Response to Reviewers

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.  
\* Confusing presentation of the DIABLO algorithm. The authors should write out the DIABLO algorithm in full.

Move supplement information of existing methods into introduction

Pseudo Algo -> supp? (see if space)

Supplementary%20Information/S1%20Fig.pdf

\* 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)  
 to be 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.

In agreement with the reviewer, we have extended the simulation analyses to address a number of other questions:

1. Relationship between the covariance (between datasets) and the error rate and number of variables selected
2. Improving the classification error rate of the full design by adding more uncorrelated (independent) information through additional components
3. Selecting all discriminatory variables when comparing error rates between DIABLO and existing classification methods.

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 degree of correlation () between datasets and fold-change (δ) 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 (see revised Figure 2B). The resulting dataset was of the form:

+ , where j = 1, 2, 3

The discriminatory matrix () was generated using the following model:

, where

where the loadings, **w**1, **w**2, and **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 , , were 3-vectors drawn from a multivariate normal distribution with a mean value of -δ/2 and a mean value of δ/2 for G2. The covariance between pairs of components was set to cov(, ) = (for ) where =[0,5,10,15] and cov(, ) =0 (for *i*=*j*).

../analyses/simulation_study/Figures/Figure1.pdf

1. Effects of varying the degree of covariance between datasets on the error rate and number of variables selected

Conclusion: as the covariance between datasets increases, the error rate will increase with the full design but not the null design

1. Improving the classification error rate of the full design by adding more uncorrelated (independent) information (explain defalation)

Conclusion: adding additional components allows more independent (uncorrelated) information to be used to predict the class labels, resulting in improved prediction.

1. Selecting all discriminatory variables when comparing error rates between DIABLO and existing classification methods.

Conclusion: DIABLO performs comparably to existing methods.

30/08: simu data: extending keepX retrieved the nonCorDis variables and improved classif error rate, AS to apply a one-side t-test to see if there is any significant improvement?

Try intermediate designs:

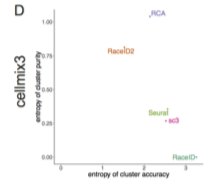
Add more components: 30/08: trying models with 1 comp but keepX small and see whether we retrieve the variables on the second comp.

Change correlation values (weak, medium, strong): 30/08: AS currently generated Errorate vs effect size vs correlation to investigate the relationship and performance. -> change color scheme and x axis label for easier interpretation.

New simulation framework

Simulate from real data? (see Yugene paper supp)

Next steps: on simple figure that will summarise the ‘winner’ (full or null) depending on correlation and discrimination structure of the data.



\* 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.

Add methods used on real data to benchmark simulation (it will be limited, focus on selected features, and error rate when applicable).

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

* Compare CV error rate in training vs. validation data. discuss why luminal a and b have poor error rates, help with interpretation, one sentence discussion about PAM50

\* 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).  
expand explanation in the supp: method

# Minor comments  
\* The authors have integrated DIABLO into their mixOmics R package, and it seems well-documented.  
\* What is the runtime and memory footprint of DIABLO?  
add memory footprint to each of the analyses including model fitting and cross-validation

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.

Move existing methods from supplement to intro

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 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.

30/08:

* simu data: extending keepX retrieved the nonCorDis variables and improved classif error rate
* currently putting a fine tuning grid to see whether we identify the correct variables, so far so good.

For the simulation study: use 2 components

Selecting 60 on 1 comp: would this improve performance?? And that may answer reviewer 1

Try tuning too

Classification performance on real data? Figure 2 (compare types of features selected between supervised and unsupervised methods)

Breast cancer: run the other 4 relevant methods to calculate error rate (same number of features? Compare overlap features). And run on the test set

Wang datasets: see if any test set?

Asthma: report error rate

Benchmark data sets: put here error rates and we discuss

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)?   
- explain parameter tuning: show error rate vs keepX.

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?  
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.  
3. alpha is missing in the objective of equation in p.6, l.7 and has inconsistent sub/superscripts in the equation  
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.  
5. The methods MOFA and JIVE have missing or malformatted citations in the text on p.14  
6. Typo on p.2 l.31: validation  
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.

The figure displays the number of unique features. Some features were duplicated across components.  
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.  
9. In the supplement the grid parameters for simulation are inconsistent within the text and with the Figure 1a.  
10. The correlations in Fig. S2 for the uncorrelated simulation setting seem to me still rather high. Could the authors comment on this?  
11. Fig. 3a (names of proteins) and Fig. 4f are illegible