Dr. Kim-Anh Lê Cao
Snr Lecturer, Statistical Genomics
School of Mathematics & Statistics
Melbourne Integrative Genomics
The University of Melbourne VIC 3010
T: +61 (0)3834 43971
@: kimanh.lecao@unimelb.edu.au

THE UNIVERSITY OF MELBOURNE

FACULTY OF SCIENCE

Nov 30, 2018

Dear Editor,

Please find attached a revision to our manuscript 'DIABLO: an integrative approach for identifying key molecular drivers from multi-omic assays' as a research article for the Systems Biology category in Bioinformatics.

We would like to thank the reviewers for their comments and we have substantially extended our simulation analyses and improved the clarity of the text. We have made extensive changes in the main manuscript and the supplemental material to address their comments.

The method is implemented in the open source R package mixOmics (now moved to Bioconductor), and our R scripts in R markdown format, along with detailed tutorials are available on our companion website http://www.mixOmics.org/mixDIABLO. We look forward to your reply.

Yours sincerely,

Dr. Kim-Anh LÊ CAO

Reviewer: 1

Comments to the Author

Summary

Singh, et al. present a new method, DIABLO, for supervised biomarker discovery of multiple 'omics datasets. More specifically, DIABLO is designed to overcome the computational challenge of identifying molecular features in different datasets predictive of a phenotypic response (e.g. cancer subtype). This is an important and difficult challenge given the scale of 'omics data and that it is increasingly common for researchers to take multiple types of molecular measurements (e.g. mRNA, miRNA, protein expression, …) per sample. Singh, et al. use a matrix factorization approach, specifically a generalized version of canonical correlation analysis to incorporate supervision in the form of phenotypic labels. They demonstrate DIABLO on simulated and real data, including a breast cancer and asthma dataset, and compare supervised/unsupervised and integrative/non-integrative approaches. The results on both simulated and real data are somewhat mixed.

Overall, the DIABLO method is novel and interesting, as are the applications and some of the analysis. However, despite these contributions, I recommend that the authors revise their manuscript for two main reasons. First, in multiple places the manuscript reads like a draft and requires major edits. Second, the analysis of the results of DIABLO on simulated and real data is incomplete. I elaborate on these and other points below.

Major comments

(1) In many places the manuscript reads like a draft and/or is missing key details, and also includes many typos. These include:

* Limited motivation for the new supervised approach. In particular, there is no substantive review of related work. As such, the authors' claim that existing "supervised strategies are unable to capture the shared information across multiple biological domains when identifying the key molecular drivers associated with a phenotype" (page 3) is unsupported.

We agree with the reviewer that compressive review of current knowledge and gaps would benefit the motivation of our approach. We had provided a summary figure in our Supplemental Material Fig S1, but we also have better detailed one paragraph in the introduction, which reads as:

Many strategies (component-based, message-passing, Bayesian methods, network-analysis, classification schemes) have been proposed for multi-omics data integration to answer various questions, incorporating experimental data as well as curated data from biological databases (see Suppl. Fig. S1, Zeng and Lumley 2018; Ritchie *et al.* 2015; Bersanelli *et al.* 2016; Meng *et al.* 2016; Huang *et al.* 2017; Rohart *et al.* 2017b). These include data-driven methods for identifying novel phenotypic clusters such as Similarity Network Fusion (Wang *et al.*, 2014), Bayesian Consensus Clustering (Kirk *et al.*, 2012), and methods for extracting common sources of variation such as joint Non-negative Matrix Factorization (Zhang *et al.*, 2012), Joint and Individual Variation Explained (Lock *et al.*, 2013), sparse MultiBlock Partial Least Squares (Li *et al.*, 2012), regularized and sparse Generalized Canonical Correlation Analysis (Tenenhaus and Tenenhaus, 2011; Tenenhaus *et al.*, 2014) and Multi-Omics Factor Analysis (Argelaguet *et al.*, 2018). Other methods such as Passing Attributes between Networks for Data Assimilation (Glass *et al.*, 2013), Sparse Network regularized Multiple Non-negative Matrix Factorization (Zhang *et al.*, 2011) and Reconstructing Integrative Molecular Bayesian NETworks (Zhu *et al.*, 2012) can be used to incorporate curated data with experimental data in order to reconstruct biological networks. All of these methods are examples of unsupervised multi-omics data integration, that is, without the need of sample labels that categorize samples based on a certain phenotype or trait. However, researchers are also interested in multi-omics biomarkers that are predictive of disease, *i.e.* supervised methods in which molecular patterns that span across biological domains explain or characterise a known phenotype.

Supervised data integration approaches for the classification of multiple phenotypes (*e.g.* PAM50 breast cancer phenotypes) include multi- step approaches that concatenate all data prior to applying a classification model, or ensemble-based in which a classification model is applied separately to each omics data and the resulting predictions are combined based on average or Majority vote (Günther *et al.*, 2012). These approaches can be biased towards certain omics data types, and do not account for interactions between omic layers (Aben *et al.*, 2016; Ma *et al.*, 2016). Recently, classification approaches such as Network smoothed t- statistics Support Vector Machines (Cun and Fröhlich, 2013), Generalized Elastic Net (Sokolov *et al.*, 2016), and adaptive Group-Regularized ridge regression (van de Wiel *et al.*, 2016) have incorporated curated biological data such as PPI data, genetic pathway data, and type of methylation probes. These methods are still limited to single omics data such that, either the concatenation or ensemble-based schemes must be applied to incorporate additional data-types. Other approaches include The Analysis Tool for Heritable and Environmental Network Associations (ATHENA) based on a Grammatical Evolution Neural Network that integrates multi- omics data for the prediction of

clinical outcomes (Kim *et al.*, 2013). However, the approach requires initial filtering, feature selection and modelling independently on each omics dataset prior to integration.

