Ugidos, M.; Tarazona Campos, S.; Prats-Montalbán, JM.; Ferrer, A.; Conesa, A. (2020). MultiBaC: A strategy to remove batch effects between different omic data types. Statistical Methods in Medical Research. 29(10):2851-2864. https://doi.org/10.1177/0962280220907365

MultiBaC: a strategy to remove batch effects between different omic data types

Statistical Methods in Medical Research
XX(X):2–17

DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

Manuel Ugidos[1,*], Sonia Tarazona[2,*], José M. Prats-Montalbán[2], Alberto Ferrer[2] and Ana Conesa[3]

Abstract

Diversity of omic technologies has expanded in the last years together with the number of omic data integration strategies. However, multiomic data generation is costly and many research groups cannot afford research projects where many different omic techniques are generated, at least at the same time. As most researchers share their data in public repositories, different omic datasets of the same biological system obtained at different labs can be combined to construct a multiomic study. However, data obtained at different labs or moments in time are typically subjected to batch effects that need to be removed for successful data integration. While there are methods to correct batch effects on the same data types obtained in different studies, they cannot be applied to correct lab or batch effects across omics. This impairs multiomic meta-analysis. Fortunately, in many cases, at least one omics platform –i.e. gene expression- is repeatedly measured across labs, together with the additional omic modalities that are specific to each study. This creates an opportunity for batch analysis. We have developed MultiBaC, a strategy to correct batch effects from multiomic datasets distributed across different labs or data acquisition events. Our strategy is based on the existence of at least one shared data type which allows data prediction across omics. We validate this approach both on simulated data and on a case where the multiomic design is fully shared by two labs, hence batch effect correction within the same omic modality using traditional methods can be compared with the MultiBaC correction across data types. Finally we apply MultiBac to a true multiomic data integration problem to show that we are able to improve the detection of meaningful biological effects.

Keywords

Batch effect correction, Multiomic integration, multivariate methods, biostatistics

Introduction

Over the last decade, high-throughput omic technologies such as transcriptomics, metabolomics, proteomics or epigenomics have become routine assays in many biological research laboratories. Increasingly, combinations of these methods are proposed to address complex questions about the molecular regulation of genomes and the physiology of cellular systems. As different omic assays target different biomolecules or chemical modifications, the combined study of these various molecular layers has the potential to provide insights into the complex regulatory networks that operate in living cells. However, simultaneously generating multiple omic measurements of the same molecular system for one particular study might be difficult. Challenges arise due to budgetary restrictions, time and sample limitations, or simply because of the convenience of a sequential analysis of the data in order to make informed decisions for follow up experiments. At the same time, researchers are not longer restricted to their own experimental capacities in order to obtain multiomic information, as facilities offer these assays on a commercial basis. Widespread editorial policies requiring omic data deposition in public repositories before publication of results have created a wealth of molecular data available to researchers for reuse. As a consequence, scientists have the opportunity to combine compatible data generated in other labs to compose a suitable multiomic dataset without the need of repeating experiments already performed by somebody else. Unfortunately, combining data obtained by different people and/or at different moments in time has an important drawback. Data will almost unavoidably be affected by technical biases associated to the experimentation event that, especially for high throughput molecular assays, may result in important levels of noise contaminating the biological signal. This unwanted source of variation is commonly known as “batch effect” and is very frequently seen as the first component of variability in the omic dataset, standing out over the experimental conditions under of study.

[1] Gene expression and RNA Metabolism Laboratory, Instituto de Biomedicina de Valencia, Consejo Superior de Investigaciones Científicas (CSIC), Valencia, Spain
[2] Multivariate Statistical Engineering Group, Department of Applied Statistics, Operations Research and Quality, Universitat Politècnica de València, Valencia, Spain
[3] Microbiology and Cell Science Department, Institute of Food and Agricultural Sciences, University of Florida, Gainesville, FL, United States
[*] These authors contributed equally to this work

Corresponding author:
Ana Conesa
Email: aconesa@ufl.edu

Batch effects significantly impair the power of statistical algorithms to detect significant true effects as they increase measurement errors and data variability. Removing batch effects becomes then necessary in order to obtain meaningful results from statistical analyses (1; 2). Provided that the omic experiment has been designed in such a way that batch effects are not confounded with the effects of interest (e.g. treatment, disease, cell type, etc.), the so-called Batch Effect Correction Algorithms (BECAs) can be used to remove, or at least mitigate, systematic biases. Therefore these methods are extremely useful to combine data from different laboratories or measured at different times.

Several BECAs for omic data have been proposed. Limma (3) applies linear models while the ComBat method (4) from sva R package (5) estimates batch effects as the sum of an additive and a multiplicative effect with an empirical Bayes approach. RUV (6) estimates the unwanted variation from negative control genes that are known a priori to be unaffected by the biological factor of interest. We proposed the ARSyN approach (7), that relies on the ANOVA-Simultaneous Components Analysis (ASCA) framework (8; 9) to decompose the omic signal into experimental effects, the batch effect and residuals. ARSyN applies Principal Component Analysis (PCA) to estimate the systematic variation due to batch effect and then removes it from the original data.

These methods have been traditionally applied to remove batch effects from omic data of the same type, as for example gene expression, and have been instrumental for the combination of data from the public domain into meta-analyses to reveal novel biological insights that cannot be discovered with small sample sizes (10; 11; 12; 13; 14). However, while removing batch effects from a single omic data type with an appropriate experimental design is relatively straightforward, it can become unapproachable when dealing with multiomic datasets. In the multiomic scenario, each omic modality may be measured by a different lab or at a different moment in time, and so it is obtained within a different batch. When this is the case, the batch effect will be confounded with the "omic type effect" and impossible to remove from the data. However, in some scenarios, the multiomic batch effect can be corrected.

In this work, we present the novel MultiBaC method, which is the first BECA dealing with batch effect correction in multiomic datasets. MultiBaC is able to remove batch effects across different omics generated within separate batches provided that at least one common omic data type is included in all the batches considered. Although this may seem a strong requirement, in practice there are many studies that include at least gene expression or popular histone marks as part of their multiomic design and hence provide opportunities for data combination across omic modalities. For example, stress response in yeast has been studied at the transcriptional rate (15; 16; 17), translational rate (18) and RNA-binding of global proteins (19), in three different studies that also included RNA-seq profiling. A method that corrects batch effects across omics will allow for the integration of these data in one single analysis that jointly evaluates different layers of transcriptional regulation by leveraging public resources and without the need of generating additional data. In this work we demonstrate that MultiBaC is effective in removing

batch effects without introducing additional biases and that outperforms adaptation of existing strategies to the multiomic batch problem. MultiBaC is therefore an effective tool to reuse existing datasets to perform meta-analysis across omics technologies.

Data

A yeast multiomic dataset obtained at different laboratories

We collected data from Gene Expression Omnibus (GEO) database pertaining to three different studies that analyzed the effects of glucose starvation in yeast. These studies used equivalent yeast strains and experimental conditions, but differed in the types of omic technologies profiled. Lab A (Department of Biochemistry and Molecular Biology, Universitat de València) collected gene expression (RNA, with accession number GSE11521) and transcription rates (GRO, with accession number GSE1002) (15; 16; 17). Lab B (Department of Molecular and Cellular Biology, Harvard University) obtained gene expression (RNA) and translation rates (RIBO), with accession number GSE56622 (18). Finally, Lab C (Department of Biology, Johns Hopkins University) measured gene expression (RNA) and global PAR-CLIP data (PAR-CLIP) with accession number GSE43747 (19). Therefore, labs had one shared (RNA) and one distinct (GRO, RIBO and PAR-CLIP, respectively) data types. This distributed multiomic scenario represents the type of correction problem MultiBaC addresses.

