BatchServer: a web server for batch effect evaluation, visualization and correction

Tiansheng Zhu[1,2,3], Guo-Bo Chen[4], Chunhui Yuan[2,3], Rui Sun[2,3], Fangfei Zhang[2,3], Xiao Yi[2,3], Shuigen Zhou[1,*], Tiannan Guo[2,3,*]

[1]Shanghai Key Lab of Intelligent Information Processing, and School of Computer Science, Fudan University, China.

[2]Zhejiang Provincial Laboratory of Life Sciences and Biomedicine, Key Laboratory of Structural Biology of Zhejiang Province, School of Life Sciences, Westlake University, 18 Shilongshan Road, Hangzhou 310024, Zhejiang Province, China.

[3]Institute of Basic Medical Sciences, Westlake Institute for Advanced Study, 18 Shilongshan Road, Hangzhou 310024, Zhejiang Province, China.

[4]Clinical Research Institute, Zhejiang Provincial People's Hospital, People's Hospital of Hangzhou Medical College, Hangzhou, Zhejiang, China

Corresponding Author.

Shuigen Zhou: sgzhou@fudan.edu.cn;

Tiannan Guo: guotiannan@westlake.edu.cn

Abstract

Batch effects are unwanted data variations that may obscure biological signals, leading to bias or errors in subsequent data analyses. Effective evaluation and elimination of batch effects are necessary for omics data analysis. In order to facilitate the evaluation and correction of batch effects, here we present BatchSever, an open-source R/Shiny based user-friendly interactive graphical web platform for batch effects analysis. In BatchServer we introduced autoComBat, a modified version of ComBat, which is the most widely adopted tool for batch effect correction. BatchServer uses PVCA (Principal Variance Component Analysis) and UMAP (Manifold Approximation and Projection) for evaluation and visualizion of batch effects. We demonstate its application in multiple proteomics and transcriptomic data sets. BatchServer is provided at https://lifeinfo.shinyapps.io/batchserver/ as a web server. The source codes are freely available at https://github.com/guomics-lab/batch_server.

Keywords: batch effects, data pre-process, ComBat, bioinformatics, web server

INTRODUCTION

High-throughput omics data, such as mass spectrometric (MS) -based proteomics data and next generation sequencing (NGS)-based data, are usually generated with the unwanted systematic variation or so-called “batch effects”. Batch effects result from technical variations, different laboratory conditions, reagent lots, and personnels [1, 2]. If not handled properly, batch effects may distort biological signals [3] and mislead downstream analyses [4]. Proper evaluation and correction of batch effects are thus essential in analyzing large-scale omics data.

Principal variance component analysis (PVCA) [5] and uniform manifold approximation and projection (UMAP)[6] are two widely adopted methods to identify batch effects by visualization. PVCA leverages Principal Component Analysis (PCA) and Variance Components Analysis (VCA), which fits a mixed linear model to estimate the proportion of variation of each factor. PCA [7] is a matrix factorization algorithm, for selecting top principal components as a represention of a multi-dimensional data set. VCA is a method for assessing the proportion of variance attributable to the main effects of random effect variables and the interaction of variables. PVCA combines the characteristics of both PCA and VCA, and has been applied to evaluate the batch effects and effectiveness of batch effect correction [5]. As an emerging non-linear dimensionality reduction method, UMAP is increasing used for visualization of single cell cytometry and transcriptome data [6]. We applied these two methods to evaluate batch effects.

The most intuitive way to correct batch effects is to normalize data, which adjusts the global properties of the data by comparing individual samples. However, simply exploiting normalization is sometimes insufficient to remove complex batch effects [1]. To effectively eliminate batch effects, several methods have been developed, including singu-

lar value decomposition (SVD) [8], surrogate variable analysis (SVA) [9], exploBATCH [10], BatchI [11] and ComBat [12]. Among those methods, ComBat based on a parametric or non-parametric empirical Bayes strategy is arguably the most widely method for batch correction. ComBat has been reported to be reliable and robust to outliers when tested over a large number of samples [2, 13]. However, users need to determine whether to use parametric bayes-based method or nonparametric bayes-based method. This is actually dependent on the distribution of data that is usually unknown unless evaluated [12]. Here, we modified ComBat, and implemented automated determinion of this parameter. The modified ComBat is called autoComBat. We further developed an open source web server – BatchServer that integrates PVCA, UMAP and autoComBat to provide researchers with an easy-to-use interface to evaluate and correct potential batch effects in large-scale omics data.

MATERIALS AND METHODS

BatchServer is written in R and calls several R packages, including Shiny and Shinydashboard for R/Shiny web interface; pvca for batch effect evaluation; plotly, umap and ggplot2 for batch effect visualization; sva for batch effect correction. BatchServer is inherited from the interactive micro-service framework Shiny, integrating web interface in app.R, ui.R for the interface degisn, and server.R for data processing, other source files are called through global.R.

RESULTS AND DISCUSSION

The architecture design of BatchServer

To provide an easy-to-use interactive user interface for visualize the batch effects by figures and tables with option of download of the batch corrected data matrix, we designed the architecture of BatchServer as three main layers (**Figure 1**). The data input layer is provided for uploading input files of data files and sample information files and for interactively selecting batches and covariate names. The data processing layer is responsible for batch effect estimation, visualization and correction. The data output layer can display and download data processing results.

