Analytica Chimica Acta 1044 (2018) 66–76

Contents lists available at ScienceDirect

Analytica Chimica Acta

journal homepage: www.elsevier.com/locate/aca

KIMBLE: A versatile visual NMR metabolomics workbench in KNIME

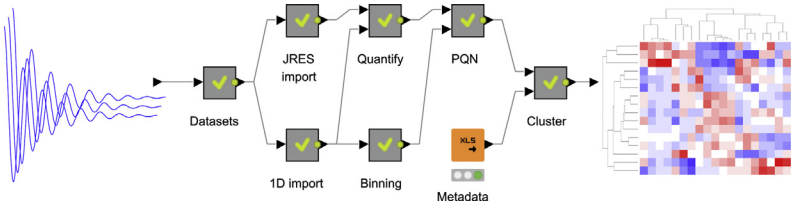
Aswin Verhoeven*, Martin Giera, Oleg A. Mayboroda

Center for Proteomics and Metabolomics, Leiden University Medical Center, Albinusdreef 2, 2333ZA, Leiden, The Netherlands

HIGHLIGHTS

- KIMBLE is a novel KNIME-based NMR metabolomics workflow platform.
- KIMBLE supports both targeted and untargeted NMR metabolomics.
- KIMBLE can be extended and adapted to specific projects and biofluids.
- KIMBLE is self-documenting and combines data, algorithms and software in one file.

GRAPHICAL ABSTRACT



ARTICLE INFO

Article history:
Received 9 April 2018
Received in revised form
25 July 2018
Accepted 26 July 2018
Available online 30 July 2018

Keywords:
Nuclear magnetic resonance
Metabolomics
Metabolite quantification
Workflow
Virtual machine
Biofluid

ABSTRACT

The problem of reproducibility of scientific research is a serious issue in biomedical sciences. In addition to experimental repeatability, limiting the (pre-) analytical variance is also essential. To address this problem in the field of metabolomics, we have designed KIMBLE, the KNIME-based Integrated MetaBoLomics Environment, a novel platform for the processing and analysis of NMR metabolomics data. It consists of an elaborate NMR metabolomics workflow in the KNIME workflow management system that handles both targeted and untargeted metabolomics. The workflow provides a self-documenting way of transforming raw time-domain NMR data into metabolic insights. Parameters for the quantification of a number of interesting metabolites in urine are included in the workflow, and several useful statistical analysis and visualization tools are incorporated as well. The workflow comes with an interesting sports-induced ketosis dataset so that new users can easily get acquainted with the platform. The user is free to adapt and extend the workflow to his or her personal needs. The KIMBLE workflow, the KNIME software and all the required libraries are installed in a VirtualBox virtual machine that allows for facile installation and use by non-experts.

1. Introduction

Over the last two decades the “omics” disciplines (genomics, transcriptomics, proteomics and metabolomics) gradually became an indispensable part of biomedical research practice. As a consequence the biological and medical sciences became increasingly exposed to complex analytical technologies such as e.g. NMR [1,2]. These developments have also led to increasing complexity and diversity of the obtained data and the necessary data analysis approaches [3]. This, in turn, brings the question of data consistency into focus. The technical solutions for controlling the analytical or/and pre-analytical variance are well covered in the literature [4]. However, the degree of variance introduced by various data processing workflows remains largely unexplored. For instance, several software packages, both commercial and free, are currently in use for the processing and analysis of NMR metabolomics datasets. The processing of the Free Induction Decay (FID) typically happens in the spectrometer software, for example with the Topspin software from Bruker Biospin [5]. Untargeted metabolomics can be performed in Bruker's Amix software, Automics [6], or performed with Matlab [7], R, or Python code. Targeted analysis can be performed using the commercial Chenomx [8] software or

* Corresponding author.
E-mail address: a.verhoeven@lumc.nl (A. Verhoeven).

https://doi.org/10.1016/j.aca.2018.07.070

the open-source BATMAN [9] package. Subsequent univariate or multivariate analysis can take place in Simca [10], or in self-written scripts in one of the aforementioned languages. Another option is to use a web-based metabolomics tool, such as Metaboanalyst [11]. The selection of an optimal combination of tools is not straightforward. Suboptimal choices, which include multiple, largely undocumented user- or workgroup-specific protocols, affect the reproducibility of data analysis and make it prone to errors. A possible solution for improving the transparency and shareability of data analysis is to organize all data processing steps in a single workflow within a workflow management system. The task of a workflow manager is to provide an integrated and self-documenting data handling platform, bringing a diverse range of software tools together under one umbrella.

Several initiatives with this goal are already underway. NMRProcFlow [12] is a tool dedicated to the processing of NMR spectra but it is not flexible nor easily extendible. Pathomx [13] has an esthetically pleasing workflow editor, but internally uses Pandas dataframes and is therefore limited by the system memory. A metabolomics workflow built on top of a more general workflow platform, such as Taverna [14] or Galaxy [15], would offer superior flexibility. This is the approach taken by Workflow4Metabolomics [16], which is built on top of the Galaxy platform. It provides a web-based application and supplies user friendly tools and a workflow editor canvas to combine those. However, it does not allow loops and the process of adding your own tools and scripts to the workflow is rather complex.

Many of the aforementioned shortcomings can be bypassed with the KNIME platform. KNIME is an open-source data mining and workflow management system developed by the KNIME AG in Zurich, Switzerland, and the University of Konstanz [17]. The basic KNIME Analytics Platform is free to use, and its components are licensed under various open-source licenses. KNIME allows the user to process and manipulate data tables using a system of nodes and connectors that can be visually arranged in a workflow editor. Every node represents an operation on the data table, while every connector transfers the result of one node to the input of the next node. A large number of tools are available in the node repository, including nodes for loop constructs, data filtering, statistical tests, machine learning and visualization. This way, even users with no coding experience can construct elaborate data processing workflows. For more advanced users, Java, Python and R scripting nodes are available. Recently, dedicated mass-spectrometry (MS) nodes have been added to the KNIME node repository [18], and these have been used to implement a workflow for MS metabolomics [19].

Here, we present a novel NMR metabolomics data processing workflow built on the KNIME platform – KIMBLE: the KNIME-based Integrated MetaBoLomics Environment. The workflow contains all steps for processing and transforming raw NMR time domain data into a metabolomics dataset (bins and metabolite concentrations), connect it with metadata, and perform statistical analyses. Moreover, by installing KNIME, the workflow, and all necessary Python and R libraries in a virtual machine (VM), the platform can easily be shared/distributed as a single entity: the virtual machine image. The only requirement is that the user's computer satisfies the system requirements and has Oracle VirtualBox installed, which is available for free for most common desktop operating systems.

To illustrate the versatility of the platform, we have implemented several advanced metabolomics algorithms in convenient KNIME nodes and integrated these into a single workflow. It contains nodes for procedures such as adaptive binning and PQN normalization. The KIMBLE workflow can be used as-is, or can be easily extended by the user. Merging metadata with the experimental data can be accomplished with two or three nodes, and various univariate and multivariate statistical analysis tools are available for the analysis of the combined table. However, the most important part of KIMBLE is the node containing an algorithm for metabolite quantification. Thus, apart from the sample preparation and NMR measurements themselves, the whole metabolomics workflow is integrated into one platform that can be easily modified and extended for specific purposes.

The present version of KIMBLE is tailored for urine metabolomics. However, it can be adapted and extended to other biofluids. In addition to project-specific workflow adaptations, users will often find that their workflow extensions are broadly applicable. These extensions can be integrated with the KIMBLE template and shared with colleagues and the broader scientific community. For example, a future branch for mass-spectrometry based metabolomics could be combined with the existing NMR workflow to form a powerful combined system.

2. Results

An outline of the KIMBLE workflow is presented in Fig. 1. The workflow consists of a number of interconnected “nodes”. Each node represents an operation on data. Most of the nodes in the main workflow are metanodes, an abstraction mechanism in which a subworkflow of more basic nodes act as a single node. The nodes can have functions such as data import, data processing and data visualization. The nodes are attached to each other with connector lines, each line representing a single data table being handed from one node to the next. The black connection points on the left side of a node are user interface (UI) elements for input tables, those on the right side of the node represent the output tables. An example of a node with its input and output table connections is given in Fig. 2. The various stages of the workflow in Fig. 1 are indicated by the background colors. The workflow reflects the routine protocol for urinary metabolomics that is currently in use in our laboratory in which two NMR experiments are performed on every sample: NOESY1D [20] and 2D J-resolved spectroscopy (JRES) [21] [22]. The data workflow can be divided into roughly 6 main parts.

