Published online 07 October 2014

Nucleic Acids Research, 2014, Vol. 42, No. 21 **e161**
doi: 10.1093/nar/gku864

svaseq: removing batch effects and other unwanted noise from sequencing data

Jeffrey T. Leek[*]

Department of Biostatistics, Johns Hopkins Bloomberg School of Public Health Baltimore, MD 21212, US

Received June 24, 2014; Revised August 20, 2014; Accepted September 08, 2014

ABSTRACT

It is now known that unwanted noise and unmodeled artifacts such as batch effects can dramatically reduce the accuracy of statistical inference in genomic experiments. These sources of noise must be modeled and removed to accurately measure biological variability and to obtain correct statistical inference when performing high-throughput genomic analysis. We introduced surrogate variable analysis (sva) for estimating these artifacts by (i) identifying the part of the genomic data only affected by artifacts and (ii) estimating the artifacts with principal components or singular vectors of the subset of the data matrix. The resulting estimates of artifacts can be used in subsequent analyses as adjustment factors to correct analyses. Here I describe a version of the sva approach specifically created for count data or FPKMs from sequencing experiments based on appropriate data transformation. I also describe the addition of supervised sva (ssva) for using control probes to identify the part of the genomic data only affected by artifacts. I present a comparison between these versions of sva and other methods for batch effect estimation on simulated data, real count-based data and FPKM-based data. These updates are available through the sva Bioconductor package and I have made fully reproducible analysis using these methods available from: https://github.com/jtleek/svaseq.

INTRODUCTION

Batch effects and other technological artifacts introduce spurious correlation, create bias and add variability to the results of genomic experiments (1–3). The basic problem is that batch effects introduce a new source of signal into the data that can be confused with the signal an analyst is looking for. This signal is consistent across transcripts, exons or genes and so may lead to gross errors in the calculation of statistical significance, estimates of effect sizes or other statistical measures (4,5). These types of noise also prevent analysts from appropriately modeling biological variation and group-specific changes in gene expression (6). Unfortunately we rarely know all of the potential artifacts in most high-throughput experiments (4,7). In some cases, it is possible to rely on the date the samples were processed as a surrogate for unmeasured artifacts (8) and correct for them to get statistically accurate results. However, each new technology may suffer from different artifacts and it may take time for the community to discover which variables must be measured and included in an analysis (9).

In 2007 we introduced surrogate variable analysis (sva) as a conceptual approach to statistical modeling of genomic data when artifacts are unknown or unmeasured (4) (Figure 1) and subsequently improved the estimation algorithm (5) (Figure 2). We proposed modeling the data as a combination of known variables of interest, known adjustment variables and unknown and unmeasured artifacts. A simple version of this model might relate gene expression for gene i on sample j (g_{ij}) to the phenotype for that sample (y_j), the known batch variable for that sample (a_j) and an unknown artifact on that sample (u_j):

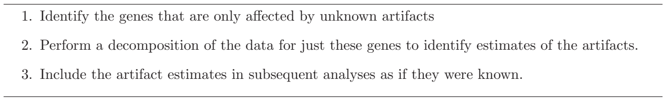
$$\underbrace{g_{ij}}_{\text{gene expression}} = \underbrace{b_{i0}}_{\text{baseline expression}} + \underbrace{b_{i1}y_j}_{\text{phenotype effect}} + \underbrace{c_i a_j}_{\text{known batch}} + \underbrace{d_i u_j}_{\text{unknown artifact}} + \underbrace{e_{ij}}_{\text{meas. error}} \tag{1}$$

If only a single gene is measured, it is difficult to estimate the unknown artifact (u_j) from the data directly, since all the coefficients (b, c, d) are unknown. But we noticed (4) that if many genes are measured, it is possible that for some genes the coefficients for b_{i1} and c_i may be equal to zero. For these specific genes the model reduces to:

$$\underbrace{g_{ij}}_{\text{gene expression}} = \underbrace{b_{i0}}_{\text{baseline expression}} + \underbrace{d_i u_j}_{\text{unknown artifact}} + \underbrace{e_{ij}}_{\text{measurement error}}$$

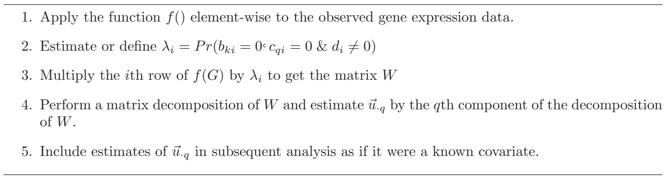
Our next insight was that even though d_i and u_j are unknown, you do not need to know either of them exactly to get correct statistical inference for the parameters for the

[*] To whom correspondence should be addressed. Tel: +1 410 955 1166; Fax: +1 410 955 0958; Email: jtleek@gmail.com



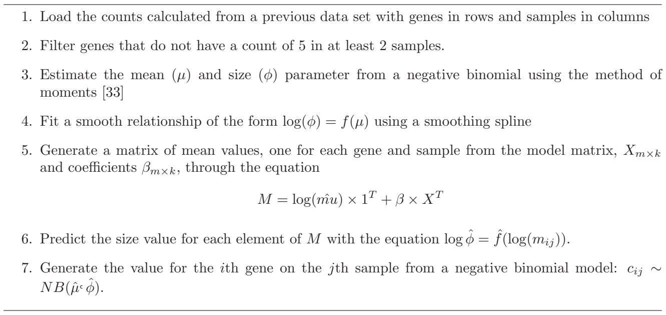
1. Identify the genes that are only affected by unknown artifacts
2. Perform a decomposition of the data for just these genes to identify estimates of the artifacts.
3. Include the artifact estimates in subsequent analyses as if they were known.

Figure 1. Surrogate variable analysis (sva). The general sva framework for identifying unknown artifacts in genomic data has three steps (4,5).



1. Apply the function $f()$ element-wise to the observed gene expression data.
2. Estimate or define $\lambda_i = Pr(b_{ki} = 0\text{ʿ}\, c_{qi} = 0 \;\&\; d_i \neq 0)$
3. Multiply the ith row of $f(G)$ by λ_i to get the matrix W
4. Perform a matrix decomposition of W and estimate $\vec{u}_{\cdot q}$ by the qth component of the decomposition of W.
5. Include estimates of $\vec{u}_{\cdot q}$ in subsequent analysis as if it were a known covariate.