Note that most of the introduction section has been rewritten.

* Confusing presentation of the DIABLO algorithm. The authors should write out the DIABLO algorithm in full.
* Confusing presentation of the sGCCA algorithm. The notation for a_h^k is inconsistent, and more importantly, all a_h^k are completely missing from the objective of the optimization problem. Further, the authors do not review how sGCCA solves the optimization problem.

* Never defining sPLSDA
* Never defining N_new
* "validatio" —> "validation" (page 2)

The method section has been entirely rewritten. Section 1.1 presents sGCCA and Section 1.2 introduces DIABLO. All notations have been checked and made consistent all throughout. Acronyms have been defined and typos fixed.

(2) The results on simulated and real data are also concerning, particularly in the comparatively worse performance of DIABLO (full) at classification (full refers to the design matrix which controls which omics datasets are "connected").
* It is concerning that DIABLO (full) has the worst phenotypic classification performance on simulated data. The authors claim that there is a tradeoff between discrimination and correlation, and that DIABLO is better at selecting interpretable variables. This makes sense, but is unexplored and incomplete. The authors should extend their simulation analysis to show settings in which DIABLO (full) is at least as good as existing methods, and whether the design matrix can be used to achieve stronger classification performance even in the current simulated data setup.

We decided to thoroughly extend our simulation analyses to address these important questions. We agree with the reviewer that our discussion about the discrimination and correlation trade-off needed further exploration. Therefore, in our new simulation scheme we have further studied the relationship between the covariance between datasets, classification performance (error rate) and number of variables selected. Changes appear in the main document **subsection 3.1** Correlation and discrimination trade-of, **Suppl Section S1 and Suppl Fig S3**.

The updated simulation scheme detailed in Suppl S1 is depicted in the figure below, where:

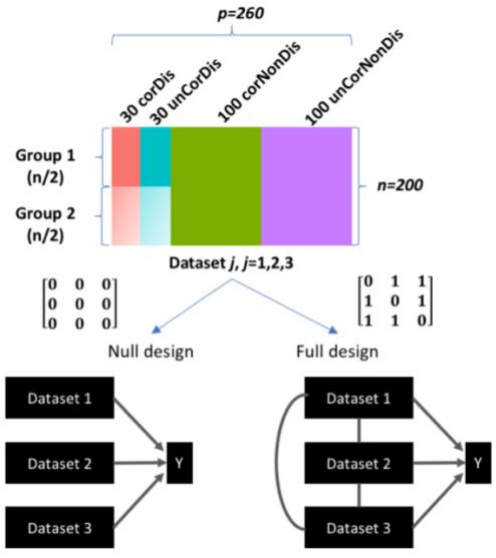


Figure. Simulated multi-omics data. *Each simulated dataset consisting of four types of variables: 30 correlated-discriminatory (corDis) variables, 30 uncorrelated-discriminatory (unCorDis) variables, 100 correlated-nondiscriminatory (corNonDis) variables and 100 uncorrelated-nondiscriminatory (unCorNonDis) variables*

Three datasets were simulated each with 200 observations (n) and 260 variables (p). The 200 observations were split equally over two groups (G1 and G2), whereas the 260 variables were generated by varying the covariance ($\sigma^2_{XY} = [0{,}5{,}10{,}15]$) between datasets and fold-change (δ=[0,1,2]) between G1 and G2: 30 correlated-discriminatory (corDis) variables, 30 uncorrelated-discriminatory (unCorDis) variables, 100 correlated-nondiscriminatory (corNonDis) variables, and 100 uncorrelated-nondiscriminatory (unCorNonDis) variables were simulated (see Figure 1A). The resulting dataset is of the form:

$$X_j = \left[X_j^{corDis} \mid X_j^{unCorDis} \mid X_j^{corNonDis} \mid X_j^{unCorNonDis} \right] + E_j, \text{ where j = 1, 2, 3}$$

The matrix containing correlated and discriminatory variables, X_j^{corDis} was generated using the following model:

$$X_j^{corDis} = \boldsymbol{u}_j^{corDis} \boldsymbol{w}_j^t, \text{ where } \|\boldsymbol{w}\| = 1, j = 1{,}2{,}3$$

where the loadings, $\mathbf{w}_1$, $\mathbf{w}_2$, and $\mathbf{w}_3$ were 30-vectors, and the elements were drawn from a uniform distribution in the interval of [-0.3, 0.2] U [0.2, 0.3]. For G1, the outer components $\boldsymbol{u}_1^{corDis}$, $\boldsymbol{u}_2^{corDis}$, $\boldsymbol{u}_3^{corDis}$ were 3-vectors drawn from a multivariate normal distribution with a mean value of 0 and a mean value of δ=[0,1,2] for G2. The covariance

between pairs of components was set to cov($\boldsymbol{u}_i^{corDis}$, $\boldsymbol{u}_j^{corDis}$) = σ_{ij}^2 (for $i \neq j$) where σ_{ij}^2=[0,5,10,15] and cov($\boldsymbol{u}_i^{corDis}$, $\boldsymbol{u}_j^{corDis}$) =0 (for *i=j*).

