Enhancer Linking by Methylation/Expression Relationships with the R package *ELMER* version 2

Tiago Chedraoui Silva[1,2,*] Simon G. Coetzee[2] Lijing Yao[4] Dennis J. Hazelett[2] Houtan Noushmehr[1,3] Benjamin P. Berman[2,*]

[1]Department of Genetics, Ribeirão Preto Medical School, University of São Paulo, Ribeirão Preto, Brazil
[2]Center for Bioinformatics and Functional Genomics, Cedars-Sinai Medical Center, Los Angeles, CA, USA
[3]Department of Neurosurgery, Henry Ford Hospital, Detroit, MI, USA
[4]Roche Sequencing Solutions

∗ Corresponding authors, email tiago.chedraouisilva@cshs.org or Benjamin.Berman@cshs.org

Abstract

Recent studies indicate that DNA methylation can be used to identify changes at transcriptional enhancers and other cis-regulatory elements in primary human samples. A systematic approach to inferring gene regulatory networks has been provided by the R/Bioconductor package *ELMER* (Enhancer Linking by Methylation/Expression Relationships), which first identifies DNA methylation changes in distal regulatory elements and correlates these with expression of nearby genes to identify direct transcriptional targets. Next, *ELMER* performs a transcription factor binding motif analysis and integrates with expression profiling of all human transcription factors, to identify master regulatory TFs and place each differentially methylated regulatory element into the context of an altered gene regulatory network (GRN).

Here we present a completely updated version of the package (*ELMER* v. 2.0), which uses the latest Bioconductor data structures including the popular `MultiAsssayExperiment`, supports multiple reference genome assemblies as well as the DNA methylation platforms Infinium MethylationEPIC and Infinium HumanMethylation450, and provides a "Supervised" analysis mode for paired sample study designs (such as treated vs. untreated replicate samples). It also supports data import from the new NCI Genomic Data Commons (GDC) database. The new version is substantially re-written, improving stability, performance, and extensibility. It also uses improved databases for transcription factor binding domain families and binding motif specificities, and has newly designed output plots for publication-quality figures.

Below, we describe the methods and new features of ELMER v. 2.0, and present two use case demonstrating how the tool can be used to analyze TCGA data in either Unsupervised or Supervised mode. *ELMER* (v2.0.0) is available as an R/Bioconductor package at `https://github.com/tiagochst/ELMER`. Also, *ELMER.data* (v2.0.0), which provides auxiliary data required to perform the analysis, is available at `https://github.com/tiagochst/ELMER.data`.

Keywords

DNA methylation, gene regulatory networks, enhancers, chromatin interactions, chromQTLs, transcription factor binding sites, epigenetics, computational tools

Introduction

Motivated by our discovery of transcriptional enhancers in tissue DNA methylation data [1], and subsequent approaches to linking these enhancers to transcriptional targets using a chromQTL approach [2] (reviewed in [3]), we developed the the R/Bioconductor *ELMER* (Enhancer Linking by Methylation/Expression Relationships) package, a tool which infers regulatory element landscapes and transcription factor networks from cancer methylomes [4].

This tool combined DNA methylation and gene expression data from human tissues to infer multi-level cis-regulatory networks through several steps which included the identification of distal enhancer probes with significantly altered DNA methylation levels in primary tumor tissues compared to normal tissues, followed by the identification of putative target genes, and a comprehensive gene regulatory network analysis which combined transcription factor motifs at the altered enhancers with TF expression to identify the underlying master regulators. This approach identified several known and unknown master regulators in TCGA data, such as GATA3 and FOXA1 in breast cancer, and P63 and SOX2 in squamous cell lung carcinoma. [4, 5].

Based on user feedback and a full review of the source code, we identified and implemented a number of software improvements, which are summarized in table 1: (i) The original package contained no standard data structure to handle multiple assays (DNA methylation, gene expression and clinical data), which would be required for an integrative genomic data analysis. Recently, the Bioconductor team provided such a data structure through the MultiAssayExperiment package. (ii) All auxiliary databases (human TF list, classification of TF in families , gene annotation, DNA methylation annotation and motif occurrences within probe sites) used in the package were created and maintained manually, thereby making the upgrade process laborious; thus, we automated this process. (iii) The package was developed to analyze primary tumor tissue samples compared to normal tissues samples, thus not allowing arbitrary subgroups to be compared (for instance mutants vs. non-mutants, treated vs. untreated, etc.) (iv) Our original approach used known epigenomic markers for enhancers to constrain the genomic regions searched for differential methylation. However, this selection could limit our algorithm to identifying regulatory networks for tissue types that exist in the epigenomic databases; we found this constraint problematic, and thus now search *all* distal regulatory regions without any such filter. (v) The function used to download data from The Cancer Genome Atlas (TCGA) data portal [6] broke when the TCGA site was shutdown and its data transfered to The NCI's Genomic Data Commons (GDC) [7]; we now have a more general data provider interface that supports GDC as the default provider. (vi) The package only supported data aligned to Genome Reference Consortium GRCh37 (hg19), and we now provide support for Genome Reference Consortium GRCh38 (hg38). (vii) There was no support to the recent HumanMethylationEPIC (EPIC) array [8]. In addition to the specific improvements listed above, we substantially re-wrote most of the code to be more efficient and maintainable, and improved most of the output plots generated.

Here, we present a new version of the R *ELMER* package, which addresses all the issues described above. The new version of *ELMER* (v2.0.0) is available as an R/Bioconductor package at `https://github.com/tiagochst/ELMER`. And, the new version of *ELMER.data* (v2.0.0), which provides auxiliary data required to perform the analysis, is available at `https://github.com/tiagochst/ELMER.data`.

Table 1. Main differences between ELMER old version (v.1) and the new version (v.2)

Features	ELMER Version 1	ELMER Version 2
Primary data structure	mee object (custom data structure)	MAE object (Bioconductor data structure)
Auxiliary data	Manually created	Programmatically created
Number of human TFs	1,982	1,987 (Uniprot database [9])
Number of TF motifs	91	640 (HOCOMOCO v10 database [10])
TF classification	78 families	80 families and 308 subfamilies (TFClass database [11])
Analysis performed	Normal vs tumor samples	Group 1 vs group 2
Statistical grouping	unsupervised only	unsupervised or supervised using labeled groups
TCGA data source	The Cancer Genome Atlas (TCGA) (not available)	The NCI's Genomic Data Commons (GDC)
Genome of reference	GRCh37 (hg19)	GRCh37 (hg19)/GRCh38 (hg38)
DNA methylation platforms	HumanMethylation450	HumanMethylationEPIC and HumanMethylation450
Graphical User interface (GUI)	None	TCGAbiolinksGUI

Methods

Operation

The R *ELMER* package version 2.0.0 requires R version 3.4.0 or higher. It is open-source under the The GNU General Public License v3.0 (GPL-3) and can run on any operating system. It depends mainly on two R/Bioconductor packages: *TCGAbiolinks* [12] to download cancer data from NCI Genomic Data Commons (GDC), and *MultiAssayExperiment* [13] to create an R object with an integrative data structure. Finally, we provide a pipeline to perform *ELMER* analysis on cancer samples compared to normal tissue samples from GDC; however, it should be noted that to perform its analysis for some cancer types, a minimum of 16 Gb memory in R is required.

The latest development version can be installed in an R session from GitHub using the *devtools* package [14]:

Listing 1. "Install ELMER version 2.0 from github"

```
install.packages("devtools", dependencies = TRUE)
devtools::install_github("tiagochst/ELMER", dependencies = TRUE, build_vignettes
    = TRUE)
devtools::install_github("tiagochst/ELMER.data", dependencies = TRUE, build_
    vignettes = TRUE)
```

Having successfully installed the *ELMER* package, it can be loaded using `library("ELMER")` and a detailed vignette is available which can be viewed by typing `browseVignettes("ELMER")`

and `browseVignettes("ELMER.data")`. Figure 1 shows an overview of the workflow of the *ELMER* package.

