The ENmix User's Guide

Zongli Xu, Liang Niu, Leping Li and Jack A. Taylor

Modified: February 9, 2016. Compiled: February 9, 2016

1 Introduction

Illumina HumanMethylation450 BeadChip array measurements have intrinsic levels of background noise that degrade methylation measurement. The ENmix package provides an efficient data preprocessing tool designed to reduce background noise and improve signal for DNA methylation estimation. The package utilizes a novel model-based background correction method, *ENmix*, that significantly improve accuracy and reproducibility of methylation measures. The data structure used by the ENmix package is compatible with several other related R packages, such as minfi, wateRmelon and ChAMP, providing straightforward integration of ENmix-corrected datasets for subsequent data analysis. The software is designed to support large scale data analysis, and provides multi-processor parallel computing wrappers for commonly used data preprocessing methods, including BMIQ probe design type bias correction and ComBat batch effect correction. In addition the ENmix package has selectable complementary functions for efficient data visualization (such as data distribution plotting), quality control (identification and filtering of low quality data points, samples, probes, and outliers, along with imputation of missing values), inter-array normalization (3 different quantile normalizations), identification of probes with multimodal distributions due to SNPs and other factors, and exploration of data variance structure using principal component regression analysis plots. Together these provide a set of flexible and transparent tools for preprocessing of EWAS data in a computationally-efficient and user-friendly package.

2 Citation

If you are using ENmix package, please cite this publication:

Zongli Xu, Liang Niu, Leping Li and Jack A. Taylor ENmix: a novel background correction method for Illumina HumanMethylation450 BeadChip, Nucleic Acids Research, 2015

3 Compatibility with other related R packages

The ENmix uses the data structure provided by R minfi packages as input and output, and thus is fully compatible with the minfi package. The same data structures were also used by several other R packages, such as ChAMP and wateRmelon, so the output from ENmix functions can be easily utilized in these packages for further analysis. Here are some examples:

Example 1: mixed use of minfi and ENmix functions

```
> library(ENmix)
> #minfi functions to read in data
> sheet <- read.450k.sheet(file.path(find.package("minfiData"),
+  "extdata"), pattern = "csv$")
> rgSet <- read.450k.exp(targets = sheet, extended = TRUE)
> #ENmix function for control plot
> plotCtrl(rgSet)
> #minfi functions to extract methylation and annotation data
> mraw <- preprocessRaw(rgSet)
> beta<-getBeta(mraw, "Illumina")
> anno=getAnnotation(rgSet)
> #ENmix function for fast and accurate distribution plot
> multifreqpoly(beta,main="Data distribution")
> multifreqpoly(beta[anno$Type=="I",],main="Data distribution, type I")
> multifreqpoly(beta[anno$Type=="II",],main="Data distribution, type II")
> #ENmix background correction
> mset<-preprocessENmix(rgSet, bgParaEst="oob", dyeCorr=TRUE, nCores=6)
> #minfi functions for further preprocessing and analysis
> gmSet <- preprocessQuantile(mset)
> bumps <- bumphunter(gmSet, design = model.matrix(~ gmSet$status), B = 0,
+ type = "Beta", cutoff = 0.25)
```

Example 2: add ENmix background correction step into ChAMP pipeline

```
> library(ENmix)
> library(ChAMP)
> testDir=system.file("extdata",package="ChAMPdata")
> myLoad=champ.load(directory=testDir)
> #ENmix background correction
> mset<-preprocessENmix(myLoad$rgSet,bgParaEst="oob", nCores=6)
> #remove probes filtered by champ.load()
> mset=mset[rownames(myLoad$beta),]
> #update myLoad object with background corrected intensity data
> myLoad$mset=mset
> myLoad$beta=getBeta(mset)
```

```
> myLoad$intensity=getMeth(mset)+getUnmeth(mset)
> #continue ChAMP pipeline
> myNorm=champ.norm()
```