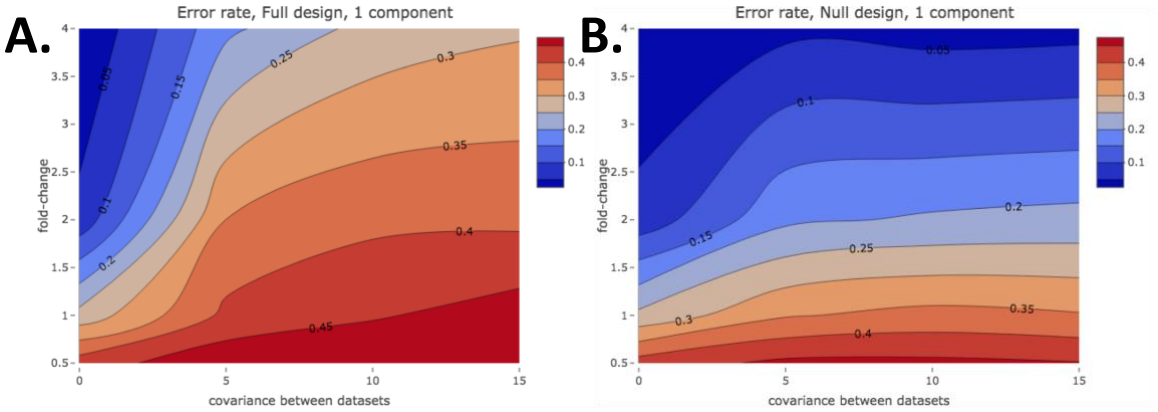
The matrix containing uncorrelated and discriminatory variables, $X_j^{unCorDis}$ was generated using the following model:

$$X_j^{unCorDis} = \boldsymbol{u}_j^{unCorDis}\boldsymbol{w}_j^t, \text{ where } \|\boldsymbol{w}\| = 1, j = 1,2,3$$

where the loadings, $\mathbf{w}_1$, $\mathbf{w}_2$, and $\mathbf{w}_3$ were 30-vectors, and the elements were drawn from a uniform distribution in the interval of [-0.3, 0.2] U [0.2, 0.3]. For G1, the outer components $\boldsymbol{u}_1^{unCorDis}$, $\boldsymbol{u}_2^{unCorDis}$, $\boldsymbol{u}_3^{unCorDis}$ were 3-vectors drawn from a multivariate normal distribution with a mean value of 0 and a mean value of δ=[0,1,2] for G2. The covariance between pairs of components was set to cov($\boldsymbol{u}_i^{unCorDis}$, $\boldsymbol{u}_j^{unCorDis}$) = $\sigma_{ij} i \neq j \sigma_{ij} \boldsymbol{u}_i^{unCorDis}\boldsymbol{u}_j^{unCorDis}$0 for all *i* and *j*.

The nondiscriminatory variables (corNonDis and unCorNonDis) were generated by drawing 100-vectors each with 200 elements, from a multivariate normal distribution with a mean of 0. For correlated variables, the covariance between pairs of components was set to cov($\boldsymbol{u}_i^{corDis}$, $\boldsymbol{u}_j^{corDis}$) = σ_{ij}^2 (for $i \neq j$) where σ_{ij}=[0,5,10,15] and cov($\boldsymbol{u}_i^{corDis}$, $\boldsymbol{u}_j^{corDis}$) =0 (for *i=j*). For uncorrelated variables the covariance between pairs of components was set to cov($\boldsymbol{u}_i^{unCorDis}$, $\boldsymbol{u}_j^{unCorDis}$) = 0 for all *i* and *j*. $\boldsymbol{E}_j$ is a 200 x 260 residual matrix where each element is drawn from a normal distribution with zero mean and variance equal to 0.5.

For each simulated set of datasets with a given covariance and fold-change level, DIABLO models were constructed either with the null or full design and their performance was evaluated using 10-fold cross-validation. This procedure was repeated 20 times and the classification error rates were averaged.

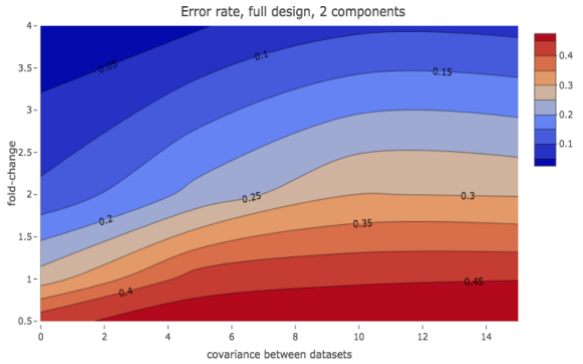


The figure above (**Suppl. Fig. S3**) depicts contour plots of the error rate for different degrees of covariance and fold-change (signal) either using the full or null design in the diablo models (selecting 60 variables on 1 comp). As can be observed in **A**, increasing the covariance between datasets leads to an increase in the classification error rate (blue to red) for a given fold-change. For the Null design in **B** however, the error rate is similar for given effect size, irrespective of the covariance between datasets.

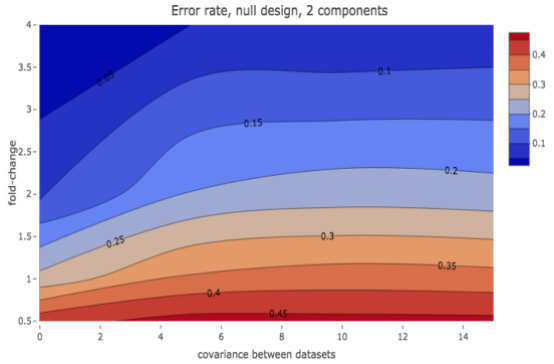
Our second step was then to study whether the addition of components improves the classification performance of DIABLO_full. The underlying assumption of DIABLO_full is that a common source of variance exists between datasets. We hypothesized that the high classification error rate was due to the fact that the model seeks for highly correlated components across datasets, however a good classification tasks requires uncorrelated information to be extracted. One way to add uncorrelated information in DIABLO_full is to consider the other dimensions of the model (i.e. the components built on the residual data after the deflation step).