Figure 2. General sva estimation framework. In this general framework, Step 1 allows for transformations specific to different data types, Step 2 allows for either estimating or defining the probabilities of being affected by unknown artifacts but not known variables and Step 4 allows for a variety of matrix decompositions and factor analysis approaches.



1. Load the counts calculated from a previous data set with genes in rows and samples in columns
2. Filter genes that do not have a count of 5 in at least 2 samples.
3. Estimate the mean (μ) and size (ϕ) parameter from a negative binomial using the method of moments [33]
4. Fit a smooth relationship of the form $\log(\phi) = f(\mu)$ using a smoothing spline
5. Generate a matrix of mean values, one for each gene and sample from the model matrix, $X_{m\times k}$ and coefficients $\beta_{m\times k}$, through the equation

$$M = \log(\hat{mu}) \times 1^T + \beta \times X^T$$

6. Predict the size value for each element of M with the equation $\log\hat{\phi} = \hat{f}(\log(m_{ij}))$.
7. Generate the value for the ith gene on the jth sample from a negative binomial model: $c_{ij} \sim NB(\hat{\mu}\text{ʿ}\,\hat{\phi})$.

Figure 3. Approach for simulating RNA-seq data with Polyester package (34).

phenotype variable b_{i1}, you just have to know their linear combination $d_i \times u_j$ (4,5). We showed that if you collect the data for all genes where there is no effect of phenotype or known batch ($b_{i1} = 0$ and $c_i = 0$) and subtracted the mean of each gene to remove the baseline effect, the matrix form of the model is:

$$\underbrace{G}_{m_u\times n} = \underbrace{\vec{d}}_{m_u\times 1}\underbrace{\vec{u}}_{1\times n} + \underbrace{E}_{m_u\times n}$$

where m_u is the number of genes where $b_{1i} = 0$ and $c_i = 0$ and n is the sample size. We then pointed out that you could apply a matrix decomposition like the singular value decomposition or principal components analysis to this subset of the data to estimate surrogates for the unknown artifacts u. Later we also showed that if the number of genes that are not affected by phenotype or known batch ($b_{1i} = 0$ and $c = 0$) but are affected by unknown artifacts (d_i ? 0) is large enough, you can obtain consistent estimates of a linear transformation of u_i (10). You can then include the estimate of batch effects in downstream models to remove artifacts, dependence between genes and improve statistical inference (4,5). These insights inspired the general form of the sva approach shown in Figure 1. The difference between the two-step and iteratively re-weighted sva algorithms is the approach to estimating the genes affected by unknown artifacts (Step 1) (4,5).

In this paper I discuss two new extensions of the sva approach for sequencing data. The first extension deals with Step 1 of the sva approach (Figure 2). The idea is to use external information to estimate the genes that are likely to be affected by unknown artifacts. The original sva algorithm attempted to identify genes affected by artifacts directly from the data itself, but sometimes there is external information about which genes are unlikely to be differential expressed. This external information could be control probes (7) or estimates of batch-related probes from previous studies (11). Supervised sva uses this information in Step 1 of the sva approach, reducing computational time and reliance on estimation procedures for identifying the right genes to use for artifact estimation. The second idea, svaseq, is based on performing an appropriate transformation of the count or Fragments Per Kilobase Of Exon Per Million Fragments Mapped (FPKM) data during Steps 1 and 2 of the sva approach (12). Here I focus on the moderated log transform, which has been widely adopted both for the analysis of sequence count data and FPKM estimates. I then perform a thorough and reproducible comparison between the standard methods for removing batch effects from sequencing data. I demonstrate that svaseq and supervised sva perform comparably to existing approaches for removing batch effects from sequencing data.

MATERIALS AND METHODS

General form of the surrogate variable analysis mathematical model

The general form of the simple model in Equation (1) is

$$E[f(g_{ij})|\vec{y}, \vec{b}, \vec{u}, \vec{c}, \vec{d}, \vec{a}])$$
$$= b_{0i} + \sum_{k=1}^{K} b_{ki}y_{kj} + \sum_{\ell=1}^{L} c_{\ell i}a_{\ell j} + \sum_{q=1}^{Q} d_{qi}u_{qj} \qquad (2)$$

or in the matrix form is

$$E[f(G)]) = \underbrace{BY}_{\text{phenotype}} + \underbrace{CA}_{\text{artifacts}} + \underbrace{DU}_{\text{unknown artifacts}}$$

where the function *f*() has been applied component wise to each element of G and there may be multiple phenotypes, artifacts or unknown artifacts and I have dropped the explicit conditioning for ease of notation. The general sva algorithm then proceeds in the following four concrete steps.

In Step 5, the covariates may be included in a standard linear regression modeling analysis on an appropriately transformed scale or the covariates can be directly used in software that models counts with generalized linear models (GLMs) including edgeR (13) and DESeq (14). They can be directly included in these models as they are estimated on the same scale as standard link functions for GLMs.

Relationship of surrogate variable analysis to other approaches

The general form of the sva algorithm relies on the idea that there is a subset of genes, probes or transcripts that are affected by unknown batch effects or other artifacts

