An Introduction to the **skewr** Package

Ryan Putney Steven Eschrich Anders Berglund

May 2, 2018

1 Introduction

The skewr package is a tool for visualizing the output of the Illumina Human Methylation 450k BeadChip(450k) to aid in quality control. It creates a panel of nine plots. Six of the plots represent the density of either the methylated intensity or the unmethylated intensity given by one of three subsets of the 485,577 total probes. These subsets include Type I-red, Type I-green, and Type II. The remaining three distributions give the density of the β-values for these same three subsets. Each of the nine plots optionally displays the distributions of the "rs" SNP probes and the probes associated with imprinted genes[1] as a series of 'tick' marks located above the x-axis.

Importance of the data

DNA methylation is an epigenetic modification of the genome believed to play a role in gene expression. Hypermethylation and hypomethylation have the potential to either silence or 'turn on' specific genes, respectively. For this reason, the role that methylation plays in disease processes, such as cancer, is of particular interest to researchers. The introduction of 450k which can efficiently query the methylation status of more than 450,000 CpG sites has given rise to the need of being able to analyze the generated data in an equally efficient, as well as accurate, manner.

450k Design

The 450k utilizes two separate assay methods on a single chip. Approximately 135,000 of the total number of probes on the 450k are of the Infinium Type I technology, as used on the preceding 27k BeadChip. The Type I probes have both a methylated and an unmethylated probe type for each CpG site. The final base of these probe types are designed to match either the methylated **C** or the **T** which results from bisulfite conversion of an unmethylated **C**. A

single base extension results in the addition of a colored dye. The color of the fluorescent dye used to measure the intensities of these probes is the same for both the methylated and unmethylated types. The Type II probes utilize one probe for both the methylated and unmethylated CpG locus. The single base extension step itself determines the methylation status of the locus. If the interrogated locus has a methylated **C**, then a green-labeled **G** is added, while a red-labeled **A** is added if the locus has the converted **T** denoting an unmethylated locus. For this reason, the level of methylation for the type II probes is always measured in the green channel, while level of unmethylation in the red. These factors mean that there are potentially six subsets of the total signal intensity: Type I-red methylated, Type I-red unmethylated, Type I-green methylated, Type I-green unmethylated, Type II methylated, and Type II unmethylated.

The preferred metric for evaluating the level of methylation for a given probe is usually the β-value, which is calculated as a ratio of the methylated signal intensity over the sum of the methylated and unmethylated signal intensities and a small offset value α, which is usually 100:

$$\beta = \frac{M}{M + U + \alpha}$$

The β-value has the advantages of being the Illumina recommended metric and of being natural and straightforward[2].

Differences in the performance of the Type I and Type II probes[3, 4] and the existence of dye bias introduced by the two-color design[5], however, have been observed. Much work has been done to correct these confounding factors, but their efficacy is usually judged in β space.

Proposed Use of skewr

skewr is designed to visualize the array data in $\log_2$ intensity space. By analyzing the data in this way, we believe that a more accurate understanding of the biological variation may be attained.

As a step in that direction, we found that the $\log_2$ distributions of the intensities may be modeled as a mixture of skew-normal distributions. A three-component model fits the Type I intensity distributions well, while two components generally fit the Type II distributions the best. We have observed a difficulty in fitting a Skew-normal mixture model to the Type II intensity distributions, especially the unmethylated probes. We have verified, however, that the two-component model works very well for samples that carry a reasonable assumption of purity.

The location of the individual components themselves may give insight into the true signal-to-noise ratio, as well as possible mechanisms of non-specific binding. The posterior probabilities for the intensities of individual probes may prove useful in distinguishing the true biological variability in the methylation levels.

Finite Mixture of Skew-normal Distributions

skewr utilizes the mixsmsn package to estimate the parameters for the Skew-normal components that make up the finite mixture model for the intensity distributions[6]. mixsmsn deals with the family of distributions known as the scale mixtures of the skew-normal distributions(SMSN)[7].

If a random variable Z has a skew-normal distribution with location parameter μ, scale parameter σ^2, and skewness, or shape, parameter λ, its density is given by:

$$\psi(z) = 2\phi(z; \mu, \sigma^2)\Phi\left(\frac{\lambda(z-\mu)}{\sigma}\right), \tag{1}$$

where $\phi(\cdot; \mu, \sigma^2)$ is the probability density function and $\Phi(\cdot)$ is the cumulative distribution function, both of the univariate normal distribution. Random variable Z is then denoted as $Z \sim \text{SN}(\mu, \sigma^2, \lambda)$.