4 Setting up the data

The first step is to import array raw data files (*.idat) using functions provided in R package minfi to create an object of `RGChannelSetExtended`.

```
> library(ENmix)
> require(minfi)
> #see minfi user's guide for the format of sample_sheet.txt file
> targets <- read.table("./sample_sheet.txt", header=T)
> rgSet <- read.450k.exp( targets = targets, extended = TRUE)
> # or read in all idat files under a directory
> rgSet <- read.450k.exp(base = "path_to_directory_idat_files",
+ targets = NULL, extended = TRUE, recursive=TRUE)
```

When methylation IDAT raw data files are not available, such as in many publically available datasets, users can use methylated (M) and unmethylated (U) intensity data to create an object of `MethylSet`.

```
> M<-matrix_for_methylated_intensity
> U<-matrix_for_unmethylated_intensity
> pheno<-as.data.frame(cbind(colnames(M), colnames(M)))
> names(pheno)<-c("Basename","filenames")
> rownames(pheno)<-pheno$Basename
> pheno<-AnnotatedDataFrame(data=pheno)
> anno<-c("IlluminaHumanMethylation450k", "ilmn12.hg19")
> names(anno)<-c("array", "annotation")
> mdat<-MethylSet(Meth = M, Unmeth = U, annotation=anno,
+ phenoData=pheno)
```

As an example for testing, users can use IDAT files provided in R data package minfiData to create an object of `RGChannelSetExtended`.

```
> library(ENmix)
> require(minfi)
> require(minfiData)
> sheet <- read.450k.sheet(file.path(find.package("minfiData"),
+ "extdata"), pattern = "csv$")
> rgSet <- read.450k.exp(targets = sheet, extended = TRUE)
```

5 Quality Control

5.1 Internal control probes

Illumina 450k chip incorporated 15 different types of internal control probes (total of 848 probes). The control plots generated using the data by Illumina GenomeStudio software are very useful to inspect experimental process and data quality. However, GenomeStudio only works on the windows operating system, it is time consuming to generate these plots for larger dataset, and there is also no option to save the plots into file. The function `plotCtrl` can generate similar plots for each type of control.

```
> plotCtrl(rgSet)
```

See Illumina Infinium HD Methylation Assay for detailed description on how to interpret these control figures. Here is a list of control types:

Control types	Number of probes
Sample-Independent Controls	
STAINING	4
EXTENSION	4
HYBRIDIZATION	3
TARGET REMOVAL	2
RESTORATION	1
Sample-Dependent Controls	
BISULFITE CONVERSION I	12
BISULFITE CONVERSION II	4
SPECIFICITY I	12
SPECIFICITY II	3
NON-POLYMORPHIC	4
NORM_A	32
NORM_C	61
NORM_G	32
NORM_T	61
NEGATIVE	613

These controls can also be plotted in user specified order to check how experimental factors affect methylation measures, such as batch, plate, array or array location.

```
> pinfo=pData(rgSet)
> IDorder=rownames(pinfo)[order(pinfo$Slide,pinfo$Array)]
> plotCtrl(rgSet,IDorder)
```