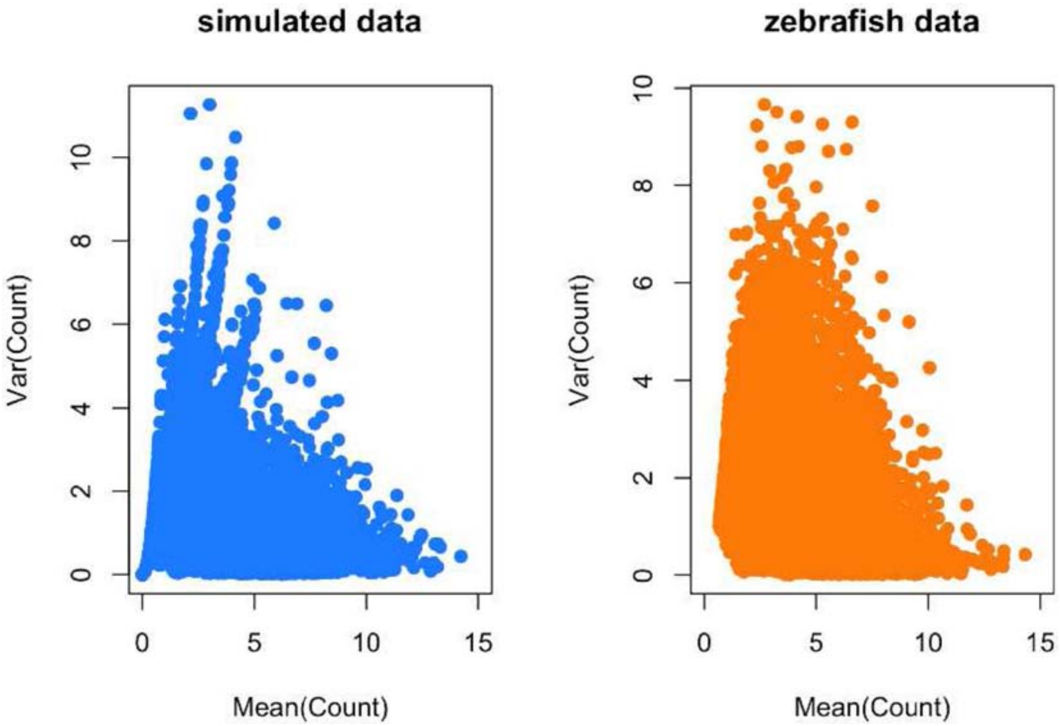


Figure 4. Distribution of means and variances for simulated and real Zebrafish data. To confirm that my simulation procedure produced reasonable simulated counts, I plotted the gene-specific means and variances for (**left panel**) the simulated data set and (**right panel**) the observed Zebrafish data set. The two distributions are qualitatively similar. Additional checks on the simulation procedure are provided in the simulated data analysis at http://jtleek.com/svaseq/simulateData.html.

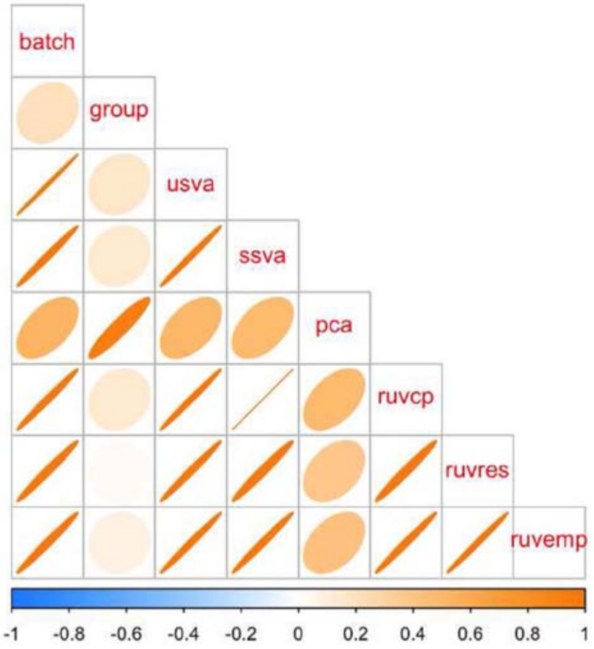


Figure 5. Correlation between simulated batch and group variables and various batch estimates. Light circles indicate low correlation and dark, tight ellipses indicate high correlation. In this case, all estimates that respect multiple sources of signal (sva and RUV based) methods are highly correlated with the simulated batch effect. Principal components estimates a linear combination of the group and batch variable and has lower concordance with the true simulated batch and the other estimates. Additional details at http://jtleek.com/svaseq/simulateData.html.

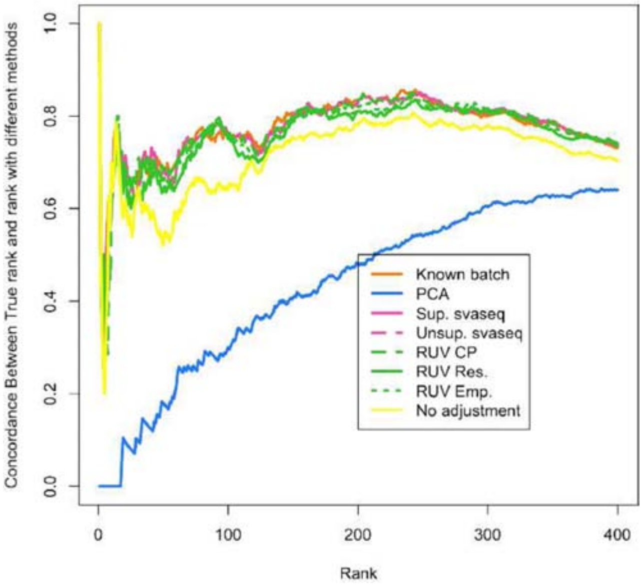


Figure 6. Differential expression results for simulated data. A concordance at the top plot (CAT plot) shows the fraction of DE results that are concordant between the analysis with the true batch and the analyses using different batch estimates. Supervised (pink solid) and unsupervised (pink dotted) sva for sequencing, RUV with control probes (green dashed), RUV with empirical controls (green dotted) and residual RUV (green solid) all outperform not adjusting for batch effects (yellow) while principal components analysis (blue) performs worse than no adjustment. Additional details at http://jtleek.com/svaseq/simulateData.html.

but are not affected by the biological relationship of interest. This is the main way that a sva approach is distinguished from a standard principal component regression. Standard principal component-based regression methods, such as EIGENSTRAT (15), are sufficient when the number of probes, genes or features expected to show a signal is small. Then the principal components will be consistent estimates of a linear transformation of the artifacts and not the phenotype/outcome of interest (16).