Simulated data

A synthetic multiomic dataset was created that reproduces the scenario described in the yeast example. In this case, we simulated three different omic data types from two labs, one of them being the common data type. Each omic data matrix was generated with the MOSim multiomic simulation tool (20). As MOSim does not model batch effects, we analyzed several yeast experimental datasets (15; 16; 17; 18; 19; 21; 22) to estimate the magnitude of a reasonable batch effect by fitting a linear model that included the batch effect and the interaction between treatment and batch. We observed that the coefficients of the model follow a normal distribution with mean equal to zero. Hence, we simulated different datasets with varying magnitudes of batch effects, by adding to the MOSim simulated data batch effects generated from a normal distribution with increasing values of the starndard deviation values. We modeled three batch effect levels: low, moderate and high, being magnitudes low and moderate present in real experimental data, while the high level was an extreme scenario never observed in the evaluated datasets. A detailed description of batch effect simulation can be found in Supplementary Materials 1.

Proof of concept data

We validated MultiBaC on two multiomic datasets that shared all omics modalities (GEO accession numbers GSE33136 (21) and GSE24488 (22)). In both GEO studies transcription rates and gene expression data were available and the experimental conditions compared were room temperature versus heat-shock stress in yeast. We denote these datasets "proof of concept" data because both omic data types are available from both studies and, hence, traditional BECAs for a single omic can be applied and compared to MultiBaC correction. Each of the two laboratories considered applied a different technology to obtain omic measurements: study 1 (GSE33136) used microarrays while study 2 (GSE24488) used sequencing techniques. In order to make both datasets comparable, GSE24488 data were normalized using voom() transformation from limma R package (3).

Methods

ARSyN method

ARSyN (ASCA Removal of Systematic Noise) was presented by Nueda et al. (7) and is a batch effect correction approach that relies on the ANOVA-Simultaneous Component Analysis (ASCA) framework. Let x_{ijr} be the gene expression of gene x, measured at time i, under treatment j and for replicate r. Therefore, x_{ijr} can be decomposed as in any ANOVA model as:

$$x_{ijr} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + (\alpha\beta\gamma)_{ijr} \tag{1}$$

where μ is an offset term, α_i the treatment, β_j the batch effect, $(\alpha\beta)_{ij}$ the interaction effect between batch and treatment, and $(\alpha\beta\gamma)_{ijr}$ the individual variation (residuals). If our omic data matrix X contains N genes in columns and M samples in rows, the previous equation can be expressed using matrix notation as:

$$\mathbf{X} = 1m^t + \mathbf{X_a} + \mathbf{X_b} + \mathbf{X_{ab}} + \mathbf{X_{abg}} \tag{2}$$

where m is an N size vector containing the estimations of μ for each gene, matrices $\mathbf{X_a}$, $\mathbf{X_b}$ and $\mathbf{X_{ab}}$ contain the estimations of parameters α_i, β_j and $(\alpha\beta)_{ij}$ respectively, and $\mathbf{X_{abg}}$ contains the residuals $(\alpha\beta\gamma)_{ijr}$. Once this ANOVA-like decomposition is obtained, a PCA is applied on each submatrix, the number of principal components is determined for each case, and the resulting ASCA model is:

$$\mathbf{X} = 1m^t + \overbrace{\mathbf{T_a}\mathbf{P_a^t} + \mathbf{E_a}}^{\mathbf{X_a}} + \overbrace{\mathbf{T_b}\mathbf{P_b^t} + \mathbf{E_b}}^{\mathbf{X_b}} + \overbrace{\mathbf{T_{ab}}\mathbf{P_{ab}^t} + \mathbf{E_{ab}}}^{\mathbf{X_{ab}}} + \overbrace{\mathbf{T_{abg}}\mathbf{P_{abg}^t} + \mathbf{E_{abg}}}^{\mathbf{X_{abg}}} \tag{3}$$

where $\mathbf{T_i}$ and $\mathbf{P_i}$ are the scores and loadings matrices from the PCA on each matrix $\mathbf{X_i}$, respectively. After estimating the effects with ASCA, ARSyN corrects the batch effect by subtracting undesirable effects from original data according to the following equation:

$$\mathbf{X}^* = \mathbf{X} - \overbrace{(\mathbf{T_b}\mathbf{P_b^t} + \mathbf{T_{ab}}\mathbf{P_{ab}^t})}^{\text{Batch and interaction effects}} \tag{4}$$

where $\mathbf{X}^*$ is the corrected matrix without batch or interaction batch-treatment effects.

MultiBaC: A multiomic batch effect correction strategy

MultiBaC (**Multi**omic **Ba**tch **C**orrection) method was conceived to correct batch effects across different omic data types provided that at least one omic modality is repeated in the batches. For the sake of simplicity in the formulation of the MultiBac method we consider that batch effect arises from different labs generating data, although the method is generally applicable to any other batch sources such as time or lab technician. Let us consider a minimal size problem example with two labs, each one of them measuring two different omic data types, one of them in common (Figure 1(a)). We denote $\mathbf{X_1}$ as the common data type from lab 1, $\mathbf{X_2}$ as the common data type from lab 2, $\mathbf{K}$ as the non-common data type from lab 1 and $\mathbf{Z}$ as the non-common data type from lab 2. One important feature of MultiBaC is that the different omics studied in each lab do not have to share the variable space. This allows to combine gene-related omics (e.g. RNA-seq) with other technologies such as proteomics or metabolomics. However, within each lab the same samples should have been measured with the different omic technologies, hence the number of samples must be the same for all the omics.

MultiBaC assumes that there exists a relationship between two different omic data types that does not depend on the laboratory. Basically, MultiBaC applies a multivariate PLS regression (23) to model the non-common omic data matrix as a function of the common omic measurements. The models are then used to predict the missing measurements what results in complete multiomic datasets in all laboratories. Next, traditional BECA methods are applied to correct the batch effect from the original matrices. MultiBaC proceeds through three steps (Figure 1(b)):

In the *Modelling step*, PLS models are built for each lab, where the common omic data type is used as the explanatory matrix $\mathbf{X}$ and the non-common omic is used as the response matrix $\mathbf{Y}$. The PLS model can be expressed as $\mathbf{Y} = \mathbf{XB} + \mathbf{E}$, where $\mathbf{B}$ is the regression coefficient matrix and $\mathbf{E}$ is the residuals matrix. $\mathbf{B}$ can be estimated as:

$$\mathbf{B} = \mathbf{W}^*\mathbf{C}^\mathbf{T} = \mathbf{W}(\mathbf{P}^\mathbf{T}\mathbf{W})^{-\mathbf{1}}\mathbf{C}^\mathbf{T} \tag{5}$$

where $\mathbf{W}$ is the $\mathbf{X}$-weight matrix, $\mathbf{P}$ is the $\mathbf{X}$-loading matrix and $\mathbf{C}$ is the $\mathbf{Y}$-weight matrix.

