OXFORD

Briefings in Bioinformatics, 2022, **23(2)**, 1–16
https://doi.org/10.1093/bib/bbab535
Problem Solving Protocol

TIGER: technical variation elimination for metabolomics data using ensemble learning architecture

Siyu Han, Jialing Huang, Francesco Foppiano, Cornelia Prehn, Jerzy Adamski, Karsten Suhre, Ying Li, Giuseppe Matullo, Freimut Schliess, Christian Gieger, Annette Peters and Rui Wang-Sattler

Corresponding author: Rui Wang-Sattler, Tel: +49 89 31873978; E-mail: rui.wang-sattler@helmholtz-muenchen.de

Abstract

Large metabolomics datasets inevitably contain unwanted technical variations which can obscure meaningful biological signals and affect how this information is applied to personalized healthcare. Many methods have been developed to handle unwanted variations. However, the underlying assumptions of many existing methods only hold for a few specific scenarios. Some tools remove technical variations with models trained on quality control (QC) samples which may not generalize well on subject samples. Additionally, almost none of the existing methods supports datasets with multiple types of QC samples, which greatly limits their performance and flexibility. To address these issues, a non-parametric method TIGER (Technical variation elImination with ensemble learninG architEctuRe) is developed in this study and released as an R package (https://CRAN.R-project.org/package=TIGERr). TIGER integrates the random forest algorithm into an adaptable ensemble learning architecture. Evaluation results show that TIGER outperforms four popular methods with respect to robustness and reliability on three human cohort datasets constructed with targeted or untargeted metabolomics data. Additionally, a case study aiming to identify age-associated metabolites is performed to illustrate how TIGER can be used for cross-kit adjustment in a longitudinal analysis with experimental data of three time-points generated by different analytical kits. A dynamic website is developed to help evaluate the performance of TIGER and examine the patterns revealed in our longitudinal analysis (https://han-siyu.github.io/TIGER_web/). Overall, TIGER is expected to be a powerful tool for metabolomics data analysis.

Keywords: metabolomics, machine learning, ensemble learning, predictive modelling, longitudinal analysis

Introduction

Metabolomics provides a unique perspective to quantitatively characterize small molecule (<1500 Dalton), metabolites, which can represent the metabolic status of a subject. Metabolomics analyses facilitate the identification of biomarkers and improve the understanding of

Siyu Han is a doctoral candidate at School of Medicine, Technical University of Munich, Germany, and Helmholtz Zentrum München, German Research Center for Environmental Health (GmbH). His research interests include machine learning and its applications in life science.
Jialing Huang is a doctoral candidate at Ludwig Maximilian University of Munich, Germany, and Helmholtz Zentrum München, German Research Center for Environmental Health (GmbH). Her research interests include integration of OMICs data to study metabolic diseases.
Francesco Foppiano is a master student at the Ludwig Maximilian University of Munich, Germany, and Helmholtz Zentrum München, German Research Center for Environmental Health (GmbH). His research interests include molecular epidemiology and population genetics.
Cornelia Prehn is the Head of Metabolomics Lab at Metabolomics and Proteomics Core Facility, Helmholtz Zentrum München, German Research Center for Environmental Health (GmbH). Her research interest is to precisely and efficiently quantify metabolite concentration profiles.
Jerzy Adamski is a professor at the National University of Singapore, University of Ljubljana, Slovenia and Technical University of Munich, Germany. His research interest is to identify the factors responsible for the pathogenesis of complex metabolic diseases such as diabetes, obesity, cardiovascular disorders and to characterize biomarkers supporting diagnosis and therapy developments.
Karsten Suhre is a professor at Weill Cornell Medicine and director of the Bioinformatics Core, Qatar. His research interests are at the intersection of metabolomics and genomics and their role in human health.
Ying Li is an associate professor at the College of Computer Science and Technology, Key Laboratory of Symbol Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun, China. Her research topics include machine learning, bioinformatics and computational biology.
Giuseppe Matullo is a professor in Human Genetics and group leader of the Genomics Variation, Complex Diseases and Population Medicine Unit at the Turin University, Italy, focusing on research activity on gene–environment interaction, quantitative trait loci and biomarker identification from several omics analyses in complex diseases.
Freimut Schliess is the Director Science & Innovation at Profil Institut für Stoffwechselforschung (GmbH) and focuses on the translation of metabolomics data into a stratification of persons with diabetes in the setting of both clinical diabetes research and diabetes care.
Christian Gieger is the Head of the Research Unit of Molecular Epidemiology at the Helmholtz Zentrum München, German Research Center for Environmental Health (GmbH). His research focuses on deciphering the molecular mechanisms of complex diseases like type 2 diabetes or obesity.
Annette Peters is a professor at the Ludwig Maximilian University of Munich, Germany, and is the director of the Institute of Epidemiology at the Helmholtz Zentrum München, German Research Center for Environmental Health (GmbH). Her focus is on the development and progression of metabolic, respiratory and allergic diseases, as well as heart diseases and mental health.
Rui Wang-Sattler is a group leader at the Helmholtz Zentrum München, German Research Center for Environmental Health (GmbH). Her research focuses on identification of candidate biomarkers and better understand the pathophysiological mechanisms of metabolic diseases.
Received: September 8, 2021. **Revised:** November 1, 2021. **Accepted:** November 18, 2021
© The Author(s) 2022. Published by Oxford University Press.
This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/4.0/), which permits non-commercial re-use, distribution, and reproduction in any medium, provided the original work is properly cited. For commercial re-use, please contact journals.permissions@oup.com

biological pathways in health and metabolic disease such as diabetes [1]. Despite significant advances in recent years, unwanted technical variation still remains a critical issue in the current metabolomics workflow [2] and prevents the translation of metabolomics analyses in personalized healthcare. Liquid chromatography-mass spectrometry (LC-MS) is the most widely used technique for metabolomics studies because of its high sensitivity and metabolite coverage [3–5]. But the unwanted technical variation can be introduced by changes in experimental conditions such as batch effects, temporal drifts, temperature changes and variation of analytical platforms. In metabolomics experiments, biological variation of interest is inevitably confounded with systematic errors which can be categorized as intra- and inter-batch technical variation [6]. Intra-batch variation generally refers to the incremental changes in instrumental response during the measurement of a batch of samples [7]. And inter-batch variation occurs when samples in a large-scale study have to be separated into batches [8]. Intra- and inter-batch variations can cause detectable differences between samples irrespective of biological variation and further lead to false discoveries [9, 10].

Technical noises and systematic errors in metabolomics data are hard, even impossible to control [10, 11]. Therefore, many practical methods and tools (Table 1) have been proposed to normalize data and remove unwanted variation, so that samples from different batches can be combined and compared [12]. One of the most common approaches is to use normalization factor (NF). For each metabolite of each batch, the raw metabolite values of the QC samples in this batch act as test values, while the averaged metabolite values of the QC samples from all batches are used as target value. The NF is the median [13] or mean [14] of the ratios of the test values relative to the target value. Each metabolite of each batch has its own NF which measures the deviation caused by the potential variations. For subject samples, the deviation can be finally offset by dividing the values of subject samples by the corresponding NF. In this case, the method can be viewed as fitting a linear equation where the calculated NF acts as the slope term. In addition to linear regression, local polynomial regression (LOESS) is also being applied to eliminate technical variation [15, 16]. LOESS has an advantage over linear equation with respect to model flexibility. Compared to a straight line determined by NF, LOESS fits a polynomial curve by assigning more weights to points near the one whose response is being estimated. An LOESS model for technical variation removal is usually trained with the information of injection order, the idea of which is that the training samples temporally close to a given test sample may better characterize the temporal drifts it suffers than those that are far away. Recently, a machine learning-based method, SERRF (Systematic Error Removal using Random Forest) [17], has been developed to normalize large-scale untargeted metabolomics data. The study of SERRF demonstrates that technical variation in the intensity of one compound can be modelled by the intensities of other compounds. SERRF builds random forest (RF) [18] model to regress unwanted systematic variation and has surpassed many popular methods, including NOMIS (Normalization using Optimal selection of Multiple Internal Standards) [19], cubic splines normalization [20] and a method based on support vector machine (SVM) [21], on several benchmark datasets. Another state-of-the-art tool WaveICA [22] was developed for the cases where QC samples are not available. WaveICA utilizes the wavelet transform [23] and independent component analysis (ICA) [24, 25] to capture and remove technical variation. Based on the assumption that metabolite intensities may display temporal trends over the injection order, WaveICA first uses the wavelet transform to decompose the trend into multi-scale data with different frequencies. Then ICA is used to detect and remove unwanted variation in the multi-scale data. And the normalized data are finally reconstructed using the inverse wavelet transform. The experiments show that WaveICA outperforms ComBat (combatting batch effects when Combining Batches) [26], a well-known QC-free method based on empirical Bayes, as well as QC-RLSC (Quality Control-based Robust LOESS Signal Correction) [27].

Many valuable and helpful methods have been developed, but the current approaches are subject to several limitations. One underlying assumption of NF is that the increase in one metabolite value can be balanced by the decrease in the values of another metabolites, but it has been argued that this assumption is only valid in limited practical scenarios [19, 28]. LOESS's normalization capability is boosted by its more advanced algorithm, but the model of LOESS neglects the potential associations between metabolites and tends to overfit the data. Regarding data structure, LOESS can hardly be applied to datasets that contain more variables (metabolites) than samples [17]. SERRF is one of the most robust methods, but its model trained on QC samples cannot guarantee to yield satisfactory results on subject samples—this is also the issue of most QC-based methods. WaveICA is not afflicted by the common issues of QC-based methods, but it still has its own drawbacks. WaveICA assumes that biological variation mainly exists in the data with high frequency, whereas the temporal drifts are in the low-frequency part. During the normalization, the biological variation in the low-frequency area might be removed as well.