The detailed instructions for BatchServer are provided in the Readme section of the online web page. Here we briefly descript the usage of BatchServer. For data input, a data file and a sample information file are required. The format of these files can be tab-delimited, space-separated, comma-delimited or an Excel file. BatchServer provides a sample test data file by the bladderbatch package in the Readme section. The user can upload these two files in the “Data Input” menu, then click the “Submit” button. The data read module will read, process, and store the files for subsequent uses. Users can evaluate whether the data have batch effects using PVCA or UMAP with the online server. Both methods will display the visualization of batch effects. Once the batch effect is present, it can be adjusted using improved ComBat. Users can check and download the results of batch effect evaluation. The corrected data can be also downloaded.

Imputation of missing values

Missing values are common in omics data sets due to technical or biological issues [14]. The mechanisms for missing values occurrence are complex and can be estimated in a number of ways [15]. Most statistical and machine learning methods do not allow a large portion of missing values; therefore, BatchServer provides four common and com-

putationally efficient ways to replace missing values by ‘1’, ‘0’, ‘10% of the minimum’, or ‘the minimum’, where the minimum is the minimal value in the upload data matrix.

AutoComBat: automated ComBat

Although parametric and nonparametric ComBat often brought out similar results, the parametric Bayes implantation was much computationally faster than its nonparametric one. The choice of using parametric or non-parametric Bayes implementation should be upon the goodness-of-fit test for the underlying distribution of the data set. Here we developed an automated ComBat called autoCombat in the BatchServer.

ComBat is essentially a linear model that attempts to eliminate batch effects directly from the data sets. Y_{ijg} is defined to represent the feature (e.g. protein or gene) g of sample j from batch i. Define a model that assumes

$$Y_{ijg} = \alpha_g + X\beta_g + \gamma_{ig} + \delta_{ig}\epsilon_{ijg}, \quad (1)$$

here α_g is the overall value of g, X is a design matrix for sample conditions, and β_g is the vector of the regression coefficients. The error term, ϵ_{ijg}, is assumed to follow a normal distribution $N \sim (0, \sigma_g^2)$. γ_{ig} and δ_{ig} represent the additive and multiplicative batch effects of batch i for feature g, respectively. If (2) holds, ComBat will use the parametric Bayes method, otherwise it will use the non-parametric Bayes method to estimate γ_{ig} and δ_{ig}^2.

$$\gamma_{ig} \sim N(\gamma_i, \tau_i^2), \text{ and } \delta_{ig}^2 \sim \text{Inverse Gamma}(\lambda_i, \theta_i) \quad (2)$$

The hyperparameters γ_i, τ_i^2, λ_i, θ_i are estimated empirically from data using the method of moments[12]. When using ComBat, users must manually modify the default parameters to determine whether to use the parametric or non-parametric empirical Bayes method. For ease of use, we employed Kolmogorov-Smirnov Goodness of Fit Test (K-S) to test whether the additive parameter γ_{ig} fits normal distribution and the multiplicative

parameter δ^2_{ig} fits inverse gamma distribution for each batch at the significance level of 0.05 [16], respectively. In other words, we incorprated a 'autoComBat" module in BatchServer which automatically switches between parametric or non-parametric Bayes method upon the detected distribution underlying the input data.

The autoComBat in BatchServer provides an additional option 'auto' for parameter "par.prior", that automatically determines the use of parametric or non-parametric Bayes method. For a better visual experience, BatchServer not only plots all batches interactively, but also improves the visualization by highlighting and changing the colors of output lines and points of ComBat. Using the default parameters, we can verify the prior plot of $\hat{\gamma}$ and $\hat{\delta}$ of batch '1' which passed the K-S test (**Figure 2A**). Batch '4' did not pass the K-S test which indicated parametric (red) and empirical (blue) estimation was not fit well (**Figure 2B**).

Application of PVCA and UMAP

The PVCA method used to assess the proportion of each variable source for a given data set, but it is not universally applicable by the limitation of only accepting ExpressionSet objects. To improve the usability, we wrapped the pvcaBatchAcessess function in the pvca package and simplified the input data format to integrate the improved PVCA into BatchServer. On the other hand, UMAP was integrated into BatchServer to visualize and judge the batch effects of high dimensional omics data.

Illustrative applications of BatchSever

To test the performance of BatchServer, we used one proteomics data set and three transcriptomics data sets for the evaluation because the latter are more accessible in the literature. As a benchmark, we used transcriptomics datasets are from the default dataset from bladderbatch package and ComBat package. We also applied the BatchServer to a

published proteomic data of NCI-60 dataset cell lines [17]. The missing values of all the four data sets were replaced by '1'. Each data set was log2 transformed and quantile normalized as suggested by Wilson [13]. The data before and after autoComBat adjusted were analyzed and evaluated by pvca and umap module.

