Introduction to *metadeconfoundR*

Till Birkner*[1]*, Sofia K. Forslund

[1] tillb@online.de

Edited: November 06, 2019; Compiled: October 19, 2020

Contents

1 Introduction 1

2 Quick start 3

3 Use cases 3

4 `sessionInfo()` 3

1 Introduction

Package

metadeconfoundR 0.1.8

When analyzing multi omics datasets, the search for features that could serve as biomarkers is an important aspect. Because these biomarkers might be used in clinical settings for disease diagnosis etc., it is extremely important to minimize false positives. One possible error source are confounding variables: The biomarker is not directly linked to the disease but influenced by a third (confounding) variable, that in turn is linked to the disease.

In this project work, the R package MultideconfoundR was developed to address this issue. It first uses univariate statistics to find associations between omics features and disease status or metadata. Using nested linear model comparison post hoc testing, those associations are checked for confounding effects from other covariates/metadata and a status label is returned. The tool is able to handle large datasets in a reasonable time, by parallel processing.

metadeconfoundR was developed to conduct confounder-aware biomarker analysis of large scale multi-omics datasets in parallel. This analysis is a two step process.

First, significant associations between single omics features (like gut microbial OTUs) and metadata (like drug administration) are identified (fig1, left). Based on the data type of the respective metadata, either `wilcox.test()` (for binary), `cor.test()` (for continuous numerical) or `kruskal.test()` (for neither numerical nor binary) is used. All three tests are rank-based to minimize assumptions about data distribution. In addition to collecting p-values for all computed tests, effect size is measured if possible. In case of binary data, Cliff's Delta is computed by an algorithm based on the `orddom::orddom()` function. For continuous data the "estimate" component from `cor.test()` is used. Since there is no effect size for

categorical data with more than 2 levels, no value is reported here. It is recommended to introduce binary pseudo-variables for each level of the categorical metadata to circumvent this drawback.

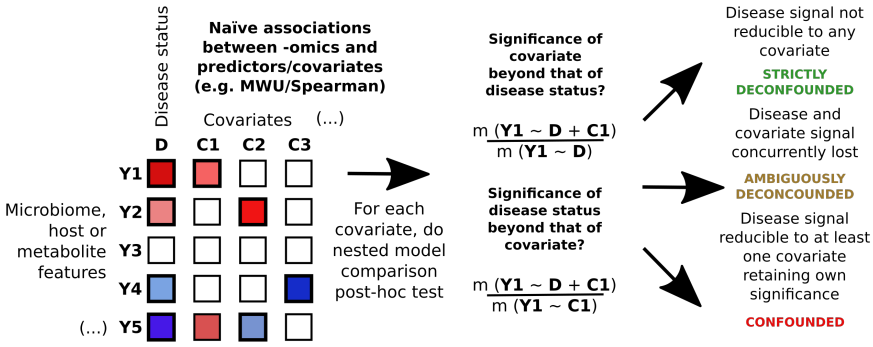

Figure 1: **Overview of main statistical approach used to determine confounding status**
(left) Each omics feature (Y) is independently tested for association to any of the predictors/covariates (D/C); Used test depends on data type of covariate. (center) For each identified feature (Y1) ↔ covariate (D/C) pair a set of linear models (either including an additional covariate or not) is fitted and a likelihood ratio is computed. (right) Based on likelihood ratio a status for the feature (Y1) ↔ covariate (D/C) pair is reported. Figure kindly provided by Sofia Forslund.

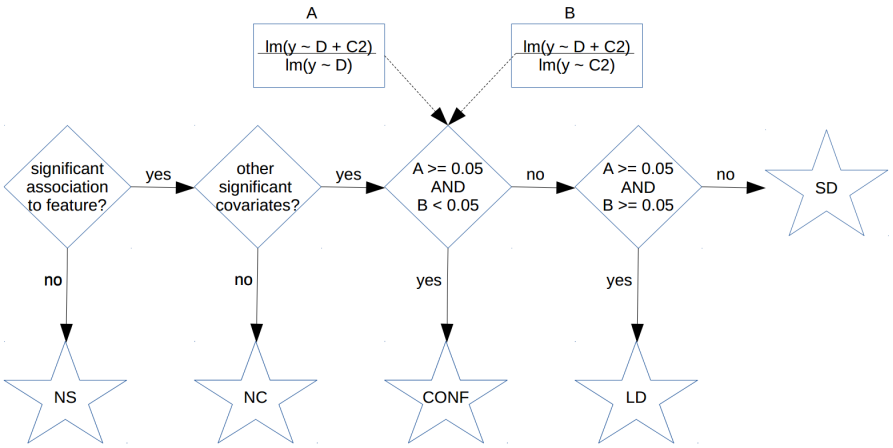

Figure 2: **Status labeling process in more detail**
For each feature ↔ covariate combination these steps are done. A and B are the linear model likelihood ratios. Should there be more than one "other significant covariate" (C2), linear model likelihood ratio comparison has to be repeated for every single one of them. Whenever CONF is reached, the name of C2 is returned as label. Only when SD is reached for all C2, this will be returned as label. (y = feature, D = current covariate, C2 = other significant covariates, NS = not significant, NC = no covariates (i.e. trivially deconfounded), CONF = confounded, LD = laxly deconfounded, SD = strictly deconfounded)

