Batch effect

Mikhail Dozmorov
Fall 2017

Batch effects

- Batch effects are widespread in high-throughput biology. They are artifacts not related to the biological variation of scientific interests.

- For instance, two microarray experiments on the same technical replicates processed on two different days might present different results due to factors such as room temperature or the two technicians who did the two experiments.

- Batch effects can substantially confound the downstream analysis, especially meta-analysis across studies.
Batch sources

http://bib.oxfordjournals.org/content/14/4/469.long
The effect of batch removal

http://bib.oxfordjournals.org/content/14/4/469.long

Accounting for batch effects

- In statistical modeling, batch effects can be included as covariates (additional predictors) in the model (preferred method).
- For exploratory analysis, we often attempt to “eliminate” or “adjust for” such unwanted variation in advance, by subtracting the estimated effect from each variable (ComBat, SVA).
**Batch removal methods**

Two main approaches:

- Location-scale (LS)
- Matrix-factorization (MF)

---

**Batch removal methods**

Location-scale

- LS method assumes a model for the location (mean) and/or scale (variance) of the data within the batches.

- Adjusts the batches in order to agree with these models
Batch removal methods

Matrix-factorization

- MF techniques assume that the variation in the data corresponding to batch effects is independent on the variation corresponding to the biological variable of interest
- Capture non-biological variability in a small set of factors
- Factors can be estimated through some matrix factorization methods

ComBat

- “Eliminate” the impact of the (known) batch variable on the observed values
- Can provide information about variables of interest, whose effect should be retained in the data
- Works best if batch and variable of interest are not confounded
ComBat

**ComBat - Location-scale method**

The core idea of ComBat was that the observed measurement $Y_{ijg}$ for the expression value of gene $g$ for sample $j$ from batch $i$ can be expressed as

$$Y_{ijg} = \alpha_g + X\beta_g + \gamma_{ig} + \delta_{ig}e_{ijg}$$

where $X$ consists of covariates of scientific interests, while $\gamma_{ig}$ and $\delta_{ig}$ characterize the additive and multiplicative batch effects of batch $i$ for gene $g$.

https://www.bu.edu/jlab/wp-assets/ComBat/Abstract.html

ComBat

After obtaining the estimators from the above linear regression, the raw data $Y_{ijg}$ can be adjusted to $Y_{ijg}^*$:

$$Y_{ijg}^* = Y_{ijg} - \hat{\delta}_{ig} - X\hat{\beta}_g - \hat{\gamma}_{ig} + \hat{\alpha}_g + X\hat{\beta}_g$$

For real application, an empirical Bayes method was applied for parameter estimation.

https://www.bu.edu/jlab/wp-assets/ComBat/Abstract.html
What if the batch variable is unknown?

- Manifests as systematic “unwanted variation” in data
- Can be identified using e.g. control genes (“housekeeping” genes, spike-ins)
- Represent themselves as residuals after eliminating known signal

Methods to account for

- Include estimated unwanted variation as covariate(s) in the statistical model
- RUV, sva packages commonly used in genomics

SVA

When batches were unknown, the surrogate variable analysis (SVA) was developed.

The main idea was to separate the effects caused by covariates of our primary interests from the artifacts not modeled.

Now the raw expression value $Y_{jg}$ of gene $g$ in sample $j$ can be formulated as:

$$Y_{jg} = \alpha_g + X\beta_g + \sum_{k=1}^{K} \lambda_{kg} \eta_{kj} + \epsilon_{jg}$$

where $\eta_{kj}$s represent the unmodeled factors and are called as “surrogate variables”.

13/20

14/20
SVA

Once again, the basic idea was to estimate $\eta_{ij}$s and adjust them accordingly.

An iterative algorithm based on singular value decomposition (SVD) was derived to iterate between estimating the main effects $\hat{\alpha}_g + X\hat{\beta}_g$ given the estimation of surrogate variables and estimating surrogate variables from the residuals $r_{ij} = Y_{ij} - \hat{\alpha}_g - X\hat{\beta}_g$

sva package in Bioconductor

- Contains ComBat function for removing effects of known batches.
- Assume we have:
  - edata: a matrix for raw expression values
  - batch: a vector named for batch numbers.

```r
modcombat = model.matrix(~1, data=as.factor(batch))
combat_edata = ComBat(dat=edata, batch=batch, mod=modcombat, par.prior=TRUE, prior.plot=FALSE)
```

SVASEQ

For sequencing data, svaseq, the generalized version of SVA, suggested applying a moderated log transformation to the count data or fragments per kilobase of exon per million fragments mapped (FPKM) first to account for the nature of discrete distributions.

Instead of a direct transformation on the raw counts or FPKM, remove unwanted variation (RUV) adopted a generalized linear model for $Y_{jk}$.

BatchQC - Batch Effects Quality Control

A Bioconductor package with a GUI (shiny app).

https://github.com/mani2012/BatchQC
What to use

“ComBat, an Empirical Bayes method, outperformed the other five programs by most metrics”


References


- Batch effects and the importance of EDA, https://kbroman.wordpress.com/2012/04/25/microarrays-suck/