The pwrEWAS User's Guide

Stefan Graw, Devin C. Koestler

3 July 2018

Abstract

pwrEWAS is a computationally efficient tool to estimate power in EWAS as a function of sample and effect size for two-group comparisons of DNAm (e.g., case vs control, exposed vs non-exposed, etc.). Detailed description of in-/outputs, instructions and an example, as well as interpretations of the example results are provided in the following vignette.

Package

pwrEWAS 1.0

Contents

Introduction

When designing an epigenome-wide association study (EWAS) to investigate the relationship be-tween DNA methylation (DNAm) and some exposure(s) or phenotype(s), it is critically important to assess the sample size needed to detect a hypothesized difference with adequate statistical power. However, the complex and nuanced nature of DNAm data make direct assessment of statistical power challenging. To circumvent these challenges and to address outstanding need for a user-friendly interface for EWAS power evaluation, we have developed pwrEWAS. The current imple-mentation of pwrEWAS accommodates power estimation for two-group comparisons of DNAm (e.g., case vs control, exposed vs non-exposed, etc.). Power is calculated by means of a Monte Carlo approach in which DNAm data are randomly generated from one of several different existing data datasets, chosen to cover the most common tissue-types used in EWAS. In addition to specify-ing the tissue type to be used for DNAm profiling, users are required to specify the sample size, number of differentially methylated CpGs, effect size(s) (e.g., Δ_β), target false discovery rate (FDR) and the number of simulated data sets, and have the option of selecting from several different sta-tistical methods to perform differential methylation analyses. pwrEWAS reports the marginal power, marginal type I error rate, marginal FDR, and false discovery cost (FDC) .The R-Shiny web inter-face allows for easy input of user-defined parameters and includes an advanced settings button that offers additional options pertaining to data generation and computation.

Dependencies

This document has the following dependencies

```
library(shiny)
library(shinyBS)
library(ggplot2)
library(parallel)

library(car)
library(CpGassoc)
library(truncnorm)
library(limma)
library(genefilter)
```

Installation

pwrEWAS can be installed from guthub with the following R code:

```
devtools::install_github("stefangraw/pwrEWAS")
```

Usage

```
out = pwrEWAS(minTotSampleSize = 10,
                    maxTotSampleSize = 50,
                    SampleSizeSteps = 10,
                    NcntPer = 0.5,
                    targetDelta = c(0.2, 0.5),
                    J = 100000,
                    targetDmCpGs = 100,
                    tissueType = "Adult (PBMC)",
                    detectionLimit = 0.01,
                    DMmethod = "limma",
                    FDRcritVal = 0.05,
                    core = 4,
                    sims = 50)

myPlotCI3D(out$powerArray)
myDensityPlots(out$deltaArray, detectionLimit = 0.01)
```

Input parameter

Parameter	Decription
minTotSampleSize	Minimum total sample size
maxTotSampleSize	Maximum total sample size
SampleSizeSteps	Sample size increments
NcntPer	Percentage sample group 1
targetDelta	Target maximum difference in mean methylation
J	Number of CpGs tested/simulated
targetDmCpGs	Target number of DM CpGs
tissueType	Tissue type
detectionLimit	Detection Limit
DMmethod	Method of DM analysis
FDRcritVal	Target FDR
core	Threads
sims	Number of simulated data sets

Output parameter

Running pwrEWAS will result in an object with the following three attributes: meanPower, powerArray, and deltaArray. The first attribute, meanPower, is a 2D matrix with empirically estimated marginal mean power for sample sizes and target Δ_β's (averaged over simulated data sets). The second attribute, powerArray, provides the full set of empirically estimated marginal power for sample sizes, target Δ_β's, and simulated data sets in a 3D matrix. The third attribute, deltaArray, contains a 3D matrix with simulated Δ_β's for sample sizes, targetΔ_β, and simulated data sets.