The first data set of transcriptomics data is a combined data matrix of GSE19804 and GSE10072 based on their shared probes, which are microarray-based transcriptomic data of 227 lung cancer disease (118 cancers and 109 controls) in female non-smokers [2]. That naturally brings out a data that were generated from two distinct batches (we named them as batch1 and batch2). **Figure 3A** showed an estimation of proportions of each factor by pvca: 1) before using autoComBat (no correction) 71.57% variation by batch, 16.87% by biological signals (tumor and normal), 0.73% to interaction between batch and biological types, and 10.84% left to residual variation; 2) after autoComBat correction, variation by batch reduced to nearly zero and consequently intensified biological variation considerably. Figure 3B and 3C showed consistent effect of batch variation correction using UMAP with Figure 3A . The results were identical when using nonparametric and auto parameters (Figure 3A, B, C) because neither batch passed the K-S test (Figure 3D). Although the computational cost for method choice was negligible for ComBat (Supplementary Table 1), the nonparametric estimation was always computational expensive than the parametric one. We also tested another two transcriptomics datasets. Both showed substantial batch effects as visualized by our BatchServer, but after applying automated procedures the batch effects were well controlled, leading to enhanced biological signals. Details were presented in Supporting Information.

The proteomics data set are from NCI-60 cell lines[17], containing 3,171 proteins and 60 samples of 12 batches. This data set was generated by Pressure Cyling Technology

coupled with data independent acquisition (PCT-DIA)[18]. The potential factor of batch effects could be from the different tumor type or from PCT. We processed this data set with the same procedure to show the performance of batch effect evaluation and correction of BatchServer. The results are shown in **Figure 4.** The batch accounted for 3.02% variability in the data, while biological type captures 33.74% variability (Figure 4A column 1). After batch effect correction, biological type variability were enhanced (larger than 45%) and percent of batch effect fell to 0 (figure 4A column 2 to 4). That gives consistent results using UMAP (Figure 4BC), which shows batch effects were further reduced and ability to distinguish between tumor types were increased. This suggests that the batch effect of proteome data can be effectively adjusted by BatchServer.

CONCLUSIONS

We developed a web server called BatchServer to facilitate the evaluation, visualization, and correction of batch processing results for large-scale omics data sets. The automated ComBat can automatically select parametric or non-parametric empirical Bayes methods for batch calibration. It also integrates PVCA and UMAP to evaluate and visualize potential batch effects. BatchServer has an R/Shiny graphical user interface for enhanced usability and is easy to install on a personal computer or server, thus providing a convenient service to process batch effect for the community. It can be effectively applied to multiple omics data sets.

ASSOCIATED CONTENT

Supporting Information.

The following files are available free of charge.

The second and the third transcriptomic data sets description

Supplementary Figure 1-3. (file type docx)

Supplementary Figure 1: Performance of improved ComBat by the second transcriptomic data set.

Supplementary Figure 2: Visualization of BatchServer by the third transcriptomic data set.

Supplementary Figure 3: Prior plot of batch fitting effect by improved ComBat by the third transcriptomic data set.

Supplementary Table 1-3. (file type docx)

Supplementary Table 1: Time consumed (seconds) for batch effect adjust using ComBat in transcriptomic data set 1.

Supplementary Table 2: Time consumed (seconds) for batch effect adjust using improved ComBat in transcriptomic data set two.

Supplementary Table 3: Time consumed (seconds) for batch effect adjust using ComBat in transcriptomic data set 3.

AVAILABILITY OF DATA AND MATERIALS

The proposed BatchServer and its manual are deployed at https://lifeinfo.shinyapps.io/batchserver/ and tested with Chrome, Firefox and IE browsers. Source codes and testing data are freely available at https://github.com/guomics-lab/batch_server with license for freely available to academic researchers.

ABBREVIATIONS

MS, mass spectrometry; NGS, next generation sequencing; PCA, principal components analysis; PVCA, Principal Variance Component Analysis; UMAP, Manifold Approximation and Projection; SVD, Singular Value Decomposition; t-SNE, t-distributed stochastic

neighbor embedding; K-S test, Kolmogorov–Smirnov test; DIA, data independent acquisition.

CONFLICT OF INTEREST

The authors declare that they have no competing interests.

ACKNOWLEDGMENTS

This work was supported by National Natural Science Foundation of China (General Program) (Grant No. 81972492 to T.G.), Zhejiang Provincial Natural Science Foundation for Distinguished Young Scholars (Grant No. LR19C050001 to T.G.), Hangzhou Agriculture and Society Advancement Program (Grant No. 20190101A04 to T.G.).

AUTHORS' CONTRIBUTIONS

TZ design and implemented the web application; TG, SZ, and TZ wrote the manuscript and conceived the project; GC, CY, RS and FZ revised and modified the manuscript and provided many useful suggestions. RS and XY conducted the software testing. All authors have read and approved the final manuscript.