Recently, the routine use of common, well-characterized QC samples has been recognized as a valuable tool to improve the external validity of large-scale metabolomics studies. [12, 29, 30]. QC samples cannot only help eliminate temporal drifts, batch effects and other technical variations, but also help achieve better inter-laboratory reproducibility so that data from different centres can be effectively compared [5]. With the growth of QC-provided metabolomics datasets, the development of highly flexible and readily adaptable

Table 1. Overview of four popular normalization methods and TIGER

Method	Algorithm	Metabolite type	QC-based	Injection order required	Multiple QC supported	Reference
NF	Linear regression	Targeted and untargeted	Yes	No	No	[13, 14]
LOESS	Local polynomial regression	Targeted and untargeted	Yes	Yes	No	[15, 16]
SERRF	Random forest	Untargeted	Yes	Yes	No	[17]
WaveICA	Wavelet transform	Untargeted	No	Yes	No	[22]
TIGER	Ensemble learning	Targeted and untargeted	Yes	No	Yes	This study

QC-based methods become a necessity. Now more and more QC-based methods are employing machine learning algorithms to normalize datasets and remove technical variation. Metabolite values of QC samples and subject samples are typically used as training and test data. A traditional machine learning model is generally trained and fine-tuned on a training set that is highly representative of unseen examples. However, QC samples may not be fully representative of subject samples. Due to underfitting and overfitting issues, many QC-based methods only yield weak performance on subject samples [22, 31]. As to performance assessment, the same QC samples are often divided into two subsets for model training and evaluation, which may make the evaluation result too optimistic—strong evaluation performance on QC samples does not guarantee a satisfactory result on subject samples. To improve generalization and achieve reliable evaluation, datasets of many cohorts or studies contain multiple types of QC samples, sometimes corresponding to different metabolite concentration levels. The availability of different QC samples provides more flexibility and better reproducibility for metabolomics data pre-processing. However, almost none of the existing methods is able to process the datasets with different kinds of QC samples.

To address these issues, a novel method TIGER (Technical variation elImination with ensemble learninG architEctuRe) is developed based upon an adaptable ensemble learning architecture. TIGER is evaluated with three datasets constructed with targeted or untargeted LC-MS metabolomics data. Moreover, a case study is performed to illustrate how TIGER can be applied to longitudinal analysis.

TIGER is released as an R package which can be installed in R via command `install.packages("TIGERr")`. The package manual provides detailed descriptions and examples to help users apply TIGER to different scenarios (see Supplementary File—R Package Manual). A dynamic website (https://han-siyu.github.io/TIGER_web/) is also developed, which enables users to interactively compare TIGER with other popular methods and review the patterns revealed in our longitudinal analysis.

Methods

TIGER eliminates the technical variation using an adaptable ensemble learning architecture. This section describes the framework of the architecture and illustrates how this architecture is adapted to build TIGER.

Architecture of Ensemble Learning

The ensemble learning architecture designed in this study is inspired by the idea of super learner [32] and tailored for the cases where training samples are limited and cannot fully represent test samples. The architecture is comprised of several base models and one meta model. The base models are trained with different hyperparameter sets, which could mitigate the potential overfitting issue caused by one specific hyperparameter set. The meta model assigns weights to base models such that high-performing base models can obtain great weights, but information of underperforming learners can be considered as well.

Base Model

The ensemble model has n base models. Each base model is determined by θ_i, a hyperparameter set from pool $\{\theta_1, \theta_2, ..., \theta_n\}$. The base model $\phi_i(\cdot)$ is a machine learning model of the form:

$$\phi_i(y \sim X) = \varphi(y \sim X|\theta_i). \quad (1)$$

where X and y represent variable and response; $\varphi(\cdot)$ denotes a function that can be generalized to various approximators, such as SVM, k-nearest neighbours (k-NN) or other user-defined functions.

When fitting base models, a specific training set $\mathcal{D} = \{y \sim X|m \in 1:M, d \in 1:D\}$ including M instances and D variables will be shuffled randomly and split into K folds for cross-validation (CV). For any fold $k \in \{1, 2, ..., K\}$, base models are trained with $K-1$ training folds $\{y_{-k} \sim X_{-k}\}$ and tested on the remaining validation fold X_k:

$$\hat{y}_{i,k} = \phi_i(X_k|y_{-k} \sim X_{-k}), \quad (2)$$

where $\hat{y}_{i,k}$ is the predicted result of the validation fold $\{X_k\}$ produced by $\phi_i(\cdot|y_{-k} \sim X_{-k})$, the base model ϕ_i trained with the training folds $\{y_{-k} \sim X_{-k}\}$. The base models here are only used to evaluate their performances. All base models will be retrained with the whole training set and used to normalize new data. Finally, values of $\hat{y}_{i,k}$ are collected and concatenated:

$$\hat{y}_i = \{\hat{y}_{i,1}, \hat{y}_{i,2}, ..., \hat{y}_{i,k}\}. \quad (3)$$

Algorithm 1. Overview of TIGER Algorithm

Input: $Q_{D,B}$: QC samples with D metabolites from B batches
Input: $S_{D,B}$: subject samples to be normalised
Output: $S'_{D,B}$: normalised data of subject samples
▷ start iteration of normalisation process
for $d \in D$ **do**
  ▷ d: index of objective metabolite
  ▷ step 1: variable selection
  $v_Q = \text{highlyCorrelatedMetabolites}(Q_{d,B})$
  $v_S = \text{highlyCorrelatedMetabolites}(S_{d,B})$
  $v = v_Q \cap v_S$ ▷ intersection of v_Q and v_S
  ▷ v: selected variables for metabolite d
  for $b \in B$ **do**
    ▷ b: index of batch
    ▷ step 2: model construction
    training set $\mathcal{D}_{d,b} = \{Q_{d,b}, Q_{v,b}\}$
    ▷ $Q_{d,b}$: data of objective metabolite
    ▷ $Q_{v,b}$: data of selected metabolites
    build ensemble model $\Phi_{d,b}(\cdot)$ with $\mathcal{D}_{d,b}$ ▷ to Alg. 2
    ▷ step 3: data correction
    apply $\Phi_{d,b}(\cdot)$ to $S_{d,b}$
  concatenate results
output normalised dataset $S'_{D,B}$

Meta Model

Meta model $\Phi(\cdot)$ corresponds to a model of the form:

$$\Phi(y \sim X) = \sum_i^n w_i \phi_i(y \sim X). \tag{4}$$

Unlike many blending models that average the predictions of all ensemble members, the architecture here assigns each base model a weight w_i which is transformed from the loss of each validation set via a softmax-like formula:

$$w_i = \frac{\exp(-\ell_i)}{\sum_i^n \exp(-\ell_i)}, \tag{5}$$

where ℓ_i, the loss of the corresponding base model $\phi_i(\cdot)$, is defined as follows:

$$\ell_i = \mathcal{L}(\hat{y}_i, y), \tag{6}$$

$$\mathcal{L}(\hat{y}, y) = \frac{1}{M} \sum_{m=1}^{M} \frac{|\hat{y}^{(m)} - y^{(m)}|}{y^{(m)}}. \tag{7}$$

Here, $\mathcal{L}(\cdot)$ is the loss function that measures error ratio; $\hat{y}_i^{(m)}$ denotes the m-th values predicted by $\phi_i(\cdot)$ in CV, and $y^{(m)}$ is the corresponding target value. The numerator of Eq. 5 calculates the exponential of the negative loss of each individual base model, and the denominator sums over the exponentials. This ensures base models with small losses have more weights, but learners having large losses can also get some weights, which alleviates the overfitting problem caused by one specific hyperparameter set. And the resulting ensemble model might outperform the constituent models.

TIGER Algorithm

TIGER iterates many times during the correction (Alg. 1). Only one metabolite will be processed in one iteration. In each iteration, TIGER constructs ensemble models separately for different batches. The model construction can be outlined in three steps: variable selection, model construction and data correction.

Algorithm 2. Procedures of training ensemble model in TIGER

Input: $\mathcal{D} = \{y, X\}$: training set with response y and variables X
Input: $\{\theta_1, \theta_2, ..., \theta_n\}$: hyperparameter sets for random forest
Input: K: number of folds for cross-validation
Output: Normalised metabolomics data
▷ start to build base models
compute error ratio y' by Eq. 9
$\mathcal{D}_{RF} = \{y', X\}$
shuffle $\mathcal{D}_{RF}$ and split the dataset into K folds
for $i \in \{1, 2, ..., n\}$ **do**
  for $k \in \{1, 2, ..., K\}$ **do**
    train $\psi(y'_{-k} \sim X_{-k}|\theta_i)$ using the training folds
    predict $\hat{y}'_{i,k} = \psi(X_k|y'_{-k} \sim X_{-k})$ ▷ error ratio
    transform $\hat{y}_{i,k} = y_k/(\hat{y}'_{i,k} + 1)$ ▷ concentration
  get $\hat{y}_i$ by Eq. 3
  train $\psi(y' \sim X|\theta_i)$ using the whole training set
  get base model $\phi_i(y \sim X)$ by Eq. 1 and Eq. 8
▷ start to fit meta model
for $i \in \{1, 2, ..., n\}$ **do**
  compute loss $\ell'_i = \mathcal{L}(\hat{y}_i, \bar{y})$ by Eq. 7
for $i \in \{1, 2, ..., n\}$ **do**
  compute weight w_i by Eq. 5
fit meta model $\Phi(y \sim X) = \sum_i^n w_i \phi_i(y \sim X)$

Variable Selection

The correlation coefficients (CCs) of any two metabolites are calculated separately using training (i.e. QC samples in a general case) and test samples (i.e. subject samples). For an objective metabolite to be normalized, the metabolites with CCs greater than 0.5 in both training and test samples are selected. Then the intersection of the selected metabolites determines t highly-correlated metabolites. To ensure stable and consistent performances of the models in different iterations, the number of highly-correlated metabolites is limited to a specific range $[t_{min}, t_{max}]$. If $t < t_{min}$, t_{min} metabolites with top CCs in both training and test samples will be selected. If $t > t_{max}$, only top t_{max} of t metabolites will be selected. By default, TIGER selects 5 to 10 variables to train models.

Both Pearson product–moment correlation and Spearman's rank correlation are supported in TIGER to compute CCs. Partial correlation [33] is also supported for variable selection, such that pairwise CCs are conditioned against the correlation with all other metabolites, and indirect associations between distantly related metabolites are ignored [34]. In this study, TIGER uses rank-based CCs.

The training set constructed with the data of highly correlated metabolites not only reduces the model complexity but also considers the potential interactions

between metabolite values. Moreover, the selected variables also imply the variation introduced by temporal drifts, well position effects and other unobserved noises. Thus, the model can capture the relevant information even if the data of injection order and well position are not available. In practice, users can also explicitly include injection order and well position into the training data to train TIGER (see Supplementary File—R Package Manual).