Therefore, considering a PLS model for each lab, for our minimal size problem, we will have the following PLS models:

$$PLS_1 : \mathbf{K} = \mathbf{B_1}\mathbf{X_1} + \mathbf{E_1}$$

$$PLS_2 : \mathbf{Z} = \mathbf{B_2}\mathbf{X_2} + \mathbf{E_2}$$

Q^2-based cross-validation (CV) optimization, proposed by Tenenhaus (24), is applied to select the optimal number of components for the PLS models, since Q^2 measures the marginal contribution of each component to the predictive power of the model. A good Q^2 value > 0.7 is required to ensure that the model has a good prediction performance and can be used to infer the missing data modality. In the *Prediction step*, MultiBaC will estimate the missing omic data type for each lab by using the previously obtained PLS coefficient matrices:

$$\hat{\mathbf{Z}}_\mathbf{1} = \mathbf{X_1}\mathbf{B_2}$$

$$\hat{\mathbf{K}}_\mathbf{2} = \mathbf{X_2}\mathbf{B_1}$$

Note that, for predicting $\hat{\mathbf{Z}}_\mathbf{1}$, the coefficient matrix relating $\mathbf{X_2}$ and $\mathbf{Z}$ is used, that is, $\mathbf{B_2}$. And the procedure is analogous for $\hat{\mathbf{K}}_\mathbf{2}$. Remember that we aim to predict the omic information that was not initially available for each lab. This will allow us to remove the batch effect on non-common information with traditional methods using the original and the predicted information, i.e., $\mathbf{K}$ and $\hat{\mathbf{K}}_\mathbf{2}$ for instance.

Finally, in the *Correction step* MultiBaC applies ARSyN to remove batch effect from every omic data type. Available data are used for the common omic, while predicted data must be used for the rest of omics.

$$\mathbf{X}^* = ARSyN(\mathbf{X_1},\ \mathbf{X_2})$$

$$\mathbf{K}^* = ARSyN(\mathbf{K},\ \hat{\mathbf{K}}_\mathbf{2})$$

$$\mathbf{Z}^* = ARSyN(\hat{\mathbf{Z}}_\mathbf{1},\ \mathbf{Z})$$

where * means corrected matrix. Typically, we will discard now the predicted and corrected omic matrices $\hat{\mathbf{K}}_\mathbf{2}^*$ and $\hat{\mathbf{Z}}_\mathbf{1}^*$, and use the original and corrected matrices, $\mathbf{K}^*$ and $\mathbf{Z}^*$, for further statistical analyses.

[insert figure 1]

Other multiomic batch effect correction approaches

In addition to MultiBaC strategy, we also adapted two other existing and conceptually different methodologies that theoretically could be applicable for solving the multiomic batch effect problem. These strategies were compared with the MultiBaC method.

Missing data imputation strategy: In this approach, the values for the non-common omic data types are considered missing values for the laboratories where these omic data types are not available. Imputation of missing values is carried on with the multivariate method **T**rimmed **S**cores **R**egression (TSR) (25), and then a BECA is applied (e.g. ARSyN). TSR models the structure in Figure 2(a) containing missing values (NA) as a unique matrix, and employs the latent space of the whole matrix to impute missing data according to the relation between observed variables in each batch by using the common information as an inner reference.

Product transfer model: The Joint-Y PLS (JY-PLS) methodology presented by García Muñoz et al. (26) is based on PLS regression and assumes that both PLS response matrices ($\mathbf{X_1}$ and $\mathbf{X_2}$ in Figure 2(b)) share the same latent structure. Note that response matrices in this model are $\mathbf{X_1}$ and $\mathbf{X_2}$ (the common data type). Basically, JY-PLS builds a PLS model between $\mathbf{K}$ and $\mathbf{X_1}$ and another PLS model between $\mathbf{Z}$ and $\mathbf{X_2}$ by forcing $\mathbf{X_1}$ and $\mathbf{X_2}$ to share the same weight matrix (Q^T), i.e the same latent space. The ARSyN batch effect corrected common data type ($\mathbf{X}^*$) is used for the JY-PLS inversion step in order to obtain $\mathbf{K}^*$ and $\mathbf{Z}^*$, that is, the non-common batch effect corrected matrices. In brief, the inversion step tries to transfer a new set of responses which are the corrected data (e.g $\mathbf{X_1^*}$), in order to obtain which observations of the non-common omic could be in agreement with that set of responses (i.e $\mathbf{K}^*$).

[insert figure 2]

Validation strategies

Latent space concordance. This validation strategy was used to assess the performance of the methods on simulated data by evaluating if original data (before batch effect addition) and batch effect corrected data share the latent space in a PCA model. Latent space concordance (R^2) measures how well the variability structure of originally simulated matrices is able to explain the variability of corrected matrices and the higher the R^2 the better the concordance. We applied latent structure concordance by estimating a PCA model with the original data and computing R^2 score for the corrected data after projection onto that PCA model. In order to remove rotation effect differences, which could decrease the R^2 score, the PROCRUSTES algorithm (27; 28) was applied in this step.

Differential expression analysis. Assuming that batch effects impair combination of different experiments but do not affect the inner information structure of one experiment, we consider that

differential expression (DE) analysis applied to individual omic matrices should give the same or very similar results before and after batch effect correction. In order to assess the concordance of such DE results, we considered the original data as the true values and the corrected data as the predicted values, and we used three different scores based on the number of DE genes obtained from each dataset: False Discovery Rate (FDR), Sensitivity (SE) and Specificity (SP). FDR measures the false positive rate, i.e. the percentage of genes declared as DE after correction but non-DE in the original matrix. SE assesses the ability to detect, after correction, all the DE genes obtained from the original data. Finally, SP appraises the ability to detect, after correction, all the initially non-DE genes. Differential expression analysis were performed using limma R package (3).

Results and Discussion

Simulated data

Simulated datasets were used to test the performance of MultiBac method at removing batch effects and preserving the structure of the original data. We studied the latent space concordance between original (without batch effects) and MultiBaC-corrected data at increasing magnitudes of batch effect and interaction values between batch and experimental conditions (Figure 3(a), upper left panel). R^2 was high (> 0.7) and very similar for the three tested methods at all batch magnitudes except for the highest values. Moreover, the intensity of the batch-condition interaction had little effect on the R^2 values. These results indicate that tested batch correction methods successfully recovered the latent structure of the unbiased data when batch effects were within limits observed in real datasets.

Next, we compared the differential expression analysis results on the original simulated data and on the batch effect corrected data to evaluate if detection of differentially expressed features was maintained after the batch effect correction. Taking the differential expression results from data without batch effect as the true reference, we computed the False Discovery Rate (FDR), Sensitivity (SE) and Specificity (SP) (Figure 3(a)). The performance of the compared methods regarding these three indicators was greatly affected by the magnitude of the interaction effect between the batch and the experimental condition, while the batch effect magnitude did not seem to have an important effect. FDR (bottom-left plot) is lower for MultiBaC than for the other two methods in all cases. In general, this indicator varies from 0 to 20%, while it reaches more than 50% in some cases for TSR or JY-PLS. In addition, MultiBaC FDR was less affected by the effect of the interaction when compared to the other methods. The increase in false positives caused SP rate (bottom-right plot) to generally decrease at high interaction magnitudes, but JY-PLS and MultiBaC performances were very similar, with scores above 80% in all cases, including at high interaction levels. Regarding SE results (top-right plot), MultiBaC was once again the best method, with SE above 95% in all simulations. This means that MultiBaC recovers all the originally

differentially expressed genes, regardless the magnitude of the interaction effect. Altogether we conclude that MultiBaC outperforms compared methods and results in batch corrected data where no apparent additional biases have been introduced.

