UCLA Health | Jonsson Comprehensive Cancer Center

Batch effects: problem and solution

Jaron Arbet

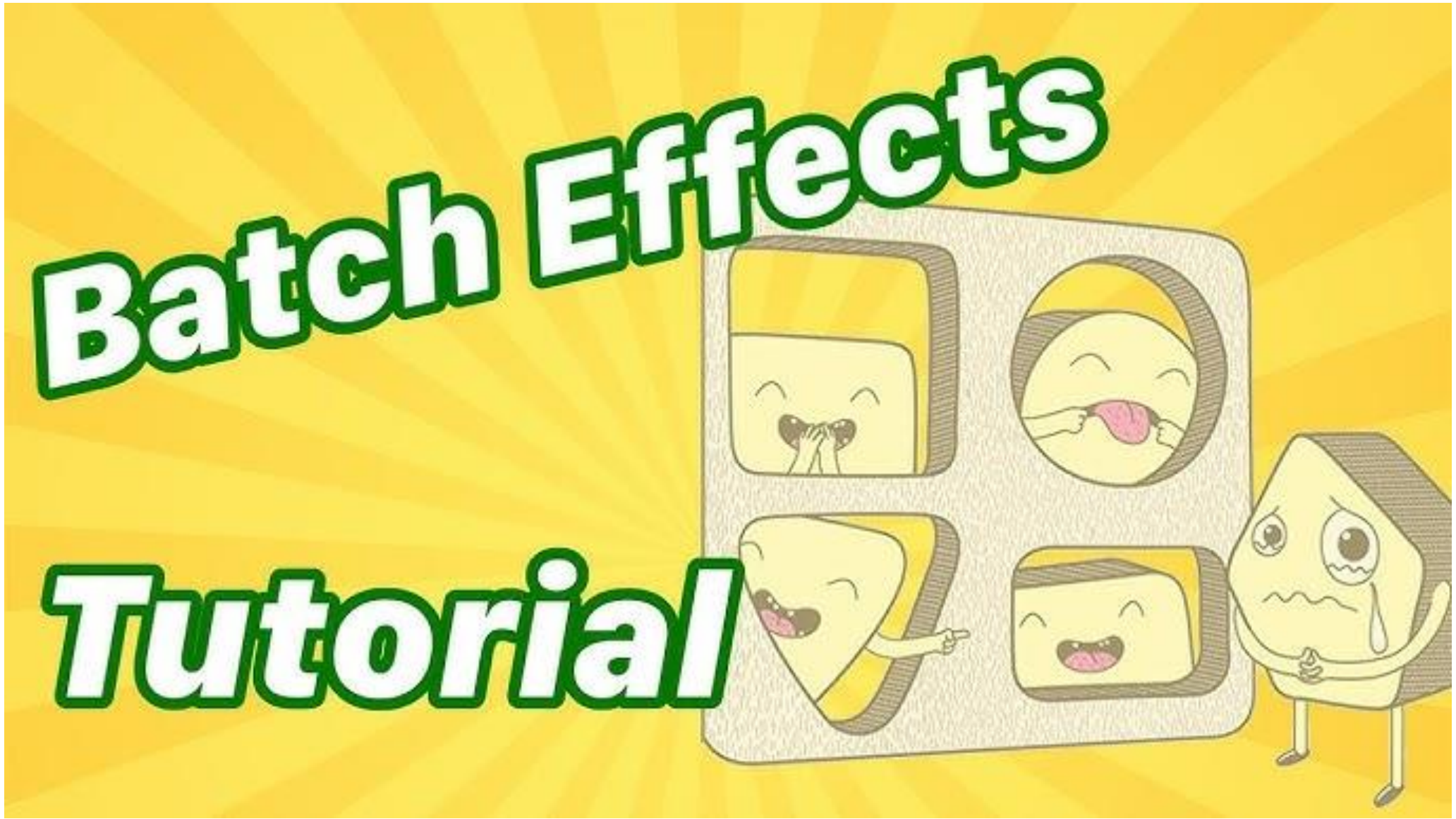


Image from: https://www.youtube.com/watch?v=3wjbxDex7_8

Background

- We routinely measure high-dimensional ‘omics data:
  - Genomics (SNPs, SNVs, CNA, SVs), mRNA, proteomics, epigenomics, metabolomics, microbiome, *etc.*
- Each sample must be processed in a lab, often with complex protocols
- Minor differences in protocols (batch, lab, personnel, date, temperature, day of week, *etc.*) can have large impact on measurements. Collectively, these variables are known as “**batch effects**”.
- But more broadly, batch effects are any variables that ***confound*** the relationship btwn the main outcome and ‘omics data