Example

Running pwrEWAS

```
library(devtools)
devtools::install_github("stefangraw/pwrEWAS")
library(pwrEWAS)
set.seed(1234)
out = pwrEWAS(minTotSampleSize = 20,
                    maxTotSampleSize = 220,
                    SampleSizeSteps = 20,
                    NcntPer = 0.5,
                    targetDelta = c(0.1, 0.2, 0.4),
                    J = 100000,
                    targetDmCpGs = 100,
                    tissueType = "Adult (PBMC)",
                    detectionLimit = 0.01,
                    DMmethod = "limma",
                    FDRcritVal = 0.05,
                    core = 6,
                    sims = 50)
## [2018-07-03 15:25:38] Finding tau...done [2018-07-03 15:27:43]
## [2018-07-03 15:27:43] Running simulation ...done [2018-07-03 16:06:18]
```

When running pwrEWAS, first τ will be determined. The beginning and finish of this process willl be printed with time stamps ("[time stamp] Finding tau. . . done [time stamp]"). Next, pwrEWAS will run the simulations to empirically estimate power. Likewise, the beginning and finish of this procress will be printed with time stamps: "[time stamp] Running simulation . . . done [time stamp]".

Running pwrEWAS will result in an object out, that stores the three attributes:

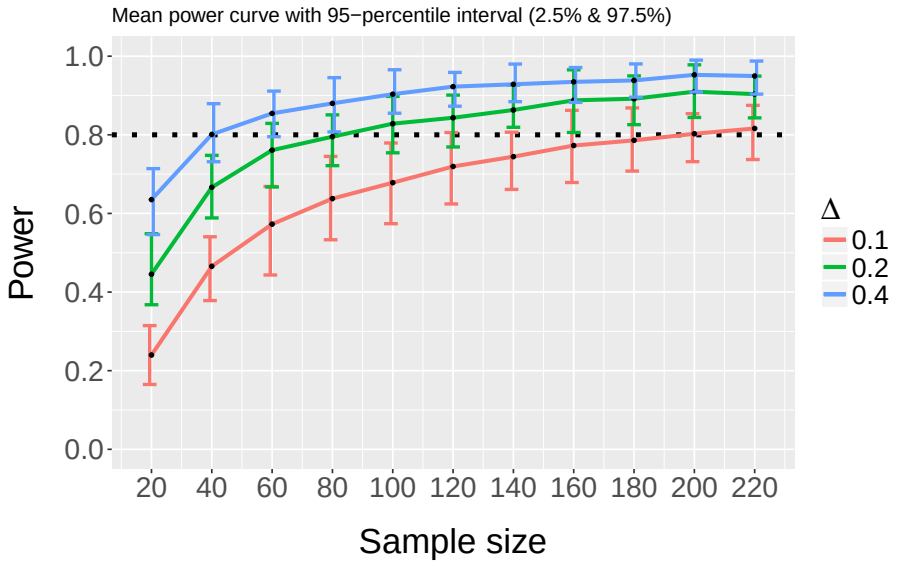
```
str(out)
## List of 3
##  $ meanPower : num [1:11, 1:3] 0.24 0.465 0.573 0.638 0.678 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:11] "20" "40" "60" "80" ...
##   .. ..$ : chr [1:3] "0.1" "0.2" "0.4"
##  $ powerArray: num [1:50, 1:11, 1:3] 0.255 0.228 0.237 0.317 0.2 ...
##   ..- attr(*, "dimnames")=List of 3
##   .. ..$ : chr [1:50] "1" "2" "3" "4" ...
##   .. ..$ : chr [1:11] "20" "40" "60" "80" ...
##   .. ..$ : chr [1:3] "0.1" "0.2" "0.4"
##  $ deltaArray:List of 3
##   ..$ 0.1: num [1:6400, 1:11] -0.0454 -0.01307 -0.00665 0.00751 -0.03075 ...
##   .. ..- attr(*, "dimnames")=List of 2
##   .. .. ..$ : NULL
##   .. .. ..$ : chr [1:11] "20" "40" "60" "80" ...
##   ..$ 0.2: num [1:5650, 1:11] 0.13095 -0.06722 -0.00559 -0.00918 0.05659 ...
##   .. ..- attr(*, "dimnames")=List of 2
##   .. .. ..$ : NULL
##   .. .. ..$ : chr [1:11] "20" "40" "60" "80" ...
##   ..$ 0.4: num [1:5350, 1:11] 0.0919 -0.0289 0.0602 0.1717 0.0265 ...
##   .. ..- attr(*, "dimnames")=List of 2
##   .. .. ..$ : NULL
##   .. .. ..$ : chr [1:11] "20" "40" "60" "80" ...
```

