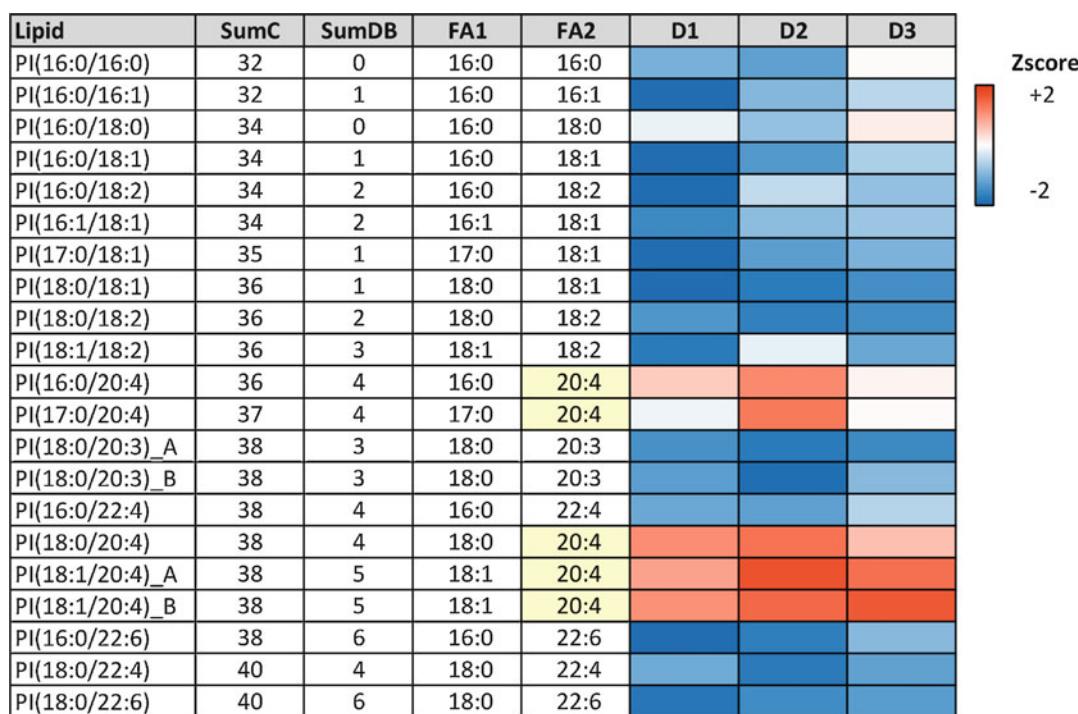


Fig. 4 Intrasubclass differences based on the fatty acid composition. PI lipids containing the fatty acid 20:4 were high (positive Zscore) in the endothelial cells of lung tissues of three 20-month-old donors [27], whereas the rest of the PIs were low (negative Zscore). Data used to generate this figure is from [27] from the original Supplemental Table S4 [27]

the associated study. Each classifier (or ontology term) is assigned a p-value, enabling the identification of global trends in the lipidome (Table 1). The tool also displays the classifiers as nodes in a network along with the associated lipids such that the user can see the exact lipids in the enriched ontology term (Fig. 5).

4.3 Integrating ‘Omics Data

The interaction of lipids with other cellular components, including proteins, genes, and metabolites, can enable a greater understanding of the mechanisms behind the observations in your lipidomics data generated in steps 1–4 above. LipidMaps protein database contains thousands of entries across ten model organisms that contain protein and gene identifiers for proteins and genes associated with lipid metabolism and homeostasis. For humans, there are 2273 entries with over 1100 unique entrez gene and gene symbols, uniprot identifications, and protein entries. To identify proteins or genes that may have a role in the lipid signatures noted in Subheading 4.2, perform the following steps with your associated proteomics (and/or transcriptomics) results.

Table 1

Lipid category and subclass enrichment in Ebola virus disease [28], using Fisher's exact test with *p*-value < 0.05

EVD— increased in fatalities vs controls		Test. performed	Classifier	Count. query	Count. universe	%. query	%. universe	<i>p</i> - value	<i>q</i> - value	FDR. Fold. change
Category	Glycerolipid			43/ 115	94/379	37.4	24.8	0.012	0.037	1.5
Sub class	DG(16/ 115	19/379	13.9	5.0	0.003	0.008	2.8
Sub class	PE(20/ 115	27/379	17.4	7.1	0.002	0.008	2.4
Sub class	PG(17/ 115	20/379	14.8	5.3	0.002	0.008	2.8
Total chain carbon by all	Glycerophospholipid with a total number of chain carbon of 36			18/ 51	42/212	35.3	19.8	0.025	0.225	1.8
Total chain carbon by all	PC with a total number of chain carbon of 32			1/1	3/83	100.0	3.6	0.048	1.000	27.7
Specific chains by all	PC(with the chain 16:0			2/2	15/110	100.0	13.6	0.022	1.000	7.3
Total number of DB by all	SM(d with a total number of chain unsaturation of 0			2/3	2/28	66.7	7.1	0.037	0.074	9.3

- Step 1. Download LipidMaps protein database (<http://www.lipidmaps.org/data/proteome/download.php>).
- Step 2. Select the entries for the model organism of interest.
e.g., *Homo sapiens*,
- Step 3. Choose the molecular identifier that aligns with your 'omics results and check for (and remove, if appropriate) duplicate identifier values. Also, copy the gene_name column, as this provides some initial details on the function of the identifier.
e.g., protein_entry,
- Step 4. Copy the molecular identifier chosen in **step 3** and determine which lipid associated entries were detected in your proteomics data.

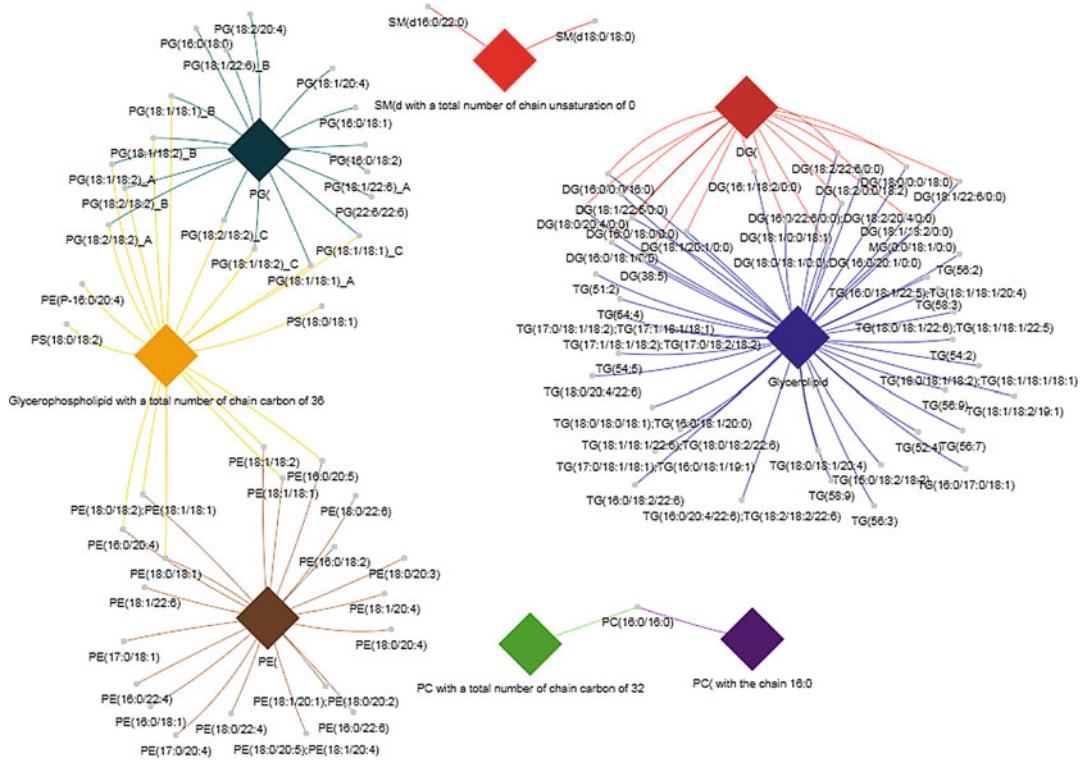


Fig. 5 Lipid enrichment network result performed using Lipid Mini-On. The network shows statistically significant ontology terms (diamond shapes) and the individual lipids associated with those terms were found to be enriched in human Ebola virus disease fatalities versus controls. The network was produced from an enrichment test using a Fisher's Exact test with a *p*-value cutoff of <0.05. The query list contained lipids that were statistically elevated in fatalities versus the controls . Data used for this analysis is from [28] from the original Supplemental Table S2 of publication [28]

- If using Excel, highlight both the LipidMaps column and the column with your data and use the highlight “duplicate values” function. Then organize your data columns by cell color or font to group lipid associated proteins (Fig. 6).
 - At this step you will have a list of all lipid-associated proteins (and/or genes) in your ‘omics data.
- To obtain information on the protein(s) or gene (s) of interest, search for the identifier in protein or gene databases such as Uniprot (<https://www.uniprot.org/>) or GeneCards (<https://www.genecards.org/>).
- Examine the proteins with functions related to the lipidomics results from Subheading 4.2 (Fig. 7)

Step 5. To associate the proteins (or genes) to pathways, copy the list of lipid associated proteins (genes) into DAVID

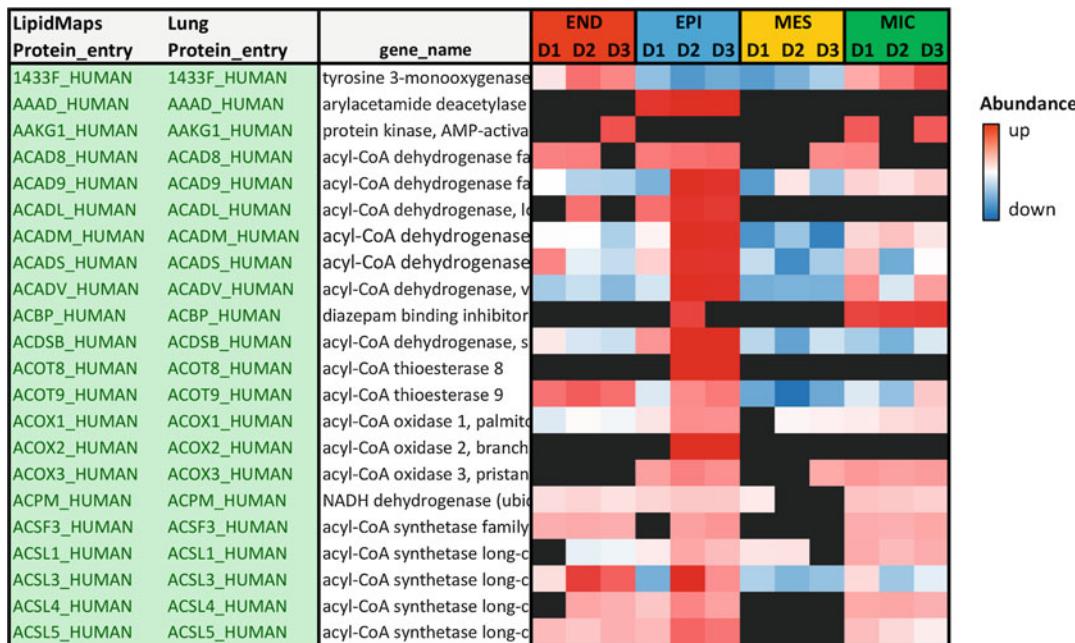


Fig. 6 Identifying lipid-associated proteins in proteomics data. Proteomics data from a human lung study [27] was compared against the LipidMaps protein database using the protein_entry identifier. Matching protein_entry identifiers were highlighted in green. The gene_name was also tracked for the protein (gene) descriptor. The proteomics data shows lipid-associated proteins identified in three donors (D1, D2, and D3) of sorted endothelial cells (END), epithelial cells (EPI), mesenchymal cells (MES), and immune cells (MIC) from the lung tissue. Proteins that are higher in the associated cell type are in red, and those that are down are in blue. Black cells indicate that the protein was not identified in that cell type for the associated donor. Proteomics data presented in this figure is from [27] from the original Supplemental Table S2 of publication [27].

[46, 47] or Pathview [48, 49] (other software tools and codes are available [16, 50]).

- DAVID—proteins/genes associated with pathways. Below is one example:
 - Go to <https://david.ncifcrf.gov/>
 - Click on “Functional Annotation”,
 - Copy and paste list of lipid-associated proteins,
 - Select identifier used in **step iii**,
 - Select “Gene List” as list type,
 - Click “Submit List”,
 - Check species is correct (*Homo sapiens* is default),
 - Click on “Pathways”,
 - Explore options provided,

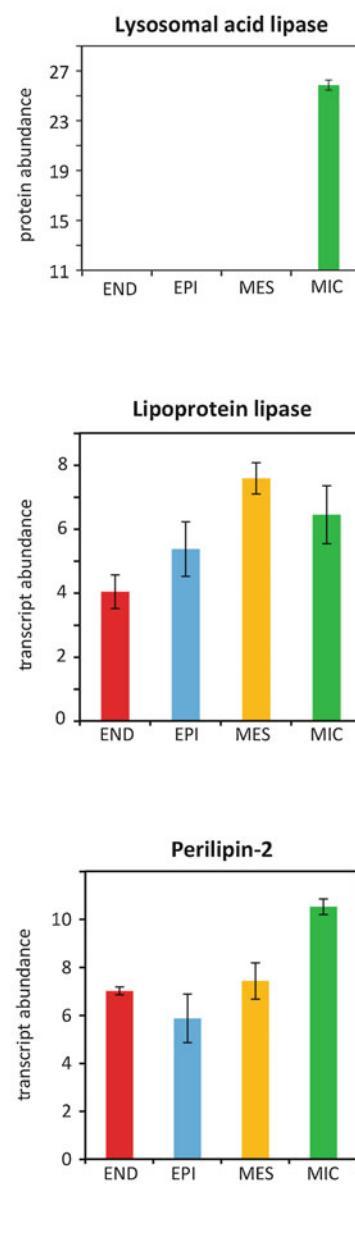
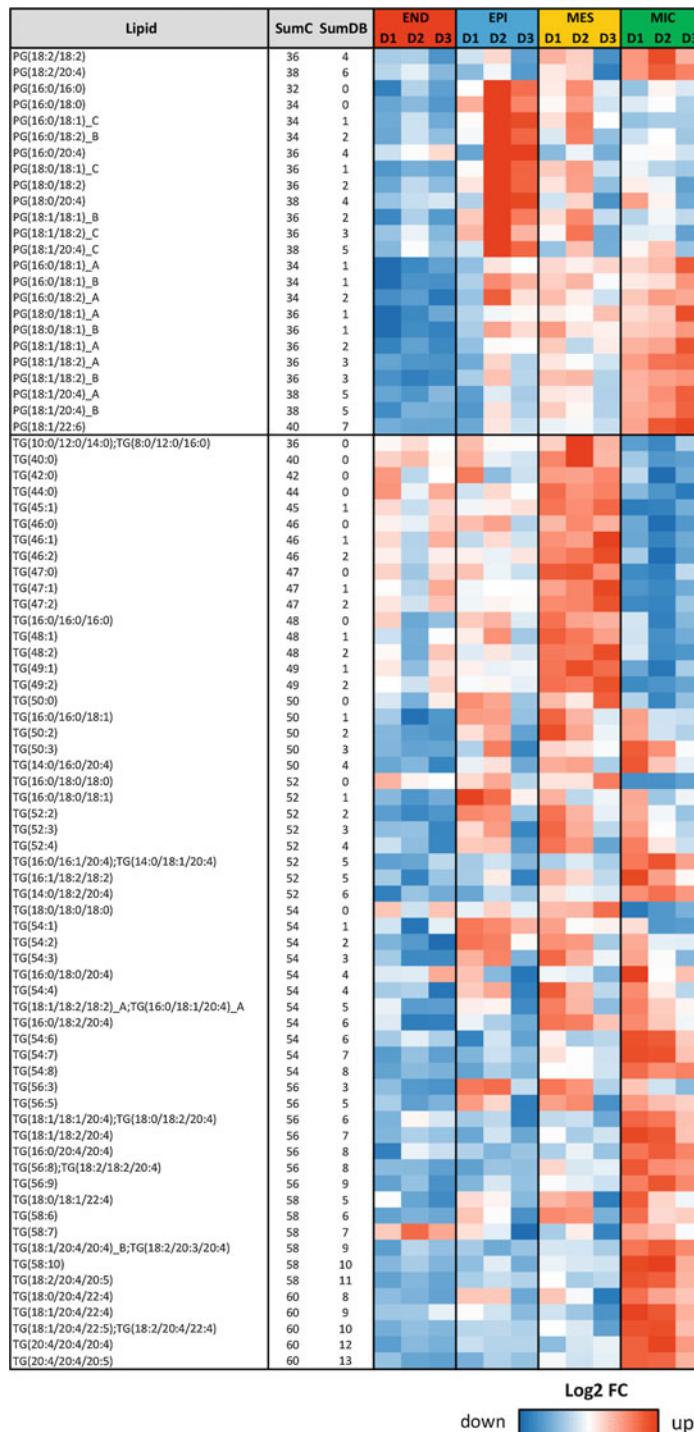


Fig. 7 Integrating lipidomics, proteomics, and transcriptomics data to understand the lung lipidome. Lipidomics analysis of cell-sorted lung tissue from 3 20-month-old female donors revealed intrasubclass trends for PG and TG lipids [27]. For the PGs, lipids with shorter retention times elevated in immune (MIC) cells and were found to be BMP lipids [27], which are known to be associated with late endosomes and lysosomes.

- (b) Pathview—visualize proteins (or genes) with abundance information on pathways
 - (i) Go to <https://pathview.uncc.edu/analysis>
 - (ii) Upload gene data by clicking on “choose file”,
 - Pathview offers examples of data format and analyses to the right of the file upload and selection area.
 - (iii) Select the options and pathways aligning with your data (e.g., sphingolipid metabolism),
 - (iv) Click “Submit”,
 - (v) Review results and download as needed.

5 Summary

In this chapter, I discussed and provided an approach of using the lipid common name to gather information and understanding of lipidomics data. In addition, steps for the integration of other ‘omics data to assist with the interpretation of the patterns identified in the lipidomics data was provided. Given most individual lipids do not have a known specific function and the lack of pathways or networks for specific lipids, the goal of this chapter is to provide the reader with a step-by-step approach to analyze their lipidomics data for extracting biological information leading to interpretation.

- Step 1. Organize the statistical results file based on LipidMaps classification scheme.
- Step 2. Organize within each subclass by the sum of the total number of chain carbons (SumC) and then the sum of the total number of chain double bonds (SumDB).
- Step 3. Examine the fatty acid composition at both the intersubclass and intrasubclass level.
- Step 4. Perform enrichment analysis.

Fig. 7 (continued) Proteomics supported the presence of BMPs in MIC cells, as lysosomal acid lipase, a protein associated with lysosomes, was only identified in MIC cells. Protein abundance is LFQ intensity. For TGs, organization of the lipids as shown in Subheading 4.2 step 2 revealed that TGs with shorter total chain carbons and no or low number of total double bonds showed elevated levels in mesenchymal cells (MES), whereas longer chained, more polyunsaturated fatty acids showed elevated levels in the MIC cells. Examining transcriptomics data revealed that lipoprotein lipase, which hydrolyzes TGs, was expressed the greatest in MES cells, and perilipin-2, which is associated with lipid droplets, was the greatest in MIC cells. Transcript values are log 2 abundances. *END* endothelial cells, *EPI* epithelial cells, three lung tissue donors (D1, D2, and D3). Lipidomics data presented in this figure is modified from [27] from the original Supplemental Table S4 of publication [28]. Protein and transcript graphs were modified from the supplemental Fig. S1 and S2 from the publication [27]

Step 5. Integrate other ‘omics data.

6 Data Deposition

All mass spectrometry datasets generated from the human Ebola virus disease study were previously published [51] and deposited in Mass Spectrometry Interactive Virtual Environment (MassIVE) at the University of California at San Diego under the code MSV000080129. The cell-sorted lung lipidome data was previously published [27] and the data deposited at MassIVE under the code MSV000081973.

Acknowledgments

I would like to thank Jeremy Clair and Ernesto S. Nakayasu for their comments and careful review of the manuscript. This work was supported by an administrative supplement to grant U19AI106772, provided by the National Institute of Allergy and Infectious Diseases (NIAID), National Institute of Health (NIH) (USA). Research conducted on the lung samples were supported by grant HL122703 from the National Heart Lung Blood Institute of NIH.

References

1. Gross RW, Han X (2011) Lipidomics at the interface of structure and function in systems biology. *Chem Biol* 18(3):284–291. <https://doi.org/10.1016/j.chembiol.2011.01.014>
2. Rustam YH, Reid GE (2018) Analytical challenges and recent advances in mass spectrometry based lipidomics. *Anal Chem* 90(1):374–397. <https://doi.org/10.1021/acs.analchem.7b04836>
3. Agmon E, Stockwell BR (2017) Lipid homeostasis and regulated cell death. *Curr Opin Chem Biol* 39:83–89. <https://doi.org/10.1016/j.cbpa.2017.06.002>
4. Holthuis JC, Menon AK (2014) Lipid landscapes and pipelines in membrane homeostasis. *Nature* 510(7503):48–57. <https://doi.org/10.1038/nature13474>
5. Hilvo M, Denkert C, Lehtinen L, Muller B, Brockmoller S, Seppanen-Laakso T, Budczies J, Bucher E, Yetukuri L, Castillo S, Berg E, Nygren H, Sysi-Aho M, Griffin JL, Fiehn O, Loibl S, Richter-Ehrenstein C, Radke C, Hyotylainen T, Kallioniemi O, Iljin K, Oresic M (2011) Novel theranostic opportunities offered by characterization of altered membrane lipid metabolism in breast cancer progression. *Cancer Res* 71(9):3236–3245. <https://doi.org/10.1158/0008-5472.can-10-3894>
6. Lydic TA, Goo YH (2018) Lipidomics unveils the complexity of the lipidome in metabolic diseases. *7(1):4*. <https://doi.org/10.1186/s40169-018-0182-9>
7. Zhao YY, Miao H, Cheng XL, Wei F (2015) Lipidomics: novel insight into the biochemical mechanism of lipid metabolism and dysregulation-associated disease. *Chem Biol Interact* 240:220–238. <https://doi.org/10.1016/j.cbi.2015.09.005>
8. Lamari F, Mochel F, Saudubray JM (2015) An overview of inborn errors of complex lipid biosynthesis and remodelling. *J Inherit Metab Dis* 38(1):3–18. <https://doi.org/10.1007/s10545-014-9764-x>
9. Dautel SE, Kyle JE, Clair G, Sontag RL, Weitz KK, Shukla AK, Nguyen SN, Kim YM, Zink EM, Luders T, Frevert CW, Gharib SA, Laskin J, Carson JP, Metz TO, Corley RA, Ansong C (2017) Lipidomics reveals dramatic lipid compositional changes in the maturing

- postnatal lung. *Sci Rep* 7:40555. <https://doi.org/10.1038/srep40555>
10. Quehenberger O, Armando AM, Brown AH, Milne SB, Myers DS, Merrill AH, Bandyopadhyay S, Jones KN, Kelly S, Shaner RL, Sullards CM, Wang E, Murphy RC, Barkley RM, Leiker TJ, Raetz CR, Guan Z, Laird GM, Six DA, Russell DW, McDonald JG, Subramaniam S, Fahy E, Dennis EA (2010) Lipidomics reveals a remarkable diversity of lipids in human plasma. *J Lipid Res* 51(11):3299–3305. <https://doi.org/10.1194/jlr.M009449>
 11. van Meer G, de Kroon AI (2011) Lipid map of the mammalian cell. *J Cell Sci* 124(Pt 1):5–8. <https://doi.org/10.1242/jcs.071233>
 12. Hu T, Zhang JL (2018) Mass-spectrometry-based lipidomics. *J Sep Sci* 41(1):351–372. <https://doi.org/10.1002/jssc.201700709>
 13. Hyotylainen T, Oresic M (2015) Optimizing the lipidomics workflow for clinical studies—practical considerations. *Anal Bioanal Chem* 407(17):4973–4993. <https://doi.org/10.1007/s00216-015-8633-2>
 14. Hyotylainen T, Oresic M (2016) Bioanalytical techniques in nontargeted clinical lipidomics. *Bioanalysis* 8(4):351–364. <https://doi.org/10.4155/bio.15.244>
 15. Kyle JE, Crowell KL, Casey CP, Fujimoto GM, Kim S, Dautel SE, Smith RD, Payne SH, Metz TO (2017) LIQUID: an-open source software for identifying lipids in LC-MS/MS-based lipidomics data. *Bioinformatics* 33(11):1744–1746. <https://doi.org/10.1093/bioinformatics/btx046>
 16. Misra BB, Mohapatra S (2019) Tools and resources for metabolomics research community: a 2017–2018 update. *Electrophoresis* 40(2):227–246. <https://doi.org/10.1002/elps.201800428>
 17. Fahy E, Subramaniam S, Murphy RC, Nishijima M, Raetz CR, Shimizu T, Spener F, van Meer G, Wakelam MJ, Dennis EA (2009) Update of the LIPID MAPS comprehensive classification system for lipids. *J Lipid Res* 50(Suppl):S9–S14. <https://doi.org/10.1194/jlr.R800095-JLR200>
 18. Fahy E, Subramaniam S, Brown HA, Glass CK, Merrill AH Jr, Murphy RC, Raetz CR, Russell DW, Seyama Y, Shaw W, Shimizu T, Spener F, van Meer G, VanNieuwenhze MS, White SH, Witztum JL, Dennis EA (2005) A comprehensive classification system for lipids. *J Lipid Res* 46(5):839–861. <https://doi.org/10.1194/jlr.E400004-JLR200>
 19. Folch J, Lees M, Sloane Stanley GH (1957) A simple method for the isolation and purification of total lipides from animal tissues. *J Biol Chem* 226(1):497–509
 20. Bligh EG, Dyer WJ (1959) A rapid method of total lipid extraction and purification. *Can J Biochem Physiol* 37(8):911–917. <https://doi.org/10.1139/o59-099>
 21. Matyash V, Liebisch G, Kurzchalia TV, Shevchenko A, Schwudke D (2008) Lipid extraction by methyl-tert-butyl ether for high-throughput lipidomics. *J Lipid Res* 49(5):1137–1146. <https://doi.org/10.1194/jlr.D700041-JLR200>
 22. Liebisch G, Vizcaino JA, Kofeler H, Trottmüller M, Griffiths WJ, Schmitz G, Spener F, Wakelam MJ (2013) Shorthand notation for lipid structures derived from mass spectrometry. *J Lipid Res* 54(6):1523–1530. <https://doi.org/10.1194/jlr.M033506>
 23. Han X (2016) Lipidomics for studying metabolism. *Nat Rev Endocrinol* 12(11):668–679. <https://doi.org/10.1038/nrendo.2016.98>
 24. Grosch S, Schiffmann S, Geisslinger G (2012) Chain length-specific properties of ceramides. *Prog Lipid Res* 51(1):50–62. <https://doi.org/10.1016/j.plipres.2011.11.001>
 25. Veldhuizen R, Nag K, Orgeig S, Possmayer F (1998) The role of lipids in pulmonary surfactant. *Biochim Biophys Acta* 1408(2–3):90–108
 26. Clair G, Reehl S, Stratton KG, Monroe ME, Tfaily MM, Ansong C, Kyle JE (2019) Lipid Mini-On: mining and ontology tool for enrichment analysis of lipidomic data. *Bioinformatics* 35(2):4507–4508. <https://doi.org/10.1093/bioinformatics/btz250>
 27. Kyle JE, Clair G, Bandyopadhyay G, Misra RS, Zink EM, Bloodsworth KJ, Shukla AK, Du Y, Lillis J, Myers JR (2018) Cell type-resolved human lung lipidome reveals cellular cooperation in lung function. *Sci Rep* 8(1):13455. <https://doi.org/10.1038/s41598-018-31640-x>
 28. Kyle JE, Burnum-Johnson KE (2019) Plasma lipidome reveals critical illness and recovery from human Ebola virus disease. *Proc Natl Acad Sci U S A* 116(9):3919–3928. <https://doi.org/10.1073/pnas.1815356116>
 29. Harayama T, Riezman H (2018) Understanding the diversity of membrane lipid composition. *Nat Rev Mol Cell Biol* 19(5):281–296. <https://doi.org/10.1038/nrm.2017.138>
 30. van Meer G, Voelker DR, Feigenson GW (2008) Membrane lipids: where they are and how they behave. *Nat Rev Mol Cell Biol* 9(2):112–124. <https://doi.org/10.1038/nrm2330>

31. Ghosh A, Nishtala K (2017) Biofluid lipidome: a source for potential diagnostic biomarkers. *Clin Transl Med* 6(1):22. <https://doi.org/10.1186/s40169-017-0152-7>
32. Borghini I, Barja F, Pometta D, James RW (1995) Characterization of subpopulations of lipoprotein particles isolated from human cerebrospinal fluid. *Biochim Biophys Acta* 1255 (2):192–200
33. Koch S, Donarski N, Goetze K, Kreckel M, Stuerenburg HJ, Buhmann C, Beisiegel U (2001) Characterization of four lipoprotein classes in human cerebrospinal fluid. *J Lipid Res* 42(7):1143–1151
34. Mahley RW (2016) Central nervous system lipoproteins: ApoE and regulation of cholesterol metabolism. *Arterioscler Thromb Vasc Biol* 36(7):1305–1315. <https://doi.org/10.1161/atvaha.116.307023>
35. Harrington MG, Fonteh AN, Oborina E, Liao P, Cowan RP, McComb G, Chavez JN, Rush J, Biringer RG, Huhmer AF (2009) The morphology and biochemistry of nanostructures provide evidence for synthesis and signaling functions in human cerebrospinal fluid. *Cerebrospinal Fluid Res* 6(10). <https://doi.org/10.1186/1743-8454-6-10>
36. Bouatra S, Aziat F, Mandal R, Guo AC, Wilson MR, Knox C, Bjorndahl TC, Krishnamurthy R, Saleem F, Liu P, Dame ZT, Poelzer J, Huynh J, Yallou FS, Psychogios N, Dong E, Bogumil R, Roehring C, Wishart DS (2013) The human urine metabolome. *PLoS One* 8(9):e73076. <https://doi.org/10.1371/journal.pone.0073076>
37. Dashti M, Kulik W, Hoek F, Veerman EC, Peppelenbosch MP, Rezaee F (2011) A phospholipidomic analysis of all defined human plasma lipoproteins. *Sci Rep* 1:139. <https://doi.org/10.1038/srep00139>
38. Kim SH, Yang JS, Lee JC, Lee JY, Lee JY, Kim E, Moon MH (2018) Lipidomic alterations in lipoproteins of patients with mild cognitive impairment and Alzheimer's disease by asymmetrical flow field-flow fractionation and nanoflow ultrahigh performance liquid chromatography-tandem mass spectrometry. *J Chromatogr A* 1568:91–100. <https://doi.org/10.1016/j.chroma.2018.07.018>
39. Kontush A, Lhomme M, Chapman MJ (2013) Unraveling the complexities of the HDL lipidome. *J Lipid Res* 54(11):2950–2963. <https://doi.org/10.1194/jlr.R036095>
40. Serna J, Garcia-Seisdedos D, Alcazar A, Lasuncion MA, Bustos R, Pastor O (2015) Quantitative lipidomic analysis of plasma and plasma lipoproteins using MALDI-TOF mass spectrometry. *Chem Phys Lipids* 189:7–18. <https://doi.org/10.1016/j.chemphyslip.2015.05.005>
41. Wiesner P, Leidl K, Boettcher A, Schmitz G, Liebisch G (2009) Lipid profiling of FPLC-separated lipoprotein fractions by electrospray ionization tandem mass spectrometry. *J Lipid Res* 50(3):574–585. <https://doi.org/10.1194/jlr.D800028-JLR200>
42. Hodson L, Skeaff CM, Fielding BA (2008) Fatty acid composition of adipose tissue and blood in humans and its use as a biomarker of dietary intake. *Prog Lipid Res* 47(5):348–380. <https://doi.org/10.1016/j.plipres.2008.03.003>
43. Sales S, Graessler J, Ciucci S, Al-Atrib R, Vihervaara T, Schuhmann K, Kauhanen D, Sysi-Aho M, Bornstein SR, Bickle M, Cannistraci CV, Ekroos K, Shevchenko A (2016) Gender, contraceptives and individual metabolic predisposition shape a healthy plasma lipidome. *Sci Rep* 6:27710. <https://doi.org/10.1038/srep27710>
44. Chua EC, Shui G, Lee IT, Lau P, Tan LC, Yeo SC, Lam BD, Bulchand S, Summers SA, Puvanendran K, Rozen SG, Wenk MR, Gooley JJ (2013) Extensive diversity in circadian regulation of plasma lipids and evidence for different circadian metabolic phenotypes in humans. *Proc Natl Acad Sci U S A* 110 (35):14468–14473. <https://doi.org/10.1073/pnas.1222647110>
45. Yuana Y, Sturk A, Nieuwland R (2013) Extracellular vesicles in physiological and pathological conditions. *Blood Rev* 27(1):31–39. <https://doi.org/10.1016/j.blre.2012.12.002>
46. Huang d W, Sherman BT, Lempicki RA (2009) Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nat Protoc* 4(1):44–57. <https://doi.org/10.1038/nprot.2008.211>
47. Huang da W, Sherman BT, Lempicki RA (2009) Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. *Nucleic Acids Res* 37(1):1–13. <https://doi.org/10.1093/nar/gkn923>
48. Luo W, Brouwer C (2013) Pathview: an R/Bioconductor package for pathway-based data integration and visualization. *Bioinformatics* 29(14):1830–1831. <https://doi.org/10.1093/bioinformatics/btt285>
49. Luo W, Pant G, Bhavnasi YK, Blanchard SG Jr, Brouwer C (2017) Pathview Web: user friendly pathway visualization and data integration. *Nucleic Acids Res* 45(W1):W501–w508. <https://doi.org/10.1093/nar/gkx372>

50. Pedersen HK, Forslund SK, Gudmundsdottir V, Petersen AO, Hildebrand F, Hyotylainen T, Nielsen T (2018) A computational framework to integrate high-throughput ‘omics’ datasets for the identification of potential mechanistic links. *Nat Protoc* 13(12):2781–2800. <https://doi.org/10.1038/s41596-018-0064-z>
51. Eisfeld AJ, Halfmann PJ, Wendler JP, Kyle JE, Burnum-Johnson KE, Peralta Z, Macmura T, Walters KB, Watanabe T, Fukuyama S, Yamashita M, Jacobs JM, Kim YM, Casey CP, Stratton KG, Webb-Robertson BM, Gritsenko MA, Monroe ME, Weitz KK, Shukla AK, Tian M, Neumann G, Reed JL, van Bakel H, Metz TO, Smith RD, Waters KM, N’Jai A, Sahr F, Kawaoka Y (2017) Multi-platform ‘Omics analysis of human Ebola virus disease pathogenesis. *Cell Host Microbe* 22(6):817–829.e818. <https://doi.org/10.1016/j.chom.2017.10.011>



Chapter 8

Overview of Tandem Mass Spectral and Metabolite Databases for Metabolite Identification in Metabolomics

Zhangtao Yi and Zheng-Jiang Zhu

Abstract

Liquid chromatography–mass spectrometry (LC-MS) is one of the most popular technologies in metabolomics. The large-scale and unambiguous identification of metabolite structures remains a challenging task in LC-MS based metabolomics. Tandem mass spectral databases provide experimental and in silico MS/MS spectra to facilitate the identification of both known and unknown metabolites, which has become a gold standard method in metabolomics. In addition, metabolite knowledge databases offer valuable biological and pathway information of metabolites. In this chapter, we have briefly reviewed the most common and important tandem mass spectral and metabolite databases, and illustrated how they could be used for metabolite identification.

Key words Metabolite identification, Metabolomics, Tandem mass spectrum, Metabolite database

1 Introduction

Metabolomics comprehensively measures metabolites in a biological system to provide mechanistic insights of biological and disease-related events. With technology advances in mass spectrometry, hundreds to thousands of metabolites can now be quantitatively profiled from biological samples. Liquid chromatography–mass spectrometry (LC-MS) is one of the most popular technologies in metabolomics due to its high sensitivity, selectivity, and throughput. However, the major bottleneck of metabolomics has been the challenge of determining the structural identities of the MS peaks found to be dysregulated in the metabolomic profiling [1, 2]. The use of accurate mass for identifying metabolites in LC-MS based metabolomics is highly challenging. The problem is that there are many possible molecular formulas within the tolerance, and numerous structural isomers for each molecular formula. The alternative approach which matches the accurate precursor mass (MS1) and tandem mass spectrum (MS/MS or MS2) with those from the standard spectral library to identify metabolites has

Table 1
Statistics of tandem mass spectral databases

Database	Compounds with spectra	Number of Spectra	In silico spectra	Free Search	Free download
NIST 17	13,808	574,826	×	\$	\$
METLIN	>200,000	NA	√	√	✗
MoNA	~75,000	261,917	√	√	✓
mzCloud	> 8000	~2,000,000 ^a	×	√	✗
HMDB	2265	22,247 ^b	×	√	✓
LipidBlast	119,200	212,516	√	–	✓

^aThe number of recalibrated spectra in mzCloud

^bTotal experimental LC-MS/MS spectra in HMDB

become the “gold” standard method [1]. This method requires the availability of standard MS/MS spectra. In past decades, many efforts have been made to expand the existing tandem mass spectral databases such as METLIN [3], NIST [4], and MassBank [5] (Table 1). Clearly, all mass spectral databases are hindered by the lack of chemical standards for many cellular metabolites, and suffer from having uncharacterized spectral variations across different LC-MS instruments, acquisition condition, and laboratories. Other efforts have also been made to theoretically predict the MS/MS spectra in silico [6, 7]. Limited by the high diversity of chemical structures of metabolites and the relatively small size of training dataset, the accuracy for the in silico prediction approach still requires a substantial improvement. Molecular and metabolic pathways can also be utilized for metabolite identification by mapping dysregulated metabolic features into the metabolic network, such as Mummichog [8] and PIUMet [9]. In this approach, metabolic pathway databases are generally required.

Many metabolomics-related databases have been constructed and are either freely accessible or commercially available to provide chemical structures, physiochemical properties, spectral information, biological functions, and pathway information. In this chapter, we have provided an overview of current databases for metabolomics, especially the supportive databases for metabolite identification (Table 1), including tandem mass spectral databases like NIST, METLIN, MassBank, and mzCloud; metabolite knowledge databases such as HMDB; and metabolic pathway databases such as KEGG [10], MetaCyc [11] and Reactome [12]. However, common chemical databases such as PubChem and ChemSpider are not covered in this chapter.

2 METLIN

METLIN was first developed in 2003 by Scripps Center for Metabolomics led by Prof. Gary Siuzdak [13] and is one of the largest tandem mass spectral databases for metabolites. METLIN now covers more than 300,000 compounds (data from Chapter 10), including metabolites, drugs, toxicants, peptides and so on. Metadata in METLIN includes compound name, structure, formula, mass, CAS number, KEGG ID, HMDB ID, and PubChem ID. The commercial availabilities of compounds are also provided. Most importantly, over 200,000 compounds have either experimental or in silico-predicted tandem mass spectra. Among them, experimental tandem mass spectra were acquired from authentic chemical standards. All experimental tandem mass spectra were collected under a strict experimental protocol on high-resolution Agilent QTOF instruments (6500 series). MS2 spectra were acquired using four collision energy levels (0, 10, 20, and 40 eV) on both positive and negative modes. The MS/MS spectra at 0 eV represent the in-source fragmentation, and facilitate the annotation of in-source fragments in untargeted metabolomics. All MS2 spectra in METLIN were curated and recalibrated. The possible chemical structures were also assigned to product ion using the bioinformatic tool MetFrag. Originally, METLIN only contains the experimental tandem mass spectra, but recently it also incorporated the in silico-predicted MS2 spectra. The in silico prediction of MS2 spectra were performed by input–output kernel regression (IOKR)-based machine learning algorithm [3]. The experimental MS2 spectra in METLIN were used for training the machine learning algorithm. METLIN is freely available through the web server (<https://metlin.scripps.edu>) with a user-friendly interface. METLIN web server has provided multiple searching capabilities including single, batch, precursor ion, neutral loss, accurate mass, and fragment searches. MS/MS spectral match in METLIN was based on X-Rank similarity algorithm, which provides the robust identification result when matching experimental spectra from different instrument platforms. The only drawback of METLIN is that the MS2 spectra are not downloadable. For a more detailed description of METLIN, please refer to Chapter 10.

3 NIST 17 Tandem Mass Spectral Libraries

NIST mass spectral libraries are developed by the NIST mass spectrometry data center led by Dr. Stephen E. Stein (Gaithersburg, MD, USA). The NIST spectral libraries aim to facilitate chemical compound identification in GC- and LC-MS by providing standard mass spectra. These libraries were originally developed as GC-MS

spectral libraries (by electron ionization) and later incorporated LC-MS based tandem mass spectra. NIST Tandem mass spectral library were acquired using a range of fragmentation conditions including positive and negative modes from ion-trap and collision cells such as QTOF, Orbitrap, and QqQ. The tandem mass spectra include both low-resolution ones from QqQ and ion trap, and high-resolution ones from QTOF and Orbitrap, and cover a wide range of compounds including small molecules and peptides, so NIST libraries are not specific for metabolites. The NIST tandem mass spectral libraries are updated every 3 years, and NIST17 is the latest version (<https://chemdata.nist.gov/dokuwiki/doku.php?id=chemdata:nist17>).

NIST 17 tandem spectral libraries contain a total of 13,808 compounds (118,082 precursor ions, and 574,826 tandem spectra), and 1904 biologically relevant peptides in an additional library. Each compound entry in the library contains compound name, formula, CAS number, InChIKey, MS/MS spectra, data acquisition condition, and so on. MS/MS spectra were collected from multiple collision energy levels over a wide range of instrument platforms, mainly from Thermo LTQ Orbitrap Elite, Agilent QTOF 6530, Micromass Quattro Micro QqQ, Thermo Finnigan LTQ Orbitrap Velos, and so on. For LTQ Orbitrap instruments, spectra from both HCD and IT/ion trap collision type were provided. NIST 17 also includes tandem mass spectra for multiple common ion adducts, such as $[M+H]^+$, $[M+H-H_2O]^+$, $[M+Na]^+$, and $[M+NH_4]^+$ in positive mode and $[M-H]^-$ and $[M-H-H_2O]^-$ in negative mode.

For compound identification, NIST provides the MS search software (now version 2.3) to search the libraries. With the search software, one can browse or search against the libraries by exact mass, formula, precursor or product ion mass, CAS number, and so on. However, for batch query, one has to use the other software MS PepSearch. Mass spectra in libraries can also be individually exported as msp format, and used in other software tools for metabolite identification. Notably, NIST 17 has also included a new search method, hybrid similarity search (HSS) [14]. HSS is very useful to annotate new chemical structures that do not have tandem mass spectra in the libraries, because it allows matching the experimental spectrum to a library spectrum with a neutral loss using the m/z differences between their precursor ions. In summary, NIST 17 tandem mass spectral libraries feature a broad coverage of small molecules, and MS2 spectra from various types of ion adducts with different collision energy levels and instruments. However, NIST 17 is not freely available, and it lacks biological information or links to metabolite databases for metabolomics research. NIST 17 should be used as a tool to elucidate compound structures.

4 MassBank and MoNA

MassBank was originally designed as an open-source and public mass spectral repository to collect and store freely available tandem mass spectra. MassBank web server was first developed in Japan [5] (2010, <http://www.massbank.jp>) and later as a mirror in Europe (2012, <https://massbank.eu/MassBank>). Tandem mass spectra in MassBank were contributed from different research groups with author names tagged to each entry. MassBank allows the users to freely download and upload mass spectra. MassBank now encompasses a total of experimental mass spectra of around 40,000 CID spectra, 13,000 EI spectra and other types of spectra. CID spectra were mostly acquired from TOF, QqQ, and ion trap instruments. In 2013, an R package RMassBank [15] was developed as a recalibration tool for uploaded spectra.

Recently, MoNA (MassBank of North America) has been developed by Prof. Oliver Fiehn Lab in University of California-Davis, and incorporated all spectra in MassBank. In addition, MoNA also collected tandem mass spectra from other databases such as ReSpect (<http://spectra.psc.riken.jp>), GNPS (<https://gnps.ucsd.edu>), and MetaboBASE (from Bruker), together with Fiehn lab in-house libraries such as Fiehn HILIC, and Vaniya/Fiehn natural products library. In silico spectral libraries for lipids, such as LipidBlast, were also included in MoNA. Therefore, MoNA is currently the most comprehensive tandem spectral database in the world. Now, MoNA has 122,171 experimental spectra, of which 96,996 spectra are specified as LC-MS spectra, and 139,746 in silico spectra. MoNA covered MS2 spectra for common ion adducts obtained from both QTOF and Orbitrap instruments. Each entry in MoNA contains the submitter, mass spectrum, compound name, downloadable 2D structure, classification provided by ClassyFire [16], InChIKey, InChI, SMILES, and metadata including instrument, chromatographic condition, and so on.

MoNA web server allows both “quick search” by exact mass and “similarity search” by uploading an MS/MS spectrum (e.g., in an MGF file). When one is browsing the mass spectrum in MoNA, it also allows for searching “similar spectra” in the databases. All spectra from MoNA are freely downloaded as either JSON or msp format. Distribution of most spectra in MoNA is under license of Creative Commons by Attribution (CC-BY). The downloaded spectra can be imported into other software tools such as NIST MS search software and MS-DIAL for metabolite identification. Each spectrum in MoNA is given an identifier—SPLASH [17], which is a hashed string that provides an unambiguous, database-independent identifier. The use of SPLASH allows for the quick and accurate query and cross-reference of the spectra across different databases.

5 mzCloud

mzCloud (<https://www.mzcloud.org>) is a commercial tandem mass spectral database hosted by HighChem and has a focus on spectra acquired only from Orbitrap instruments. Both MS/MS and multistage MSⁿ spectra were acquired using chemical standards at various collision energies using both collision-induced dissociation (CID) and higher-energy collisional dissociation (HCD). Currently, mzCloud includes 23,800 spectra trees covering 209,012 precursors from over 8000 compounds. All raw tandem mass spectra were postprocessed by filtering and recalibration to improve the spectral quality. Fragment ions in tandem mass spectra were annotated through applying the general fragmentation rules and fragmentation mechanisms published in peer-reviewed journals, manual evaluation, and ab initio quantum chemical computation. mzCloud can be freely accessed through the web server. It supports MS/MS spectral similarity match in single or batch mode for metabolite identification. For compounds not included in the database, mzCloud enables the de novo metabolite structural elucidation via identifying the possible substructures in the query tandem mass spectrum through comparing the product ion spectra of structurally related compounds. However, all tandem mass spectra in mzCloud cannot be downloaded, and all spectra are exclusively from Orbitrap instruments.

6 HMDB

HMDB (human metabolome database, <http://www.hmdb.ca>) was developed by the Canada metabolomics innovation center led by Prof. David Wishart. HMDB is the most comprehensive and *Homo sapiens*-only metabolite knowledge database. HMDB is now updated to version 4.0 [18] and covers a total of 114,100 metabolites. Each metabolite entry includes massive metadata information such as name, formula, MDL Molfile, InChI, SMILES, chemical taxonomy, physiological data, normal and abnormal concentrations, related genomic and enzymatic information, pathway information, links to other databases such as PubChem, ChemSpider, KEGG, BioCyc, and related spectral characterizations. Only a small portion of metabolites in HMDB have provided the tandem mass spectra. Specifically, there are 22,247 experimental tandem spectra from 2265 compounds in HMDB. Most of these were imported from MassBank and other spectral databases. In HMDB 4.0 release, *in silico*-predicted MS2 spectra obtained from CFM-ID [19] were also incorporated. In addition to tandem mass spectra, HMDB also includes NMR spectra for metabolite identification. The HMDB provides many different search functions, including

ChemQuery structure search, molecular weight search, and LC-MS/MS search. For a more detailed description of HMDB, please refer to Chapter 11. All information in HMDB can be freely downloaded.

7 LIPID MAPS and LipidBlast

LIPID MAPS is a lipid knowledge database developed by The LIPID MAPS consortium [20, 21]. It includes several different databases including structure database (LMSD), gene/proteome database (LMPD), in silico structure database (LMISSD), and database of computationally generated bulk lipid species (COMP_DB). Among them, LMSD is the most important lipid database with a total of 43,403 biologically relevant lipids from eight categories, namely, fatty acyls (FA), glycerolipids (GL), glycerophospholipids (GP), sphingolipids (SP), sterol lipids (ST), prenol lipids (PR), saccharolipids (SL), and polyketides (PK). Each category also includes various subclassification hierarchies. Each lipid in LMSD has a unique LIPID MAPS identification number—“LM_ID” identifier [22]. One could search the lipid class via classification-based, text/ontology-based, and structure-based searches. LMISSD is an in silico-generated lipid structure databases through the theoretical and computational expanding of head-groups and chains from different lipid classes accounting for a total number of lipids >one million. Each entry in LMSD and LMISSD has a LIPID MAPS ID, MDL Molfile, SMILES, InChI, InChIKey, exact mass, formula, systematic name, and a set of abbreviations for sum composition, chain composition, and exact structure. All of this data can be freely downloaded as structure-data file (SDF) format.

For some lipids in LMSD, experimental tandem mass spectra were collected using chemical standards. However, these spectra are only provided as images and cannot be used for spectral similarity match. The experimental spectra for lipids are not as important as metabolites, since fragmentation of lipids are well studied and easier to be predicted and simulated. There are many computational tools to generate in silico-predicted MS₂ spectra for lipids such as LipidBlast [23] and in silico prediction tools in LIPID MAPS. LipidBlast is one of the largest stand-alone in silico tandem spectral databases for lipid identification, which covers a total of 212,516 predicted spectra of 119,200 compounds from 26 lipid compound classes. Spectra from common ion adducts such as [M+H]⁺, [M+Na]⁺, [M +NH₄]⁺, and [M-H]⁻ are covered. LipidBlast can be freely downloaded (<https://fiehnlab.ucdavis.edu/projects/LipidBlast>). Other databases such as MoNA and LMSD have incorporated the spectra in LipidBlast. LipidBlast can be imported to other MS/MS search tool such as NIST MS Search and NIST MS PepSearch for lipid identification.

8 Metabolic Pathway Databases

The metabolic pathway database contains information about metabolic pathways, metabolic reactions and related genes and enzymes [24]. All of the information is important to interpret the biological functions of metabolites that are found to be dysregulated in metabolomics experiment. In addition, several important software tools such as Mummichog [8], PIUMet [9], and MetDNA [25] also use the metabolic pathway and metabolic reactions for metabolite identification.

Kyoto Encyclopedia of Genes and Genomes (KEGG) pathway database (<https://www.genome.jp/kegg/>) is one of the most curated and comprehensive pathway databases, which contains a collection of metabolic pathway maps representing the knowledge on the molecular interaction, reaction, and relation networks for metabolism. Currently, KEGG includes a total of 530 pathway maps from 491 eukaryotes and 5379 prokaryotes, composed of 18,505 metabolites and other small compounds and 11,146 reactions. Massive genomic and enzymatic information are also incorporated. Every metabolite in KEGG has a KEGG ID and contains name, classification, reaction, pathway, reference, and so on. All data can be freely searched in KEGG web server or via KEGG API (<https://www.kegg.jp/kegg/rest/>).

MetaCyc (<https://MetaCyc.org>) is a curated metabolic pathway database with 2666 metabolic pathways from 2960 different organisms. It contains experimental data about chemical compounds, reactions, enzymes, and metabolic pathways curated from more than 54,000 publications, making it the largest collection and reference database of metabolic pathways. Data in MetaCyc can be searched in web server by keyword, ontology, identifier, and so on. All data can be freely downloaded for local analysis or imported into other database. Each metabolite entry has InChI, InChIKey, SMILES, standard Gibbs free energy, reaction, enzymatic regulation information, and references.

Reactome (<https://reactome.org/>) is a free, open-source pathway database and provides tools for the visualization, interpretation and analysis of pathway knowledge. It contains around 20,000 pathways from different species covering around 80,000 reactions. Genomic and enzymatic information are also included. Reactome provides a pathway browser in which the pathway can be freely browsed in an expandable hierarchy or browsed by typing a keyword. Each pathway entry contains compounds and proteins involved. All data and software tools in Reactome are freely available for download.

Despite the increase of pathway knowledge, there are still some missing pathways and metabolites. To complement the gap, in silico metabolite databases such as MINEs [26] and

MyCompoundID [27] provide computationally generated metabolic network and reaction information derived from known metabolites and biochemical reactions. This enlarged pathway information can further expand the metabolite knowledge databases and facilitate annotating new metabolites.

Acknowledgments

The work has been supported by National Key R&D Program of China (2018YFA0800902), National Natural Science Foundation of China (Grants 21575151) and Chinese Academy of Sciences Major Facility-based Open Research Program. Z.-J. Z. is supported by Thousand Youth Talents Program.

References

- Vinaixa M, Schymanski EL, Neumann S, Navarro M, Salek RM, Yanes O (2016) Mass spectral databases for LC/MS- and GC/MS-based metabolomics: state of the field and future prospects. *Trends Anal Chem* 78:23–35
- Domingo-Almenara X, Montenegro-Burke JR, Benton HP, Siuzdak G (2018) Annotation: a computational solution for streamlining metabolomics analysis. *Anal Chem* 90:480–489
- Guilas C, Montenegro-Burke JR, Domingo-Almenara X, Palermo A, Warth B, Hermann G, Koellensperger G, Huan T, Uritboonthai W, Aisporna AE et al (2018) METLIN: a technology platform for identifying knowns and unknowns. *Anal Chem* 90:3156–3164
- Yang X, Neta P, Liang Y, Stein SE (2017) Extending a comprehensive reference tandem mass spectral library for more reliable metabolite identification. 65th Annual ASMS conference on mass spectrometry and allied topics, Indianapolis, Indiana, June 4–8, 2017
- Horai H, Arita M, Kanaya S, Nihei Y, Ikeda T, Suwa K, Ojima Y, Tanaka K, Tanaka S, Aoshima K (2010) MassBank: a public repository for sharing mass spectral data for life sciences. *J Mass Spectrom* 45:703–714
- Kind T, Tsugawa H, Cajka T, Ma Y, Lai Z, Mehta SS, Wohlgemuth G, Barupal DK, Shewalter MR, Arita M et al (2018) Identification of small molecules using accurate mass MS/MS search. *Mass Spectrom Rev* 37:513–532
- Blazenovic I, Kind T, Ji J, Fiehn O (2018) Software tools and approaches for compound identification of LC-MS/MS data in metabolomics. *Meta* 8:31
- Li S, Park Y, Duraisingham S, Strobel FH, Khan N, Soltow QA, Jones DP, Pulendran B (2013) Predicting network activity from high throughput metabolomics. *PLoS Comput Biol* 9:e1003123
- Pirhaji L, Milani P, Leidl M, Curran T, Avila-Pacheco J, Clish CB, White FM, Saghatelian A, Fraenkel E (2016) Revealing disease-associated pathways by network integration of untargeted metabolomics. *Nat Methods* 13:770–776
- Kanehisa M, Furumichi M, Tanabe M, Sato Y, Morishima K (2016) KEGG: new perspectives on genomes, pathways, diseases and drugs. *Nucleic Acids Res* 45:D353–D361
- Caspi R, Billington R, Fulcher CA, Keseler IM, Kothari A, Krummenacker M, Latendresse M, Midford PE, Ong Q, Ong WK et al (2018) The MetaCyc database of metabolic pathways and enzymes. *Nucleic Acids Res* 46:D633–D639
- Fabregat A, Jupe S, Matthews L, Sidiropoulos K, Gillespie M, Garapati P, Haw R, Jassal B, Korninger F, May B (2017) The reactome pathway knowledgebase. *Nucleic Acids Res* 46:D649–D655
- Smith CA, O'Maille G, Want EJ, Qin C, Trauger SA, Brandon TR, Custodio DE, Abagyan R, Siuzdak G (2005) METLIN: a metabolite mass spectral database. *Ther Drug Monit* 27:747–751
- Burke MC, Mirokhin YA, Tchekhovskoi DV, Markey SP, Heidbrink Thompson J, Larkin C, Stein SE (2017) The hybrid search: a mass spectral library search method for discovery of modifications in proteomics. *J Proteome Res* 16:1924–1935
- Stravs MA, Schymanski EL, Singer HP, Hollender J (2013) Automatic recalibration and

- processing of tandem mass spectra using formula annotation. *J Mass Spectrom* 48:89–99
- 16. Feunang YD, Eisner R, Knox C, Chepelev L, Hastings J, Owen G, Fahy E, Steinbeck C, Subramanian S, Bolton E (2016) ClassyFire: automated chemical classification with a comprehensive, computable taxonomy. *J Chem* 8:61
 - 17. Wohlgemuth G, Mehta SS, Mejia RF, Neumann S, Pedrosa D, Pluskal T, Schymanski EL, Willighagen EL, Wilson M, Wishart DS et al (2016) SPLASH, a hashed identifier for mass spectra. *Nat Biotechnol* 34:1099–1101
 - 18. Wishart DS, Feunang YD, Marcu A, Guo AC, Liang K, Vázquez-Fresno R, Sajed T, Johnson D, Li C, Karu N et al (2018) HMDB 4.0: the human metabolome database for 2018. *Nucleic Acids Res* 46:D608–D617
 - 19. Allen 'F, Greiner R, Wishart D (2014) Competitive fragmentation modeling of ESI-MS/MS spectra for putative metabolite identification. *Metabolomics* 11:98–110
 - 20. Sud M, Fahy E, Cotter D, Brown A, Dennis EA, Glass CK, Merrill AH Jr, Murphy RC, Raetz CR, Russell DW (2006) LMSD: lipid maps structure database. *Nucleic Acids Res* 35:D527–D532
 - 21. Fahy E, Sud M, Cotter D, Subramaniam S (2007) LIPID MAPS online tools for lipid research. *Nucleic Acids Res* 35:W606–W612
 - 22. Fahy E, Subramaniam S, Murphy RC, Nishijima M, Raetz CR, Shimizu T, Spener F, van Meer G, Wakelam MJ, Dennis EA (2009) Update of the LIPID MAPS comprehensive classification system for lipids. *J Lipid Res* 50: S9–S14
 - 23. Kind T, Liu KH, Lee DY, DeFelice B, Meissen JK, Fiehn O (2013) LipidBlast in silico tandem mass spectrometry database for lipid identification. *Nat Methods* 10:755–758
 - 24. Fiehn O, Barupal DK, Kind T (2011) Extending biochemical databases by metabolomic surveys. *J Biol Chem* 286:23637–23643
 - 25. Shen X, Wang R, Xiong X, Yin Y, Cai Y, Ma Z, Liu N, Zhu Z-J (2019) Metabolic reaction network-based recursive metabolite annotation for untargeted metabolomics. *Nat Commun* 10:1516
 - 26. Jeffries JG, Colastani RL, Elbadawi-Sidhu M, Kind T, Niehaus TD, Broadbelt LJ, Hanson AD, Fiehn O, Tyo KE, Henry CS (2015) MINES: open access databases of computationally predicted enzyme promiscuity products for untargeted metabolomics. *J Chem* 7:44
 - 27. Huan T, Tang C, Li R, Shi Y, Lin G, Li L (2015) MyCompoundID MS/MS search: metabolite identification using a library of predicted fragment-ion-spectra of 383,830 possible human metabolites. *Anal Chem* 87:10619–10626



Chapter 9

METLIN: A Tandem Mass Spectral Library of Standards

J. Rafael Montenegro-Burke, Carlos Guijas, and Gary Siuzdak

Abstract

Untargeted mass spectrometry metabolomics studies rely on accurate databases for the identification of metabolic features. Leveraging unique fragmentation patterns as well as characteristic dissociation routes allows for structural information to be gained for specific metabolites and molecular classes, respectively. Here we describe the evolution of METLIN as a resource for small molecule analysis as well as the tools (e.g., Fragment Similarity Search and Neutral Loss Search) used to query the database and their workflows for the identification of molecular entities. Additionally, we will discuss the functionalities of isoMETLIN, a database of isotopic metabolites, and the latest addition to the METLIN family, METLIN-MRM, which facilitates the analysis of quantitative mass spectrometry data generated with triple quadrupole instrumentation.

Key words Untargeted metabolomics, Spectral database, MS/MS spectra, Metabolite identification

1 Introduction

As the interest of the scientific community fell on endogenous metabolites almost 20 years ago due to their implications in diagnostics and biomarker discovery, technical challenges had to be overcome in order to use mass spectrometry or metabolomics for these biomedical purposes. Particularly, chromatographic data processing and metabolite characterization/identification of untargeted metabolomic experiments represented the biggest obstacles for researchers worldwide [1]. While the development of chromatographic data processing tools is covered in this book, this chapter will focus on the metabolite database METLIN and its associated tools, from inception with a few hundred metabolites to the most current version with over ~600,000 compounds (November 2019).

METLIN was developed to assist metabolomics research across a broad range of disciplines, especially to facilitate the identification

J. Rafael Montenegro-Burke and Carlos Guijas contributed equally to this work.

of metabolites in LC-MS based metabolomics and bridge the gap to other “omics” sciences such as genomics and proteomics [1]. The molecular or macromolecular identification process to assign names and annotations to genes and proteins from genomics and proteomics experiments has been made possible not only due to great efforts in their respective communities but also due to the predicative sequence of nucleosides and amino acids. On the other hand, the large chemical diversity found in the metabolome as well as the vast number of different molecular entities limits the prediction of fragmentation patterns (MS/MS) in MS-based experiments thereby limiting the ability to make identifications [2].

METLIN was first developed in 2003 and contained a few hundred metabolites and MS/MS spectra [1]. However, its growth in terms of the number of metabolites and MS/MS spectra but also the incorporation of new tools has been continuous. For example, after it was made publicly available in 2005, METLIN grew from a few hundred to more than 10,000 metabolites with their respective MS/MS spectra by 2012 [3]. Additionally, tools to facilitate and automate the identification of known and unknown metabolites have been integrated [4–6].

Concomitantly, other academic, public, and private entities have developed metabolite/small molecule databases, which could be classified into two categories: (1) pathway-centric and (2) compound-centric [7]. METLIN belongs to the latter as it contains chemical structures, spectral profiles and its tools are designed to facilitate metabolite identification.

Similar databases containing MS/MS spectra include MassBank, HMDB, GNPS, MoNA, LIPID MAPS, NIST 14, and mzCloud [8–12]. For a more detailed comparison between these databases we direct the reader to the work of Vinaixa et al. [2]. Briefly some of the main differences are data collection and curation processes, instrumentation used to generate data, standard reference materials, types of molecules, source and origin of molecule, and accessibility. In METLIN, MS/MS spectra have been acquired at the Scripps Center for Metabolomics following strict protocols and only using standard reference material that is either commercially available or has been synthesized. Early on in the development process, it was decided against the use of complex biological samples for reference fragmentation spectra in order to avoid interfering molecules and provide the best spectral quality. This is unlike other databases where MS/MS spectra have been acquired in different laboratories under different conditions and instrumentation. Over the years, METLIN has also incorporated metabolites from a wide range of molecular classes and biological or chemical origins. Therefore, the broad spectrum of small molecules available in METLIN include endogenous metabolites such as lipids, amino acids, nucleotides and carbohydrates; toxicants have recently been included to assist exposome research [13]. Further,

drugs and drug secondary metabolites can be found in METLIN as well, as the use of untargeted metabolomics in drug development has gained traction in recent years [14].

2 Tools for the Identification of Known and Unknown Metabolites

2.1 Basic Search Engines

In a traditional untargeted metabolomics workflow, dysregulated features (i.e., specific m/z values with their respective retention time values) will be searched against a database for putative metabolite identifications based on accurate mass. METLIN provides three search options: (1) Simple, (2) Batch, and (3) Advanced Search (Fig. 1). Simple Search allows the search of both m/z values and neutral masses within a selected mass tolerance. Several adducts in both positive and negative polarities can be selected. Batch Search has the same functions and capabilities as Simple Search, however, several m/z values can be searched simultaneously, facilitating the annotation of adducts and common losses that stem from the same metabolite. Similarly, ions with a different molecular origin can be easily distinguished and linked to other putative m/z values with this search feature. Advanced Search, on the other hand, allows the user to search based on other metabolite information such as molecular formula, metabolite names, SMILES and KEGG, CAS and MID numbers. These searches provide a list of metabolite names (or molecular formulas) that could potentially correspond to the dysregulated feature of interest. However, given the low elemental diversity in biomolecules (C, H, N, O, P, and S), this list can contain tens to hundreds of putative metabolite identifications.

2.2 Identification of Known Metabolites: MS/MS Spectrum Match Search

In order to reduce the list of putative identifications obtained from m/z value searching, experimental MS/MS spectra are compared against spectral libraries in terms of fragmentation patterns (m/z of fragments and their intensities) (Fig. 1). MS/MS Spectrum Match Search is a tool that allows the autonomous identification of metabolites. Here, users upload the fragmentation profile as a table of m/z values and intensities, and enter the mass of the precursor with a specific tolerance, collision energy, and polarity. This tool then searches, compares, and scores the similarity of the experimental spectra with the reference spectra in the library for all metabolites within the selected mass tolerance, relying on a modified X-Rank similarity algorithm [15]. Without a doubt the ~300,000 molecular standards with MS/MS spectra are METLIN's most valuable contribution to metabolomics research. The fragmentation spectra for these ~600,000 molecules have been acquired at four collision energies (0, 10, 20, and 40 V) in both positive- and negative-ion mode. While low collision energy spectra were previously not extensively used in the identification process, they have been

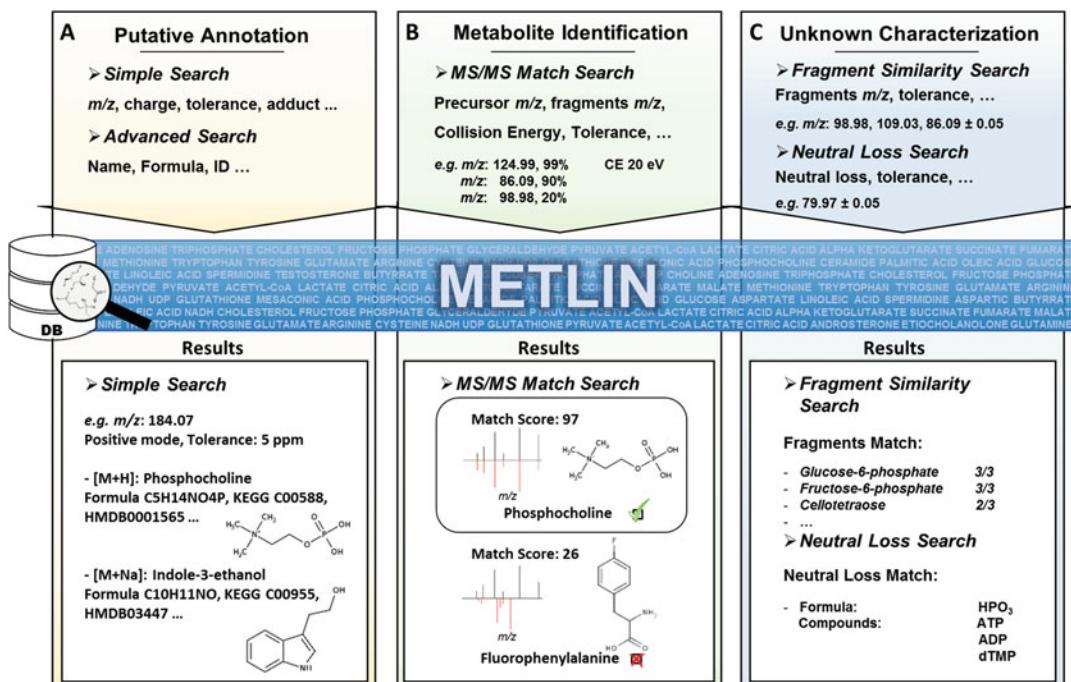


Fig. 1 METLIN search functions for small molecule identification. **(a)** Simple and Batch Search allow users to search small molecules against a database of 1 million compounds based on both m/z values and neutral masses within a selected mass tolerance. Advanced Search allows searches based on metabolite information such as molecular formula, metabolite names, SMILES and KEGG, and CAS and MID numbers. **(b)** With the MS/MS Spectrum Match Search, experimental and library MS/MS spectra can be searched, matched, and scored in an automatic way. **(c)** Fragment Similarity Search and Neutral Loss Search aid the identification of metabolites or chemical structures by searching m/z values of the fragments or neutral losses, respectively, regardless of the precursor mass. *Analytical Chemistry* 2018, 90, 3156–3164. (Figure 1, with permission from ACS and RightsLink)

recently used as a second layer in metabolite annotation algorithms and data deconvolution [16]. The combination of METLIN's large spectral library and annotation tools significantly simplifies data analysis and improves the confidence of putative identifications.

2.3 Identification of Unknown Metabolites

Interestingly, spectral libraries serve a dual purpose in the identification/characterization of metabolic features. As mentioned above, the main use of these libraries is to compare experimental spectra with reference spectra for the purpose of identification (up to level 2 according to the Metabolomics Standards Initiative) [2, 17]. However, given the large number of metabolites and the broad range of chemistries, no library is complete despite considerable efforts dedicated to increasing their populations. Furthermore, the number of metabolites in nature still is a subject of debate but estimates in the million range are not hyperbolized. Such numbers dwarf the ~20,000 genes and proteins (without taking

posttranslational modifications into account). Thus, the second purpose of spectral libraries is to aid the identification/characterization of known metabolites without MS/MS spectra available and unknown metabolites, whose structures have not been previously described in any library or resource. For this particular purpose, two tools, Fragment Similarity Search and Neutral Loss Search, have been developed and continually refined over the last 11 years [4]. These tools leverage the large number of spectral information and the dissociation routes similarities between compounds with related structures and chemical moieties.

2.3.1 Fragment Similarity and Neutral Loss Search

The Fragment Similarity Search algorithm was originally implemented into METLIN to find chemical similarities between the desired unknown features and the known molecules available in the library based on the experimental MS/MS acquired by the user and the over 4 million high-resolution MS/MS spectra in the METLIN library. Specifically, the Fragment Similarity Search algorithm relies on a shared peak count method and facilitates the identification of the molecule of interest or molecular class by prioritizing molecules with a larger chemical fragment overlap [4].

The Neutral Loss Search algorithm was designed as a complementary tool to Fragment Similarity Search. While in Fragment Similarity Search shared fragments (i.e., ions with the same m/z value) provide structural information of the ion of interest, similar structures with different molecular formulas, ergo different masses and m/z values, would generate fragments with different masses and no similarity would be determined. However, in Neutral Loss Search, mass differences between precursor ions and their fragments ions can provide structural information based on common “leaving groups” in their respective dissociation routes. Both these tools leverage the vast number of carefully curated MS/MS spectra from a wide range of small molecular entities in the METLIN library to facilitate the identification of not only known molecules with no available MS/MS spectra, but also of unknown molecules which have not been previously described in any form.

Here, we provide an example using these tools to identify unknown metabolites detected in a murine macrophage cell line (RAW264.7). The analysis was performed using an I-class UPLC system coupled to a Synapt G2-Si mass spectrometer (Waters Corp. Milford, MA). After the unsuccessful identification of a metabolic feature using accurate mass and MS/MS spectra matching procedures described in Subheadings 2.1 and 2.2 respectively, Fragment Similarity Search and Neutral Loss Search tools were employed to gain structural information and therefore clues to molecular identity in the following steps:

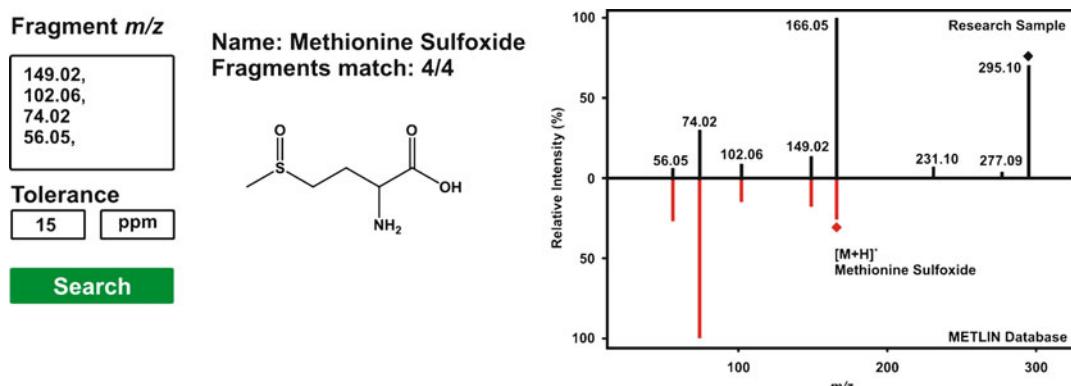


Fig. 2 Fragment Similarity Search facilitates the identification of unknown metabolites where no MS/MS spectral data are available. The fragments of an unknown metabolite were searched against METLIN and all of the four fragments were found to match with methionine sulfoxide. The comparison between experimental and library MS/MS spectra implies high structural similarities

1. Selected fragments from the murine macrophage metabolite MS/MS spectrum were searched against the METLIN MS/MS spectral database using Fragment Similarity Search tool. High intensity fragments are more likely to provide structural information of the metabolic feature, and in some cases, these can be very specific to the metabolite of interest. On the other hand, lower mass fragments are commonly shared with many molecules, making it harder to gain useful structural information from their analysis. Additionally, larger errors in the mass accuracy are inherent for low mass fragments given the mass accuracy definition [18]. In this case, we selected the fragments 149.02, 102.06, 74.02, and 56.05 from the unknown feature of interest for analysis, which resulted in many hits to molecules in METLIN. However, Fragment Similarity Search orders the results based on the number of matching fragments. Methionine sulfoxide (Met sulfoxide) matched 4 of the 4 fragments searched, hinting at a metabolite containing such a chemical structure or moieties (Fig. 2).
2. In order to gain more structural information, other fragments (higher mass) were searched as described above. The higher mass fragment searches did not yield any results that matched a particular compound or molecular class.
3. Assuming that Met sulfoxide with a monoisotopic neutral mass of 165.05 is part of the molecule of interest and we can observe a prominent fragment with the mass 166.05, the next step is to use the mass of the molecule to identify what chemical structures could constitute the rest of the molecule. After calculating the mass difference between the precursor ion and the potential methionine sulfoxide fragment ($295.10 - 166.05 = 129.04$), we use the Neutral Loss Search

tool and search for 129.04 within a selected ppm window. Most of the Neutral Loss Search results consist of molecules that contain glutamic acid (Glu), suggesting Glu is a second piece of the unknown metabolite.

4. After using Fragment Similarity Search and Neutral Loss Search, we have gathered some information about the possible chemical structures of the metabolite of interest. We then deduced how the two pieces might be connected. A plausible link between methionine sulfoxide and glutamic acid is an amide bond, which would form the dipeptide Glu Met sulfoxide.
5. Advanced Search can be used to search for such a peptide in the METLIN database by name and molecular formula. Unfortunately, this peptide is not available in the library. However, a similar dipeptide containing Glu and methionine (Met) is available with MS/MS spectra. While the only difference between the structures is the oxidation of the sulfur atom in methionine, such a small modification can have considerable differences in the dissociation routes and those differences should be considered.
6. The MS/MS spectra between the metabolite of interest (putatively Glu Met sulfoxide) and of the dipeptide Glu Met are compared next to try to further characterize the metabolite identity. Several fragments with a mass difference of an oxygen atom (15.99) are observed and were attributed to the oxidation of the sulfur atom in methionine. An additional feature of METLIN is the MetFrag algorithm developed by S. Neumann and coworkers, which indicates putative structures for the different fragment ions in the MS/MS library [19]. Based on MetFrag's analysis, we saw that several of the predicted structures that have a mass difference of 15.99 actually correspond to fragments that contain the thiol moiety. This further increases the confidence in the putative identification of the unknown feature as Glu Met sulfoxide (Fig. 3).

3 METLIN Family

3.1 isoMETLIN

In parallel to the development of METLIN, other related databases have been released relying in the growing number of analytical standards available in METLIN to reach new goals in the field of metabolomics. isoMETLIN was implemented in 2014 as a database for isotope-based metabolomics [20]. With an analogous interface to METLIN, isoMETLIN provides accurate mass of all computed isotopologs accrued in METLIN, compounds with a different number of isotope-labeled atoms and consequently, different m/z values. isoMETLIN Search includes the most commonly used

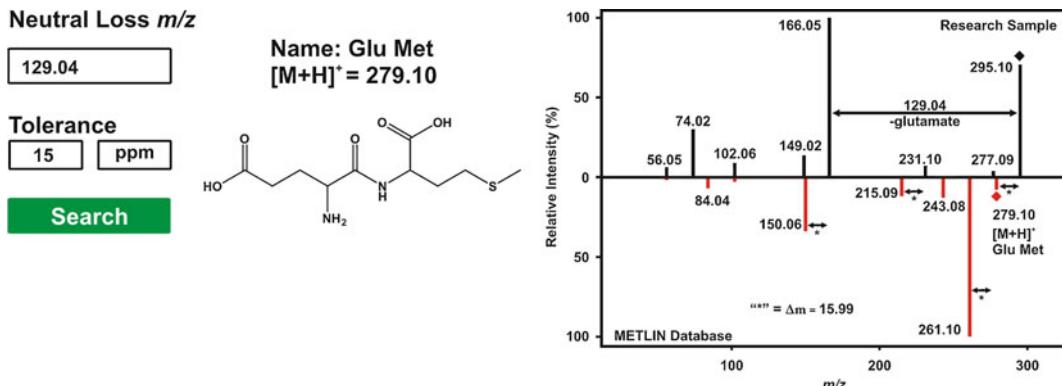


Fig. 3 Neutral loss search aides the identification of unknown metabolites based on mass differences between precursor ions and their fragments ions (i.e., common “leaving groups”). A Neutral Loss Search of 129.04 ($295.10 - 166.05 = 129.04$) yielded 168 results, where ~70% contain a glutamic acid moiety. Based on the masses of Met sulfoxide and Glu, a dipeptide is a likely option. Such a peptide is not available in the library; however, a similar compound Glu Met contains MS/MS spectra for comparison. Several fragments, which contain the thiol moiety, have a mass difference of 15.99 (monoisotopic mass of oxygen atom) corresponding to the oxidation of sulfur in methionine

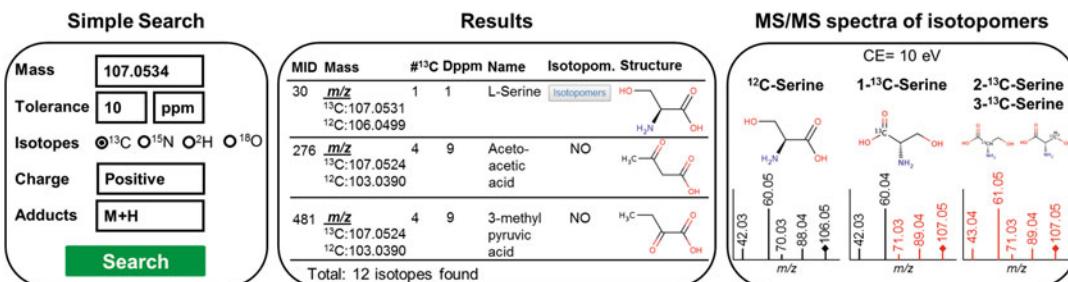


Fig. 4 isoMETLIN Simple Search menu allows searching the m/z of isotopologs within a selected mass tolerance. Type of isotopic labeling, ion charge, and ion adducts can also be selected. This search renders a list of all possible metabolites taking all possible isotopic combinations into account. Additionally, MS/MS spectra can be accessed when available

stable isotopes in labelling experiments, such as ^{13}C , ^{15}N , ^2H , and ^{18}O . Even though isotopologs can be discerned by accurate mass measurements, the further analysis of their MS/MS spectra is necessary to determine the position of the isotopic label within the same isotopologs, a pivotal feature for the investigation of metabolic pathways [21, 22]. To accomplish this, isoMETLIN also incorporates the MS/MS spectra for hundreds of isotopomers (same isotopolog with different location of labeled atoms) to help in tracing the isotopic label, providing a vast amount of information about the de novo synthesis of metabolites in certain pathways (Fig. 4). Although the principal application of isoMETLIN’s

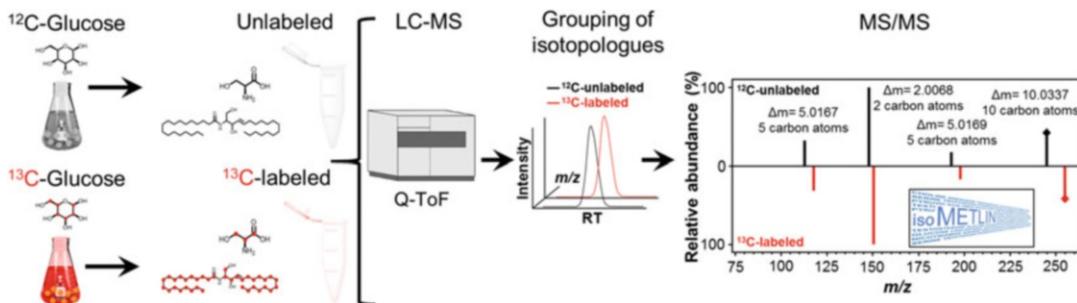


Fig. 5 Systematic generation of MS/MS spectra of uniformly labeled metabolites in isoMETLIN. *Pichia pastoris* was grown in unlabeled and ^{13}C -labeled glucose, producing uniformly labeled metabolites after several generations. Unlabeled and labeled metabolite extracts were analyzed by high-resolution untargeted metabolomics to generate pairs of unlabeled and labeled putative metabolites. Finally, the MS/MS spectra of identified pairs was incorporated into isoMETLIN after a careful curation

capabilities is the analysis of metabolic fluxes, which constitutes an emerging “omics” discipline by itself, other uses of isoMETLIN include isotope dilution quantitative metabolomics and identification of compounds, as is shown below.

3.1.1 Untargeted Generation of MS/MS Data Using Isotope-Labeled Microorganisms

Similar to METLIN, isoMETLIN fragmentation spectra was initially acquired on qToF instruments at different collision energies, using authentic isotope-labeled standards. However, a major shortcoming in populating a spectral database with MS/MS spectra of metabolite isotopologs is that the number of isotopomers increases with molecular weight (number of atoms) and that most isotopomers are not commercially available. To address this limitation, a novel approach using uniformly labeled microorganisms was recently developed [23]. Briefly, two metabolite extracts were generated by growing *Pichia pastoris* in ^{12}C -glucose- and ^{13}C -glucose-containing media (Fig. 5a). Before the acquisition of the fragmentation spectra, the labeling efficiency of the yeast metabolites was verified to be above 99%. These two unlabeled and uniformly labeled metabolite extracts were analyzed by different LC-MS platforms in an untargeted way. The unlabeled metabolites were grouped with all their isotopologs containing the isotopic trace and the MS/MS spectra of hundreds of fully labeled metabolites was generated (Fig. 5). It is worth mentioning that the reduced cost and time efforts are crucial advantages of this systematic workflow of generating MS/MS spectra for several hundreds of isotopic-labeled metabolites.

3.1.2 Identification of Metabolites Using Isotopes

An additional application of fully labeled MS/MS spectra is the gain of structural information for metabolite identification. By leveraging the mass differences between analogous fragments of isotopologs, the number of C atoms in each fragment can be

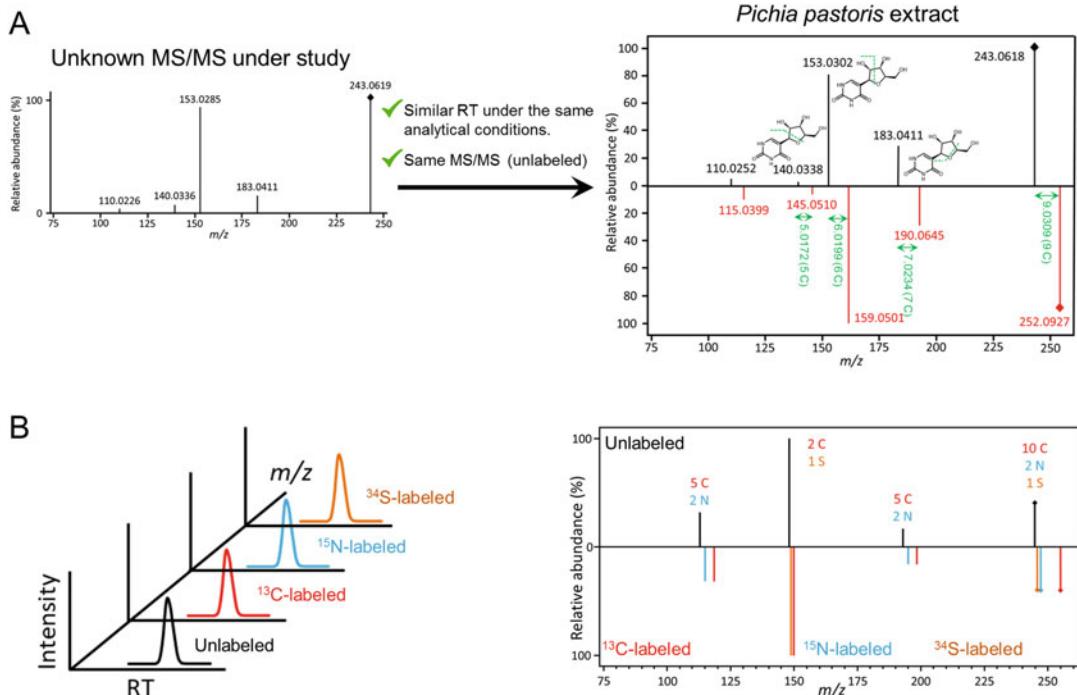


Fig. 6 (a) The MS/MS of an unknown metabolite is matched with the MS/MS of the *Pichia pastoris* extract unlabeled compound. To consider the *Pichia pastoris* pair of compounds for identification, the retention time of the unknown metabolite should match under the same analytical conditions. The number of carbons of each fragment can be used to determine the structures of both parent and fragment ions. In this example, pseudouridine was identified. (b) Proposed parallel analysis of microorganisms labeled with other stable isotopes. MS/MS spectra of fully labeled known metabolites can be incorporated into isoMETLIN, whereas MS/MS spectra of unknown metabolites can be used for their identification as described in (a)

determined. This information is of great interest for the identification of metabolites, whose MS/MS is not available, as it is the case in the identification of pseudouridine (Fig. 6a). Furthermore, two additional examples of identified metabolites using uniformly labeled *Pichia pastoris* can be found in the work of Guijas et al. [23].

Currently, these spectra are being incorporated into isoMETLIN to aid researchers in the process of identifying of metabolites and the incorporation of analogous data generated by the analysis of microorganisms uniformly labeled with other isotopes, such as ¹⁵N and ³⁴S is a future goal. Not only will this allow the addition of new MS/MS spectra of uniformly labeled metabolites containing these two atoms, but it will also generate new layers of MS/MS spectra that can be complementary to the information provided by the ¹³C-labeled metabolites for the identification of metabolites (Fig. 6b).

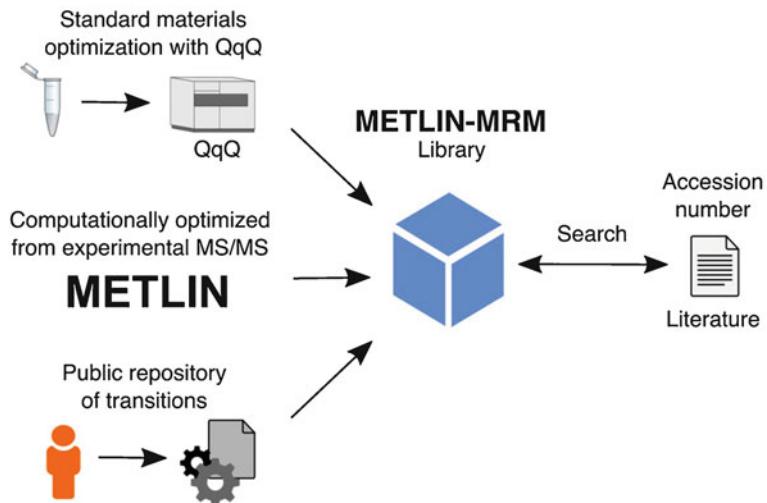


Fig. 7 METLIN-MRM ensembles three types of transitions: (1) transitions experimentally optimized by standard materials with QqQ via an established protocol, (2) transitions computationally (comp.) optimized by using METLIN's MS/MS spectra, and (3) public repository transitions. Experimentally and computationally optimized transitions were optimized for sensitivity and selectivity, respectively. METLIN-MRM also serves as a public repository (PR), in which the community can populate the library with new transitions

3.2 METLIN-MRM

The METLIN-MRM library consists of a small-molecule transitions for multiple-reaction monitoring (MRM) for more than 15,500 unique small molecules. It was developed to streamline absolute quantitation, which typically is accomplished with triple-quadrupole (QqQ) mass spectrometers configured to monitor a particular set of precursor–product ion transitions. However, in order to determine the transitions for the different molecules of interest, each target molecule must be optimized with pure standard materials. In the library, three different types of transitions are available: (1) traditional experimentally optimized transitions, (2) computationally optimized experimental transitions, and (3) public repository transitions (Fig. 7). Experimentally optimized transitions were acquired for more than 1000 molecules in both positive and negative mode by following established protocols [24]. These small molecule transitions were optimized for the highest intensity to achieve low limits of detection.

In addition to experimentally acquired data, transitions for more than 14,000 and 4700 molecules in positive and negative mode, respectively, were computationally optimized by using the METLIN spectral library [23] (acquired at different CE on a qToF instrument) by ranking empirical MS/MS fragments according to their selectivity (uniqueness of a product fragment for a given molecule). The developed ranking algorithm compares the MS/MS spectra of compounds with precursors within a ± 0.7 Da

window and selects fragments with the best selectivity without compromising sensitivity. This strategy enables high-throughput quantitation analysis as transitions are no longer required to be optimized with standard reference materials and, by the same token, minimizes errors caused by interfering molecules as transitions less likely to be masked are selected. For more detailed information about the algorithms and the selection process, we refer the readers to the main published work [25].

Lastly, METLIN-MRM also serves as a public repository, in which community members can upload transitions. Submitted lists of transitions are assigned with a unique accession number that can be used as a reference for publications. This process ultimately facilitates the deposition of transitions used in experiments and scientific literature into a standardized and searchable database, thereby increasing the traceability and reproducibility of experiments and the reuse/sharing of optimized transitions. Currently, 3300 transitions for more than 1500 small molecules are available from peer-reviewed publications with their corresponding original source (doi).

4 METLIN Population

METLIN contains experimental tandem mass spectrometry data on over 600,000 molecular standards and also has structures on over a million small molecules corresponding to a wide range of molecular classes. Over the years, small molecular entities have been incorporated into the library without bias for a particular class or type of compounds. Endogenous metabolites spanning the three-

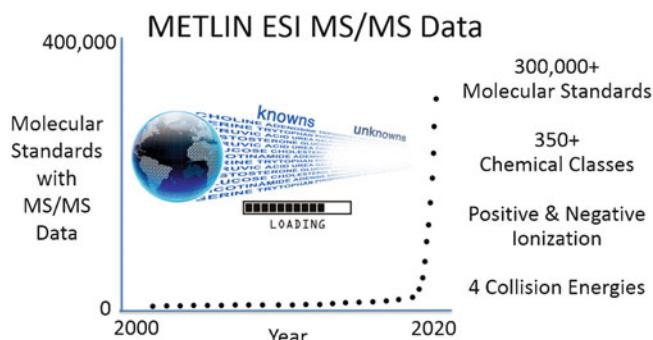


Fig. 8 Experimental electrospray ionization tandem mass spectrometry data has been generated on over ~300,000 molecular standards and incorporated into METLIN. This data was generated in both positive and negative ionization modes and at multiple collision energies (0, 10, 20, and 40 V). The rapid increase in the rate of METLIN growth is the culmination of multiple analytical and informatic challenges being overcome in 2018. *Analytical Chemistry* 2018, 90, 13128–13129. (Figure 1, with permission from ACS and RightsLink)

domain system (archaea, bacteria, and eukarya), modified metabolites, synthetic drugs and toxicants. The growth of the MS/MS database has been exponential in the last year, growing from 14,000 in 2017 to over 600,000 in November 2019 in both positive- and negative-ion mode at multiple collision energies (Fig. 8) [26]. A growth curve we anticipate will continue into the future.

5 Conclusion

In this chapter, METLIN's evolution since its beginnings, its tools for metabolite identification, and future developments have been discussed. It has adapted and developed its tools to not only facilitate the identification of known compounds (with or without MS/MS spectra) but also the discovery of unknown compounds. In the coming years, hundreds of thousands of small molecules will be characterized using the same strict MS/MS protocols and data curation. Further, given the ubiquitous coupling of preionization techniques and mass spectrometry instrumentation for untargeted metabolomics, tools for retention time prediction are currently being developed. Ultimately, it is safe to say that METLIN's goal from its inception of facilitating metabolomics research has not changed, although its library and tools have continuously evolve to meet the changing needs of modern metabolomics research as well as other chemical entities.

Acknowledgments

This research was partially funded by National Institutes of Health grants R35 GM130385, P30 MH062261, P01 DA026146 and U01 CA235493; and by Ecosystems and Networks Integrated with Genes and Molecular Assemblies (ENIGMA), a Scientific Focus Area Program at Lawrence Berkeley National Laboratory for the U.S. Department of Energy, Office of Science, Office of Biological and Environmental Research, under contract number DE-AC02-05CH11231. This research benefited from the use of credits from the National Institutes of Health (NIH) Cloud Credits Model Pilot, a component of the NIH Big Data to Knowledge (BD2K) program.

References

1. Smith CA, Maille GO, Want EJ, Qin C, Trauger SA, Brandon TR, Custodio DE, Abagyan R, Siuzdak G (2005) METLIN: a metabolite mass spectral database. *Ther Drug Monit* 27(6):747–751. <https://doi.org/10.1097/01.ftd.0000179845.53213.39>
2. Vinaixa M, Schymanski EL, Neumann S, Navarro M, Salek RM, Yanes O (2016) Mass spectral databases for LC/MS- and GC/MS-based metabolomics: state of the field and future prospects. *TrAC Trends Anal Chem* 78:23–35. <https://doi.org/10.1016/j.trac.2015.09.005>
3. Tautenhahn R, Cho K, Uritboonthai W, Zhu Z, Patti GJ, Siuzdak G (2012) An accelerated workflow for untargeted metabolomics using the METLIN database. *Nat Biotechnol* 30:826. <https://doi.org/10.1038/nbt.2348>
4. Benton HP, Wong DM, Trauger SA, Siuzdak G (2008) XCMS2: processing tandem mass spectrometry data for metabolite identification and structural characterization. *Anal Chem* 80:6382–6389. <https://doi.org/10.1021/ac800795f>
5. Benton HP, Ivanisevic J, Mahieu NG, Kurczy ME, Johnson CH, Franco L, Rinehart D, Valentine E, Gowda H, Ubhi BK, Tautenhahn R, Gieschen A, Fields MW, Patti GJ, Siuzdak G (2015) Autonomous metabolomics for rapid metabolite identification in global profiling. *Anal Chem* 87(2):884–891. <https://doi.org/10.1021/ac5025649>
6. Montenegro-Burke JR, Phommavongsay T, Aisporna AE, Huan T, Rinehart D, Forsberg E, Poole FL, Thorgersen MP, Adams MWW, Krantz G, Fields MW, Northen TR, Robbins PD, Niedernhofer LJ, Lairson L, Benton HP, Siuzdak G (2016) Smartphone analytics: mobilizing the lab into the cloud for omic-scale analyses. *Anal Chem* 88(19):9753–9758. <https://doi.org/10.1021/acs.analchem.6b02676>
7. Fiehn O, Barupal DK, Kind T (2011) Extending biochemical databases by metabolomic surveys. *J Biol Chem* 286(27):23637–23643. <https://doi.org/10.1074/jbc.R110.173617>
8. Horai H, Arita M, Kanaya S, Nihei Y, Ikeda T, Suwa K, Ojima Y, Tanaka K, Tanaka S, Aoshima K, Oda Y, Kakazu Y, Kusano M, Tohge T, Matsuda F, Sawada Y, Hirai MY, Nakanishi H, Ikeda K, Akimoto N, Maoka T, Takahashi H, Ara T, Sakurai N, Suzuki H, Shibata D, Neumann S, Iida T, Tanaka K, Funatsu K, Matsuura F, Soga T, Taguchi R, Saito K, Nishioka T (2010) MassBank: a public repository for sharing mass spectral data for life sciences. *J Mass Spectrom* 45(7):703–714. <https://doi.org/10.1002/jms.1777>
9. Wishart DS, Tzur D, Knox C, Eisner R, Guo AC, Young N, Cheng D, Jewell K, Arndt D, Sawhney S, Fung C, Nikolai L, Lewis M, Coutouly M-A, Forsythe I, Tang P, Srivastava S, Jeroncic K, Stothard P, Amegbey G, Block D, Hau DD, Wagner J, Miniaci J, Clements M, Gebremedhin M, Guo N, Zhang Y, Duggan GE, MacInnis GD, Weljie AM, Dowlatabadi R, Bamforth F, Clive D, Greiner R, Li L, Marrie T, Sykes BD, Vogel HJ, Querengesser L (2007) HMDB: the human metabolome database. *Nucleic Acids Res* 35(suppl_1):D521–D526. <https://doi.org/10.1093/nar/gkl923>
10. Wang M, Carver JJ, Phelan VV, Sanchez LM, Garg N, Peng Y, Nguyen DD, Watrous J, Kapono CA, Luzzatto-Knaan T, Porto C, Bouslimani A, Melnik AV, Meehan MJ, Liu W-T, Crüsemann M, Boudreau PD, Esquenazi E, Sandoval-Calderón M, Kersten RD, Pace LA, Quinn RA, Duncan KR, Hsu C-C, Floros DJ, Gavilan RG, Kleigrewe K, Northen T, Dutton RJ, Parrot D, Carlson EE, Aigle B, Michelsen CF, Jelsbak L, Sohlenkamp C, Pevzner P, Edlund A, McLean J, Piel J, Murphy BT, Gerwick L, Liaw C-C, Yang Y-L, Humpf H-U, Maansson M, Keyzers RA, Sims AC, Johnson AR, Sidebottom AM, Sedio BE, Klitgaard A, Larson CB, Boya PCA, Torres-Mendoza D, Gonzalez DJ, Silva DB, Marques LM, Demarque DP, Pociute E, O'Neill EC, Briand E, Helfrich EJN, Granatosky EA, Glukhov E, Ryffel F, Houson H, Mohimani H, Kharbush JJ, Zeng Y, Vorholt JA, Kurita KL, Charusanti P, McPhail KL, Nielsen KF, Vuong L, Elfeki M, Traxler MF, Engene N, Koyama N, Vining OB, Baric R, Silva RR, Mascuch SJ, Tomasi S, Jenkins S, Macherla V, Hoffman T, Agarwal V, Williams PG, Dai J, Neupane R, Gurr J, Rodríguez AMC, Lamasa A, Zhang C, Dorrestein K, Duggan BM, Almaliti J, Allard P-M, Phapale P, Nothias L-F, Alexandrov T, Litaudon M, Wolfender J-L, Kyle JE, Metz TO, Peryea T, Nguyen D-T, VanLeer D, Shinn P, Jadhav A, Müller R, Waters KM, Shi W, Liu X, Zhang L, Knight R, Jensen PR, Palsson BØ, Pogliano K, Linington RG, Gutiérrez M, Lopes NP, Gerwick WH, Moore BS, Dorrestein PC, Bandeira N (2016) Sharing and community curation of mass spectrometry data with global natural products social molecular networking. *Nat Biotechnol* 34:828. <https://doi.org/10.1038/nbt.3597>

11. Tsugawa H, Kind T, Nakabayashi R, Yukihira D, Tanaka W, Cajka T, Saito K, Fiehn O, Arita M (2016) Hydrogen rearrangement rules: computational MS/MS fragmentation and structure elucidation using MS-FINDER software. *Anal Chem* 88(16):7946–7958. <https://doi.org/10.1021/acs.analchem.6b00770>
12. Sud M, Fahy E, Cotter D, Brown A, Dennis EA, Glass CK, Merrill AH Jr, Murphy RC, Raetz CRH, Russell DW, Subramaniam S (2006) LMSD: LIPID MAPS structure database. *Nucleic Acids Res* 35(suppl_1): D527–D532. <https://doi.org/10.1093/nar/gkl838>
13. Warth B, Spangler S, Fang M, Johnson C, Forsberg E, Granados A, Domingo-Almenara X, Huan T, Rinehart D, Montenegro-Burke JR, Hilmers B, Aisporna AE, Hoang L, Uritboonthai W, Benton HP, Richardson S, Williams A, Siuzdak G (2017) Exposome-scale investigations guided by global metabolomics, pathway analysis, and cognitive computing. *Anal Chem* 89(21):11505–11513. in press
14. Wishart DS (2016) Emerging applications of metabolomics in drug discovery and precision medicine. *Nat Rev Drug Discov* 15:473. <https://doi.org/10.1038/nrd.2016.32>
15. Mylonas R, Mauron Y, Masselot A, Binz P-A, Budin N, Fathi M, Viette V, Hochstrasser DF, Lisacek F (2009) X-rank: a robust algorithm for small molecule identification using tandem mass spectrometry. *Anal Chem* 81(18):7604–7610. <https://doi.org/10.1021/ac900954d>
16. Domingo-Almenara X, Montenegro-Burke JR, Guijas C, Majumder ELW, Benton HP, Siuzdak G (2019) Autonomous METLIN-guided in-source fragment annotation for untargeted metabolomics. *Anal Chem* 91(5):3246–3253. <https://doi.org/10.1021/acs.analchem.8b03126>
17. Sumner LW, Amberg A, Barrett D, Beale MH, Beger R, Daykin CA, Fan TW-M, Fiehn O, Goodacre R, Griffin JL, Hankemeier T, Hardy N, Harnly J, Higashi R, Kopka J, Lane AN, Lindon JC, Marriott P, Nicholls AW, Reily MD, Thaden JJ, Viant MR (2007) Proposed minimum reporting standards for chemical analysis. *Metabolomics* 3(3):211–221. <https://doi.org/10.1007/s11306-007-0082-2>
18. Brenton AG, Godfrey AR (2010) Accurate mass measurement: terminology and treatment of data. *J Am Soc Mass Spectrom* 21(11):1821–1835. <https://doi.org/10.1016/j.jasms.2010.06.006>
19. Wolf S, Schmidt S, Müller-Hannemann M, Neumann S (2010) In silico fragmentation for computer assisted identification of metabolite mass spectra. *BMC Bioinformatics* 11(1):148. <https://doi.org/10.1186/1471-2105-11-148>
20. Cho K, Mahieu N, Ivanisevic J, Uriboothai W, Chen Y Jr, Siuzdak G, Patti GJ (2014) isoMETLIN: a database for isotope-based metabolomics. *Anal Chem* 86(19):9358–9361. <https://doi.org/10.1021/ac5029177>
21. Kurczy ME, Forsberg EM, Thorgersen MP, Poole FL, Benton HP, Ivanisevic J, Tran ML, Wall JD, Elias DA, Adams MWW, Siuzdak G (2016) Global isotope metabolomics reveals adaptive strategies for nitrogen assimilation. *ACS Chem Biol* 11(6):1677–1685. <https://doi.org/10.1021/acscchembio.6b00082>
22. Badur MG, Metallo CM (2018) Reverse engineering the cancer metabolic network using flux analysis to understand drivers of human disease. *Metab Eng* 45:95–108. <https://doi.org/10.1016/j.ymben.2017.11.013>
23. Guijas C, Montenegro-Burke JR, Domingo-Almenara X, Palermo A, Warth B, Hermann G, Koellensperger G, Huan T, Uriboothai W, Aisporna AE, Wolan DW, Spilker ME, Benton HP, Siuzdak G (2018) METLIN: a technology platform for identifying knowns and unknowns. *Anal Chem* 90(5):3156–3164. <https://doi.org/10.1021/acs.analchem.7b04424>
24. Lange V, Picotti P, Domon B, Aebersold R (2008) Selected reaction monitoring for quantitative proteomics: a tutorial. *Mol Syst Biol* 4(1):222. <https://doi.org/10.1038/msb.2008.61>
25. Domingo-Almenara X, Montenegro-Burke JR, Ivanisevic J, Thomas A, Sidibé J, Teav T, Guijas C, Aisporna AE, Rinehart D, Hoang L, Nordström A, Gómez-Romero M, Whiley L, Lewis MR, Nicholson JK, Benton HP, Siuzdak G (2018) XCMS-MRM and METLIN-MRM: a cloud library and public resource for targeted analysis of small molecules. *Nat Methods* 15(9):681–684. <https://doi.org/10.1038/s41592-018-0110-3>
26. Guijas C, Siuzdak G (2018) Reply to comment on METLIN: a technology platform for identifying knowns and unknowns. *Anal Chem* 90(21):13128–13129. <https://doi.org/10.1021/acs.analchem.8b04081>



Chapter 10

Metabolomic Data Exploration and Analysis with the Human Metabolome Database

David S. Wishart

Abstract

The Human Metabolome Database (HMDB) is a comprehensive, online, digital database designed to support the analysis and interpretation of metabolomic data acquired from human and/or mammalian metabolomic studies. This chapter covers three methods or protocols pertinent to using the HMDB: (1) understanding the general layout of the HMDB; (2) exploring the contents of a typical HMDB “MetaboCard”; and (3) an example of how HMDB can be used in a metabolomics study on human glioblastoma.

Key words Database, Metabolomics, Human, Disease, Data analysis

1 Introduction

The Human Metabolome Database (HMDB) is the primary vehicle for relaying knowledge collected from the Human Metabolome Project (HMP) to the public. The HMP was launched in 2005 [1] as a multi-institutional project based at the University of Alberta in Canada. Now entering its 15th year of operation, the HMP has been using advanced metabolomic techniques combined with computer-aided literature/text mining to compile as much information about the human metabolome as possible. Over the past 15 years the HMP has experimentally characterized a number of human biofluids and excreta, including the metabolome of human cerebrospinal fluid [2], human serum [3], human urine [4], human saliva [5], and human feces [6]. This information along with other information compiled from computer-aided text mining and manual curation has been used to assemble a much more extensive collection of human metabolites for many more tissues and biofluids. This information is periodically released to the public in the form of an online database, known as the HMDB, which is located at www.hmdb.ca [7, 8].

Simply stated, the HMDB is an open-access, web-enabled database that contains detailed information about essentially all known human metabolites. This detailed information includes fully referenced data about their biological roles, physiological concentrations, disease associations, chemical reactions, metabolic pathways, and reference mass spectrometry (MS) or nuclear magnetic resonance (NMR) spectra. Over the past 12 years [7], the HMDB has grown by a factor of nearly 50X and has rapidly evolved to meet the changing needs of both the metabolomics and clinical research communities. The most recent release of the HMDB (HMDB 4.0) contains 114,100 metabolites that are grouped into 4 major classes: (1) Detected and quantified, (2) Detected but not quantified, (3) Expected, and (4) Predicted [8]. “Detected” metabolites are those with measured concentrations or experimental confirmation of their existence in human biofluids, cells, or tissues. “Expected” metabolites are those with a known structure for which biochemical pathways are known and where human intake/exposure is frequent, but the compound has yet to be detected in the body. “Predicted” compounds represent a small number of highly feasible and biologically reasonable metabolites that have been generated through computational or “*in silico*” biotransformations using the program known as BioTransformer [9].

The HMDB is widely used by the metabolomics community and receives >5 million web hits each year. It is frequently accessed by researchers wishing to learn more about the structure, nomenclature, functions, or pathways of identified metabolites in both human and mammalian metabolomic studies. The HMDB is also used to obtain referential concentrations for medical diagnostic work, to identify compounds through spectral comparisons, and to learn more about the pathways or processes that involve metabolites identified in metabolomic experiments. Even though the HMDB explicitly includes the word “human” in its title, almost all of the metabolites and much of the information about these metabolites can be readily used when studying the metabolome of other mammalian models (mouse, rat, dog, cat, cow, etc.).

This chapter provides an overview of the HMDB layout, its contents, and an example of how to use it in a metabolomic study.

2 Materials

The HMDB is an online, web-accessible database. It is reachable via any modern computer or web-enabled device (smart phone or tablet) equipped with a reasonably up-to-date web browser (Google Chrome, Windows Internet Explorer, Apple Safari, Mozilla Firefox, or Opera) and a moderately high-resolution screen.

3 Methods

Four methods are covered here: (1) exploring the general layout of the HMDB; (2) exploring the contents of a typical HMDB “Meta-Card”; and (3) an example of how HMDB can be used in metabolomics study on human glioblastoma. All activities can be done with almost any internet-compatible device although a larger screen (18 cm or larger) is preferable.

3.1 Exploring the HMDB Layout

1. Open a preferred web browser and enter the following URL (web address): <http://www.hmdb.ca>.
2. The HMDB home page should be visible (Fig. 1).
3. On the top right is a Search box with a pull-down menu (with options for metabolites, diseases, pathways, proteins, reactions) and a blue search button. Users may enter text into the search box to search any of the categories in the pull-down menu. Enter the name “Alanine” into the search box and press the search button (or press the return key). A “Search Results” page should appear listing more than 100 matches to Alanine with the most likely match appearing at the top of the list (Fig. 2). See Note 1 regarding a more detailed explanation of what is seen in this page.
4. Press the “back” button on your browser or the “HMDB” icon (top left of the Search Results page) to return to the HMDB home page.