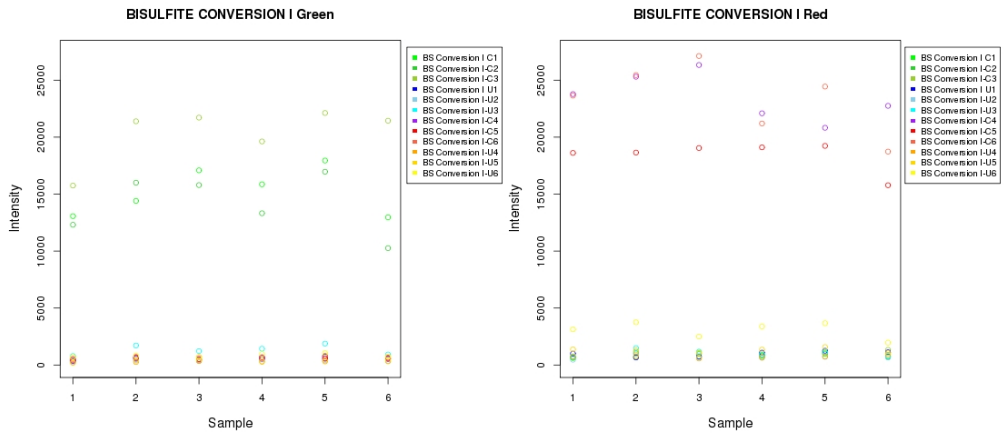

Figure 1: Bisulfite conversion controls for type I probes

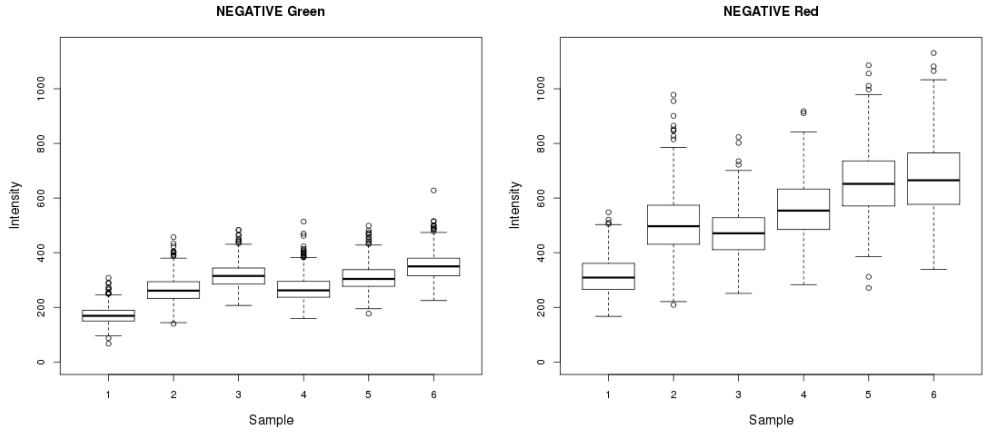

Figure 2: Negative control probes

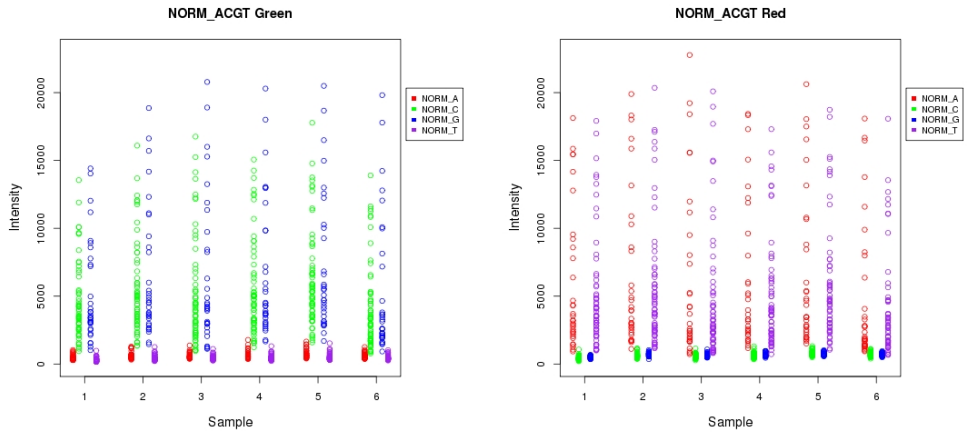

Figure 3: NORM ACGT control probes

5.2 Data distribution

Methylation intensity or beta value distribution plots are very useful for data summary, visual inspection and identification of outlier samples. Density plot is routinely generated using R function `multidensity`. However, the function is computationally intensive, and can take several hours to produce density plots for a large methylation dataset. Furthermore, density plot is difficult to understand for many investigators, also as noted in the man page of `multidensity`, density plot may not be able to display data distribution accurately for some data because of the smooth function which may lead part of the distribution to be out of range, and may obscure important details in data distribution.