[insert figure 3]

Proof of concept data

We further validated MultiBaC with proof of concept data, where the same two omics (gene expression and transcriptional rates) had been measured by two different laboratories. Consequently, traditional BECAs can be applied on each omic data type to remove the laboratory effect and compared to the across-data types batch effect correction by MultiBaC. ARSyN was used as BECA method. For MultiBaC, we assumed that gene expression was the common omic and transcriptional rates were non-common between labs. We evaluated results by comparing PCA plots from original data and both ARSyN and MultiBaC corrected data (Figure 3(b)). As expected the PCA of the original data showed a strong effect of the laboratory that was captured by the first principal component (PC). However, after ARSyN correction, the first PC separated between the experimental conditions while the second PC discriminated the omic data types, indicating that batch effects had been efficiently removed. MultiBaC correction results also showed that the first principal component was related to the experimental condition, as desired. A small residual batch effect was noticeable at the second PC for the control condition of gene expression but the strongest effect was related to the omic data type, similarly to ARSyN correction. High batch and interaction effect magnitudes have been almost totally removed from transcription rate data, indicating that in scenarios with realistic batch and interaction effects MultiBaC provides excellent results. Therefore, this example illustrates that MultiBaC performance on experimental data is equivalent to established BECAs with the advantage that MultiBaC can be applied when specific omic data types are not included in all batches.

MultiBaC application to a real problem

Lastly, we applied MultiBaC to the real distributed multiomic dataset, with three labs having gene expression (RNA) as common omic data type and a second omic assay as non-common (namely GRO, RIBO and PAR-CLIP). These data showed a pronounced batch effect (Figure 4(a)) that stood out above omic methodology and experimental condition. MultiBac was successful at correcting these biases (Figure 4(a)). After correction, PCA clustered samples by omic type rather than by laboratory and, within each technology, separation of samples from the two experimental conditions was observed (Figure 4(b)), suggesting that technical noise was removed to reveal biological information. We further evaluated that MultiBac preserved the biological information between experimental conditions by comparing differential expression calls between corrected and non-corrected data (Table 1), as well as the number

of common genes in both analyses. We computed FDR, SE and SP by taking the original data as the true reference. Although original data do not represent a real true reference without batch effect as happened in simulated data, these results are still useful to compare the effect of MultiBaC correction with ARSyN performance (only applied on RNA data) in terms of differential expression results.

Table 1. Differential expression results for the yeast multiomic dataset obtained at different labs. First column (Original) contains the number of differentially expressed genes (DEG) for each omic computed from original data. Second column (Corrected) contains the same results but computed from corrected data. Third column (Common) displays the number of DEG that are common to both analyses. FDR, SE and SP (columns 4-6) were calculated in percentage by assuming original results as true. Differential expression for omics with the symbol $*$ was computed without adjusting p-values. Last row (TOTAL) shows the number of DEG obtained in at least one omic.

	Original	Corrected	Common	FDR	SP	SE
	n^o of genes			%		
GRO	3075	2616	2615	0.038	99.950	85.041
RNA	2440	2487	2440	1.889	98.253	1
RIBO*	109	87	87	0	1	79.817
PAR*	653	607	601	0.988	99.089	92.037
TOTAL (unique)	4135	4445	3906			

The sensitivity to detect true positives (SE) was high, around 80% in the worst case (RIBO-seq), while the specificity exceeded 98% in all cases and FDR was always below 2%. RNA measurements can be considered as a control since the correction was made with the ARSyN method. In this case, a small increase in RNA number of DEGs revealed that correction slightly affected differential expression results, even when traditional BECAs and MultiBaC were applied. This is expected as the removal of batch effects reduces the variability within experimental conditions and hence improves the differential expression results. Even so, most DEGs were recovered after correction and we can state that MultiBaC preserves most of the biological information in the original data, as happens with any other traditional BECA.

Genes declared as differentially expressed in at least one of the omics (4135 for the original set and 4445 for the corrected set) were selected for clustering analysis in order to check if gene profiles across omics and conditions changed after correction. K-means algorithm (29; 30) was applied for clustering analysis and each cluster was labeled by its pattern of change (Table 2) across omic data types (results in Supplementary Materials 2)

Table 2. Clusters characterization. Each cluster obtained is characterized by a differential behavior shared by all genes in that cluster. Up or down means up- or down-regulated genes in treatment condition versus control condition.

Cluster	Pattern
1	GRO down
2	GRO and RNA down
3	GRO up and PAR down
4	RNA down
5	PAR down
6	GRO up
7	RNA up
8	GRO and PAR down
9	PAR up

The number of genes in each cluster before and after MultiBaC correction is summarized in Figure 4(b), as well as the number of shared genes between pre- and post-correction clusters. The diagonal of this table reveals the number of genes whose pattern was not affected by the correction. The cells in yellow show the most important changes in trend after correction (at least 10% of the number of genes in the diagonal). These changes were, however, subtle in magnitude and after a more detailed analysis per cluster we concluded that only 42 genes inverted their trend from up to down regulation or viceversa for RNA. Among these 42 genes, 21 were classified as "become positive" (BP) genes, since they were initially down-regulated and after correction they became up-regulated. The other 21 "become negative" (BN) genes followed the opposite behavior, that is, they were initially up-regulated and after the correction they were down-regulated.

A functional enrichment analysis of these 42 genes did not return any significant result, which means that these genes are involved in many different functions but their change in trend when correcting batch effect is not related to any specific functional category. In order to further understand why these genes changed their trend, we compared their expression values to those of 100 randomly selected up-regulated genes for RNA (RG) that did not change their trend after correction (Figure 4(c)). We found that BP genes were originally up-regulated in Lab A despite of being down regulated when performing the average between labs. The same happens for BN genes, they were initially down regulated in one lab. Interestingly, for RG randomly selected genes, the mean value was the same for all labs and there was no discordant information. This result suggests that MultiBac corrects genes with a true laboratory associated bias. For other genes MultiBaC slightly modified the value of the fold-change without introducing a switch in the direction (sign) of the change.

Finally, we compared MultiBaC results with those from P. L. Nagy et al., 2003 (18) (Lab B in our example). They focused their analysis on two groups of genes: RNA & Ribosome Occupancy (RO) up-regulated (G1) and RNA up- & RO down-regulated (G2) (see Figure 4(d)). In (18), RO denotes the ratio between RIBO and RNA values, while the ratio between GRO and RNA is named as Polymerase Occupancy (PO) and we used here the same notation. Regarding RO ratios, there are no large differences between the original and the corrected state. However, the PO ratio is greater after correction. This result agrees and improves the conclusions of the cited paper, where PO values were approximately the half of RNA values. This means that MultiBaC correction improved the relationship between omics improving accuracy and in agreement with previous studies.

