`methyvim`: Targeted and Model-free Differential Methylation Analysis in R

Nima S. Hejazi[*1,2], **Rachael V. Phillips**[1], **Alan E. Hubbard**[1], **and Mark J. van der Laan**[1,3]

[1]**Group in Biostatistics, University of California, Berkeley**
[2]**Center for Computational Biology, University of California, Berkeley**
[3]**Department of Statistics, University of California, Berkeley**

Abstract We present `methyvim`, an R package implementing a general algorithm for the nonparametric estimation of treatment effects on DNA methylation at CpG sites throughout the genome, complete with straightforward statistical inference for such estimates. The approach leverages variable importance measures derived from statistical parameters arising causal inference, defined in such a manner that they may be used to obtain targeted estimates of the relative importance of individual CpG sites with respect to a binary treatment assigned at the phenotype level. The procedure implemented is computationally efficient, incorporating a preliminary screening step to isolate a subset of sites for which there is cursory evidence of differential methylation as well as a unique multiple testing correction to control the False Discovery Rate with the same rigor as if all sites were tested. This technique for analysis of differentially methylated positions provides an avenue to incorporate flexible state-of-the-art machine learning algorithms into the estimation of differential methylation effects without the loss of interpretable statistical inference.

Keywords

DNA methylation, differential methylation, epigenetics, causal inference, statistical variable importance, machine learning, targeted minimum loss-based estimation

[*]`nhejazi@berkeley.edu`

Introduction

DNA methylation is a fundamental epigenetic process known to play an important role in the regulation of gene expression. DNA methylation mostly commonly occurs at CpG sites and involves the addition of a methyl group (CH_3) to the fifth carbon of the cytosine ring structure to form 5-methylcytosine. Numerous biological and medical studies have implicated DNA methylation as playing a role in disease and development [1]. Perhaps unsurprisingly then, biotechnologies have been developed to rigorously probe the molecular mechanisms of this epigenetic process. Modern assays, like the Illumina *Infinium* HumanMethylation BeadChip assay, allow for quantitative interrogation of DNA methylation, at single-nucleotide resolution, across a comprehensive set of CpG sites scattered across the genome; moreover, the computational biology community has invested significant effort in the development of tools for properly removing technological effects that may contaminate biological signatures measured by such assays [2, Dedeurwaerder et al. [3]]. Despite these advances in both biological and bioninformatical techniques, most statistical methods available for differential analysis of data produced by such assays rely on over-simplified models that do not readily extend to such high-dimensional data structures without restrictive modeling assumptions and the use of inferentially costly hypothesis testing corrections. When these standard assumptions are violated, estimates of the population-level effect of an exposure or treatment may suffer from large bias. What's more, reliance on restrictive and misspecified statistical models naturally leads to biased effect estimates that are not only misleading in assessing effect sizes but also result in false discoveries as these biased estimates are subject to testing and inferential procedures. Such predictably unreliable methods serve only to produce findings that are later invalidated by replication studies and add still further complexity to discovering biological targets for potential therapeutics. Data-adaptive estimation procedures that utilize machine learning provide a way to overcome many of the problems common in classical methods, controlling for potential confounding even in high-dimensional settings; however, interpretable statistical inference (i.e., confidence intervals and hypothesis tests) from such data-adaptive estimates is challenging to obtain [4].

In this paper, we briefly present an alternative to such statistical analysis approaches in the form of a nonparametric estimation procedure that provides simple and readily interpretable statistical inference, discussing at length a recent implementation of the methodology in the `methyvim` R package. Inspired by recent advances in statistical causal inference and machine learning, we provide a computationally efficient technique for obtaining targeted estimates of nonparametric *variable importance measures* (VIMs) [5], estimated at a set of pre-screened CpG sites, controlling for the False Discovery Rate (FDR) as if all sites were tested. Under standard assumptions (e.g., identifiability, strong ignorability) [6], targeted minimum loss-based estimators of regular asymptotically linear estimators have sampling distributions that are asymptotically normal, allowing for reliable point estimation and the construction of Wald-style confidence intervals [7, van der Laan and Rose [8]]. As counterfactuals are contrary-to-fact – that is, defined in terms of missing data – in the context of DNA methylation, we define the counterfactual outcomes under a binary treatment as the observed methylation (whether Beta- or M-) values a CpG site would have if all subjects were given the treatment and the methylation values a CpG site would have if treatment were withheld from all subjects. Although these counterfactual outcomes are, of course, impossible to observe, they have statistical analogs that may be reliably estimated from observed data [6]. We describe an algorithm that incorporates the estimation of VIMs using *targeted minimum loss-based estimation* (**TMLE**) [9], deferring detailed description and analysis of the statistical methodology to work outside the scope of the present manuscript. This methodology assesses the individual importance of a given CpG site by utilizing state-of-the-art machine learning algorithms in the estimation of a targeted VIM, initially proposed and explored in [10]. In the present work, we focus on a software package, `methyvim`, that implements a variant of this methodology, specifically tailored to differential methylation analysis.

For a general discussion of the framework of targeted minimum loss-based estimation and the role this approach plays in statistical causal inference, the interested reader is invited to consult van der Laan and Rose [7] and van der Laan and Rose [8]. For a more general introduction to (statistical) causal inference, Pearl [6] and Hernan and Robins [11] may be of interest.