Examples

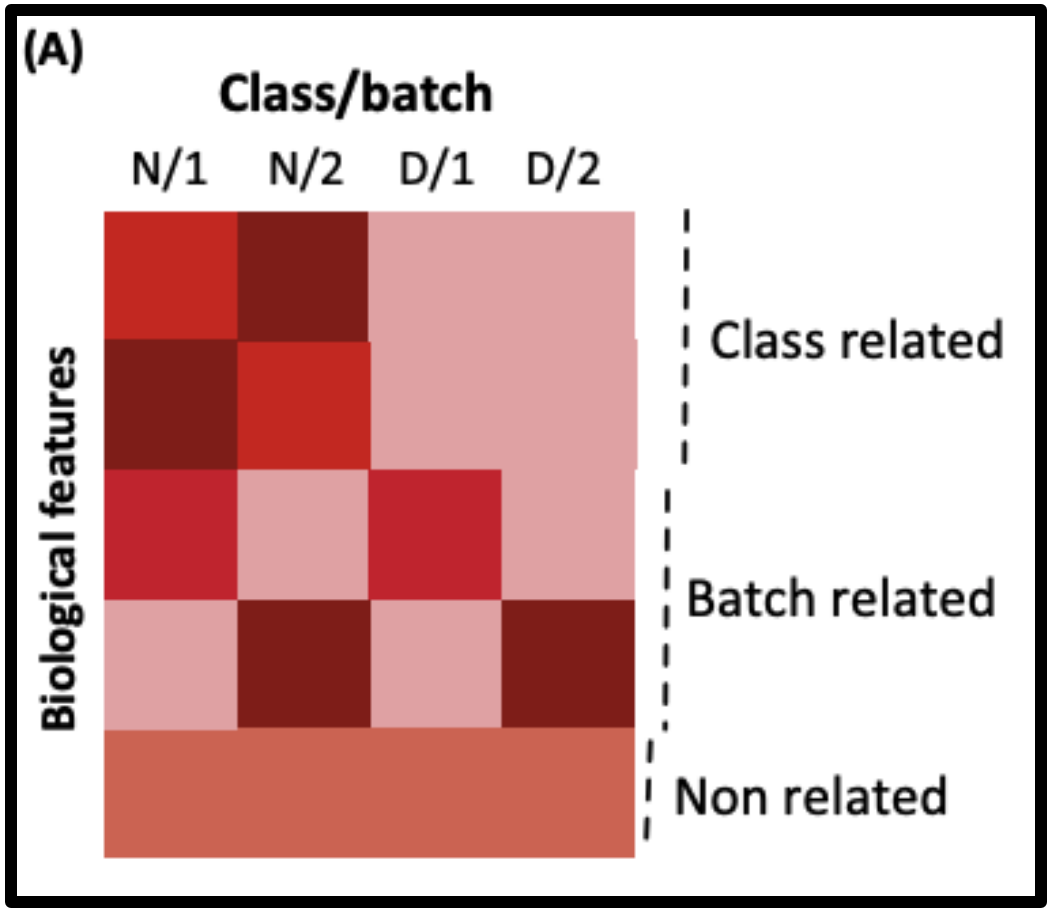


Table 1 | **Batch effects seen for a range of high-throughput tecl**

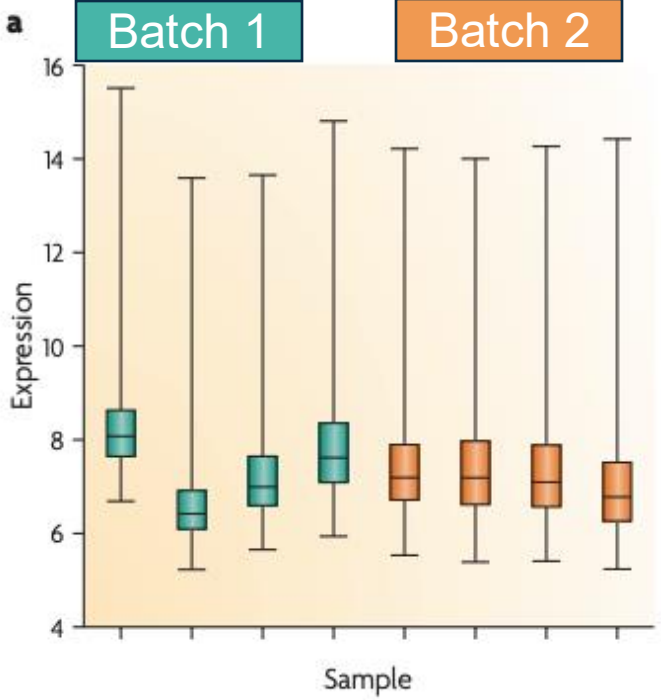
Study description*	Known variable used as a surrogate		
	Surrogate‡	Confounding (%)§	Susceptible features (%)‖
Data set 1: gene expression microarray, Affymetrix (N_p = 22,283)	Date	29.7	50.5
Data set 2: gene expression, Affymetrix (N_p = 4167)	Date	77.6	73.7
Data set 3: mass spectrometry (N_p = 15,154)	Processing group	100	51.7
Data set 4: copy number variation, Affymetrix (N_p = 945,806)	Date	29.2	99.5
Data set 5: copy number variation, Affymetrix (N_p = 945,806)	Date	12.2	83.8
Data set 6: gene expression, Affymetrix (N_p = 22,277)	Processing group	NA	83.8
Data set 7: gene expression, Agilent (N_p = 17,594)	Date	NA	62.8
Data set 8: DNA methylation, Agilent (N_p = 27,578)	Processing group	NA	78.6
Data set 9: DNA sequencing, Solexa (N_p = 2,886)	Date	24.2	32.1

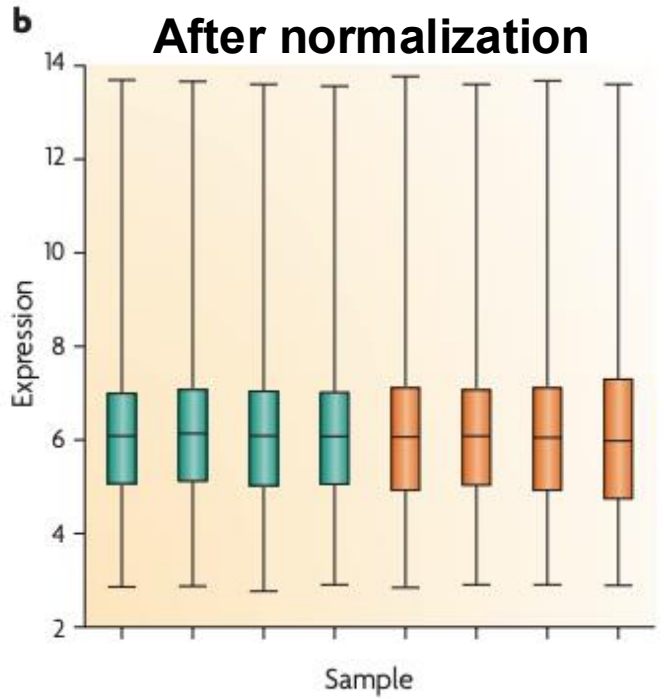
Across 9 datasets, **32.1% to 99.5%** of features were significantly associated with **date** of processing, or **processing group**