[insert figure 4]

Conclusion

Many methods have been proposed to efficiently remove unwanted effects from omic data, such as effects related to lab, machine, protocol, etc., which are known in general as batch effects. These approaches (BECAs) deal with just one omic data type at a time and, to the best of our knowledge, no strategy has been suggested yet for the multiomic context, where each omic may have been produced in a different lab, by a different person or at a different period. Obviously, when two different omics have been generated in two different batches, it is difficult, if not impossible, to distinguish between the effect of the batch and the effect of the omic type itself. However, it is possible to estimate the batch effect between different omics when there is at least one common omic data type in all the batches. In this work we introduce MultiBaC, a new methodology to correct batch effects when integrating multiomic datasets in this scenario. Thus, the only requisite to apply MultiBaC is that one omic data type must be shared by all the batches to allow batch effect estimation and removal.

In this work we showed the application of MultiBac to integrated different omic technologies obtained for the same biological system at different labs. However, MultiBaC could be in principle applied in other situations such as experiments where the same omic data type has been generated by two different techniques or protocols. One example could be metabolomics obtained with Gas Chromatography (GC) and High-Pressure Liquid Chromatography (HPLC), where a few metabolites are shared by both protocols but the rest of metabolites are specific of each protocol. The common metabolites would constitute the common information and MultiBaC can be applied to remove the protocol effect so both datasets can be joined in a single analysis.

To prove the ability of MultiBaC to correct batch effect, we applied the method on simulated multiomic scenarios. As there are not established multiomic batch correction methods, we adapted and applied two suitable existing algorithms (JY-PLS and TSR) and compared them to our MultiBaC

approach. The performance of MultiBac and the other methods naturally depends on the magnitude of the batch effect and on how much this effect interacts with the effect of the experimental factor of interest. MultiBaC correction worked extremely well at batch levels expected for these technologies. Batch magnitude affected the latent structure similarity between original and corrected data but it did not affect differentially expressed genes (DEG). With extreme interaction magnitudes MultiBaC performance was compromised although it was still the best approach. We concluded that our results under the moderate interaction scenario represent very well the MultiBaC performance with real interaction effects. All in all, our analyses showed a good performance of the correction methods in realistic scenarios with MultiBaC outperforming in all simulated scenarios when correcting real experimental datasets with a strong laboratory effect. In the "proof of concept" dataset, where traditional BECAs could also be applied, results obtained with ARSyN and MultiBaC were very similar according to the PCA. MultiBaC performance was slightly less powerful than ARSyN method since MultiBaC does not estimate the batch and interaction effects from the non-common omic, while ARSyN does. Thus, the estimation and correction of the unwanted variation is not the same and should be more accurate for ARSyN. Nonetheless, MultiBaC almost completely removed the batch effect. Finally, in our "real yeast multiomic dataset", differential expression together with clustering analysis proved that lab effect was removed while the effects of experimental factors were preserved in all the omics. Few genes changed their trend after correction but the comparison with previously published results showed that results after correction were more meaningful, reliable and concordant with such studies.

In conclusion, MultiBaC is effective at removing non-biological noise from multiomic data collected at different studies, and makes these datasets comparable. We anticipate MultiBac will be a useful tool for the reutilisation of existing data for multiomic integration analyses and in facilitating experimental designs that involved the generation of multiple and diverse omic assays.

Funding

This work is part of a research project that is totally funded by Conselleria d'Educació, Cultura i Esport (Generalitat Valenciana) through PROMETEO grants programme for excellence research groups.

References

[1] P. Kupfer, R. Guthke, D. Pohlers, R. Huber, D. Koczan, and R. W. Kinne, "Batch correction of microarray data substantially improves the identification of genes differentially expressed in Rheumatoid Arthritis and Osteoarthritis," *BMC Medical Genomics*, vol. 5, 2012.

[2] J. Gregori, L. Villarreal, O. Méndez, A. Sánchez, J. Baselga, and J. Villanueva, "Batch effects correction improves the sensitivity of significance tests in spectral counting-based comparative discovery

proteomics," *Journal of Proteomics*, vol. 75, no. 13, pp. 3938–3951, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1874391912002758

[3] M. E. Ritchie, B. Phipson, D. Wu, Y. Hu, C. W. Law, W. Shi, and G. K. Smyth, "{limma} powers differential expression analyses for {RNA}-sequencing and microarray studies," *Nucleic Acids Research*, vol. 43, no. 7, p. e47, 2015.

[4] C. Li, W. E. Johnson, and A. Rabinovic, "Adjusting batch effects in microarray expression data using empirical Bayes methods," *Biostatistics*, vol. 8, no. 1, pp. 118–127, 04 2006. [Online]. Available: https://dx.doi.org/10.1093/biostatistics/kxj037

[5] J. T. Leek, W. E. Johnson, H. S. Parker, E. J. Fertig, A. E. Jaffe, and J. D. Storey, *sva: Surrogate Variable Analysis*, 2016.

[6] J. A. Gagnon-Bartsch and T. P. Speed, "Using control genes to correct for unwanted variation in microarray data," *Biostatistics (Oxford, England)*, vol. 13, no. 3, pp. 539–552, jul 2012. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/22101192https://www.ncbi.nlm.nih.gov/pmc/PMC3577104/

[7] M. J. Nueda, A. Ferrer, and A. Conesa, "ARSyN: A method for the identification and removal of systematic noise in multifactorial time course microarray experiments," *Biostatistics*, vol. 13, no. 3, pp. 553–566, 2012.

[8] J. J. Jansen, H. C. J. Hoefsloot, J. van der Greef, M. E. Timmerman, J. A. Westerhuis, and A. K. Smilde, "Asca: analysis of multivariate data obtained from an experimental design," *Journal of Chemometrics*, vol. 19, no. 9, pp. 469–481. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/cem.952

[9] A. K. Smilde, A. Ferrer, A. Conesa, H. C. J. Hoefsloot, J. A. Westerhuis, M. Talón, and M. J. Nueda, "Discovering gene expression patterns in time course microarray experiments by ANOVA–SCA," *Bioinformatics*, vol. 23, no. 14, pp. 1792–1800, 05 2007. [Online]. Available: https://doi.org/10.1093/bioinformatics/btm251

[10] B. N. Keel, C. M. Zarek, J. W. Keele, L. A. Kuehn, W. M. Snelling, W. T. Oliver, H. C. Freetly, and A. K. Lindholm-Perry, "Rna-seq meta-analysis identifies genes in skeletal muscle associated with gain and intake across a multi-season study of crossbred beef steers," *BMC Genomics*, vol. 19, no. 1, p. 430, 2018. [Online]. Available: https://doi.org/10.1186/s12864-018-4769-8

[11] M. D. Li, T. C. Burns, A. A. Morgan, and P. Khatri, "Integrated multi-cohort transcriptional meta-analysis of neurodegenerative diseases," *Acta Neuropathologica Communications*, vol. 2, no. 1, p. 93, 2014. [Online]. Available: https://doi.org/10.1186/s40478-014-0093-y

[12] M. Andres-Terre, H. M. McGuire, Y. Pouliot, E. Bongen, T. E. Sweeney, C. M. Tato, and P. Khatri, "Integrated, multi-cohort analysis identifies conserved transcriptional signatures across multiple respiratory viruses," *Immunity*, vol. 43, no. 6, pp. 1199–1211, 2019/07/11 2015. [Online]. Available: https://doi.org/10.1016/j.immuni.2015.11.003