A number of extensions and variations on the sva algorithm have been introduced. For Step 1 in the sva algorithm, identifying probes only associated with unmeasured artifacts, it has been proposed to use control probes (7,12). For Step 2 of the sva algorithm, estimating latent factors only associated with unmeasured artifacts, it has been proposed to use factor analysis (17), independent components analysis (18) and principal components analysis (19). Another extension of the surrogate variable approach in Step 2 has been to model known sources of technical or biological covariation between the measurements for probes, for example in eQTL studies (20,21).

Supervised sva (ssva)

Supervised sva (ssva) sets $\lambda_i = 1$ for all negative controls and $\lambda_i = 0$ for all other genes in Step 2 of the sva algorithm. The assumption is that control probes will capture all of the variation due to unknown artifacts and none of the variation due to the phenotype. Control probes may miss biological artifacts. For example, we showed that trans-eQTL that are associated with multiple gene expression levels may act like an artifact when measuring the association between gene expression and phenotype (4). These artifacts may be missed by the ssva approach. However ssva is particularly useful for unfortunate experimental designs where the phenotype variable and unknown artifacts are highly correlated (8), making empirical estimates unstable (7).

Moderated log link sva (svaseq)

The second extension involves the choice of function $f()$ in (1). In our original work, we used the identity function for data measured on an approximately symmetric and continuous scale. For sequencing data, which are often represented as counts, a more suitable model may involve the use of a moderated log function (22,23). For example in Step 1 of the algorithm we may first transform the gene expression measurements by applying the function $log(g_{ij} + c)$ for a small positive constant. In the analyses that follow I will set $c = 1$. The choice of the moderating constant is an important one and is beyond the scope of this manuscript. Intuitively a choice of $c = 0$ corresponds to no moderation and as c increases you decrease the variation in the data. After performing Steps 1–5 of the sva estimation algorithm, the estimated covariates are included in downstream models as adjustment variables. For the analyses that follow, I will use the limma package (24) with the voom method (25) for differential expression analysis. The voom method is an approach for estimating the mean–variance relationship when performing differential expression analysis on sequencing experiments.

Combining svaseq and ssva

Supervised svaseq proceeds by applying the transformation $log(g_{ij} + c)$ to the gene expression count data in Step 1 and setting $\lambda_i = 1$ for all negative controls and $\lambda_i = 0$ for all other genes in Step 2 of the sva algorithm.

Zebrafish data

I use data from Zebrafish sensory neurons with three control samples and three gallein treated samples as the comparison groups (26). These data are available as part of the *zebrafishRNASeq* Bioconductor package. I loaded the data and filtered as described in the removing unwanted variation in sequencing data (RUVSeq) package. Then I estimated batch effects using supervised and unsupervised sva for sequencing, principal components analysis, RUV with control probes, RUV with empirical controls and residual RUV. I compared the model estimates and I compared differential expression analysis results when each of the different batch effect estimates was included in the model in place of the study variable.

ReCount data

ReCount is a database of pre-processed RNA-sequencing data, processed to be comparable across samples (27). In this analysis, I downloaded pre-counted RNA-sequencing datasets measuring gene expression in two separate Hapmap populations (28,29). For my analysis, I downloaded the count data from ReCount and downloaded the pedigree information from the Hapmap website. I then performed differential expression analyses looking for differences in expression between males and females and estimated unknown latent structure. I calculated estimates of batch effects using unsupervised sva for sequencing, principal components analysis, RUV with empirical control probes, and RUV on residuals. I compared the estimates to the variable indicating whether the data came from the Pickrell or Montgomery study. I compared two scenarios, one where the sex and study variables were balanced and one where they were imbalanced. I also compared the differential expression analysis results when each of the different batch effect estimates was included in the model in place of the study variable.

GEUVADIS data

I downloaded the processed GEUVADIS (30,31) Ballgown object (32) from:

https://github.com/alyssafrazee/ballgown_code

I then subset the data to only the non-duplicated samples (31) and performed a differential expression analysis comparing populations. I calculated estimates of batch effects using unsupervised sva for sequencing, principal components analysis, RUV with empirical control probes and RUV on residuals. I compared the estimates to the variable indicating which lab the sequencing was performed in. I also compared the differential expression analysis results when each of the different batch effect estimates was included in the model in place of the laboratory variable.

Simulating data

I simulated data from a negative binomial model for count based RNA-sequencing data (Figure 3) (33). For complete details see the simulated data R markdown document and accompanying HTML file.

I estimated the model parameters from the Zebrafish data described above. I simulated two scenarios, one where the group and batch variable were not correlated and one where they were correlated. Here we consider both batch effects that are correlated with the outcome and batch effects that are orthogonal. This is a critical distinction as unsupervised methods that estimate batch effects directly from the data will often perform worse when batch and outcome are correlated unless this relationship is explicitly modeled. Data were simulated with the *Polyester* R package (34). Then I estimated batch effects using supervised and unsupervised sva for sequencing, principal components analysis, RUV with control probes, RUV with empirical controls and residual RUV. I compared the model estimates to the true simulated batch variable and I also compared differential expression analysis results when each of the different batch effect estimates was included in the model in place of the study variable.

Code and availability

ssva and svaseq are currently implemented in the sva Bioconductor package version 3.11.2 or greater (http://www.bioconductor.org/packages/devel/bioc/html/sva.html). All data and code used to perform this analysis are available as R markdown files (35) available from: https://github.com/jtleek/svaseq. You can view the individual analyses as webpages at:

(i) Zebrafish analysis: http://jtleek.com/svaseq/zebrafish.html
(ii) ReCount analysis: http://jtleek.com/svaseq/recount.html
(iii) GEUVADIS analysis: http://jtleek.com/svaseq/geuvadis.html
(iv) Simulated data analysis: http://jtleek.com/svaseq/simulateData.html

RESULTS

Simulated data

I estimated simulation parameters from the Zebrafish data as described in the methods. I then performed several checks to confirm that (i) the data generated by the simulated model recapitulated the qualitative behavior of the data used to estimate the model parameters (Figure 4), (ii) that data generated without signal did not show statistically significant results, (iii) data could be simulated with differential expression signal and (iv) that data with batch effects displayed the expected conservative bias of *P*-values (36) (See supplementary analysis files (http://jtleek.com/svaseq/zebrafish.html).