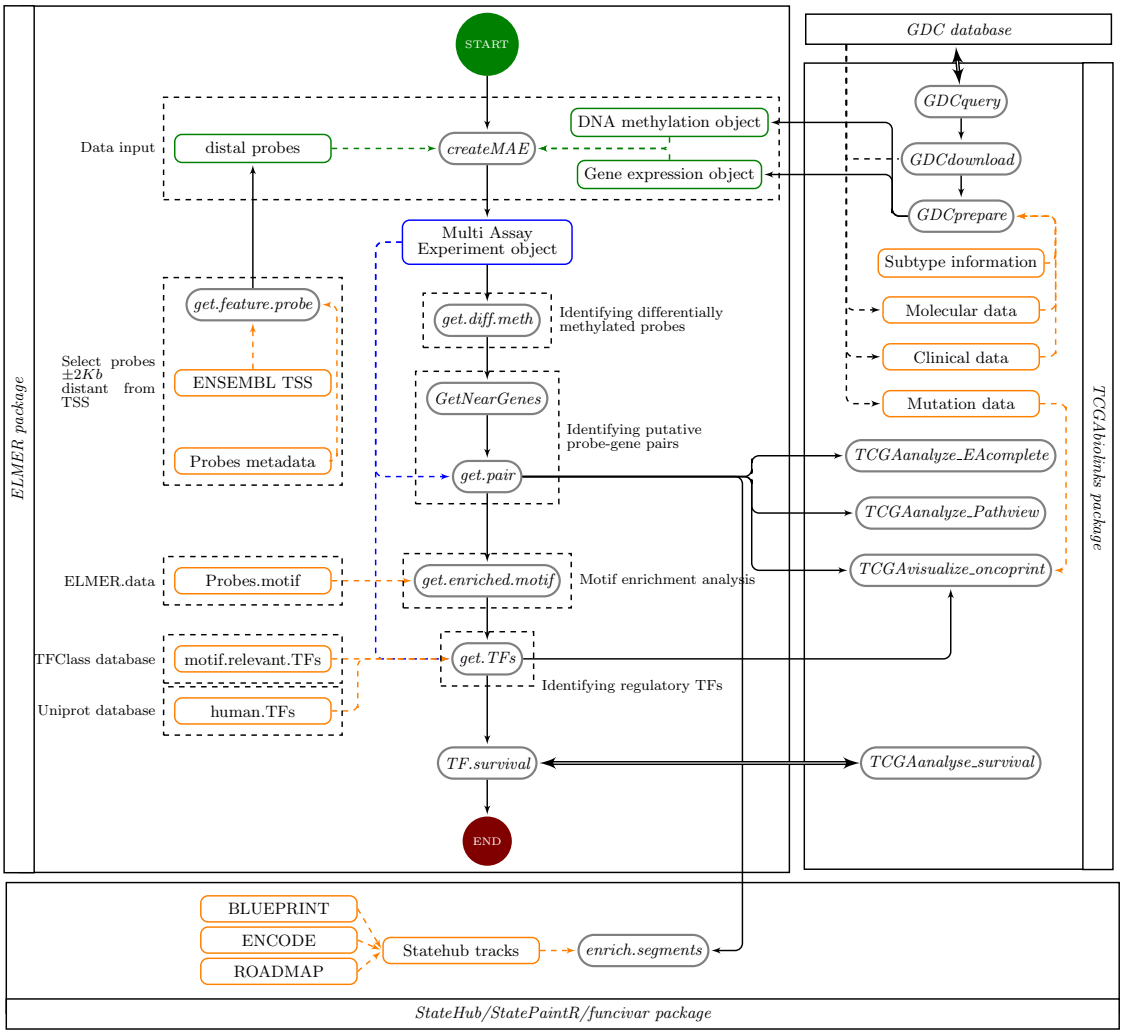

Figure 1. ELMER workflow: ELMER receives as input a DNA methylation object, a gene expression object (both can be either a matrix or a SummarizedExperiment object) and a Genomic Ranges (GRanges) object with distal probes to be used as filter which can be retrieved using the *get.feature.probe* function. The function *createMAE* will create a *Multi Assay Experiment* object keeping only samples that have both DNA methylation and gene expression data. Genes will be mapped to genomic position and annotated using ENSEMBL database [15], while for probes it will add annotation from `http://zwdzwd.github.io/InfiniumAnnotation`. This *MAE* object will be used as input to the next analysis functions. First, it identifies differentially methylated probes followed by the identification of their nearest genes (10 upstream and 10 downstream) through the *get.diff.meth* and *GetNearGenes* functions respectively. For each probe it will verify if any of the nearby genes were affected by its change in the DNA methylation level and a list of gene and probes pairs will be outputted from *get.pair* function. For the probes in those pairs, it will search for enriched regulatory Transcription Factors motifs with the *get.enriched.motif* function. Finally, the enriched motifs will be correlate with the level of the transcription factor through the *get.TFs* function. In the figure green Boxes represents user input data, blue boxes represents output object, orange boxes represents auxiliary pre-computed data and gray boxes are functions.

Implementation

Here we describe each of following analysis steps showed in figure 1. For more details, please also check the original ELMER paper Yao et al. [4].

- Organize data as a *MultiAssayExperiment* object
- Identify distal probes with significantly different DNA methylation level when comparing two sample groups.
- Identify putative target genes for differentially methylated distal probes, using methylation vs. expression correlation
- Identify enriched motifs for each probe belonging to a significant probe-gene pair
- Identify master regulatory Transcript Factors (TF) whose expression associate with DNA methylation changes at multiple regulatory regions.

Organization of data as a *MultiAssayExperiment* object

To facilitate the analysis of experiments and studies with multiple samples the Bioconductor team created the *SummarizedExperiment* class [16], a data structure able to store data and metadata for a single experiment but not for data spanning several experiments for the same sample. To overcome this problem, recently, the MultiAssay SIG (Special Interest Group) created the MultiAssayExperiment class[13] a data structure to manage and preprocess multiple assays for integrated genomic analysis. This data structure is now an input for all main functions of *ELMER*, and can be generated by *createMAE* function.

To perform *ELMER* analyses, we need to populate a *MultiAssayExperiment* with a DNA methylation matrix or *SummarizedExperiment* object from HM450K or EPIC platform; a gene expression matrix or SummarizedExperiment object for the same samples; a matrix mapping DNA methylation samples to gene expression samples; and a matrix with sample metadata (i.e. clinical data, molecular subtype, etc.). If TCGA data are used, the the last two matrices will be automatically generated. If using non-TCGA data, the matrix with sample metadata should be provided with at least a column with a patient identifier and another one identifying its group which will be used for analysis, if samples in the methylation and expression matrices are not ordered and with same names, a matrix mapping for each patient identifier their DNA methylation samples and their gene expression samples should be provided to the *createMAE* function. Based on the genome of reference selected, metadata for the DNA methylation probes, such as genomic coordinates, will be added from Zhou et al. [17] [17]; and metadata for gene expression and annotation is added from Ensembl database [15] using biomaRt [18].

Selecting distal probes

Probes from HumanMethylationEPIC (EPIC) array and Infinium HumanMethylation450 (HM450) array are removed from the analysis if they have either internal SNPs close to the $3'$ end of the probe; non-unique mapping to the bisulfite-converted genome; or off-target hybridization due to partial overlap with non-unique elements [19]. This probe metadata information is included in *ELMER.data* package, populated from the source file at `http://zwdzwd.github.io/InfiniumAnnotation` [19]. To limit ELMER to the analysis of distal elements, probes located in regions of $\pm 2kb$ around transcription start sites (TSSs) were removed.

Identification of differentially methylated CpGs (DMCs)

For each distal probe, samples of each group (group 1 and group 2) are ranked by their DNA methylation beta values, those samples in the lower quintile (20% samples with the lowest methylation levels) of each group are used to identify if the probe is hypomethylated in group 1 compared to group 2, using an unpaired one-tailed t-test. The 20% is a parameter to the *diff.meth* function called *minSubgroupFrac*. For the (ungrouped) cancer case, this is set to 20% as in [4], because we typically wanted to be able to detect a specific molecular subtype among the tumor samples; these subtypes often make up only a minority of samples, and 20% was chosen as a lower bound for the purposes of statistical power (high enough sample numbers to yield t-test p-values that could overcome multiple hypothesis correction, yet low enough to be able to capture changes in individual molecular subtypes occurring in 20% or more of the cases.) This number can be set arbitrarily as an input to the *diff.meth* function and should be tuned based on sample sizes in individual studies. In the *Supervised* mode, where the comparison groups are implicit in the sample set and labeled, the *minSubgroupFrac* parameter is set to 100%. An example would be a cell culture experiment with 5 replicates of the untreated cell line, and another 5 replicates that include an experimental treatment.

To identify hypomethylated DMCs, a one tailed t-test is used to rule out the null hypothesis: $\mu_{group1} \geq \mu_{group2}$, where μ_{group1} is the mean methylation within the lowest group 1 quintile (or other percentile as specified by the *minSubgroupFrac* parameter) and μ_{group2} is the mean within the lowest group 2 quintile. Raw p-values are adjusted for multiple hypothesis testing using the Benjamini-Hochberg method [20], and probes are selected when they had adjusted p-value less than 0.01 (which can be configured using the *pvalue* parameter). For additional stringency, probes are only selected if the methylation difference: $\Delta = \mu_{group1} - \mu_{group2}$ was greater than 0.3. The same method is used to identify hypermethylated DMCs, except we use the *upper* quintile, and the opposite tail in the t-test is chosen.

Identification of putative target gene(s)

For each differentially methylated distal probe (DMC), the closest 10 upstream genes and the closest 10 downstream genes are tested for inverse correlation between methylation of the probe and expression of the gene (the number 10 can be changed using the *numFlankingGenes* parameter). To select these genes, the probe-gene distance is defined as the distance from the probe to the transcription start site specified by the ENSEMBLE gene level annotations [15] accessed via the R/Bioconductor package biomaRt [18, 21]. By choosing a constant number of genes to test for each probe, our goal is to avoid systematic false positives for probes in gene rich regions. This is especially important given the highly non-uniform gene density of mammalian genomes. Thus, exactly 20 statistical tests were performed for each probe, as follows.

For each probe-gene pair, the samples (all samples from both groups) are divided into two groups: the M group, which consisted of the upper methylation quintile (the 20% of samples with the highest methylation at the enhancer probe), and the U group, which consists of the lowest methylation quintile (the 20% of samples with the lowest methylation.) The 20% ile cutoff is a configurable parameter *minSubgroupFrac* in the *get.pair* function. As with its usage in the *diff.meth* function, the default value of 20% is a balance, allowing for the identification of changes in a molecular subtype making up a minority (i.e. 20%) of cases, while also yielding enough statistical power to make strong predictions. For larger sample sizes or other experimental designs, this could be set even lower.

For each candidate probe-gene pair, the Mann-Whitney U test is used to test the null hypothesis

that overall gene expression in group M is greater than or equal than that in group U. This non-parametric test was used in order to minimize the effects of expression outliers, which can occur across a very wide dynamic range. For each probe-gene pair tested, the raw p-value P_r is corrected for multiple hypothesis using a permutation approach as follows. The gene in the pair is held constant, and x random methylation probes are chosen to perform the same one-tailed U test, generating a set of x permutation p-values P_p. We chose the x random probes only from among those that were "distal" (farther than $2kb$ from an annotated transcription start site), in order to draw these null-model probes from the same set as the probe being tested. An empirical p-value P_e value was calculated using the following formula (which introduces a pseudo-count of 1):

$$P_e = \frac{num(P_p \leq P_r) + 1}{x + 1}$$

Notice that in the *Supervised* mode, no additional filtering is necessary to ensure that the M and U group segregate by sample group labels. The two sample groups are segregated by definition, since these probes were selected for their differential methylation, with the same directionality, between the two groups.