[13] V. Sandhu, K. J. Labori, A. Borgida, I. Lungu, J. Bartlett, S. Hafezi-Bakhtiari, R. E. Denroche, G. H. Jang, D. Pasternack, F. Mbaabali, M. Watson, J. Wilson, E. H. Kure, S. Gallinger, and B. Haibe-Kains, "Meta-analysis of 1,200 transcriptomic profiles identifies a prognostic model for pancreatic ductal adenocarcinoma," *JCO Clinical Cancer Informatics*, no. 3, pp. 1–16, 2019, pMID: 31070984. [Online]. Available: https://doi.org/10.1200/CCI.18.00102

[14] H. Huang, C.-C. Liu, and X. J. Zhou, "Bayesian approach to transforming public gene expression repositories into disease diagnosis databases," *Proceedings of the National Academy of Sciences*, vol. 107, no. 15, pp. 6823–6828, 2010. [Online]. Available: https://www.pnas.org/content/107/15/6823

[15] V. Pelechano and J. E. Pérez-Ortín, "There is a steady-state transcriptome in exponentially growing yeast cells," *Yeast*, vol. 27, no. 7, pp. 413–422. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/yea.1768

[16] J. Garcıa-Martınez, A. Aranda, and J. Pérez-Ortın, "Genomic run-on evaluates transcription rates for all yeast genes and identifies gene regulatory mechanisms," *Molecular Cell*, vol. 15, no. 2, pp. 303–313, 2019/01/29 2004. [Online]. Available: https://doi.org/10.1016/j.molcel.2004.06.004

[17] V. Pelechano, S. Chávez, and J. Pérez-Ortín, "A complete set of nascent transcription rates for yeast genes," *PloS one*, vol. 5, no. 11, pp. e15 442; e15 442–e15 442, 11 2010. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/21103382

[18] B. M. Zid and E. K. O'Shea, "Promoter sequences direct cytoplasmic localization and translation of mrnas during starvation in yeast," *Nature*, vol. 514, no. 7520, pp. 117–121, 10 2014. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/25119046

[19] M. A. Freeberg, T. Han, J. J. Moresco, A. Kong, Y.-C. Yang, Z. J. Lu, J. R. Yates, and J. K. Kim, "Pervasive and dynamic protein binding sites of the mrna transcriptome in saccharomyces cerevisiae," *Genome biology*, vol. 14, no. 2, pp. R13–R13, 02 2013. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/23409723

[20] C. Martínez-Mira, A. Conesa, and S. Tarazona, "Mosim: Multi-omics simulation in r," *bioRxiv*, p. 421834, 01 2018. [Online]. Available: http://biorxiv.org/content/early/2018/09/20/421834.abstract

[21] A. McKinlay, C. L. Araya, and S. Fields, "Genome-wide analysis of nascent transcription in saccharomyces cerevisiae," *G3 (Bethesda, Md.)*, vol. 1, no. 7, pp. 549–558, 12 2011. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/22384366

[22] L. Castells-Roca, J. García-Martínez, J. Moreno, E. Herrero, G. Bellí, and J. Pérez-Ortín, "Heat shock response in yeast involves changes in both transcription rates and mrna stabilities," *PloS one*, vol. 6, no. 2, pp. e17 272–e17 272, 02 2011. [Online]. Available: https://www.ncbi.nlm.nih.gov/pubmed/21364882

[23] S. Wold and M. Sjostrom, "PLS-Regression: A basic tool of chemometrics," pp. 109–130, 2001.

[24] M. Tenenhaus, *La régression PLS: théorie et pratique*. Editions Technip, 1998. [Online]. Available: https://books.google.es/books?id=OesjK2KZhsAC

[25] A. Folch-Fortuny, R. Vitale, O. E. de Noord, and A. Ferrer, "Calibration transfer between NIR spectrometers: New proposals and a comparative study," *Journal of Chemometrics*, vol. 31, no. 3, p. e2874, 2017. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/cem.2874

[26] S. García Muñoz, J. F. MacGregor, and T. Kourti, "Product transfer between sites using Joint-Y PLS," *Chemometrics and Intelligent Laboratory Systems*, vol. 79, no. 1-2, pp. 101–114, 2005.

[27] J. M. Andrade, M. P. Gómez-Carracedo, W. Krzanowski, and M. Kubista, "Procrustes rotation in analytical chemistry, a tutorial," *Chemometrics and Intelligent Laboratory Systems*, vol. 72, no. 2, pp. 123–132, 2004.

[28] J. R. Hurley and R. B. Cattell, "The procrustes program: Producing direct rotation to test a hypothesized factor structure," *Behavioral Science*, vol. 7, no. 2, pp. 258–262, 1962. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/bs.3830070216

[29] J. A. Hartigan, *Clustering Algorithms*, 99th ed. New York, NY, USA: John Wiley & Sons, Inc., 1975.

[30] J. Hartigan and M. Wong, "A K-Means Clustering Algorithm," *Applied Statistics*, vol. 28, no. 1, pp. 100–108, 1979. [Online]. Available: https://www.jstor.org/stable/2346830

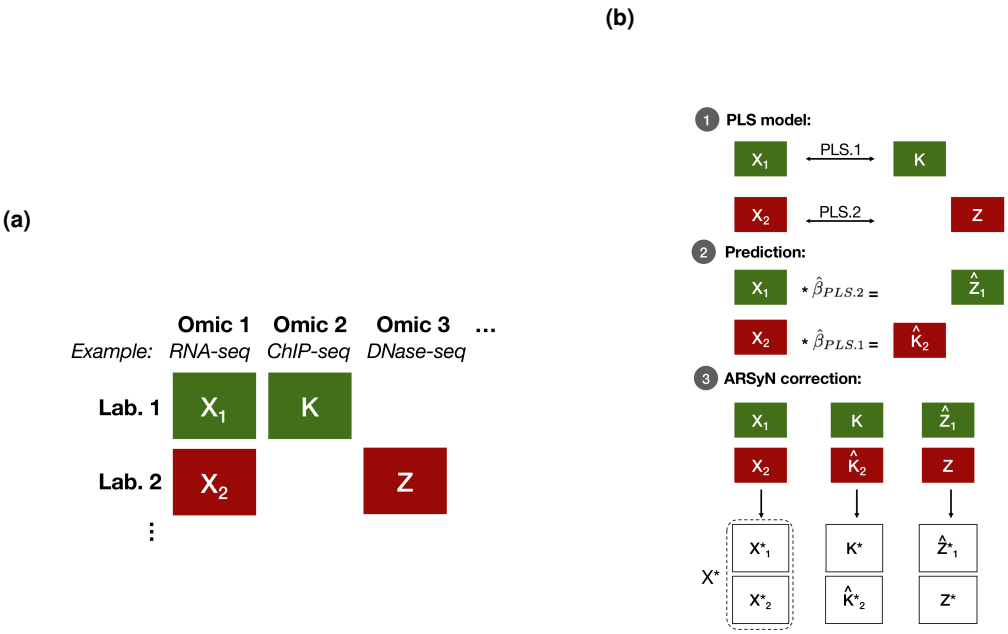


Figure 1. Description of MultiBaC method to correct batch effects in multiomic data from different laboratories. (a) Minimal size problem example in which one omic data type is shared by both laboratories and each laboratory may have other omic data types in an exclusive manner. (b) Overview of MultiBaC strategy, which combines PLS regression with conventional ARSyN batch effect correction. 1: A PLS model is built per laboratory to explain the non-common omic with the shared one. 2: For each laboratory, the initially missing omic is predicted. 3: ARSyN correction is applied on each omic data type by using predicted data.