The figure below panels **C** and **D** (**Suppl. Fig. S3**) show the contour plots of the classification error rate for different degrees of covariance and fold-change (signal) either using the full or null design as in **A** and **B** but when the components of the second dimension of DIABLO have been added. Since the sets of components in dimension 1 are orthogonal to the set of components in dimension 2, the variables selected on the second set of components are uncorrelated with those from the first component but still predictive of the outcome, resulting in a lower error rate.

C.



D.



To conclude on these simulation analyses, we have shown that DIABLO_full's performance can be affected by the covariance structure between datasets, and that the performance can be improved when adding orthogonal information (components) in the model.

* On simulated data, the authors only perform limited benchmarking against existing approaches, only comparing to sPLSDA, and do not provide an explanation for this missing analysis. There is more extensive benchmarking on real data.

The aim of the simulation study was mainly to evaluate the classification performance of the DIABLO designs and whether the methods were able to identify the correct variables. Therefore, we could only include the supervised methods (i.e. Concatenation and Ensemble scheme with sPLS-Discriminant Analysis).

* Some of the results on real data are not well-explained and/or do not have sufficient context.

Unfortunately we are lacking space in the main manuscript to fully describe the data. The information is presented in **Suppl Section S2**, along with the pre-processing steps.

* For example, there are quite substantial error rates for predicting PAM50 breast cancer subtypes (ranging from ~5-50%). How are these results to be interpreted? The PAM50 subtypes have known clinical implications, so if the authors find subtypes that are refined or different from PAM50, they should provide some sort of validation (e.g. with clinical data such as survival). Otherwise, what is the point of using supervision?

The purpose of the Breast Cancer analysis was to focus on the application of DIABLO to multiclass phenotypes and the biological relevance of the variables selected. We describe that some phenotypes are easier to separate (Basal, Her2) as compared to other subtypes (LumA, and LumB), as we depict using various illustration such as component plots, and heatmaps. It was not our intent to identify new subtypes, but rather to investigate whether there were multi-omics panels that could predict PAM50 subtypes. Therefore downstream validation with survival such as through analysis has not been considered. However, and to answer Reviewer 2 comments, we have added a section on classification performance as we have access to an independent test set in this study:

3.4 Competitive classification performance of DIABLO

In the breast cancer study we used independent test data to compare DIABLO, Elastic Net classifiers and both Concatenation-based and ensemble-based schemes based on the sPLSDA (see Suppl. S5 for details). Parameters of each integrative method were tuned using 5x5-fold CV on the training datasets to identify the optimal model, before assessing the methods' prediction performance on the test data. We found that DIABLO models performed similarly to Concatenation-based classifiers and out- performed Ensemble-based classifiers (Suppl. Table S2). Concatenation- based classifiers were biased towards the more predictive variables (mRNA or CpGs), whereas DIABLO selected variables evenly across datasets and had similar error rates between training and test datasets.

* The description of the "multilevel DIABLO" approach is confusing, and does not seem to be discussed in the Methods (though the authors say it is in the Results on page 23).

Due to the word limit restrictions, this section was moved to the supplementary materials and was not corrected in the main text. We apologize for this over sight. Additional details for this approach can be in **Suppl Section S7** and is stated as follow:

For multivariate analyses, A multilevel approach separates the within subject variation matrix (X_w) and the between subject variation (X_b) for a given dataset (X) (Westerhuis *et al.*, 2010; Liquet *et al.*, 2012), ie. $X = X_w + X_b$. In the case of a two-repeated measured problem (e.g. pre vs post challenge), the within subject variation matrix is similar to calculating the net difference for each individual between the data obtained for pre and post challenge. For each omics dataset, the within-subject variation matrix was extracted prior to applying DIABLO. In the asthma study, the multilevel approach (called variance decomposition step) was applied to the cell-type, gene and metabolite module datasets.

Minor comments

* The authors have integrated DIABLO into their mixOmics R package, and it seems well-documented.

Thank you!

* What is the runtime and memory footprint of DIABLO?

Computational footprint of DIABLO has been largely reduced since the first submission of this manuscript. Runtime and memory usage are reported in the following table on simulated data, comparing a single omics analysis using sPLS-DA and an integrative analysis using DIABLO in mixOmics, with a macbook pro 2013, 2.6GHz, 16Go Ram. The `tune` function is used to identify the number of variables to select from each dataset using cross-validation and grid of values for the number of components and number of features to select per component (the tune method is currently not implemented for NA values). The `model` is the final model run based on the tuned parameter. V6.3.2 is the current version in mixOmics in Bioconductor.

	Single 'omics sPLS-DA				*N*-integration DIABLO			
N	1000				1000			
P	20,000				10,000; 10,000			
NA	no		yes		no		yes	
	`model`	`tune`	`model`	`tune`	`model`	`tune`	`model`	`tune`
v6.1.1 (sec)	13.7	160	370	42 min	172	40 min	-	-
v6.3.2 (sec)	1.0	12	3.2	20 sec	2	18 sec	3.4	29
× faster	14	13	115	126	86	133	-	-
v6.1.1 (Go)	6.1	78	16.7	200	9.4	202	-	-
v6.3.2 (Go)	0.8	4.6	2.4	14.1	0.8	5.1	2.4	18.8
× better	8	17	7	14	12	40	-	-

Reviewer: 2

Comments to the Author

In their manuscript the authors present a method (DIABLO) to integrate data from multiple omics in a semi-supervised manner providing a balance between unsupervised methods that do not take into account known labels and supervised methods that do not take into account correspondence structures between omics. The method is based upon sGCCA (Tenenhaus et al al 2014) by including the labels as an additional block and implemented as part of the mixOmics package.