At the start of the workflow, indicated in yellow, a list of file paths is created where the experimental data is located. The list of paths is also scanned in order to label the QC samples as such. Here the user can choose to import all the spectra of a project at once, or only a limited subset of the spectra as a test run. The list is shuffled so that the spectra will be imported in random order, which means that the subset of spectra will have approximately the same distribution of patients as the whole project. At this point a subset of spectra is also assigned to be used for PQN normalization and adaptive binning calculations.

Indicated in light blue is the “import” section of the workflow. The time-domain data of the experiments are imported using the Python NMRGlue library [23]. The import script does basic processing such as Fourier transformation, zeroth-order phase correction, baseline correction and setting the chemical shift reference. A shearing transformation is performed on the JRES spectra. Interpolation is used to define a new chemical shift axis that is identical for every imported spectrum so that the spectral points of different spectra line up perfectly. Every recorded spectrum is represented by a row in the output table, with the spectra themselves stored in list cells. The chemical shift axis is stored separately as a regular table. The two tables are carried through the workflow using metanodes with 2 input and output connectors. The spectra are available from the lower connectors, the axis data from the upper. To save space and increase performance, the 2D JRES spectra are not imported at full resolution. Instead, sum projections of three bands in the indirect dimension are calculated in the Python script and loaded into the KIMBLE workflow. The band edges are chosen to sum individual peaks in singlets, doublets and

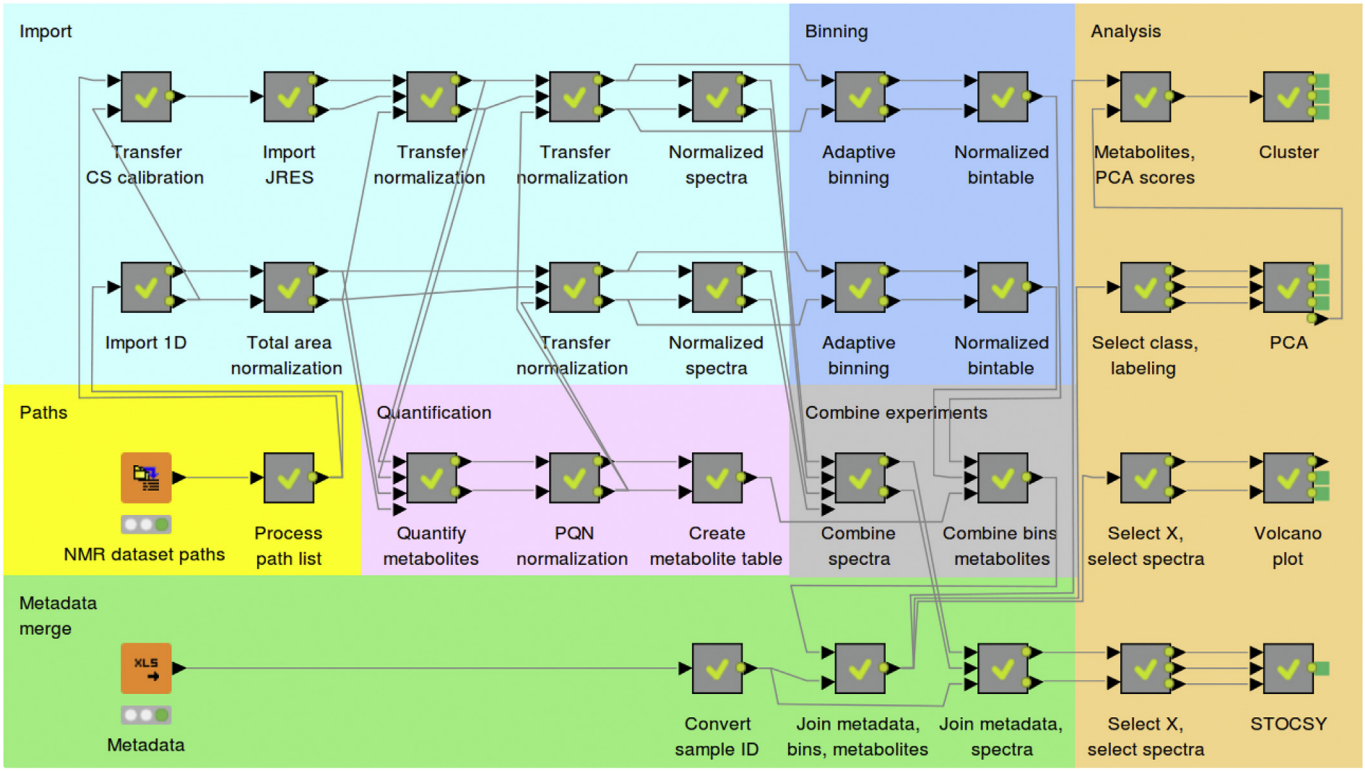


Fig. 1. The KIMBLE workflow. The various functional branches are indicated by color. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

triplets with the most common J-coupling values. This is illustrated in Fig. 3. After import an additional column is added to the axis table to label specific sections of the spectrum that do not represent normal metabolites, such as the part of the spectrum dominated by the water peak, the part of the spectrum that contains the reference peak, the urea peak and areas in the spectrum with no signal or pure baseline (the "Define excluded areas" and "Define urea" metanodes). Subsequently the scaling factor for total area normalization is calculated (metanode "Total area normalization"). The water peak, the reference peak, the urea peak and the baseline areas should not be used for generating this factor; by using the labelling in the previous metanodes these intervals can be easily excluded. The normalization factor is stored as a floating point number in a separate column and is not actually applied to the data at this point. If normalization is required, it can be applied on the fly. The advantage is that the spectral data is not copied, saving a lot of disk I/O and storage space. Normalization factors are generated for the JRES and 1D spectra separately. Combining data obtained from different NMR experiments requires that these are normalized in the same way, therefore in the "Transfer scaling" node the 1D normalization factors are transferred to the JRES branch while the JRES normalization factors are erased.

The part of the workflow in Fig. 1 that performs binning for exploratory analysis is indicated with a dark blue background. In exploratory metabolomics the main component of the data processing approach is data binning, also known as bucketing. Equidistant binning information is already added to the axis table at the import stage. This, however, is a suboptimal form of binning: the bin edges may fall on top of important peaks, making the data very sensitive to peak shifts. To minimize this problem the workflow uses an alternative binning strategy, called adaptive binning. In the "Form adaptive bins" node we use a modified version of the adaptive binning algorithm reported by De Meyer et al. [24]. In brief, the position of minimum standard deviation within every equidistant bin is identified, and the bin is split into two halves. The second half of one bin is then combined with the first half of the next bin, forming a new bin. The result of this procedure can be inspected in the workflow itself and an example is shown in Fig. 4. For the sake of performance, the standard deviation at every point of the spectrum is calculated from a reasonable random subset of all spectra, for example 50. In case of the JRES spectra, this process is applied individually to each of the three bands. The adaptive binning information is stored in the chemical shift axis table, the spectral data is left unaltered. The bin values can be easily obtained by combining the two tables and using the standard KNIME "GroupBy" node. At this point a second normalization factor, namely probabilistic quotient normalization (PQN) that is often performs better than the total area normalization, is merged with the data [25]. This can be calculated from the bins themselves, but we have opted to use actual metabolite concentrations calculated in a different part of the workflow in order to minimize noise. To calculate the PQN normalization factor for each spectrum we firstly use the average metabolite concentration. Then, the ratio of the concentrations in each spectrum with respect to the mean concentration is calculated. Finally, the PQN scaling factor is calculated as the median of these ratios. In the "Adaptive binning" metanode the integral is taken of the sections of the spectra contained in the bins to obtain the bin values. These bin values are stored as list cells in an additional column in the data table. Information about the positions of the bin edges is stored in a separate table; hence the presence of a third output connector of this metanode. The "Normalized bintable" node applies the PQN normalization factor, drops the full resolution spectra from the table, and stores each bin in an easily accessible table column.