Y is a random variable with an SMSN distribution if:

$$Y = \mu + U^{-1/2}Z, \tag{2}$$

where μ is the location parameter, $Z \sim \text{SN}(0, \sigma^2, \lambda)$, and U is a positive random variable given by the distribution function $H(\cdot, \nu)$. Then U becomes the scale factor, and its distribution $H(\cdot, \nu)$ is the mixing distribution indexed by the parameter ν which may be univariate or multivariate. Since skewr is only dealing with the skew-normal distribution of the SMSN family U always has the value 1, and the parameter ν has no significance. Therefore,

$$Y = \mu + Z, \quad Z \sim \text{SN}(0, \sigma^2, \lambda) \tag{3}$$

A finite mixture of skew-normal distributions for a random sample $\mathbf{y} = (y_1, y_2, \ldots, Y_n)$ with g number of components is given by:

$$f(y_i, \Theta) = \sum_{j=1}^{g} p_j \psi(y_i; \theta_j), \quad p_j \geq 0, \quad \sum_{j=1}^{g} p_j = 1, \quad i = 1, \ldots, n, \quad j = 1, \ldots, g, \tag{4}$$

where $\theta_j = (\mu_j, \sigma_j^2, \lambda_j)$ are the parameters for the j^{th} skew-normal component, $p_1, \ldots, p_g$ are the mixing probabilities, or weights, for the components, and Θ is a vector of all the parameters: $\Theta = ((p_1, \ldots, p_g), \theta_1, \ldots, \theta_g)$.

Finally, the estimated posterior probability $\hat{z}_{ij}$ is given by:

$$\hat{z}_{ij} = \frac{\hat{p}_j \text{SN}(y_i; \hat{\mu}_j, \hat{\sigma}_j^2, \hat{\lambda}_j)}{\sum_{k=1}^{g} \hat{p}_k \text{SN}(y_i; \hat{\mu}_k, \hat{\sigma}_k^2, \hat{\lambda}_k)}, \quad i = 1, \ldots, n, \quad j = 1, \ldots, g, \tag{5}$$

which is essentially the probability of finding a given point in the j^{th} component over the probability of finding it in the mixture model.

2 Getting Started

Installation

skewr is a package that is designed to work with several preexisting packages. Therefore, a fair number of dependencies are required. The following packages must be installed:

```
packages.install('mixsmsn')
source('http://bioconductor.org/biocLite.R')
biocLite(c('skewr', 'methylumi', 'minfi', 'wateRmelon',
           'IlluminaHumanMethylation450kmanifest', 'IRanges'))
```

And to run this vignette as written:

```
biocLite('minfiData')
```

Load skewr

```
library(skewr)
library(minfiData)
```

3 Sample Session

skewr provides a convenience function for retrieving clean barcode names from all idat files found in the path, or vector of paths, given as a parameter. `getMethyLumiSet` is a wrapper function utilizing the `methylumIDAT` function provided by methylumi[8]. `getMethyLumiSet` will process all idat files in the path to the directory given by the first argument, or default to the working directory if none is given. A vector of barcodes may be provided if only those specific idat files are to be processed. The default output will be a raw MethyLumiSet, unless a normalization method is specified when calling `getMethyLumiSet`. It is unlikely that preprocessing will be desired when calling `getMethyLumiSet` unless one is only interested in a single normalization method. It is much more likely that `getMethyLumiSet` will be called to assign the raw `MethyLumiSet` object to a label. Then the `preprocess` method provided by the skewr may be called to return a number of objects with different normalization methods applied[9, 1].

As an example of the use of the `getBarcodes` function provided by skewr, we will start by retrieving the barcodes of some idat files provided by minfiData[10].

```
baseDir <- system.file("extdata/5723646052", package = "minfiData")
barcodes <- getBarcodes(path = baseDir)
barcodes
```

```
## [1] "5723646052_R02C02" "5723646052_R04C01" "5723646052_R05C02"
```

```
methylumiset.raw <- getMethyLumiSet(path = baseDir, barcodes =  barcodes[1:2])
methylumiset.illumina <- preprocess(methylumiset.raw, norm = 'illumina',
                                    bg.corr = FALSE)
```

To allow for a more efficient demonstration, however, the rest of the vignette will utilize a `MethyLumiSet` object that is supplied by the Bioconductor package wateRmelon. This reduced data set only contains 3363 features out of the 485,577 total probes. We will only use one sample, but skewr will plot panels for all samples contained within the `MethyLumiSet` passed to it.

```
data(melon)
melon.raw <- melon[,11]
```

Additional normalization methods may be performed as follows:

```
melon.illumina <- preprocess(melon.raw, norm = 'illumina', bg.corr = TRUE)
melon.SWAN <- preprocess(melon.raw, norm = 'SWAN')
melon.dasen <- preprocess(melon.raw, norm = 'dasen')
```