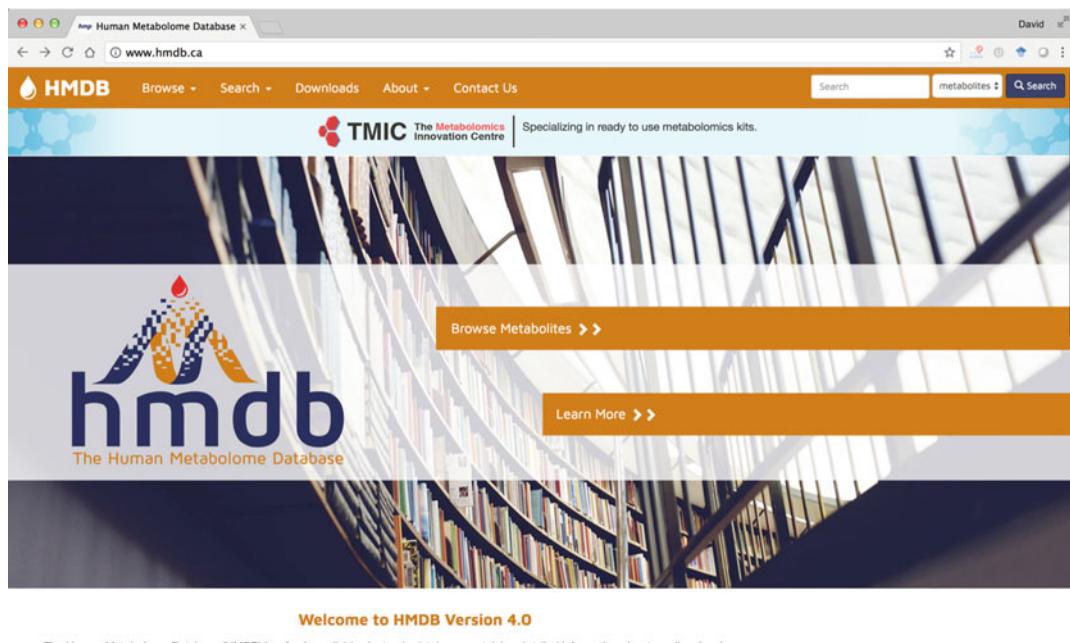


Fig. 1 A screenshot of the HMDB home page

The screenshot shows the HMDB website interface. At the top, there's a navigation bar with links for Browse, Search, Downloads, About, and Contact Us. A search bar contains the text "Alanine". Below the search bar, a message says "Searching metabolite for Alanine returned 122 results." A page header indicates "Displaying metabolites 1 - 25 of 122 in total". There are page navigation buttons for 1, 2, 3, 4, 5, Next, and Last.

Below the navigation, there are two filter sections: "Filter by metabolite status (default all)" and "Filter by biospecimen".

The first result listed is L-Alanine (HMDB0000161, 56-41-7). It includes its chemical structure (CC(C(=O)O)N), a brief description mentioning it's a non-essential amino acid, and a note about its high concentration in meat. It also lists its matched synonyms: (S)-(-)-Alanine, (S)-Alanine, and α-Alanine.

The second result listed is (alpha-D-mannosyl)7-beta-D-mannosyl-diacetylchitobiosyl-L-asparagine, isoform A (protein) (HMDB0062251, 302-72-7). It includes its chemical structure (CC(C(O)C(=O)O)C(O)C(=O)N), a brief description mentioning it's a derivative of alanine, and a note about its synthesis from alanine.

Fig. 2 A screenshot of the “Search Results” page for a search using Alanine as a text entry

5. Use a mouse or computer track pad to select “Browse” on the HMDB home page menu. A list of 14 browsing options will be displayed (Fig. 3). These are described in more detail in **Note 2**. Selecting any of these options will generate a specially formatted page that allows users to freely browse the data in each of these categories.
6. Use a mouse or computer track pad to select “Search” on the HMDB home page menu. A list of 11 search or query functions will be displayed (Fig. 4). These are described in more detail in **Note 3**. Selecting any of these options will generate a specially formatted page that allows users to search the HMDB according to the selected search type.
7. Use a mouse or computer track pad to select the “Downloads” option on the HMDB home page menu. Selecting this will generate a formatted, scrollable list of different files (for different versions of the HMDB) that can be downloaded to a local computer. These include protein and gene sequences (in FASTA format), chemical structures (in SDF format), metabolite/protein data (in XML format) and MS/NMR spectra (in XML format). The size of each file is displayed on the right. Clicking the dark blue “Download” button initiates the file download to your computer or device.

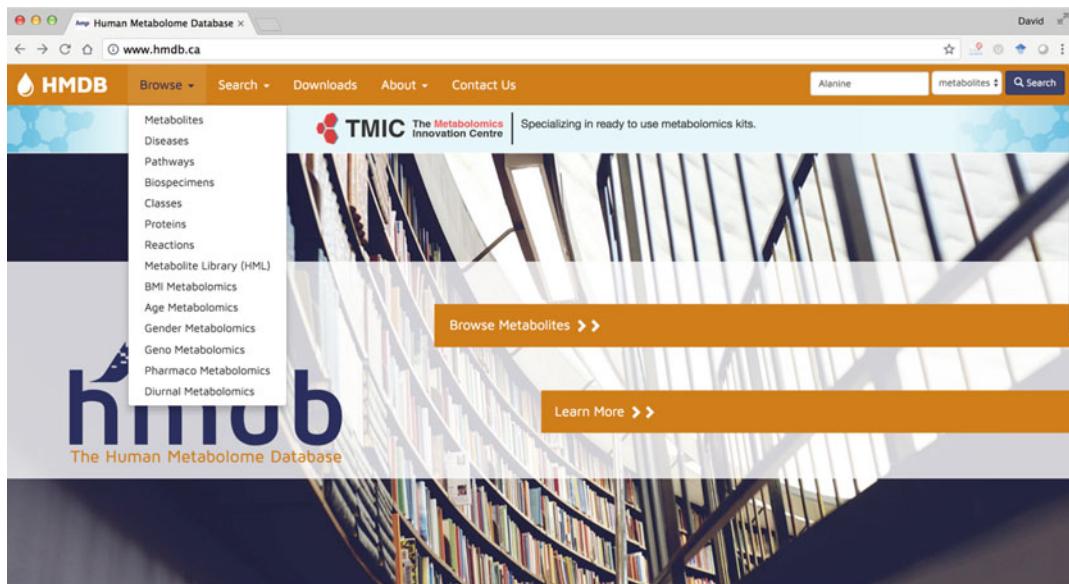


Fig. 3 A screenshot of the HMDB home page with the “Browse” menu displayed

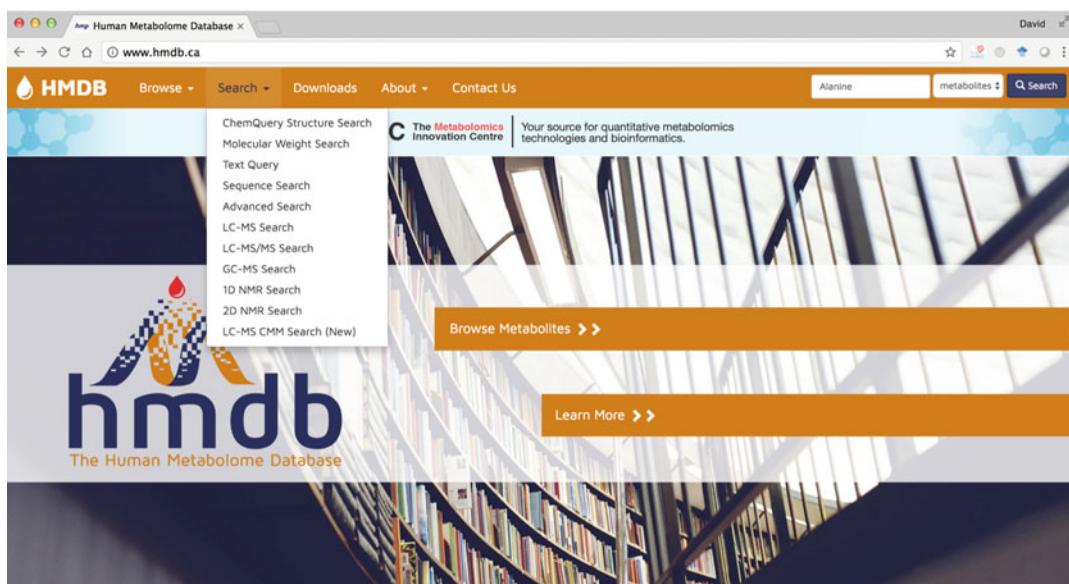


Fig. 4 A screenshot of the HMDB home page with the “Search” menu displayed

8. Press the “back” button on your browser or the “HMDB” icon (top left of the Search Results page) to return to the HMDB home page.
9. Use a mouse or computer track pad to select “About” on the HMDB home page menu. A list of 9 useful links concerning the HMDB will be displayed. These are described in more detail in **Note 4**. The function of the remaining links on the HMDB home page are either self-evident or redundant and will not be discussed here. Users are encouraged to click on the menu options found under “Search” and “Browse” to further explore each of the searching and browsing functions in the HMDB.
10. This completes the HMDB overview.

3.2 Exploring the Content of the HMDB MetaboCard

1. Go to the HMDB home page (<http://www.hmdb.ca>) and use your mouse or track pad to select “Browse” on the HMDB home page menu.
2. Select “Metabolites” from the pull-down menu (it is at the top of the pull-down list).
3. Scroll down the page, move the mouse pointer to the name of the first compound in the table (1-methylhistidine) and click on it. Users may alternately click on the tan-colored HMDB identifier (HMDB0000001) as this will yield the same result.
4. A “MetaboCard” for 1-Methylhistidine should appear that shows the Record Information along with some of the Metabolite identification data (Fig. 5). Additional details about the MetaboCard concept are given in **Note 5**.
5. Use a mouse or computer track pad to scroll down the page to view more of the Metabolite Identification data (*see Note 6*).
6. Use a mouse or computer track pad to scroll down the page to view more of the Chemical Taxonomy data (Fig. 6). Additional information about this section is given in **Note 7**.
7. Use a mouse or computer track pad to scroll down the page to view more of the Ontology data (Fig. 7). Additional information about this section is given in **Note 8**.
8. Use a mouse or computer track pad to scroll down the page to view more of the Physical Property data (Fig. 8).
9. Use a mouse or computer track pad to scroll down the page to view more of the Spectral data and Biological Property data (Fig. 9). Additional information about these sections is given in **Note 9**.
10. Use a mouse or computer track pad to scroll down the page to view data on the Normal Concentrations, Abnormal

The screenshot shows the HMDB website interface. At the top, there's a navigation bar with links for Browse, Search, Downloads, About, and Contact Us. A search bar is also present. Below the header, a banner displays 'Showing metabolocard for 1-Methylhistidine (HMDB0000001)'. The main content area is divided into several sections: 'Record Information' (listing version 4.0, status as detected and quantified, creation date 2005-11-16, update date 2019-01-11, and various IDs); 'Metabolite Identification' (listing the common name 1-Methylhistidine and a detailed description about its dietary sources and medical implications); and other tabs like Identification, Taxonomy, Ontology, Physical properties, Spectra, Biological properties, Concentrations, Links, References, and XML.

Fig. 5 A screenshot of the upper portion of the 1-Methylhistidine MetaboCard

This screenshot shows the 'Chemical Taxonomy' section of the MetaboCard. It includes a table with rows for 'Description', 'Kingdom', 'Super Class', 'Class', 'Sub Class', 'Direct Parent', 'Alternative Parents', and 'Substituents'. Each row contains a brief description or a list of categories. For example, 'Alternative Parents' lists various chemical classes such as L-alpha-amino acids, imidazolyl carboxylic acids, aralkylamines, and heteroaromatic compounds. The 'Substituents' row lists histidine derivatives and azole compounds.

Fig. 6 A screenshot of the Chemical Taxonomy data field from the 1-Methylhistidine MetaboCard

The screenshot shows the HMDB interface for the compound HMDB0000001. The top navigation bar includes links for Identification, Taxonomy, Ontology, Physical properties, Spectra, Biological properties, Concentrations, Links, References, enzymes (2), Show 2 proteins, and XML. The main content area is titled 'Ontology'.

Category	Details
Physiological effect	Health effect: Health condition: <ul style="list-style-type: none"> ◦ Kidney disease ◦ Diabetes mellitus type 2 Metabolism and nutrition disorders: <ul style="list-style-type: none"> ◦ Obesity
Disposition	Source: <ul style="list-style-type: none"> ◦ Endogenous Biological location: Tissue and substructures: <ul style="list-style-type: none"> ◦ Muscle ◦ Skeletal muscle Biofluid and excreta: <ul style="list-style-type: none"> ◦ Saliva ◦ Feces ◦ Urine ◦ Blood ◦ Cerebrospinal fluid Subcellular: <ul style="list-style-type: none"> ◦ Cytoplasm
Process	Naturally occurring process:

Fig. 7 A screenshot of the Ontology data field from the 1-Methylhistidine MetaboCard

The screenshot shows the HMDB interface for the compound HMDB0000001. The top navigation bar includes links for Identification, Taxonomy, Ontology, Physical properties, Spectra, Biological properties, Concentrations, Links, References, enzymes (2), Show 2 proteins, and XML. The main content area is titled 'Physical Properties'.

Category	Details																																	
State	Solid																																	
Experimental Properties	<table border="1"> <thead> <tr> <th>Property</th> <th>Value</th> <th>Reference</th> </tr> </thead> <tbody> <tr> <td>Melting Point</td> <td>249</td> <td>http://download.cappchem.com/data/Properties-of-Amino-Acids.pdf</td> </tr> <tr> <td>Boiling Point</td> <td>Not Available</td> <td>Not Available</td> </tr> <tr> <td>Water Solubility</td> <td>Not Available</td> <td>Not Available</td> </tr> <tr> <td>LogP</td> <td>Not Available</td> <td>Not Available</td> </tr> </tbody> </table>	Property	Value	Reference	Melting Point	249	http://download.cappchem.com/data/Properties-of-Amino-Acids.pdf	Boiling Point	Not Available	Not Available	Water Solubility	Not Available	Not Available	LogP	Not Available	Not Available																		
Property	Value	Reference																																
Melting Point	249	http://download.cappchem.com/data/Properties-of-Amino-Acids.pdf																																
Boiling Point	Not Available	Not Available																																
Water Solubility	Not Available	Not Available																																
LogP	Not Available	Not Available																																
Predicted Properties	<table border="1"> <thead> <tr> <th>Property</th> <th>Value</th> <th>Source</th> </tr> </thead> <tbody> <tr> <td>Water Solubility</td> <td>6.93 g/L</td> <td>ALOGPS</td> </tr> <tr> <td>logP</td> <td>-3.1</td> <td>ChemAxon</td> </tr> <tr> <td>pKa (Strongest Acidic)</td> <td>1.96</td> <td>ChemAxon</td> </tr> <tr> <td>pKa (Strongest Basic)</td> <td>9.25</td> <td>ChemAxon</td> </tr> <tr> <td>Physiological Charge</td> <td>0</td> <td>ChemAxon</td> </tr> <tr> <td>Hydrogen Acceptor Count</td> <td>4</td> <td>ChemAxon</td> </tr> <tr> <td>Hydrogen Donor Count</td> <td>2</td> <td>ChemAxon</td> </tr> <tr> <td>Polar Surface Area</td> <td>81.14 Å²</td> <td>ChemAxon</td> </tr> <tr> <td>Rotatable Bond Count</td> <td>3</td> <td>ChemAxon</td> </tr> <tr> <td>Refractivity</td> <td>42.39 m³·mol⁻¹</td> <td>ChemAxon</td> </tr> </tbody> </table>	Property	Value	Source	Water Solubility	6.93 g/L	ALOGPS	logP	-3.1	ChemAxon	pKa (Strongest Acidic)	1.96	ChemAxon	pKa (Strongest Basic)	9.25	ChemAxon	Physiological Charge	0	ChemAxon	Hydrogen Acceptor Count	4	ChemAxon	Hydrogen Donor Count	2	ChemAxon	Polar Surface Area	81.14 Å²	ChemAxon	Rotatable Bond Count	3	ChemAxon	Refractivity	42.39 m³·mol⁻¹	ChemAxon
Property	Value	Source																																
Water Solubility	6.93 g/L	ALOGPS																																
logP	-3.1	ChemAxon																																
pKa (Strongest Acidic)	1.96	ChemAxon																																
pKa (Strongest Basic)	9.25	ChemAxon																																
Physiological Charge	0	ChemAxon																																
Hydrogen Acceptor Count	4	ChemAxon																																
Hydrogen Donor Count	2	ChemAxon																																
Polar Surface Area	81.14 Å²	ChemAxon																																
Rotatable Bond Count	3	ChemAxon																																
Refractivity	42.39 m³·mol⁻¹	ChemAxon																																

Fig. 8 A screenshot of the Physical Property data field from the 1-Methylhistidine MetaboCard

The screenshot shows the HMDB MetaboCard for 1-Methylhistidine. At the top, there's a navigation bar with tabs like Identification, Taxonomy, Ontology, Physical properties, Spectra, Biological properties, Concentrations, Links, References, enzymes (2), Show 2 proteins, and XML. Below the navigation bar, there are two main sections: Spectra and Biological Properties.

Spectra:

Spectrum Type	Description	Splash Key	View in MoNA
GC-MS	GC-MS Spectrum - GC-MS (2 TMS)	splash10-0002-961000000-d0147d3e28362f91174a	View in MoNA
GC-MS	GC-MS Spectrum - GC-MS (3 TMS)	splash10-0gbd-4941000000-cee19577c72eed95aec	View in MoNA
GC-MS	GC-MS Spectrum - GC-MS (Non-derivatized)	splash10-0002-961000000-d0147d3e28362f91174a	View in MoNA
GC-MS	GC-MS Spectrum - GC-MS (Non-derivatized)	splash10-0gbd-4941000000-cee19577c72eed95aec	View in MoNA
Predicted GC-MS	Predicted GC-MS Spectrum - GC-MS (Non-derivatized) - 70eV, Positive	splash10-00dm-950000000-e50450f2dc8a5d4b3655	View in MoNA
Predicted GC-MS	Predicted GC-MS Spectrum - GC-MS (1 TMS) - 70eV, Positive	splash10-0odi-931000000-d67eb07e6f5ecb74263	View in MoNA

Biological Properties:

- Cellular Locations:** Cytoplasm
- Biospecimen Locations:** Blood, Cerebrospinal Fluid (CSF), Feces, Saliva, Urine
- Tissue Locations:** Muscle, Skeletal Muscle

Pathways:

Name	SMPDB/Pathwhiz	KEGG
Histidine Metabolism		
Histidinemia		Not Available

Fig. 9 A screenshot of the Spectral data and Biological Property data fields from the 1-Methylhistidine MetaboCard

Concentrations, and Associated Disorders and Diseases (Fig. 10). More information about these sections is given in **Note 10**.

11. Use a mouse or computer track pad to scroll down the page to view External Links and References (Fig. 11).
12. Use a mouse or computer track pad to scroll down the page to view the Enzymes page (Fig. 12). More information about the enzymes is given in **Note 11**.
13. This completes the HMDB MetaboCard overview.

3.3 Using the HMDB for Disease Studies (Glioblastoma)

1. Go to the HMDB home page (<http://www.hmdb.ca>) and use your mouse or track pad to select “Search” on the HMDB home page menu.
2. The drop-down menu will display several search options including structure, sequence, text, spectral and molecular weight searches. For this example select the “LC-MS Search” option (Fig. 13).
3. As explained in **Note 12**, the following m/z values were found to be significantly altered in cerebrospinal fluid samples between patients with glioblastoma (brain cancer) and healthy controls: 149.0444, 117.0183, 119.0339, 91.0388, 147.0763, 193.0341, 308.0911. Enter these numbers into the “Query Masses” box as shown in Fig. 14.

Normal Concentrations								
Biospecimen	Status	Value	Age	Sex	Condition	Reference	Details	
Blood	Detected and Quantified	7.7 +/- 1.9 uM	Adult (>18 years old)	Both	Normal	7061274		
Blood	Detected and Quantified	14.4 +/- 2.3 uM	Adult (>18 years old)	Both	Normal	7061274		
Blood	Detected and Quantified	19.6 +/- 2.6 uM	Adult (>18 years old)	Both	Normal	7061274		
Blood	Detected and Quantified	12.7 +/- 2.9 uM	Adult (>18 years old)	Both	Normal	7061274		
Show more...								
Abnormal Concentrations								
Biospecimen	Status	Value	Age	Sex	Condition	Reference	Details	
Blood	Detected and Quantified	51.2 +/- 17.6 uM	Adult (>18 years old)	Female	Pregnancy with fetuses with trisomy 18	23535240		
Blood	Detected and Quantified	43.4 +/- 22.3 uM	Adult (>18 years old)	Female	Pregnancy	23535240		
Blood	Detected and Quantified	50.7 (12.9) uM	Adult (>18 years old)	Female	Early preeclampsia	22494326		
Blood	Detected and Quantified	50.0 (14.6) uM	Adult (>18 years old)	Female	Pregnancy	22494326		
Show more...								
Associated Disorders and Diseases								
Disease References	Kidney disease							
	1. Raj DS, Ouwendyk M, Francoeur R, Pierratos A: Plasma amino acid profile on nocturnal hemodialysis. <i>Blood Purif.</i> 2000;18(2):97-102. [PubMed:10838467]							
	Early preeclampsia							
	1. Bahado-Singh RO, Akolekar R, Mandal R, Dong E, Xia J, Kruger M, Wishart DS, Nicolaides K: Metabolomics and first-trimester prediction of early-onset preeclampsia. <i>J Matern Fetal Neonatal Med.</i> 2012 Oct;25(10):1840-7. doi: 10.3109/14767058.2012.680254. Epub 2012 Apr 28. [PubMed:22494326]							
	Pregnancy							

Fig. 10 A screenshot of the Normal and Abnormal Concentrations data fields from the 1-Methylhistidine MetaboCard

External Links	
DrugBank ID	DB04151
Phenol Explorer Compound ID	Not Available
FoodDB ID	FDB012119
KNAPSAcK ID	Not Available
Chemspider ID	83153
KEGG Compound ID	C01152
BioCyc ID	CPD-1823
BIGG ID	Not Available
Wikipedia Link	Not Available
METLIN ID	3741
PubChem Compound	92105
PDB ID	HIC
ChEBI ID	50599
References	
Synthesis Reference	Jain, Rahul; Cohen, Louis A. Regiospecific alkylation of histidine and histamine at N-1 (t). <i>Tetrahedron</i> (1996), 52(15), 5363-70.
Material Safety Data Sheet (MSDS)	Download (PDF)
General References	<ol style="list-style-type: none"> Colombani PC, Kovacs E, Frey-Rindova P, Frey W, Langhans W, Arnold M, Wenk C: Metabolic effects of a protein-supplemented carbohydrate drink in marathon runners. <i>Int J Sport Nutr.</i> 1999 Jun;9(2):181-201. [PubMed:10362454] Nicholson JK, Foxall PJ, Spraul M, Farrant RD, Lindon JC: 750 MHz 1H and 1H-13C NMR spectroscopy of human blood plasma. <i>Anal Chem.</i> 1995 Mar 1;67(5):793-811. [PubMed:7762816] Myint T, Fraser GE, Lindsted KD, Knutson SF, Hubbard RW, Bennett HW: Urinary 1-methylhistidine is a marker of meat consumption in Black and in White California

Fig. 11 A screenshot of the External Links and References data fields from the 1-Methylhistidine MetaboCard

Enzymes

1. Beta-Ala-His dipeptidase

General function: Involved in metallopeptidase activity
 Specific function: Preferential hydrolysis of the beta-Ala- β -His dipeptide (carnosine), and also anserine, Xaa- β -His dipeptides and other dipeptides including homocarnosine
 Gene Name: CNDP1
 Uniprot ID: Q96KN2
 Molecular weight: 56691.6

References

1. Fleisher LD, Rassin DK, Wisniewski K, Salwen HR: Carnosinase deficiency: a new variant with high residual activity. Pediatr Res. 1980 Apr;14(4 Pt 1):269-71. [PubMed:7375183]

2. Protein arginine N-methyltransferase 3

General function: Involved in protein methyltransferase activity
 Specific function: Methylates (mono and asymmetric dimethylation) the guanidino nitrogens of arginyl residues in some proteins
 Gene Name: PRMT3
 Uniprot ID: O60678
 Molecular weight: 59902.7

References

1. Raghavan M, Lindberg U, Schutt C: The use of alternative substrates in the characterization of actin-methylating and carnosine-methylating enzymes. Eur J Biochem. 1992 Nov 15;210(1):311-8. [PubMed:1446680]

Fig. 12 A screenshot of the Enzymes and Transporters page from the 1-Methylhistidine MetaboCard

Spectra Search Mass Spectrum

LC-MS Search LC-MS/MS Search GC-MS Search 1D NMR Search 2D NMR Search

Query Masses (Da)

Enter one mass per line (maximum 700 query masses per request)

Ionization

Ion Mode Positive

Adduct Type

- Unknown
- M+H
- M+H₂O
- M+H-H₂O
- M+NH₄-H₂O
- M+Li
- M+Na

Hold Ctrl (Windows) or Command (Mac) to select multiple adducts

Molecular Weight Tolerance ±

Fig. 13 A screenshot of the “LC-MS Search” page from the HMDB

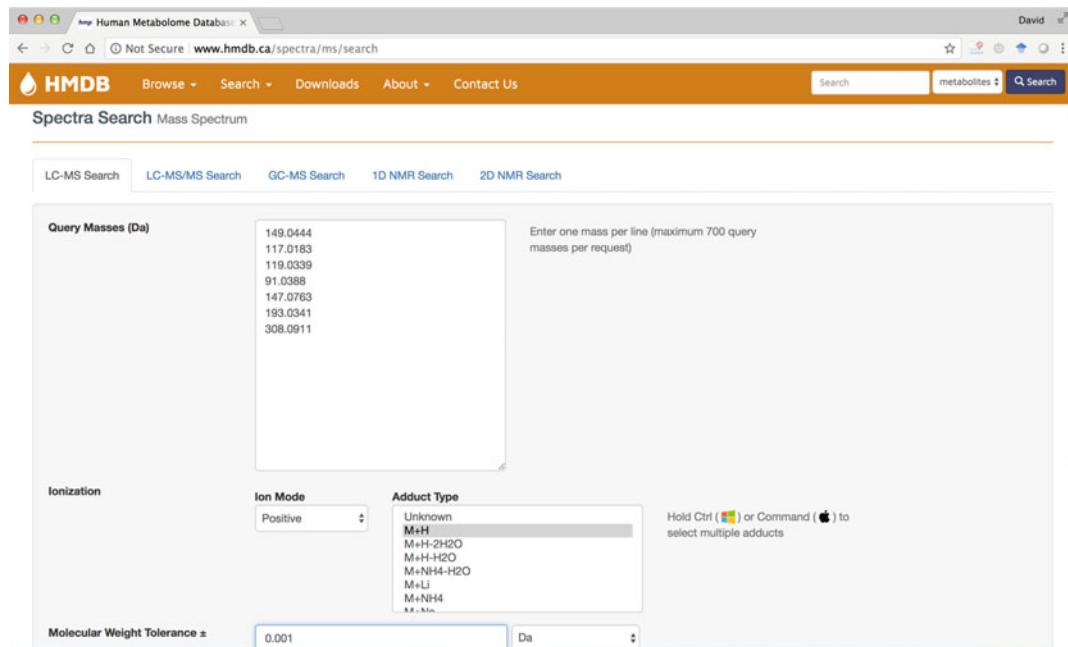


Fig. 14 A screenshot of the “LC-MS Search” page with the 149.0444, 117.0183, 119.0339, 91.0388, 147.0763, 193.0341, 308.0911 m/z values entered into the “Query Masses” box

4. Ensure the Ion Mode is marked positive and the molecular weight tolerance is set to 0.001 Da. Choose the adduct type to be “M + H”. Press the “Search” button.
5. After a few seconds, a list of the top scoring hits will appear as shown in Fig. 15. *See Note 13* for a more detailed explanation of how to interpret the resulting table.
6. From the list of top matches, click on any HMDB hyperlink under the “Compound” column on the left. For example, click on the HMDB0000606 MetaboCard link. This opens up the HMDB MetaboCard for d-2-hydroxyglutarate (Fig. 16).
7. Read through the information on d-2-hydroxyglutarate to learn more about its known links to cancer and its role as an oncometabolite. Click on the corresponding pathway links or pathway thumbnail images in the MetaboCard to learn more about the metabolism and mechanism of action of this oncometabolite (*see Note 14*).
8. Repeat the above process for the remaining identified metabolites including: fumarate, succinate, lactate, glutamine, isocitrate, and glutathione. *See Note 15* for more detailed information about how these metabolites were identified.
9. This concludes the section on using the HMDB for disease studies.

Compound	Name	Formula	Monoisotopic Mass	Adduct	Adduct M/Z	Delta (ppm)
HMDB0000428	3-Hydroxyglutaric acid	C5H8O5	148.0372	M+H	149.0444	0 m/z calculator
HMDB0059655	2-Hydroxyglutarate	C5H8O5	148.0372	M+H	149.0444	0 m/z calculator
HMDB0011676	D-Xylo-1,5-lactone	C5H8O5	148.0372	M+H	149.0444	0 m/z calculator
HMDB00001900	Ribonolactone	C5H8O5	148.0372	M+H	149.0444	0 m/z calculator
HMDB0000606	D-2-Hydroxyglutaric acid	C5H8O5	148.0372	M+H	149.0444	0 m/z calculator
HMDB0000426	Citramalic acid	C5H8O5	148.0372	M+H	149.0444	0 m/z calculator
HMDB0000694	L-2-Hydroxyglutaric acid	C5H8O5	148.0372	M+H	149.0444	0 m/z calculator
HMDB0062739	3-methylmalate(2-)	C5H8O5	148.0372	M+H	149.0444	0 m/z calculator
HMDB0038890	Methyl 3-methyl-1-buteneyl disulfide	C6H12S2	148.0380	M+H	149.0453	6 m/z calculator

Fig. 15 A screenshot showing the results of the “LC-MS Search” using the seven entered masses or *m/z* values

Metabolite Identification

Common Name	D-2-Hydroxyglutaric acid
Description	In humans, D-2-hydroxyglutaric acid is formed by a hydroxacyacid-oxoacid transhydrogenase whereas in bacteria it is formed by a 2-hydroxyglutarate synthase. D-2-Hydroxyglutaric acid is also formed via the normal activity of hydroxacyacid-oxoacid transhydrogenase during conversion of 4-hydroxybutyrate to succinate semialdehyde. The compound can be converted to alpha-ketoglutaric acid through the action of a 2-hydroxyglutarate dehydrogenase (EC 1.1.99.2). In humans, there are two such enzymes (D2HGDH and L2HGDH). Both the D and the L stereoisomers of hydroxyglutaric acid are found in body fluids. D-2-Hydroxyglutaric acid is a biochemical hallmark of the inherited neurometabolic disorder D-2-hydroxyglutaric aciduria (OMIM: 600721). And the genetic disorder glutaric aciduria II. D-2-Hydroxyglutaric aciduria (caused by loss of D2HGDH or gain of function of IDH) is rare, with symptoms including cancer, macrocephaly, cardiomyopathy, mental retardation, hypotonia, and cortical blindness. An elevated urine level of D-2-hydroxyglutaric acid has been reported in patients with spondyloenchondroplasia (OMIM: 271550). D-2-Hydroxyglutaric acid can be converted to alpha-ketoglutaric acid through the action of 2-hydroxyglutarate dehydrogenase (D2HGDH). Additionally, the enzyme D-3-phosphoglycerate dehydrogenase (PHGDH) can catalyze the NADH-dependent reduction of alpha-ketoglutarate (AKG) to D-2-hydroxyglutarate (D-2HG). Nyhan et al. (1995) described 3 female patients, 2 of them sibs, who were found to have excess accumulation of D-2-hydroxyglutaric acid in the urine. The phenotype was quite variable, even among the sibs, but included mental retardation, macrocephaly with cerebral atrophy, hypotonia, seizures, and involuntary movements. One of the patients developed severe intermittent vomiting and was given a pyloromyotomy. The electroencephalogram demonstrated hypersynchrony. There was an increased concentration of protein in cerebrospinal fluid, an unusual finding in inborn errors of metabolism. D-2-Hydroxyglutaric acid can also be produced via gain-of-function mutations in the cytosolic and mitochondrial isoforms of isocitrate dehydrogenase (IDH). IDH is part of the TCA cycle and this compound is generated in high abundance when IDH is mutated. Since D-2-hydroxyglutaric acid is sufficiently similar in structure to 2-oxoglutarate (2OG), it is able to inhibit a range of 2OG-dependent dioxygenases, including histone lysine demethylases (KDMs) and members of the ten-eleven translocation (TET) family of 5-methylcytosine (5mC) hydroxylases. This inhibitory effect leads to alterations in the hypoxia-inducible factor (HIF)-mediated hypoxic response and alterations in gene expression through global epigenetic remodeling. Depending on the circumstances, D-2-hydroxyglutaric acid causes a cascading effect that leads genetic perturbations and malignant transformation. Depending on the circumstances, D-2-hydroxyglutaric acid can act as an oncometabolite, a neurotoxin, an acidogen, and a metabotxin. An oncometabolite is a compound that promotes tumour growth and survival. A neurotoxin is a compound that is toxic to neurons or neural tissue. An acidogen is an acidic compound that induces acidosis, which has multiple adverse effects on many organ systems. A metabotxin is an endogenously produced metabolite that causes adverse health effects at chronically high levels. As an oncometabolite, D-2-hydroxyglutaric acid is a competitive inhibitor of multiple alpha-ketoglutarate-dependent dioxygenases, including histone demethylases and the TET family of 5mC hydroxylases. As a result, high levels of 2-hydroxyglutarate lead to genome-wide histone and DNA methylation alterations, which in turn lead to mutations that ultimately cause cancer (PMID: 29038145). As a neurotoxin, D-2-hydroxyglutaric acid mediates its neurotoxicity through activation of N-methyl-D-aspartate receptors. D-2-Hydroxyglutaric acid is structurally similar to the excitatory amino acid glutamate and stimulates neurodegeneration by mechanisms similar to glutamate, NMDA, or mitochondrial toxins (PMID: 12113528). As an acidogen, D-2-hydroxyglutaric acid is classified as an alpha hydroxy acid belonging to the general class of compounds known as organic acids. Chronically high levels of D-2-hydroxyglutaric acid are a feature of the inborn error of metabolism called D-2-hydroxyglutaric aciduria. Abnormally high levels of organic acids in the blood (organic acidemia), urine (organic aciduria), the brain, and other tissues lead to general metabolic acidosis. Acidosis typically occurs when arterial pH falls below 7.35. In infants with acidosis, the initial symptoms include poor feeding, vomiting, loss of appetite, weak muscle tone (hypotonia), and lack of energy (lethargy). These can progress to heart abnormalities, kidney abnormalities, liver damage, seizures, coma, and possibly death. These are also the characteristic symptoms of

Fig. 16 A screenshot of the MetaboCard for D-2-hydroxyglutarate

4 Notes

1. The Search Results page has several components. At the top left of the page are “change page” or page selector icons that allow users to quickly select and display different result pages consisting of 25 metabolites per page. Users can select a page by number or they may select the next page or the last page of results. Below the page selector panel is a greyed-in area for filtering or limiting the displayed results. Users can filter by metabolite status (detect, expected, predicted) or by biospecimen type (blood, urine, saliva, etc.), which refers to the biofluid or tissue location(s) where the metabolite is found. Using a mouse or track-pad, users can click on one of the desired check boxes after which they can press on the “Apply Filter” button on the far right. Applying the filter generates a new “Search Results” page with the results filtered according to the user selection. Below the filter selection box is a scrollable that displays the hits from the search. On the left side is a tan-colored button that displays the compound’s HMDB identifier. Clicking on this button will display the compound’s corresponding MetaboCard. The number below each MetaboCard button is the CAS (Chemical Abstract Services) identifier, if available. To the right of the MetaboCard button (in the same metabolite row) is the common name and the IUPAC name of the compound along with the text in which the word “Alanine” has matched to the text in the corresponding MetaboCard (highlighted in yellow). The biofluid location in which the metabolite has been found is indicated in the dark grey boxes at the top of each metabolite row. On the far right of each metabolite row is a 2D structure of the metabolite drawn in an electronically neutral format.
2. The HMDB has 14 different browsing options. The most frequently chosen option is the “Metabolite” category, which is listed at the top of the pull-down menu. When selected, the metabolite browse option allows users to interactively view, scroll through and filter metabolite data. The data is presented in a structured table that provides synoptic information (name, structure, chemical formula, biofluid source) about the 110,000+ metabolites in the HMDB. The filtering option allows users to select metabolites based on their status (quantified, expected, predicted, etc.), biospecimen type (urine, blood, etc.), general origin (microbial, food, endogenous, etc.), and subcellular location. Each metabolite is linked to a specific “MetaboCard,” which is described in more detail in Subheading 3.2. Selecting the “Diseases” browsing option allows users to view a structured, scrollable table that lists (age and gender-specific) metabolite concentration data for

>650 different diseases along with their published or online references. Likewise, selecting the “Pathways” browsing option allows users to view a structured, scrollable table containing data on nearly 50,000 metabolic pathways. The pathway table contains expandable thumbnail images along with hyperlinks to the metabolites in each pathway. Other browsing categories offered in HMDB’s Browse menu allow users to view metabolites via their biospecimen or biofluid of origin (“Biospecimens”), by their chemical class, as determined via ClassyFire [10], (“Class”), by the 5700+ known enzymes or protein transporters that act on human metabolites (“Proteins”), or by the 18,000+ enzymatic reactions in the HMDB (“Reactions”). Other browsing options support the display of known metabolites (along with their references) that vary with body mass index or BMI (“BMI Metabolomics”), with age (“Age Metabolomics”), with gender or sex (“Gender Metabolomics”), diurnal cycle (“Diurnal Metabolomics”), drug intake (“Pharmaco Metabolomics”), and single nucleotide polymorphism or SNP type (“Geno Metabolomics”).

3. The HMDB has 11 different search options. The most frequently chosen option is the “ChemQuery Structure Search” which is listed at the top of the pull-down menu. This search function allows users to draw a structure using the MarvinSketch applet from ChemAxon and to search for similar structures throughout the HMDB’s collection of 110,000+ structures. The applet also allows users to paste in a SMILES [11] or InChI [12] identifier directly into the drawing canvas to automatically generate the corresponding query structure. These structure searches may be done via structure similarity scores or by substructure matching and the results may be filtered by molecular weight, number of results or metabolite status. Other searches that are supported include a simple “Molecular Weight Search,” a “Text Search” that supports Boolean (AND, OR, NOT) text queries, a BLAST-based “Sequence Search” (to look for protein or gene sequence similarities among HMDB’s 5000+ protein/gene sequences) and a wide variety of spectral searches (MS and NMR). The MS (LC-MS, LC-MS/MS, LC-MS CMM, and GC-MS) searches allow users to put in lists of *m/z* peaks, mass tolerances and to select various adducts (for LC-MS data). These queries are then searched against HMDB’s collection of 350,000 MS spectra. The NMR searches allow users to paste in lists of chemical shifts (for 1D or 2D NMR spectra) and to search against HMDB’s collection of ~4000 ¹H and ¹³C NMR spectra. Another important search offering is the HMDB’s “Advanced Search” (for metabolites or concentrations). This search option allows users to selectively search for numbers, numeric ranges or text from

more than 30 HMDB data fields using a variety of conditions (matches, does not match, starts with, etc.). Effectively, “Advanced Search” functions as a user-friendly structured query language (SQL) search tool.

4. The “About” pull-down menu offers 10 options that describe additional details about the HMDB. The “About the HMDB” option provides a short description of the current version of the HMDB. The “Release Notes” contains detailed information about each HMDB release over the past 12 years while the “Citing the HMDB” provides detailed publication information about how to properly cite the database. The “Statistics” option lists very extensive and up-to-date information regarding the numbers of compounds, names, synonyms, spectra, proteins, genes, reactions, diseases, and so on. in the HMDB. This information is updated regularly. The “Data Sources” provides information about how the HMDB was assembled and where the information was obtained while the “Other Databases” offers short descriptions and hyperlinks to a number of popular, alternative metabolomics databases. Perhaps the most useful link in the “About” menu is the “Help/Tutorial” option. This links users to the HMDB tutorial video, a 10-min video (also available through YouTube) that provides a nice introduction to the HMDB and how to use its many browse, search and query options.
5. Most of the viewable data in the HMDB is contained in series of synoptic summary tables called “MetaboCards” or metabolism information cards. Each MetaboCard, which is associated with a specific metabolite, contains >130 data fields with approximately two-thirds of the data fields associated with chemical or physicochemical data and the other one third associated with biological or biomedical data. The data fields in each MetaboCard are grouped into 14 distinctive categories which include: (1) record information, (2) metabolite identification, (3) chemical taxonomy, (4) chemical ontology, (5) physical properties, (6) spectra, (7) biological properties, (8) normal concentrations, (9) abnormal concentrations, (10) associated disorders, (11) external links, (12) references, (13) enzymes, and (14) transporters.
6. The Metabolite Identification data field in the HMDB contains 13 different pieces of information that can be used to identify or characterize a metabolite. These include common names, the IUPAC name, synonyms, structure images (2D and 3D), structure files, the chemical formula, the molecular weight, the SMILES string, and InChI identifiers. The most useful piece of information in the identification field is the metabolite description. This typically provides a short 100–200 word description of the metabolite and its biological role(s) or origin. Every

metabolite in the HMDB has a hand-written metabolite description.

7. The Chemical Taxonomy data field contains 10 pieces of information that describe each metabolite's structural features and their relationships to each other. The HMDB's taxonomy is automatically generated via the program called ClassyFire [10]. ClassyFire takes a chemical structure, analyzes it and then classifies it into a fully defined, hierarchical taxonomy. The ClassyFire chemical taxonomy (like the Linnean taxonomy in biology) includes a chemical kingdom, superclass, class, and subclass as well as the chemical parents and substituents. Each descriptor in the ClassyFire taxonomy has a formal, written definition and many of these definitions are accessible via hyperlinks in the HMDB. The point of a structural taxonomy is to enable consistent chemical classifications and consistent chemical descriptions across labs and across publications. The ClassyFire chemical taxonomy has been adopted by most of the world's major chemical databases, including PubChem [13], LIPID MAPS [14], ChEBI [15], and others.
8. The Ontology data field contains four subfields that describe each metabolite's function in the HMDB through a structured language and a standard set of definitions. An ontology is a set of concepts in a subject area that shows their properties and the relations between them. The point of an ontology is to enable consistent descriptions and to assist in computational text mining. HMDB's functional ontology is modeled after the Gene Ontology or GO [16] that is widely used in molecular biology and genetics. The main categories used in HMDB's Ontology cover (1) Physiological effects; (2) Disposition (source or origin); (3) Process (involvement biological pathways as well as natural or industrial processes); and (4) Role (industrial or biological application).
9. The Spectra data field contains a scrollable list of all the GC-MS, LC-MS, LC-MS/MS and NMR spectra collected for each metabolite in the HMDB. The spectra listed in this field include both experimentally observed as well as predicted spectra. Each spectrum has a short description of the instrument or collection conditions and a link to enable facile viewing of the selected spectrum with one of two interactive, online spectral viewers (JSpectraViewer is the most commonly used). The Biological Properties field contains four subfields that display detailed information about the known cellular locations of the metabolite, the known biofluids or excreta in which the metabolite is found, the tissue in which the metabolite is found as well as the biological pathways that use, produce or process the metabolite. The pathways subfield shows the name of the

pathway as well as clickable thumbnail views of the pathways as made available by either by KEGG [17] or SMPDB [18].

10. Many MetaboCards in the HMDB have relatively comprehensive information on Normal Concentration and Abnormal Concentrations. These two data fields contain specific information about metabolite concentrations in different biofluids or excreta. Information about age, sex, and (if abnormal) the associated condition is presented along with links to the PubMed references and any additional notes. The diseases or conditions identified in the Abnormal Concentrations field are further elaborated in the Associated Disorders and Diseases data field, which includes disease references and, if appropriate, links to the Online Mendelian Inheritance of Man (OMIM) database [19].
11. The last data field in the HMDB displays the enzymes and/or transporters that are associated with each metabolite. These data fields or data boxes are framed with a green border to make them visibly distinct from each metabolite's MetaboCard. Each of these enzyme/protein boxes contains the protein name (which is hyperlinked) along with brief descriptions about the general function, specific function, gene name, UniProt ID, estimated molecular weight, and known reactions. Clicking on the protein name (or the "Enzyme Details" box on the right) will open up a richly detailed "Protein Card" containing dozens of pieces of information about the protein, its amino acid and DNA sequence, its functions, pathways, reactions, GO classification, and various measurable gene or protein properties.
12. The example chosen here is fictitious but is based on published information from several metabolomic studies on glioblastoma (a form of brain cancer). Assume that a high resolution (OrbiTrap or QTOF) MS experiment has been completed on a set of 30 cerebrospinal fluid (CSF) samples collected from 30 patients with glioblastoma and another set of 30 CSF samples from 30 patients with no cancer symptoms (but with suspected meningitis, spinal fractures, etc.) that required the mandatory collection of CSF. Using the positive ion mode, peaks with the following m/z values were found to be significantly increased in the cancer patients: 149.0444, 117.0183, 119.0339, 91.0388 while the peaks with the following m/z values were found to be significantly decreased in the cancer patients relative to the non-cancer "controls": 147.0763, 193.0341, 308.0911. Our task is to use the HMDB to find out what these compounds are, how they might be involved in glioblastoma and to identify which biochemical or signaling pathways they may be associated with.

13. The table that displays the mass hits includes 7 subtables with a subtable of mass matches for each of the 7 query masses. The first subtable shows matches for the m/z 149.0444 query, the second subtable shows mass matches for the m/z 117.0183 query, and so on. In some cases up to 10–12 metabolites with identical masses or m/z values (isobaric compounds) will be displayed. Also shown in each table or subtable is the name of the compound, the molecular formula, the monoisotopic mass of the parent compound, the adduct type (selected by the user), the adduct mass, and the difference between the query mass and the matching mass (given as Delta, in parts per million or ppm). One of the challenges in working with only m/z data (or only LC-MS data) is that multiple candidates or multiple metabolites can match to a given m/z value. Usually additional information, such as retention time, expected abundance in the given biofluid, additional MS/MS spectra or selected information regarding biological context must be used to decide which unique compound(s) are most likely matching to the observed MS spectra.
14. Reading through the MetaboCard for D-2-hydroxyglutarate will reveal that it has been implicated as an oncometabolite as well as a key metabolite in a rare inborn of metabolism called D-2-hydroxyglutaricaciduria. Scrolling down the MetaboCard will show more details regarding D-2-hydroxyglutarate's structure, synonyms, chemical classifications, concentrations in different biofluids, pathways, and the enzymes/transporters that bind it or act on it.
15. While multiple “hits” are possible with the mass list provided in this example, the results of this particular LC-MS study on glioblastoma CSF should ideally show that D-2-hydroxyglutarate, fumarate, succinate, and lactate have increased in concentration in cancer patients, while glutamine, isocitrate, and glutathione have decreased in cancer patients. The metabolites that increased in concentration are known oncometabolites and appear to catalyze further mutations and epigenetic changes (D-2-hydroxyglutarate, fumarate) or contribute to immunosuppression, metastasis, inflammation, and other well-known hallmarks of cancer (succinate and lactate). This simple example was chosen for didactic purposes. Identifying compounds using only m/z data is often risky and leads to multiple mass-matching redundancies. If MS/MS data were available it would have been possible to search for similar MS/MS spectra via HMDB’s “MS/MS Search” option. Finding matching MS/MS spectra (either observed or predicted) to any query MS/MS spectrum provides important supporting evidence regarding the actual identity of the compound or compounds of interest.

Acknowledgments

This work was supported by Genome Alberta (a division of Genome Canada), The Canadian Institutes of Health Research (CIHR), Western Economic Diversification (WED), and the Canada Foundation for Innovation (CFI).

References

1. Wishart DS (2007) Proteomics and the human metabolome project. *Expert Rev Proteomics* 4:333–335
2. Mandal R, Guo AC, Chaudhary KK, Liu P, Yallou FS, Dong E et al (2012) Multi-platform characterization of the human cerebrospinal fluid metabolome: a comprehensive and quantitative update. *Genome Med* 4:38
3. Psychogios N, Hau DD, Peng J, Guo AC, Mandal R, Bouatra S et al (2011) The human serum metabolome. *PLoS One* 6:e16957
4. Bouatra S, Aziat F, Mandal R, Guo AC, Wilson MR, Knox C et al (2013) The human urine metabolome. *PLoS One* 8:e73076
5. Dame ZT, Aziat F, Mandal R, Krishnamurthy R, Bouatra S, Borzouie S et al (2015) The human saliva metabolome. *Metabolomics* 11:1864–1883
6. Karu N, Deng L, Slac M, Guo AC, Sajed T, Huynh H et al (2018) A review on human fecal metabolomics: methods, applications and the human fecal metabolome database. *Anal Chim Acta* 1030:1–24
7. Wishart DS, Tzur D, Knox C, Eisner R, Guo AC, Young N et al (2007) HMDB: the human Metabolome database. *Nucleic Acids Res* 35 (Database issue):D521–D526
8. Wishart DS, Feunang YD, Marcu A, Guo AC, Liang K, Vázquez-Fresno R et al (2018) HMDB 4.0: the human metabolome database for 2018. *Nucleic Acids Res* 46(D1):D608–D617
9. Djoumbou-Feunang Y, Fiamoncini J, Gil-della-Fuente A, Greiner R, Manach C, Wishart DS (2019) BioTransformer: a comprehensive computational tool for small molecule metabolism prediction and metabolite identification. *J Cheminform* 11:2
10. Djoumbou Feunang Y, Eisner R, Knox C, Chepelev L, Hastings J, Owen G et al (2016) ClassyFire: automated chemical classification with a comprehensive, computable taxonomy. *J Cheminform* 8:61
11. Weininger D (1988) SMILES, a chemical language and information system. I. Introduction to methodology and encoding rules. *J. Chem Inf Comput Sci* 28:31–36
12. Heller S, McNaught A, Stein S, Tchekhovskoi D, Pletnev I (2013) InChI—the worldwide chemical structure identifier standard. *J Chem* 5:7
13. Kim S, Thiessen PA, Bolton EE, Chen J, Fu G, Gindulyte A et al (2016) PubChem substance and compound databases. *Nucleic Acids Res* 44(Database issue):D1202–D1213
14. Sud M, Fahy E, Cotter D, Brown A, Dennis EA, Glass CK et al (2007) LMSD: LIPID MAPS structure database. *Nucleic Acids Res* 35(Database issue):D527–D532
15. Hastings J, Owen G, Dekker A, Ennis M, Kale N, Muthukrishnan V et al (2016) ChEBI in 2016: improved services and an expanding collection of metabolites. *Nucleic Acids Res* 44 (Database issue):D1214–D1219
16. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM et al (2000) Gene ontology: tool for the unification of biology. The gene ontology consortium. *Nat Genet* 25:25–29
17. Kanehisa M, Furumichi M, Tanabe M, Sato Y, Morishima K (2017) KEGG: new perspectives on genomes, pathways, diseases and drugs. *Nucleic Acids Res* 45(Database issue):D353–D361
18. Jewison T, Su Y, Disfany FM, Liang Y, Knox C, Maciejewski A et al (2014) SMPDB 2.0: big improvements to the small molecule pathway database. *Nucleic Acids Res* 42(Database issue):D478–D484
19. Amberger JS, Bocchini CA, Scott AF, Hamosh A (2019) OMIM.org: leveraging knowledge across phenotype-gene relationships. *Nucleic Acids Res* 47(Database issue):D1038–D1043



Chapter 11

De Novo Molecular Formula Annotation and Structure Elucidation Using SIRIUS 4

Marcus Ludwig, Markus Fleischauer, Kai Dührkop, Martin A. Hoffmann, and Sebastian Böcker

Abstract

SIRIUS 4 is the best-in-class computational tool for metabolite identification from high-resolution tandem mass spectrometry data. It offers de novo molecular formula annotation with outstanding accuracy. When searching fragmentation spectra in a structure database, it reaches over 70% correct identifications. A predicted fingerprint, which indicates the presence or absence of thousands of molecular properties, helps to deduce information about the compound of interest even if it is not contained in any structure database. Here, we present best practices and describe how to leverage the full potential of SIRIUS 4, how to incorporate it into your own workflow, and how it adds value to the analysis of mass spectrometry data beyond spectral library search.

Key words Metabolomics, LC–MS/MS, Annotation, Molecular formula, Structure prediction, SIRIUS, Metabolite identification

1 Introduction

Comprehensive identification of small molecules is one of the most urgent needs in metabolomics and related fields such as in natural products research, biomarker discovery, and environmental science. Yet, this task remains highly challenging. Liquid-chromatography tandem mass spectrometry (LC–MS/MS) is one of the most prominent analytical techniques to identify biomolecules. The mere mass of a compound is not sufficient to determine the correct molecular formula, let alone its structure. Tandem mass spectrometry provides additional information but is non-trivial to interpret. Usually, metabolite identification is performed by searching fragmentation spectra in a spectral library [15, 36, 41, 44]. However, spectral libraries are—and always will be—highly incomplete. This represents a major obstacle, particularly for secondary metabolism analysis. During the last years multiple tools were developed for searching in structure databases which are orders of magnitudes

larger compared to spectral libraries; this includes CFM-ID [1], DEREPLICATOR+ [25], MAGMa [30], MetFrag [32, 45], MIDAS [40], MS-FINDER [37], and CSI:FingerID [7].

Currently, the best performing tool for this task is CSI:FingerID, successor of FingerID [13]. It is part of SIRIUS 4 [9], a software for metabolite identification from high-resolution fragmentation spectra. SIRIUS started off as a method for de novo molecular formula identification, but now integrates CSI:FingerID to offer combined molecular formula annotation and structure database search. SIRIUS performs metabolite identification in a two-step approach: Firstly, the molecular formula of the query compound is determined via isotope pattern analysis and fragmentation trees. Second, SIRIUS uses CSI:FingerID to predict a molecular fingerprint from the given spectrum and fragmentation tree. This predicted fingerprint can be searched against a structure database to identify the most likely candidate. Searching CASMI 2016 [33] positive ion mode spectra in a database of 0.5 million structures of biological interest resulted in 74.0% correct identifications [9]. When searching in PubChem [20], which contains many millions of structures, CSI:FingerID still achieves an identification rate of 39.4% (74.8% in the top 10). These rates were reached without using meta-information such as citation frequencies or production volumes; using such meta-information can be very harmful in practice [2].

Whereas spectral library search will only allow a “peek through the keyhole,” SIRIUS enables untargeted identification to draw a more complete picture of a metabolic system [5]. It is understood that not every existing biomolecule is or will be contained in structure databases. But even for these instances SIRIUS offers valuable insight by providing a predicted molecular fingerprint to assist de novo structure elucidation and by searching in databases of hypothetical structures such as the in silico generated MINE databases [17]. Comprehensive compound identification is not a luxury but an indispensable step to answer biological questions. Compared to spectral library search SIRIUS offers highly increased coverage; compared to searching compounds only by mass it offers tremendously improved accuracy. Here, we present how to use SIRIUS to systematically annotate your compounds, and provide insight on common practices, judging the results and necessary prerequisites of your data.

2 What Data Can Be Processed by SIRIUS?

SIRIUS processes high-resolution, high mass accuracy fragmentation spectra, but also uses first stage of mass spectrometry (MS1) data. The statistical model of SIRIUS and the machine learning model of CSI:FingerID were trained on tandem mass spectra

(MS/MS) created by collision-induced dissociation (CID), as commonly applied in LC–MS/MS experiments. Most of the training compounds were ionized by electrospray ionization (ESI). However, it has been reported that SIRIUS is also able to analyze compounds from GC–MS data which has been acquired using the soft ionization method dopant-assisted atmospheric pressure chemical ionization (dAPCI) and subsequently fragmenting ions by CID [22]. At present, SIRIUS only handles single-charged compounds.

3 Preprocessing

SIRIUS is specialized in metabolite identification and relies on other tools for proper preprocessing. Input spectra must be in centroid mode (peak picked). Besides, further preprocessing of the data is highly beneficial for good results. Open source software exists for feature finding, to group isotope peaks of each compound, estimate adducts, and reject all MS/MS which cannot be assigned to a proper feature in the MS1. OpenMS [31] and MZmine 2 [27] both provide export functions tailored to the needs for SIRIUS.

It is beyond the scope of this chapter to go into the details of the different preprocessing steps, but *see Chapter 4* in this book for details on OpenMS processing. Unfortunately, we cannot propose optimal parameters, since these depend on the data. A metabolomics OpenMS workflow to preprocess data for SIRIUS may use the following OpenMS tools: FeatureFinderMetabo, MetaboliteAdductDecharger, and SiriusAdapter. The SiriusAdapter can be used either to directly run SIRIUS or to export .ms-files for SIRIUS to import.

SIRIUS benefits from the following preprocessing steps:

- A reasonably averaged MS1 is more accurate than using a single MS1 spectrum. Determining the masses and intensities of the compound’s isotope pattern using the chromatographic peaks can reduce errors.
- When measuring multiple MS/MS spectra of the same compound, in particular at different collision energies, it is beneficial to analyze a merged spectrum rather than the individual spectra. Fragmentation spectra can be grouped by their corresponding MS1 feature. SIRIUS will merge all grouped spectra. This is preferred over directly providing a merged spectrum as input for SIRIUS.
- MS/MS spectra which cannot be assigned to any MS1 feature should be rejected; these spectra are likely of bad quality.

- MS/MS spectra with low total intensity or very few signal peaks should be rejected. Usually it is difficult to confidently identify the corresponding compounds.

It is usually not necessary to preprocess fragmentation spectra by removing “noise peaks” or recalibrating masses; such preprocessing can substantially worsen results, as signal peaks may be removed or masses shifted into the wrong direction. SIRIUS can decide for itself which of the peaks in the spectrum are noise, but it cannot recover the masses of accidentally removed signal peaks. To this end, be cautious when using intensity thresholds. If the data is noisy and necessitates “noise peak” removal, use a low intensity threshold to remove as few signal peaks as possible. Furthermore, we propose to use a low MS1 intensity threshold and not-to-restrictive parameters for feature detection. A high number of spurious features might pose a problem for MS1-only analysis. But here, we concentrate on metabolite identification based on fragmentation spectra, and spurious features can easily be recognized because these will not produce significant signal peaks within the fragmentation spectrum. Using liberal parameters will help to detect more low intensity isotope peaks and include them into the compound’s isotope pattern.

Instrumental setup has huge impact on spectrum quality and some setups might be more suitable for structure elucidation with computational tools. *See Tip 1* for more information.

Tip 1: Spectra Quality

High quality spectra are indispensable to obtain good compound annotations. Spectra of high quality possess many signal peaks with intensities considerably above the noise level and mass errors of less than 10 ppm. On the other hand, few high-intensity signal peaks and mass errors of over 15 ppm indicate a spectrum of bad quality. It is understood that some molecules produce few fragments. But the information content of a spectrum increases with the number of (non-noise) peaks; identifying a compound from one peak is mere guessing. A proper instrumental setup can facilitate peak-rich spectra. Instead of using a single collision energy, spectra should be measured at multiple energies and merged. Alternatively, a ramped collision energy can be used to cover a large range of energies. In both cases, we expect to see more fragmentation peaks and, hence, better results.

Broad isolation windows favor chimeric spectra, being composed of fragments from more than one compound. Such chimeric spectra will interfere with fragmentation tree computation and also complicate the identification of structures via CSI:FingerID. In addition, broad isolation windows will result in isotope patterns

for all fragments. Selecting only the monoisotopic peak for fragmentation makes it easier to interpret the fragmentation spectrum. SIRIUS provides an option to account for isotopes in the fragmentation spectrum, but this assumes that the isolation window is broad and isotope patterns of fragments are undisturbed. Unfortunately, filtering is imperfect in practice: An isolation window of width, say, 3 Da may select 100% of the monoisotopic peak, 80% or the first and 50% of the second isotope peak. This will distort the isotope patterns of fragments in a non-trivial way. At present, SIRIUS cannot deal with distorted fragment isotopes patterns.

Compound identification benefits from choosing an instrumental setup which minimizes chimeric spectra, and favors peak-rich and low noise fragmentation spectra.

4 Metabolite Identification

SIRIUS identifies metabolites in two steps: namely molecular formula annotation and searching in a structure database. Both steps can be performed on a complete dataset using a single command; but users are advised to manually validate all results, including intermediate results. Here, we will explain the usage of SIRIUS step-by-step. For the sake of a more vivid description we will refer to the graphical user interface (GUI) of SIRIUS. All computations can be performed via the command-line interface (CLI), using the GUI as a mere visualization tool for final results (*see* Subheading 5).

An overview of the SIRIUS GUI is displayed in Fig. 1. The analysis starts with importing the data; this is done via the import dialog or drag-and-drop. SIRIUS imports spectra from .csv, .ms, or .mgf files. Imported compounds are displayed in the compound list located in the left panel. To find specific compounds, use the search field above the panel. Start computations by clicking the *Compute All* button or by selecting a set of compounds and using the context menu (right-click). If only a single compound is selected, additional parameters can be specified such as the known molecular formula.

4.1 Molecular Formula Annotation

SIRIUS finds the most likely molecular formula by considering all possible molecular formulas, and is able to annotate biomolecules with a molecular formula missing from any database. Necessary parameters for SIRIUS are:

Elements	Set of considered elements. Some elements can be auto-detected if an isotope pattern is given (<i>see</i> Tip 2).
ppm	Allowed mass deviation in ppm. This is the maximum value a molecular formula explanation is allowed to deviate from the peaks' measured mass. Molecular formulas with a



Fig. 1 The *SIRIUS* Overview tab displays the spectrum and fragmentation trees of the top molecular formula candidates. The best candidate C₂₄H₃₈O₃ is selected; the corresponding explained spectrum and fragmentation tree are shown. The left panel contains a searchable list of all compounds; selected compounds are highlighted. The data and results of the first selected compound are displayed in all the views to the right of the compound list. The upper panel provides functionalities to import spectra, save and load workspaces, export result tables, start computations, and display their status in the jobs panel. The *SIRIUS* overview tab displays various scores for each molecular formula candidate and can be sorted accordingly

higher mass error are ignored. Note that for all peaks below 200 Da an absolute error is assumed which corresponds to the specified deviation in ppm at 200 Da.

Set of considered ion types. For details see Tip 4.

Number of candidates to be displayed. Fragmentation trees are computed for all molecular formula candidates using the Critical Path³ heuristic from [8]. The top k fragmentation trees are recomputed using an exact algorithm; here, k corresponds to the number of displayed candidates plus 10. Hence, a larger number of displayed candidates increases running times.

Considered ion types

Candidates

Depending on the dataset, anticipated elements and ion types can be selected. Select a reasonable set of elements. The mass deviation is the maximum allowed deviation. Spectra measured on

an instrument with advertised sub-ppm mass accuracy might still have much larger mass deviation (e.g., if not properly calibrated or because of bad peak picking). More restrictive parameters, in particular for the allowed elements, can make computations substantially faster. Never select all uncommon elements at once. This will lead to a combinatorial explosion of potential molecular formulas; running times will increase dramatically; the number of correct molecular formula annotations will decrease. SIRIUS provides scoring profiles for Q-TOF and Orbitrap, which mainly change some background parameters. In case you are unsure if your data really has the instrument's advertised accuracy, use the default profile and set your allowed mass deviation accordingly.

Fragmentation trees are computed from a merged spectrum combining all input fragmentation spectra. Isotope pattern analysis is performed on a merged MS1 spectrum or using the isotope pattern provided by a preprocessing tool. A fragmentation spectrum which possesses peaks broadly distributed across the whole mass range presents more information to SIRIUS than a spectrum composed of either low or high mass peaks only.

4.1.1 Judging Results

Molecular formula annotation results are displayed in the *Sirius Overview* tab (see Fig. 1). Candidates are ranked by the sum of isotope pattern and fragmentation tree score (see Tip 2 on isotopes and Tip 3 on fragmentation trees). Colored bars for each score ease comparison between candidates. Each candidate molecular formula has an adduct. At this stage, this is an ion type; after structure database search with CSI:FingerID this adduct corresponds to an adduct type (compare Figs. 1 and 3 and see Tip 4).

The displayed attributes are:

Score	Overall score by which candidates are ranked. This is the sum of isotope and tree score.
Isotope score	Similarity score comparing the measured isotope pattern with the theoretical pattern for each candidate molecular formula. Usually, a score close to zero or low in comparison to the remaining candidates indicates an incorrect molecular formula, or at least an annotation of low confidence. Besides being the incorrect candidate, this might indicate improper data quality such as high intensity deviation or a low number of detected isotope peaks. The scored isotope

Tree score	pattern is highlighted in the <i>merged MSI</i> and can be assessed via the <i>Spectrum view</i> tab.
Explained peaks	Score of the computed fragmentation tree.
Total explained intensity	The number of peaks in the spectrum which can be explained by the fragmentation tree. A high number of unexplained peaks indicates an incorrect annotation, a noisy spectrum, or two compounds being fragmented simultaneously.
Median absolute mass deviation	Summed relative intensity of all explainable peaks. Values of 95% or higher indicate good quality; for values below 80%, results should be interpreted with care.
	The median absolute mass deviation of explained peaks in ppm. Low deviations are clearly desirable.

Selecting a molecular formula candidate displays the corresponding fragmentation tree and spectrum in which explained peaks are highlighted. The merged MS1 spectrum displays the selected isotope pattern. Mass errors of each fragment are shown to spot unlikely explanations; the displayed fragmentation tree can be colored accordingly. The user can inspect fragmentation tree annotations in varying degree of detail; individual fragments may support or contradict a particular molecular formula candidate. The user may decide by manual validation how well a candidate is supported.

Tip 2: Isotope Pattern and Element Detection

CAUTION: If no isotope pattern is provided and compounds are expected to contain elements beside CHNOPS, we strongly recommend to restrict molecular formulas to those from a molecular structure database. Do not select all uncommon elements for molecular formula annotation with SIRIUS. This will lead to a combinatorial explosion of potential molecular formulas; running times will increase dramatically.

Isotope patterns offer valuable information about elemental composition. The presence of uncommon elements that result in characteristic isotope pattern changes can be automatically detected [24]. Detectable elements are sulfur, chlorine, bromine, boron, and selenium. When detected, SIRIUS adds these elements to the

default set of elements CHNOP to determine the molecular formula. A predictor for silicon is disabled by default, as it results in a relatively large number of false positive predictions; the silicon isotope pattern is not “special” enough to permit a reliable auto-detection. In contrast to [24], the current version of SIRIUS uses a deep neural network for auto-detection of elements. Automated detection can be enabled or disabled via the compute dialog. Not considering elements which are extremely unlikely, substantially improves running times and may slightly improve results [24]. SIRIUS may still choose a molecular formula which *does not contain* an element with positive auto-detection, just as it might choose a molecular formula which does not contain any other enabled element. The final score of each molecular formula candidate is a combination of the fragmentation tree score and the isotope pattern score.

Tip 3: Fragmentation Trees

A fragmentation tree annotates peaks in the fragmentation spectrum with molecular formulas and identifies likely losses between the fragments—similar to “fragmentation diagrams” created by experts. The calculated tree must not be understood as ground truth but can be used to derive information about the measured compound’s fragmentation [29]. Fragmentation trees are also used to identify the molecular formula of an unknown compound. For every molecular formula candidate of the precursor ion, a separate fragmentation tree is computed which best explains the spectrum, as evaluated by a Maximum A Posteriori estimator [3]. This estimation takes into account information such as mass deviations, intensities, common losses, and loss sizes. The overall best-scoring fragmentation tree corresponds to the most likely molecular formula explanation. In addition, CSI:FingerID uses the fragmentation tree to predict the compound’s molecular fingerprint.

A simplified example of a fragmentation tree is presented in Fig. 2. A fragmentation tree is computed from the fragmentation spectrum given the (candidate) molecular formula of the precursor ion. Initially, a fragmentation graph is constructed in the following way: For every fragment peak, all possible molecular formula explanations are computed. These explanations must be subformulas of the precursor molecular formula—a fragment only loses, but never gains new atoms. Every such molecular formula is a node in the graph. Nodes are connected by an edge if one node is a subformula of another node—this represents a potential loss. Using combinatorial optimization, the best-scoring fragmentation tree is computed which explains every peak at most once. Unexplained peaks are considered noise.

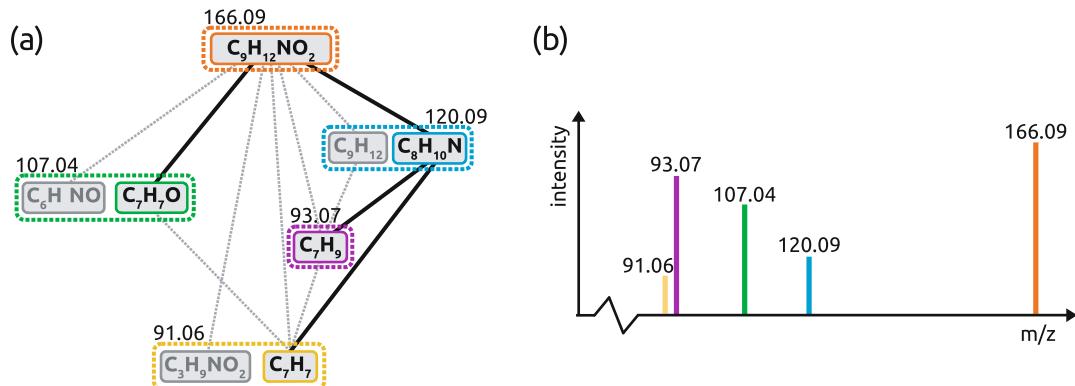


Fig. 2 Example of a fragmentation tree computed from a fragmentation graph in (a), given the spectrum in (b). The molecular formula of the neutral precursor is assumed to be $\text{C}_9\text{H}_{12}\text{NO}_2$. Molecular formulas are computed for all fragment peaks and serve as the nodes of the graph; nodes with the same color indicate molecular formulas corresponding to the same peaks. Nodes are connected by edges if one node is a subformula of another, thereby creating the fragmentation graph. A fragmentation tree is a connected subgraph which explains each color (peak) at most once and has no cycles. The best-scoring fragmentation tree, corresponding to a Maximum A Posteriori estimator, is computed by combinatorial optimization. The optimal fragmentation tree is indicated by solid lines; nodes which are not used are grayed out. These computations are repeated for each molecular formula candidate explaining the precursor mass, and the best such fragmentation tree is reported

Tip 4: Ion and Adduct Types

SIRIUS differentiates between ion types and adduct types. Default ion types for positive ion mode spectra are protonation, sodium, and potassium; default ion types for negative ion mode spectra are deprotonation and chlorine. Adduct types can be seen as sub-types of an ion type. For example, the ion type protonation includes adduct types “intrinsically charged” ($[\text{M}]^+$), “protonated” ($[\text{M} + \text{H}]^+$), “protonated with water loss” ($[\text{M} - \text{H}_2\text{O} + \text{H}]^+$), and “ammonium group” ($[\text{M} + \text{NH}_4]^+$).

Adduct types cannot be determined from the fragmentation spectrum—the fragments $[\text{C}_4\text{H}_6\text{O}_2 + \text{NH}_4]^+$ and $[\text{C}_4\text{H}_9\text{NO}_2 + \text{H}]^+$ result in the exact same peak; and so will $[\text{C}_5\text{H}_7]^+$ and $[\text{C}_5\text{H}_8\text{O} - \text{H}_2\text{O} + \text{H}]^+$. That is why SIRIUS considers ion types, not adduct types, during the molecular formula annotation step. Multiple adduct types of the determined ion type can be considered for structure database search with CSI:FingerID (see Figs. 3 and 4). When a specific ion type plus adduct type is provided by the user, it will be used during all computation steps. Users can specify additional ion and adduct types within the GUI or by modifying the config file.



Fig. 3 Additional candidates are added to the *SIRIUS Overview* tab after searching with CSI:FingerID in a structure database considering adduct types $[\text{M} + \text{H}]^+$, $[\text{M} + \text{NH}_4]^+$ and $[\text{M} - \text{H}_2\text{O} + \text{H}]^+$. Molecular formulas $\text{C}_{24}\text{H}_{40}\text{O}_4$ and $\text{C}_{24}\text{H}_{38}\text{O}_3$ differ by an in-source loss of H_2O and are not distinguishable by MS/MS since in both cases, the ion $[\text{C}_{24}\text{H}_{38}\text{O}_3 + \text{H}]^+$ is fragmented; hence, both have identical score. (The same holds for the pairs $\text{C}_{22}\text{H}_{33}\text{N}_2\text{O}_2$ vs. $\text{C}_{22}\text{H}_{36}\text{N}_3\text{O}_2$ and $\text{C}_{18}\text{H}_{39}\text{N}_4\text{O}_2\text{P}$ vs. $\text{C}_{18}\text{H}_{36}\text{N}_3\text{O}_2\text{P}$). Displayed is the resolved fragmentation tree for $[\text{C}_{24}\text{H}_{40}\text{O}_4 - \text{H}_2\text{O} + \text{H}]^+$, where an H_2O loss has been added to its top

Tip 5: Molecular Fingerprint

A molecular fingerprint is a binary vector of fixed length where each position corresponds to a specific molecular property; for example, position # 393 may encode the presence or absence of a benzene ring as a substructure. In general, a “1” indicates this specific substructure is present in the molecule, a “0” indicates it is not. There exist several types of fingerprints, such as PubChem CACTVS fingerprints,¹ Klekota–Roth fingerprints [21], and MACCS fingerprints. Given a molecular structure, the corresponding fingerprint can be deterministically computed. Unfortunately, different structures can have the same molecular fingerprint.

Molecular fingerprints can be used to perform similarity search in a structure database. A common way to compare molecular

¹ http://ftp.ncbi.nlm.nih.gov/pubchem/specifications/pubchem_fingerprints.pdf

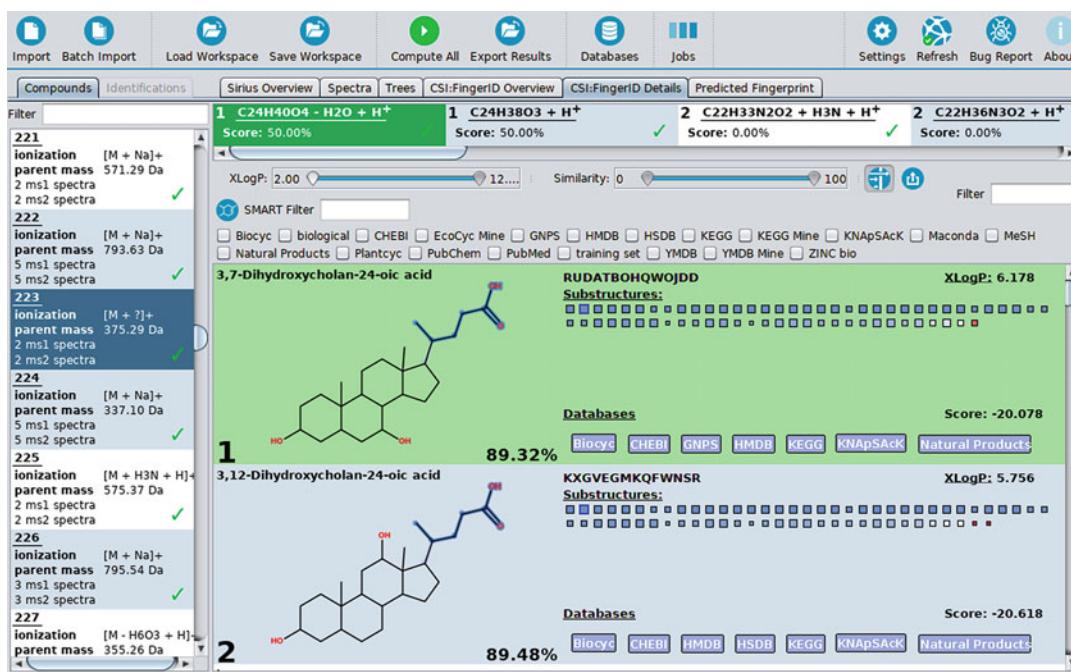


Fig. 4 The *CSI:FingerID Details* tab displays structure candidates for a selected molecular formula. The highlighted molecular property, which is predicted to be present in the query, is contained in the top 2 hits. Candidates are sorted by their score which is displayed on the right-hand side. Numbers in percent indicate the Tanimoto similarity between the predicted fingerprint and the fingerprint of each candidate. Candidates can be filtered by database, SMARTS string and XlogP value

structures using fingerprints is the Tanimoto similarity, also known as Jaccard index. Identical fingerprints produce a similarity of 1, whereas two structures not sharing a single molecular property have a Tanimoto of 0. Clearly, the similarity value depends on the choice of fingerprint type.

CSI:FingerID predicts a variety of molecular properties from several fingerprint types; only those molecular properties were selected which could also be predicted in evaluations. Given a spectrum and corresponding fragmentation tree, CSI:FingerID predicts a probabilistic fingerprint, *see* Subheading 4.3. This predicted fingerprint is compared to the deterministic fingerprints from a structure database to find the best match. The *CSI:FingerID Overview* tab also displays, for every structure candidate, the Tanimoto similarity against the predicted fingerprint. However, CSI:FingerID uses a different scoring function to rank candidates, which results in a larger number of correct identifications [7, 23].

4.2 Searching in Structure Databases

After the molecular formula has been identified, the compound is searched in a structure database. Firstly, a molecular fingerprint of the query (*see* Tip 5) is predicted from the spectrum and fragmentation tree. Next, this predicted fingerprint is compared to (and

scored against) fingerprints of structures in a database, to find the best matching structure. It must be understood that the molecular fingerprints of the candidate structures are fixed, known and independent of our tools.

To predict the molecular fingerprint, we have to know the molecular formula, ion type, and adduct type of the query. By default, not only the top-scoring molecular formula but multiple high-scoring molecular formula candidates are considered, applying a soft score threshold: All molecular formula candidates with a score above 0.75 of the optimal score are considered. To this end, we iterate over all possible combinations of molecular formula candidate and adduct type. The ion type of the query is determined by the molecular formula candidate; but various adduct types can be specified to search the database, *see* Tip 4 on ion and adduct types. When searching in the database, candidate structures must match the estimated molecular formula of the neutral molecule. Fragmentation trees of different adduct types differ as, say, a neutral loss is added to the top of the tree. These trees have exactly the same score. For each molecular formula and adduct type with candidate structures in the database, the resolved fragmentation tree is displayed in the *SIRIUS Overview* tab, *see* Fig. 3.

Scored structure candidates are displayed in the *CSI:FingerID Overview* tab. The *CSI:FingerID Details* tab allows to examine the scored structures in more detail for each molecular formula and adduct type separately (*see* Fig. 4).

As a default, users should search compounds in the PubChem database, and filter results to the biocompound structure database or a subset thereof (*see* Tip 6). You may accept those query identifications for which there is a high-scoring structure candidate in the restricted database; potentially, this is even the highest-scoring candidate for all of PubChem. For those cases where no reasonable candidate was found in the biocompound structure database, and for cases where the best PubChem candidate scores substantially better than the best biocompound candidate, you can extend your search space to all of PubChem. Obviously, it makes much sense to *integrate biochemical background knowledge* at this point: This may be information about the organism the sample was taken from, or information about the biochemical preparation of the sample. Such meta-information is not integrated into SIRIUS and CSI:FingerID, as this integration is highly non-trivial; but it is straightforward how to integrate the information manually.

4.2.1 Judging Results

Users should check if the best structure candidate agrees with the best molecular formula candidate. Sometimes, CSI:FingerID decides that, based on its machine learning model and the given candidate structures, a structure with a different molecular formula better agrees with the data. Users should verify if the selected structure database does not contain any structures for the best-

scoring molecular formula candidate; this can be an indication that the selected database is too restrictive. Besides, check if the correct adduct type has not been selected for database search.

CSI:FingerID ranks structure candidates by a logarithmic posterior probability [23], so that scores are negative numbers and zero is the optimum. Additionally, the predicted Tanimoto similarity is displayed. Since this is based on the predicted probabilistic fingerprint, this similarity usually underestimates the Tanimoto similarity between the true fingerprints. Candidates can be filtered by database, XlogP values [38, 39] predicted using the Chemistry Development Kit [35, 43], or a specific SMARTS string. Structures are linked to database entries; clicking on the database icon opens the appropriate website. One CSI:FingerID candidate structure may link to several “3D structures” in a database, as CSI:FingerID ignores stereochemistry in its computations. The number of PubMed citations² is also displayed in the *CSI:FingerID Overview* tab. This value can contribute valuable information for the identification, for example, as a sanity check. But on startup, these values must not be used to filter results: Doing so, we ignore the actual experimental data and potentially make our decisions based solely on prior knowledge [2].

The example in Fig. 4 shows two top-scoring structure candidates. Both are structurally very similar and consequently, also have similar scores. The user may decide which structure is more likely, based on background knowledge about the sample. Comparing the, say, top 5 hits may also help to get an idea about a “core” structure which CSI:FingerID predicts to be present. Blue and red squares next to each candidate molecular structure represent its molecular properties. Blue properties are predicted to be present by CSI:FingerID and also present in the candidate; red properties are predicted to be absent but are present in the candidate. The size of the square represents the quality (F1 score, harmonic mean of precision and recall) of the predictor, as determined beforehand in cross validation; but a large F1 score does not guarantee that the prediction is correct for *this* query. In contrast, the saturation of the color indicates how sure CSI:FingerID is about the property, for this query. One specific property—a carboxyl group attached to a carbon chain—has been highlighted in Fig. 4; it is present in the predicted fingerprint and in the first two candidates. A score close to zero and many blue squares usually indicates a confident identification—in this example, CSI:FingerID is very certain that the correct structure is at least very similar to the top hit. Even in case the best structure candidate is not correct, it is often structurally similar to the correct one and can help to elucidate the structure or answer the underlying biological question. Be warned that CSI:

² <https://www.ncbi.nlm.nih.gov/pubmed>

FingerID scores between different query compounds are usually not comparable; be cautious when using this score to differentiate between true and bogus identifications.

As explained in Subheading 4.1.1, users can also examine the fragmentation tree to decide how well a candidate is supported: For example, are specific side chains supported by fragments, losses or even fragmentation cascade in the fragmentation tree?

Tip 6: Some Notes on Database Size

CSI:FingerID correctly identifies 39.4% of CASMI 2016 positive ion mode spectra when searching in PubChem (in a structure-disjoint cross-validation setup). Searching in PubChem is difficult because it contains many millions of structures. If the search is performed in a database with 0.5 million structures of biological interest, correct identifications increase to 74.0% [9]. To further increase identification rates, we might even be more restrictive and search in HMDB [44] or ChEBI [12]. Limiting CSI:FingerID search to the same structures which are contained in spectral libraries will even result in identification rates comparable to spectral library search! Does this mean it is advisable to search in a database with as few structures as possible? Clearly not! Results will look great in evaluation as long as all reference structures are contained in the restricted database. But in application, many compounds will be absent from the database, meaning you cannot find them at all.

Furthermore, there are—often ignored—side effects of searching in small databases. Firstly, the measured data becomes less important. You can easily identify a compound from one peak if you limit the candidate list to a few structures. Unfortunately, doing so does not increase the identification’s confidence. It merely means that one candidate better matches the data compared with the other candidates, always assuming the correct structure is present in the candidate list. Second, incorrect identifications can be hard to spot, because they still “make sense”: If all candidates in our database are frequently cited structures, then any identification (including the incorrect ones) will be a frequently cited structure and, hence, “reasonable.”

Clearly, there is a trade-off between small and large databases. In a small database, many relevant biomolecules are missing. On the other hand, searching in PubChem decreases the number of correct identifications even though many PubChem structures are very unlikely to be actual biomolecules. CSI:FingerID provides a bio-compound database with 0.5 million structures of biological interest, containing structures from ChEBI [12], KNApSAcK [34], HMDB [44], KEGG [18], HSDB [10], MaConDa [42], BioCyc

[4], UNDP [11], a biological subset of ZINC [16], GNPS [41], MassBank [15], and MeSH-annotated PubChem compounds [20, 26]. In application, it is reasonable to search in this biocompound database, which is much smaller than PubChem, but still much more diverse than spectral libraries. For those queries where we find no reasonable explanation in the biocompound database, we can then consider the PubChem candidates.

4.3 Beyond Structure Database Search

It is understood that certain query biomolecules are not contained in any structure database. But even for such difficult instances, SIRIUS and CSI:FingerID can assist in structural elucidation. Recall that the SIRIUS molecular formula annotation step (Sub-heading 4.1) is done *de novo*. Hence, molecular formulas can be determined even for “novel compounds” absent from any structure database. Even if a structure is not contained in the structure databases, CSI:FingerID may find a very similar structure. Furthermore, CSI:FingerID allows the user to search in custom databases which may contain hypothetical structures, to identify “novel compounds.”

But one key feature sets CSI:FingerID apart from other computational tools for structure elucidation: Predicting the molecular fingerprint of the query compound does not require any molecular structure database! The fingerprint is predicted from fragmentation spectrum and tree, and contains information about thousands of molecular properties. From that, we may draw conclusions what kind of substructures the query compound contains; and this information may be sufficient to decide if it is worth to further investigate the examined compound.

4.3.1 Judging Results

The predicted fingerprint is displayed in the *Predicted Fingerprint* tab, *see* Fig. 5. Most molecular properties are described by SMARTS (SMiles ARbitrary Target Specification) strings.³ SMARTS allows a flexible encoding of substructures; for example, a property might be described as “a methyl group bound to a hetero atom.” Since SMARTS strings are usually hard to visualize, SIRIUS displays a set of example structures from the training data that have a particular molecular property.

A posterior probability is predicted for every molecular property. Estimates close to 1 indicate the property is likely being present in the query compound, whereas estimates close to 0 indicate it is not. But be careful: Since CSI:FingerID predicts thousands of properties, even some “rather certain predictions” must be wrong. A 98% chance of being present also corresponds to 2% chance of being absent; if 1000 molecular properties are predicted at this level of certainty, then 20 predictions are wrong. Also be

³ <http://www.daylight.com/dayhtml/doc/theory/theory.smarts.html>

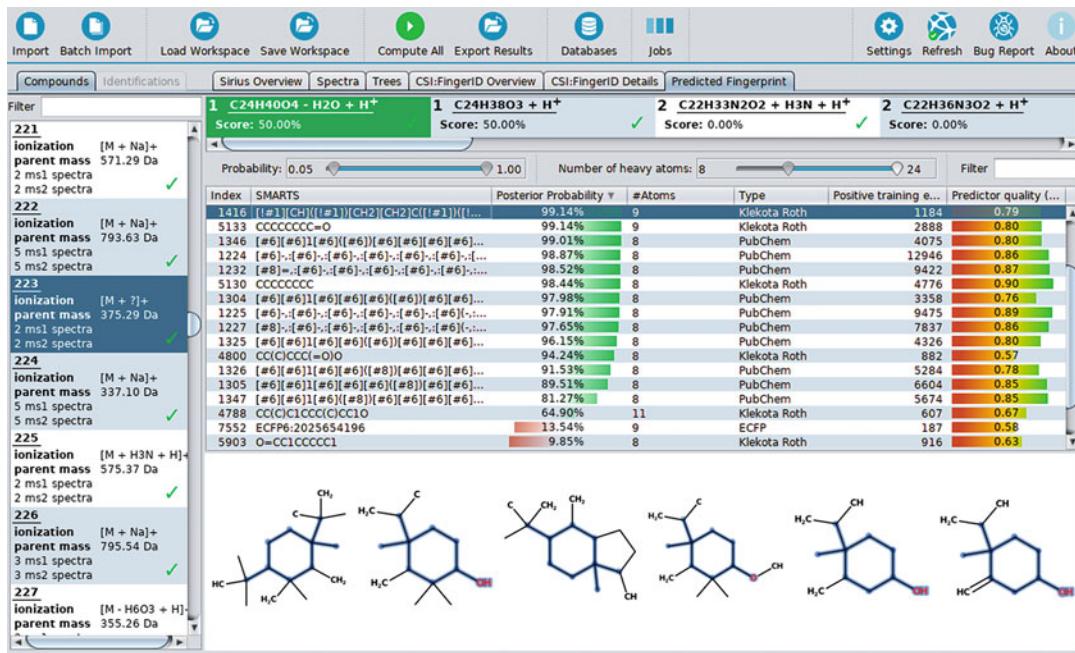


Fig. 5 The *Predicted Fingerprint* tab displays a predicted molecular fingerprint for each molecular formula candidate. The molecular fingerprint is predicted independently from any database. It can help deducing structural information on the compound even if the compound is not present in any structure database. Highlighted is a property mainly consisting of a ring. Training examples are displayed at the bottom. As shown, the oxygen is not a mandatory part of the substructure. The posterior probability of each property is also visualized as a color bar, to allow the user to swiftly distinguish properties predicted being present and absent. Green bars going to the right encode presence, red bars going to the left encode absence

reminded that these probabilities are *estimates*. To provide additional information on the quality of a prediction, the F1 score—a measure of the predictor quality—is displayed. The F1 score is the harmonic mean of precision (fraction of correct yes-predictions among all yes-predictions) and recall (fraction of correct yes-predictions among all yes-instances). A high F1 score indicates a good predictor, and 1.0 is the optimum. There is no general rule on what is a “good” F1 score; as a rule of thumb for this decision, one may assume that the F1 score equals precision and recall. Since many properties are rare and only present in few structures, the number of positive training examples is another indicator for the generalizability of the predictor. To help the user to concentrate on the most promising predictions, properties can be sorted by posterior probability, F1 score, or the number of atoms. The last option is useful to consider only larger, presumably more informative substructures.

5 Using SIRIUS in Automated Workflows

SIRIUS offers a powerful command-line interface (CLI) which allows for a flexible integration of SIRIUS into automated workflows. Technically speaking, the SIRIUS GUI is a visualization of the CLI functionality. Therefore, every task that can be done via the graphical user interface, can also be executed using the CLI. Corresponding to the two step approach in the GUI, the CLI provides self-contained sub tools for molecular formula identification and structure elucidation with separate parameter sets.

Furthermore, CLI and GUI share the same input and output formats. Both, CLI and GUI store the computed results in the SIRIUS project-space (see Fig. 6) which in turn can also be an input for the GUI or the CLI. This allows the user to review results in the GUI that have been computed with an automated workflow using the CLI.

5.1 The SIRIUS Project-Space

The SIRIUS project-space is a standardized directory structure that is organized in a three hierarchy levels, namely the *project level*, the *compound level*, and the *method level* (see Fig. 6 for details).

On the *project level*, each compound corresponds to one sub-directory (*compound level*) storing the input data, parameters, and results of the different analysis methods. These data is continuously written to the project-space, so that it represents the actual progress of a SIRIUS analysis. Further, the `.progress` file gives an overview about the progress of the ongoing analysis. On the *compound level*, each method provided by SIRIUS stores its results in its own sub-directory (*method level*). This allows the user to redo one analysis step without having to recompute the intermediate results it depends on. Further, SIRIUS is able to transfer intermediate results to a new project-space, so different parameters can easily be evaluated without having to recompute intermediate results. Since a project-space can be imported into the GUI, the user is able to judge intermediate results using the GUI before executing further analysis steps. Project-spaces can be read and written as an uncompressed directory or a compressed zip archive when using the `.sirius` file extension.

In addition to the *method level* results, the project-space contains summaries of these results on the *project level* and the *compound level*. These summaries are in `csv` format to provide easy access to the results for further downstream analysis, data sharing, and data visualization. The summaries are not imported into SIRIUS but are (re-)created based on the actual results every time a project-space is exported.

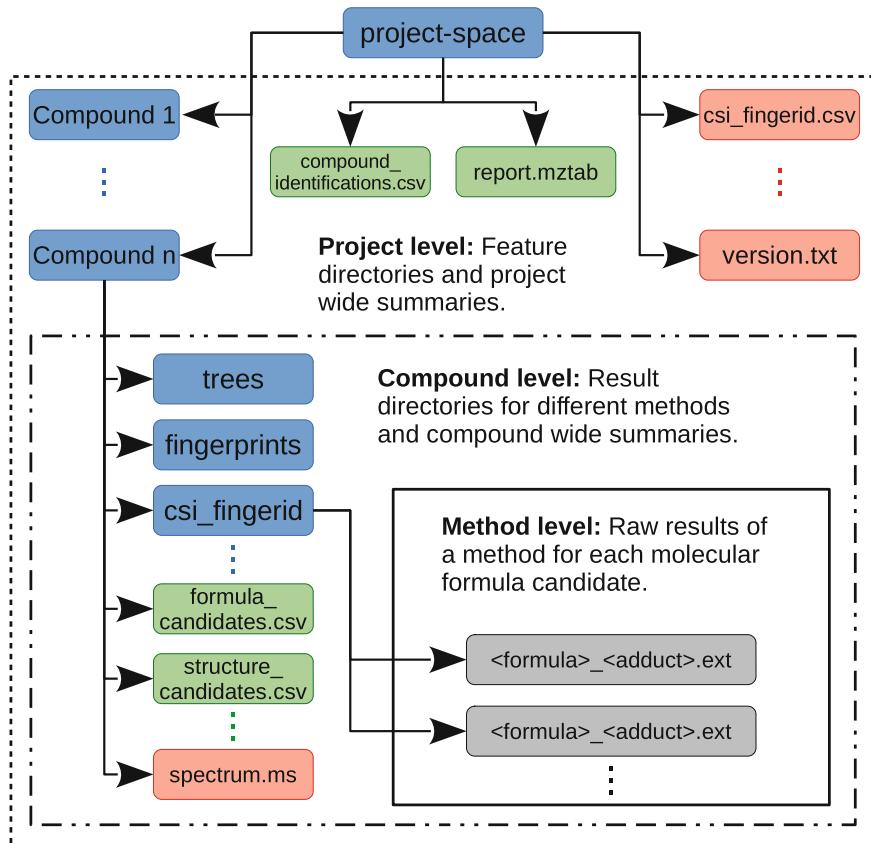


Fig. 6 The SIRIUS project-space is a standardized directory structure that stores results, summarized results, input data, parameters, and version information of a SIRIUS analysis. It is organized on three levels, namely the *project level* (dashed line), the *compound level* (dashed-and-dotted line), and the *method level* (solid line). The *compound level* contains sub-directories (blue) for each compound, summaries (green) about the whole dataset, and additional information (red) about the version of SIRIUS that created the output. The *compound level* contains a sub-directory for each method that was applied to the compound as well as the summaries of these methods results. Further, it contains additional information, such as the input data and the parameters used for the computations. On the *method level*, SIRIUS stores the results of a specific method for a given compound (grey)

5.2 Standardized Project-Space Summary with mzTab-M

The project-space is a SIRIUS-specific format that allows the user to access all results and analysis details, but may not be optimal for sharing this data with third party tools or data archives. For this purpose, SIRIUS provides an analysis report (`report.mztab`) in the standardized mzTab-M format [14]. All results summarized in this report are linked to the results in the corresponding SIRIUS project-space, allowing the user to share summarized results using mzTab-M without losing the connection to the detailed results provided in the project-space. Furthermore, SIRIUS passes meta-information such as scan numbers and identifiers of the input data into this analysis report. This allows for an easy combination of the SIRIUS results with the results of other analyses such as MS1-based quantification.

6 Custom Databases

Users may define their own structure databases to search in. These “custom databases” can be created via GUI and CLI. In the GUI, the *Databases* button opens a dialogue listing existing databases. New ones can be created with one click. Structures are imported by inserting structure descriptors (InChI or SMILES) into the import field; one structure per line. Custom databases are useful in case the users has a limited set of structures of interest. When screening for pollutants or drugs, a list of suspected structures can be collected in advance.

When searching with CSI:FingerID it does not matter if the structures in the database are known biomolecules or if these are hypothetical structures, which have not yet been discovered in any organism. Clearly, it is not reasonable to search in an arbitrarily large database. Databases of hypothetical structures have to be compiled with care to avoid combinatorial explosion. Available tools are BioTransformer [6] and the *in silico* generated MINE databases [17]. Currently, there exist MINE extensions for Ecocyc [19], YMDB [28], and KEGG [18]. But in principle, any existing structure database can be extended by such methods. Say, you are interested in finding new bile acids. A database of hypothetical bile acids can be created by applying biotransformations to known bile acids. This new database can then be searched with CSI:FingerID to find new bile acids synthesized by the investigated organisms.

7 Conclusion

To leverage the full potential of metabolomics, we need to overcome the limitations of spectral library search. This chapter presented concepts behind SIRIUS and CSI:FingerID, best-in-class computational tools for metabolite identification from high-resolution tandem mass spectra. We stress that computational tools currently cannot replace experts, but are meant to assist them. As a consequence, users must not accept identifications blindly but verify them properly. Here, we gave some advice on how this can be done.

SIRIUS ships with a command line tool which makes it easy to run computations on compute clusters and properly integrate it into automated workflows. Popular mass spectrometry data processing tools can create input files for SIRIUS, and SIRIUS outputs results in the standardized mzTab-M format to facilitate integration. The metabolomics community benefits from new computational tools, but tool development also benefits from the communities’ input and more public training spectra. Finally, method development is an ongoing process, and SIRIUS is evolving to further improve metabolite identification.

References

1. Allen F, Greiner R, Wishart D (2015) Competitive fragmentation modeling of ESI-MS/MS spectra for putative metabolite identification. *Metabolomics* 11(1):98–110. <https://doi.org/10.1007/s11306-014-0676-4>
2. Böcker S (2017) Searching molecular structure databases using tandem MS data: are we there yet? *Curr Opin Chem Biol* 36:1–6. <https://doi.org/10.1016/j.cbpa.2016.12.010>. <https://authors.elsevier.com/a/1UF-u4sz6LvFfy>
3. Böcker S, Dührkop K (2016) Fragmentation trees reloaded. *J Cheminform* 8:5. <https://doi.org/10.1186/s13321-016-0116-8>. <http://www.jcheminf.com/content/8/1/5>
4. Caspi R, Altman T, Billington R, Dreher K, Foerster H, Fulcher CA, Holland TA, Keseler IM, Kothari A, Kubo A, Krummenacker M, Latendresse M, Mueller LA, Ong Q, Paley S, Subhraveti P, Weaver DS, Weerasinghe D, Zhang P, Karp PD (2014) The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. *Nucleic Acids Res* 42(D1):D459–D471. <https://doi.org/10.1093/nar/gkt1103>. <http://nar.oxfordjournals.org/content/42/D1/D459.abstract>
5. da Silva RR, Dorrestein PC, Quinn RA (2015) Illuminating the dark matter in metabolomics. *Proc Natl Acad Sci U S A* 112(41):12549–12550. <https://doi.org/10.1073/pnas.1516878112>
6. Djoumbou-Feunang Y, Fiamoncini J, Gil-de-la Fuente A, Greiner R, Manach C, Wishart DS (2019) BioTransformer: a comprehensive computational tool for small molecule metabolism prediction and metabolite identification. *J Cheminf* 11(1):2
7. Dührkop K, Shen H, Meusel M, Rousu J, Böcker S (2015) Searching molecular structure databases with tandem mass spectra using CSI: FingerID. *Proc Natl Acad Sci U S A* 112(41):12580–12585. <https://doi.org/10.1073/pnas.1509788112>
8. Dührkop K, Lataretu MA, White WTJ, Böcker S (2018) Heuristic algorithms for the maximum colorful subtree problem. In: Proceedings of workshop on algorithms in bioinformatics (WABI 2018). Leibniz international proceedings in informatics (LIPIcs), vol 113. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, pp 23:1–23:14. <https://doi.org/10.4230/LIPIcs.WABI.2018.23>. <http://drops.dagstuhl.de/opus/volltexte/2018/9325>
9. Dührkop K, Fleischauer M, Ludwig M, Aksenov AA, Melnik AV, Meusel M, Dorrestein PC, Rousu J, Böcker S (2019) Sirius 4: a rapid tool for turning tandem mass spectra into metabolite structure information. *Nat Methods*. <https://doi.org/10.1038/s41592-019-0344-8>
10. Fonger GC, Hakkinen P, Jordan S, Publicker S (2014) The National Library of Medicine's (NLM) Hazardous Substances Data Bank (HSDB): background, recent enhancements and future plans. *Toxicology* 325:209–216. <https://doi.org/10.1016/j.tox.2014.09.003>
11. Gu J, Gui Y, Chen L, Yuan G, Lu HZ, Xu X (2013) Use of natural products as chemical library for drug discovery and network pharmacology. *PLoS One* 8(4):1–10
12. Hastings J, Owen G, Dekker A, Ennis M, Kale N, Muthukrishnan V, Turner S, Swainston N, Mendes P, Steinbeck C (2016) ChEBI in 2016: improved services and an expanding collection of metabolites. *Nucleic Acids Res* 44(D1):D1214–D1219. <https://doi.org/10.1093/nar/gkv1031>. <http://europapmc.org/articles/PMC4702775>
13. Heinonen M, Shen H, Zamboni N, Rousu J (2012) Metabolite identification and molecular fingerprint prediction via machine learning. *Bioinformatics* 28(18):2333–2341. <https://doi.org/10.1093/bioinformatics/bts437>
14. Hoffmann N, Rein J, Sachsenberg TT, Hartler J, Haug K, Mayer G, Alka O, Dayalan S, Pearce JTM, Rocca-Serra P et al (2019) mzTab-M: a data standard for sharing quantitative results in mass spectrometry metabolomics. *Anal Chem* 91(5):3302–3310. <https://doi.org/10.1021/acs.analchem.8b04310>
15. Horai H, Arita M, Kanaya S, Nihei Y, Ikeda T, Suwa K, Ojima Y, Tanaka K, Tanaka S, Aoshima K, Oda Y, Kakazu Y, Kusano M, Tohge T, Matsuda F, Sawada Y, Hirai MY, Nakanishi H, Ikeda K, Akimoto N, Maoka T, Takahashi H, Ara T, Sakurai N, Suzuki H, Shibata D, Neumann S, Iida T, Tanaka K, Funatsu K, Matsuura F, Soga T, Taguchi R, Saito K, Nishioka T (2010) MassBank: a public repository for sharing mass spectral data for life sciences. *J Mass Spectrom* 45(7):703–714. <https://doi.org/10.1002/jms.1777>
16. Irwin JJ, Sterling T, Mysinger MM, Bolstad ES, Coleman RG (2012) ZINC: a free tool to discover chemistry for biology. *J Chem Inf Model* 52(7):1757–1768
17. Jeffryes JG, Colastani RL, Elbadawi-Sidhu M, Kind T, Niehaus TD, Broadbelt LJ, Hanson AD, Fiehn O, Tyo KEJ, Henry CS (2015) MINEs: open access databases of computationally predicted enzyme promiscuity products for

- untargeted metabolomics. *J Cheminform* 7:44. <https://doi.org/10.1186/s13321-015-0087-1>
18. Kanehisa M, Sato Y, Kawashima M, Furumichi M, Tanabe M (2016) KEGG as a reference resource for gene and protein annotation. *Nucleic Acids Res* 44(D1):D457–D462
 19. Keseler IM, Mackie A, Santos-Zavaleta A, Billington R, Bonavides-Martínez C, Caspi R, Fulcher C, Gama-Castro S, Kothari A, Krummenacker M, Latendresse M, Muñiz-Rascado L, Ong Q, Paley S, Peralta-Gil M, Subhraveti P, Velázquez-Ramírez DA, Weaver D, Collado-Vides J, Paulsen I, Karp PD (2017) The EcoCyc database: reflecting new knowledge about Escherichia coli k-12. *Nucleic Acids Res* 45:D543–D550
 20. Kim S, Thiessen PA, Bolton EE, Chen J, Fu G, Gindulyte A, Han L, He J, He S, Shoemaker BA, Wang J, Yu B, Zhang J, Bryant SH (2016) PubChem substance and compound databases. *Nucleic Acids Res* 44:D1202–D1213. <https://doi.org/10.1093/nar/gkv951>
 21. Klekota J, Roth FP (2008) Chemical substructures that enrich for biological activity. *Bioinformatics* 24(21):2518–2525. <https://doi.org/10.1093/bioinformatics/btn479>
 22. Larson EA, Hutchinson CP, Lee YJ (2018) Gas chromatography-tandem mass spectrometry of lignin pyrolyzates with dopant-assisted atmospheric pressure chemical ionization and molecular structure search with CSI:FingerID. *J Am Soc Mass Spectrom* 29(9):1908–1918. <https://doi.org/10.1007/s13361-018-2001-3>
 23. Ludwig M, Dürkop K, Böcker S (2018) Bayesian networks for mass spectrometric metabolite identification via molecular fingerprints. *Bioinformatics* 34(13):i333–i340. <https://doi.org/10.1093/bioinformatics/bty245>. Proceedings of Intelligent Systems for Molecular Biology (ISMB 2018)
 24. Meusel M, Hufsky F, Panter F, Krug D, Müller R, Böcker S (2016) Predicting the presence of uncommon elements in unknown biomolecules from isotope patterns. *Anal Chem* 88(15):7556–7566. <https://doi.org/10.1021/acs.analchem.6b01015>
 25. Mohimani H, Gurevich A, Shlemov A, Mikheenko A, Korobeynikov A, Cao L, Shcherbin E, Nothias LF, Dorrestein PC, Pevzner PA (2018) Dereplication of microbial metabolites through database search of mass spectra. *Nat Commun* 9(1):4035. <https://doi.org/10.1038/s41467-018-06082-8>
 26. Nelson SJ, Johnston WD, Humphreys BL (2001) Relationships in medical subject headings. In: Bean CA, Green R (eds) Relationships in the organization of knowledge. Kluwer Academic Publishers, Dordrecht, pp 171–184. <http://www.nlm.nih.gov/mesh/meshrels.html>
 27. Pluskal T, Castillo S, Villar-Briones A, Oresic M (2010) MZmine 2: modular framework for processing, visualizing, and analyzing mass spectrometry-based molecular profile data. *BMC Bioinf* 11:395. <https://doi.org/10.1186/1471-2105-11-395>
 28. Ramirez-Gaona M, Marcu A, Pon A, Guo AC, Sajed T, Wishart NA, Karu N, Djoumbou Feunang Y, Arndt D, Wishart DS (2017) YMDB 2.0: a significantly expanded version of the yeast metabolome database. *Nucleic Acids Res* 45:D440–D445
 29. Rasche F, Svatos A, Maddula RK, Böttcher C, Böcker S (2011) Computing fragmentation trees from tandem mass spectrometry data. *Anal Chem* 83(4):1243–1251. <https://doi.org/10.1021/ac101825k>
 30. Ridder L, van der Hooft JJJ, Verhoeven S, de Vos RCH, Bino RJ, Vervoort J (2013) Automatic chemical structure annotation of an LC-MS(n) based metabolic profile from green tea. *Anal Chem* 85(12):6033–6040. <https://doi.org/10.1021/ac400861a>
 31. Röst HL, Sachsenberg T, Aiche S, Bielow C, Weisser H, Aicheler F, Andreotti S, Ehrlich HC, Gutenbrunner P, Kenar E, Liang X, Nahnsen S, Nilse L, Pfeuffer J, Rosenberger G, Rurik M, Schmitt U, Veit J, Walzer M, Wojnar D, Wolski WE, Schilling O, Choudhary JS, Malmström L, Aebersold R, Reinert K, Kohlbacher O (2016) OpenMS: a flexible open-source software platform for mass spectrometry data analysis. *Nat Methods* 13(9):741–748. <https://doi.org/10.1038/nmeth.3959>
 32. Ruttkies C, Schymanski EL, Wolf S, Hollender J, Neumann S (2016) MetFrag relaunched: incorporating strategies beyond in silico fragmentation. *J Cheminform* 8:3. <https://doi.org/10.1186/s13321-016-0115-9>
 33. Schymanski EL, Ruttkies C, Krauss M, Brouard C, Kind T, Dürkop K, Allen FR, Vaniya A, Verdegem D, Böcker S, Rousu J, Shen H, Tsugawa H, Sajed T, Fiehn O, Ghesquière B, Neumann S (2017) Critical assessment of small molecule identification 2016: automated methods. *J Cheminf* 9:22. <https://doi.org/10.1186/s13321-017-0207-1>
 34. Shinbo Y, Nakamura Y, Altaf-Ul-Amin M, Asahi H, Kurokawa K, Arita M, Saito K, Ohta D, Shibata D, Kanaya S (2006) KNAP-SACK: a comprehensive species-metabolite relationship database. In: Saito K, Dixon RA, Willmitzer L (eds) Plant metabolomics.

- Biotechnology in agriculture and forestry, vol 57. Springer, Berlin, pp 165–181
35. Steinbeck C, Han Y, Kuhn S, Horlacher O, Luttmann E, Willighagen E (2003) The Chemistry Development Kit (CDK): an open-source Java library for chemo- and bioinformatics. *J Chem Inf Comput Sci* 43:493–500
36. Tautenhahn R, Cho K, Uritboonthai W, Zhu Z, Patti GJ, Siuzdak G (2012) An accelerated workflow for untargeted metabolomics using the METLIN database. *Nat Biotechnol* 30(9):826–828. <https://doi.org/10.1038/nbt.2348>
37. Tsugawa H, Kind T, Nakabayashi R, Yukihira D, Tanaka W, Cajka T, Saito K, Fiehn O, Arita M (2016) Hydrogen rearrangement rules: computational ms/ms fragmentation and structure elucidation using MS-FINDER software. *Anal Chem* 88(16):7946–7958. <https://doi.org/10.1021/acs.analchem.6b00770>
38. Wang R, Fu Y, Lai L (1997) A new atom-additive method for calculating partition coefficients. *J Chem Inf Comput Sci* 37(3):615–621. <https://doi.org/10.1021/ci960169p>
39. Wang R, Gao Y, Lai L (2000) Calculating partition coefficient by atom-additive method. *Perspect Drug Discov Des* 19(1):47–66. <https://doi.org/10.1023/A:1008763405023>
40. Wang Y, Kora G, Bowen BP, Pan C (2014) MIDAS: a database-searching algorithm for metabolite identification in metabolomics. *Anal Chem* 86(19):9496–9503. <https://doi.org/10.1021/ac5014783>
41. Wang M *et al* (2016) Sharing and community curation of mass spectrometry data with Global Natural Products Social molecular networking. *Nat Biotechnol* 34(8):828–837. <https://doi.org/10.1038/nbt.3597>
42. Weber RJM, Li E, Bruty J, He S, Viant MR (2012) MaConDa: a publicly accessible mass spectrometry contaminants database. *Bioinformatics* 28(21):2856–2857. <https://doi.org/10.1093/bioinformatics/bts527>
43. Willighagen EL, Mayfield JW, Alvarsson J, Berg A, Carlsson L, Jeliazkova N, Kuhn S, Pluskal T, Rojas-Chertó M, Spjuth O, Torrance G, Evelo CT, Guha R, Steinbeck C (2017) The Chemistry Development Kit (CDK) v2.0: atom typing, depiction, molecular formulas, and substructure searching. *J Cheminf* 9(1):33. <http://dx.doi.org/10.1186/s13321-017-0220-4>
44. Wishart DS, Feunang YD, Marcu A, Guo AC, Liang K, Vázquez-Fresno R, Sajed T, Johnson D, Li C, Karu N, Sayeeda Z, Lo E, Assempour N, Berjanskii M, Singhal S, Arndt D, Liang Y, Badran H, Grant J, Serra-Cayuela A, Liu Y, Mandal R, Neveu V, Pon A, Knox C, Wilson M, Manach C, Scalbert A (2018) HMDB 4.0: the human metabolome database for 2018. *Nucleic Acids Res* 46(D1): D608–D617. <http://dx.doi.org/10.1093/nar/gkx1089>
45. Wolf S, Schmidt S, Müller-Hannemann M, Neumann S (2010) In silico fragmentation for computer assisted identification of metabolite mass spectra. *BMC Bioinf* 11:148. <https://doi.org/10.1186/1471-2105-11-148>



Chapter 12

Annotation of Specialized Metabolites from High-Throughput and High-Resolution Mass Spectrometry Metabolomics

Thomas Naake, Emmanuel Gaquerel, and Alisdair R. Fernie

Abstract

High-throughput mass spectrometry (MS) metabolomics profiling of highly complex samples allows the comprehensive detection of hundreds to thousands of metabolites under a given condition and point in time and produces information-rich data sets on known and unknown metabolites. One of the main challenges is the identification and annotation of metabolites from these complex data sets since the number of authentic standards available for specialized metabolites is far lower than an account for the number of mass spectral features. Previously, we reported two novel tools, *MetNet* and *MetCirc*, for putative annotation and structural prediction on unknown metabolites using known metabolites as baits. *MetNet* employs differences between *m/z* values of MS1 features, which correspond to metabolic transformations, and statistical associations, while *MetCirc* uses MS/MS features as input and calculates similarity scores of aligned spectra between features to guide the annotation of metabolites. Here, we showcase the use of *MetNet* and *MetCirc* to putatively annotate metabolites and provide detailed instructions as to how those can be used. While our case studies are from plants, the tools find equal utility in studies on bacterial, fungal, or mammalian xenobiotic samples.