Characterization of chromatin state context of enriched probes using FunciVar

Unlike version 1 of *ELMER*, we now consider *all* distal probes in the identification of regulatory elements. DNA methylation is known to affect several different classes of distal chromatin state element, including active enhancers, poised enhancers, and insulators. In order to provide functional interpretation of the regulatory elements identified by *ELMER*, we perform a chromatin state enrichment analysis of the probes within significant probe-gene pairs, using the *statePaintR* tools from the `statehub.org` suite[22], along with our new FunciVar package [23]. Enrichment of the putative pairs within chromatin states is calculated against a background model that uses the distal probe set that the putative pairs are drawn from.

Motif enrichment analysis

In order to identify enriched motifs and potential upstream regulatory TFs, all probes with occurring in significant probe-gene pairs are combined for motif enrichment analysis. HOMER (Hypergeometric Optimization of Motif EnRichment)[24] is used to find motif occurrences in a $\pm 250bp$ region around each probe, using HOCOMOCO (HOmo sapiens COmprehensive MOdel COllection) v10 [10] . Transcription factor (TF) binding models are available at `http://hocomoco.autosome.ru/downloads` (using the HOMER specific format with threshold score levels corresponding to p-value $< 1^{-4}$).

For each probe set tested (i.e. the set of all probes occurring in significant probe-gene pairs), a motif enrichment Odds Ratio and a 95% confidence interval are calculated using following formulas:

$$p = \frac{a}{a+b}$$

$$P = \frac{c}{c+d}$$

$$OddsRation = \frac{\frac{p}{(1-p)}}{\frac{P}{1-P}} = \frac{p(1-P)}{P(1-p)} = \frac{ad}{bc}$$

$$SD = \sqrt{\frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}}$$

where a is the number of probes within the selected probe set that contain one or more motif occurrences; b is the number of probes within the selected probe set that do not contain a motif occurrence; c and d are the same counts within the entire array probe set (drawn from the same set of distal-only probes using the same definition as the primary analysis). A probe set was considered significantly enriched for a particular motif if the 95% confidence interval of the Odds Ratio was greater than 1.1 (specified by option *lower.OR*, 1.1 is default), and the motif occurred at least 10 times (specified by option *min.incidence*, 10 is default) in the probe set.

Identification of master regulator TFs

When a group of enhancers is coordinately altered in a specific sample subset, this is often the result of an altered upstream *master regulator* transcription factor in the gene regulatory network. *ELMER* tries to identify such transcription factors corresponding to each of the TF binding motifs enriched from the previous analysis step. For each enriched motif, *ELMER* takes the average DNA methylation of all distal probes (in significant probe-gene pairs) that contain that motif occurrence (within a $\pm 250bp$ region), and compares this average DNA methylation to the expression of each gene annotated as a human TF.

A statistical test is performed for each motif-TF pair, as follows. All samples are divided into two groups: the M group, which consists of the 20% of samples with the highest average methylation at all motif-adjacent probes, and the U group, which consisted of the 20% of samples with the lowest methylation. This step is performed by the *get.TFs* function, which takes *minSubgroupFrac* as an input parameter, again with a default of 20%. For each candidate motif-TF pair, the Mann-Whitney U test is used to test the null hypothesis that overall gene expression in group M is greater or equal than that in group U. This non-parametric test was used in order to minimize the effects of expression outliers, which can occur across a very wide dynamic range. For each motif tested, this results in a raw p-value (P_r) for each of the human TFs. All TFs are ranked by their $-log_{10}(Pr)$ values, and those falling within the top 5% of this ranking were considered candidate upstream regulators. The best upstream TFs which are known to recognize to specific binding motif are automatically extracted as putative regulatory TFs, and rank ordered plots are created to visually inspect these relationships, as shown in the example below. Because the same motif can be recognized by many transcription factors of the same binding domain family, we define these relationships at both the family and subfamily classification level using the classifications from TFClass database [11]. Use of this database is a major change from version 1 of ELMER, which used custom curations for DNA binding domain families. Use of the TFClass database is preferable because it is well curated and regularly updated to reflect new findings.

Use Case 1: Breast Invasive Carcinoma (unsupervised approach)

Here, we describe how to perform *ELMER* analysis on TCGA BRCA (Breast Invasive Carcinoma) data retrieved from the GDC server. We assume that the user has an R environment with the packages *ELMER* (v2.0.0 or newer) and *ELMER*.data (v2.0.0 or newer) installed (see Installation documentation on the Bioconductor website). We first describe how the data can be downloaded and organized to the default *ELMER* input, followed by the following analysis steps:

- Identification of distal probes with significant differential DNA methylation (i.e. DMCs) in tumor vs. normal samples
- Identification of putative target gene(s) for differentially methylated distal probes

- Characterization of chromatin state context of significant probe regions using FunciVar
- Identification of enriched motifs within set of probes in significant probe-gene pairs
- Identification of master regulator Transcription Factors (TF) for each enriched motif

In addition to these standard steps, we also show how to compare the putative probe-gene pairs to those derived from deep-sequenced ChIA-PET data from MCF7 cells (as shown in [4]). This use case uses all data available with the recommended thresholds, and can take up to 10 hours to complete and requires a machine with more than 16GB of RAM. For a simpler example, please take a look at the included vignette (see supplemental files).

Downloading TCGA data

The function *getTCGA* uses the TCGAbiolinks package [12] to download TCGA data for all samples for a given disease (such as BLCA, LGG, GBM). Its main arguments are the *genome* that if set to "hg19" will download data from GDC legacy archive, and if set to "hg38" it will download data from the main GDC harmonized data portal.

Listing 2. "Step 1: Downloading TCGA data from GDC database"

```
library(ELMER)
getTCGA(disease = "BRCA", # TCGA disease abbreviation (BRCA,BLCA,GBM, LGG, etc)
        basedir = "DATA", # Where data will be downloaded
        genome  = "hg38") # Genome of refenrece "hg38" or "hg19"
```

If the *getTCGA* function called before was successful it will create the following objects and folders:

```
--- DATA/BRCA/
  |----------- BRCA_meth_hg38.rda (object with DNA methylation)
  |----------- BRCA_RNA_hg38.rda (object with gene expression)
  |----------- BRCA_clinic.rda   (object with indexed clinical information)
  |----------- Raw/ (folder: contains All raw data from GDC)
```

Selecting distal probes

The function *get.feature.probe*, shown in Listing 3, is used to select HM450K/EPIC probes located away from any TSS (at least $2Kb$ away). Its main arguments are the genome of reference ("hg38"/"hg19") and DNA methylation platform ("450K"/"EPIC"). The *feature* argument is used to limit the region of probes; as we want all distal probes, we set it to NULL.

Listing 3. "Step 2: Selection of probes within biofeatures"

```
# get distal probes that are 2kb away from TSS
distal.probes <- get.feature.probe(feature = NULL,
                                   genome = "hg38",
                                   met.platform = "450K")
# 168644 probes
```

Organizing data into a MultiAssayExperiment object

The function *createMAE* is used to organize the gene expression and DNA methylation data into a MultiAssayExperiment (MAE) object. Listing 4 shows how to use it with the data created in the previous steps. Its main arguments are described below:

- *exp* : An R object or a path to a file containing a gene expression matrix or SummarizedExperiment with with gene counts.
- *met* : An R object or a path to a file containing a DNA methylation matrix or SummarizedExperiment with beta values.
- *met.platform*: DNA methylation platform. "EPIC" for Infinium MethylationEPIC or "450K" for Infinium HumanMethylation450.
- *genome*: The genome of reference ("hg19" or "hg38") used to select the correct metadata. Genes genomic ranges will be annotated using ENSEMBL database and DNA methylation probes using metadata available at `http://zwdzwd.github.io/InfiniumAnnotation`.
- *linearize.exp*: this step will take the $log_2(gene\ expression + 1)$ in order to linearize the relationship between gene expression and DNA methylation.
- *filter.probes*: genomic ranges (i.e. distal regions) within which probes from DNA methylation data should be kept.
- *met.na.cut*: maximum percentage of empty values (NA) a probe might have to be considered in the analysis. Default is 20% (i.e if 50% of samples has empty values for a given probe, it will be removed).
- *colData*: A matrix with samples metadata (i.e. clinical data , molecular subtype information). If argument TCGA is set to *TRUE* this matrix will be created automatically. In this case *colData* argument is optional.
- *sampleMap*: A matrix mapping DNA methylation data and gene expression data to samples. *ELMER* uses only samples with both data. Otherwise it will be removed. If argument TCGA is set to *TRUE* this matrix will be created automatically. In this case *sampleMap* argument is optional.

Listing 4. "Step 3: Create MultiAssayExperiment"

```
mae <- createMAE(exp = "DATA/BRCA/BRCA_RNA_hg38.rda",
                 met = "DATA/BRCA/BRCA_meth_hg38.rda",
                 met.platform = "450K",
                 genome = "hg38",
                 linearize.exp = TRUE,
                 filter.probes = distal.probes,
                 met.na.cut = 0.2,
                 save = TRUE,
                 TCGA = TRUE)
```

Listing 5 shows information about the object created. There are 866 samples with both gene expression and DNA methylation data, and among those 5 are metastatic samples, 778 are Primary Solid Tumor and 83 are Solid Tissue Normal.