`getSNparams` will subset the probes by either methylated, `M`, or unmethylated, `U`, and `I-red`, `I-green`, or `II`. The subset of probes are then fitted to a skew-normal finite mixture model. The mixture modelling is provided by the `smsn.mix` method of the mixsmsn package. Assign the return value or each of the six `getSNparams` calls to a separate label.

```
sn.raw.meth.I.red <- getSNparams(melon.raw, 'M', 'I-red')
sn.raw.unmeth.I.red <- getSNparams(melon.raw, 'U', 'I-red')
sn.raw.meth.I.green <- getSNparams(melon.raw, 'M', 'I-green')
sn.raw.unmeth.I.green <- getSNparams(melon.raw, 'U', 'I-green')
sn.raw.meth.II <- getSNparams(melon.raw, 'M', 'II')
sn.raw.unmeth.II <- getSNparams(melon.raw, 'U', 'II')
```

If you want to compare the raw data for your experiment with the same data after a normalization method has been performed, you would carry out the same steps for each of your normalized `MethyLumiSet`s. For example:

```
sn.dasen.meth.I.red <- getSNparams(melon.dasen, 'M', 'I-red')
sn.dasen.unmeth.I.red <- getSNparams(melon.dasen, 'U', 'I-red')
sn.dasen.meth.I.green <- getSNparams(melon.dasen, 'M', 'I-green')
sn.dasen.unmeth.I.green <- getSNparams(melon.dasen, 'U', 'I-green')
sn.dasen.meth.II <- getSNparams(melon.dasen, 'M', 'II')
sn.dasen.unmeth.II <- getSNparams(melon.dasen, 'U', 'II')
```

Before the panel plots can be made, the values returned by `getSNparams` must be put into a list so that probes with the same assay and channel are in a separate list object with the methylated probes listed first followed by the unmethylated. Note that skewr is designed to create a series of panel plots for an entire experiment. It will not work if one attempts to index a single sample and its accompanying skew-normal models. See section 4 for information on how to better view a single sample within an experiment.

```
raw.I.red.mixes <- list(sn.raw.meth.I.red, sn.raw.unmeth.I.red)
raw.I.green.mixes <- list(sn.raw.meth.I.green, sn.raw.unmeth.I.green)
raw.II.mixes <- list(sn.raw.meth.II, sn.raw.unmeth.II)
```

The `panelPlots` method takes the original `MethyLumiSet` object as the first parameter. The following parameters consist of the listed `I-red`, `I-green`, and `II` models, in that order.

```
panelPlots(melon.raw, raw.I.red.mixes, raw.I.green.mixes,
           raw.II.mixes, norm = 'Raw')
```

The `samp.num` parameter of `panelPlots`, may also be specified if plots for only one sample out of an experimenter is wanted. The `samp.num` is an integer indexing the sample column within the `MethyLumiSet` object.

Of course, the listing of the Skew-normal objects may be done in the call to `panelPlots`.

```
panelPlots(melon.dasen,
           list(sn.dasen.meth.I.red, sn.dasen.unmeth.I.red),
           list(sn.dasen.meth.I.green, sn.dasen.unmeth.I.green),
           list(sn.dasen.meth.II, sn.dasen.unmeth.II), norm = 'dasen')
```

`panelPlots` will produce a panel plot for each sample in your experiment, Figure 1. Producing panel plots of the same experiment with different types of normalization applied may allow a better understanding of what the normalization method is actually doing to the data.