In the Zebrafish study there was a single simulated batch effect. I simulated two scenarios, in the first scenario the batch effect and the group effect had low correlation. As expected, all methods that aim to estimate batch effects while taking into account multiple sources of signal (svaseq and RUV methods) produce estimates that are highly correlated with the simulated batch effect. The estimate of batch based on principal components is biased, because the principal component is estimating a linear combination of the group and batch variable (Figure 5). In this scenario all the *P*-value distributions are approximately correct, with the exception of the analysis using principal component based estimates of batch. This is because the principal components do not estimate accurate versions of the batch effect and bias the statistical significance calculation (see: http://jtleek.com/svaseq/simulateData.html for plots).

I next fit models relating the gene expression counts to the simulated group phenotype (p) and batch effect estimates ($\hat{u}$) using the following model:

$$log(g_{ij}+1)|\vec{y},\vec{b},\vec{u},\vec{d} = b_{0i} + b_{1i}y_j + d_{1i}\hat{u}_j + e_{ij} \quad (3)$$

I accounted for the potential relationship between mean and variance using the voom method (25). I then estimated how concordant the differential expression results were with the results we obtained when we fit model (6) using the true simulated batch variable using concordance at the top plots (CAT plots, Figure 6) (37). For each ranking, these plots show the fraction of results that are the same between the analysis using the true batch variable and the batch variable estimated with different methods. Supervised and unsupervised sva for sequencing, RUV with control probes, RUV with empirical controls and residual RUV all outperform not adjusting for batch effects while principal components analysis performs worse than no adjustment. The reason is that the principal component estimate is correlated with group and absorbs some of the signal due to that variable.

We performed an identical analysis where the batch was now correlated with the group variable. Qualitatively similar results hold in this second simulated scenario with one exception. The empirical RUV methods attempt to define control probes by identifying genes that do not show differential expression with respect to batch. But when batch and group are correlated, this may also through away genes that show signal with respect to the group variable. Similar the residual RUV approach estimates the batch variable after taking the residuals from the model fit of the counts on the group variable. However, when batch and group are correlated, this again removes batch signal and leads to slightly lower performance of the RUV approaches (4). Unsupervised svaseq does not use the control probes but avoids some of these difficulties by iteratively identifying probes associated with group but not associated with batch (5) (Figure 7). The *P*-value histograms here show a strong difference between supervised and unsupervised approaches. The unsupervised approaches attempt to estimate the artifacts, but they are correlated with the group. Since the estimates are off, the statistical significance calculations are not correct (see: http://jtleek.com/svaseq/simulateData.html for plots). But the supervised methods correct the statistical significance calculations accurately.

Zebrafish data

Next I performed an analysis on the Zebrafish data as described in the methods section. Here, the batch variable is

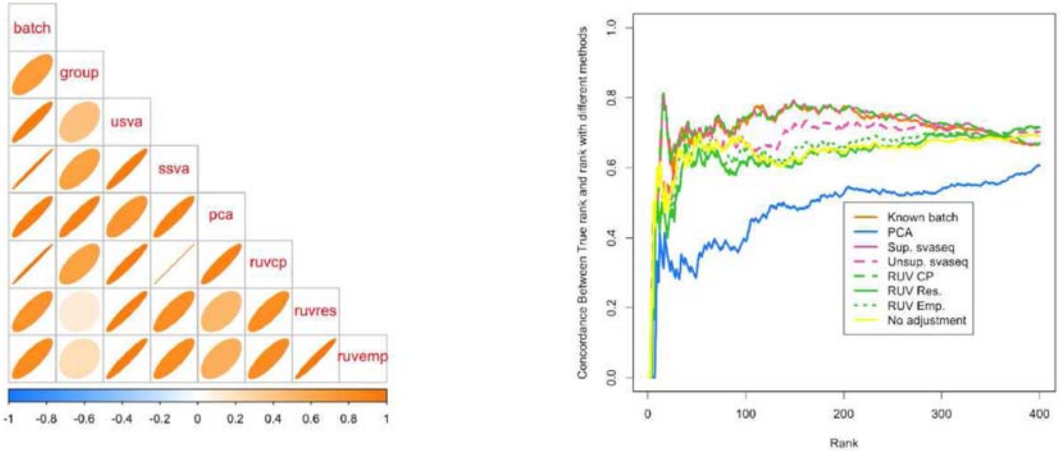


Figure 7. Comparison of batch effect results when group and batch are correlated. (**a**) A plot of the correlation between the different batch estimates and the batch variable analogous to Figure 5. (**b**) A concordance at the top plot measuring concordance between the analysis using the true batch variable and the various estimates analogous to Figure 6. Here the unsupervised RUV approaches using empirical control probes and residuals perform worse than no adjustment, because the methods can not distinguish signal from the known group variable and the unknown batch variable. Additional details at http://jtleek.com/svaseq/simulateData.html.