REFERENCES

1. Leek JT, Scharpf RB, Bravo HC, Simcha D, Langmead B, Johnson WE, Geman D, Baggerly K, Irizarry RA: **Tackling the widespread and critical impact of batch effects in high-throughput data**. *Nat Rev Genet* 2010, **11**(10):733-739.
2. Lazar C, Meganck S, Taminau J, Steenhoff D, Coletta A, Molter C, Weiss-Solis DY, Duque R, Bersini H, Nowe A: **Batch effect removal methods for microarray gene expression data integration: a survey**. *Brief Bioinform* 2013, **14**(4):469-490.
3. Goh WWB, Wong L: **Advanced bioinformatics methods for practical applications in proteomics**. *Brief Bioinform* 2019, **20**(1):347-355.
4. Chen C, Grennan K, Badner J, Zhang D, Gershon E, Jin L, Liu C: **Removing batch effects in analysis of expression microarray data: an evaluation of six batch adjustment methods**. *PLoS One* 2011, **6**(2):e17238.
5. Boedigheimer MJ, Wolfinger RD, Bass MB, Bushel PR, Chou JW, Cooper M, Corton JC, Fostel J, Hester S, Lee JS *et al*: **Sources of variation in baseline gene expression levels from toxicogenomics study control animals across multiple laboratories**. *Bmc Genomics* 2008, **9**:285.
6. Becht E, McInnes L, Healy J, Dutertre CA, Kwok IWH, Ng LG, Ginhoux F, Newell EW: **Dimensionality reduction for visualizing single-cell data using UMAP**. *Nat Biotechnol* 2018, **37**(1):38-44.
7. Wold S: **Principal component analysis**. 1987, **2**(1):37-52.

8. Alter O, Brown PO, Botstein D: **Singular value decomposition for genome-wide expression data processing and modeling**. *Proc Natl Acad Sci U S A* 2000, **97**(18):10101-10106.
9. Leek JT, Storey JD: **Capturing heterogeneity in gene expression studies by surrogate variable analysis**. *PLoS Genet* 2007, **3**(9):1724-1735.
10. Nyamundanda G, Poudel P, Patil Y, Sadanandam A: **A Novel Statistical Method to Diagnose, Quantify and Correct Batch Effects in Genomic Studies**. *Sci Rep* 2017, **7**(1):10849.
11. Papiez A, Marczyk M, Polanska J, Polanski A: **BatchI: Batch effect Identification in high-throughput screening data using a dynamic programming algorithm**. *Bioinformatics* 2019, **35**(11):1885-1892.
12. Johnson WE, Li C, Rabinovic A: **Adjusting batch effects in microarray expression data using empirical Bayes methods**. *Biostatistics* 2007, **8**(1):118-127.
13. Goh WWB, Wang W, Wong L: **Why Batch Effects Matter in Omics Data, and How to Avoid Them**. *Trends Biotechnol* 2017, **35**(6):498-507.
14. McGurk KA, Dagliati A, Chiasserini D, Lee D, Plant D, Baricevic-Jones I, Kelsall J, Eineman R, Reed R, Geary B *et al*: **The use of missing values in proteomic data-independent acquisition mass spectrometry to enable disease activity discrimination**. *Bioinformatics* 2019.
15. Wei R, Wang J, Su M, Jia E, Chen S, Chen T, Ni Y: **Missing Value Imputation Approach for Mass Spectrometry-based Metabolomics Data**. *Sci Rep* 2018, **8**(1):663.
16. MA S: **Tests Based on EDF Statistics in Goodness-of-Fit Techniques**. *Marcel Dekker* 1986:97-194.
17. Guo T, Luna A, Rajapakse VN, Koh CC, Wu Z, Liu W, Sun Y, Gao H, Menden MP, Xu C *et al*: **Quantitative Proteome Landscape of the NCI-60 Cancer Cell Lines**. *iScience* 2019, **21**:664-680.
18. Guo T, Kouvonen P, Koh CC, Gillet LC, Wolski WE, Rost HL, Rosenberger G, Collins BC, Blum LC, Gillessen S *et al*: **Rapid mass spectrometric conversion of tissue biopsy samples into permanent quantitative digital proteome maps**. *Nat Med* 2015, **21**(4):407-413.

LEGENDS OF FIGURES

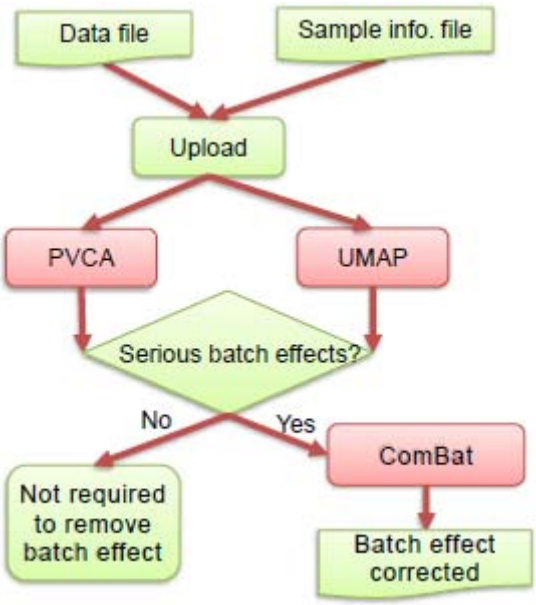
Figure 1. The architecture of BatchServer. Users can submit data and sample information files via the data input layer. Once submitted, files are processed by the data process layer. Thereafter, the user is required to specify potential batch effect columns and interactively set variables, if any. The data processing layer then calculates and returns the results. Finally, the data output layer will display the results interactively through figures and tables.