In the pink part of the workflow, the quantitation of specific urinary metabolites is performed. This is also known as targeted

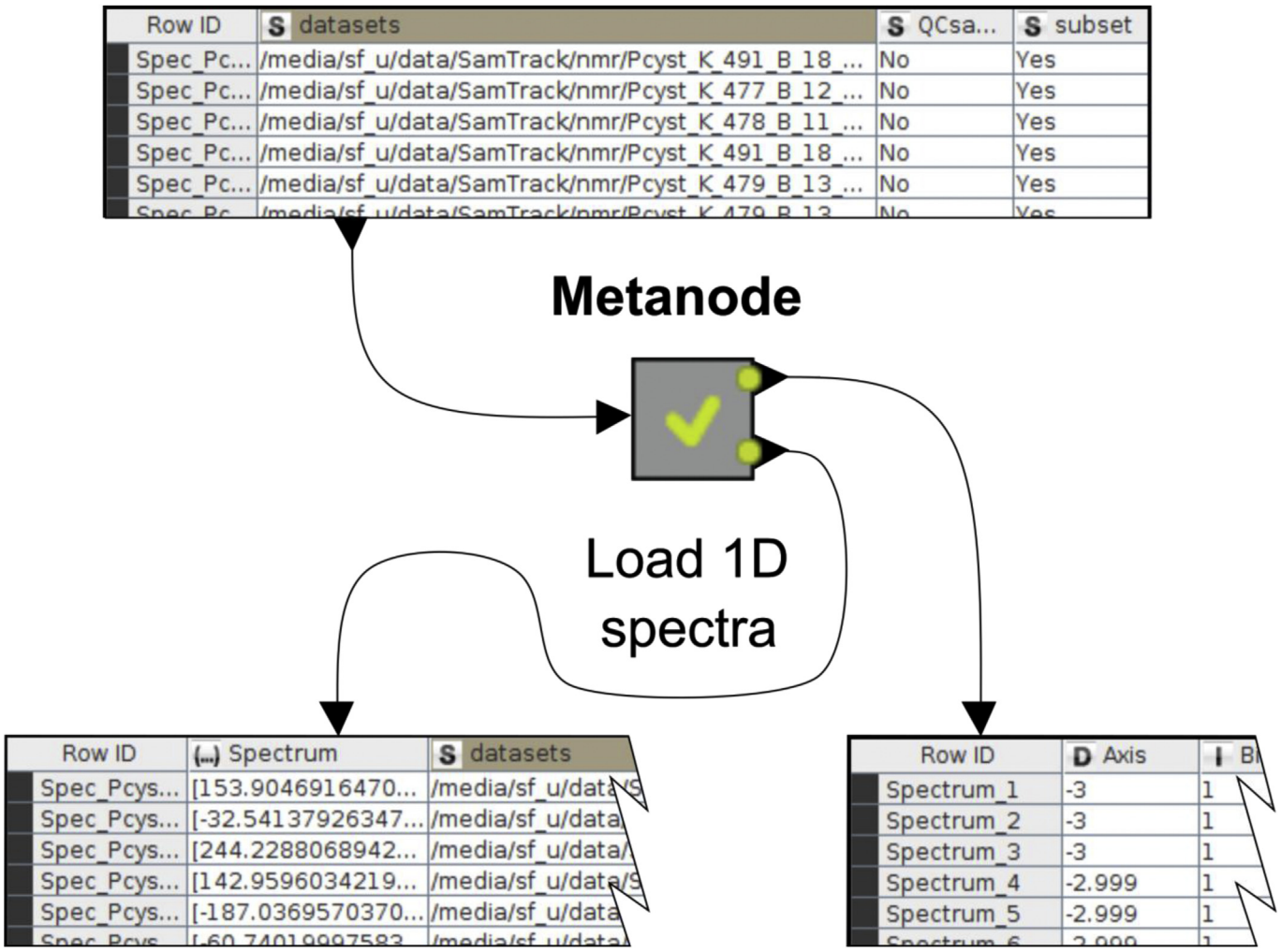


Fig. 2. The table structures of the input and output connectors of the import metanode. The input table contains the paths to the NMR data. The upper output table shows a small section of the chemical shift axis table, while the lower output table contains the NMR spectra.

analysis. The algorithm uses data from both the JRES and the NOESY1D experiments in order to get superior quantification accuracy than can be achieved with the experiments separately. The "Quantify metabolites" metanode embeds a complex network of other nodes and metanodes that represent the quantification algorithm. For each metabolite the algorithm requires a set of parameters that can be entered in the "Metabolite parameters" node. Adding a new metabolite to the quantification algorithm corresponds to adding a new line to the "Metabolite parameters" table. Below is a short summary of this algorithm.

The metanode starts with the analysis of the JRES spectrum which offers superior resolution with respect to the NOESY1D. Of the three available projection bands, it takes the best projection for a given multiplet. The band is chosen by the user and is typically the one with the highest intensity and the least overlap with peaks of other metabolites. In several steps, the baseline around the peak is corrected and the spectra are aligned so that the peaks of one metabolite for different samples line up. The result is a small interval that includes the target peak but nothing more. Subsequently the integral of this interval is taken. A visual inspection of the steps can be performed on the matrix of mini-plots that is one of the outputs of the quantification metanode (see Fig. 5). The procedure is repeated for the 1D spectrum. Here the procedure is typically more difficult than for the JRES because the 1D spectrum is far more crowded. This is expected to lead to more variation and outliers than for the integrals of the JRES peaks. However, the integrals of the 1D peaks give a better representation of the actual metabolite concentration than the JRES integrals because the JRES experiment suffers from loss of polarization during the pulse sequence (due to relaxation and imperfect refocussing) [22]. Therefore the ratio between the area of the TSP peak and the area of the metabolite is ill-defined in the JRES spectrum. Nevertheless, the JRES integrals show a good correlation with the NOESY1D integrals in cases where peak overlap is not an issue. So instead of using the JRES TSP peak to calculate the concentrations, JRES integrals are scaled to the same domain as the NOESY1D integrals. These scaled JRES integrals are then converted to concentrations using the area of the NOESY1D TSP peak. The accuracy of the quantification was assessed by comparing the creatinine concentration as determined from the NMR spectra by the workflow with the clinical chemistry values (see Fig. 6). This comparison shows an excellent agreement, which gives confidence that many more metabolites can be reliably quantified in this manner. These concentrations can be exported or analyzed, but because of the dilution effects of urine, it is often preferable to perform PQN normalization on concentrations, or convert the concentrations to concentration ratios with respect to creatinine.

For a useful statistical analysis the NMR data needs to be combined with metadata, which is what takes place in the part of the

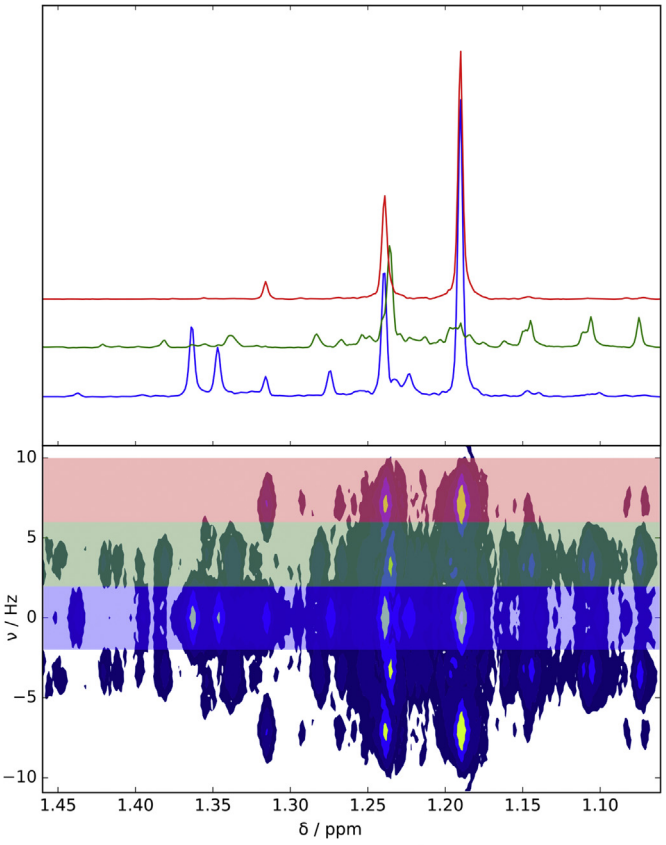


Fig. 3. An example of a part of the JRES spectrum and the three projection bands.

workflow with a green background. If the metadata is in an Excel file, the merging process can be accomplished in KNIME with an Excel reader node and a Joiner node. The metadata and the experimental data do not need to have identically sorted rows, as the tables are combined by matching the sample IDs in user-specified columns, similar to a join operation in a relational database. The metabolite concentrations, the NOESY1D bins, and the JRES bins, can be analyzed separately, or they can be combined in one large table and be analyzed together.