ENmix's frequency polygon plot provides a better alternative for inspection of data distribution. It can accurately reflect data distribution and, like histogram it is easy to understand. It is also much faster, and only take a few minutes to produce a distribution plot for >1000 samples.

```
> mraw <- preprocessRaw(rgSet)
> #total intensity plot is userful for data quality inspection
> #and identification of outlier samples
> multifreqpoly(assayData(mraw)$Meth+assayData(mraw)$Unmeth,
+ xlab="Total intensity")
> #Compare frequency polygon plot and density plot
> beta<-getBeta(mraw, "Illumina")
> anno=getAnnotation(rgSet)
> beta1=beta[anno$Type=="I",]
> beta2=beta[anno$Type=="II",]
> library(geneplotter)
> jpeg("dist.jpg",height=900,width=600)
> par(mfrow=c(3,2))
> multidensity(beta,main="Multidensity")
> multifreqpoly(beta,main="Multifreqpoly",xlab="Beta value")
> multidensity(beta1,main="Multidensity: Infinium I")
> multifreqpoly(beta1,main="Multifreqpoly: Infinium I",xlab="Beta value")
> multidensity(beta2,main="Multidensity: Infinium II")
> multifreqpoly(beta2,main="Multifreqpoly: Infinium II",xlab="Beta value")
> dev.off()
```

See the following figures (Figure 4) generated from the above code. When type I and type II probes are plotted separately (Fig 4 bottom 4 panels) the difference in modes between type I and II probes can be appreciated. But when all probes are plotted together (Fig 4 top panels), the multidensity plot obscures these differences, while they remain readily apparent in the multifreqpoly plot. In addition, the multidensity plots appear to suggest that probes range in value from <0 to >1, whereas multifreqpoly correctly show the range from 0 to 1.