Listing 5. "Verifying MultiAssayExperiment"

```
> mae
A MultiAssayExperiment object of 2 listed
 experiments with user-defined names and respective classes.
 Containing an ExperimentList class object of length 2:
 [1] DNA methylation: RangedSummarizedExperiment with 135331 rows and 866 columns
 [2] Gene expression: RangedSummarizedExperiment with 57035 rows and 866 columns
Features:
 experiments() - obtain the ExperimentList instance
 colData() - the primary/phenotype DataFrame
 sampleMap() - the sample availability DataFrame
 `$`, `[`, `[[` - extract colData columns, subset, or experiment
 *Format() - convert ExperimentList into a long or wide DataFrame
 assays() - convert ExperimentList to a list of rectangular matrices
> table(mae$definition)
         Metastatic Primary solid Tumor Solid Tissue Normal
                  5                 778                  83
```

Identification of distal probes with significant differential DNA methylation (i.e. DMCs) in tumor vs. normal samples

The function *get.diff.meth* is used to identify regions differently methylation between two groups. Listing 6 shows how to use it to select hypomethylated probes in "Primary solid tumor" samples when compared to "solid tissue normal" samples ($FDR \leq 0.01$,$\Delta mean \geq 0.3$), using those samples in the lower quintile ($percentage = 0.2$) of DNA methylation levels for each probe. Its main arguments are described below:

- *data* A multiAssayExperiment with DNA methylation and Gene Expression data.
- *group.col* A column defining the groups of the sample. You can view the available columns using: colnames(MultiAssayExperiment::colData(data)).
- *group1* A group from group.col. *ELMER* will run group1 vs group2. That means, if direction is hyper, get probes hypermethylated in group 1 compared to group 2.
- *group2* A group from group.col. *ELMER* will run group1 vs group2. That means, if direction is hyper, get probes hypermethylated in group 1 compared to group 2.
- *diff.dir* Differential methylation direction. It can be "hypo" which is only selecting hypomethylated probes in group 1 when compared to group 2; "hyper" which is only selecting hypermethylated probes;
- *minSubgroupFrac* A number ranging from 0 to 1, specifying the fraction of extreme samples from group 1 and group 2 that are used to identify the differential DNA methylation. The default is 0.2 because we typically want to be able to detect a specific (possibly unknown) molecular subtype among tumor; these subtypes often make up only a minority of samples, and 20% was chosen as a lower bound for the purposes of statistical power. If you are using pre-defined group labels, such as treated replicates vs. untreated replicated, use a value of 1.0 (*Supervised* mode)
- *pvalue* A number specifying the significant P value (adjusted P value by Benjamini-Hochberg procedure) cutoff for selecting significant hypo/hyper-methylated probes. Default is 0.01.

- *sig.dif* A number specifying the smallest DNA methylation difference as a cutoff for selecting significant hypo/hyper-methylated probes. Default is 0.3.

Listing 6. "Identify significantly different DNA methylation probes in tumor and normal samples"

```
diff.probes <- get.diff.meth(data = mae,
                             group.col = "definition",
                             group1 = "Primary solid Tumor",
                             group2 = "Solid Tissue Normal",
                             diff.dir = "hypo", # Get probes hypometh. in group 1
                             cores = 1,
                             minSubgroupFrac = 0.2, # % group samples  used.
                             pvalue = 0.01,
                             sig.dif = 0.3,
                             dir.out = "Results_hypo/",
                             save = TRUE)
```

If the *save* argument is set to TRUE, in the *dir.out* folder two files will be created: *getMethdiff.hypo.probes.csv* containing all probes from the DNA methylation data with the difference means of the groups and the significance values, *getMethdiff.hypo.probes.significant.csv* will contain only probes that respect the thresholds. Table 2 shows the first rows of *getMethdiff.hypo.probes.significant.csv* file.

probe	pvalue	Primary.solid.Tumor˙Minus˙Solid.Tissue.Normal	adjust.p
cg00001809	1.97e-35	-0.32	1.26e-34
cg00008695	1.62e-67	-0.44	3.72e-66
cg00009553	6.84e-31	-0.525	3.61e-30

Table 2. First three rows of getMethdiff.hypo.probes.significant.csv file.

Also, users are able to verify the DNA methylation levels of a selected probe using the auxiliary function *metBoxPlot*, as showed in Listing 7. This function creates a boxplot for all samples and another one for the 20% of samples used in the analysis for each probe as showed in plots 2 and 6.

Listing 7. "Verify probes used in the comparison"

```
metBoxPlot(mae,
           group.col = "definition",
           group1 = "Primary solid Tumor",
           group2 = "Solid Tissue Normal",
           diff.dir = "hypo",
           probe = "cg14058239",
           minSubgroupFrac = 0.2)
```

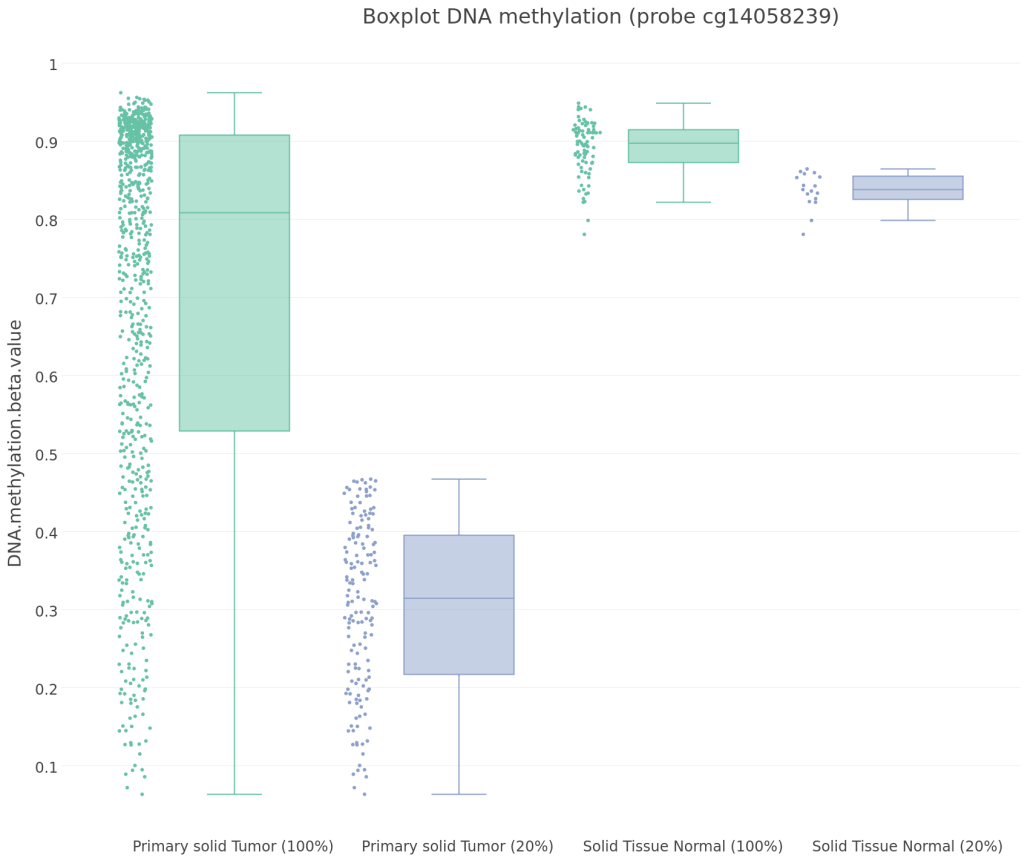

Figure 2. Probe cg00001809 DNA methylation boxplot. Due to the distribution of DNA methylation levels, if considered all samples in the comparison the probe would not be identified as differently methylated, but if considered the existence of different molecular subtypes and using only the 20% of samples, this probe will be identified as differently methylated.

Identification of putative target gene(s) for differentially methylated distal probes

The function *get.pair* is used to link enhancer probes with methylation changes to target genes with expression changes and report the putative target gene for selected probes. Listing 8 shows how to select the 20 nearest genes (10 downstream and 10 upstream) and evaluate if each pair is anti-correlated (probes with higher methylation levels have lower gene expression levels). Its main arguments are described below:

- *nearGenes* Output of *GetNearGenes* function.
- *minSubgroupFrac* A number ranging from 0 to 1, specifying the fraction of extreme samples that define group U (unmethylated) and group M (methylated), which are used to link probes to genes. The default is 0.4 (the lowest quintile of samples is the U group and the highest quintile samples is the M group) because we typically want to be able to detect a specific (possibly unknown) molecular subtype among tumor; these subtypes often make up only a minority of samples, and 20% was chosen as a lower bound for the purposes of statistical

power. If you are using pre-defined group labels, such as treated replicates vs. untreated replicated, use a value of 1.0 (*Supervised* mode).

- *permu.size* Number of permutation. Default is 10000. *Note*: This parameter can strongly impact run time.
- *pvalue* Raw p-value cutoff for defining significant pairs. Default is 0.001.
- *Pe* Empirical p-value cutoff for defining significant pairs. Default is 0.001.
- *filter.probes* Should probes be filtered by selecting only those which has at least a certain number of samples below and above a certain cut-off? If true, arguments *filter.probes* and *filter.percentage* will be used.
- *filter.portion* A number specify the cut point to define binary methylation level for probe loci. Default is 0.3. When beta value is above 0.3, the probe is methylated and vice versa. For one probe, the percentage of methylated and unmethylated samples should be above filter.percentage value. Only used if *filter.probes* is TRUE.
- *filter.percentage* Minimum percentage of samples to be considered in methylated and unmethylated for the filter.portion option. Default 5%. Only used if *filter.probes* is TRUE.