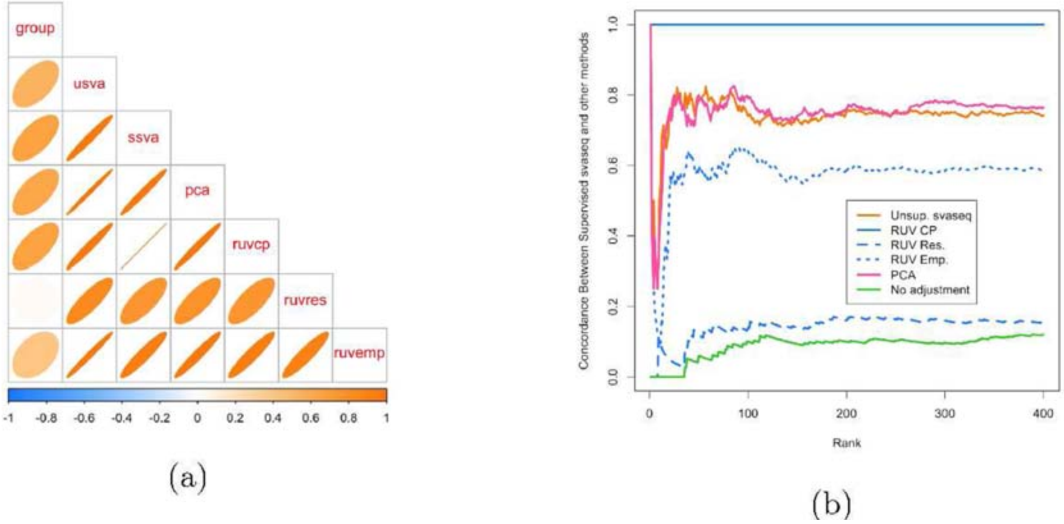


Figure 8. Comparison of batch effect results on Zebrafish data. (**a**) A plot of the correlation between the different batch estimates analogous to Figure 5, but with no gold standard. (**b**) A concordance at the top plot measuring concordance between the analysis using the supervised SVA estimates and the various other batch estimates analogous to Figure 6. The control probes RUV approach (blue solid in (b)) and supervised sva approach produce identical results. The unsupervised sva (orange solid) and principal components (pink solid) approaches are most similar to the supervised estimates in this scenario. Additional details at http://jtleek.com/svaseq/zebrafish.html.

not known, but we do have negative control probes which can be used to estimate the batch effects. When comparing the batch estimates, I noted that the supervised sva estimates and the RUV control probes estimates were perfectly correlated ($R^2 = 1$) and that they produced identical differential expression results (Figure 8a). The unsupervised sva and principal components approaches are most similar to the supervised estimates from SVA or RUV for the Zebrafish data.

ReCount data

For the ReCount data I generated an artificial batch effect by combining the data from two different studies of gene expression in two different populations (28,29). I used sex as the outcome variable in the analysis and then estimated batch effects using the same set of proposed approaches. I next fit models relating the gene expression counts (g) to sex variable (p, phenotype variable representing sex) and batch effect estimates ($\hat{u}$, representing the study the samples came from) using the following model:

$$log(g_{ij} + 1)|\vec{p}, \vec{b}, \vec{u} = b_{0i} + b_{1i}y_j + d_{1i}\hat{u}_j + e_{ij} \qquad (4)$$

I accounted for the potential relationship between mean and variance using the voom method (25). I then estimated how concordant the differential expression results were with the results we obtained when we fit model 6 using the true simulated batch variable using concordance at the top plots (Figure 8b) (37).

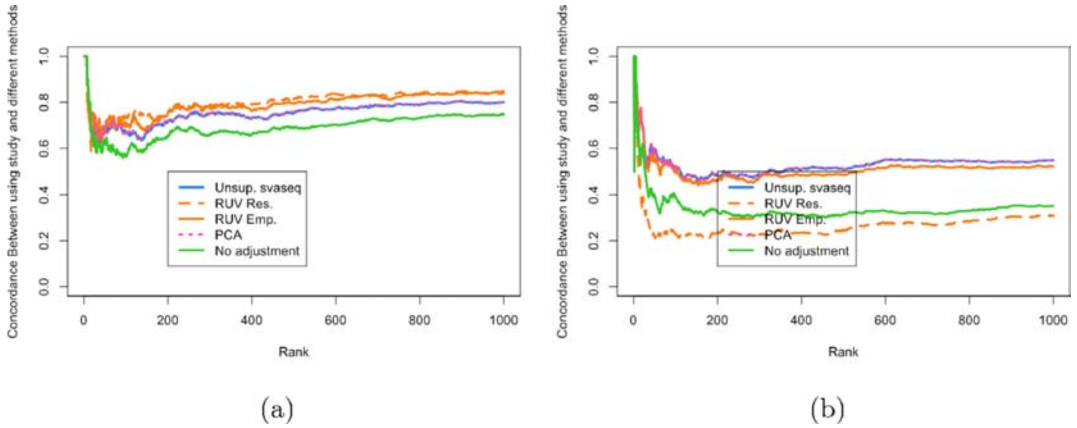


Figure 9. Comparison of differential expression results for ReCount experiment. (**a**) A concordance at the top plot measuring concordance between the analysis using the true study and the various other batch estimates analogous to Figure 6. (**b**) A concordance at the top plot measuring concordance between the analysis using the true study and the various other batch estimates analogous to Figure 6 when data were resampled to make the sex and study variables moderately correlated ($r^2 = 0.33$.) When sex and study are uncorrelated, RUV performs slightly better and when sex and study are correlated, svaseq performs slightly better. Additional details at http://jtleek.com/svaseq/recount.html.

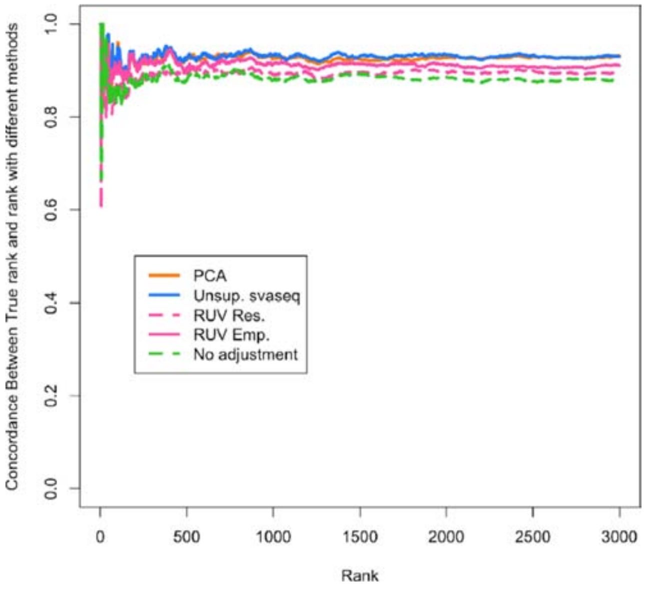


Figure 10. Differential expression results for GEUVADIS data. A concordance at the top plot (CAT plot) shows the fraction of DE results that are concordant between the analysis with the true laboratory and the analyses using different batch estimates. Unsupervised sva for sequencing (blue) and principal components analysis (orange) outperform the RUV based methods (pink) and no batch adjustment (green). Additional details at http://jtleek.com/svaseq/geuvadis.html.