Overall, the method and the results are presented in a clear manner and the method seems to provide a good balance between supervised and unsupervised approaches. The authors demonstrate the ability of the method to find correlated discriminative features in simulations and convincingly show that the method is able to infer discriminative and biological meaningful components in several applications. More care could be taken when discussing the relationship of the proposed methods to existing approaches and in evaluating its predictive performance.

Major comments:

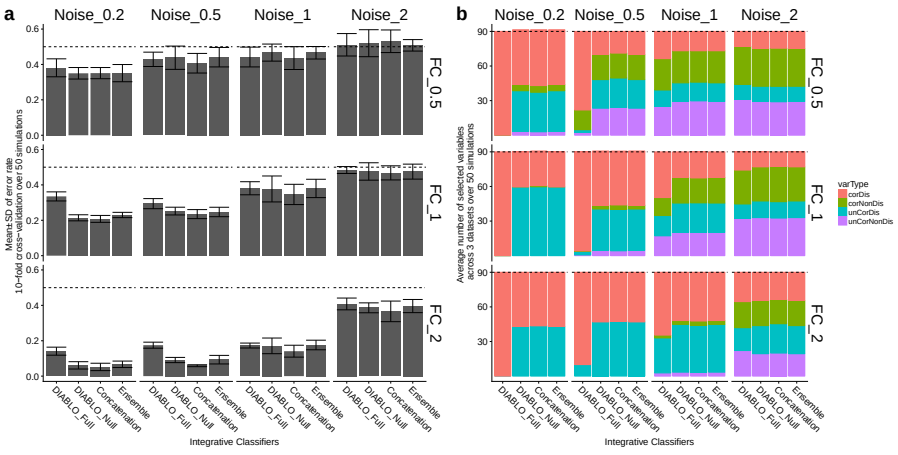
1. In the introduction the authors comment on supervised and unsupervised methods, however they do not relate their method to existing methods that aim at partly supervised integration of multiple data types such as for example sparse Multi-Block Partial Least Squares or sparse supervised CCA. This relationship and the contributions should be discussed more carefully.

This comment re-joins the comment from reviewer 1, see our answer in point **(1).**

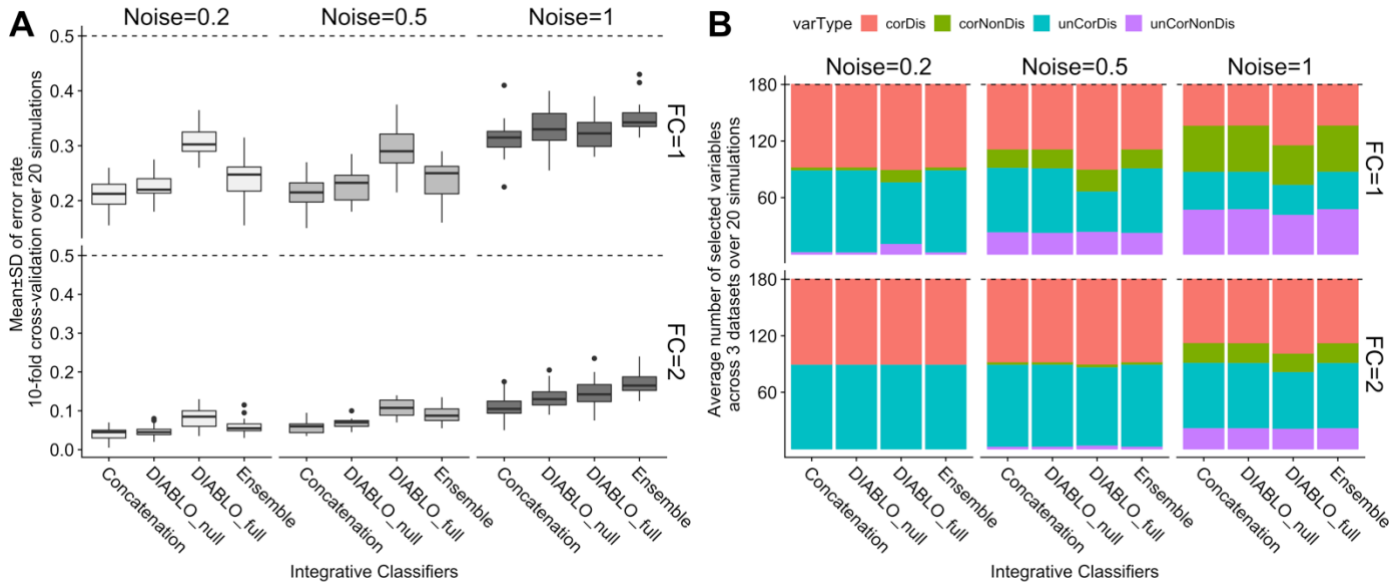
2. The authors convincingly demonstrate that the method is very good at finding biological meaningful components that well discriminate phenotypic groups. In terms of predictive performance there seems to be a risk, when concentrating on correlations between data sets, that DIABLO (with full or partly full design matrix) could overlook single strongly discriminative features in a data set when these have little correlation to other omic data sets. For example, in the simulation study DIABLO_Full mainly discovers correlated discriminative features. Would the method be able to discover all 180 discriminative features if 60 instead of 30 variables were selected from each data set?