In the second step, all hits are checked for confounding effects (fig 1, center and right) and a status is reported for each feature ↔ metadata combination (fig 2). A “hit" here is defined as a feature ↔ metadata association with small enough fdr-corrected p-value and big enough effect size. Thresholds for both parameters can be set via `QCutoff` and `DCutoff` when starting the analysis. Since confounding of signal can only happen with more than one different metadata associated to a certain feature, all features with only one significant metadata are trivially deconfounded and get status “No Covariates (NC)".
The actual confounder detection is done by a set of two likelihood ratio tests of nested linear models: For each feature with multiple covariates

2 Quick start

Install via tar.gz file: `install.packages("metadeconfoundR_0.1.1.tar.gz", repos = NULL)`

[]/home/till/R/x86$_64$−pc−$linux$−gnu−$library$/3.6/$BiocStyle$/$resources$/tex/$Bioconductor$

3 Use cases

4 `sessionInfo()`

```
toLatex(sessionInfo())
```

- R version 3.6.3 (2020-02-29), `x86_64-pc-linux-gnu`
- Locale: `LC_CTYPE=C.UTF-8`, `LC_NUMERIC=C`, `LC_TIME=C.UTF-8`, `LC_COLLATE=C`, `LC_MONETARY=C.UTF-8`, `LC_MESSAGES=C.UTF-8`, `LC_PAPER=C.UTF-8`, `LC_NAME=C`, `LC_ADDRESS=C`, `LC_TELEPHONE=C`, `LC_MEASUREMENT=C.UTF-8`, `LC_IDENTIFICATION=C`
- Running under: `Ubuntu 18.04.4 LTS`
- Matrix products: default
- BLAS: `/usr/lib/x86_64-linux-gnu/blas/libblas.so.3.7.1`
- LAPACK: `/usr/lib/x86_64-linux-gnu/lapack/liblapack.so.3.7.1`
- Base packages: base, datasets, grDevices, graphics, methods, stats, utils
- Other packages: metadeconfoundR 0.1.8
- Loaded via a namespace (and not attached): BiocManager 1.30.10, BiocStyle 2.14.4, MASS 7.3-51.5, Matrix 1.2-18, R6 2.4.1, Rcpp 1.0.4.6, assertthat 0.2.1, backports 1.1.7, bigmemory 4.5.36, bigmemory.sri 0.1.3, boot 1.3-24, callr 3.4.3, cli 2.0.2, codetools 0.2-16, colorspace 1.4-1, compiler 3.6.3, crayon 1.3.4, desc 1.2.0, digest 0.6.25, doParallel 1.0.15, dplyr 1.0.2, ellipsis 0.3.1, evaluate 0.14, fansi 0.4.1, foreach 1.5.0, formatR 1.7, futile.logger 1.4.3, futile.options 1.0.1, generics 0.0.2, ggplot2 3.3.1, glue 1.4.1, grid 3.6.3, gtable 0.3.0, htmltools 0.4.0, iterators 1.0.12, knitr 1.28, lambda.r 1.2.4, lattice 0.20-40, lifecycle 0.2.0, lme4 1.1-23, lmtest 0.9-37, magrittr 1.5, minqa 1.2.4, munsell 0.5.0, nlme 3.1-144, nloptr 1.2.2.1, parallel 3.6.3, pillar 1.4.4, pkgbuild 1.0.6, pkgconfig 2.0.3, pkgload 1.1.0, plyr 1.8.6, prettyunits 1.1.1, processx 3.4.2, ps 1.3.3, purrr 0.3.4, reshape2 1.4.4, rlang 0.4.8,

rmarkdown 2.1, roxygen2 7.1.0, rprojroot 1.3-2, rstudioapi 0.11, scales 1.1.0,
splines 3.6.3, statmod 1.4.34, stringi 1.4.6, stringr 1.4.0, testthat 2.3.2, tibble 3.0.1,
tidyselect 1.1.0, tools 3.6.3, vctrs 0.3.4, withr 2.2.0, xfun 0.14, xml2 1.3.2,
yaml 2.2.1, zoo 1.8-8