The final part of the workflow, shown in orange, is dedicated to the statistical analysis of the combined metadata and experimental data. As examples, we have implemented metanodes for principal component analysis (PCA) and hierarchical cluster analysis (HCA). The PCA metanode uses the R "PCAmethods" package under the hood. It yields loadings, scores, and an R2/Q2 plot in order to judge the quality of the PCA analysis. The HCA node produces a heatmap which is clustered both with respect to the variables (where it uses the Pearson correlation as a distance measure) and with respect to the samples (based on the Euclidian distance). In this example, the metabolite concentrations are combined with the PCA scores, which allows to associate the PCA scores with specific metabolites. The Statistical Total Correlation Spectroscopy (STOCSY) node shows both the correlation and the covariance of the spectral data with respect to a dependent variable. The volcano plot node shows the relation between the statistical significance of a concentration change with the relative magnitude of the change. There are many other nodes available for statistical analysis, including univariate tests, decision trees, and support vector machines, that can be added to the statistical analysis part of the workflow. As an example, we demonstrate the power of KIMBLE for multivariate analysis by analyzing the effects on the urine metabolome of a fasting person during physical exercise as compared to resting. The exercise segment consists of $3 \times 2 \times 15$ km back-and-forth stretches of cycling, with one sequence of cycling and recovery

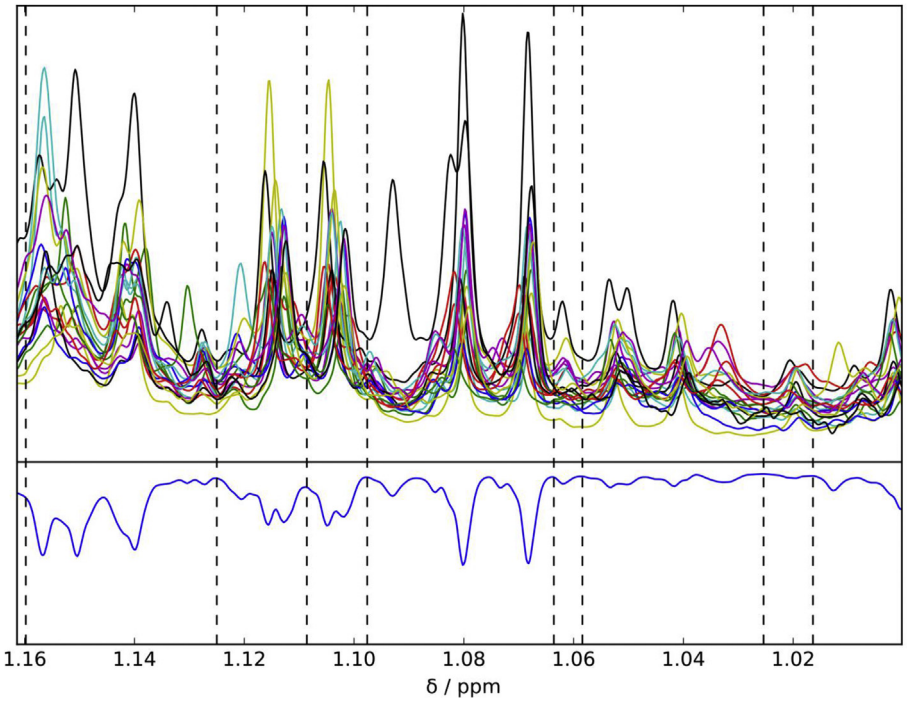


Fig. 4. Multiple urine spectra are shown superimposed. The corresponding standard deviation that the forms the reference spectrum is shown mirrored along the vertical axis in blue. The local minima of this reference spectrum form the bin edges of the adaptive binning process. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

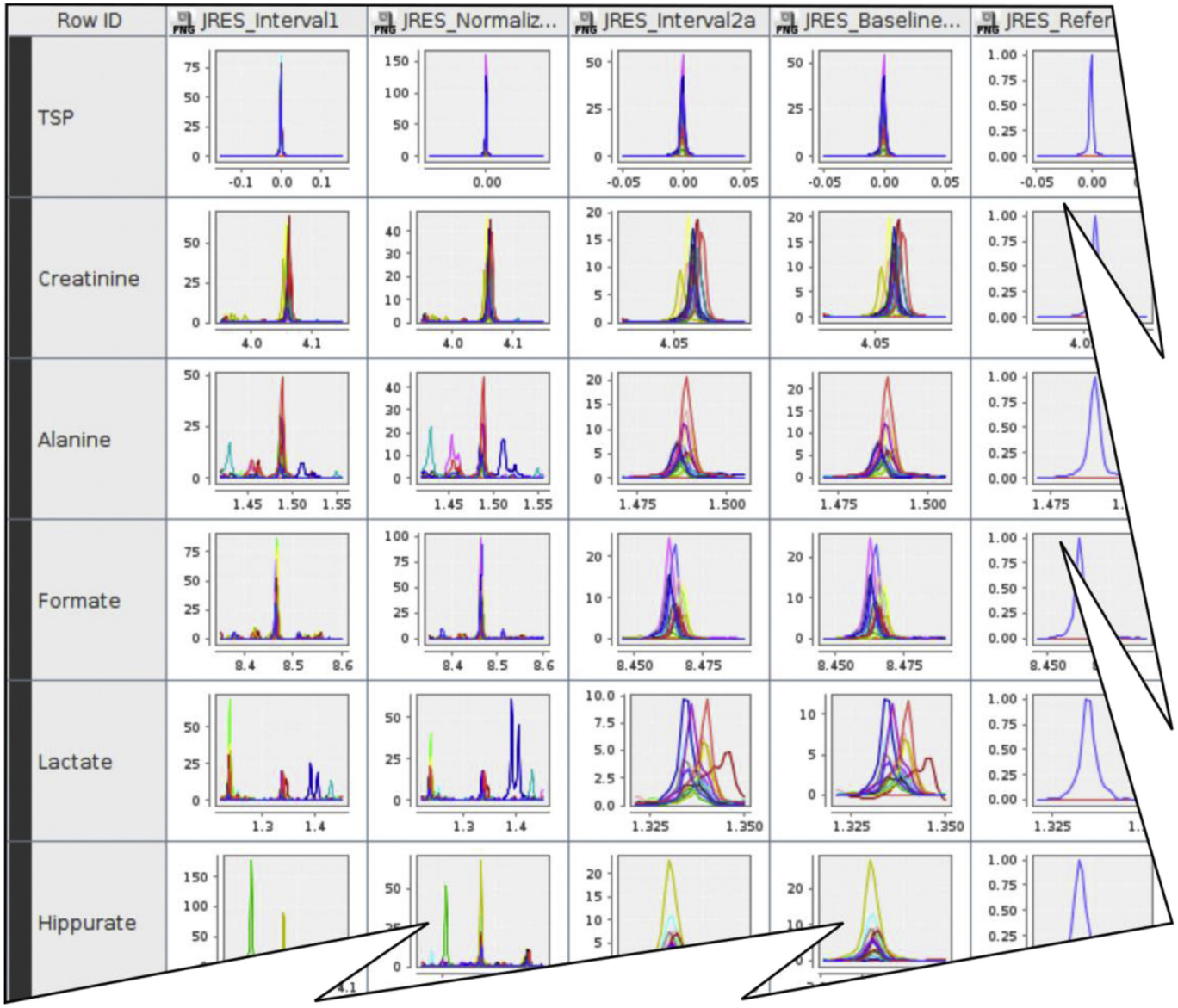


Fig. 5. Small part of the diagnostic output from the quantification metanode. The columns represent steps of the quantification algorithm, progressing from left to right. Every row represents a metabolite. A problem with the quantification of lactate for one of the spectra is clearly visible.