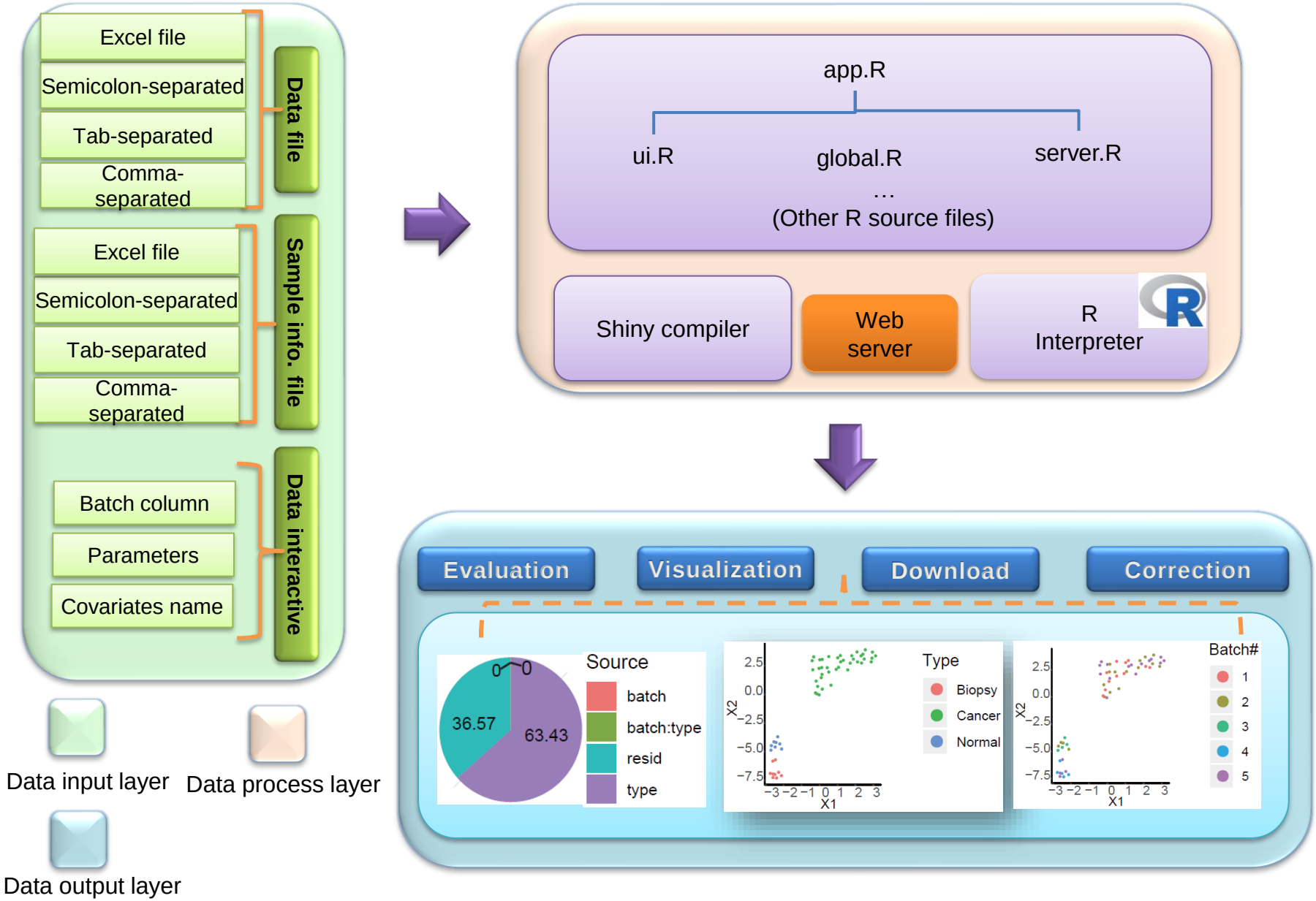
Figure 2. The screenshot of prior plot of γ and δ of (A) batch '1' which passed the K-S test and (B) batch '4' which didn't pass the K-S test using density plot and normal Q-Q plot of BatchServer. Red and blue lines/points indicate parametric and empirical estimation, respectively. The first 50 rows of bladder data in bladderbatch Bioconductor package were used as input data.