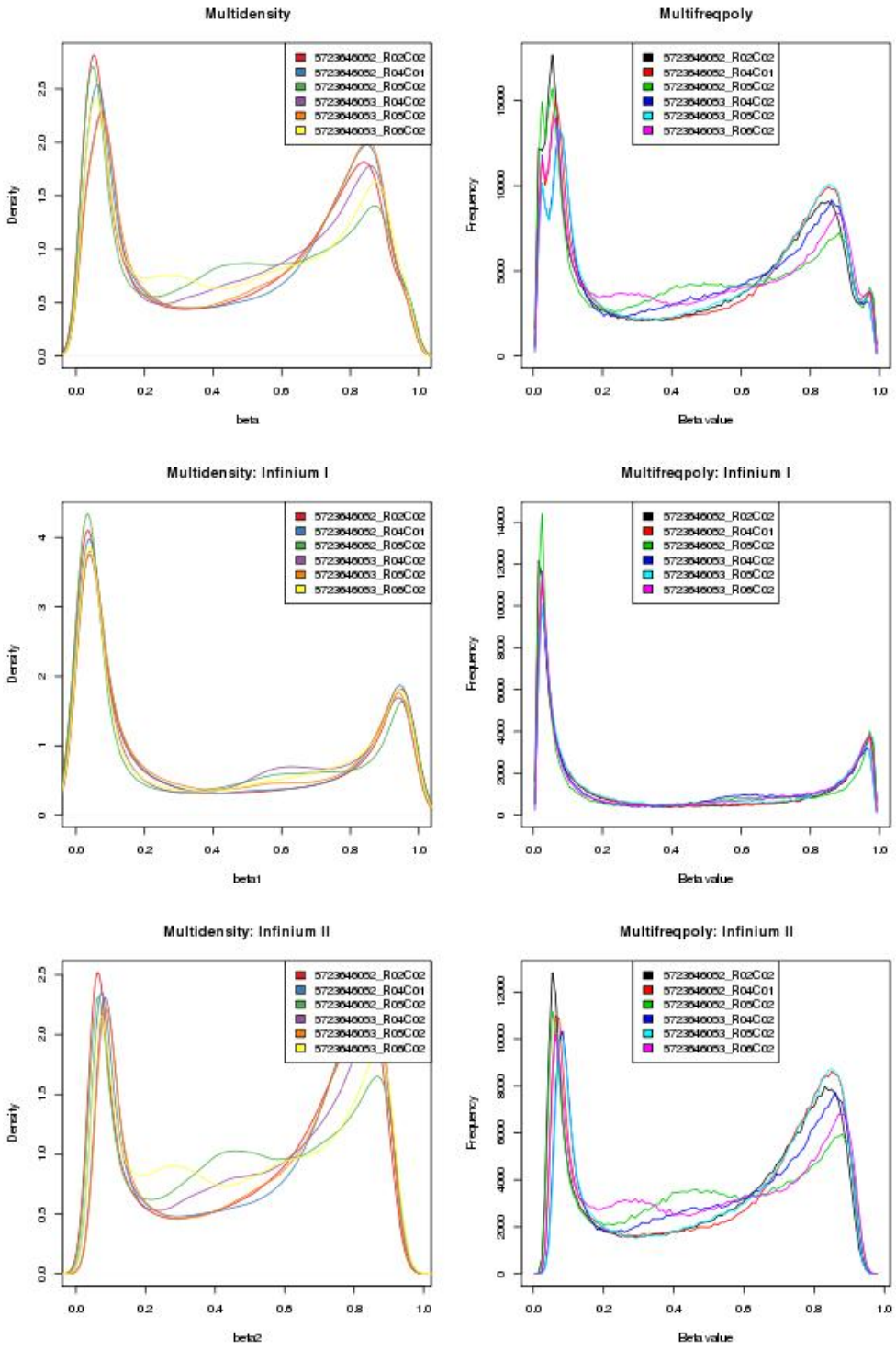

Figure 4: Methylation beta value distribution plots for all probes (top 2 panels) and for type I (middle panels) and II (bottom panels) probes separately. The smoothing function in multidensity plots (panels on left) results in misleading range and mode information which are more accurately depicted in the multifreqpoly plots (panels on right)

5.3 QC information, filtering of low quality samples and probes

Data quality measures, including detection P values, number of beads for each methylation read and average intensities for bisulfite conversion probes can be extracted using the function `QCinfo` from an object of `RGChannelSetExtended`. According default or user specified quality score thresholds, the `QCinfo` can also identify and export a list of low quality samples and CpG probes. Outlier samples in total intensity or beta value distribution were often excluded before further analysis. Such samples were tricky to be identified, by default the argument outlier=TRUE will trigger the function to identify these outlier samples automatically. Quality score figures from `QCinfo` can be used to guide the selection of quality score thresholds.

```
> qc<-QCinfo(rgSet)
```

5.4 Filtering out outliers and low quality data values

Outlier and low quality data values can have large impact on association statistical tests. Function `rm.outlier` can filter out these data points and replace them as missing values. Outliers are defined as values smaller than 3 times IQR from the lower quartile or larger than 3 times IQR from the upper quartile. Some statistical methods do not allow missing values, argument *impute=TRUE* in the function can be specified to impute missing data using k-nearest neighbors method.

```
> #filter out outliers
> b1=rm.outlier(beta)
> #filter out low quality and outlier values
> b2=rm.outlier(beta,qcscore=qcscore)
> #filter out low quality and outlier values, remove rows and columns
> # with too many missing values
> b3=rm.outlier(beta,qcscore=qcscore,rmcr=TRUE)
> #filter out low quality and outlier values, remove rows and columns
> # with too many missing values, and then do imputation
> b3=rm.outlier(beta,qcscore=qcscore,rmcr=TRUE,impute=TRUE)
```

6 Background correction

Function `preprocessENmix` incorporats a model based background correction method *ENmix*, which models methylation signal intensities with a flexible exponential-normal mixture distribution, together with a truncated normal distribution to model background noise. Users can also specify a list of poor performance CpGs to be excluded before background correction using argument *exCpG*.