Methods

Data Structure

We consider an observed data structure $O = (W_k, A, Y_j)$, where $Y_j, j = 1, \ldots J$ represents a given CpG site of interest, $A \in \{0, 1\}$ represents a binary phenotype-level treatment, and $W_k, k = 1, \ldots K(j)$ is a matrix of the observed values of CpG sites in the same neighborhood as j as well as any potential phenotype-level confounders (e.g., sex, age). We consider having access to measurements on a large number G of CpG sites (e.g., 850,000, as measured by the Illumina MethylationEPIC BeadChip arrays), and as a matter of practicality, let $J \leq G$, so that J indexes only those CpG sites that pass an arbitrary pre-screening procedure; note that equality is only obtained in the case that all sites pass the pre-screening procedure. Further, we let $K(j)$ index the set of all CpG sites so that k, the realization of $K(j)$ for a given CpG site j, is a set of the CpG sites that neighbor the CpG site j, where the definition of a neighborhood is left as user-specified; thus, W_k is a matrix of observed methylation values for the set of CpG sites k that neighbor a particular target site j as well as any potential phenotype-level confounders, where the latter does not vary between realizations of $K(j)$. As a

variable importance measure, we see to estimate a target parameter (ψ_j) that quantifies the effect of changing a treatment A on the methylation of a CpG site j, accounting for any potential confounding from phenotype-level covariates and the observed methylation at the set of sites k that neighbor the site j. After a complete run of the procedure, we have access to j VIM estimates ψ_j, each corresponding to the degree to which a given CpG site j was responsive to a change in the binary treatment A, after controlling for potential confounding. When machine learning estimators are employed in estimating ψ_j, this produces a nonparametric variable importance measure of differential methylation, allowing for the identification of differentially methylated positions (DMPs). We note that $\psi(j)$, for any given j, is a function of the *true* data-generating distribution; so, to formalize, we consider observing n iid copies of O (i.e., $O_1, \ldots, O_n$), where we have $O \sim P \in \mathcal{M}$, which is to say that the random variable O is governed by an unknown probability distribution P, assumed only to reside in a nonparametric statistical model $\mathcal{M}$ that places no restrictions on the data-generating process.

Implementation

The core functionality of this package is made available via the eponymous `methyvim` function, which implements a statistical algorithm designed to compute targeted estimates of VIMs, defined in such a way that the VIMs represent parameters of scientific interest in computational biology experiments; moreover, these VIMs are defined such that they may be estimated in a manner that is very nearly assumption-free, that is, within a *fully nonparametric statistical model*. **The statistical algorithm consists in several major steps**:

1. Pre-screening of genomic sites is used to isolate a subset of sites for which there is cursory evidence of differential methylation. For the sake of computational feasibility, targeted minimum loss-based estimates of VIMs are computed only for this subset of sites. Currently, the available screening approach adapts core routines from the `limma` R package. Future releases will support functionality from other packages (e.g., `randomForest`, `tmle.npvi`). Following the style of the function for performing screening via `limma`, users may write their own screening functions and are invited to contribute such functions to the core software package by opening pull requests at the GitHub repository.

2. Nonparametric estimates of VIMs, for the specified target parameter, are computed at each of the CpG sites passing the screening step. The VIMs are defined in such a way that the estimated effects is of an exposure/treatment on the methylation status of a target CpG site, controlling for the observed methylation status of the neighbors of that site. Currently, routines are adapted from the `tmle` R package. Future releases will support doubly-robust estimates of these VIMs (via the `drtmle` package) and add parameters for continuous treatments/exposures (via the `tmle.npvi` and `txshift` R packages).

3. Since pre-screening is performed prior to estimating VIMs, we make use of a multiple testing correction uniquely suited to such settings. Due to the multiple testing nature of the estimation problem, a variant of the Benjamini and Hochberg procedure for controlling the False Discovery Rate (FDR) is applied [12]. Specifically, we apply a modified marginal Benjamini and Hochberg step-up False Discovery Rate controlling procedure for multi-stage analyses (FDR-MSA), which has established theoretical guarantees to control the FDR at the same rate as if all sites were tested (i.e., without screening) [13].

Parameters of Interest For the CpG sites that pass the pre-screening step, a user-specified target parameter of interest is estimated independently at each site. *In all cases, an estimator of the parameter of interest is constructed via targeted minimum loss-based estimation*.

For discrete-valued treatments or exposures:

- The average treatment effect (ATE): The effect of a binary exposure or treatment on the observed methylation at a target CpG site is estimated, controlling for the observed methylation at all other CpG sites in the same neighborhood as the target site, based on an additive form. Often denoted $\psi_0 = \psi_0(1) - \psi_0(0)$, the parameter estimate represents the additive difference in methylation that would have been observed at the target site had all observations received the treatment versus the counterfactual under which none received the treatment.

- The relative risk (RR): The effect of a binary exposure or treatment on the observed methylation at a target CpG site is estimated, controlling for the observed methylation at all other CpG sites in the same neighborhood as the target site, based on a geometric form. Often denoted, $\psi_0 = \frac{\psi_0(1)}{\psi_0(0)}$, the parameter estimate represents the multiplicative difference in methylation that would have been observed at the target site had all observations received the treatment versus the counterfactual under which none received the treatment.