every 1.5 h. After every cycling stretch a urine sample was collected. The resting segment of the project was performed on a different day on the same person and comprised of sitting at a desk while maintaining similar intervals for the urine collection. During the experiment the person only consumed electrolyte drinks to prevent dehydration. The PCA scores plot in Fig. 7 clearly shows the greater spread of scores for the exercise day, with the scores moving away from the resting scores as the cycling effort progresses. The vertical clustergram in the HCA diagram in Fig. 10 clearly show three groups. One group is formed by a pair of urine samples taken after waking up. The second group of observations contains the urine samples taken while resting, but also includes urine samples at early times during exercise day. Finally, there is a third group of urine samples taken at times when physical fatigue becomes obvious. The HCA horizontal clustergram shows the ketone bodies neatly grouped together, and the heatmap shows that after 4 rides the ketone body concentrations start increasing vastly, while the relative concentration of threonine, trigonelline and glycine goes down. The second principal component clearly is associated with these compounds. These results show that at a point, as expected, the body of the rider entered a state of ketosis. This is also reflected in the STOCSY plot constructed from the exercise day spectra in Fig. 8, where the peaks of the ketone bodies clearly stand out. For the study subject ketosis kicks in after 60 km of moderately intense biking while fasting. The volcano plot of the quantified metabolites and the bins shown in Fig. 9 shows that there may be more metabolites that change significantly during exercise, but discussing these is beyond the scope of this article. The data for this simple but interesting mini-project is included in the virtual machine so that the user can easily get familiar with KIMBLE.

3. Discussion

The question of reproducibility (or more correctly lack of reproducibility) of research in the “omics” disciplines is frequently discussed in the literature. Traditionally the analytical or pre-

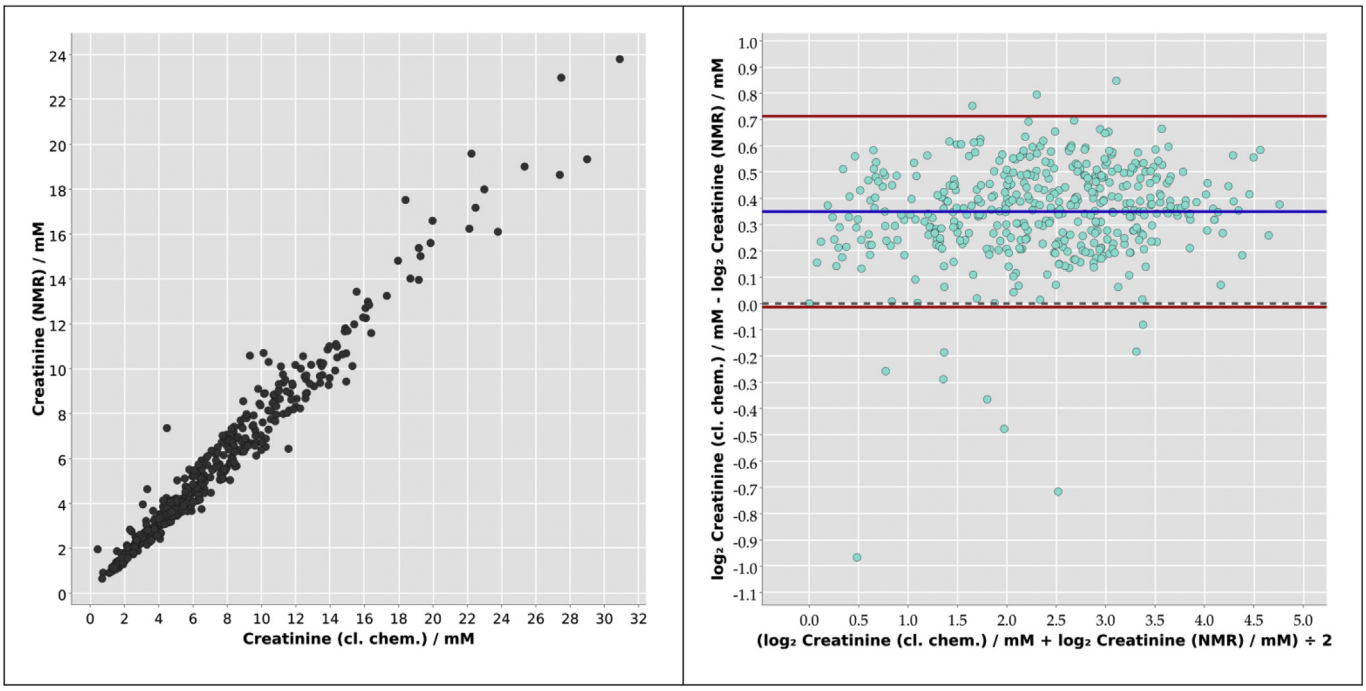


Fig. 6. Creatinine correlation. A) Scatter plot of the creatinine concentrations as determined by standard clinical chemistry methods (x-axis) against the concentrations as determined by NMR (y-axis). B) Logarithmic Bland-Altman plot showing the agreement between the two methods.

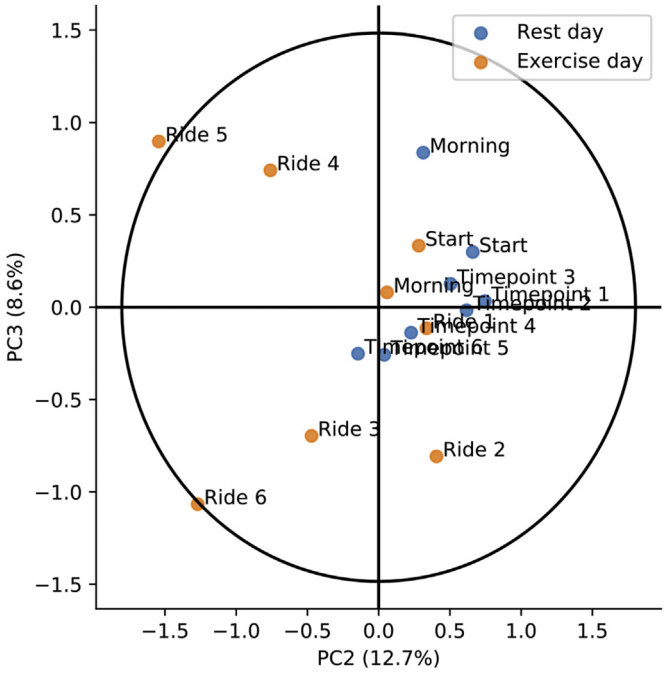


Fig. 7. Scores plot of second and third principal components of the principal component analysis (PCA) of the combined bin table of the JRES and the NOESY1D experiments.

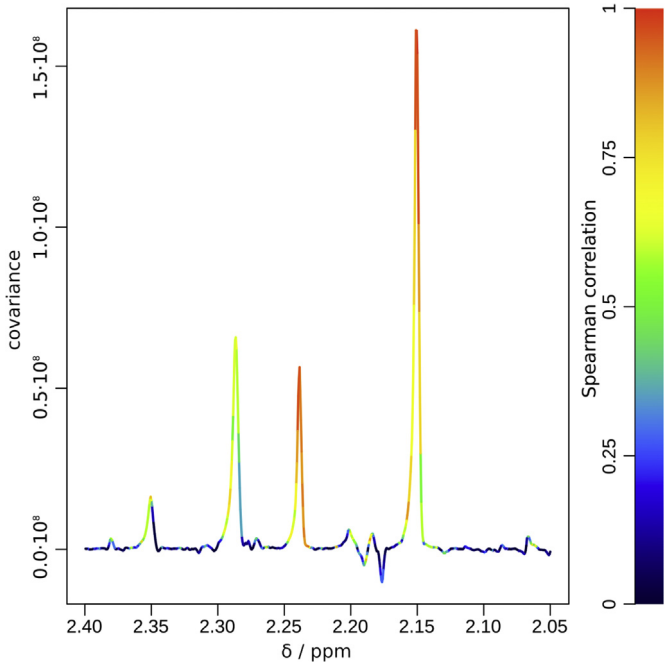


Fig. 8. STOCSY spectrum of the JRES projection selecting singlets and center peaks of multiplets.