Model Construction

The architecture described in the previous section is a general framework which can be extended and adapted to various use cases. In the implementation of TIGER, the ensemble learning architecture is further tailored to suit the case of technical variation removal. Alg. 2 shows how the model is constructed in TIGER using our ensemble learning architecture.

Specifically, the base model $\varphi(\cdot)$ in TIGER is defined as:

$$\varphi(y \sim X|\theta_i) = \frac{y}{\psi(y' \sim X|\theta_i) + 1}, \tag{8}$$

where $\psi(\cdot)$ is an RF model trained with θ_i which is one hyperparameter combination from the hyperparameter pool. By default, the pool contains {*mtry_percent* = {0.2, 0.4, 0.6, 0.8}, *nodesize_percent* = {0.2, 0.4, 0.6, 0.8}}. Users can include more hyperparameters into the pool (see Supplementary File—R Package Manual for further details). Response y in Eq. 8 denotes the raw values of an objective metabolite, while variable X is the raw data of y's highly correlated metabolites. The RF model $\psi(\cdot)$ is to predict the error ratio of y, denoted by y' and defined as:

$$y' = \frac{y - \bar{y}}{\bar{y}}, \tag{9}$$

where $\bar{y}$ is the mean or median of y. $\bar{y}$ can be calculated based on the whole dataset or each batch. By default, TIGER computes mean-based $\bar{y}$ from the whole dataset ($\bar{y} = \bar{y}_{all}$). But batch-specific $\bar{y} = \bar{y}_{batch}$ is also supported in TIGER. In this case, the obtained metabolite value will be additionally multiplied by a factor, $\bar{y}_{all}/\bar{y}_{batch}$, to offset inter-batch variations. This will make the algorithm more aggressive and competent to process the datasets with strong batch effects. When fitting the meta model, $\bar{y}$ is used as the target value to compute the loss in Eq. 6. If a training set has more than one kind of QC sample, then $\bar{y}$ and y' will be calculated separately for each kind.

Overall, for a specific training set $\mathcal{D} = \{y \sim X\}$, TIGER first transforms raw metabolite values of an objective metabolite y into the corresponding error ratio y' (Eq. 9). Then y' and X are fed into an RF model $\psi(\cdot)$. Accordingly, the predicted result of $\psi(\cdot)$ is also an error ratio denoted by $\hat{y}'$. Before being passed to the meta model, the predicted error ratio is converted back to metabolite values by equation $\hat{y}_i = y/(\hat{y}' + 1)$. The output of TIGER's base models $\hat{y}_i$ and $\bar{y}$ are passed to meta model, and the loss is obtained by computing $\ell_i' = \mathcal{L}(\hat{y}_i, \bar{y})$. Thus, the weight assigned to each base model is based on how close the predicted metabolite value $\hat{y}_i$ is to the target value $\bar{y}$.

Data Correction

Given the cross-validated weights, base models can be combined together for data correction. The outcomes are the weighted sums of the predicted results of all base models, which are also the corrected values of the objective metabolite converted from predicted error ratios.

In the implementation of TIGER, test samples are scheduled to be processed on-the-fly during the model construction, rather than being corrected in a separate stage, to avoid redundant computational costs. Parallel computing is supported to accelerate computational speed.

Evaluation Criteria

We use relative standard deviation (RSD, also known as the coefficient of variation), mean absolute percentage error (MAPE) and principal component analysis (PCA) to evaluate TIGER.

As one of the most widely used metrics, RSD is a unitless and standardized measure defined as the ratio of the standard deviation to the arithmetic mean.

MAPE is one of the most common measures in computational fields, used to evaluate the difference ratio between true values and predicted values (Eq. 7). A lower MAPE value indicates the predicted value is closer to the target value which in this study is defined as the average of the corresponding metabolite values.

Considering that some datasets only provide one kind of QC, while training and testing using the same kind of QC may lead to an over-optimistic evaluation, we further use PCA plots to compare the clusters of samples before and after applying different normalization methods. Identical samples in the PCA plot should be clustered together.

Data Description

Data from KORA (Cooperative Health Research in the Region of Augsburg) study [35], P20 Negative (negative mode, Functional Cardio-Metabolomics study) [17] and Amide of WaveICA [22] are used to evaluate normalization methods. Data in the KORA study are the concentrations of targeted metabolites, while data in P20 Negative and Amide measure the compounds or peak intensities of untargeted metabolomics data (Table 2).

Targeted Metabolomics Datasets

We use targeted metabolomics data of three timepoints from the KORA cohort: the baseline survey (KORA S4, examined between 1999 and 2001), the first follow-up study (KORA F4, 2006–2008) and the

Table 2. Summary of the datasets used in this study

Dataset	Targeted				Untargeted	
	KORA S4	KORA F4 (Original)	KORA F4 (Remeasured)	KORA FF4	P20 Negative	Amide
Number of Batches/Plates[1]	22	38	4	29	4	4
Number of Subjects	1614	3061	288	2218	1174	644
Number of Variables[2]	103	103	103	103	268	6402
Number of QC Samples						
QC	114	–	22	145	125	85
QC1	22	38	4	29	–	–
QC2	22	38	4	29	–	–
QC3	22	38	4	29	–	–
Median of RSD						
QC	0.0921	–	0.0884	0.1178	0.2752	0.5139
QC1	0.0946	0.1225	0.0770	0.1090	–	–
QC2	0.0799	0.1251	0.0678	0.1065	–	–
QC3	0.0797	0.1070	0.0724	0.1093	–	–
Median of MAPE						
QC	0.0741	–	0.0697	0.0947	0.2256	0.4171
QC1	0.0726	0.0993	0.0537	0.0870	–	–
QC2	0.0603	0.0977	0.0475	0.0818	–	–
QC3	0.0630	0.0878	0.0546	0.0884	–	–

[1]Plates for KORA S4, F4, FF4. Batches for P20 Negative and Amide. [2]KORA data include 103 metabolites. P20 Negative contains 268 lipids. Amide has 6402 peaks.

second follow-up (KORA FF4, 2013–2014), as well as the accompanying QC samples to construct targeted metabolomics datasets. The metabolite profiling of KORA S4 (March–April 2011), F4 (August 2008–March 2009) and FF4 (February–October 2019) serum samples spans more than a decade during which analytical procedures have been upgraded several times. The samples of KORA F4 were measured with the analytical kit Absolute*IDQ*® p150 (p150, BIOCRATES Life Sciences AG, Innsbruck, Austria), while the samples of KORA S4 and FF4 were quantified with Absolute*IDQ*® p180 (p180). To evaluate the technical variation introduced by different kits, samples of 288 individuals from the F4 study were remeasured using the p180 kit (September–October 2019). During the measurement, three different manufacturer-provided QC samples, denoted by QC1, QC2 and QC3, were allocated to each 96-well kit plate. For the p180 kit, each plate additionally quantified five identical pooled EDTA-plasma QC samples (Sera Laboratories International Ltd., Hull, United Kingdom) [36], denoted by QC. Manufacturer-provided QC1, QC2 and QC3 varied due to the platform update, but pooled EDTA-plasma QC remained the same. We use KORA F4 (Original) and F4 (Remeasured) to distinguish the two subsets of the KORA F4 dataset (Table 2).

Quality inspection [37, 38] is applied to the metabolomics data. Kits p150 and p180 allow simultaneous quantification of 163 and 188 metabolites, respectively. Only metabolites that meet the following five criteria will be selected: (1) the overlap between p150 and p180; (2) at least 50% of its measured sample values are above the limits of detection (LOD) of its corresponding plates; (3) missing values <10% (4) median RSD of different QC samples <25% (5) the spearman CCs between KORA F4 (Remeasured) and F4 (Original) >0.5. In total, 103 metabolites satisfy all criteria and are selected to construct the target metabolomics datasets (see Table S1 for detailed quality inspection).

The latest surveyed dataset KORA FF4 is used for method evaluation where QC are selected as training samples, while QC1, QC2, QC3 and subject samples are used as test data (see Figure 1 for an example variable from KORA FF4). In case study section, datasets of three time-points (KORA S4, F4 and FF4) are used for longitudinal analysis.

Untargeted Metabolomics Datasets

P20 Negative and Amide are two ready-to-use datasets provided by SERRF and WaveICA. Relevant quality inspection has been conducted in their original studies. P20 Negative dataset is based on plasma samples and acquired using a validated lipidomics assay [17]. In P20 Negative, 1174 subject samples and 125 identical QC samples were separated into four batches (Figure 1). Each sample has 268 lipids (Table 2).

Dataset Amide is based on plasma samples processed with a UHPLC-QTOF/MS system [22]. A total of 644 subject samples and 85 identical QC samples were collected from four batches, and each sample has 6402 detected peaks (Figure 1 and Table 2).

Both P20 Negative and Amide only provide one kind of QC sample, 80% of which will be used as training sets, while the remaining QC samples and all subject samples are used as test sets.

Evaluation Results

We first evaluate how our ensemble learning architecture performs on the task of metabolomics data normalization. Second, TIGER is benchmarked against four popular methods, NF, LOESS, SERRF and WaveICA, using KORA FF4, P20 Negative and Amide.

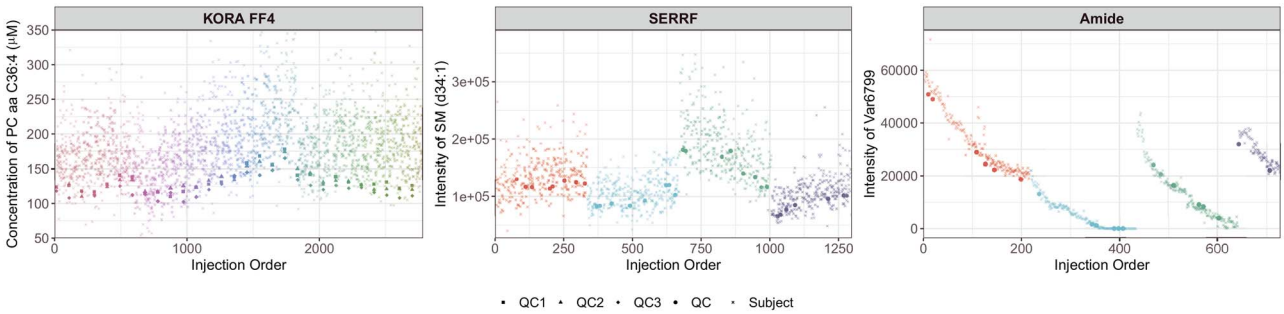

Figure 1. Raw data of selected metabolites from datasets KORA FF4, P20 Negative and Amide. The x-axis indicates the injection order, and the y-axis represents metabolite concentration (μM) or intensity. Samples from different batches are distinguished by colours.