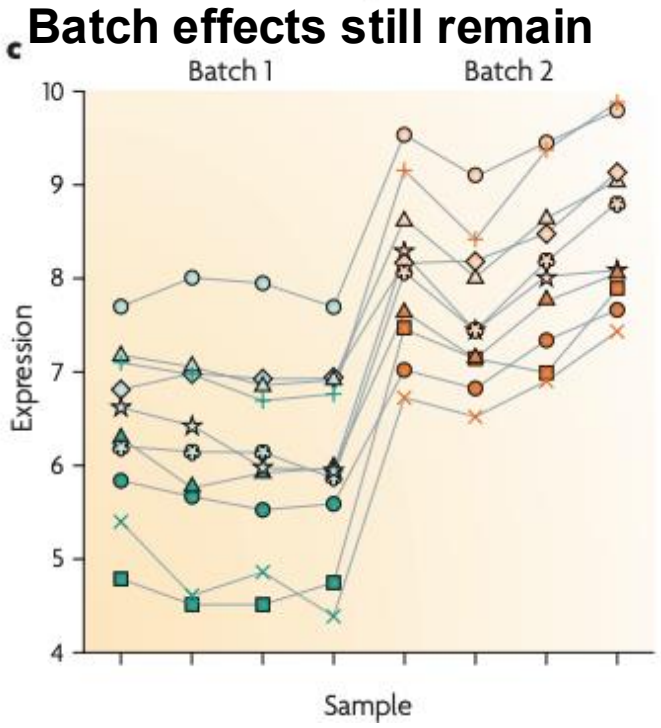
Goh 2017
Leek 2010

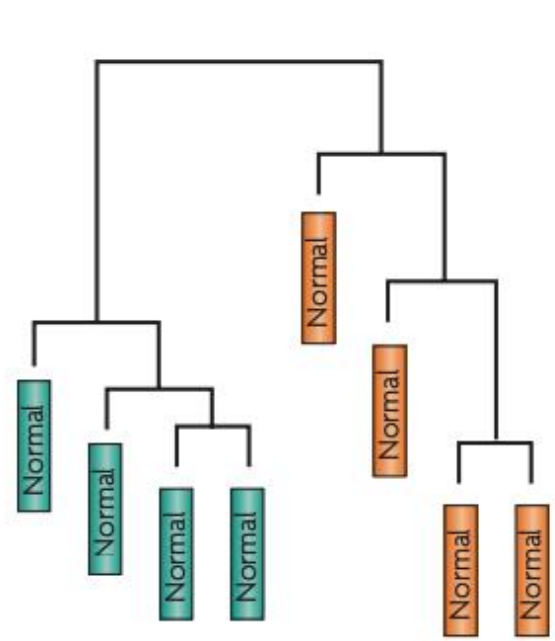
Normalization

- Batch effect correction and “Normalization” are NOT the same thing!
- Some Normalization methods attempt to correct for some batch effects, but batch effects can still remain post-normalization:









Leek 2010

Solutions

1. Good experimental design

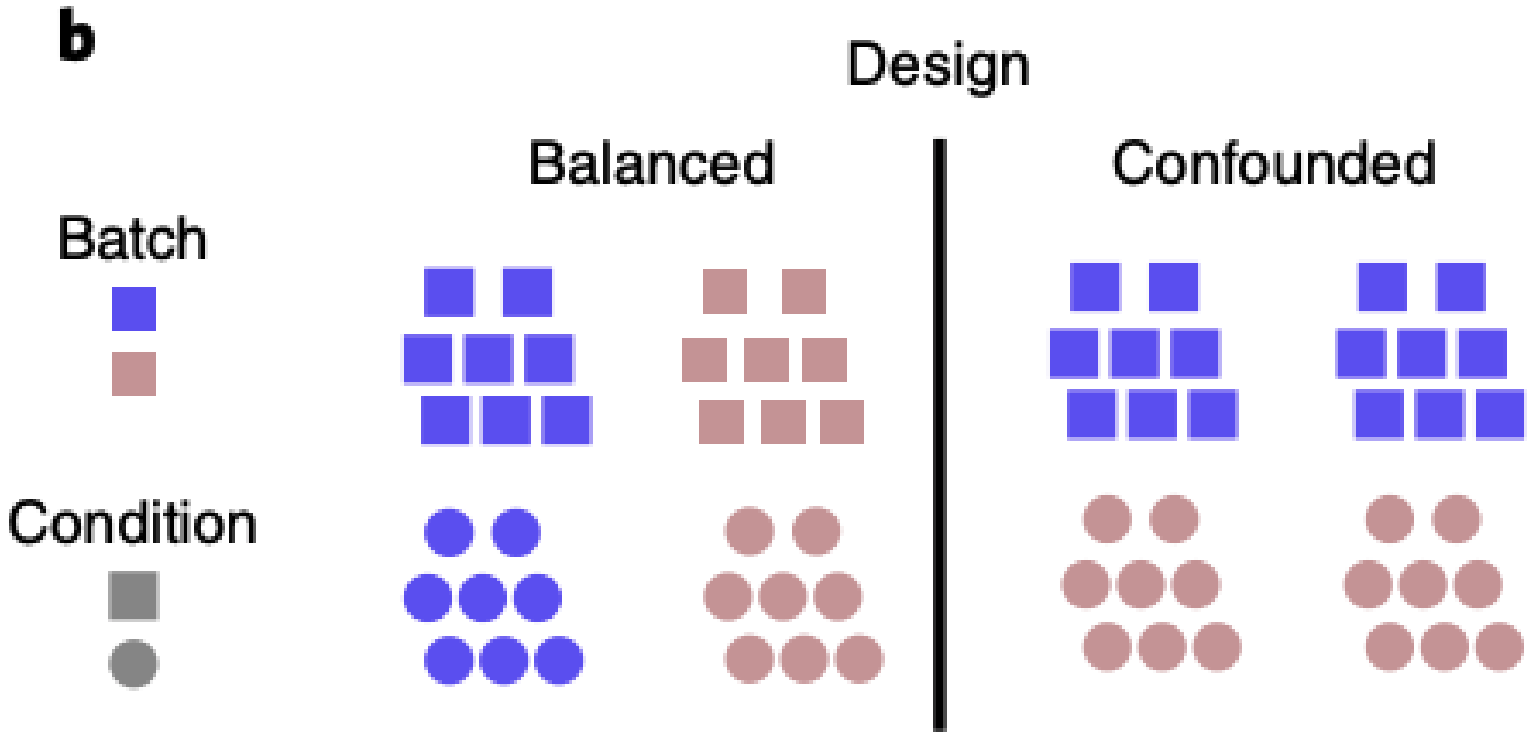
2. Identify **measured** features that are clear Batch effects; adjust for them in your analyses