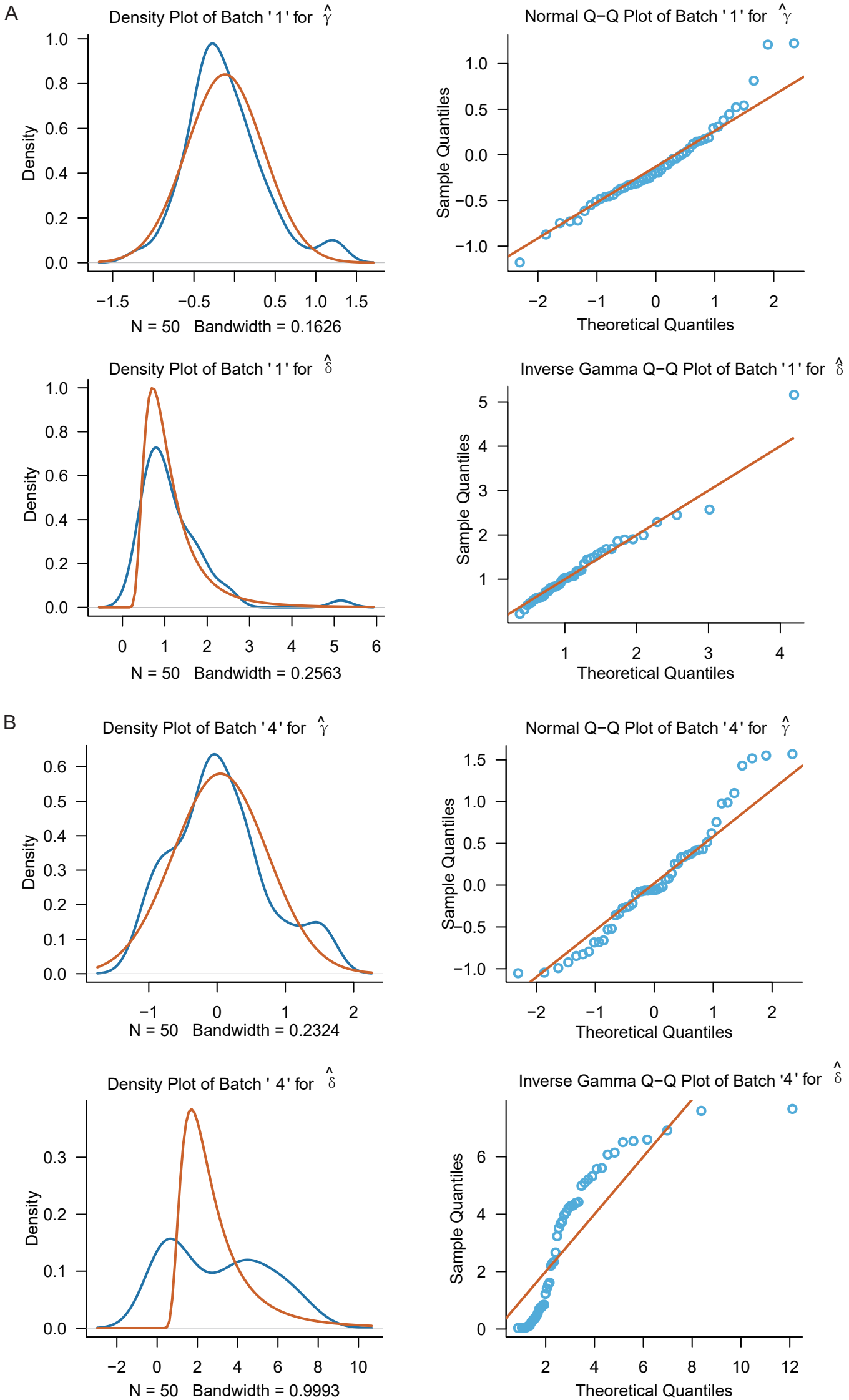
Figure 3. Improved performance of ComBat by the first transcriptomics data set. (A) Pie plots of batch effect using PVCA with no correction and par.prior set to nonparametric, parametric or auto for autoComBat. The number in each pie chart is the percentage of the source of variation corresponding to the respective color. 'type' means the variation caused by the tumor type. 'batch' means the variation caused by the batch number. 'batch:type' means interaction effects and 'resid' represent residual variance (unexplained variance by the chosen effects in the model, which may be caused by individual heterogeneous and other factors). (B) UMAP plots show the clustering of (B) biological interesting type and (C) batch# before and after batch effect correction by parametric ComBat, non-parametric ComBat and autoComBat, respectively. (D) Priori plots of batch effect by autoComBat.