Evaluation of Ensemble Learning Architecture

To validate the ensemble learning architecture, the ensemble models whose base models are trained with three machine learning algorithms of different complexities, namely k-NN [39], random forest [40] and extreme gradient boosting (XGB) [41, 42], are compared with the corresponding models without using ensemble technique. The non-ensemble models are trained with the same hyperparameter sets (Table S2) as the ensemble models and tuned with a 5-fold cross-validated grid search. All models are trained with QC and tested with QC1, QC2 and QC3 and subject samples from the KORA FF4 dataset.

Figure 2 demonstrates the improvement of our ensemble learning architecture to the models trained with original algorithms. As shown in Figure 2, all ensemble models show some improvements upon their predecessors, which demonstrates the effectiveness of our ensemble learning architecture. k-NN and XGB give the best and the worst results among non-ensemble models. After adopting the ensemble learning strategy, these two algorithms yielded the least and the greatest improvement. Model complexity may account for this fact (see discussion). Algorithm RF is selected to build the base models in TIGER as it helps to achieve the best performance on both RSD and MAPE with moderate complexity. After incorporating the RF algorithm into the ensemble learning architecture, the medians of RSD and MAPE are„ 14.43% and 14.79% better than the original model.

Evaluation on Targeted Metabolomics Data

From the evaluation of KORA FF4's 103 metabolites, we found that the performances of TIGER, NF, LOESS, SERRF and WaveICA on the three manufacturer-provided QC samples are very similar (Table S3). Compared to the other four methods, TIGER has the lowest median of RSD. For example, for QC2 from KORA FF4, the median of RSD is reduced from 0.1065 (raw) to 0.0509 after being processed by TIGER (Figure 3). All four methods are able to lower the RSD, which means the normalized values are less dispersed than raw values. In addition to RSD, low medians of MAPE of TIGER, NF and SERRF are observed (Table S3), suggesting good performances of these three methods. However, the metabolite values seem to further deviate from the target values after being processed by LOESS and WaveICA—the median of MAPE (QC2) increases from 0.0818 (raw) to 0.5444 and 0.2537, respectively (Figure 3). TIGER achieves the best performance among all investigated methods and strikes a superior balance between RSD and MAPE.

The detailed performances of TIGER and the other four methods on each metabolite are available at our dynamic website.

Evaluation on Untargeted Metabolomics Data

Untargeted datasets P20 Negative and Amide are further used to evaluate the performances of different methods. As dataset Amide contains stronger technical variation (see Table 2) than KORA FF4 and P20 Negative, the target value (in Eq. 7), an argument of TIGER's programme, is configured to compute based on each batch.

Figure 3 shows that TIGER effectively lowered the technical errors and improved the data quality of the P20 Negative dataset—the median of RSD is reduced by 19.06% (from 0.2741 of raw data to 0.0835), while the median of MAPE is reduced by 14.67% (from 0.2117 of raw data to 0.0650).

The evaluation results of the Amide dataset show that all four methods can lower the RSD values of QC samples (Figure 3 and Table S3), while WaveICA has the lowest median of RSD. In terms of the error ratio, the evaluation reveals similar good performance among TIGER, LOESS, NF and SERRF, while WaveICA gets the highest median of MAPE.

PCA analysis is further performed to evaluate how each method generalizes to subject samples. Figure 4 shows four clusters representing four batches of raw data of P20 Negative and Amide, respectively. The QC and subject samples in P20 Negative and Amide are clustered together without distinct batch differences after being normalized by TIGER, SERRF and WaveICA, which demonstrates that these methods effectively eliminate the technical variation. By contrast, after normalized by NF, QC and subject samples still contain strong batch effects, which means NF underfits the data. From the RSD and MAPE results (Figure 3), it seems that LOESS achieves the best performance on the Amide dataset.

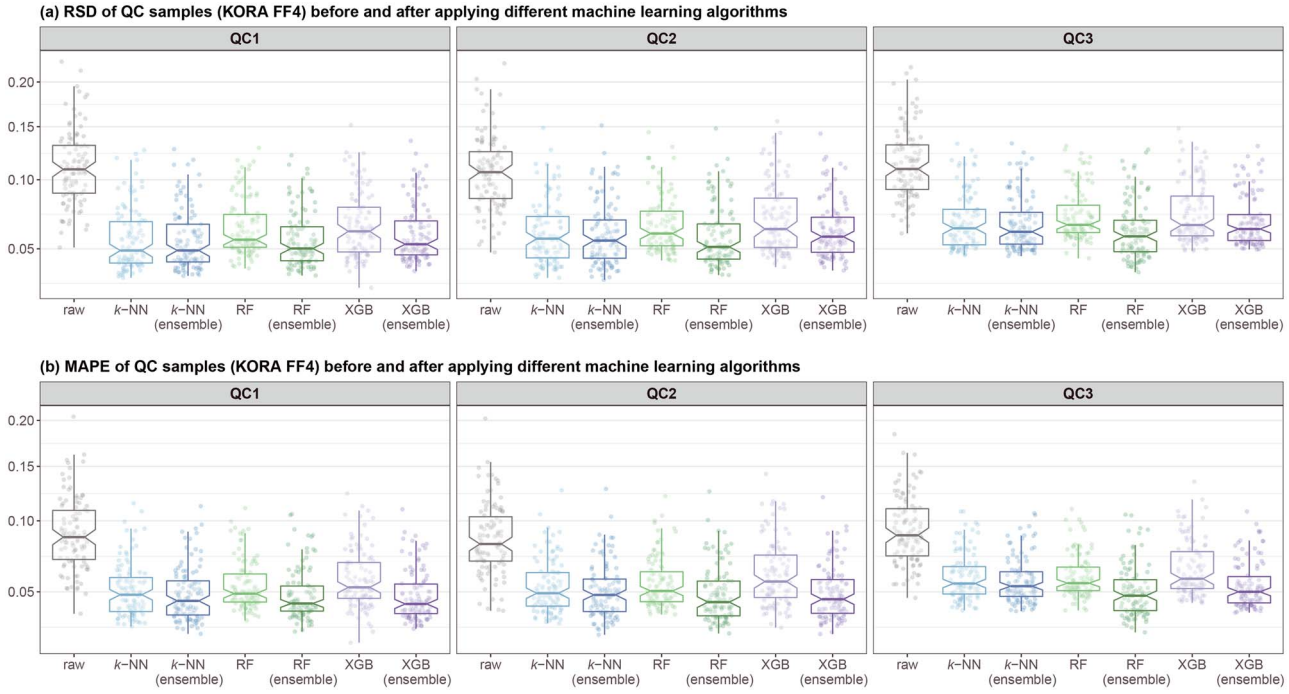

Figure 2. Performance evaluation of general and ensemble architecture of *k*-nearest neighbours (*k*-NN), random forest (RF) and extreme gradient boosting (XGB). The results of non-ensemble models are obtained from the corresponding fine-tuned model (see Table S2 for the hyperparameter list). Ensemble models are trained using the same hyperparameters as the corresponding non-ensemble models. Each dot in each box plot represents the corresponding metric value of one metabolite, while the box shows the overall distribution of the metric values of all metabolites. The RSD **(a)** and MAPE **(b)** results show that all ensemble architectures yield performance improvement over their predecessors which are constructed without using ensemble learning techniques. Please note that the y-axis has been sqrt-transformed.

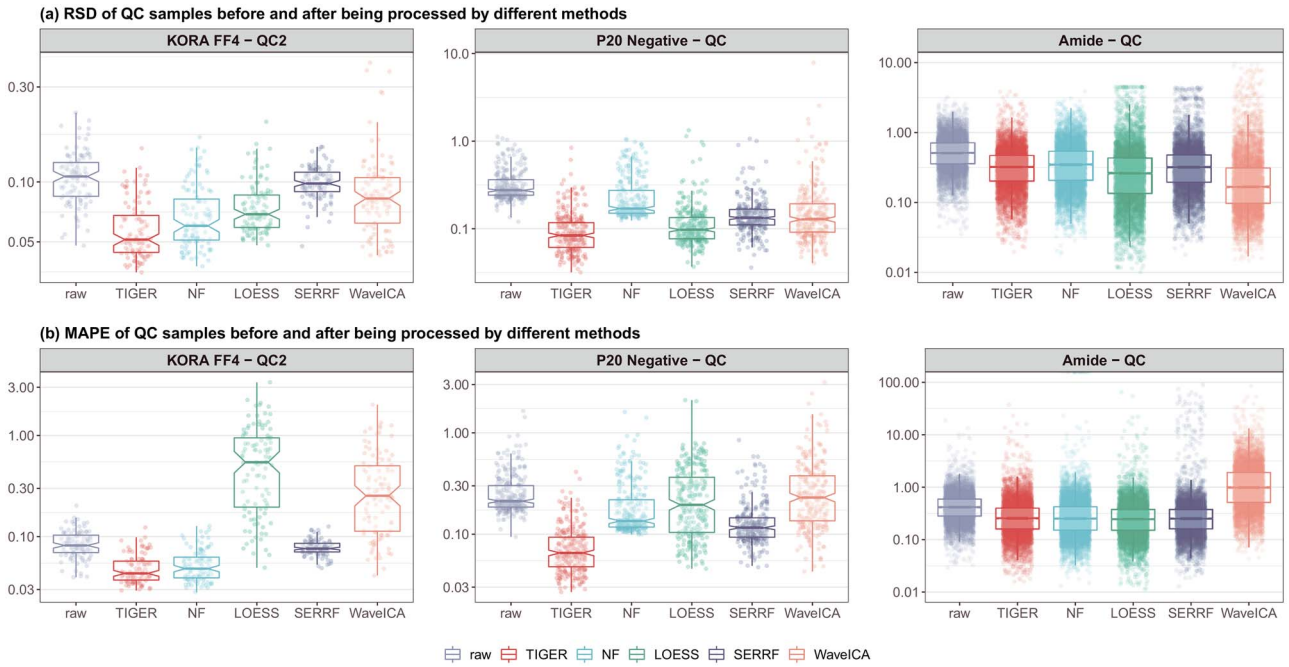

Figure 3. Box plot of RSD **(a)** and MAPE **(b)** results on QC samples from datasets KORA FF4, P20 Negative and Amide. Each dot in each box plot represents the corresponding metric value of one metabolite or variable, while the box shows the overall distribution of the metric values of all metabolites. The performance in this figure is yielded by TIGER's ready-to-use R package, thus the performance of the ensemble RF model here slightly differs from the result in Figure 2 which is obtained from the fine-tuned model of each architecture. Please note that the y-axis has been log10-transformed.