It is recommended to print the attribute "meanPower", as it summarizes the attribute "powerArray". meanPower will be provide a 11x3 table with the average power by sample size as rows (20-260; steps=20) and by target Δ_β as columns (0.1, 0.2, 0.4):

```
dim(out$meanPower)
## [1] 11  3
print(out$meanPower)
##           0.1       0.2       0.4
## 20  0.2397924 0.4454216 0.6351795
## 40  0.4654836 0.6664461 0.8010337
## 60  0.5727758 0.7609746 0.8545229
## 80  0.6377192 0.7952403 0.8799925
## 100 0.6781776 0.8283171 0.9033560
## 120 0.7194872 0.8434184 0.9225365
## 140 0.7444723 0.8631514 0.9284496
## 160 0.7727335 0.8877466 0.9347372
## 180 0.7858769 0.8917593 0.9383745
## 200 0.8026668 0.9097081 0.9525265
## 220 0.8161831 0.9035558 0.9496127
```

The attribute "powerArray" should primarily be to create a power plot, but can also be used to investigate the power results for the individual simulations. In pwrEWAS included is a function "myPlotCI3D", that will create a power plot, where power (y-axis) is shown as a function of sample sizes (x-axis) for different effect sizes (seperate colored lines). For each sample size, the mean power as well as the 95%tile interval (2.5% and 97.5%) is shown.

```
dim(out$powerArray) # simulations x sample sizes x effect sizes
## [1] 50 11  3
myPlotCI3D(out$powerArray)
```



The power plots indicates, that for an expected maximum difference of mean DNAm of 0.1, 0.2, and 0.4, a total sample size of at least about 40, 80, and 200 patients, is required to archive 80% power for detecting these differences, respectively.

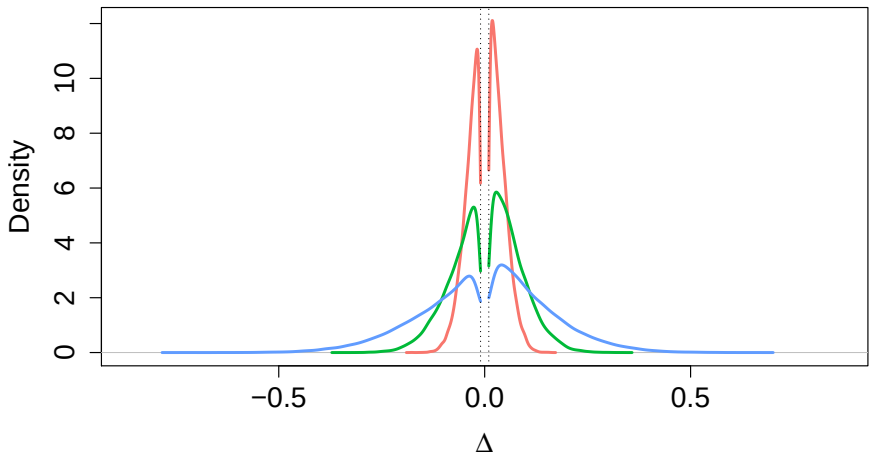
The third attribute "deltaArray" contains the simulated differences in mean DNAm. Even though the maximum value of simulated differences in greater than the target differences in DNAm, more than 99% of the simulated differences is smaller than the target differences in DNAm.

```
# maximum value of simulated differences by target value
lapply(out$deltaArray, max)
## $`0.1`
## [1] 0.1594534
##
## $`0.2`
## [1] 0.333524
##
## $`0.4`
## [1] 0.6560669

# percentage of simulated differences to be within the target range
mean(out$deltaArray[[1]] < 0.1)
## [1] 0.9955682
mean(out$deltaArray[[2]] < 0.2)
## [1] 0.9951891
mean(out$deltaArray[[3]] < 0.4)
## [1] 0.9956839
```