analytical factors considered to be the main sources of the variance threatening the reproducibility. However, the data processing workflow can affect the data consistency as well. A continuously growing pool of software packages and data analysis approaches is difficult to control and standardize. The current report presents a possible solution for data analysis standardization. The presented KIMBLE workflow offers a simple, transparent and most importantly shareable solution by calling all required packages from a dedicated workflow manager. The workflow manager allows the data to be inspected after every node, also after subsequent nodes have been executed. The data is always stored with the workflow itself, so that it is always clear how the data is obtained, assuring reproducible data processing and analysis. This could lead to enormous storage requirements, but KNIME does not copy unaltered table columns when processing a node, but keeps referring

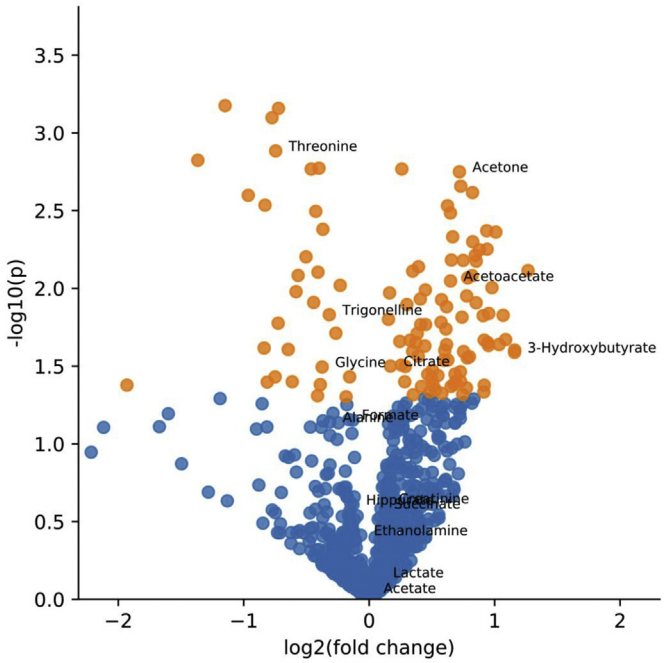


Fig. 9. Volcano plot of the output of the volcano plot metanode that performs a linear regression on all bins and metabolites individually with respect to the timepoint.

to the original copy. The ultimate reproducibility can be achieved by storing a separate virtual machine for every project, so that the workflow and the data is always stored together with the version of KNIME and the Python/R libraries that were used to build the workflow. This can be easily achieved by working in a copy of the original KIMBLE virtual machine by using the clone functionality of VirtualBox. To carry this line of reasoning to its natural conclusion, the manuscript in which the results of the workflow are presented could be composed in a word processor in the virtual machine.

An advantage of KNIME as compared to MATLAB or R is that tables that are too large to fit in memory are partly saved to disk. This makes the workflow more scalable, but disk access is generally slower than RAM access. Tables are therefore stored row by row, making row access of the top rows of a large table fast, but column access much slower. The metabolomics workflow has been constructed in such a way that every observation (NMR spectrum) is represented by a row in the table with the spectrum stored in a list cell, so that the performance of the workflow scales linearly with the size of the projects (see Table 1). List cells can be converted to table columns as needed.

An obvious weakness of our solution is the relatively low execution speed. Indeed, as a java-based platform KNIME is not optimized for fast numerical computations. Moreover, running in a virtual machine and manipulating the main tables largely on disk are slowing the workflow as well. Nevertheless, workflow run speeds are acceptable as can be seen in Table 1. Optimal speed can be achieved by using a fast solid-state drive instead of a traditional hard-disk drive. Also running the workflow on “bare silicon” instead of a virtual machine can potentially increase execution speed. The guidelines provided in the “install_kimble.txt” file in the virtual machine can help a knowledgeable user with installing the workflow environment on a PC with an Ubuntu-based operating system. However, it should be noted that this reduces reproducibility, because it ties the workflow to a specific computer setup.

The most unique and probably most valuable output generated by the workflow is the table of metabolite concentrations. The main weakness of untargeted metabolomics is the lack of a simple path from a significant bin to the annotated metabolite. This, in turn, complicates the interpretation of the data. The quantitative analysis of a limited/pre-selected subset of metabolites is becoming a main trend in the field. However, peak shifts caused by changes in pH and ionic strength lead to inter-individual and even intra-individual variations that make urine a worst-case scenario for metabolite quantification with NMR. In the workflow information from 1D and JRES spectra are combined to solve the variability problem and achieve optimal quantitation.

The authors are open to extending the KIMBLE platform with new functionality provided to us by the community. If it is decided to add a new node/tool to the platform, an acknowledgement or reference will be added to an annotation in the workflow itself. Future users of KIMBLE are kindly requested to include these references to their manuscripts if these external tools are in the workflow branches that lead to the nodes that generate the final results.

4. Conclusions

KNIME is a workflow manager and data mining platform. We have investigated its suitability for metabolomics data processing and analysis. The workflow presented in this article is a promising starting point for NMR metabolomics of urine and can be easily adapted to other biofluids. Both the data and the processing algorithms are stored in the workflow, making the data processing and analysis completely reproducible. Because of the “spill to disk” approach of KNIME, the workflow performance scales linearly with the number of spectra. Except for the KNIME software itself, the metabolomics workflow also depends on several Python and R libraries. By installing KNIME and the libraries in a Linux virtual machine, the components are combined in one entity (the virtual machine image) that can be easily archived and shared with others. With the large and still growing library of nodes that KNIME provides, users that have no programming experience can apply, adapt, and extend the workflow as well. The workflow branch that performs targeted metabolomics shows good agreement with the clinical chemistry results for creatinine quantification.

5. Materials and methods

5.1. Urine samples

442 urine samples were chosen from a Polycystic Kidney Disease (PKD) cohort for demonstrating the performance of the quantification algorithm. These include urine samples from PKD patients, patients suffering from other renal diseases, and healthy people. The urine samples used as an example in the multivariate analysis section were produced voluntarily by one of the authors of this manuscript. Urine samples were collected from the same person on two different days: an exercise day and a resting day. One sample was taken just after waking up and one sample was collected just before the first cycling session. The remaining 6 samples were taken after every cycling session during the recovery period. The distance of a single stretch was 15 km, resulting in a total cycling distance of 90 km. The cyclist aimed to maintain a constant speed of 28 km/h on straits. The length of time for one cycling session combined with the recovery period afterwards was 1.5 h. The person consumed a sports electrolyte drink but fasted otherwise. On a different day 8 urine samples were taken at similar timepoints as on the exercise day. In between sampling points the person was sitting at a desk, also fasting but avoiding physical exercise.