However, the PCA evaluation shows that the batch effect can still be detected in its normalized Amide dataset, although QC samples cluster tightly. This suggests that LOESS suffers from the overfitting problem with a favourable result on QC samples but a substandard performance on subject samples. Evaluated on P20 Negative and Amide dataset, TIGER achieves the most compact clusters for both QC and subject samples, which

demonstrates that TIGER is an appropriate candidate for technical variation removal.

Case Study

This case study is conducted to illustrate how TIGER can be integrated into longitudinal analysis where batch/-plate correction and systematic errors removal are critical and indispensable.

Data Imputation

KORA F4 (Original) has a few missing values which need to be imputed before correction. To ensure an accurate result, a template dataset with simulated missing values is created using KORA F4 (Original). Out of 3061 individuals of 103 metabolites (315 283 data points), six missing values are scattered in the data of four metabolites (PC aa C32:2, PC ae C42:4, SM (OH) C14:1 and SM C20:2). A subset without missing values is first extracted from F4 (Original). For each of these four metabolites, we randomly remove 10 values with the assumption that data of one specific metabolite are missing completely at random (MCAR) [43]. The resulting template dataset is then imputed by four popular algorithms [44, 45] namely predictive mean matching (PMM) [46], classification and regression trees (CART) [47], k-NN imputation [48] and Bayesian linear regression (Norm) [49]. The results are evaluated with MAPE which measures the difference between original values and imputed values. Method PMM achieved the best performance with an MAPE of 0.1220, compared with 0.1301 for k-NN, 0.1323 for Norm and 0.1588 for CART. PMM is used to impute the missing values in KORA F4 (Original).

Data Correction and Cross-Kit Adjustment

In this case study, we aim to correct datasets of three time-points, namely KORA S4, F4 (Original) and FF4. The three datasets exhibit obvious inter-batch technical variation (see the distributions of raw data in Figure 5). Thus, data correction and adjustment are absolutely necessary to ensure a reliable longitudinal analysis. Considering that only KORA FF4 and F4 (Remeasured) used the same kinds of manufacturer-provided QC samples (QC1, QC2 and QC3), but all three datasets provided the same kind of pooled EDTA-plasma QC sample (QC, five per plate), we use QC to correct the datasets of three time-points. And KORA F4 (Original) will be adjusted using the 288 repeated measurements of F4 (Remeasured).

We use KORA FF4 as a reference dataset to which the other datasets are aligned for inter-batch correction. Intra-batch technical variation within the data from 29 plates of KORA FF4 is first eliminated with TIGER. When normalizing KORA S4, the target values ($\bar{y}$ in Eq. 9) are calculated from KORA FF4 so that TIGER attempts to remove the inter-batch technical variation by minimizing the discrepancy between KORA S4 and FF4. TIGER by default eliminates the technical variation within the input dataset, in which the target values are automatically computed using the input dataset itself. With the help of TIGER's flexibility functionality, the target values can be calculated from a reference dataset and then passed to the programme as an argument. This will enable TIGER to align the input dataset with the reference dataset (see Supplementary File–R Package Manual).

The adjustment of dataset KORA F4 consists of two steps. KORA F4 (Remeasured) was generated from four plates of the p180 kit. Data correction is first performed through the method we used for KORA S4 to combat intra- and inter-batches, such that the values in KORA F4 (Remeasured) are aligned to FF4. In the second step, the samples with repeated measurements in KORA F4 (Original) are used as training samples for cross-kit adjustment. In a broad sense, the noises introduced by different kits can be categorized as inter-batch technical variation, but the noises are further amplified due to the change of analytical kit. In our experiment, we regard each of these 288 subject samples as a QC sample. And the remeasured data are used as target values to minimize the difference between KORA F4 (Remeasured) and F4 (Original). After the cross-kit adjustment, dataset KORA F4 (Original) is comparable to FF4.

To evaluate the quality of the adjusted data, RSD of QC1, QC2 and QC3 are calculated on raw F4 (Original) and adjusted F4 (Original). After TIGER's adjustment, the median of RSD calculated on QC1, QC2 and QC3 reduce from 0.1225, 0.1251 and 0.1070 to 0.0967, 0.0953 and 0.0914, respectively. Compared with the data dispersion indicated by RSD, we are more concerned about how close the adjusted original remeasured data are to the normalized remeasured data. Because QC1, QC2 and QC3 are different in KORA F4 (Original) and F4 (Remeasured), we cannot directly compute MAPE on these manufacturer-provided QC samples. To investigate how well TIGER performs on cross-kit adjustment, 4-fold stratified CV is conducted using the 288 samples with remeasurements in normalized KORA F4 (Remeasured) and F4 (Original). MAPE is computed on each validation fold to quantify the difference between the adjusted original data and the normalized remeasured data. In KORA F4 (Original), the 288 original measured samples were spread across 38 plates. For each plate, we randomly and equally split the samples into four groups. Groups of each plate are combined together to construct four folds for stratified CV. After TIGER's adjustment, the median and mean of MAPE were reduced by 16.81% (from 0.2501 to 0.0814) and 19.29% (from 0.2958 to 0.1029), respectively, which indicates that TIGER is effective for cross-kit adjustment.

Analysis for ageing Trends

We further use raw and TIGER normalized datasets of KORA S4, F4 and FF4 to investigate age-associated metabolites. To weaken the influence of diseases and medical treatments on metabolite concentration, subject

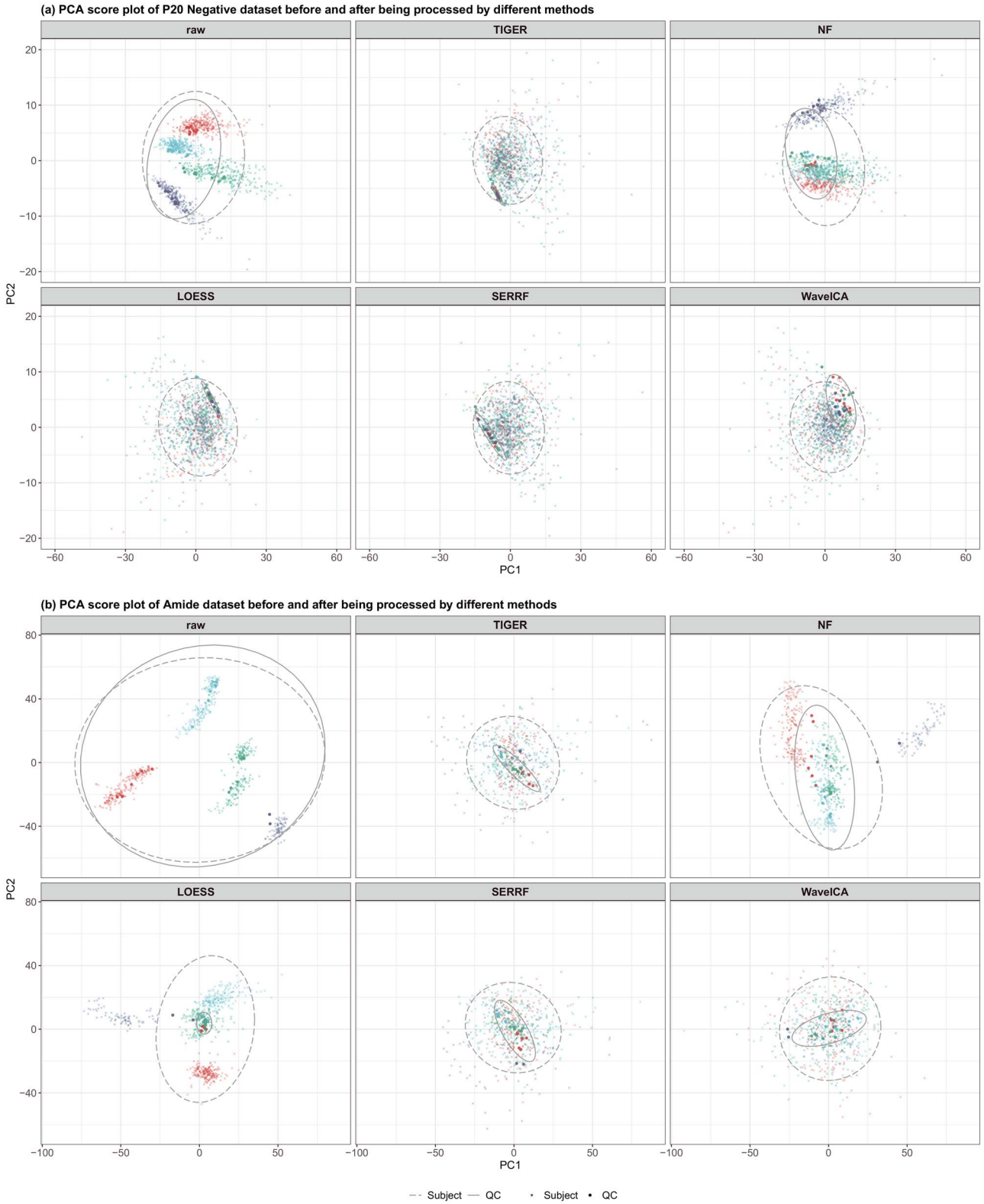

Figure 4. PCA plots of P20 Negative **(a)** and Amide **(b)** dataset. QC samples and subject samples are represented by bigger solid points and smaller partially transparent dots. Samples from different batches, marked with different colours, are expected to mix together after being normalized by a method. **(a)** For P20 Negative dataset, the QC samples are tightly clustered after being processed by TIGER, and the subject samples from different batches are also mixed together. By contrast, NF still has evident batch effects in both QC and subject samples, which proves that NF underfits the data. The patterns of QC samples in the panels of LOESS, SERRF and WaveICA demonstrate noticeable intensity drifts and noises remain in the data they corrected. **(b)** As to the Amide dataset, NF still underfits the data, while LOESS overfits the data. In the panel of LOESS, it can be noted that the QC samples cluster in one group, but subject samples of different batches still gather in their respective communities. This overfitting problem leads to a deceptive good performance of LOESS in Figure 3. TIGER, SERRF and WaveICA yield better results than LOESS and NF, and the PCA plots show that TIGER achieves more reliable results and a better balance between RSD and MAPE than its competitors.

