使用ComBat方法校正站点效应

Alex / 2024-11-04 / free_learner@163.com / AlexBrain.cn

本文介绍在R环境下使用ComBat方法校正站点效应（site effect）。

一、背景

磁共振数据对于扫描仪和参数都非常敏感，不同机器或者扫描参数采集的数据具有明显的差异，这使得将不同站点或者不同参数采集的数据融合在一起进行分析变得困难。ComBat方法是目前校正站点效应常用的方法。注意ComBat也可以校正其他类型的差异，比如一批数据是用不同版本的软件处理的，那么这种软件版本带来的差异也可以使用ComBat方法进行校正，因为ComBat并不假设差异的来源。不过ComBat是否适用于所有的场景，也许还有待于进一步验证。关于ComBat的原理请参考如下文献：

- Fortin, J., Parker, D., Tunç, B., Watanabe, T., Elliott, M.A., Ruparel, K., Roalf, D.R., Satterthwaite, T.D., Gur, R.C., Gur, R.E., Schultz, R.T., Verma, R., & Shinohara, R.T. (2017). Harmonization of multi-site diffusion tensor imaging data. *NeuroImage*, 161, 149-170.
- Fortin, J., Cullen, N.C., Sheline, Y.I., Taylor, W.D., Aselcioglu, I., Cook, P.A., Adams, P.B., Cooper, C.M., Fava, M., McGrath, P.J., McInnis, M.G., Phillips, M.L., Trivedi, M.H., Weissman, M.M., & Shinohara, R.T. (2017). Harmonization of cortical thickness measurements across scanners and sites. *NeuroImage*, 167, 104-120.

二、使用neuroCombat包

1. `neuroCombat`就是上面文献作者开发的包，除了R环境，在Python和Matlab环境下也开发了类似的工具。关于`neuroCombat`的介绍：https://github.com/Jfortin1/ComBatHarmonization
2. 安装`neuroCombat`

```
library(devtools)
install_github("jfortin1/neuroCombat_Rpackage")
```

3. 基本用法

```
library(neuroCombat)
## 准备数据
## 假设all_dat表示包含所有数据的数据框，每行表示一个被试，其中10-100列是磁共振数据，每一列可能表示一个脑区或者体素。
mri_mat <- t(as.matrix(all_dat[, 10:100]))
## site_info表示每个被试数据对应的站点或者参数类别。
site_info <- all_dat$Site
## design_mat表示感兴趣的变量（比如诊断、年龄和性别）构成的设计矩阵，目的是为了在去除站点效应过程中，不要去除这些感兴趣变量相关的变异。
design_mat <- model.matrix(~Diagnosis+Age+Sex, data = all_dat)
## 校正数据，校正后的数据存放在adjust_dat$dat.combat矩阵里，每列表示一个被试
adjust_dat <- neuroCombat(dat=mri_mat, batch=site_info, mod=design_mat)
```

三、使用combat.enigma包

1. `neuroCombat`包的一个不足之处是，无法将在一个数据集拟合的模型参数运用到另一个另一个数据集上。比如，在机器学习中我们往往需要在训练集中去除站点效应，同时把模型参数直接运用到测试集上（所有数据一起去除站点效应会造成数据泄漏）。通过简单搜索，发现combat.enigma包提供了相关功能。

2. 安装`combat.enigma`

```
install.packages('combat.enigma')
```

3. 基本用法

```
library(combat.enigma)
## 准备数据
## 假设all_dat表示包含所有数据的数据框，每行表示一个被试，其中10-100列是磁共振数据，每一列可能表示一个脑区或者体素。
mri_dat <- all_dat[, 10:100]
## site_info表示每个被试数据对应的站点或者参数类别。
site_info <- all_dat$Site
## design_mat表示感兴趣的变量（比如诊断、年龄和性别）构成的设计矩阵，目的是为了在去除站点效应过程中，不要去除这些感兴趣变量相关的变异。
design_mat <- model.matrix(~Diagnosis+Age+Sex, data = all_dat)
## 校正数据，校正后的数据存放在adjust_dat$dat.combat矩阵里，每行表示一个被试
combat_model <- combat_fit(mri_dat, site_info, design_mat)
adjust_dat <- combat_apply(combat_model, mri_dat, site_info, design_mat)
```

不同于`neuroCombat`，`combat.enigma`校正过程包含两个步骤，首先拟合模型参数，再使用模型参数去除数据中的站点效应，这样使用场景更加灵活。通过比较，在同样数据下，

`neuroCombat` 和 `combat.enigma` 得到的结果是完全一致的。

四、校正前后简单比较

下图中是在我自己的数据上测试的结果，左边表示校正前两个站点的数据分布，右边表示校正后两个站点的数据分布，似乎校正后两个站点的方差更接近了。这里只展示了一个脑区的结果，其他脑区校正前后的结果也是类似的。