Key words Annotation, Plant metabolite, Specialized metabolite, Unknown metabolite, Metabolic modification, Molecular networking

1 Introduction

Based on estimations, the plant kingdom contains between 100,000 and 1 million metabolites [1, 2] with any species containing more than 5000 metabolites [3]. While the core central metabolism is comparable to nonplant species, a wide array of specialized (formerly known as secondary) metabolites contributes to the high number of predicted metabolites in plants. As summarized in Alseekh and Fernie [4], it is estimated that plants are able to synthesize about 27,000 different alkaloids [5–7], 10,000 different phenolics [6–8] and 16,000 different terpenoid compounds [6, 7]. Given this, one crucial aspect of plant metabolomics lies in

the de novo annotation of natural products requiring software solutions that are not based on previously reported metabolites (cf. [9]), that is, to select metabolite candidates for further identification in an unbiased way.

High-throughput mass spectrometry offers an unbiased approach to capture in a comprehensive manner the metabolic state of cells and tissues under given experimental conditions. For highly complex biological matrices (e.g., from plant tissues), hundreds to thousands of metabolites are typically detected within a liquid chromatography (LC) mass spectrometry (MS) profile [10] imposing difficulties in accurately processing this high number of metabolites [4]. While for gas chromatography (GC)-MS extensive database entries for primary metabolites exist (e.g., [11]), this is less common for LC-MS data [10]. Often LC-MS data contain a high proportion of unknown metabolites and it is a critical challenge in metabolic profiling to select metabolic features for further identification. Often the metabolites measured by LC-MS belong to the class of specialized metabolites. These metabolites typically display a highly diverse chemistry [12], exhibit a high dynamic range depending on environmental conditions [3] and on enzymatic capabilities of the species or ecotype (e.g., [13]). They, furthermore, usually display strong compartmentalization at the tissue, cellular, and subcellular levels [14], requiring diverse and deliberate sampling and methods which intensifies the analysis of these highly complex samples.

Metabolic profiles can be acquired either in MS1 or in MS2 mode. While MS1 profiling allows for rapid throughput of many samples, MS2 captures structural information on metabolic features by fragmentation. For MS2, two modes of data acquisition exist: data-dependent acquisition (DDA) and data-independent acquisition (DIA). While DDA selects a certain number of precursor ions for fragmentation, DIA uses all ions (or ions in a certain m/z range) for fragmentation and thus allows to comprehensively capture fragmentation data on detected metabolic features.

Different software solutions exist to reconstruct metabolic networks and/or putatively annotate metabolites (reviewed in [10]). Starting from MS1 data, typically peak tables are annotated for isotopes, fragments and adducts (e.g., by CAMERA [15] that interacts directly with the processed peak data from xcms or AStream [16] that reduces redundancy between various chemical forms in one compounds and removed noise artifacts and potential sample contamination). Several software solutions exist to putatively annotate metabolic features within metabolomics datasets: xMSannotator [17] employs metabolic pathway associations by combining data-driven approaches (using intensity profiles, retention time, mass defect, isotope, and adduct characteristics) and chemical, metabolic, and environmental databases. PlantMat [18] matches features of the dataset against combinatorial enumerations of

predefined aglycones and decorating building blocks (e.g., glycosyl subunits). Another tool, mummichog [19], uses network analysis for pathway enrichment, prediction of functional activity and putative metabolite identification. While these tools are valuable when it comes to analysis of well-defined metabolomes with a good coverage of metabolites (`xMSannotator`), known aglycones (PlantMat) or availability of genome-scale metabolic models (mummichog), other software tools are needed to putatively annotate de novo metabolites from MS1 data.

For MS2 data, Mass2Motifs [9, 20] applies Latent Dirichlet Allocation to metabolomics data sets to assign conserved fragments and losses of metabolites that reflect chemical substructures. This approach allows to group molecules based on shared substructures regardless of spectral similarity. Another approach is MetFamily, to detect regulated metabolite families by hierarchical cluster analysis of MS2 spectra and linking these clusters to MS1 data [21].

We previously created two novel open-source software tools, `MetNet` [22] and `MetCirc` [23], for dedicated putative annotation of metabolic features acquired by MS1 and MS2 (DDA and DIA alike) mass spectrometry. Both `MetNet` and `MetCirc` use molecular networking to guide metabolite annotation. Network construction from high-resolution MS1 data was pioneered by Breitling et al. [24] who proposed to use pair-wise differences of metabolic features for network construction. Similarly, `MetNet` uses the information-rich output of MS1 metabolic profiling to link features based on structural information and quantitative information (i.e., intensity values). In brief, `MetNet` matches all pair-wise differences of metabolic features against a table of known metabolic transformations that typically occur or are specifically searched for (e.g., glycosylation or hydroxylation). Furthermore, `MetNet` creates networks from quantitative data by statistically associating metabolic features to each other. Metabolites that share a specific pathway are normally coregulated and show a statistical association (e.g., [25–27]). The two results are combined to a consensus network that contains both information from structural and quantitative data and enables to formulate structural hypotheses on unknown mass signals linked to known metabolites.

For MS2 data, `MetCirc` uses molecular networking [28, 29] based on spectral similarity scores between MS2 features based on a modified normalized dot product (NDP, see [28] for further details) taking into account either shared fragments or main neutral losses. Small molecules, as produced by plants, often return few fragments during MS2 fragmentation. Sometimes only a few fragments are shared among MS2 spectra and diagnostic for a certain compound family. Compared to the classically used MS/MS similarity scoring based on shared fragments, we completed this scoring procedure by taking into account a second similarity measure incorporating shared neutral losses, which help to obtain

biochemically meaningful MS/MS groupings that can be further analyzed. While MetCirc was initially designed to facilitate the exploration of MS2 metabolomics data obtained from cross-species/cross-tissues comparative experiments, MetCirc also improves the dereplication of known and unknown metabolites within its interactive Shiny interface.

In this chapter, we provide stepwise protocols for the use of MetNet and MetCirc in order to achieve compound class grouping of MS1 and MS2 data and for the annotation of structural similarities among clustered MS2. To this end, we employed a data set of MS2 spectra produced from the UPLC-qTOF⁺MS analysis of polar to semipolar extracts of *Nicotiana attenuata* and collection of MS2 spectra using a DIA method (Metabolights MTBLS335). *N. attenuata* has been developed as ecological model system to elucidate the role of specialized metabolites in mediating plant-insect interactions. Main specialized metabolites detected in positive ionization mode from leaf methanolic extracts comprise alkaloids, flavonoid glycosides, hydroxycinnamic-polyamine conjugates, O-acyl sugars, and 17-hydroxygeranylinalool diterpene glycosides (17-HGL-DTG) [30, 31]. A previous metabolomics study reported that approximately 360 nonredundant compound-level deconvoluted MS2 spectra are typically collected by DIA MS2 from *N. attenuata* leaf extracts, of those around 60 compounds are annotated according to levels 1 and 2 of the Metabolomics Standard Initiative [30–32]. For this dataset, scores higher than 0.8 were retrieved for less than 10% of the MS2 data set when querying (in 2015) MS2 spectra against entries from the Massbank public depository [28]. The abovementioned 17-HGL-DTG compound class corresponds to abundant 17-hydroxygeranylinalool backbones decorated with sugar and malonyl moieties and are employed as efficient defenses against phytophagous insects [33]. The 17-HGL-DTG chemotype comprises more than 40 predicted compounds in a given plant tissue and the rich fragmentation patterns produced by these compounds provide sufficient resolution for *a priori* structure annotation by mass spectral alignment [32]. Hence, an MS2 cluster enriched in 17-HGL-DTs is used to illustrate MetCirc and MetNet annotation strategies from plant metabolomics profiles.

Noteworthy, while both programs were written with the analysis of plant metabolism in mind, they will be equally applicable to annotate specialized metabolites of other organisms as well as xenobiotics.

2 Materials

2.1 Software Requirements and Packages

1. R software (currently version ≥ 3.5 required) available via <https://www.r-project.org/bin/>.
2. *optional:* install RStudio (v1.0.153 or higher) available via <https://www.rstudio.com/products/rstudio/download/>.
3. Internet connection to download BiocManager from CRAN (<https://cran.r-project.org/>) and MetNet and/or MetCirc package from Bioconductor (<https://bioconductor.org/>).
4. MetNet v1.0.0, MetCirc v1.12.0 (as used here, *see Notes 1 and 2*).
5. *optional:* amap v0.8-16 for clustering (as used here) or a similar package.
6. Internet explorer, recommended: Firefox (version 63.0 or higher) or Google Chrome (version 71.0.3578.80 or higher) (MetCirc).

2.2 Input Data

1. $m \times n$ peak table of m MS1 features containing m/z values, retention time and intensity values for n samples (MetNet, *see Notes 3–5*). A peak table can be acquired by following the protocol of Shimizu et al. [34] using an Orbitrap mass spectrometer or similar and running the below-mentioned xcms/CAMERA script.
2. MS2 data from DDA or DIA mass spectrometry formatted as a peak table or as a .msp file (MetCirc, *see Note 4*). A peak table can be acquired according to Li et al. [28] using an Orbitrap or qTOF mass spectrometer.

2.3 Hardware Requirements

1. Desktop computer.
2. *preferably:* R Server environment with ≥ 64 GB RAM for large peak tables with several thousand metabolic features (MetNet).

3 Methods

3.1 Prerequisites

1. *if required:* Install R (<https://cran.r-project.org/bin/>) depending on the operating system.
2. *optional:* Install RStudio (<https://www.rstudio.com/products/rstudio/download/>) depending on the operating system.
3. Open R Session, either by opening R, Rstudio or (under Linux) by entering R to the Terminal and hitting Enter.
4. Install BiocManager by entering to the R Session and hitting enter:

```
if (!requireNamespace("BiocManager", quietly = TRUE))
  install.packages("BiocManager")
```

3.2 MetNet

1. Install MetNet via Bioconductor framework by entering to the R Session and hitting enter:

```
BiocManager::install("MetNet", version = "3.8")
```

2. Load the namespace of the package and attach it to the search space to the R environment by entering to the R Session and hitting Enter.

```
library(MetNet)
```

3. Load the peak table to the R session. This step will differ depending on how the peak table is stored. If the peak table is stored in tabular format in a file (here in the peaklist.txt file) load the file by

```
peaklist <- read.table("path_to_peaklist.txt_file/peaklist.txt")
```

MetNet is compatible with the output of xcms/CAMERA [15, 35] pipeline in R. Alternatively to the step by loading an existing peak table by `read.table` a peak list can also be created from NetCDF/mzXML by using xcms/CAMERA, for example, by entering to the R console.

```
xset <- xcmsSet(method = "centWave", ppm = 30, snthresh = 10, peakwidth = c(5,20))
xset2 <- group(xset, method = "density", minfrac = 0.5, minsamp = 2, bw = 2, mzwid = 0.025)
xset3 <- retcor(xset2, family = "s", plottype = "m", missing = 1, extra = 1, span = 1)
xset4 <- group(xset3, method = "density", bw = 2, mzwid = 0.025, minfrac = 0.5,
minsamp = 2)
xset5 <- fillPeaks(xset4, method = "chrom")
an <- xsAnnotate(xset5)
anF <- groupFWHM(an, perfwhm = 0.6)
anI <- findIsotopes(anF, mzabs = 0.01)
anIC <- groupCorr(anI, cor_eic_th = 0.75, graphMethod = "lpc")
anFA <- findAdducts(anIC, polarity = "positive")
peaklist <- getPeaklist(anFA)
```

Parameters will depend on the specifications of the employed chromatography and mass spectrometry system according to [36].

4. *if required*: After import of the peak table the peak table needs to be normalized (e.g., each column by its 75% quantile and log2 transformation of intensity values) (*see Note 6*),
5. Define the transformation that will be looked for (and *if required*: associated changes in retention time, *see Note 7*). As an example we want to search for transformations that are

based on hydroxylation (associated with earlier retention time on a reverse-phase column, +15.99), malonylation (higher retention time, +86.00), and addition of rhamnose (earlier retention time, +146.06), glucose (earlier retention time, +162.05). Create the table with R by entering the command lines below and hitting Enter.

```
transformation <- rbind(
  c("Hydroxylation (-H)", "O", 15.9949146221, "-"),
  c("Malonyl group (-H2O)", "C3H2O3", 86.0003939305, "+"),
  c("Rhamnose (-H2O)", "C6H10O4", 146.0579088094, "-"),
  c("Glucose (-H2O)", "C6H10O5", 162.0528234315, "-"))
transformation <- data.frame(group = transformation[,1],
  formula = transformation[,2],
  mass = as.numeric(transformation[,3]), rt = transformation[,4])
```

6. Create the adjacency matrix that reports a connection between two metabolic features if the respective difference in *m/z* values matches to a transformation defined in the `data.frame` object `transformation`. Molecular weight difference w_X is defined by $w_X = |w_A - w_B|$, where w_A is the molecular weight of substrate A, and w_B is the molecular weight of product B [24] (typically, *m/z* values will be used as a proxy for the molecular weight since the molecular weight is not directly derivable from MS1 data, *see Note 7*). The function `createStructuralAdjacency` (*see Note 8*) will then return this adjacency matrix together with another matrix telling about the type of connection (in the above case if there is addition/loss of hydroxyl, malonyl, rhamnose or glucose moiety). Enter the following to the R console and hit Enter.

```
struct_adj <- createStructuralAdjacency(peaklist, transformation,
  ppm = 10)
```

7. *optional:* Check the proposed connection by their retention time difference and remove false positive connection that had by chance the same/similar *m/z* difference but elute before or after the reference feature (*see Notes 3 and 9–11*). For example on data acquired by reverse-phase LC-MS, if there is a connection between features based on the difference of glucose addition and the feature with the higher *m/z* value elutes later, `rtCorrection` will remove the connection between these features, since the addition of glucose should render the feature more polar, thus eluting earlier. The expected retention time shift has to be given in the column “`rt`” of the `transformation` object. To adjust for retention time enter to the R console and hit Enter.

```
struct_adj_rt <- rtCorrection(struct_adj, peaklist, transformation)
```

8. Create the adjacency matrix based on statistical associations of intensity values between metabolic features. The `createStatisticalAdjacency` function provided by `MetNet` allows for an approach proposed by [37] in that it uses different statistical frameworks and combines the results from the different statistical models into a consensus adjacency matrix. Currently, the following methods are implemented: Least absolute shrinkage and selection operator [38], Random Forest [39], Pearson and Spearman correlation (including partial and semi-partial correlation) [25], context likelihood of relatedness (CLR) [40] the algorithm for the reconstruction of accurate cellular networks (ARACNE) [41] and constraint-based structure learning [42] (Bayesian networks, *see Notes 12–15*). By way of example, we will create the consensus adjacency matrix by using Pearson and Spearman correlation and the models CLR and ARACNE by entering to the R console and hitting Enter.

```
stat_adj <- createStatisticalAdjacency(peaklist_cut,
model = c("pearson", "spearman", "clr", "aracne"), correlation_adjust = "BH")
```

9. Combine the adjacency matrices from structural and statistical association. Only these connections will be reported in the final “consensus” network that are shared in the two adjacency matrix are present in both adjacency matrices (intersection is taken, *see Note 14*). To create the consensus matrix enter to the R console and hit Enter.

```
cons_adj <- combineStructuralStatistical(struct_adj[[1]], stat_adj)
cons_adj <- combineStructuralStatistical(struct_adj_rt[[1]], stat_adj)
```

10. *if required:* After this step the `MetNet` pipeline is finished. Commonly used network analysis tools (e.g., from `igraph` or `sna` package) can be employed to analyze the network. Known metabolites should guide the putative annotation of unknown metabolites to formulate structural hypotheses (cf. Fig. 1).

Here we will retrieve the components from the network (subgraphs in which any two vertices are connected to each other) that will contain possible associated metabolites by entering to the R console and hitting Enter.

```
library(igraph)
net <- graph_from_adjacency_matrix(cons_adj, mode = "undirected", diag = FALSE)
net_comp <- components(net)
```

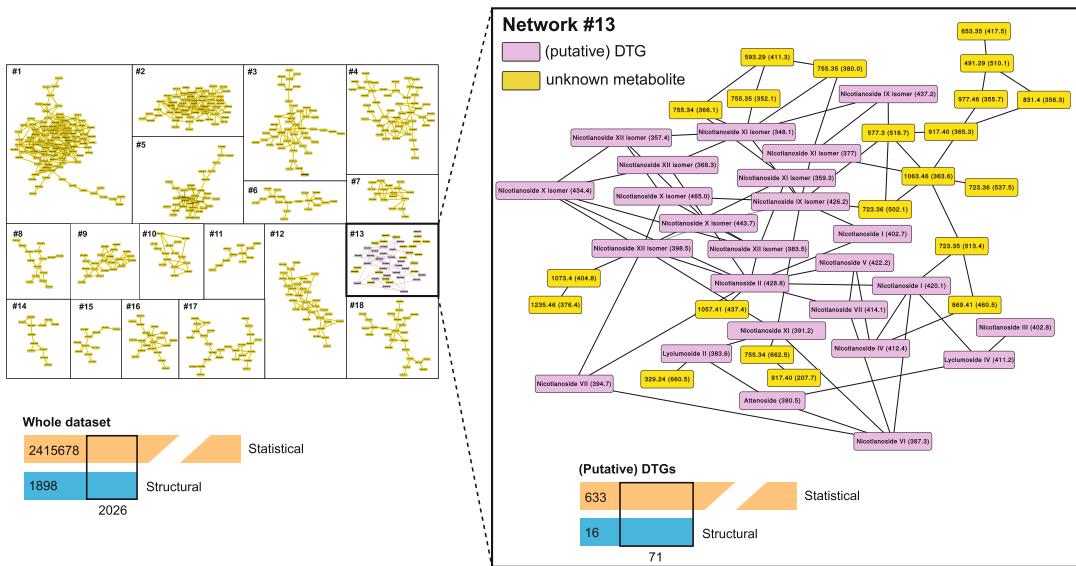


Fig. 1 Typical result of MetNet pipeline. MetNet created based on a *Nicotiana* MS2 dataset, a network considering the m/z shifts corresponding to hydroxylation, malonylation, rhamnosylation, and glucosylation and statistical associations using Pearson and Spearman correlation as well as CLR and ARACNE as statistical models (see also [22] for further details). For the whole dataset, 2,417,704 and 3924 links between metabolic features were determined for statistical and structural associations (intersection of 2026 links). Displayed are network components with 10 or more members. Network #13 is enriched in diterpene glycosides (DTG) but also contains unknown metabolites collected in the dataset. Based on m/z differences to known DTG, these unknowns can be putatively annotated. For network #13, 704 and 87 links between metabolic features were determined for statistical and structural annotations (intersection of 71 links)

3.3 MetCirc

1. Install MetCirc via Bioconductor framework by entering to the R Session and hitting enter:

```
BiocManager::install("MetCirc", version = "3.8")
```

2. Load the namespace of the package and attach it to the search space to the R environment by entering to the R Session and hitting Enter.

```
library(MetCirc)
```

3. Load MS2 data and convert to the MSP class by either two ways:
 - (a) Load a data frame (here msms.txt) with column “id” (unique identifiers for MS/MS features), and columns “mz” and “intensity” comprising the fragment ions and their intensities (see Notes 16–19). The data frame can be loaded to the R session by entering and hitting Enter.

```
msms <- read.table("path_to_msms.txt/msms.txt")
msp <- convert2MSP(msms)
```

- (b) Load a data frame in .msp file format, a typical data format for storing MS/MS libraries. Required properties of such a data frame are the name of the metabolite (row entry “NAME:”), the m/z value of the precursor ion (“PRECURSORMZ” or “EXACTMASS:”), the number of peaks of the feature (“Num Peaks:”), and information on fragments and peak areas (see Note 20). To load the file in .MSP format and convert it to the MSP class object, enter the following to the R session and hit Enter.

```
msms <- read.table("path_to_msms.msp/msms.msp")
msp <- convertMSP2MSP(msms)
```

4. *optional*: Calculate neutral losses from precursor ions by entering to the R console and hitting Enter.

```
msp <- msp2FunctionalLossesMSP(msp)
```

5. *optional*: Combine several MSP objects (here `msp1`, `msp2`) to one MSP object (here `msp`) by entering to the R console and hitting Enter.

```
msp <- combine(msp1, msp2)
```

6. Bin the fragments. Due to slight differences in m/z values over measurements, fragments might have m/z values which differ from other fragments even though they are in theory identical. Binning will bin together fragment ions which are similar to allow for comparison between m/z values (see Notes 21–23) by entering to the R console and hitting enter.

```
binnedMSP <- binning(msp = msp, tol = 0.01)
```

7. Similarity between two MS/MS features is calculated according to the NDP. For a considered MS/MS pair, peak intensities of shared m/z values for precursor/fragment ions and neutral losses are employed as weights $W_{S1,i}$ and $W_{S2,i}$ within the following formula:

$$\text{NDP} = \frac{\sum (W_{S1,i} \cdot W_{S2,i})^2}{\sum W_{S1,i}^2 \cdot \sum W_{S2,i}^2},$$

with S1 and S2 the spectra 1 and 2, respectively, of the i th of j common peaks. Weights are calculated according to $W = [\text{peak intensity}]^m \cdot [m/z]^n$, with $m = 0.5$ and $n = 2$ as default values as suggested by MassBank [28]. Calculate similarity (see Notes 24–25) according to the NDP by entering to the R console and hitting enter.

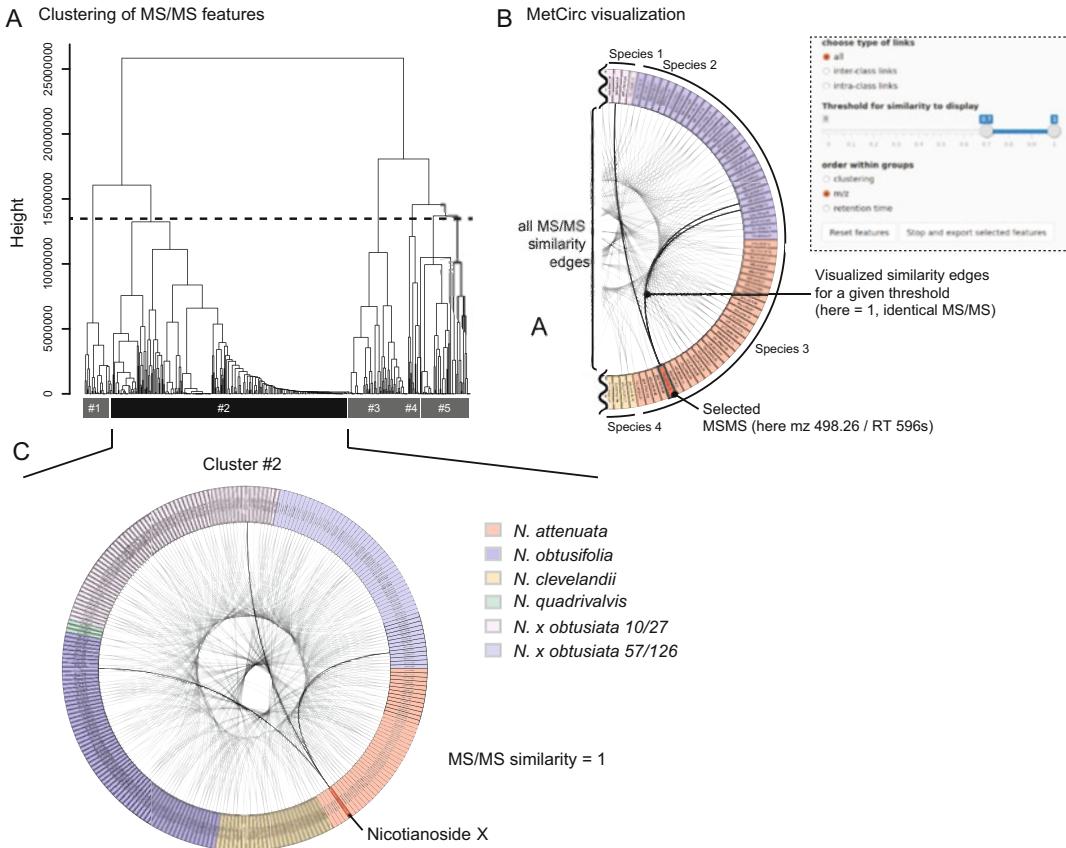


Fig. 2 Typical result of MetCirc pipeline. (a) Hierarchical clustering of MS/MS features based on distance between similarity scores prior to MetCirc analysis. Clustering will extract clusters containing highly similar MS/MS features (with high similarity scores). (b) Interface of the MetCirc Shiny application. Within the Shiny application MS/MS features can be navigated, reordered, annotated, and links between MS2 features thresholded and selected based on the type of links. (c) Exemplary MetCirc output for diterpene glycosides (DTG) for six *Nicotiana* species. For all panels, MS2 spectra were collected from leaves of six *Nicotiana* species and used for processing through the MetCirc pipeline. MetCirc visualization is built for Cluster #2 enriched in DTGs. From this compound class, nicotianoside X previously identified in *Nicotiana attenuata* is selected in the interface. Similarity threshold is fixed to 1 (cross-species MS2 identity) which readily allows for the dereplication (reidentification) of nicotianoside X in three of the five other *Nicotiana* species tested

```
similarityMat <- createSimilarityMatrix(binnedMSP)
```

8. *optional:* Clustering of MS/MS features based on similarity score (cf. Fig. 2a). In case the library of MS/MS feature is high in number, highly related features can be extracted using clustering. By way of example, to define five clusters by Spearman correlation as distance measure and retrieve the similarity matrix with members of cluster 2, enter the following to the R console and hit Enter.

```

hClustMSP <- amap::hcluster(similarityMat, method = "spearman")
cutTreeMSP <- cutree(hClustMSP, k = 5)
cluster2 <- names(cutTreeMSP)[as.vector(cutTreeMSP) == "2"]
similarityMat_cluster2 <- similarityMat[cluster2, cluster2]

```

9. The adjacency matrix can be visualized within an interactive shiny session. To start the visualization enter to the R console (*see Note 26*) and hit Enter.

```
selectedFeatures <- shinyCircos(similarityMat, msp = msp)
```

The shiny interface will be loaded, upon which the type of link to be displayed can be selected in the left-side panel (inter-class links for links between groups, intra-class links for links within groups, all for all links within and between groups). Ordering inside of groups can be changed by selecting the respective option in the sidebar panel (based on similarity (“clustering”), *m/z* value (“*m/z*,” in ascending order), retention time (“retention time,” in ascending order, *see Fig. 2b*). In the “Appearance” tab, the size of the plot can be changed, the precision of the displayed values for the *m/z* and retention time values, as well as if a legend is displayed or not.

10. Change the slider input (“Threshold for similarity to display,” *see Fig. 2b*) to threshold similarity scores within the plot (*see Notes 27 and 28*).
11. Navigate through MS/MS features and display information by single-clicking on the sectors in the circular plot. Update a selected feature by changing information (name, class, information, adduct) in the sidebar panel and click “Update annotation” to save changes to the MSP-object `msp`. A highly similar, known feature should be used as a reference (cf. [32]). On exiting the application via “Stop and export selected features” (*see Note 29*), the updated MSP-object will be retrievable by.

```
selectedFeatures$msp
```

12. *optional:* Permanently select MS/MS features by double-clicking on the sector. Reset all permanently selected features by clicking on “Reset features.” Permanently selected features will be returned to the `selectedFeatures` object when exiting the application via “Stop and export selected features” (*see Note 29*).

4 Notes

1. Consult the vignette of `MetNet` for updated functionality or further information (<https://bioconductor.org/packages/release/bioc/vignettes/MetNet/inst/doc/MetNet.pdf>).
2. Consult the vignette of `MetCirc` for updated functionality or further information (<https://bioconductor.org/packages/release/bioc/vignettes/MetCirc/inst/doc/MetCirc.pdf>).
3. Chromatographic resolution has to be sufficiently good and reliable to assure that batch-wise retention time alignment correction is technically feasible.
4. Required data-sets have to be generated using high-resolution MS instruments (mass accuracy ideally <15 ppm).
5. Sampling of diverse conditions/tissues is preferred to have strong differences between samples which will be captured by statistical methods.
6. `MetNet` does not impose any requirements for data normalization, filtering, etc. However, the user has to make sure that the data is properly preprocessed. These include division by internal standard, \log_2 transformation of intensity values, noise filtering, removal of features that do not represent mass features/metabolites, removal of isotopes, etc.
7. Occurrence of stereoisomers and structural isomers impose problems/redundancy for network inference that should be taken into account during network interpretation.
8. The `ppm` parameter in the function `createStructuralAdjacency` should be set according to the mass spectrometer setup.
9. For ambiguous metabolic transformations, the retention time correction can also be omitted by setting the respective data frame entry to “?”.
10. Expert knowledge is required to assess how a specific transformation will change the polarity of a molecule and running the `rtCorrection` function should be handled with care.
11. Depending on the position of addition/loss of a specific group, the retention time shift can be reverted (cf. [22]).
12. Make sure to use the correct abbreviations for the statistical methods for `createStatisticalAdjacency`: “lasso” for LASSO, “randomForest” for Random Forest, “clr” for CLR, “aracne” for ARACNE, “pearson” for Pearson correlation, “pearson_partial” or “pearson_sempartial” for partial or semi-partial Pearson correlation, “spearman” for Spearman correlation, “spearman_partial” or “spearman_sempartial” for partial

or semipartial Spearman correlation, “bayes” for constraint-based structure learning.

13. For the function `createStatisticalAdjacency`, additional parameters can be passed to the functions `lasso`, `randomForest`, `clr`, `aracne`, `correlation`, `bayes`, and/or `consensusAdjacency` (e.g., for alpha value adjustment or adjustment of epsilon parameter in `clr`).
14. The parameter `threshold` in `createStatisticalAdjacency` and `combineStructuralStatistical` should be set accordingly, if another method than “central.graph” (default) is used.
15. Choice of statistical methods should depend on the specific question in mind and the size of the peak table, typically LASSO, Random Forest, and constraint-based structure learning will take a long calculation time for large data sets.
16. When loading the data frame in tabular format, make sure that in the “id” column, all entries belonging to one MS/MS feature are identical for all fragments, that is, the column “id” has to be a unique descriptor for MS/MS features.
17. When loading the data frame in tabular, make sure that the intensity values are in percent (%), not decimal) with the intensity value with the highest intensity has a value of 100.0 (%), instead of 1.00) and all other fragments have the respective relative intensity (e.g., 10% means 10% smaller intensity than the feature with 100%).
18. When loading the data frame in tabular format, the “id” column has to contain the *m/z* value of the precursor ion, and optionally the retention time (add “`rt = TRUE`” to `convert2MSP`).
19. When loading the data frame in tabular format, the data frame may contain additional information that will be stored in the MSP object and may help with derePLICATION: `rt` (retention time of features), `names` (name of the MS/MS feature), `information` (additional information about the MS/MS feature), `classes` (class of the MS/MS feature), `adduct` (adduct ion name of MS/MS feature). Make sure that for all instances (except `rt`), all entries per MS/MS feature are identical.
20. When loading the data frame in .MSP file format, the data frame may optionally contain the row entries “RETENTION-TIME:” (containing the retention time of the precursor), “CLASS” (class of MS/MS feature), “ADDITION-NAME:”/“PRECURSORTYPE:” (adduct ion type of MS/MS feature), and “INFORMATION:” (additional information about MS/MS feature) for each MS/MS feature within the data frame.

21. The `tol` parameter in the `binning` function has to be set according to the mass spectrometry setting (e.g., 0.01 means that fragments with a *m/z* difference of 0.01 to a bin will be put in this bin).
22. The function `binning` bins fragments together based on minimal distance to bins which were calculated either by the mean or the median of fragments according to the `tol` parameter. The user can specify by the `method` argument if median or mean will be used.
23. The function `binning` expects a vector (group argument) which comprises membership of the entries in the `msp` object, to a compartment, species, individual, and so on. If group is not specified `binning` will create an internal dummy variable group (“a” with the length of the `msp` object).
24. For calculation of the similarity matrix, a matrix that has per row one MS/MS feature and in the corresponding columns the fragments, entries of the matrix are intensity values (in %) for a specific fragment ion (the function `binning` will return such a matrix).
25. For calculation of the similarity matrix, care has to be taken with setting the parameters `m` and `n` (default to $m = 0.5$ and $n = 2$ as suggested by MassBank). When peak intensities should not be taken into account, `m` should be set to $m = 0$.
26. Before running `shinyCircos`, make sure that the features in `similarityMat` and `msp` are identical and in same order.
27. Typically, MS/MS features are further analyzed that share a NDP score ≥ 0.55 (cf. [43]); however, depending on the grade of fragmentation also MS/MS pairs can be analyzed.
28. Threshold the similarity score < 1 to exclude identical features from visualization.
29. When stopping `shinyCircos`, always do via “Stop and export selected features,” only via this way will the selected features and modified `msp` file be exported to the R session.

Acknowledgments

T.N. acknowledges support by the IMPRS-PMPG program and A.R.F. the support of Max Planck Society. E. G. acknowledges the support by the Deutsche Forschungsgemeinschaft Excellence Initiative to the University of Heidelberg and by the Centre National de la Recherche Scientifique.

References

1. Dixon RA, Strack D (2003) Phytochemistry meets genome analysis, and beyond. *Phytochemistry* 62:815–816
2. Rai A, Saito K, Yamazaki M (2017) Integrated omics analysis of specialized metabolism in medicinal plants. *Plant J* 90:764–787
3. Fernie AR, Trethewey RN, Krotzky AJ et al (2004) Metabolite profiling: from diagnostics to systems biology. *Nat Rev Mol Cell Biol* 5:763–769
4. Alseekh S, Fernie AR (2018) Metabolomics 20 years on: what have we learned and what hurdles remain? *Plant J* 94:933–942
5. Ziegler J, Facchini PJ (2008) Alkaloid biosynthesis: metabolism and trafficking. *Annu Rev Plant Biol* 59:735–769
6. Wink M (2004) Phytochemical diversity of secondary metabolites. In: Goodman RM (ed) *Encyclopedia of plant and crop science*. Marcel Dekker, New York, pp 915–919
7. Wink M (2015) Modes of action of herbal medicines and plant secondary metabolites. *Medicines (Basel)* 2:251–286
8. Tohge T, Alseekh S, Fernie AR (2014) On the regulation and function of secondary metabolism during fruit development and ripening. *J Exp Bot* 65:4599–4611
9. Van Der Hooft JJJ, Wandy J, Young F et al (2017) Unsupervised discovery and comparison of structural families across multiple samples in untargeted metabolomics. *Anal Chem* 89:7569–7577
10. Perez De Souza L, Naake T, Tohge T et al (2017) From chromatogram to analyte to metabolite. How to pick horses for courses from the massive web resources for mass spectral plant metabolomics. *Gigascience* 6:1–20
11. Kopka J, Schauer N, Krueger S et al (2005) GMD@CSB.DB: the Golm metabolome database. *Bioinformatics* 21:1635–1638
12. D'auria JC, Gershenson J (2005) The secondary metabolism of *Arabidopsis thaliana*: growing like a weed. *Curr Opin Plant Biol* 8:308–316
13. Li X, Svedin E, Mo HP et al (2014) Exploiting natural variation of secondary metabolism identifies a gene controlling the glycosylation diversity of dihydroxybenzoic acids in *Arabidopsis thaliana*. *Genetics* 198:1267
14. Sweetlove LJ, Fernie AR (2013) The spatial organization of metabolism within the plant cell. *Annu Rev Plant Biol* 64:723–746
15. Kuhl C, Tautenhahn R, Bottcher C et al (2012) CAMERA: an integrated strategy for compound spectra extraction and annotation of liquid chromatography/mass spectrometry data sets. *Anal Chem* 84:283–289
16. Alonso A, Julia A, Beltran A et al (2011) AStream: an R package for annotating LC/MS metabolomic data. *Bioinformatics* 27:1339–1340
17. Uppal K, Walker DI, Jones DP (2017) xMSannotator: an R package for network-based annotation of high-resolution metabolomics data. *Anal Chem* 89:1063–1067
18. Qiu F, Fine DD, Wherritt DJ et al (2016) PlantMAT: a metabolomics tool for predicting the specialized metabolic potential of a system and for large-scale metabolite identifications. *Anal Chem* 88:11373–11383
19. Li SZ, Park Y, Duraisingham S et al (2013) Predicting network activity from high throughput metabolomics. *PLoS Comput Biol* 9: e1003123
20. Van Der Hooft JJ, Wandy J, Barrett MP et al (2016) Topic modeling for untargeted substructure exploration in metabolomics. *Proc Natl Acad Sci U S A* 113:13738–13743
21. Treutler H, Tsugawa H, Porzel A et al (2016) Discovering regulated metabolite families in untargeted metabolomics studies. *Anal Chem* 88:8082–8090
22. Naake T, Fernie AR (2019) MetNet: metabolite network prediction from high-resolution mass spectrometry data in R aiding metabolite annotation. *Anal Chem* 91:1768–1772
23. Naake T, Gaquerel E (2017) MetCirc: navigating mass spectral similarity in high-resolution MS/MS metabolomics data. *Bioinformatics* 33:2419–2420
24. Breitling R, Ritchie S, Goodenow D et al (2006) Ab initio prediction of metabolic networks using Fourier transform mass spectrometry data. *Metabolomics* 2:155–164
25. Steuer R (2006) Review: on the analysis and interpretation of correlations in metabolomic data. *Brief Bioinform* 7:151–158
26. Morreel K, Saeys Y, Dima O et al (2014) Systematic structural characterization of metabolites in *Arabidopsis* via candidate substrate-product pair networks. *Plant Cell* 26:929–945
27. Gaquerel E, Kuhl C, Neumann S (2013) Computational annotation of plant metabolomics profiles via a novel network-assisted approach. *Metabolomics* 9:904–918
28. Li D, Baldwin IT, Gaquerel E (2015) Navigating natural variation in herbivory-induced secondary metabolism in coyote tobacco

- populations using MS/MS structural analysis. *Proc Natl Acad Sci U S A* 112:E4147–E4155
29. Watrous J, Roach P, Alexandrov T et al (2012) Mass spectral molecular networking of living microbial colonies. *Proc Natl Acad Sci U S A* 109:E1743–E1752
30. Gaquerel E, Heiling S, Schoettner M et al (2010) Development and validation of a liquid chromatography-electrospray ionization-time-of-flight mass spectrometry method for induced changes in *Nicotiana attenuata* leaves during simulated herbivory. *J Agric Food Chem* 58:9418–9427
31. Li DP, Heiling S, Baldwin IT et al (2016) Illuminating a plant's tissue-specific metabolic diversity using computational metabolomics and information theory. *Proc Natl Acad Sci USA* 113:E7610–E7618
32. Heiling S, Khanal S, Barsch A et al (2016) Using the knowns to discover the unknowns: MS-based dereplication uncovers structural diversity in 17-hydroxygeranylinalool diterpene glycoside production in the Solanaceae. *Plant J* 85:561–577
33. Heiling S, Schuman MC, Schoettner M et al (2010) Jasmonate and ppHsystemin regulate key malonylation steps in the biosynthesis of 17-hydroxygeranylinalool diterpene glycosides, an abundant and effective direct defense against herbivores in *Nicotiana attenuata*. *Plant Cell* 22:273–292
34. Shimizu T, Watanabe M, Fernie AR et al (2018) Targeted LC-MS analysis for plant secondary metabolites. *Methods Mol Biol* 1778:171–181
35. Smith CA, Want EJ, O'maille G et al (2006) XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Anal Chem* 78:779–787
36. Patti GJ, Tautenhahn R, Siuzdak G (2012) Meta-analysis of untargeted metabolomic data from multiple profiling experiments. *Nat Protoc* 7:508–516
37. Marbach D, Costello JC, Kuffner R et al (2012) Wisdom of crowds for robust gene network inference. *Nat Methods* 9:796
38. Tibshirani R (1996) Regression shrinkage and selection via the lasso. *J Roy Stat Soc B Met* 58:267–288
39. Breiman L (2001) Random forests. *Mach Learn* 45:5–32
40. Faith JJ, Hayete B, Thaden JT et al (2007) Large-scale mapping and validation of *Escherichia coli* transcriptional regulation from a compendium of expression profiles. *PLoS Biol* 5:e8
41. Margolin AA, Nemenman I, Basso K et al (2006) ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics* 7:S7
42. Scutari M (2010) Learning Bayesian networks with the bnlearn R package. *J Stat Softw* 35:1–22
43. Wolfender JL, Nuzillard JM, Van Der Hooft JJJ et al (2019) Accelerating metabolite identification in natural product research: toward an ideal combination of LC-HRMS/MS and NMR profiling, in silico databases and chemometrics. *Anal Chem* 91(1):704–742



Chapter 13

Feature-Based Molecular Networking for Metabolite Annotation

Vanessa V. Phelan

Abstract

The Global Natural Product Social Molecular Networking (GNPS) platform leverages tandem mass spectrometry (MS/MS) data for annotation of compounds. Molecular networks aid in the visualization of the chemical space within a metabolomics experiment. Recently, molecular networking has been combined with feature detection methods to yield Feature-Based Molecular Networking (FBMN). FBMN allows for the discrimination of isomers within the molecular network, incorporation of quantitative information generated by the feature detection tools into visualization of the molecular network, and compatibility with forthcoming *in silico* annotation tools. This chapter provides step-by-step methods for generating a molecular network to annotate microbial natural products using the Global Natural Product Social Molecular Networking (GNPS) Feature-Based Molecular Networking (FBMN) workflow.

Key words Molecular networking, Secondary metabolism, Feature annotation, Specialized metabolites, Natural Products, GNPS

1 Introduction

In liquid chromatography–mass spectrometry (LC-MS) based metabolomics workflows, annotation of a metabolite as an “identified metabolite” (confidence level 1) [1] typically requires comparing the accurate mass and retention time of a molecular feature to an authentic standard. While this approach has proven to be very robust in well-studied systems, it requires *a priori* knowledge of metabolites of interest in the biological sample as well as availability of authentic standards. In microbial metabolomics, specialized metabolites (also known as secondary metabolites, natural products, quorum sensing molecules, and small molecular virulence factors) are often of particular interest to the research community due to their roles in treating human disease, in contributing to infection, and as integral components of molecular communication within microbiome communities [2–4]. However, microbial specialized metabolite discovery is often hampered by the

rediscovery of characterized structures; most of these metabolites are not commercially available.

To address the limitation of molecular annotation in microbial metabolomics, Global Natural Products Social Molecular Networking (GNPS, <https://gnps.ucsd.edu>) was developed [5]. GNPS is an online bioinformatics platform for the analysis, sharing, and curation of tandem mass spectrometry (MS/MS) data. While GNPS was originally developed with microbial metabolites in mind, the tools within GNPS can be applied to any MS/MS metabolomics dataset.

The core principle of molecular networking at GNPS is the assumption that molecules of similar structure will fragment similarly during MS/MS data acquisition. Therefore, two structurally related molecules will likely have similar fragment ion spectra. To create a molecular network, all MS/MS spectra are aligned together to determine fragmentation similarity, and assembled into a molecular network. To aid in dereplication of molecules in the molecular network, each MS/MS spectrum of a metabolite of interest is scored based upon its spectral similarity to libraries of MS/MS spectra of known compounds. In practice, annotation (confidence level 2) [1] of MS/MS spectra of an experiment occurs in GNPS via automated dereplication (identification of knowns), whereby all MS/MS spectra collected during an experiment are scored against all MS/MS libraries (<https://gnps.ucsd.edu/ProteoSAFe/libraries.jsp>), including compound collections from the FDA and NIH, third party spectral libraries, and user-submitted annotations. If a user makes their data public via GNPS, that data will undergo continuous identification—the periodic and automatic reanalysis of the dataset against updated GNPS MS/MS libraries.

Further, the annotation of unknowns (not present in spectral libraries) can be made based on upon the molecular networks. Microorganisms often produce a variety of specialized metabolites from a single biosynthetic pathway. By leveraging molecular networking's grouping of similar MS/MS spectra of an entire dataset into structurally related molecular families of metabolites [6], rapid annotation (confidence level 3) [1] of analogues is possible. Molecular networks can be visualized using either the in-browser visualization or in Cytoscape [7].

While molecular networking has been broadly utilized in metabolomics, a general drawback of using the classic workflow is the MSCluster algorithm [8], which combines nearly identical MS/MS spectra to simplify visualization of a molecular network, does not take into account retention time. Therefore, the MS/MS spectra of isobaric species are combined into a single representative consensus MS/MS spectrum in classic molecular networking, despite the isobaric compounds separating chromatographically. Recently, feature finding programs have been integrated with molecular

networking [9] via the Feature-Based Molecular Networking (FBMN) workflow to correct this limitation.

Feature finding is a data reduction step in LC-MS analysis, whereby a feature is defined as a mass to charge (m/z) ratio, retention time pair and the corresponding abundance. Ultimately, processing LC-MS data using feature detection programs results in a quantitative representation of all metabolites within a metabolomics dataset. In FBMN, LC-MS/MS data is processed via feature finding, yielding abundances from MS data and a single representative MS/MS spectrum is selected per feature. This workflow allows users to import quantitative information derived from the feature detection tools into the molecular networks, discriminate isomers by retention time within the molecular network, and is compatible with in silico annotation of MS/MS spectra. Additionally, metadata such as bioactivity, media, and strain can be incorporated into the network to allow for rapid visual interpretation of the LC-MS/MS data. This approach allows for annotation of metabolites [1] based upon accurate mass and retention time of a molecular feature to an authentic standard (confidence level 1), a MS/MS spectral library (confidence level 2), and annotation of analogues of known compounds (confidence level 3).

Herein, we use raw LC-MS/MS data from two *Pseudomonas aeruginosa* PA14 strains, wild-type (WT) and a deletion mutant in the *rhlR* gene (*rhlR*) to illustrate the utility of FBMN. The *rhlR* gene is a key transcriptional regulator of the quorum sensing machinery of *P. aeruginosa* and regulates the production of a number of molecular families including the phenazines, quinolones, rhamnolipids, and siderophores [10–12]. FBMN provides a platform to rapidly annotate known microbial metabolites and their analogues as well as prioritizing features for further analysis by visualizing differential abundance via the molecular network.

2 Materials

2.1 Data

The raw data (Bruker .d file format and centroided mzXML file format) and the metadata table used for the step-by-step instructions below, a batch file containing the settings for processing the raw data in MZmine2 [13], and all resulting files from the MZmine2 and GNPS processing workflows can be downloaded free of charge at <https://massive.ucsd.edu> or accessed via the GNPS menu option “MassIVE Datasets” via accession number MSV000083500 (see Note 1).

2.2 mzMine 2

The current version of MZmine2 is available free of charge from the project web page <http://mzmine.github.io>.

2.3 GNPS

Academic, government, and not-for-profit researchers can create a free account on the GNPS website <https://gnps.ucsd.edu> and use the associated computational workflows free of charge.

2.4 Cytoscape

The current version of Cytoscape is available free of charge from the project web page <https://cytoscape.org>.

3 Methods**3.1 Create a Metadata File**

The metadata file provides the ability to incorporate sample property data into visualization and analysis of FBMN in Cytoscape.

The metadata format is a tab separated text file, that must be generated using a text editor (*see Note 2*).

In order for the metadata file to be processed properly in GNPS, the format of the file must be correct (Fig. 1). The first column must be titled “filename” in all lowercase. The file names in the “filename” column must be the full filename. The filenames should each be unique, identical to the names of the ones used for the feature finding process described in Subheading 3.2, and not contain any path information. The columns containing metadata must be prefixed by “ATTRIBUTE_” (*see Note 3*). The “ATTRIBUTE_” columns should contain simple descriptors. For example, the “ATTRIBUTE_Strain” column in Fig. 1 contains two descriptors called “GNPSGROUPS”: WT and rhlR, which denote the two strains of *P. aeruginosa* being compared.

3.2 Perform FBMN Compatible Feature Finding

1. Start MZmine2 (*see Note 4*). The MZmine2 settings used for the sample data are outlined in Table 1.
2. Import raw data files: Select the files to be analyzed under the menu option “Raw data methods → Raw data import” (*see Note 5*).
3. In MZmine2, a sequence of steps must be performed to correctly process the raw data for both feature finding and export to GNPS for molecular networking (*see Note 6*). The MZmine2 settings used for the sample data are outline in Table 1 (*see Note 7*).
 - (a) Mass detection:
 - Select all raw data files. Perform mass detection on MS level 1 under the menu option “Raw data methods → Mass detection → Set filter: MS level 1” (*see Note 8*).
 - Select all raw data files. Perform mass detection on MS level 2 under the menu option “Raw data methods → Mass detection → Set filter: MS level 2” (*see Note 9*).
 - (b) Build chromatograms: Select all raw data files. Perform building the chromatograms by choosing and applying

	A	B	C	D
1	filename	ATTRIBUTE_Strain		
2	9177.mzXML	rhlR		
3	9178.mzXML	WT		
4	9208.mzXML	rhlR		
5	9209.mzXML	WT		
6	9238.mzXML	rhlR		
7	9239.mzXML	WT		
8				

Fig. 1 Correct format for the metadata table file. The metadata file describes sample properties which will allow greater flexibility for analysis and visualization of the molecular network

appropriate settings (see Note 10) under the menu option “Raw data methods → Chromatogram builder.” New files will be created within the “Peak lists” window with the suffix *chromatograms*.

- (c) Deconvolute chromatograms: Select all peak lists with the suffix *chromatograms*. Perform chromatogram deconvolution by choosing and applying appropriate settings (see Note 11) under the menu option “Peak list methods → Peak detection → Chromatogram deconvolution.” New files will be created within the “Peak lists” window the suffix *chromatograms deconvoluted*.
- (d) Isotopic peaks grouper: Select all peak lists with the suffix *chromatograms deconvoluted*. Perform isotopic peaks grouper by choosing and applying appropriate settings (see Note 12) under the menu options “Peak list methods → Isotopes → Isotopic peaks grouper.” New files will be created within the “Peak lists” window the suffix *chromatograms deconvoluted deisotoped*.
- (e) Align features: Select all peak lists with the suffix *chromatograms deconvoluted deisotoped*. Perform feature alignment by choosing and applying appropriate settings under the menu option “Peak list methods → Alignment → Join Aligner.” A new file called *Aligned peak list* will be created within the “Peak lists” window.
- (f) Detect missing peaks: Select the peak list *Aligned peak list*. Perform gap-filling by choosing and applying appropriate settings (see Note 13) under the menu option “Peak list methods → Gap-filling → Peak finder (multithreaded).” A new file called *Aligned peak list gap-filled* will be created within the “Peak lists” window.
- (g) Filter for MS/MS peaks: Select the peak list *Aligned peak list gap-filled*. Perform peak list filtering for MS/MS by selecting *Keep all rows that match all criteria* under the

Table 1
MZmine2 parameter settings used for sample dataset

Parameter	Setting
<i>Mass detection (MS1)</i>	
Scans	MS level: 1; any polarity; any spectrum type
Mass detector	Centroid; noise level 1.0E3
Mass list name	Masses
<i>Mass detection (MS2)</i>	
Scans	MS level: 2; any polarity; any spectrum type
Mass detector	Centroid; noise level 1.0E2
Mass list name	Masses
<i>Chromatogram builder</i>	
Scans	MS level: 1; any polarity; any spectrum type
Mass list	Masses
Min time span (min)	0.05
Min height	3.0E3
<i>m/z</i> tolerance	0.01 <i>m/z</i> or 20.0 ppm
Suffix	Chromatograms
<i>Chromatogram deconvolution</i>	
Suffix	Deconvoluted
Algorithm	Baseline cut-off; min peak height 1.0E3; peak duration range (min) 0.00–2.00; baseline level 5.0E3
<i>m/z</i> center calculation	MEDIAN
<i>Isotopic peaks grouper</i>	
Name suffix	Deisotoped
<i>m/z</i> tolerance	0.01 <i>m/z</i> or 20.0 ppm
Retention time tolerance	0.25 absolute (min)
Monotonic shape	Checked
Maximum charge	4
Representative isotope	Most intense
<i>Join aligner</i>	
Peak list name	Aligned peak list
<i>m/z</i> tolerance	0.01 <i>m/z</i> or 20.0 ppm
Weight for <i>m/z</i>	0.8
Retention time tolerance	0.5 absolute (min)

(continued)

Table 1
(continued)

Parameter	Setting
Weight for RT	0.2
Require same charge state	Checked
<i>Peak finder (multithreaded)</i>	
Name suffix	Gap-filled
Intensity tolerance	10.0%
<i>m/z</i> tolerance	0.01 <i>m/z</i> or 20.0 ppm
Retention time tolerance	0.5 absolute (min)
<i>Peak list rows filter</i>	
Name suffix	Filtered
Minimum peaks in a row	3
Peak duration range	Checked 0.00–2.0
Keep or remove rows	Keeps rows that match all criteria
Keep only peaks with MS2 scan (GNPS)	Checked
Reset the peak number ID	Checked
<i>Duplicate peak filter</i>	
Name suffix	Filtered
Filter mode	OLD AVERAGE
<i>m/z</i> tolerance	0.005 <i>m/z</i> or 10.0 ppm
RT tolerance	0.5 absolute (min)
<i>Export for/submit to GNPS</i>	
Filename	FBMN example
Mass list	Masses
Filter rows	ONLY WITH MS2

“Keep or remove rows” option and checking the box for “Keep only peaks with MS2 scans (GNPS)” (*see Note 14*) under “Peak list methods → Filtering → Peaklist row filter.” A new file called “Aligned peak list gap-filled filtered” will be created within the “Peak lists” window.

- (h) Filter for duplicate peaks: Select the peak list called *Aligned peak list gap-filled filtered*. Perform duplicate peak rows filtering by choosing and applying appropriate settings (*see Note 15*) under the menu option “Peak list

methods → Filtering → Duplicate peak filters.” A new file called *Aligned peak list gap-filled filtered* will be created within the “Peak lists” window.

- Export files for GNPS: Select the peak list called *Aligned peak list gap-filled filtered*. Perform export of files under the menu option “Peak list methods → Export for/Submit to GNPS.” Under the “file-name” option in the export window, provide a name for the files that will be created in the chosen folder. Under “mass list” choose *masses* and select *ONLY WITH MS2* under “Filter rows.” After pressing OK, navigate to the export folder. Within this folder, two files will be created: an mgf file which will be used for molecular networking and the associated Peak Area Quanitification Table (a csv file labeled with “quant” which contains the MS1 peak area abundance information for each sample).

3.3 Perform FBMN

1. Navigate to the GNPS website (<https://gnps.ucsd.edu>) and sign in using your user name and password.
2. Navigate to the homepage (<https://gnps.ucsd.edu/ProteoSAFe/static/gnps-splash.jsp>) and scroll down to “Advanced Analysis Tools” option. Under “Feature Networking,” choose “Analyze.” The analysis web page specific to the FBMN workflow will open.
3. Set up and run the GNPS FBMN workflow. The GNPS FBMN settings used for the sample data are outlined in Table 2 (see Note 16).
 - (a) Workflow Selection: Provide a descriptive title (see Note 17).
 - (b) File Selection: Upload files to GNPS by clicking “Select Input Files.” The “Select Input Files” interface will open. Navigate to the “Upload Files” pane. Upload the *mgf* file, the *Peak Area Quantification Table* csv file, and the *Sample Metadata Table* created in the previous steps by dragging the files to the “File Drag and Drop” area (see Note 18). After upload of the files to the appropriate folder, navigate to the “Select Input Files” pane. Locate the files you want to analyze. Select the *mgf* file and click the “MS2 mgf” button. The *mgf* file will now be listed under “Selected MS2 MGF file” in the “Selected Files” pane. Follow the same procedure for the *Peak Area Quantification Table* and the *Sample Metadata Table*. *Speclibs* will be automatically listed under the “Selected Library Files” section of the “Selected Files” pane (see Note 19). Close the window by clicking “Finish Selection.”

Table 2
GNPS FBMN workflow parameter settings used for sample dataset

Parameter	Setting
<i>Workflow selection</i>	
Title	FBMN example PA14 rhlR cosine 0.70
<i>File selection</i>	
MS2 MGF file	FBMN example.mgf
Peak area quantification table	FBMN example_quant.csv
Sample metadata table	FBMN example metadata.txt
<i>Basic options</i>	
Quantification table source	MZmine2
Precursor ion mass tolerance	0.05 Da
Fragment ion mass tolerance	0.1 Da
<i>Advanced network options</i>	
Min pairs Cos	0.70
Network TopK	10
Minimum matched fragment ions	4
Maximum connected component size	100
Maximum shift between precursors	500 Da
<i>Advanced library search options</i>	
Library search min matched peaks	4
Search analogs	Don't search
Top results to report per query	1
Score threshold	0.7
Maximum analog search mass difference	100.0 Da (default value)
<i>Advanced filtering options</i>	
Minimum peak intensity	0.0
Filter precursor window	Filter
Filter peaks in 50 Da window	Filter
Filter library	Filter
<i>Advanced quantification options</i>	
Normalization per file	No norm
Aggregation method for peak abundances per group	Mean
<i>Advanced external tools</i>	

(continued)

Table 2
(continued)

Parameter	Setting
Run dereplicator	Don't run
<i>Advanced extras</i>	
Supplementary pairs	

- (c) Basic Options: Choose *MZmine2* as the “Quantification Table Source.” Change “Precursor Ion Mass Tolerance” and “Fragment Ion Mass Tolerance” settings to a value that is appropriate for the data (*see Note 20*).
 - (d) Workflow Submission: Verify that your e-mail address is correct. Click “Submit.”
4. When the FBMN analysis job is completed, navigate to the menu item “Jobs” on GNPS and click the DONE status. Under “Advanced Views—Experimental Views,” Click “Direct Cytoscape Preview/Download.” After the GNPS Cytoscape Downloader finishes loading, click “Download Cytoscape File” (*see Note 21*).

3.4 Visualize the FBMN in Cytoscape

1. Locate and open the Cytoscape file downloaded from GNPS. In the resulting molecular network, each node represents the MS/MS spectrum from an MS1 feature (*see Note 22*). All metadata associated with the network is shown in the “Table Panel,” including the “GNPSGROUPS” descriptors of the “ATTRIBUTE_Strain” column of the metadata table. Additionally, the “Table Panel” lists information about matches to the GNPS libraries.
2. Different parameters can be set to enhance the visualization of data features within the molecular network by modifying the settings in the “Node,” “Edge,” or “Network” tabs within the “Style” tab of the “Control Panel” window. The default settings for the automatically generated Cytoscape file are *node color*: blue; *edge color*: white; *background color*: gray; *node label*: Compound name (*see Note 23*). To recapitulate the molecular network in Fig. 2a, apply the following steps:
 - (a) To highlight the connectivity similarity between nodes, load cosine settings. To do this, select the “Edge” tab within the “Style” tab of the “Control Panel.” For the “Width” setting, select *cosine* for the “Column” and *Continuous Mapping* for the “Mapping Type.” Double-click the “Current Mapping” box and set the desired minimum (0) and maximum (30) values. Within the molecular network, the thicker edges indicate a stronger relationship between the MS2 spectra represented by the nodes.

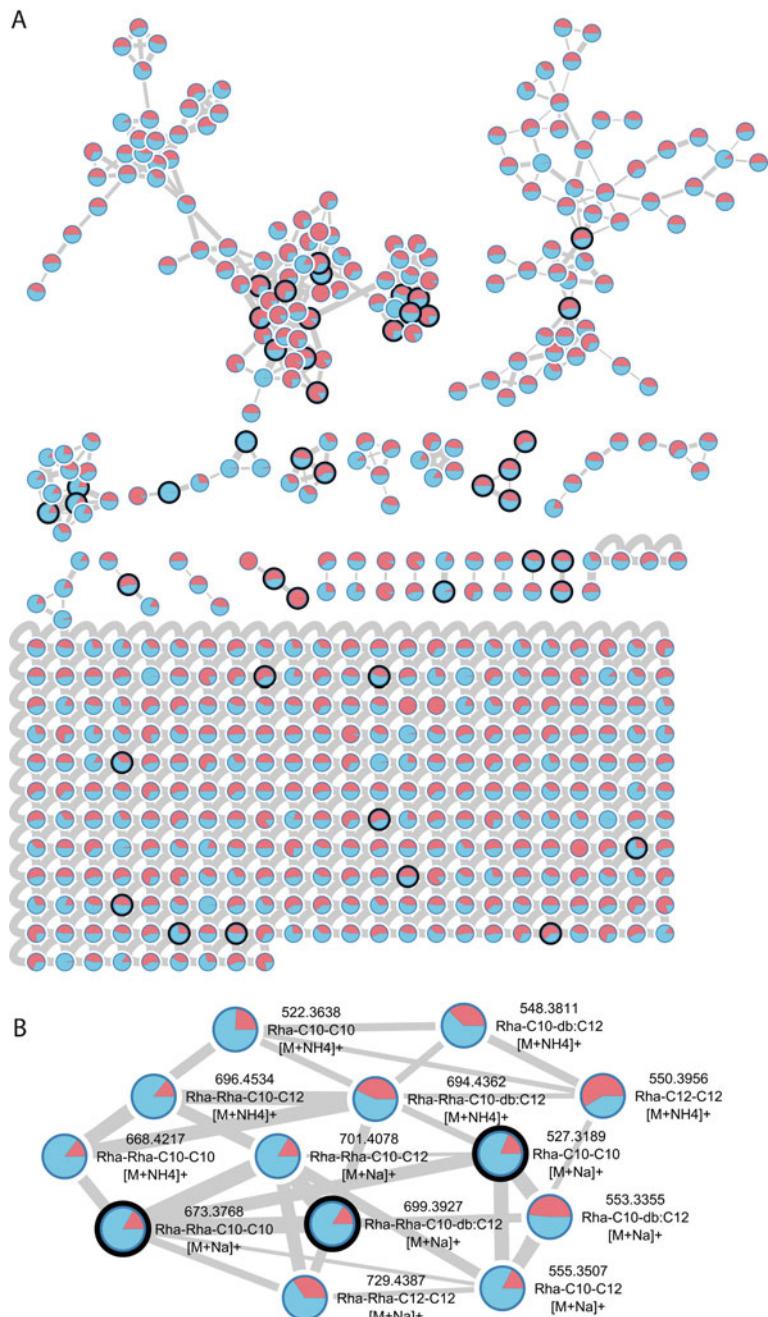


Fig. 2 Feature-based molecular network. **(a)** The total feature-based molecular network. Each feature is represented by a node. The pie chart indicates the abundance of each feature in the different GNPSGROUPS, rhIR (pink) and WT (blue). Nodes that have MS/MS matches to the GNPS libraries are outlined in black. **(b)** The rhamnolipid molecular family. Three features had MS/MS matches to the GNPS libraries: m/z 527.3189 (Rha-C10-C10), m/z 673.3768 (Rha-Rha-C10-C10), and m/z 699.3927 (Rha-Rha-C10-db:C12). As molecules with similar MS/MS spectra will cluster into a molecular family, annotation of other nodes can be propagated through the molecular family based upon their difference in m/z values to library matches

- (b) To visualize the precursor mass on each node, select the “Node” tab. For the “Label” setting, select *precursor mass* for the “Column” and *Passthrough Mapping* for the “Mapping Type.”
- (c) To visualize which nodes have corresponding MS2 spectral matches to the GNPS libraries, select the “Node” tab. For the “Border Paint” setting, select *Compound_Name* for the “Column” and *Discrete Mapping* for the “Mapping Type.” Select the boxes with compound names, right click, select edit, then edit Select Discrete Mapping Values and choose black.
- (d) To visualize the group average abundances, pie charts are used. Select the “Node” tab. For the “Image/Chart 1” setting, click on the Def. box. In the “Data” tab, remove the default columns from the “Selected Columns” panel and add the appropriate columns (GNPSGROUP:rhlR and GNPSGROUP:WT for the sample data) from the “Available Columns” panel. These two columns will now be listed within the “Selected Column” panel. Under the “Options” tab, select the desired color for the two columns—pink for 1 (GNPSGROUP:rhlR) and blue for 2 (GNPSGROUP:WT). Click “Apply” (*see Note 24*).

3.5 Use the FBMN to Perform Initial Analysis

1. The visualization of experimental data can be used to guide secondary metabolite annotation and analysis.
 - (a) Differential abundance (*see Note 25*): The pie chart visualization can be used to quickly identify features that have differential abundance between sample groups (Fig. 2). The “cluster index” column of the “Node Table” panel of the “Table Panel” in Cytoscape corresponds to the feature number in the Peak Area Quanitification Table generated by the MZmine2 workflow. Therefore, the abundance differences displayed by the pie chart in the molecular network can be validated simply by finding the feature number in the Peak Area Quanitification Table for each sample and averaging the abundance values for each group (*see Note 26*).
 - (b) Annotation: In the molecular network, all nodes with matches to the GNPS libraries are indicated by a black border. Click on a node with a black border. The annotation of that node will be listed within the “Compound_Name” column of the “Node Table” panel of the “Table Panel.”
 - (c) Propagating an annotation: Locate the node annotated “Rhamnolipid Rha-C10-C10” (*m/z* 527.3197). As molecular networking compares the similarity of

MS/MS spectra, the nodes connected to “Rhamnolipid Rha-C10-C10” form a molecular family of structurally related molecules based on the MS/MS spectra. Therefore, the MS1 mass difference between nodes and the raw MS/MS spectra (*see Note 26*) can be leveraged to annotate (confidence level 3) the rest of the nodes within the molecular family (Fig. 2b).

4 Notes

1. The step-by-step instructions for the analysis described was performed using MZmine2 on the Windows platform. Therefore, adjustments may be required if a different computing platform or feature finding program is used. For information regarding how to convert proprietary raw MS data to open source formats, consult the instrument vendor.
2. Tab separated text files can be generated using text editors including Microsoft Excel, Notepad++ (Windows), gedit (Linux), TextWrangler (Mac OS).
3. Metadata in metabolomics studies is important for defining the experimental methodology as well as facilitating reproducibility. In the context of FBMN, metadata can be used to differentially label or color components of the molecular network. Therefore, it is important to consider what scientific questions you would like to investigate using FBMN. Relevant metadata for FBMN includes, but is not limited to, sample type, biological source, genotype, treatment, geolocation, sample preparation, bioactivity measurements, and so on.
4. Information about the compatibility of FBMN with other feature finding programs can be found on the GNPS documentation website (<https://ccms-ucsd.github.io/GNPSDocumentation/featurebasedmolecularnetworking/>).
5. Raw data import file compatibility requirements for MZmine2 can be found in the HELP menus. The sample data was collected on a Bruker Maxis HD, lock mass was applied, and the data was converted from .d to centroided .mzXML using Bruker CompassXport.
6. MZmine2 has extensive documentation under HELP menus describing the options available for data processing. There are many more options available in MZmine2 than those illustrated in the step by step instructions. While each setting can be set individually, if you anticipate analyzing many similar datasets, a batch .xml file can be saved, loaded, and configured for multiple uses.

7. The MZmine2 settings for FBMN need to be optimized for each dataset. The settings appropriate for the example data may not be appropriate for your data.
8. Set an appropriate intensity threshold for MS1 feature finding by clicking the “...” box next to the mass detector option. You can use the “Show Preview” window to assess the effect of your intensity threshold on your data. Minimally, the value should correspond the minimum value set for triggering an MS2 scan event used when setting up your MS2 data collection.
9. Set an appropriate intensity threshold for MS2 feature finding by clicking the “...” box next to the mass detector option. You can use the “Show Preview” window to assess the effect of your intensity threshold on your data. This threshold should be set to the representative noise level in the MS2 data. If this threshold is set too high, export for GNPS processing will be negatively impacted.
10. Choose settings appropriate for the LC separation and mass accuracy of your experiment.
11. While not used in this example, *m/z* range for MS2 scan pairing (Da) and RT range for MS2 scan pairing (min) can be enabled to more precisely match an MS2 scan with an MS1 feature.
12. The settings chosen for Isotope peaks grouper are dependent upon the chromatographic peak shape, duty cycle time of the mass spectrometer, and the MS1 mass accuracy.
13. Gap-filling is a critical step in MZmine2 feature finding. Following alignment, the resulting peak list may contain missing peaks due to deficient peak detection or a mistake in alignment of different peak lists. A missing peak does not imply that the peak does not exist. In most cases, a missing peak indicates that it went undetected in previous steps. To verify that the feature finding is correct, the peak list table can be visualized by right-clicking on the “aligned peak list gap-filled” and choosing “Show peak list table.” It is important to always verify the peak list by validating randomly selected features in the raw data. Peak finding (multithreaded) allows the processing to be performed in parallel instead of sequentially.
14. There are many different filtering options available in MZmine2. For further details, access the HELP information.
15. Removal of duplicate peaks ensures that duplicate features generated during the previous processing steps are removed. This filtering step can be performed before applying the Peaks rows filter, if desired.
16. The settings for the FBMN GNPS workflow must be optimized for each individual dataset. For more information about the various settings, consult the GNPS documentation web page.

17. Using a descriptive title that encompasses the setting used for FBMN is extremely useful. One of the advantages of the GNPS platform is that a user can perform iterative analyses of the same data set concurrently in order to identify the most optimal settings for the analysis of their dataset.
18. If the files are too large for the Drag and Drop feature of the Select Input Files interface, the files can be uploaded to GNPS via an ftp server. Directions to perform file upload via ftp is available within the GNPS documentation.
19. GNPS annotates MS/MS spectra by comparing each spectrum to MS/MS libraries. The list of available MS/MS libraries can be accessed via the GNPS website.
20. The appropriate “Precursor Ion Mass Tolerance” and “Fragment Ion Mass Tolerance” settings are dictated by the mass accuracy and mass resolution of the mass spectrometer used for MS/MS data collection. In addition to Basic Options, the FBMN workflow can be adjusted by modifying the various Advanced Options. Learn more about these settings on the GNPS documentation pages.
21. In addition to visualizing the molecular network in Cytoscape, the network can be viewed and analyzed via the in-browser visualization. For further information about the different types of analyses that can be performed in GNPS, access the GNPS documentation.
22. It is important to note that during the feature finding processing in MZmine2, only the features that have a corresponding MS/MS spectrum are retained for molecular networking analysis. If a feature does not have an associated MS/MS spectrum, it will not be in the molecular network. Whether a feature has a corresponding MS/MS spectrum is dependent upon the settings used during data acquisition as well as during data processing in MZmine2.
23. Tutorials for how to use Cytoscape, including a manual and videos, are available on the Cytoscape website.
24. If the visualization adjustments of the molecular network in Cytoscape do not automatically change, they can be shown by utilizing the “Show Graphic Details” option within the “View” menu.
25. Cytoscape has many different settings that can be used to differentiate groups. Node size, shape, color, etc. can be adjusted to reflected different attributes of the samples such as origin, and bioactivity.
26. FBMN is a tool to aid in the analysis of MS data. The settings used in the FBMN workflow, including those for feature finding and generating the molecular network, will influence the final molecular network output. It is very important to validate the observations from FBMN with the raw data.

Acknowledgments

We thank L. Dietrich (Columbia University) for kindly providing the *P. aeruginosa* strains and M. Wang (University of California, San Diego) for providing feedback on the manuscript. Our work on microbial metabolomics has been supported by the National Institutes of Health (National Institute of General Medical Sciences grants K01 GM103809 and R35 GM128690), the ALSAM Foundation (L.S. Skaggs Professorship and Therapeutical Innovation Award), and the University of Colorado.

References

1. Sumner LW, Amberg A, Barrett D, Beale MH, Beger R, Daykin CA, Fan TW, Fiehn O, Goodacre R, Griffin JL, Hankemeier T, Hardy N, Harnly J, Higashi R, Kopka J, Lane AN, Lindon JC, Marriott P, Nicholls AW, Reily MD, Thaden JJ, Viant MR (2007) Proposed minimum reporting standards for chemical analysis Chemical Analysis Working Group (CAWG) Metabolomics Standards Initiative (MSI). *Metabolomics* 3(3):211–221. <https://doi.org/10.1007/s11306-007-0082-2>
2. Garg N, Luzzatto-Knaan T, Melnik AV, Caraballo-Rodriguez AM, Floros DJ, Petras D, Gregor R, Dorrestein PC, Phelan VV (2017) Natural products as mediators of disease. *Nat Prod Rep* 34(2):194–219. <https://doi.org/10.1039/c6np00063k>
3. Newman DJ, Cragg GM (2016) Natural products as sources of new drugs from 1981 to 2014. *J Nat Prod* 79(3):629–661. <https://doi.org/10.1021/acs.jnatprod.5b01055>
4. Phelan VV, Liu WT, Pogliano K, Dorrestein PC (2011) Microbial metabolic exchange—the chemotype-to-phenotype link. *Nat Chem Biol* 8(1):26–35. <https://doi.org/10.1038/nchembio.739>
5. Wang M, Carver JJ, Phelan VV, Sanchez LM, Garg N, Peng Y, Nguyen DD, Watrous J, Kapono CA, Luzzatto-Knaan T, Porto C, Bouslimani A, Melnik AV, Meehan MJ, Liu WT, Crusemann M, Boudreau PD, Esquenazi E, Sandoval-Calderon M, Kersten RD, Pace LA, Quinn RA, Duncan KR, Hsu CC, Floros DJ, Gavilan RG, Kleigrewe K, Northen T, Dutton RJ, Parrot D, Carlson EE, Aigle B, Michelsen CF, Jelsbak L, Sohlenkamp C, Pevzner P, Edlund A, McLean J, Piel J, Murphy BT, Gerwick L, Liaw CC, Yang YL, Humpf HU, Maansson M, Keyzers RA, Sims AC, Johnson AR, Sidebottom AM, Sedio BE, Klitgaard A, Larson CB, CAB P, Torres-Mendoza D, Gonzalez DJ, Silva DB, Marques LM, Demarque DP, Pociute E, O'Neill EC, Briand E, Helfrich EJN, Granatosky EA, Glukhov E, Ryffel F, Houson H, Mohimani H, Kharbush JJ, Zeng Y, Vorholt JA, Kurita KL, Charusanti P, McPhail KL, Nielsen KF, Vuong L, Elfeki M, Traxler MF, Engene N, Koyama N, Vining OB, Baric R, Silva RR, Mascuch SJ, Tomasi S, Jenkins S, Macherla V, Hoffman T, Agarwal V, Williams PG, Dai J, Neupane R, Gurr J, Rodriguez AMC, Lamsa A, Zhang C, Dorrestein K, Duggan BM, Almaliti J, Allard PM, Phapale P, Nothias LF, Alexandrov T, Litaudon M, Wolfender JL, Kyle JE, Metz TO, Peryea T, Nguyen DT, VanLeer D, Shinn P, Jadhav A, Muller R, Waters KM, Shi W, Liu X, Zhang L, Knight R, Jensen PR, Palsson BO, Pogliano K, Linington RG, Gutierrez M, Lopes NP, Gerwick WH, Moore BS, Dorrestein PC, Bandeira N (2016) Sharing and community curation of mass spectrometry data with global natural products social molecular networking. *Nat Biotechnol* 34(8):828–837. <https://doi.org/10.1038/nbt.3597>
6. Nguyen DD, Wu CH, Moree WJ, Lamsa A, Medema MH, Zhao X, Gavilan RG, Aparicio M, Atencio L, Jackson C, Ballesteros J, Sanchez J, Watrous JD, Phelan VV, van de Wiel C, Kersten RD, Mehnaz S, De Mot R, Shank EA, Charusanti P, Nagarajan H, Duggan BM, Moore BS, Bandeira N, Palsson BO, Pogliano K, Gutierrez M, Dorrestein PC (2013) MS/MS networking guided analysis of molecule and gene cluster families. *Proc Natl Acad Sci U S A* 110(28):E2611–E2620. <https://doi.org/10.1073/pnas.1303471110>
7. Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D, Amin N, Schwikowski B, Ideker T (2003) Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome*

- Res 13(11):2498–2504. <https://doi.org/10.1101/gr.1239303>
8. Frank AM, Bandeira N, Shen Z, Tanner S, Briggs SP, Smith RD, Pevzner PA (2008) Clustering millions of tandem mass spectra. J Proteome Res 7(1):113–122. <https://doi.org/10.1021/pr070361e>
9. Olivon F, Grelier G, Roussi F, Litaudon M, Touboul D (2017) MZmine 2 data-preprocessing to enhance molecular networking reliability. Anal Chem 89(15):7836–7840. <https://doi.org/10.1021/acs.analchem.7b01563>
10. Schuster M, Greenberg EP (2007) Early activation of quorum sensing in *Pseudomonas aeruginosa* reveals the architecture of a complex regulon. BMC Genomics 8:287. <https://doi.org/10.1186/1471-2164-8-287>
11. Whiteley M, Lee KM, Greenberg EP (1999) Identification of genes controlled by quorum sensing in *Pseudomonas aeruginosa*. Proc Natl Acad Sci U S A 96(24):13904–13909
12. Winson MK, Camara M, Latifi A, Foglino M, Chhabra SR, Daykin M, Bally M, Chapon V, Salmond GPC, Bycroft BW, Lazdunski A, Stewart GSAB, Williams P (1995) Multiple N-acyl-L-homoserine lactone signal molecules regulate production of virulence determinants and secondary metabolites in *Pseudomonas aeruginosa*. Proc Natl Acad Sci USA 92(20):9427–9431. <https://doi.org/10.1073/pnas.92.20.9427>
13. Pluskal T, Castillo S, Villar-Briones A, Oresic M (2010) MZmine 2: modular framework for processing, visualizing, and analyzing mass spectrometry-based molecular profile data. BMC Bioinformatics 11:395. <https://doi.org/10.1186/1471-2105-11-395>



Chapter 14

A Bioinformatics Primer to Data Science, with Examples for Metabolomics

W. Stephen Pittard, Cecilia “Keeko” Villaveces, and Shuzhao Li

Abstract

With the increasing importance of big data in biomedicine, skills in data science are a foundation for the individual career development and for the progress of science. This chapter is a practical guide to working with high-throughput biomedical data. It covers how to understand and set up the computing environment, to start a research project with proper and effective data management, and to perform common bioinformatics tasks such as data wrangling, quality control, statistical analysis, and visualization, with examples on metabolomics data. Concepts and tools related to coding and scripting are discussed. Version control, knitr and Jupyter notebooks are important to project management, collaboration, and research reproducibility. Overall, this chapter describes a core set of skills to work in bioinformatics, and can serve as a reference text at the level of a graduate course and interfacing with data science.

Key words Bioinformatics, Metabolomics, Data science, Quality control, Data management, Cloud computing, Scripting, Data visualization

1 Introduction

The field of bioinformatics received the first major boost by genome sequencing [1]. Over the past three decades, bioinformatics became infused with many disciplines. At times, it is difficult to recognize bioinformatics as a single research field, because of its coverage of broad topics, from molecular evolution, structure of macromolecules, drug screening, and machine learning to network models. When commemorating the 50th anniversary of the DNA helix, Walter Gilbert said that “[it] seems to me that molecular biology is dead. DNA-based thinking has penetrated the whole of biology, and the separate field no longer exists” [2]. The same thing can be said of bioinformatics today—biology has become an information science, where larger and larger volumes of data lead the new waves of innovation. Bioinformatics has penetrated biology. The most exciting bioinformatics research has been combined with

particular scientific domains, which evolved along genomics, epigenomics, proteomics, microbiome, and now metabolomics.

The goal of this chapter is to describe a core set of skills to do bioinformatics, at the level of a graduate course and interfacing with data science. This is written as an introductory reference of working with high throughput biomedical data, covering data management, scripting, version control, data wrangling, quality control, statistical analysis and visualization. We will use metabolomics data as examples, but the principles are rather generic. An in-depth guide of the essential tools, including Python and R programming environments, Git for version control, and Docker for virtualization, is provided in Chapter 15.

2 Computing Environment and Setup

Selecting the most appropriate computational environment is an important decision for researchers involved in even modest forms of data analysis. The choice will be a function of several things including the type and size of data, whether collaboration is involved, as well as the desired outcome (e.g., a software product or package in support of a publication). In response to the proliferation of omics data, there has been a corresponding development of proprietary and open source tools and frameworks designed to ease the management and comprehension of this information. This is advantageous since researchers and analysts do not have to develop major software tools to accomplish in-depth data transformation and analysis. However, they do have to invest time in understanding how to deploy such tools to leverage their full potential. In particular, combining disparate tools into pipelines, while selecting appropriate options for each of the component programs, can require significant experimentation when seeking to optimize for the data to generate or test project hypotheses.

It is also possible that a project might involve the use of a number of tools, languages, and operating systems which in turn requires the familiarity with different platform's exchange formats, and user interfaces. This can be mitigated somewhat by the availability of web-based portals that insulate the user from the underlying details of data management and analysis while providing the benefit of near immediate access to results that can be examined using, for example, a genome browser or alignment tools common to bioinformatics workflows. Tools such as BLAST, Galaxy, and the UCSC Genome browser fall into this category. These frameworks are also available for installation on local computers upon which the user can install as much or as little of the toolset as desired while retaining the option of using the public versions of these tools.

In this case the researcher, or someone acting on their behalf, must manage the technical aspects of installation, deployment, and

training for the tool. For purposes of classroom education, use of Web services and GUIs might be preferable as it allows the students to focus on the science behind a given tool without the added distraction of having to understand the underlying computational environment. However, for researchers and analysts, being able to “look under the hood” is important since it permits a nuanced use of the tool that is necessary or more productive. Moreover, being able to author “scripts” and programs to obtain, manipulate, and visualize data is useful to accelerate the pace of one’s research.

When considering the above, it becomes apparent that some degree of self-sufficiency and familiarity with informatics tools is useful if not necessary to accelerate the pace of research. Publication agreements can also require the data and code made available. The creation of digital assets that can be easily tracked greatly enhances the reproducibility of published research, and it usually necessitates the involvement of informatics aware personnel. Even before the publication step, it is helpful for a researcher to have facility with data management approaches and visualization packages essential to data exploration and the formation of hypotheses. In this case, being productive with the “command line” interface becomes important as it provides greater flexibility than is customarily available in form-based web tools. It also enables the creation of “glue” code to wrap, supplement, or replace existing software tools.

2.1 Desktops, Laptops, Clusters, and the Cloud

Analysis can be accomplished in a number of environments depending on the scope and complexity of the project. Modern computing hardware, even at the laptop level, provides sufficient memory, multicore CPUs, and disk space to accomplish many data transformation and analysis activities, including the use of parallel libraries (depending on the language). Desktop computers can provide even greater flexibility and expandable components, which can also accommodate virtual servers designed to host one or more operating systems should there be a need for such. Cheap hardware, in combination with comparably priced networking equipment, motivated the formation of Beowulf clusters in the late 1990s and early 2000s which allowed researchers to leverage the collective power of various computers by presenting them as an integrated cluster. Users submit “jobs” into a work “queue” and wait for results according to an established priority. While the Beowulf model is still in use, it is more common to experience formalized clusters managed by knowledgeable personnel responsible for the continued operation of the resource and, optionally, assist users with questions. The advantage for the researcher is that the infrastructure is managed on their behalf, there is a single point of entry, and the resource can generally be expanded without impacting existing work.

A significant evolution of this approach was introduced in 2006 when Amazon launched Amazon Web Services, which provided scalable Infrastructure as a Service (IaaS) allowing knowledgeable

users the ability to provision and organize compute “instances,” storage, and networking on-demand. This approach provides flexibility in the architecture of environments customized to the needs of a specific project. The service is available as a “pay what you use” model, while significantly discounted reserved resources are available. Google and Microsoft followed suit and offer competing resources at a similar level and cost. All of these vendors provide additional services such as machine learning and deep learning, database management, data streaming which can be arranged and connected in novel ways which implies that fully leveraging these services requires dedicated skills and training. These vendors, Amazon, Microsoft, and Google are colloquially known as “Cloud Computing” providers in reference to the idea that all hardware, networking, and storage are managed off-premise as well in reference to the idea that the at-scale capabilities on offer are seemingly infinite.

2.2 Operating Systems for Bioinformatics

A key aspect of a computational environment is the availability of software resources and libraries. Complex dependencies between these resources and libraries may determine the cost and feasibility of a project, even if new software development is not involved. Part of the problem is that they can be dependent on a particular operating system. Furthermore, how one interacts with the operating system directly impact one’s productivity. Three major operating systems are widely used today: Microsoft Windows, Apple Mac OS, and Linux, with many people being more familiar with the first two. Microsoft Windows is the most prevalent operating system for computing at the consumer and corporate levels wherein office productivity is a priority. Thus, the environment is developed primarily to support spreadsheet use, email, web browsing, and business applications. Nonetheless, it is still possible to use Microsoft Windows as a host operating system for computational research especially with the addition of Windows 10 “power shell” which offers a “shell/interpreter” tool for those accustomed to command line development within a UNIX environment. Microsoft, like Apple Mac OS, is a proprietary system the cost of which is typically embedded within the purchase price of a computer although some organizations have established licensing agreements with Microsoft to offer discounted or subsidized licenses. Many open source tools are available for both Windows and Mac OS.

Apple Mac OS is also a popular proprietary operating system (including versions of Mac OS, OS X, macOS) which is tightly integrated with Apple hardware to provide a unified, stable environment in accordance with an established set of user interface design considerations. The Mac OS operating system is based upon a variant of UNIX called BSD (Berkeley Software Distribution). For the end user this means that a form of UNIX is accessible

on any Apple hardware running OS. It is interesting to note that the GUI (the “Finder”) is a process that runs on top of UNIX. For those interested in command line access, it can be obtained via the Terminal application. Apple Mac OS also offers a development interface called Xcode which is primarily for those interested in creating applications for Mac OS and iOS, while it is used to manage projects in various languages. For those interested in treating their Apple Mac hardware primarily as a UNIX box, there is the Homebrew project (<https://brew.sh/>), which enables the management and installation of common UNIX packages such as the GNU suite of tools (gcc, make, etc.). The advantage of this approach is that one can continue to use the conveniences of the Apple Mac GUI while having full access to a UNIX-based development and shell scripting environment.

Virtualization technologies have been commonly used to serve multiple operating systems on the same hardware, and to support cloud computing. These include the two main categories of virtual machines and containers. These technologies help modularize computing and make the deployment of software applications more compatible and efficient. The Docker containers are described in details in Chapter 15.

2.3 UNIX/Linux

The terms UNIX and Linux are often used interchangeably, but technically that is not accurate. The term UNIX originally applied to a proprietary operating system developed out of Bell Labs in the early 1970s, which was originally intended for internal use. As the system grew in utility and popularity, licenses were offered to Berkeley University for further development, which resulted in BSD, the first major branch of the ATT code base. Organizations such as IBM and HP developed their own “flavors” of UNIX, some based on ATT or BSD, and offered these distributions under various business models. The GNU project was created by Richard Stallman and became the leader of the free software movement. The year of 1991 saw the release of the first Linux kernel by Linus Torvalds. Subsequently, Linux became the dominant system to run web servers, and led to the boom of an open source software ecosystem. The name Linux is commonly employed generically though there are a number of specific distributions (abbreviated as “distros”) such as Ubuntu, Debian, GenToo, and Fedora as well as commercially supported distros such as Redhat and SUSE.

While Linux is freely available, vendors can offer “for cost” support models to help individuals and organizations become productive. The organizations behind each of the respective distros can offer enthusiastic support freely simply to encourage adoption of those particular distributions. As an example, the Ubuntu distribution has a large number of “baked in” tools for the processing of genomic data which is appealing to researchers within that domain.

Linux has become the standard for high performance computation and is the foundation of major cloud computing vendors.

2.4 GUI vs. the Command Line

The convenience and intuitive nature of GUI (graphical user interface) is attractive as it simplifies the management of files, folders, and computer settings by offering a visual interface to perform these common actions without requiring in depth training or orientation as a prerequisite to productivity. Moreover, there are a number of desktop applications written specifically to leverage the capabilities of these operating systems which can further simplify the manipulation of data. For example, there are commercial packages for the analysis of genomic data that run natively on both Windows and Mac OS. However, in order for a GUI to be available, the exact task has to be defined and coded into the GUI design. This places a lot of limit on what one can do within a GUI. A typical task in bioinformatics is converting data formats, which relies mostly on the use of text manipulation package such as **sed** or **awk** or the creation of a script in **Python** or **R** to first preprocess the data. Without such knowledge the researcher will be at a disadvantage and reliant upon others to accomplish what is a basic yet essential task.

The phrase “command line” refers to the use of typed commands at a **shell** (e.g., Bash, ksh) prompt that in turn invokes specific programs available within a given computational environment. Microsoft Windows has a command prompt. Shell is native to Linux and Mac OS, which runs on top of a variant of the UNIX operating system. Bioinformatics often involves command lines to execute and connect different programs. This creates the capability to perform and automate complex tasks.

2.5 The Shell

Learning how to interact with the UNIX operating system does require a commitment on the user’s part, though a facility with it can be acquired in stages over time and in response to the demands of a given project. The basic components involve the use of a SHELL which is the interpreter for any command a user might enter at the command line prompt. Think of a shell as an interpreter of interactive input although scripts containing an arbitrary number of shell commands, including programming constructs, can be developed for execution within the shell context. As an example, it is possible to create a number of shell scripts written in bash for the maintenance of data over time. These are typically designed and implemented for system administration tasks, though Linux offers powerful text manipulation tools which can be helpful in the management of scientific data. However, many researchers choose to develop scripts in Python or R, with the benefit of packages written specifically to work with high throughput data. Figure 1 is an example of using the shell on Apple Mac OS. It is outside the

```
$ pwd
/Users/esteban/MyFiles
$ ls
logistic.R          prinicpal_components_analysis.R
notebook.py
$
$ echo $SHELL
/bin/bash
$
```

Fig. 1 Screenshot of BASH example on Mac OSX

scope of this text to provide a detailed primer on the Bash shell, though it represents an important development tool.

Shells are native to Linux and Mac OS. To communicate with remote servers, including cloud infrastructure, one relies mostly on Shells. Very recently, Microsoft has started including “Windows subsystem for Linux,” to provide full Linux functions within the Windows 10 operating system.

3 Data Management

3.1 Organizing Your Projects

Independent of the operating system you choose, one of the primary uses of GUI is to manage files and folders on your local system or attached to it in cases wherein you are using a shared filesystem or a form of network attached storage. GUIs provide an easy way to create, rename, move, and remove folders to reflect the needs of your work. All of this can be accomplished using the command line/shell, although that requires facility with the specific command names to perform efficiently. One area in which the SHELL is quite helpful is when needing to rename or manipulate a large number of files according to a general pattern in which cases a Bash script can be written. However, generally speaking it is fine to use a GUI to manage your project organization. The key is to choose a setup and naming convention that will make sense particularly after the passage of time such that when you return the structure will be understandable. Also consider that you might later involve a collaborator in which case you will need to explain the structure if it is not already evident. While organizing your project is a highly subjective activity here are some tips that will help you:

1. Do not stockpile code, data, and spreadsheets into a single folder. Create meaningfully sub folders up front to contain highly related information even if you do not yet have a significant amount of information to manage.

For example, all sequencing data should be in a folder and perhaps sub folders depending on the experimental design of your project. While this seems obvious, it is very easy to initiate a project by creating one folder with some basic data, perhaps

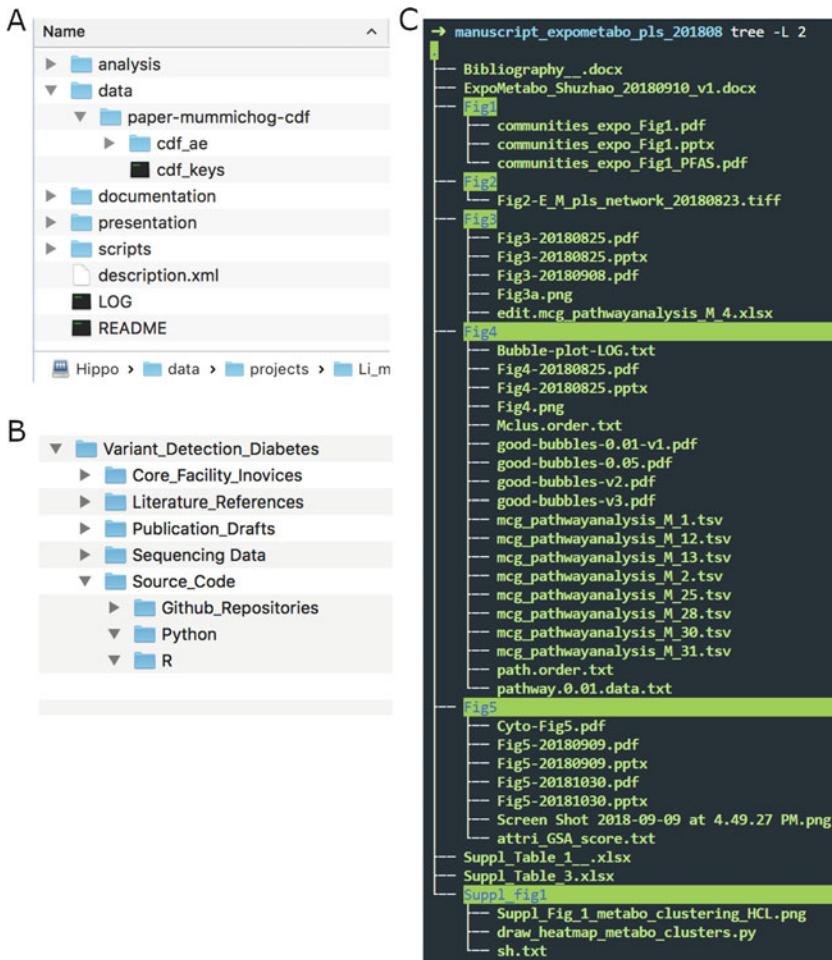


Fig. 2 File directories as examples for data management. **(a)** A project folder on a remote Linux server, mounted to MacOS desktop via sshfs. This is an automated directory structure on this server, so that scripts are run periodically to gather project statistics and generate reports. **(b)** A project folder with a focus on DNA sequencing. **(c)** A manuscript folder, where data are organized by figures

some invoices from a Core facility, PDFs of relevant publications, and example code. This might be manageable in the short term, but as more data arrives, it begins to dominate the folder which makes it difficult to locate the documents. It is always possible to create new subfolders within the originally created folder as more information accumulates though it becomes difficult to cleanly move information into the newer folders if naming conventions are not uniform. Figure 2 gives a few examples of organizing data for a project.

2. Choose names that avoid spaces, especially if you have collaborators using Linux.

It is possible to create folders and filenames that contain spaces within the name (e. g. “Sequencing Folder”). Note that

when accessing these files from a program or command line tool, they must be quoted so the shell (or code) will not interpret the name as being two distinct files. While this is a minor point, it is easier to use a fully specified name without spaces up front rather than retro fitting such a change. While doing this might not make a difference for your work it might very well make a difference to your collaborator who is attempting to programmatically manipulate your folder structure, particularly the sequencing files. Names should be specific and informative, and adding the date is a good idea (e.g., “SmithJ_Sequencing_Data_May_20_2019”), because the time stamp attached by the operating system may get lost after the files are moved around a few places.

3. Place data on a volume that is backed up or replicated to an off-premise site.

This may sound obvious, but projects frequently begin by using a local hard drive that might be personal property or not part of a research network and thus cannot access the drives or sharing services available within an organization. Many universities and research organizations have shared drives that can be mounted which is useful for backing up data although one must ensure that such shares can be accessed off campus. Services such as Dropbox or Box allow use of shared drives anywhere, and in cases where network activity is interrupted, the changes are synchronized once connectivity has returned. There is wide variation in how organizations offer backup and archival services so it is incumbent upon the researcher to make regular copies of project information. Obviously, in cases where the data is of considerable size (terabytes or petabytes) accommodations must be made at the outset to avoid moving the data except where necessary given that network traversal can be slow. At the individual level, it is possible to use Amazon Web Services or Google Cloud to stockpile project data at a reasonable cost, assuming your organization’s data management policies permit such an approach.

4. Use Git repositories to maintain source code. Use Github-like services for distribution and collaboration.

Git is arguably the most popular version control tool. Git and Github will be discussed in greater detail in a Chapter 15. Git allows users to manage changes to source code over time in a way that tracks all modifications with the added capability of being able to arbitrarily revert to a previous version. While this seems like more of a convenience for programmers it is useful for anyone engaged in the development of scripts, involved code, or pipelines that will eventually become part of a published or shared work. Services such as Github allow users to create repositories for hosting and sharing Git repositories

(abbreviated “repos”) at no cost. The repository is safely backed up and archived long term. Keep in mind that Github is only for source code as opposed to the data that the code will apply to. While it is possible, and advisable, to include a set of demo data within the repository, it should be small in size.

3.2 Notebook Tools

Data reproducibility is fundamental to science because a scientific conclusion is only valid if it can be validated by peers. With regard to publications that include analysis and code to arrive at a set of results, a general reader should be able to realize the same result given reasonable access to the code and data referenced in the publication.

Scripting notebooks are a useful tool for sharing the full process of data analysis. With respect to R, there is the **knitr** (<https://yihui.name/knitr>) package that provides dynamic report generation which permits the mixture of text, equations, images, and R code within a given document notebook that can be used to create results based on user supplied data. The format of the notebook can be Markdown, LaTeX, or Rnw all of which can be used to develop large scale publications in PDF or HTML formats. Thus, it is easy to create publication ready documents that can easily be regenerated in response to newly supplied data. Another popular tool is **Jupyter Notebook** (<https://jupyter.org/>), which also offers a notebook style framework for report code sharing and report generation. In addition to Python and R, Jupyter supports over 40 programming languages. By keeping code and results in a web browser, it keeps research record, facilitates collaboration, and makes good tutorials.

4 Coding and Scripting

4.1 Before You Start Coding

Software development usually requires specialized tools and workflow. Most bioinformatics data analysis does not require software development but scripting. Bioinformatics is often a lengthy process of many steps, each step to accomplish a specific task. It is critical to identify what tasks one need to complete in order to accomplish the overall goal. One has to know what the question is before choosing the right tool. Scripting is usually used to wrangle data and “glue” different steps. Interactive terminals and notebook tools (e.g., knitr and Jupyter notebooks) are often preferred over complex IDEs (integrated development environments). But it is still a good idea to version control the scripts. Should a different version produce a different result, one need to track down the source of the change.

Even without involving serious software development, a few practical tips are good to have. Do not repeat yourself—if a piece of code accomplishes a common task, make it reusable as a function,

class or module. Keep it simple and stupid—readability will save you and, more importantly, others, a lot of time later. Adopt conventions whereas possible, especially common data formats. These include csv, tsv, XML, and JSON. Since most programming languages have mature tools to handle common data formats, this reduces development cost and make the work more interoperable.

Learning to use a code editor is mandatory, because one has to know what is the command given to the computer and what is the input data. Microsoft Word is the most popular WYSIWYG editor, but it cannot be used for code editing. Because the user has to have absolute transparency over the code, and WYSIWYG editors mask the content by visual elements. A code editor works on plain text and gives you total transparency. The default text editor on MS Windows is Notepad, on Mac OS TextEdit. Among the popular free code editors are vim, Atom and Visual Studio Code. Beginners should avoid the complexity of IDEs (Integrated Development Environment, e.g., Eclipse), in order to focus on the mastering of the basic programming concepts. Rstudio, Matlab, and IPython are not exactly programming IDEs, but instead are interactive environments for data analysis.

4.2 What Languages to Use

It is easy to incite hot debates when choosing a programming language. In the field of bioinformatics, Python has already become the most popular language (replacing Perl). Java has wide applications in enterprise software development, and C/C++ is often preferred for performance. Javascript is now gaining prominence due to its central role in web development. With many mature libraries and a clean, elegant design, Python enables rapid scripting and development without compromising a lot of performance, since many libraries are implemented in C. Also highly popular, R is irreplaceable in the field of bioinformatics. R is designed as a statistical programming environment, centered on data analysis. Preferred by statisticians, many statistical tools were first published in R. The successful Bioconductor project further boosted its popularity. Both Python and R are explained in detail in Chapter 15.

The most practical choice, however, is dependent on several factors of a project. It takes time to gain proficiency in a language—one has to weigh over the time investment when learning a new tool. Domain specific applications are an important consideration. To work with relational databases, knowledge of Structured Query Language (SQL) is required to select and manipulate data from the database. One may also consider embedding SQL capability within software code and pipelines. Sometimes, the availability of one function determines the design of a project. If a commercial license is required, the cost and maintenance needs to be planned carefully. Moving a piece of software from a laptop to cloud could require a new license.

The real-world work environment often consists of mixed, heterogeneous computing components, which demand multiple programming languages. The maintenance cost of software programs could be grossly underestimated. The libraries the software is built upon evolve over time, which can lead to version conflicts and complex dependency issues. Docker container is a tool to produce a snapshot of all related resources of a software project, thus addressing this issue. If the components of software or different software programs communicate via standard interfaces, especially via a network protocol, the dependency issues will be isolated within each component. This also means, programs do not have to be written in the same language to communicate effectively. As a lot of computing moves into the web and cloud, this new paradigm is well accommodated in the latest trend in microservices.

5 Common Bioinformatics Tasks

There are bioinformatics tools for specific domains (e.g. BLAST and BLAT for searching DNA sequences, and XCMS, mzMine, and OpenMS for processing metabolomics data) (Chapters 2–4 in this book). Different from those, this section covers a set of common tasks that are often carried out by scripting using Shell, Python or R. These include data handling, analysis, and visualization. One needs the flexibility of scripting to accomplish the tasks, and some general guiding principles are explained here.

To copy, move, rename and back up batch files, interactive commands in the Shell are the right tool. Even simple commands can be written in a script and automated. For example, one can use one command to compress files and another to copy to a remote computer; a script consisted of these two lines of code can be run automatically at 12:15 am every night (using a systems service like Cron on UNIX). Readers can explore more on data management and Shell scripting in their own practice. In this section, we will cover quality control, data wrangling, common statistics, and visualization using metabolomics data as examples (**Note 1**).

5.1 Performing Quality Control (QC) and Quality Assurance (QA)

Before spending 2 years to analyze your data, it is critical to know that the data are not garbage due to some malfunction of an instrument. For high-throughput data, abnormality is easy to spot by examining two things: signal intensity acquired on a sample (Figs. 3 and 4), and how the measurements in a sample correlate with other samples (Fig. 5). If an instrument fails during the analysis of a sample, the signal intensity will not be consistent for this sample, resulting often low signal intensity. For samples of the same type of biological matrix (e.g., human serum), the basic pattern of intensity should be the same, therefore manifesting high correlation coefficients between these samples.

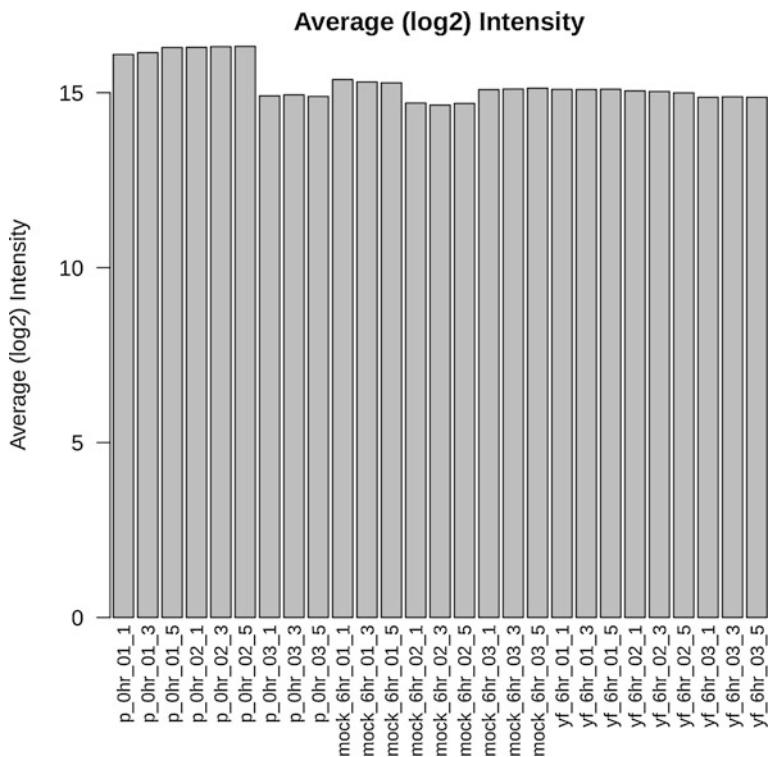


Fig. 3 Bar plot of average intensity of all replicates in the moDC dataset, inspected for quality control

Quality control should already be incorporated into the experimental protocols. Commonly, metabolomics experiments include spike-in exogenous chemicals as controls, which are chosen as stable isotope labeled molecules that are absent in biological samples. These control molecules should give consistent peaks in the data of each sample, and visualized for quality control. Similarly, one can plot the signals from common endogenous metabolites, while their signals may not be consistent for biological reasons. An example of phenylalanine (protonated molecular weight 166.0863, consistently seen in human plasma using positive ESI on LC-MS) is shown in Fig. 4. Sometimes, injection replicates are performed on mass spectrometry metabolomics. These are technical replicates and expected to generate the same result (Fig. 5a).

The above tasks are usually associated with the laboratory that performs the experiments, and they should be completed before the data are handed over to the next stage of projects. For people responsible for the data analysis, these QC results should be examined before time and effort is spent on downstream analysis. In addition, PCA and MA plots are good ways to check for anomalies in the data (Figs. 6 and 7).

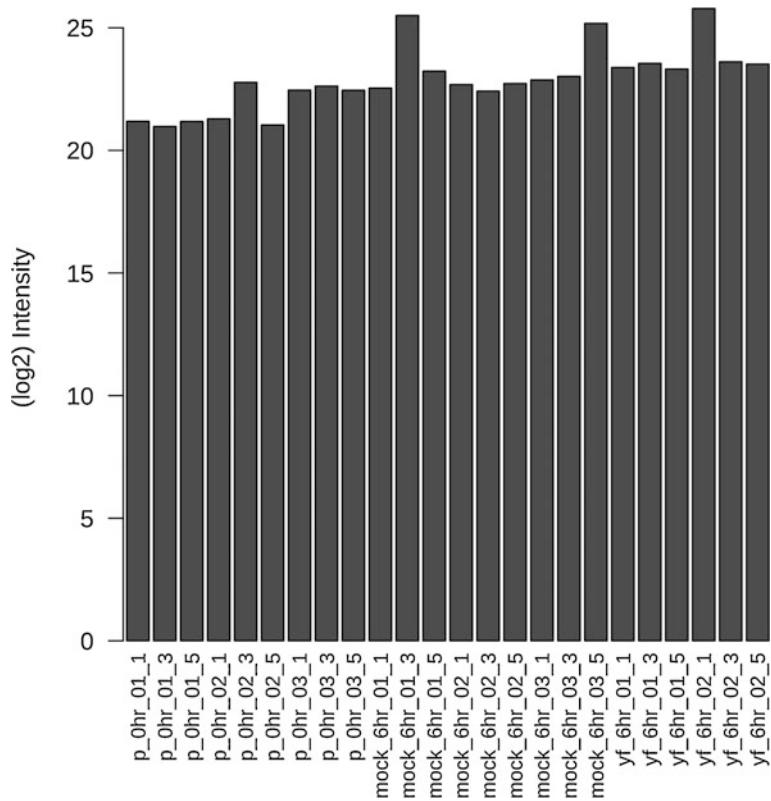


Fig. 4 Bar plot of phenylalanine intensity in all samples

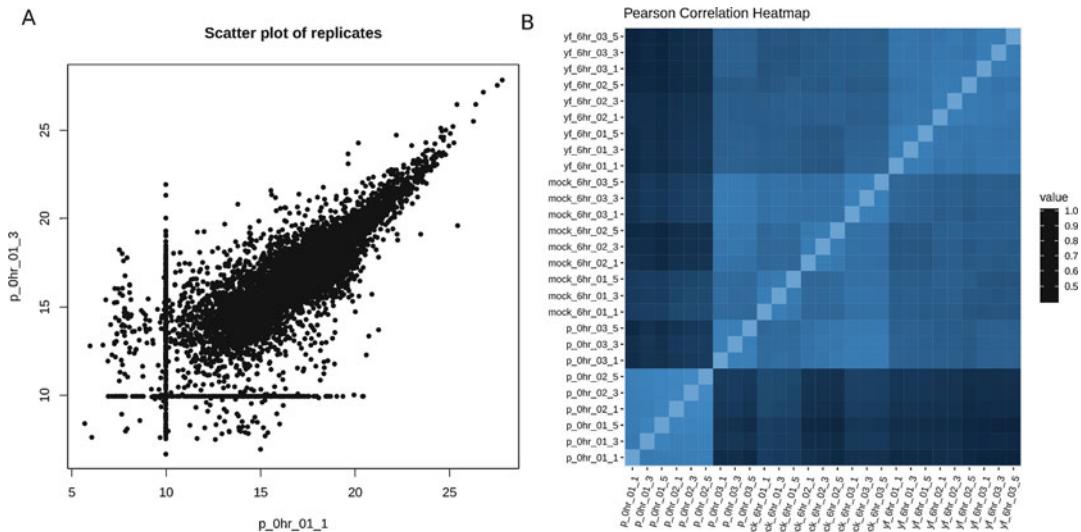


Fig. 5 Using correlation between samples to examine data quality. **(a)** Scatter plot of all features between two technical replicates. Each dot is a feature. The high number of values close to 10 (forming two lines) reflects imputed values in the data. This scatter plot is typical for metabolomics and transcriptomics data, where reproducibility is better for features of higher intensities. **(b)** Pair-wise Pearson correlation coefficients for all samples plotted on a heatmap, where the color is scaled by the coefficient value

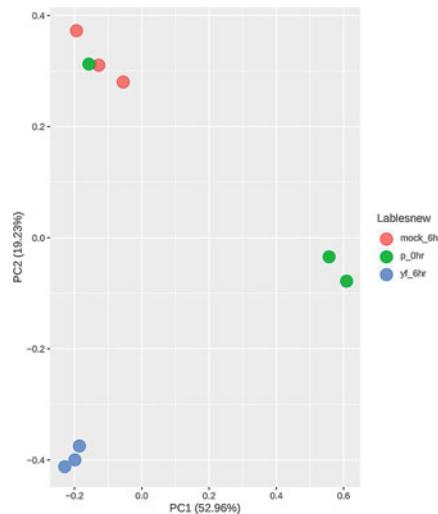


Fig. 6 Using PCA plot to examine grouping patterns of biological samples. Each dot represents a sample. One of the 0 h samples groups with the mock 6 h samples. This raises a flag for further investigation

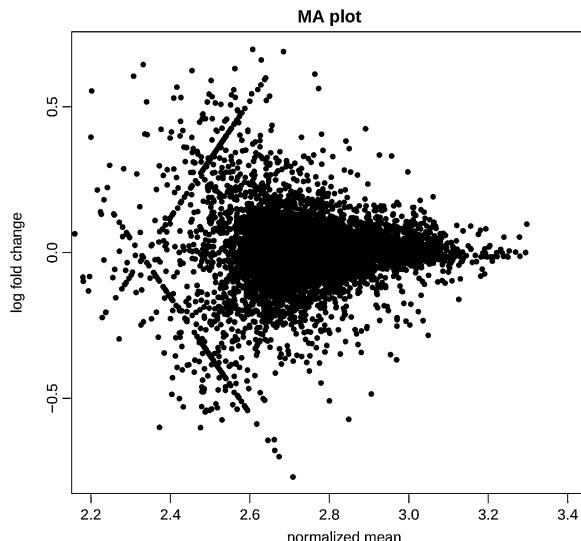


Fig. 7 The MA plot of two replicates or samples should have a flat trendline, otherwise it indicates a bias in data distribution. The straight lines of features are from imputed values, similar to those in Fig. 5a

5.2 Data Transformation, Scaling and Normalization

For high-throughput data, it is often useful to filtering the data for missing values and low intensities, so that data of higher quality is used for subsequent analysis. Many statistical methods do not cope with missing data, and imputation needs to be performed. It is common in metabolomics data to replace missing values by limit

of detection. Similar to transcriptomics, the measurement errors in metabolomics data tend to be multiplicative. After log transformation (usually by log₂), the data are more likely to follow a normal distribution.