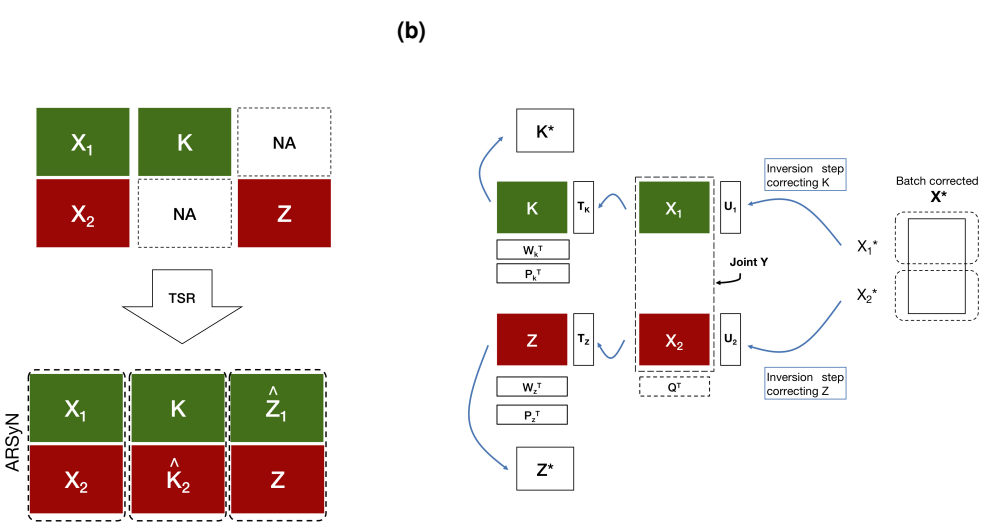


Figure 2. Outline of alternative methods for multiomic batch correction. The matrix notation used is the same as in Figure 1(a). (a) TSR: After TSR, a traditional BECA (e.g. ARSyN) can be applied. (b) JY-PLS: $\mathbf{W_i}$, $\mathbf{P_i}$ and $\mathbf{T_i}$ are weights, loadings and scores of $\mathbf{K}$ and $\mathbf{Z}$ matrices, respectively. $\mathbf{U_i}$ are the scores for X matrices. $\mathbf{Q^T}$ is the matrix of common weights of $\mathbf{X}$ matrices. $\mathbf{X}^*$ is used in the JY-PLS inversion step to obtain $\mathbf{K}^*$ and $\mathbf{Z}^*$.

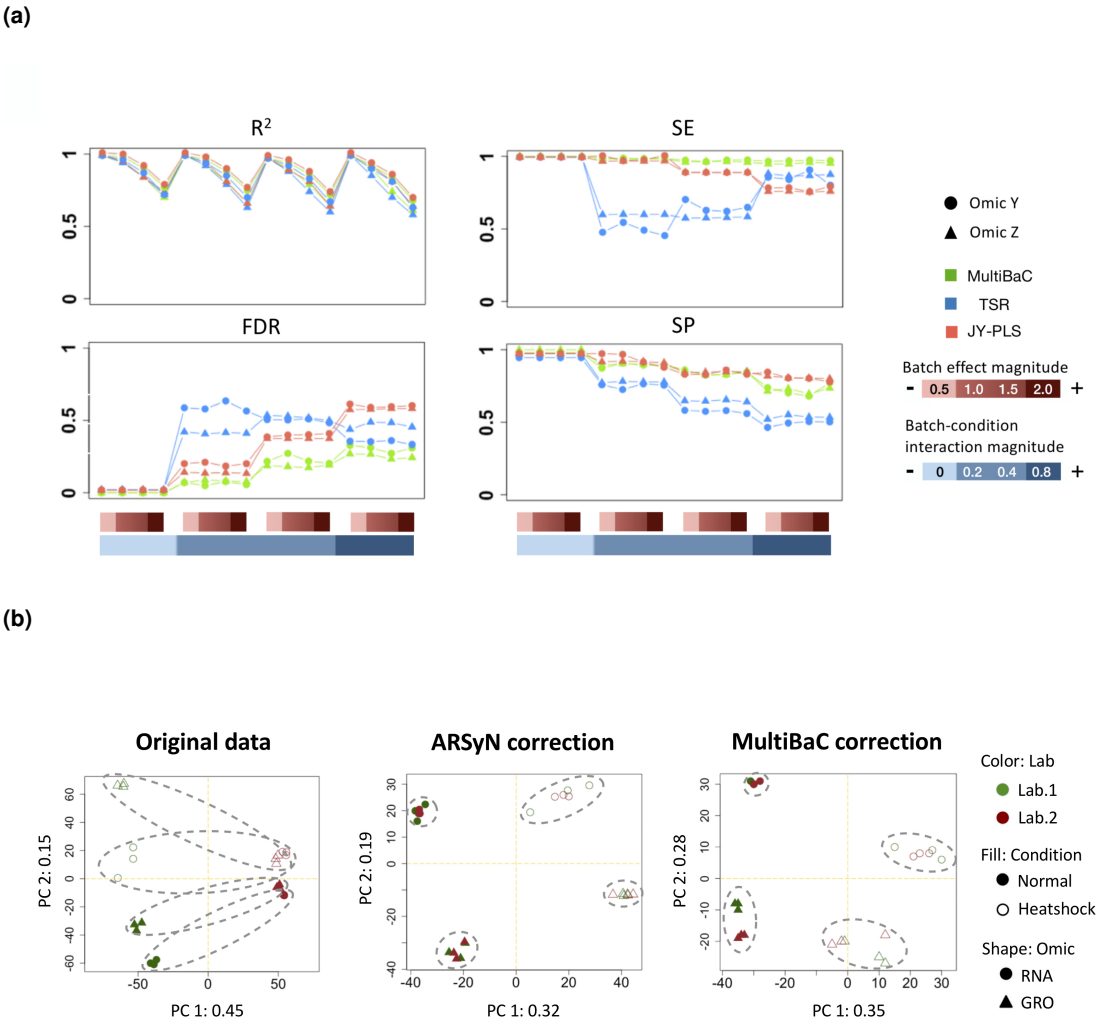
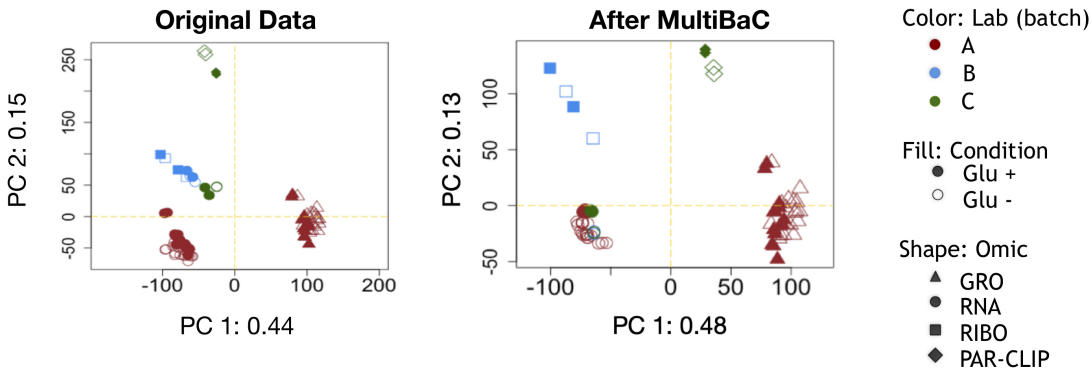