In the previous analysis we had simulated 60 predictive variables per dataset (30 correlated and 30 uncorrelated), however as the reviewer suggests we can also allow for all methods to select 60 variables per dataset (180 in total) in order to determine whether all 180 discriminative features were selected. **Figure 1** in the manuscript has been updated and now includes the selection of 180 variables. The text in **section 3.1** has also been updated accordingly.

Before:

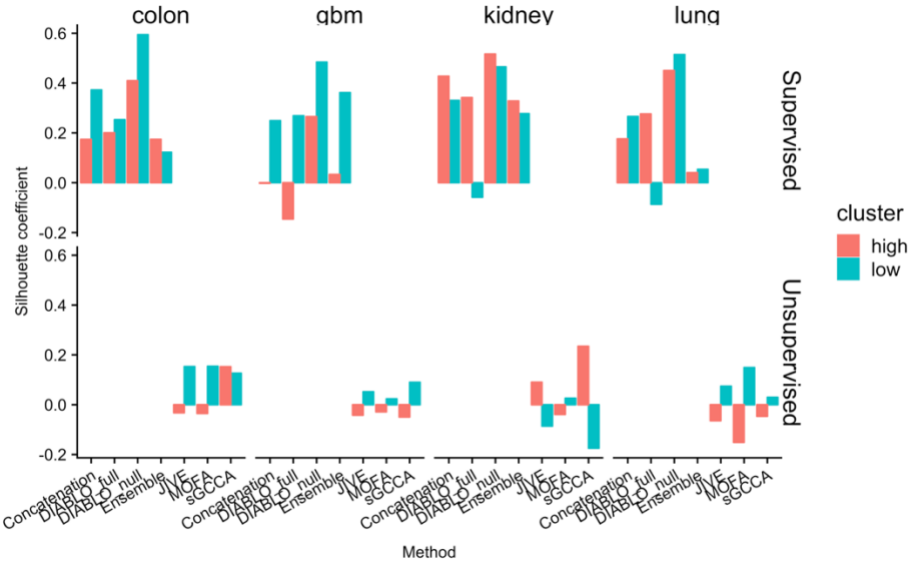


After:



In addition, it would also be good to see a method comparison in terms of classification performance on real data (e.g. on the benchmark data sets by Wang et al 2014) using independent test sets.

The purpose of the benchmarking analyses was to compare the types of variables selected across different datasets and integrative methods. However, since the methods differed with respect to variable selection, and level of supervision and cohorts lacked independent test datasets, a comparison with respect classification performance was neglected. In the revised analyses we have performed internal validation to assess for consistency within the phenotypic groups in the benchmarking experiments using the silhouette coefficient Since all integrative methods that have been included in the benchmarking experiments are component-based methods, the first two principal components were used to compute the average silhouette coefficient per dataset, per group for all methods. **Suppl. Fig S10** has been added as:



Supplementary Figure S10. Internal validation of high and low phenotypic groups for all method in the benchmarking experiments.

The silhouette for each data i, was computed as the normalized difference between two average distances (a_i and b_i), where a_i is the average distance between i and all points within its own cluster and bi is the average distance between i and all points that are not in its cluster ($s(i) = \frac{b(i)-a(i)}{\max\{a(i),b(i)\}}$). The silhouette ranges from -1 to 1, 1 being a strong indicator of cluster membership and -1 being a weak indicator of cluster membership. As can be observed, the supervised methods show stronger silhouette coefficients

as compared to unsupervised methods. This is because the principal components are associated with the phenotype of interest. DIABLO_Null consistently out-performed the methods with a higher average silhouette coefficient with respect to both phenotypic groups (high and low survival). The silhouette coefficients for the other methods were variable, however, whether this translates to a lower predictive performance in independent test data remains to be observed.

In addition, we added a classification performance comparison on the Breast Cancer PAM50 subtypes, that includes independent test sets (610 samples in the test datasets: mRNA, miRNA, and CpGs). Given the limitation of the Concatenation scheme where all variables must be present in both training and test datasets, we removed the proteins dataset for this comparative analysis. The summary of the results is presented in **Suppl. Table S2** (see below) and we added a new **section 3.4**:

3.4 Competitive classification performance of DIABLO

In the breast cancer study we used independent test data to compare DIABLO, Elastic Net classifiers and both Concatenation-based and ensemble-based schemes based on the sPLSDA (see Suppl. S5 for details). Parameters of each integrative method were tuned using 5x5-fold CV on the training datasets to identify the optimal model, before assessing the methods' prediction performance on the test data. We found that DIABLO models performed similarly to Concatenation-based classifiers and out- performed Ensemble-based classifiers (Suppl. Table S2). Concatenation- based classifiers were biased towards the more predictive variables (mRNA or CpGs), whereas DIABLO selected variables evenly across datasets and had similar error rates between training and test datasets.

Supplementary Table S2. Classification error rates [average error (sd)] of DIABLO, Concatenation-based and Ensemble-based sPLSDA and Elastic Net (enet) classifiers on the Breast Cancer study (see Suppl. Section S5 for details).

Dataset	*p*	Train	Test
Diablo_null	mRNA: 60 miRNA: 42 CpGs: 22	0.21 (0.0091)	0.19
Diablo_full	mRNA: 55 miRNA: 17 CpGs: 17	0.22 (0.0057)	0.21
Concatenation_sPLSDA	mRNA: 60 miRNA: 0 CpGs: 0	0.15 (0.013)	0.18
Concatenation_enet	mRNA: 38 miRNA: 2 CpGs: 118	0.14 (0.0072)	0.20
Ensemble_sPLSDA	mRNA: 60 miRNA: 55 CpGs: 40	0.25 (0.014)	0.28
Ensemble_enet	mRNA: 96 miRNA: 45 CpGs: 127	0.11 (0.0016)	0.23

3. To find sparse solutions the method requires the users to choose the number active variables per dataset. However, it is unclear how users should make an informed decision on this quantity, as an exhaustive grid search can be very expensive. How sensitive is the method to this choice (which possibly could lead to strong over- or under-fitting)?