Estimating the VIM corresponding to the parameters above, for discrete-valued treatments or exposures, requires two separate regression steps: one for the treatment mechanism (propensity score) and one for the outcome regression. Technical details on the nature of these regressions are discussed in Hernan and Robins

[11], and details for estimating these regressions in the framework of targeted minimum loss-based estimation are discussed in van der Laan and Rose [7].

Support for continuous-valued treatments or exposures is *planned but not yet available*. Future releases will allow users to assess continuous-valued treatments by relying on parameters estimable through implementations available in the following software packages:

- A nonparametric variable importance measure (NPVI) [14] (R package `tmle.npvi`). The effect of a continuous-valued exposure or treatment (the observed methylation at a target CpG site) on an outcome of interest is estimated, controlling for the observed methylation value at all other CpG sites in the same neighborhood as the target site. This uses a parameter that compares values of the treatment against a user-specified reference value taken to be the null value. In particular, the implementation to be provided is designed to assess the effect of differential methylation at the target CpG site on an outcome of interest (e.g., survival), providing a nonparametric evaluation of the impact of methylation at the target site.
- The causal effect of shifting the value of an observed intervention, defined as the counterfactual outcome under a posited shift of a continuous-value treatment of interest using stochastic intervention policies [15, Díaz and van der Laan [16], Hejazi et al. [17]] (R package `txshift`). The value of an outcome of interest under an unobserved value of the (continuous-valued) treatment, specified through a user-provided *additive shift* of the observed treatment, may be data-adaptively estimated. This allows for the effect of changes/shifts in the observed methylation at a CpG site on an outcome of interest to be evaluated, providing a nonparametric evalution of the relative importance of a particular target CpG site with respect to another variable measured in the same study.

Class `methytmle` We have adopted a class `methytmle` to help organize the functionality within this package. The `methytmle` class builds upon the `GenomicRatioSet` class provided by the `minfi` package so all of the slots of `GenomicRatioSet` are contained in a `methytmle` object. The new class introduced in the `methyvim` package includes several new slots:

- `call` - the form of the original call to the `methyvim` function.
- `screen_ind` - indices identifying CpG sites that pass the screening process.
- `clusters` - non-unique IDs corresponding to the manner in wich sites are treated as neighbors. These are assigned by genomic distance (bp) and respect chromosome boundaries (produced via a call to `bumphunter::clusterMaker`).
- `var_int` - the treatment/exposure status for each subject. Currently, these must be binary, due to the definition of the supported targeted parameters.
- `param` - the name of the target parameter from which the estimated VIMs are defined.
- `vim` - a table of statistical results obtained from estimating VIMs for each of the CpG sites that pass the screening procedure.
- `ic` - the measured array values for each of the CpG sites passing the screening, transformed into influence curve space based on the chosen target parameter.

We refer the reader to the package vignette, "`methyvim`: Targeted Data-Adaptive Estimation and Inference for Differential Methylation Analysis," included in any distribution of the software package, for further details.

Operation

A standard computer with the latest version of R and Bioconductor 3.6 installed will handle applications of the `methyvim` package.

Use Cases

To examine the practical applications and the full set of utilities of the `methyvim` package, we will use a publicly available example data set produced by the Illumina 450K array, from the `minfiData` R package.

Preliminaries: Setting up the Data We begin by loading the package and the data set. After loading the data, which comes in the form of a raw `MethylSet` object, we perform some further processing by mapping to the genome (with `mapToGenome`) and converting the values from the methylated and unmethylated channels to Beta-values (via `ratioConvert`). These two steps together produce an object of class `GenomicRatioSet`, provided by the `minfi` package.

```
suppressMessages(library(minfiData))
data(MsetEx)
mset <- mapToGenome(MsetEx)
grs <- ratioConvert(mset)
grs
```

```
## class: GenomicRatioSet
## dim: 485512 6
## metadata(0):
## assays(2): Beta CN
## rownames(485512): cg13869341 cg14008030 ... cg08265308 cg14273923
## rowData names(0):
## colnames(6): 5723646052_R02C02 5723646052_R04C01 ...
##   5723646053_R05C02 5723646053_R06C02
## colData names(13): Sample_Name Sample_Well ... Basename filenames
## Annotation
##   array: IlluminaHumanMethylation450k
##   annotation: ilmn12.hg19
## Preprocessing
##   Method: Raw (no normalization or bg correction)
##   minfi version: 1.21.2
##   Manifest version: 0.4.0
```

We can create an object of class `methytmle` from any `GenomicRatioSet` object simply invoking the S4 class constructor `.methytmle`:

```
library(methyvim)
```

```
## methyvim v1.3.1: Targeted Variable Importance for Differential Methylation Analysis
```

```
grs_mtmle <- .methytmle(grs)
grs_mtmle
```

```
## class: methytmle
## dim: 485512 6
## metadata(0):
## assays(2): Beta CN
## rownames(485512): cg13869341 cg14008030 ... cg08265308 cg14273923
## rowData names(0):
## colnames(6): 5723646052_R02C02 5723646052_R04C01 ...
##   5723646053_R05C02 5723646053_R06C02
## colData names(13): Sample_Name Sample_Well ... Basename filenames
## Annotation
##   array: IlluminaHumanMethylation450k
##   annotation: ilmn12.hg19
## Preprocessing
##   Method: Raw (no normalization or bg correction)
##   minfi version: 1.21.2
##   Manifest version: 0.4.0
## Target Parameter:
## Results:
## Object of class "data.frame"
## data frame with 0 columns and 0 rows
```

Additionally, a `GenomicRatioSet` can be created from a matrix with the function `makeGenomicRatioSetFromMatrix` provided by the `minfi` package.