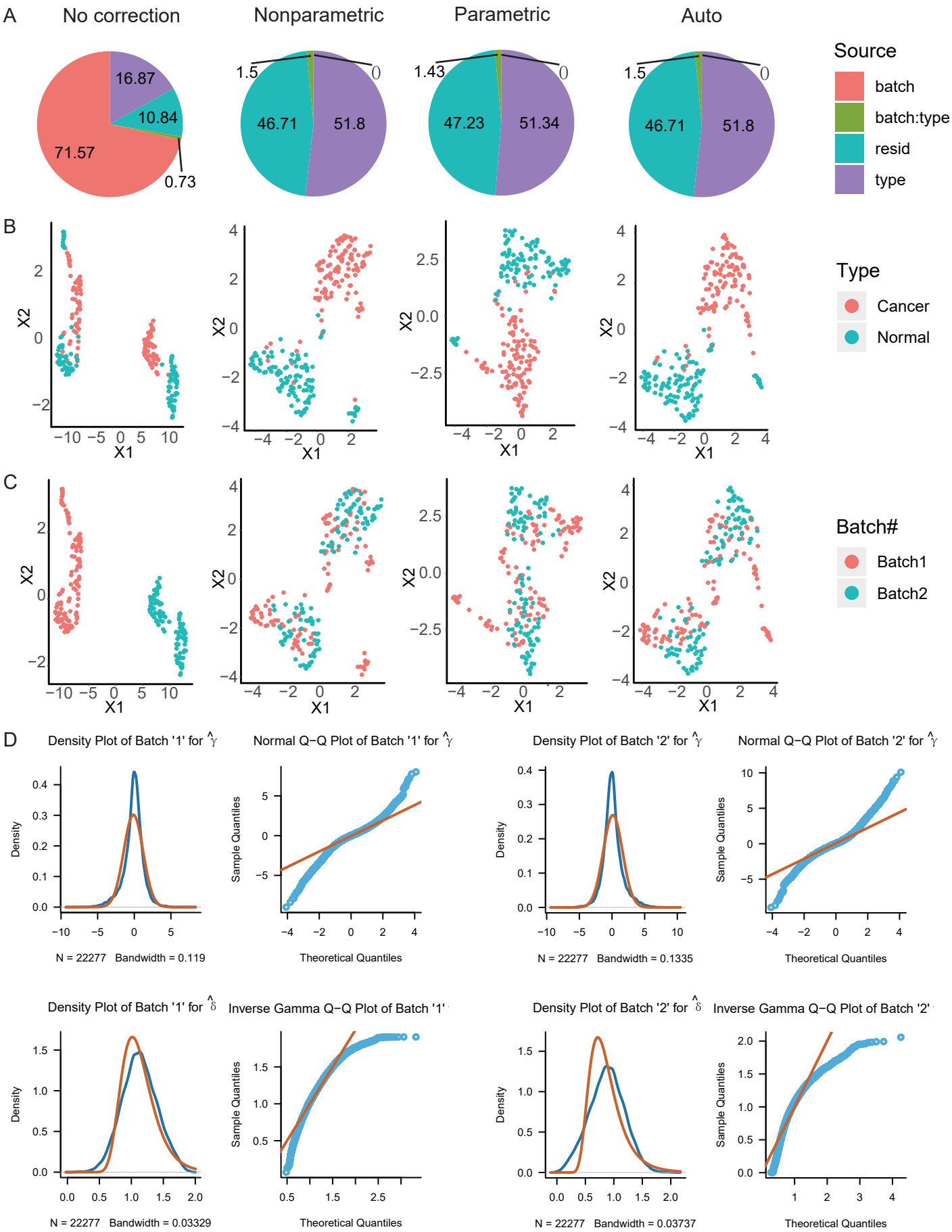
Figure 4. Performance of BatchServer by proteomics data set. The four columns are, in order, original data with no correction, corrected data with 'nonparametric', corrected data with 'parametric' and correct data with 'automatic' by autoCombat. (A) Pie plots of batch effect using PVCA. (B) UMAP visualization of the clustering of biological effect enhanced after batch effect correction. (C) UMAP show the degree of accumulation and dispersion of batches before and after batch effect correction.