Although variation is expected in data, it often arises due to uncontrolled factors such as inconsistent specimen collection, or instrument drift. This sort of variation can obscure insight one may gain from what was supposed to be measured. Data normalization, scaling and standardization attempt to suppress unwanted variation while minimally affecting the quality of the data. These preprocessing methods are often required before further analysis can be done as a way to prepare the data so that only what is meant to be analyzed can be analyzed. Some algorithms, such as PCA and Gradient descent in machine learning, require scaling, whereas for others, such as t-test there is no effect but scaling may help clearer presentation of the result.

Scaling is transforming the data by applying a function to it. Normalization and standardization are both types of scaling. Standardization is changing the data to have a mean of zero and a standard deviation of one. Normalization for -omics data usually refers to scaling each sample to a similar distribution of feature intensities. Common methods of normalization include mean centering, LOESS (local weighted regression) and quartile normalization. Methods more specific to metabolomics data are available [3]. Of note, LC-MS metabolomics tends to have many missing values for low-abundance features (similar to single-cell RNA-seq data). Caution has to be taken using common normalization methods. In our example code, we used a strategy to guide normalization using only the more abundant metabolites.

5.3 Common Statistical Analyses

The field of metabolomics has a good share of complex statistics. Multivariate methods are often used to project data into latent spaces, and PLS-DA (partial least square discriminant analysis) is among the popular methods to compare biological classes. Principal component analysis (PCA) is not directly used for statistical analysis, but often used for dimension reduction and visualization. Due to its unsupervised nature, PCA is also used for exploration and quality control. In Fig. 6, one baseline sample is not grouped with other baseline samples, but with a treatment group. This figure raises a red flag on that particular sample and demands further investigation of the cause.

Less complex but often preferred are the univariate statistics. If the data are known to follow normal distribution, ANOVA (Analysis of variance, three or more groups) or t-test (two groups) can be used. They can be further divided into categories for independent samples or related samples. If the distribution of data is not normal or unknown, nonparametric methods such as Mann–Whitney U test can be used. For continuous outcome, linear regression is

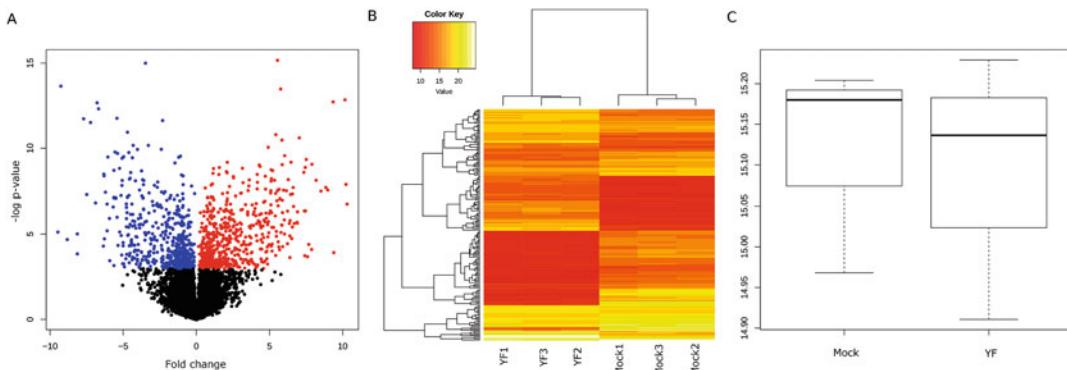


Fig. 8 Visualization of results from a statistical test. (a) Volcano plot of all features. The *Y*-axis indicates the significance of a feature in the statistical test, and the *X*-axis shows the direction and magnitude of the difference between two groups. (b) A heatmap of the significant features ($\text{FDR} < 0.05$ here) the statistical test. Each column represents a sample and cell of the heatmap represents a feature of one sample. The six samples are separated into two biological groups, matched to the experimental design. (c) Boxplot to visualize the mean level and standard deviation of one feature in the two biological groups

commonly used. A more extensive review on this topic can found in Gardinassi et al., 2017 [4]. In our example code, we show the comparison between two experimental groups using the Student's *t*-test, visualized via a volcano plot, a heatmap, and a box plot (Fig. 8).

It is important to note that multiple testing is a major concern in -omics data (also see Chapter 20). The chance of getting a small *p*-value increases greatly when one performs the same test hundreds or thousands of times. False discovery rate (FDR) is a common approach to adjust the result by multiple test correction, implemented with various algorithms. The reporting of FDR is considered necessary in scientific publications using metabolomics data. These above statistical methods are available as both R and Python programming tools. An example of more user-friendly tools is MetaboAnalyst, which enables users to perform data analysis by interacting with graphical web interface (see Chapter 17). After the statistical analysis of metabolite features, pathway and network analysis can be performed to bring the data to the relevant biological context (see Chapter 19).

5.4 Visualization

A large portion of scientific findings are communicated via figures. For data intensive publications, the visual presentation has even more special challenges. A good example is the rising of Circos plots. Circos was initially developed for genome projects, but has since been applied to many domains to illustrate the relationships between complex data [5]. One can condense a great amount of data of many different types of plots, such as line plots, histograms, tiles and heatmap, into one figure. This type of figures has to be

generated by computer scripts, as manual artistry can no longer cope with the complexity and precision.

A few principles apply to scientific figures, regardless how the figures are generated. Firstly, the type of figures or plots should be determined by the underlying message. For instance, a Venn diagram is to visualize the overlap between two or more categories, but will be ineffective if there are more than four categories, in which case UpSet is a better solution (<http://caleydo.org/>). Secondly, illustrations are made to deliver a clear message. Optional visual elements, if not related to the message, should be avoided. This includes the parsimony of using colors. Readers will wonder what the color codes for if a color is used in a figure. If the purpose is not clear, the result will be more confusing than helpful. The real estate on a figure should be prioritized for the elements that deliver the message. If the pattern is shown by a heatmap is clear in an area of 1 square inch, there is no need to plot the heatmap in 2 square inches. But if the sample labels are important to the message, they need to have an appropriate portion of the area and be clearly readable. Thirdly, figures should adhere to the instructions by print journals from the start. Journals generally require figures to be submitted in separate files, and of enough resolution (e.g., 300 dpi). Figures and their captions should not be embedded for submission, because the publishers need to typeset the manuscripts according to their standards. Many journals specify limited sizes to choose for figures, for example, to accommodate 2-column or 3-column format of the print. Keeping fonts consistent for all figures is required by many journals. All journals post their author instructions on the web, and readers should at least familiarize with at least a few of them. Finally, scientific results have to reproducible. Most publications today undergo revisions. The manuscripts and corresponding data analyses should be properly versioned. In fact, for a manuscript involving complex data analysis, it is also a good way to organize the underlying data clearly by each figure, so that the figure can always be regenerated even after modifications (Fig. 2c). We illustrate here a few common types of figures in our example code, including barplots, boxplots, scatter plots, heatmaps, and volcano plots (Note 2).

6 Notes

1. Code example. We use the moDC dataset from the Mummi-chog paper [6] to illustrate major steps in processing, data analysis and presentation. This was an experiment of infecting immune cells by yellow fever virus (YFV). Three experimental groups are baseline (0 h), mock infection (6 h of culture but no virus), and YFV infection (6 h). Each group included three biological samples, and each biological sample was run in

triplicate on a mass spectrometer coupled with liquid chromatography. The example data and code are available at the supplemental website: <https://metabolomics-data.github.io>.

2. It is easier to learn the bioinformatics skills by working on a project. The computer programming world changes quickly and most people learn without taking a class. People learn differently, and one may have to try various things to find a suitable approach. Online help can be found via websites such as Stack Overflow (<https://stackoverflow.com/>) and BioStars (<https://www.biostars.org>).

Acknowledgments

This work has been funded, in part, by the US national Institutes of Health via grants UH2 AI132345 (Li), U2C ES030163 (Jones, Li, Morgan, Miller), U01 CA235493 (Li, Xia, Siuzdak), U2C ES026560 (Miller), P30 ES019776 (Marsit), P50 ES026071 (McCauley), and the US EPA grant 83615301 (McCauley).

References

1. Zauhar RJ (2001) University bioinformatics programs on the rise. *Nat Biotechnol* 19(3):285
2. Gilbert W (2003) Life after the helix. *Nature* 421:315–316
3. De Livera AM, Olshansky G, Simpson JA, Creek DJ (2018) NormalizeMets: assessing, selecting and implementing statistical methods for normalizing metabolomics data. *Metabolomics* 14 (5):54
4. Gardinassi LG, Xia J, Safo SE, Li S (2017) Bioinformatics tools for the interpretation of metabolomics data. *Curr Pharmacol Rep* 3 (6):374–383
5. Krzywinski M et al (2009) Circos: an information aesthetic for comparative genomics. *Genome Res* 19:1639–1645
6. Li S, Park Y, Duraisingham S, Strobel FH, Khan N, Soltow QA, Jones DP, Pulendran B (2013) Predicting network activity from high throughput metabolomics. *PLoS Comput Biol* 9(7):e1003123



Chapter 15

The Essential Toolbox of Data Science: Python, R, Git, and Docker

W. Stephen Pittard and Shuzhao Li

Abstract

The daily work in data science involves a set of essential tools: the programming languages Python and R, the version control tool Git and the virtualization tool Docker. Proficiency in at least one programming language is required for data science. R is tied to a computing environment that focuses on statistics, in which many new algorithms in genomics and biomedicine are first published. Python has a root in system administration, and is a superb language for general programming. Version control is critical to managing complex projects, even if software development is not involved. Docker container is becoming a key tool for deployment, portability, and reproducibility. This chapter provides a self-contained practical guide of these topics so that readers can use it as a reference and to plan their training.

Key words Bioinformatics, Data science, Python, R, Git, Docker, Version control, Virtualization

1 Introduction

The critical role of data science permeates many aspects of our society. As for introductory bioinformatics, the basic concepts of data management, analysis and visualization are covered in Chapter 14, with real examples of metabolomics. Chapter 16 describes predictive modeling and machine learning. Other specialized domains, such as high-performance computing, are not covered in this book. This chapter covers the toolset of data science in general: the programming languages Python and R, the version control tool Git and the virtualization tool Docker. From our experience, these are the essential tools required for data-intense work on a daily basis.

Proficiency in at least one programming language is required for data science, depending on application domains. Python has a root in system administration, and is a superb language for general programming. R is tied to a computing environment that focuses on statistics, in which many new algorithms in genomics and biomedicine are first published. Both Python and R are interpreted,

high-level languages. They allow the direct interactions via command line with the language interpreter, thus easing the learning curve and facilitating code prototyping. Between saving time writing code and computing efficiency, both languages lean toward the former. However, if the libraries used in an application are implemented efficiently (e.g., written in C language), computing performance needs not suffer. It is the excellent ecosystems of available libraries and packages that establish the popularity of these two languages in data science.

Version control is critical to managing complex projects, even if software development is not involved. Docker container is becoming a key tool for deployment, portability, and reproducibility. Excellent tutorials and documentations are available for each of these tools. This chapter is intended to be a self-contained practical guide for readers to learn the topics in a structured pace. All these tools are free to use and available on Microsoft Windows, Apple Mac OS, and Linux operating systems.

2 Python for Data Science

The Python programming language was created by Guido van Rossum in early 1990s. The name came from the British television series “Monty Python” (not from the snake); thus, a flavor of humor is seen occasionally in its code and documentation. It started on a modern and elegant design, and code readability is enforced by its syntax. Together with other features, Python is suited for both small and large projects, and is used in many complex, enterprise-level software applications.

2.1 Installation and Relevant Packages

Python is included by default in Apple Mac OS and Linux distributions, as it also serves system administrative functions. Thus, besides the Python package manager *pip*, Python libraries can be installed via the software manager of the operating systems. For scientific computing, several libraries are indispensable and called the “SciPy stack” (as presented at <https://www.scipy.org>), including *NumPy*, *SciPy*, *Matplotlib*, and *pandas*. The basic numerical array data structure is provided by *NumPy*, and *pandas* enables major modern data structures similar to those in R. *Matplotlib* provides comprehensive plotting functions. The name of *Matplotlib* reflects its initial similarity to Matlab, a popular commercial tool for data analysis and visualization. But the library has been rewritten for object oriented programming. *Matplotlib* is the basis of other Python plotting tools, including *seaborn*. In addition, comprehensive methods of machine learning are implemented in the *scikit-learn* library.

Custom installation of Python on Mac OS and Linux is often necessary to upgrade to newer versions, or accommodate multiple versions. Python can be downloaded from its official site, <https://>



Fig. 1 Common Python libraries for scientific computing that are shipped with the Anaconda distribution. The logos may be trademarked by individual parties. The tool stack is compatible with all three major operating systems

www.python.org, for Linux/UNIX, Mac OS, Windows, and other operating systems. For beginners, the simplest way is to install Anaconda, a software platform for data analytics and scientific computing, which includes all the tools discussed in this section (Fig. 1). Instruction at <http://docs.continuum.io/anaconda/install/>). At the time of writing, users should start on Python 3.7 because Python 2 will not be supported beyond 2020.

2.2 Interactive Computing with Python SciPy Stack

Once Anaconda (or another Python distribution with the above libraries) is installed, we can start working on data—the possibilities are almost endless! Evoke the interactive environment by typing “python” in the command line shell and pressing the ENTERkey. This interactive environment is marked by “>>>” starting every line, and the code in the current line is executed every time the ENTER key is pressed:

```
>>> 165.079 + 1.0073
166.0863
```

Import *NumPy* and *pandas* libraries:

```
>>> import numpy as np
>>> import pandas as pd
```

Try a few simple things:

```
>>> n = 5
>>> a = [2, 3, 4, 5, 6]
>>> a2 = np.array(a)
```

```

>>>
>>> type(n)
<class 'int'>
>>> type(a)
<class 'list'>
>>> type(a2)
<class 'numpy.ndarray'>
>>>
>>> # help will show the supporting documentation on the
object, "q" to quit
>>> help(a2)

```

This above snippet shows: n is defined as an integer, a as a list and a2 as a n-dimensional array in NumPy. Same as in R, content behind “#” is a comment and ignored from execution. Next, we download the sample data “modc_ae_2012.txt” from our GitHub site, and place it under this work directory (where we started this Python interactive environment). We use the input function `read_csv` in *pandas* to read the data:

```

>>> mydata = pd.read_csv("modc_ae_2012.txt", sep="\t")
>>> mydata.shape
(7995, 31)
>>> mydata.head()
      mz        time    mz.min    mz.max p_Ohr_01_1 ...
0  85.02783  59.68820  85.02783  85.02783   15.5810 ...
1  85.04717 124.75120  85.04709  85.04739   14.4754 ...
2  85.06532  68.66651  85.06517  85.06547   14.4223 ...
3  85.10073  16.48022  85.10050  85.10078   14.5249 ...
4  86.05951  67.78485  86.05949  86.05980   10.6674 ...
[5 rows x 31 columns]
>>>

>>> mydata.mean(0) [:8]
mz            479.377606
time          165.949377
mz.min        479.327571
mz.max        479.427072
p_Ohr_01_1    16.091406
p_Ohr_01_3    16.144007
p_Ohr_01_5    16.290922
p_Ohr_02_1    16.295431
dtype: float64

```

In the above snippet, we read our example data into an object `mydata`, which is a table of 7995 rows and 31 columns. A peek of

the data (first 5 rows) is printed out by `mydata.head()`. We calculate the mean value of each column by `mydata.mean(0)`. The dot “.” after an object calls for its attribute or method.

2.3 Concepts in Data Structure

For simplicity, we take a subset of the data (first 4 rows and 6 columns) for this section.

```
>>> sliced_data = mydata.iloc[0:4, 0:6]
>>> sliced_data
      mz      time    mz.min    mz.max p_0hr_01_1  p_0hr_01_3
0   85.02783    59.68820   85.02783   85.02783     15.5810
16.0425
1   85.04717   124.75120   85.04709   85.04739     14.4754
14.2709
2   85.06532    68.66651   85.06517   85.06547     14.4223
15.0515
3   85.10073    16.48022   85.10050   85.10078     14.5249
13.2573
>>> type(sliced_data)
<class 'pandas.core.frame.DataFrame'>
>>>
>>> sliced_data.iloc[0]
      mz      85.02783
      time      59.68820
      mz.min    85.02783
      mz.max    85.02783
      p_0hr_01_1  15.58100
      p_0hr_01_3  16.04250
Name: 0, dtype: float64
>>> type(sliced_data.iloc[0])
<class 'pandas.core.series.Series'>
>>>
>>> sliced_data.iloc[0].to_numpy()
array([85.02783,  59.6882 ,  85.02783,  85.02783,  15.581    ,
       16.0425 ])
>>> type( sliced_data.iloc[0].to_numpy() )
<class 'numpy.ndarray'>
>>>
>>> list(sliced_data.iloc[0].to_numpy())
[85.02783000000001, 59.6882, 85.02783000000001,
 85.02783000000001, 15.581, 16.0425]
```

In the above example, we checked the type of each object. We have `sliced_data` as a “DataFrame.” The method `sliced_data.iloc[0]` gets the first row of `sliced_data`, and is a “Series.” These are data structures *pandas* uses to facilitate applications data science. Series is a one-dimensional labeled array of the same data type, and DataFrame is a 2-dimensional labeled data

structure with columns of potentially different types. DataFrame is similar to the data frame in R, and Series to the vector in R. We can convert the Series to NumPy array by `.to_numpy()`, and further to a regular list in Python by `list()`.

2.4 Plotting

We import *Matplotlib* here for data plotting.

```
>>> import matplotlib.pyplot as plt
>>> X = mydata['p_0hr_01_1']
>>> Y = mydata['p_0hr_01_3']
>>> plt.figure(figsize=(5,5))
<Figure size 500x500 with 0 Axes>
>>> plt.plot(X, Y, 'r.', markersize=2)
[<matplotlib.lines.Line2D object at 0x7f359301b780>]
>>> plt.title("Scatter plot of two samples")
Text(0.5, 1.0, 'Scatter plot of two samples')
>>> plt.savefig("Figure2.png")
```

The above snippet places the metabolite intensities of two samples in our example data into variables `X` and `Y`, and creates a scatter plot to visualize their relationship. The saved figure is shown here as Fig. 2. The function `plt.savefig()` can further specify figure resolution and formats such as PDF, PNG, or SVG.

This completes our introductory section on Python for data science. To continue onto the next level, readers are encouraged to complete two training materials: the 10-min tutorial linked on *pandas* website (<https://pandas.pydata.org/>), and the 6-h course of programming in Python on the website of Software Carpentry (<https://software-carpentry.org/lessons/>). With additional experience, readers are also encouraged to use iPython and Jupyter notebook in place of the default interactive environment.

3 R for Data Science

The R language is an open source implementation created by Ross Ihaka and Robert Gentleman based on the S programming language which was originally created at AT&T Bell Laboratories. R provides an interactive environment for data manipulation, visualization, and statistical computing that includes strong support for programming constructs and robust input and output capabilities. R is extensible via the “package” mechanism which facilitates the addition of user-contributed code. The Comprehensive Archive Network (CRAN) offers 14,348 contributed packages at the time of writing, across many application domains. R benefits from enthusiastic user support and a highly motivated community of developers, statisticians, researchers, and data scientists. An extended version of this text in notebook format is posted at our GitHub site <https://metabolomics-data.github.io/>.

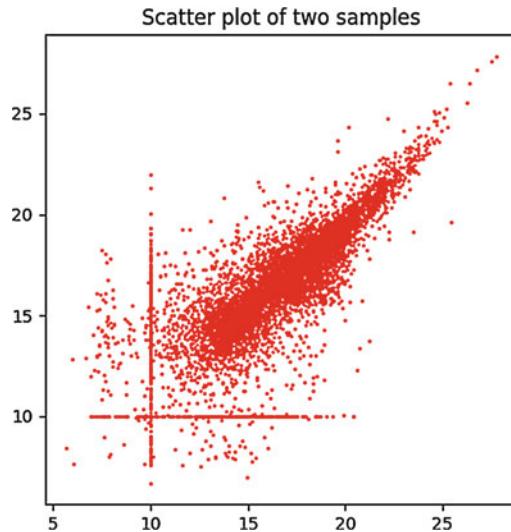


Fig. 2 Example of scatter plot in Python

3.1 Installation and Packages

The R project website <https://cran.r-project.org/> offers single click installers for Microsoft Windows, Apple Mac OS, and Linux (in the form of packages) designed to provide the base R interactive environment, language, graphics capability, and a foundational set of statistical functions. A default installation of R is comprised of core *packages* which offer basic functionality that can be extended with user contributed packages via the `install.packages()` command. As an example, the following will install the *actuar* package onto the local system. The `library()` function is then used to load the package into the workspace after which functions within the package may be used.

```
> install.packages("actuar")
Installing package into '/Users/esteban/r_packages'
(as 'lib' is unspecified)
--- Please select a CRAN mirror for use in this session ---
trying URL 'https://mirrors.nics.utk.edu/cran/bin/macosx/el-
capitan/contrib/3.5/actuar_2.3-1.tgz'
Content type 'application/x-gzip' length 2044668 bytes (1.9
MB)
=====
downloaded 1.9 MB

The downloaded binary packages are in
/var/folders/wh/z0v5hqgx3dzdfgbr_3w0000gn/T//Rtmpy1m6A7/down-
loaded_packages

> library(actuar).
```

The associated Bioconductor project <https://www.bioconductor.org/> provides 1436 packages specifically for the exploration of high-throughput genomic data. Management of Bioconductor tools involves a different process that first requires the installation of the *BiocManager* package.

```
> if (!requireNamespace("BiocManager", quietly = TRUE))
    install.packages("BiocManager")
> BiocManager::install()
```

To then install a Bioconductor package, simply supply the desired package name(s) as a vector:

```
> BiocManager::install("xcms") # Install the xcms package
> BiocManager::install(c("xcms", "GenomicFeatures"))
```

3.2 Getting Help

A good starting place for R assistance is the “Getting Help” page at <https://www.r-project.org/help.html> which presents FAQs and steps on how to walk through demonstrations of various functions and packages. R also contains the *browseVignettes()* command which provides a web listing of all available vignettes / guided tours associated with a given package. Assistance is also available via Stack Overflow <http://stackoverflow.com> with bioinformatics specific support being available at the BioStars forum <https://www.biostars.org/>. Please read the respective terms of use for both sites prior to the submission of questions.

3.3 Variables and Data Structures

Variables are containers for data involving four primary types: numeric, character, logical, and factor. Variables are assigned using the symbol “`<-`” or “`=`”; however, the former is *strongly* preferred. R is largely an interactive environment that retains variables and data structures during and, optionally, across sessions. The interpreter accepts input after the “`>`” character.

```
> A <- 2.5    # The "<-" is the preferred method of assignment

> A = 2.5    # This is equivalent to the above although using
the "=" is
               # strongly discouraged except when setting function
arguments.
> A
[1] 2.5
```

R variables are case sensitive (“MyVar” vs. “myvar”), cannot contain spaces (“my var”), or begin with symbols (., \$, %, ^, &, *, #, (,), !). Character variables typically represent qualitative or labeled information that might later be converted to factors, which are variables which can only take a collection of predefined values. Numeric variables typically represent measured information although integers could also represent a coded or binary “Yes/No” response. Logical variables take the value “TRUE” or “FALSE,” and are usually the result of some comparison within a programming construct (*if else*) or are used when indexing into vectors, matrices, lists, or data frames.

3.3.1 Vectors

Vectors are containers for homogeneous data that in turn can be assembled into matrices. Vectors can be explicitly created using the “**c**” function or be returned from other R functions.

```
> 1:5 # Create a vector from 1, 2, 3, 4, 5
[1] 1 2 3 4 5

> rnorm(8)      # Generate 8 random values from a Normal
Distribution
[1] -0.75633261   0.70375541 -0.05123946   1.07488741
0.82288562 -0.45921131
[7] -0.48127334   0.58647522

> y <- 1:10 # A vector with 10 elements (1 .. 10)

> y <- c(1,2,3,4,5,6,7,8,9,10) # Same as above yet using the
"c" function
```

Consider the following example which simulates the collection of height measurements of eight people. This example also introduces the bracket notation which uses logical expressions to identify subsets of information.

```
> height <- c(59,70,66,72,62,66,60,60) # create a vector of
8 heights

> height[1:5] # Get first 5 elements
[1] 59 70 66 72 62

> height[c(1,5)] # Get just first and fifth elements
[1] 59 62
```

This example shows the use of more involved logical conditions within the brackets.

```
# Find values between 60 and 70
> height[height > 60 & height < 70]
[1] 66 62 66

# Find heights between 60 and 70 inclusive
> height[height > 60 & height <= 70]
[1] 70 66 62 66
```

Vectors can be computed upon and used as input to functions.

```
> weight <- c(117,165,139,142,126,151,120,166) # weight
(in lbs)

> weight/100
[1] 1.17 1.65 1.39 1.42 1.26 1.51 1.20 1.66

> mean(weight)
[1] 140.75
```

3.3.2 Matrices

A matrix can be considered as a vector with a dimension attribute. To this end, the ***dim()*** function is used to impose a matrix structure onto an existing vector. The ***matrix()*** function can also be used.

```
> set.seed(123) # Makes this example reproducible
> mymatrix <- rnorm(9)
> mymatrix
[1] -0.56047565 -0.23017749  1.55870831  0.07050839
0.12928774  1.71506499
[7]  0.46091621 -1.26506123 -0.68685285

> dim(mymatrix) <- c(3,3)
> mymatrix
     [,1]      [,2]      [,3]
[1,] -0.5604756 0.07050839  0.4609162
[2,] -0.2301775 0.12928774 -1.2650612
[3,]  1.5587083 1.71506499 -0.6868529

> mymatrix <- matrix(rnorm(9),nrow=3,ncol=3)
> mymatrix
     [,1]      [,2]      [,3]
```

```
[1,]  0.7013559 -0.2179749 -0.6250393
[2,] -0.4727914 -1.0260044 -1.6866933
[3,] -1.0678237 -0.7288912  0.8377870
```

As with vectors, matrices can be indexed using bracket notation which, in this case, will require two indices, one for rows and another for columns. Omitting an index implies that all rows or columns are desired.

```
> newmat[1,1] # First row and first column
[1] -0.5604756

> newmat[1,] # First row and all columns
[1] -0.56047565  0.07050839  0.46091621

> newmat[1:2,] # First two rows
[,1]      [,2]      [,3]
[1,] -0.5604756 0.07050839  0.4609162
[2,] -0.2301775 0.12928774 -1.2650612
```

3.3.3 Lists

Lists provide a way to store heterogeneous data within a single structure. Newcomers to R usually do not create lists except when (1) writing a function that needs to return heterogeneous information or (2) as a precursor to creating a data frame. Lists are more commonly encountered in results returned by common statistical modeling activities such as regression, decision trees, and random forests.

```
> data(mtcars) # Load mtcars into the environment
> mylm <- lm(mpg ~ wt, data = mtcars)
```

Use the `names()` function to see the list elements.

```
> names(mylm)
[1] "coefficients"    "residuals"        "effects"         "rank"
[5] "fitted.values"   "assign"          "qr"              "df."
residual"
[9] "xlevels"          "call"            "terms"           "model"
```

The “\$” symbol is used to address specific list elements.

```
> mylm$rank
[1] 2
```

```
> mylm$call
lm(formula = mpg ~ wt, data = mtcars)

> mylm$coefficients
(Intercept)           wt
37.285126   -5.344472
```

3.3.4 Data Frames

Data frames are tightly coupled collections of variables organized into a rectangular format. Data frames can be constructed from vectors, lists, matrices or by reading in CSV files, or from information being returned by APIs (Application Programming Interfaces). Best of all, data frames can hold different data types across multiple columns. There are a number of data frames built into R that can be used to illustrate how to work with them. The **data()** function can be used to load a built-in data frame.

```
data(mtcars). # Load the mtcars data frame into the environment

# Determine structure. It looks like a list
> str(mtcars)
'data.frame': 32 obs. of 11 variables:
 $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
 $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
 $ disp: num  160 160 108 258 360 ...
 $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
 $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
 ...
 $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
 $ qsec: num  16.5 17 18.6 19.4 17 ...
 $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
 $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
 $ gear: num  4 4 4 3 3 3 4 4 4 ...
 $ carb: num  4 4 1 1 2 1 4 2 2 4 ...

> nrow(mtcars) # How many rows does it have ?
[1] 32

> ncol(mtcars) # How many columns are there ?
[1] 11
```

The previously described bracket notation can be used with data frames just as easily.

```
> mtcars[1:3,] # Get the first three rows
```

```

mpg cyl disp hp drat    wt  qsec vs am gear carb
Mazda RX4     21.0   6 160 110 3.90 2.620 16.46  0  1     4     4
Mazda RX4 Wag 21.0   6 160 110 3.90 2.875 17.02  0  1     4     4
Datsun 710    22.8   4 108  93 3.85 2.320 18.61  1  1     4     1

# Get the first three rows and only columns one and four
> mtcars[1:3,c(1,4)]
      mpg   hp
Mazda RX4     21.0 110
Mazda RX4 Wag 21.0 110
Datsun 710    22.8  93

# Find all rows wherein the mpg is >= 30 and then select
# columns 2 through 6

> mtcars[mtcars$mpg >= 30.0,2:6]
      cyl disp hp drat    wt
Fiat 128      4 78.7 66 4.08 2.200
Honda Civic    4 75.7 52 4.93 1.615
Toyota Corolla 4 71.1 65 4.22 1.835
Lotus Europa   4 95.1 113 3.77 1.513

```

3.4 Functions

Functions are a very important part of the R language especially given that all packages are comprised of predefined functions to assist with a large variety of tasks. Users communicate within R almost entirely through functions, thus consider writing one to avoid repeating the interactive entry of commands. However, before writing new code it is wise to first determine if there is not already an existing function that performs the desired actions. A significant idea associated with software development is “Don’t Repeat Yourself” commonly abbreviated as “DRY.” Functions are created using the *function()* directive and are stored as R objects just like any other variable. Functions allow easy reuse of code that can optionally be used in the creation of a package for distribution on CRAN. Information on a preexisting function can be obtained using the “?” character before the function name of interest.

```

> ?mean
mean                               package:base
Documentation

```

Arithmetric Mean

Description:

Generic function for the (trimmed) arithmetic mean.

Usage:

```
mean(x, ...)
```

3.5 S3 and S4 Objects

Organizing high-throughput data types such as geospatial, time series, and experimental information can be challenging using a single data frame or matrix. To provide a meaningful representation might involve a combination of structures organized into an “object” that makes intuitive sense for users. As an example, the output from the linear modeling *lm()* function involves a list of eleven elements (some of which are lists themselves) known to be useful in model evaluation.

```
> mylm <- lm(mpg~., data=mtcars)
> names(mylm)
[1] "coefficients"   "residuals"      "effects"        "rank"
[5] "fitted.values"  "assign"         "qr"             "df."
residual"
[9] "xlevels"        "call"          "terms"          "model"
```

R provides “generic” functions such as *plot()*, *summary()*, and *print()* which, based on the object type, will “dispatch” work to an underlying “method” that knows what to do with the specific object type in question. In the case of the *print()* function, there are three methods relating to regression although the user need not worry about which to call as the S3 Object system handles this:

```
> grep("lm", methods(print), value=TRUE)
[1] "print.glm"           "print.lm"          "print.summary.glm"
[4] "print.summary.lm"

> print(mylm)

Call:
lm(formula = mpg ~ ., data = mtcars)

Coefficients:
(Intercept)          cyl          disp          hp          drat
               12.30337     -0.11144      0.01334     -0.02148      0.78711
              wt
-3.71530
qsec            vs          am          gear          carb
              0.82104     0.31776      2.52023      0.65541     -0.19942
```

It is important to understand that various objects exist within the typical R environment although novice and intermediate users will primarily be consumers of objects as opposed to developers. In R, there are three object systems, S3, S4, and Reference Classes. Creating objects involve considerations customarily associated with Object Oriented Programming (OOP) which are beyond the scope of the text. Of note, to call for an attribute of an object, we use “\$” in R, which is the equivalent of the dot “.” in Python. The dot in a string has no lexical meaning in R.

3.6 Graphics

The R language provides a powerful environment for the visualization of scientific data. It provides publication quality graphics, which are fully programmable and reproducible. There are a number of useful output types (PDF, JPEG, PNG, SVG) in addition to the default high resolution on-screen graphics capability. R Graphics can be confusing given that there are three primary user-focused graphics systems: Base, Lattice, and ggplot2.

3.6.1 Base Graphics

Base graphics is the default display package included in every R installation. It has both high and low level routines which provides flexibility in developing customized plots. Base graphics is well documented and there is significant support available via Google.

3.6.2 Lattice Graphics

Lattice graphics <http://lattice.r-forge.r-project.org/> is an implementation of Trellis graphics outlined in the “Visualization Data” text by William Cleveland. Lattice simplifies the creation of conditioned plots with automatic creation of axes, legends, and other annotations. The user can specify a formula interface similar to that used in predictive models and aggregation commands, thus allowing the user to leverage previous R knowledge.

3.6.3 ggplot2

ggplot2 <https://ggplot2.tidyverse.org/> is a relative new comer that is written as an implementation of Leland Wilkinon’s “Grammar of Graphics” which attempts to decompose a visualization into semantic components designed to facilitate a better understanding of the data. By discussing the visualization in terms of accepted vocabulary, the *ggplot2* package enables flexible exploration of data without the need to commit to a fixed set of specific chart types unless desired. *ggplot2* is part of the “tidyverse” (<http://tidyverse.org>) which is a collection of R packages that share a common set of design philosophies, grammar, and data structures that seek to simplify the management, organization, and display of data. However, the *ggplot2* package can be used independently of the larger *tidyverse* package.

4 Git for Version Control

Data science often involves writing scripts and software programs, which are designed to perform discrete tasks during the exploratory phases of the effort. As the needs of the project become more apparent, these scripts are frequently updated and grow in complexity which increases the likelihood of software errors (“bugs”) being introduced at which point the developer must then locate the last known functional version and also address the bug impacting the latest version. This can be difficult given that folders can contain files with arbitrary names which then necessitates the examination of multiple files to locate the error free version. The difficulty in this scenario can be compounded by the introduction of collaborators and additional analysts seeking to participate in the effort. Also consider that software developed in support of an eventual publication should have a transparent trackable version history to facilitate resolution of reported bugs which in turn can enhance reproducibility of published results.

4.1 *Git*

Git is an open source tool for managing revisions to a set of files that experience change over time, that is, version control. Many tools for version control have been developed in the past few decades, but Git has become arguably the most popular. It was originally developed in 2005 by Linus Torvalds to manage the development of the Linux kernel. While Git can work with a general set of text files, it is more commonly employed as part of collaborative software development projects to track modifications to a “repository” (a collection of files within a folder structure). Even If there is no intent for public distribution, the developer benefits from the ability to revert to previous versions or create new “branches” for novel development without impacting the reference branch. Git follows a distributed model that allows developers to “clone” repositories, along with any associated change history, and submit modifications for subsequent (re)integration into the reference copy, commonly known as the “master branch.” Git can assist in the detection and resolution of conflicts wherein changes to a file are made by multiple people.

According to the 2018 Stack Overflow Developer Survey, 87.2% of respondents who use version control in their projects use Git. It is ideal for managing large, distributed projects involving hundreds of participants though it is also appropriate for the laboratory-based informaticist developing code in support of a publication. Git is language agnostic in that it views source code as simple text files to be monitored for changes. The functionality of Git is not a function of the selected programming language.

4.2 Availability and Installation

Operating system specific installers are available at <https://git-scm.com/downloads>. The clients provide a built-in GUI tool although the method of interaction described in this text will relate to the command line client available which is available via the Terminal application in Apple Mac OS and Linux. The Windows install also comes with a git shell that can be launched as a standard application.

4.3 A Common Git Workflow

There are formal methodologies for software development projects although use of Git does not involve or require the adoption of any specific approach. A common generic development workflow using git involves the following steps:

1. Create a repository using “git init” or “git clone.”
2. Create/modify files in the working repository using a text editor.
3. Add file(s) to be tracked using “git add.”
4. Use “git status” frequently to see the current state of files and commits.
5. When the tracked files represent a logical point of progress then commit changes using “git commit” or “git commit -a” for subsequent commits.

Repeat steps 2–5 until the project is complete or at a logical stopping point. There are many possible variations to this workflow though it represents a useful starting point.

4.4 Key Concepts

The following represent essential ideas for becoming productive with Git. Each concept will be considered in detail.

Repository: A folder that has been placed under git control.

Origin: Every repository has an “origin,” which could be local (on a hard drive) or a remote hosting service such as Github or GitLab.

Branch: Each repository has a reference or “master” branch from which copies can be made for purposes of experimentation, testing, and bug fixing without impacting the availability of the master branch.

Tracking: The activity of monitoring files for changes.

Commit: The activity of saving or taking a snapshot of existing modifications. Each project will usually experience multiple commits over time.

4.5 Getting Started

Git requires some basic identity information such as e-mail address although this does not have to be an actual working address. Consider though that this information is what Git will use in the meta data of the repository so should it be distributed in the future, end users will have appropriate contact information. Launch a

terminal on Apple Mac OS or Linux or the Git Shell on Microsoft windows. This will display a window containing a shell prompt that typical includes a “\$” prompt or some variation thereof.

```
/home/ubuntu/MyProject $ git config --global user.email "john-doe@doe.com"
/home/ubuntu/MyProject $ git config --global user.name "John Doe"
```

4.6 Create a Repository

Git repositories can be created from existing folders or new ones. The following commands demonstrate the creation of a new folder and repository. Note that there are no files to manage just yet.

```
/home/ubuntu $ mkdir MyProject
/home/ubuntu $ cd MyProject/
/home/ubuntu/MyProject $ git init
Initialized empty Git repository in /home/ubuntu/MyProject/.git/
/home/ubuntu/MyProject $ ls
/home/ubuntu/MyProject $
/home/ubuntu/MyProject $ ls -a      # List hidden files
. .. .git
```

It is important to understand that working inside of a git repository is no different than working in any other folder. Git does not attempt to validate any code you develop nor does it alter any user created files. However, git does manage the content of the .git folder that is created by the “git init” command. This folder exists to capture all the metadata for the repository. On Linux and Apple systems any file beginning with a period character is generally hidden to prevent accidental deletion. The presence of the .git folder is what makes the MyProject folder an actual git repository. Removing the .git folder would not remove any data files, but any revision control information would be lost.

4.7 Adding Content

Next, create some files using a text editor (e.g., vi, emacs).

```
/home/ubuntu/MyProject $ vi Readme.md
/home/ubuntu/MyProject $ vi regression.R
/home/ubuntu/MyProject $ cat Readme.md

# MyProject
```

This is the readme file for this project.

More details will follow

```
/home/ubuntu/MyProject $ cat regression.R
data(mtcars)
mylm <- lm(mpg~., data=mtcars)
summary(mylm)
```

Determine the status of these files according to git:

```
/home/ubuntu/MyProject $ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be
committed)

  README.md
  regression.R

nothing added to commit but untracked files present (use "git
add" to track)
```

The results of the “status” command indicate that the two new files are not currently being tracked. It also provides information on how to initiate tracking which means that git will monitor the files for changes. After adding the file, enter the “git status” command which will show that the files are now “staged” for a “commit.” Additional changes can be made and git will track them.

```
/home/ubuntu/MyProject $ git add *
/home/ubuntu/MyProject $ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

    new file:   README.md
    new file:   regression.R
```

4.8 Making a Commit

Making a commit in git will take a “snapshot” of all changes and register them. This is accomplished using the “git commit” command. For an active development project, commits should be made frequently. Remember that a benefit of the git system is that any commit can be reverted should it result in a software error.

```
$ /home/ubuntu/MyProject $ git commit
```

This will invoke the default editor associated with the current user account. For Ubuntu this will be “nano.” The following screen illustrates a typical commit message which should be informative.

```
This is my first commit
# Please enter the commit message for your changes. Lines
starting
# with '#' will be ignored, and an empty message aborts the
commit.
#
# On branch master
#
# Initial commit
#
# Changes to be committed:
#       new file:   Readme.md
#       new file:   regression.R
#
```

After exiting from the commit editor the following conformation of the commit activity can be observed:

```
[master (root-commit) f025b99] This is my first commit
2 files changed, 7 insertions(+)
create mode 100644 Readme.md
create mode 100644 regsession.R
```

Next, follow this up with a “git status.” Git indicates that all changes have been noted and committed to the master branch. At this point, more files can be created or new edits can be made to the existing files followed by more commits.

```
/home/ubuntu/MyProject $ git status
On branch master
nothing to commit, working tree clean
```

The “git log” provides a summary of the commit activity.

```
/home/ubuntu/MyProject $ git log
commit f025b99c53def7b6216d92d8113982d387610615 (HEAD -> master)
Author: John Doe <johndoe@doe.com>
Date:   Tue Jun 18 19:35:24 2019 +0000

This is my first commit
```

An alternative form of the log command provides a more compact summary. Note that the number to the left is a “hash number” that uniquely identifies each commit which simplifies the process of retrieving or reverting to a previous commit.

```
/home/ubuntu/MyProject $ git log --pretty=oneline
f025b99c53def7b6216d92d8113982d387610615 (HEAD -> master)
This is my first commit
```

4.9 Inspecting Changes to Files

This example demonstrates what happens when modifications are made to a file that is currently being tracked for changes but has yet to be committed. This is accomplished using the “git diff” command. At this point, neither of the two files in the repository have been modified since the commit.

```
/home/ubuntu/MyProject $ git diff
/home/ubuntu/MyProject $
```

Modify the regression.R file which currently looks like:

```
data(mtcars)
mylm <- lm(mpg~., data=mtcars)
summary(mylm)
```

Here are the changes.

```
# Predict the mpg of a car
data(mtcars)
mylm <- lm(mpg~hp+wt, data=mtcars)
summary(mylm)
```

The “git difference” command will note the recent changes which in this case are (1) the addition of a new line at the top and (2) a change in the line containing the call to the R lm() function. A full discussion of the change format is beyond the scope of this text though it is apparent that a new line 1 was added and the new line 3 (line 2 in the previous version) was modified.

```
/home/ubuntu/MyProject $ git diff
diff --git a/regression.R b/regression.R
index cffffd6c..b7db2aa 100644
--- a/regression.R
+++ b/regression.R
@@ -1,3 +1,4 @@
+# Predict the mpg of a car
  data(mtcars)
-mylm <- lm(mpg~.,data=mtcars)
+mylm <- lm(mpg~hp+wt,data=mtcars)
 summary(mylm)
```

Following this up with a call to “git status” confirms that a change has been made and the file is staged for a subsequent commit although it is not required. More changes can be made to file including file deletions or additions.

```
/home/ubuntu/MyProject $ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
    (use "git checkout -- <file>..." to discard changes in
     working directory)

          modified:   regression.R

no changes added to commit (use "git add" and/or "git commit
-a")
```

At this point, apply a commit to take a snapshot of the current changes. Note that output of the git status command indicates that the “git commit -a” option may be used to update what will be committed while the actual commit is taking place.

```
/home/ubuntu/MyProject $ git commit -a

Added a descriptive header line and changed lm predictor
variables
```

```

# Please enter the commit message for your changes. Lines
starting
# with '#' will be ignored, and an empty message aborts the
commit.
#
# On branch master
# Changes to be committed:
#       modified:   regression.R
#
[master 457c2ee] Added a descriptive header line and changed lm
predictor variables
1 file changed, 2 insertions(+), 1 deletion(-)

```

Now check the status and commit event log to confirm that the master branch is up to date.

```

/home/ubuntu/MyProject $ git status
On branch master
nothing to commit, working tree clean

/home/ubuntu/MyProject $ git log
commit 457c2ee2a189af1f067bc2d82eb01b01f4d9f206 (HEAD -> mas-
ter)
Author: John Doe <johndoe@doe.com>
Date:   Tue Jun 18 20:42:27 2019 +0000

        Added a descriptive header line and changed lm predictor
variables

commit f025b99c53def7b6216d92d8113982d387610615
Author: John Doe <johndoe@doe.com>
Date:   Tue Jun 18 19:35:24 2019 +0000

This is my first commit

```

4.10 Using Branches

An attractive feature of Git is the ability to create copies of the master branch into arbitrarily named experimental branches without impacting the master branch code. Think of any new branches as being derivative safe copies of previously committed code which can be managed using the usual git command set. Changes to a derivative branch can, if desired, be integrated back into the master branch or remain in a separate branch for more testing. The “git branch” command indicates any existing branches and places an

asterisk next to the currently active branch which in this case is the “master” branch.

```
/home/ubuntu/MyProject $ git branch
* master
/home/ubuntu/MyProject $
```

This example will create a branch called “experiment” although the current branch is still the master branch. The “git branch <branchname>” creates a new branch name “branchname.”

```
/home/ubuntu/tester $ git branch experiment
/home/ubuntu/tester $ git branch
experiment
* master
```

The command “git checkout experiment” will checkout the contents of the master branch into the branch named “experiment” and make “experiment” the current branch:

```
/home/ubuntu/tester $ git checkout experiment
Switched to branch 'experiment'
/home/ubuntu/tester $ git branch
* experiment
  master
```

The only thing that has changed is the current branch. The underlying folder contents are the same. Create a new file called “logistic_regression.R” with the following contents and commit the change with a commit message of “Added a file for logistic regression.”

```
# Logistic Regression Example
mtcars$am <- factor(mtcars$am)
myglm <- glm(am~.,data=mtcars)
summary(myglm)

/home/ubuntu/tester $ git add logistic_regression.R
/home/ubuntu/tester $ git commit
[experiment 3c8a02f] Added a file for logistic regression
1 file changed, 4 insertions(+)
create mode 100644 logistic_regression.R

/home/ubuntu/tester $ ls
```

```
Readme.md logistic_regression.R regression.R
```

Remember that the new file has been added within the experimental branch and will NOT impact the **master** branch. To verify this, switch back to the master branch by using the “git checkout master” command.

```
/home/ubuntu/tester $ git branch
* experiment
  master
/home/ubuntu/tester $ git checkout master
Switched to branch 'master'

/home/ubuntu/tester $ git branch
  experiment
* master

/home/ubuntu/tester $ ls
Readme.md regression.R
```

Observe that the **master** branch does not include the `logistic_regression.R` file since it exists only within the **experiment** branch. It is possible to merge the changes made in the **experiment** branch into the **master** branch by using the “git merge” command.

```
/home/ubuntu/tester $ git branch
  experiment
* master

/home/ubuntu/tester $ git merge experiment
Updating 457c2ee..3c8a02f
Fast-forward
 logistic_regression.R | 4 +++
 1 file changed, 4 insertions(+)
 create mode 100644 logistic_regression.R

/home/ubuntu/tester $ ls
Readme.md logistic_regression.R regression.R
```

Once the merge is complete the experiment branch can be removed.

```
/home/ubuntu/tester $ git branch
  experiment
* master
```

```
/home/ubuntu/tester $ git branch -d experiment
Deleted branch experiment (was 3c8a02f).

/home/ubuntu/tester $ git branch
* master
```

4.11 Sharing Repositories

Using git is an effective way to manage any number of development projects in a way that preserves changes made during the project lifecycle which enhances reproducibility and serves as documentation of the project’s development trajectory. The examples presented thus far use a locally created repository which could be shared with other users although that is optional. However, it is common for developers to share projects using a hosting service such as Github that supports over 36 million developers and 100 million repositories. Ref <https://github.com/> Github offers a free hosting plan for unlimited public and private repositories, issue and bug tracking, and project management tools. Proceed to <https://github.com/join?source=header-home> to create an account and select the free plan.

Many newcomers to Git are already familiar with GitHub having been directed to download a repository by a publication or a colleague. Knowledge of git commands is usually not essential to simply using software in a repository although to participate in development efforts does require such knowledge. Login to the previously created Github account and create a New repository by clicking the “+” button in the upper right corner of the Github screen (Fig. 3):

After selecting “New Repository” a screen will be displayed that prompts for information including the desired repository name, an optional description, whether the repository should be public (leave this selected), and whether to initialize it with a README file (leave selected). After providing this information, click the green “Create repository” at the bottom of the screen (Fig. 4):

The next screen will display a summary of the repository from which the green “Clone or Download” button can be clicked. After clicking this button, in the pop-up box the URL for the repository will be shown. Click the clipboard icon to copy the URL.

4.12 Cloning the Repository Locally

As Git is decentralized, a repository can be cloned by any number of parties interested in contributing to a project. To clone the repository locally, open a Terminal and enter the following git command which will download the repository into the current folder.

```
/home/ubuntu $ git clone https://github.com/wsperm/mycoolrepo.
git
```

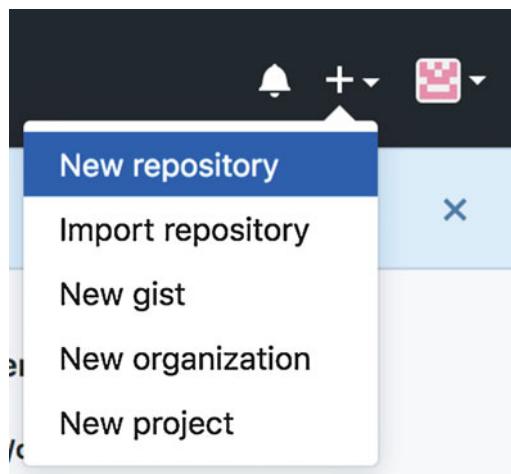


Fig. 3 Creating a new repository in Github

The screenshot shows the 'Create repository' form on GitHub. It includes the following fields and options:

- Owner:** wspem
- Repository name ***: mycoolrepo (with a green checkmark)
- Description (optional):** Repository for Testing
- Visibility:** Public (selected) - Anyone can see this repository. You choose who can commit.
- Visibility:** Private - You choose who can see and commit to this repository.
- Initialize this repository with a README:** (checkbox checked) - This will let you immediately clone the repository to your computer.
- Add .gitignore:** None
- Add a license:** None
- Create repository** button (green)

Fig. 4 Repository information screen on Github

```

Cloning into 'mycoolrepo'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0

```

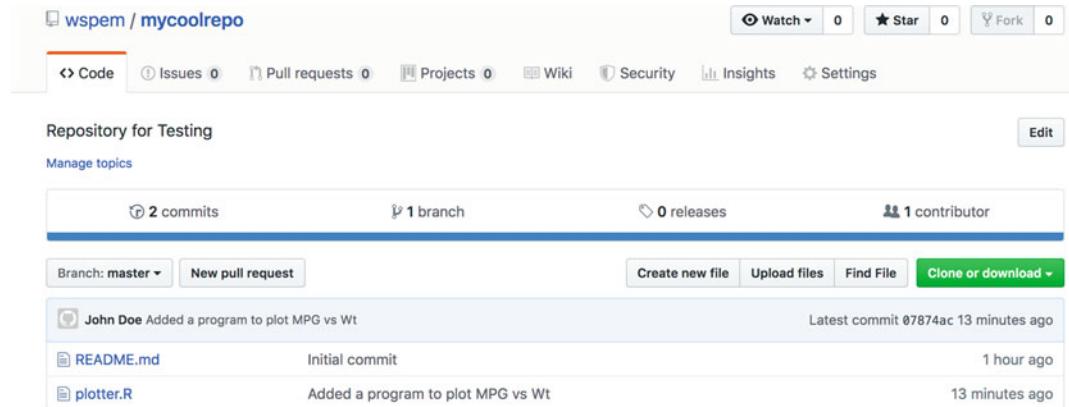


Fig. 5 Github Repository main page

```
Unpacking objects: 100% (3/3), done.
```

```
/home/ubuntu $ ls -a
. .. .git README.md
```

The contents of the repository will match the contents reflected on the Github website. There is only a single README.md file which was created at the same time that the repository itself was created.

```
/home/ubuntu $ cd mycoolrepo/
/home/ubuntu/mycoolrepo $ ls
README.md
```

An important git concept mentioned earlier was that of “origin” which up until now has not been a consideration since the repository created in the previous section was local. However, since the “mycoolrepo” repository was created on Github and then cloned locally, the origin of the repository should reflect that sequence of events. The command “git remote -v” will confirm the correct point of origination of “mycoolrepo.”

```
/home/ubuntu $ cd mycoolrepo/
/home/ubuntu/mycoolrepo $ git remote -v
origin      https://github.com/wsphem/mycoolrepo.git (fetch)
origin      https://github.com/wsphem/mycoolrepo.git (push)
```

Files can still be added, modified, or deleted just as with the earlier local repository so create a new file called plotter.R with the following contents.

```
# Program to plot MPG vs Wt
data(mtcars)
plot(mpg~wt,data=mtcars)
```

Now use “git status”.

```
/home/ubuntu/mycoolrepo $ git status
On branch master
Your branch is up to date with 'origin/master'.

Untracked files:
  (use "git add <file>..." to include in what will be
committed)

      plotter.R

nothing added to commit but untracked files present (use "git
add" to track)
```

This is a familiar result based on the earlier work with the local MyProject repository. At this point add and commit the file and then check the status.

```
/home/ubuntu/mycoolrepo $ git add plotter.R
/home/ubuntu/mycoolrepo $ git commit
git commit
[master 07874ac] Added a program to plot MPG vs Wt
  1 file changed, 4 insertions(+)
  create mode 100644 plotter.R

/home/ubuntu/mycoolrepo $ git status
On branch master
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
```

The message indicates that the commit worked though given that a new file has been added the local version of the repository, it is now “ahead” of the origin repository currently residing on Github. Work could continue and commits could be made but this local copy of the mycoolrepo repository will be out of sync with the version on Github. There is a command that will allow one to “push” these changes to the master branch located at the origin which is

```
/home/ubuntu/mycoolrepo $ git remote -v
origin    https://github.com/wspem/mycoolrepo.git (fetch)
origin    https://github.com/wspem/mycoolrepo.git (push)
```

Push up the changes. This will require knowing the user id and password for the Github account created earlier.

```
/home/ubuntu/mycoolrepo $ git push origin
Username for 'https://github.com': wspem
Password for 'https://wspem@github.com':
Counting objects: 3, done.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 347 bytes | 347.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/wspem/mycoolrepo.git
  e61a151..07874ac  master -> master

/home/ubuntu/mycoolrepo $ git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```

Refer back to the web page associated with the Github repository which in this example is <https://github.com/wspem/mycoolrepo>. Note that the file plotter.R that was added locally is now present in the Github repository (Fig. 5).

4.13 Next Steps

This walkthrough has provided an overview of Git and Github basics although there remains a significant degree of capability that cannot be covered due to space constraints. Understanding how to manage software conflicts and work with repositories owned by other developers is also important information. In any case, the material presented in this section represents a solid basis upon which laboratory-based developers can efficiently build reproducible software projects.

5 The Docker Container

A software application of any real-world complexity will depend on other software and libraries to function. These software and libraries will further depend on computing environments including services from the operating systems. Some of those may be available only on certain operating systems (e.g., on Linux but not Windows). The access to multiple operating systems, without

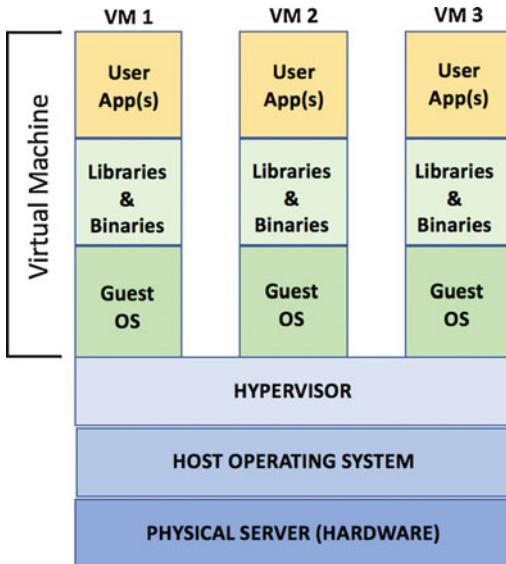


Fig. 6 Diagram of virtualized systems

purchasing a new computer, can be provided by the use of virtual machines. For example, *VirtualBox* allows Linux to run on a local Windows system. In this example the following terms would apply (Fig. 6):

Server: The actual physical hardware.

Host OS: The operating system installed on the Server (Microsoft Windows).

Hypervisor: The software which enables virtualized services (e.g., Virtual Box).

Guest OS: The operating system installed within a virtualized environment (Linux).

Use of a hypervisor allows for one or more guest operating systems, assuming the server hardware provides adequate storage and memory resources. The guest operating system must be installed including any supporting system and applications libraries. The hypervisor provides access to the functioning virtual computers, and corresponding operating systems, each of which can access the local system's hardware (CD/DVD, mouse, USB ports, etc.). Figure 2 shows an Apple laptop running *VirtualBox* that is used to host a copy of Mint Linux. Virtualized systems are particularly useful in situations that require a significant number of applications which justifies the effort involved in creating and configuring a virtual machine along with the supporting OS and application libraries (Fig. 7).

A disadvantage of virtual machines is the resource requirement (CPU cycles, memory and disk space) on the host computer. In effect, the host system can become over-subscribed. A second disadvantage is it assumes the user has sufficient knowledge to

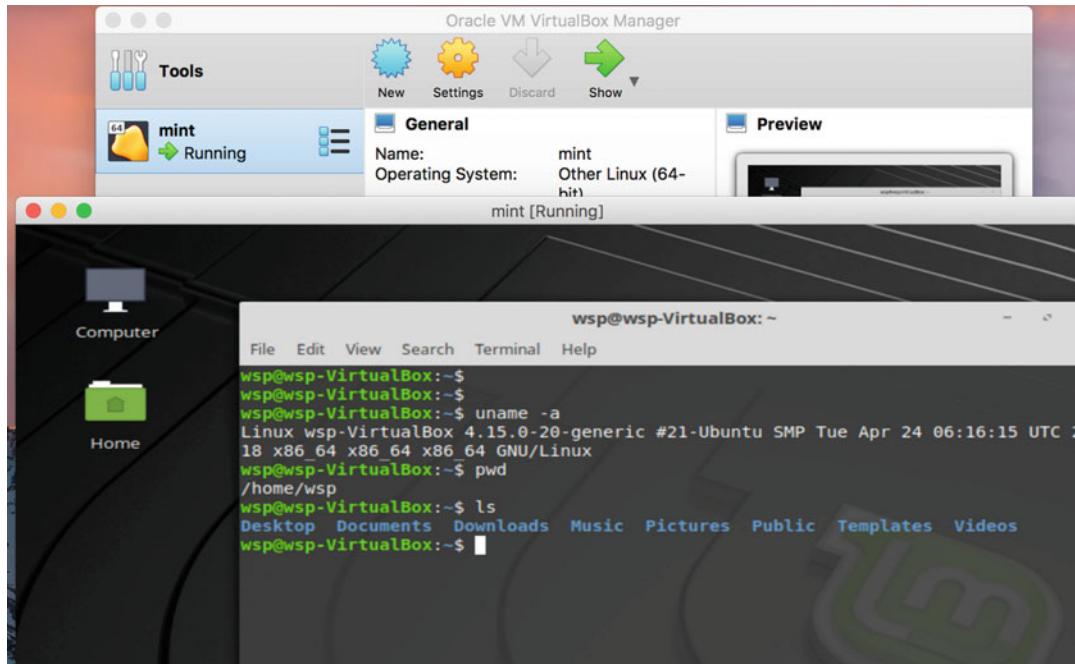


Fig. 7 Linux Mint running in VirtulBox on Apple Mac OS

install and manage the necessary operating systems and application packages. Many times, a project requires a single application in which case, using a virtual machine represents more cost than it is worth. A lighter alternative technology is containers, and the best example of containers is Docker.

Another reason of using containers, perhaps more important to data science, is the dependency/compatibility issue of computer software. The libraries that a software application depends on will change over time and have many versions, and some be out of maintenance after a while. Therefore, a software application often fails to run because its dependency on those libraries breaks. The solution is to have a snapshot of the exact versions and copies of these libraries that the software application is created to run on (which is also important to the reproducibility of science). Even though one could create a virtual machine image for this, but to do it regularly as part of daily routine, containers are the answer. In practice, containers often reduce the complexity of installing or deploying a software application.

5.1 *Installing Docker*

Free installers for Apple Mac OS and Microsoft Windows are available from Docker Hub which requires the creation of an account. The install provides both the Docker daemon and client CLI command line interface.

Microsoft Windows—<https://docs.docker.com/docker-for-windows/install/>.

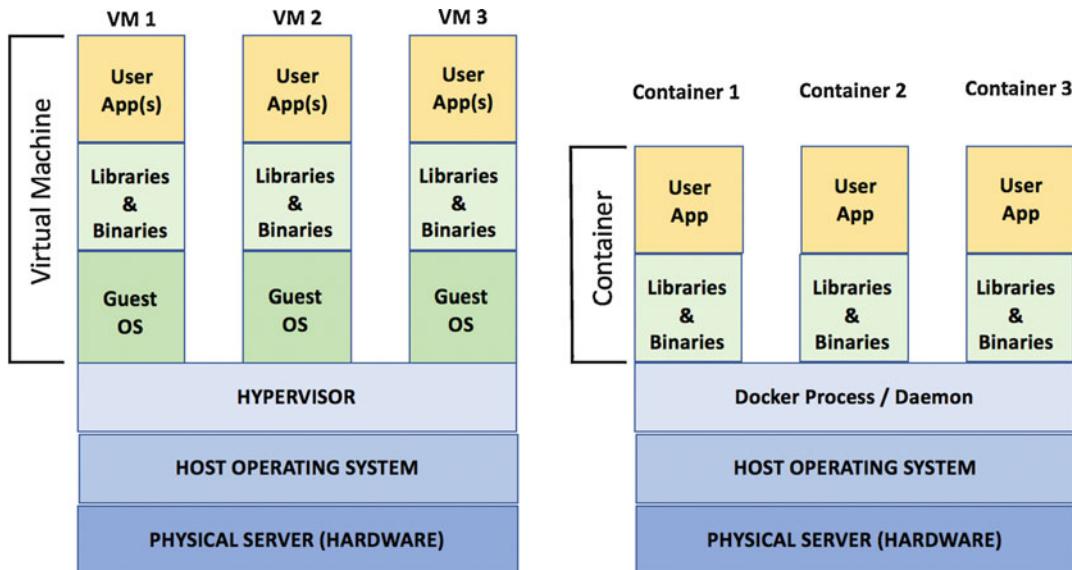


Fig. 8 Comparing a virtual machine and docker

Apple Mac OS—<https://docs.docker.com/docker-for-mac/install/>.

Linux—Refer to package repository for the specific distribution.

After the installation of Docker the “hello-world” image can be used to validate the installation.

```
$ docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
1b930d010525: Pull complete
Digest: sha256:41a65640635299bab090f783209-
c1e3a3f11934cf7756b09cb2f1e02147c6ed8
Status: Downloaded newer image for hello-world:latest
```

Hello from Docker!

This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:

1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
(amd64)
3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.

4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

To try something more ambitious, you can run an Ubuntu container with:

```
$ docker run -it ubuntu bash
```

Share images, automate workflows, and more with a free Docker ID:

<https://hub.docker.com/>

For more examples and ideas, visit:

<https://docs.docker.com/get-started/>

In the Docker domain, the concept of a virtual machine is replaced by a “container” which employs an “image” of a specific tool or application, as well as any dependencies, that can be executed under the control of a Docker process running on the individual’s computer. Docker containers can be executed in isolation although it is easy to enable communication between containers to create services from independent components. The differences between a Virtual Machine and a Docker setup can be observed in the following figure. Note the absence of a guest operating system in the Docker architecture. Also, the Hypervisor is replaced by a Docker process that manages containers and image (Fig. 8).

5.2 Key Docker Concepts

From an end user point of view, the terminology for Docker involves a basic knowledge of the following ideas:

Client/CLI—This is the CLI (command line interface) that allows the user to issue various docker commands. The client usually runs on the same host as the Docker daemon process. The CLI offers a suite of commands for managing containers, images, volumes, and networking between containers.

Docker Process/Daemon—This is the persistent process that handles client requests, checks with the registry, and manages containers. This usually runs on the same host as the Docker client.

Container—A packaging construct designed to insulate docker images from the environment in which they will execute. A container can be thought of as an actively running image with which the end user can interact.

Image—A collection of bundled dependencies necessary to execute an application. Base images are those not reliant upon other images. Examples are operating systems (e.g., Ubuntu, Debian). Child images are those created on top of Base images.

Docker Registry—A registry is a collection of images that can be searched and deployed. Consider it as a reference point for locating and initiating containers. An example is Docker Hub <https://www.docker.com/products/docker-hub>.

```
$ docker search rocker --limit 10
NAME          DESCRIPTION          STARS      OFFICIAL      AUTOMATED
rocker/rstudio    RStudio Server image  257       [OK]
rocker/shiny      [OK]
rocker/tidyverse   Version-stable build of R, rstudio, and R pa...  71       [OK]
rocker/r-base      Basic R for Rocker And Official 'r-base'  66       [OK]
rocker/verse       Adds tex & related publishing packages to ve...  32       [OK]
rocker/r-ver       Reproducible builds to fixed versions of R  24       [OK]
rocker/geospatial  Docker-based Geospatial toolkit for R, built...  20       [OK]
rocker/rstudio-stable Build RStudio based on a debian:stable (debi...  14       [OK]
rocker/shiny-verse  Rocker Shiny image + Tidyverse R packages. U...  7        [OK]
rocker/binder      Adds binder to rocker/tidyverse, providing J...  1        [OK]
$
```

Fig. 9 The search command in docker

5.3 Finding Useful Images

There are a number of prebuilt images relating to bioinformatics and supporting languages including Python and R. For example, the Rocker project <https://www.rocker-project.org/> offers Docker Containers for the R Environment as well as instructions on how to run the images. Image discovery is straightforward by using the search capability which is part of the default installation of Docker. The “search” command is used to locate images of interest. Note also that there is a web-based search tool Located at <https://hub.docker.com/search?q=rocker&type=image> that offers an intuitive user interface (Fig. 9)

The “docker search” command has a number of options that can help filter output though the basic format given above is typical. The NAME and DESCRIPTION files document the image label and any supplied information. The STARS field is a community rating of the image which is how the output is sorted. Once the desired image is found it is simple to execute it. Below is an example to run “rocker/r-base.”

```
$ docker run -it rocker/r-base
Unable to find image 'rocker/r-base:latest' locally
latest: Pulling from rocker/r-base
2666d10a4f80: Pull complete
2c0f31f3b229: Pull complete
8978e71a606b: Pull complete
3a18d5b41e17: Pull complete
3b9876199949: Pull complete
1ecd21a8af49: Pull complete
Digest: sha256:35099e4073c8aff2c616add01d7ffd452bf79c78a893-
b08a9ac30e8114cab247
Status: Downloaded newer image for rocker/r-base:latest

R version 3.6.0 (2019-04-26) -- "Planting of a Tree"
Copyright (C) 2019 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
```

```
Type 'license()' or 'licence()' for distribution details.
```

Natural language support but running in an English locale

R is a collaborative project with many contributors.

Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.

Type 'q()' to quit R.

>

The command “`docker run -it rocker/r-base`” will (1) contact the local Docker process/daemon to see if there is a local image named “`rocker/r-base:latest`”. If not found, the Docker process can be configured to (2) consult a registry to determine the existence of the image. Once located, the Docker process will (3) “pull” (download) the image along with any supporting images. The Docker process will (4) then start a container on the host which the user can then engage to experiment with the new metabolomics package without impacting the version of R residing on the laptop. Once the experiment has concluded the user can stop the container, by exiting R or using “`docker stop`,” and remove the image if desired or develop a new analysis pipeline that could then be converted to an image (Fig. 10).

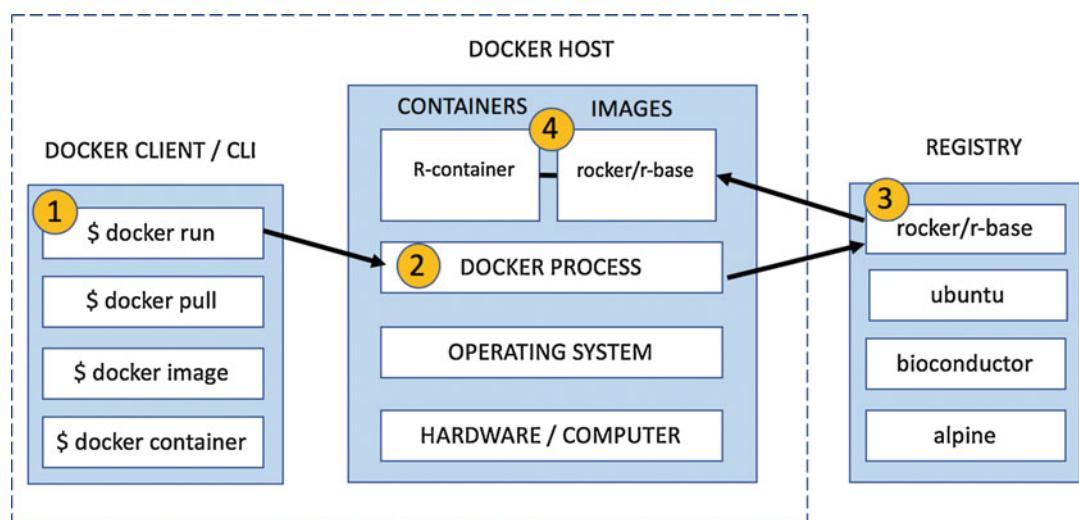


Fig. 10 Workflow of a command run on docker

5.4 Managing Containers and Images

A notable side effect of running containers is that any associated image might require additional image layers that contribute to the overall download size. Images remain on the local hard drive unless explicitly removed. The Docker client includes commands to manage containers and images as well as tools to create them. Start the following container using the `ubuntu:latest` image command.

```
$ docker run -it --name=local_ubuntu ubuntu:latest
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
5b7339215d1d: Pull complete
14ca88e9f672: Pull complete
a31c3b1caad4: Pull complete
b054a26005b7: Pull complete
Digest: sha256:9b1702dcfe32c873a770a32cf306dd7fc1c4f-
d134adf8783db68defc8894b3c
Status: Downloaded newer image for ubuntu:latest
root@06949ec991be:/#
```

The “`-it`” options indicate a requested interactive session with the container via a terminal / command line prompt. The “`--name = local_ubuntu`” is an optionally assigned label for the container which simplifies identification should there be other active containers on the system.

The `docker “ps”` command will show all resident containers. The “`-a`” option shows inactive containers also. From another terminal / command line, enter \$ `docker ps -a` (Fig. 11).

Notice that the container has a “CONTAINER ID” assigned by the Docker process although the more intuitive name “`local_ubuntu`” may also be used to identify and manage the container. The container can be stopped using the Docker CLI via “stop” (Fig. 12).

Stopping the container does not remove it from the system. If a later attempt is made to restart it via the `docker run` command then the following would occur:

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
c4ec51577936	ubuntu:latest	"/bin/bash"	9 seconds ago	Up 9 seconds		local_ubuntu

Fig. 11 Using the `ps` command

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
c4ec51577936	ubuntu:latest	"/bin/bash"	9 seconds ago	Up 9 seconds		local_ubuntu

Fig. 12 Using the `stop` command

```
$ docker run -it --name=local_ubuntu ubuntu:latest
docker: Error response from daemon: Conflict. The container name "/local_ubuntu" is already in use by container "c4ec515779368cb14d4b9d79fac86d53e4bb018125277eaf6a-b65e9438d89814". You have to remove (or rename) that container to be able to reuse that name.
See 'docker run --help'.
```

To remove the `local_ubuntu` container from the system (Fig. 13).

The container has been removed; however, the supporting image still remains locally. The “`rmi`” command can be used to delete the local image (Fig. 14):

In general, it is not necessary to remove containers and images from the system unless disk space becomes an issue in which case these approaches can be used to perform cleanup. One way to prevent a container from persisting locally is to specify the “`-rm`” option when running the `docker`. After the container is stopped, the container will be deleted. Run the `ubuntu:latest` image as before:

```
$ docker run -it --rm --name=local_ubuntu ubuntu:latest
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
5b7339215d1d: Pull complete
14ca88e9f672: Pull complete
a31c3b1caad4: Pull complete
b054a26005b7: Pull complete
Digest: sha256:9b1702dcfe32c873a770a32cf306dd7fc1c4f-
```

\$ docker container rm local_ubuntu						
CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
S						

Fig. 13 Using the `rm`. command to remove Ubuntu container

```
$ docker images -a
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
ubuntu          latest   4c108a37151f    44 hours ago  64.2MB
$ docker rmi 4c108a37151f
Untagged: ubuntu@sha256:9b1702dcfe32c873a770a32cf306dd7fc1c4fd134adfb783db68defc8894b3c
Deleted: sha256:4c108a37151f54439950335c409802e948883e00c93fdb751d206c9a9674c1f6
Deleted: sha256:7c1abf1dbbf02a48330a7317ab45a6091d53e2e9cc062f0f3dbd2b7539947a6
Deleted: sha256:5a614dda4a54650168ee2cd30ce2e39576dad5c9a0d1907c02445687b4ea5090
Deleted: sha256:bd042113a73a5c9c6680990740446b7324afb39e243ade3d33bdaa9ffaf8d294
Deleted: sha256:ba9de9d8475e7f5e40086358a1353b3cc080994fc6d31e4272dd3acb69b0151e
$
```

Fig. 14 Using `rmi` command to delete local image

```
$ docker images -a
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
ubuntu          latest   4c108a37151f  44 hours ago  64.2MB
$ docker ps -a
CONTAINER ID    IMAGE      COMMAND      CREATED      STATUS      PORTS      NAMES
ee1ef120852d   ubuntu:latest "/bin/bash"  2 minutes ago  Up 2 minutes           local_ubuntu
```

Fig. 15 Using the ps command to verify the existence of the container and image

```
$ docker images -a
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
ubuntu          latest   4c108a37151f  44 hours ago  64.2MB
$ docker ps -a
CONTAINER ID    IMAGE      COMMAND      CREATED      STATUS      PORTS      NAMES
$ docker ps -a
```

Fig. 16 Using the ps command to verify the existence of image but not the container

```
$ docker search rocker --limit=5
NAME          DESCRIPTION          STARS      OFFICIAL      AUTOMATED
rocker/rstudio RStudio Server image 257        [OK]
rocker/tidyverse Version-stable build of R, rstudio, and R pa... 71        [OK]
rocker/r-base   Basic R for Rocker And Official 'r-base'       66        [OK]
rocker/geospatial Docker-based Geospatial toolkit for R, built... 20        [OK]
rocker/rstudio-stable Build RStudio based on a debian:stable (debi... 14        [OK]
$
```

Fig. 17 Rocker images which support the R language

```
d134adfb783db68defc8894b3c
Status: Downloaded newer image for ubuntu:latest
root@ee1ef120852d:/#
```

From another Terminal window use images and ps commands to verify the existence of the container and image (Fig. 15):

Back in the ubuntu:latest shell, type the word “exit” to end the container session:

```
root@ee1ef120852d:/# exit
exit
```

Next, verify that the container has not been maintained although the image has (Fig. 16).

5.5 A More Practical Use Case

As seen earlier, the rocker/r-base image offers command line access to the R environment although the rocker project offers images in support of the R language (Fig. 17) .

The rocker/rstudio image in particular provides R language support along with the RStudio IDE which in this case is a web-based service accessible locally via a browser. Because of this, some options will need to be supplied to appropriately map the RStudio port (typically 8787) to a local system port which can also be 8787 assuming no other local service is using it. The RStudio IDE also requires the specification of a password at run time. The

default user is “rstudio.” This is how the associated docker run command would look:

```
$ docker run -e PASSWORD="testpass" --name=local_rstudio -p 8787:8787 rocker/rstudio

[s6-init] making user provided files available at /var/run/s6/etc...exited 0.
[s6-init] ensuring user provided files have correct perms...
exited 0.
[fix-attrs.d] applying ownership & permissions fixes...
[fix-attrs.d] done.
[cont-init.d] executing container initialization scripts...
[cont-init.d] add: executing...
Nothing additional to add
[cont-init.d] add: exited 0.
[cont-init.d] userconf: executing...
[cont-init.d] userconf: exited 0.
[cont-init.d] done.
[services.d] starting services
[services.d] done.
```

The “-e” option will set the PASSWORD within the container. Note that this command will be running in attached mode which means that the terminal window in which it was entered will be blocked/engaged until the container is terminated. This is not necessarily a problem since the point of user interaction will be the RStudio login panel. In order to run `rocker/rstudio` as a detached container, include the “-d” option which will leave the terminal free for input.

Next, launch a web browser and provide a URL of `http://localhost:8787`. This will bring up the RStudio login page. Provide a user name of “rstudio” and a password of “testpass” (Fig. 18).

After clicking the “Sign In” button, the RStudio IDE will be presented for engagement. At this point the user can install any additional packages and create code as with any instance of R (Fig. 19).

5.6 Sharing Data

Remember that the container is running under the supervision of the Docker process and the containerized R environment will in no way impact any locally existing versions of R. However, a problem with this specific example is that there is nowhere to save code and have it persist except within the container itself. For example, a user might have existing R code in a local folder (e.g., `metabolomics_r_code`) intended for use with the container. Or, in working with the container, the user might want to save code to the local

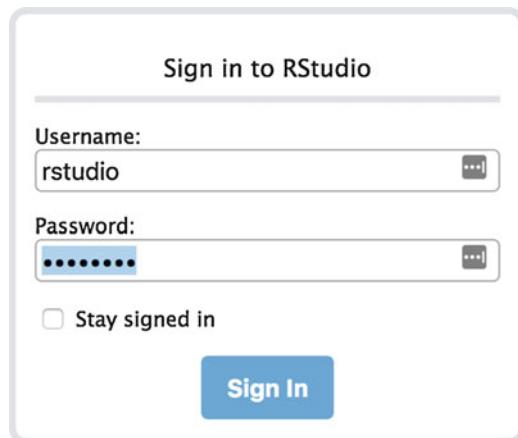


Fig. 18 Login panel for Rocker/R-studio

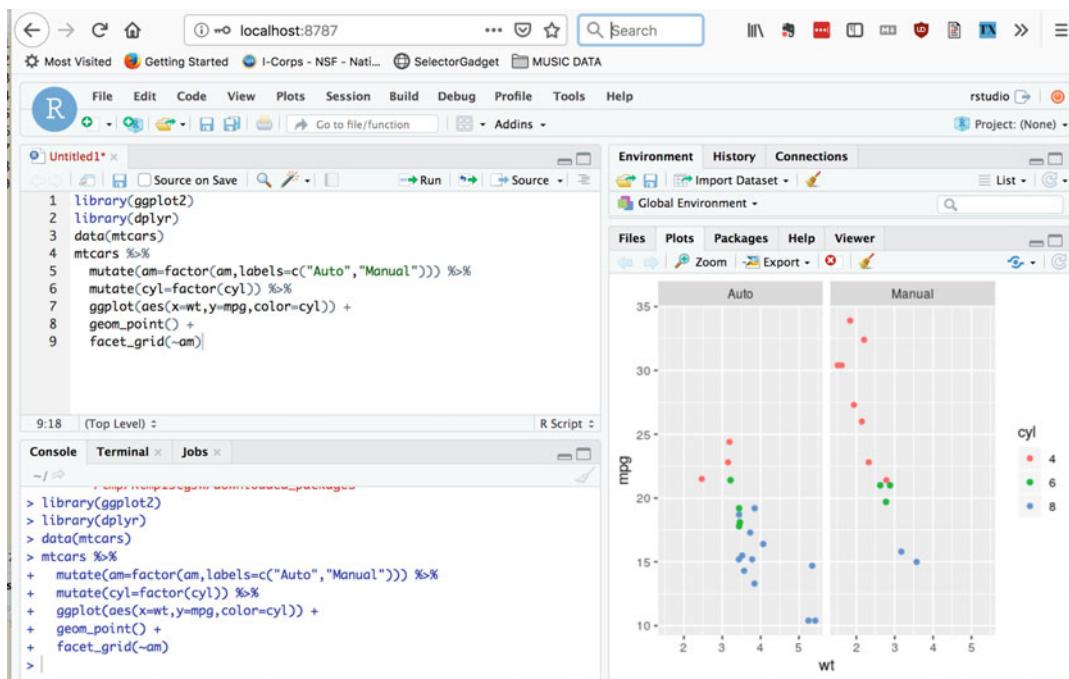


Fig. 19 Rocker/R-studio container

system outside of the container or with future containers. Docker has a solution for this in the form of volume mapping which can be specified on the run command line. Stop the running rocker/verse instance either by typing Control-C in the terminal window used to launch the container or by using “`docker stop`.”

The user wants to expose the `metabolomics_r_code` folder to the `rocker/rstudio` container. The folder resides within `/Users/esteban`.

```
$ cd ~
$ cd metabolomics_r_code/
$ pwd
/Users/esteban/metabolomics_r_code
$ ls
linear_regression.R logistic_regression.R support_vector_
machine.R

$ docker run -e PASSWORD="testpass" --name=local_rstudio_vol
-v /Users/esteban/metabolomics_r_code:/home/rstudio/metabolomics_r_code -p 8787:8787 rocker/rstudio

[s6-init] making user provided files available at /var/run/s6/etc...exited 0.
[s6-init] ensuring user provided files have correct perms...
exited 0.
[fix-attrs.d] applying ownership & permissions fixes...
[fix-attrs.d] done.
[cont-init.d] executing container initialization scripts...
[cont-init.d] add: executing...
Nothing additional to add
[cont-init.d] add: exited 0.
[cont-init.d] userconf: executing...
[cont-init.d] userconf: exited 0.
[cont-init.d] done.
[services.d] starting services
[services.d] done.
```

As before, use a browser to login to the interface at `http://localhost:8787`.

Once the RStudio interface is displayed, notice the existence of the `metabolomics_r_code` folder which is now associated with the same folder within `/Users/esteban` (Fig. 20).

Clicking the folder will verify the existence of the user code (Fig. 21).

These files may be edited or removed with any changes within this container being reflected in `/Users/esteban/metabolomics_r_code`. Even after the `local_rstudio_vol` container is stopped or removed the changes will persist in the local folder. The “`-v`” option is what makes this possible. In this case the run command included:

```
-v /Users/esteban/metabolomics_r_code:/home/rstudio/metabolomics_r_code
```

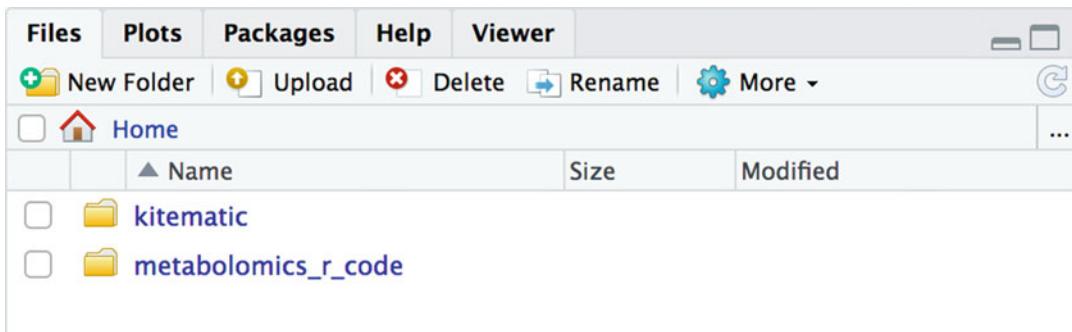


Fig. 20 Shared folder in R-studio container

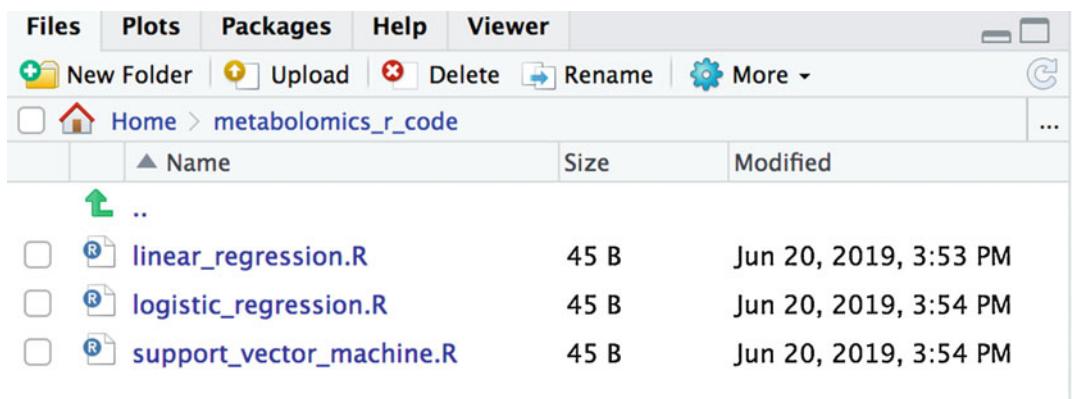


Fig. 21 User files

Notice the semicolon character that separates the two path specifications. The path on the left, /Users/esteban/metabolomics_r_code refers to the folder currently residing on the user's computer. The second path, /home/rstudio/metabolomics_r_code, indicates where in the container the local folder should be mapped to. The /home/rstudio folder exists in the folder, and depending on the image being used, this might need to change, though within the rocker domain the /home/rstudio folder will exist.

There are many R packages available in a Docker image format, so it is important to thoroughly search Docker Hub to take advantage of their existence. The rocker project is useful as it is the Bioconductor project which provides tools for the analysis of high-throughput genomic data.

More specifically there is a Docker image for the xcms R package for processing and visualization of chromatographically separated and single-spectra mass spectral data.

```
$ docker search bioconductor --limit=5 --format '{{.Name}} \t
```

```

{{.Description}}
bioconductor/release_base      release base container
bioconductor/release_core      release core container
bioconductor-devel_core2       Automated Build Bioconductor
Develement Core...
bioconductor-devel_base2       Automated Build Bioconductor
Develement Base...
bioconductor-devel_mscore2     Automated Build Bioconductor
Develement msco...
$
$ docker search xcms --limit=5 --format '{{.Name}} \t {{.
Description}}'
yufree/xcmsrocker   Rocker image for metabolomics study
payamemami/xcms-container
yufree/xcms      Non-Target Data Analysis Environment with CR...
pcm32/xcms-camera
wilsontom/xcms-dockerdev

```

5.7 Dockers for Python

The Jupyter project (<https://jupyter.org/>) provides a notebook system that supports over 40 languages, although it is strongly associated with the Python language. There are docker images available for Jupyter (Fig. 22):

Note that the “docker search” command has an option called “format” which permits selection of specific output fields. The default is to provide information on five fields that, depending on the length of the Description field, might result in too much information. The format option accepts a template specification to extract only those field names of interest.

```

$ docker search jupyter --limit=10 --format '{{.Name}}'
jupyter/datascience-notebook
jupyter/all-spark-notebook
jupyterhub/jupyterhub
jupyter/scipy-notebook
jupyter/tensorflow-notebook

```

NAME	DESCRIPTION	STARS	OFFICIAL	AUTO
jupyter/datascience-notebook	Jupyter Notebook Data Science Stack from ht...	487		
jupyter/all-spark-notebook	Jupyter Notebook Python, Scala, R, Spark, Me...	233		
jupyterhub/jupyterhub	JupyterHub: multi-user Jupyter notebook serv...	211		[OK]
jupyter/scipy-notebook	Jupyter Notebook Scientific Python Stack fro...	172		
jupyter/tensorflow-notebook	Jupyter Notebook Scientific Python Stack w/ ...	146		
jupyter/pyspark-notebook	Jupyter Notebook Python, Spark, Mesos Stack ...	106		
jupyter/minimal-notebook	Minimal Jupyter Notebook Stack from https://...	79		
jupyter/base-notebook	Small base image for Jupyter Notebook stacks...	68		
jupyterhub/singleuser	single-user docker images for use with Jupy...	21		[OK]
jupyter/nbviewer	Jupyter Notebook Viewer	15		[OK]

Fig. 22 Jupyter images on docker

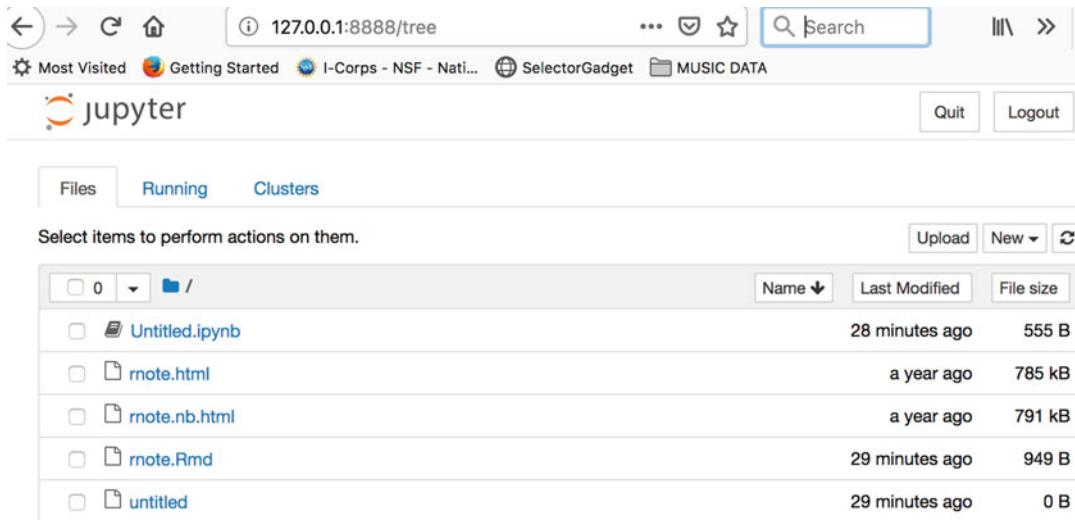
```
jupyter/pyspark-notebook
jupyter/minimal-notebook
jupyter/base-notebook
jupyterhub/singleuser
jupyter/nbviewer

$ docker search jupyter --limit=10 --format '{{.Name}} \t {{.
Description}}'
jupyter/datascience-notebook Jupyter Notebook Data Science
Stack from ht...
jupyter/all-spark-notebook Jupyter Notebook Python,
Scala, R, Spark, Me...
jupyterhub/jupyterhub JupyterHub: multi-user Jupyter
notebook serv...
jupyter/scipy-notebook Jupyter Notebook Scientific Python
Stack fro...
jupyter/tensorflow-notebook Jupyter Notebook Scientific Python
Stack w/ ...
jupyter/pyspark-notebook Jupyter Notebook Python, Spark,
Mesos Stack ...
jupyter/minimal-notebook Minimal Jupyter Notebook Stack
from https://...
jupyter/base-notebook Small base image for Jupyter Notebook
stacks...
jupyterhub/singleuser single-user docker images for use
with Jupy...
jupyter/nbviewer Jupyter Notebook Viewer
```

Launching Jupyter.

Based on the previous search for Jupyter notebooks, there is a `jupyter/minimal-notebook` image. The command will start the image in a local container and associate the local folder `/Users/esteban/notebooks` to the `/home/joyvan` folder inside the container. Also note that the container provides specific URL information to access the notebook (Fig. 23).

```
$ docker run -p 8888:8888 -v /Users/esteban/notebooks:/home/
joyvan jupyter/minimal-notebook
Executing the command: jupyter notebook
[I 21:10:08.657 NotebookApp] JupyterLab extension loaded from
/opt/conda/lib/python3.7/site-packages/jupyterlab
[I 21:10:08.658 NotebookApp] JupyterLab application directory
is /opt/conda/share/jupyter/lab
[I 21:10:08.661 NotebookApp] Serving notebooks from local
directory: /home/joyvan
[I 21:10:08.662 NotebookApp] The Jupyter Notebook is running
at:
```

**Fig. 23** Jupyter main page container

```
[I 21:10:08.662 NotebookApp] http://(aa292bbbba5f or
127.0.0.1):8888/?token=664e4773930180eacef53877390f30a39-
d758595ab0b61dc
```

```
[I 21:10:08.663 NotebookApp] Use Control-C to stop this server
and shut down all kernels (twice to skip confirmation).
```

```
[C 21:10:08.683 NotebookApp]
```

To access the notebook, open this file in a browser:

```
file:///home/jovyan/.local/share/jupyter/runtime/
nbsviewer-6-open.html
```

Or copy and paste one of these URLs:

```
http://(aa292bbbba5f or 127.0.0.1):8888/?toke-
n=664e4773930180eacef53877390f30a39d758595ab0b61dc
```