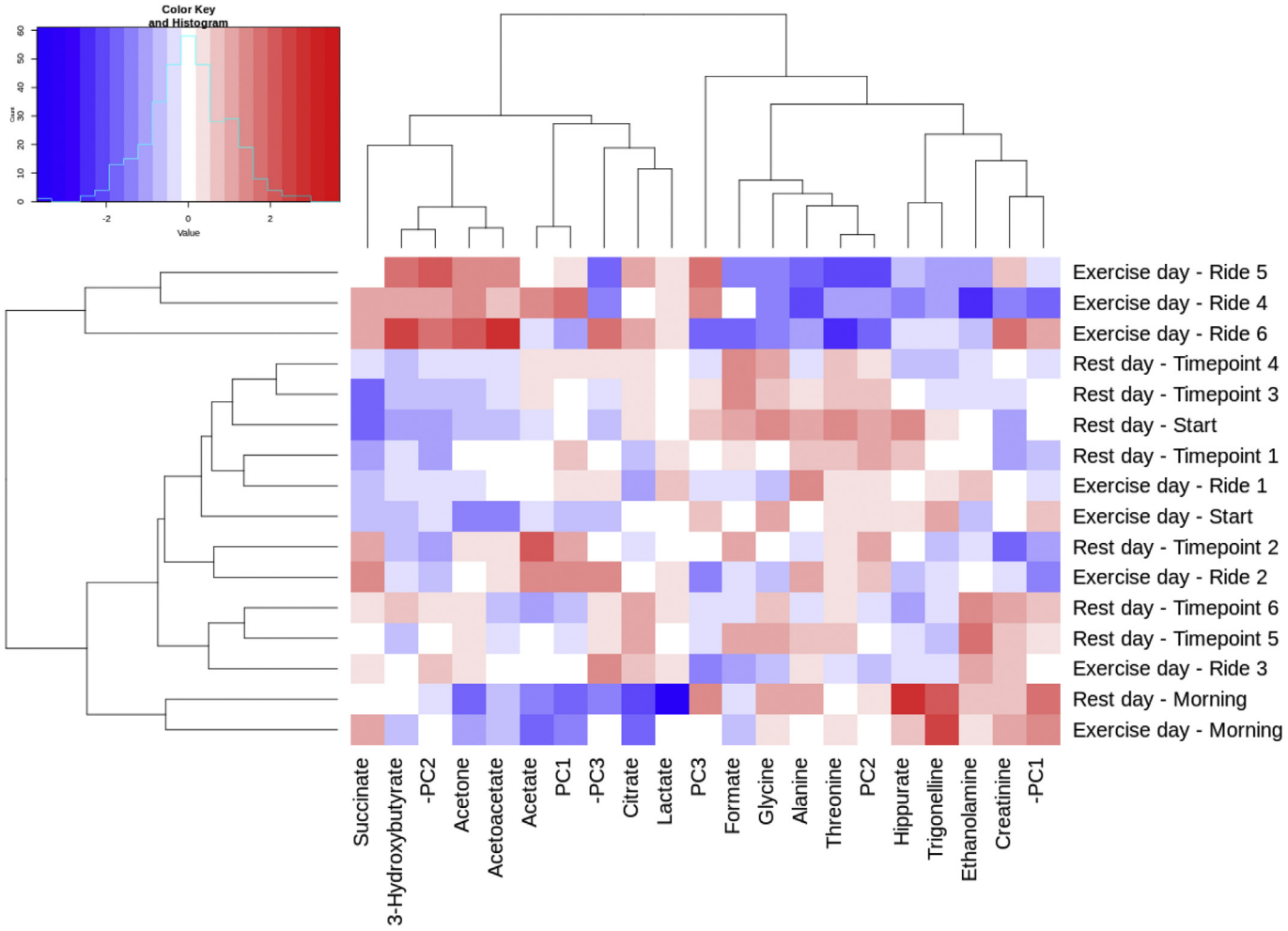


Fig. 10. Heatmap of the urine metabolome comparison between a person at rest and performing physical exercise. The metabolite concentrations are mean-centered and auto-scaled. PC1, PC2, etc. represent scores of the PCA analysis. Variables are clustered according the Pearson correlation (average linkage), while observation are clustered based on Euclidian distance (average linkage).

Table 1
The duration of the various sections of the workflow installed in a virtual machine with 10 GB RAM and 4 cores, running on a PC with 16 GB RAM and an Intel Core i7 with a 3.4 GHz clock frequency.

Number of samples	Time for import of NOESY1D and JRES spectra/s (time per sample/s)	Time for the quantification of 4 metabolites/s (time per sample/s)	Time for binning/s (time per sample/s)
50	78 (1.56)	136 (2.72)	73 (1.46)
100	143 (1.43)	238 (2.38)	107 (1.07)
200	293 (1.47)	437 (2.19)	173 (0.87)
400	568 (1.42)	836 (2.09)	310 (0.78)

5.2. Sample preparation

Prior to analysis, the urine samples were thawed and 700 µl of each sample was manually transferred to a 96 deep-well plate and centrifuged at 1550 g for 5 min. Using a Gilson 215 liquid handler, 630 µl of urine were mixed with 70 µl of pH 7.4 phosphate buffer (1.5 M) in 100% D_2O containing 4 mM TSP and 2 mM NaN_3. A customized Gilson 215 liquid handler was used to transfer the samples to a 5.0 mm Bruker NMR tube rack.

5.3. NMR experiments and processing

1H NMR data were collected using a Bruker 600 MHz AVANCE II spectrometer equipped with a 5 mm TCI cryogenic probe head and a z-gradient system. A Bruker SampleJet sample changer was used for sample insertion and removal. All experiments were recorded at 300 K. A fresh sample of 99.8% methanol-d4 was used for temperature calibration [26] before each batch of measurements. Duration of 90° pulses were automatically calibrated for each individual sample using a homonuclear-gated mutation experiment [27] on the locked and shimmed samples after automatic tuning and matching of the probe head. One-dimensional (1D) 1H NMR spectra were recorded using the first increment of a NOESY pulse sequence [20] with presaturation ($\gamma B_1 = 50$ Hz) during a relaxation delay of 4 s and a mixing time of 10 m s for efficient water suppression [28]. Initial shimming was performed using the TopShim

(Bruker Corporation, 2011) tool on a random mix of urine samples from the study, and subsequently the axial shims were optimized automatically before every measurement. 16 scans of 65,536 points covering 12,335 Hz were recorded. J-resolved spectra (JRES) were recorded with a relaxation delay of 2 s and 2 scans for each increment in the indirect dimension. A data matrix of 40 × 12,288 data points was collected covering a sweep width of 78 × 10,000 Hz.

5.4. Spectral processing in KIMBLE

The Free Indiction Decay (FID) of the NOESY1D experiment was imported into the KIMBLE workflow and subsequently zero-filled to 65,536 complex points prior to Fourier transformation. An exponential window function was applied with a line-broadening factor of 1.0 Hz. The spectra were automatically phase and baseline corrected and automatically referenced to the internal standard, with the methyl resonance of trimethylsilylpropionate (TSP) set to 0.0 ppm. The time-domain JRES data was also imported into the workflow. A sine-shaped window function was applied and the data was zero-filled to 256 × 16,384 complex data points prior to Fourier transformation. In order to remove the skew, the resulting data matrix was tilted along the rows by shifting each row (k) by 0.5*(128-k) points and symmetrised about the central horizontal lines.

5.5. Software

The workflow is running in KNIME Analytics Platform version 3.5.3, using the following extensions: Interactive R Statistics Integration, Javascript Views, JFreeChart, NGS Tools, Python Integration, Python Scripting Extension, Statistics Nodes, and Streaming Execution. The workflow depends on the R platform (version 3.4.4) and the R packages pls, AUC, psych, Rserve, XML, moments, readxl, Hmisc, gplots, plotrix, and ggplot2, pcaMethods, and sva, obtained from either the CRAN repository or the Bioconductor website. The workflow also uses Python (version 2.7.15) in combination with the following Python libraries: numpy, scipy, matplotlib, pandas, protobuf, sklearn, sklearn-pandas, and seaborn, obtained from either the Ubuntu repositories or the Python Package Index (pip). Python 3.6.5 is also available, but is not used in the template workflow at this point. All software listed above is installed and running in a 64-bit Xubuntu 18.04 virtual machine (VM). 10 GB of RAM, 120 GB of dynamically allocated storage (actual size 6.55 GB) and 4 CPU cores were assigned to the VM. 8 GB of heap space was assigned to KNIME. The virtual machine hypervisor is VirtualBox 5.2.12, supported by VirtualBox Guest Additions version 5.2.12 installed in the VM. The KNIME workspace folder and the folder for temporary files are located on the host system, and are accessed from within the VM via the VirtualBox shared folders feature. The hypervisor is running on a 64-bit Windows 7 Enterprise host system. It should be noted that the same VM can run on other operating system for which VirtualBox is available. The host system is a PC with 16 GB of RAM and an Intel Core i7-3770 CPU running at 3.40 GHz.

Availability

KIMBLE can be downloaded from: http://cpm.lumc.nl/kimble/.

Acknowledgements

The authors are grateful to Prof. Dr. Johan W. de Fijter for supplying a training sample selection for the quantification algorithm.