Listing 8. "Step 5: Identify putative target genes for differentially methylated distal probes"

```
# For each differently methylated probes we will get the
# 20 nearby genes (10 downstream and 10 upstream)
nearGenes <- GetNearGenes(data = mae,
                          probes =  diff.probes$probe,
                          numFlankingGenes = 20,
                          cores = 1)

# This step is the most time consuming. Depending on the size of the groups
# and the number of probes found previously it migh take hours
Hypo.pair <- get.pair(data = mae,
                      nearGenes = nearGenes,
                      group.col = "definition",
                      group1 = "Primary solid Tumor",
                      group2 = "Solid Tissue Normal",
                      permu.dir = "Results_hypo/permu",
                      permu.size = 10000,
                      minSubgroupFrac = 0.4,# 40% of samples to create U and M
                      pvalue = 0.001,
                      Pe = 0.001,
                      filter.probes = TRUE,
                      filter.percentage = 0.05,
                      filter.portion = 0.3,
                      dir.out = "Results_hypo",
                      cores = 1,
                      label = "hypo")
# Number of pairs: 2950
```

The output of this function is shown in table 3. Probe and GeneID columns shows the significant pair and P_e shows the adjusted p-value.

To visualize the relationship between the probe-gene pairs inferred, there are two auxiliary functions in ELMER. The function *schematic.plot*, showed in Listing 9, which will plot genes and

Probe	GeneID	Symbol	Distance	Sides	Raw.p	Pe
cg05120309	ENSG00000196405	EVL	0	L3	3.04e-56	9.999000099990002e-5
cg25343204	ENSG00000106541	AGR2	0	R1	3.99e-56	9.999000099990002e-5
cg14058239	ENSG00000141424	SLC39A6	0	R1	5.17e-56	9.999000099990002e-5

Table 3. First three rows of getPair.hypo.pairs.significant.csv file.

probes in a specified genomic region, highlighting the significant pairs identified by plotting a genomic interactions track and highlight the genes in the pair in red (Figure 3). Also, using the function *scatter.plot* (Listing 10) it is possible to visualize the correlation between gene expression and DNA methylation levels at probe (Figure 4).

Listing 9. "Schematic plot to visualize gene-probe pairs"

```
# by probe and with detail about DNA methylation
schematic.plot(data = mae,
               group.col = "definition",
               group1 = "Primary solid Tumor",
               group2 = "Solid Tissue Normal",
               pair = Hypo.pair,
               statehub.tracks = "hg38/ENCODE/mcf-7.16mark.segmentation.bed",
               byProbe = "cg04723436")
```

Listing 10. "Scatter plot to visualize correlation between gene expression and DNA methylation levels at probe"

```
scatter.plot(data = mae,
             byPair = list(probe = "cg04723436",
                           gene = "ENSG00000107485"),
             save = T,
             category = "definition",
             lm = TRUE)
```

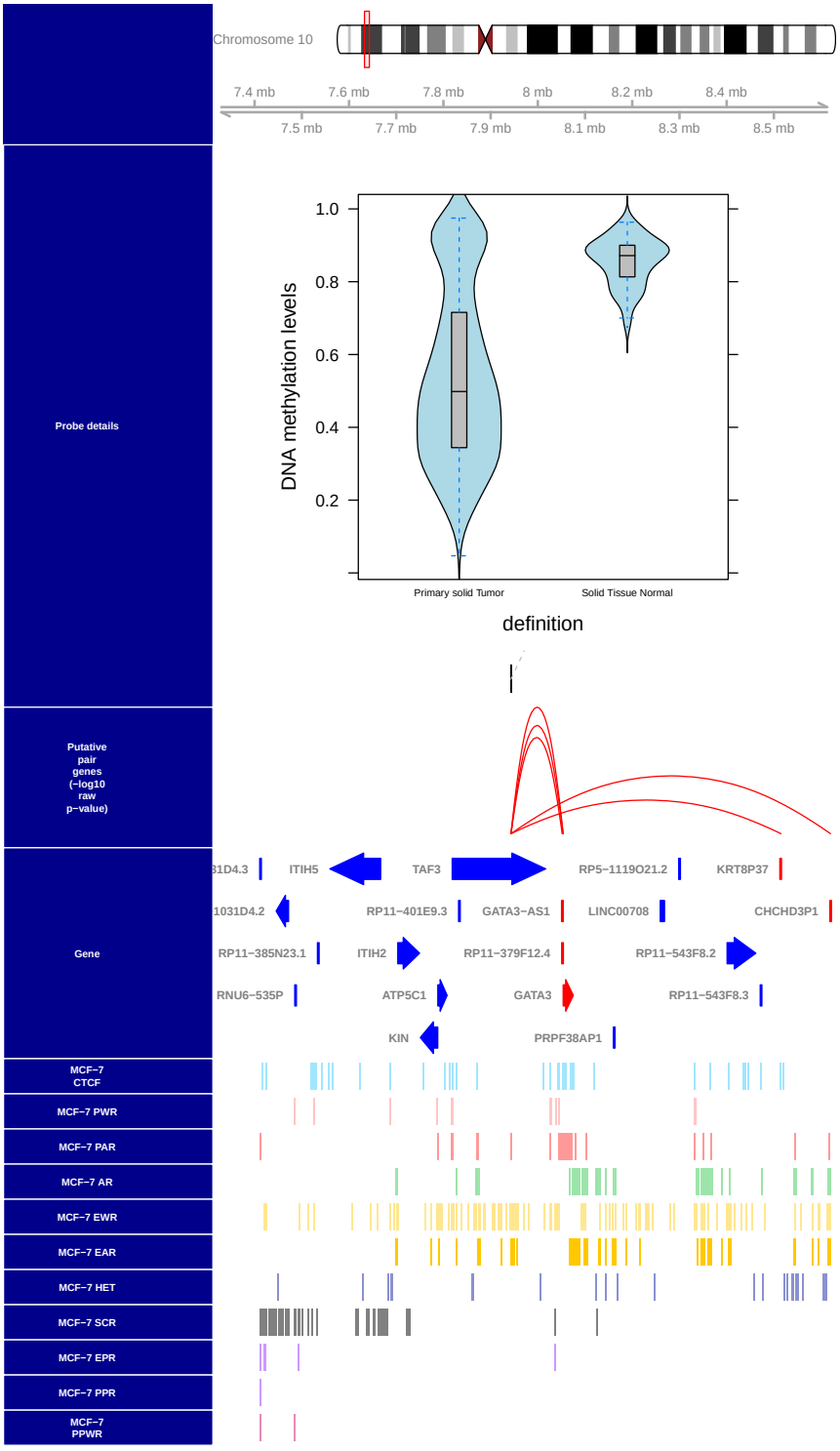

Figure 3. Plot probe gene pairs with annotation track for MCF-7 cell line from `StateHub.org`. Significant probes and gene pairs are highlighted in red.

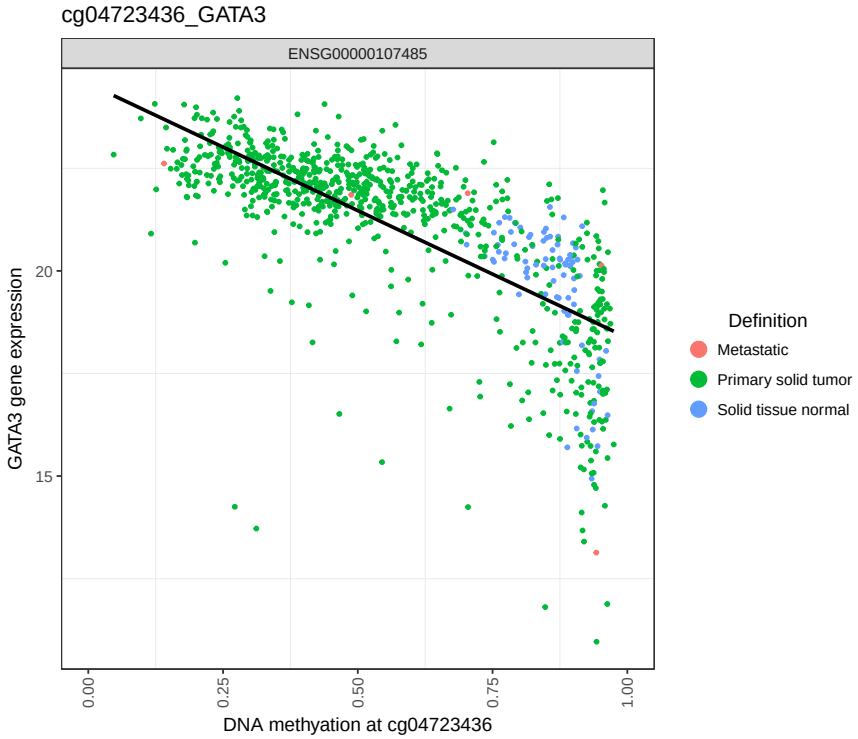

Figure 4. Scatter plot for significant probe (cg04723436) gene (GATA3) pair.

Characterization of chromatin state context of significant probe regions using FunciVar

To understand and compare our set of probes identified in the probe-gene pairs inferred we used chromatin state of IHEC cell types from `http://statehub.org/`, to calculate the relative enrichment of different states (see Additional file for the code). This procedure uses code from the statepaintR [22] and FunciVar [23] packages. Figure 5 shows the enrichment for 14 encode cells lines. The plot shows enrichment for active region (AR),enhancer active region (EAR), weak enhancer (EWR) and active promoter region (PAR) in MCF-7 cell (human breast adenocarcinoma cell line) while for other cell lines this enrichment is not visible.