```
http://127.0.0.1:8888/?token=664e4773930180eacef53877390-
f30a39d758595ab0b61dc
```

5.8 Summary

This introduction has provided basic information on how to identify and run Docker containers which represent a lightweight alternative to virtualization techniques involving the use of hypervisor software. Dockers are particularly useful for containing applications involving multiple languages or run times as well as those delivered via the web such as RStudio, Jupyter Notebooks, and Galaxy. The Docker mechanism neatly packages these tools, and supporting components, into images on behalf of the end user who does not need to install or compile code. All that is required is installation of the Docker package for the local host. Docker also offers the ability

to create images using the Dockerfile mechanism as well as the docker-compose command which enables the deployment of multiple containers simultaneously.

Because these four tools, Python, R, Git, and Docker, are widely used in the information technologies, these are very transferable skills. For the same reason, one can easily find good online resources and get help. Stack Overflow (<http://stackoverflow.com>) is a popular site to search answers for or ask these technical questions. We also encourage readers to use and contribute to our supplemental site on GitHub (<https://metabolomics-data.github.io/>).

Acknowledgments

This work has been funded in part by the US National Institutes of Health via grants UH2 AI132345 (Li), U2C ES030163 (Jones, Li, Morgan, Miller), U01 CA235493 (Li, Xia, Siuzdak), U2C ES026560 (Miller), P30 ES019776 (Marsit), P50 ES026071 (McCauley), and the US EPA grant 83615301 (McCauley).



Chapter 16

Predictive Modeling for Metabolomics Data

**Tusharkanti Ghosh, Weiming Zhang, Debasish Ghosh,
and Katerina Kechris**

Abstract

In recent years, mass spectrometry (MS)-based metabolomics has been extensively applied to characterize biochemical mechanisms, and study physiological processes and phenotypic changes associated with disease. Metabolomics has also been important for identifying biomarkers of interest suitable for clinical diagnosis. For the purpose of predictive modeling, in this chapter, we will review various supervised learning algorithms such as random forest (RF), support vector machine (SVM), and partial least squares-discriminant analysis (PLS-DA). In addition, we will also review feature selection methods for identifying the best combination of metabolites for an accurate predictive model. We conclude with best practices for reproducibility by including internal and external replication, reporting metrics to assess performance, and providing guidelines to avoid overfitting and to deal with imbalanced classes. An analysis of an example data will illustrate the use of different machine learning methods and performance metrics.

Key words Metabolomics, Mass spectrometry, Supervised learning, Performance Metrics, Predictive Modeling

1 Introduction

In the past 20 years, there has been a dramatic increase in the development and use of high-throughput technologies for measuring various types of biological activity. Common examples include transcriptomics (the measurement of gene expression) and proteomics (the measurement of protein levels). The focus of this chapter is on metabolomics, which involves the measurement of small compounds, referred to here as metabolites, on a high-throughput basis. As products of activity at the protein level, metabolites represent an intermediate level between regulatory processes such as methylation and transcription, and the full spectrum of physiological and disease states. One appealing feature of metabolites is their ability to be used as clinical biomarkers, and for this reason, metabolomics has been extensively applied for finding biomarkers and

studying physiological processes and phenotypic changes associated with disease [1–4].

Metabolomics experiments fall into two categories: targeted and untargeted. Targeted metabolomics experiments measure ions from known biochemically annotated metabolites. By contrast, untargeted metabolomics experiments measure all possible ions within a predefined mass range and as a result may also include ions that do not map to known metabolites [5–8]. The main objective of metabolomics is to quantify and characterize the whole spectrum of metabolites. There are a variety of platforms by which metabolites can be measured. Examples include gas chromatography mass spectrometry (GC-MS) and liquid chromatography mass spectrometry (LC-MS) [9–11]. At a high level, these platforms input a sample, fragment it into ions, and separate them using physical properties in order to generate spectra for the sample. The fragmented ion spectra are then selected based on their physical properties (e.g., the retention time and the mass–charge ratio). In many instances, these properties can be used to map the ions to known metabolites.

Metabolomics data pose a variety of analytical challenges [12, 13]; thus, carefully constructed analytical pipelines need to be developed in order to preprocess and normalize the data. Once the data are normalized, one can proceed with various downstream analytical tasks, such as differential expression analysis, clustering, classification, network discovery, and visualization. In this chapter, we focus on the particular task of classification, which also goes by the name of prediction, supervised learning and biomarker discovery. We give an overview on some of the most commonly used methods for classification, along with an illustrative example using a dataset from our group. The structure of this chapter is as follows. In Subheading 2, we provide a short review of missing values and techniques for missing value imputation in metabolomics data. We then briefly describe the most commonly used supervised learning methods (Random Forest, Support Vector Machine, and Partial Least Square-Discriminant Analysis). In Subheading 3, we lay out a framework on fitting prediction models and their practical issues. This is followed by an illustrative example of data analysis and performance evaluation in Subheading 4, and the chapter ends with a short discussion in Subheading 5. In this chapter, we interchangeably use the term supervised learning, predictive modeling and machine learning.

2 Methods

2.1 Missing Values

Supervised learning methods require complete data; however, untargeted metabolomic data is prone to missing values, where the data matrix contains zeros in one or more entries. Some studies

have reported 20–30% missing values in datasets generated using untargeted MS [14, 15]. It is difficult to deduce whether a missing value is a genuine absence of a feature, a feature below the lower limit of detection of the machine, or the failure of the algorithms employed to identify real signals from the background. In practice, statisticians have defined three types of mechanisms that lead to missing values: missing at random (MAR), missing not at random (MNAR) and missing completely at random (MCAR) [16–18]. MCAR means that the missingness mechanism is completely random and depends neither on the observed data nor on the missing data. Scientifically plausible reasons that are compatible with missing completely at random include random errors or stochastic fluctuations of peak detection during the acquisition process of the raw data (incomplete derivations of signals). MAR means that the probability of a variable being missing is fully accounted for by other observed variables. Missing not at random (MNAR) means that the missingness mechanism depends on the unobserved values. If analysts believe MNAR to hold, there are unfortunately no ways to assess this assumption using observed data. A practical strategy is to collect as much covariate information as possible in order to make the MAR assumption plausible with the observed data.

There have been several attempts in the literature to deal with missing values for metabolomics data. For example, fillPeaks [19] in the XCMS software package has many missing value imputation tools available. A practical rule of thumb is to impute missing values by a small value or zero. This is problematic in that this leads to distortions of the distribution of missing variables and can cause the standard deviations to be underestimated [20]. Finally, Zhan et al. developed kernel-based approaches which explicitly modeled the missingness into a differential expression analysis [21]. Other imputation strategies include imputing missing values by zero, half of the minimum value or by the mean or median of observed values. More advanced methods use, random forest (RF) [20, 22], singular value decomposition (SVD) [23, 24], and k -nearest neighbors (kNN) [25]. The choice of these methods can influence the data analyses and inferences [14, 22, 26]. It is therefore extremely crucial to select the most suitable method for tackling missing values before moving forward with prediction. Recent work has compared performance of various missing value imputation methods [14, 25, 27, 28] on MS metabolomics data [20].

2.2 Classification

Methods: An Early Look

It is important to note that what we now call supervised learning dates back to over 80 years ago, when Sir R. A. Fisher introduced the use of linear discriminant analysis (LDA) [29]. This was a generative model in which the features conditional on class label were modeled as a multivariate normal distribution with a mean vector that depended on group, and a common covariance matrix.

This was generalized to quadratic discriminant analysis, in which the covariance matrix also depends on the group. Linear discriminant analysis possessed two desirable properties:

1. Since the multivariate normal distribution is fully specified by the mean vector and covariance matrix, it is relatively simple to compute.
2. The classification rule from LDA is linear in the predictors and thus simple to interpret.

While this methodology is well established, there are two challenges with modern metabolomics data that make the utility of LDA less effective. First, in most situations, the number of metabolites being measured is greater than the sample size, which means that the covariance matrix will not be directly estimable from the observed data. Second, there is an increasing recognition that the linear classification rule might be too restrictive and that analysts should consider other nonlinear classifiers. This will motivate the classification tools we describe in Subheadings 2.3–2.5.

A second technique that dates back to the 1950s and has been used extensively in machine learning is Naïve Bayes [30]. In this framework, we assume the features are conditionally independent given the group label and model the likelihood ratio of the feature given the group label. Based on the product of these likelihood ratios, we are able to assign a new observation to a predicted group. The term “Naïve” comes from the fact that we assume that the features are statistically independent when, in fact, we know that they are not. That said, Naïve Bayes has been shown to be an effective tool in classification problems [31], and it can handle the situation when the number of metabolites measured is greater than the sample size.

2.3 Decision Tree

A Decision Tree (DT) is a supervised machine learning model, that outputs a hierarchical structure to classify subjects [32]. It is a nonlinear classifier which is mainly used for classifying nonlinearly separable data. The objective of a decision tree is to develop a model that predicts the value of a response variable based on several predictor variables. Figure 1 shows an example of a hypothetical DT, which divides the data into two categories based on two input variables. DT used in data mining can be classified into two groups:

- Classification tree: The predicted outcome is a categorical variable, representing two or more classes to which the observation belongs.
- Regression tree: The predicted value is a continuous variable.

DT is also known as Classification and Regression Trees (CART), which was first introduced in the machine learning literature [33]. The main difference between classification and regression trees is the criteria on which the split-point decision is made.

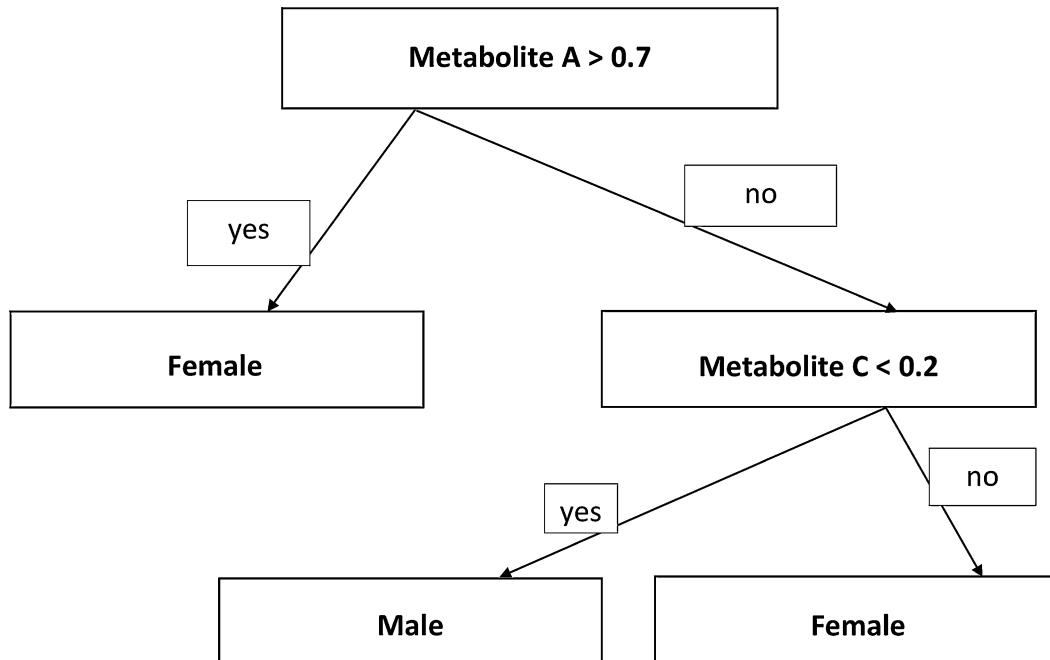


Fig. 1 A simple decision tree that splits the data into two gender groups based on two metabolites

2.4 Random Forest

A Random Forest (RF) is an extremely reliable classifier and robust to overfitting. It constructs an ensemble of DTs, which means an aggregation of tree-structured predictors [34]. In RF, each tree is independently constructed using a bootstrap sample of the original data (the “bagged sample”). This training data is used to build the classification model. The data that was not sampled using the bootstrap is referred to as the out-of-bag sample. Since these data were not used in model building, they can be used as a test data set, which can be used to evaluate classification accuracy in an unbiased manner, by calculating the “out-of-bag error” [35]. A measure of the variable importance of classification is also computed by considering the difference between the results from the original and randomly permuted versions of the data set. Cross-validation is not needed since RF is estimated from the bootstrap samples.

RF has become popular as a biomarker detection tool in various metabolomics studies [36, 37]. RF has the strength to deal with missing and data [34, 38] and overfitting issues [39, 40]. In addition, it can also tackle high-dimensional data sets without feature elimination as a requirement [41].

2.5 Support Vector Machines

Support Vector Machines (SVM) have been previously used in the analysis of several omics studies, particularly gene expression data [42–44]. A simple figure of an SVM is shown in Fig. 2. The main characteristics that define the concept of SVMs are (a) the criteria they use to categorize nonlinear relationships (b) the set of training

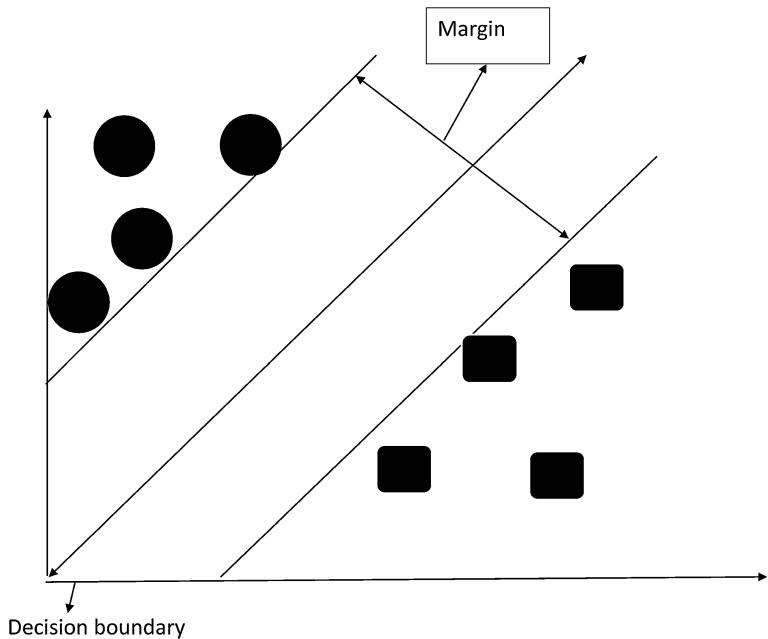


Fig. 2 A simple graphical representation of SVM

sets that are necessary to optimize the linear classifier; (c) the use of kernel machines to transform the variable into a higher order nonlinear space where linear separability holds; (d) utility in terms of performance and efficiency for high dimensional data sets.

A major drawback of SVM is its restrictions to binary classification problems. For example, it can only discriminate between two classes where the data points are categorized by two classes in n -dimensional space, where n corresponds to the number of metabolites in our context. A hyperplane is constructed that separates the data points from the two classes. The hyperplane coefficients are determined based on the variable (metabolite) importance for discriminating between two classes.

SVM can yield a hyperplane of $p-1$ dimension in p dimensional space. The main purpose of SVM is to optimize the largest margin. In practice, a separation often does not exist as the data points cannot always be linearly separated. In such nonlinear cases, a kernel substitution is adopted to map the data to a higher order dimension. The maximum-margin hyperplane was the original algorithm developed as a [linear classifier](#) [45]. An extension to create nonlinear classifiers was proposed by applying the [kernel trick](#) to maximum-margin hyperplanes [46]. The advantage of using the kernel trick is that it can substitute the linear kernel with other robust kernels, such as the Gaussian kernel [47]. Also in the family of nonlinear supervised learners are deep neural networks (DNN), which construct a nonlinear function from input variables to outcome variables using a combination of convolution filters and hidden layers [48].

2.6 Partial Least Squares-Discriminant Analysis

Partial least squares-discriminant analysis (PLS-DA) is a supervised technique widely used in metabolomics studies [49–52]. It is mainly constructed on the rotation of metabolite abundances in order to maximize the covariance between the independent variables (metabolite abundances) and the corresponding response variable (classes) in high-dimension by finding a linear subspace of the predictors [53]. PLS-DA is an extension of classical PLS regression which was implemented for solving linear equations and estimating parameters of interest. PLS-DA method has been extensively used in various metabolomics studies for disease classification and biomarker detection [50, 54–56]. Furthermore, PLS-DA can also be used for dimension reduction, and feature selection by ranking the loading vectors in decreasing order [52, 57, 58].

Orthogonal PLS (OPLS)-DA was developed as an improvement to PLS-DA in order to discriminate two or more classes of metabolites using multivariate data [59, 60]. The main advantage of OPLS-DA over PLS-DA is that a single component is used as a predictor where the other components constitute the orthogonal contrasts for analysis of variance, which are independent linear comparisons between the classes of a component.

Multilevel PLS-DA is another classification technique that can be used to classify multivariate data from crossover designed studies [61]. For example, each subject in a controlled experimentation setup undergo treatment in a random order [62]. Multilevel PLS-DA can be thought as a multivariate extension of the paired *t*-test [61].

3 Practical Issues in Fitting Prediction Models

3.1 Feature Selection

Feature Selection (FS) is an important step in successful data mining procedures [63], such as SVMs [64, 65] and Naïve Bayes [66], to enhance performance and reduce computational efficiency. However, FS is not a necessary criterion for some supervised algorithms, such as SVM due to its reliance on regularization, which is the process of adding information to prevent overfitting in order to enhance the predictive accuracy and interpretability of the supervised learning model. The purpose of feature selection is similar to model selection [67], which tries to find a compromise between high predictive accuracy and a model with few predictors. The insignificant input features in a supervised model may lead to overfitting. Hence, it is reasonable to ignore those input features with negligible or no effect on the output. For example, in the example later in this chapter, the objective is to infer the relationship between gender and their corresponding metabolite features. However, if the sample identifier or any other redundant column is included as one of the input features, it may cause overfitting. FS

is generally used as a preprocessing tool, in order to reduce the dimension of a data set by only selecting subsets of features (metabolites), on which a supervised learning is employed. Some well-known extensions of these FS techniques are Recursive Feature Elimination, L1 norm SVM [68], and Sequential Minimal Optimization (SMO) [69].

One of the most commonly used measure in FS is the Variable Importance Score (VIS), which evaluates features using a model-based approach [70] by ranking the features according to their relevance in a classification problem [71]. The main advantage of using VIS is that incorporates the correlation structure between the predictors (metabolite features) into the importance calculation.

3.2 Cross-Validation

The classification performance of supervised learners is crucial to determine their predictive power and accuracy. Generally, the validation procedures are implemented by assuming the model on a training set and then testing it on an independent set (validation data set). However, in practical situations, due to the relatively small number of samples and unavailability of an unbiased independent validation data set, cross-validation (CV) can be applied by splitting a data set into training and test sets. Using k -fold cross validation [36], the training data set is split into k subsets (folds) of almost equal size, that is, where $k-1$ training sets consist of $x\%$ of the data and the remaining $(100-x)\%$ data is contained in the k th test data set. Ideally, $x\%$ far exceeds $(100-x)\%$, and x is usually chosen as 90, 80 or 70. Leave-One-Out-CV is a special case of CV, where k is equal to the total number of data points.

3.3 Metrics for Evaluation

There are several potential metrics by which one can evaluate a prediction model. The most common metric that is used in practice is the classification accuracy, meaning the proportion of predictions from the model that are correct based on the gold standard label. An alternative classification metric is given by the receiver operating characteristic (ROC) curve. Assume that we have two groups, disease and control and that higher values of the model correspond to a greater probability of having disease. We will let the model output be Y and group label be D , where $D = 0$ means control and $D = 1$ means diseased. One can define the false positive rate based on a cutoff c by $\text{FP}(c) = P(Y > c | D = 0)$. Similarly, the true positive rate is $\text{TP}(c) = P(Y > c | D = 1)$. The true and false positive rates can then be summarized by the receiver operating characteristic (ROC) curve, which is a graphical presentation of $\text{TP}(c), \text{FP}(c)$ for all possible cutoff values of c . The ROC curve shows the tradeoff between increasing true positive and false positive rates. Then, the area under the ROC curve (AUC) can be measured for the curve and is a summary based on how well the model can distinguish between two diagnostic groups (diseased/control). Other commonly used metrics are defined in terms of $\text{TP}(c)$ and $\text{FP}(c)$ as below:

Sensitivity (SENS): $\text{SENS} = \text{TP}(c)$.

Specificity (SPEC): $\text{SPEC} = 1 - \text{FP}(c)$.

Precision (PREC): $\text{TP}(c) / (\text{TP}(c) + \text{FP}(c))$.

Recall (REC): $\text{REC} = \text{SENS} = \text{TP}(c)$.

False Discovery Rate (FDR): $\text{FP}(c) / (\text{TP}(c) + \text{FP}(c))$.

The predicted classes are conventionally computed based on the cut-off c ($=50\%$) for the probabilities. However, the cutoff (threshold) value can be tuned to control the FDR depending on the problem setting in order to attain maximum predictive accuracy.

Calibration is another property that has been espoused for risk prediction models. Well-calibrated models are those in which the predicted risk matches the observed risk for individuals. The manner in which this is typically assessed is by comparing the risk predictions from the model to some nonparametric (i.e., non-model-based) estimate; the closer the predictions are, the better calibrated the model is. Calibration has been advocated in the risk prediction [72]. As a matter of course, nonparametric estimates of risk models require binning of covariates or categorization of predicted values in order to deal with the inherent sparsity that exists with using continuous covariates. One method of performing calibration, in the binary outcome setting, is to use the Hosmer-Lemeshow goodness of fit statistic [73]; smaller values of the statistic correspond with better calibrated models.

In the calibration setting, what is important is understanding the distribution of the predicted probabilities, or equivalently, the risk scores, from the fitted model. Calibration of the model then is equivalent to modeling the distribution of risk scores; a useful quantity for accomplishing this is the predictiveness curve [74].

3.4 Imbalanced Classes

In numerous data sets, there are unequal numbers of cases in each class. In this instance, the classifier is biased toward better performance of the larger (or majority) class, compared to the smaller (or minority) class. Often, the research question is much more focused on performance of discriminating the minority class from the majority class. But the size of the minority class may be limited by the difficulty, expense, or time of obtaining the rarer type of sample. This unequal distribution between classes of a data set is referred to as the imbalanced class problem [75].

In such cases, the main interest lies in the correct classification of the “minority class” [76]. Classes with fewer samples or no sample have a low prior probability and low error cost [77]. The relation between the distribution of samples in the training set and costs of misclassification can be controlled by setting a prior probability at each class.

Several methods have been discussed for tackling imbalanced data [78, 79], and two techniques which have been extensively applied in the last decade are resampling and cost-sensitive learning. In resampling, the approach is to either oversample the minority class or undersample the majority class. For example, the minority class can be oversampled by producing duplicates [80] or under sampling (removing samples) of the majority class [81, 82]. One major drawback of under sampling is that the majority class may lose some information, if a large part of majority class in a small training set is not considered. In cost-sensitive learning, the approach is to assign a cost misclassification of the minority class and minimize the overall cost function [83, 84]. Both the resampling and cost-sensitive learning approaches are considered to be more effective in terms of predictive accuracy than by using equal class prior constraints [85].

3.5 TRIPOD Guidelines

A recent scientific initiative has focused on developing more reproducible approaches to the building, evaluation and validation of prediction models. A document resulting from this effort that helps in this goal is the TRIPOD (Transparent reporting of a multivariable prediction model for individual prognosis or diagnosis) statement, which provides recommendations for fair reporting of studies developing, validating, or updating a prediction model. It consists of a 22-item checklist detailing vital information that must be incorporated in a prediction model study report [86]. For our example analysis below, we provide supporting information to illustrate how our analysis satisfies the 22-item TRIPOD checklist (Appendix 1).

4 Illustrative Example

4.1 Data

For the predictive comparison of classifiers, we obtained a LC-MS metabolomics dataset from <https://www.metabolomicsworkbench.org> (Project ID: PR00038). The data were generated from subjects enrolled in the Genetic Epidemiology of Chronic Obstructive Pulmonary Disease Gene study (COPD-Gene) [87, 88]. Plasma from 131 subjects was collected from the COPDGene study cohort and analyzed using untargeted LC-MS (C18+ and HILIC+) metabolomics. The lipid fraction of the human plasma collected from current and former smokers was analyzed using Time of Flight (ToF) liquid chromatograph (LC) (Agilent 6210 Series) and a Quadrupole ToF mass spectrometer (Agilent 6520) which yielded combined data on 2999 metabolite features. Data were annotated, normalized and preprocessed using the methods described in [87, 88].

COPD is an extremely heterogeneous disease comprising multiple phenotypes. The 131 subjects were either current or former

smokers with various chronic obstructive pulmonary disease (COPD) phenotypes including airflow obstruction, radiologic emphysema, and exacerbations. Within this set there were 56 males and 75 females. For additional information about the cohort, sample collection and data storage data generation, see [87].

4.2 Training and Test Sets

We split the data (131 samples) into 70% (93 samples) training and 30% (38 samples) test (evaluation) data. For the training data, we use fivefold CV, where we split the training data (93 training samples) into 5 different subsets (or fivefolds). We used the first four-folds to train the data and left the last (fifth) fold as holdout-test dataset. We then trained the algorithms against each of the folds and computed (average over fivefolds) the metrics for the training dataset. The test dataset ($n = 38$ samples) is used to provide an unbiased evaluation of the best model fit on the training dataset. The test dataset can be regarded as an external dataset which provides the gold standard used to evaluate the models, using ROC curves and other metrics for evaluation. For model validation, we predicted the performance of the test data using the trained models for all the three classifiers.

4.3 Feature Ranking and Variable Importance

In this section, we implemented different predictive models using metabolite abundances as the predictor variables and Gender (Male/Female) as the response based on the training dataset. We then computed the Variable Importance Score, which is a measure of feature relevance to gender for each metabolite (see Subheading 3.1). These scores are nonparametric in nature, and range between 0 and 100. They are subsequently used to rank all the features to the classification of our response variable, that is, Gender. Metabolites with high values are considered to more relevant features in classification problem.

In the dataset, the top five metabolites are detected as feature metabolites out of 2999 metabolites in the training set with fivefold Cross-Validation for three different classifiers (Fig. 3a–c). Among them, C39 H79 N7 O + 7.3314843, N-palmitoyl-D-sphingosyl-1-(2-aminoethyl)phosphonate, and C43 H86 N2 O2 are considered to be significant metabolite features based on RF and SVM classifiers. However, zeta-Carotene, unannotated metabolite (mass: 2520.6355 and retention time: 1.5409486), 5-Hydroxyisourate +4.668069, C13 H28 N2 O4, and Tyrosine* +2.3151746 are identified as good predictors based on PLS-DA.

4.4 Model Validation

In this section, we evaluated the performance of all the three classifiers based on the 30% test data of 38 samples using the trained models. Here, we present ROC curves for all the predictive models of the testing data used to compute the diagnostic potential of a classifier in this clinical metabolomics application. From the ROC

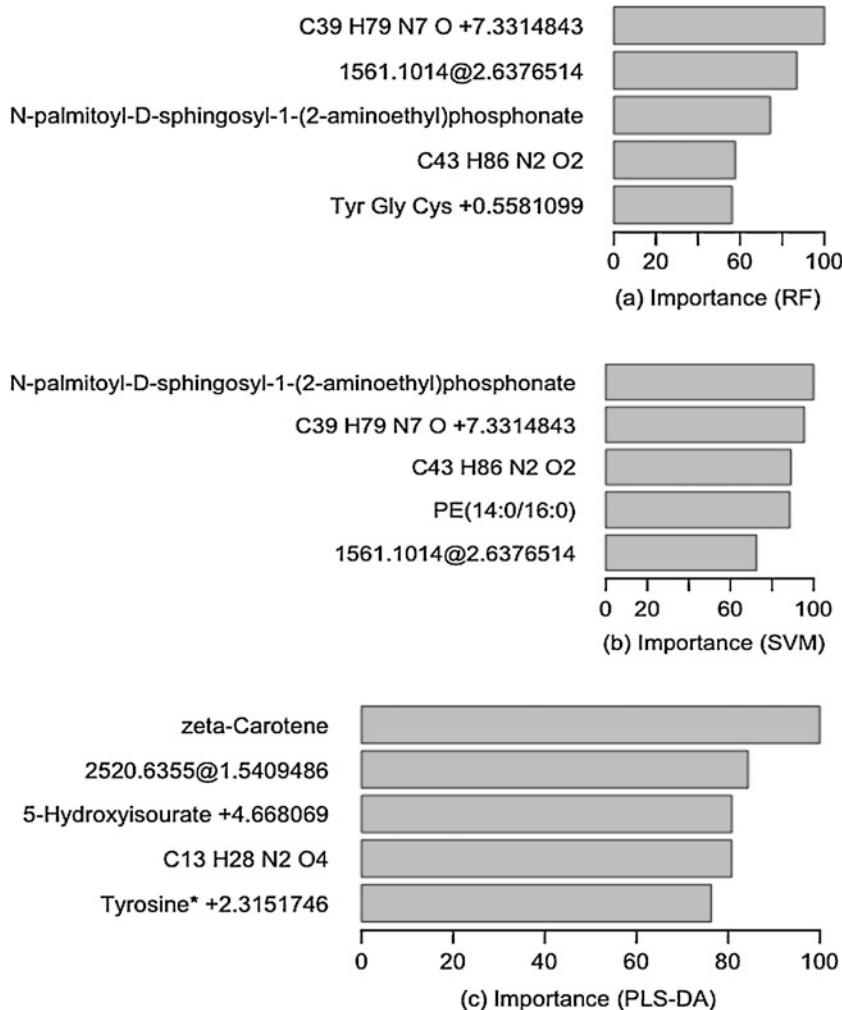


Fig. 3 Metabolite relevant feature ranking bar plots (top five metabolites) using Variable Important Scores ranging from 0 to 100. **(a)** Random Forest, **(b)** Support Vector Machine (SVM), and **(c)** Partial Least Square-Discriminant Analysis (PLS-DA) for the training dataset

curves, the three methods perform similarly (Fig. 4). Table 1 shows the performance metrics of the testing data evaluated for all the classifiers. In this testing dataset, we use AUC as our metric to choose the best performing classifier. Based on this metric RF has a small advantage over the other methods (0.87 versus 0.86), but with other metrics the other methods have a small advantage. In addition, we also computed the Variable Importance Score on the test dataset. The top five metabolites for all the three classifiers using the test dataset were exactly the same selected using the training dataset with fivefold CV in the previous section.

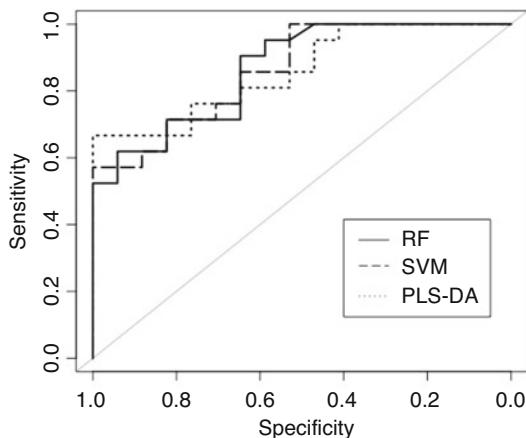


Fig. 4 ROC curves of the testing dataset obtained from three classification algorithms (RF, SVM, and PLS-DA)

Table 1

Metrics (area under curve (AUC), sensitivity (SENS), specificity (SPEC), precision (PREC), recall (REC)) to evaluate the performance of classification on testing dataset

Metrics/methods	AUC	SENS	SPEC	PREC	REC
RF	0.87	0.71	0.64	0.71	0.71
SVM	0.86	0.76	0.71	0.76	0.76
PLS-DA	0.86	0.81	0.65	0.74	0.81

5 Summary

Biomarker detection in the field of metabolomics is popular both in the context of prognostic and diagnostic studies. In this chapter, we discussed the most commonly used supervised learning algorithms, feature selection methods, and performance metrics, used in the downstream analyses of metabolomics studies. In addition, we also reported predictive accuracy of three classifiers on an example human plasma LC/MS test dataset to predict gender. Even though there were advantages of one method compared to the other depending on the metric, our results cannot be held as a comprehensive comparison of these methods, since different classifiers perform differently depending on the datasets. We encourage investigators to explore a variety of methods. For more detailed discussions of biomarker detection and predictive accuracy, see [89–92]. The R code for this chapter is posted at the supplemental website, <https://metabolomics-data.github.io/>. Appendix 2 lists selected open source tools that implement supervised learning algorithms.

TRIPOD Checklist for Predictive Modeling for Metabolomics Data

Section/topic	Item	Checklist item	Section
<i>Title and abstract</i>			
Title	1	Identify the study as developing and/or validating a multivariable prediction model, the target population, and the outcome to be predicted	See title
Abstract	2	Provide a summary of objectives, study design, setting, participants, sample size, predictors, outcome, statistical analysis, results, and conclusions	See abstract
<i>Introduction</i>			
Background and objectives	3a	Explain the medical context (including whether diagnostic or prognostic) and rationale for developing or validating the multivariable prediction model, including references to existing models	Subheading 4.1
	3b	Specify the objectives, including whether the study describes the development or validation of the model or both	Internal validation, Subheading 4.4

(continued)

Section/topic	Item	Checklist item	Section
<i>Methods</i>			
Source of data	4a	Describe the study design or source of data (e.g., randomized trial, cohort, or registry data), separately for the development and validation data sets, if applicable	Subheading 4.1
	4b	Specify the key study dates, including start of accrual; end of accrual; and, if applicable, end of follow-up	Subheading 4.1, see [87]
Participants	5a	Specify key elements of the study setting (e.g., primary care, secondary care, general population) including number and location of centers	N/A
	5b	Describe eligibility criteria for participants	Subheading 4.1, see [87]
	5c	Give details of treatments received, if relevant	N/A
Outcome	6a	Clearly define the outcome that is predicted by the prediction model, including how and when assessed	Subheading 4.1
	6b	Report any actions to blind assessment of the outcome to be predicted	N/A

(continued)

Section/topic	Item	Checklist item	Section
Predictors	7a	Clearly define all predictors used in developing or validating the multivariable prediction model, including how and when they were measured	2999 predictors, for more details see [87]
	7b	Report any actions to blind assessment of predictors for the outcome and other predictors	N/A
Sample size	8	Explain how the study size was arrived at	Subheading 4.1, see [87]
Missing data	9	Describe how missing data were handled (e.g., complete-case analysis, single imputation, multiple imputation) with details of any imputation method	The data was already preprocessed and imputed, see Subheading 4.1
Statistical analysis methods	10c	For validation, describe how the predictions were calculated	Subheading 3.3
	10d	Specify all measures used to assess model performance and, if relevant, to compare multiple models	Subheading 3.3
	10e	Describe any model updating (e.g., recalibration) arising from the validation, if done	N/A

(continued)

Section/topic	Item	Checklist item	Section
Risk groups	11	Provide details on how risk groups were created, if done	N/A
Development vs. validation	12	For validation, identify any differences from the development data in setting, eligibility criteria, outcome, and predictors	N/A
<i>Results</i>			
Participants	13a	Describe the flow of participants through the study, including the number of participants with and without the outcome and, if applicable, a summary of the follow-up time. A diagram may be helpful	Subheading 4.1, <i>see [87]</i>
	13b	Describe the characteristics of the participants (basic demographics, clinical features, available predictors), including the number of participants with missing data for predictors and outcome	Subheading 4.1, <i>see [87]</i>
	13c	For validation, show a comparison with the development data of the distribution of important variables (demographics, predictors, and outcome)	Subheadings 4.3 and 4.4

(continued)

Section/topic	Item	Checklist item	Section
Model performance	16	Report performance measures (with CIs) for the prediction model	N/A
Model-updating	17	If done, report the results from any model updating (i.e., model specification, model performance)	Subheading 4.4
<i>Discussion</i>			
Limitations	18	Discuss any limitations of the study (such as nonrepresentative sample, few events per predictor, missing data)	Subheading 4.1, see [87]
Interpretation	19a	For validation, discuss the results with reference to performance in the development data, and any other validation data	N/A
	19b	Give an overall interpretation of the results, considering objectives, limitations, results from similar studies, and other relevant evidence	Subheadings 4.4 and 5
Implications	20	Discuss the potential clinical use of the model and implications for future research	Subheadings 4.4 and 5. However, performance of the model is data-driven
<i>Other information</i>			

(continued)

Section/topic	Item	Checklist item	Section
Supplementary information	21	Provide information about the availability of supplementary resources, such as study protocol, web calculator, and data sets	Subheading 4.1, see [87]
Funding	22	Give the source of funding and the role of the funders for the present study	NIH

Selected Open Source (R/Bioconductor/Web-Based) Tools for Supervised Learning Algorithms

Method	Source	Reference
PLS-DA	Bioconductor (roppls)	[93]
PLS-DA, RF, and SVM	Bioconductor (biosigner)	[94]
SVM, RF	Bioconductor (MLSeq)	[95]
RF, SVM, PLS-DA	Metaboanalyst http://www. metaboanalyst.ca/	[96]
PCA, PLS-DA, RF	Bioconductor (statTarget)	[97]
Feature selection, metric evaluation	Bioconductor (OmicsMarker)	[98]
Sparse PLS-DA	Bioconductor (mixOmics)	[99]
Feature selection, metric evaluation	CRAN (lilikoi)	[100]
Probabilistic principal component analysis	CRAN (MetabolAnalyze)	[101]
Kernel-based metabolite differential analysis	CRAN (KMDA)	[21]
PLS-DA, OPLS-DA	CRAN (muma)	[102]
RF	CRAN (RFmarkerDetector)	[103]
RF, SVM, PLS-DA	CRAN (caret)	[104]

References

1. Maniscalco M, Fuschillo S, Paris D, Cutignano A, Sanduzzi A, Motta A (2019) Clinical metabolomics of exhaled breath condensate in chronic respiratory diseases. *Adv Clin Chem* 88:121–149. <https://doi.org/10.1016/bs.acc.2018.10.002>
2. Pujos-Guillot E, Petera M, Jacquemin J, Centeno D, Lyan B, Montoliu I, Madej D, Pietruszka B, Fabbri C, Santoro A, Brzozowska A, Franceschi C, Comte B (2018) Identification of pre-frailty sub-phenotypes in elderly using metabolomics. *Front Physiol* 9:1903. <https://doi.org/10.3389/fphys.2018.01903>
3. Sarode GV, Kim K, Kieffer DA, Shibata NM, Litwin T, Czlonkowska A, Medici V (2019) Metabolomics profiles of patients with Wilson disease reveal a distinct metabolic signature. *Metabolomics* 15(3):43. <https://doi.org/10.1007/s11306-019-1505-6>
4. Wang X, Zhang A, Sun H (2013) Power of metabolomics in diagnosis and biomarker discovery of hepatocellular carcinoma. *Hepatology* 57(5):2072–2077
5. Caesar LK, Kellogg JJ, Kvalheim OM, Cech NB (2019) Opportunities and limitations for untargeted mass spectrometry metabolomics to identify biologically active constituents in complex natural product mixtures. *J Nat Prod* 82:469. <https://doi.org/10.1021/acs.jnatprod.9b00176>
6. Liu LL, Lin Y, Chen W, Tong ML, Luo X, Lin LR, Zhang HL, Yan JH, Niu JJ, Yang TC (2019) Metabolite profiles of the cerebrospinal fluid in neurosyphilis patients determined by untargeted metabolomics analysis. *Front Neurosci* 13:150. <https://doi.org/10.3389/fnins.2019.00150>
7. Sanchez-Arcos C, Kai M, Svatos A, Gershenson J, Kunert G (2019) Untargeted metabolomics approach reveals differences in host plant chemistry before and after infestation with different pea aphid host races. *Front Plant Sci* 10:188. <https://doi.org/10.3389/fpls.2019.00188>
8. Wang R, Yin Y, Zhu ZJ (2019) Advancing untargeted metabolomics using data-independent acquisition mass spectrometry technology. *Anal Bioanal Chem* 411:4349. <https://doi.org/10.1007/s00216-019-01709-1>
9. Allwood JW, Xu Y, Martinez-Martin P, Palau R, Cowan A, Goodacre R, Marshall A, Stewart D, Howarth C (2019) Rapid UHPLC-MS metabolite profiling and phenotypic assays reveal genotypic impacts of nitrogen supplementation in oats. *Metabolomics* 15(3):42. <https://doi.org/10.1007/s11306-019-1501-x>
10. Fang J, Zhao H, Zhang Y, Wong M, He Y, Sun Q, Xu S, Cai Z (2019) Evaluation of gas chromatography-atmospheric pressure chemical ionization tandem mass spectrometry as an alternative to gas chromatography tandem mass spectrometry for the determination of polychlorinated biphenyls and polybrominated diphenyl ethers. *Chemosphere* 225:288–294. <https://doi.org/10.1016/j.chemosphere.2019.03.011>
11. Lohr KE, Camp EF, Kuzhumparambil U, Lutz A, Leggat W, Patterson JT, Suggett DJ (2019) Resolving coral photoacclimation dynamics through coupled photophysiological and metabolomic profiling. *J Exp Biol* 222:jeb195982. <https://doi.org/10.1242/jeb.195982>
12. Baumeister TUH, Ueberschaar N, Schmidt-Heck W, Mohr JF, Deicke M, Wichard T, Guthke R, Pohnert G (2018) DeltaMS: a tool to track isotopologues in GC- and LC-MS data. *Metabolomics* 14(4):41. <https://doi.org/10.1007/s11306-018-1336-x>
13. Gilmore IS, Heiles S, Pieterse CL (2019) Metabolic imaging at the single-cell scale: recent advances in mass spectrometry imaging. *Annu Rev Anal Chem* (Palo Alto Calif) 12:201. <https://doi.org/10.1146/annurev-anchem-061318-115516>
14. Do KT, Wahl S, Raffler J, Molnos S, Laimighofer M, Adamski J, Suhre K, Strauch K, Peters A, Gieger C, Langenberg C, Stewart ID, Theis FJ, Grallert H, Kastenmuller G, Krumsiek J (2018) Characterization of missing values in untargeted MS-based metabolomics data and evaluation of missing data handling strategies. *Metabolomics* 14(10):128. <https://doi.org/10.1007/s11306-018-1420-2>
15. Liggi S, Hinz C, Hall Z, Santoru ML, Poddighe S, Fjeldsted J, Atzori L, Griffin JL (2018) KniMet: a pipeline for the processing of chromatography-mass spectrometry metabolomics data. *Metabolomics* 14(4):52. <https://doi.org/10.1007/s11306-018-1349-5>
16. Fielding S, Fayers PM, McDonald A, McPherson G, Campbell MK (2008) Simple imputation methods were inadequate for missing not at random (MNAR) quality of life data. *Health Qual Life Outcomes* 6(1):57

17. Schafer JL, Graham JW (2002) Missing data: our view of the state of the art. *Psychol Methods* 7(2):147
18. Steyerberg EW, van Veen M (2007) Imputation is beneficial for handling missing data in predictive models. *J Clin Epidemiol* 60(9):979
19. Smith CA, Want EJ, O'Maille G, Abagyan R, Siuzdak G (2006) XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Anal Chem* 78(3):779–787. <https://doi.org/10.1021/ac051437y>
20. Wei R, Wang J, Su M, Jia E, Chen S, Chen T, Ni Y (2018) Missing value imputation approach for mass spectrometry-based metabolomics data. *Sci Rep* 8(1):663. <https://doi.org/10.1038/s41598-017-19120-0>
21. Zhan X, Patterson AD, Ghosh D (2015) Kernel approaches for differential expression analysis of mass spectrometry-based metabolomics data. *BMC Bioinformatics* 16:77. <https://doi.org/10.1186/s12859-015-0506-3>
22. Gromski PS, Xu Y, Kotze HL, Correa E, Ellis DI, Armitage EG, Turner ML, Goodacre R (2014) Influence of missing values substitutes on multivariate analysis of metabolomics data. *Metabolites* 4(2):433–452. <https://doi.org/10.3390/metabo4020433>
23. Kumar N, Hoque MA, Shahjaman M, Islam SM, Mollah MN (2017) Metabolomic biomarker identification in presence of outliers and missing values. *Biomed Res Int* 2017:2437608. <https://doi.org/10.1155/2017/2437608>
24. Sun X, Langer B, Weckwerth W (2015) Challenges of inversely estimating Jacobian from metabolomics data. *Front Bioeng Biotechnol* 3:188. <https://doi.org/10.3389/fbioe.2015.00188>
25. Lee JY, Styczynski MP (2018) NS-kNN: a modified k-nearest neighbors approach for imputing metabolomics data. *Metabolomics* 14(12):153. <https://doi.org/10.1007/s11306-018-1451-8>
26. Di Guida R, Engel J, Allwood JW, Weber RJM, Jones MR, Sommer U, Viant MR, Dunn WB (2016) Non-targeted UHPLC-MS metabolomic data processing methods: a comparative investigation of normalisation, missing value imputation, transformation and scaling. *Metabolomics* 12(5):93. <https://doi.org/10.1007/s11306-016-1030-9>
27. Chen MX, Wang SY, Kuo CH, Tsai IL (2019) Metabolome analysis for investigating host-gut microbiota interactions. *J Formos Med Assoc* 118(Suppl 1):S10–S22. <https://doi.org/10.1016/j.jfma.2018.09.007>
28. Shen X, Zhu ZJ (2019) MetFlow: an interactive and integrated workflow for metabolomics data cleaning and differential metabolite discovery. *Bioinformatics* 35:2870. <https://doi.org/10.1093/bioinformatics/bty1066>
29. McLachlan, Geoffrey J (2004) Discriminant analysis and statistical pattern recognition. Wiley-Interscience, Hoboken, N.J. John Wiley & Sons. & Wiley InterScience (Online Service)
30. McCallum A, Nigam K (1998) A comparison of event models for naive Bayes text classification. In: AAAI-98 workshop on learning for text categorization, vol 1. Citeseer, pp 41–48
31. Wang Q, Garrity GM, Tiedje JM, Cole JR (2007) Naive Bayesian classifier for rapid assignment of rRNA sequences into the new bacterial taxonomy. *Appl Environ Microbiol* 73(16):5261–5267
32. Quinlan JR (1986) Induction of decision trees. *Mach Learn* 1(1):81–106
33. Breiman L (2017) Classification and regression trees. Routledge, Boca Raton
34. Liaw A, Wiener M (2002) Classification and regression by randomForest. *R News* 2(3):18–22
35. Gislason PO, Benediktsson JA, Sveinsson JR (2006) Random forests for land cover classification. *Pattern Recogn Lett* 27(4):294–300
36. Chen T, Cao Y, Zhang Y, Liu J, Bao Y, Wang C, Jia W, Zhao A (2013) Random forest in clinical metabolomics for phenotypic discrimination and biomarker selection. *Evid Based Complement Alternat Med* 2013:298183
37. Scott I, Lin W, Liakata M, Wood J, Vermeer CP, Allaway D, Ward J, Draper J, Beale M, Corol D (2013) Merits of random forests emerge in evaluation of chemometric classifiers by external validation. *Anal Chim Acta* 801:22–33
38. Ho TK (1998) Nearest neighbors in random subspaces. In: Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR). Springer, pp 640–648
39. Biau G (2012) Analysis of a random forests model. *J Mach Learn Res* 13 (Apr):1063–1095
40. Hapfelmeier A, Hothorn T, Ulm K, Strobl C (2014) A new variable importance measure

- for random forests with missing data. *Stat Comput* 24(1):21–34
41. Menze BH, Kelm BM, Masuch R, Himmelreich U, Bachert P, Petrich W, Hamprecht FA (2009) A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data. *BMC Bioinformatics* 10(1):213
42. Maker AV, Hu V, Kadkol SS, Hong L, Brugge W, Winter J, Yeo CJ, Hackert T, Buchler M, Lawlor RT, Salvia R, Scarpa A, Bassi C, Green S (2019) Cyst fluid biosignature to predict Intraductal papillary mucinous neoplasms of the pancreas with high malignant potential. *J Am Coll Surg* 228:721. <https://doi.org/10.1016/j.jamcollsurg.2019.02.040>
43. Tkachev V, Sorokin M, Mescheryakov A, Simonov A, Garazha A, Buzdin A, Muchnik I, Borisov N (2018) FLOating-window projective separator (FlowPS): a data trimming tool for support vector machines (SVM) to improve robustness of the classifier. *Front Genet* 9:717. <https://doi.org/10.3389/fgene.2018.00717>
44. Yerukala Sathipati S, Ho SY (2018) Identifying a miRNA signature for predicting the stage of breast cancer. *Sci Rep* 8(1):16138. <https://doi.org/10.1038/s41598-018-34604-3>
45. Cortes C, Vapnik V (1995) Support-vector networks. *Mach Learn* 20(3):273–297
46. Boser BE, Guyon IM, Vapnik VN (1992) A training algorithm for optimal margin classifiers. In: Proceedings of the fifth annual workshop on computational learning theory. ACM, pp 144–152
47. Bishop CM (2006) Pattern recognition and machine learning. Springer, Berlin
48. Ripley BD (1994) Flexible non-linear approaches to classification. In: From statistics to neural networks. Springer, Berlin, pp 105–126
49. Contreras-Jodar A, Nayan NH, Hamzaoui S, Caja G, Salama AAK (2019) Heat stress modifies the lactational performances and the urinary metabolomic profile related to gastrointestinal microbiota of dairy goats. *PLoS One* 14(2):e0202457. <https://doi.org/10.1371/journal.pone.0202457>
50. Park HG, Jang KS, Park HM, Song WS, Jeong YY, Ahn DH, Kim SM, Yang YH, Kim YG (2019) MALDI-TOF MS-based total serum protein fingerprinting for liver cancer diagnosis. *Analyst* 144:2231. <https://doi.org/10.1039/c8an02241k>
51. Quiros-Guerrero L, Albertazzi F, Araya-Valverde E, Romero RM, Villalobos H, Poveda L, Chavarria M, Tamayo-Castillo G (2019) Phenolic variation among *Chamaecrista nictitans* subspecies and varieties revealed through UPLC-ESI(–)-MS/MS chemical fingerprinting. *Metabolomics* 15(2):14. <https://doi.org/10.1007/s11306-019-1475-8>
52. Wang J, Yan D, Zhao A, Hou X, Zheng X, Chen P, Bao Y, Jia W, Hu C, Zhang ZL, Jia W (2019) Discovery of potential biomarkers for osteoporosis using LC-MS/MS metabolomic methods. *Osteoporos Int* 30:1491. <https://doi.org/10.1007/s00198-019-04892-0>
53. Grissa D, Petera M, Brandolini M, Napoli A, Comte B, Pujos-Guillot E (2016) Feature selection methods for early predictive biomarker discovery using untargeted metabolomic data. *Front Mol Biosci* 3:30. <https://doi.org/10.3389/fmolb.2016.00030>
54. Bayci AWL, Baker DA, Somerset AE, Turkoglu O, Hothem Z, Callahan RE, Mandal R, Han B, Bjorndahl T, Wishart D, Bahado-Singh R, Graham SF, Keidan R (2018) Metabolomic identification of diagnostic serum-based biomarkers for advanced stage melanoma. *Metabolomics* 14(8):105. <https://doi.org/10.1007/s11306-018-1398-9>
55. Catav SS, Elgin ES, Dag C, Stark JL, Kucukayyuz K (2018) NMR-based metabolomics reveals that plant-derived smoke stimulates root growth via affecting carbohydrate and energy metabolism in maize. *Metabolomics* 14(11):143. <https://doi.org/10.1007/s11306-018-1440-y>
56. Guo JG, Guo XM, Wang XR, Tian JZ, Bi HS (2019) Metabolic profile analysis of free amino acids in experimental autoimmune uveoretinitis rat plasma. *Int J Ophthalmol* 12(1):16–24. <https://doi.org/10.18240/ijo.2019.01.03>
57. Rodrigues-Neto JC, Correia MV, Souto AL, Ribeiro JAA, Vieira LR, Souza MT Jr, Rodrigues CM, Abdelnur PV (2018) Metabolic fingerprinting analysis of oil palm reveals a set of differentially expressed metabolites in fatal yellowing symptomatic and non-symptomatic plants. *Metabolomics* 14(10):142. <https://doi.org/10.1007/s11306-018-1436-7>
58. Wong M, Lodge JK (2012) A metabolomic investigation of the effects of vitamin E supplementation in humans. *Nutr Metab (Lond)* 9(1):110. <https://doi.org/10.1186/1743-7075-9-110>

59. Li Y, Chen M, Liu C, Xia Y, Xu B, Hu Y, Chen T, Shen M, Tang W (2018) Metabolic changes associated with papillary thyroid carcinoma: a nuclear magnetic resonance-based metabolomics study. *Int J Mol Med* 41(5):3006–3014. <https://doi.org/10.3892/ijmm.2018.3494>
60. Rezig L, Servadio A, Torregrossa L, Miccoli P, Basolo F, Shintu L, Caldarelli S (2018) Diagnosis of post-surgical fine-needle aspiration biopsies of thyroid lesions with indeterminate cytology using HRMAS NMR-based metabolomics. *Metabolomics* 14(10):141. <https://doi.org/10.1007/s11306-018-1437-6>
61. Westerhuis JA, van Velzen EJ, Hoefsloot HC, Smilde AK (2010) Multivariate paired data analysis: multilevel PLSDA versus OPLSDA. *Metabolomics* 6(1):119–128
62. Liquet B, Le Cao KA, Hocini H, Thiebaut R (2012) A novel approach for biomarker selection and the integration of repeated measures experiments from two assays. *BMC Bioinformatics* 13:325. <https://doi.org/10.1186/1471-2105-13-325>
63. Liu H, Motoda H (1998) Feature extraction, construction and selection: a data mining perspective, vol 453. Springer Science & Business Media, Norwell
64. Guyon I, Elisseeff A (2003) An introduction to variable and feature selection. *J Mach Learn Res* 3:1157–1182
65. Weston J, Elisseeff A, Schölkopf B, Tipping M (2003) Use of the zero-norm with linear models and kernel methods. *J Mach Learn Res* 3(Mar):1439–1461
66. Mladenic D, Grobelnik M (1999) Feature selection for unbalanced class distribution and naive bayes. In: ICML 1999, pp 258–267
67. Bozdogan H (1987) Model selection and Akaike's information criterion (AIC): the general theory and its analytical extensions. *Psychometrika* 52(3):345–370
68. Guan W, Zhou M, Hampton CY, Benigno BB, Walker LD, Gray A, McDonald JF, Fernández FM (2009) Ovarian cancer detection from metabolomic liquid chromatography/mass spectrometry data by support vector machines. *BMC Bioinformatics* 10(1):259
69. Platt J (1998) Sequential minimal optimization: a fast algorithm for training support vector machines
70. Kuhn M, Johnson K (2013) Applied predictive modeling, vol 26. Springer, New York
71. Behnamian A, Millard K, Banks SN, White L, Richardson M, Pasher J (2017) A systematic approach for variable selection with random forests: achieving stable variable importance values. *IEEE Geosci Remote Sens Lett* 14(11):1988–1992
72. Van Calster B, Vickers AJ (2015) Calibration of risk prediction models: impact on decision-analytic performance. *Med Decis Making* 35(2):162–169
73. Agresti A (2002) Categorical data analysis. Wiley, New York
74. Huang Y, Sullivan Pepe M, Feng Z (2007) Evaluating the predictiveness of a continuous marker. *Biometrics* 63(4):1181–1188
75. Holder LB, Haque MM, Skinner MK (2017) Machine learning for epigenetics and future medical applications. *Epigenetics* 12(7):505–514. <https://doi.org/10.1080/15592294.2017.1329068>
76. Chen C, Liaw A, Breiman L (2004) Using random forest to learn imbalanced data, vol 110. University of California, Berkeley, pp 1–12
77. Breiman L, Friedman J, Olshen RA, Stone CJ (1984) Classification and regression trees. Chapman & Hall, New York
78. Japkowicz N (2000) Learning from imbalanced data sets: a comparison of various strategies. In: AAAI workshop on learning from imbalanced data sets. Menlo Park, CA, pp 10–15
79. Maloof MA (2003) Learning when data sets are imbalanced and when costs are unequal and unknown. In: ICML-2003 workshop on learning from imbalanced data sets II, pp 2–1
80. Ling CX, Li C (1998) Data mining for direct marketing: problems and solutions. In: KDD 1998, pp 73–79
81. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) SMOTE: synthetic minority over-sampling technique. *J Artif Intell Res* 16:321–357
82. Kubat M, Matwin S (1997) Addressing the curse of imbalanced training sets: one-sided selection. In: ICML 1997. Citeseer, pp 179–186
83. Domingos P (1999) Metacost: a general method for making classifiers cost-sensitive. In: KDD 1999, pp 155–164
84. Cateni S, Colla V, Vannucci M (2014) A method for resampling imbalanced datasets in binary classification tasks for real-world problems. *Neurocomputing* 135:32–41
85. Drummond C, Holte RC (2003) C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In: Workshop on learning from imbalanced datasets II. Citeseer, pp 1–8

86. Collins GS, Reitsma JB, Altman DG, Moons KG (2015) Transparent reporting of a multi-variable prediction model for individual prognosis or diagnosis (TRIPOD): the TRIPOD statement. *BMC Med* 13(1):1
87. Cruickshank-Quinn CI, Jacobson S, Hughes G, Powell RL, Petracche I, Kechris K, Bowler R, Reisdorff N (2018) Metabolomics and transcriptomics pathway approach reveals outcome-specific perturbations in COPD. *Sci Rep* 8(1):17132
88. Regan EA, Hokanson JE, Murphy JR, Make B, Lynch DA, Beaty TH, Curran-Everett D, Silverman EK, Crapo JD (2010) Genetic epidemiology of COPD (COPD-Gene) study design. *COPD* 7(1):32–43. <https://doi.org/10.3109/15412550903499522>
89. Andersen SL, Briggs FBS, Winnike JH, Natanzon Y, Maichle S, Knagge KJ, Newby LK, Gregory SG (2019) Metabolome-based signature of disease pathology in MS. *Mult Scler Relat Disord* 31:12–21. <https://doi.org/10.1016/j.msard.2019.03.006>
90. Lee HS, Seo C, Hwang YH, Shin TH, Park HJ, Kim Y, Ji M, Min J, Choi S, Kim H, Park AK, Yee ST, Lee G, Paik MJ (2019) Metabolomic approaches to polyamines including acetylated derivatives in lung tissue of mice with asthma. *Metabolomics* 15(1):8. <https://doi.org/10.1007/s11306-018-1470-5>
91. Long NP, Yoon SJ, Anh NH, Nghi TD, Lim DK, Hong YJ, Hong SS, Kwon SW (2018) A systematic review on metabolomics-based diagnostic biomarker discovery and validation in pancreatic cancer. *Metabolomics* 14(8):109. <https://doi.org/10.1007/s11306-018-1404-2>
92. Regan EA, Hersh CP, Castaldi PJ, DeMeo DL, Silverman EK, Crapo JD, Bowler RP (2019) Omics and the search for blood biomarkers in COPD: insights from COPD-Gene. *Am J Respir Cell Mol Biol* 61:143. <https://doi.org/10.1165/rcmb.2018-0245PS>
93. Thévenot EA (2016) ropls: PCA, PLS (-DA) and OPLS (-DA) for multivariate analysis and feature selection of omics data
94. Rinaudo P, Boudah S, Junot C, Thévenot EA (2016) Biosigner: a new method for the discovery of significant molecular signatures from omics data. *Front Mol Biosci* 3:26
95. Zararsiz G, Goksuluk D, Korkmaz S, Eldem V, Duru IP, Unver T, Ozturk A, Zararsiz MG, klaR M, biocViews Sequencing, R (2014) Package ‘MLSeq’
96. Xia J, Psychogios N, Young N, Wishart DS (2009) MetaboAnalyst: a web server for metabolomic data analysis and interpretation. *Nucleic Acids Res* 37(suppl_2):W652–W660
97. Luan H, Ji F, Chen Y, Cai Z (2018) statTarget: a streamlined tool for signal drift correction and interpretations of quantitative mass spectrometry-based omics data. *Anal Chim Acta* 1036:66–72
98. Determan Jr CE, Determan Jr MCE (2015) Package ‘OmicsMarkR’
99. Rohart F, Gautier B, Singh A, Le Cao K-A (2017) mixOmics: an R package for ‘omics feature selection and multiple data integration. *PLoS Comput Biol* 13(11):e1005752
100. Al-Akwaa FM, Yunis B, Huang S, Alhajaji H, Garmire LX (2018) Lilikoi: an R package for personalized pathway-based classification modeling using metabolomics data. *GigaScience* 7(12):giy136
101. Gift N, Gormley IC, Brennan L, Gormley MC (2010) Package ‘MetabolAnalyze’
102. Gaude E, Chignola F, Spiliopoulos D, Spitaleri A, Ghitti M, García-Manteiga JM, Mari S, Musco G (2013) Muma, an R package for metabolomics univariate and multivariate statistical analysis. *Curr Metabol* 1(2):180–189
103. Palla P (2015) Information management and multivariate analysis techniques for metabolomics data. Università degli Studi di Cagliari
104. Kuhn M (2008) Building predictive models in R using the caret package. *J Stat Softw* 28(5):1–26



Chapter 17

Using MetaboAnalyst 4.0 for Metabolomics Data Analysis, Interpretation, and Integration with Other Omics Data

Jasmine Chong and Jianguo Xia

Abstract

MetaboAnalyst (www.metaboanalyst.ca) is an easy-to-use, comprehensive web-based tool, freely available for metabolomics data processing, statistical analysis, functional interpretation, as well as integration with other omics data. This chapter first provides an introductory overview to the current MetaboAnalyst (version 4.0) with regards to its underlying design concepts and user interface structure. Subsequent sections describe three common metabolomics data analysis workflows covering targeted metabolomics, untargeted metabolomics, and multi-omics data integration.

Key words Web server, Multivariate statistics, Enrichment analysis, Metabolic pathway analysis, Multi-omics integration

1 Introduction and Exploring MetaboAnalyst 4.0

Rapid advances in analytical chemistry, mass spectrometry (MS), and nuclear magnetic resonance (NMR) spectroscopy have dramatically increased the size and speed at which metabolomics data can be obtained. However, the inability of researchers to extract meaningful biological insights from these increasingly large and complex datasets has now become a major barrier in current metabolomics research and applications. There is an urgent demand for user-friendly and easily accessible bioinformatic tools to bridge the gap between data generation and biological insights.

MetaboAnalyst is an easy-to-use web-based tool suite that permits users to perform a wide variety of metabolomics data analysis tasks. It was first released in 2009 with a single module for metabolomics data processing and statistical analysis [1] and has since been continuously updated to meet the ever-evolving needs for metabolomics data analysis [2–4]. The current release (v4.0) contains 12 modules which can be placed into four general categories: (1) exploratory statistical analysis, (2) functional analysis, (3) data integration and systems biology, and (4) data processing

and utility functions. The MetaboAnalyst web server has been carefully designed so that each module workflow follows an intuitive yet consistent process beginning with data uploading, followed by data processing and analysis. MetaboAnalyst supports the analysis of various common data types obtained from both targeted and untargeted metabolomics. Its companion R package, MetaboAnalystR, was made available in 2018 to complement the web server with improved flexibility, transparency, and scalability for advanced and high-throughput metabolomics data analysis [5]. The package was recently upgraded to version 2.0 to support raw spectral pre-processing and functional interpretation for LC-MS based global metabolomics [6].

The objective of this chapter is to introduce the reader to three typical metabolomics data analysis workflows using MetaboAnalyst 4.0. The chapter is divided into four sections. Section 1 focuses on introducing the overall layout and design concepts of MetaboAnalyst. Section 2 provides an example of a targeted metabolomics data workflow, including commonly used statistical analyses as well as pathway and enrichment analysis. Section 3 goes through an untargeted metabolomics workflow from raw MS peaks to functional interpretation. Finally, Section 4 describes an example of using MetaboAnalyst for integrative analysis of metabolomics and transcriptomics data.

This section will guide users through MetaboAnalyst, specifically how to understand its modular workflow, navigation tree, analysis report generation, and R Command History.

1.1 *Understanding the MetaboAnalyst Layout*

1. To begin, open a preferred web browser and enter the MetaboAnalyst web address: <https://www.metaboanalyst.ca/>.
2. At the top of the home page, click the “>>click here to start<<” hyperlink to enter the “Module Overview” page.
3. In the center of the “Module Overview” page is a circular “clock” showing 12 different modules currently available in MetaboAnalyst 4.0 (Fig. 1). All modules follow the same general flow: data upload, data processing and data analysis. To briefly demonstrate this, select the “Statistical Analysis” button on the top left of the circular panel.
4. The Statistical Analysis data upload page should now be visible (Fig. 2). On the left-hand side is the navigation tree, which guides users through all analysis steps of the selected module. This navigation tree is specific for each module. Note the “Upload” hyperlink is highlighted in blue, representing the current step. On the right-hand side is the R Command History panel, which displays all underlying R commands as they occur in real-time. The aim of this panel is to improve the reproducibility and transparency of the MetaboAnalyst web server. These R commands can be used directly by the companion MetaboAnalystR package to reproduce one’s results [5, 6].

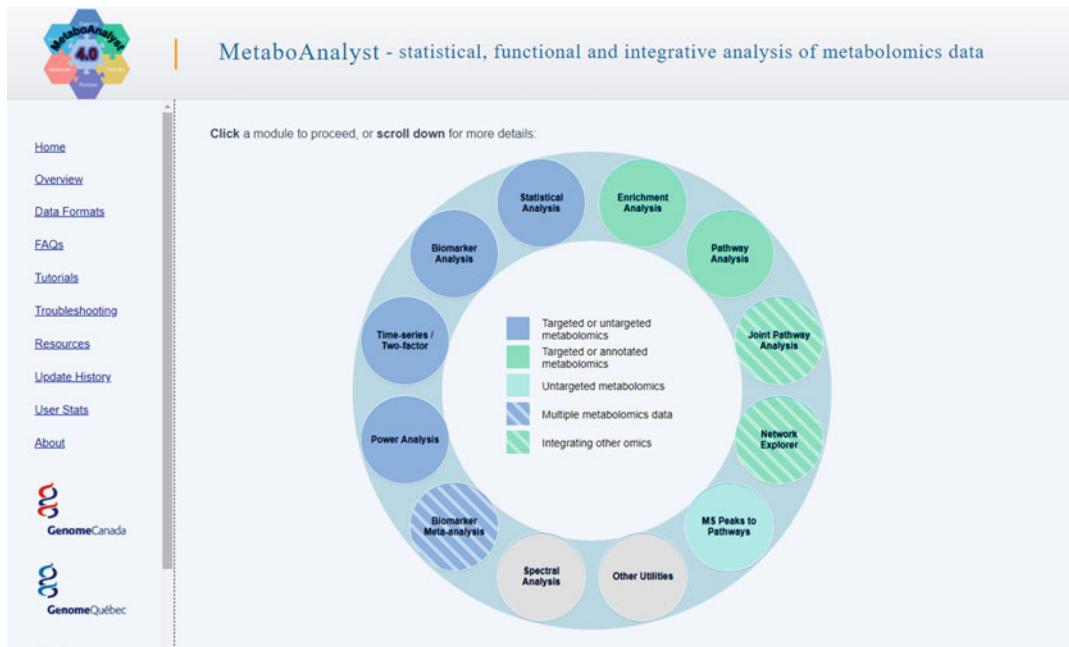


Fig. 1 A screenshot of the MetaboAnalyst module overview

The screenshot shows the "Upload your data" section of the MetaboAnalyst interface. On the left, a sidebar has "Upload" selected, with options for Processing, Normalization, Statistics, Download, and Exit. The main area has two tabs: "1) Upload your data" and "2) Try our test data". The "1) Upload your data" tab contains fields for Tab-delimited text (.txt) or comma-separated values (.csv) files. It includes dropdowns for Data Type (Concentrations, Spectral bins, Peak intensity table), Format (Samples in rows (unpaired)), and Data File (Choose File). There is also a "Zipped Files (.zip)" section with similar fields for Data Type (NMR peak list, MS peak list, MS spectra), Data File (Choose File), and Pair File (Choose File). To the right, there is an "R Command History" panel showing "no commands found". The "2) Try our test data" tab lists several pre-loaded datasets:

Data Type	Description
Concentrations	Metabolite concentrations of 77 urine samples from cancer patients measured by 1H NMR (Eisner R. et al.). Group 1 - cachexic; group 2 - control.
Concentrations	Metabolite concentrations of 39 rumen samples measured by proton NMR from dairy cows fed with different proportions of barley grain (Amelai BN. et al.). Group label - 0, 15, 30, or 45 - indicating the percentage of grain in diet.
NMR spectral bins	Binned 1H NMR spectra of 50 urine samples using 0.04 ppm constant width (Psichogios NG. et al.). Group 1 - control; group 2 - severe kidney disease.
NMR peak lists	Peak lists and intensity files for 50 urine samples measured by 1H NMR (Psichogios NG. et al.). Group 1 - control; group 2 - severe kidney disease.

Fig. 2 A screenshot of the Statistical Analysis data upload page within MetaboAnalyst

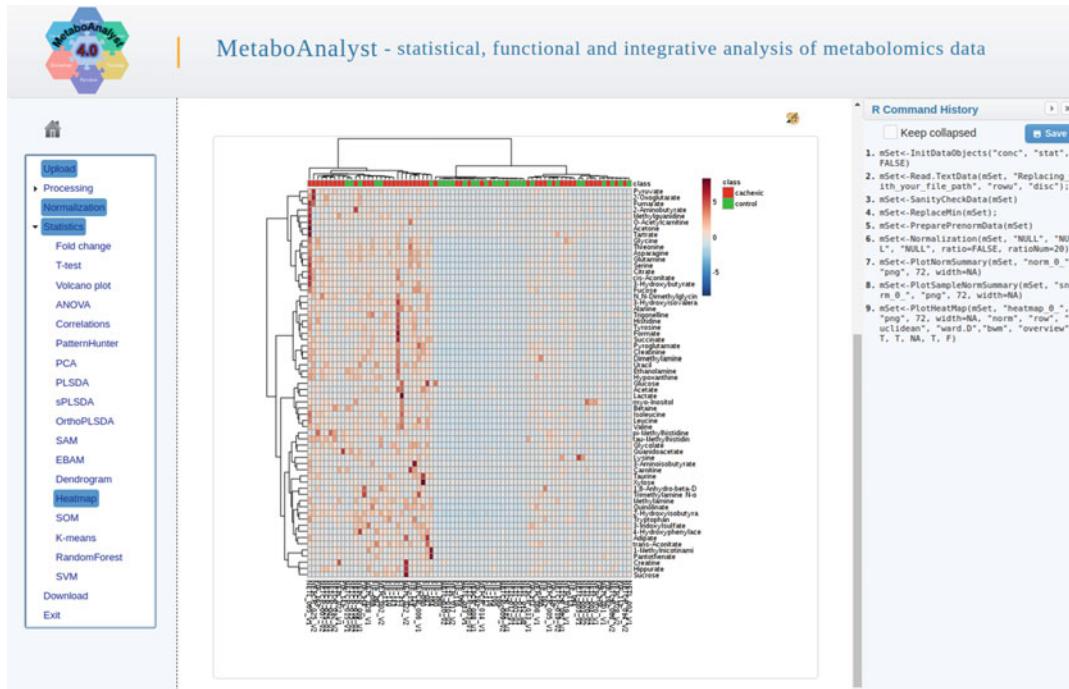


Fig. 3 A screenshot of the heatmap generated in the Statistical Analysis module within MetaboAnalyst. To the left of the image is the navigation tree and to the right is the R Command History panel

5. Select any example dataset listed at the bottom of the page and click “Submit.” For the purposes of this section, we will speed through the next few pages. A detailed guide to data processing and statistical analysis is provided below in Subheading 2.
6. The next page is the Data Integrity Check, click “Skip” on the bottom of the page. Note that the R commands are updated on the right upon each user action. After is the Normalization page, keep all options set to “None”, press “Normalize” (bottom left) and then “Proceed” (bottom right). Following this is the “Analysis View” page, which shows all the analysis options available in MetaboAnalyst. Click “Heatmaps” under the “Cluster Analysis” subheading. A heatmap should now be visible in the center of the screen (Fig. 3). Take some time to adjust all available parameters at the top of the page to update the heatmap, such as switching the “View Mode” from “Overview” to “Detail View.”
7. To download this plot (and any other plot in MetaboAnalyst) as a high-resolution image, click the paint palette icon at the top right of the page. The “Graphics Center” dialog will now appear, where users can adjust the format, resolution, and size of their image to download. Click “Submit” and the download link will now appear at the bottom of the dialog. Right click the hyperlink and select “Save link as...” to download the image.

8. Click the “Download” hyperlink from the left-side navigation tree to enter the “Results Download” page. Here, users can download all results and the R Command History from the “Download.zip” hyperlink in the downloads table. Press the blue “Generate Report” button to create a PDF report of the entire analysis. A blue “Analysis Report” hyperlink will appear on the right, containing a link to the created report. This report contains all selected parameters, generated figures and tables. Click on the link and the report will appear in a new tab of the web-browser. Scroll through the report, which first begins with an introduction to the selected module, then data processing steps, followed by the analyses and R Command History. Note that the Analysis Report contains the updated heatmap figure. To save the report, right-click the page and select “Save as...” to download it.
9. Press the “Logout” button to exit the session and return to the MetaboAnalyst home page.

2 A Targeted Metabolomics Workflow

A targeted metabolomics study involves the identification and quantification of a predefined set of metabolites. It is often applied for biomarker discovery (i.e., identification of key metabolites associated with a phenotype) or to study the dynamics of certain metabolic pathways of interest. The aim of analyzing targeted metabolomics data is therefore to uncover important metabolites that are (1) significantly different between phenotypes, and/or (2) functionally coherent with known molecular and metabolic mechanisms. The next section demonstrates how to use MetaboAnalyst to perform various statistical and functional analyses of data generated from targeted metabolomics studies.

2.1 Data Upload and Processing

Metabolomics data contains both biologically relevant variation—that is, changes in the metabolome induced by different experimental conditions, and unwanted variation introduced during sample preparation, instrumentation, and even data preprocessing (e.g., peak picking, grouping, and alignment). The aim of data processing is to enhance biologically relevant signals while reducing unrelated variations in the data [7]. Caution must be applied at this step to select the most appropriate methods as it can greatly influence the downstream interpretation of results [8, 9]. Researchers should keep in mind the aim of their study, the structure of their data, as well as the statistics they wish to use [7–9]. Methods for data processing within MetaboAnalyst include missing value imputation, filtering, scaling, centering, and transformations. In this section we demonstrate how users can process their data using MetaboAnalyst 4.0.

1. Go to the MetaboAnalyst home page and click “>>click here to start<<” to enter the “Module Overview” page. Select the “Statistical Analysis” button to enter the corresponding data upload page. There are two sections—the top section allows users to upload their own data, and the bottom section allows users to explore the module using an example data. Select the first dataset (77 urine metabolomics data from a study on cancer cachexia [10]) and click the “Submit” button at the bottom of the page.
2. The “Data Integrity Check” page should appear next. This page informs users how their data was read in by MetaboAnalyst (e.g., summary of samples, features, missing values, and metadata) and if their data passed all quality checks. As this dataset passed the integrity check and no missing values were detected, click the “Skip” button on the bottom right of the page.
3. The “Normalization” page should now appear. The normalization methods are categorized into three groups: (1) sample normalization, (2) data transformation, and (3) data scaling. Sample normalization is used to adjust potential overall differences among samples (i.e., those caused by volume or mass differences during sample preparation). Metabolite concentrations in urine samples should be adjusted for the dilution effect. MetaboAnalyst provides many different methods for this purpose including the traditional normalization by a reference compound (i.e., *Creatinine* in this case), normalization by sum/median, as well as the probabilistic quotient normalization (PQN) [11] using a particular reference sample or a pooled average sample from a reference group. For demonstration purposes, we will use the traditional approach. Select “Normalize by a reference feature” under the subheading “Sample normalization.” A dialog will appear. Select “Creatinine” and click “Submit” to close the dialog. Next, set Data transformation to “Log transformation” and Data scaling to “None.” Click “Normalize” to process the data.
4. To view a graphical summary of the resulting data normalization, click “View Result.” The distribution of the normalized data now follows a “bell-shaped” curve compared to the original data. Click the “X” on the top right of the dialog to close the dialog and press “Proceed” on the bottom right of the page.
5. The “Data Analysis Exploration” page should now be displayed. MetaboAnalyst 4.0 currently offers 18 statistical methods for metabolomics data analysis. These methods are organized into five general categories: (1) univariate analysis, (2) chemometrics analysis, (3) significant feature identification,



Fig. 4 A screenshot of a volcano plot created using an example dataset within MetaboAnalyst highlighting glucose in a boxplot

(4) cluster analysis, and (5) classification & feature selection. For the sake of brevity, only a handful of commonly used statistical methods will be demonstrated.

2.2 Univariate Analysis to Identify Significant Features

6. Click “Volcano plot” to view the volcano plot of the data using default parameters. Points highlighted in red are significant based on the default p -value cutoff of 0.1 and fold-change threshold of 2.0. Click on the single red point to view a boxplot showing the concentrations of the selected feature within each group (Fig. 4). Glucose shows a greater concentration in cachexic patients as compared to controls.
7. There are two icons on the top-right of the plot. Click on the mini table icon to enter the “Feature Details View” of the detailed result table from volcano analysis. This table shows compound names, fold-changes and p -values of statistically significant features. Click on any feature name to view boxplot summaries of both the original and normalized concentrations.

2.3 Multivariate Exploratory Data Analysis

Multivariate methods can be classified as unsupervised or supervised. Unsupervised methods do not consider class labels during model building and are therefore suitable for data exploration to identify inherent patterns in the data. In comparison, supervised methods consider class labels and are often used for biomarker discovery and classification. The next part will provide instructions

on how to perform one unsupervised method—principal component analysis (PCA), and two supervised methods—partial least squares discriminant analysis (PLS-DA) and random forest (RF).

8. To obtain a visual overview of the data, click “Principal Component Analysis (PCA)” from the “Data Analysis Exploration” page or the “PCA” link from the navigation tree to enter the PCA analysis page. The results from PCA are shown in six tabs. The first tab shows pairwise score plots between the top five principal components (PCs). From this default view, it appears that the two groups overlap significantly with each other.
9. Click the “3D Plot” (the fourth tab) to explore the PCA results in an interactive 3D score (on the left) and loading plot (on the right) of the first three PCs. The score plot is used to explore the underlying structure of the data, and the loading plot is used to assess which features have the greatest influence on each component. Use your mouse to rotate and zoom in and out of each plot (Fig. 5). Note that rotations are synchronized between the two 3D plots. From the direction of separation in the score plot, users can easily identify features with strong contributions to the separation in the loading plot. Click on any feature in the loading plot to view a boxplot comparing the concentrations of the selected feature between the two groups.
10. Move on to supervised multivariate statistical methods. Click the “PLS-DA” hyperlink from the navigation tree to enter the PLS-DA analysis page. This page contains seven tabs. The second tab shows the 2D Score Plot of the PLS-DA classification model. Compared to PCA, the PLS-DA score plot shows better separation between the two groups.

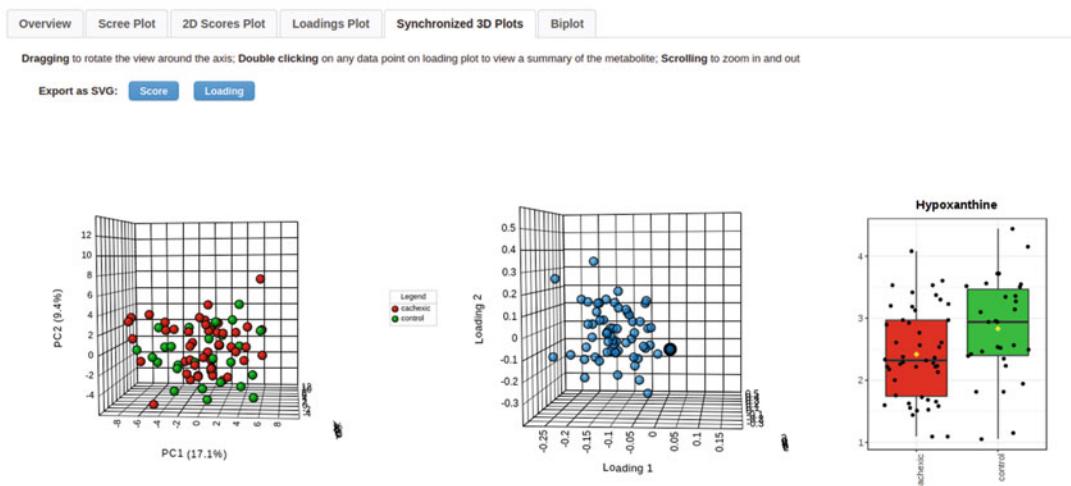


Fig. 5 A screenshot of a 3D PCA score and loading plot. *Hypoxanthine* is highlighted with a boxplot

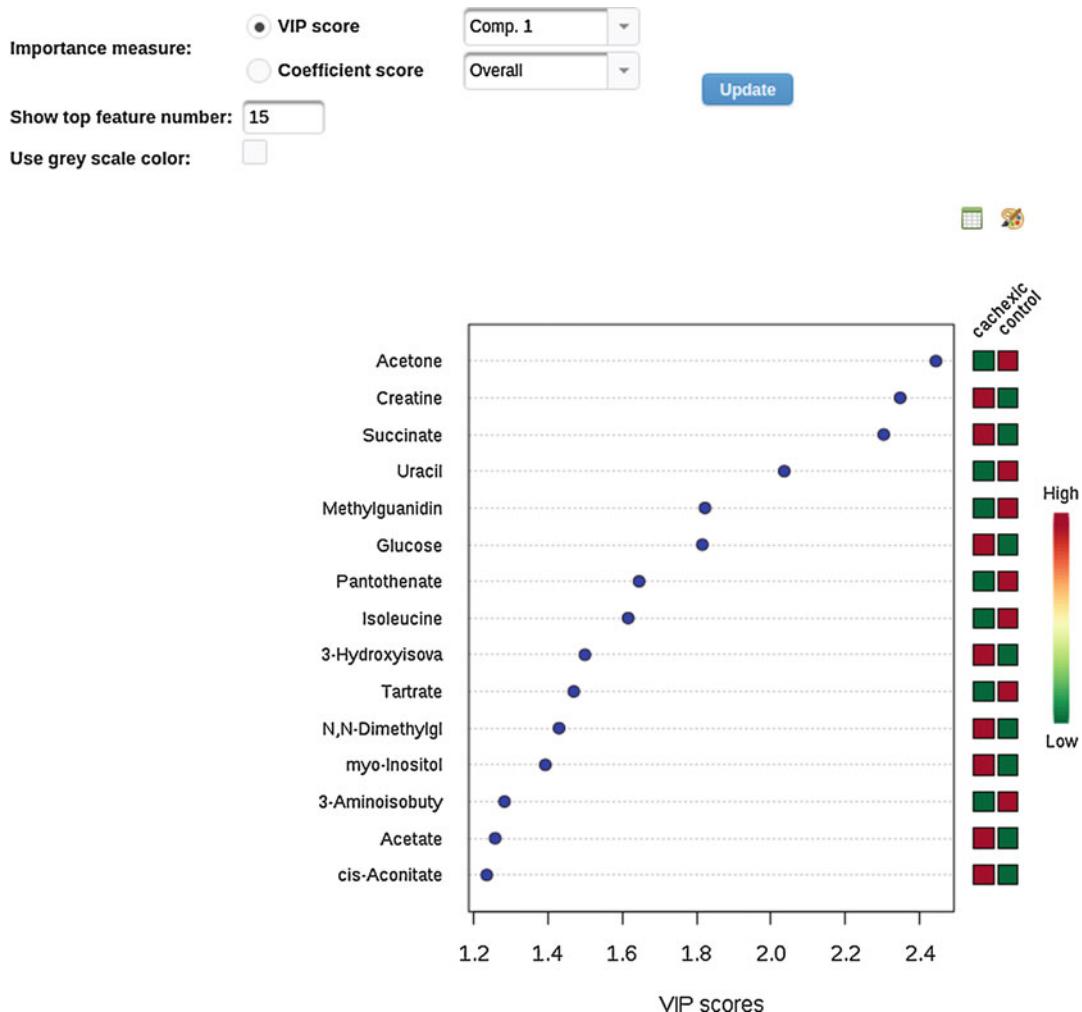


Fig. 6 A screenshot of a PLS-DA VIP scores plot highlighting the top 15 most important features

11. Next, click “3D Plot” to view the three-dimensional PLS-DA score and loading plots (further details in **step 9** above). The 3D score plot shows good separation between cachexic and control patients. The synchronized loading plot can be used to explore which features drive the separation between the two groups. For instance, click upon the upper right most feature (*Acetone*). The box plot of *Acetone* shows higher concentration in controls compared to cachexic patients.
12. To identify the important features of the PLS-DA model, select the “Imp. Features” tab. A plot summarizing the top most important features will appear (Fig. 6). MetaboAnalyst supports two importance measures that summarize the contribution a variable makes to the PLS-DA model—the variable importance in projection (VIP) and the weighted sum of

absolute regression coefficients (coef.). By default, it shows the top 15 features according to their VIP scores. The colored boxes to the right of the plot indicate whether the average concentration of a feature is high or low relative to each other. From the plot, *acetone*, *uracil*, and *methylguanidin* are higher in controls, while *creatine* and *succinate* are higher in cachexic patients.

13. Lastly, we will explore the Random Forest (RF) method. RF is an ensemble machine learning method. It builds and combines multiple decision trees to improve accuracy and prediction. Click the “RandomForest” link on the navigation tree. By default, MetaboAnalyst sets the number of trees built to 500. Users can adjust the number of decision trees to see if the results can be improved. For instance, change this to 2000 and then click “Update.” Notice that the OOB error improves from 0.26 to 0.247. However, these values may change due to the random nature of this algorithm.
14. To view the important features selected by the RF classifier, select the “Var. Importance” tab. The plot is similar to the VIP score plot from **step 12** except here, features are selected based on how much they increase OOB error when left out. The top three features are *uracil*, *isoleucine*, and *creatine*.

2.4 Functional Interpretation of Targeted Metabolomics Data

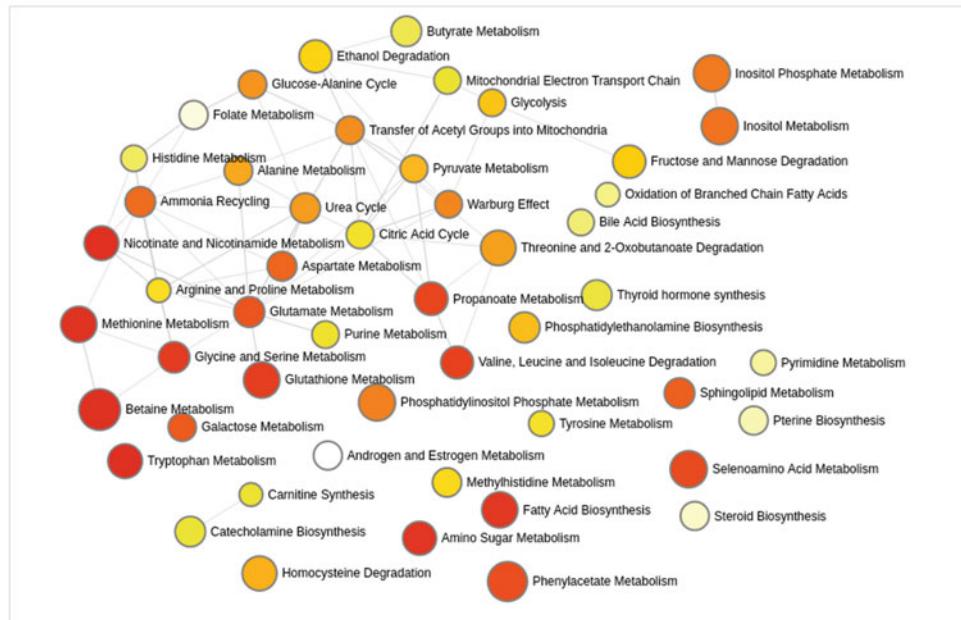
MetaboAnalyst provides two modules for functional interpretation of targeted metabolomics data: (1) metabolite set enrichment analysis, and (2) metabolic pathway analysis. Both methods are based on existing biological knowledge to determine if specific groups of metabolites (i.e., those involved in pathways [12] or metabolite sets [13]) are significantly enriched in users’ data. Users can directly upload a metabolite concentration table, or a list of important metabolites obtained from the previous statistical analysis. The next section will first go through metabolic enrichment analysis followed by metabolic pathway analysis.

2.4.1 Metabolite Set Enrichment Analysis

1. Return to the MetaboAnalyst home page by clicking the home icon on the top of the navigation tree. Click “>>click here to start<<” to enter the “Module Overview” page and select the “Enrichment Analysis” button to enter the data upload page.
2. On the Enrichment Analysis upload page, click the “A concentration table” panel and select the example data named “Data 1”. This is the same data used previously for statistical analysis. Click “Submit” to upload the data.
3. The “Compound Name/ID Standardization” page should appear. The aim of compound name standardization is to match compounds from the user’s data to those curated in the MetaboAnalyst knowledgebase. Compounds without

exact matches will be highlighted in yellow with a “View” button shown in the last column to allow users to perform approximate search. As there are no compounds without matches, press the “Submit” button at the bottom of the page.

4. Next is the Data Integrity Check page. Click “Skip” to continue.
5. The “Normalization overview” page should now be displayed. Keep the Sample Normalization and Data Transformation options to none, and select “Auto scaling” under the Data Scaling options. Click “Normalize” and then “Proceed”.
6. The following page shows the parameters for enrichment analysis. For demonstration purposes, select the “Pathway-associated metabolite sets [SMPDB],” which consists of 99 pathways from the Small Molecule Pathway Database [14], and keep all other parameters to their default. Click “Submit” to continue.
7. The Enrichment Analysis results page consists of two tabs, each with two parts. The first part of each tab consists of a graphical representation of the results, either as a network or a bar chart. The bottom part of each tab shows a detailed table of the enrichment analysis results.
8. Beginning with the Network View (Fig. 7), each circle within the network (node) represents a metabolite set and is colored and sized based on its *p*-value and fold enrichment (hits/expected), respectively. Two metabolite sets are linked together with a line (edge) if they share >20% of metabolites. The network is also interactive, permitting users to drag and drop nodes, move left and right, as well as zoom in and out. From the network view, *Betaine Metabolism* and *Glycine and Serine Metabolism* are two of the most enriched metabolite sets and are also connected with several other metabolite sets.
9. Scroll down to the enrichment analysis results table. Listed are the pathway names, pathway size, number of compound hits, and enrichment statistics. Click “View” under the Details columns for further information about a pathway. A popup will appear. At the top of the popup is a bar-chart of all matched compounds, colored based on its association to a phenotype. On the bottom are the common names of all metabolites in the selected pathway, with matches highlighted in red.
10. Return to the parameters page and this time select “Pathway-associated metabolite sets (KEGG),” which consists of 80 pathways from the KEGG database [12]. Keep all other parameters as default and click “Submit” at the bottom of the page.
11. From the network, based on their larger size and red coloring, *Glycine, serine and threonine metabolism* and *Valine,*



	Metabolite Set	Total	Hits	Statistic	Expected	P value	Holm P	FDR	Details
	Tryptophan Metabolism	60	5	10.522	1.3158	5.6649E-5	0.0041353	0.0031692	View
	Nicotinate and Nicotinamide Metabolism	37	3	10.48	1.3158	1.2147E-4	0.0087458	0.0031692	View
	Betaine Metabolism	21	2	14.311	1.3158	1.4515E-4	0.010305	0.0031692	View
	Methionine Metabolism	43	4	11.386	1.3158	1.7852E-4	0.012496	0.0031692	View
	Amino Sugar Metabolism	33	3	10.105	1.3158	2.2099E-4	0.015248	0.0031692	View
	Fatty Acid Biosynthesis	35	2	11.385	1.3158	2.9572E-4	0.020109	0.0031692	View
	Glycine and Serine Metabolism	59	10	8.6753	1.3158	3.0389E-4	0.020361	0.0031692	View

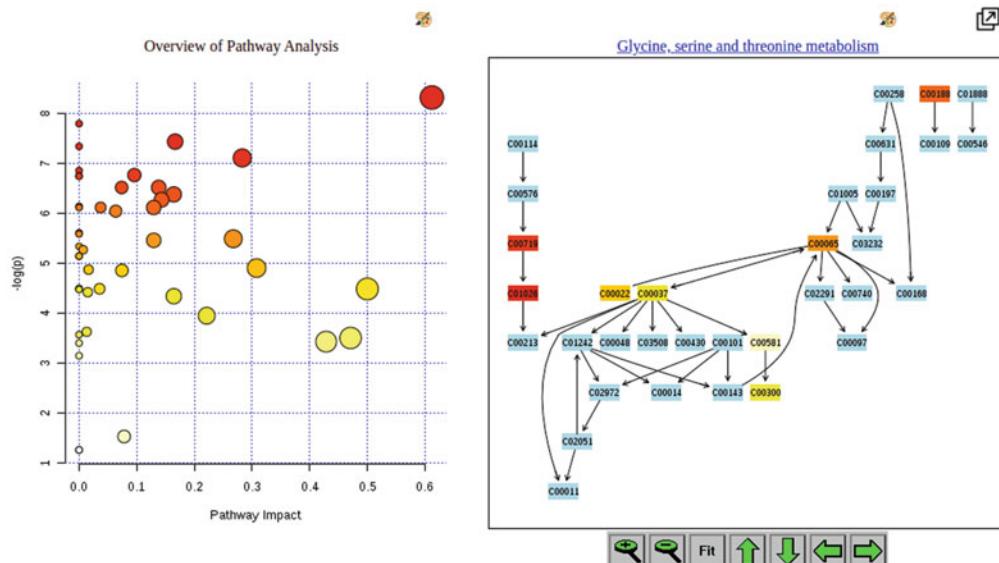
Fig. 7 A screenshot of the interaction network of the SMPDB pathway metabolite set created using the Enrichment Analysis module of MetaboAnalyst

leucine and isoleucine biosynthesis are two of the top most enriched pathway sets. Enrichment analysis using the two different metabolite sets consistently identified alterations in glycine metabolism, which is consistent with cachexia literature [13, 15–17].

2.4.2 Metabolic Pathway Analysis

1. Return to the MetaboAnalyst home page by selecting the home icon on the top of the navigation tree. Click “>>click here to start<<” to enter the “Module Overview” page and select the “Pathway Analysis” button to enter the data upload page.
2. On the Pathway Analysis upload page, click the “A concentration table” panel and select the empty box next to “Use the example data.” This is the same data used previously for enrichment analysis. Click “Submit” to upload the data.
3. The compound standardization, data integrity check, and normalization are identical to steps 3–5 in Section 2.4.1 above.

4. The next page shows all the options for metabolic pathway analysis. Keep the default parameters for the pathway analysis algorithm and KEGG Version. As the example samples were obtained from humans, select “*Homo sapiens* (KEGG)” and click “Submit” at the bottom of the page to perform pathway analysis.
5. The Pathway Analysis results are displayed on the following page. The “Overview of Pathway Analysis” plot on the left of the screen provides a graphical summary of the pathway analysis. In the plot, all matched pathways are represented as circles. The color and size of each circle corresponds to its *p*-value and pathway impact value, respectively. From this plot, *Glycine, serine and threonine metabolism* was the most enriched pathway (largest *p*-value) and had a pathway impact of 0.612. Click upon any circle and its corresponding pathway plot will appear to the right (Fig. 8). Matched compounds in the pathway plot are colored in red, orange, or yellow based on its corresponding *p*-value. Press on any of these matched compounds and a popup will appear showing a boxplot of the concentrations of the selected metabolite between the two



Click the corresponding Pathway Name to view its graphical presentation; click Match Status to view the pathway compounds (with matched ones highlighted).

Pathway Name	Match Status	<i>p</i>	-log(<i>p</i>)	Holm <i>p</i>	FDR	Impact	Details
Glycine, serine and threonine metabolism	8/33	2.4586E-4	8.3107	0.01131	0.0059936	0.61192	KEGG SMP
Valine, leucine and isoleucine biosynthesis	4/8	4.1491E-4	7.7874	0.018671	0.0059936	0.0	KEGG SMP
Aminocycl-tRNA biosynthesis	13/48	5.9417E-4	7.4283	0.026144	0.0059936	0.16667	KEGG
Valine, leucine and isoleucine degradation	3/40	6.5285E-4	7.3342	0.028073	0.0059936	0.0	KEGG SMP

Fig. 8 A screenshot of the graphical overview of Pathway Analysis results in MetaboAnalyst, highlighting *Glycine, serine and threonine metabolism*

groups, as well as its *p*-value and pathway importance score. The pathway viewer is interactive, permitting zooming in and out and moving around the viewer.

6. Scroll down the page to view the numeric details of the Pathway Analysis results. The table contains the names of all matched pathways, the number of compound matches from the uploaded data, statistical values, and links to KEGG and SMPDB. Clicking on any pathway name will update the pathway plot above.
7. Click the “Download” hyperlink from the navigation tree. This page permits users to download all generated plots, tables, and analysis report. Click “Exit” on the navigation tree to end this session.

3 An Untargeted Metabolomics Workflow

Untargeted metabolomics, also known as global metabolomics, aims to measure all possible metabolites within samples without a priori knowledge of the metabolome. Due to its high sensitivity and wide coverage, high resolution LC-MS is currently the dominant platform in global metabolomics [18, 19]. A typical LC-MS based metabolomics experiment can generate 10,000 or more peaks (features) characterized by their mass and retention times. However, as a single peak can potentially match multiple compounds within the given mass range, peak annotation requires significant efforts to search through compound databases and perform tandem MS experiments. Due to this challenge, functional interpretation of global metabolomics data is not straightforward, as classical metabolic pathway enrichment analysis requires metabolites as input, not MS peaks. To address this bottleneck, Li et al. proposed a novel approach, named mummichog, to directly infer pathway activities from peak lists by leveraging the collective power of metabolic pathways, without requiring a priori metabolite identification [20]. This algorithm assumes that a certain degree of random errors during individual peak assignment will not change the collective behavior jointly determined by all metabolites involved in the pathways. This concept has been recently adapted to the popular Gene Set Enrichment Analysis (GSEA) algorithm [21] in MetaboAnalystR 2.0 and MetaboAnalyst 4.0. Mummichog is based on over representation analysis (ORA) to test if certain pathways are enriched in the significant peaks as compared to null models based on peak lists of the same size randomly drawn from the inputted peak list. In comparison, GSEA is a cutoff free method that evaluates the overall differences of two distributions based on Kolmogorov–Smirnov tests. The following section will go through

this module to showcase the functional interpretation of preranked MS peaks.

3.1 Functional Interpretation of MS Peaks

1. From the MetaboAnalyst home page, click “>>click here to start<<” to enter the “Module Overview” page. Next, select the “MS Peaks to Paths” button to enter the data upload page. Notice that the page is divided into two parts, the top where users can upload their own peak list or peak intensity table and the bottom where users can select an example dataset to explore the module.
2. To use the example data, select the radio button next to IBD. The example data contains peak list or peak intensity table data obtained from a subset of fecal samples from 12 pediatric IBD patients and 12 controls collected from the Integrative Human Microbiome Project [22]. Click “Submit” to continue.
3. The “Data Integrity Check” page should appear, which informs users whether the quality of the uploaded data was suitable for further analysis. Click “Proceed” to move forward.
4. Next is the “Library View,” where users can select which algorithm to use to calculate pathway enrichment (mummichog, GSEA, or their integration) and their desired species-specific pathway library. For the mummichog algorithm, users can either use the “Default cutoff,” which uses the top 10% or top 500 peaks (whichever is smaller based on the user’s uploaded MS peaks) as significant or specify a *p*-value cutoff to indicate which features will be considered significant. For demonstration purposes, we will first use the mummichog algorithm and then the third approach—integrating mummichog and GSEA. Select the mummichog algorithm and click the button next to “Default *p*-value cutoff.” Keep the pathway library selection to “*Homo sapiens* [MFN],” which was constructed from the human BiGG and Edinburgh genome-scale metabolic networks [23].
5. The following page shows the predicted pathway activity using the mummichog algorithm. The pathway summary plot at the top of the page displays all matched pathways as circles (Fig. 9). The color and size of each circle corresponds to its *p*-value and enrichment factor, respectively. The enrichment factor of a pathway is calculated as the ratio between the number of significant pathway hits and the expected number of hits within the pathway. The plot is also interactive. Hover over any circle to view the name of the pathway, the enrichment factor, and -log₁₀ *p*-value. Further, double-click any circle to view the list of potential compound hits from the uploaded peak list to those in the selected pathway. Significant hits are shown in red and nonsignificant hits are blue. Meanwhile, the pathway enrichment analysis results table below contains the total size of the pathway, the number of compound matches, and

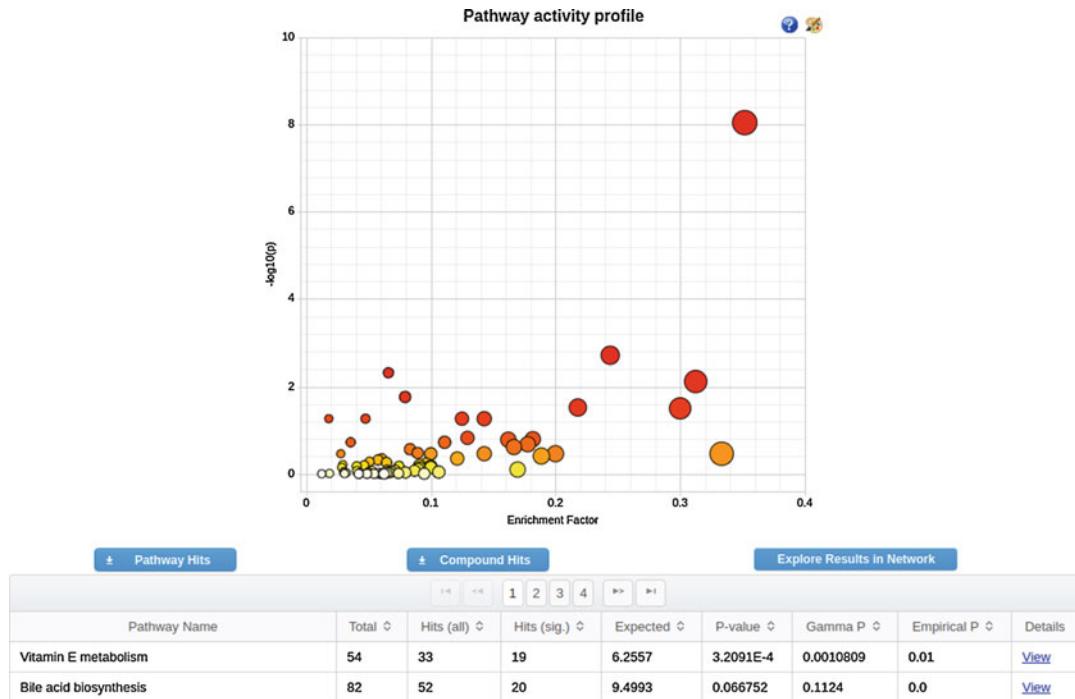


Fig. 9 A screenshot of the pathway activity profile plot in MetaboAnalyst summarizing the results of the mummichog algorithm in the MS Peaks to Paths module

raw/adjusted *p*-values. The top ranked pathways are *Vitamin E metabolism* and *Bile acid biosynthesis*.

6. Users can download the pathway results table from the “Pathway Hits” button at the top left of the table, as well as all matched metabolites from the uploaded list of MS peaks from the “Compound Hits” button at the top center of the table. Users can also visualize the pathway results in the context of a KEGG Global Metabolic Network by clicking the “Explore Results in Network” button.
7. Return to the “Library View” by selecting “Set parameter” from the navigation tree. For the pathway algorithms, maintain the mummichog algorithm as checked and also select GSEA. Keep the pathway library selection to “Homo Sapiens [MFN]” and press “Submit” to perform the integrated mummichog and GSEA analysis.
8. The results of the combined mummichog and GSEA *p*-values are shown on the following page (Fig. 10). The page is split into two parts, with the top showing a plot summarizing the combined pathway analysis, and the bottom containing a detailed results table. The integrated pathway summary plot shows the results of the mummichog (*y*-axis) and GSEA (*x*-axis) *p*-value combination, with the circles representing

Integrated Pathway Activity Profile

Mouseover any circle to view its name; click to view compound matches. More details are in the table at bottom

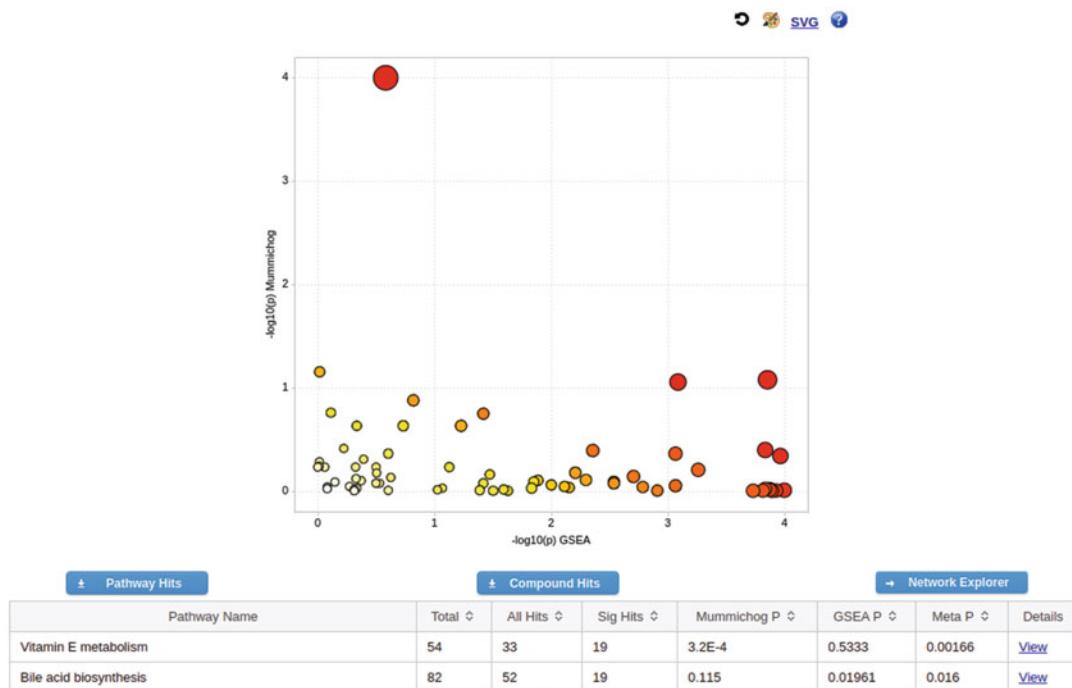


Fig. 10 A screenshot of the pathway summary plot integrating mummichog (*y*-axis) and GSEA (*x*-axis) *p*-values. The size and color of the circles correspond to the transformed combined *p*-values

matched pathways. The size and color of the circles correspond to their transformed combined *p*-values. *Bile acid biosynthesis* was consistently identified as one of the most perturbed pathways in pediatric IBD patients as compared to healthy controls using each algorithm individually as well as integrated. Importantly, bile acids are well-known to play an important role in the onset or progression of inflammatory bowel disease [24–26].

- Click on the “Download” hyperlink from the navigation tree to enter the “Download” page. Here, the analysis report, R command history, images, and results tables can be downloaded. Alternatively, click the home icon to exit the session.

4 Knowledge-Based Multi-Omics Data Integration

Generation of multiple omics data from the same set of samples is becoming increasingly common nowadays. However, efficient multi-omics data integration and interpretation has yet to be fully

realized. One common approach is to analyze each omics individually and then combine the resulting lists of significant molecules onto existing biological knowledgebases for joint analysis and visualization. For instance, to integrate metabolomics and proteomics/transcriptomics data, a common approach is to co-project metabolites and genes/proteins onto the same metabolic pathways or networks for visual exploration as well as for enrichment analysis. The following section will illustrate multi-omics integration first based on pathways using the Joint Pathway Analysis module, followed by network-based integration using the Network Explorer module.

4.1 Joint Pathway Analysis

1. From the MetaboAnalyst home page, click “>>click here to start<<” to enter the “Module Overview” page. Select the “Joint Pathway Analysis” button on the middle-right of the circular panel to enter the Data Upload page.
2. On the “Joint Upload” page, click the checkbox next to “Use our example data” to use the example data. A list of genes and metabolites will show in the corresponding text areas and the organism will be set to *Homo sapiens*. This example data contains a subset of transcriptome and metabolome data from a study of intrahepatic cholangiocarcinoma [27]. Click “Submit” to continue.
3. The “Compound/Gene Mapping” page is similar to the “Compound Name Standardization” page from the enrichment/pathway analysis modules. The page contains two tabs, showing the results of the compound and gene name mapping, respectively. Compounds or genes with no matches will be highlighted in red. Scroll down and press “Submit.”
4. The “Analysis Parameters” page allows users to specify parameters for enrichment analysis, topology analysis, underlying pathway databases, and integration method. There are three types of pathways—the gene-metabolite pathways, gene-centric pathways, and metabolite-centric pathways. Please note, many KEGG pathways contain only genes/proteins and are only available when gene-centric pathways are selected. By default genes and metabolites are merged into a single list which is used to perform enrichment analysis against pathways containing both genes and metabolites. Keep all default parameters and press “Submit.”
5. The result from Joint Pathway Analysis follows the same design as the Pathway Analysis result (Fig. 11). Briefly, the page is separated into two, with the top half containing the pathway visualization and the bottom half showing a detailed results table (refer to steps 5–6 in Section 2.4.2 for further details). The “Overview of Pathway Analysis” plot displays all matched pathways as circles, with the color and size of each circle corresponding to its *p*-value and pathway impact value,

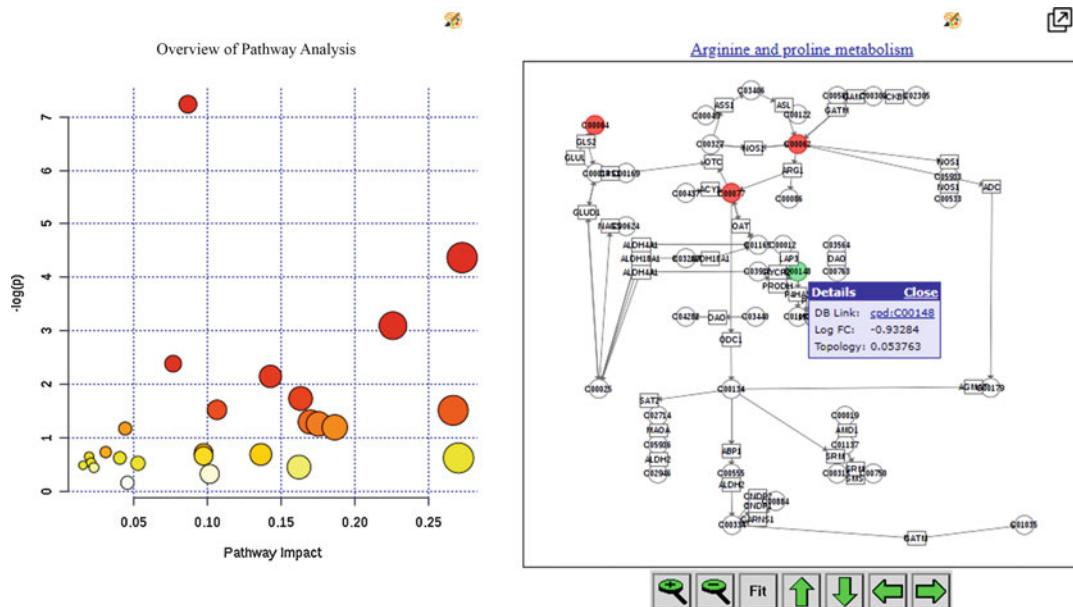


Fig. 11 A screenshot of the graphical overview of the Joint Pathway Analysis results in MetaboAnalyst, highlighting *Arginine and proline metabolism*

respectively. Click upon any circle for its corresponding pathway plot to appear on the right.

6. Scroll down to the results table. One of the top ranked pathways is *Arginine and proline metabolism*, with four matched metabolites/genes and a *p*-value of 0.0050855. Click upon the pathway name to visualize the pathway above. The matched features are highlighted red if upregulated and green if downregulated. Further, circles represent matched metabolites and squares represent genes. Double-click any highlighted node to get more details about the selected gene or metabolite. At this moment, it is advisable to also check the results from gene-centric pathways, as transcriptome measurement is much more comprehensive as compared to targeted metabolomics.
7. After further exploration, click the “Download” link on the navigation tree (left-side menu) to download all results. After downloading, click the “Logout” button to exit the session.

4.2 Network Exploration of Multi-Omics Data

In the previous section, multi-omics integration was limited to predefined pathways. Pathways represent high-quality, well-characterized knowledge that can provide explicit mechanistic understanding of the data. However, there are many known interactions and associations that are not captured in current pathway databases, which has significantly limited the utility of this approach. A natural extension is to use a network-based approach. Compared to pathways, networks are very flexible and inclusive—they can integrate

multiple types of information (including pathways) and may represent systems-level behavior. In the following section, we demonstrate how to use the Network Explorer module to map metabolites/genes onto known molecular interactions and molecule-phenotype association networks.

8. From the MetaboAnalyst home page, press “>>click here to start<<” to enter the “Module Overview” page. Next, select the “Network Explorer” button in the lower right corner of the circle to enter the Data Upload page.
9. For demonstration purposes, click the “try our example data” hyperlink on the top right of the page. A dialog will appear with two options of example datasets, providing details such as ID type, a brief description of the data, and instrumentation used. Select the Metabolite-genes data, which is the same intrahepatic cholangiocarcinoma (ICC) dataset used above in the Joint Pathway Analysis module. Press “Yes” to use the data and then “Submit” to continue.
10. The “Compound/Gene Mapping” page is identical to the one from the Joint Pathway Analysis. Refer to **step 3** in Section 4.1 above for further details. Scroll down and press the “Submit” button.
11. The next page shows the five knowledge-based networks: (1) the KEGG Global Metabolic Network, (2) metabolite-disease interaction networks, (3) gene-metabolite interaction networks, (4) metabolite-metabolite interaction networks, and (5) metabolite-gene-disease interaction networks. Details of how these networks were obtained can be found in the MetaboAnalyst 4.0 and FAQs from the website [2]. Click on the hyperlink for the “Metabolite-Gene-Disease Interaction Network” at the bottom of the page to use this network.
12. The “Mapping Overview” page should appear next. This page provides a detailed summary of the mapping of the uploaded data to the selected “Metabolite-Gene-Disease Interaction Network.” Click “Proceed” to continue to the network visualization.
13. The “Network Viewer” page displays the default layout of the uploaded data onto the “Metabolite-Gene-Disease Interaction Network.” This network viewer consists of four parts, the top toolbar for network customization, the left showing the Node Explorer table, the center with the interaction network, and the right with the Function Explorer menu.
14. In the interaction network, metabolites are represented as diamonds, genes as circles, and diseases as squares. The matched features are sized based on their position in the

interaction network, with important nodes (i.e., bigger) holding key positions in the network. The Node Explorer table provides two topological measures: node degree and betweenness centrality. Click the checkbox next to *L-Arginine* to zoom into this node. *L-Arginine* is linked to several blood-related disorders. Details for all selected nodes will appear in the “Current Selections” box in the bottom left corner of the page.

15. For further functional insights into the uploaded multi-omics data, pathway and enrichment analysis can be performed. To demonstrate this, in the Function Explorer panel (right-side), keep “Query” set to all nodes and select “Reactome” from the “Database” dropdown menu. Click “Submit” to perform pathway analysis. The top pathways are *Peptide ligand-binding receptors*, *Alternative complement activation*, and *Vasopressin-like receptors*. To view the features involved in these pathways, click the checkboxes next to the pathway names (Fig. 12).
16. MetaboAnalyst provides several features for users to customize the network and perform other analyses such as finding the shortest path between any two nodes in the network. Take time to explore these features.
17. Following network visualization, click on the “Download” hyperlink from the navigation tree (now at the top of the

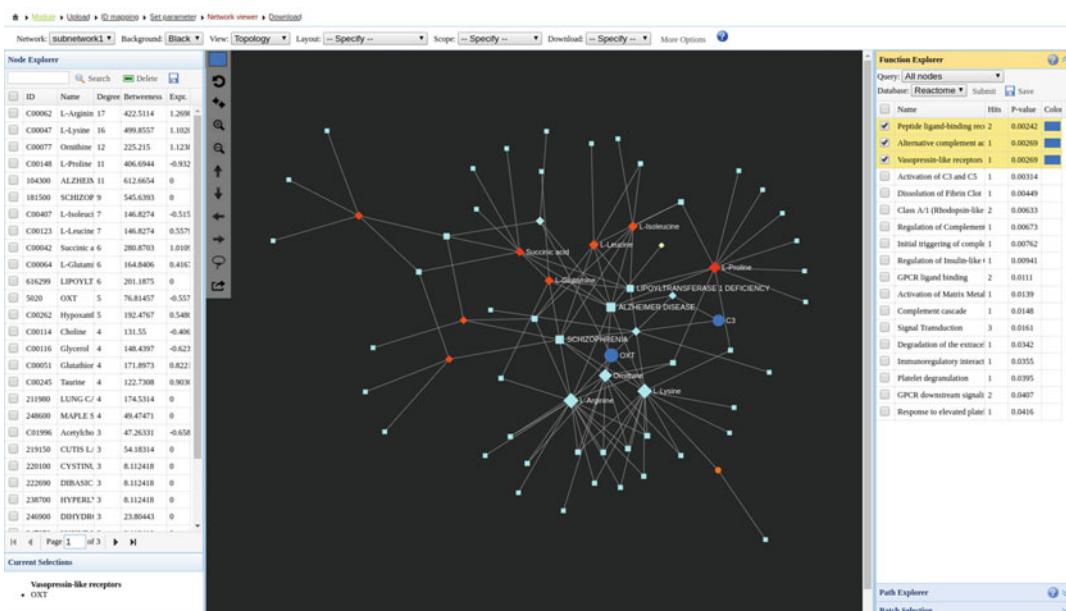


Fig. 12 A screenshot of the metabolite-gene-disease interaction network created using example data within the Network Explorer module of MetaboAnalyst

screen) to enter the “Download” page. Alternatively, click the home icon to exit the session.

5 Conclusion

MetaboAnalyst is one of the most widely used bioinformatics tools for metabolomics data analysis and interpretation. This chapter first introduced the general design concepts of MetaboAnalyst 4.0, and worked through several common workflows for targeted metabolomics, untargeted metabolomics, and multi-omics data integration. The underlying concepts and approaches of individual modules have been previously discussed in-depth [7, 28–33]. It should be noted that study design and raw data preprocessing are also essential components of a metabolomics data workflow and heavily influence the interpretation of results. We therefore encourage readers to familiarize themselves with proper experimental and preprocessing approaches to ensure high data quality and reduce unwanted variation [34–40]. For advanced users with basic knowledge in R, we recommend coupling the MetaboAnalystR package together with the web server to leverage the strengths of both approaches to maximize the potential of their data from raw spectra to biological insights.

Acknowledgement

This work has been supported in part by the US National Institutes of Health grant U01 CA235493, Natural Sciences and Engineering Research Council of Canada and Canada Research Chairs program.

References

1. Xia J, Psychogios N, Young N, Wishart DS (2009) MetaboAnalyst: a web server for metabolomic data analysis and interpretation. *Nucleic Acids Res* 37:W652–W660
2. Chong J, Soufan O, Li C, Caraus I, Li S, Bourque G, Wishart DS, Xia J (2018) MetaboAnalyst 4.0: towards more transparent and integrative metabolomics analysis. *Nucleic Acids Res* 46:W486–W494. <https://doi.org/10.1093/nar/gky310>
3. Xia J, Mandal R, Sinelnikov IV, Broadhurst D, Wishart DS (2012) MetaboAnalyst 2.0—a comprehensive server for metabolomic data analysis. *Nucleic Acids Res* 40:W127–W133
4. Xia J, Sinelnikov IV, Han B, Wishart DS (2015) MetaboAnalyst 3.0—making metabolomics more meaningful. *Nucleic Acids Res* 43: W251–W257
5. Chong J, Xia J (2018) MetaboAnalystR: an R package for flexible and reproducible analysis of metabolomics data. *Bioinformatics* 34:4313–4314. <https://doi.org/10.1093/bioinformatics/bty528>
6. Chong J, Yamamoto M, Xia J (2019) MetaboAnalystR 2.0: from raw spectra to biological insights. *Meta* 9:57
7. van den Berg RA, Hoefsloot HC, Westerhuis JA, Smilde AK, van der Werf MJC (2006) Scaling, and transformations: improving the biological information content of metabolomics data. *BMC Genomics* 7:142. <https://doi.org/10.1186/1471-2164-7-142>.

8. Temmerman L, Livera A, Bowne JB, Sheedy JR, Callahan DL, Nahid A, Souza D, Schoofs L, Tull DL, McConville M (2012) Cross-platform urine metabolomics of experimental hyperglycemia in type 2 diabetes. *Diabetes Metab S* 6:002
9. De Livera AM, Dias DA, De Souza D, Rupasinghe T, Pyke J, Tull D, Roessner U, McConville M, Speed TP (2012) Normalizing and integrating metabolomics data. *Anal Chem* 84:10768–10776. <https://doi.org/10.1021/ac302748b>
10. Eisner R, Stretch C, Eastman T, Xia J, Hau D, Damaraju S, Greiner R, Wishart DS, Baracos VE (2011) Learning to predict cancer-associated skeletal muscle wasting from ¹H-NMR profiles of urinary metabolites. *Metabolomics* 7:25–34
11. Dieterle F, Ross A, Schlötterbeck G, Senn H (2006) Probabilistic quotient normalization as robust method to account for dilution of complex biological mixtures. Application in ¹H NMR metabolomics. *Anal Chem* 78:4281–4290
12. Kanehisa M, Goto S, Sato Y, Furumichi M, Tanabe M (2012) KEGG for integration and interpretation of large-scale molecular data sets. *Nucleic Acids Res* 40:D109–D114. <https://doi.org/10.1093/nar/gkr988>
13. Puchalska P, Crawford PA (2017) Multi-dimensional roles of ketone bodies in fuel metabolism, signaling, and therapeutics. *Cell Metab* 25:262–284. <https://doi.org/10.1016/j.cmet.2016.12.022>
14. Jewison T, Su Y, Disfany FM, Liang Y, Knox C, Maciejewski A, Poelzer J, Huynh J, Zhou Y, Arndt D et al (2014) SMPDB 2.0: big improvements to the small molecule pathway database. *Nucleic Acids Res* 42:D478–D484. <https://doi.org/10.1093/nar/gkt1067>
15. Flint TR, Janowitz T, Connell CM, Roberts EW, Denton AE, Coll AP, Jodrell DI, Fearon DT (2016) Tumor-induced IL-6 reprograms host metabolism to suppress anti-tumor immunity. *Cell Metab* 24:672–684. <https://doi.org/10.1016/j.cmet.2016.10.010>
16. Ham DJ et al (2014) Glycine administration attenuates skeletal muscle wasting in a mouse model of cancer cachexia. *Clin Nutr* 33 (3):448–458
17. Cui P et al (2019) Metabolic derangements of skeletal muscle from a murine model of glioma cachexia. *Skelet Muscle* 9(1):3
18. Gowda GN, Djukovic D (2014) Overview of mass spectrometry-based metabolomics: opportunities and challenges. In: Mass spectrometry in metabolomics. Springer, New York, pp 3–12
19. Theodoridis G, Gika HG, Wilson ID (2008) LC-MS-based methodology for global metabolite profiling in metabolomics/metabolomics. *Trac Trend Anal Chem* 27:251–260. <https://doi.org/10.1016/j.trac.2008.01.008>
20. Li S, Park Y, Duraisingham S, Strobel FH, Khan N, Soltow QA, Jones DP, Pulendran B (2013) Predicting network activity from high throughput metabolomics. *PLoS Comput Biol* 9:e1003123
21. Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA, Paulovich A, Pomeroy SL, Golub TR, Lander ES et al (2005) Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc Natl Acad Sci U S A* 102:15545–15550. <https://doi.org/10.1073/pnas.0506580102>
22. Integrative HMPRNC (2014) The integrative human microbiome project: dynamic analysis of microbiome-host omics profiles during periods of human health and disease. *Cell Host Microbe* 16:276–289. <https://doi.org/10.1016/j.chom.2014.08.014>
23. Li S, Pozhitkov A, Ryan RA, Manning CS, Brown-Peterson N, Brouwer M (2010) Constructing a fish metabolic network model. *Genome Biol* 11:R115. <https://doi.org/10.1186/gb-2010-11-11-r115>
24. Tiraterra E, Franco P, Porru E, Katsanos KH, Christodoulou DK, Roda G (2018) Role of bile acids in inflammatory bowel disease. *Ann Gastroenterol* 31:266–272. <https://doi.org/10.20524/aog.2018.0239>
25. Ogilvie LA, Jones BV (2012) Dysbiosis modulates capacity for bile acid modification in the gut microbiomes of patients with inflammatory bowel disease: a mechanism and marker of disease? *Gut* 61:1642–1643. <https://doi.org/10.1136/gutjnl-2012-302137>.
26. Duboc H, Rajca S, Rainteau D, Benarous D, Maubert MA, Quervain E, Thomas G, Barbu V, Humbert L, Desprès G et al (2013) Connecting dysbiosis, bile-acid dysmetabolism and gut inflammation in inflammatory bowel diseases. *Gut* 62:531–539. <https://doi.org/10.1136/gutjnl-2012-302578>
27. Murakami Y, Kubo S, Tamori A, Itami S, Kawamura E, Iwaisako K, Ikeda K, Kawada N, Ochiya T, Taguchi Y (2015) Comprehensive analysis of transcriptome and metabolome analysis in intrahepatic Cholangiocarcinoma and hepatocellular carcinoma. *Sci Rep* 5:16294
28. Worley B, Powers R (2013) Multivariate analysis in metabolomics. *Curr Metabolomics*

- 1:92–107. <https://doi.org/10.2174/2213235X11301010092>.
29. Vinaixa M, Samino S, Saez I, Duran J, Guinovart JJ, Yanes O (2012) A guideline to univariate statistical analysis for LC/MS-based untargeted metabolomics-derived data. *Metabolomics* 2:775–795. <https://doi.org/10.3390/metabo2040775>
30. Rubingh CM, Bijlsma S, Derkx EP, Bobeldijk I, Verheij ER, Kochhar S, Smilde AK (2006) Assessing the performance of statistical validation tools for megavariate metabolomics data. *Metabolomics* 2:53–61. <https://doi.org/10.1007/s11306-006-0022-6>
31. Bartel J, Krumsiek J, Theis FJ (2013) Statistical methods for the analysis of high-throughput metabolomics data. *Comput Struct Biotechnol J* 4:e201301009. <https://doi.org/10.5936/csbj.201301009>
32. Marco-Ramell A, Palau-Rodriguez M, Alay A, Tulipani S, Urpi-Sarda M, Sanchez-Pla A, Andres-Lacueva C (2018) Evaluation and comparison of bioinformatic tools for the enrichment analysis of metabolomics data. *BMC Bioinformatics* 19(1). <https://doi.org/10.1186/s12859-017-2006-0>
33. Hackstadt AJ, Hess AM (2009) Filtering for increased power for microarray data analysis. *BMC Bioinformatics* 10:11. <https://doi.org/10.1186/1471-2105-10-11>
34. Hendriks MMWB, van Eeuwijk FA, Jellema RH, Westerhuis JA, Reijmers TH, Hoefsloot HCJ, Smilde AK (2011) Data-processing strategies for metabolomics studies. *Trac Trend Anal Chem* 30:1685–1698. <https://doi.org/10.1016/j.trac.2011.04.019>
35. Tautenhahn R, Patti GJ, Rinehart D, Siuzdak G (2012) XCMS online: a web-based platform to process untargeted metabolomic data. *Anal Chem* 84:5035–5039. <https://doi.org/10.1021/ac300698c>
36. Myers OD, Sumner SJ, Li S, Barnes S, Du X (2017) Detailed investigation and comparison of the XCMS and MZmine 2 chromatogram construction and chromatographic peak detection methods for preprocessing mass spectrometry metabolomics data. *Anal Chem* 89:8689–8695. <https://doi.org/10.1021/acs.analchem.7b01069>
37. Dudzik D, Barbas-Bernardos C, Garcia A, Barbas C (2018) Quality assurance procedures for mass spectrometry untargeted metabolomics. A review. *J Pharm Biomed Anal* 147:149–173. <https://doi.org/10.1016/j.jpba.2017.07.044>
38. Lange E, Tautenhahn R, Neumann S, Gröpl C (2008) Critical assessment of alignment procedures for LC-MS proteomics and metabolomics measurements. *BMC Bioinformatics* 9:375
39. Tautenhahn R, Bottcher C, Neumann S (2008) Highly sensitive feature detection for high resolution LC/MS. *BMC Bioinformatics* 9:504. <https://doi.org/10.1186/1471-2105-9-504>
40. Zhou B, Xiao JF, Tuli L, Ressom HW (2012) LC-MS-based metabolomics. *Mol BioSyst* 8:470–481



Chapter 18

Using Genome-Scale Metabolic Networks for Analysis, Visualization, and Integration of Targeted Metabolomics Data

Jake P. N. Hattwell, Janna Hastings, Olivia Casanueva,
Horst Joachim Schirra, and Michael Witting

Abstract

Interpretation of metabolomics data in the context of biological pathways is important to gain knowledge about underlying metabolic processes. In this chapter we present methods to analyze genome-scale models (GSMs) and metabolomics data together. This includes reading and mining of GSMs using the SBTAB format to retrieve information on genes, reactions, and metabolites. Furthermore, the chapter showcases the generation of metabolic pathway maps using the Escher tool, which can be used for data visualization. Lastly, approaches to constrain flux balance analysis (FBA) by metabolomics data are presented.

Key words Metabolic networks, Metabolomics, Analysis, Visualization, Integration

1 Introduction

1.1 Genome-Scale Models

Similar to how an insect struggling in a spider's web sends vibrations across the threads, any originally localized disturbance can result in effects "reverberating" in a complex pattern across the entire metabolic network. To improve understanding of the complex metabolic networks, present in nature, genome-scale modeling has emerged in recent years as a means of representing a biological system in a format that is informative and usable for experiments. A genome-scale model (GSM, or GSML for genome-scale metabolic network) is a digital reconstruction that represents the global metabolic processes that occur within an organism [1, 2]. Such a model should be able to be constrained and optimized to facilitate both qualitative and quantitative investigation of the biological system of interest, usually taking the form of *in silico* simulations. GSMS help to expedite research by providing a comprehensive knowledge base of the organism's metabolism as well as reducing time and financial costs.

Genome-scale models aim to amass as much information about a biochemical system as possible. The core of a GSM is built using information about genes, the reactions that the products of those genes catalyze, and the metabolites and chemical compounds inside the system [1, 3, 4]. Currently, genes map directly to reactions using Boolean operations, creating gene associations that control whether reactions can be active. Reactions consume and produce metabolites, forming a network of metabolites linked by reactions.

The information in GSMS must be carefully cultivated through both automated and manual curation [5]. Refinement through correction of errors and addition of new information ensures that the model can be used as a reliable source of information, both as a reference tool and for contextualizing data. This results in a resource that allows for rapid identification of active pathways and trends in the biological network, which can provide context to -omics data.

As advancements in -omics technologies continue to reveal the complexity of biological systems, different ways to analyze metabolism have evolved. Genetic manipulation of metabolic genes allows for the study of their influence on different phenotypic traits, while metabolomics allows for the characterization of the metabolic status of a given organism or biological system. Both techniques can be performed with decent throughput, but each only captures a small part of what is a genome-wide multi-omic system. Genome-scale modeling and an accompanying technique, flux balance analysis (FBA), allows for the computational prediction of intracellular turnover rates (or fluxes) for all metabolic reactions in a cell or in an organism. This integrated method overcomes several of the limitations of traditional techniques.

These in silico tools have become very powerful, and a large number of GSMS for different organisms have been reconstructed from genomic information and published [6–9]. While the first GSM of the gram-negative bacteria *Haemophilus influenzae* was rather small [10], a trend toward more complex, multicellular organisms and full-body metabolic reconstructions is underway as our technology and methods advance [11, 12].

1.2 *Caenorhabditis elegans* GSM

The small nematode *Caenorhabditis elegans* was the first multicellular organism with a completely sequenced genome [13]. Several GSMS have been published for *C. elegans* [14–17]. Recently, these different models have been merged into a consensus GSMN christened WormJam (from “worm jamboree,” i.e., curation meetings of the WormJam community) [18, 19]. This model currently represents one of the best curated models for *C. elegans*, but work is still constantly being undertaken to improve it further. The WormJam model will be used as the example model throughout this chapter.

1.3 Metabolomics and GSMS

While GSMS represent large knowledge bases of metabolism in a given organism, the discrepancies between these models and metabolomic measurements can be significant [20]. One of the key challenges that faces the use of GSMS with metabolomic data is accurately mapping the identities of metabolites with the model with those experimentally measured. For example, metabolites found in a GSM may fall below the limit of detection of current metabolomics techniques or it may be present in very specific stages or conditions. Conversely, due to a large gap in the knowledge of metabolites, many of them might not yet be annotated and incorporated into the GSM. Despite these limitations, metabolomics data can be successfully combined with FBA for several applications.

In addition to visualizing metabolomic data on pathway diagrams, it is also possible to improve the *in silico* modeling potential of genome-scale models through the integration of metabolomics and transcriptomics data. Flux balance analysis (FBA) is a constraint-based technique that can be used to predict fluxes of various conditions, which can be of a genetic or environmental nature. Attractively, FBA does not require kinetic information about reactions, but instead uses a combination of elementary pathway analysis and linear programming techniques to optimize a model. This optimization is commonly for the production of biomass, but can be changed to other biological processes of interest.

When unconstrained, the number of possible flux distributions is essentially infinite. For this reason, experimental -omics data are used to reduce the system to an allowable solution space, which is then optimized, or to clarify the objective for optimization.

A variety of -omics data can be integrated with GSMS in a number of different ways to obtain biologically relevant FBA solutions, as illustrated in Fig. 1. But as mentioned in the introduction, only those experimentally measured genes and metabolites can be used to constrain an FBA that are present in the GSM. In this chapter we present different protocols demonstrating how GSM and metabolomics data can work together. This includes the use of GSMS as a metabolome database for annotation of metabolomics data and the use of targeted metabolomics data to combine with flux balance analysis. In particular, we explore two methods for the integration of metabolomic data into *in silico* experiments. The first (*MetabFBA*) uses endometabolomic data to enhance the objective function of the model [20], and the second (*MetaboTools*) is the use of exometabolomic data to constrain an FBA via exchange fluxes [21, 22].

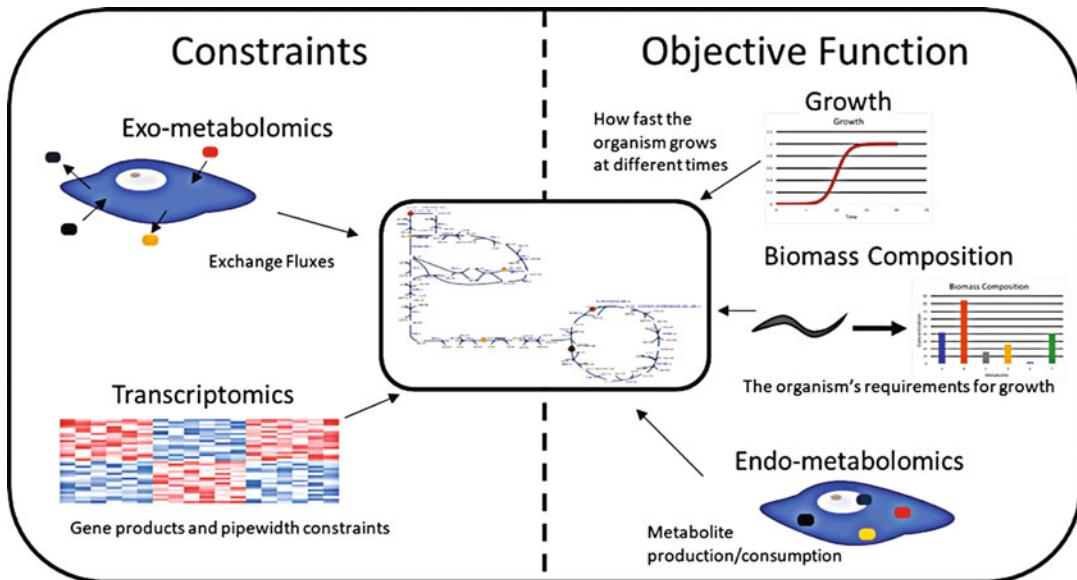


Fig. 1 Integration of -omics data with flux balance analysis. This illustration shows different ways that -omics data can be incorporated together with a GSM for FBA. The methods are sorted with respect to the different omics data sources and whether the data are used to apply constraints to the model (left) or to inform the objective function that will be optimized (right)

2 Materials

The protocols in this chapter were tested on a computer with Windows 10 installed. Since the R and Python programs are compatible with all major operating systems, the examples here are expected to work on other computer systems. The specific software components used here are the following:

1. **R 3.5.2**
 - The *tidyverse* package (1.2.1).
 - *Rstudio 1.1.463*.
2. **Python 3.6+**
 - Python is freely downloadable from <https://www.python.org/downloads/>. The methods within this chapter have been tested to work in both Python 3.6 and Python 3.7.
 - *COBRApy* and *Escher* packages for Python (Installation instructions detailed in Subheading 3.2.1).
3. **Demo Scripts:** Demonstration scripts and test data are available for download at the following git repository: https://github.com/michaelwitting/MiMB_CompMetabo.

3 Methods

These protocols assume basic knowledge and experience of both R and Python. All R scripts use the `tidyverse` package and especially the “pipe” `%>%` operator to avoid nested function calls. If you are not familiar with the `tidyverse` we suggest reading the chapters in this online web tutorial <https://r4ds.had.co.nz/>.

In the first example, the statistical programming language R (*see Note 1*) is used for inspection of the SBTAB files and preparation of metabolomics data. The `tidyverse` package is used to allow for easy access to the data as well as to facilitate data filtering and manipulation. However, similar scripts, data inspection, and manipulation can be performed in other languages such as Python. The second part utilizes Python and the packages `cobra` and `escher` to generate Escher maps which can be used for metabolomics data visualization. Python is then used to demonstrate a basic flux balance analysis.

3.1 Reading and Searching for Information in SBTAB Files

Metabolic models are typically stored in the SBML format, an XML derivative customized for the use in systems biology. However, this format is hardly human readable and contains a lot of information to aid the computer in processing the file. In contrast to SBML files, the Systems Biology Table (SBTab) format allows humans directly to work with the model. Both formats are interconvertible and free web tools for conversion are available (<https://www.sbtab.net/sbtab/default/converter.html>). The SBTAB format comprises a series of separate files that individually contain the reactions, metabolites, genes, and other information present in the GSM. Because the data in the individual data files are related to each other, the same identifiers are used across all data. This makes it possible to cross-reference and search for information (e.g., isolate metabolites from a single reaction and retrieve information on them from the compound table).

We use an example model, which is a truncated version of the WormJam model containing only information related to the tricarboxylic acid (TCA) cycle, pentose phosphate pathway (PPP), and glycolysis pathway from the WormJam model. This minimal example contains the following SBTAB files:

- compartments-SBTAB.tsv contains the information on all the model involved compartments,
- compounds-SBTAB.tsv contains the chemical information of all metabolites,
- genes-SBTAB.tsv contains all related metabolic genes,
- pathways-SBTAB.tsv contains the information on pathways and mapping to external pathway databases (e.g., KEGG),

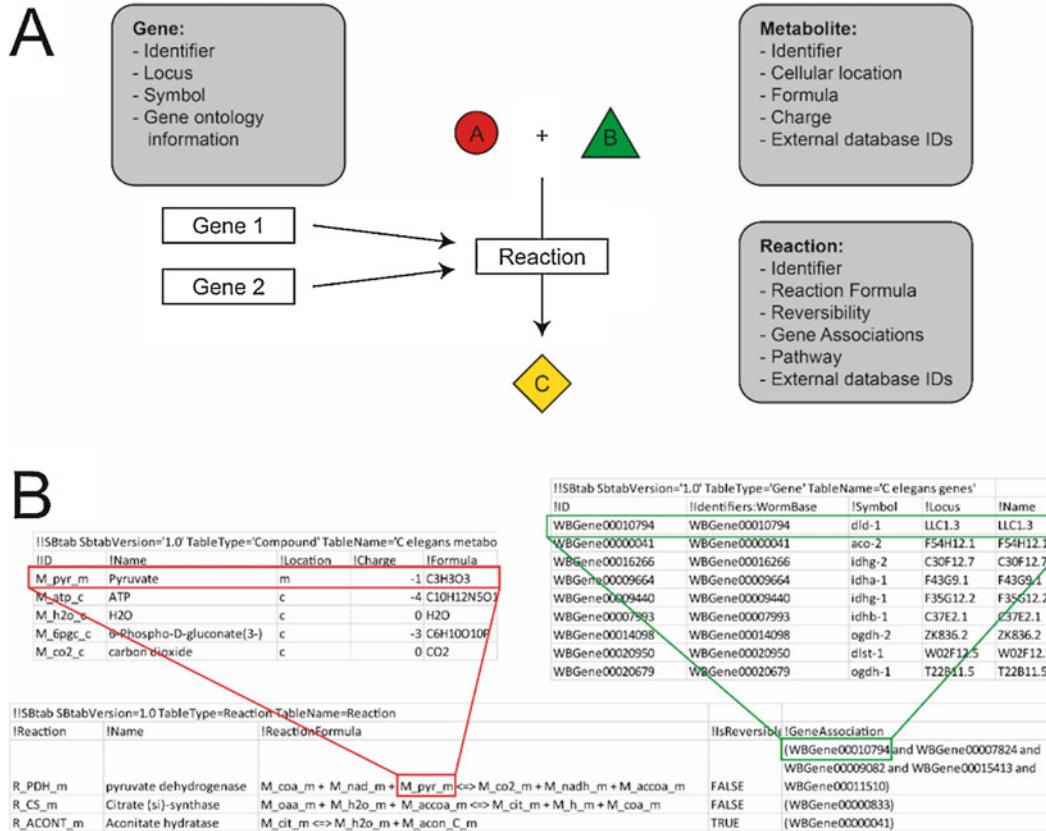


Fig. 2 (a) The three most common classes of entities in genome-scale models, and some of the information they contain. In this representation, Genes 1 and 2 are metabolic genes required for the catalysis of the reaction, which converts A and B into C. **(b)** Example tables from SBTAB format

- reactions-SBTAB.tsv contains all reactions related to the three pathways mentioned above.

The model contains in total 29 reactions, 49 genes, and 51 compounds. These reactions, genes and compounds were derived from other previously published models and have been additionally manually checked and curated. A specialty of the WormJam model is the detailed curation of metabolite structures. GSMS use metabolite structures with charges appropriate for the cytosolic pH of 7.3 and mass balanced reactions, that is, reactions in which atoms are neither created nor destroyed. However, in metabolomics experiments typically neutral structures are reported. In order to overcome problems in mapping neutral metabolite names from metabolomics studies to charged structures, the compound table contains both the charged and neutral form of the metabolites (Fig. 2).

3.1.1 Reading of SBTAB Files

1. All SBTAB files have to be stored in the same folder. All files are read at the same time by the function below, which reads all files having a file extension of “-SBTab.tsv” and creates a data frame (called a **tibble**) for each file.