See the following paper for the detailed description of the method:

Zongli Xu, et. al. ENmix: a novel background correction method for Illumina HumanMethylation450 BeadChip, Nucleic Acids Research, 2015

If argument QCinfo is specified, the low quality samples and probes identified by function `QCinfo` will be excluded before ENmix background correction. Using argument *exSample* and *exCpG*, User can also specify a list of samples or probes to be excluded before background correction.

```
> qc=QCinfo(rgSet)
> mdat<-preprocessENmix(rgSet, bgParaEst="oob", dyeCorr=TRUE, QCinfo=qc,
+ exCpG=NULL, nCores=6)
```

7 Inter-array normalization

Function `normalize.quantile.450k` can be used to perform quantile normalization on methylation intensity values.

```
> mdat<-normalize.quantile.450k(mdat, method="quantile1")
```

8 Parallel computing wrappers

DNA methylation analysis is computationally intensive. To take advantage of the widely available multi-core computation resources, we provided a few parallel computing wrappers to speed up the analysis.

8.1 Probe type bias correction

Majority of probes on Illumina 450K and EPIC BeadChips are type II probes. Although type II probes facilitate increased array genome coverage, they were shown to have decreased dynamic range and reproducibility compared to type I probes. Taking advantage of the high spatial correlation of DNA methylation levels along the human genome, The RCP (Regression on Correlated Probes) method utilizes nearby (<25 bp) type I and II probe pairs to derive the quantitative relationship between probe types and then recalibrates type II probe measurements using type I probes as referents.

```
> beta<-rcp(mdat)
```

The `BMIQ` function in R package wateRmelon can also be used to reduce probe type bias. However the funtion is computation intensive and can take very long time for large dataset. Therefore we here provided a multi-core parallel computing wrapper in the function `bmiq.mc` to speed up the process.

```
> beta<-bmiq.mc(mdat, nCores=6)
```

8.2 Batch effect correction

Function `ComBat.mc` is a multi-core parallel computing wrapper for the `ComBat` function in R package sva.

```
> batch<-factor(pData(mdat)$Slide)
> betaC<-ComBat.mc(beta, batch, nCores=6, mod=NULL)
```

9 Principal component regression analysis plot

First, principal component analysis will be performed in standardized beta value matrix (standardized for each CpG), and then the specified number of top principal components (that explain most data variation) will be used to perform linear regression with each specified variables, such as batch or environmental variables. Regression P values will be plotted to explore methylation data variance structure and to identify possible confounding variables to guide association statistical analysis.

```
> cov<-data.frame(group=pData(mdat)$Sample_Group,
+     slide=factor(pData(mdat)$Slide))
> pcrplot(beta, cov, npc=6)
```

10 Multimodal CpGs

Function `nmode.mc` uses an empirical approach to identify multimodal distributed CpGs (SNP like probes). When measured in a population of people the majority of CpGs on the Illumina HumanMethylation450 BeadChip have unimodal distributions of DNA methylation values with relatively small between-person variation. However, some CpGs (typically around 10,000 in 450k array often seemingly the result of SNPs in the probe region) may have multimodal distributions of methylation values with sizeable differences between modes and large between-person variation. These multimodal distributed data are usually caused by SNP effect, problematic probe design or other unknown artifacts instead of actual methylation level and thus should be excluded from DNA methylation analysis. Researchers have often excluded CpGs based on SNP annotation information. However, because SNP annotation always depends on population origin, we found that this approach alone may exclude many well-distributed (unimodal) CpGs, while still failing to identify other multi-modal CpGs. We developed an empirical approach to identify CpGs that are obviously not uni-modally distributed, so that researchers can make more informed decisions about whether to exclude them in their particular study populations and analyses.