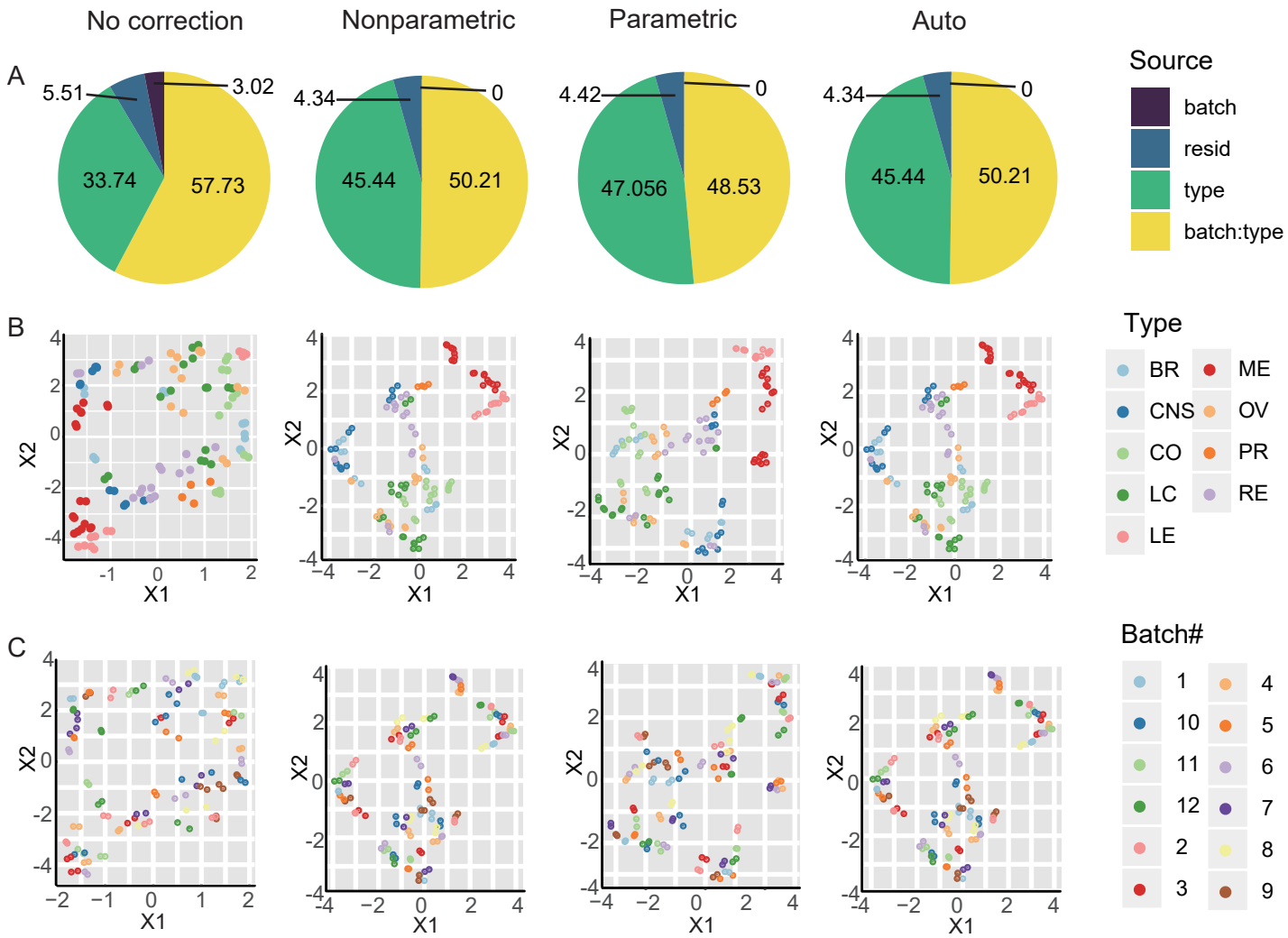
For TOC only


Data file
Sample info. file
Upload
PVCA
UMAP
Serious batch effects?
No
Yes
ComBat
Not required to remove batch effect
Batch effect corrected


Excel file
Semicolon-separated
Tab-separated
Comma-separated
Data file
Excel file
Semicolon-separated
Tab-separated
Comma-separated
Sample info. file
Batch column
Parameters
Covariates name
Data interactive
app.R
ui.R
global.R
server.R
…
(Other R source files)
Shiny compiler
Web server
R Interpreter
Evaluation
Visualization
Download
Correction
Source
batch
batch:type
resid
type
36.57
63.43
0
0
Type
Biopsy
Cancer
Normal
Batch#
1
2
3
4
5
X1
X2
Data input layer
Data process layer
Data output layer


A
Density Plot of Batch '1' for $\hat{\gamma}$
Density
N = 50 Bandwidth = 0.1626
Normal Q−Q Plot of Batch '1' for $\hat{\gamma}$
Sample Quantiles
Theoretical Quantiles
Density Plot of Batch '1' for $\hat{\delta}$
Density
N = 50 Bandwidth = 0.2563
Inverse Gamma Q−Q Plot of Batch '1' for $\hat{\delta}$
Sample Quantiles
Theoretical Quantiles
B
Density Plot of Batch '4' for $\hat{\gamma}$
Density
N = 50 Bandwidth = 0.2324
Normal Q−Q Plot of Batch '4' for $\hat{\gamma}$
Sample Quantiles
Theoretical Quantiles
Density Plot of Batch ' 4' for $\hat{\delta}$
Density
N = 50 Bandwidth = 0.9993
Inverse Gamma Q−Q Plot of Batch '4' for $\hat{\delta}$
Sample Quantiles
Theoretical Quantiles


A
No correction
Nonparametric
Parametric
Auto
16.87
10.84
71.57
0.73
1.5
0
46.71
51.8
1.43
0
47.23
51.34
1.5
0
46.71
51.8
Source
batch
batch:type
resid
type
B
X1
X2
Type
Cancer
Normal
C
X1
X2
Batch#
Batch1
Batch2
D
Density Plot of Batch '1' for γ̂
Normal Q−Q Plot of Batch '1' for γ̂
Density Plot of Batch '2' for γ̂
Normal Q−Q Plot of Batch '2' for γ̂
Density
Sample Quantiles
Theoretical Quantiles
N = 22277 Bandwidth = 0.119
N = 22277 Bandwidth = 0.1335
Density Plot of Batch '1' for δ̂
Inverse Gamma Q−Q Plot of Batch '1'
Density Plot of Batch '2' for δ̂
Inverse Gamma Q−Q Plot of Batch '2'
N = 22277 Bandwidth = 0.03329
N = 22277 Bandwidth = 0.03737


No correction
Nonparametric
Parametric
Auto
A
5.51
3.02
33.74
57.73
4.34
0
45.44
50.21
4.42
0
47.056
48.53
4.34
0
45.44
50.21
Source
batch
resid
type
batch:type
B
X1
X2
Type
BR
CNS
CO
LC
LE
ME
OV
PR
RE
C
Batch#
1
10
11
12
2
3
4
5
6
7
8
9