```
# load required libraries -----
-----
library(tidyverse)

# helper function -----
-----
read_sbtab <- function(folderPath) {

  # get all files
  sbtab_files <- list.files(folderPath,
  pattern = "-SBTab.tsv$",
  full.names = TRUE)

  # make new list
  sbtab_list <- list()

  # iterate over all files
  for(i in 1:length(sbtab_files)) {

    #get current file and add to list
    sbtab_file <- sbtab_files[i]
    sbtab_list[[i]] <- read_tsv(sbtab_file,
    comment = "!!")

  }

  # correct names
  sbtab_names <- str_replace_all(basename(sbtab_files),
  "-SBTab.tsv", "")
  names(sbtab_list) <- paste0(sbtab_names,

  # make tibble for each table
  for(i in 1:length(sbtab_list)) {

    assign(names(sbtab_list)[i],
    sbtab_list[[i]],
    envir = parent.frame())

  }

}
```

As this function is highly used, we store it in a separate .R file which is sourced at the beginning of each script. This custom function is stored in an .R file named “loadSBTab.R”, which can be afterward sourced from any other .R file to have access to the reading function.

```
# load required libraries -----
-----
library(tidyverse)

# source
source("R/loadSBTab.R")

# load tables -----
-----
read_sbtab("SBTab")
```

2. Reactions can be printed by typing `reactions_table` at the command prompt, which prints the first ten reactions in the output console. In order to more closely inspect individual reactions the `$` operator can be used to access the reaction formula (`$'!ReactionFormula'`). This command prints all 29 reactions. Let us now examine closer the reaction catalyzed by pyruvate synthase.

```
> reactions_table$'!ReactionFormula'[20]
[1] "M_coa_m + M_nad_m + M_pyr_m <=> M_co2_m + M_nadh_m +
M_accoa_m"
```

The WormJam model uses BiGG identifiers for the metabolites. The leading `M` denotes the entity as a metabolite. The ID ends with a `c`, `m`, `n`, or `e` coding for cytosolic, mitochondrial, nuclear, and extracellular location of the metabolite, respectively. The files in this chapter only contain entities from the cytosol and mitochondria. The numbers and letters in between denote the metabolite itself. `M_pyr_c` and `M_pyr_m` are both identifiers for pyruvate, but with different subcellular locations. Based on the metabolite identifiers it can be seen that the reaction occurs in the mitochondrion.

3. Since the BiGG metabolite identifier follows certain rules, it can be identified by regular expressions. Metabolite identifiers follow the following regular expression `"M_\\w+(c|m|n|e)"`. Using the `str_extract_all()` function it is possible to isolate all metabolites from the reactions stored in the

reaction table. The following section of code isolates all unique metabolites IDs and counts their occurrence in the reactions.

```
# isolate all compounds and count occurrence -----
-----
metabolite_counts <- reactions_table$'!ReactionFormula' %>%
  map(.f=~/str_extract_all(.x, "M_\\w+_(c|m|e|n)")) %>%
  unlist() %>%
  tibble('!ID' = .) %>%
  add_count('!ID') %>%
  distinct('!ID', .keep_all = TRUE) %>%
  arrange(desc(n))
```

The result is a data frame with the BiGG identifiers followed by the count. The highest count was found for the metabolite M_h_c which represents cytosolic protons. These protons are often required for mass balancing of reactions and not informative. The data frame can be filtered to remove uninformative metabolites or hub metabolites (e.g., H₂O).

```
# filter to remove hub metabolites -----
-----
hub_metabolites <- c("M_h_c", "M_h_m", "M_h2o_c", "M_h2o_m")

metabolite_counts %>%
  filter(!'!ID' %in% hub_metabolites)
```

After this filtering, ADP remains as the metabolite with the highest occurrence.

4. In the next step the genes related to the pyruvate synthase are extracted. Gene assignments are stored in the column '\$'GeneAssociation' and WormBase identifiers "WBGene\d+".

```
> reactions_table$'!GeneAssociation'[20]
[1] "(WBGene00010794 and WBGene00007824 and WBGene00009082 and
WBGene00015413 and WBGene00011510)"
```

Five different genes are associated with this reaction. They are connected with an AND operator which indicates that all of them are required. If a list of genes is required, they can be isolated from this field with the `str_extract_all()` function similar to the metabolites. The regular expression for gene identifiers is "WBGene\\d+"

```
reactions_table$`!GeneAssociation'[20] %>%
  map(.f=~str_extract_all(.x, "WBGene\\d+")) %>%
  unlist()
```

This returns all the gene identifiers as vector. The enzymes required for this reaction form a large complex and genes returned encode different subparts of this complex.

5. Lastly, we can Isolate all reactions that contain a specific metabolite either as reactant or product using the `str_detect()` function. The code below isolates all reactions that contain pyruvate (independent of subcellular location).

```
# get all reactions with a pyruvate involved -----
-----
filtered_reactions_table <- reactions_table %>%
  filter(str_detect(.$`!ReactionFormula`, "M_pyr"))

filtered_reactions_table
```

3.1.2 Interaction and ID Mapping Between Tables

1. So far only entries from the reaction table have been isolated, without additionally retrieving data from the compound or gene table. In order to obtain this linked information (e.g., gene symbols for each gene related to pyruvate synthase), we can isolate all WormBase identifiers and query the gene table. First the gene identifiers are isolated and stored in a vector.

```
# get genes involved in pyruvate synthase reaction -----
-----
gene_list <- reactions_table$`!GeneAssociation'[20] %>%
  map(.f=~str_extract_all(.x, "WBGene\\d+")) %>%
  unlist()
```

This vector can be then used to filter the gene table using the `filter` function.

```
# filter gene table according to entries in gene_list -----
-----
genes_table %>%
  filter(.`!ID` %in% gene_list)
```

The two commands can be combined into a single command, where first a new data frame containing the WormBase IDs is generated and is then joined with the information from the gene table.

```
# single command to obtain a filter gene table -----
-----
reactions_table$`!GeneAssociation'[20] %>%
  map(.f=~str_extract_all(.x, "WBGene\\d+")) %>%
  unlist() %>%
  tibble(`!ID` = .) %>%
  left_join(., genes_table, by = c(`!ID`))
```

Since Wormbase identifiers act as machine-readable unique resource identifiers, and are very rarely used in normal work, this command returns the human-readable gene symbols, in our case *dld-1*, *dlat-1*, *dlat-2*, *pdhb-1*, and *pdhb-2*.

- Likewise chemical information for a metabolite can be retrieved. In contrast to genes, the function `str_detect()` is used, which utilizes regular expression. The search string “`M_pyr`” is used, which returns the entries for the cytosolic and mitochondrial pyruvate.

```
# get chemical information for pyruvate -----
-----
compounds_table %>% filter(str_detect(`!ID`, "M_pyr"))
```

The data table for the metabolites contains all information, such as sum formula, chemical structure, and identifiers for the charged and neutral structures. A specific metabolite with its subcellular localization can be retrieved using the following.

```
# get chemical information for pyruvate in the cytosol -----
-----
compounds_table %>% filter(`!ID` == "M_pyr_c")
```

- Multiple entries can be retrieved using a vector with the IDs of the metabolites. The resulting code lines are analogous to the ones used for genes in **step 1**.

```
# get all metabolites related to pyruvate synthase reaction -----
metabolite_list <- reactions_table$`!ReactionFormula'[20] %>%
  map(.f=~str_extract_all(.x, "M_\\w+_(c|m|e|n)")) %>%
  unlist()

# filter gene table according to entries in metabolite list -----
compounds_table %>% filter(`!ID` %in% metabolite_list)
```

```
# single command to obtain a filtered compound table -----
-----
reactions_table$`!ReactionFormula'[20] %>%
  map(.f=~str_extract_all(.x, "M_\\w+_(c|m|e|n)")) %>%
  unlist() %>%
  tibble(`!ID` = .) %>%
  left_join(., compounds_table, by = c(`!ID`))
```

4. Data in the compound table contain both the charged and neutral versions of metabolites. The neutral version can be used to calculate *m/z* values for metabolite annotation. All columns that contain information on the neutral metabolite have a “neutral” in the column name and can be selected in the code using this. Based on the last script, we will now only select columns containing the neutral part for all metabolites related to pyruvate synthase reaction.

```
# single command to obtain a filtered compound table and select
only neutral
reactions_table$`!ReactionFormula'[20] %>%
  map(.f=~str_extract_all(.x, "M_\\w+_(c|m|e|n)")) %>%
  unlist() %>%
  tibble(`!ID` = .) %>%
  left_join(., compounds_table, by = c(`!ID`)) %>%
  select(contains("neutral"))
```

3.1.3 Generation of a Suspect List for Pathway Screening

1. The complete information for all metabolites in the reactions can be retrieved from the previous script (*see Note 2*). Based on these data, a suspect list for inspection or annotation of metabolomics data can be generated. First all metabolites are isolated from the reactions and their chemical data is retrieved. The list of metabolites might contain certain molecules that cannot be measured by the relevant analytical technique (e.g., protons or water in the case of mass spectrometry). These compounds and others that represent only generic compounds with no explicit chemical structure are removed from the list. Afterward, only unique entries are selected using the `distinct()` function.

```
# isolate all metabolites from all reactions and leave
unique -----
neutral_metabolites <- reactions_table$`!ReactionFormula` %>%
  map(.f=~str_extract_all(.x, "M_\\w+_(c|m|e|n)")) %>%
  unlist() %>%
```

```
tibble(`!ID` = .) %>%
left_join(., compounds_table, by = c("!ID")) %>%
filter(!.$`!ID` %in% metabolite_exclusion) %>%
select(contains("neutral")) %>%
distinct()
```

2. One package that allows for the calculation of m/z values from neutral masses or formulas is *masstrixR* (Witting and Schmitt-Kopplin, under review). Data has to be supplied in a specific format. Therefore a new data frame called `suspect_List` is created.

```
# create new tibble that can be used with masstrixR -----
-----
suspect_List <- tibble(
  id = neutral_metabolites$`!Notes:ChEBI_neutral`,
  smiles = NA,
  inchi = neutral_metabolites$`!Notes:InChi_neutral`,
  inchiky = neutral_metabolites$`!Notes:InChIKey_neutral`,
  formula = neutral_metabolites$`!Notes:FORMULA_Neutral`,
  name = neutral_metabolites$`!Notes:ChEBI_Name_neutral`,
  exactmass = NA
)
```

3. The *masstrixR* package can be obtained from GitHub (*see Note 3*). After loading the package, the adducts that shall be covered have to be defined. Metabolites from glycolysis, PPP, and TCA cycle are typically detected in negative ionization mode as [M-H]⁻ adduct.

```
# load masstrixR library -----
-----
library(masstrixR)

# create compound list with adduct masses
-----
neg_adducts <- c(" [M-H] -")
neg_suspect_List <- as_tibble(prepareCompoundList(suspect_List,
adductList = neg_adducts))
```

This compound list can be used to create EIC traces (e.g., in XCMS, or to annotate nontargeted metabolomics data).

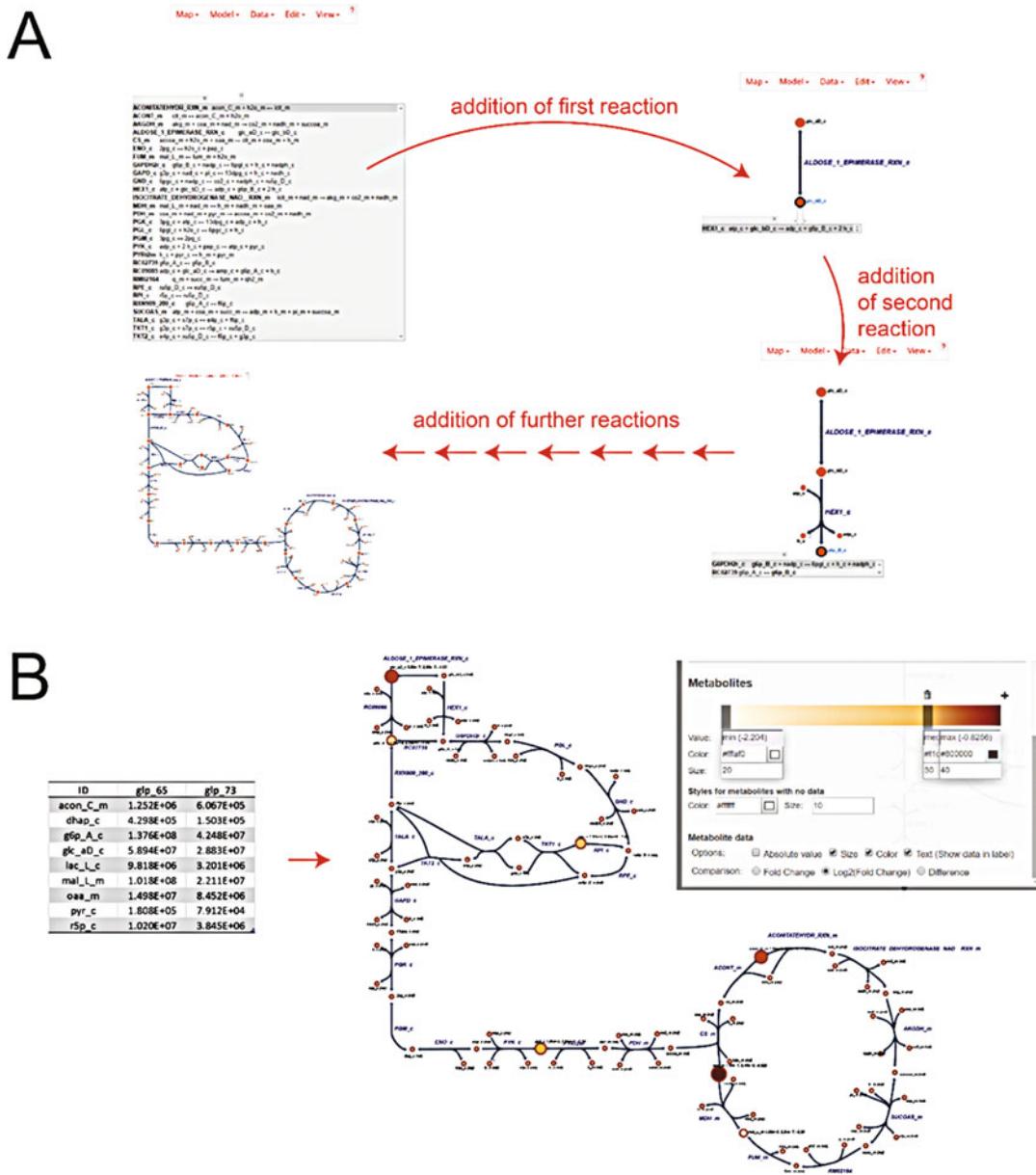


Fig. 3 (a) The Escher workflow. Starting with a blank window, reactions are added to the map in a chainlike fashion, creating a pathway. (b) Overlaying an example table of metabolites onto the generated pathway map. The Escher Builder window allows for customization of the overlay, through the settings pane

3.2 Generation of Escher Pathway Map for Visualization of Metabolic Fluxes

Escher is a versatile tool for the generation of biological pathway maps that can be used to visualize metabolic systems and give context to omics data [23]. Several template *Escher* maps are available for use on the *Escher* website; however, for organism-specific pathways or figure generation, a custom map may be desired. This section details the generation of an *Escher* pathway map from scratch (Fig. 3).

To create a map, a Constraint Based Reconstruction and Analysis (COBRA) formatted model is needed. Instructions for the generation of a COBRA model can be found in a *Nature Protocols* paper [2].

3.2.1 Installation of COBRApy and Escher

1. To install the *COBRApy* and *Escher* modules, after installing Python 3.6+, open a command prompt window, and enter the following commands:

```
python -m pip install --user cobra
python -m pip install --user escher
```

This will install the *COBRApy* and *Escher* packages onto your account on the computer. To install the packages for all users, simply omit the “*--user*” parameter; however, administrator privileges are required.

3.2.2 Generation of Escher Pathway Map

1. A map can be generated using the SBML file of the model, Python 3, and the *COBRApy* and *Escher* packages. In Python, enter the following commands.

When a file name is presented (nonitalicized), replace this with the path to the corresponding file from the Demo Scripts in Subheading 2.

Lines beginning with a # represent comments to explain the code and do not need to be entered.

```
#import dependancies
import cobra
from escher import Builder
#read the SBML model, create an Escher Builder object, and open
the Escher server
model = cobra.io.read_sbml_model("WormJam_Toy_Network.xml")
escher_model = Builder(model=model)
escher_model.display_in_browser()
```

This will start up a local server on your computer and open up the *Escher* Builder in a browser window.

2. To begin adding reactions, select “Edit” → “Add reaction mode” from the top menu, and click anywhere to add your first reaction. Type in the Reaction identifier from your model, and *Escher* will automatically place it on the canvas. You can then click on any of the metabolites that participate in the reaction to add another reaction. The selection tool and rotation tool can be used to arrange items on the canvas. More information about high level use of *Escher* can be found in the

publication [23] and the documentation site (<https://escher.readthedocs.io>). A sample *Escher* diagram of the WormJam model's glycolysis, PPP, and TCA cycle can be found with the accompanying files.

3. It is important to save your work using “Map” → “Save map JSON”. Whilst *Escher* is hosted locally, it does not save the work you have done, so this must be done manually. To load a *Escher* map you have worked on previously, use “Map” → “Load map JSON”. SVG and PNG formatted images of your pathway maps can also be exported using the “Map” menu. The sample map can be loaded in this way.

3.3 Overlay of Metabolomics Data on Escher Pathway Maps

3.3.1 Preparation of Metabolomics Data

Data obtained from targeted metabolomics can be visualized on *Escher* maps (see Subheading 3.3.2). Here we present how such dataset can be prepared for the visualization using functions from the R *tidyverse* package. As an example, data from Hastings et al. is used [20]. Different strains of *C. elegans* have been followed over some time and examined using HILIC-MS/MS.

1. First the metabolomics data is loaded from .tsv file. Additionally, a file containing mapping information between the metabolite names from the result file and the model IDs is read.

```
# read files -----
-----
metabolomics_data <- read_tsv("MetabolomicsData/metabolomics_table.tsv")
model_mapping <- read_tsv("MetabolomicsData/metabo-model-mapping.tsv")
```

2. Next, metabolites that belong to the glycolysis, TCA cycle, or pentose phosphate pathway are filtered and retained based using the filter() function. This reduces the number of metabolites to 18.

```
# filter metabolites from glycolysis, TCA and PPP -----
-----
metabolomics_data_filtered <- metabolomics_data %>%
  filter(str_detect('METABOLIC PATHWAY',
    "Glycolysis|TCA|Pentose phosphate pathway"))
```

3. In order to calculate the group means and standard deviation we convert the data to a long format using the gather() function.

```
# generate long format from data table -----
-----
metabolomics_long <- gather(metabolomics_data_filtered, key,
value,
-`Compound (Metabolite)`,
-`METABOLIC PATHWAY`,
convert = TRUE, na.rm = TRUE)
```

4. The grouping of the different samples is hard-coded in the sample name. The column with the sample name can be split into separate columns using the `separate()` function. This will generate additional columns, which are named time, strain, Day, and sampleID.

```
# generate long format and split sample column to sub groups
-----
metabolomics_long <- gather(metabolomics_data_filtered, key,
value,
-`Compound (Metabolite)`,
-`METABOLIC PATHWAY`,
convert = TRUE, na.rm = TRUE) %>%
separate(into = c("time", "strain", "Day", "sampleID"),
col = key,
sep = "-")
```

5. The newly generated data can be then grouped accordingly and the `summarize` function is used to calculate the mean, standard deviation, and the number of samples (n) for each metabolite and sample group.

```
# group data and summarize -----
-----
metabolomics_summary <- metabolomics_long %>%
group_by(`Compound (Metabolite)`, time, strain) %>%
summarize(mean = mean(value),
sd = sd(value),
n = n())
```

6. Next the data is merged with the table containing the mapping of the measured and model metabolites. Since not all metabolites that have been measured so far are on the metabolic model only metabolites that have a model ID are kept.

```
metabolomics_summary <- metabolomics_summary %>% left_join(..,
                                         model_mapping,
                                         by = c("Compound (Metabolite)" =
"METABOLITE")) %>%
filter(!is.na(MODEL))
```

7. On Escher maps only two sample conditions can be compared at the moment. Therefore, data has to be selected accordingly. The code below uses only metabolite and data points fitting to the glp condition and measured at 65 or 73 h. Afterward, a data frame suitable for mapping to Escher maps is generated and written to a .csv file.

```
# use filter function to get strain and time points of
interest -----
metabolomics_summary %>% filter(strain == "glp",
time %in% c("65", "73")) %>%
ungroup() %>%
unite(sampleID, strain, time) %>%
select(sampleID, MODEL, mean) %>%
spread(key = sampleID, value =mean) %>%
rename(ID = MODEL) %>%
write_csv("Escher/escher_metabolites.csv")
```

3.3.2 Mapping and Interpretation of Data

The resulting CSV can then be loaded into *Escher* through the map JSON file created in Subheading 3.2.2. An example map of the TCA cycle, PPP, and glycolysis pathway is available in the Demo Scripts in Subheading 2.

1. To overlay data onto the pathway maps, we begin by importing dependencies.

```
import cobra
from escher import Builder
```

2. We next load the previously constructed pathway map and open it in a browser window. The file name here can be replaced with the path to the map you generated in Subheading 3.2.2.

```
escher_model = Builder(map_json=" TCA_PPP_Glycolysis.json")
escher_model.display_in_browser()
```

3. In the *Escher* Builder window, use the menu to select “Data” → “Load Metabolite Data”. Navigate to the “escher_metabolites.csv” file and select it to overlay the data onto the pathway map. Transcriptomic data can also be used in this step to further augment the visualization, by selecting “Data” → “Load Gene Data” or “Data” → “Load Reaction Data”, depending on how your data is formatted. The heatmap is customizable from the “View” “Settings” option of the main menu.

3.4 Integrating Endo- and Exometabolomic Data with FBA

3.4.1 Using Endometabolome Data to Constrain a Flux Balance Analysis

To incorporate data from the endometabolome into the model, time series metabolomic data of the cells of interest are required. MetabFBA is a method which integrates intracellular metabolomic changes with the objective function of the FBA problem, requiring the model to allow for the production or consumption of metabolites within the system between time points [20]. The central logic of MetabFBA is shown in Fig. 4.

If we compare an example metabolite (Fig. 4b), called M , the concentration drops during Timeframe f1, indicating that the demand of M by reactions is greater than the generation of M by reactions. Conversely in Timeframe f2, the concentration of M increases; thus, the supply must have exceeded the demand. The changes in concentration of metabolites across a timeframe provide information on the changes that are occurring within the system of interest. The implication that these changes are a result of net production or consumption of metabolites over time is used by MetabFBA to augment the objective function, adding the demand and supply of changing metabolites as variables to be maximized. In this section we present a basic Python implementation of MetabFBA.

1. The experimental metabolomics data must first be mapped to the model before any further analysis can occur. A series of t -tests are used to assess if the concentrations of metabolites have changed significantly; thus, replicates are needed. To begin, we generate a table of metabolites and their concentrations at various time points. An example of this can be seen in the file “MetabolomicsData/metabolomics_table.tsv”. Additionally, a table that allows for the conversion of experimental metabolite identifiers in the dataset with the identifiers used within the model must also exist (e.g., “MetabolomicsData/metabolite-model-mapping.tsv”).
2. The metabolite data table is then converted to a “Differences” table (example: “Metabolomicsdata/metabo-diffs.tsv”), in which each column represents the difference in concentrations for each timeframe (the time between adjacent time points), and each row is a different metabolite. To do this, we perform t -tests for each combination of metabolite and timeframe in to

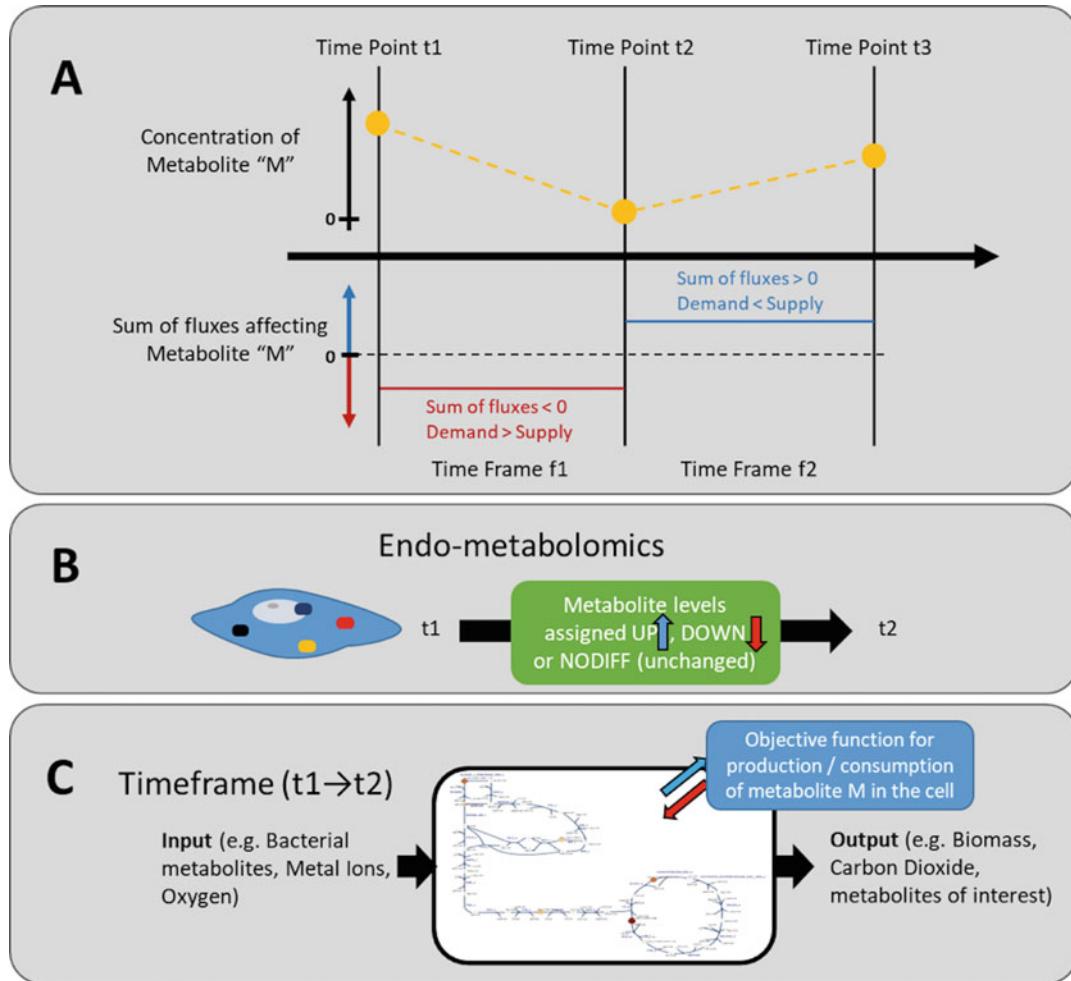


Fig. 4 Overview of MetabFBA approach: (a) Schematic overview of the MetabFBA approach, where t1 and t2 are two consecutive time points ($t_2 > t_1$). (b) Schematic representation of the underlying assumption that metabolite level changes between time points correspond to sustained differences in fluxes within the timeframe

determine significance. If the change is significantly increased or decreased, then the value of that cell in the table is set to “UP” or “DOWN,” respectively. If there is no significant change, we use “NODIFF” as the value of the cell.

3. To integrate the metabolomics data into the model, Python is used, alongside the *COBRApy* module, and Python’s in-built CSV module. In this example, the data tables are saved as TSV files. First, we import the necessary modules, create variables for the locations of the relevant files and load the data into Python. We load the mapping table as a dictionary of key-value pairs and the differences table are loaded into nested lists. A list

of metabolites present in the data is also generated. The code presented here is also available from the Demo Scripts.

```
import csv
import cobra
import cobra.flux_analysis
from cobra.core import Reaction

mapping_file = "PATH_TO_MAPPING_TABLE"
metabolite_file = "PATH_TO_DIFFERENCES_TABLE"
model_file = "PATH_TO_MODEL_FILE"

with open(mapping_file,"r") as file:
    mapping_data_input = list(csv.reader(file,delimiter="\t"))
    mapping_data = {entry[0]:entry[1] for entry in mapping_data_input}

with open(metabolite_file,'r') as file:
    metabolite_data = list(csv.reader(file,delimiter="\t"))
    metabolites = [row[0] for row in metabolite_data][1:]
```

4. Next, we define a function that will allow us to add reactions for the increase and decrease of metabolites that change significantly in the data.

```
def addMetabolomicsReaction(model, metabolites, react_name,
coefficient_str=1):
    print(react_name,":",metabolites)
    # Build a reaction
    coefficient_str = str(coefficient_str)
    reaction_str = coefficient_str + (" + "+coefficient_str).join(
metabolites) + " <=> " + react_name+"_c"
    reaction = Reaction(react_name)
    reaction.name = react_name
    reaction.subsystem = 'Metabolomics integration'
    # Add the reaction to the model
    model.add_reactions([reaction])
    reaction.build_reaction_from_string(reaction_str)
    #print(reaction,"::::::",reaction_str)
    return reaction
```

This function takes a COBRA model object, a list of metabolites, the name of the reaction, and a coefficient that is set to 1 by default. It then constructs the COBRA Reaction object that converts the metabolites list to a single “lumped” metabolite and returns it.

5. We next define a function to integrate the metabolomics data. This function accepts the model and a timeframe as arguments, then finds the column in the data relating to the timeframe. The “UP” and “DOWN” tagged metabolites are then separated into their own lists and converted to the metabolite identifiers used in the model using the mapping table. The previously defined `addMetabolomicsReaction` function is used to generate a demand reaction for both the increasing and decreasing metabolites, before the new demand reactions are incorporated alongside the Biomass reaction of the model into the objective function of the FBA problem.

```
def integrate_metabolomics(model,group):

    ## generate a list of the UP, DOWN and NODIFF assignments to
    metabolites for the column "group"

    metabo_diff_values = [row[metabolite_data[0].index(group)]]
    for row in metabolite_data][1:]

        # filter the increased and decreased metabolites and
        translate the identifiers to model identifiers using the
        mapping data

    up_mets_codes = [metabolites[i] for i in range(len
    (metabolites)) if metabo_diff_values[i] == "UP"]

    up_mets_codes = [mapping_data[i] for i in up_mets_codes if
    i in mapping_data and mapping_data[i] != ""]

    down_mets_codes = [metabolites[i] for i in range(len
    (metabolites)) if metabo_diff_values[i] == "DOWN"]

    down_mets_codes = [mapping_data[i] for i in down_mets_codes if
    i in mapping_data and mapping_data[i] != ""]

    # arbitrary threshold

    threshold = 2.5

    # If there are changed metabolites, add a sink reaction for
    them.

    if (len(up_mets_codes)>0):

        up_mets_react = addMetabolomicsReaction(model,
        up_mets_codes, "up_metabolites")
```

```

model.metabolites.up_metabolites_c.compartment = 'c'

model.add_boundary(model.metabolites.up_metabolites_c,
type='sink',lb=0,ub=threshold)

if (len(down_mets_codes)>0):

    down_mets_react = addMetabolomicsReaction(model,
down_mets_codes, "down_metabolites")

model.metabolites.down_metabolites_c.compartment = 'c'

model.add_boundary(model.metabolites.down_metabolites_c,
type='sink',lb=-1*threshold,ub=0)

# Add the demand reactions to the model objective

if (len(up_mets_codes)>0 and len(down_mets_codes)>0):

    model.objective = model.reactions.BIO0100.flux_expression +
model.reactions.up_metabolites.flux_expression - model.
reactions.down_metabolites.flux_expression

elif (len(up_mets_codes)>0):

    model.objective = model.reactions.BIO0100.flux_expression +
model.reactions.up_metabolites.flux_expression

elif (len(down_mets_codes)>0):

    model.objective = model.reactions.BIO0100.flux_expression -
model.reactions.down_metabolites.flux_expression

return model

```

6. Finally, we utilize the function to constrain the model with metabolomic data and perform a parsimonious FBA.

```

model = cobra.io.read_sbml_model(model_file)
model = integrate_metabolomics(model, "INSERT_COLUMN_NAME")
solution = cobra.flux_analysis.pfba(model)
print(solution.fluxes["BIO0100"])

```

The solution of the flux balance analysis will be displayed. The methods presented here may improve the quality of predictions alone; however, the model has not yet been

constrained in any other fashion, and thus configuring the growth medium and incorporating transcriptomic data will both greatly improve the predictive accuracy of the model through further constraining the solution space (**Note 4**).

3.4.2 Using Exometabolome Data to Constrain a Flux Balance Analysis

The exometabolome can also be used to constrain an FBA. The logic of the approach is related to the use of endometabolomic data, but rather than modifying the objective function, the data are used to constrain the FBA directly via using exchange fluxes between the cells and the external medium:

Extracellular metabolomic data can be obtained through the analysis of spent medium of cell culture. By analyzing the changes in metabolite concentration over time and the quantity of cells in the culture, it is then possible to constrain an FBA experiment. Differences in the exometabolome over time should be a result of transport of metabolites in and out of cells, which can then be used to place qualitative and/or quantitative constraints on the exchange reactions: The limits of detection of a specific metabolite and the concentration of that metabolite in the media can be used to place qualitative constraints on the respective exchange reaction. Furthering the use of extracellular metabolomic data, the absolute differences in metabolite concentrations can be used to constrain exchange reactions in a quantitative fashion.

This method to constrain an FBA with exometabolomic data has been incorporated into the *MetaboTools* toolbox [22], and the *MetaboTools* protocols article provides an excellent step-by-step description of the workflow and also contains tutorials of two use cases. Thus, the present chapter will not provide detailed instructions on the use of *MetaboTools*, as any description by us would merely duplicate already existing resources. Instead, the reader is referred to the detailed description of the protocol in the *MetaboTools* paper [22]. *MetaboTools* is part of the COBRA toolbox (<https://github.com/opencobra/cobratoolbox>), and detailed descriptions on the installation and use of the COBRA toolbox are provided in a previous protocol paper [24]. An applied example of the *MetaboTools* technique is the use of extracellular metabolomic data to constrain the human genome-scale model in order to study the metabolic profiles of cancer cells [21, 25].

4 Notes

1. Similar data reading and manipulation can be performed in other programming languages (e.g., Python). R was chosen since it is used frequently in metabolomics data analysis.
2. The generation of a suspect list for LC-MS is possible with the WormJam model, since it stores also neutral metabolites. In

case of other models only containing the charged version the charged formula and the charge state can be used to generate a neutral formula, which, in turn, can be used to calculate neutral mass and adduct m/z values.

3. The R package `masstrixR` can be installed from GitHub like a normal package using the `devtools::install_github()` function.