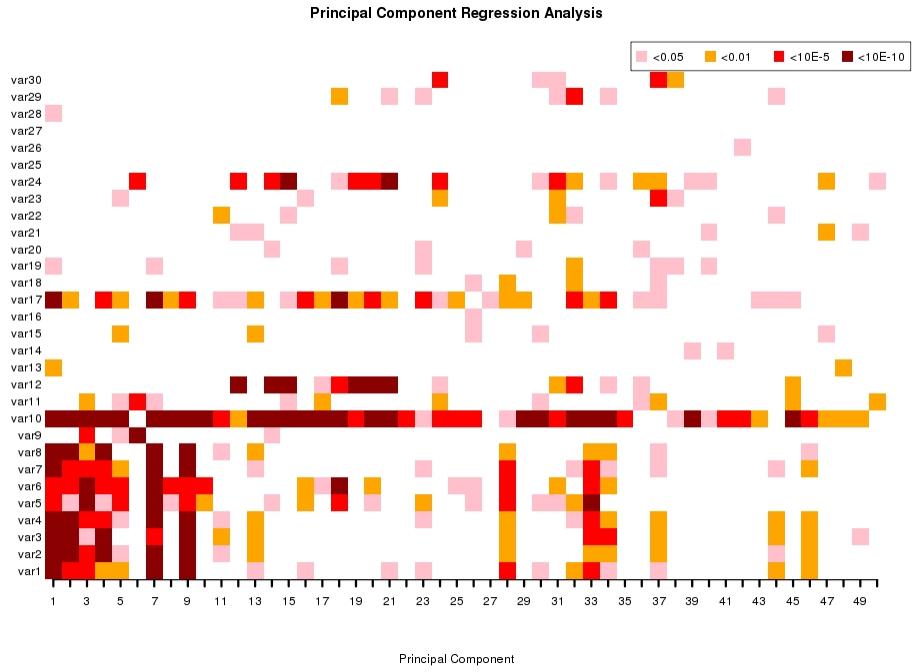

Figure 5: Example principal component regression p value plot of raw data generated using 450K methylation data from a published study

See online supplementary materials of the following paper for an evaluation of the method using published EWAS data.

Zongli Xu, Liang Niu, Leping Li and Jack A. Taylor ENmix: a novel background correction method for Illumina HumanMethylation450 BeadChip, Nucleic Acids Research, 2015

```
> nmode<- nmode.mc(beta, minN = 3, modedist=0.2, nCores = 5)
```

11 Example Analysis

Working with IDAT raw methylation data files

11.1 Example 1

```
> library(ENmix)
> #read in data
> sheet <- read.450k.sheet(file.path(find.package("minfiData"),
+     "extdata"), pattern = "csv$")
```

```
> rgSet <- read.450k.exp(targets = sheet, extended = TRUE)
> #QC info
> qc<-QCinfo(rgSet)
> #background correction and dye bias correction
> #exclude bad samples and probes according to qc
> mdat<-preprocessENmix(rgSet, bgParaEst="oob", dyeCorr=TRUE,
+                       QCinfo=qc, nCores=6)
> #inter-array normalization
> mdat<-normalize.quantile.450k(mdat, method="quantile1")
> #probe-type bias adjustment
> beta<-rcp(mdat)
```