In the original data, the batch effect and the group variable are nearly perfectly orthogonal. In this situation the empirical and residual RUV approaches produce estimates of the batch variable more highly correlated than the unsupervised svaseq approach (see http://jtleek.com/svaseq/recount.html) and produce correspondingly more similar differential expression results to using the true study variable as an adjustment in the differential expression analysis (Figure 9a). However, I next re-sampled the data to mimic a scenario where the group and batch variable showed modest correlation ($r^2 = 0.33$). In this scenario the unsupervised sva and principal components analysis approaches outperform the empirical control RUV approach. The residual RUV approach performs worse than no adjustment for study, because signal due to the batch variable was removed when the residuals from the model relating sex to phenotype was calculated (Figure 9b).

GEUVADIS data

The *Ballgown* R package https://github.com/alyssafrazee/ballgown (32) can be used to analyze abundance data from assembled transcriptome data from *Cufflinks* (38). I loaded data from the GEUVADIS project (30,31) that we recently processed using *Cufflinks* and *Ballgown* (32). I selected only the non-duplicated samples and performed a differential expression analysis comparing different populations. I then compared the estimated batch effects using the various approaches to the known lab where the samples were processed, one of the variables that showed the highest association with assembled transcript levels (31).

I assessed concordance between the batch effect estimates and the lab variable by fitting the model:

$$\hat{u}_j = b_0 + \sum_{k=1}^{K} b_k 1(\text{Sample j belongs to lab k}) + e_j \qquad (5)$$

and then performed an ANOVA to compare the model including lab to the null model of no association with lab. The unsupervised sva and principal components estimates showed significantly higher F-statistics for concordance (482 and 456, respectively) compared to the RUV approach (106 and 109 for RUV residual and empirical, respectively). I next fit models relating the gene expression counts (g) to the population phenotype (p, phenotype variable representing population) and batch effect estimates ($\hat{u}$, representing the study the samples came from) using the following model:

$$log(g_{ij} + 1)|\vec{p}, \vec{b}, \vec{u} = b_{0i} + b_{1i} y_j + d_{1i} \hat{u}_j + e_{ij} \qquad (6)$$

I compared the results to the differential expression model where I included the known lab variable as an adjustment in place of $\hat{u}$. The svaseq and principal components adjusted analyses showed greater concordance with the lab adjusted analysis, as expected since the batch estimates were more highly correlated with this known variable (Figure 10).

DISCUSSION

Here I have described the general sva framework and I have introduced two extensions of the sva approach. The first takes advantage of known control probes to simplify the sva algorithm and the second addresses the distribution of count and FPKM data typically observed in sequencing experiments. The question of whether to use FPKM or count based approaches for the analysis of RNA-sequencing data is beyond the scope of this paper. However, I have demonstrated in this paper that regardless of the choice for measurement summary, svaseq can be applied to remove batch effects.

I have shown that sva-based approaches perform comparably to other batch effect estimation procedures for sequencing when the group and unknown batch variables are uncorrelated and outperform other approaches when the batch and group variable are correlated. These extensions are currently available from the devel branch of the sva software http://bioconductor.org/packages/devel/bioc/html/sva.html and all analyses are fully reproducible and available as R markdown documents from https://github.com/jtleek/svaseq.

FUNDING

National Institutes of Health (NIH) [R01 GM10570502, GM083084 to J.L.]. Funding for open access charge: NIH [R01 GM10570502].

Conflict of interest statement. None declared.