3. Estimate **hidden/unmeasured** batch effects; adjust for them in your analyses

Goh 2017
Büttner 2019

1. Good experimental design

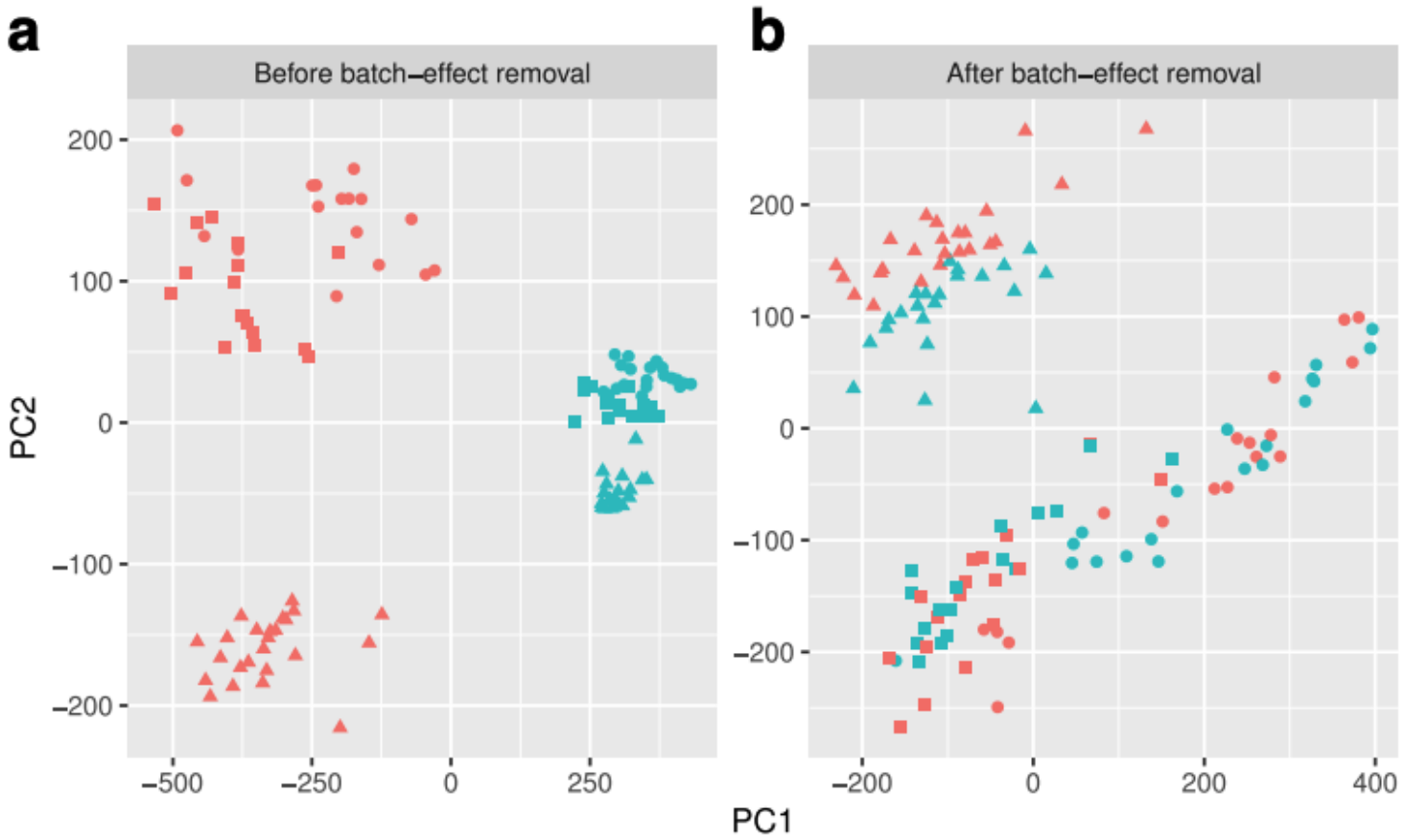
- When comparing groups (trt vs control, disease vs normal), evenly distribute them between processing batches:



2. Identify measured batch effects

PCA:

- Test clinical features *vs.* top PCs
- Scatterplots of top PCs colored by clinical features