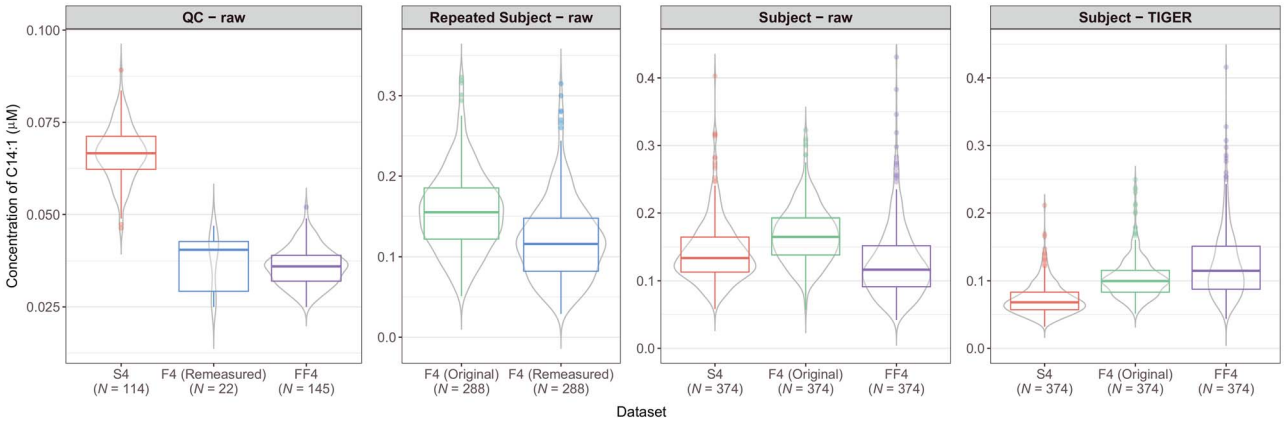

Figure 5. The distribution of the metabolite concentration of C14:1 (Tetradecenoylcarnitine) from the KORA-derived datasets. *N* denotes the number of samples. The box of S4 in the first plot is higher than that of both F4 (Remeasured) and FF4 for the identical QC samples. The second plot shows that the values of F4 (Original) are higher than the values in F4 (Remeasured). The last two plots show the concentration of C14:1 of S4, F4 and FF4 for the 374 individuals with and without TIGER's normalization. The raw values of subject samples theoretically suffer from similar technical variations to the raw values of QC and remeasured samples.

samples of KORA are screened according to their corresponding phenotype data. For each time-point, we exclude individuals with obesity (body mass index, BMI, higher than 35 kg/m^2), or with hypertension (systolic blood pressure higher than 160 mmHg), or with type 1 or type 2 diabetes. Non-fasting samples are also removed. The remaining data of three time-points consist of 374 individuals. The average age of these participants from KORA S4, F4 and FF4 were 61.75, 68.75 and 75.75 years old. The mean values of BMI, fasting glucose and haemoglobin A1c (HbA1c) were relatively stable during the 14 years investigation, whereas the mean of systolic blood pressure decreased (Table S4).

We use a linear mixed-effects model [50] with a random intercept to investigate the relationships between each of the 103 metabolites and age (Table S5). Overall, age is found to be significantly associated with 73 metabolites (*P*-value <4.85e-04 = 0.05/103). By contrast, age is associated with 89 metabolites in the raw data. We notice 38 metabolites show the different significance of age associations in raw and TIGER normalized data. Figure 5 shows the concentration distribution of metabolite C14:1 (Tetradecenoylcarnitine). Estimated on raw data, the regression coefficient of C14:1 ~ age is -5.68e-04 (*P*-value = 0.0038). When fitting the model using the data processed by TIGER, the regression coefficient goes to 3.16e-03 (*P*-value = 3.04e-91). The positive correlation found in C14:1 is consistent with recent longitudinal research, conducted on cohort Wisconsin Registry for Alzheimer's Prevention (WRAP), which demonstrates that C14:1 and C18:1 (Octadecenoylcarnitine) are among the most age-dependent metabolites [51]. Another study [52] also shows that many fatty acids, including C14:1 and C18:1, are significantly increased in midlife. The positive correlation between C14:1 and age can only be revealed in the corrected data, which confirms TIGER's effectiveness in technical variation removal. We also notice the positive correlation between age and metabolite C18:1. Fitting the models with raw and TIGER corrected data, the regression coefficients of C18:1 ~ age are 2.45e-3 (*P*-value = 8.13e-53) and 3.14e-3 (*P*-value = 5.81e-90), respectively. This result implies TIGER does not impair true biological variations within the data.

Using TIGER's dynamic website, readers can interactively examine the associations between age and the concentrations of different metabolites. The ratio and sum of the concentrations of multiple metabolites are also supported to maximize the availability of our website.

Discussion

In this study, we developed TIGER, a reliable method for metabolomics data normalization powered by an adaptable ensemble learning architecture. Evaluated with targeted and untargeted metabolomics datasets, TIGER outperforms four widely used methods (NF, LOESS, SERRF and WaveICA) on both intra- and inter-batch technical variation removal. A case study is performed to illustrate how TIGER improves the detection of true ageing-associated metabolites in a longitudinal analysis.

For many machine learning tasks, models are trained and fine-tuned on training sets which can be regarded as representative of the unseen examples. And more and more advanced machine learning models have been developed to capture the complex but subtle structure hidden within the data. However, a highly sophisticated model may not be robust enough for a dataset with limited sample size or a large quantity of noises, which is often the case in the bioinformatics field. To tackle these issues, an ensemble learning architecture was devised in this study. Instead of turning a weak learner into a strong one or searching for a specific hyperparameter combination that achieves the lowest training error, the architecture improves a model's robustness to noises and mitigates the risk of overfitting by considering the output from multiple learners, though some of them may yield mediocre outcomes. The main idea underlying the proposed architecture is that even weak- performing

base models may help lower the potential variation and contribute information to the ensemble model, which could theoretically outperform a fine-tuned individual model.

TIGER selects RF, an algorithm with moderate complexity, to train its base models. Models trained with high complexity algorithms, such as XGB, can capture not only patterns of technical variation but also random noises in the training set, thus may not generalize well on the unseen examples which greatly differs from the training data. A model with low complexity, such as *k*-NN, is less prone to overfit the training set, but, on the other hand, has limited hyperparameters, which means it only has limited base models to improve accuracy. TIGER by default only selects 5 to 10 highly correlated variables to build ensemble models. We additionally evaluated the impact of different variable numbers on TIGER's performance using the KORA FF4 dataset (Table S6). In the most extreme scenario, TIGER is able to perform normalization with only one available variable and obtain a median of RSD of 0.0570 (QC2) and a median of MAPE of 0.0581 (QC2). In our evaluation, only one type of QC sample (pooled EDTA-plasma) was used as training data, but we expect the performance can be further boosted if the training set can be constructed with additional types of QC samples. This helps the meta model compute more reliable weights to ensure a better generalization ability. The application scope of this ensemble learning architecture is not limited to remove unwanted technical variation and systematic errors in metabolomics data. Its remarkable improvements in generalization ability and satisfactory performance on noisy data may play a positive role in other data modelling tasks.

In addition to RSD, we also used MAPE as well as PCA plots to evaluate the performance of different methods, as RSD alone may not be sufficient to assess one method when raw data are expected to be as close as possible to the corresponding target values. In addition, using RSD alone may result in over-optimistic results and the problem of overcorrection, in which case biological variations are also removed during the normalization. In the case study, MAPE was our primary criterion for quantifying TIGER's performance, as (1) we expect to evaluate how much the adjusted KORA F4 (Original) data differ from F4 (Remeasured) data; (2) the repeated subject samples involve both technical variation and their own biological variation–RSD is no longer applicable. In this scenario, MAPE is more practical than RSD for a reliable evaluation.

We further noticed though LOESS and WaveICA produced results with high error ratios of MAPE, they displayed different patterns with the increase of technical variation. The MAPE of LOESS was higher than that of WaveICA on KORA FF4, the most stable dataset in this study, but when evaluated on the Amide dataset, LOESS already surpassed WaveICA in terms of MAPE. In fact, LOESS achieved the most balanced result among the five methods, if we only consider the QC samples (Figure 3). But when focusing on subject samples, we found the normalized data still showed strong batch effects (Figure 4). The problem is that the LOESS-based model captures data patterns that only exist in QC samples and only partially reflect the whole datasets, thus yielding unfavourable results for subject samples. Although QC samples can be split into training (80%) and test sets (20%), the evaluation still entails the risk of producing over-optimistic results. Therefore, the evaluation in this scenario can be deceptive and should be viewed with caution. LOESS overfitted the data though the QC samples in Amide are the mix of the aliquots of all subject samples. The method may yield worse results if the training samples are more distinct from subject samples. To avoid potential bias, we highly recommend using different kinds of evaluation metrics and methods, including, but not limited to, RSD, MAPE and PCA, to perform data evaluation. Furthermore, it is also beneficial to introduce several kinds of QC samples into the acquisition process of metabolomics data. In the evaluation of the KORA FF4, identical pooled EDTA-plasma QC samples (five on each plate) were used to train QC-based methods, while manufacturer-provided QC plasma samples (QC1, QC2 and QC3) and subject serum samples were used to construct the test set and evaluate performance. The overall data distribution of the QC can greatly differ from the distributions of QC1, QC2 and QC3. Hence, a method with good results on QC1, QC2 and QC3 will theoretically generalize robustly on subject samples. Although plasma and serum metabolite concentration profiles are different, as we have previously shown with 377 KORA individuals, they are also highly correlated [53]. The strong performance of TIGER in the evaluation of the KORA dataset demonstrates that TIGER is effective in reducing technical variation, even if the training data are not fully representative of the test data.