Differential Methylation Analysis For this example analysis, we'll treat the condition of the patients as the exposure/treatment variable of interest. The `methyvim` function requires that this variable either be `numeric` or easily coercible to `numeric`. To facilitate this, we'll simply convert the covariate (currently a `character`):

```
var_int <- (as.numeric(as.factor(colData(grs)$status)) - 1)
```

n.b., the re-coding process results in "normal" patients being assigned a value of 1 and cancer patients a 0.

Now, we are ready to analyze the effects of cancer status on DNA methylation using this data set. We proceed as follows with a targeted minimum loss-based estimate of the Average Treatment Effect.

```
methyvim_cancer_ate <- methyvim(data_grs = grs, var_int = var_int,
                                vim = "ate", type = "Beta", filter = "limma",
                                filter_cutoff = 0.20, obs_per_covar = 2,
                                parallel = FALSE, sites_comp = 250,
                                tmle_type = "glm"
                               )
```

Note that we set the `obs_per_covar` argument to a relatively low value (just 2, even though the recommended value, and default, is 20) for the purposes of this example as the sample size is only 10. We do this only to exemplify the estimation procedure and it is important to point out that such low values for `obs_per_covar` will compromise the quality of inference obtained because this setting directly affects the definition of the target parameter.

Further, note that here we apply the `glm` flavor of the `tmle_type` argument, which produces faster results by fitting models for the propensity score and outcome regressions using a limited number of parametric models. By contrast, the `sl` (for "Super Learning") flavor fits these two regressions using highly nonparametric and data-adaptive procedures (i.e., via machine learning). Obtaining the estimates via GLMs results in each of the regression steps being less robust than if nonparametric regressions were used.

We can view a table of results by examining the `vim` slot of the produced object, most easily displayed by simply printing the resultant object:

```
methyvim_cancer_ate
```

```
## class: methytmle
## dim: 485512 6
## metadata(0):
## assays(2): Beta CN
## rownames(485512): cg13869341 cg14008030 ... cg08265308 cg14273923
## rowData names(0):
## colnames(6): 5723646052_R02C02 5723646052_R04C01 ...
##   5723646053_R05C02 5723646053_R06C02
## colData names(13): Sample_Name Sample_Well ... Basename filenames
## Annotation
##   array: IlluminaHumanMethylation450k
##   annotation: ilmn12.hg19
## Preprocessing
##   Method: Raw (no normalization or bg correction)
##   minfi version: 1.21.2
##   Manifest version: 0.4.0
## Target Parameter: Average Treatment Effect
## Results:
##                     lwr_ci       est_ate        upr_ci        var_ate
## cg14008030 -0.11956597277 -0.0314159619  0.0567340489 0.002022705230
## cg20253340 -0.08850636712 -0.0588661418 -0.0292259165 0.000228691940
## cg21870274 -0.09499057246 -0.0291189817  0.0367526091 0.001129494604
## cg17308840 -0.04626018088 -0.0071524518  0.0319552773 0.000398119136
## cg00645010 -0.02677328296 -0.0134655543 -0.0001578256 0.000046099449
## cg27534567  0.06745648461  0.1157118536  0.1639672225 0.000606148645
## cg08258224  0.13650451332  0.3050884951  0.4736724770 0.007398104677
## cg20275697 -0.30210263597 -0.1231299608  0.0558427144 0.008337988977
## cg24373735 -0.04283533168  0.0076666975  0.0581687266 0.000663904349
## cg12445832 -0.06082410546  0.0150395574  0.0909032203 0.001498150599
## cg01097950 -0.04392159158  0.0658323796  0.1755863507 0.003135655504
```

```
## cg01782097 -0.01082071508  0.0010232901  0.0128672954 0.000036516155
##                                                                pval n_neighbors
## cg14008030 0.484846808198562295721956161287380382418632 51           0
## cg20253340 0.000099174260505874870181078351016878968948 73           0
## cg21870274 0.386253718236077125958161104790633544325828 55           2
## cg17308840 0.719994333171475764387992057891096919775009 16           2
## cg00645010 0.047340069041464802324092886465223273262381 55           2
## cg27534567 0.000002602935551169070311973863934840167644 32           1
## cg08258224 0.000389591369062320855808090325211878734990 02           1
## cg20275697 0.177515452774292231685748788549972232431173 32           0
## cg24373735 0.766048934364622624570984044112265110015869 14           0
## cg12445832 0.697602171797029990685246048087719827890396 12           0
## cg01097950 0.239737652131245593256636539081227965652942 66           0
## cg01782097 0.865530222445773045336636641877703368663787 84           1
##            n_neighbors_control max_cor_neighbors
## cg14008030                   0                NA
## cg20253340                   0                NA
## cg21870274                   1        0.94435796
## cg17308840                   1        0.94435796
## cg00645010                   2        0.52368097
## cg27534567                   0        0.93629683
## cg08258224                   0        0.93629683
## cg20275697                   0                NA
## cg24373735                   0                NA
## cg12445832                   0                NA
## cg01097950                   0                NA
## cg01782097                   1       -0.39410834
##  [ reached getOption("max.print") -- omitted 238 rows ]
```

Finally, we may compute FDR-corrected p-values, by applying a modified procedure for controlling the False Discovery Rate for multi-stage analyses (FDR-MSA) [13]. We do this by simply applying the `fdr_msa` function.