Figure 3. Performance of MultiBaC correction. (a) Simulated data results. *Top-left*: Latent space concordance (R^2). *Bottom-left*: False Discovery rate (FDR). *Bottom-right*: Specificity (SP). *Top-right*: Sensitivity (SE). Rectangles at the bottom represent the batch (top) and interaction (bottom) magnitudes as explained in Supplementary Materials 1. (b) Proof of concept data results. *Left*: PCA score plot for original data. First principal component (main source of variability) groups samples by lab instead of by omic or treatment. *Middle*: PCA score plot for ARSyN batch-corrected data. First principal component groups samples by condition, so batch effect is completely removed. *Right*: PCA score plot for MultiBaC batch-corrected data. First principal component groups samples by condition as in ARSyN correction but a residual batch effect is shown by the second component for normal condition in gene expression data. Dashed line ellipses are grouping samples from different batches by omic-condition factor.

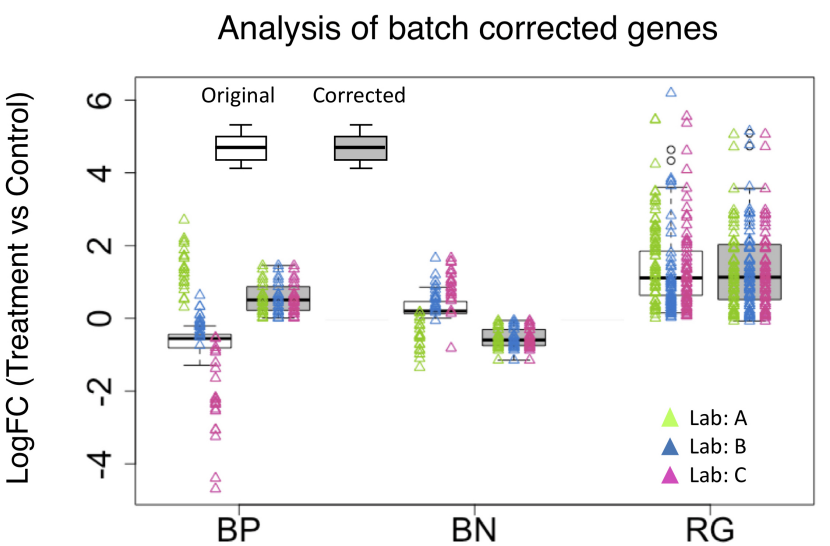
(a)



(b)

		CORRECTED									TOTAL
		1	2	3	4	5	6	7	8	9	
ORIGINAL	1	214	172	0	2	2	0	119	159	24	692
	2	5	146	0	40	69	0	8	58	0	326
	3	0	0	386	65	140	6	6	0	0	603
	4	0	13	17	261	48	0	2	0	6	347
	5	0	2	72	9	345	2	26	21	0	477
	6	0	0	75	59	0	196	2	0	0	332
	7	0	16	91	6	39	12	288	51	1	504
	8	10	4	0	0	52	0	2	236	0	304
	9	0	2	56	100	2	75	38	0	48	321
TOTAL		229	355	697	542	697	291	491	525	79	3906

(c)



(d)

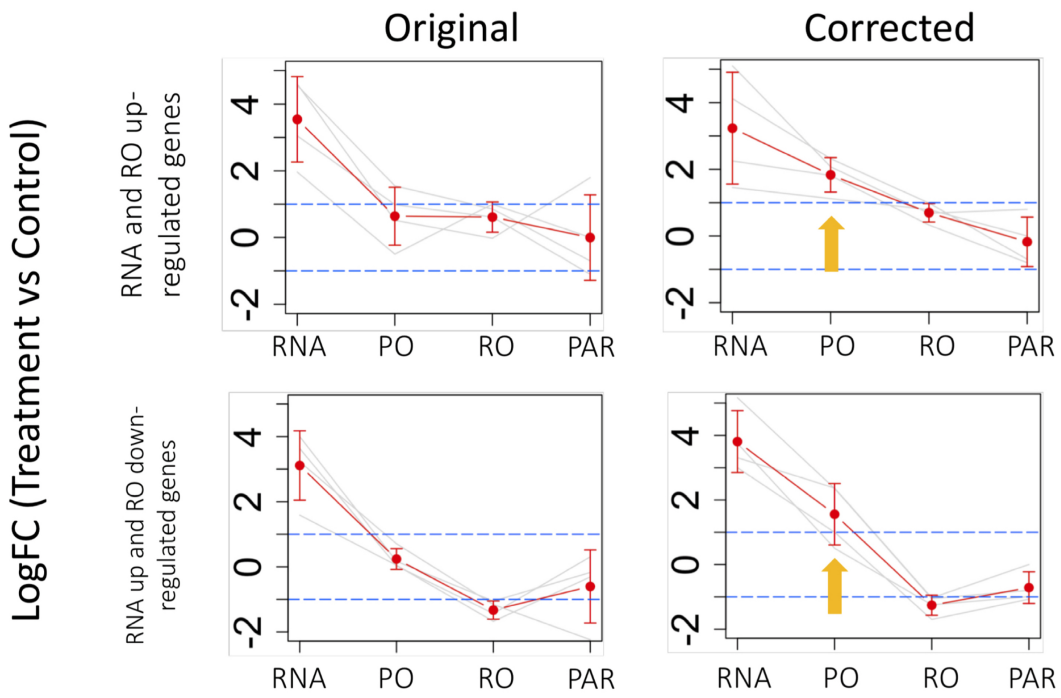


Figure 4. MultiBaC results on the ”distributed yeast multiomics dataset” data. (a) PCA score plot of the global matrix with all the omic data types (merged by genes) after MultiBaC correction. Dashed line ellipses are grouping samples from different batches by omic-condition factor. (b) Number of genes shared between clusters generated from original data (rows) and corrected data (columns). Diagonal cells contain genes that have been assigned to the same pattern before and after correction. Yellow cells correspond to clusters with an important number (at least 10% of diagonal box) of genes whose trend changed after correction. (c) RNA values of 42 genes that have changed the sign of their logFC after correction. Become Positive Genes (BP) are genes that were down-regulated in the original data (white boxes) but up-regulated after correction (gray boxes). Become Negative genes (BN) had the opposite behavior. Random Genes (RG) are 100 up-regulated genes randomly selected. Triangles show the logFC value for each single gene in each lab. (d) LogFC values per omic before and after MultiBaC correction. First row: RNA and Ribosome Occupancy (RO) up-regulated genes. Second row: RNA up-regulated but RO down-regulated genes. Each line corresponds to the profile of a gene in the corresponding group. The doted central line is the average profile of all the genes in the group, and the segment at each point represents the mean value $\pm$ the standard deviation. Dashed lines remark the logFC threshold values $+1$ and -1. Yellow arrows indicate the increase of Polymerase Occupancy (PO) values after correction.