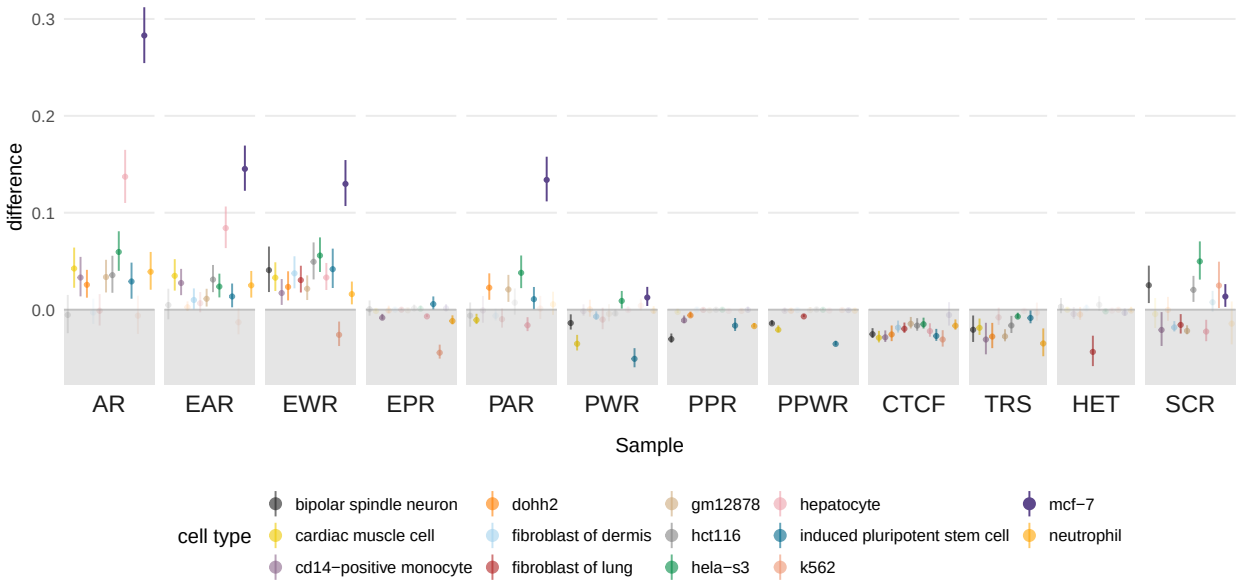

Figure 5. Enrichment of paired probes and chromatin states of encode cells. The plot shows enrichment for active region (AR),enhancer active region (EAR), weak enhancer (EWR) and active promoter region (PAR) for MCF-7 cell. Acronyms - AR: Active region, EAR: active enhancer, EWR: Weak Enhancer, EPR: poised enhancer, PAR: active promoter, PWR: Weak Promoter, PPR: poised promoter, PPWR: Weak Poised Promoter, CTCF: architectural complex, TRS: transcribed, HET: heterochromatin, SCR: Polycomb Repressed Silenced

Identification of enriched motifs within set of probes in significant probe-gene pairs

The function *get.enriched.motif* is used to identify enriched motif in a set of probes. The main arguments are described below:

- *lower.OR* The motif with lower boundary of 95% confidence interval for Odds Ratio $\geq$ *lower.OR* are the significantly enriched motifs.
- *min.incidence* Minimum number of probes having the motif signature (default: 10) required for a motif to be enriched.

Listing 11. "Step 6: Motif enrichment analysis on the selected probes"

```
enriched.motif <- get.enriched.motif(data = mae,
                                     min.motif.quality = "DS",
                                     probes = unique(Hypo.pair$Probe),
                                     dir.out = "Results_hypo",
                                     label = "hypo",
                                     min.incidence = 10,
                                     lower.OR = 1.1)
```

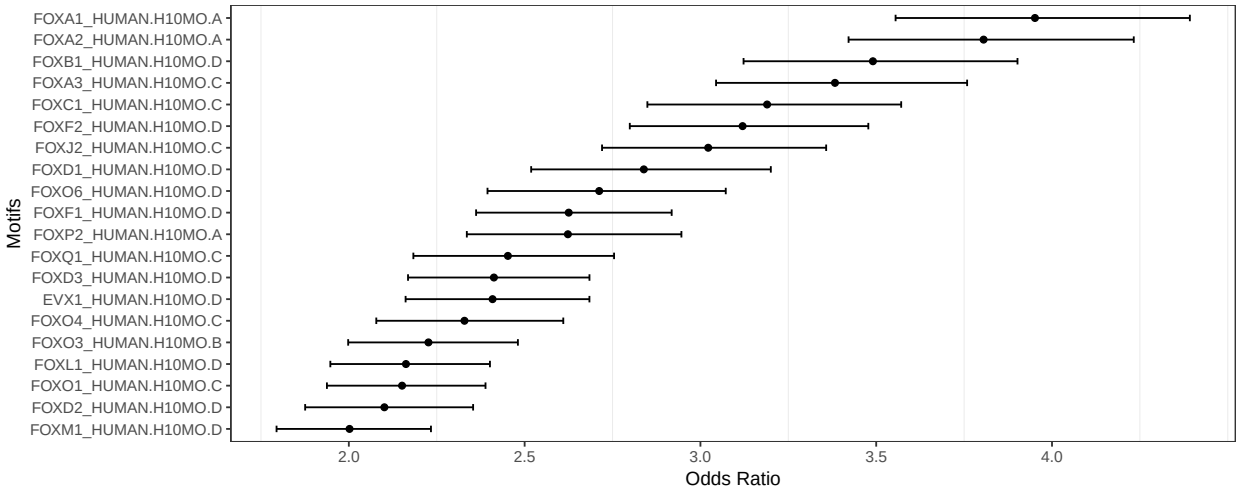

Figure 6. Motif enrichment plot shows the enrichment levels ($OR \geq 2.0$) for the selected motifs. This plot represents a subset of the enriched motifs for $lower.or = 1.1$, only selected for representational purposes.

Identification of master regulator Transcription Factors (TF) for each enriched motif

The function *get.TFs* is used to identify regulatory TF whose expression associates with TF binding motif DNA methylation which.

Listing 12. "Step 7: Identify regulatory Transcript Factors"

```
TF <- get.TFs(data = mae,
```

```
                group.col = "definition",
                group1 = "Primary solid Tumor",
                group2 = "Solid Tissue Normal",
                minSubgroupFrac = 0.4, # Set to 1 if supervised mode
                enriched.motif = enriched.motif,
                dir.out = "Results_hypo",
                cores = 1,
                label = "hypo")

paste(sort(unique(TF$top.potential.TF.family)),collapse = ",")
# "EMX1,ESR1,FOXA1,GATA3,HOMEZ,LMX1B,MYB,MZF1,NR2F6,OVOL1,PBX1,RARA,SPDEF,ZKSCAN1
    ,ZSCAN16"
paste(sort(unique(TF$top.potential.TF.subfamily)),collapse = ",")
# "AR,EMX1,FOXA1,FOXD2,GATA3,HOMEZ,LMX1B,MYB,NR2E3,PBX1,ZKSCAN1,ZSCAN16"
```

The result of this function is shown in table 4 and in figure 7.

motif	top.potential.TF.family	top.potential.TF.subfamily	potential.TF.family	potential.TF.subfamily	top˙5percent˙TFs
AIRE.C	NA	NA	NA	NA	FOXA1;GATA3;ESR1;SPD...
ALX1.B	NA	NA	NA	NA	FOXA1;GATA3;ESR1;SPD...
ALX3.D	NA	NA	NA	NA	FOXA1;GATA3;ESR1;SPD...
ALX4.D	NA	NA	NA	NA	FOXA1;GATA3;ESR1;RAR...
ANDR.A	ESR1	AR	ESR1;AR	AR	FOXA1;GATA3;ESR1;SPD...
ARI3A.D	NA	NA	NA	NA	FOXA1;GATA3;ESR1;RAR...
ARI3A.S	NA	NA	NA	NA	FOXA1;GATA3;ESR1;RAR...
ARI5B.C	NA	NA	NA	NA	FOXA1;GATA3;ESR1;SPD...
ARX.D	NA	NA	NA	NA	FOXA1;GATA3;ESR1;RAR...
BARH1.D	EMX1	NA	EMX1;LBX2	NA	FOXA1;GATA3;ESR1;SPD...
BARH2.D	EMX1	NA	EMX1;LBX2	NA	FOXA1;GATA3;ESR1;CXX...
BARX1.D	EMX1	NA	EMX1;LBX2	NA	FOXA1;GATA3;ESR1;RAR...
BARX2.D	EMX1	NA	EMX1;LBX2	NA	FOXA1;GATA3;ESR1;CXX...
BATF3.D	NA	NA	NA	NA	FOXA1;GATA3;ESR1;SPD...
BATF.A	NA	NA	NA	NA	FOXA1;GATA3;ESR1;CXX...
BATF.S	NA	NA	NA	NA	FOXA1;GATA3;ESR1;CXX...
BCL6.C	ZSCAN16	NA	ZSCAN16;OVOL1	NA	FOXA1;GATA3;ESR1;RAR...
BHE22.D	NA	NA	NA	NA	FOXA1;GATA3;ESR1;SPD...

Table 4. First twenty rows of the *getTF.hypo.significant.TFs.with.motif.summary.csv* file created by *get.Tfs* function (suffix "_HUMAN.H10MO" was removed from motifs names). First column shows the enriched motif, "top_5percent_TFs" shows the top 5% TFs ranked (the same as all TFs to the left of the dashed line in figure 7), "potential.TFs.family" are the TF from the "top_5percent" that belongs to the same family as the TF of the motif, "top.potential.TFs.family" is the highest ranked TF belonging to the same family as the TF of the motif (same as the first TF from "potential.TFs.family" column). The columns "potential.TFs.subfamily" and "top.potential.TFs.subfamily" are the same as "potential.TFs.family" and "top.potential.TFs.family" but considering the subfamily classification instead. For example, the motif ANDR has two TFs in the top 5% that belongs to the same TF family (Steroid hormone receptors): ESR1 and AR, but if considering subfamilies only AR in considered.