The sparse methods implemented in the mixOmics R-library, including DIABLO use soft-thresholding to replace the ℓ_1 penalty by the number of variables to select from each component. This improves the usability by the user who can determine a suitable grid for their purposes. A smaller classification model may be favored if the features will be follow-up via candidate experiments or validation studies, where large model are useful for perform gene-set enrichment analyses, as more clearly specified in the **Methods section.**

2.3 Parameters tuning

- The number of variables to select per dataset and per component. A grid composed of a small number of variables (<50 with steps of 5 or 10) may suffice as we did not observe substantial changes in the classification performance during our case study analyses. The variable selection size can also be guided according to the downstream biological interpretation. For example, a gene-set enrichment analysis may require a larger set of features than a literature-search interpretation.

The grid search is also mentioned in the Discussion section:

Selecting the optimal number of variables requires repeated CV to ensure unbiased classification error rate evaluation. A grid approach was deemed reasonable and provided very good performance results, but several iterations to refine the grid may be required depending on the complexity of the classification problem. The grid search algorithm is efficient (Rohart *et al.*, 2017a), but we advise using a broad filtering strategy to alleviate computational time when dealing with extremely large datasets (> 50,000 features each).

We provide a tune function in mixOmics that performs parameter tuning using a repeated and stratified cross-validation (Rohart *et al.*, 2017). In regards to over and under-fitting, this can be assessed using cross-validation which provides an estimate of the generalizable test error.

Minor comments:

1. In 'parameter tuning' it is unclear what is meant by 'first component' in l.27 p. 9. Which design matrix is used to calculate this component?

To choose an appropriate design matrix, we suggest the user to first consider the pairwise correlation between components obtained from a PLS model. PLS only applies for the integration of 2 data sets and there is no design matrix needed for this method. Once the first set of components is extracted from PLS, the correlation between them will indicate the degree of correlation and will inform the design matrix for DIABLO.

2. The description of visualisation outputs on p. 10 would profit from an illustration in a supplementary figure or including pointers to a corresponding figure in subsequent analyses.

We thank the review for this suggestion, we have referred to exemplar figures in this section:

2.4 DIABLO visualisation outputs

To facilitate the interpretation of the integrative analysis, several types of graphical outputs were proposed and implemented in mixOmics. *Sample plots* include a consensus plot which depicts the samples by calculating the average of the components from each dataset (Fig 3A). Omic-specific sample plots can also be obtained by plotting components associated to each dataset (Suppl. Fig. S14).

Variable plots give more insights into the variables that were selected by DIABLO. Our new circos plot represents correlations between and within selected variables from each dataset. The association between variables is computed using a similarity score that is analogous to a Pearson correlation coefficient (see González *et al.* 2012); this association is displayed as a color-coded link inside the plot to represent a positive or negative correlation above a user-specified threshold. The selected variables are represented on the side of the plot, with side colors indicating each omics type, optional line plots represent the expression levels in each phenotypic group (see Suppl. Fig. S20).

Clustered Image Maps (CIM) based on the Euclidean distance and the complete linkage display an unsupervised clustering between the selected variables (centered and scaled) and the samples (see Suppl. Fig. S15). Color bars represent the sample phenotypic groups (columns) and the type of omics (rows) variables (see González *et al.* 2012).

3. alpha is missing in the objective of equation in p.6, l.7 and has inconsistent sub/superscripts in the equation

All notations were amended in the Methods section (we do not need an \alpha coefficient in Eq. (1)).

4. The description on p. 7 uses different notation and naming for loadings/coefficients vectors than on p.6 and differs

again from the description in “Prediction distances”. In general, it would be helpful to make this more consistent and avoid duplicated descriptions if possible on these two pages.

We have fully amended our notations and rewritten the Methods section.

5. The methods MOFA and JIVE have missing or malformatted citations in the text on p.14

This has been corrected in the revised draft.

6. Typo on p.2 l.31: validation

This has been corrected in the revised draft.

7. Why is the set size different in Fig. 2a between methods? To my understanding the same number of features were used from each method.

For each method, 2 sets of components were retained and 30 variables were selected for each dataset, resulting in 30 variables x 2 components x 3 datasets = 180 variables per method. Although the first and second components are orthogonal, there might be some overlap between variables selected on both components. The set size depicts the number of unique features and thus leads to the discrepancy observed by the reviewer. We describe this occurrence in Supplementary Figure S4 of the revised manuscript draft.

8. The message of Fig. 2c (upper panel) is unclear. Do the two large clusters correspond to the two components? What do the grey lines represent? A description thereof should be included into the caption.

Each network depicted the multi-omics biomarker panel (mRNA, miRNA and CpGs) identified using the different methods. The gray circles depict modules based on the edge betweenness index from the igraph R-library. For the colon cancer dataset we observed modules (clusters circled by gray lines) that included features that selected on the two components. However, this was not true for all the other cancers datasets. The caption for Figure 2 has been amended as:

Fig. 2. Benchmark for colon cancer. A) Overlap of features selected by supervised or unsupervised methods. B) Number of correlated variables in the biomarker panels for various Pearson correlation cut-offs. C) Top: network modularity of each multi-omic biomarker panel. Gray circles depict modules based on the edge betweenness index from the igraph R-library. Bottom: consensus component plots depicting the separation of subjects in the high and low survival groups. Similar patterns were observed for kidney, gbm and lung cancer datasets, see Suppl. Figs S5-S9

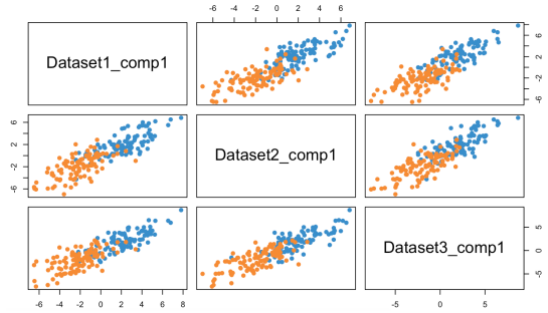
9. In the supplement the grid parameters for simulation are inconsistent within the text and with the Figure 1a.