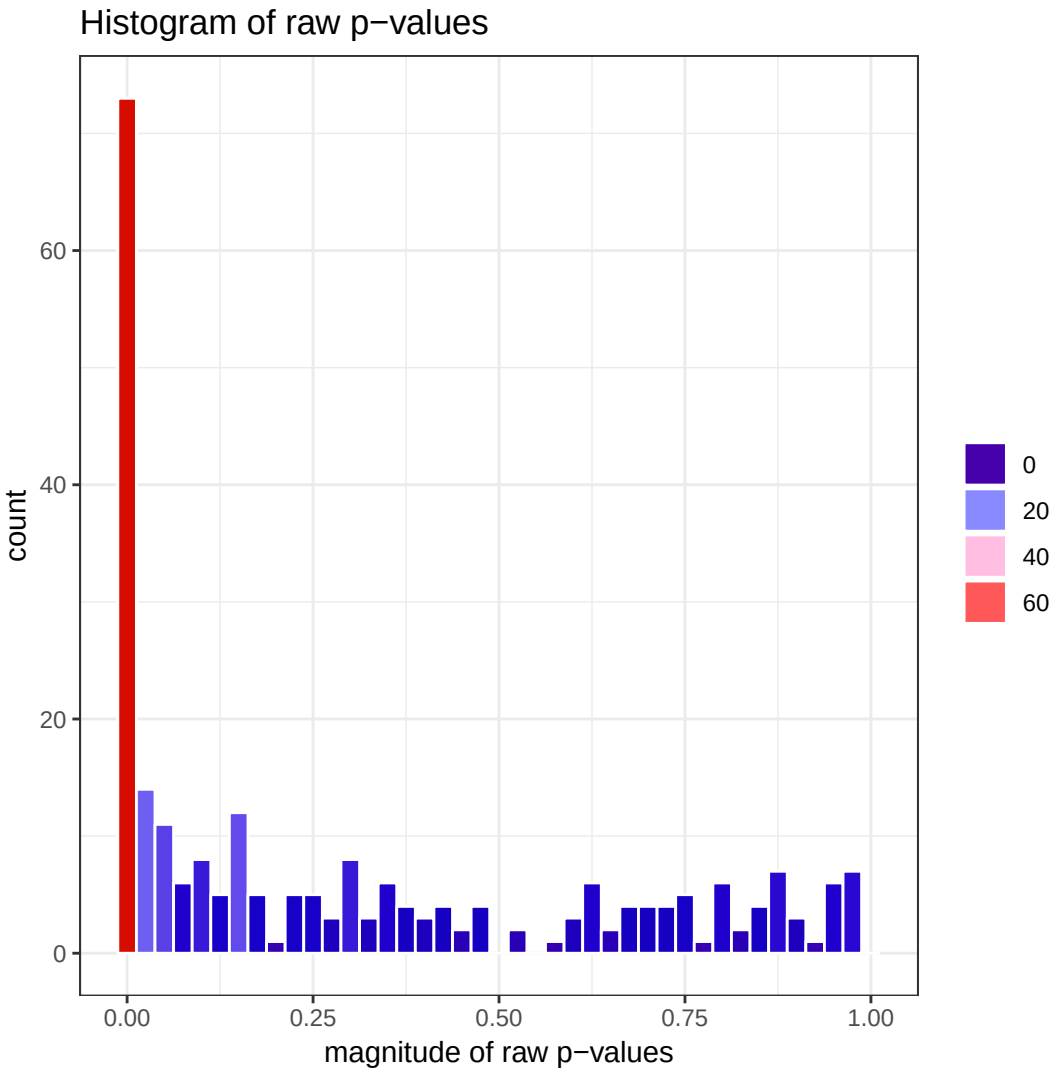
```
fdr_p <- fdr_msa(pvals = methyvim_cancer_ate@vim$pval,
                 total_obs = nrow(methyvim_cancer_ate))
```

Having explored the results of our analysis numerically, we now proceed to use the visualization tools provided with the `methyvim` R package to further enhance our understanding of the results.