```
# install masstrixR -----
-----
devtools::install_github("michaelwitting/masstrixR")
```

4. Colijn et al. provide a convenient method for the integration of transcriptomics data in [26].

References

1. Palsson BO (2015) Systems biology: constraint-based reconstruction and analysis. Cambridge University Press, Cambridge
2. Thiele I, Palsson BO (2010) A protocol for generating a high-quality genome-scale metabolic reconstruction. *Nat Protoc* 5:93
3. Lee D-S (2010) Interconnectivity of human cellular metabolism and disease prevalence. *J Stat Mech* 2010(12):14
4. Bergdahl B, Sonnenschein N, Machado D, Herrgård M, Förster J (2015) Genome-scale models. In: Villadsen J (ed) Fundamental bioengineering, 1st edn. John Wiley & Sons, Inc., Hoboken, New Jersey. <https://doi.org/10.1002/9783527697441.ch06>
5. Tian M, Kumar P, STP G, Reed JL (2017) Metabolic modeling for design of cell factories. In: Nielsen J, Hohmann S (eds) Systems biology. John Wiley & Sons, Inc., Hoboken, New Jersey. <https://doi.org/10.1002/9783527696130.ch3>
6. Brunk E et al (2018) Recon3D enables a three-dimensional view of gene variation in human metabolism. *Nat Biotechnol* 36:272
7. de Oliveira Dal'Molin CG et al (2010) Ara-GEM, a genome-scale reconstruction of the primary metabolic network in Arabidopsis. *Plant Physiol* 152(2):579–589
8. Sigurdsson MI et al (2010) A detailed genome-wide reconstruction of mouse metabolism based on human recon 1. *BMC Syst Biol* 4 (1):140
9. Ebrahim A et al (2015) BiGG models: a platform for integrating, standardizing and sharing genome-scale models. *Nucleic Acids Res* 44 (D1):D515–D522
10. Edwards JS, Palsson BO (1999) Systems properties of the *Haemophilus influenzae* Rd metabolic genotype. *J Biol Chem* 274 (25):17410–17416
11. Krauss M et al (2012) Integrating cellular metabolism into a multiscale whole-body model. *PLoS Comput Biol* 8(10):e1002750
12. Yilmaz LS, Walhout AJM (2017) Metabolic network modeling with model organisms. *Curr Opin Chem Biol* 36:32–39
13. C. elegans Sequencing Consortium (1998) Genome sequence of the nematode *C. elegans*: a platform for investigating biology. *Science* 282(5396):2012–2018
14. Büchel F et al (2013) Path2Models: large-scale generation of computational models from biochemical pathway maps. *BMC Syst Biol* 7 (1):116
15. Gebauer J et al (2016) A genome-scale database and reconstruction of *Caenorhabditis elegans* metabolism. *Cell Syst* 2(5):312–322
16. Yilmaz LS, Walhout AJ (2016) A *Caenorhabditis elegans* genome-scale metabolic network model. *Cell Syst* 2(5):297–311
17. Ma L et al (2017) Systems biology analysis using a genome-scale metabolic model shows that phosphine triggers global metabolic suppression in a resistant strain of *C. elegans*. *bioRxiv*
18. Witting M et al (2018) Modeling meets metabolomics—the WormJam consensus model as basis for metabolic studies in the model

- organism *Caenorhabditis elegans*. Front Mol Biosci 5:96
- 19. Hastings J et al (2017) WormJam: a consensus *C. elegans* metabolic reconstruction and metabolomics community and workshop series. Worm 6(2):e1373939
 - 20. Hastings J et al (2019) Multi-omics and genome-scale modeling reveal a metabolic shift during *C. elegans* aging. Front Mol Biosci 6:2
 - 21. Aurich MK et al (2015) Prediction of intracellular metabolic states from extracellular metabolomic data. Metabolomics 11(3):603–619
 - 22. Aurich MK, Fleming RMT, Thiele I (2016) MetaboTools: a comprehensive toolbox for analysis of genome-scale metabolic models. Front Physiol 7:327
 - 23. King ZA et al (2015) Escher: a web application for building, sharing, and embedding data-rich visualizations of biological pathways. PLoS Comput Biol 11(8):e1004321
 - 24. Schellenberger J et al (2011) Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox v2.0. Nat Protoc 6:1290
 - 25. Aurich MK, Fleming RMT, Thiele I (2017) A systems approach reveals distinct metabolic strategies among the NCI-60 cancer cell lines. PLoS Comput Biol 13(8):e1005698
 - 26. Colijn C, Brandes A, Zucker J, Lun DS, Weiner B, Farhat MR, Cheng T-Y, Moody DB, Murray M, Galagan JE (2009) Interpreting expression data with metabolic flux models: predicting *Mycobacterium tuberculosis* mycolic acid production. PLoS Comput Biol 5(8):e1000489. <https://doi.org/10.1371/journal.pcbi.1000489>



Chapter 19

Pathway Analysis for Targeted and Untargeted Metabolomics

Alla Karnovsky and Shuzhao Li

Abstract

Recent advances in analytical techniques, particularly LC-MS, generate increasingly large and complex metabolomics datasets. Pathway analysis tools help place the experimental observations into relevant biological or disease context. This chapter provides an overview of the general concepts and common tools for pathway analysis, including Mummichog for untargeted metabolomics. Examples of pathway mapping, MetScape, and Mummichog are explained. This serves as both a practical tutorial and a timely survey of pathway analysis for label-free metabolomics data.

Key words Metabolomics, Untargeted metabolomics, MetScape, Mummichog, Pathway analysis, Metabolic network

1 Introduction

The wealth of information of biochemical reactions, accumulated over more than a century of scientific investigations, is often found in biochemistry textbooks organized into one pathway per chapter. Many of the metabolic pathways, such as urea cycle, tricarboxylic acid cycle, glycolysis, and oxidative phosphorylation, were elucidated in the first half of the past century. Pathways consist of reactions, and reactions are defined by reactants and products and often catalyzed by enzymes. Enzymes are proteins that are encoded in the genome sequence. The emergence of high-throughput gene expression studies in the 1990s prompted the development of pathway analysis techniques to help contextualize and interpret the results. The purpose of these techniques is to help reduce the data involving hundreds of altered genes to smaller and more interpretable sets of altered biological “concepts,” helping generate testable hypotheses. Application of these techniques allowed researchers to move from assessing changes in individual genes, to evaluating significance of pathways and other meaningful biological categories. As pathways represent prior knowledge, the collective

statistical power can also identify patterns that are missed at individual gene level. Later, as metabolomics data became available, similar methods were applied to perform pathway analysis on metabolites.

1.1 Pathway Analysis Approaches for Metabolomics Data

Various mathematical and computational formalisms can be used to model metabolic pathways. For example, an ordinary differential equation can incorporate the concentrations of metabolites, reaction rates, and mass balance, to model a metabolic reaction nicely. However, most metabolomics experiments measure the averaged concentration of each metabolite at a particular moment of a biological specimen, and do not capture the precise information that is required for such models. A simplified approach is flux balance analysis, which assumes steady state of all reactions in a closed system. More often, statistical models not kinetic models are used in biomedical investigations of the metabolomes. Stable isotope labeling combined with metabolomics provides a powerful tool to trace reactions and pathways. We refer the readers to Chapter 6 for isotope tracing and Chapter 18 for flux balance analysis. This chapter will focus on the pathway-based bioinformatics analysis of label-free metabolomics data.

Analytical approaches in metabolomics are often referred in two categories: targeted and untargeted. Targeted metabolomics is the measurement of defined groups of chemically characterized and biochemically annotated metabolites [1]. In contrast, untargeted metabolomics is the comprehensive analysis of all measurable metabolites in a sample. In addition to a set of identified metabolites, untargeted experiments can yield thousands unidentified features, of which, a fraction may represent unique metabolites [2]. Common bioinformatics approaches, such as enrichment tests, can be used for the analysis of targeted and the known portion of untargeted metabolomics data (Fig. 1a). Incorporating the unknown portion of untargeted experiments requires distinct bioinformatics techniques (Fig. 1b). Therefore, in this chapter, we will discuss these approaches separately.

1.2 Pathway Databases and Genome Scale Metabolic Models

A number of databases catalog information on metabolic pathways, metabolites, metabolic reactions, enzymes and the genes that encode them (**Note 1**). Kyoto Encyclopedia of Genes and Genome (KEGG) pioneered this area and is still widely used [3]. Extensive literature search and expert curation set the initial stage. Later, genome sequences enabled a new approach of metabolic reconstruction. That is, proteins encoding metabolic enzymes can be inferred from genome sequences, and further linked to their respective biochemical reactions. The reactions are connected into metabolic pathways and networks, under the term “genome scale metabolic models” (see Chapter 18). BioCyc [4], RECON [5], and EHMN [6] were some early notable examples. The latter two were merged into a community model now named RECON3D

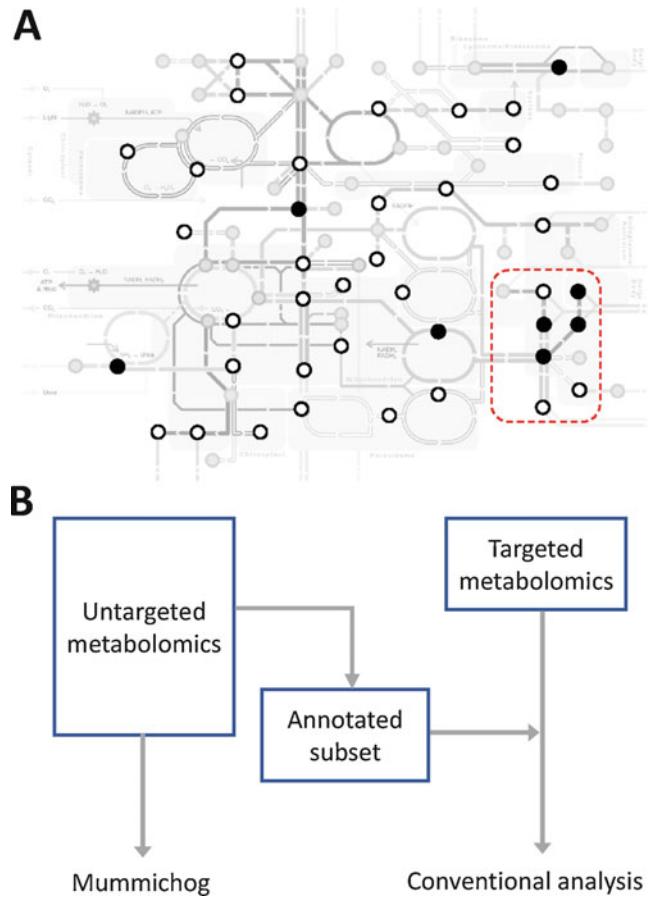


Fig. 1 Pathway analysis for metabolomics. **(a)** Measured metabolites are mapped to metabolic pathways. Each circle represents a metabolite, whereas filled circles are significant metabolites selected based on in study question. A metabolomics experiment only measures a subset of metabolites. Most common analysis is the enrichment of significant metabolites on a pathway, as indicated by the red box with dashed lines. The background is based on Chakazul's Metabolic Metro Map (<https://en.wikipedia.org/wiki/User:Chakazul>). **(b)** Flowchart to choose an approach of pathway analysis. For untargeted metabolomics, one can analyze it directly using the Mummichog algorithm, or perform annotation and treat the annotated subset as targeted metabolomics

[7]. The rapidly expanding number of available genomic sequences and amount of gene expression data popularized this approach. Subsequently, a number of more detailed organism-specific metabolic reconstructions have been developed [5–9]. In addition to detailed information about metabolic pathway topology and individual components of pathways, some of these include information about subcellular compartments where the metabolic reactions occur [9], and describe metabolic enzyme complexes and

transporters [5]. Different from these genome-centric approaches, a recent more metabolite-centric effort is Small Molecule Pathway Database (SMPDB) [10].

2 Materials

This chapter gives an overview of the concepts in metabolomics pathway and network analysis. One needs a computer with a modern web browser and Internet access to replicate the exercises, which are given on two software programs, *Cytoscape* with the *MetScape* app and *Mummichog*. Two datasets are provided on our GitHub site (<https://metabolomics-data.github.io/>) to be used for these examples.

3 Metabolite Identifiers and Pathway Mapping

To query a database and a pathway, a compatible identifier is required for each metabolite. Due to the head start, KEGG identifiers are commonly used by other software. When one uses other databases such as PubChem and HMDB, options are given to link to KEGG identifiers. Metabolite identifier conversion can be via some web tools, which are listed as notes at the end of this Chapter. Conventions like InChI and SMILES provide structural information of the molecule and are encouraged to be used along with identifiers and common names.

Structural code and common name and structural code (e.g., InChI or SMILES).

If the goal is to map a list of metabolites on pathways, statistical tests are not necessarily involved. KEGG and Biocyc provide tools for mapping and “painting” metabolites on pathways. To map human metabolites to KEGG pathways:

Step 1. Go to https://www.genome.jp/kegg/tool/map_pathway2.html.

Step 2. Enter “hsa” in the “Search against” box.

Step 3. Copy the list of metabolites in KEGG identifiers, one metabolite per line, paste to use here, then click “Exec.” If users do not have a list of metabolites to use, example data can be found on our companion GitHub page.

Step 4. On the result page, click individual pathways to explore.

The metabolites from your input should be colored by pink or your designated color.

4 Statistical Tests and Application to Targeted Metabolomics

From the field of transcriptomics, pathway analysis can be classified into three groups: overrepresentation analysis (e.g., DAVID [11]), functional class scoring (e.g., GSEA [12]), and approaches that utilize pathway topology (reviewed in [13]) (**Note 2**). Tools in the first two categories do not take into account the interactions between members of a pathway, while the third group of methods considers such interactions. Extensive experience of pathway and network analysis of genomic data, particularly in cancer, has shown that consensus pathway topology can be altered in a disease condition [14]. In addition to gene expression data, the development of functional enrichment techniques had a significant impact on the analysis of epigenetics and GWAS data [12, 15–18]. These pathway analysis tools provide intuitive results that can be easily interpreted by biologists. Their popularity has stimulated the development of similar tools for metabolomics [19–22].

The overrepresentation test is found in almost all software packages for pathway analysis, using either Fisher's exact test or hypergeometric test (the two are similar for large sample sizes). Using the example in Fig. 1a, there are 40 metabolites measured in the illustrated experiment. In the pathway defined by the red dashed circle, 4 of the 7 metabolites are significant. Out of the total 40 metabolites, 8 are significant. Thus, the significance of the pathway is computed using Fisher's exact test as:

$$P = \frac{\binom{7}{4} \binom{33}{4}}{\binom{40}{8}} = 0.02$$

A few limitations apply to this approach. The coverage of many targeted analysis is small, thus insufficient to investigate all the pathways. The curation of pathways themselves could be limited, and only a handful of metabolites from a given pathway are typically measured. Overlap between pathways and correlations between metabolites from different pathways can induce positive dependence between pathways, requiring special adjustments in order not to inflate the significance of such pathways. Overrepresentation analysis and functional class scoring methods used by most existing metabolite enrichment tools do not address these problems efficiently. The application of GSEA method should be cautioned on a separate issue: the level of metabolites in a pathway can change in different directions. Should an enzyme be dysfunctional in a pathway, it would result in the accumulation of the upstream metabolite but depletion of the downstream metabolite. Therefore, the ranking method in GSEA may not apply to such data.

Commercial software programs are available for metabolomics pathway analysis. MetaCore and Ingenuity Pathway Analysis are among the popular choices. So far, these software programs only work with targeted metabolomics data (**Note 3**).

5 Pathway Visualization

Automated layout of networks has progressed well in the recent years. However, pathway visualization is more challenging because logics of chemical structures and conversions are often desired, and human conventions are often preferred. As a result, manual layout of metabolic pathways is prevailing. Early in the development, KEGG has coded their pathways in a markup language, which are reused by many other bioinformatic tools. Some of these tools utilize the static pathways charts [23, 24], while others make them interactive [25, 26]. One such tool called Paintomics can load metabolite and gene expression measurements and visualize them over KEGG pathway maps [24]. A more interactive tool, Wanted, has been developed for exploration of experimental metabolomics data in the context of metabolic pathways, originally from plants [25, 26]. However, it can be used for any data: users can load KEGG maps or build their own pathways. Another metabolomics pathways analysis tool MetPA [27], that is now part of the comprehensive data analysis package MetaboAnalyst, in addition to pathway mapping, calculates pathway impact based on normalized centrality measure of a given compound relative to the other compounds.

One of the limitations of visualizing data over pathways charts stems from the fact that metabolites are often involved in multiple pathways. In order to understand the overall effect of altered level of a given metabolite, the user has to go through multiple pathways and to understand the connections between them. An alternative to this approach is building a network of genes/metabolites where each node is unique and nodes from multiple pathways can be linked together. Such networks provide an easy way to connect multiple pathways and build gene/compound centric maps enabling quick data exploration and logical, well-informed hypothesis generation.

As the software ecosystem moves toward the web, native javascript based pathway visualization starts to emerge. Esher is an excellent example of this [28]. Esher allows for building, viewing, and sharing pathway visualization online. A practical example of Esher is given in Chapter 18.

6 Example Using Metscape

MetScape [29] is a plugin for a widely used network visualization program CytoScape [30]. It allows users to upload a list of metabolites with experimentally determined concentrations and map them to reactions, genes and pathways. It also supports identification of enriched biological pathways from expression profiling data, building the networks of genes and metabolites involved in these pathways, and allow users to visualize the changes in the gene/metabolite data over time/experimental conditions. MetScape is using human metabolic pathways, although, it can also map mouse and rat genes to their human homologs. A screenshot of CytoScape 3.7.1/MetScape 3.1.3 is shown in Fig. 2. To replicate this example:

- Step 1. Obtain *CytoScape* from <https://cytoscape.org/>. Install *CytoScape* by following instructions.
- Step 2. Launch *CytoScape*. From the “Apps” menu, open “App Manager.” Search “MetScape” and install it.
- Step 3. Now MetScape can be found under the “Apps” menu. Click the “MetScape” tab, open “Build Network” then “Pathway-based.”
- Step 4. Download a copy of the example data from our companion GitHub page if not done so. This is the same file as the KEGG mapping exercise.
- Step 5. Input the example data file using the “Select” button in the input box in the left panel. After the data file is read (ignore warnings), click “Build Network” at the bottom of the left panel. The network of metabolites similar to Fig. 2 should appear in the main panel.

7 Untargeted Metabolomics Analysis

The advantage of untargeted metabolomics is to measure as many metabolites as possible without requiring a priori hypothesis. For data generated on high-resolution mass spectrometry, most peaks are likely to be unique individual features. When some of these peaks match to a chemical library that was generated on the same platform in the same lab, the annotation of the library can be transferred to these peaks, resulting in a data table of partial annotation. Sometimes, the annotated peaks are presented separately as targeted data. In addition to reference-based annotation, computational methods are available to perform annotation by combining database searches with heuristic rules (e.g., isotopic patterns, correlation of peak intensities). Furthermore, it becomes possible to retrospectively annotate peaks in archived high-resolution data.

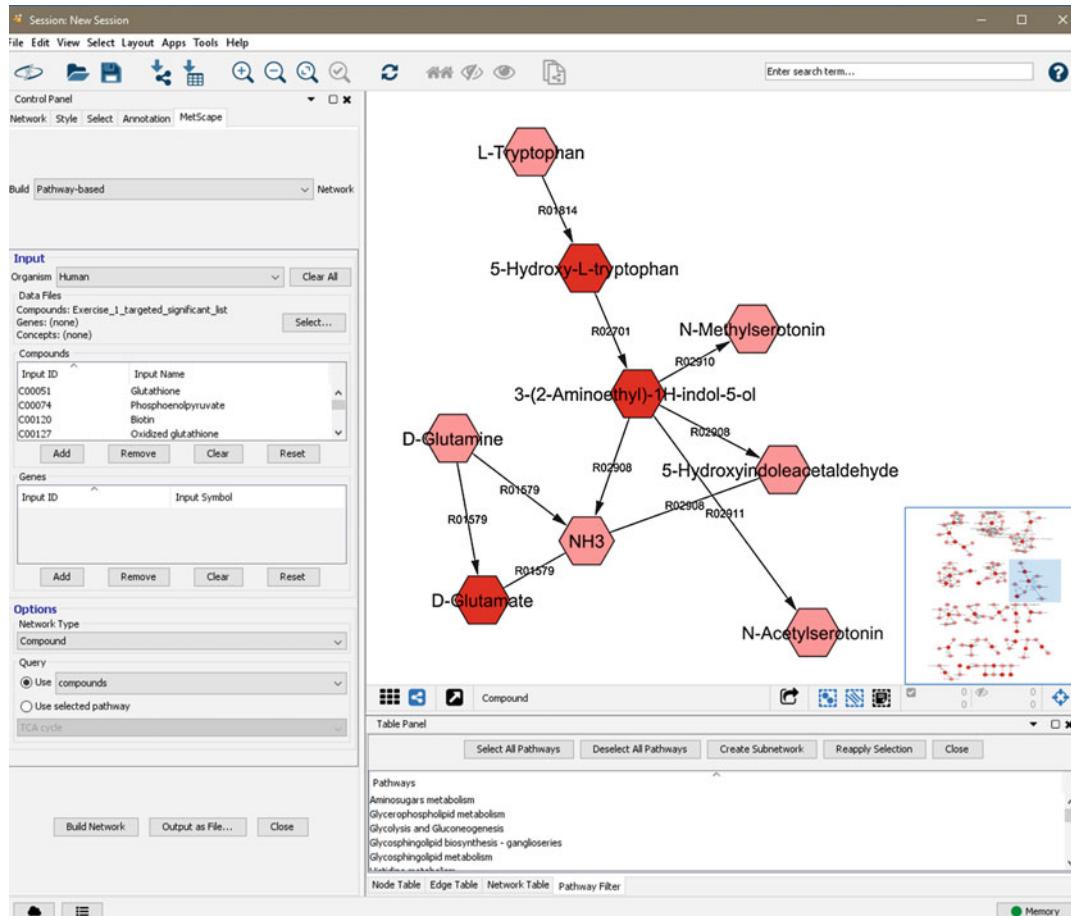


Fig. 2 Screenshot of MetScape within the CytoScape framework. Using our example data, a metabolite network is formed by connecting known metabolic reactions that contain the input metabolites. The full network is shown in the thumbnail and the metabolites of tryptophan pathway are shown in the center of the window

There is no consensus on the best approach of processing and analyzing untargeted metabolomics. It is reasonable to expect a computational annotation tool to group related features into metabolites and to produce a list of metabolites of high-quality annotation. This “*in silico*” list can be treated as targeted data and analyzed using targeted tools as discussed in previous sections, while the peaks without high-quality annotation will be ignored.

A common mistake in untargeted LC-MS metabolomics is to search each feature in a public database, and use the top or an arbitrary match for pathway analysis. Since the chromatographic information is usually irrelevant in such database search, the top match is usually only one of many possible candidates for the true metabolite. The arbitrary selection of a candidate bears a small probability of finding the true metabolite, and this selection is not

necessarily better based on the smallest mass difference between the query and the candidate. When this process is repeated for many metabolites, one ends up using a list of mostly false metabolite identifiers for downstream analysis.

The proper approach should accommodate the ambiguity of metabolite identification in the statistical framework of pathway analysis. This was the design of the Mummichog algorithm [31]. From each feature measured in LC-MS, a list of possible candidate metabolites is computed (no more than one metabolite can be true per feature). The algorithm searches pathways using the combination of all candidates, and calculates an enrichment *p*-value for each combination per pathway. Given that most of the candidates and combinations are false predictions, only few results are expected to land on the true metabolites and correct pathways. The problem becomes how to distinguish those real signals from random data. The distribution of random data can be simulated by permutating the input data. By contrasting the pathway enrichment result started from statistically significant features against random data, the likelihood of finding the correct pathways and metabolites can be quantified.

Not all metabolites are assigned to a pathway, and many metabolites are shared by multiple pathways. A network model of metabolic reaction is free of the arbitrary boundaries between pathways, and the unbiased network topology allows us to test network modules that are enriched by a metabolomics experiment. Both pathway analysis and network module analysis are provided by Mummichog. This enables rapid generation of high-quality of hypotheses directly from untargeted metabolomics without requiring prior annotation, thus accelerating the rate of scientific discovery. Mummichog has been incorporated into the popular web platforms of on XCMS Online [32] and MetaboAnalyst [33]. It is also available as an open-source Python package, and a standalone server (linked via mummichog.org), which is a simple place to use the tool. Mummichog does not deal with feature level statistics. Users will produce a *p*-value per mass feature using their method of choice (e.g., *t*-test or linear regression to test significance according to the study design). A table of mass features (mass to charge ratio, retention time and *p*-values) is used as input to Mummichog. Mummichog only supports LC-MS data at the moment, but ongoing development aims to support more data types and more upstream processing pipelines. The example output from Mummichog is shown in Fig. 3. Users can use the example data from our GitHub site to replicate these results.

This strategy of matching patterns in projected spaces between mass spectrometry and biological models can be broadly applied, including the integration with other omics data. PIUMet is a software tool that extends the Mummichog approach by including protein interaction data, thus integrating protein and gene

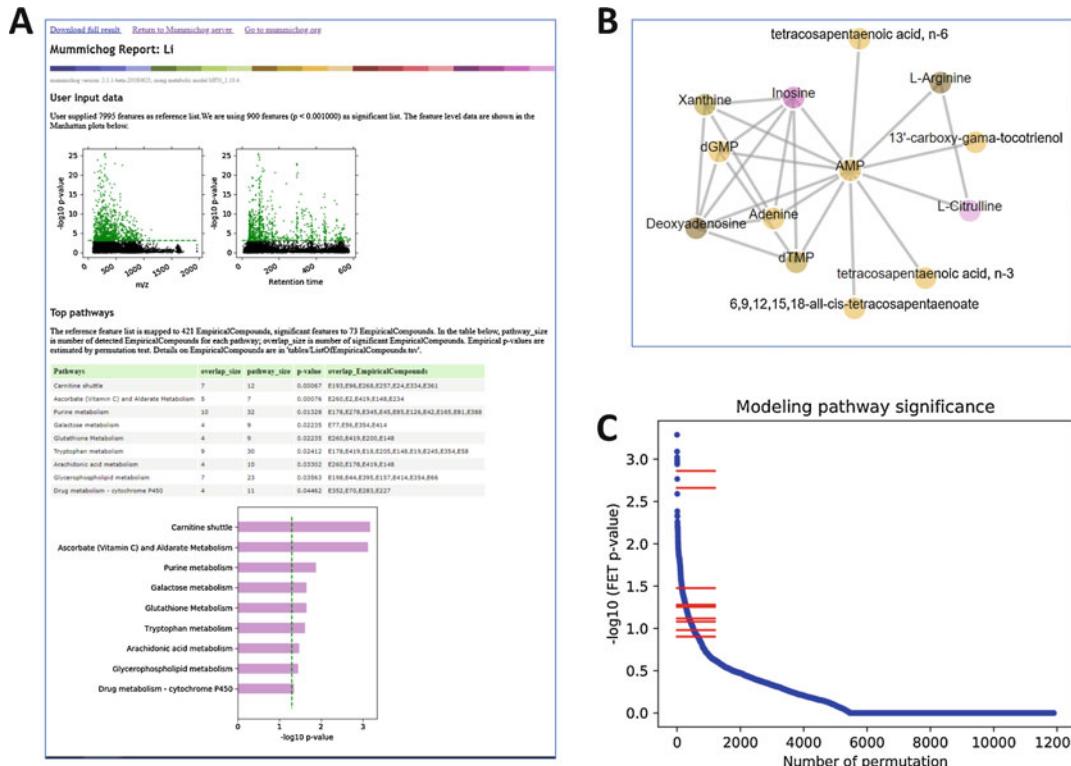


Fig. 3 Example output from Mummichog (version 2.1). (a) The summary report contains a replot of user input data as two Manhattan plots. Significant pathways are shown as a table and a bar plot. Not included but scrollable in the screenshot is the module analysis and list of metabolites of interest. (b) Example of a significant network module. (c) Mummichog computes an empirical p -value by comparing the pathway FET p -values (red) against those from permutation data (blue). Details of the algorithm are given in Li et al. (2013) [31]

expression data with untargeted metabolomics [34]. Because metabolomics data are often orthogonal to gene expression and other data types, multi-omics integration is often a data-driven process (see Chapter 23). Prior annotation is not required to perform such data driven integration until the step of interpreting the data, where Mummichog can be applied. This works well to include the global measurement, since discovery of novel biology is a main motivation of many studies. The online resources related to this chapter are listed in Note 4.

8 Notes

- There is a relatively low coverage of experimentally measured metabolites by biological pathway databases. Most existing tools rely on the same set of pathway databases that provide a genome-centric view of the metabolome. In contrast, experimental metabolomics provides a chemistry-centric view of

metabolome determined by a chosen analytical method. The areas of overlap of these two views typically include primary metabolism, while the coverage of secondary and lipid metabolism is scarce [35]. Further, exogenous compounds that are routinely detected by metabolomics assays are not included in pathway maps. As a result, often in current studies, only half of detected named polar metabolites and even fewer lipids can be mapped onto metabolic pathways. The improvement of pathway curation based on experimental measurement of metabolites will be an important area of scientific investigation.

2. Both pathways and networks can be mathematically represented as graphs. Since pathways are predefined with scientific conventions, this infusion of human knowledge could increase the sensitivity of analysis. Analysis via network structure can be less biased, because networks do not have the boundaries between pathways.
3. Statistical significance is not the same thing as biological significance. If one or few metabolites lead to critical biological insight, pathway analysis is not even necessary. After pathway analysis, one should validate the findings in some form, especially with untargeted metabolomics data. Mummichog can shift the step of metabolite annotation and confirmation after pathway analysis. Chemical standards and MSⁿ experiments are a good way to validate metabolites (*see Chapter 1* for reporting standards).
4. Resources.

Metabolic pathways and models.

KEGG: <http://www.genome.jp/kegg/kegg2.html>

BioCyc: <https://biocyc.org>

Recon: <https://www.vmh.life/#human/all>

Metabolite ID conversion.

<https://cts.fiehnlab.ucdavis.edu>

<https://www.metaboanalyst.ca/faces/upload/ConvertView.xhtml>

Metabolomics data analysis tools.

MetaboAnalyst: <https://www.metaboanalyst.ca>

MetScape: <http://metscape.ncibi.org/>

MetExplore: <https://metexplore.toulouse.inra.fr/>

Metabox/Met-DA: <http://metda.fiehnlab.ucdavis.edu>

XCMS Online: <https://xcmsonline.scripps.edu>

Mummichog: <http://mummichog.org>

Network visualization tools.

Cytoscape: <http://cytoscape.org>

Gehpi: <https://gephi.org>

Acknowledgments

This work has been funded in part by the US national Institutes of Health via grants U01CA235487 (Karnovsky, Michailidis), U19 AI090023 (Pulendran), U2C ES026560 (Marsit), and U01 CA235493 (Li, Xia, Siuzdak).

References

1. Roberts LD, Souza AL, Gerszten RE, Clish CB (2012) Targeted metabolomics. *Curr Protoc Mol Biol.* Chapter 30:Unit 30.32.31-24. <https://doi.org/10.1002/0471142727.mbm3002s98>
2. Mahieu NG, Patti GJ (2017) Systems-level annotation of a metabolomics data set reduces 25000 features to fewer than 1000 unique metabolites. *Anal Chem* 89(19):10397–10406. <https://doi.org/10.1021/acs.analchem.7b02380>
3. Kanehisa M, Goto S, Hattori M, Aoki-Kinoshita KF, Itoh M, Kawashima S, Katayama T, Araki M, Hirakawa M (2006) From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Res* 34 (database issue):D354–D357
4. Caspi R, Altman T, Dreher K, Fulcher CA, Subhraveti P, Keseler IM, Kothari A, Krummenacker M, Latendresse M, Mueller LA, Ong Q, Paley S, Pujar A, Shearer AG, Travers M, Weersinghe D, Zhang P, Karp PD (2012) The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. *Nucleic Acids Res* 40(Database issue):D742–D753. <https://doi.org/10.1093/nar/gkr1014>
5. Duarte NC, Becker SA, Jamshidi N, Thiele I, Mo ML, Vo TD, Srivas R, Palsson BO (2007) Global reconstruction of the human metabolic network based on genomic and bibliomic data. *Proc Natl Acad Sci U S A* 104(6):1777–1782. <https://doi.org/10.1073/pnas.0610772104>
6. Ma H, Sorokin A, Mazein A, Selkov A, Selkov E, Demin O, Goryanin I (2007) The Edinburgh human metabolic network reconstruction and its functional analysis. *Mol Syst Biol* 3:135. <https://doi.org/10.1038/msb4100177>
7. Thiele I, Swainston N, Fleming RM, Hoppe A, Sahoo S, Aurich MK, Haraldsdottir H, Mo ML, Rolfsson O, Stobbe MD, Thorleifsson SG, Agren R, Bolling C, Bordel S, Chavali AK, Dobson P, Dunn WB, Endler L, Hala D, Hucka M, Hull D, Jameson D, Jamshidi N, Jonsson JJ, Juty N, Keating S, Nookaei I, Le Novere N, Malys N, Mazein A, Papin JA, Price ND, Selkov E Sr, Sigurdsson MI, Simeonidis E, Sonnenschein N, Smallbone K, Sorokin A, van Beek JH, Weichert D, Goryanin I, Nielsen J, Westerhoff HV, Kell DB, Mendes P, Palsson BO (2013) A community-driven global reconstruction of human metabolism. *Nat Biotechnol* 31(5):419–425. <https://doi.org/10.1038/nbt.2488>
8. Sigurdsson MI, Jamshidi N, Steingrimsson E, Thiele I, Palsson BO (2010) A detailed genome-wide reconstruction of mouse metabolism based on human Recon 1. *BMC Syst Biol* 4:140. <https://doi.org/10.1186/1752-0509-4-140>
9. Hao T, Ma HW, Zhao XM, Goryanin I (2010) Compartmentalization of the Edinburgh human metabolic network. *BMC Bioinformatics* 11:393. <https://doi.org/10.1186/1471-2105-11-393>
10. Jewison T, Su Y, Disfany FM, Liang Y, Knox C, Maciejewski A, Poelzer J, Huynh J, Zhou Y, Arndt D, Djoumbou Y, Liu Y, Deng L, Guo AC, Han B, Pon A, Wilson M, Rafatnia S, Liu P, Wishart DS (2014) SMPDB 2.0: big improvements to the small molecule pathway database. *Nucleic Acids Res* 42(Database issue):D478–D484. <https://doi.org/10.1093/nar/gkt1067>
11. Huang DW, Sherman BT, Tan Q, Collins JR, Alvord WG, Roayaei J, Stephens R, Baseler MW, Lane HC, Lempicki RA (2007) The DAVID gene functional classification tool: a novel biological module-centric algorithm to functionally analyze large gene lists. *Genome Biol* 8(9):R183. <https://doi.org/10.1186/gb-2007-8-9-r183>
12. Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA, Paulovich A, Pomeroy SL, Golub TR, Lander ES, Mesirov JP (2005) Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc Natl Acad Sci U S A* 102(43):15545–15550. <https://doi.org/10.1073/pnas.0506580102>
13. Khatri P, Sirota M, Butte AJ (2012) Ten years of pathway analysis: current approaches and outstanding challenges. *PLoS Comput Biol* 8

- (2):e1002375. <https://doi.org/10.1371/journal.pcbi.1002375>
14. Creixell P, Reimand J, Haider S, Wu G, Shibata T, Vazquez M, Mustonen V, Gonzalez-Perez A, Pearson J, Sander C, Raphael BJ, Marks DS, Ouellette BFF, Valencia A, Bader GD, Boutros PC, Stuart JM, Linding R, Lopez-Bigas N, Stein LD (2015) Pathway and network analysis of cancer genomes. *Nat Methods* 12(7):615–621. <https://doi.org/10.1038/nmeth.3440>
 15. Huang DW, Sherman BT, Tan Q, Kir J, Liu D, Bryant D, Guo Y, Stephens R, Baseler MW, Lane HC, Lempicki RA (2007) DAVID bioinformatics resources: expanded annotation database and novel algorithms to better extract biology from large gene lists. *Nucleic Acids Res* 35(Web Server issue):W169–W175. <https://doi.org/10.1093/nar/gkm415>
 16. Lee PH, O'Dushlaine C, Thomas B, Purcell SM (2012) INRICH: interval-based enrichment analysis for genome-wide association studies. *Bioinformatics* 28(13):1797–1799. <https://doi.org/10.1093/bioinformatics/bts191>
 17. Segre AV, Groop L, Mootha VK, Daly MJ, Altshuler D (2010) Common inherited variation in mitochondrial genes is not enriched for associations with type 2 diabetes or related glycemic traits. *PLoS Genet* 6(8):e1001058. <https://doi.org/10.1371/journal.pgen.1001058>
 18. Cavalcante RG, Lee C, Welch RP, Patil S, Weymouth T, Scott LJ, Sartor MA (2014) Broad-enrich: functional interpretation of large sets of broad genomic regions. *Bioinformatics* 30(17):i393–i400. <https://doi.org/10.1093/bioinformatics/btu444>
 19. Cavalcante RG, Patil S, Weymouth TE, Bendinskas KG, Karnovsky A, Sartor MA (2016) ConceptMetab: exploring relationships among metabolite sets to identify links among biomedical concepts. *Bioinformatics* 32(10):1536–1543. <https://doi.org/10.1093/bioinformatics/btw016>
 20. Lopez-Ibanez J, Pazos F, Chagoyen M (2016) MBROLE 2.0-functional enrichment of chemical compounds. *Nucleic Acids Res* 44(W1):W201–W204. <https://doi.org/10.1093/nar/gkw253>
 21. Xia J, Wishart DS (2016) Using MetaboAnalyst 3.0 for comprehensive metabolomics data analysis. *Curr Protoc Bioinformatics* 55:14.10.11–14.10.91. <https://doi.org/10.1002/cpbi.11>
 22. Hernandez-de-Diego R, Tarazona S, Martinez-Mira C, Balzano-Nogueira L, Furio-Tari P, Pappas GJ Jr, Conesa A (2018) PaintOmics 3: a web resource for the pathway analysis and visualization of multi-omics data. *Nucleic Acids Res* 46(W1):W503–w509. <https://doi.org/10.1093/nar/gky466>
 23. Paley SM, Karp PD (2006) The pathway tools cellular overview diagram and Omics viewer. *Nucleic Acids Res* 34(13):3771–3778. <https://doi.org/10.1093/nar/gkl334>
 24. Garcia-Alcalde F, Garcia-Lopez F, Dopazo J, Conesa A (2011) Paintomics: a web based tool for the joint visualization of transcriptomics and metabolomics data. *Bioinformatics* 27(1):137–139. <https://doi.org/10.1093/bioinformatics/btq594>
 25. Junker BH, Klukas C, Schreiber F (2006) VANTED: a system for advanced data analysis and visualization in the context of biological networks. *BMC Bioinformatics* 7:109. <https://doi.org/10.1186/1471-2105-7-109>
 26. Klukas C, Schreiber F (2010) Integration of -omics data and networks for biomedical research with VANTED. *J Integr Bioinform* 7(2):112. <https://doi.org/10.2390/biec09-2010-112>
 27. Xia J, Wishart DS (2010) MetPA: a web-based metabolomics tool for pathway analysis and visualization. *Bioinformatics* 26(18):2342–2344. <https://doi.org/10.1093/bioinformatics/btp418>
 28. King ZA, Drager A, Ebrahim A, Sonnenschein N, Lewis NE, Palsson BO (2015) Escher: a web application for building, sharing, and embedding data-rich visualizations of biological pathways. *PLoS Comput Biol* 11(8):e1004321. <https://doi.org/10.1371/journal.pcbi.1004321>
 29. Karnovsky A, Weymouth T, Hull T, Tarcea VG, Scardoni G, Laudanna C, Sartor MA, Stringer KA, Jagadish HV, Burant C, Athey B, Omenn GS (2012) MetScape 2 bioinformatics tool for the analysis and visualization of metabolomics and gene expression data. *Bioinformatics* 28(3):373–380. <https://doi.org/10.1093/bioinformatics/btr661>
 30. Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D, Amin N, Schwikowski B, Ideker T (2003) Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res* 13(11):2498–2504. <https://doi.org/10.1101/gr.1239303>
 31. Li S, Park Y, Duraisingham S, Strobel FH, Khan N, Soltow QA, Jones DP, Pulendran B (2013) Predicting network activity from high throughput metabolomics. *PLoS Comput Biol* 9(7):e1003123. <https://doi.org/10.1371/journal.pcbi.1003123>

32. Huan T, Forsberg EM, Rinehart D, Johnson CH, Ivanisevic J, Benton HP, Fang M, Aisporna A, Hilmers B, Poole FL, Thorgeresen MP, Adams MWW, Krantz G, Fields MW, Robbins PD, Niedernhofer LJ, Ideker T, Majumder EL, Wall JD, Rattray NJW, Goodacre R, Lairson LL, Siuzdak G (2017) Systems biology guided by XCMS online metabolomics. *Nat Methods* 14(5):461–462. <https://doi.org/10.1038/nmeth.4260>
33. Chong J, Soufan O, Li C, Caraus I, Li S, Bourque G, Wishart DS, Xia J (2018) MetaboAnalyst 4.0: towards more transparent and integrative metabolomics analysis. *Nucleic Acids Res* 46(W1):W486–W494. <https://doi.org/10.1093/nar/gky310>
34. Pirhaji L, Milani P, Leidl M, Curran T, Avila-Pacheco J, Clish CB, White FM, Saghatelian A, Fraenkel E (2016) Revealing disease-associated pathways by network integration of untargeted metabolomics. *Nat Methods* 13(9):770–776. <https://doi.org/10.1038/nmeth.3940>
35. Barupal DK, Haldya PK, Wohlgemuth G, Kind T, Kothari SL, Pinkerton KE, Fiehn O (2012) MetaMapp: mapping and visualizing metabolomic data by integrating information from biochemical pathways and chemical and mass spectral similarity. *BMC Bioinformatics* 13:99. <https://doi.org/10.1186/1471-2105-13-99>



Chapter 20

Application of Metabolomics to Renal and Cardiometabolic Diseases

Casey M. Rebholz and Eugene P. Rhee

Abstract

Metabolomics has been increasingly applied to study renal and related cardiometabolic diseases, including diabetes and cardiovascular diseases. These studies span cross-sectional studies correlating metabolites with specific phenotypes, longitudinal studies to identify metabolite predictors of future disease, and physiologic/interventional studies to probe underlying causal relationships. This chapter provides a description of how metabolomic profiling is being used in these contexts, with an emphasis on study design considerations as a practical guide for investigators who are new to this area. Research in kidney diseases is underlined to illustrate key principles. The chapter concludes by discussing the future potential of metabolomics in the study of renal and cardiometabolic diseases.

Key words Metabolomics, Study design, Cardiometabolic disease, Renal disease, Cardiovascular disease, Biomarker, Replication

1 Introduction

1.1 Current Use of Cardiometabolic and Renal Disease Biomarkers in Research and Practice

Biomarkers of cardiometabolic disease, including kidney disease, are already being widely used in both research and clinical settings. There are many biomarkers that can help with diagnosing and monitoring disease progression, stratifying the population based on risk of developing a clinical outcome, and serving as surrogate markers of hard clinical outcomes in clinical trials. Some key biomarkers are peptides or proteins, such as brain natriuretic peptide and cardiac troponin, respectively. Elevated levels of cardiac troponin indicate myocardial injury, which, in the setting of myocardial ischemia, is sufficient to diagnose myocardial infarction [1] and in the appropriate clinical context trigger acute treatment. Other diagnostic biomarkers are metabolites, such as glucose in diabetes or uric acid with gout. Blood levels of cholesterol and triglycerides are used to diagnose dyslipidemias, but can also be used along with other clinical and lifestyle factors to estimate risk of cardiovascular disease, originally with the Framingham risk equation and more

recently with the American Heart Association/American College of Cardiology Pooled Cohort Risk Equation, and guide decisions regarding whether or not to treat with lipid-lowering therapy [2, 3]. For kidney disease, blood levels of cystatin C (a peptide) and creatinine (a metabolite), as well as urine levels of protein (albuminuria) are the basis for diagnosis and staging [4, 5]. Furthermore, change in estimated glomerular filtration rate (eGFR), which is calculated with blood levels of creatinine, has been adopted as surrogate endpoint for research studies [6].

Despite their utility in various settings, current biomarkers also have important limitations. This is well illustrated with existing renal biomarkers. Creatinine, the main endogenous marker of kidney function—or more specifically, kidney filtration (“glomerular filtration rate”, GFR)—is confounded by muscle mass, dietary intake of protein, and tubular secretion [7]. Cystatin C, a more recently characterized kidney filtration marker, is influenced by other non-GFR determinants including inflammation [8]. Blood elevations of creatinine and cystatin C, although indicative of kidney function, are not pathologic; that is, they do not make a causal contribution to disease pathogenesis, in contrast to elevated glucose levels in diabetes or elevated uric acid levels in gout. Although there is considerable evidence from both clinical trials and observational studies for change in albuminuria as a surrogate outcome for chronic kidney disease progression, and albuminuria may be both a marker and mediator of disease progression, its use is limited in part due to high within person variability thereby hampering the ability to differentiate between clinically significant change and random error [9–12]. Finally, creatinine, cystatin C, and urine albumin all share the limitation of nonspecificity—they do not provide strong insight into the underlying cause of kidney disease. Common conditions such as diabetes and hypertension, as well as less common conditions such as autoimmune disease, allergic drug reaction, or inherited structural anomalies can all cause kidney disease, and alterations in these biomarkers are unable to differentiate specific etiologies.

1.2 Opportunity for Metabolomics to Improve Biomarker Research

There is a substantial need for the discovery of new biomarkers of renal and other cardiometabolic diseases to overcome existing limitations. As outlined in more detail elsewhere in this book, current metabolomic platforms can detect >1000 known metabolites and often many more unknown ion features, many of which will be characterized and added to the growing list of known metabolites over time [13–15]. Alterations in metabolites could yield new insights into disrupted metabolism characteristic of prevalent disease states as well as future disease prognosis, and ideally, provide mechanistic insight and outline new treatment targets as well. Further, by integrating both endogenous metabolism and exogenous inputs, such as diet, medications, and the microbiome,

metabolite alterations have the potential to capture functional mediators of disease that are not assessed by other molecular domains.

2 How to Apply Metabolomics to Clinical Questions

In the process of designing and implementing a metabolomics study to address a clinical question, it is essential for the investigator to take certain factors into consideration ranging from study design and analytic considerations to clinical characteristics of study participants and details about biospecimen collection and storage. New studies on the metabolomics of cardiometabolic diseases should at least meet the standards set by previously conducted research of this kind and ideally would expand upon the existing research in terms of rigor and validity.

2.1 Designing a Population or Clinical Study

Different study designs have their own strengths and weaknesses, and their suitability depends on the study question of interest as well as practical constraints such as sample size and available resources. *Cross-sectional*, *cohort*, and *case-control* studies are types of epidemiologic, observational study designs that allow for the description of free-living populations with respect to their exposures (including metabolites) and disease outcomes.

Cross-sectional metabolomics studies assess the metabolome and the outcome of interest at the same point in time, allowing for the metabolomic characterization of a prevalent disease state. Cross-sectional studies can compare metabolite levels in individuals with or without a disease—for example, coronary disease, type 2 diabetes, CKD—or can test the association (or correlation) of metabolite levels with related continuous traits. For example, one study measured ~500 blood metabolites in 1735 participants of the KORA F4 study, followed by replication in 1164 individuals in the TwinsUK registry. This study demonstrated the substantial impact of kidney function—in KORA F4, 112 metabolites were significantly associated with eGFR, 54 of which replicated in TwinsUK [16]. For select metabolites most strongly correlated with eGFR, the authors then examined their association with measured GFR among 200 participants of the AASK study, highlighting c-mannosyltryptophan and pseudouridine as alternative or complementary markers of kidney function. Cross-sectional studies have also identified numerous blood metabolites that correlate with other cardiometabolic traits such as body mass index and insulin sensitivity, including branched chain amino acids and lipids enriched for saturated fatty acids [17, 18]. Thus, cross-sectional studies have been useful in highlighting metabolite perturbations that coincide with abnormal pathophysiologic conditions. Although these studies have the potential to identify novel disease

biomarkers and risk factors, the fact that metabolites and outcomes are being assessed at the same time precludes conclusions about the direction of causality.

Unlike cross-sectional analyses, observational studies allow for the establishment of temporality, whereby metabolomic profiling can be conducted at a baseline time point and the outcome events occur subsequently such that the exposure (metabolite alteration) precedes disease onset. These types of studies are useful for identifying early (preclinical) markers of disease risk that could be used to risk stratify the population, and potentially to identify causal disease pathways and therapeutic targets. Observational studies can have different designs, including *cohort* and *case-control* studies. A cohort study considers all subjects available in a given study population, some of whom have the outcome of interest (cases). Cohort studies have the advantage of having a well-characterized underlying study population and allow for multiple study questions to be addressed. For example, metabolomics data generated at a baseline time point in a large cohort can be tested for association with any longitudinal outcome that has been rigorously assessed over time, such as new onset kidney disease, diabetes, or cardiovascular disease. For metabolites that emerge as potential risk markers, normal reference ranges can be determined across all individuals in the cohort and the correlation with various demographic factors and comorbidities can be explored. Extensive profiling of a given cohort also permits other avenues of discovery, for example conducting a genome wide association study of metabolite levels if genotyping data has also been generated [19], and ultimately even pursuing Mendelian randomization analyses [20].

Assaying all samples in a given cohort study is not always feasible, considering the cost of metabolomic profiling and the value of nonrenewable biological specimens, and alternatives are required to answer specific questions of interest more efficiently. In a case-control study, one compares individuals who have the outcome of interest (cases) with individuals who do not but are otherwise similar (controls). Often, cases and controls are selected from a larger cohort, or in other words are “nested” within the cohort study. For example, one study utilized a nested case-control study design within the Chronic Renal Insufficiency Cohort, consisting of 200 cases with rapid progression of kidney disease (defined as GFR loss >3 ml/min per year) and 200 controls with stable kidney function over time who were matched to cases on baseline eGFR and proteinuria [21]—this subset of 400 individuals was selected from the nearly ~4000 recruited into the parent cohort. This study identified a panel of three metabolites (arginine, methionine, and threonine) for which higher levels were strongly and independently associated with rapid kidney disease progression, were not associated with baseline eGFR, and might serve as markers of renal metabolic capacity.

Selection bias is an important limitation that can arise when designing a case-control study, as it can be challenging to assemble an appropriate control group. Case-control studies often are more appropriate for highlighting associations for wider validation and characterization. Indeed, the nested case-control study outlined above serves as a springboard for consortial studies that seek to identify metabolomic predictors of CKD progression across many thousands of individuals within the context of the Chronic Kidney Disease Biomarkers Consortium (<https://www.ckdbiomarkersconsortium.org/>).

As already noted, the main advantage of a case-control design is the ability to enrich for cases, which may represent a small fraction of the overall cohort, and the ability to constrain overall sample number through the identification of otherwise similar controls (typically ~1–3 fold the number of selected cases). A third kind of observational study design, a *case-cohort* study, allows for preservation of the underlying cohort through random selection of the subcohort, rather than trying to find “matching” controls that closely resemble cases, while making sure to enrich for cases with the outcome of interest. In other words, cases are selected as in a case-control study, but are ultimately compared to a random sampling of the overall cohort rather than matched controls. In either case, conducting metabolomics research using a case-control or case-cohort study limits the use of the data to analyses of the disease status of the case group or to subsetting the data for analyses within the subcohort.

Physiologic or *interventional* studies involve the purposeful manipulation of a parameter, such as medication or diet, or performing some other manipulation of the study subject. These types of studies provide a controlled setting in which to investigate the metabolomic effects of the intervention, that is, to probe cause and effect in a way that is not possible with observational studies. In part, this advantage derives from the temporal relationship between intervention and change in the metabolome. In addition, these studies sometimes permit use of each participant as his or her own biologic control, comparing the metabolome pre- and postintervention, thus reducing the concerns about confounding introduced by comparisons across different individuals.

One subtype of a physiologic or interventional study is a metabolic challenge study. For example, the human metabolome (HuMet) study was designed to assess the dynamics of the human metabolome in 15 healthy, young men in response to different metabolic challenges: 36-h fasting period, standard liquid diet, oral glucose tolerance test, physical activity test, oral lipid tolerance test, and cold stress test [22]. The investigators collected a variety of biospecimens including plasma, urine, breath gas, and exhaled breath condensate at multiple time points throughout each intervention and conducted metabolomic profiling on the

biospecimens. They were able to demonstrate expected changes in known metabolites (e.g., low levels of insulin and glucose during fasting and an increase in their plasma levels 30 min after the oral glucose tolerance test). The repeated measures of the metabolome is a notable strength of this study, which, in the context of a metabolic challenge allows one to attribute change in the metabolome to the intervention as well as to determine the kinetics and stability of these changes.

Feeding studies and other types of diet intervention studies are a valuable design for understanding the effect of diet on cardiometabolic outcomes. The food metabolome reflects metabolites that are derived from food itself as well as those altered through the metabolism of food by tissues and the microbiota [23]. Characterizing the metabolomics of dietary intake, especially within feeding studies, allows for the identification of biomarkers of food as well as insights about diet-modifiable pathways leading to disease outcomes [24]. For example, one group conducted a controlled cross-over feeding trial in which 19 healthy volunteers received four dietary interventions that varied in level of adherence to the World Health Organization healthy eating guidelines for 3 days each in a randomized order [25]. Metabolomic profiling was conducted on biospecimens that were pooled from urine collected at three time points each day. The investigators were able to develop diet-specific urinary metabolite models which were then validated in two independent study populations, that is, the INTERMAP UK cohort ($n = 225$) and a healthy eating Danish cohort ($n = 66$). These results provide proof-of-concept for metabolomic phenotyping of dietary intake, an objective approach that would vastly improve upon currently used tools for assessing diet such as patient recall and daily food diaries.

Another important type of intervention study is one that evaluates the effect of a medication on metabolites pertinent to cardiometabolic and renal outcomes. Of course, these kinds of studies have already been conducted for individual metabolites such as glucose, uric acid, and cholesterol, which are known to be causal disease mediators. In theory, unbiased metabolomic approaches could be used to provide a more nuanced understanding of metabolic pathways that are affected by pharmacologic agents, particularly if the mechanism of action is unknown or incompletely elucidated. Once the metabolomic response to treatment has been characterized, metabolomics could then be used to distinguish between individuals who would be expected to respond to treatment and individuals who would not be expected to respond to treatment. Ultimately, conducting metabolomics in treatment trials could inform a personalized medicine strategy of providing beneficial treatment to responders and avoiding cost and harmful effects by not prescribing such therapy to nonresponders [26]. These are important avenues of future investigation in the field.

Finally, in addition to a focused examination of how dietary or pharmacologic inputs impact the metabolome, physiologic study designs can seek to understand organ specific metabolite production and handling. For example, one study profiled individuals who underwent “planned” myocardial infarctions, that is, patients with hypertrophic obstructive cardiomyopathy undergoing alcohol induced ablation of their septal coronary arteries—this is an elective surgical procedure that intentionally infarcts a portion of septal muscle in patients whose thickened heart chambers obstruct optimal flow of blood out of the heart [27]. In addition to enabling precisely timed blood sampling, this study utilized invasive catheterization of the coronary sinus, a venous channel that receives blood from all of the coronary veins and empties into the right atrium, permitting direct assessment of metabolite release from the ischemic heart. In another example leveraging invasive catheterization, metabolites were profiled in plasma obtained from the aorta and renal vein, permitting direct assessment of the heterogeneous impact of kidney function across the metabolome [28]. As expected, this analysis showed that many metabolite levels are substantially lower in the renal vein than the aorta because of renal clearance; however, there was substantial heterogeneity in the magnitude of decrease, attributable to the different chemical characteristics of circulating metabolites (size, polarity, protein binding, etc.) as well as the different mechanisms by which the kidneys handle metabolites, including filtration, reabsorption, secretion, and metabolism.

2.2 Statistics and the Challenge of Multiple Testing

Multiple testing is a fundamental challenge in metabolomics research. The large number of compounds detectable using currently available metabolomic platforms is an important strength, permitting an expansive and unbiased discovery approach, with the opportunity not just to identify single metabolites of interest but to also highlight select metabolic pathways. However, the large number of metabolites assayed and tested statistically also raises important analytical challenges, specifically an increase in the type one error rate, the detection of false positive findings. There are a variety of analytic approaches that can be used to account for multiple testing, including adjusting the threshold for determining statistical significance. One widely used approach is a Bonferroni correction, where statistical significance is adjusted for the number of metabolites measured ($0.05/n$), and if relevant the number of outcomes examined as well. This can be overly conservative for study designs that include both discovery analysis and independent replication, particularly given the significant intercorrelation between metabolite measurements and the ability to evaluate metabolite alterations in the context of biochemical pathways. A widely used alternative to Bonferroni correction is the Benjamini–Hochberg procedure (false discovery rate) [29, 30]. Some

investigators use data reduction techniques to reduce the number of statistical tests, for example, by limiting the analysis to only known metabolites (and not unknown metabolites or ion features). An additional approach is to restrict analysis to metabolites with relatively low intra-person, day to day variability, on the premise that highly variable metabolites are unlikely to serve as reliable disease biomarkers in the clinic. In one study of diabetic kidney disease and progression to end stage kidney disease, the investigators only considered metabolites that had a correlation coefficient ≥ 0.4 on paired, fasting samples drawn several years apart, narrowing the number of metabolites considered from 262 to 119 [31].

Traditional approaches to examining the association between metabolites and outcomes examine each metabolite individually. Given the high-dimensional nature of metabolomics data as well as the intercorrelation among metabolites, more systems-based approaches (covered in more detail elsewhere in the textbook) that consider metabolites in aggregate and in the context of biochemical pathways may be more appropriate depending on the study of interest. Analytical approaches need to be aligned with the overarching goal of the study. If the goal is to identify and develop a clinically useful biomarker, then traditional regression methods that model each metabolite individually and apply the most stringent significance thresholds may be most appropriate. If the goal is to use metabolomics data primarily for biologic insight and as a springboard for laboratory-based interrogation, then systems-based approaches with more permissive significance thresholds may be most appropriate (and the full breadth of data, including metabolites that have high intra-person variability, should be analyzed). In either case, the concern for false positive discovery can be ameliorated by increasing statistical power through increased sample size, an option that will become more tractable as the throughput and costs of metabolomics platforms continue to improve. In clinical research of cardiometabolic and renal disease, fortunately, there are large samples of individuals available through medical registries and cohort studies who are available for metabolomic profiling (although primarily for European and Caucasian study populations). Enhancing sample size may remain a challenge for less common diseases or for non-Caucasian study populations.

2.3 Controlling for Confounding Factors

The measurement and adjustment for potential confounding factors is a critical consideration in metabolomics studies of renal and cardiometabolic disease. Body mass index and diabetes status have an enormous influence on the metabolome, in large part because insulin action and sensitivity modulates blood levels of sugars, amino acids, and lipids [32, 33]. Kidney function also has an enormous influence on the metabolome, as highlighted by both observational and physiologic studies described above. Age,

gender, dietary habits, and other demographic factors also impact the metabolome [34, 35]. Careful consideration of these factors is essential in both study design, to ensure that they are catalogued and measured, as well as in analysis, where models may need to be adjusted to account for these potential confounders.

A discussion of confounding in kidney disease illustrates some key challenges. As noted, the kidneys significantly impact the metabolome, particularly small, polar molecules which are normally excreted in urine. However, inverse correlation with GFR does not necessarily indicate that a metabolite undergoes glomerular filtration, as the kidneys also contribute to metabolite clearance through reabsorption, secretion, and metabolism. In addition, loss of kidney function can cause alterations in diet, gut microbiota, insulin sensitivity, and other parameters that may impact the metabolome [36]. Finally, a metabolite in theory could have an inverse relationship with GFR if it contributes directly to the pathogenesis of kidney disease, that is, it causes kidney damage. Hypothetically, statistical approaches should vary depending on the questions of interest. For studies that seek to identify novel causal mediators of kidney disease, the association between metabolites and outcomes (e.g., progression of kidney disease) should not necessarily be adjusted for clinical measures of kidney function, as this could attenuate or abrogate potentially meaningful signals. In practice, however, almost all metabolomics studies of kidney disease to date have adjusted for clinical measures of kidney function. In part, this is because so many metabolites end up reaching statistical significance without adjusting for kidney function, making it difficult to prioritize select findings. In addition, this reflects the fact that most investigators in the field have a background in clinical science and epidemiology, with a focus on the discovery of clinically useful biomarkers, which mandates rigorous adjustment to identify signals with predictive power beyond GFR.

When adjusting for kidney function, the standard in the field is to adjust for GFR and proteinuria. Multiple studies have been conducted to examine which metabolites are cross-sectionally associated with both measured GFR and estimated GFR [16, 37–40]. There is comparatively less evidence examining the cross-sectional association between blood metabolites and proteinuria. One study has examined the cross-sectional association between the metabolome and proteinuria as assessed using 24-h urine collections in the African American Study of Kidney Disease and Hypertension and the Modification of Diet in Renal Disease study [41] and found that 58 metabolites (~9% of measured compounds) were significantly associated with proteinuria after accounting for demographic characteristics, serum albumin, mGFR, and other clinical risk factors.

Medications are another major determinant of the metabolome. Kidney disease patients as well as participants in other clinical studies of preexisting cardiometabolic diseases are often treated with medications for comorbid conditions such as hypertension, diabetes, and dyslipidemia. However, self-report of medication use is suboptimal, and there is variability in the type of medications (even within specific classes of drugs) and degree to which patients adhere to their prescribed medications. Metabolomics has the potential to identify markers of drug use, including the drug itself or one of its catabolites, but this has been a relatively underutilized application in studies to date [42].

For all metabolomics studies, diet needs to be considered as a potential confounder. As already noted, the metabolome is impacted by dietary intake, reflecting breakdown products of foods that are consumed, compounds that are metabolized in the body by tissue and microbiota, and secondary effects of the hormonal responses triggered by caloric input [23, 24]. When the nutritional metabolome is not of interest to the study question but rather considered to be a confounding factor between the metabolome and kidney disease status or risk, the investigator needs to adjust for the fasting versus fed state. Study participants will vary in terms of time since their last meal and with respect to the composition of their diet. It may suffice to measure and adjust for time since last meal. As an additional measure to account for variability in food composition, one could adjust for an overall diet quality score (e.g., the Healthy Eating Index) [43].

2.4 Specimen Type, Storage, and Handling

The nature of samples used for metabolomics analysis requires careful consideration. For large cohort studies that have completed enrollment, samples will have already been collected with corresponding demographic and clinical data. Access to these samples often requires direct collaboration with investigators overseeing the cohort studies, or in some cases, can be obtained through public channels (e.g., the National Institute of Diabetes and Digestive and Kidney Diseases Central Repository or the National Heart, Lung, and Blood Institute Biologic Specimen and Data Repository Information Coordinating Center). In addition to large observational cohorts, biospecimens are also often collected for randomized clinical trials of different pharmacologic or lifestyle interventions, although access to these specimens may be restricted in use. Finally, prospective collection by the investigator or a group of collaborators is an attractive approach to sample accrual when appropriate, for example for physiologic or interventional studies or when rare or unusual study populations are required.

Whether samples have already been collected or will be collected prospectively, it is important to recognize that procedures for sample collection, processing, and storage can all impact the levels of metabolites in a given biological specimen. First, all sampling

approaches, along with the subsequent generation of metabolomics data and integration with clinical data, need to be approved by the relevant regulatory board for human research. *Fasting status* is critical to determine for all samples. *Timing of sample collection*, independent of feeding state, is also important as the role of circadian oscillation on the metabolome is increasingly recognized [44]. A few small studies have directly compared results between *serum* and *plasma*, suggesting overall concordance in results, although serum appears to have higher levels of some metabolites including some reflective of the clotting process [45–48]. For plasma, different *anticoagulants* may have different effects. For example, citrate will clearly impact the measurement of citrate levels and may interfere with the measurement of isomers, whereas heparin may impact lipids based on its ability to inhibit lipoprotein lipase [49]. Samples that are shipped, stored for an extended period of time, or sub aliquoted may undergo *freeze-thaw cycles* which could alter the composition of the metabolome. Previous studies have reported on the stability of the metabolome at different storage temperatures and after undergoing multiple freeze-thaw cycles [50, 51]. For example, one study demonstrated stability of metabolite concentrations over four freeze-thaw cycles, but degradation of samples was apparent when they were left at room temperature for 12 h or longer before processing and storing [52]. Thus, investigators should be aware of exactly how samples for their study were collected and processed, and the duration of time intervals between phlebotomy and storage at –80 °C. Stability of course also varies depending on the metabolite, with amino acids and neutral lipids demonstrating particular robustness across various sample handling and storage conditions, whereas molecules with high energy phosphate bonds would be expected to dissipate more readily.

In theory, a range of biospecimens are amenable to metabolomic profiling, including blood (plasma or serum), urine, saliva, cerebral spinal fluid, stool, and even tissue homogenates. All have been examined to date, but in practice blood and urine have been studied the most extensively in renal and cardiometabolic disorders. As such, a brief discussion of urine metabolomics is warranted. One challenge with urinary analyses relates to the wide range of dilution, resulting in metabolite levels that may fall above or below the linear dynamic range of detection. To account for between-person variability in dilution, metabolite levels can be indexed to urine creatinine. Alternatively, urine collected over a 24-h period may be more informative than spot urine which is collected at a random point in time (during a clinical or study visit) since 24-h urine specimens provide a cumulative measure of the metabolome which varies diurnally. The choice of blood versus urine may depend on the pathophysiologic aspect of kidney disease of interest. For example, urine may be a more suitable medium for studying tubular secretion or reabsorption [53]. The simultaneous measure of urine and

blood could provide unique insights about the transport of compounds across kidney cells, that is, fractional excretion which is the fraction of the metabolite filtered by the kidney that is ultimately excreted, providing insight on metabolite absorption and secretion [53]. However, the current availability of studies with metabolomics results in both urine and blood is limited.

2.5 Replication of Discovery Findings

Currently available metabolomic platforms, particularly untargeted approaches, allow for the identification of many known metabolites and many more unknown compounds or peaks/features. Following such discovery work, it is necessary to replicate findings to provide more confidence in their validity. One technique for validation is to randomly subset the data (e.g., two-thirds and one-third samples) using one subset for discovery and the second subset for validation. The difficulty with this technique is that one is reducing statistical power by analyzing small datasets. If feasible, a preferred alternative is to use an independent, external study population to replicate the initial discovery findings. However, with an external replication, lack of consistent results could be due to differences between the discovery and replication cohorts. An additional challenge for replication across studies is nonoverlap of metabolomic platforms with respect to hardware, procedures, quality control methods, and metabolite identification, thereby precluding the ability to systematically meta-analyze data generated across disparate research groups. The Chronic Kidney Disease Biomarkers Consortium recently conducted a methodological study in which the same specimens were run on two nontargeted platforms [14], yielding several observations. First, there was strong overlap of known metabolites across platforms, particularly among abundant metabolites such as amino acids and lipids, with generally better coefficients of variation (CVs) across technical replicates for these than the other known metabolites. Second, the exercise of using different platforms on the same sample set can lead to the expansion of known metabolite coverage for both platforms. Third, agreement for the majority of overlapping measurements across platforms was strong, suggesting that meta-analysis across platforms is feasible, although there is a relative paucity of these kinds of studies to date, in stark contrast to genomics and genome wide association studies. Despite the areas of overlap, however, this study also reinforced a general challenge in metabolomics, that no single analytic approach is truly comprehensive and that careful understanding of the methods utilized is crucial. For kidney and cardiovascular disease biomarker research, the simultaneous deployment of different methodologies across different cohorts may be prudent, to cast the widest net for discovery, and to minimize the risk inherent in selecting one analytical approach.

2.6 Data Analysis and Computational Cost of Metabolomics Research

As metabolomics methods develop further, the analytic cost of measuring the metabolome decreases, but the computational cost of big data will remain, especially as the ability to run larger batches and identify more metabolites increases. Key advances that have allowed for more efficient measurement of the metabolome include the use of robotics, automated data processing software for metabolite identification and quantification, and improvement in quality control procedures [54]. As such, it is now possible to conduct metabolomics in large-scale epidemiologic studies, yielding an increasingly large amount of data in terms of sample size and identified metabolites.

The most productive metabolomics projects providing meaningful and translatable findings will be led by a team of scientists, each offering their own expertise in basic science, physiology, clinical medicine, bioinformatics, statistics, and epidemiology. This type of team science requires the support by home institutions, to recognize the need of infrastructure and career development of young scientists. There is a need to train early career researchers to be able to meaningfully analyze metabolomics data and to appreciate both the physiological significance of detectable metabolites as well as how to effectively use statistical techniques. Chapters 14–16 in this book provide a foundation of data science and statistics. Other chapters give more specific treatment of topics in metabolomics research. Readers are also encouraged to visit the supplemental website (<https://metabolomics-data.github.io>) for more training materials.

3 Summary

In summary, the potential of metabolomics research of renal and cardiometabolic diseases is great, with substantial progress made to date. The majority of past studies have been observational in nature, either cross-sectional or observational cohort studies. However, the identification of novel clinical biomarkers, elucidation of new biology, and identification of actionable therapeutic targets will require larger and more diverse studies permitting rigorous validation and meta-analysis, as well as more physiologic and interventional study designs (e.g., to understand how diet and medications alter the relevant metabolic pathways and how these responses impact disease risk and to assign organ specificity to metabolomic alterations). When applying metabolomics to study questions in kidney disease and related cardiometabolic diseases, it is important to consider the impact of conducting many statistical tests on statistical power, to control for confounding factors, to understand details of specimen collection and handling, and to select a specimen type best suited for the study question. Given the broad range

of considerations, the application of metabolomics to clinical disease is best conducted by a multidisciplinary team with sufficient technical, computational, and content-related expertise.

References

- Thygesen K, Alpert JS, Jaffe AS, Chaitman BR, Bax JJ, Morrow DA, White HD, Executive Group on behalf of the Joint European Society of Cardiology/American College of Cardiology/American Heart Association/World Heart Federation Task Force for the Universal Definition of Myocardial I (2018) Fourth universal definition of myocardial infarction (2018). *Glob Heart* 13(4):305–338. <https://doi.org/10.1016/j.ghart.2018.08.004>
- Anderson KM, Odell PM, Wilson PW, Kannel WB (1991) Cardiovascular disease risk profiles. *Am Heart J* 121(1 Pt 2):293–298
- Goff DC Jr, Lloyd-Jones DM, Bennett G, Coady S, D'Agostino RB, Gibbons R, Greenland P, Lackland DT, Levy D, O'Donnell CJ, Robinson JG, Schwartz JS, Shero ST, Smith SC Jr, Sorlie P, Stone NJ, Wilson PW, Jordan HS, Nevo L, Wnek J, Anderson JL, Halperin JL, Albert NM, Bozkurt B, Brindis RG, Curtis LH, De Mets D, Hochman JS, Kovacs RJ, Ohman EM, Pressler SJ, Sellke FW, Shen WK, Smith SC Jr, Tomaselli GF, American College of Cardiology/American Heart Association Task Force on Practice G (2014) 2013 ACC/AHA guideline on the assessment of cardiovascular risk: a report of the American College of Cardiology/American Heart Association task force on practice guidelines. *Circulation* 129(25 Suppl 2):S49–S73. <https://doi.org/10.1161/01.cir.0000437741.48606.98>
- Kidney Disease: Improving Global Outcomes (KDIGO) (2013) KDIGO clinical practice guideline for the evaluation and management of chronic kidney disease. *Kidney Int* 3 (1):1–150
- Inker LA, Schmid CH, Tighiouart H, Eckfeldt JH, Feldman HI, Greene T, Kusek JW, Manzi J, Van Lente F, Zhang YL, Coresh J, Levey AS, Chronic Kidney Disease Epidemiology Collaboration Investigators (2012) Estimating glomerular filtration rate from serum creatinine and cystatin C. *N Engl J Med* 367 (1):20–29. <https://doi.org/10.1056/NEJMoa1114248>
- Levey AS, Inker LA, Matsushita K, Greene T, Willis K, Lewis E, De Zeeuw D, Cheung AK, Coresh J (2014) GFR decline as an endpoint for clinical trials in CKD: a scientific workshop sponsored by the National Kidney Foundation and the US Food and Drug Administration. *Am J Kidney Dis* 64(6):821–835
- Stevens LA, Coresh J, Greene T, Levey AS (2006) Assessing kidney function--measured and estimated glomerular filtration rate. *N Engl J Med* 354(23):2473–2483. <https://doi.org/10.1056/NEJMra054415>
- Stevens LA, Schmid CH, Greene T, Li L, Beck GJ, Joffe MM, Froissart M, Kusek JW, Zhang YL, Coresh J, Levey AS (2009) Factors other than glomerular filtration rate affect serum cystatin C levels. *Kidney Int* 75(6):652–660. <https://doi.org/10.1038/ki.2008.638>
- HJL H, Greene T, Tighiouart H, Gansevoort RT, Coresh J, Simon AL, Chan TM, Hou FF, Lewis JB, Locatelli F, Praga M, Schena FP, Levey AS, Inker LA, Chronic Kidney Disease Epidemiology C (2019) Change in albuminuria as a surrogate endpoint for progression of kidney disease: a meta-analysis of treatment effects in randomised clinical trials. *Lancet Diabetes Endocrinol* 7(2):128–139. [https://doi.org/10.1016/S2213-8587\(18\)30314-0](https://doi.org/10.1016/S2213-8587(18)30314-0)
- Coresh J, Heerspink HJL, Sang Y, Matsushita K, Arnlov J, Astor BC, Black C, Brunskill NJ, Carrero JJ, Feldman HI, Fox CS, Inker LA, Ishani A, Ito S, Jassal S, Konta T, Polkinghorne K, Romundstad S, Solbu MD, Stempniewicz N, Stengel B, Tonelli M, Umesawa M, Waikar SS, Wen CP, Wetzel JFM, Woodward M, Grams ME, Kovesdy CP, Levey AS, Gansevoort RT, Chronic Kidney Disease Prognosis C, Chronic Kidney Disease Epidemiology C (2019) Change in albuminuria and subsequent risk of end-stage kidney disease: an individual participant-level consortium meta-analysis of observational studies. *Lancet Diabetes Endocrinol* 7(2):115–127. [https://doi.org/10.1016/S2213-8587\(18\)30313-9](https://doi.org/10.1016/S2213-8587(18)30313-9)
- Inker LA, Levey AS, Pandya K, Stoycheff N, Okparavero A, Greene T, Chronic Kidney Disease Epidemiology C (2014) Early change in proteinuria as a surrogate end point for kidney disease progression: an individual patient meta-analysis. *Am J Kidney Dis* 64(1):74–85. <https://doi.org/10.1053/j.ajkd.2014.02.020>
- Waikar SS, Rebholz CM, Zheng Z, Hurwitz S, Hsu CY, Feldman HI, Xie D, Liu KD, Mifflin TE, Eckfeldt JH, Kimmel PL, Vasan RS, Bonventre JV, Inker LA, Coresh J, Chronic Kidney

- Disease Biomarkers Consortium I (2018) Biological variability of estimated GFR and albuminuria in CKD. *Am J Kidney Dis* 72(4):538–546. <https://doi.org/10.1053/j.ajkd.2018.04.023>
13. Schrimpe-Rutledge AC, Codreanu SG, Sherrod SD, McLean JA (2016) Untargeted metabolomics strategies-challenges and emerging directions. *J Am Soc Mass Spectrom* 27(12):1897–1905. <https://doi.org/10.1007/s13361-016-1469-y>
 14. Rhee EP, Waikar SS, Rebholz CM, Zheng Z, Perichon R, Clish CB, Evans AM, Avila J, Denburg MR, Anderson AH, Vasan RS, Feldman HI, Kimmel PL, Coresh J, Consortium CKDB (2019) Variability of two metabolomic platforms in CKD. *Clin J Am Soc Nephrol* 14(1):40–48. <https://doi.org/10.2215/CJN.07070618>
 15. Mahieu NG, Patti GJ (2017) Systems-level annotation of a metabolomics data set reduces 25000 features to fewer than 1000 unique metabolites. *Anal Chem* 89(19):10397–10406. <https://doi.org/10.1021/acs.analchem.7b02380>
 16. Sekula P, Goek ON, Quaye L, Barrios C, Levey AS, Romisch-Margl W, Menni C, Yet I, Gieger C, Inker LA, Adamski J, Gronwald W, Illig T, Dettmer K, Krumsiek J, Oefner PJ, Valdes AM, Meisinger C, Coresh J, Spector TD, Mohney RP, Suhre K, Kastenmuller G, Kottgen A (2016) A metabolome-wide association study of kidney function and disease in the general population. *J Am Soc Nephrol* 27(4):1175–1188. <https://doi.org/10.1681/ASN.2014111099>
 17. Ho JE, Larson MG, Ghorbani A, Cheng S, Chen MH, Keyes M, Rhee EP, Clish CB, Vasan RS, Gerszten RE, Wang TJ (2016) Metabolomic profiles of body mass index in the Framingham heart study reveal distinct Cardiometabolic phenotypes. *PLoS One* 11(2):e0148361. <https://doi.org/10.1371/journal.pone.0148361>
 18. Newgard CB, An J, Bain JR, Muehlbauer MJ, Stevens RD, Lien LF, Haqq AM, Shah SH, Arlotto M, Slentz CA, Rochon J, Gallup D, Ilkayeva O, Wenner BR, Yancy WS Jr, Eisenson H, Musante G, Surwit RS, Millington DS, Butler MD, Svetkey LP (2009) A branched-chain amino acid-related metabolic signature that differentiates obese and lean humans and contributes to insulin resistance. *Cell Metab* 9(4):311–326. <https://doi.org/10.1016/j.cmet.2009.02.002>
 19. Kastenmuller G, Raffler J, Gieger C, Suhre K (2015) Genetics of human metabolism: an update. *Hum Mol Genet* 24(R1):R93–R101. <https://doi.org/10.1093/hmg/ddv263>
 20. Sekula P, Del Greco MF, Pattaro C, Kottgen A (2016) Mendelian randomization as an approach to assess causality using observational data. *J Am Soc Nephrol* 27(11):3253–3265. <https://doi.org/10.1681/ASN.2016010098>
 21. Rhee EP, Clish CB, Wenger J, Roy J, Elmariah S, Pierce KA, Bullock K, Anderson AH, Gerszten RE, Feldman HI (2016) Metabolomics of chronic kidney disease progression: a case-control analysis in the chronic renal insufficiency cohort study. *Am J Nephrol* 43(5):366–374. <https://doi.org/10.1159/000446484>
 22. Krug S, Kastenmuller G, Stuckler F, Rist MJ, Skurk T, Sailer M, Raffler J, Romisch-Margl W, Adamski J, Prehn C, Frank T, Engel KH, Hofmann T, Luy B, Zimmermann R, Moritz F, Schmitt-Kopplin P, Krumsiek J, Kremer W, Huber F, Oeh U, Theis FJ, Szymczak W, Hauner H, Suhre K, Daniel H (2012) The dynamic range of the human metabolome revealed by challenges. *FASEB J* 26(6):2607–2619. <https://doi.org/10.1096/fj.11-198093>
 23. Scalbert A, Brennan L, Manach C, Andres-Lacueva C, Dragsted LO, Draper J, Rappaport SM, van der Hooft JJ, Wishart DS (2014) The food metabolome: a window over dietary exposure. *Am J Clin Nutr* 99(6):1286–1308. <https://doi.org/10.3945/ajcn.113.076133>
 24. Guasch-Ferre M, Bhupathiraju SN, Hu FB (2018) Use of metabolomics in improving assessment of dietary intake. *Clin Chem* 64(1):82–98. <https://doi.org/10.1373/clinchem.2017.272344>
 25. Garcia-Perez I, Posma JM, Gibson R, Chambers ES, Hansen TH, Vestergaard H, Hansen T, Beckmann M, Pedersen O, Elliott P, Stamler J, Nicholson JK, Draper J, Mathers JC, Holmes E, Frost G (2017) Objective assessment of dietary patterns by use of metabolic phenotyping: a randomised, controlled, crossover trial. *Lancet Diabetes Endocrinol* 5(3):184–195. [https://doi.org/10.1016/S2213-8587\(16\)30419-3](https://doi.org/10.1016/S2213-8587(16)30419-3)
 26. Tolstikov V (2016) Metabolomics: bridging the gap between pharmaceutical development and population health. *Metabolites* 6(3):E20. <https://doi.org/10.3390/metabo6030020>
 27. Lewis GD, Wei R, Liu E, Yang E, Shi X, Martinovic M, Farrell L, Asnani A, Cyrille M, Ramanathan A, Shaham O, Berriz G, Lowry PA, Palacios IF, Tasan M, Roth FP, Min J, Baumgartner C, Keshishian H, Addona T, Mootha VK, Rosenzweig A, Carr SA, Fifield MA, Sabatine MS, Gerszten RE (2008)

- Metabolite profiling of blood from individuals undergoing planned myocardial infarction reveals early markers of myocardial injury. *J Clin Investig* 118(10):3503–3512. <https://doi.org/10.1172/JCI35111>
28. Rhee EP, Clish CB, Ghorbani A, Larson MG, Elmariah S, McCabe E, Yang Q, Cheng S, Pierce K, Deik A, Souza AL, Farrell L, Domos C, Yeh RW, Palacios I, Rosenfield K, Vasan RS, Florez JC, Wang TJ, Fox CS, Gersztten RE (2013) A combined epidemiologic and metabolomic approach improves CKD prediction. *J Am Soc Nephrol* 24(8):1330–1338. <https://doi.org/10.1681/ASN.2012101006>
 29. Curtin F, Schulz P (1998) Multiple correlations and Bonferroni's correction. *Biol Psychiatry* 44(8):775–777
 30. Benjamini Y, Hochberg Y (1995) Controlling the false discovery rate - a practical and powerful approach to multiple testing. *J R Stat Soc B* 57(1):289–300
 31. Niewczas MA, Sirich TL, Mathew AV, Skupien J, Mohney RP, Warram JH, Smiles A, Huang X, Walker W, Byun J, Karoly ED, Kensicki EM, Berry GT, Bonventre JV, Pennathur S, Meyer TW, Krolewski AS (2014) Uremic solutes and risk of end-stage renal disease in type 2 diabetes: metabolomic study. *Kidney Int* 85(5):1214–1224. <https://doi.org/10.1038/ki.2013.497>
 32. Warren B, Rehbolz CM, Sang Y, Lee AK, Coresh J, Selvin E, Grams ME (2018) Diabetes and trajectories of estimated glomerular filtration rate: a prospective cohort analysis of the atherosclerosis risk in communities study. *Diabetes Care* 41(8):1646–1653. <https://doi.org/10.2337/dc18-0277>
 33. Bell EK, Gao L, Judd S, Glasser SP, McClellan W, Gutierrez OM, Safford M, Lackland DT, Warnock DG, Muntner P (2012) Blood pressure indexes and end-stage renal disease risk in adults with chronic kidney disease. *Am J Hypertens* 25(7):789–796. <https://doi.org/10.1038/ajh.2012.48>
 34. Yin X, Subramanian S, Willinger CM, Chen G, Juhasz P, Courchesne P, Chen BH, Li X, Hwang SJ, Fox CS, O'Donnell CJ, Muntendam P, Fuster V, Bobeldijk-Pastorova I, Sookoian SC, Pirola CJ, Gordon N, Adourian A, Larson MG, Levy D (2016) Metabolite signatures of metabolic risk factors and their longitudinal changes. *J Clin Endocrinol Metab* 101(4):1779–1789. <https://doi.org/10.1210/jc.2015-2555>
 35. Bhupathiraju SN, Guasch-Ferre M, Gadgil MD, Newgard CB, Bain JR, Muehlbauer MJ, Ilkayeva OR, Scholtens DM, Hu FB, Kanaya AM, Kandula NR (2018) Dietary patterns among Asian Indians living in the United States have distinct Metabolomic profiles that are associated with Cardiometabolic risk. *J Nutr* 148(7):1150–1159. <https://doi.org/10.1093/jn/nxy074>
 36. Ramezani A, Raj DS (2014) The gut microbiome, kidney disease, and targeted interventions. *J Am Soc Nephrol* 25(4):657–670. <https://doi.org/10.1681/ASN.2013080905>
 37. Coresh J, Inker LA, Sang Y, Chen J, Shafi T, Post WS, Shlipak MG, Ford L, Goodman K, Perichon R, Greene T, Levey AS (2018) Metabolomic profiling to improve glomerular filtration rate estimation: a proof-of-concept study. *Nephrol Dial Transplant* 34(5):825–833. <https://doi.org/10.1093/ndt/gfy094>
 38. Titan SM, Venturini G, Padilha K, Tavares G, Zatz R, Bensenor I, Lotufo PA, Rhee EP, Thadhani RI, Pereira AC (2019) Metabolites related to eGFR: evaluation of candidate molecules for GFR estimation using untargeted metabolomics. *Clin Chim Acta* 489:242–248. <https://doi.org/10.1016/j.cca.2018.08.037>
 39. Goek ON, Doring A, Gieger C, Heier M, Koenig W, Prehn C, Romisch-Margl W, Wang-Sattler R, Illig T, Suhre K, Sekula P, Zhai G, Adamski J, Kottgen A, Meisinger C (2012) Serum metabolite concentrations and decreased GFR in the general population. *Am J Kidney Dis* 60(2):197–206. <https://doi.org/10.1053/j.ajkd.2012.01.014>
 40. Ng DP, Salim A, Liu Y, Zou L, Xu FG, Huang S, Leong H, Ong CN (2012) A metabolomic study of low estimated GFR in non-proteinuric type 2 diabetes mellitus. *Diabetologia* 55(2):499–508. <https://doi.org/10.1007/s00125-011-2339-6>
 41. Luo S, Coresh J, Tin A, Rehbolz CM, Appel LJ, Chen J, Vasan RS, Anderson AH, Feldman HI, Kimmel PL, Waikar SS, Kottgen A, Evans AM, Levey AS, Inker LA, Sarnak MJ, Grams ME, Chronic Kidney Disease Biomarkers Consortium I (2019) Serum Metabolomic alterations associated with proteinuria in CKD. *Clin J Am Soc Nephrol* 14(3):342–353. <https://doi.org/10.2215/CJN.10010818>
 42. Guo L, Milburn MV, Ryals JA, Lonergan SC, Mitchell MW, Wulff JE, Alexander DC, Evans AM, Bridgewater B, Miller L, Gonzalez-Garay ML, Caskey CT (2015) Plasma metabolomic profiles enhance precision medicine for volunteers of normal health. *Proc Natl Acad Sci U S A* 112(35):E4901–E4910. <https://doi.org/10.1073/pnas.1508425112>
 43. Krebs-Smith SM, Pannucci TE, Subar AF, Kirkpatrick SI, Lerman JL, Tooze JA, Wilson MM, Reedy J (2018) Update of the healthy eating index: HEI-2015. *J Acad Nutr Diet*

- 118(9):1591–1602. <https://doi.org/10.1016/j.jand.2018.05.021>
44. Grant LK, Ftouni S, Nijagal B, De Souza DP, Tull D, McConville MJ, Rajaratnam SMW, Lockley SW, Anderson C (2019) Circadian and wake-dependent changes in human plasma polar metabolites during prolonged wakefulness: a preliminary analysis. *Sci Rep* 9(1):4428. <https://doi.org/10.1038/s41598-019-40353-8>
45. Breier M, Wahl S, Prehn C, Fugmann M, Ferrari U, Weise M, Banning F, Seissler J, Grallert H, Adamski J, Lechner A (2014) Targeted metabolomics identifies reliable and stable metabolites in human serum and plasma samples. *PLoS One* 9(2):e89728. <https://doi.org/10.1371/journal.pone.0089728>
46. Wedge DC, Allwood JW, Dunn W, Vaughan AA, Simpson K, Brown M, Priest L, Blackhall FH, Whetton AD, Dive C, Goodacre R (2011) Is serum or plasma more appropriate for inter-subject comparisons in metabolomic studies? An assessment in patients with small-cell lung cancer. *Anal Chem* 83(17):6689–6697. <https://doi.org/10.1021/ac2012224>
47. Lin Z, Zhang Z, Lu H, Jin Y, Yi L, Liang Y (2014) Joint MS-based platforms for comprehensive comparison of rat plasma and serum metabolic profiling. *Biomed Chromatogr* 28(9):1235–1245. <https://doi.org/10.1002/bmc.3152>
48. Ishikawa M, Tajima Y, Murayama M, Senoo Y, Maekawa K, Saito Y (2013) Plasma and serum from nonfasting men and women differ in their lipidomic profiles. *Biol Pharm Bull* 36(4):682–685
49. Brunner MP, Shah SH, Craig DM, Stevens RD, Muehlbauer MJ, Bain JR, Newgard CB, Kraus WE, Granger CB, Sketch MH Jr, Newby LK (2011) Effect of heparin administration on metabolomic profiles in samples obtained during cardiac catheterization. *Circ Cardiovasc Genet* 4(6):695–700. <https://doi.org/10.1161/CIRCGENETICS.111.960575>
50. Gika HG, Theodoridis GA, Wilson ID (2008) Liquid chromatography and ultra-performance liquid chromatography-mass spectrometry fingerprinting of human urine: sample stability under different handling and storage conditions for metabolomics studies. *J Chromatogr A* 1189(1–2):314–322. <https://doi.org/10.1016/j.chroma.2007.10.066>
51. Teahan O, Gamble S, Holmes E, Waxman J, Nicholson JK, Bevan C, Keun HC (2006) Impact of analytical bias in metabolomic studies of human blood serum and plasma. *Anal Chem* 78(13):4307–4318. <https://doi.org/10.1021/ac051972y>
52. Anton G, Wilson R, Yu ZH, Prehn C, Zukunft S, Adamski J, Heier M, Meisinger C, Romisch-Margl W, Wang-Sattler R, Hveem K, Wolfenbuttel B, Peters A, Kastenmuller G, Waldenberger M (2015) Pre-analytical sample quality: metabolite ratios as an intrinsic marker for prolonged room temperature exposure of serum samples. *PLoS One* 10(3):e0121495. <https://doi.org/10.1371/journal.pone.0121495>
53. Kottgen A, Raffler J, Sekula P, Kastenmuller G (2018) Genome-wide association studies of metabolite concentrations (mGWAS): relevance for nephrology. *Semin Nephrol* 38(2):151–174. <https://doi.org/10.1016/j.semnephrol.2018.01.009>
54. Tzoulaki I, Ebbels TM, Valdes A, Elliott P, Ioannidis JP (2014) Design and analysis of metabolomics studies in epidemiologic research: a primer on -omic technologies. *Am J Epidemiol* 180(2):129–139. <https://doi.org/10.1093/aje/kwu143>



Chapter 21

Using the IDEOM Workflow for LCMS-Based Metabolomics Studies of Drug Mechanisms

Anubhav Srivastava and Darren J. Creek

Abstract

Rapid advancements in metabolomics technologies have allowed for application of liquid chromatography mass spectrometry (LCMS)-based metabolomics to investigate a wide range of biological questions. In addition to an important role in studies of cellular biochemistry and biomarker discovery, an exciting application of metabolomics is the elucidation of mechanisms of drug action (Creek et al., *Antimicrob Agents Chemother* 60:6650–6663, 2016; Allman et al., *Antimicrob Agents Chemother* 60:6635–6649, 2016). Although it is a very useful technique, challenges in raw data processing, extracting useful information out of large noisy datasets, and identifying metabolites with confidence, have meant that metabolomics is still perceived as a highly specialized technology. As a result, metabolomics has not yet achieved the anticipated extent of uptake in laboratories around the world as genomics or transcriptomics. With a view to bring metabolomics within reach of a nonspecialist scientist, here we describe a routine workflow with IDEOM, which is a graphical user interface within Microsoft Excel, which almost all researchers are familiar with. IDEOM consists of custom built algorithms that allow LCMS data processing, automatic noise filtering and identification of metabolite features (Creek et al., *Bioinformatics* 28:1048–1049, 2012). Its automated interface incorporates advanced LCMS data processing tools, mzMatch and XCMS, and requires R for complete functionality. IDEOM is freely available for all researchers and this chapter will focus on describing the IDEOM workflow for the nonspecialist researcher in the context of studies designed to elucidate mechanisms of drug action.

Key words Metabolomics, IDEOM, Drug mechanism, Mode of action, Microsoft Excel, LCMS, Data processing

1 Introduction

The main challenges in untargeted metabolomics studies include the low signal to noise ratio, metabolite identification and comparative data visualization. IDEOM attempts to address all these issues through its user-friendly graphical user interface built in Microsoft Excel. The MS Excel spreadsheet operates through a number of VBA macros that allow automated data processing of high-resolution LCMS data. These macros allow for automatic generation and execution of scripts in the VBA and R environment [1]

(www.r-project.org) using adjustable parameters. Using powerful inbuilt tools for LCMS data processing—mzMatch [2] and XCMS [3]—it generates a list of putative metabolites with associated confidence levels and peak intensities, with intuitive data visualization and standard statistical tests. This chapter will provide a step-by-step guide for the use of IDEOM to investigate the mode of action (MoA) of a drug.

The main approaches used during the early stages of drug discovery are target-based screens and phenotypic screens. While target-based screens are a popular approach to identify small molecules that will act by a predefined mode of action, phenotypic screening has proved to be a successful avenue for the identification of many drug candidates [4]. For example, in the field of infectious diseases it can be very efficient to screen large chemical libraries for their ability to inhibit pathogen growth, giving rise to a large number of hit compounds that can be triaged and optimized to provide lead drug candidates [5, 6]. The obvious issue with drug discovery based on phenotypic screens has been limited information about the drug target responsible for the observed phenotype. The knowledge of the mode of action or target is vital as it can provide a structural basis for chemical optimization of the hit compound (e.g., for improving binding), it can underpin the rational design of combination therapies and can also help in monitoring for resistance to the therapeutic compounds in the clinical setting.

Metabolomics studies have been shown to help in finding the mode of action of novel and existing pharmaceutical compounds [7–11]. Typically, this involves *in vitro* incubation of the relevant cell population with the investigational drug, followed by untargeted metabolomics analysis. However, care must be taken to ensure that the metabolic profile observed following drug incubation is representative of the specific action of the drug. While higher drug concentrations and longer incubations may lead to a more definitive assay outcome (e.g., resulting in cell death/growth arrest), these conditions will likely lead to off-target affects, thus confounding identification of the metabolic perturbation primarily responsible for the action of the drug. On the other hand, low concentrations and/or short incubations may not allow for sufficient pharmacological activity to an extent whereby the metabolic profile can be differentiated from the untreated control. Therefore, a balance must be achieved both in duration and concentration of the drug in MoA studies using metabolomics, whereby the optimal conditions induce specific perturbations to metabolism without causing cell death. Ideally, some simple activity assays should be conducted with escalating drug exposure to determine the highest achievable sublethal concentration and duration conditions that will be suitable for metabolomics analysis. The addition of time-course, concentration-ranging, and negative control (e.g., inactive chemical analogues and compounds that kill by distinct

mechanisms) samples also improves interpretation of the link between specific metabolic responses and mechanisms of drug action. Consideration should also be given to the sampling conditions, as the biomass requirements for metabolomics analysis (e.g., >1 million cells/sample for most human cell lines) may differ from standard activity assay conditions, and the metabolism must be quenched for analysis in a short but practical time window which will allow experimental reproducibility.

In this chapter, we present an excerpt from a study [7] where LCMS-based metabolomics data was analyzed using the IDEOM workflow to find the mode of action of antimalarial compounds. This study used compounds from the Malaria Box, an open-access collection of “hit” antimalarial compounds that were identified by phenotypic screening and assembled by the Medicines for Malaria Venture [12]. The lack of information on the mode of action of these compounds is a significant hurdle to further development of these hits. This study used simple *in vitro* incubation of cultures of the malaria parasite, *Plasmodium falciparum*, with the test compounds followed by LCMS-based metabolomics analysis using IDEOM, and generated MoA information which will be crucial for future drug discovery efforts based on these hits.

2 Materials and Methods

2.1 Cell Culture and Drug Incubations for Metabolomics Analysis

Trophozoite stage asexual *P. falciparum* (3D7) parasites (causative agents of malaria) were cultured using a previously defined method [13], using human RBCs (obtained from the Australian Red Cross Blood Service) at 7–8% parasitemia (iRBC) and 3% hematocrit in RPMI 1640 with hypoxanthine and 0.5% Albumax (Gibco) at 37 °C under a specialized gas mix (95% N₂, 4% CO₂, 1% O₂). For drug-induced metabolic perturbation experiments, 200 µL iRBCs were incubated with 1 µM of test compounds for 5 h in 96-well plates in quadruplets. This short incubation with test compounds was intended to induce metabolic perturbations in the malaria parasite without causing significant mortality in the cell population, as *P. falciparum* *in vitro* activity assays are usually performed using a 48 – 72-h incubation.

2.2 Metabolite Extraction for Metabolomics Analysis

Following sublethal drug incubation, culture medium was removed and the metabolism of cells was quenched by resuspending the cells in ice-cold PBS. All the subsequent steps were performed on ice. Cells were washed by centrifugation at 1000 × *g* for 5 min and PBS was removed. Metabolite extraction was performed by adding 135 µL methanol (containing internal standards) followed by rapid mixing. Samples were then gently mixed for 1 h at 4 °C and then centrifuged at 3000 × *g* for 5 min to precipitate proteins and other insoluble molecules. The supernatant, which contained the

metabolite extract, was transferred to glass LC-MS vials and stored at -80°C before LC-MS analysis. A 10 μL aliquot of each sample was pooled together to generate a quality control (QC) sample which was used to monitor sample stability and analytical reproducibility.

2.3 LC-MS Based Metabolomics Analysis

The method of choice for this study was LC-MS, using hydrophilic interaction liquid chromatography (HILIC) and high-resolution (Orbitrap) mass spectrometry (MS). Briefly, Samples (10 μL) were injected onto a Dionex RSLC U3000 LC system (Thermo) fitted with a ZIC-pHILIC column (5 μm , 4.6 by 150 mm; Merck). 20 mM ammonium carbonate (A) and acetonitrile (B) were used as the mobile phases with a 30 min gradient starting from 80% B to 40% B over 20 min, followed by washing at 5% B for 3 min and reequilibration at 80% B. Mass spectrometry utilized a Q-Exactive MS (Thermo) with a heated electrospray source operating in positive and negative modes (rapid switching) and a mass resolution of 35,000 from m/z 85 to 1050. Before each batch of samples, blanks and mixtures of authentic standards (234 metabolites) were analyzed, and 2 pooled extracts were analyzed in data-dependent tandem mass spectrometry (MS/MS) mode to facilitate downstream metabolite identification where necessary. Pooled QC samples were analyzed periodically throughout each batch.

2.4 LC-MS Data Analysis Using IDEOM

Data was analyzed using IDEOM pipeline as described in this chapter (see Subheading 3). Briefly, raw files were converted to mzXML with msconvert [14], LC-MS peak signals were extracted with the Centwave algorithm in XCMS [15], samples were aligned and artifacts were filtered with mzMatch [2] and additional data filtering and feature identification was performed based on accurate mass and retention time [16]. The parameters used in automated IDEOM filtering for this study are demonstrated in Fig. 1, and the annotated peak metadata and chromatogram images allowed further manual data filtering to remove features that were not reliably detected across replicates and those with very low-quality chromatographic peaks. This resulted in reduction of the 15,000 detected LC-MS peaks, which includes many noise and electrospray artifacts, to a manageable 460 reproducible peaks with reasonable putative identifications. Metabolite identification (level 1 confidence according to the Metabolomics Standards Initiative (MSI) [17] corresponded to confidence score of 10 in IDEOM based on accurate mass and retention time for metabolites that were present in the standard mixture. Other features were putatively annotated (MSI level 2) based on accurate mass and predicted retention time using the IDEOM database. Metabolite abundance was determined by peak height and normalized to the average for untreated samples from the same batch. Univariate statistical analyses utilized Welch's t test ($\alpha = 0.05$) and Pearson's correlation (Microsoft

XCMS (Centwave)		Confidence levels	
Method (file type):	mzXML	arbitrary	
ppm:	2	Standard RT within 5%	9
peak width (min):	5	Calculated RT within 50%	7
peak width (max):	100		
S/N threshold:	3		
Prefilter (# points):	3		
Prefilter (intensity):	1000		
Mzdiff:	0.001		
mzMatch		ID-dependent rejection	
Mzmatch grouping RT window:	0.5 min	Xenobiotics	4.5
Mzmatch grouping m/z ppm:	5 ppm	RT outside window	3
Relative Std Dev (RSD) filter:	0.50 TECHNICAL		
Noise filter (codadw):	0.80		
Intensity filter (LOQ):	100000		
Minimum detections #:	4		
RT window for related peaks:	0.10 min		
IDEOM		Peak-dependent rejection	
RT for id of authentic standards:	5.0 %	Below intensity filter	0.5
RT for id for calculated RT:	50.0 %	RSD filter	0.4
PPM for mass identification:	3.0 ppm	Shoulder/duplicate peak	0.2
Ignore related peaks before RT:	0.0 min	common adducts/fragments/isotopes	0.1
RT window for complex adducts:	0.50 min	not more than blank control	0
RT window for Duplicatepeaks:	0.33 min		
RT window for Shoulderpeaks:	1.0 min		
Intensity ratio for Shoulderpeaks:	5 to 1		
Intensity limit Duplicatepeaks:	1 %		
r2 limit for duplicatepeaks	0.99		
Preferred DB:	PfalcDB pfa	Confidence modifiers	add
		Preferred DB	1
		Related peak (mzMatch)	-2
		Filtering threshold: (adjust in macro)	5
STATISTICS:			
		Minimum detectable intensity:	10000
		Statistical P-value:	0.05
		Study design:	Unpaired

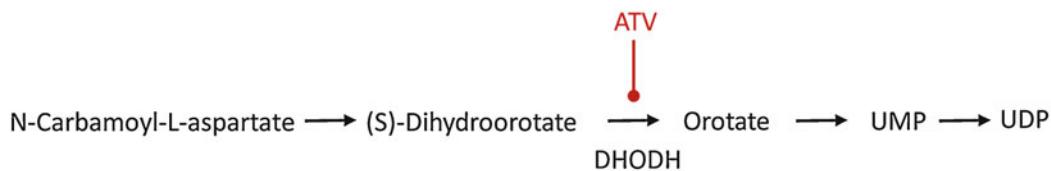
Fig. 1 Parameters used for automated IDEOM filtering and feature identification from raw metabolomics data in the atovaquone mode of action study [7]

Excel) and multivariate analysis was performed by using the Metabolomics package in R [18] through IDEOM.

2.5 Mode of Action of Atovaquone Confirmed by IDEOM-Based Analysis

Although 100 compounds were tested in this study, for the purposes of this chapter we focused on the mode of action of a known antimalarial drug, atovaquone. The comparison sheet (see IDEOM processing) showed the metabolite intensities of drug-treated samples expressed relative to the “control” group, which was DMSO (vehicle control) in this study (IDEOM file with entire data set can be downloaded from the supplemental material for this study [7]). As expected [19], specific changes in the levels of metabolites involved in pyrimidine biosynthesis were observed in metabolite extracts from parasites treated with atovaquone. The observed increase in levels of *N*-carbamoyl-L-aspartate and dihydroorotate in atovaquone-treated cultures, and the decrease in downstream

	A	B	C	D	E	F	G	H	I	J	K
	Sort	Trend Sort	Import Peaks	Search	Tools	Graphs	Export	Reference	Pathway	max intensity	
1	Mas	RT	FORMU	Isomer	Putative metabolite	Reference	Map	Pathway		ATQ1	DMSO1
428	158.03	10.83	C5H6N2O4	3	(S)-Dihydroorotate	18	Nucleotide Metabolism	Pyrimidine metabolism	211580	76.57	1
429	176.04	16.56	C5H8N2O5	2	N-Carbamoyl-L-aspartate	8	Nucleotide Metabolism	Pyrimidine metabolism	133683333	39.92	1
430	128.06	14.62	C5H8N2O2	3	5,6-Dihydrothymine	8	Nucleotide Metabolism	Pyrimidine metabolism	10114805	0.86	1
431	404	16.14	C9H14N2O1	1	UDP	8	Nucleotide Metabolism	Pyrimidine metabolism	636850	0.69	1
432	324.04	14.83	C9H13N2O5	4	UMP	10	Nucleotide Metabolism	Pyrimidine metabolism	173588	0.25	1
433	156.02	9.912	C5H4N2O4	2	Orotate	8	Nucleotide Metabolism	Pyrimidine metabolism	365120	0.00	1



Pyrimidine biosynthesis

Fig. 2 Top panel: Snapshot of the comparison sheet highlighting the metabolites involved in pyrimidine biosynthesis in the atovaquone mode of action study [7]. Bottom panel: Schematic representation of the pyrimidine biosynthesis pathway in malaria parasite and the step inhibited by atovaquone (ATV: Atovaquone, DHODH: dihydroorotate dehydrogenase)

metabolites orotate, UMP, and UDP, compared to DMSO control (Fig. 2) was indicative of inhibition of dihydroorotate dehydrogenase (DHODH) activity. This is consistent with the known mechanism of action of atovaquone involving the primary inhibition of ubiquinol oxidation by the cytochrome bc1 complex [19], which provides the essential cofactor for DHODH. This indirect inhibition of *P. falciparum* DHODH by atovaquone has been independently genetically validated as the primary mechanism of action of atovaquone [20]. Furthermore, this specific impact of atovaquone on pyrimidine pathway metabolites has been demonstrated in two independent studies that analyzed these metabolites using targeted LCMS analysis [8, 21].

3 IDEOM Workflow

3.1 Installation

The MS Excel-based IDEOM template can be downloaded from <http://mzmatch.sourceforge.net/ideom.html>. It does not require any installation, and can be loaded directly in a recent version of Microsoft Excel (2010 or later). For full functionality, users require

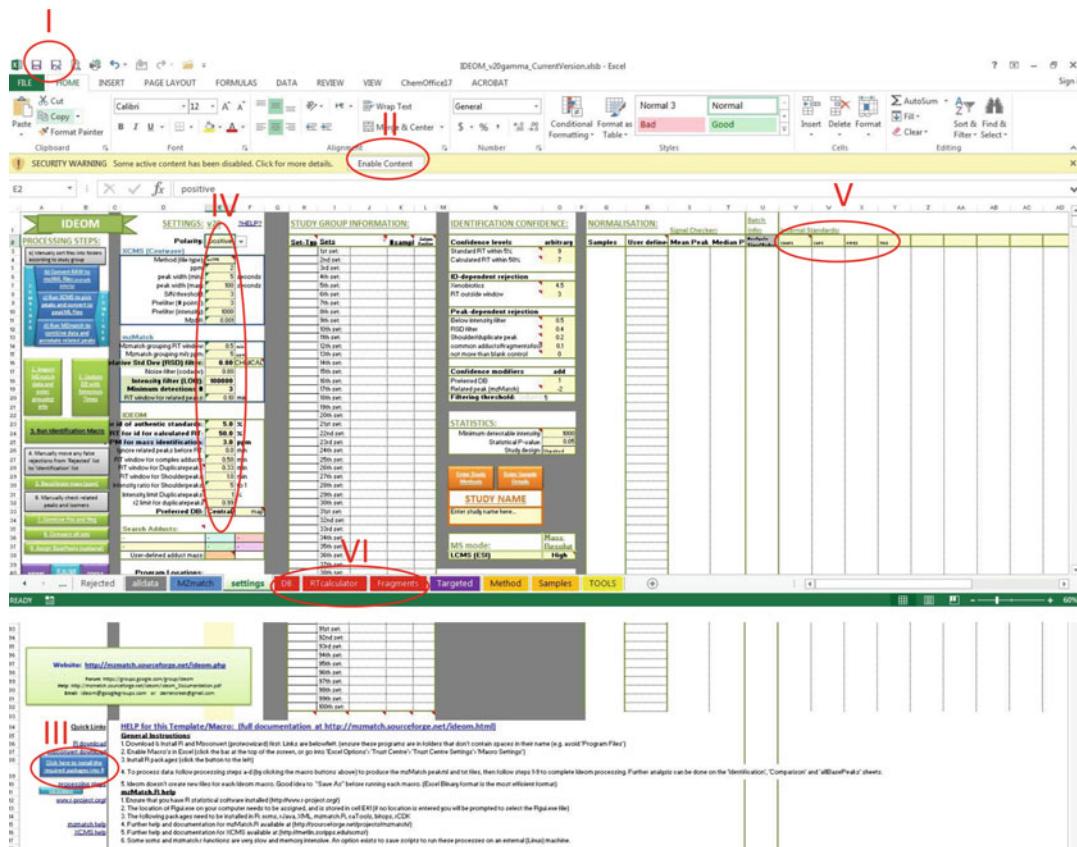


Fig. 3 Screenshot of the IDEOM template as it appears when the file is first opened. Areas circled in red indicate the location of I: “Save As” button, II: “Enable Content” button to allow macros to run, III: Button for installation of the necessary R packages, IV: Adjustable parameters for IDEOM, mzMatch and XCMS functions, V: Cells to enter names of the internal standards, VI: worksheets containing metabolite database, retention time and fragment information. See information in the “Getting Started” section for more details

a current installation of R (www.r-project.org) and proteowizard tools, which can be downloaded from <http://proteowizard.sourceforge.net/downloads.shtml>. Hyperlinks to Xcalibur (Thermo) and web browsers are included in the IDEOM template; however, these programs are not required for successful implementation of IDEOM.

3.2 Getting Started

Figure 3 gives a screenshot of the IDEOM template as it appears when the file is first opened.

1. Open the IDEOM.xlsb file in MS Excel and “Save As” with your study name (e.g., mystudy1.xlsb).
2. Enable Macros by clicking the “Enable content” button in the “Security Warning” ribbon.

3. (First time only) Install all required R packages by clicking the appropriate blue button in the help section at the bottom of the page. (You may need to scroll down to row 57 to find this). The blue buttons execute functions in R (not within Excel); you may be prompted to select the “Rgui.exe” file on your computer. Please agree to any Windows security warnings.
4. All settings for XCMS, mzMatch, and IDEOM processing are located in column E.
5. (optional) Up to 10 internal/external standards may be entered into cells U2:AD2.
6. (optional) Red sheets contain advanced settings and databases.

3.3 General Information

Ensure to save the IDEOM template as a macro-enabled workbook in “.xslb” Excel binary format. It is recommended to save the file before running each step. Macros are activated by clicking the colored buttons. In-cell hyperlinks in Excel are activated by double-click and web links are activated by single-click. The IDEOM template includes a number of Excel “worksheets” (Fig. 3) whose features are described in Table 1.

3.4 Processing Steps

3.4.1 Raw Data Processing

The raw data processing steps can be executed by clicking the corresponding blue buttons at the top-left of the settings sheet (Fig. 4)

1. Manually sort files into folders according to study group.
 - (a) Create a working directory which is *not* within “My Documents” or “Program Files,” create folders for each group of replicates in the study (control, treatment, blank, QC, etc.) and move relevant raw files into these folders.
 - (b) This step may be skipped if the user wishes to process all files without grouping.
 - (c) Avoid spaces in folders and filenames. It is recommended to replace space with underscore in the names. For example; use “IDEOM_trial” rather than “IDEOM trial.”
2. Convert RAW to mzXML files and split polarity.
 - (a) Run **step (b)** (click blue button) if dealing with raw files. (Skip this step if you already have mzxml files).
 - (b) This step uses msconvert (or ReAdW), through R, to convert raw LCMS files to the .mzXML format. Msconvert is recommended.
 - (c) The location of msconvert.exe (or ReAdW.exe) on your computer needs to be assigned, and can be stored in cell E43.
 - (d) msconvert.exe needs to be in the same folder as the Pwiz dll files (keep all Pwiz files together). (ReAdW requires zlib1.dll to be in your windows directory to work correctly.)

Table 1
Description of worksheets in the IDEOM file

Sheet	Description
Settings	Home page and the starting point for all analysis. It contains many basic settings, help documentation and the main macro buttons for processing data. Default settings are suitable for 4.6 mm ZIC-pHILIC chromatography (with ammonium carbonate/ACN, 0.3 mL/min) coupled to the Q-Exactive Orbitrap
MZmatch	Blank sheet to allow import of the peak list from mzMatch (or other tools)
Alldata	Information about every peak set in the peak list is written to this sheet
Rejected	Peak sets with putative identification, but confidence below 5, are copied to this sheet. Most of these peaks are noise or artifacts, but users may wish to scan them manually for false rejections
Identification	All identified metabolites with confidence of 5 or above are copied to this sheet
allBasepeaks	All base peaks (from mzMatch related peaks function) are copied to this sheet unless the peaks are not significant in any group (less than blank)
Comparison	Comparative peak intensity data is placed here after running the “compare all sets” macro. Many functions for evaluation and visualization are available on this sheet
DB	This sheet contains the full metabolite database and associated metadata. If required, additional metabolites may be added to the bottom of the list, or additional property columns to the right of existing columns. Additional information may also be added to existing columns, but do not insert new columns between existing data
RTcalculator and fragments	These contain important tables of information required for the IDEOM macros. Advanced users may wish to update these tables with instrument-specific values
Targeted	This sheet allows a targeted analysis of specific metabolites
Tools	This sheet demonstrates the utilization of IDEOM’s user-defined Excel functions, suitable for manual calculations such as mass determinations
Method and samples	Extra sheets to allow uploading of experiment metadata

- (e) TIC CHECKER: This function is available in the R script menu at the bottom of the page and allows users to run this additional script in R. It is not essential, but provides a preliminary opportunity to check signal reproducibility of the mzXML files.
3. Run XCMS (and mzmatch) to pick peaks and convert to peakML files.
 - (a) This step uses the Centwave function in xcms (through R) to pick peaks, and then mzmatch converts each individual file to peakml format.

IDEOM

PROCESSING STEPS:

- a) Manually sort files into folders
- b) Convert RAW to mzXML files (and split)
- c) Run XCMS to pick peaks and convert to peakML files
- d) Run MZmatch to combine data and annotate related peaks

SETTINGS: v19 [?HELP?](#)

Polarity: combined

XCMS (Centwave)

Method (file type):	mzXML
ppm:	2
peak width (min):	5 seconds
peak width (max):	100 seconds
S/N threshold:	3
Prefilter (# points):	3
Prefilter (intensity):	1000
Mzdiff:	0.001

mzMatch

Mzmatch grouping RT window:	0.5 min
Mzmatch grouping m/z ppm:	5 ppm
Relative Std Dev (RSD) filter:	0.50 TECHNICAL
Noise filter (codadw):	0.80
Intensity filter (LOQ):	100000
Minimum detections #:	4
RT window for related peaks:	0.10 min

IDEOM

RT for id of authentic standards:	5.0 %
RT for id for calculated RT:	50.0 %
PPM for mass identification:	3.0 ppm
Ignore related peaks before RT:	0.0 min
RT window for complex adducts:	0.50 min
RT window for Duplicatepeaks:	0.33 min
RT window for Shoulderpeaks:	1.0 min
Intensity ratio for Shoulderpeaks:	5 to 1
Intensity limit Duplicatepeaks:	1 %
r2 limit for duplicatepeaks	0.99
Preferred DB:	PfalcDB pfa

Search Adducts:

-	-	-
-	-	-
User-defined adduct mass:		

Fig. 4 Screenshot from the settings sheet with the raw data processing steps (blue buttons) and IDEOM data processing steps (green buttons)

Table 2
List of automated mzMatch functions

mzMatch function	Description
Group	Groups peaks across replicate samples based on specified mass and RT windows (optional, requires files to be assigned to group folders before starting)
RSD filter	(Relative Standard Deviation) removes features with high variability in peak intensity for replicates within a group (optional, requires previous group function)
Combine	Combines all groups/samples into one peakml file (peak list)
Noise filter	Peak shape is assessed by CoDA-DW score (0–1)
Intensity filter	Features are removed if no sample has a peak above the intensity threshold
Detections filter	Peaks must be present in a minimum number of samples
Gap-filler	Fills gaps (missing values) in the peak intensity table where peaks were detected in some samples but not others. This requires access to the raw data (mzxml files) to extract the relevant ion signals from the given <i>m/z</i> and retention time windows
Related peaks annotation	Annotates peaks that are likely to be ESI artifacts (e.g., isotopes, adducts (Na^+ , K^+ , Cl^- , ACN, etc.), fragments, multiply charged species, dimers, multimers, complex adducts or FT artifact signals) based on retention time, peak shape and correlation of peak intensities across samples
Output	Generates output files in peakml format and a peak list table in .txt format

- (b) Settings for the Centwave function can be changed in cells E4–E11 of the Settings sheet — for full documentation see xcms help (metlin.scripps.edu/xcms).
- (c) Briefly, the parameters are:
- PPM: mass deviation from scan to scan.
 - Peakwidth: range for baseline peak width.
 - S/N threshold: Minimum signal to noise ratio.
 - Prefilter: number of scans greater than a given intensity threshold.
4. Run MZmatch to combine data and annotate related peaks.
 This step uses mzMatch to group individual peakml files, filter peaks and annotate related peaks. Automated mzMatch functions are listed in Table 2.
- Settings can be changed in cells E14–E20, and E25 on the Settings sheet, additional parameters can be altered by saving the R script and running it externally — for documentation see mzmatch help (mzmatch.sourceforge.net). The default settings should be a good starting point, though you might need to change RSD filter (E16) if the data is variable, or Minimum Detections # (E19) if there are less than 3 replicates in any group. Gapfiller requires mzXML files to be in the same

location as when they were converted to peakml files (**step c**), therefore it is a good idea to run **steps (c)** and **(d)** together (on the same computer) by using the “COMBINED” button. Gapfiller is very memory intensive so if the process crashes at this step, try closing the R session and continuing the script manually from the gapfiller stage. If this does not work then tighten the filters (RSD, min detections) to make smaller peakml files. Another option would be to run it on a computer with a higher RAM capacity.

3.5 IDEOM Processing

The main IDEOM data processing steps and associated parameters are found on the settings sheet (Fig. 4). While most IDEOM processing is automated, some study-specific input is required from the user. Optimal results are achieved by clicking **steps 1–9** (green buttons) and following the on-screen prompts. **Steps 1** and **2** can be run in any order, but must both be completed before running the main processing macro (**step 3**)

1. Import MZmatch data and enter grouping info.
 - (a) Use this function to import data from the mzMatchoutput.txt file produced by mzMatch. (If you have already entered data manually, or by the “import example data” or “import Mzmine data” buttons, you may press cancel at the “import file” screen to skip this step.)
 - (b) The second part of this function asks the user to enter grouping information. “Autofill” can be used if the prefix of the sample names refers to the grouping, otherwise manually select groups using the “add” buttons.
 - (c) Set-Type needs to be selected for each group using the drop-down lists (Table 3). Always set one group as “Treatment” and another as “Control” to allow statistical comparisons.
 - (d) If there are more than 15 sample groups, there is a second tab with space for 15 more groups. IDEOM currently supports a maximum of 30 groups (Fig. 5).
 - (e) The third part of this function plots average sample intensities to allow a quick check of whether the data is consistent. Internal (external) standards will also be plotted if details have been entered in cells U2-AD2 of the settings sheet.
 - (f) The fourth part gives the option of normalizing the data either by TIC, median, or user-defined values (column R on settings sheet). Normalization is not routinely recommended for LCMS data due to nonlinear responses and the unpredictability of ion-suppression.

Table 3
Description of set types (groups) in the settings sheet

Set types (groups)	Description
Blank	Solvent blank(s), used as the background reference for the groups significance filter
Control	Base group for intensity comparisons. Only one control can be set at any time. If you wish to compare results to multiple controls please use “Save As” to get different Excel files corresponding to each control when you run the “Compare All Sets” macro
Treatment	Initial 2-sample intensity comparison, and used for the QC: RSD if no QC is present. Only one treatment can be set at any time
Sample	Any real sample groups that are not selected as the initial “control” or “treatment” group should be set as “Sample.” Undefined groups are assumed to be “sample”
QC	This group is used for the RSD filter and is included in graphs of individual samples, but not in the comparison of results
Standards and Exclude	These groups are excluded from all functions, but the peak intensity data is retained in the data matrix on all sheets

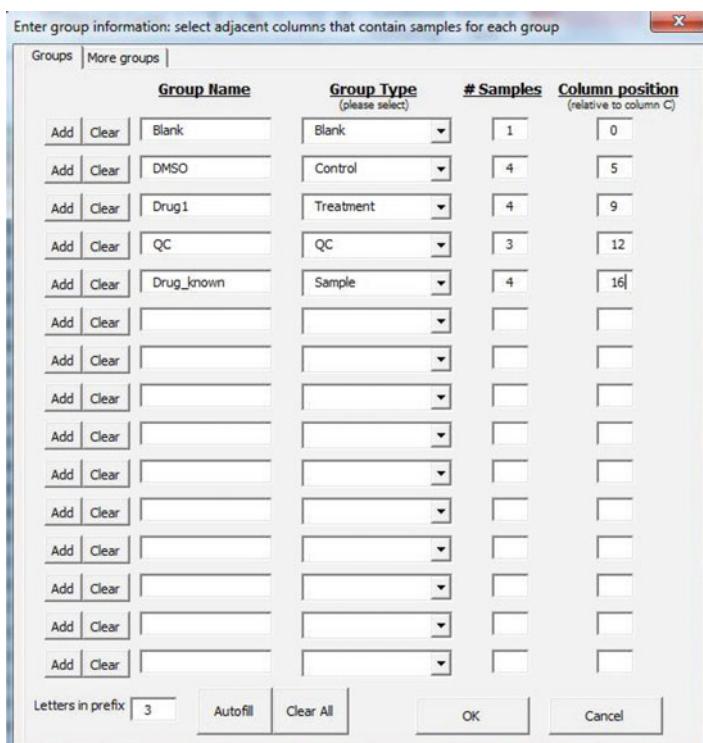


Fig. 5 Screenshot of the window where group information can be entered while importing the mzMatch data. Refer to Table 3 for description of Group Types

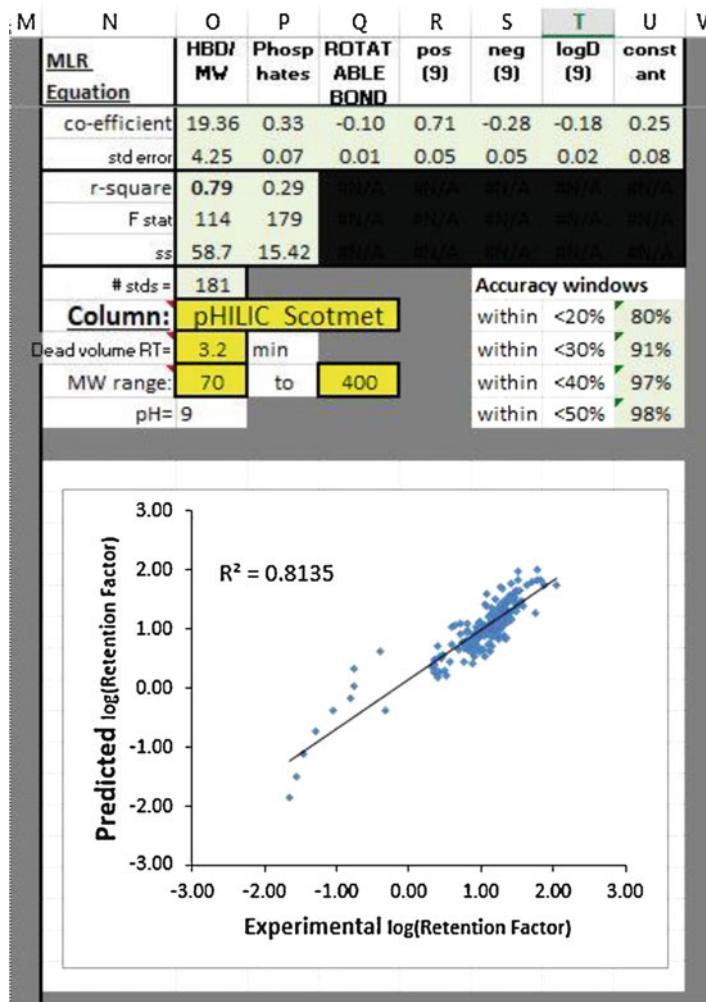


Fig. 6 Retention time prediction model on the RTcalculator sheet, showing values associated with the current multiple linear regression model, and a plot of the model fit for the current metabolite standards

- (g) The grouping functionality in IDEOM assumes that replicates in the mzMatch output data are in adjacent columns. This sample ordering should automatically occur if using the mzMatch preprocessing workflow described above, otherwise the column order should be adjusted in the txt file prior to import.
2. Update DB with Retention Times.
 - (a) This function enters standard retention times into the database (DB sheet), and (optional) enters predicted retention times for other metabolites [22].
 - (b) A list of retention times for the particular LCMS method, obtained from analysis of authentic standards (ideally in

Table 4
List of RSD parameters for each group of replicates

RSD parameters	Description
Strict	If any group has RSD larger than this setting, the metabolite is considered unreliable and given a low confidence score (0.4)
Technical	If the QC group (or Treatment group in the absence of a QC group) has RSD larger than this setting, the metabolite is given a low confidence score (0.4)
Generous	No specific RSD filter is applied, although if no group has an RSD below the filter threshold then the peak is not significant and has confidence level of 0
Exclusions	Groups with mean intensity below the LOQ threshold are excluded from this test (as they often have higher RSD)

the same batch), is required (create this list either using the Targeted Sheet in IDEOM, or externally using targeted analysis software such as Toxid, Tracefinder, or Xcalibur).

- (c) The list of standard RTs may be either imported from any Excel-readable file, or entered directly into columns A and B on the “RTcalculator” sheet.
- (d) If importing .csv files of retention times: “_” in metabolite names will be replaced with “,”.
- (e) All authentic standards (column A) must have names that exactly match those in the DB sheet.
- (f) RT calculator uses physicochemical properties in the DB sheet to predict retention times based on a multiple linear regression model with the authentic standards (QSPR approach [22]).
- (g) You have the opportunity to check the model fit before annotating all metabolites in the database (Fig. 6).
- (h) If there is no good prediction model you can still use this function to upload standard retention times for those metabolites where you have authentic standards.
- (i) RTcalculator sheet parameters:
 - The column (cell O8) and dead volume (mins) should be entered before running this macro (cell O9). Other data in columns N-U show the accuracy of the current RT prediction model.
 - The mass range for application of the prediction model is defined in cells O10 and Q10. The default model, “pHILIC_Scotmet”, is suitable for the ZIC-pHILIC method with ammonium carbonate and acetonitrile mobile phase (see LCMS method in the example above) for metabolites within the MW range 70–400.

Table 5
Description of columns in Identification, Rejected, All data, and all Base peaks sheets

Column	Description
A	Neutral exact mass (from mzMatch)
B	Retention time (from mzMatch) in minutes
C	Formula from DB with closest match to mass (if within ppm window)
D	Number of isomers in DB with this exact formula
E	Metabolite name: Best match from DB for this mass and RT (bold type if there is a standard RT for this metabolite in the DB)
F	Confidence level (arbitrary out of 10) according to parameters on “settings” sheet
G	Records whether the metabolite is in a “preferred database” (from DB)
H	Map: The general area of metabolism for this metabolite (usually from KEGG) Note: Columns G and H can be changed by choosing a different header in cell G1 or H1
I	Mass error (in ppm) from nearest match in DB
J	RT error relative to authentic standard (name is bold) or predicted RT as % of RT (NB: Where no predicted RT exists in database the cell is colored red)
K	Alt ppm: Mass error for the next closest mass in the DB (if within ppm window) — If font is red it is a double-charged match, if background is colored it is an adduct match — See settings D36:F38 for adduct colors
L	Groups: Records which sample groups have significant peaks detected for this metabolite (peaks > blanks, and RSD < RSDfilter)
M	BP: Basepeak (if identified) for that peak (from mzMatch: Largest coeluting peak with correlated peak shape and intensity trend across samples) NB: This does not update when other identifications are changed to a different isomer
N	Mzdiff: Mass difference between this peak and the basepeak
O	Relationship: Relationship to the basepeak (according to mzMatch)
P	Addfrag: Common adduct, isotope, fragment or neutral-loss: Based on filters and formulas the “fragments” sheet. Possible complex adducts of larger coeluting peaks (in duplicatepeaks window) are also annotated here. Note: a “complex adduct” may sometimes be the parent of two fragment ions
Q	% error of detected ^{13}C -isotope intensity from the theoretical ^{13}C -isotope intensity (note: This relies on filtered peaks, and does not go back to the raw data)
R	Related peaks: A list of common related peaks (isotopes, multicharged, adducts, fragments) that the macro has detected for this metabolite
S	RSD (relative standard deviation) for QC samples (or for treatment group if no QC is assigned)
T	Maximum RSD for all included sample groups
U	Maximum intensity from all included samples
V	Relation id (from mzMatch)

(continued)

Table 5
(continued)

Column	Description
W	Peak intensity ratio for mean of “treatment” group vs. mean of “control” group
X	P-value for unpaired <i>t</i> -test between “treatments” and “controls”
Y	Adduct of formula match to mass (H, Na, double-charge, etc.)
Z	Polarity (in combined files the first-named polarity is that with the biggest peak. All sample intensities in combined file are taken from the polarity with the biggest peak for each metabolite)
AA	Number of detected peaks in included groups
NEXT #	Peak intensities from all samples present on the mzmatch sheet
Extra columns	Other functions such as “compare with medium,” “compare 2 other groups,” and “isotope search” will add extra columns. Users may add additional columns to the right without affecting macro performance

- Columns W:X allow standard retention times to be uploaded to the database without being included in the prediction model (e.g., for large metabolites outside the validated mass range).
- Columns Z:AI allow predicted retention times to be uploaded to the database based on class properties, according to specific annotations in the DB sheet (Default values are for the pHILIC method above).
- Headers E1:J1 can be adjusted to other phys-chem properties (from drop-down menus) if a different RT prediction model is required (select “manual” column).
 - (j) If this function is not implemented then metabolite identification will only be based on exact mass (not retention time), hence there will be many more false-identifications.

3. Run Identification Macro.

Before running this macro, check all settings in column E of the Settings sheet. It is worth noting that the settings for RSD filter, intensity filter, minimum detections and related peaks window (under mzMatch settings, cells E16:E20) are also used in IDEOM functions.

- (a) RT and mass windows (E23:E25) may need to be changed depending on the quality of the data.
- (b) Preferred DB (E33) should be chosen from the drop-down list. “Central” refers to central pathways in KEGG (Cofactors and vitamins, amino acid, nucleotide, carbohydrate, lipid, and energy metabolism). Additional

Table 6
Description of columns in comparisons sheet

Column	Description
A	Neutral exact mass (from mzMatch)
B	Retention time (from mzMatch) in minutes
C	Formula from DB with closest match to mass (if within ppm window)
D	Number of isomers in DB with this exact formula
E	Metabolite name: Best match from DB for this mass and RT (bold type if there is a standard RT for this metabolite in the DB)
F	Confidence level (arbitrary out of 10) according to parameters on “settings” sheet
G	Map: The general area of metabolism for this metabolite (usually from KEGG)
H	Pathway: List of biochemical pathways for this metabolite (usually from KEGG)
I	Max intensity: Maximum LCMS intensity in any sample for that metabolite. Note: Columns G, H, and I can be changed by choosing a different header in cell G1, H1, or I1
J	Mean intensity of each included group relative to the “control” group (as set when the “comparison” macro was run). Significant (<i>t</i> -test) values are in bold
Next #	<i>P</i> -values for unpaired <i>t</i> -test between each included group and the control
Next #	Mean intensity for each included group
Next #	Standard deviation for each included group
Next #	Relative standard deviation for each included group
Next #	Fisher ratio for each included group, relative to the control group
Additional	Additional columns are added every time you sort by correlation of intensity trends relative to a specific metabolite. You may add your own additional columns to the right of existing data without affecting performance. Please do not insert columns between existing data

organisms/annotations may be added to the database using the purple “Annotate DB” macro, or by manually appending a column to the end of the DB sheet (column BN).

- (c) Additional adducts (other than H^+/H^-) can be searched by selecting them in cells D36:F38 on the settings sheet. (mzMatch has already corrected the data for 1 proton). It is recommended to select “Na” and “ NH_3 ” in positive mode data, “Cl” and “Formic acid” in negative mode data, and “double-charge” for both ionization polarities.
- (d) This macro runs most of the IDEOM functions to filter data and identify peaks, data is recorded in the “alldata” sheet, and then copied to the other results sheets.

The following list of functions is applied to each peak:

Table 7

List of shortcuts obtained by double-clicking in columns of Identification, Rejected and all Base peaks sheets

Column	Description of shortcut
A	(Mass) Plots the mass spectrum for all coeluting peaks. Red peaks are RelatedPeaks (from mzMatch). Hover mouse over lines to get annotation information
B	(Retention time) plots intensity and RT of all isomeric peaks for this mass from all samples. There is also the option for a quick link to look at EIC for a specific RAW file in Xcalibur, or to produce EIC(s) of this mass from a peakml file
C	(Formula) checks all possible adducts for formulas, then gives option to copy a mass for you to paste into either Xcalibur or ChemSpider search engine, or run a process in R to find possible chemical formulae
D	(Isomers) shows a list of isomers in DB with this exact formula, also shows % retention time error, databases and pathways (from DB)
E	(Name) plots a graph of intensities for each sample. Also, single-click gives a dropdown list that enables you to select any isomer from the DB with the selected formula (data in columns F:J will change depending on this selection)
F	(Confidence) - In rejected sheet only - looks to see if this metabolite has been detected in the “identification” sheet. If present, the retention time of each peak is shown
G	(PreferredDB) — If hyperlinks activated — Searches ChemSpider for the metabolite name (search engine can be changed in the hyperlinks table, e.g., PubChem or Google)
H	(DB) — If hyperlinks activated — Searches online databases for the metabolite (e.g., KEGG, HMDB, Metacyc, or Lipidmaps)
J	(RT% err) shows a list of isomers in DB with this exact formula, also shows % retention time error, databases, and pathways (same as column D)
K	(altPPM) shows information about metabolites with the alternate formula that gives this ppm error
W	(TvsCTRL) plots a graph of intensities for each sample on a log scale

Table 8

List of shortcuts obtained by double-clicking in columns of Comparison sheet

Column	Description of shortcut
A–E	Same as identification sheet (<i>see</i> Table 7 above)
F	(Confidence) gives all the information about this metabolite from the “identification” sheet
G	(Map) — If hyperlinks activated — Searches for metabolite in ChemSpider (search engine can be changed in the hyperlinks table, e.g., PubChem or Google)
H	(Pathway) — If hyperlinks activated — Searches for metabolite in online databases
I	(Groups) gives a plot of individual sample intensities

- (a) *Group detection vs. noise:* If a peak is present for every sample in a group and each peak is greater than every peak in the blanks, and the RSD is below the RSD filter threshold, then the metabolite detection is considered significant and that group name is appended to the “groups” column (L). Peaks that are not significant in any group are assigned the lowest confidence (0).
- (b) *Charge:* If ^{13}C isotope is present, the detected charge-state is recorded and taken into consideration for all mass (or m/z)-dependent functions.
- (c) The RSD (relative standard deviation) is calculated for each group of replicates. The maximum RSD is entered in column T, and the RSD for the QC's is entered in column S. If no QC's are assigned then the RSD for the “Treatment” group is entered in column S. The RSD parameters are listed in Table 4.
- (d) *Intensity filter:* The maximum intensity for all included samples is entered in column U. If this is less than the Intensity filter (LOQ) then a low confidence is given (0.5).
- (e) *Formula assignment:* based on the closest match to the database (within mass window) — if 2 matches within specified mass window the mass error for the furthest formula is noted in “altppm” column (K).
- (f) *Other adducts:* All Basepeaks (see mzmatch) are checked for other “search adducts” according to the user selections in cells D36:F38 on the settings sheet — If a formula has already been assigned, the mass error for the adduct is noted in the “altppm” column (K).
- (g) *Metabolite identification:* for all formulae is based on
 - (a) standard retention time (if present), or
 - (b) predicted retention time (if present).
- (h) *Isomers:* If metabolite assignment remains ambiguous, the first matching metabolite in the DB is assigned. NOTE: DB has been sorted to give preference to the selected “preferred DB,” then to “central” metabolites, then to metabolites present in more source databases (KEGG, MetaCyc, etc.), then to lowest metabolite ID number in source databases.
- (i) *Fragments:* Known fragments of specific metabolites are removed (according to the table in “Fragments” sheet columns Q:U, from standards analyzed on the HILIC-Orbitrap system).
- (j) *Other related peaks:* Related peaks determined by mzMatch are annotated in the “addfrag” column

(P) according to possible mass differences in the “Fragments” sheet (columns A:C). Annotations in Column C of the “Fragments” sheet designate whether specific adducts/fragments are automatically rejected with confidence level (0.4) or merely annotated with this information about a potential relationship.

- (k) *Isotopes:* ^{13}C Isotope abundances are recorded in column Q as the % error relative to the theoretical isotope abundance for the putative metabolite formula, or for unidentified peaks, as the theoretical number of carbon atoms in the formula.
- (l) *Confidence:* Arbitrary confidence levels are assigned based on retention time error, or annotation as “Xenobiotic” in “Map” column of DB sheet. Confidence is modified by whether the metabolite is in the “preferred” database, and mzMatch annotation as either “base” or “related” peak. See columns N:O of settings sheet for more details. Confidence levels for each filter may be adjusted if desired.
- (m) *Filtering:* All metabolites with a confidence score of 5 or greater are copied to the “Identification” sheet, those below 5 are copied to the “rejected” sheet.
- (n) *Duplicate annotations:* If duplicates remain with the same identification, the identity of the smaller peak is changed to the next most likely isomer that satisfies the retention-time window.
- (o) *Additional calculations:* detailed information is entered into columns A:AA for each metabolite.

4. Manually move any false rejections from “Rejected” list to “Identification” list (optional).

This is an optional step for an expert user to thoroughly interrogate the list of putatively identified metabolites that were rejected by the IDEOM algorithms. If you believe a rejected metabolite should be retained in the data-set, double-click the “Confidence” value to see if that metabolite is already in the identification sheet. Wrongly rejected metabolites can be moved to the “Identification” sheet by using the “Retrieve Row” function (Table 5).

5. Recalibrate mass (ppm).

This macro can be started from either this settings sheet, or from the button at the top of the “identification” sheet. It will plot the relationship between mass and mass accuracy (ppm error) for all “identified” metabolites, with authentic standards in red, and fit a fifth order polynomial function (this should allow for the calibration errors observed on Thermo Orbitrap). If the polynomial function appears to fit the data, agree to