This has been corrected in the revised draft of the manuscript.

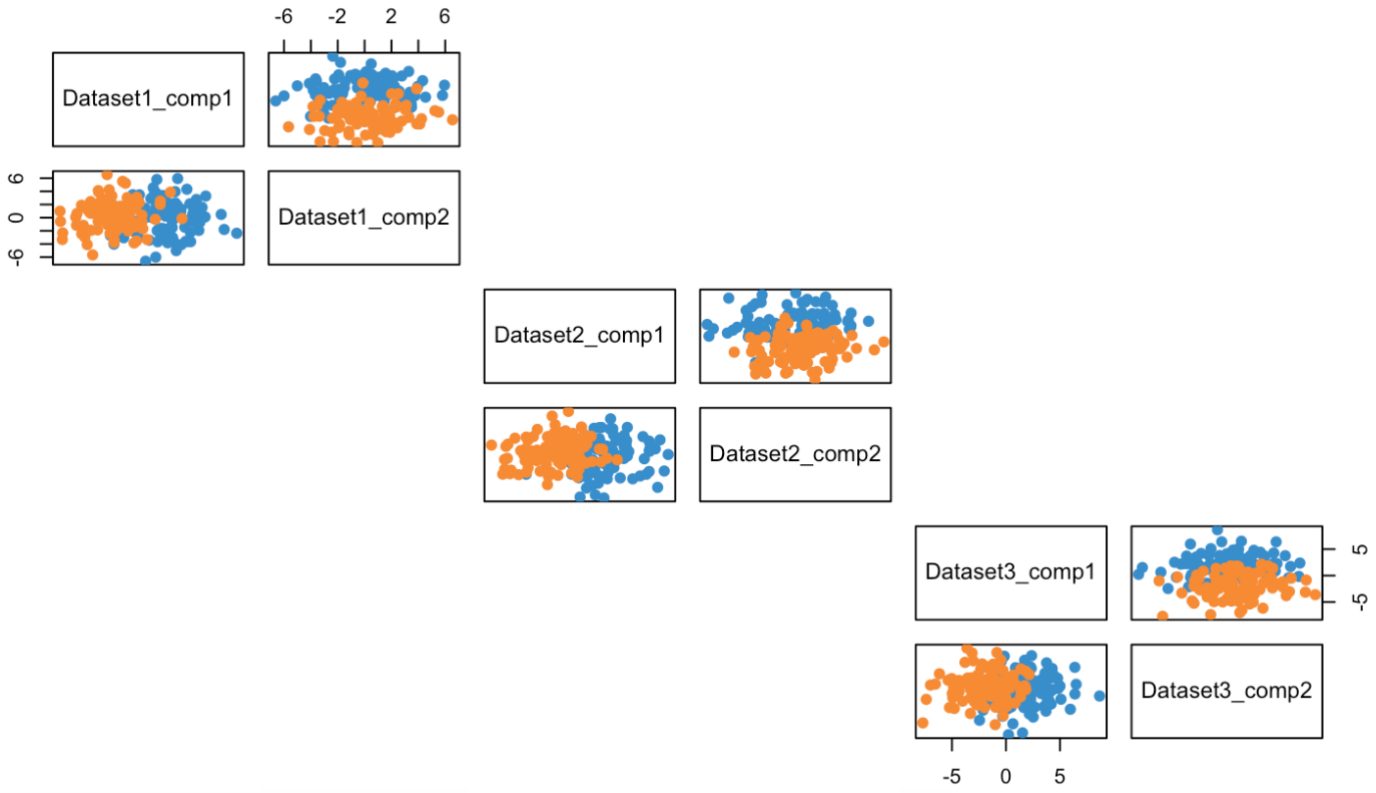
10. The correlations in Fig. S2 for the uncorrelated simulation setting seem to me still rather high. Could the authors comment on this?

Th figure was depicting an extreme case where the level of discrimination (i.e. fold-change) and correlation was very high. The fold-change is simulated as the difference between the centroids of the two groups, resulting in a high level of discrimination between groups and therefore a high degree of correlation between components of different datasets. Our scatterplot matrices created confusion as they seem to depict that the classification occurs using the components of different blocks (see below) when instead it occurs separately for each omic-type and the predictions are combined using a voting scheme (average, majority, or weighted majority, as described in the Methods section 2.2).

Previous figure: association between component 1 of different datasets. However, this plot may be misinterpreted by the reader as a depiction of the classification boundary used to discriminate the two phenotypic groups.



Correct figure (below) component plots for each dataset and each component. Given the orthogonality of components, each added component brings uncorrelated information that explains the variation in the response.



Given this confusion we have decided to remove this Supplemental figure entirely for the revised manuscript draft to avoid confusion.

11. Fig. 3a (names of proteins) and Fig. 4f are illegible

We have revised all figures to improve readability and have remove such difficult to read panels to the Supplementary Materials.