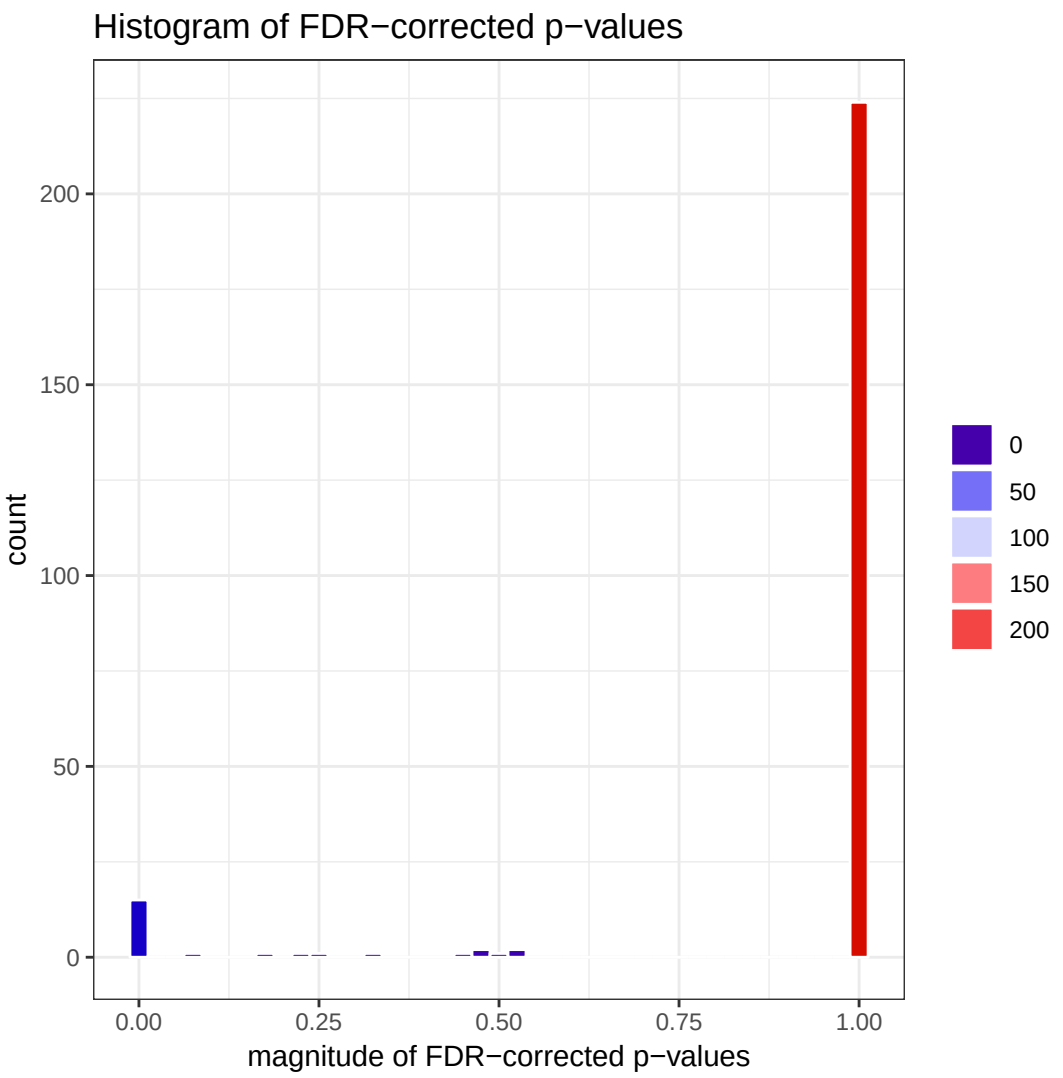
Visualization of Results While making allowance for users to explore the full set of results produced by the estimation procedure (by way of exposing these directly to the user), the `methyvim` package also provides *three* (3) visualization utilities that produce plots commonly used in examining the results of differential methylation analyses.

A simple call to `plot` produces side-by-side histograms of the raw p-values computed as part of the estimation process and the corrected p-values obtained from using the FDR-MSA procedure.

```
plot(methyvim_cancer_ate, type = "raw_pvals")
```



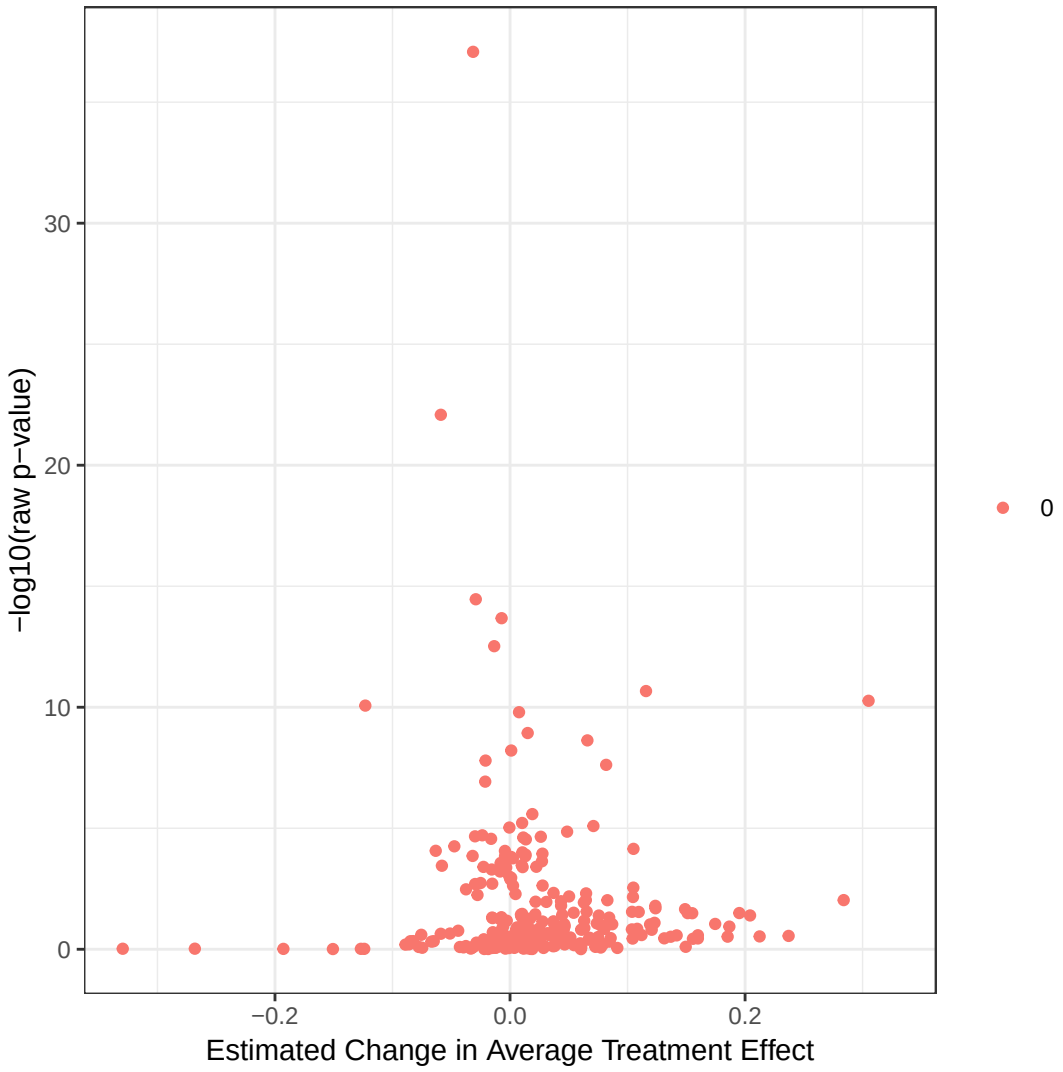

```
plot(methyvim_cancer_ate, type = "fdr_pvals")
```



