Here are some information that I collected about batch effects

There are two functions in SVA package "comBat" and "svm"

comBat is for known batches (out case)

sva is for finding unknown batches (not our case)

combat needs log transformed data so output is log normazlized

but DeSeq needs not scaled data (they to not work together well!)

<http://seqanswers.com/forums/archive/index.php/t-46890.html>

*"If you have known batches, just include the batch variable in the design for DESeq2.*

*We don't recommend testing on transformed counts.*

*If you have unknown batches, you can use svaseq or other packages. We are writing up a workflow which will be released in a few weeks and includes svaseq and RUVSeq.*

*But briefly, add the SVA surrogate variables (columns of 'sv') to the colData, and then add these to the design. E.g., for two surrogate variables:*

*dds$SV1 <- svseq$sv[,1]*

*dds$SV2 <- svseq$sv[,2]*

*design(dds) <- ~ SV1 + SV2 + condition*

*dds <- DESeq(dds)"*

*For machine Learning*

***Michael Love***

*09-30-2014, 11:49 AM*

*limma has a function which easily removes batch effects from a matrix:  
  
http://web.mit.edu/~r/current/arch/i386\_linux26/lib/R/library/limma/html/removeBatchEffect.html  
  
(you'd want the input to be on the scale of log2 of counts, and the rlog or VST output is log2 scale)*

<https://support.bioconductor.org/p/76745/>

*While it's not the ideal experimental design (better would be to have distributed all states within each library preparation batch in a block design, or even randomized), it is still possible to analyze all the samples together using a design ~batch + state. I assume the colData looks something like this (with replicates in addition):*

*batch state*

*1 1*

*1 2*

*1 5*

*2 3*

*2 4*

*2 5*

*Be sure that these columns are factors, not numerics.*

*What happens when you run DESeq2 with a design of ~batch + state, is that it will use the samples from state 5 to estimate the batch effect. So if you only have a few samples, this can be a very noisy estimate of the batch differences for each gene, but it's the best you can do given you want to make comparisons across batch.*

*ADD REPLY •*[*link*](https://support.bioconductor.org/p/76745/#76791)*modified 5 months ago • written 5 months ago by*[*Michael Love*](https://support.bioconductor.org/u/5822/)

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*Hi Riccardo,*

*if you use exactly one SV, my proposal and Mike's approach will likely to be quite similar.*

*However, Mike's proposal is more robust as well as close to a "textbook" solution, so easy to communicate as well.*

*sva uses a quite complex algorithm, so the additionally variability caused by that might hamper the potential advantages.  So I would recommend Mike's proposal.*

*Bernd*

<https://support.bioconductor.org/p/76099/>

*Just to be clear, there's an important difference between removing a batch effect and modelling a batch effect. Including the batch in your design formula will model the batch effect in the regression step, which means that the raw data are not modified (so the batch effect is not removed), but instead the regression will estimate the size of the batch effect and subtract it out when performing all other tests. In addition, the model's residual degrees of freedom will be reduced appropriately to reflect the fact that some degrees of freedom were "spent" modelling the batch effects. This is the preferred approach for any method that is capable of using it (this includes DESeq2). You would only remove the batch effect (e.g. using limma's removeBatchEffect function) if you were going to do some kind of downstream analysis that can't model the batch effects, such as training a classifier.*

*I don't have experience with ComBat, but I would expect that you run it on log-transformed CPM values, while DESeq2 expects raw counts as input. I couldn't tell you how to properly use the two methods together.*

*modified 5 months ago • written 5 months ago by*[*Ryan C. Thompson*](https://support.bioconductor.org/u/5618/)

<https://support.bioconductor.org/p/62954/>

*Dear list,*

*I am analyzing a RNAseq dataset with a very obvious batch-effect using the DESeq2 (v1.4.5) R package. I account for the batch-effect in my model and in terms of differential expression it really improves the analysis. Now, I would like to visualize the data by principal component analysis and hierarchical clustering, which requires the adjusted values.  However, I cannot figure out how to get the batch-corrected values. Thanks you for the help!*

*Best,*

*Salah*

*hi Praful,*

*Yes, you should just add batch as another factor in the design.*

*We do not have any kind of functionality in DESeq2 for differential expression for non-count data. Using the batch within the count-based GLM is the recommended approach, as we have information about the statistical properties of the raw counts.*

*modified 15 months ago • written 15 months ago by*[*Michael Love*](https://support.bioconductor.org/u/5822/)*♦ 7.9k*

<https://www.bioconductor.org/packages/3.3/bioc/vignettes/DESeq2/inst/doc/DESeq2.pdf>

**3.12.1 Linear combinations**

The simplest case is the linear combination, or linear dependency problem, when two variables contain exactly the same information, such as in the following sample table. The software cannot fit an effect for batch and condition, because they produce identical columns in the model matrix. This is also referred to as “perfect confounding”. A unique solution of coefficients (the βi in the formula in Section 4.1) is not possible.

## batch condition

## 1 1 A

## 2 1 A

## 3 2 B

## 4 2 B

Another situation which will cause problems is when the variables are not identical, but one variable can be formed by the combination of other factor levels. In the following example, the effect of batch 2 vs 1 cannot be fit because it is identical to a column in the model matrix which represents the condition C vs A effect.

## batch condition

## 1 1 A

## 2 1 A

## 3 1 B

## 4 1 B

## 5 2 C

## 6 2 C

In both of these cases above, the batch effect cannot be fit and must be removed from the model formula. There is just no way to tell apart the condition effects and the batch effects. The options are either to assume there is no batch effect (which we know is highly unlikely given the literature on batch effects in sequencing datasets) or to repeat the experiment and properly balance the conditions across batches. A balanced design would look like:

## batch condition

## 1 1 A

## 2 1 B

## 3 1 C

## 4 2 A

## 5 2 B

## 6 2 C

Our situation is something in between.