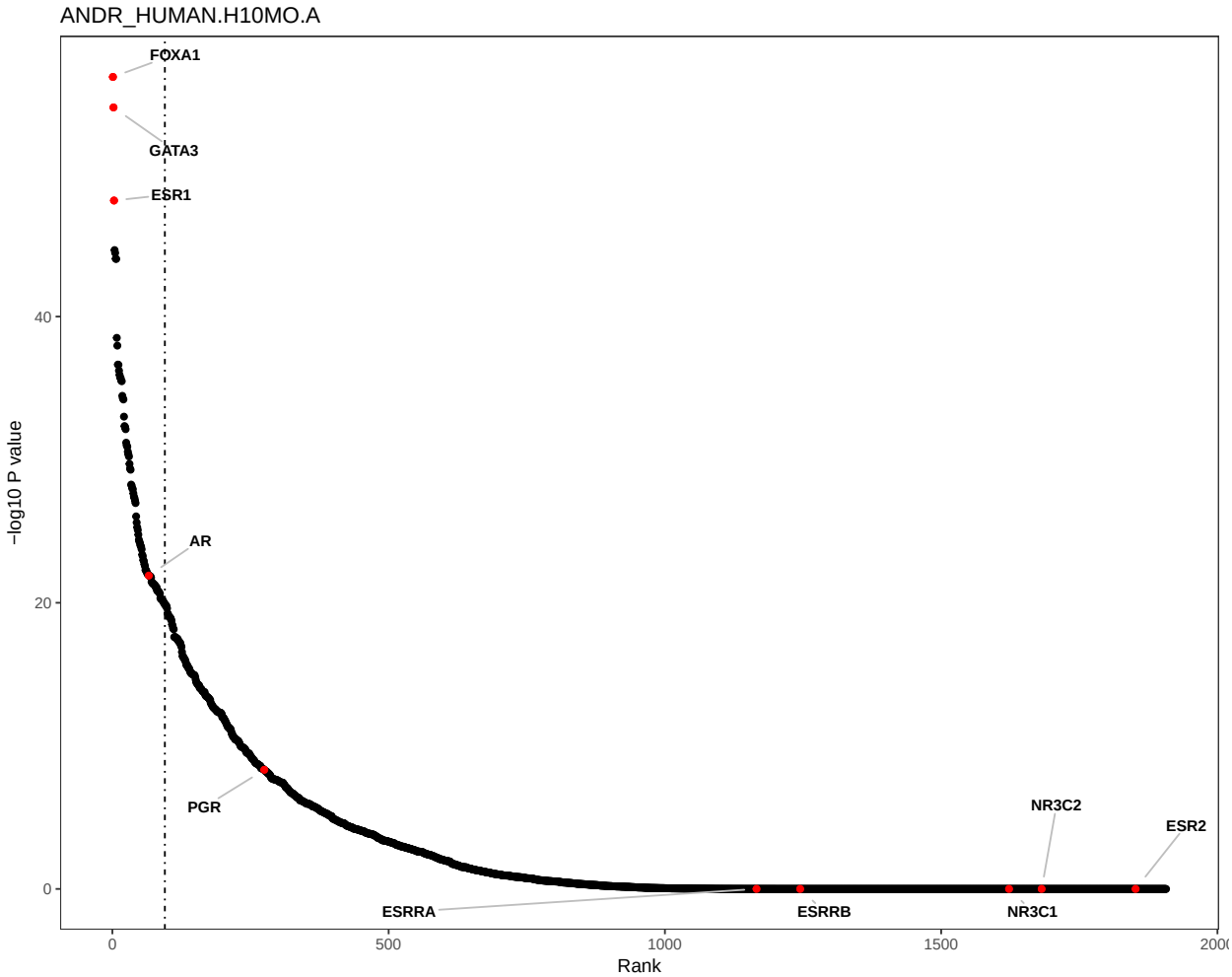

Figure 7. TF ranking plot shows statistic $-log_{10}(P-value)$ assessing the anti-correlation level of TFs expression level with average DNA methylation level at sites with a given motif. By default, the top 3 associated TFs and the TF family members (dots in red) that are associated with that specific motif are labeled in the plot. But there is also a option show highlight only TF sub-family members (TCClass database classification) which is showed in the vignette.

Comparing inferred results with MCF-7 chIA-PET

As shown in Yao et al. [4], we compared the putative pairs inferred to the chromatin loops derived from deep-sequenced ChIA-PET data from MCF7 cells [25]. First, we identify the number of *ELMER* pairs overlapping the ChIA-PET loops, then we repeat using randomly generated pairs with properties similar to the *ELMER* pairs. For each true ELMER probe in a probe-gene pair, we randomly select a different probe from the complete set of distal probes. We then choose the nth nearest gene to the random probe, where n is the same as the adjacency of the true ELMER probe (i.e. if the true probe is linked to the second gene upstream, the random probe will also be linked to its second gene upstream). Thus, the random linkage set has both the same number of probes and the same number of linked genes as the true set. One hundred such random datasets were generated to arrive at a 95% CI (+/-1.96* SD). The result is shown in Figure 8. Of the 2108 putative pairs identified in breast cancer tumors 321 (approximately 15.2%) were also identified as loops in the MCF7 ChIA-PET data. This was a three-fold enrichment over randomized probe-gene

pairs (see Additional file for the code).

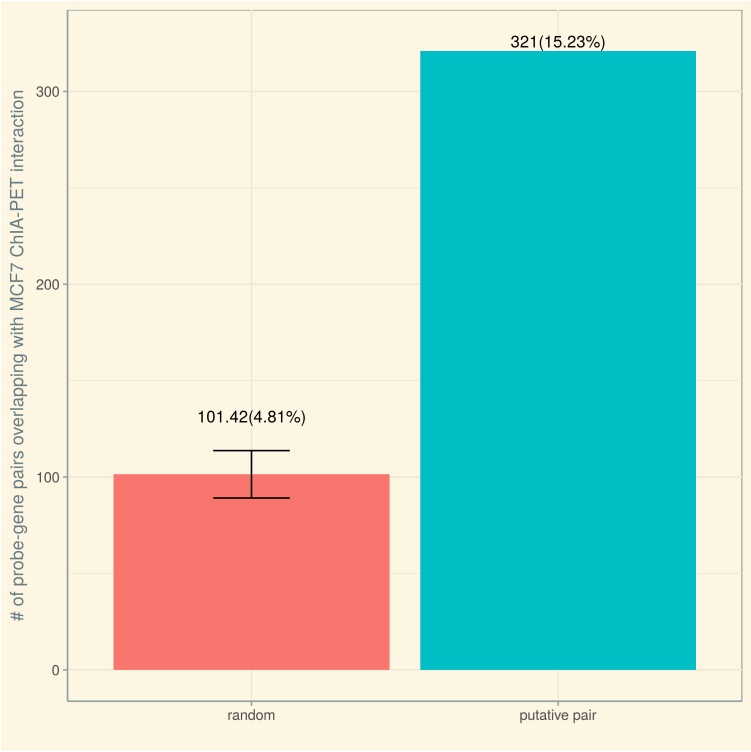

Figure 8. The graph shows the comparison of the number of probe-gene pairs identified within MCF7 ChIA-PET data using the putative pairs from BRCA vs. random pairs

Use Case 2: G-CIMP-low and G-CIMP-high (supervised approach)

We also performed *ELMER* analysis on GBM (glioblastoma multiforme) and LGG (Low grade glioma) data retrieved from TCGA comparing two know groups (IDH mutant and wild-type) [26]. The main arguments changed were the percentage of samples used to identify the differently methylated probes in function *get.diff.meth* which was set to 100% (use all samples from each group) and the percentage of samples used to identify the putative gene-probe pairs in function *get.pair* and the regulatory TFs in function *get.TFs* which were also set to 100% (Results and code can be found in the supplemental files.

Discussion

We present a new version of *ELMER*, an R/Bioconductor package that allows users to infer altered gene regulatory networks and master regulators, by linking expression changes to DNA methylation changes of nearby cis-regulatory elements. This version is greatly improved in terms of stability, performance, and extensibility. It also adds a number of new features, including support for support Human genome build 38, to the Infinium MethylationEPIC BeadChip array, import of datasets from the NCI Genomic Data Commons (GDC), and a new Supervised mode for analysis of paired sample study designs. It also performs motif and master regulator analysis using an expanded set of motifs and a well-maintained database for TF binding domain families and binding site preferences. It also provides greater support for Bioconductor standards (such as MultiAssayExperiment), and improved user interaction by improving messages and error handling, as well as newly designed, publication quality output plots. Our case study performed on a TCGA Breast Invasive Carcinoma (BTCGA-BRCA) dataset showed that GATA3, ESR1, FOXA1 were identified, consistent with research presented by Theodorou et al. [27] and our earlier work [4].

Data availability

The TCGA data was downloaded from the NCI Genomic Data Commons (GDC) data portal [7] using TCGAbiolinks R/Bioconductor package [12, 28]. Gene annotations were retrieved from ENSEMBLE [15] database via biomaRt R/Bioconductor package[18, 21]. DNA methylation microarrays metadata were retrieved from `http://zwdzwd.github.io/InfiniumAnnotation`[19]. Transcription factor (TF) binding models can be downloaded at HOCOMOCO database (`http://hocomoco.autosome.ru/`) [10]. The list of human TF can be accessed at `http://www.uniprot.org/` [9]. The classification of human transcription factors (TFs) can be viewed at `http://tfclass.bioinf.med.uni-goettingen.de/tfclass` [11].