Remark: The plots displayed above may also be generated as side-by-side histograms in a single plot object. This is the default for the `plot` method and may easily be invoked by specifying no additional arguments to the `plot` function, unlike in the above.

While histograms of the p-values may be generally useful in inspecting the results of the estimation procedure, a more common plot used in examining the results of differential methylation procedures is the volcano plot, which plots the parameter estimate along the x-axis and $-\log_{10}$(p-value) along the y-axis. We implement such a plot in the `methyvolc` function:

```
methyvolc(methyvim_cancer_ate)
```

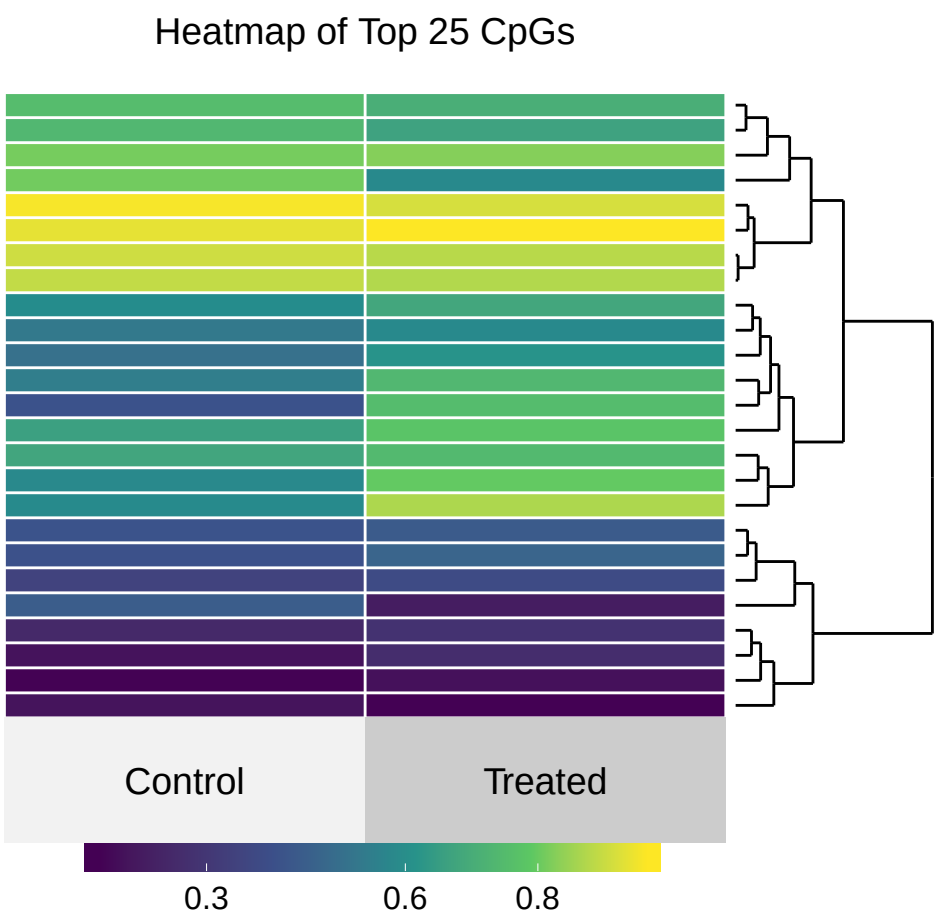


The purpose of such a plot is to ensure that very low (possibly statistically significant) p-values do not arise from cases of low variance. This appears to be the case in the plot above (notice that most parameter estimates are near zero, even in cases where the raw p-values are quite low).