To get a better understanding of how the differences in mean DNAm look like, pwrEWAS provides a density plot, where the distribution of siumlated differences in mean DNAm is plotted by target differences in DNAm (Δ_β). The color theme matches the colors of the power plot. Simulated differences within the detection limit around zero are removed, as they are here not defined as real/meaningful differences.

```
myDensityPlots(out$deltaArray, detectionLimit = 0.01)
```



SessionInfo

- R version 3.4.1 (2017-06-30), `x86_64-w64-mingw32`
- Locale: `LC_COLLATE=English_United States.1252`, `LC_CTYPE=English_United States.1252`, `LC_MONETARY=English_United States.1252`, `LC_NUMERIC=C`, `LC_TIME=English_United States.1252`
- Running under: `Windows 7 x64 (build 7601) Service Pack 1`
- Matrix products: default
- Base packages: base, datasets, graphics, grDevices, methods, parallel, stats, utils
- Other packages: abind 1.4-5, BiocStyle 2.6.1, car 3.0-0, carData 3.0-1, CpGassoc 2.60, devtools 1.13.5, doParallel 1.0.11, foreach 1.4.4, genefilter 1.60.0, ggplot2 2.2.1, iterators 1.0.9, limma 3.34.9, nlme 3.1-131, pwrEWAS 0.0.0.9000, shiny 1.1.0, shinyBS 0.61, truncnorm 1.0-8
- Loaded via a namespace (and not attached): annotate 1.56.2, AnnotationDbi 1.40.0, backports 1.1.2, Biobase 2.38.0, BiocGenerics 0.24.0, bit 1.1-14, bit64 0.9-7, bitops 1.0-6, blob 1.1.1, bookdown 0.7, cellranger 1.1.0, codetools 0.2-15, colorspace 1.3-2, compiler 3.4.1, curl 3.2, data.table 1.11.4, DBI 1.0.0, digest 0.6.15, evaluate 0.10.1, forcats 0.3.0, foreign 0.8-69, git2r 0.21.0, grid 3.4.1, gtable 0.2.0, haven 1.1.1, htmltools 0.3.6, httpuv 1.4.3, httr 1.3.1, IRanges 2.12.0, knitr 1.20, later 0.7.3, lattice 0.20-35, lazyeval 0.2.1, magrittr 1.5, Matrix 1.2-10, memoise 1.1.0, mime 0.5, munsell 0.5.0, openxlsx 4.1.0, pillar 1.2.3, plyr 1.8.4, promises 1.0.1, R6 2.2.2, Rcpp 0.12.17, RCurl 1.95-4.10, readxl 1.1.0, rio 0.5.10, rlang 0.2.1, rmarkdown 1.10, rprojroot 1.3-2, RSQLite 2.1.1, S4Vectors 0.16.0, scales 0.5.0, splines 3.4.1, stats4 3.4.1, stringi 1.1.7, stringr 1.3.1, survival 2.41-3, tibble 1.4.2, tools 3.4.1, withr 2.1.2, xfun 0.2, XML 3.98-1.11, xtable 1.8-2, yaml 2.1.19, zip 1.0.0

References