References

[1] A. Verhoeven, E. Slagboom, M. Wuhrer, M. Giera, O.A. Mayboroda, Automated quantification of metabolites in blood-derived samples by NMR, Anal. Chim. Acta 976 (2017) 52–62, https://doi.org/10.1016/j.aca.2017.04.013.
[2] S. Kostidis, R.D. Addie, H. Morreau, O.A. Mayboroda, M. Giera, Quantitative NMR analysis of intra- and extracellular metabolism of mammalian cells: a tutorial, Anal. Chim. Acta 980 (2017) 1–24, https://doi.org/10.1016/j.aca.2017.05.011.
[3] T. Huan, E.M. Forsberg, D. Rinehart, C.H. Johnson, J. Ivanisevic, H.P. Benton, M. Fang, A. Aisporna, B. Hilmers, F.L. Poole, M.P. Thorgersen, M.W.W. Adams, G. Krantz, M.W. Fields, P.D. Robbins, L.J. Niedernhofer, T. Ideker, E.L. Majumder, J.D. Wall, N.J.W. Rattray, R. Goodacre, L.L. Lairson, G. Siuzdak, Systems biology guided by XCMS Online metabolomics, Nat. Methods 14 (2017) 461–462, https://doi.org/10.1038/nmeth.4260.
[4] I. Kohler, A. Verhoeven, R.J. Derks, M. Giera, Analytical pitfalls and challenges in clinical metabolomics, Bioanalysis 8 (2016) 1509–1532, https://doi.org/10.4155/bio-2016-0090.
[5] TopSpin, Bruker BioSpin GmbH, Silberstreifen 4, 76287 Rheinstetten, Germany, n.d. https://www.bruker.com/products/mr/nmr/nmr-software/software/topspin/overview.html.
[6] J. He, C. Jin, B. Xia, Automics: an integrated platform for NMR-based metabonomics spectral processing and data analysis, BMC Bioinf. 10 (2009) 83, https://doi.org/10.1186/1471-2105-10-83.
[7] MATLAB, MathWorks, inc., 1 Apple hill drive Natick, MA 01760–2098, USA, n.d. https://www.mathworks.com/products/matlab.html.
[8] Chenomx NMR Suite, Chenomx inc., 4232–10230 Jasper Ave, Edmonton, Alberta, Canada, n.d. https://www.chenomx.com/.
[9] J. Hao, W. Astle, M. De Iorio, T.M. Ebbels, BATMAN—an R package for the automated quantification of metabolites from nuclear magnetic resonance spectra using a Bayesian model, Bioinformatics 28 (2012) 2088–2090, https://doi.org/10.1093/bioinformatics/bts308.
[10] SIMCA, Umetrics/Sartorius-Stedim, n.d. http://umetrics.com/products/simca.
[11] J. Xia, I.V. Sinelnikov, B. Han, D.S. Wishart, MetaboAnalyst 3.0—making metabolomics more meaningful, Nucleic Acids Res. 43 (2015) W251–W257, https://doi.org/10.1093/nar/gkv380.
[12] D. Jacob, C. Deborde, M. Lefebvre, M. Maucourt, A. Moing, NMRProcFlow: a graphical and interactive tool dedicated to 1D spectra processing for NMR-based metabolomics, Metabolomics 13 (2017), https://doi.org/10.1007/s11306-017-1178-y.
[13] M.A. Fitzpatrick, C.M. McGrath, S.P. Young, Pathomx: an interactive workflow-based tool for the analysis of metabolomic data, BMC Bioinf. 15 (2014) 396, https://doi.org/10.1186/s12859-014-0396-9.
[14] K. Wolstencroft, R. Haines, D. Fellows, A. Williams, D. Withers, S. Owen, S. Soiland-Reyes, I. Dunlop, A. Nenadic, P. Fisher, J. Bhagat, K. Belhajjame, F. Bacall, A. Hardisty, A. Nieva de la Hidalga, M.P. Balcazar Vargas, S. Sufi, C. Goble, The Taverna workflow suite: designing and executing workflows of Web Services on the desktop, web or in the cloud, Nucleic Acids Res. 41 (2013) W557–W561, https://doi.org/10.1093/nar/gkt328.
[15] J. Goecks, A. Nekrutenko, J. Taylor, Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences, Genome Biol. 11 (2010), https://doi.org/10.1186/gb-2010-11-8-r86. R86.
[16] Y. Guitton, M. Tremblay-Franco, G. Le Corguillé, J.-F. Martin, M. Pétéra, P. Roger-Mele, A. Delabrière, S. Goulitquer, M. Monsoor, C. Duperier, C. Canlet, R. Servien, P. Tardivel, C. Caron, F. Giacomoni, E.A. Thévenot, Create, run, share, publish, and reference your LC–MS, FIA–MS, GC–MS, and NMR data analysis workflows with the Workflow4Metabolomics 3.0 Galaxy online infrastructure for metabolomics, Int. J. Biochem. Cell Biol. 93 (2017) 89–101, https://doi.org/10.1016/j.biocel.2017.07.002.
[17] M.R. Berthold, N. Cebron, F. Dill, T.R. Gabriel, T. Kötter, T. Meinl, P. Ohl, C. Sieb, K. Thiel, B. Wiswedel, KNIME: the Konstanz information miner, in: Data Anal. Mach. Learn. Appl, Springer, 2008, pp. 319–326. http://www.springer.com/gp/book/9783540782391.
[18] H.L. Röst, T. Sachsenberg, S. Aiche, C. Bielow, H. Weisser, F. Aicheler, S. Andreotti, H.-C. Ehrlich, P. Gutenbrunner, E. Kenar, X. Liang, S. Nahnsen, L. Nilse, J. Pfeuffer, G. Rosenberger, M. Rurik, U. Schmitt, J. Veit, M. Walzer, D. Wojnar, W.E. Wolski, O. Schilling, J.S. Choudhary, L. Malmström, R. Aebersold, K. Reinert, O. Kohlbacher, OpenMS: a flexible open-source software platform for mass spectrometry data analysis, Nat. Methods 13 (2016) 741–748, https://doi.org/10.1038/nmeth.3959.
[19] S. Liggi, C. Hinz, Z. Hall, M.L. Santoru, S. Poddighe, J. Fjeldsted, L. Atzori, J.L. Griffin, KniMet: a pipeline for the processing of chromatography–mass spectrometry metabolomics data, Metabolomics 14 (2018), https://doi.org/10.1007/s11306-018-1349-5.
[20] A. Kumar, R.R. Ernst, K. Wüthrich, A two-dimensional nuclear Overhauser enhancement (2D NOE) experiment for the elucidation of complete proton-proton cross-relaxation networks in biological macromolecules, Biochem. Biophys. Res. Commun. 95 (1980) 1–6, https://doi.org/10.1016/0006-291X(80)90695-6.
[21] W.P. Aue, J. Karhan, R.R. Ernst, Homonuclear broad band decoupling and two-dimensional J-resolved NMR spectroscopy, J. Chem. Phys. 64 (1976) 4226–4227, https://doi.org/10.1063/1.431994.
[22] P.J.D. Foxall, J.A. Parkinson, I.H. Sadler, J.C. Lindon, J.K. Nicholson, Analysis of

biological fluids using 600 MHz proton NMR spectroscopy: application of homonuclear two-dimensional J-resolved spectroscopy to urine and blood plasma for spectral simplification and assignment, J. Pharmaceut. Biomed. Anal. 11 (1993) 21–31, https://doi.org/10.1016/0731-7085(93)80145-Q.

[23] J.J. Helmus, C.P. Jaroniec, Nmrglue: an open source Python package for the analysis of multidimensional NMR data, J. Biomol. NMR 55 (2013) 355–367, https://doi.org/10.1007/s10858-013-9718-x.

[24] T. De Meyer, D. Sinnaeve, B. Van Gasse, E. Tsiporkova, E.R. Rietzschel, M.L. De Buyzere, T.C. Gillebert, S. Bekaert, J.C. Martins, W. Van Criekinge, NMR-based characterization of metabolic alterations in hypertension using an adaptive, intelligent binning algorithm, Anal. Chem. 80 (2008) 3783–3790, https://doi.org/10.1021/ac7025964.

[25] F. Dieterle, A. Ross, G. Schlotterbeck, H. Senn, Probabilistic quotient normalization as Robust method to account for dilution of complex biological mixtures. Application in 1H NMR metabonomics, Anal. Chem. 78 (2006) 4281–4290, https://doi.org/10.1021/ac051632c.

[26] M. Findeisen, T. Brand, S. Berger, A 1H-NMR thermometer suitable for cryoprobes, Magn. Reson. Chem. 45 (2007) 175–178, https://doi.org/10.1002/mrc.1941.

[27] P.S.C. Wu, G. Otting, Rapid pulse length determination in high-resolution NMR, J. Magn. Reson. 176 (2005) 115–119, https://doi.org/10.1016/j.jmr.2005.05.018.

[28] W.S. Price, Water signal suppression in NMR spectroscopy, Annu. Rep. NMR Spectrosc. 38 (1999) 289–354, https://doi.org/10.1016/S0066-4103(08)60040-X.