Software availability

This section will be generated by the Editorial Office before publication. Authors are asked to provide some initial information to assist the Editorial Office, as detailed below.

1. URL link to where the software can be downloaded from or used by a non-coder (AUTHOR TO PROVIDE; optional)
2. The source code of *ELMER* is available at `https://github.com/tiagochst/ELMER` and the auxiliary data files are available `https://github.com/tiagochst/ELMER.data`.
3. Link to source code as at time of publication (*F1000Research* TO GENERATE)
4. Link to archived source code as at time of publication (*F1000Research* TO GENERATE)
5. *ELMER* is available under the GNU General Public License version 3 (GNU GPL3)

Author contributions

BPB and HN conceived the study. TCS developed the new version of *ELMER* and *ELMER*.data package. TCS, SC and DH prepared the first draft of the manuscript. All authors were involved in

the revision of the draft manuscript and have agreed to the final content. LY conceived and developed the original version of the *ELMER* package, and graciously advised on the improvements in this new version. BPB, TCS, SC are authors of the *ELMER* Bioconductor package.

Competing interests

No competing interests were disclosed

Grant information

This work has been supported by a grant from Henry Ford Hospital (H.N.) and by the São Paulo Research Foundation (FAPESP) (2016/01389-7 to T.C.S. & H.N. and 2015/07925-5 to H.N.), T.C.S. and B.P.B. were supported by the NCI Informatics Technology for Cancer Research program, NIH/NCI grant 1U01CA184826, and Genomic Data Analysis Network NIH/NCI grant 1U24CA210969.

Acknowledgments

Lijing Yao for her assistance assistance with explanation of the codes of the initial version of *ELMER* tool available at `https://github.com/lijingya/ELMER` and `https://github.com/lijingya/ELMER.data`.

References

1. Benjamin P Berman, Daniel J Weisenberger, Joseph F Aman, Toshinori Hinoue, Zachary Ramjan, Yaping Liu, Houtan Noushmehr, Christopher PE Lange, Cornelis M van Dijk, Rob AEM Tollenaar, et al. Regions of focal dna hypermethylation and long-range hypomethylation in colorectal cancer coincide with nuclear lamina-associated domains. *Nature genetics*, 44(1):40–46, 2012.

2. Dvir Aran and Asaf Hellman. Dna methylation of transcriptional enhancers and cancer predisposition. *Cell*, 154(1):11–13, 2013.

3. Lijing Yao, Benjamin P Berman, and Peggy J Farnham. Demystifying the secret mission of enhancers: linking distal regulatory elements to target genes. *Critical reviews in biochemistry and molecular biology*, 50(6):550–573, 2015.

4. Lijing Yao, Hui Shen, Peter W Laird, Peggy J Farnham, and Benjamin P Berman. Inferring regulatory element landscapes and transcription factor networks from cancer methylomes. *Genome biology*, 16(1):105, 2015.

5. Tiago C Silva, Antonio Colaprico, Catharina Olsen, Fulvio D'Angelo, Gianluca Bontempi, Michele Ceccarelli, and Houtan Noushmehr. Tcga workflow: Analyze cancer genomics and epigenomics data using bioconductor packages. *F1000Research*, 5, 2016.

6. Katarzyna Tomczak, Patrycja Czerwinska, Maciej Wiznerowicz, et al. The cancer genome atlas (tcga): an immeasurable source of knowledge. *Contemp Oncol (Pozn)*, 19(1A):A68–A77, 2015.

7. Robert L Grossman, Allison P Heath, Vincent Ferretti, Harold E Varmus, Douglas R Lowy, Warren A Kibbe, and Louis M Staudt. Toward a shared vision for cancer genomic data. *New England Journal of Medicine*, 375(12):1109–1112, 2016.

8. Infinium methylationepic kit. `https://www.illumina.com/products/by-type/microarray-kits/infinium-methylation-epic.html`. Accessed: 2017-05-30.

9. Rolf Apweiler, Amos Bairoch, Cathy H Wu, Winona C Barker, Brigitte Boeckmann, Serenella Ferro, Elisabeth Gasteiger, Hongzhan Huang, Rodrigo Lopez, Michele Magrane, et al. Uniprot: the universal protein knowledgebase. *Nucleic acids research*, 32(suppl 1):D115–D119, 2004.

10. Ivan V Kulakovskiy, Ilya E Vorontsov, Ivan S Yevshin, Anastasiia V Soboleva, Artem S Kasianov, Haitham Ashoor, Wail Ba-Alawi, Vladimir B Bajic, Yulia A Medvedeva, Fedor A Kolpakov, et al. Hocomoco: expansion and enhancement of the collection of transcription factor binding sites models. *Nucleic acids research*, 44(D1):D116–D125, 2016.

11. Edgar Wingender, Torsten Schoeps, and Jürgen Dönitz. Tfclass: an expandable hierarchical classification of human transcription factors. *Nucleic acids research*, 41(D1):D165–D170, 2013.

12. Antonio Colaprico, Tiago C Silva, Catharina Olsen, Luciano Garofano, Claudia Cava, Davide Garolini, Thais S Sabedot, Tathiane M Malta, Stefano M Pagnotta, Isabella Castiglioni, et al. Tcgabiolinks: an r/bioconductor package for integrative analysis of tcga data. *Nucleic acids research*, page gkv1507, 2015.

13. MultiAssay SIG. *MultiAssayExperiment: Software for the integration of multi-omics experiments in Bioconductor*, 2017. URL `https://github.com/waldronlab/MultiAssayExperiment/wiki/MultiAssayExperiment-API`. R package version 1.2.0.

14. Hadley Wickham and Winston Chang. *devtools: Tools to Make Developing R Packages Easier*, 2016. URL `https://CRAN.R-project.org/package=devtools`. R package version 1.12.0.

15. Andrew Yates, Wasiu Akanni, M Ridwan Amode, Daniel Barrell, Konstantinos Billis, Denise Carvalho-Silva, Carla Cummins, Peter Clapham, Stephen Fitzgerald, Laurent Gil, et al. Ensembl 2016. *Nucleic acids research*, page gkv1157, 2015.

16. Wolfgang Huber, Vincent J Carey, Robert Gentleman, Simon Anders, Marc Carlson, Benilton S Carvalho, Hector Corrada Bravo, Sean Davis, Laurent Gatto, Thomas Girke, et al. Orchestrating high-throughput genomic analysis with bioconductor. *Nature methods*, 12(2):115–121, 2015.

17. Wanding Zhou, Peter W Laird, and Hui Shen. Comprehensive characterization, annotation and innovative use of infinium dna methylation beadchip probes. *Nucleic Acids Research*, page gkw967, 2016.

18. Steffen Durinck, Paul T Spellman, Ewan Birney, and Wolfgang Huber. Mapping identifiers for the integration of genomic datasets with the r/bioconductor package biomart. *Nature protocols*, 4(8): 1184–1191, 2009.

19. Wanding Zhou, Peter W. Laird, and Hui Shen. Comprehensive characterization, annotation and innovative use of infinium dna methylation beadchip probes. *Nucleic Acids Research*, 45(4):e22, 2017. doi: 10.1093/nar/gkw967. URL `+http://dx.doi.org/10.1093/nar/gkw967`.

20. 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.

21. Steffen Durinck, Yves Moreau, Arek Kasprzyk, Sean Davis, Bart De Moor, Alvis Brazma, and Wolfgang Huber. Biomart and bioconductor: a powerful link between biological databases and microarray data analysis. *Bioinformatics*, 21(16):3439–3440, 2005.

22. Simon G Coetzee, Zachary Ramjan, Huy Q Dinh, Benjamin P Berman, and Dennis J Hazelett. Statehub-statepaintr: rapid and reproducible chromatin state evaluation for custom genome annotation. *bioRxiv*, page 127720, 2017.

23. Funcirvar. `https://github.com/Simon-Coetzee/funcivar`. Accessed: 2017-06-09.

24. Sven Heinz, Christopher Benner, Nathanael Spann, Eric Bertolino, Yin C Lin, Peter Laslo, Jason X Cheng, Cornelis Murre, Harinder Singh, and Christopher K Glass. Simple combinations of lineage-determining transcription factors prime cis-regulatory elements required for macrophage and b cell identities. *Molecular cell*, 38(4):576–589, 2010.

25. Guoliang Li, Xiaoan Ruan, Raymond K Auerbach, Kuljeet Singh Sandhu, Meizhen Zheng, Ping Wang, Huay Mei Poh, Yufen Goh, Joanne Lim, Jingyao Zhang, et al. Extensive promoter-centered chromatin interactions provide a topological basis for transcription regulation. *Cell*, 148(1):84–98, 2012.

26. Michele Ceccarelli, Floris P Barthel, Tathiane M Malta, Thais S Sabedot, Sofie R Salama, Bradley A Murray, Olena Morozova, Yulia Newton, Amie Radenbaugh, Stefano M Pagnotta, et al. Molecular profiling reveals biologically discrete subsets and pathways of progression in diffuse glioma. *Cell*, 164 (3):550–563, 2016.

27. Vasiliki Theodorou, Rory Stark, Suraj Menon, and Jason S Carroll. Gata3 acts upstream of foxa1 in mediating esr1 binding by shaping enhancer accessibility. *Genome research*, 23(1):12–22, 2013.

28. TC Silva, A Colaprico, C Olsen, F D'Angelo, G Bontempi, M Ceccarelli, and H Noushmehr. Tcga workflow: Analyze cancer genomics and epigenomics data using bioconductor packages [version 2; referees: 1 approved, 2 approved with reservations]. *F1000Research*, 5(1542), 2016. doi: 10.12688/f1000research.8923.2.