11.2 Example 2: A more elaborated example

```
> library(ENmix)
> #read in data
> sheet <- read.450k.sheet(file.path(find.package("minfiData"),
+     "extdata"), pattern = "csv$")
> rgSet <- read.450k.exp(targets = sheet, extended = TRUE)
> #control plots
> plotCtrl(rgSet)
> #QC info
> qc<-QCinfo(rgSet)
> mraw <- preprocessRaw(rgSet)
> beta<-getBeta(mraw, "Illumina")
> #distribution plot
> multifreqpoly(beta,main="Methylation Beta value distribution")
> #Search for multimodal CpGs
> #sample size in this example data is too small for this purpose!
> bb=beta; bb[qc$detP>0.05 | qc$nbead<3]=NA #exclude low quality data first
> nmode<-nmode.mc(bb, minN = 3, modedist=0.2, nCores = 6)
> outCpG = names(nmode)[nmode>1]
> #background correction and dye bias correction
> mdat<-preprocessENmix(rgSet, bgParaEst="oob", dyeCorr=TRUE,
+                       QCinfo=qc, exCpG=outCpG, nCores=6)
> #user also can exclude bad samples and CpGs later by running:
> #mdat<-preprocessENmix(rgSet, bgParaEst="oob", dyeCorr=TRUE,nCores=6)
> #mdat <- mdat[,!(colnames(mdat) %in% qc$badsample)]
> #mdat <- mdat[!(rownames(mdat) %in% qc$badCpG),]
> #inter-array normalization
> mdat<-normalize.quantile.450k(mdat, method="quantile1")
> #probe-type bias adjustment
```

```
> beta<-rcp(mdat, nCores=6)
> # Principal component regression analysis plot
> cov<-data.frame(group=pData(mdat)$Sample_Group,
+     slide=factor(pData(mdat)$Slide))
> pcrplot(beta, cov, npc=6)
> #filter out low quality and outlier values, remove rows and columns
> #with too many missing value, and then do imputation
> beta <- rm.outlier(beta,qcscore=qc,rmcr=TRUE,impute=TRUE)
> #batch correction
> #using M values instead of beta values maybe better at this step
> batch<-factor(pData(mdat)[colnames(beta),]$Slide)
> betaC<-ComBat.mc(beta, batch, nCores=6, mod=NULL)
```

12 SessionInfo

- R version 3.2.0 (2015-04-16), `x86_64-unknown-linux-gnu`
- Locale: `LC_CTYPE=en_US.UTF-8`, `LC_NUMERIC=C`, `LC_TIME=en_US.UTF-8`, `LC_COLLATE=en_US.UTF-8`, `LC_MONETARY=en_US.UTF-8`, `LC_MESSAGES=en_US.UTF-8`, `LC_PAPER=en_US.UTF-8`, `LC_NAME=C`, `LC_ADDRESS=C`, `LC_TELEPHONE=C`, `LC_MEASUREMENT=en_US.UTF-8`, `LC_IDENTIFICATION=C`
- Base packages: base, datasets, graphics, grDevices, methods, stats, utils
- Loaded via a namespace (and not attached): tools 3.2.0

13 References

Zongli Xu, Liang Niu, Leping Li and Jack A. Taylor, ENmix: a novel background correction method for Illumina HumanMethylation450 BeadChip, Nucleic Acids Research, 2015

Illumina Inc., Infinium HD Assay Methylation Protocol Guide, Illumina, Inc. San Diego, CA.

Aryee MJ, Jaffe AE, Corrada-Bravo H, Ladd-Acosta C, Feinberg AP, Hansen KD and Irizarry RA (2014). Minfi: A flexible and comprehensive Bioconductor package for the analysis of Infinium DNA Methylation microarrays. Bioinformatics, 30(10), pp. 13631369.

Pidsley, R., CC, Y.W., Volta, M., Lunnon, K., Mill, J. and Schalkwyk, L.C. (2013) A data-driven approach to preprocessing Illumina 450K methylation array data. BMC genomics, 14, 293.

Teschendorff AE et. Al (2013). A beta-mixture quantile normalization method for correcting probe design bias in Illumina Infinium 450 k DNA methylation data. Bioinformatics.

Johnson, WE, Rabinovic, A, and Li, C (2007). Adjusting batch effects in microarray expression data using Empirical Bayes methods. Biostatistics 2007 8(1):118-127.