REFERENCES

1. Akey,J.M., Biswas,S., Leek,J.T. and Storey,J.D. (2007) On the design and analysis of gene expression studies in human populations. *Nat. Genet.*, **39**, 807–808.
2. Sebastiani,P., Solovieff,N., Puca,A., Hartley,S.W., Melista,E., Andersen,S., Dworkis,D.A., Wilk,J.B., Myers,R.H., Steinberg,M.H. *et al.* (2010) Genetic signatures of exceptional longevity in humans. *Science*, **2010**, doi:10.1126/science.1190532.
3. Lambert,C.G. and Black,L.J. (2012) Learning from our GWAS mistakes: from experimental design to scientific method. *Biostatistics*, **13**, 195–203.
4. Leek,J. and Storey,J. (2007) Capturing heterogeneity in gene expression studies by ‘Surrogate Variable Analysis’. *PLoS Genet.*, **3**, e161.
5. Leek,J. and Storey,J. (2008) A general framework for multiple testing dependence. *PNAS*, **105**, 18718–18723.
6. Hansen,K.D., Wu,Z., Irizarry,R.A. and Leek,J.T. (2011) Sequencing technology does not eliminate biological variability. *Nat. Biotechnol.*, **29**, 572–573.
7. Gagnon-Bartsch,J.A. and Speed,T.P. (2012) Using control genes to correct for unwanted variation in microarray data. *Biostatistics*, **13**, 539–552.
8. Leek,J.T., Scharpf,R.B., Bravo,H.C., Simcha,D., Langmead,B., Johnson,W.E., Geman,D., Baggerly,K. and Irizarry,R.A. (2010) Tackling the widespread and critical impact of batch effects in high-throughput data. *Nat. Rev. Genet.*, **11**, 733–739.
9. Kircher,M., Heyn,P. and Kelso,J. (2011) Addressing challenges in the production and analysis of illumina sequencing data. *BMC Genomics*, **12**, 382.
10. Leek,J. (2011) Asymptotic conditional singular value decomposition for high-dimensional genomic data. *Biometrics*, **67**, 344–352.
11. Parker,H.S., Bravo,H.C. and Leek,J.T. (2014) Removing batch effects for prediction problems with frozen surrogate variable analysis. Peer J., **2**, e561.
12. Risso,D., Ngai,J., Speed,T. and Dudoit,S. (2014) Normalization of RNA-seq data using factor analysis of control genes or samples. Nat. Biotechnol., **32**, 896–902.
13. Robinson,M.D., McCarthy,D.J. and Smyth,G.K. (2010) edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, **26**, 139–140.
14. Anders,S. and Huber,W. (2010) Differential expression analysis for sequence count data. *Genome Biol.*, **11**, R106.
15. Price,A.L., Patterson,N.J., Plenge,R.M., Weinblatt,M.E., Shadick,N.A. and Reich,D. (2006) Principal components analysis corrects for stratification in genome-wide association studies. *Nat. Genet.*, **38**, 904–909.
16. Anderson,T.W. (1963) Asymptotic theory for principal component analysis. *Ann. Math. Stat.*, **34**, 122–148.
17. Friguet,C., Kloareg,M. and Causeur,D. (2009) A factor model approach to multiple testing under dependence. *J. Am. Stat. Assoc.*, **104**, 1406–1415.
18. Teschendorff,A.E., Zhuang,J. and Widschwendter,M. (2011) Independent surrogate variable analysis to deconvolve confounding factors in large-scale microarray profiling studies. *Bioinformatics*, **27**, 1496–1505.
19. Fan,J., Han,X. and Gu,W. (2012) Estimating false discovery proportion under arbitrary covariance dependence. *J. Am. Stat. Assoc.*, **107**, 1019–1035.
20. Listgarten,J., Kadie,C., Schadt,E.E. and Heckerman,D. (2010) Correction for hidden confounders in the genetic analysis of gene expression. *Proc. Natl. Acad. Sci. U.S.A.*, **107**, 16465–16470.
21. Stegle,O., Parts,L., Durbin,R. and Winn,J. (2010) A Bayesian framework to account for complex non-genetic factors in gene expression levels greatly increases power in eQTL studies. *PLoS Comput. Biol.*, **6**, e1000770.
22. Frazee,A.C., Sabunciyan,S., Hansen,K.D., Irizarry,R.A. and Leek,J.T. (2014) Differential expression analysis of RNA-seq data at single-base resolution. Biostatistics, **15**, 413–426.
23. Bullard,J.H., Purdom,E., Hansen,K.D. and Dudoit,S. (2010) Evaluation of statistical methods for normalization and differential expression in mRNA-Seq experiments. *BMC Bioinformatics*, **11**, 94.
24. Smyth,G.K. (2005) Limma: linear models for microarray data. In: *Bioinformatics and Computational Biology Solutions Using R and Bioconductor*. Springer, New York, pp. 397–420.
25. Law,C.W., Chen,Y., Shi,W. and Smyth,G.K. (2014) Voom: precision weights unlock linear model analysis tools for RNA-seq read counts. Genome Biol., **15**, R29.
26. Ferreira,T., Wilson,S.R., Choi,Y.G., Risso,D., Dudoit,S., Speed,T.P. and Ngai,J. (2014) Silencing of odorant receptor genes by G Protein βγ signaling ensures the expression of one odorant receptor per olfactory sensory neuron. *Neuron*, **81**, 847–859.
27. Frazee,A.C., Langmead,B. and Leek,J.T. (2011) ReCount: a multi-experiment resource of analysis-ready RNA-seq gene count datasets. *BMC Bioinformatics*, **12**, 449.
28. Pickrell,J., Marioni,J., Pai,A., Degner,J., Engelhardt,B., Nkadori,E., Veyrieras,J., Stephens,M., Gilad,Y. and Pritchard,J. (2010) Understanding mechanisms underlying human gene expression variation with RNA sequencing. *Nature*, **464**, 768–772.
29. Montgomery,S., Sammeth,M., Gutierrez-Arcelus,M., Lach,R., Ingle,C., Nisbett,J., Guigo,R. and Dermitzakis,E. (2010) Transcriptome genetics using second generation sequencing in a Caucasian population. *Nature*, **464**, 773–777.
30. Lappalainen,T., Sammeth,M., Friedländer,M.R., AC't Hoen,P., Monlong,J., Rivas,M.A., Gonzàlez-Porta,M., Kurbatova,N., Griebel,T., Ferreira,P.G. *et al.* (2013) Transcriptome and genome sequencing uncovers functional variation in humans. *Nature*, **501**, 506–511.
31. AC't Hoen,P., Friedländer,M.R., Almlöf,J., Sammeth,M., Pulyakhina,I., Anvar,S.Y., Laros,J.F., Buermans,H.P., Karlberg,O., Brännvall,M. *et al.* (2013) Reproducibility of high-throughput

mRNA and small RNA sequencing across laboratories. *Nat. Biotechnol.*, **31**, 1015–1022.

32. Frazee,A.C., Pertea,G., Jaffe,A.E., Langmead,B., Salzberg,S.L. and Leek,J.T. (2014) Flexible isoform-level differential expression analysis with Ballgown. bioRxiv, doi:10.1101/003665.
33. Savani,V. and Zhigljavsky,A. (2006) Efficient estimation of parameters of the negative binomial distribution. *Commun. Stat.—Theory Methods*, **35**, 767–783.
34. Frazee,A.C., Jaffe,A.E., Langmead,B. and Leek,J. (2014) Polyester: simulating RNA-seq datasets with differential transcript expression. bioRxiv, doi:10.1101/006015.
35. Xie,Y. (2014) knitr: a comprehensive tool for reproducible research in R. In: Victoria,S, Friedrich,L and Roger,DP (eds). *Implementing Reproducible Research*. CRC Press, Boca Raton, pp. 3–30.
36. Leek,J. and Storey,J.D. (2011) The joint null criterion for multiple hypothesis tests. *Stat. Appl. Genet. Mol. Biol.*, **10**, 28.
37. Irizarry,R.A., Warren,D., Spencer,F., Kim,I.F., Biswal,S., Frank,B.C., Gabrielson,E., Garcia,J.G., Geoghegan,J., Germino,G. *et al.* (2005) Multiple-laboratory comparison of microarray platforms. *Nat. Methods*, **2**, 345–350.
38. Trapnell,C., Roberts,A., Goff,L., Pertea,G., Kim,D., Kelley,D.R., Pimentel,H., Salzberg,S.L., Rinn,J.L. and Pachter,L. (2012) Differential gene and transcript expression analysis of RNA-seq experiments with TopHat and Cufflinks. *Nat. Protoc.*, **7**, 562–578.