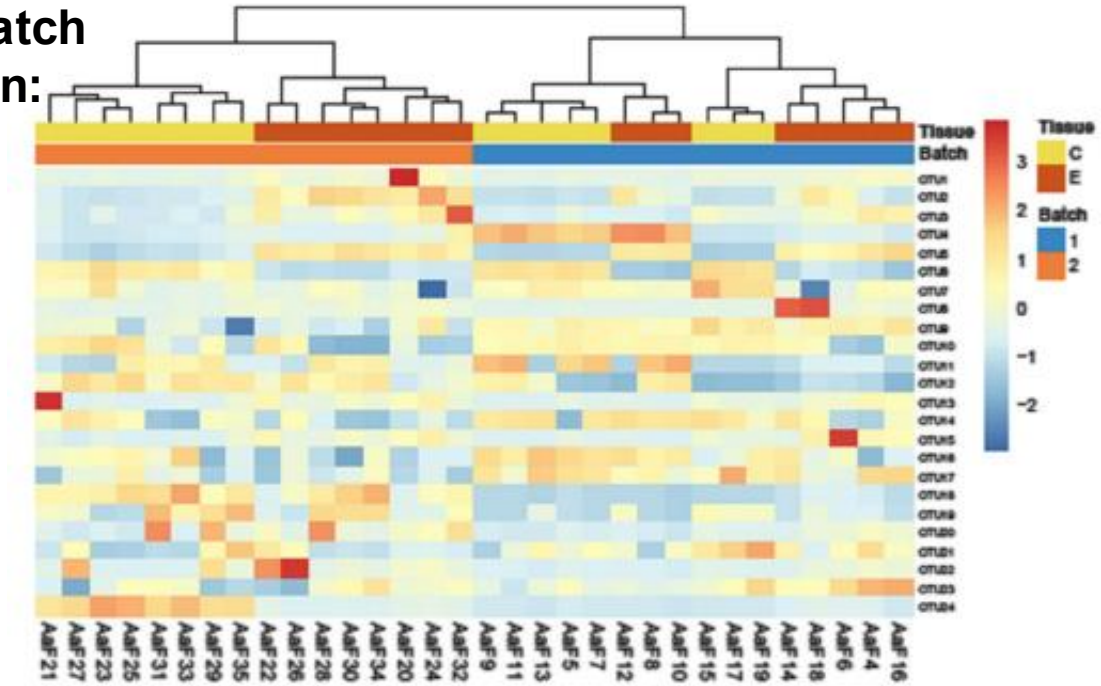


Clustered heatmaps:

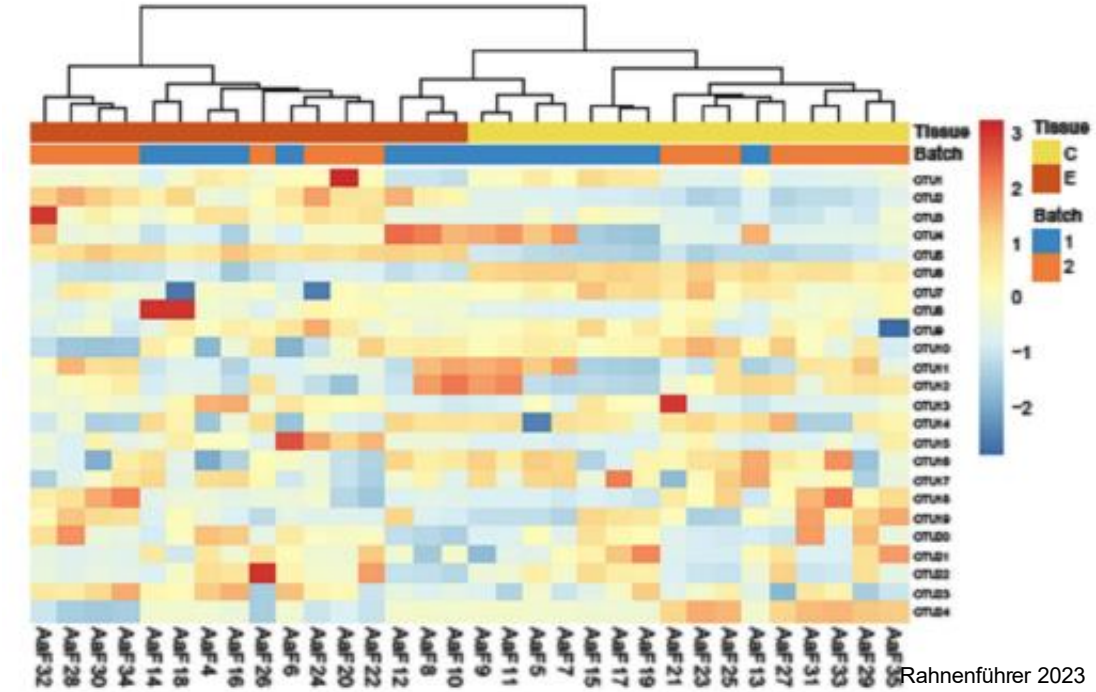
- Use hierarchical clustering to sort rows of features and columns of patients based on their molecular profiles

- Plot covariate bars at top. Do any covariates drive the sorting?

Before batch correction:

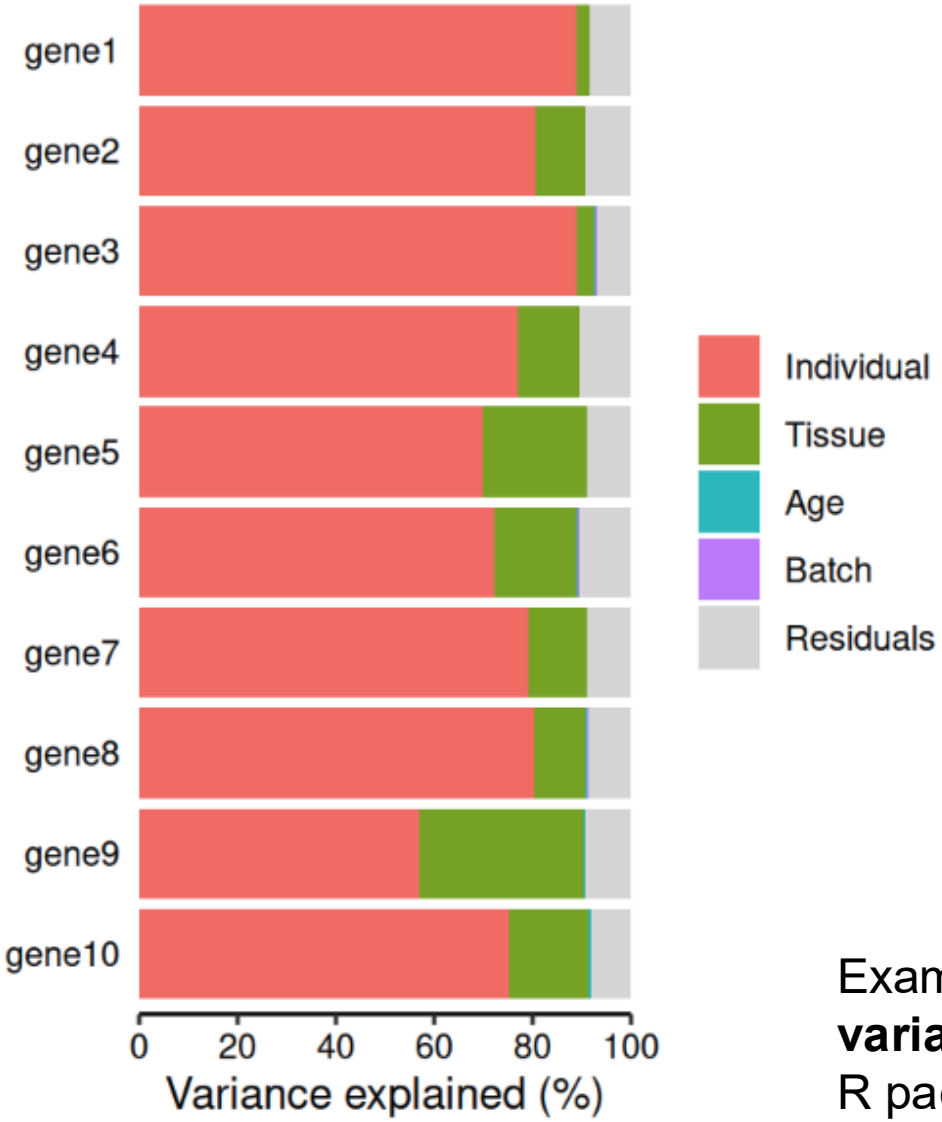


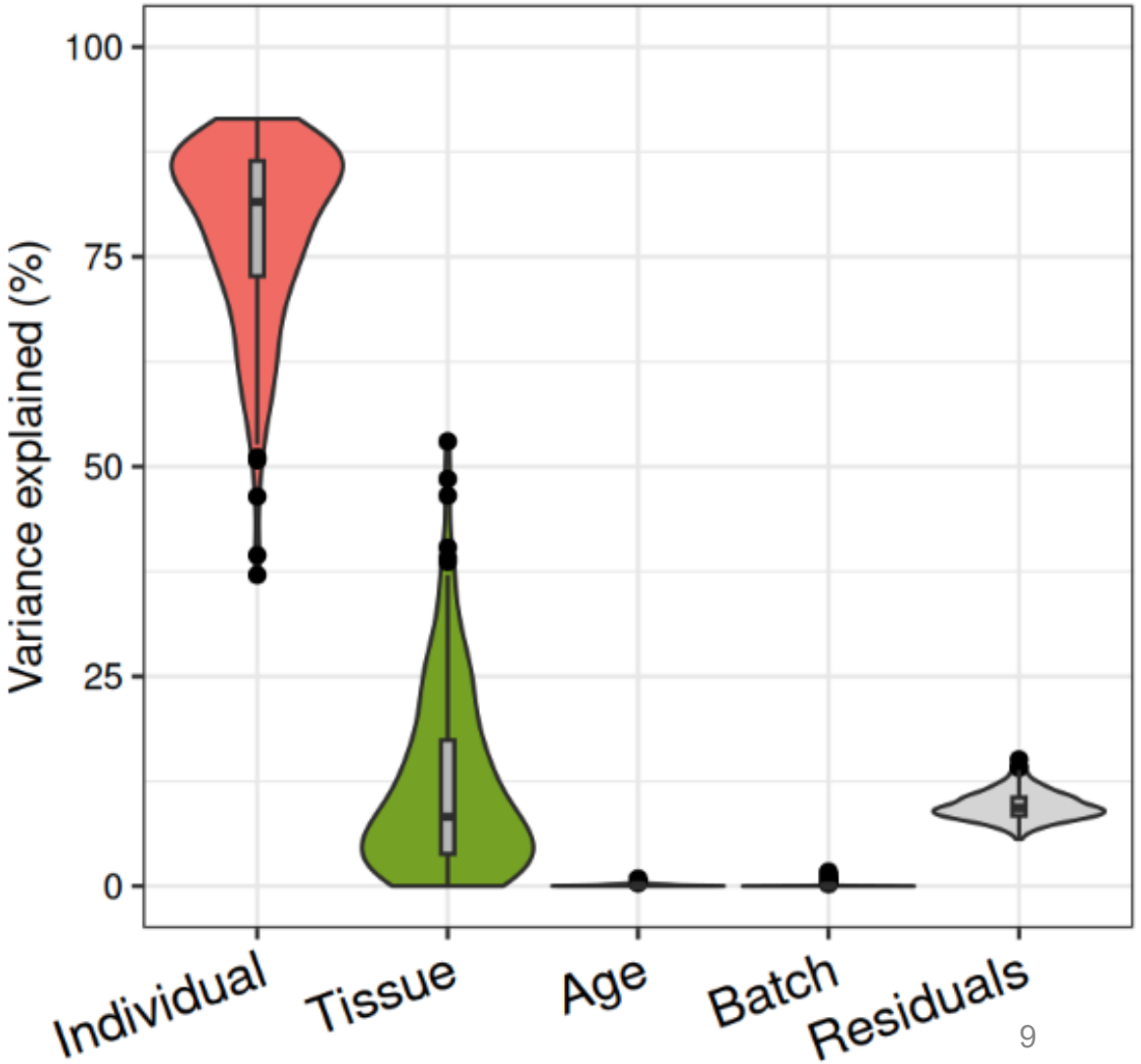
After batch correction:



Comparing effect sizes

Test association btwn each potential batch effect with all molecular features; compare distributions of effect sizes





Example from **variancePartition** R package

2. How to adjust for known batch effects?

1. Adjust for them as covariates in linear models
2. “Residualize”: for a given molecular feature, regress it against the batch effects and use the residuals as the new “batch-corrected feature”
3. ComBat: popular software to adjust for known batch effects

- I prefer (1) since most interpretable, but (3) better if small sample size

Johnson 2007

3. Estimate hidden/unmeasured batch effects

Surrogate variable analysis (SVA)

1. Identify biological variables where you want to preserve the molecular differences (*e.g.* tissue type, prognosis, trt vs ctrl)
2. For each molecular feature, use a linear model to regress out the above variables
3. Input the residuals from (2) into PCA
4. Identify the top K PCs to keep: these represent latent sources of variation **unrelated to the biology you care about** (1)
5. For downstream analyses, use linear models that adjust for PCs from (4)

…Or maybe simple PCA directly on the molecular features is good enough?

Zhou *et al. Genome Biology* (2022) 23:210
https://doi.org/10.1186/s13059-022-02761-4

Genome Biology

RESEARCH Open Access

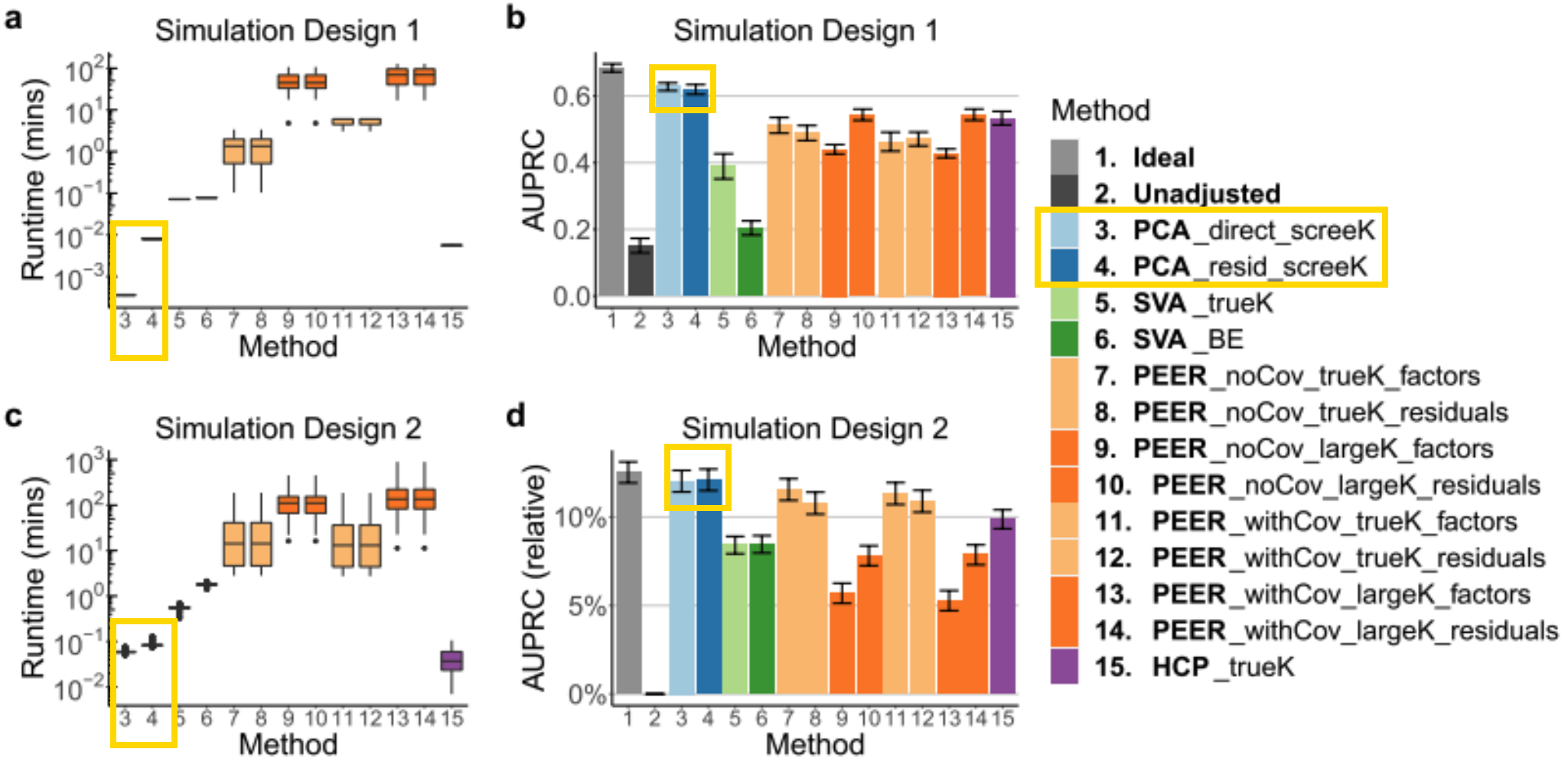
PCA outperforms popular hidden variable inference methods for molecular QTL mapping

Heather J. Zhou[1], Lei Li[2], Yumei Li[3], Wei Li[3] and Jingyi Jessica Li[1,4,5,6*]