Another observation is that as the technical variation increases, the performance of NF becomes weaker. This linear regression-based method was only inferior to TIGER in the evaluation of KORA FF4, but it was surpassed by LOESS, SERRF and WaveICA on P20 Negative and Amide. We speculated that this might result from the ineptitude of the linear regression in capturing complex data patterns. The raw data of KORA FF4 used in this study are already of quite high quality, thus NF is able to further improve the data quality [54]. When being applied to a dataset with a large quantity of noises, such as Amide, NF underfits the data and fails to yield satisfactory results. We also noticed that SERRF, contrasting with NF, gradually outperformed LOESS and WaveICA, with the increase of technical variation. The favourable performances achieved by TIGER and SERRF may be partly owing to the robustness of the RF algorithm that underlies the methods. The ensemble learning architecture further improved TIGER's stability and effectiveness, thus achieving better overall performances than SERRF across the three datasets.

Data acquisition of a large data cohort often spans many years and utilizes different analytical platforms, which makes data inevitably contain various unwanted noises. In the case study, TIGER was applied to a longitudinal analysis which involved three time-points data measured with two kits (p150 and p180). We demonstrated that TIGER can considerably reduce intra- and inter-batch/plate effects introduced by different kits and improve the homogeneity and comparability of data, which helps identify true candidate biomarkers of disease-associated metabolites. Previous studies have shown that the concentrations of many metabolites, including C14:1 and C18:1, are positively correlated with age [37, 51, 52, 55]. Our case study shows that many other metabolites are also associated with age (see Table S5 and interactive website). Many associations can only be revealed after removing the potential technical variation. It would be of great significance to perform more in-depth investigations and replicate these novel findings with further studies.

Aiming to offer practical and customizable functions for technical variation elimination, TIGER can also be used to eliminate the systematic errors introduced by different analytical kits. To our knowledge, TIGER is the first ready-to-use tool that supports cross-kit adjustment. Cross-kit adjustment in this study was evaluated on the kits of two different versions. The performance of adjusting data from two completely different kits is not tested. Adjusting data from entirely distinct sources is not recommended, and the results may be misleading. TIGER makes no assumption about data source and is theoretically applicable to numeric data measured with various techniques. But in this study, TIGER was only evaluated with data from LC-MS analysis.

In terms of computational speed, as one dataset generally has only a small number of QC samples for model training, the limited increase of samples may not make a big impact on the complexity of TIGER. Additionally, because TIGER by default only selects 5–10 highly correlated variables to train each model, the increase of variables will not increase the complexity of each ensemble model. However, the running time can grow with the number of variables and batches as TIGER builds different ensemble models for different metabolites and batches. For a dataset with D variables and B batches or plates, TIGER needs to train $D \times B$ models to normalize the whole dataset. Evaluated on a general hardware environment configured with a processor Intel® Core™ i9-10885, 32 GB memory and 64-bit Windows 10 OS, TIGER took 2 min 28.85 s, 2 min 21.75 s and 57 min 30.46 s to process datasets KORA FF4 (103 variables, 22 plates), P20 Negative (268 variables, four batches) and Amide (6402 variables, four batches) in parallel with 8 cores. In this case, TIGER takes around 140 s to build 1000 models when 20–30 QC samples per batch are available for model training. We also evaluated the running time of TIGER under different numbers of highly correlated variables (Table S6). The result shows that the running time increases approximately linearly with the number of variables. TIGER trades model complexity for a robust prediction and requires more processing time than other competing methods, but the increased cost is still acceptable.

In sum, TIGER has the following merits:

(i) TIGER integrates the RF algorithm into an innovative ensemble learning architecture. Benefiting from this advanced architecture, TIGER is robust to outliers, free from model tuning and less likely to be affected by specific hyperparameters. Although many QC-based methods require at least 10 QC samples per batch to perform reliable normalization, TIGER in our evaluation achieved strong performance with only five QC samples. The good generalization ability makes TIGER also suitable for small-scale datasets.

(ii) TIGER is highly reliable and robust. As shown with multiple criteria (RSD, MAPE and PCA) and evaluated using three different kinds of metabolite datasets, the overall performance of TIGER surpasses four popular methods, including NF, LOESS, SERRF and WaveICA. TIGER is also less prone to underfit or overfit the data.

(iii) TIGER is remarkably flexible and versatile. TIGER supports targeted and untargeted metabolomics data and is competent to perform intra- and inter-batch technical variation removal as well as cross-kit adjustment to ensure data obtained from different analytical assays can be effectively combined and compared. Moreover, unlike many existing tools that only support one kind of QC sample, TIGER takes advantage of all available QC samples to build ensemble models, which helps the model generalize well on unseen examples.

(iv) TIGER is readily accessible and convenient to use. Released as an R package, TIGER is compatible with almost all widely used operating systems (OS), including Windows, UNIX/Linux and macOS. TIGER is accepted by Comprehensive R Archive Network (CRAN) and can be installed easily in R. In contrast to many tools that rely on users to manually perform parameter optimization, which requires users to be familiar with the algorithms and related arguments, TIGER can effectively process the datasets with its default set-up. TIGER also supports parallel computing to make full use of computing resources and accelerate the process.

Availability

Data

KORA FF4's QC data are included in TIGER's R package. The cohort data that support the findings of this study are operated by Helmholtz Zentrum München and available via KORA platform https://www.helmholtz-muenchen.de/en/kora/index.html upon reasonable request.

P20 Negative dataset, provided by SERRF as a demo dataset, is available at https://slfan2013.github.io/SERRF-online/. The dataset is downloaded and used with the consent of the authors.

Amide dataset is included in WaveICA's R package and available at https://github.com/dengkuistat/WaveICA. The dataset was downloaded and used in compliance with the copyright policy of the publisher.

R Package

The R implementation of TIGER, TIGERr, is a free R package under the GNU General Public Licence. The package is developed with the help of dependencies ppcor [33], randomForest [40], caret [56] and pbapply [57]. The documentation is generated with roxygen2 [58].

The package has been included in CRAN. Users can simply install TIGER in R via command `install.packages("TIGERr")`, and an appropriate version, as well as dependencies, will be installed automatically. The package is also available at CRAN (https://CRAN.R-project.org/package=TIGERr) and GitHub (https://github.com/HAN-Siyu/TIGER).

Dynamic Website

The dynamic website (accessible at https://han-siyu.github.io/TIGER_web/) is a shiny-based application [59] which supports interactive figures for the detailed results of KORA-derived datasets. Packages including shinydashboard [60], flexdashboard [61], ggplot2 [62], ggsci [63] and plotly [64] are employed in the background to control layouts, output figures and enable interactive features.

The dynamic website contains two function modules for the results of method evaluation and longitudinal analysis, respectively. Users can enter the name of the metabolite of interest to compare the results of different methods and check the distribution of its concentration. Relevant statistics will be displayed when the cursor hovers over the plots.

Key Points

- An ensemble learning architecture is developed to enhance model performance on noisy metabolomics datasets. The architecture, comprised of multiple base models and one meta model, can be adapted to a wide range of machine learning algorithms and expanded to a scope beyond the correction of metabolomics data.
- Based on our ensemble learning architecture, we developed TIGER as a novel algorithm for eliminating technical variation in metabolomics data. Benchmarked against several widely used methods, TIGER shows the most robust and reliable performance on targeted and untargeted metabolomics datasets for eliminating intra- and inter-batch technical variation. TIGER also demonstrates strong performance on cross-kit adjustment, which greatly improves data reproducibility. It enables the combination and comparison of experimental data from different analytical kits to reliably identify candidate metabolite biomarkers of interest.
- Using TIGER normalized data, many age-associated metabolites are revealed in our case study. Important patterns only appear after data correction, which proves that the removal of systematic errors is crucial for the longitudinal analysis of metabolomics data.
- TIGER is released as an R package and can be run on multiple OS platforms. Our dynamic website allows users to compare the performance of different methods and browse the associations between metabolite concentrations and age.
- Various critical metrics and methods, including, but not limited to, RSD, MAPE and PCA, should be considered to evaluate the normalized metabolomics data to ensure a reliable result and avoid the risks of overfitting and overcorrection.

Acknowledgements

We express our appreciation to all KORA study participants for their blood donation and time. The KORA-Study group consists of A. Peters (speaker), H. Schulz, L. Schwettmann, R. Leidl, M. Heier, K. Strauch and their co-workers, who are responsible for the design and conduct of the KORA study. We thank the staff from the Institute of Epidemiology, Helmholtz Zentrum München and the Genome Analysis Center Multi-omic platforms, especially J. Scarpa, N. Lindemann, Dr. W. Römisch-Margl and K. Sckell, for helping in sample logistics, data and straw collection and metabolomic measurements.
The authors are grateful to Dr. E. Zeggini (Institute of Translational Genomics, Helmholtz Zentrum München), Dr. D. Frishman (TUM School of Life Sciences, Technical University of Munich), Dr. M. Covic (Institute of Epidemiology, Helmholtz Zentrum München), Dr. C. H. Ek (Department of Computer Science and Technology, University of Cambridge) and Dr. E. Spiegel (Institute of Computational Biology, Helmholtz Zentrum München) for their valuable and constructive suggestions during the development of TIGER. The authors thank Y. Tai, R. Wang, M. Cheng, Y. Guo, H. Li and L. Fan for helping test our R package. The authors also would like to thank the editor and anonymous reviewers for handling this manuscript.

Funding

The KORA study was initiated and financed by Helmholtz Zentrum München - German Research Center for Environmental Health, which is funded by the German Federal Ministry of Education and Research and by the State of Bavaria. Furthermore, KORA research was supported within the Munich Center of Health Sciences (MC-Health), Ludwig-Maximilians-Universität, as part of LMUinnovativ.

The German Diabetes Center is supported by the German Federal Ministry of Health (Berlin, Germany). This study was supported in part by a grant from the German Federal Ministry of Education and Research to the German Center for Diabetes Research (DZD).

Part of this study was supported by the funding from the European Union's Horizon 2020 research and innovation programmes: The 210997-iPDM-GO EIT Health Innovation Project supported by the European Institute of Innovation and Technology (EIT), a body of the European Union; The Innovative Medicines Initiative 2 (IMI2), project CARDIATEAM funded under the Grant Agreement No. 821508.

