The session looks at a major issue that underlies many omics datasets, viz. batch effects. Batch effects are technical biases that may confound analysis of omics data. They are very complex and effective mitigation is highly context dependent. Do they affect identification of discriminating/causal factors when we analyze patient datasets? Do prediction models (constructed on training datasets) work well on future patients? How do you mitigate batch effects?

Session Plan

Part I, What batch effects are and how they affect biomedical data analysis and model building.

Suggested readings:


Part II, How batch effects can be measured. How do you know they are big enough to worry over?

Suggested readings:

- **PCA side-by-side boxplot** (Goh & Wong, “Protein complex-based analysis is resistant to the obfuscating consequences of batch effects – a case study in clinical proteomics”, *BMC Genomics*, 18:142, 2017)

Part III, Normalization methods and batch effect-correction methods. What are these and what are their important differences?

Suggested readings:

- Common normalization methods such as linear scaling, quantile normalization, z-score transformation, and specialized methods such as **GFS** (Belorkar & Wong, “GFS: Fuzzy preprocessing for effective gene expression analysis”, *BMC Bioinformatics*, 17(Suppl 17):540, 2016)

**Part IV, How should a normalization method be applied when there are multiple classes and batches?**

Suggested readings:

• Zhao et al., “How to do quantile normalization correctly for gene expression data analysis”, *Scientific Reports*, 10:15534, 2020

**Part V, How do normalization methods interact with batch effects and batch effect-correction methods**

Suggested readings:


**Part VI, If a dataset has lots of missing values and also batch effects, what happens and what can/should you do?**

Suggested readings:

• Some missing value-imputation methods (imputation by global mean, same-batch mean, nearest neighbours, etc.)
• Voss et al., “HarmonizR enables data harmonization across independent proteomic datasets with appropriate handling of missing values”, *Nat. Comm.*, 13: 3523, 2022
• Sun & Goh, “Why batch sensitization is important for missing value imputation”, [https://doi.org/10.21203/rs.3.rs-1328989/v1](https://doi.org/10.21203/rs.3.rs-1328989/v1)
Outline: The session looks at a major issue that underlies many omics datasets, viz. batch effects. Batch effects are technical biases that may confound analysis of omics data. They are very complex and effective mitigation is highly context dependent. Do they affect identification of discriminating/causal factors when we analyze patient datasets? Do prediction models (constructed on training datasets) work well on future patients? How do you mitigate batch effects?
What batch effects are
Batch effects

Unwanted non-biological variations due to processing time, reagent batch, handlers, etc.

Batch-class imbalance

One class forms a large fraction of a batch and another class forms a large fraction of another batch

In this situation, batch effects tend to be badly confounded with biological effects
Childhood leukemia patients

Samples from different batches are grouped together, regardless of subtypes and treatment response.
Peripheral blood mononuclear cells (PBMC)
Exercise

Do batch effects affect data analysis and model building?
In what ways?
Exercise

What makes batch-label randomization a valid control?
How batch effects are “measured”
Paired boxplots of PCs

PCA scatter plot is often used for visualizing batch effects

But it is easier to see which PC is enriched in batch effects by showing, side by side, the distribution of values of each PC stratified by class and by batch variables
χ2 test the local batch distribution against the global batch distribution

For high-dimensional data, the authors recommend to do PCA, retain the top 50 PCs, then run kBET on the reduced data
Exercise

What is good/bad about paired boxplots of PCs?

What is good/bad about kBET?

*E.g.*, what if class or batch proportions are imbalanced? What if some classes appear only in some batches?

Suggest how to improve either of the above for quantifying batch effects, or suggest a totally different approach.
Normalization & batch-effect correction
Normalization vs batch-effect correction

Normalization

*Put data into the same scale*

e.g., linear scaling, z-score, quantile normalization, GFS

Batch-effect correction

*Remove batch effects*

e.g., Combat, Harman, surrogate variable analysis, batch mean centering, GFS
Exercise

Does quantile normalization remove batch effects?

Does it make it easier to identify differentially expressed genes?

Quantile normalization

Given $n$ arrays of length $p$, form $X$ of size $p \times n$ where each array is a column.

Sort each column of $X$ to give $X_{\text{sort}}$.

Take means across rows of $X_{\text{sort}}$ and assign this mean to each elem in the row to get $X'_{\text{sort}}$.

Get $X_{\text{normalized}}$ by arranging each column of $X'_{\text{sort}}$ to have same ordering as $X$. 
Exercise

Look up batch mean centering (BMC)

Does it remove additive batch effects well?

Does it remove multiplicative batch effects well?
When class & batch are balanced

Normalization methods (e.g., quantile) do not remove batch effects

But they still easily separate the classes

Zhou et al., *J Genet & Genom*, 46:433-443, 2019
When class & batch are imbalanced, i.e., when one batch is dominated by one class

The situation deteriorates quickly …
Impact on feature selection

A Balanced

C Severely unbalanced

Zhou et al., J Genet & Genom, 46:433-443, 2019
Missing values & batch effects
Some omics data have lots of missing values (proteomics MS, scRNA-seq, etc.)

<table>
<thead>
<tr>
<th>Table Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Column 1</td>
</tr>
<tr>
<td>Data 1</td>
</tr>
<tr>
<td>Data 3</td>
</tr>
<tr>
<td>Data 5</td>
</tr>
<tr>
<td>Data 7</td>
</tr>
<tr>
<td>Data 9</td>
</tr>
</tbody>
</table>

Common missing-value imputation methods

Impute based on the mean value of the corresponding feature

Determine highly correlated variables, impute by regression

Impute based on the mean of k nearest neighbours

Hmm... this 78-yr-old is as fit as a 42-yr-old?
Exercise

You have two batches with lots of missing values

Do you normalize / remove batch effects first, or do you impute missing values first?

Do you combine the two batches and do missing-value imputation on the combined data, or do you do missing-value imputation on the two batches separately?
Why batch-sensitization is imp for missing-value imputation
Combined data into one matrix
Extract submatrices w/ few missing values
Batch correct each submatrix
Put them back together
Summary

Batch effects are insidious and unavoidable in omics data

Batch-effect correction can introduce artifacts into data

Missing values are prevalent in some omics data types (e.g., proteomics MS and scRNA-seq)

Missing-value imputation in the presence of batch effects is tricky

Batch-effect correction in the presence of missing values is tricky


Goh & Wong, “Protein complex-based analysis is resistant to the obfuscating consequences of batch effects – a case study in clinical proteomics”, *BMC Genomics*, 18:142, 2017 (PCA side-by-side boxplot)

Zhao et al., “How to do quantile normalization correctly for gene expression data analysis”, *Scientific Reports*, 10:15534, 2020