Yet another popular plot for visualizing effects in such settings is the heatmap, which plots estimates of the raw methylation effects (as measured by the assay) across subjects using a heat gradient. We implement this in the `methyheat` function:

```
methyheat(methyvim_cancer_ate, smooth.heat = TRUE, left.label = "none")
```

```
## [1] 0.1 0.3 0.6 0.8 1.0
```



Remark: Invoking `methyheat` in this manner produces a plot of the top sites (25, by default) based on the raw p-value, using the raw methylation measures in the plot. This uses the exceptional `superheat` R package [**?**], to which we can easily pass additional parameters. In particulat, we hide the CpG site labels that would appear by default on the left of the heatmap (by setting `left.label = "none"`) to emphasize that this is only an example and *not* a scientific discovery.

Summary

Here we introduce the R package `methyvim`, an implementation of a general algorithm for differential methylation analysis that allows for recent advances in causal inference and machine learning to be leveraged in computational biology settings. The estimation procedure produces straightforward statistical inference and takes great care to ensure computationally efficiency of the technique for obtaining targeted estimates of nonparametric variable importance measures. The software package includes techniques for pre-screening a set of CpG sites, controlling for the False Discovery Rate as if all sites were tested, and for visualzing the results of the analyses in a variety of ways. The anatomy of the software package is dissected and the design described in detail. The `methyvim` R package is available via the Bioconductor project.

Software availability

Latest source code (development version): `https://github.com/nhejazi/methyvim`

Bioconductor (stable release): `https://bioconductor.org/packages/methyvim`

Archived source code as at time of publication: `https://github.com/nhejazi/methyvim/releases/tag/f1000`

Documentation (development version): `https://code.nimahejazi.org/methyvim`

Software license: The MIT License, copyright Nima S. Hejazi

Author contributions

NH designed and implemented the software package, applied the tool to the use cases presented, and co-drafted the present manuscript. RP helped in designing the software and co-drafted the present manuscript.

AH and ML served as advisors for the development of this software and the general statistical algorithm it implements.

Competing interests

No competing interests were disclosed.

Grant information

NH was supported in part by the National Library of Medicine of the National Institutes of Health under Award Number T32-LM012417, by P42-ES004705, and by R01-ES021369. RP was supported by P42-ES004705. The content of this work is solely the responsibility of the authors and does not necessarily represent the official views of the various funding sources and agencies.

References

[1] Keith D Robertson. DNA methylation and human disease. *Nature reviews. Genetics*, 6(8):597, 2005.

[2] Jean-Philippe Fortin, Aurelie Labbe, Mathieu Lemire, Brent W Zanke, Thomas J Hudson, Elana J Fertig, Celia MT Greenwood, and Kasper D Hansen. Functional normalization of 450k methylation array data improves replication in large cancer studies. *bioRxiv*, 2014.

[3] Sarah Dedeurwaerder, Matthieu Defrance, Martin Bizet, Emilie Calonne, Gianluca Bontempi, and François Fuks. A comprehensive overview of infinium humanmethylation450 data processing. *Briefings in bioinformatics*, page bbt054, 2013.

[4] Maxwell W Libbrecht and William Stafford Noble. Machine learning in genetics and genomics. *Nature Reviews. Genetics*, 16(6):321, 2015.

[5] Mark J van der Laan. Statistical inference for variable importance. *The International Journal of Biostatistics*, 2(1), 2006.

[6] Judea Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2009.

[7] Mark J van der Laan and Sherri Rose. *Targeted Learning: Causal Inference for Observational and Experimental Data*. Springer Science & Business Media, 2011.

[8] Mark J van der Laan and Sherri Rose. *Targeted Learning in Data Science: Causal Inference for Complex Longitudinal Studies*. Springer Science & Business Media, 2018.

[9] Mark J van der Laan and Daniel Rubin. Targeted maximum likelihood learning. *The International Journal of Biostatistics*, 2(1), 2006.

[10] Catherine Tuglus and Mark J van der Laan. Targeted methods for biomarker discovery, the search for a standard. 2008.

[11] Miguel A Hernan and James M Robins. *Causal Inference*. Chapman & Hall/CRC Texts in Statistical Science. Taylor & Francis, 2018, forthcoming.

[12] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the royal statistical society. Series B (Methodological)*, pages 289–300, 1995.

[13] Catherine Tuglus and Mark J. van der Laan. Modified FDR controlling procedure for multi-stage analyses. *Statistical Applications in Genetics and Molecular Biology*, 8(1):1–15, January 2009. doi: 10.2202/1544-6115.1397. URL `https://dx.doi.org/10.2202/1544-6115.1397`.

[14] Antoine Chambaz, Pierre Neuvial, and Mark J van der Laan. Estimation of a non-parametric variable importance measure of a continuous exposure. *Electronic Journal of Statistics*, 6:1059, 2012.

[15] Iván Díaz Muñoz and Mark J van der Laan. Population intervention causal effects based on stochastic interventions. *Biometrics*, 68(2):541–549, 2012.

[16] Iván Díaz and Mark J van der Laan. Stochastic treatment regimes. In *Targeted Learning in Data Science: Causal Inference for Complex Longitudinal Studies*, pages 167–180. Springer Science & Business Media, 2018.

[17] Nima S Hejazi, Mark J van der Laan, and David C Benkeser. *txshift: Targeted Learning of causal effects under stochastic treatment regimes in R*, 2018. URL `https://github.com/nhejazi/txshift`. R package version 0.2.0.