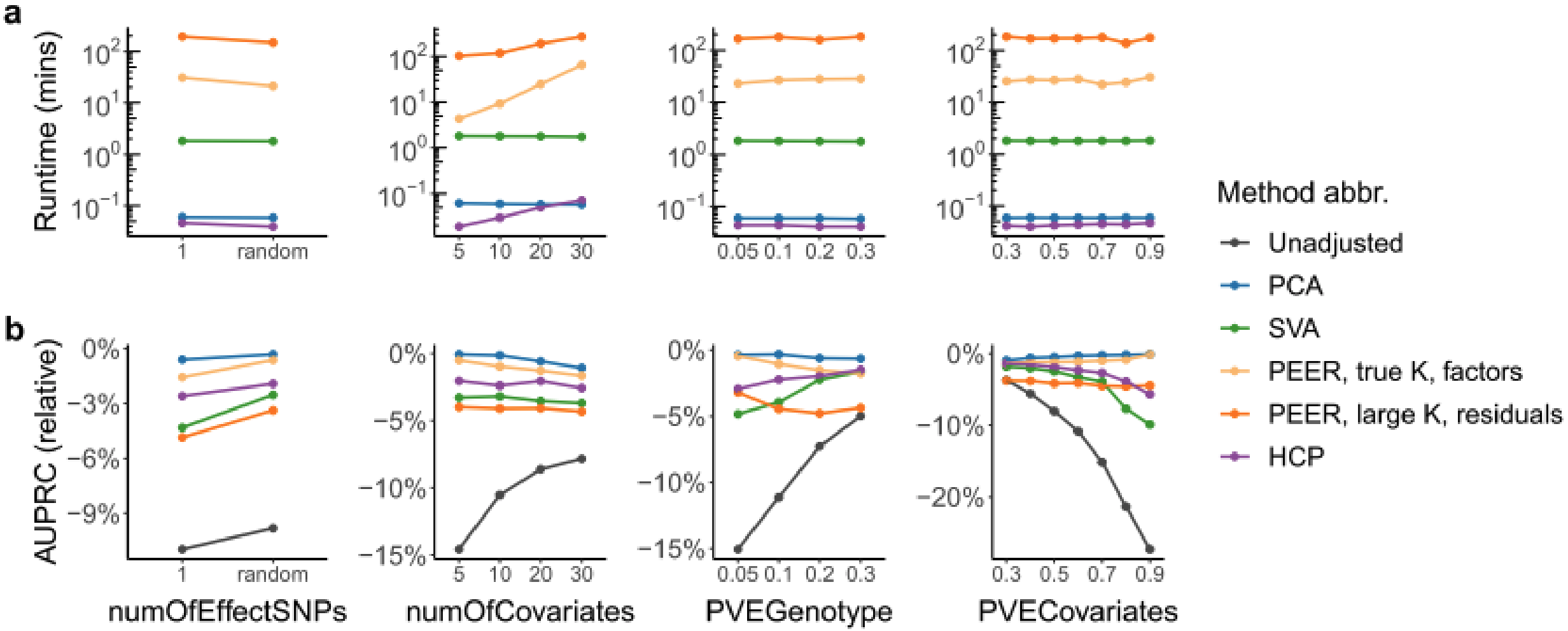
R Tutorial:
https://github.com/heatherjzhou/PCAForQTL

- Compared SVA, PEER, HCP, and simple PCA on 362 synthetic and 110 real datasets
- "We show that **PCA** not only underlies the statistical methodology behind the popular methods but is also **orders of magnitude faster, better-performing, and much easier to interpret and use**"

PCA is faster and more powerful



PCA runtime and power is robust across many settings



PCs adjust for batch effects **and** allow us to account for global differences in molecular profiles:

> “PCA can be interpreted as both a dimension reduction and a [batch effect/confounder] discovery method.
>
> by including PCs as covariates, we are controlling for the effect of the overall molecular profile on the abundance level of any individual feature
>
> Thus including molecular PCs as covariates is analogous to including genotype PCs as covariates (which is commonly done to correct for population stratification)”

Downside to PCA?

- If the top PCs are highly correlated with the main predictor of interest (*e.g.* trt vs. ctrl, disease vs normal), then you will lose power when adjusting for them.
- SVA overcomes this problem
- Remarkably this problem was not apparent in the work of Zhu 2017, but in extreme settings, could still be worth trying SVA over PCA

Recommendations

1. Always make sure to use **good study design**: balance the treatment group across processing batches
2. Use PCA and clustered heatmaps to identify **measured batch effects**. Adjust for them as covariates in linear models.
3. To account for **hidden batch effects**, use PCA first following the Zhu 2017 R package tutorial
4. If the PCs are highly correlated with the main predictor of interest, then consider using **SVA** instead.. but PCA may still work well if N is large.
5. If small N and/or small number of molecular features, **ComBat** is best option for known BEs, but can't estimate hidden BEs.

References

- Büttner, Maren, et al. "A test metric for assessing single-cell RNA-seq batch correction." Nature methods 16.1 (2019): 43-49.
- Goh, Wilson Wen Bin, Wei Wang, and Limsoon Wong. "Why batch effects matter in omics data, and how to avoid them." Trends in biotechnology 35.6 (2017): 498-507.
- Johnson, W. Evan, Cheng Li, and Ariel Rabinovic. "Adjusting batch effects in microarray expression data using empirical Bayes methods." Biostatistics 8.1 (2007): 118-127.
- Leek, Jeffrey T., and John D. Storey. "Capturing heterogeneity in gene expression studies by surrogate variable analysis." PLoS genetics 3.9 (2007): e161.
- Leek, Jeffrey T., et al. "Tackling the widespread and critical impact of batch effects in high-throughput data." Nature Reviews Genetics 11.10 (2010): 733-739.
- Rahnenführer, Jörg, et al. "Statistical analysis of high-dimensional biomedical data: a gentle introduction to analytical goals, common approaches and challenges." BMC medicine 21.1 (2023): 182.
- Zhou, Heather J., et al. "PCA outperforms popular hidden variable inference methods for molecular QTL mapping." Genome biology 23.1 (2022): 210.
- Zhu, X., et al. "Granatum: a graphical single-cell RNA-Seq analysis pipeline for genomics scientists. Genome Med 9: 108." 2017,