References

1. Sen P, Lamichhane S, Mathema VB, et al. Deep learning meets metabolomics: A methodological perspective. *Brief Bioinform* 2021;**22**(2):1531–42.
2. Pang Z, Chong J, Li S, et al. Metaboanalystr 3.0: Toward an optimized workflow for global metabolomics. *Metabolites* 2020;**10**(5):186.
3. Grebe SKG, Singh RJ. Lc-ms/ms in the clinical laboratory–where to from here? *The Clinical biochemist reviews* 2011;**32**(1):5.
4. Mapstone M, Cheema AK, Fiandaca MS, et al. Plasma phospholipids identify antecedent memory impairment in older adults. *Nat Med* 2014;**20**(4):415–8.
5. Siskos AP, Jain P, Römisch-Margl W, et al. Interlaboratory reproducibility of a targeted metabolomics platform for analysis of human serum and plasma. *Anal Chem* 2017;**89**(1):656–65.
6. de Livera AM, Dias DA, de Souza D, et al. Normalizing and integrating metabolomics data. *Anal Chem* 2012;**84**(24): 10768–76.
7. Kuligowski J, Sánchez-Illana Á, Sanjuán-Herráez D, et al. Intra-batch effect correction in liquid chromatography-mass spectrometry using quality control samples and support vector regression (qc-svrc). *Analyst* 2015;**140**(22):7810–7.
8. Tokareva AO, Chagovets VV, Kononikhin AS, et al. Normalization methods for reducing interbatch effect without quality control samples in liquid chromatography-mass spectrometry-based studies. *Anal Bioanal Chem* 2021;**413**(13):3479–86.
9. Auer PL, Doerge RW. Statistical design and analysis of rna sequencing data. *Genetics* 2010;**185**(2):405–16.
10. Hicks SC, Irizarry RA. Quantro: a data-driven approach to guide the choice of an appropriate normalization method. *Genome Biol* 2015;**16**(1):1–8.
11. Scherer A. *Batch effects and noise in microarray experiments: sources and solutions*, Vol. **868**. John Wiley & Sons, 2009.
12. Wehrens R, Hageman JA, van Eeuwijk F, et al. Improved batch correction in untargeted ms-based metabolomics. *Metabolomics* 2016;**12**(5):88.
13. Wang W, Zhou H, Lin H, et al. Quantification of proteins and metabolites by mass spectrometry without isotopic labeling or spiked standards. *Anal Chem* 2003;**75**(18):4818–26.
14. Huang J, Huth C, Covic M, et al. Machine learning approaches reveal metabolic signatures of incident chronic kidney disease in individuals with prediabetes and type 2 diabetes. *Diabetes* 2020;**69**(12):2756–65.
15. Li B, Tang J, Yang Q, et al. Performance evaluation and online realization of data-driven normalization methods used in lc/ms based untargeted metabolomics analysis. *Sci Rep* 2016;**6**(1):1–13.
16. Cleveland WS, Devlin SJ. Locally weighted regression: an approach to regression analysis by local fitting. *J Am Stat Assoc* 1988;**83**(403):596–610.
17. Fan S, Kind T, Cajka T, et al. Systematic error removal using random forest for normalizing large-scale untargeted lipidomics data. *Anal Chem* 2019;**91**(5):3590–6.
18. Breiman L. Random forests. *Machine learning* 2001;**45**(1):5–32.
19. Sysi-Aho M, Katajamaa M, Yetukuri L, et al. Normalization method for metabolomics data using optimal selection of multiple internal standards. *BMC bioinformatics* 2007;**8**(1):1–17.
20. Workman C, et al. A new non-linear normalization method for reducing variability in dna microarray experiments. *Genome Biol* 2002;**3**(9):1–16.
21. Luan H, Ji F, Chen Y, et al. stattarget: A streamlined tool for signal drift correction and interpretations of quantitative mass spectrometry-based omics data. *Anal Chim Acta* 2018;**1036**: 66–72.
22. Deng K, Zhang F, Tan Q, et al. Waveica: A novel algorithm to remove batch effects for large-scale untargeted metabolomics data based on wavelet analysis. *Anal Chim Acta* 2019;**1061**:60–9.
23. Daubechies I. *The wavelet transform, time-frequency localization and signal analysis*. Princeton University Press, 2009.
24. Hyvärinen A, Oja E. Independent component analysis: algorithms and applications. *Neural Netw* 2000;**13**(4–5):411–30.
25. Renard E, Branders S, Absil P-A. Independent component analysis to remove batch effects from merged microarray datasets. In: *International Workshop on Algorithms in Bioinformatics*. Springer, 2016, 281–92.
26. Johnson WE, Li C, Rabinovic A. Adjusting batch effects in microarray expression data using empirical bayes methods. *Biostatistics* 2007;**8**(1):118–27.
27. The Human Serum Metabolome (HUSERMET) Consortium, Dunn WB, Broadhurst D, et al. Procedures for large-scale metabolic profiling of serum and plasma using gas chromatography and liquid chromatography coupled to mass spectrometry. *Nat Protoc* 2011;**6**(7):1060–83.
28. De Livera AM, Sysi-Aho M, Jacob L, et al. Statistical methods for handling unwanted variation in metabolomics data. *Anal Chem* 2015;**87**(7):3606–15.
29. Brunius C, Shi L, Landberg R. Large-scale untargeted lc-ms metabolomics data correction using between-batch feature alignment and cluster-based within-batch signal intensity drift correction. *Metabolomics* 2016;**12**(11):1–13.
30. Li B, Tang J, Yang Q, et al. Noreva: normalization and evaluation of ms-based metabolomics data. *Nucleic Acids Res* 2017;**45**(W1):W162–70.
31. Shen X, Gong X, Cai Y, et al. Normalization and integration of large-scale metabolomics data using support vector regression. *Metabolomics* 2016;**12**(5):1–12.
32. van der Laan MJ, Polley EC, Hubbard AE. Super learner. *Stat Appl Genet Mol Biol* 2007;**6**(1).
33. Kim S. ppcor: an r package for a fast calculation to semi-partial correlation coefficients. *Communications for statistical applications and methods* 2015;**22**(6):665.
34. Krumsiek J, Suhre K, Illig T, et al. Gaussian graphical modeling reconstructs pathway reactions from high-throughput metabolomics data. *BMC Syst Biol* 2011;**5**(1):1–16.
35. Holle R, Happich M, Löwel H, et al. Kora-a research platform for population based health research. *Das Gesundheitswesen* 2005;**67**(S 01):19–25.
36. Haid M, Muschet C, Wahl S, et al. Long-term stability of human plasma metabolites during storage at- 80 c. *J Proteome Res* 2018;**17**(1):203–11.

37. Yu Z, Zhai G, Singmann P, et al. Human serum metabolic profiles are age dependent. *Aging Cell* 2012;**11**(6):960–7.
38. Wang-Sattler R, Yu Z, Herder C, et al. Novel biomarkers for pre-diabetes identified by metabolomics. *Mol Syst Biol* 2012;**8**(1):615.
39. Beygelzimer A, et al. *FNN: Fast Nearest Neighbor Search Algorithms and Applications*, 2019, R package version 1.1.3.
40. Liaw A, et al. Classification and regression by randomforest. *R news* 2002;**2**(3):18–22.
41. Chen T, Guestrin C. Xgboost: A scalable tree boosting system. In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, 785–94.
42. Wu S, Yuan Q, Yan Z, et al. *xgboost: Extreme Gradient Boosting*, 2021, R package version 1.4.1.1.
43. Rubin DB. Inference and missing data. *Biometrika* 1976;**63**(3): 581–92.
44. vanBuuren S, Groothuis-Oudshoorn K. mice: Multivariate imputation by chained equations in r. *J Stat Softw* 2011;**45**(3):1–67.
45. Van Buuren S. *Flexible imputation of missing data*. CRC press, 2018.
46. Little RJA, Rubin DB. *Statistical analysis with missing data*, volume 793. *John Wiley & Sons* 2019.
47. Doove LL, van Buuren S, Dusseldorp E. Recursive partitioning for missing data imputation in the presence of interaction effects. *Computational statistics & data analysis* 2014;**72**:92–104.
48. Torgo L. *Data Mining with R, learning with case studies*. Chapman and Hall/CRC, 2010.
49. Rubin DB. *Multiple imputation for nonresponse in surveys*, volume 81. *John Wiley & Sons* 2004.
50. Kuznetsova A, et al. lmerTest package: Tests in linear mixed effects models. *J Stat Softw* 2017;**82**(13):1–26.
51. Darst BF, Koscik RL, Hogan KJ, et al. Longitudinal plasma metabolomics of aging and sex. *Aging (Albany NY)* 2019; **11**(4):1262.
52. Pararasa C, Ikwuobe J, Shigdar S, et al. Age-associated changes in long-chain fatty acid profile during healthy aging promote pro-inflammatory monocyte polarization via ppar γ*Aging Cell* 2016;**15**(1):128–39.
53. Yu Z, Kastenmüller G, He Y, et al. Differences between human plasma and serum metabolite profiles. *PloS one* 2011;**6**(7): e21230.
54. Huang J, Covic M, Huth C, et al. Validation of candidate phospholipid biomarkers of chronic kidney disease in hyperglycemic individuals and their organ-specific exploration in leptin receptor-deficient db/db mouse. *Metabolites* 2021;**11**(2):89.
55. Chaleckis R, Murakami I, Takada J, et al. Individual variability in human blood metabolites identifies age-related differences. *Proc Natl Acad Sci* 2016;**113**(16):4252–9.
56. Kuhn M, et al. Building predictive models in r using the caret package. *J Stat Softw* 2008;**28**(5):1–26.
57. Solymos P, Zawadzki Z. *pbapply: Adding Progress Bar to '*apply' Functions*, 2020, R package version 1.4-3.
58. Wickham H, et al. *roxygen2: In-Line Documentation for R*, 2020, R package version 7.1.1.
59. Chang W, et al. *shiny: Web Application Framework for R*, 2021, R package version 1.6.0.
60. Chang W, Ribeiro BB. *shinydashboard: Create Dashboards with 'Shiny'*, 2018, R package version 0.7.1.
61. Iannone R, et al. *flexdashboard: R Markdown Format for Flexible Dashboards*, 2020, R package version 0.5.2.
62. Wickham H. *ggplot2: Elegant Graphics for Data Analysis*. New York: Springer-Verlag, 2016.
63. Xiao N. *ggsci: Scientific Journal and Sci-Fi Themed Color Palettes for 'ggplot2'*, 2018, R package version 2.9.
64. Sievert C. *Interactive Web-Based Data Visualization with R, plotly, and shiny*. Chapman and Hall/CRC, 2020.