recalibrate masses. If the curve is not a good fit, but you see a trend, consider manual recalibration efforts. After calibration, check the new plot of mass errors, and set a new ppm window to remove outliers (false-identifications). In some cases, it is worth checking the rejected peaks (bottom of “rejected” list) for alternative identifications by clicking the “altppm” column. If significant adjustment was made (e.g., >2 ppm) consider rerunning analysis from **step 3** (with the recalibrated data) to improve identification.

6. Manually check related peaks and isomers (optional).

For thorough analysis, check all the information supplied for each peak in the “identification” sheet. Another common approach is to skip this step initially, and return later to double-check specific metabolites of interest. While it is always a good idea to return to raw data for confirmation of specific metabolites, the identification sheet allows rapid access to a large amount of meta-information to simplify the process of manual data curation and metabolite identification.

- (a) *Isomers:* Sort by the Formula column and remove duplicates that appear to be due to poor chromatography (click a cell in the RT column (B:B) for a plot of relative intensities and retention times of all isomeric peaks).
- (b) Click the light-blue “add chromatograms” button to add cropped peak chromatogram images for each putative metabolite (click or hover in column A to view).
- (c) *Related Peaks:* Click the orange “Sort By Relation ID” button and look in columns M:P for ESI artifacts that were not filtered by the common adducts search (e.g., Metabolite-specific fragments). Double-click in the mass (column A) to see all coeluting peaks—red masses are “related” according to mzMatch.
- (d) Also look at “C isotope error” (column Q) to see if the main isotope intensity does not match the formula, Related Peaks (column R) to see if all the fragments and isotopes are possible from the identified structure, and “adduct” (column Y) to check whether the identified adduct (e.g., 2+, Na) is likely. (The “charge” column at the end may help this if a ^{13}C isotope was present, but it is sometimes not correct in noisy data).
- (e) *Alternative identification:* Easily change identification by clicking the metabolite name (column E), then select from the isomers in the dropdown list. Further information about these isomers can be obtained by double-clicking the isomer number (column D). Alternative formulas can be investigated for individual masses by double-clicking the formula (column C).

- (f) Use the “Remove row” button to easily transfer metabolites from the “identification” sheet to the “rejected” sheet. There is an option to merge peaks, which would be used for compounds with poor peak shape (e.g., lipids on a HILIC column) that appear with the same mass but slightly different retention times. If merging peaks it keeps the largest intensity from each sample.
 - (g) To speed up viewing data you may set Excel’s calculation to “manual” (Formulas >> Calculation Options), but be aware that you need to manually calculate if you make any changes. Some macros will automatically return calculation to “Automatic” (Formulas >> Calculation Options).
 - (h) There is an option to normalize data at this stage (in the Tools button menu; purple button) to allow normalization based on (putatively) identified metabolite peaks, rather than normalizing on total LC-MS signal or peaks (which includes much more noise).
 - (i) Columns W and X allow a rapid comparison of relative intensity and *p*-value (from *t*-test) between the 2 groups specified as “control” and “treatment.” Other 2-group comparisons may be generated from the “Tools” menu.
7. Combine Pos and Neg modes (optional).

If data is acquired in switching polarity mode (or in both polarity modes separately with identical chromatography), data from each polarity must be processed separately up to this point. This step combines all data from the positive and negative mode IDEOM files. If a metabolite gives peaks in both polarities (same formula and RT within “duplicatepeaks” window setting), data from the polarity with the highest maximum intensity is selected.

8. Compare all sets.

This macro takes the data from the “identification” sheet and summarizes it in the “Comparison” sheet, providing the most convenient presentation of data for biological interpretation (examples in Tables 6, 7 and 8). An option exists to also include all significant BasePeaks (regardless of their identification status) to allow statistical analysis of unidentified metabolites. Note that the metabolite names in the “identification” sheet are dependent on the name in the “comparison” sheet, so if a peak in the “comparison” sheet is renamed to an alternative isomer, it will be reflected in the “identification” sheet. The following functions are available in the “Comparison” sheet:

- (a) Relative Intensities for all metabolites in all groups are expressed relative to the “control” group (from the Settings sheet).

- (b) P values are for unpaired t-test against “control” group; mean peak intensities, standard deviations, Relative Standards Deviations (RSD), and Fisher ratios are also provided.
- (c) Correlation Sort allows the metabolite list to be sorted by intensity correlation (across samples) relative to a specified metabolite.
- (d) The Sort function provides a quick link to sort by any column (e.g., intensity, pathway, p -value).
- (e) Excel’s native Sort and Autofilter functionality should also be used to improve data visualization.
- (f) Columns G and H can be changed to any variable in the DB sheet.
- (g) Column I can be changed to any variable in the Identification sheet.
- (h) Double-clicking in column E (name) gives an intensity plot (with standard deviations) for that metabolite. Double-clicking in column I gives individual sample intensities.
- (i) Double-clicking in column D (Isomers) gives information about the isomers in the database.
- (j) Double-clicking in column F (confidence) gives information from the “identification” sheet that helps with determination of the identity confidence.
- (k) Various summary plots are available from the orange “Graphs” button.
- (l) Various functions are available from the purple “Tools” button.
- (m) Various export formats are available from the light blue “Export” button.

9. Assign BasePeaks (optional).

This macro attempts to improve the identification of unknown BasePeaks by looking at the current “Identification” and “Rejected” sheets to update the information in the “allBasePeaks” sheet to reflect any changes since the initial identification step. It also annotates any “related peaks” if the “basepeak” itself could not be identified.

4 Other Tools Available Within IDEOM

A range of additional metabolomics data analysis tools are available within IDEOM and can be accessed through the “Tools,” “Graphs,” and “Export” menus (Table 9). Two of the more

Table 9**Additional functions are provided which may be used in any cell just like the native Excel functions**

Excel functions	Description
Exact mass	$f\!x = \text{ExactMass}(\text{formula}, \text{Clabels} \text{ (optional)}, \text{Nlabels} \text{ (optional)}, \text{Olables} \text{ (optional)}, \text{Dlabels} \text{ (optional)})$ Returns the exact mass of a given formula. Only works for the following atoms: C, H, N, O, S, P, Cl, F, I, Br, and Se. Optional arguments allow exact mass calculation for isotopically labeled compounds with the specified number of ^{13}C , ^{15}N , ^{18}O , or deuterium atoms
Mass error (ppm) calculation	$f\!x = \text{ppmcalc}(\text{mass}, \text{Theoretical mass} \text{ (optional)}, \text{formula} \text{ (optional)})$ Calculates the mass difference (in ppm) between a given mass and a theoretical formula or mass
Formula reactor	$f\!x = \text{FormulaReactor}(\text{Formula1}, \text{Formula2} \text{ (optional)}, \text{formula loss} \text{ (optional)})$ Returns the formula that results from the addition of the 2 input formulae. If the reaction involves the loss of a second product (e.g., H_2O), this must be entered as the “formulaloss.” Alternatively, this function can determine the fragment resulting from the loss of “formulaloss” from “Formula1.” only works for the following atoms: C, H, N, O, S, P, Cl, F, I, and Br. (e.g., Formula Reactor ($\text{C}_6\text{H}_{12}\text{O}_6$, H_3PO_4 , H_2O) = $\text{C}_6\text{H}_{13}\text{O}_9\text{P}$)
Formula match from exact mass	$f\!x = \text{Formula MATCH}(\text{mass}, \text{ppm}, \text{Mass list}, \text{Formula list})$ Finds a matching formula in a database of ascending masses (e.g., the DB sheet). Mass list and Formula list need to be selected as columns in a database. If two masses either side of the search mass are within the allowable ppm error the answer is <i>italicized</i>
Formula validity check	$f\!x = \text{Formulavalid}(\text{formula})$ Checks the validity of a proposed chemical formula against 5 of Kind & Fiehn’s 7 golden rules (excluding isotope and TMS rules) [23]
Theoretical isotope abundance calculator	$f\!x = \text{IsotopeAbundance}(\text{formula}, \text{atom})$ Calculates the theoretical natural isotope abundance of a specified atom in a given formula. Only works for ^{13}C , ^2H , ^{15}N , ^{18}O , ^{34}S , ^{37}Cl , ^{81}Br
Positive charge (average)	$f\!x = \text{Pos}(\text{pH}, \text{cation}, \text{p}K_{\text{a}1}, \text{p}K_{\text{a}2} \text{ (optional)}, \text{p}K_{\text{a}3} \text{ (optional)}, \text{p}K_{\text{a}4} \text{ (optional)}, \text{p}K_{\text{a}5} \text{ (optional)})$ Calculates the average number of positive charges on a molecule at a given pH, based on the formal positive charge (cation) and a list of basic $\text{p}K_{\text{a}}$ values
Negative charge (average)	$f\!x = \text{Neg}(\text{pH}, \text{anion}, \text{p}K_{\text{a}1}, \text{p}K_{\text{a}2} \text{ (optional)}, \text{p}K_{\text{a}3} \text{ (optional)}, \text{p}K_{\text{a}4} \text{ (optional)}, \dots, \text{p}K_{\text{a}8} \text{ (optional)})$ Calculates the average number of negative charges on a molecule at a given pH, based on the formal negative charge (anion) and a list of acidic $\text{p}K_{\text{a}}$ values

commonly used additional tools are for targeted analysis of a set of metabolites, and for the analysis of data from stable-isotope labeling experiments.

4.1 Targeted Analysis

Targeted analysis for specific metabolites can be undertaken from the “Targeted” sheet by following the process indicated by the buttons at the top of the page.

1. Enter the Metabolite names in Column A.
2. Click **step 2** to upload information for each metabolite from the metabolite database (DB sheet). For unique metabolites and/or masses enter the RT, formula and/or mass directly.
3. This step uses msconvert/mzMatch (through R) to process either raw, mzXML or peakML files and filters the results according to the specified masses. Parameters are taken from the Settings sheet.
4. The results from step III are searched to return results for the specific metabolites to the targeted page. Alternatively, any text file from mzMatch, or the data on the mzMatch sheet of IDEOM, can be used for this step. If multiple peaks are present it first takes the most intense peak within the RT window setting for standard RT (settings sheet; cell E23), then the most intense peak within the RT window setting for calculated RT (settings sheet; cell E24). Metabolites with no expected RT are assigned the peak with the largest intensity.
5. The targeted analysis is commonly used to obtain retention times for authentic standards, and hence there is an option to export these results to the RT calculator which is used in the regular untargeted IDEOM processing method.

4.2 Isotope Search

This tool provides an untargeted search for labeled metabolites in data that was obtained using stable-isotope labeled precursors. Run this from the “tools” menu in “comparison,” “allbasepeaks” or “Identification” sheets to search for isotopes of putatively identified compounds. The isotope search macro looks within the RT window for related peaks. The isotope search result is the relative abundance, that is, the ratio of the isotope peak to the unlabeled peak in each specified sample.

The result is shaded yellow if the relative abundance is more than 10% greater than the expected abundance of the natural isotope of the unlabeled metabolite. This macro will only find isotopes if they were imported to IDEOM from mzMatch. In some cases, isotopes are missed because the peaks were not detected by XCMS. To conduct a more comprehensive analysis of isotopes in raw data use the “Targeted Isotopes” macro in the “export” menu.

References

1. Team RC (2014) R: a language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria
2. Scheltema RA, Jankevics A, Jansen RC, Swertz MA, Breitling R (2011) PeakML/mzMatch: a file format, Java library, R library, and tool-

- chain for mass spectrometry data analysis. *Anal Chem* 83:2786–2793
3. Smith CA, Want EJ, O'Maille G, Abagyan R, Siuzdak G (2006) XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Anal Chem* 78:779–787
 4. Moffat JG, Vincent F, Lee JA, Eder J, Prunotto M (2017) Opportunities and challenges in phenotypic drug discovery: an industry perspective. *Nat Rev Drug Discov* 16:531–543
 5. Gamo FJ, Sanz LM, Vidal J, de Cozar C, Alvarez E, Lavandera JL, Vanderwall DE, Green DV, Kumar V, Hasan S et al (2010) Thousands of chemical starting points for antimalarial lead identification. *Nature* 465:305–310
 6. Hovlid ML, Winzeler EA (2016) Phenotypic screens in antimalarial drug discovery. *Trends Parasitol* 32:697–707
 7. Creek DJ, Chua HH, Cobbold SA, Nijagal B, Macrae JI, Dickerman BK, Gilson PR, Ralph SA, McConville MJ (2016) Metabolomics-based screening of the malaria box reveals both novel and established mechanisms of action. *Antimicrob Agents Chemother* 60 (11):6650–6663
 8. Allman EL, Painter HJ, Samra J, Carrasquilla M, Llinas M (2016) Metabolomic profiling of the malaria box reveals antimalarial target pathways. *Antimicrob Agents Chemother* 60:6635–6649
 9. Kwon YK, Lu W, Melamud E, Khanam N, Bognar A, Rabinowitz JD (2008) A domino effect in antifolate drug action in *Escherichia coli*. *Nat Chem Biol* 4:602–608
 10. Vincent IM, Creek DJ, Burgess K, Woods DJ, Burchmore RJ, Barrett MP (2012) Untargeted metabolomics reveals a lack of synergy between nifurtimox and eflornithine against *Trypanosoma brucei*. *PLoS Negl Trop Dis* 6:e1618
 11. Zampieri M, Szappanos B, Buchieri MV, Trauner A, Piazza I, Picotti P, Gagneux S, Borrell S, Gicquel B, Lelievre J et al (2018) High-throughput metabolomic analysis predicts mode of action of uncharacterized antimicrobial compounds. *Sci Transl Med* 10: eaal3973
 12. Spangenberg T, Burrows JN, Kowalczyk P, McDonald S, Wells TN, Willis P (2013) The open access malaria box: a drug discovery catalyst for neglected diseases. *PLoS One* 8:e62906
 13. Trager W, Jensen J (1976) Human malaria parasites in continuous culture. *Science* 193:673–675
 14. Chambers MC, Maclean B, Burke R, Amodei D, Ruderman DL, Neumann S, Gatto L, Fischer B, Pratt B, Egertson J et al (2012) A cross-platform toolkit for mass spectrometry and proteomics. *Nat Biotechnol* 30:918–920
 15. Tautenhahn R, Bottcher C, Neumann S (2008) Highly sensitive feature detection for high resolution LC/MS. *BMC Bioinformatics* 9:504
 16. Creek DJ, Jankevics A, Burgess KE, Breitling R, Barrett MP (2012) IDEOM: an excel interface for analysis of LC-MS-based metabolomics data. *Bioinformatics* 28:1048–1049
 17. Sansone SA, Fan T, Goodacre R, Griffin JL, Hardy NW, Kaddurah-Daouk R, Kristal BS, Lindon J, Mendes P, Morrison N et al (2007) The metabolomics standards initiative. *Nat Biotechnol* 25:846–848
 18. De Livera AM, Dias DA, De Souza D, Rupasinghe T, Pyke J, Tull D, Roessner U, McConville M, Speed TP (2012) Normalizing and integrating metabolomics data. *Anal Chem* 84:10768–10776
 19. Biagini GA, Fisher N, Shone AE, Mubaraki MA, Srivastava A, Hill A, Antoine T, Warman AJ, Davies J, Pidathala C et al (2012) Generation of quinolone antimalarials targeting the *Plasmodium falciparum* mitochondrial respiratory chain for the treatment and prophylaxis of malaria. *Proc Natl Acad Sci U S A* 109:8298–8303
 20. Ganesan SM, Morrissey JM, Ke H, Painter HJ, Laroiya K, Phillips MA, Rathod PK, Mather MW, Vaidya AB (2011) Yeast dihydroorotate dehydrogenase as a new selectable marker for *Plasmodium falciparum* transfection. *Mol Biochem Parasitol* 177:29–34
 21. Cobbold SA, Chua HH, Nijagal B, Creek DJ, Ralph SA, McConville MJ (2016) Metabolic dysregulation induced in *Plasmodium falciparum* by dihydroartemisinin and other front-line antimalarial drugs. *J Infect Dis* 213:276–286
 22. Creek DJ, Jankevics A, Breitling R, Watson DG, Barrett MP, Burgess KE (2011) Toward global metabolomics analysis with hydrophilic interaction liquid chromatography-mass spectrometry: improved metabolite identification by retention time prediction. *Anal Chem* 83:8703–8710
 23. Kind T, Fiehn O (2007) Seven Golden rules for heuristic filtering of molecular formulas obtained by accurate mass spectrometry. *BMC Bioinformatics* 8:105



Chapter 22

Analyzing Metabolomics Data for Environmental Health and Exposome Research

Yuping Cai, Ana K Rosen Vollmar, and Caroline Helen Johnson

Abstract

The exposome is the cumulative measure of environmental influences and associated biological responses across the life span, with critical relevance for understanding how exposures can impact human health. Metabolomics analysis of biological samples offers unique advantages for examining the exposome. Simultaneous analysis of external exposures, biological responses, and host susceptibility at a systems level can help establish links between external exposures and health outcomes. As metabolomics technologies continue to evolve for the study of the exposome, metabolomics ultimately will help provide valuable insights for exposure risk assessment, and disease prevention and management. Here, we discuss recent advances in metabolomics, and describe data processing protocols that can enable analysis of the exposome. This chapter focuses on using liquid chromatography–mass spectrometry (LC-MS)-based untargeted metabolomics for analysis of the exposome, including (1) preprocessing of untargeted metabolomics data, (2) identification of exposure chemicals and their metabolites, and (3) methods to establish associations between exposures and diseases.

Key words Exposome, Human health, Untargeted metabolomics, Data processing

1 Introduction

In 2002, the first genome-wide association study (GWAS) was published, analyzing genetic susceptibility to myocardial infarction [1]. One year later, the Human Genome Project was completed [2]. Since then, we have witnessed the increasing success of GWAS to show relationships between base-pair/gene patterns in genomic loci and many diseases [3], including autoimmune diseases [4], breast cancer [5], colorectal cancer [6], schizophrenia [7], and type 1 and 2 diabetes [8, 9]. However, *genetic linkage* association studies have only been able to explain a small fraction of the estimated heritability of complex diseases. Instead, gene–environment interactions appear to underlie the etiology of major non-communicable chronic diseases such as cancer, diabetes, and vascular and neurodegenerative diseases. A meta-analysis of twin

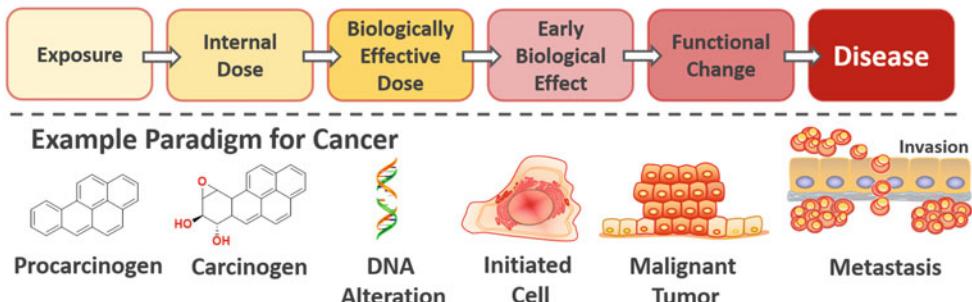
studies over the past 50 years found that heritability across all human traits is 49% [10]. These results indicate that nongenetic influences such as environmental and lifestyle risk factors could be related to different phenotypes and underlie the etiology of many diseases.

The importance of the environment as a key determinant of health, both separate from and in interaction with genetics, has motivated large-scale collaborative efforts to develop methods for comprehensively measuring environmental exposures [11–13]. While particulate matter (ambient particles and household smoke), smoking, nutrition (diets high in cholesterol, salt, and sugar), and occupational exposures are linked to about 50% of deaths globally, there are many diseases for which the exposure risk factors remain unknown [14]. In addition, the nature of these risks is different for each person due to individual susceptibility, which includes host factors such as differences in metabolism, microbiome and endogenous hormones, and social determinants of health like socioeconomic status or neighborhood. All of these factors fluctuate over the life course, with susceptibility increasing during critical windows, such as in pregnancy or times of ill health [15], making it difficult to attribute a specific exposure as causal of a particular outcome. The exposome, or “the cumulative measure of environmental influences and associated responses throughout the lifespan,” encompasses these diverse and changing external and internal exposures [16]. The exposome concept was developed in part to address the need for more comprehensive exposure assessment methods, and to provide a framework for the measurement and analysis of an individual’s exposure portfolio over an entire lifetime [16–19].

Developing methods for measuring the exposome remains a challenge given the difficulty of cataloguing an individual’s exposome at even a single point in time. High-throughput methods enable a systems-level evaluation of genes (genomics), gene expression (transcriptomics), proteins (proteomics), and metabolites (metabolomics), offering a comprehensive view of the biological changes that can occur in an individual after a specific exposure. Multi-omics approaches which combine these data can further help identify downstream chemical changes that contribute to an exposure phenotype or exposotype, or the accrued biological changes within a system that has undergone an exposure event (Fig. 1).

Exposures can thus be linked to disease end points through identification of associated biomolecular changes, or biomarker measurements [20]. Compared to other -omics technologies, metabolomics is advantageous for studying the exposome because the metabolome profiles both the exposure and the biological response. Peaks from environmental exposures are routinely observed in untargeted mass spectrometry data. Indeed, the increased incorporation of metabolomics into exposure research

a) Environmental Health Paradigm



b) Exposure and the Central Dogma of Molecular Biology

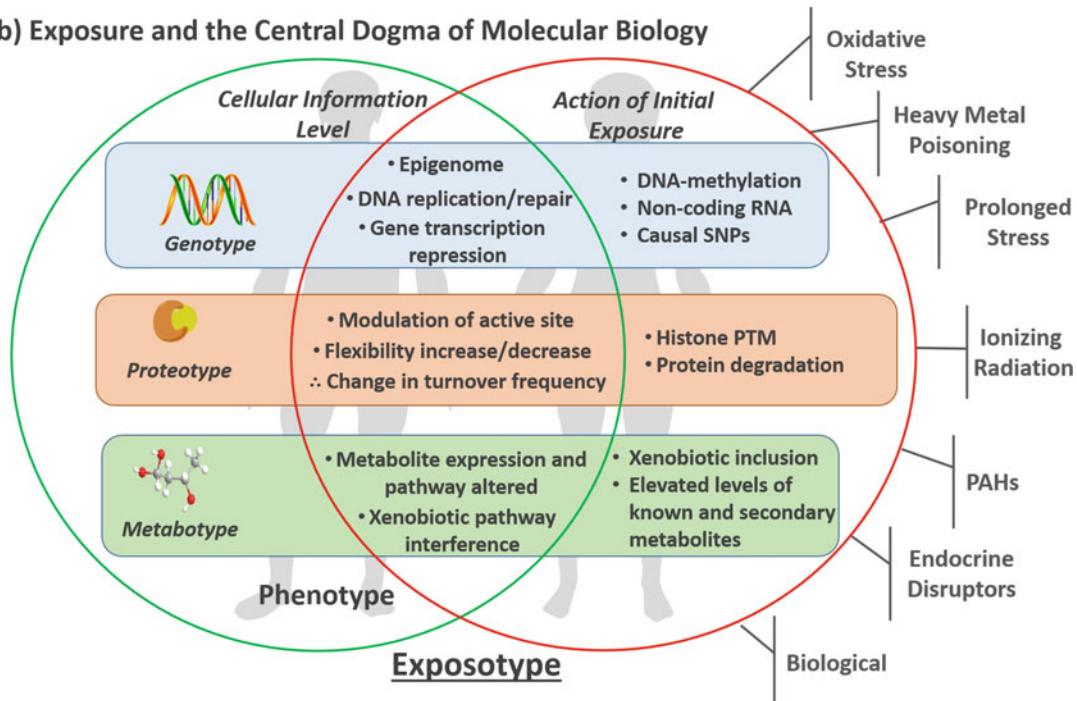


Fig. 1 Exposotype, the exposure phenotype. (a) The environmental health paradigm: exposures modify biological molecules and lead to disease. (b) Possible actions of environmental exposures on DNA, proteins, and metabolites. Please note that metabolites are not in the original Central Dogma of Molecular Biology, but informative of biological functions at both cellular and organismal levels. (Figure reproduced from [19])

can be visualized through the steady annual increase in the number of publications in PubMed from 2005, when the term “exposome” was coined, to 2019 (Fig. 2).

The metabolome, or the pool of low molecular-weight molecules in a sample, encompasses both endogenous and externally derived exposure metabolites. Endogenous metabolites comprise approximately 3000 substrates and products, the concentrations of which are determined by dietary intake, de novo synthesis, and actions of genome-encoded protein enzymatic reactions that

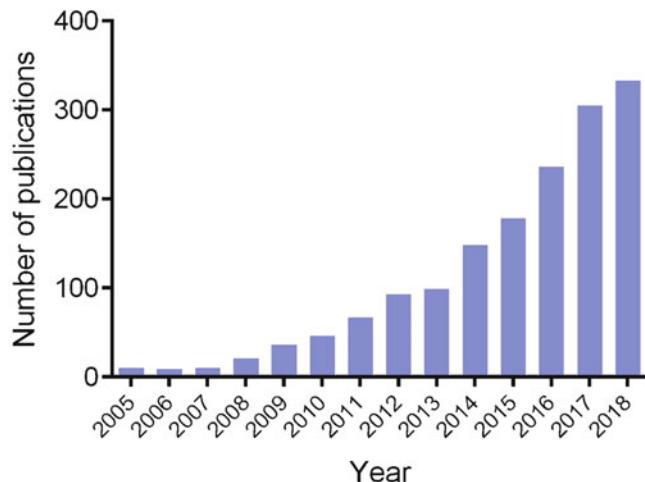


Fig. 2 Number of publications searched by keywords “exposure, metabolomics” in PubMed

convert these metabolite substrates into products. These metabolites are required to serve as intermediates of metabolic pathways that produce energy and molecules for cell growth, essential biological functions that can be monitored as surrogates of aberrant biochemistry if these processes go awry. Metabolites derived from specific external exposures include environmental chemicals (e.g., pesticides, plasticizers, heavy metals, food contaminants, deodorants, cosmetics) and pharmaceutical drugs; these exposures and their biotransformed phase I and phase II metabolites can be identified and quantified simultaneously through metabolome profiling. Untargeted metabolomics, the measurement of the metabolome at a systems-level, allows for a comprehensive, agnostic, and high-throughput capture of low molecular-weight metabolites within a biological sample [21, 22]. The nature of the approach, along with the continued advancement of high-resolution platforms such as ultraperformance liquid chromatography–mass spectrometry (UPLC-MS) and ^1H nuclear magnetic resonance (NMR) spectroscopy, offers advantages for exposomics. Simultaneous profiling of exposures and perturbed metabolic pathways or networks, along with a deep breadth of coverage, can establish the connection between exposures, biological responses, and phenotypes [23–25]. Implementing metabolomics for large-scale and systematic exposome research presents a number of challenges, especially with respect to data analysis [19, 26]. Typically, an LC-MS-based untargeted metabolomics experiment can measure more than 30,000 features/variables [27]. This results in a large, high-dimensional dataset that requires nontrivial bioinformatics analysis to identify environmental chemicals, endogenous metabolites, and the enriched metabolic pathways linking exposures to human health outcomes.

Challenges from both analytical and statistical perspectives must be addressed when analyzing metabolomics data for environmental health and exposome research, and have been detailed in multiple reviews [28–30]. From an analytical perspective, challenges include selection of appropriate data preprocessing procedures to quantify the relevant mass spectral features of exposures and metabolites; structural elucidation, or identification of external exposures and endogenous metabolites; and discovery of metabolic networks affected by specific exposures [17]. From a statistical perspective, notable challenges include not only assessing associations between a single exposure and an outcome, but accounting for mixtures of external and internal environmental exposures that change over time [31]; addressing complex correlation structures of exposome data as strong correlations between exposures are common when many exposures are measured [30, 32]; and difficulty differentiating between normal variation in outcomes and environmentally induced changes, since exposures are dynamic and there is no “reference exposome” against which to compare findings [29]. In this chapter, we describe data processing protocols that can be used to analyze metabolomics data in exposomics research, focusing on the LC-MS-based untargeted discovery of metabolites and metabolic pathways which link environmental chemical exposures to adverse human health outcomes.

2 Methods

2.1 Metabolomic Technologies for Exposome Research

2.1.1 High-Resolution Metabolomics (HRM)

Choosing the most suitable analytical platform is a key foundation for exposome research. Developments in high-resolution metabolomics (HRM) have accelerated exposome research due to their improved ability to profile a broad range of endogenous metabolites and environmental chemicals simultaneously. Modern time-of-flight (ToF) and Fourier-transform (FT) mass spectrometers confer high resolution, high accuracy, and provide the most sensitive analysis for untargeted metabolomics methodologies [22]. Benefiting from a high scan speed, ToF mass spectrometers are the most commonly used instruments for high-throughput metabolite profiling. FT mass spectrometers such as the Orbitrap™, however, have high mass resolution and mass accuracy which are not affected by low ion abundance. Therefore, they are superior for metabolite quantification, especially for environmental chemicals with a low abundance and a wide dynamic range in biological samples [33]. The data output from these HRM platforms contains a wealth of information regarding enriched metabolic pathways which can facilitate the discovery of relationships between environmental exposures and adverse health outcomes.

2.1.2 Measurement of Low Level of Exposures

The wide dynamic range of chemical compounds that are present in human samples presents a major challenge for accurate quantification, which is essential for distinguishing biological differences between groups. Typically, blood concentrations of environmental chemicals (fmol to μ mol) are 1000 times lower than endogenous metabolites (nmol to mmol) and metabolites from nutritional or pharmaceutical drug sources [34]. Biotransformed environmental chemicals produced through endogenous enzymatic reactions may also be present at very low levels. Electrospray ionization (ESI) coupled with LC is the most commonly applied technique in MS-based metabolomics research [35]. In order to quantify low-abundance environmental chemicals such as polychlorinated biphenyls [36], polycyclic aromatic hydrocarbons [37], and polybrominated diphenyl ethers [38, 39], other ionization sources are required due to their hydrophobic chemical structures. Gas-chromatography (GC) with electron impact (EI) ionization is routinely used for the analyses of these common environmental pollutants [40–42]. LC hybrids with soft ionization approaches such as atmospheric pressure chemical ionization (APCI) and atmospheric pressure photoionization (APPI) can also readily measure these low-level hydrophobic chemicals [43]. In addition, APPI uses a charge carrier (acetone or toluene) which increases sensitivity. Ion mobility LC-MS analysis has shown promise for the characterization of structurally similar environmental chemicals, due to increased resolving power and improved signal-to-noise ratio [43, 44].

2.2 Metabolomics Data Processing for Exposome Research

2.2.1 Data Preprocessing

An MS-based metabolomics experiment can result in a high-dimensional dataset containing information on tens of thousands of ions. Thus, the primary task for data preprocessing is to extract mass spectral ions that are related to signals from environmental chemicals and endogenous metabolites. In recent years, much effort has been devoted to developing algorithms and bioinformatics tools for peak detection and peak annotation. These tools have been developed in different programming languages, such as XCMS [45] using R, MZmine [46] using Java, and MetAlign [47] using C++. In addition, instrument vendors have their own proprietary software that can be used. A recent review provides a comprehensive list of current software available for metabolomics analysis [48].

Some metabolomics data preprocessing decisions such as the type of normalization approach can be influenced by exposome study design and specimen collection considerations. For example, because of the impossibility of measuring the entire life course exposome, many studies focus on exposure measurements and specimen collection during critical windows for the health outcomes being investigated [29, 49]. These critical windows may have unique physiologic signatures that must be considered when

selecting normalization approaches. A prime example is pregnancy, which is a common window targeted by exposome studies exploring early-life exposures [30]. However, the normalization of urine metabolites is unreliable when using the gold standard creatinine, due to the normal physiologic changes of pregnancy [50]. Instead, approaches like specific gravity or probabilistic quotient normalization may be more suitable for metabolomics analyses of urine during this critical window [51, 52].

2.2.2 Characterization and Identification of Exposures and Metabolites

A prerequisite for identifying causal pathways linking exposures to health outcomes is to first identify the statistically significant mass spectral signals within the metabolomics data. Since both exposures and endogenous metabolite signals will be captured at the same time in a data set, such chemical complexity can make it difficult to accurately characterize the relationships between exposures and metabolic consequences. One of the most significant analytical bottlenecks of metabolomics remains the structural elucidation of chemicals. Fortunately, the availability of bioinformatic tools and chemical databases that incorporate environmental chemicals and other exposures can help to overcome this challenge. For example, the R package Collection of Algorithms for Metabolite pRofile Annotation (CAMERA), which is embedded in the XCMS software, organizes isotopic peaks, adducts, and fragment ions by grouping ions based on retention time, integration of peak shape, and correlation of peak intensity. This reduces feature redundancy by ~50% and helps with metabolite identification [53]. xMSannotator is an R package and automated computational workflow to annotate ions in untargeted metabolomics. This annotation tool uses an integrative multicriteria scoring algorithm to assign identification to an ion, accounting for correlation strength, difference in retention time, number of matching adducts, and isotopes to categorize database matches containing information about both environmental chemicals and endogenous metabolites (ChemSpider, KEGG, HMDB, T3DB, and LipidMaps) into different confidence levels [54].

However, to obtain absolute confidence regarding metabolite identification, MS/MS data need to be acquired and the spectra compared either against metabolite databases, or preferably to standards (to also obtain matching retention time information). In the following sections, various databases developed for identification and characterization of environmental chemicals and endogenous metabolites are introduced and discussed, respectively.

To better characterize environmental chemicals from metabolomics data, several useful databases with MS/MS information have been developed and are readily available for identification of environmental metabolites in exposome research (Table 1). The METLIN database was recently modified to incorporate >700,000 chemicals from the United States Environmental

Table 1
Databases of environmental metabolites for exposome research

Database	Number of exposures or metabolites	Content	URL
METLIN Exposome database	>700,000	Chemical information, MS/MS spectra	https://metlin.scripps.edu/
Comparative Toxicogenomics Database (CTD)	130,796	Chemical–gene/protein interactions, chemical/gene–disease relationships	http://ctdbase.org/
Drugbank	11,926	Drugs, drug targets, MS/MS spectra	https://www.drugbank.ca/
Hazardous Substances Data Bank (HSDB)	6016	Chemical information, environmental fate, toxicity	https://toxnet.nlm.nih.gov/newtoxnet/hsdb.htm
Toxin Exposome Database (T3DB)	3678	Both toxin and toxin target, gene expression data, MS/MS spectra	http://www.t3db.ca/
Exposome-Explorer	876	Dietary and pollutant biomarkers, with concentration and correlation values	http://exposome-explorer.iarc.fr/

Protection Agency's (EPA) Distributed Structure-Searchable Toxicity (DSSTox) database, which includes toxicants and xenobiotics [55]. Furthermore, the database consists of a number of phase I and II mammalian biotransformation products that can help to investigate xenobiotic metabolism for exposome research, and can be better observed by LC-ESI-MS compared to the nontransformed parent compounds, which cannot be ionized through this approach. In addition, the METLIN database contains MS/MS spectra for many metabolites listed on the database, enabling greater confidence in metabolite identification. Moreover, the database is integrated into the XCMS Online data processing platform, enabling instant putative metabolite identification and toxicological pathway analysis, facilitating both metabolite identification and knowledge of altered metabolic pathways [56].

The Toxin Exposome Database (T3DB) houses 3678 toxins, with approximately one-third having MS/MS spectrum information [57]. T3DB is a comprehensive exposome resource that contains information on toxins, toxin targets, and gene expression datasets, with detailed information on chemical properties, toxic concentration in human biofluids, toxic effects, and mechanisms. The informative descriptions of each exposure can be useful to epidemiologists and toxicologists in the exposome research field. Another source of externally derived chemicals is pharmaceutical

drugs. The database Drugbank was developed to help identify these compounds, and contains information on 12,064 drugs, also with mass spectral data [58].

Other environmental chemical-specific databases are important for exposure characterization, but do not contain the MS/MS information required for metabolite identification. For example, the Comparative Toxicogenomics Database (CTD) is a large online database containing concentration values of environmental chemicals in blood, urine, and other biospecimens extracted from the scientific literature [59]. CTD includes information for chemicals, relevant genes, and proteins on toxicology, sequence, reference, species, and microarray, along with manually curated information about chemical–gene/protein interactions, and chemical–disease and gene–disease relationships. The Hazardous Substances Data Bank (HSDB) is managed by the US National Institutes of Health and focuses on the toxicology of potentially hazardous chemicals [60]. It contains information on environmental fate, human exposure, detection methods, and regulatory requirements, and is updated several times a year by a Scientific Review Panel. Exposome-Explorer, which currently houses 876 exposures, is the first database dedicated to exposure biomarkers. This database was constructed by consolidating exposure biomarker data scattered throughout the literature [61]. To identify metabolic pathways modulated by environmental chemicals, the structural elucidation of metabolites is of critical importance. For MS-based metabolomics data, the robust identification of metabolites is obtained by comparing the MS¹ mass-to-charge (m/z) ratio (the initial/precursor m/z spectrum), retention time, and MS/MS spectra to standards. Spectral similarity matching to standard metabolite libraries housed on metabolite databases has helped expedite this process.

Over the last 20 years, several freely available databases have been developed to facilitate identification of metabolites in metabolomic experiments, including METLIN [62], Human Metabolome Database (HMDB) [63], MassBank [64], and LipidMaps [65]. METLIN has a dual function in metabolomics, as it is useful for identifying both environmental exposures and endogenous metabolites. These databases were generated using pure standard compounds to provide accurate identification of metabolites from biological samples. Most of these repositories allow for a comparison of MS/MS data from a research sample to MS/MS data on the database for metabolite identification. Other platforms such as MyCompoundID [66] enable identification of unknown metabolites by comparison to a library of predicted MS/MS data from known metabolites, and from those formed by biotransformation reactions. The Global Natural Products Social Molecular Networking (GNPS) [67] resource is another platform for the storage, analysis, and sharing of MS/MS spectral data; it enables researchers

to annotate MS/MS data, and compare it to publicly available data. In addition, novel tools such as Mummichog [68] and PIUMet [69] provide putative dysregulated metabolic pathway information through bypassing metabolite identification and assessing the inter-connectivity of metabolites in known metabolic pathways. To increase accuracy in the discovery of dysregulated metabolic pathways, Metabolite Identification and Dysregulated Network Analysis (MetDNA) software was recently developed by incorporating MS/MS spectral matching followed by dysregulated network analysis (<http://metdna.zhulab.cn>). Of note, metabolite identification is optimized through the implementation of a metabolic reaction network (MRN)-based recursive algorithm. These approaches could advance the discovery of metabolic pathways affected by environmental exposures. Most of these resources and methods are described in depth in other chapters of this book.

2.2.3 Analyzing Associations Between the Exposome and Health Outcomes

Across the life course, an individual will encounter a myriad of environmental exposures which can influence human health, either individually or synergistically. A notable challenge of implementing the exposome is understanding the direct relationship between one or more exposures and a disease outcome, and identifying the factors that link environmental chemicals to adverse health outcomes. From a data analysis perspective, the core data processing step essential for the exposome lies in establishing the relationship between the exposure and health outcome.

Conventional biomonitoring of environmental exposures targets the measurement of a limited number of chemicals in biospecimens such as blood and urine. Establishing the effects of an exposure on human health is therefore restricted to a relatively small exposure panel. Traditional environmental epidemiology reflects this biomonitoring approach, as it assesses the association between a single exposure and a single response. However, these methods can result in biased and fragmentary pictures of environment–health relationships, since environmental exposures often occur as mixtures and change frequently [29]. The exposome more realistically represents the many exposures an individual will encounter throughout life, but the enormous quantity of data collected in exposome research presents challenges for analysis and interpretation.

To analyze the thousands of features typical of LC-MS experiments, metabolomics-based exposome research often employs agnostic or hypothesis-free statistical methods, examining data for patterns and correlations between exposures/metabolites and outcomes [30]. These findings can later be validated through targeted methods to describe potential mechanisms. Proponents of agnostic methods see such techniques as allowing previously unknown relationships to emerge given the complexity of exposures and outcomes, while critics warn such approaches have high chances of

false discovery, reverse causation, and spurious findings [28, 70]. Even hypothesis-free analytic methods should continue to employ core environmental epidemiology practices to assess and adjust for confounders, selection bias, measurement error, and exposure misclassification, and testing for multiple comparisons must be controlled for in analysis [29, 30, 70].

Another challenge in analyzing associations between the exposome and health outcomes is that when many exposures are measured, it is assumed that there will be strong correlations and collinearity between exposures. For example, cumulative exposure to radon, γ -ray, and long-lived radionuclides (LLR) was shown to be associated with a significant risk of lung cancer [71]. However, the Pearson correlation coefficient between radon and γ -ray exposure was 0.60 ($p < 0.001$), and the correlation between radon and LLR exposure 0.52 ($p < 0.001$) in the study. This collinearity between exposures limits our ability to elicit the effect on human health specific to each exposure. Describing the data's correlation structure and key exposures or subject attributes responsible for variation allows researchers to more precisely identify indicator exposures, which may represent groups of exposures [30]. Such indicator exposures could guide future research and measurement, or have implications for regulatory policy. For example, a study of the exposome during pregnancy integrated biomonitoring, questionnaire, environmental monitoring, and geospatial modeling data from 728 pregnant women to determine the relationships between 81 environmental exposures [30]. Strong correlations were observed within certain "families" of exposure, such as noise, water pollutants, perfluoroalkyl substances (PFAS), and air pollutants, while weak correlations were observed for groups including home and built environment exposures, metals, and phthalates, reflecting more diverse sources of exposure.

Many univariate and multivariate statistical methodologies exist to address these and other statistical challenges of exposome research, allowing for relationships between environmental exposures and health outcomes to be more effectively established. Regression methods are most commonly used in this field. To assess the statistical significance of the association between a single environmental exposure and the target biological response or health outcome, linear regression can be applied when the outcome is continuous, and logistic regression can be applied when the outcome is binary. This can be repeated for each environmental exposure measured within a cohort, and correction for multiple comparisons is performed to reduce false positive results. To obtain robust findings, the results should be validated using other datasets or cohorts. This was how the initial "environment-wide association studies" or "exposome-wide association studies" (EWAS) were conducted [72]. Because EWAS is a hypothesis-free, agnostic method of evaluating associations between exposures and health outcomes, it can help uncover novel associations [73–75].

A common problem of high-dimensional data sets is having a larger number of predictors than the sample size [76]. LASSO (least absolute shrinkage and selection operator) is a penalized regression model and variable selection method that identifies the most informative predictors for inclusion in the final model from a larger set of covariates. LASSO improves prediction precision by forcing the sum of the regression coefficients to be shrunk to zero, with the least informative coefficients given a value of zero [77]. For LASSO, the number of nonpenalized variables is limited by the number of observations. Elastic net (ENET) is another penalized regression model, which combines the advantages of ridge and LASSO regression [78]. Ridge regression effectively accommodates correlated variables, while LASSO selects variables through shrinkage [79]. The elastic net approach enables the selection of a set of correlated exposures, with the interpretability of a simple regression model. Bayesian methods are another venue for variable selection. Graphical Unit Evolutionary Stochastic Search (GUESS) and its wrapper tool R2GUESS allows for model parameterization and handling of up to several thousand of observations, hundreds of thousands of predictors, and a few outcomes simultaneously, taking advantage of graphic hardware capabilities [80, 81].

For exposome research, where the number of covariates is greater than the number of observations, dimension reduction methods such as principal components analysis (PCA) [82], canonical correlation analysis (CCA) [83], and partial least-squares (PLS) [84] are approaches for reducing the redundancy and complexity of results. By using dimension reduction methods, several components consisting of multiple covariates/predictors are defined iteratively to explain as much of the covariance between the exposures and the health outcome as possible. Table 2 lists some commonly used statistical methods and tools to assess exposome-health associations.

Table 2

Common informatics tools freely available as R packages for analyzing associations between the exposome and health outcomes

Statistical method	R package
Linear regression and logistic regression	<i>Stats and rexposome</i>
LASSO and elastic net	<i>Glmnet</i>
Bayesian variable selection regression of multivariate responses	<i>R2GUESS</i>
Principal components analysis (PCA)	<i>Prcomp</i>
Canonical correlation analysis (CCA)	<i>CCA</i>
Partial least squares (PLS)	<i>Pls</i>

2.3 Multi-omics Integration for Exposome Research

The primary goal of exposome research is to unravel biological responses to environmental exposures, ultimately linking exposures to health outcomes. Multiple layers of physiologic responses can be stimulated by environmental exposures, including DNA expression (genome), RNA expression (transcriptome), proteins (proteome), metabolites (metabolome), and microbiota (microbiome). Therefore, the diverse and rich information obtained from genomics, epigenomics, proteomics, transcriptomics, and metabolomics can be exploited to capture comprehensive biological responses to exposures. A recent example demonstrated the strength of -omics integration in environmental health through the use of transcriptomics and metabolomics to reveal the neurotoxicologic mechanisms of a herbicide (paraquat) and a fungicide (maneb) [85].

-Omics data are high-dimensional, often with thousands or tens of thousands of data points, presenting major challenges for multi-omics data integration that aims to identify exposure-mediated biomarkers. Both unsupervised and supervised algorithms form the basis of multi-omics analyses. Based on the statistical methodologies utilized, integration strategies can be categorized into matrix factorization methods, Bayesian methods, network-based methods, multiple kernel learning, and multistep-based methods. As an example, Guida et al. applied a two-step strategy to integrate epigenomics and transcriptomics data to characterize the biologically relevant long-term markers of tobacco smoke exposure [86]. Initially, the epigenomics data were regressed against the outcome, resulting in a set of outcome-associated variables (smoking-related CpG sites). Then, another regression was performed on the smoking-related epigenomics against transcriptomics data. In this study, the gene *LRRN3* was identified as a promising biomarker in which both methylation and gene expression were associated with tobacco smoke exposure. The topic of multi-omics integration is covered in more detail in Chapter 23 of this book.

Multi-omics technologies allow for agnostic analyses of the biological effects of exposure in a high-throughput manner, and integrating these data can help identify and reinforce molecular biomarkers of chemical exposures, as discussed above. However, several challenges need to be addressed in the near future for better characterization of the exposome. First, methods for efficiently extracting useful information need to be developed, as -omics approaches generate a large amount of data accompanied by redundancy. For example, it is typical for an untargeted metabolomics dataset with over 25,000 measured metabolic features to harbor fewer than 1000 unique metabolites [87]. Additionally, there are both data storage and data processing challenges associated with the large data sets generated by multi-omics experiments. This challenge can be partially addressed by improving the computational power and performance efficiency of new algorithms. Data

sharing for multiple data types is another critical aspect for consideration. At present, domain-specific data repositories are available for data sharing, including repositories for EWAS data sets (<https://nhanes.hms.harvard.edu/transmart/datasetExplorer/index>), metabolomics data (Metabolomics Workbench, <https://www.metabolomicsworkbench.org>; MetaboLights, <https://www.ebi.ac.uk/metabolights/>), and genomics data (Gene Expression Omnibus, <https://www.ncbi.nlm.nih.gov/geo/>). Future directions for multi-omics data sharing could include platforms integrating data sets on specific health outcomes.

3 Conclusions and Perspectives

Metabolomics is a vital tool for exposome research, capable of profiling both environmental chemicals and biological effects through the measurement of metabolites. In this chapter, we introduced HRM technologies and presented data processing procedures for using metabolomics data in exposome research, including methods for processing raw MS data, identifying environmental chemicals and endogenous metabolites, and analyzing associations between exposures and health outcomes. We anticipate that exposome research will flourish by integrating metabolomics and other -omics approaches, especially since several tools for data processing with application to the exposome paradigm are now available.

A series of different challenges still remain for exposome research, including analytical challenges and difficulties in data processing. From an analytical perspective, the broad dynamic range of environmental chemicals in biological specimens (e.g., blood and urine) prompts the need for new detection technologies using reconfigured mass spectrometers coupled with chromatographic separation (for example, GC/APCI/MS/MS techniques) [88]. Even though several publicly available databases now exist that house environmental chemicals, the accurate identification of these metabolites captured by metabolomics data is an urgent challenge that needs to be addressed. One promising solution to facilitate structural elucidation is to first identify the chemical class to which exposures and metabolites belong. For example, differential chemical isotope labeling LC-MS methods have been developed to profile high coverage of the submetabolome, as with the danslyation labeling method for the hydroxyl submetabolome [89]. Other informatics tools such as MetFamily [90] and CluMSID [91] apply unsupervised clustering methods to the analysis of MS/MS spectra in untargeted metabolomics. Both approaches found structural-related families to aid in compound identification. MS2LDA, utilizing the Latent Dirichlet Allocation (LDA) algorithm, was recently shown to decompose molecules into

Mass2Motifs (substructures) for molecular grouping [92]. The updated version MS2LDA+ can support the discovery and comparison of structural families across multiple samples [93].

The greatest challenges often present the greatest opportunities. With advances in personalized medicine, exposome research may play a vital role in improving medical treatments for patients, as human variability in response to exposures can impact the treatment of diseases, especially for complex cancer [94]. In addition to individual-specific biological signatures originating from inherited genes, transcripts, proteins, and metabolites, externally derived environmental exposures also contribute to human variation. Exposures can modulate host biological processes, or use the same phase I and II biotransformation enzymes, thus potentially affecting pharmaceutical drug efficacy and toxicology. Therefore, exposome studies evaluating the effects of environmental factors on human health have the potential to contribute to the development of individualized health care and personalized medicine. Single -omics data provides limited information on the pathogenesis of diseases, but by combining multiple -omics data sources (e.g., genomics, transcriptomics, proteomics, and metabolomics), exposome research can support a comprehensive understanding of the molecular profiles of each individual. One example project, Integrated Personal Omics Profiling (iPOP), aims to integrate a wide range of data to better characterize the molecular phenotypes underlying a participant's individual health status, drawing on data sources including the genome, transcriptome, proteome, and metabolome, along with information on cytokines, immune cells, clinical laboratory test results, diet, stress, and activity levels. Future individualized treatment may take into account exposome information such as exposures to environmental chemicals, dietary influences, and lifestyle.

The exposome framework encompasses not only internal exposures from metabolism and the microbiome but also external exposures ranging from socioeconomic status and neighborhood to air quality and chemical exposures. Therefore, the exposome intersects with structural and population-level determinants of health. While metabolomics research in the context of the exposome is oriented around generating individual-level data about biologically relevant exposures, innovative exposome study designs are starting to connect longitudinal and hierarchical nested exposure measures, such as measures at the community, family, and individual levels. Such studies will begin to describe how exposures change and are interrelated at multiple scales and across time in the life course [29]. This research has the potential to inform not only individual-level clinical decisions, but also risk assessment, exposure prevention, and policy interventions [95]. Metabolomics is embedded in the exposome paradigm, with the capacity to quantify both environmental exposures and biological responses, thus offering exciting tools for environmental health and exposome research.

References

1. Ozaki K, Ohnishi Y, Iida A, Sekine A, Yamada R, Tsunoda T, Sato H, Sato H, Hori M, Nakamura Y, Tanaka T (2002) Functional SNPs in the lymphotoxin-alpha gene that are associated with susceptibility to myocardial infarction. *Nat Genet* 32(4):650–654
2. Collins FS, Lander ES, Rogers J, Waterston RH, Consortium IHGS (2004) Finishing the euchromatic sequence of the human genome. *Nature* 431(7011):931–945
3. Buniello A, MacArthur JAL, Cerezo M, Harris LW, Hayhurst J, Malangone C, McMahon A, Morales J, Mountjoy E, Sollis E, Suveges D, Vrousgou O, Whetzel PL, Amode R, Guillen JA, Riat HS, Trevanion SJ, Hall P, Junkins H, Flicek P, Burdett T, Hindorff LA, Cunningham F, Parkinson H (2019) The NHGRI-EBI GWAS Catalog of published genome-wide association studies, targeted arrays and summary statistics 2019. *Nucleic Acids Res* 47(D1):D1005–D1012
4. Eyre S, Worthington J (2014) Take your PICS: moving from GWAS to immune function. *Immunity* 41(6):883–885
5. Cuzick J, Brentnall A, Dowsett M (2017) SNPs for breast cancer risk assessment. *Oncotarget* 8(59):99211–99212
6. Yao L, Tak YG, Berman BP, Farnham PJ (2014) Functional annotation of colon cancer risk SNPs. *Nat Commun* 5:5114
7. Need AC, Ge D, Weale ME, Maia J, Feng S, Heinzen EL, Shianna KV, Yoon W, Kasperaviciute D, Gennarelli M, Strittmatter WJ, Bonvicini C, Rossi G, Jayathilake K, Cola PA, McEvoy JP, Keefe RS, Fisher EM, St Jean PL, Giegling I, Hartmann AM, Moller HJ, Ruppert A, Fraser G, Crombie C, Middleton LT, St Clair D, Roses AD, Muglia P, Francks C, Rujescu D, Meltzer HY, Goldstein DB (2009) A genome-wide investigation of SNPs and CNVs in schizophrenia. *PLoS Genet* 5(2): e1000373
8. Reddy MVPL, Wang H, Liu S, Bode B, Reed JC, Steed RD, Anderson SW, Steed L, Hopkins D, She JX (2011) Association between type 1 diabetes and GWAS SNPs in the southeast US Caucasian population. *Genes Immun* 12(3):208–212
9. Willer CJ, Bonnycastle LL, Conneely KN, Duren WL, Jackson AU, Scott LJ, Narisu N, Chines PS, Skol A, Stringham HM, Petrie J, Erdos MR, Swift AJ, Enloe ST, Sprau AG, Smith E, Tong M, Doheny KF, Pugh EW, Watanabe RM, Buchanan TA, Valle TT, Bergman RN, Tuomilehto J, Mohlke KL, Collins FS, Boehnke M (2007) Screening of 134 single nucleotide polymorphisms (SNPs) previously associated with type 2 diabetes replicates association with 12 SNPs in nine genes. *Diabetes* 56(1):256–264
10. Polderman TJC, Benyamin B, de Leeuw CA, Sullivan PF, van Bochoven A, Visscher PM, Posthuma D (2015) Meta-analysis of the heritability of human traits based on fifty years of twin studies. *Nat Genet* 47(7):702
11. Vrijheid M, Slama R, Robinson O, Chatzi L, Coen M, van den Hazel P, Thomsen C, Wright J, Athersuch TJ, Avellana N, Basagana X, Brochot C, Buccini L, Bustamante M, Carracedo A, Casas M, Estivill X, Fairley L, van Gent D, Gonzalez JR, Granum B, Grazuleviciene R, Gutzkow KB, Julvez J, Keun HC, Kogevinas M, McEathan RRC, Meltzer HM, Sabido E, Schwarze PE, Siroux V, Sunyer J, Want EJ, Zeman F, Nieuwenhuijsen MJ (2014) The human early-life exposome (HELIx): project rationale and design. *Environ Health Perspect* 122(6):535–544
12. Kawamoto T, Nitta H, Murata K, Toda E, Tsukamoto N, Hasegawa M, Yamagata Z, Kayama F, Kishi R, Ohya Y, Saito H, Sago H, Okuyama M, Ogata T, Yokoya S, Koresawa Y, Shibata Y, Nakayama S, Michikawa T, Takeuchi A, Satoh H, Ch WG (2014) Rationale and study design of the Japan environment and children's study (JECS). *BMC Public Health* 14:25
13. Vineis P, Chadeau-Hyam M, Gmuender H, Gulliver J, Herceg Z, Kleijnans J, Kogevinas M, Kyrtopoulos S, Nieuwenhuijsen M, Phillips DH, Probst-Hensch N, Scalbert A, Vermeulen R, Wild CP, Consortium E (2017) The exposome in practice: design of the EXPOsOMICS project. *Int J Hyg Environ Health* 220(2):142–151
14. Rappaport SM, Barupal DK, Wishart D, Vineis P, Scalbert A (2014) The blood exposome and its role in discovering causes of disease. *Environ Health Perspect* 122(8):769–774
15. Wild CP (2012) The exposome: from concept to utility. *Int J Epidemiol* 41(1):24–32
16. Miller GW, Jones DP (2014) The nature of nurture: refining the definition of the exposome. *Toxicol Sci* 137(1):1
17. Wild CP (2005) Complementing the genome with an “exposome”: the outstanding challenge of environmental exposure measurement in molecular epidemiology. *Cancer Epidemiol Biomark Prev* 14(8):1847–1850

18. Louis GMB, Sundaram R (2012) Exposome: time for transformative research. *Stat Med* 31(22):2569–2575
19. Rattray NJW, Deziel NC, Wallach JD, Khan SA, Vasilou V, Ioannidis JPA, Johnson CH (2018) Beyond genomics: understanding exposotypes through metabolomics. *Hum Genomics* 12(1):4
20. Steinberg CEW, Sturzenbaum SR, Menzel R (2008) Genes and environment - striking the fine balance between sophisticated biomonitoring and true functional environmental genomics. *Sci Total Environ* 400(1–3):142–161
21. Nicholson JK, Lindon JC, Holmes E (1999) Metabonomics: understanding the metabolic responses of living systems to pathophysiological stimuli via multivariate statistical analysis of biological NMR spectroscopic data. *Xenobiotica* 29(11):1181–1189
22. Patti GJ, Yanes O, Siuzdak G (2012) Metabolomics: the apogee of the omics trilogy. *Nat Rev Mol Cell Biol* 13(4):263–269
23. Ellis JK, Athersuch TJ, Thomas LDK, Teichert F, Perez-Trujillo M, Svendsen C, Spurgeon DJ, Singh R, Jarup L, Bundy JG, Keun HC (2012) Metabolic profiling detects early effects of environmental and lifestyle exposure to cadmium in a human population. *BMC Med* 10:61
24. Maitre L, Villanueva CM, Lewis MR, Ibarluzea J, Santa-Marina L, Vrijheid M, Sunyer J, Coen M, Toledano MB (2016) Maternal urinary metabolic signatures of fetal growth and associated clinical and environmental factors in the INMA study. *BMC Med* 14:177
25. Baker MG, Simpson CD, Lin YS, Shireman LM, Seixas N (2017) The use of metabolomics to identify biological signatures of manganese exposure. *Ann Work Expo Health* 61(4):406–415
26. Johnson CH, Athersuch TJ, Collman GW, Dhungana S, Grant DF, Jones DP, Patel CJ, Vasilou V (2017) Yale school of public health symposium on lifetime exposures and human health: the exposome; summary and future reflections. *Hum Genomics* 11:32
27. Ivanisevic J, Zhu ZJ, Plate L, Tautenhahn R, Chen S, O'Brien PJ, Johnson CH, Marletta MA, Patti GJ, Siuzdak G (2013) Toward Omic scale metabolite profiling: a dual separation-mass spectrometry approach for coverage of lipid and central carbon metabolism. *Anal Chem* 85(14):6876–6884
28. Buck Louis GM, Smarr MM, Patel CJ (2017) The Exposome research paradigm: an opportunity to understand the environmental basis for human health and disease. *Curr Environ Health Rep* 4(1):89–98
29. Stingone JA, Louis GMB, Nakayama SF, Vermeulen RCH, Kwok RK, Cui YX, Balshaw DM, Teitelbaum SL (2017) Toward greater implementation of the Exposome research paradigm within environmental epidemiology. *Annu Rev Public Health* 38(38):315–327
30. Robinson O, Basagana X, Agier L, de Castro M, Hernandez-Ferrer C, Gonzalez JR, Grimalt JO, Nieuwenhuijsen M, Sunyer J, Slama R, Vrijheid M (2015) The pregnancy Exposome: multiple environmental exposures in the INMA-Sabadell birth cohort. *Environ Sci Technol* 49(17):10632–10641
31. Chung MK, Kannan K, Louis GM, Patel CJ (2018) Toward capturing the Exposome: exposure biomarker variability and Coexposure patterns in the shared environment. *Environ Sci Technol* 52(15):8801–8810
32. Rappaport SM (2016) Genetic factors are not the major causes of chronic diseases. *PLoS One* 11(4):e0154387
33. Go YM, Walker DI, Liang YL, Uppal K, Soltow QA, Tran V, Strobel F, Quyyumi AA, Ziegler TR, Pennell KD, Miller GW, Jones DP (2015) Reference standardization for mass spectrometry and high-resolution metabolomics applications to Exposome research. *Toxicol Sci* 148(2):531–543
34. Dennis KK, Marder E, Balshaw DM, Cui YX, Lynes MA, Patti GJ, Rappaport SM, Shaughnessy DT, Vrijheid M, Barr DB (2017) Biomonitoring in the era of the Exposome. *Environ Health Perspect* 125(4):502–510
35. Lei ZT, Huhman DV, Sumner LW (2011) Mass spectrometry strategies in metabolomics. *J Biol Chem* 286(29):25435–25442
36. Ulrich B, Stahlmann R (2004) Developmental toxicity of polychlorinated biphenyls (PCBs): a systematic review of experimental data. *Arch Toxicol* 78(5):252–268
37. Balcio glu EB (2016) Potential effects of polycyclic aromatic hydrocarbons (PAHs) in marine foods on human health: a critical review. *Toxin Rev* 35(3–4):98–105
38. Frederiksen M, Vorkamp K, Thomsen M, Knudsen LE (2009) Human internal and external exposure to PBDEs—a review of levels and sources. *Int J Hyg Environ Health* 212(2):109–134
39. Herbstman JB, Sjodin A, Kurzon M, Lederman SA, Jones RS, Rauh V, Needham LL, Tang D, Niedzwiecki M, Wang RY, Perera F (2010) Prenatal exposure to PBDEs and neurodevelopment. *Environ Health Perspect* 118(5):712–719

40. Pleil JD, Stiegel MA, Sobus JR, Tabucchi S, Ghio AJ, Madden MC (2010) Cumulative exposure assessment for trace-level polycyclic aromatic hydrocarbons (PAHs) using human blood and plasma analysis. *J Chromatogr B Analyt Technol Biomed Life Sci* 878 (21):1753–1760
41. Marek RF, Thorne PS, Wang K, DeWall J, Hornbuckle KC (2013) PCBs and OH-PCBs in serum from children and mothers in urban and rural US communities. *Environ Sci Technol* 47:3353–3361
42. Awad AM, Martinez A, Marek RF, Hornbuckle KC (2016) Occurrence and distribution of two hydroxylated polychlorinated biphenyl congeners in Chicago air. *Environ Sci Technol Lett* 3 (2):47–51
43. Zheng XY, Dupuis KT, Aly NA, Zhou YX, Smith FB, Tang KQ, Smith RD, Baker ES (2018) Utilizing ion mobility spectrometry and mass spectrometry for the analysis of polycyclic aromatic hydrocarbons, polychlorinated biphenyls, polybrominated diphenyl ethers and their metabolites. *Anal Chim Acta* 1037:265–273
44. Marquez-Sillero I, Aguilera-Herrador E, Cardenas S, Valcarcel M (2011) Ion-mobility spectrometry for environmental analysis. *Trends Anal Chem* 30(5):677–690
45. Smith CA, Want EJ, O'Maille G, Abagyan R, Siuzdak G (2006) XCMS: processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Anal Chem* 78(3):779–787
46. Katajamaa M, Miettinen J, Oresic M (2006) MZmine: toolbox for processing and visualization of mass spectrometry based molecular profile data. *Bioinformatics* 22(5):634–636
47. Lommen A (2009) MetAlign: Interface-driven, versatile metabolomics tool for hyphenated full-scan mass spectrometry data Preprocessing. *Anal Chem* 81(8):3079–3086
48. Misra BB, Mohapatra S (2019) Tools and resources for metabolomics research community: a 2017–2018 update. *Electrophoresis* 40 (2):227–246
49. Chadeau-Hyam M, Athersuch TJ, Keun HC, De Iorio M, Ebbels TMD, Jenab M, Sacerdote C, Bruce SJ, Holmes E, Vineis P (2011) Meeting-in-the-middle using metabolic profiling - a strategy for the identification of intermediate biomarkers in cohort studies. *Biomarkers* 16(1):83–88
50. MacPherson S, Arbuckle TE, Fisher M (2018) Adjusting urinary chemical biomarkers for hydration status during pregnancy. *J Expo Sci Environ Epidemiol* 28(5):481–493
51. Dieterle F, Ross A, Schlotterbeck G, Senn H (2006) Probabilistic quotient normalization as robust method to account for dilution of complex biological mixtures. Application in H-1 NMR metabolomics. *Anal Chem* 78 (13):4281–4290
52. Gagnébin Y, Tonoli D, Lescuyer P, Ponte B, de Seigneur S, Martin PY, Schappeler J, Boccard J, Rudaz S (2017) Metabolomic analysis of urine samples by UHPLC-QTOF-MS: impact of normalization strategies. *Anal Chim Acta* 955:27–35
53. Johnson CH, Ivanisevic J, Benton HP, Siuzdak G (2015) Bioinformatics: the next frontier of metabolomics. *Anal Chem* 87(1):147–156
54. Uppal K, Walker DI, Jones DP (2017) xMSannotator: an R package for network-based annotation of high-resolution metabolomics data. *Anal Chem* 89(2):1063–1067
55. Warth B, Spangler S, Fang M, Johnson CH, Forsberg EM, Granados A, Martin RL, Domingo-Almenara X, Huan T, Rinehart D, Montenegro-Burke JR, Hilmers B, Aisporna A, Hoang LT, Uritboonthai W, Benton HP, Richardson SD, Williams AJ, Siuzdak G (2017) Exposome-scale investigations guided by global metabolomics, pathway analysis, and cognitive computing. *Anal Chem* 89 (21):11505–11513
56. Tautenhahn R, Patti GJ, Rinehart D, Siuzdak G (2012) XCMS online: a web-based platform to process untargeted Metabolomic data. *Anal Chem* 84(11):5035–5039
57. Wishart D, Arndt D, Pon A, Sajed T, Guo AC, Djoumbou Y, Knox C, Wilson M, Liang YJ, Grant J, Liu YF, Goldansaz SA, Rappaport SM (2015) T3DB: the toxic exposome database. *Nucleic Acids Res* 43(D1):D928–D934
58. Wishart DS, Feunang YD, Guo AC, Lo EJ, Marcu A, Grant JR, Sajed T, Johnson D, Li C, Sayeeda Z, Assempour N, Iynkkaran I, Liu YF, Maciejewski A, Gale N, Wilson A, Chin L, Cummings R, Le D, Pon A, Knox C, Wilson M (2018) DrugBank 5.0: a major update to the DrugBank database for 2018. *Nucleic Acids Res* 46(D1):D1074–D1082
59. Davis AP, Grondin CJ, Johnson RJ, Sciaky D, McMorran R, Wiegers J, Wiegers TC, Mattingly CJ (2019) The comparative toxicogenomics database: update 2019. *Nucleic Acids Res* 47(D1):D948–D954
60. Jordan S, Fonger G, Hazard G (2017) Hazardous substances data bank: recent features and enhancements. *Abstr Am Chem Soc* 254
61. Neveu V, Moussy A, Rouaix H, Wedekind R, Pon A, Knox C, Wishart DS, Scalbert A (2017) Exposome-explorer: a manually-curated

- database on biomarkers of exposure to dietary and environmental factors. *Nucleic Acids Res* 45(D1):D979–D984
62. Smith CA, O'Maille G, Want EJ, Qin C, Trauger SA, Brandon TR, Custodio DE, Abagyan R, Siuzdak G (2005) METLIN—a metabolite mass spectral database. *Ther Drug Monit* 27(6):747–751
 63. Wishart DS, Knox C, Guo AC, Eisner R, Young N, Gautam B, Hau DD, Psychogios N, Dong E, Bouatra S, Mandal R, Sinelnikov I, Xia JG, Jia L, Cruz JA, Lim E, Sobsey CA, Srivastava S, Huang P, Liu P, Fang L, Peng J, Fradette R, Cheng D, Tzur D, Clements M, Lewis A, De Souza A, Zuniga A, Dawe M, Xiong YP, Clive D, Greiner R, Nazirova A, Shaykhutdinov R, Li L, Vogel HJ, Forsythe I (2009) HMDB: a knowledge-base for the human metabolome. *Nucleic Acids Res* 37:D603–D610
 64. Horai H, Arita M, Kanaya S, Nihei Y, Ikeda T, Suwa K, Ojima Y, Tanaka K, Tanaka S, Aoshima K, Oda Y, Kakazu Y, Kusano M, Tohge T, Matsuda F, Sawada Y, Hirai MY, Nakanishi H, Ikeda K, Akimoto N, Maoka T, Takahashi H, Ara T, Sakurai N, Suzuki H, Shibata D, Neumann S, Iida T, Tanaka K, Funatsu K, Matsuura F, Soga T, Taguchi R, Saito K, Nishioka T (2010) MassBank: a public repository for sharing mass spectral data for life sciences. *J Mass Spectrom* 45(7):703–714
 65. Fahy E, Sud M, Cotter D, Subramaniam S (2007) LIPID MAPS online tools for lipid research. *Nucleic Acids Res* 35:W606–W612
 66. Huan T, Tang CQ, Li RH, Shi Y, Lin GH, Li L (2015) MyCompoundID MS/MS search: metabolite identification using a library of predicted fragment-ion-spectra of 383,830 possible human metabolites. *Anal Chem* 87(20):10619–10626
 67. Wang MX, Carver JJ, Phelan VV, Sanchez LM, Garg N, Peng Y, Nguyen DD, Watrous J, Kapono CA, Luzzatto-Knaan T, Porto C, Bouslimani A, Melnik AV, Meehan MJ, Liu WT, Criisemann M, Boudreau PD, Esquenazi E, Sandoval-Calderon M, Kersten RD, Pace LA, Quinn RA, Duncan KR, Hsu CC, Floros DJ, Gavilan RG, Kleigrewe K, Northen T, Dutton RJ, Parrot D, Carlson EE, Aigle B, Michelsen CF, Jelsbak L, Sohlenkamp C, Pevzner P, Edlund A, McLean J, Piel J, Murphy BT, Gerwick L, Liaw CC, Yang YL, Humpf HU, Maansson M, Keyzers RA, Sims AC, Johnson AR, Sidebottom AM, Sedio BE, Klitgaard A, Larson CB, Boya CA, Torres-Mendoza D, Gonzalez DJ, Silva DB, Marques LM, Demarque DP, Pociute E, O'Neill EC, Briand E, Helfrich EJN, Granatosky EA, Glukhov E, Ryffel F, Houson H, Mohimani H, Kharbush JJ, Zeng Y, Vorholt JA, Kurita KL, Charusanti P, McPhail KL, Nielsen KF, Vuong L, Elfeki M, Traxler MF, Engene N, Koyama N, Vining OB, Baric R, Silva RR, Mascuch SJ, Tomasi S, Jenkins S, Macherla V, Hoffman T, Agarwal V, Williams PG, Dai JQ, Neupane R, Gurr J, Rodriguez AMC, Lamsa A, Zhang C, Dorrestein K, Duggan BM, Almaliti J, Allard PM, Phapale P, Nothias LF, Alexandrov T, Litaudon M, Wolfender JL, Kyle JE, Metz TO, Peryea T, Nguyen DT, VanLeer D, Shinn P, Jadhav A, Muller R, Waters KM, Shi WY, Liu XT, Zhang LX, Knight R, Jensen PR, Palsson BO, Poglino K, Linington RG, Gutierrez M, Lopes NP, Gerwick WH, Moore BS, Dorrestein PC, Bandeira N (2016) Sharing and community curation of mass spectrometry data with global natural products social molecular networking. *Nat Biotechnol* 34(8):828–837
 68. Li SZ, Park Y, Duraisingham S, Strobel FH, Khan N, Soltow QA, Jones DP, Pulendran B (2013) Predicting network activity from high throughput metabolomics. *PLoS Comput Biol* 9(7):e1003123
 69. Pirhaji L, Milani P, Leidl M, Curran T, Avila-Pacheco J, Clish CB, White FM, Saghatelian A, Fraenkel E (2016) Revealing disease-associated pathways by network integration of untargeted metabolomics. *Nat Methods* 13(9):770–776
 70. Langston MA, Levine RS, Kilbourne BJ, Rogers GL, Kershenbaum AD, Baktash SH, Coughlin SS, Saxton AM, Agboto VK, Hood DB, Litchveld MY, Oyana TJ, Matthews-Juarez P, Juarez PD (2014) Scalable combinatorial tools for health disparities research. *Int J Environ Res Public Health* 11(10):10419–10443
 71. Vacquier B, Rage E, Leuraud K, Caer-Lorho S, Houot J, Acker A, Laurier D (2011) The influence of multiple types of occupational exposure to radon, gamma rays and long-lived radionuclides on mortality risk in the French “post-55” sub-cohort of uranium miners: 1956–1999. *Radiat Res* 176(6):796–806
 72. Patel CJ, Ioannidis JP (2014) Studying the elusive environment in large scale. *JAMA* 311(21):2173–2174
 73. Patel CJ, Bhattacharya J, Butte AJ (2010) An environment-wide association study (EWAS) on type 2 diabetes mellitus. *PLoS One* 5(5):e10746
 74. Patel CJ, Cullen MR, Ioannidis JP, Butte AJ (2012) Systematic evaluation of environmental

- factors: persistent pollutants and nutrients correlated with serum lipid levels. *Int J Epidemiol* 41(3):828–843
75. Manrai AK, Cui YX, Bushel PR, Hall M, Karakitsios S, Mattingly CJ, Ritchie M, Schmitt C, Sarigiannis DA, Thomas DC, Wishart D, Balsaw DM, Patel CJ (2017) Informatics and data analytics to support exposome-based discovery for public health. *Annu Rev Public Health* 38(38):279–294
76. Sun ZC, Tao YB, Li S, Ferguson KK, Meeker JD, Park SK, Batterman SA, Mukherjee B (2013) Statistical strategies for constructing health risk models with multiple pollutants and their interactions: possible choices and comparisons. *Environ Health* 12:85
77. Tibshirani R (1996) Regression shrinkage and selection via the Lasso. *J R Stat Soc Series B Stat Methodol* 58(1):267–288
78. Zou H, Hastie T (2005) Regularization and variable selection via the elastic net (vol B 67, pg 301, 2005). *J R Stat Soc Series B Stat Methodol* 67:768–768
79. Agier L, Portengen L, Chadeau-Hyam M, Basagana X, Giorgis-Allemand L, Siroux V, Robinson O, Vlaanderen J, Gonzalez JR, Nieuwenhuijsen MJ, Vineis P, Vrijheid M, Slama R, Vermeulen R (2016) A systematic comparison of linear regression-based statistical methods to assess Exposome-health associations. *Environ Health Perspect* 124(12):1848–1856
80. Bottolo L, Richardson S (2010) Evolutionary stochastic search for Bayesian model exploration. *Bayesian Anal* 5(3):583–618
81. Liquet B, Bottolo L, Campanella G, Richardson S, Chadeau-Hyam M (2016) R2GUESS: a graphics processing unit-based R package for Bayesian variable selection regression of multivariate responses. *J Stat Softw* 69(2):1–32
82. Jiang C, Wang X, Li XY, Inlora J, Wang T, Liu Q, Snyder M (2018) Dynamic human environmental Exposome revealed by longitudinal personal monitoring. *Cell* 175(1):277
83. Wang XH, Eijkemans MJC, Wallinga J, Biesbroek G, Trzcinski K, Sanders EAM, Bogaert D (2012) Multivariate approach for studying interactions between environmental variables and microbial communities. *PLoS One* 7(11):e50267
84. Jain P, Vineis P, Liquet B, Vlaanderen J, Bodinier B, van Veldhoven K, Kogevinas M, Athersuch TJ, Font-Ribera L, Villanueva CM, Vermeulen R, Chadeau-Hyam M (2018) A multivariate approach to investigate the combined biological effects of multiple exposures. *J Epidemiol Community Health* 72(7):564–571
85. Roede JR, Uppal K, Park Y, Tran V, Jones DP (2014) Transcriptome-metabolome wide association study (TMWAS) of maneb and paraquat neurotoxicity reveals network level interactions in toxicologic mechanism. *Toxicol Rep* 1:435–444
86. Guida F, Sandanger TM, Castagne R, Campanella G, Polidoro S, Palli D, Krogh V, Tumino R, Sacerdote C, Panico S, Severi G, Kyrtopoulos SA, Georgiadis P, Vermeulen RCH, Lund E, Vineis P, Chadeau-Hyam M (2015) Dynamics of smoking-induced genome-wide methylation changes with time since smoking cessation. *Hum Mol Genet* 24(8):2349–2359
87. Mahieu NG, Patti GJ (2017) Systems-level annotation of a 25 000 features to fewer than G metabolomics data set reduces 1000 unique metabolites. *Anal Chem* 89(19):10397–10406
88. Geng DW, Jorgsten IE, Dunstan J, Hagberg J, Wang T, Ruzzin J, Rabasa-Lhoret R, van Bavel B (2016) Gas chromatography/atmospheric pressure chemical ionization/mass spectrometry for the analysis of organochlorine pesticides and polychlorinated biphenyls in human serum. *J Chromatogr A* 1453:88–98
89. Zhao S, Luo X, Li L (2016) Chemical isotope Labeling LC-MS for high coverage and quantitative profiling of the hydroxyl submetabolome in metabolomics. *Anal Chem* 88(21):10617–10623
90. Treutler H, Tsugawa H, Porzel A, Gorzolka K, Tissier A, Neumann S, Balcke GU (2016) Discovering regulated metabolite families in untargeted metabolomics studies. *Anal Chem* 88(16):8082–8090
91. Depke T, Franke R, Bronstrup M (2017) Clustering of MS2 spectra using unsupervised methods to aid the identification of secondary metabolites from *Pseudomonas aeruginosa*. *J Chromatogr B: Anal Technol Biomed Life Sci* 1071:19–28
92. van der Hooft JJ, Wandy J, Barrett MP, Burgess KEV, Rogers S (2016) Topic modeling for untargeted substructure exploration in metabolomics. *Proc Natl Acad Sci U S A* 113(48):13738–13743
93. van der Hooft JJ, Wandy J, Young F, Padmanabhan S, Gerasimidis K, Burgess KEV, Barrett MP, Rogers S (2017) Unsupervised discovery and comparison of structural families across multiple samples in untargeted metabolomics. *Anal Chem* 89(14):7569–7577

94. Lu YF, Goldstein DB, Angrist M, Cavalleri G (2014) Personalized medicine and human genetic diversity. *Cold Spring Harb Perspect Med* 4(9):a008581
95. Juarez PD, Matthews-Juarez P, Hood DB, Im W, Levine RS, Kilbourne BJ, Langston MA, Al-Hamdan MZ, Crosson WL, Estes MG, Estes SM, Agboto VK, Robinson P, Wilson S, Lichtveld MY (2014) The public health exposome: a population-based, exposure science approach to health disparities research. *Int J Environ Res Public Health* 11(12):12866–12895



Chapter 23

Network-Based Approaches for Multi-omics Integration

Guangyan Zhou, Shuzhao Li, and Jianguo Xia

Abstract

Network-based approach is rapidly emerging as a promising strategy to integrate and interpret different -omics datasets, including metabolomics. The first section of this chapter introduces the current progresses and main concepts in multi-omics integration. The second section provides an overview of the public resources available for creation of biological networks. The third section describes three common application scenarios including subnetwork identification, network-based enrichment analysis, and systems metabolomics. The section four introduces the concept of hierarchical community network analysis. The section five discusses different tools for network visualization. The chapter ends with a future perspective on multi-omics integration.

Key words Systems biology, Multi-omics integration, Data integration, Network enrichment analysis, Hierarchical community networks, Network visualization

1 Introduction

The widespread applications of various omics technologies across life sciences have shifted the main bottleneck from data generation to data analysis and interpretation. Researchers now can easily obtain comprehensive landscapes of DNA/RNA molecules, proteins or metabolites using genomics, transcriptomics, proteomics, or metabolomics technologies. Fueled by the decreasing costs of omics technologies, the past few years have witnessed a growing number of studies employing more than one type of omics technologies (i.e., multi-omics) to investigate complex diseases and biological processes [1–4]. Despite being a relatively new member of the omics family, metabolomics has gradually become a key component in recent multi-omics studies. Small compounds are not only the downstream products of the complex interactions between host genetics, life styles and environmental exposures, they also play active roles in physiology and disease through modulation of other “omics” levels [5]. How to extract biologically meaningful information from these multi-omics data have become

a key area of research in current bioinformatics and systems biology [6–9].

Different methods have been explored to help integrate and interpret multi-omics data in order to gain systems insights. Two general categories have emerged—the statistical integration and the network-based integration. Statistical integration aims to identify overall patterns or shared signatures across multiple omics datasets by drawing inference from the data themselves, without necessarily using prior knowledge. Some of the commonly used integration methods include univariate correlation, matrix-based correlation and matrix factorization [10–12]. Univariate correlation methods attempt to identify relationships between individual molecules from one omics layer to those of another omics layer. Matrix-based correlation methods seek to identify linear relationships that best explain correlation and variability both within and across omics layers. Lastly, matrix factorization methods focus on decomposing the datasets and projecting variations onto dimensionally reduced space. Although very powerful, they tend to produce very complex results which are very challenging to explain the underlying biological processes. This review will not go into detail about these statistical approaches. Please refer to the several excellent reviews for more details [12–15].

The network-based integration views the biological system as interconnected networks of molecular entities, and is primarily concerned with creating, analyzing, and visually exploring such networks [6]. To perform network-based integration, researchers need to first obtain a network relevant to the biological systems under investigation. A network can be constructed based on our current knowledge (i.e., genetic, physical, or biochemical interactions obtained from databases or previous studies). The most commonly used biological networks are protein–protein interaction (PPI) networks and metabolic networks [16]. These networks have proven to be invaluable frameworks to aid in global understanding of complex interactions among different molecules of interest [17]. Researchers first project the molecules of interest to these networks, and then try to identify important connections and modules to gain insights into the underlying system. Many algorithms have been developed to identify subnetworks enriched for user input data, which correspond to context-specific functional modules characteristic of the biological systems under study [18].

This chapter will introduce the three key components in network-based multi-omics integration—the prior knowledgebase to create the networks; the algorithms to mine the networks; and the visualization strategies to explore the networks. Researchers need to develop a good understanding of the three components in order to efficiently apply the network-based approach to obtain systems insights and to develop hypotheses. A further conceptual development of hierarchical community networks will also be introduced and illustrated.

2 Knowledge Bases and Multi-omics Data Resources

The general idea of network-based integration is to map the multi-omics experimental data into the context of our knowledge framework in the forms of different networks. For instance, proteins can bind to DNA or RNA to regulate the expression of genes, and interactions between proteins, metabolites, and lipids can control the function of molecular apparatus and signal cascades. The ensemble of these different interactions provides the knowledge framework for the use of network-based approaches to multi-omics integration.

Over the last decade, significant efforts have been invested into creating comprehensive databases on annotations of molecules, biological pathways and molecular interaction networks. Currently, curated metabolic pathways can be found in several comprehensive databases including KEGG [19], Reactome [20], MetaCyc [21], BiGG [22], Recon3D [23], and WikiPathways [24]. KEGG pathway database covers ~8500 reactions and ~16,500 compounds combining different species. MetaCyc offers a comparable number of pathways to KEGG, however, it provides a greater set of pathway attributes and larger coverage in plants, fungi, metazoans, and actinobacteria [25]. BiGG specializes in collecting genome-scale metabolic models for a given organism. Recon3D is specialized in the characterization of human metabolic model. WikiPathways is unique with its collaborative pathway editing feature which allows research communities to contribute to pathway annotation efforts. Pathways are functionally coherent and are easy to use and to understand. However, they may miss important reactions that cross the boundaries of multiple pathways. In addition, many metabolites detected in metabolomics are not covered in any metabolic pathways. To address these limitations, researchers have combined manual curation and text-mining to create more confederated biochemical reaction networks or protein–chemical interactions databases such as RHEA [26], CTD [27], and STITCH [28].

Besides metabolic pathways and networks, PPI is widely used to create molecular interaction networks. There are several comprehensive PPI databases such as IntAct [29], BioGRID [30], MINT [31], and InnateDB [32]. To control the high false positives in PPI, these databases focus on experimentally validated interactions, which only cover a subset of the proteins in several model organisms. To address these limitations, STRING database collects and integrates experimentally validated, literature-curated, and computationally predicted PPI data covering more than 2000 different organisms [33].

In addition to metabolic and PPI networks, gene regulatory networks (GRN) such as transcription factor (TF)–gene and

miRNA–gene interactions offer critical mechanistic insights into gene regulations and downstream effects. These interactions used to be based primarily on computational predictions which tend to contain very high false positives [34, 35]. To improve the situation, high-throughput experimental technologies have been developed in recent years that enabled the development of several high-quality databases on TF–gene and miRNA–gene interactions [36–38].

Multi-omics datasets have become increasingly available to the public, providing exciting opportunities for developing and benchmarking computational methods for data integration. There have been concerted efforts by several large consortia to generate multi-omics datasets to study biological responses and complex diseases (e.g., the NCI60 cell line datasets [39], The Cancer Genome Atlas (TCGA) [40], the integrative Human Microbiome Project (iHMP) [41], Metabolic Syndrome in Men (METSIM) Study [42]). Other projects such as Japanese Multi Omics Reference Panel (jMorp) [43] and MuTHER [44] aim at characterizing the molecular phenotypes of healthy individuals and twins. Finally, the Omics Discovery Index (OmicsDI) platform aims to facilitate the discovery and coordination of different omics dataset [45]. Table 1 provides a list of these examples.

3 Network-Based Integration

Biological systems can be abstracted as a series of networks comprised of interconnected molecular entities. When the networks are relatively small, researchers can directly visualize the relationships to gain global insights. For large networks, it is often necessary to first apply some graph theory to extract the most relevant subnetworks before they can be visualized. Figure 1 illustrates three different purposes using network-based approaches. In the following section, we introduce these three types of network analysis: subnetwork analysis, network enrichment analysis, and systems metabolomics.

3.1 Subnetwork Identification and Analysis

A common network-based approach is to build context-specific multi-omics subnetwork from experimental data [46–48]. This is usually achieved by connecting significant molecules identified in individual omics layers through known molecular interactions. For instance, to integrate transcriptomics and metabolomics, the list of differentially expressed genes can be mapped into PPI databases through their corresponding proteins to form PPI subnetworks while the list of differentially abundant metabolites can connect to the PPI subnetwork through known biochemical reactions. The resulting subnetworks contain both input genes/proteins and metabolites as well as a subset of their direct interacting partners. They reveal not only the relationships among those important

Table 1
Example large-scale multi-omics projects

Project	Description	Links
NCI60	Small compound screening on 60 human tumor cell lines	https://dtp.cancer.gov/discovery-development/nci-60/
TCGA	Cancer genomics project: ~20,000 samples from 33 cancer types and matched normal samples	https://www.cancer.gov/about-nci/organization/ccg/research/structural-genomics/tcga
TopMed	Large project to study heart, lung, blood, and sleep disorders	https://www.nhlbi.nih.gov/science/trans-omics-precision-medicine-topmed-program
jMorp	Large multi-omics database generated using samples from 5000 healthy Japanese volunteers	https://jmorp.megabank.tohoku.ac.jp
iHMP	Multi-omics data characterizing microbiome-host profiles in health and disease (pregnancy and preterm birth, inflammatory bowel disease, type II diabetes)	https://www.hmpdacc.org/ihmp/
MuTHER	Multi-omics resources from a range of tissues collected from a set of ~850 UK twins	http://www.muther.ac.uk/
METSIM	Population-based study to investigate nongenetic and genetic factors associated with the risk of T2D and CVD	http://www.nationalbiobanks.fi/index.php/studies2/10-metsim
OmicsDI	A meta-database for multi-omics data discovery and indexing	https://www.omicsdi.org/

molecules, but also additional molecules (i.e., interaction partners) which may be more suitable as potential biomarkers or therapeutic targets.

Different algorithms can be used to build the context-specific subnetworks. The most basic approach is to build subnetworks by identifying known interactions that exist between molecules that are prioritized from multi-omics experiments. For instance, for proteins and metabolites, we can query the metabolic interaction database to identify known relationships between enzymes and metabolites from our input (also known as zero-order interaction network). The main limitation with this approach is the potential lack of direct interactions between molecules across omics layers, resulting in very sparse subnetworks connecting a very small number of input elements. To address this issue, researchers often create subnetworks containing the input molecules, the relationships among them, and their direct interacting partners (also known as first-order interaction network). However, such networks can sometimes become too big to be visualized for meaningful interpretation. Graph mining algorithms, such as Prize-Collecting

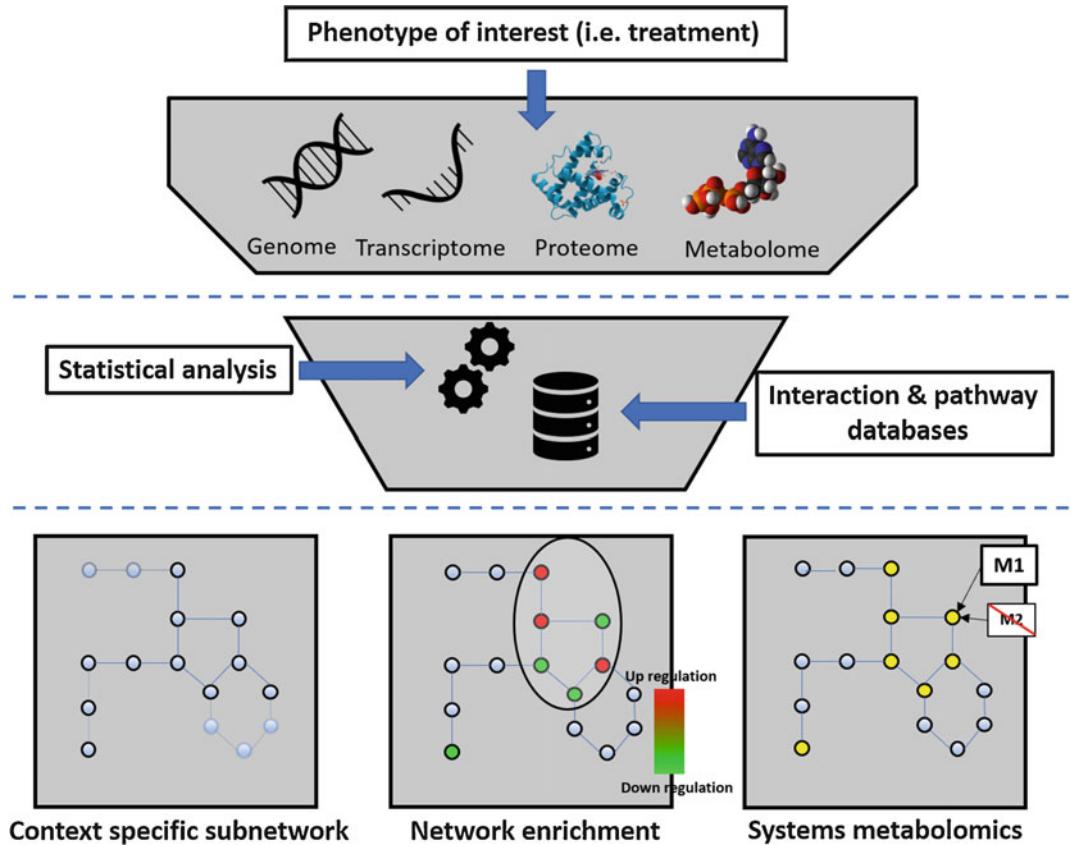


Fig. 1 The overall workflow of knowledge-driven network-based approach. Experimental data obtained from single-omics analysis are mapped onto existing molecular interaction or pathway databases to build a multi-omics network. The resulting network can be used to identify context-specific subnetworks, perform network enrichment analysis and for compound identification

Steiner Forest (PCSF) [49, 50] or jActiveModule [51], are often applied to further identify and extract the relevant context specific modules while minimizing the number of irrelevant nodes and connections.

PCSF is a well-known graph theory problem [49, 50]. Given an undirected global network and a list of input nodes with prizes, the objective is to identify one or multiple subnetworks while maximizing the prizes (input nodes), minimizing the costs (edges) and the number of subnetworks. For instance, the node prizes can correspond to p -values obtained from differential expression analysis and edge costs correspond to the inverse of interaction confidence scores. As a general method, it can be used to interpret multi-omics dataset. This algorithm has been used to identify novel Parkinson-related druggable targets [52] and enriched pathways that are affected by Kaposi's Sarcoma associated Herpesvirus infection [53].

In contrast to PCSF, jActiveModule scans the molecular interaction network to identify “network hotspots” (i.e., subnetwork with significant differential expression changes), without having a list of predefined nodes to be included in the resulting subnetworks. Specifically, it computes *p*-value at node level from expression data, which is then converted to *z*-values and aggregated at subnetwork level. To identify highest scoring subnetworks, it uses an algorithm based on simulated annealing, a method that models the physical process of heating a material and then slowly lowering the temperature. For each iteration of the algorithm, it randomly selects a node and keeps it if it improves the solution (subnetwork score). Multiple methods based on the similar underlying concept have been developed subsequently to identify active modules or network hotspots [54, 55]. Researchers have used this type of approach to identify novel biomarkers in gastric cancer [56] and novel metabolic modules that regulate macrophage polarization [57].

Random walk with restart (RWR) is another promising graph mining algorithm for the detection of novel candidates relevant to condition of interest. It aims at identifying nodes with similar functions to seed nodes, with the assumption that nodes with similar functions tend to group more closely in a network. The algorithm estimates affinity score of nodes in relation to seed nodes. The basis of the algorithm is a walker that either randomly moves to an immediate neighboring node or goes back to the starting node. After some iterations, it reaches a stable state which outputs a steady probability vector containing the affinity scores associated with nodes. The RWR algorithm has been used to predict drug–target interactions [58], lncRNA–disease associations [59], and miRNA–disease associations [60].

3.2 Network-Based Enrichment Analysis

Enrichment analysis aims to identify functions that are significantly changed by comparing experimental data with known pathways or other functionally related gene sets. A wide range of methods have been developed. Among them, overrepresentation analysis (ORA) and gene set enrichment analysis (GSEA) are two widely methods for this task [61]. Although very powerful approaches, these approaches ignore topological properties of these molecules within pathways or networks, which are essential in order to understand systems behaviour. In addition, they rely on direct overlap between input molecules and functions, and molecules with missing pathway or gene set annotation are not considered.

The network-based enrichment methods have been developed to overcome some of these issues. These approaches take advantage of graph-based statistics to exploit the connectivity information in biological pathways or molecular interaction networks. A typical approach consists of two main steps: (1) mapping the experimental data onto the pathway or network; and (2) use “topology-aware”

statistics that integrates structural information to calculate enrichment scores. Proximity measure is often used to relate user input to known biological pathways, as a fundamental principle is “guilt-by-association” which assumes that nodes with similar functions tend to be closer to each other in the networks. This approach allows the detection of enriched pathways/gene sets whose associated members do not directly overlap with those in user input. The connectivity information can also complement classical enrichment methods as an extra weighing factor, to increase discriminative power and interpretability within the network context. For instance, topological properties of a node can be used to evaluate its significance in a given pathway. Below, we will briefly introduce three methods for network-based enrichment analysis—the signaling pathway impact analysis (SPIA), the network enrichment analysis (NEA), and EnrichNet.

Signaling pathway impact analysis (SPIA) [62] is one of the earliest “topology-aware” enrichment methods. SPIA combines scores from classical enrichment analysis with network-based scores. Firstly, the algorithm matches elements from experimental data to curated biological pathways and calculate a score using ORA or GSEA. Secondly, using seed nodes as the origin, it propagates the measured expression/abundance changes to the rest of nodes in the pathways, generating a gene-level perturbation score for each node. This perturbation analysis will assign different significance to genes according to its position in the pathway (upstream or downstream) and other topological properties (i.e., degrees and betweenness). Finally, the two results are combined to generate a final score for a given pathway. By incorporating pathway structure information, SPIA shows better specificity and more sensitivity than those based on expression changes alone [62].

The network enrichment analysis (NEA) method [63] aims at integrating functional information and network connectivity. It quantifies functional set enrichment by assessing the interactions between reference gene sets and input genes. Specifically, it compares the observed number of links between them and with the expected number for the given size of input genes under null models via a network randomization algorithm to rewire the existing edges while preserving the degree distribution.

EnrichNet is another network-based enrichment analysis method, in which the enrichment score is computed by measuring proximity between input genes and reference gene sets, and does not require the nodes to be direct interacting partner of each other [64]. Firstly, RWR algorithm is used to score its distance to all reference gene sets for each seed node. Secondly, node-level distance scores are converted into distance score vectors for reference gene set. The individual distance vectors are then aggregated to form a distribution corresponding to the background model. Finally, the enrichment score of a specific reference gene set is

calculated by measuring the deviation of its distance vector to the average distribution of the background model.

3.3 Systems Metabolomics

Unlike transcriptomics or proteomics technologies, current metabolomics platforms can only cover a small portion of the metabolome. Given the chemical diversities of metabolites, no single analysis method offers a “one-size-fits-all” solution. Nonetheless, global metabolomics aims to provide an unbiased global measurement of the metabolome. Modern mass spectrometers provide ultrahigh resolution in characterizing molecular mass, and LC-MS (liquid chromatography coupled mass spectrometry) has become the dominant methods for global metabolomics in recent years [65]. These LC-MS systems routinely detect tens of thousands of metabolite “features,” defined mainly by mass-to-charge ratio (m/z) and retention time. However, metabolite identification remains a bottleneck of the field, as ambiguity is common in compound matching and robust fragmentation data are difficult to acquire for many metabolites. Fast and accurate compound annotation remains a key obstacle in interpretation of global metabolomic data and its effective integration with other omics data for systems insights [66]. Systems metabolomics has emerged in recent years to help address these issues. Here we define systems metabolomics as the approach of inferring functional insights from global metabolite profiles by leveraging prior knowledge and network structures.

One solution is shift the focal plane of analysis from individual metabolites to pathways or networks (*see Chapter 19* by Karnovsky and Li), which are known to be more tolerant against random errors [67]. Mummichog is among one of the earliest bioinformatics tools using this concept [68]. It takes as input two lists of spectral features—a significant list and a background reference peak list (i.e., all features detected in the experiment). Computationally predicted metabolites from the significant peaks are then mapped into known metabolic pathways or networks from existing databases. Enrichment analysis is performed on these putative metabolites and then compared with the results based on peaks drawn randomly from the reference peak list. Thus, mummichog borrows knowledge from prior metabolic reactions to prioritize the prediction of metabolites and to perform pathway/network analysis in one step. This method has since been successful applied to many multi-omics studies [69–71] and has been implemented in popular metabolomics tools such as XCMSonline [72] and MetaboAnalyst [73]. Figure 2 shows an example mummichog output as implemented in MetaboAnalyst 4.0.

PIUMet employs a similar concept [74] but extends the search space from metabolic pathways to an integrated weighted network composed of protein–protein interaction combined with metabolic interactions. Additionally, it allows the integrative analysis of spectral data with other omics data such as transcriptomics or

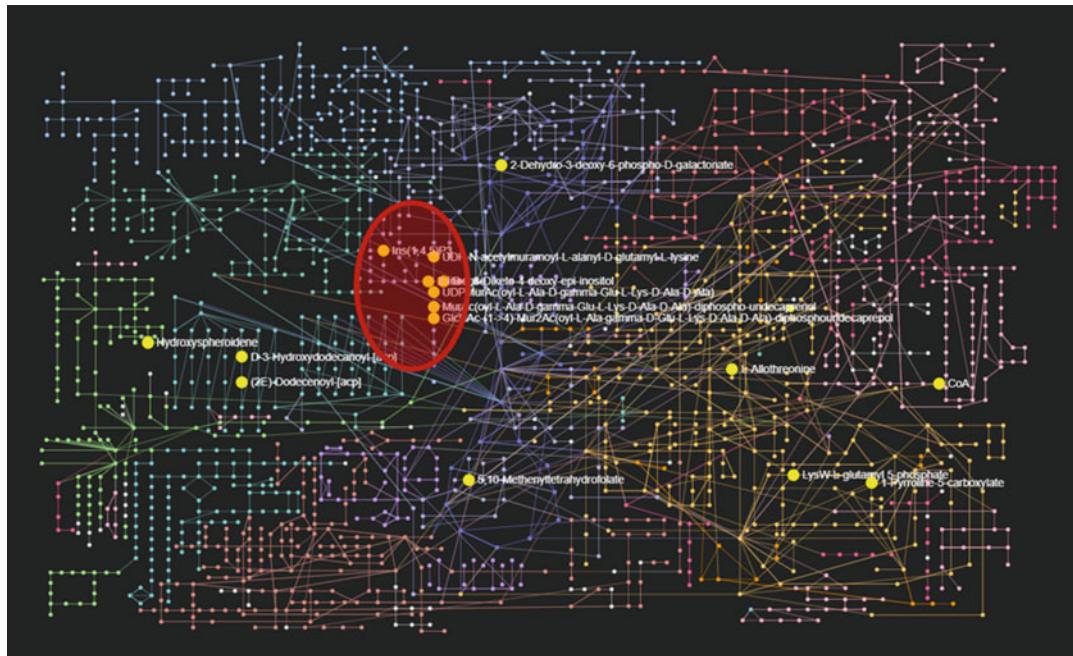


Fig. 2 An example mummichog result implemented in MetaboAnalyst. Mummichog algorithm directly maps peaks as putative metabolites onto metabolic pathways and performs enrichment analysis to identify perturbed pathways (shown in red circle)

proteomics. The first step consists of mapping the significant peak features into the global network as sets of putative metabolites. To help elucidate putative identities of altered metabolite feature, it uses PCSF algorithm to identify minimal subnetworks containing metabolite matches, assuming that metabolic reactions tend to be parsimonious.

4 Data Driven Hierarchical Community Networks

The molecular networks discussed above are the workhorses of systems biology. It is also important to consider the spatial and temporal parameters in real biological systems. If a metabolic reaction is limited in the mitochondria, a large generic PPI network may contain a lot of irrelevant information and mislead investigators to many false positives. Given that different data types are often from different compartments in human studies (e.g., gut microbiome from digestive tract, metabolomics from serum, and transcriptomics from blood cells), direct coupling based on gene activities can be very problematic. Kinetic rates may not be captured in experiments, but they are critical to data interpretation. For example, an intracellular metabolic reaction can occur in minutes, but the activation and differentiation of T cells in an immune

response can take weeks. Therefore, computational models should match the sampling and resolution of the experimental data. One of the motivations of data driven approaches is to be close to the biological truth.

Since different technologies often produce different variance structure, PLS (partial least squares, or projection to latent structure) based methods are popular for integrating multi-omics data [75]. After identifying the statistical relationships between features in different -omics spaces, the interpretation is still challenging, because the result can be still very redundant and is not necessarily fit into existing knowledge structure (e.g., a known pathway). The problem is well exemplified in transcriptomics from whole blood or white blood cells: a large portion of the transcriptomic change can be caused by the change of frequency of cell populations. A univariate parameter at the cellular scale corresponds to hundreds of features at the gene transcript level. To recognize as well as to model these hierarchical organizations in data, a useful conceptual development is hierarchical community networks (HiCoNet, <https://github.com/shuzhao-li/hiconet>). The prototype of HiCoNet was published as a multiscale, multifactorial response network to herpes zoster vaccine [76]. The authors integrated plasma metabolomics, PBMC (peripheral blood mononuclear cell) transcriptomics, plasma cytokines, and blood cell populations in the network model and used it to interpret adaptive immune responses to the vaccination. Within each data type, local communities were first detected. The association between these communities across data types was then assessed by PLS regression and permutation. Their results established significant associations between blood metabolites and intracellular transcriptome, and the importance of metabolic phenotypes during human immune responses [76]. The intermediate layer of communities provides a powerful means to incorporate knowledge mapping. For untargeted metabolomics, the community structure is likely to combine signals from isotopes, adducts and metabolites of the same classes, therefore achieving meaningful dimension reduction and cleaner result, as illustrated in Fig. 3.

This HiCoNet approach has already been applied to infectious disease [77] and exposome research (Li et al., Reproductive Toxicology, in press). The concept of leveraging community structure in each data type is compatible and operational with real biological problems. The microbiome data are easily organized into communities based on transcription profiles and taxonomy. The emerging single cell sequencing data are naturally organized by communities of subpopulations. For exposome research, because humans are usually exposed to a mixture of many environmental factors including chemicals, exposure communities are a useful means to investigate mixed exposures. The community detection algorithms are a well-studied and mature area, easily adapted to HiCoNet. The

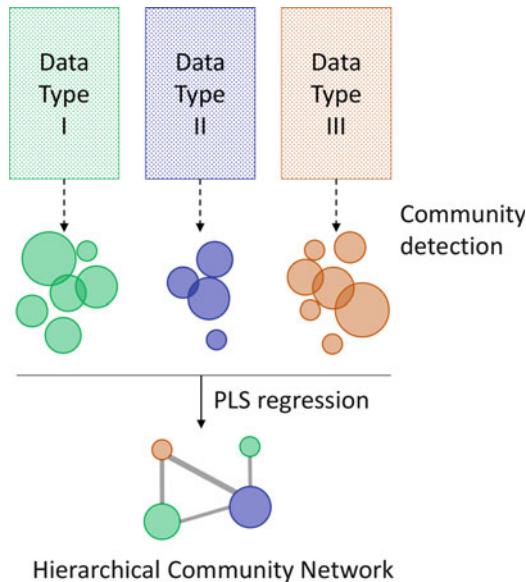


Fig. 3 A schematic illustration of hierarchical community network. The communities are detected within each data type. The association between communities of different data types is tested via PLS regression, and the significant associations form the hierarchical community network

continued computational development will aid the visualization and exploration of the resulting networks.

5 Network Visualization

Statistics alone is often insufficient to communicate the complex information within multi-omics data to researchers. A key component of network-based integration is visualization. Human eyes are very good at pattern detection and to perform visual data mining to gain deep insight or for hypothesis generation [78]. When presented properly, a network is very intuitive to navigate and to interpret. In multi-omics networks, nodes represent a biological entity such as metabolite, gene, protein or miRNA, and edges represent known interactions shared between pairs of nodes. To allow researchers to visually process large quantity of information, different visual encoding can be applied to facilitate pattern detection. For instance, node size can vary according to topological degree, node color based on expression or abundance value, and node shape based on molecule types.

Several applications are available for interactive network visualization of multi-omics data. Cytoscape is arguably the most widely used tool for analysis and visualization of biological networks [79]. Aside of the powerful features in terms of interactive

exploration and manipulation of networks, it has a plug-in system with well-defined application programming interface (API) to allow the community to contribute a wide range of analytics features. KeyPathwayMiner is an example of Cytoscape plug-in developed for integrated analysis of multi-omics data [80]. Given a global network, it attempts to identify perturbed subnetworks using expression/abundance data of omics elements. Another example is PathVisio, a popular Cytoscape plug-in developed for the visualization and analysis of biological pathways [81]. It supports multi-omics visualization of biological pathways by mapping expression values of transcriptomics and metabolomics datasets, allowing users to visually assess the general patterns. There are also several web-based applications such as Pathview and PaintOmics that support the similar types of pathway analysis [82, 83].

The past few years have witnessed a trend toward web-based network visualization. For instance, MetaboAnalyst 4.0 contains two modules—Joint Pathway Analysis and Network Explorer to support the integrative analysis and visualization of data from metabolomics, transcriptomics, and metagenomics [73]. The 3Omics is a web application developed for combined analysis of human transcriptomics, metabolomics, and proteomics data [84]. Another web-based application, OmicsNet, was developed for the creation and visualization of multi-omics biological networks in 3D space [47]. Users can map different molecules (genes/proteins, transcription factors (TF), miRNAs, and/or metabolites) of interest into the integrated knowledgebase to construct context-specific subnetworks. The interactive 3D network viewer can arrange the molecules from different omics into different “layers” to reduce the visual cluttering and to facilitate the pattern detection. An example network visualization generated using OmicsNet is given in Fig. 4, showcasing a multi-omics network composed of TF-gene, protein-protein and metabolite-enzyme interactions. Users can perform a variety of network analysis such as enrichment analysis, module detection, topology analysis, and shortest path computing. The visualization system was development using the cutting-edge WebGL technology. Given the fast development of virtual reality (VR) technology, it is envisioned that VR will play a significant role in visual analytics for multi-omics integration and systems biology [85].

6 Future Prospects

This chapter introduces several key concepts and current status in network-based multi-omics integration. Network-based approaches rely heavily on the quality and details of the underlying molecular interactions or pathways. Therefore, this approach is limited by our current knowledge and understanding of biological

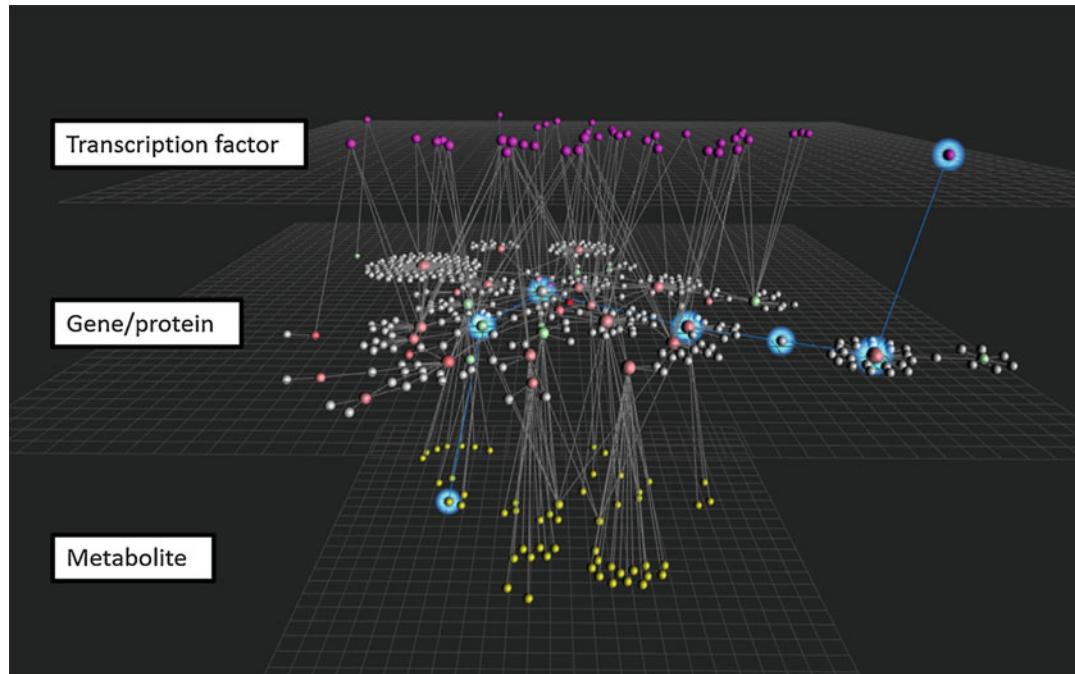


Fig. 4 An example multi-omics network generated using OmicsNet. Transcription factors are colored in purple. Genes/proteins are colored by their expression value using a green-red gradient. Grey nodes refers to predicted interacting partner. Metabolites are colored in yellow. A shortest path between a transcription factor and metabolite is highlighted in blue

processes and is primarily used to refine hypothesis or to clarify the mechanistic links that explain molecular underpinning of phenotypes within well-annotated model organisms. Despite the fact that multivariate statistics are inherently complex and difficult to interpret, there have been growing interests in recent years to develop robust mathematical approaches to reveal coherent biological changes across different omics profiles [13, 86]. Leveraging both approaches represents a promising route toward comprehensive understanding the complex multi-omics dataset.

7 Notes

1. We have reviewed network-based data integration in this chapter, while mechanism-based integration is usually based on metabolic models (*see Chapter 18* by Witting et al.).
2. The common network analysis and visualization algorithms are now implemented in popular software libraries (e.g., igraph (<https://igraph.org>) [87], NetworkX (<https://networkx.github.io/>) [88], Sigma (<http://sigmajs.org>), 3D network (<https://github.com/vasturiano/3d-force-graph>), and graph-

tools (<https://graph-tool.skewed.de/>). Popular desktop based software tools include Cytoscape (<https://cytoscape.org/>) [79] and Gephi (<https://gephi.org/>) [89].

Network-based multi-omics integration tools that are relevant to metabolomics: MetaboAnalyst (<https://metaboanalyst.ca>) [73], XCMS Online (<https://xcmsonline.scripps.edu/>) [72], Metabox (<https://github.com/kwanjeeraw/metabox>) [90], PIUMet (<http://fraenkel-nse.csbi.mit.edu/PIUMet>) [74], HiCoNet (<https://github.com/shuzhao-li/hiconet>), 3Omics (<https://3omics.cmdm.tw/>) [84], Metscape (<http://metscape.ncibi.org>) [91], PaintOmics (<http://www.paintomics.org>) [83], and OmicsNet (<https://omicsnet.ca>) [47].

Acknowledgments

This work has been funded in part by the US National Institutes of Health via grants UH2 AI132345 (Li), R01 GM124061 (Yu), U2C ES030163 (Jones, Li, Morgan, Miller), U01 CA235493 (Li, Xia, Siuzdak), Genome Canada, Genome Quebec, and Canada Research Chairs program.

References

- Hasin Y, Seldin M, Lusis A (2017) Multi-omics approaches to disease. *Genome Biol* 18(1):83
- Coleman WB (2017) Next-generation breast cancer omics. *Am J Pathol* 187(10):2130–2132
- Mach N, Ramayo-Caldas Y, Clark A, Moroldo M, Robert C, Barrey E et al (2017) Understanding the response to endurance exercise using a systems biology approach: combining blood metabolomics, transcriptomics and miR-Nomics in horses. *BMC Genomics* 18(1):187
- Villar M, Ayllon N, Alberdi P, Moreno A, Moreno M, Tobes R et al (2015) Integrated metabolomics, transcriptomics and proteomics identifies metabolic pathways affected by Anaplasma phagocytophilum infection in tick cells. *Mol Cell Proteomics* 14(12):3154–3172
- Rinschen MM, Ivanisevic J, Giera M, Siuzdak G (2019) Identification of bioactive metabolites using activity metabolomics. *Nat Rev Mol Cell Biol* 20:353–367
- Yan J, Risacher SL, Shen L, Saykin AJ (2018) Network approaches to systems biology analysis of complex disease: integrative methods for multi-omics data. *Brief Bioinform* 19(6):1370–1381
- Casci T (2012) Bioinformatics: next-generation omics. *Nat Rev Genet* 13(6):378
- Rattray NJ, Deziel NC, Wallach JD, Khan SA, Vasilious V, Ioannidis JP et al (2018) Beyond genomics: understanding exposotypes through metabolomics. *Hum Genomics* 12(1):4
- Ritchie MD, Holzinger ER, Li R, Pendergrass SA, Kim D (2015) Methods of integrating data to uncover genotype–phenotype interactions. *Nat Rev Genet* 16(2):85
- Chong J, Xia J (2017) Computational approaches for integrative analysis of the metabolome and microbiome. *Metabolites* 7(4):E62
- Gligorijevic V, Przulj N (2015) Methods for biological data integration: perspectives and challenges. *J R Soc Interface* 12(112):20150571
- Meng C, Zelezniak OA, Thallinger GG, Kuster B, Gholami AM, Culhane AC (2016) Dimension reduction techniques for the integrative analysis of multi-omics data. *Brief Bioinform* 17(4):628–641
- Bersanelli M, Mosca E, Remondini D, Giampieri E, Sala C, Castellani G et al (2016) Methods for the integration of multi-omics data:

- mathematical aspects. *BMC Bioinformatics* 17(2):S15
14. Huang S, Chaudhary K, Garmire LX (2017) More is better: recent progress in multi-omics data integration methods. *Front Genet* 8:84
 15. Tini G, Marchetti L, Priami C, Scott-Boyer M-P (2019) Multi-omics integration—a comparison of unsupervised clustering methodologies. *Brief Bioinform* 20(4):1269–1279
 16. Barabasi AL, Gulbahce N, Loscalzo J (2011) Network medicine: a network-based approach to human disease. *Nat Rev Genet* 12(1):56–68
 17. Vidal M, Cusick ME, Barabasi AL (2011) Interactome networks and human disease. *Cell* 144(6):986–998
 18. Mitra K, Carvunis AR, Ramesh SK, Ideker T (2013) Integrative approaches for finding modular structure in biological networks. *Nat Rev Genet* 14(10):719–732
 19. Kanehisa M, Goto S (2000) KEGG: Kyoto encyclopedia of genes and genomes. *Nucleic Acids Res* 28(1):27–30
 20. Fabregat A, Jupe S, Matthews L, Sidiropoulos K, Gillespie M, Garapati P et al (2017) The Reactome pathway knowledgebase. *Nucleic Acids Res* 46 (Database issue):D481–D487
 21. Caspi R, Billington R, Fulcher CA, Keseler IM, Kothari A, Krummenacker M et al (2017) The MetaCyc database of metabolic pathways and enzymes. *Nucleic Acids Res* 46 (Database issue):D633–D639
 22. King ZA, Lu J, Dräger A, Miller P, Federowicz S, Lerman JA et al (2015) BiGG models: a platform for integrating, standardizing and sharing genome-scale models. *Nucleic Acids Res* 44 (Database issue):D515–D522
 23. Brunk E, Sahoo S, Zielinski DC, Altunkaya A, Dräger A, Mih N et al (2018) Recon3D enables a three-dimensional view of gene variation in human metabolism. *Nat Biotechnol* 36(3):272–281
 24. Slenter DN, Kutmon M, Hanspers K, Ruitta A, Windsor J, Nunes N et al (2017) WikiPathways: a multifaceted pathway database bridging metabolomics to other omics research. *Nucleic Acids Res* 46 (Database issue):D661–D667
 25. Altman T, Travers M, Kothari A, Caspi R, Karp PD (2013) A systematic comparison of the MetaCyc and KEGG pathway databases. *BMC Bioinformatics* 14(1):112
 26. Alcántara R, Axelsen KB, Morgat A, Belda E, Coudert E, Bridge A et al (2011) Rhea—a manually curated resource of biochemical reactions. *Nucleic Acids Res* 40(D1):D754–D760
 27. Davis AP, Grondin CJ, Johnson RJ, Sciaky D, McMorran R, Wiegers J et al (2018) The comparative toxicogenomics database: update 2019. *Nucleic Acids Res* 47(D1):D948–D954
 28. Szklarczyk D, Santos A, von Mering C, Jensen LJ, Bork P, Kuhn M (2015) STITCH 5: augmenting protein–chemical interaction networks with tissue and affinity data. *Nucleic Acids Res* 44 (Database issue):D380–D384
 29. Kerrien S, Aranda B, Breuza L, Bridge A, Broackes-Carter F, Chen C et al (2011) The IntAct molecular interaction database in 2012. *Nucleic Acids Res* 40 (Database issue):D841–D846
 30. Chatr-Aryamontri A, Oughtred R, Boucher L, Rust J, Chang C, Kolas NK et al (2017) The BioGRID interaction database: 2017 update. *Nucleic Acids Res* 45 (Database issue):D369–D379
 31. Licata L, Brigandt L, Peluso D, Perfetto L, Iannuccelli M, Galeota E et al (2011) MINT, the molecular interaction database: 2012 update. *Nucleic Acids Res* 40 (Database issue):D857–D861
 32. Breuer K, Foroushani AK, Laird MR, Chen C, Sribnaia A, Lo R et al (2013) InnateDB: systems biology of innate immunity and beyond—recent updates and continuing curation. *Nucleic Acids Res* 41 (Database issue):D1228–D1233
 33. Szklarczyk D, Franceschini A, Wyder S, Forsslund K, Heller D, Huerta-Cepas J et al (2014) STRING v10: protein–protein interaction networks, integrated over the tree of life. *Nucleic Acids Res* 43 (Database issue):D447–D452
 34. Khan A, Fornes O, Stigliani A, Gheorghe M, Castro-Mondragon JA, van der Lee R et al (2018) JASPAR 2018: update of the open-access database of transcription factor binding profiles and its web framework. *Nucleic Acids Res* 46 (Database issue):D260–D266
 35. Agarwal V, Bell GW, Nam JW, Bartel DP (2015) Predicting effective microRNA target sites in mammalian mRNAs. *Elife* 4:e05005
 36. Han H, Shim H, Shin D, Shim JE, Ko Y, Shin J et al (2015) TRRUST: a reference database of human transcriptional regulatory interactions. *Sci Rep* 5:11432
 37. Chou C-H, Shrestha S, Yang C-D, Chang N-W, Lin Y-L, Liao K-W et al (2017) miRTarBase update 2018: a resource for experimentally validated microRNA-target interactions. *Nucleic Acids Res* 46 (Database issue):D296–D302
 38. Karagkouni D, Paraskevopoulou MD, Chatzopoulou S, Vlachos IS, Tatsoglou S, Kanellos I et al (2017) DIANA-TarBase v8: a decade-long collection of experimentally supported

- miRNA–gene interactions. *Nucleic Acids Res* 46 (Database issue):D239–D245
39. Shoemaker RH (2006) The NCI60 human tumour cell line anticancer drug screen. *Nat Rev Cancer* 6(10):813–823
40. Tomczak K, Czerwińska P, Wiznerowicz M (2015) The cancer genome atlas (TCGA): an immeasurable source of knowledge. *Contemp Oncol* 19(1A):A68
41. The Integrative HMP (iHMP) Research Network Consortium (2014) The Integrative human microbiome project: dynamic analysis of microbiome-host omics profiles during periods of human health and disease. *Cell Host Microbe* 16(3):276–289
42. Laakso M, Kuusisto J, Stančáková A, Kuulasmaa T, Pajukanta P, Lusis AJ et al (2017) The metabolic syndrome in men study: a resource for studies of metabolic and cardiovascular diseases. *J Lipid Res* 58(3):481–493
43. Tadaka S, Saigusa D, Motoike IN, Inoue J, Aoki Y, Shirota M et al (2017) jMorp: Japanese multi Omics reference panel. *Nucleic Acids Res* 46(D1):D551–D557
44. Nica AC, Parts L, Glass D, Nisbet J, Barrett A, Sekowska M et al (2011) The architecture of gene regulatory variation across multiple human tissues: the MuTHER study. *PLoS Genet* 7(2):e1002003
45. Perez-Riverol Y, Bai M, da Veiga Leprevost F, Squizzato S, Park YM, Haug K et al (2017) Discovering and linking public omics data sets using the omics discovery index. *Nat Biotechnol* 35(5):406–409
46. Yugi K, Kubota H, Hatano A, Kuroda S (2016) Trans-omics: how to reconstruct biochemical networks across multiple ‘omic’ layers. *Trends Biotechnol* 34(4):276–290
47. Zhou G, Xia J (2018) OmicsNet: a web-based tool for creation and visual analysis of biological networks in 3D space. *Nucleic Acids Res* 46(W1):W514–W522
48. Creixell P, Reimand J, Haider S, Wu G, Shibata T, Vazquez M et al (2015) Pathway and network analysis of cancer genomes. *Nat Methods* 12(7):615–621
49. Akhmedov M, Kedaigle A, Chong RE, Montemanni R, Bertoni F, Fraenkel E et al (2017) PCSF: an R-package for network-based interpretation of high-throughput data. *PLoS Comput Biol* 13(7):e1005694
50. Tuncbag N, Gosline SJ, Kedaigle A, Soltis AR, Gitter A, Fraenkel E (2016) Network-based interpretation of diverse high-throughput datasets through the omics integrator software package. *PLoS Comput Biol* 12(4):e1004879
51. Ideker T, Ozier O, Schwikowski B, Siegel AF (2002) Discovering regulatory and signalling circuits in molecular interaction networks. *Bioinformatics* 18(suppl_1):S233–S240
52. Khurana V, Peng J, Chung CY, Auluck PK, Fanning S, Tardiff DF et al (2017) Genome-scale networks link neurodegenerative disease genes to α -synuclein through specific molecular pathways. *Cell Syst* 4(2):157–170. e14
53. Sychev ZE, Hu A, DiMaio TA, Gitter A, Camp ND, Noble WS et al (2017) Integrated systems biology analysis of KSHV latent infection reveals viral induction and reliance on peroxisome mediated lipid metabolism. *PLoS Pathog* 13(3):e1006256
54. Beisser D, Klau GW, Dandekar T, Müller T, Dittrich MT (2010) BioNet: an R-package for the functional analysis of biological networks. *Bioinformatics* 26(8):1129–1130
55. Alcaraz N, List M, Dissing-Hansen M, Rehmsmeier M, Tan Q, Mollenhauer J et al (2016) Robust de novo pathway enrichment with KeyPathwayMiner 5. *F1000Res* 5:1531
56. Anvar MS, Minuchehr Z, Shahlaei M, Kheitan S (2018) Gastric cancer biomarkers; a systems biology approach. *Biochem Biophys Rep* 13:141–146
57. Jha AK, Huang SC-C, Sergushichev A, Lampropoulou V, Ivanova Y, Loguinicheva E et al (2015) Network integration of parallel metabolic and transcriptional data reveals metabolic modules that regulate macrophage polarization. *Immunity* 42(3):419–430
58. Chen X, Liu M-X, Yan G-Y (2012) Drug–target interaction prediction by random walk on the heterogeneous network. *Mol BioSyst* 8(7):1970–1978
59. Liu Y, Zeng X, He Z, Zou Q (2017) Inferring microRNA–disease associations by random walk on a heterogeneous network with multiple data sources. *IEEE/ACM transactions on computational biology. Bioinformatics* 14(4):905–915
60. Chen X, You Z-H, Yan G-Y, Gong D-W (2016) IRWRLDA: improved random walk with restart for lncRNA–disease association prediction. *Oncotarget* 7(36):57919
61. Subramanian A, Tamayo P, Mootha VK, Mukherjee S, Ebert BL, Gillette MA et al (2005) Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proc Natl Acad Sci U S A* 102(43):15545–15550
62. Tarca AL, Draghici S, Khatri P, Hassan SS, Mittal P, Kim JS et al (2009) A novel signaling pathway impact analysis. *Bioinformatics* 25(1):75–82

63. Alexeyenko A, Lee W, Pernemalm M, Guegan J, Dessen P, Lazar V et al (2012) Network enrichment analysis: extension of gene-set enrichment analysis to gene networks. *BMC Bioinformatics* 13(1):226
64. Glaab E, Baudot A, Krasnogor N, Schneider R, Valencia A (2012) EnrichNet: network-based gene set enrichment analysis. *Bioinformatics* 28(18):i451–i457
65. Dettmer K, Aronov PA, Hammock BD (2007) Mass spectrometry-based metabolomics. *Mass Spectrom Rev* 26(1):51–78
66. da Silva RR, Dorrestein PC, Quinn RA (2015) Illuminating the dark matter in metabolomics. *Proc Natl Acad Sci U S A* 112(41):12549–12550
67. Albert R, Jeong H, Barabasi AL (2000) Error and attack tolerance of complex networks. *Nature* 406(6794):378–382
68. Li S, Park Y, Duraisingham S, Strobel FH, Khan N, Soltow QA et al (2013) Predicting network activity from high throughput metabolomics. *PLoS Comput Biol* 9(7):e1003123
69. Xu X, Araki K, Li S, Han JH, Ye L, Tan WG et al (2014) Autophagy is essential for effector CD8(+) T cell survival and memory formation. *Nat Immunol* 15(12):1152–1161
70. Li S, Todor A, Luo R (2016) Blood transcriptomics and metabolomics for personalized medicine. *Comput Struct Biotechnol J* 14:1–7
71. Stewart CJ, Embleton ND, Marrs ECL, Smith DP, Fofanova T, Nelson A et al (2017) Longitudinal development of the gut microbiome and metabolome in preterm neonates with late onset sepsis and healthy controls. *Microbiome* 5(1):75
72. Huan T, Forsberg EM, Rinehart D, Johnson CH, Ivanisevic J, Benton HP et al (2017) Systems biology guided by XCMS online metabolomics. *Nat Methods* 14(5):461–462
73. Chong J, Soufan O, Li C, Caraus I, Li S, Bourque G et al (2018) MetaboAnalyst 4.0: towards more transparent and integrative metabolomics analysis. *Nucleic Acids Res* 46(W1):W486–W494
74. Pirhaji L, Milani P, Leidl M, Curran T, Avila-Pacheco J, Clish CB et al (2016) Revealing disease-associated pathways by network integration of untargeted metabolomics. *Nat Methods* 13(9):770
75. Rohart F, Gautier B, Singh A, Lê Cao K-A (2017) mixOmics: an R package for ‘omics feature selection and multiple data integration. *PLoS Comput Biol* 13(11):e1005752
76. Li S, Sullivan NL, Rouphael N, Yu T, Banton S, Maddur MS et al (2017) Metabolic phenotypes of response to vaccination in humans. *Cell* 169(5):862–877. e17
77. Gardinassi LG, Arévalo-Herrera M, Herrera S, Cordy RJ, Tran V, Smith MR et al (2018) Integrative metabolomics and transcriptomics signatures of clinical tolerance to *Plasmodium vivax* reveal activation of innate cell immunity and T cell signaling. *Redox Biol* 17:158–170
78. Pavlopoulos GA, Malliarakis D, Papanikolaou N, Theodosiou T, Enright AJ, Iliopoulos I (2015) Visualizing genome and systems biology: technologies, tools, implementation techniques and trends, past, present and future. *Gigascience* 4(1):38
79. Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D et al (2003) Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res* 13(11):2498–2504
80. Alcaraz N, Pauling J, Batra R, Barbosa E, Junge A, Christensen AG et al (2014) KeyPathway-Miner 4.0: condition-specific pathway analysis by combining multiple omics studies and networks with Cytoscape. *BMC Syst Biol* 8(1):99
81. Kutmon M, van Iersel MP, Bohler A, Kelder T, Nunes N, Pico AR et al (2015) PathVisio 3: an extendable pathway analysis toolbox. *PLoS Comput Biol* 11(2):e1004085
82. Luo W, Pant G, Bhavnasi YK, Blanchard SG Jr, Brouwer C (2017) Pathview web: user friendly pathway visualization and data integration. *Nucleic Acids Res* 45(W1):W501–W508
83. Garcia-Alcalde F, Garcia-Lopez F, Dopazo J, Conesa A (2010) Paintomics: a web based tool for the joint visualization of transcriptomics and metabolomics data. *Bioinformatics* 27(1):137–139
84. Kuo T-C, Tian T-F, Tseng YJ (2013) 3Omics: a web-based systems biology tool for analysis, integration and visualization of human transcriptomic, proteomic and metabolomic data. *BMC Syst Biol* 7(1):64
85. Sommer B, Baaden M, Krone M, Woods A (2018) From virtual reality to immersive analytics in Bioinformatics. *J Integr Bioinform* 15(2):20180043
86. Argelaguet R, Velten B, Arnol D, Dietrich S, Zenz T, Marioni JC et al (2018) Multi-Omics factor analysis-a framework for unsupervised integration of multi-omics data sets. *Mol Syst Biol* 14(6):e8124
87. Csardi G, Nepusz T (2006) The igraph software package for complex network research. *InterJournal, Complex Systems* 1695(5):1–9
88. Hagberg A, Swart P, S Chult D (2008) Exploring network structure, dynamics, and function using NetworkX (No. LA-UR-08-05495; LA-

- UR-08-5495). Los Alamos National Lab. (LANL), Los Alamos, NM (United States)
89. Bastian M, Heymann S, & Jacomy M (2009) Gephi: an open source software for exploring and manipulating networks. In: Third international AAAI conference on weblogs and social media
90. Wanichthanarak K, Fan S, Grapov D, Barupal DK, Fiehn O, Orešić M (2017) Metabox: a toolbox for metabolomic data analysis, interpretation and integrative exploration. *Plos One* 12(1):e0171046
91. Gao J, Tarcea VG, Karnovsky A, Mirel BR, Weymouth TE, Beecher CW, Cavalcoli JD, Athey BD, Omenn GS, Burant CF, Jagadish HV (2010) Metscape: a Cytoscape plug-in for visualizing and interpreting metabolomic data in the context of human metabolic networks. *Bioinformatics* 26(7):971–973

INDEX

A

- Alignment 6, 11–15, 17, 22, 26, 27, 30, 38, 40, 42, 45, 54, 55, 58, 77, 79, 87, 88, 212, 221, 231, 240, 246, 341
Annotation v, 6, 9, 12, 13, 19–21, 26, 47, 76, 87, 122, 123, 131, 141, 150–152, 185–204, 209–223, 227–241, 350, 363, 372, 389, 393–397, 429, 435–437, 439, 452, 453, 471, 475, 477
Automated data analysis pipeline (ADAP) 25–47

B

- Bioinformatics 64, 66, 78, 89, 90, 141, 228, 245–263, 265, 272, 299, 337, 358, 388, 392, 413, 450, 453, 470
Biomarkers 8, 63, 69, 79–83, 88, 90, 149, 185, 313, 314, 317, 319, 325, 341, 343, 401, 402, 404–406, 408, 409, 412, 413, 448, 454, 455, 459, 473, 475
Blood 9, 70, 71, 105, 113, 124, 125, 178, 357, 401–403, 407–412, 421, 455, 456, 460, 473, 478, 479

C

- Cardiometabolic disease 401–414
Cardiovascular disease 401, 404
Cloud computing 87, 90, 248–250

D

- Data analysis v, 5, 7, 11, 22, 25, 26, 49–51, 58, 63, 66, 67, 69, 72, 75–89, 246, 254, 255, 257, 261, 266, 314, 337–358, 384, 392, 397, 413, 422, 442, 470
Databases v, 2, 12, 56, 64, 123, 140, 149, 165, 185, 210, 248, 347, 363, 388, 422, 453, 470
Data exploration 165–183, 212, 247, 260, 279, 342–344, 354, 355, 392, 480, 481
Data integration 9, 26, 337–358, 361–385, 396, 411, 459, 472, 479, 482
Data management 49, 246, 247, 251–254, 256, 265

- Data processing v, 5, 11–22, 49–60, 79, 149, 239, 337, 338, 340, 341, 413, 419, 420, 426, 428, 430, 451, 452, 454, 456, 459, 460
Data science 7, 245–263, 265–311, 413
Data visualization 7, 51, 202, 246, 247, 256, 257, 266, 270, 279, 361–385, 392, 419, 420, 442
Disease 2, 63, 99, 121, 139, 166, 227, 313, 353, 391, 401, 420, 447, 469
Docker 7, 246, 249, 256, 265–311
Drug mechanisms 8, 419–444

E

- Enrichment analysis 85, 89, 123, 127, 133, 338, 346–348, 350, 351, 354, 357, 472, 474–476, 478

- Environmental health 447–461
Exposome v, 5, 8, 150, 447–461, 479
Extraction 67, 70, 80, 122, 421

F

- Feature annotation 227–241, 422
Feature extraction 219
Flux 66, 99–102, 104, 108, 109, 111–115, 157, 362–365, 374, 379, 380, 383, 384, 388

G

- Gas chromatography mass spectrometry (GC-MS) 3, 6, 7, 25–27, 38, 41–47, 141, 179, 181, 187, 210, 214
Genome scale metabolic networks 8, 351, 361–385
Git 246, 253, 265–311
Global Natural Product Social Molecular Networking (GNPS) 143, 150, 200, 228–230, 233–241, 455

H

- Hierarchical community networks 470, 478–480
Human v, 1, 69, 122, 165, 227, 256, 322, 349, 365, 390, 405, 421, 448, 471
Human health v, 122, 451, 456, 461

- Human metabolome database
 (HMDB) 6, 56, 58, 84, 140, 141, 144, 145, 150, 165–170, 173, 175, 176, 178–183, 199, 390, 437, 453, 455
- I**
- IDEOM 419–444
- Integration 9, 12, 22, 25, 87, 137, 197, 202, 280, 351, 354, 361–385, 395, 396, 411, 453, 459, 470, 472–478, 482
- Interpretation v, 8, 87, 88, 102, 108, 112, 114, 121–124, 133, 146, 221, 330, 337–358, 378, 421, 456, 469, 477–479
- Isotope 8, 12, 13, 52, 53, 62, 66, 71, 84, 85, 101–103, 113, 115, 116, 156–158, 186–189, 191–193, 210, 221, 231, 232, 240, 257, 388, 429, 434, 435, 438–440, 443, 444, 453, 460, 479
- L**
- LipidMaps 122, 123, 125, 128–131, 133, 145, 437, 453, 455
- Lipidome 122, 124–134
- Lipidomics 5, 9, 116, 121–134
- Liquid chromatography (LC) 2, 4, 13, 25, 210, 322, 452
- Liquid chromatography-mass spectrometry
 (LC-MS) 210, 227, 419–444, 450
- Liquid-chromatography tandem mass spectrometry
 (LC-MS/MS) 13, 140, 145, 179, 181, 185, 187, 227, 229
- M**
- Mass spectrometry (MS) 3, 5–9, 13, 14, 21, 25, 30, 49, 63, 66, 87, 88, 115, 121, 122, 134, 143, 145, 149, 160, 161, 166, 204, 209–223, 257, 337, 372, 393, 422, 448, 477
- MetaboAnalyst 8, 41, 87, 261, 337–358, 392, 395, 397, 477, 478, 481
- Metabolic activity 100, 112, 115
- Metabolic modification 161
- Metabolic networks 21, 66, 89, 351, 361–385, 470
- Metabolite databases 6, 139–147, 149, 425, 427, 433, 444, 453–455
- Metabolite identification 7, 8, 21, 26, 56–58, 71, 72, 76, 78, 83–86, 88, 139–147, 149–155, 157, 158, 161, 170, 180, 185–201, 204, 211, 350, 395, 412, 413, 419, 422, 427, 428, 440, 451, 453–456, 477
- Metabolomics v, 2, 11, 25, 49, 61, 139, 149, 165, 185, 209, 227, 246, 265, 313, 337, 362, 388, 402, 419, 448, 469
- Metabotyping 62–65, 68–70, 73, 80–82, 88, 90
- METLIN 6, 12, 20, 21, 140, 141, 149–161, 453–455
- MetScape 89, 390, 393, 394, 397
- Microsoft Excel 239, 419, 422, 424
- Mode of action 420, 421, 423
- Molecular formula 7, 57, 139, 151, 153, 155, 183, 185–204
- Molecular networking 6, 211, 227–241
- MS/MS spectra 2, 3, 6, 20, 21, 57, 140–144, 150–159, 161, 183, 187, 188, 228, 229, 237, 239, 241, 454–456, 460
- Multi-omics integration 19, 353–358, 396, 459, 469–483
- Multivariate data analysis 14, 81–83, 86–88, 343–346
- Multivariate statistics 8, 38, 87, 344, 357, 482
- Mummichog 8, 21, 140, 146, 211, 350–353, 389, 390, 395–397, 456, 477, 478
- MZmine 6, 11, 25–47, 187, 229, 230, 232, 235, 236, 238–241, 256, 430, 452
- N**
- Natural products 143, 210, 228, 455
- Network enrichment analysis 472, 474, 476
- Network visualization 357, 393, 480, 481
- Nuclear magnetic resonance
 (NMR) 3, 61, 144, 166, 337, 450
- O**
- OpenMS 6, 49–59, 187, 256
- P**
- Pathway 3, 21, 63, 99, 130, 140, 150, 166, 210, 228, 261, 338, 362, 387, 424, 450, 471, 505
- Pathway analysis 5, 8, 21, 87–89, 346, 348, 349, 352, 354–357, 363, 387–397, 481
- Peak 4, 11, 26, 50, 65, 104, 139, 153, 179, 187, 210, 231, 251, 315, 338, 393, 412, 420, 448, 477
- Performance metrics 324, 325
- Plant 5, 6, 90, 209–223, 392, 471

- Plasma 63, 65, 68–70, 72, 73, 100, 104, 105, 108, 112, 116, 122, 124–127, 257, 322, 325, 405–407, 411, 479
- Precision medicine 1, 2, 9, 473
- Predictive modeling 38, 265, 279, 313–331
- Preprocessing 5, 6, 25–47, 57, 67, 76, 78–81, 87, 187–189, 191, 320, 341, 358, 432, 451–453
- Processing v, 5, 11–22, 26, 28, 35, 49–59, 66, 79, 86–88, 149, 187, 210, 219, 229, 239, 240, 249, 256, 262, 307, 337, 338, 340–343, 365, 394, 395, 410, 411, 413, 419, 420, 423, 426–428, 430–442, 444, 451, 452, 454, 456, 459, 460
- Python 49, 51, 79, 81, 87, 246, 250, 254–256, 261, 265–311, 364, 365, 375, 379, 380, 384, 395
- Q**
- Quality control 7, 22, 50, 67, 72–75, 82, 88, 256, 257, 260, 422
- R**
- R 18, 22, 86, 143, 213–218, 220, 223, 246, 254, 265–311, 338, 358, 364, 373, 375, 385, 425, 426, 430, 453, 458
- Recommendation 7, 101, 322
- Renal disease 401, 402, 408, 409
- Replication 403, 407, 412
- Reproducible science 3, 6, 61, 66, 262
- S**
- Scripting 49, 51, 246, 249, 254–256
- Secondary metabolism 185
- Sirius 57, 58, 185–204
- Specialized metabolites 209–223, 227, 228
- Spectra 2, 12, 26, 50, 65, 139, 150, 166, 185, 211, 228, 307, 314, 338, 451, 477
- Spectral database 140, 141, 143–145, 149–161
- Structure elucidation 64, 186, 188, 200, 202
- Structure prediction 6, 140
- Study design 1–9, 67, 101–112, 327, 358, 395, 403, 405, 407, 409, 413, 452, 461
- Supervised learning 314, 315, 320, 325, 331
- Systems biology 12, 19, 21, 66, 337, 365, 470, 478, 481
- Systems medicine 8, 9
- T**
- Tandem mass spectrum 139
- U**
- Unknown metabolite 84, 150–156, 158, 210, 212, 216, 217, 408
- Untargeted 8, 11, 13, 20, 25–27, 35, 47, 52–26, 58, 64, 67, 69, 85, 86, 121–133, 141, 151, 157, 161, 186, 314, 315, 322, 338, 350–353, 358, 387–398, 412, 419, 444, 448, 450, 451, 453, 459, 460, 479
- Untargeted metabolomics 11, 13, 25, 27, 35, 69, 141, 151, 157, 314, 338, 350–353, 358, 387–398, 419, 450, 451, 453, 459, 460, 479
- V**
- Version control 7, 246, 253, 254, 265, 266, 280–294
- Virtualization 246, 249, 265, 310
- Visualization 7, 22, 27, 28, 30–32, 41, 47, 50, 51, 53, 81, 86–89, 146, 189, 202, 219, 220, 223, 228, 230, 231, 236, 238, 241, 246, 247, 256, 260, 261, 265, 266, 279, 281, 307, 314, 345, 356, 357, 361–385, 392, 393, 397, 419, 427, 442, 470, 480, 482
- W**
- Web server 84, 141, 143, 144, 146, 249, 338, 358
- Workflows 11–15, 18, 26, 27, 41–47, 49, 51, 52, 57, 58, 59, 64, 66–72, 79, 86–90, 151, 187, 202–204, 227–230, 234–236, 238, 240, 241, 246, 281, 298, 300, 338, 341–353, 358, 374, 384, 419–444, 453, 474
- X**
- XCMS 6, 11–22, 35, 210, 213, 214, 249, 256, 272, 307, 308, 315, 373, 395, 397, 420, 422, 425–247, 444, 452–454, 477, 483