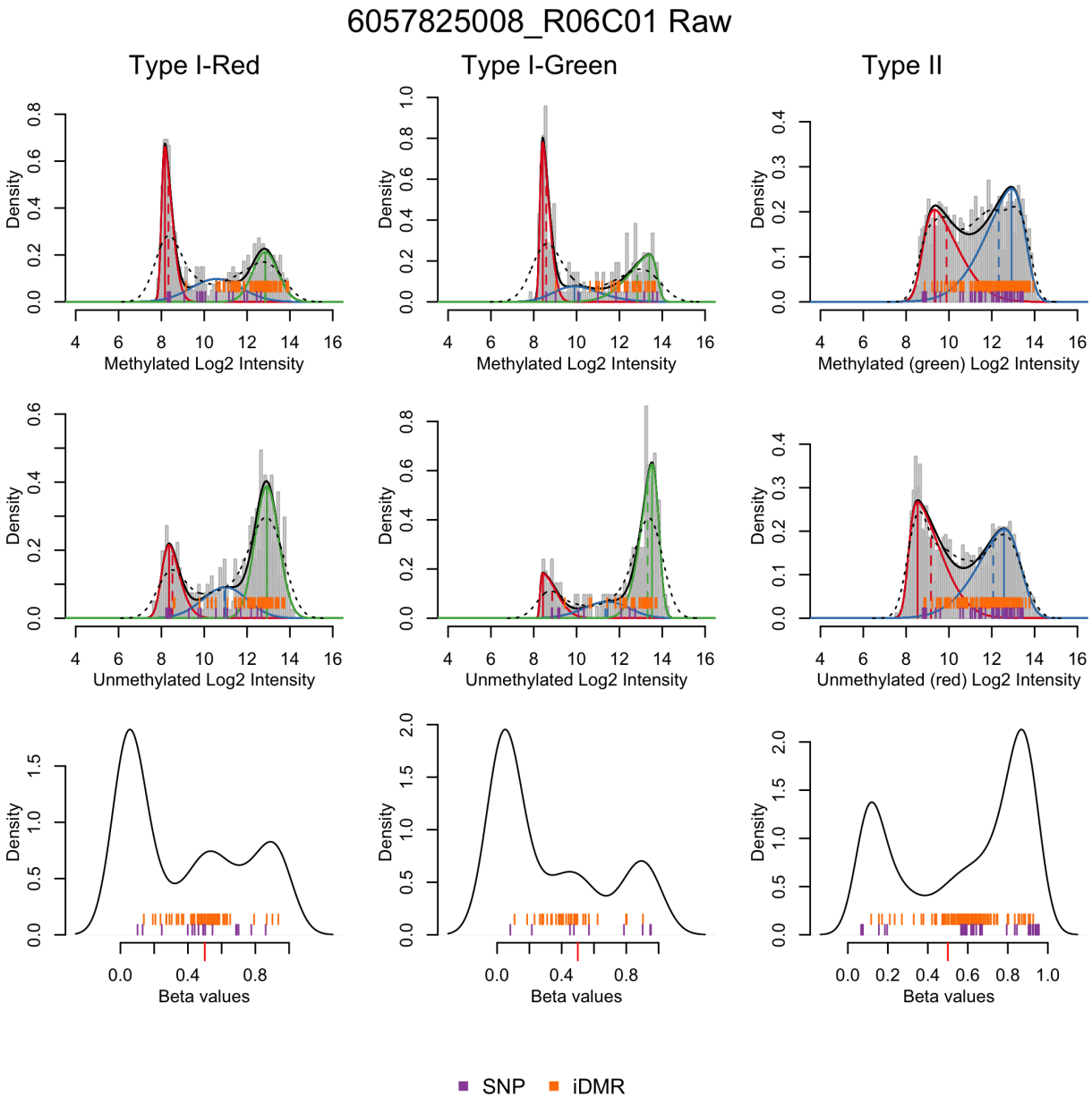

Figure 1: Panel Plot for One Sample

4 Single Plots

It is possible to call `panelPlots` with the `plot` parameter as `plot = 'frames'`. In this case, a sample number for a single sample contained within your experiment must also be passed as a parameter. If `panelPlots` is called in this manner, nine separate, large plots will be created. Each of these nine plots correspond to a single pane that would be found in the panel plot for the selected sample. In addition, each of these separate plots, except for the beta plots, will contain a legend with useful information, such as the means and modes for each of the components of the mixture. The means and modes are also contained as part of the `Skew.normal` object returned by `getSNparams` and may be accessed in the following manner:

```
class(sn.raw.meth.I.red[[1]])

## [1] "Skew.normal"

names(sn.raw.meth.I.red[[1]])

##  [1] "mu"        "sigma2"    "shape"     "pii"       "nu"
##  [6] "aic"       "bic"       "edc"       "icl"       "iter"
## [11] "n"         "obs.prob"  "means"     "modes"     "dens.list"

sn.raw.meth.I.red[[1]]$means

## [1] 12.852632 10.552378  8.326424

sn.raw.meth.I.red[[1]]$modes

## [1] 12.851167 10.559855  8.168894
```

To generate each panel as a single plot, see examples Figure 2 and Figure 3:

```
panelPlots(melon.raw, raw.I.red.mixes, raw.I.green.mixes,
           raw.II.mixes, plot='frames', frame.nums=c(1,3), norm='Raw')
```

The graph includes the histogram of the intensities expressed as probabilities as would be produced by the generic `hist` function. The dotted line represents the kernel density estimation. The colored curves are the probability distributions of the individual components with the vertical solid and dotted lines being the mode and mean, respectively, for each component. The legend identifies the mean and the mode and provides a floating point value, rounded off to three decimal places, for each. The solid black line is the sum of the components, representing the fit of the Skew-normal mixture model.

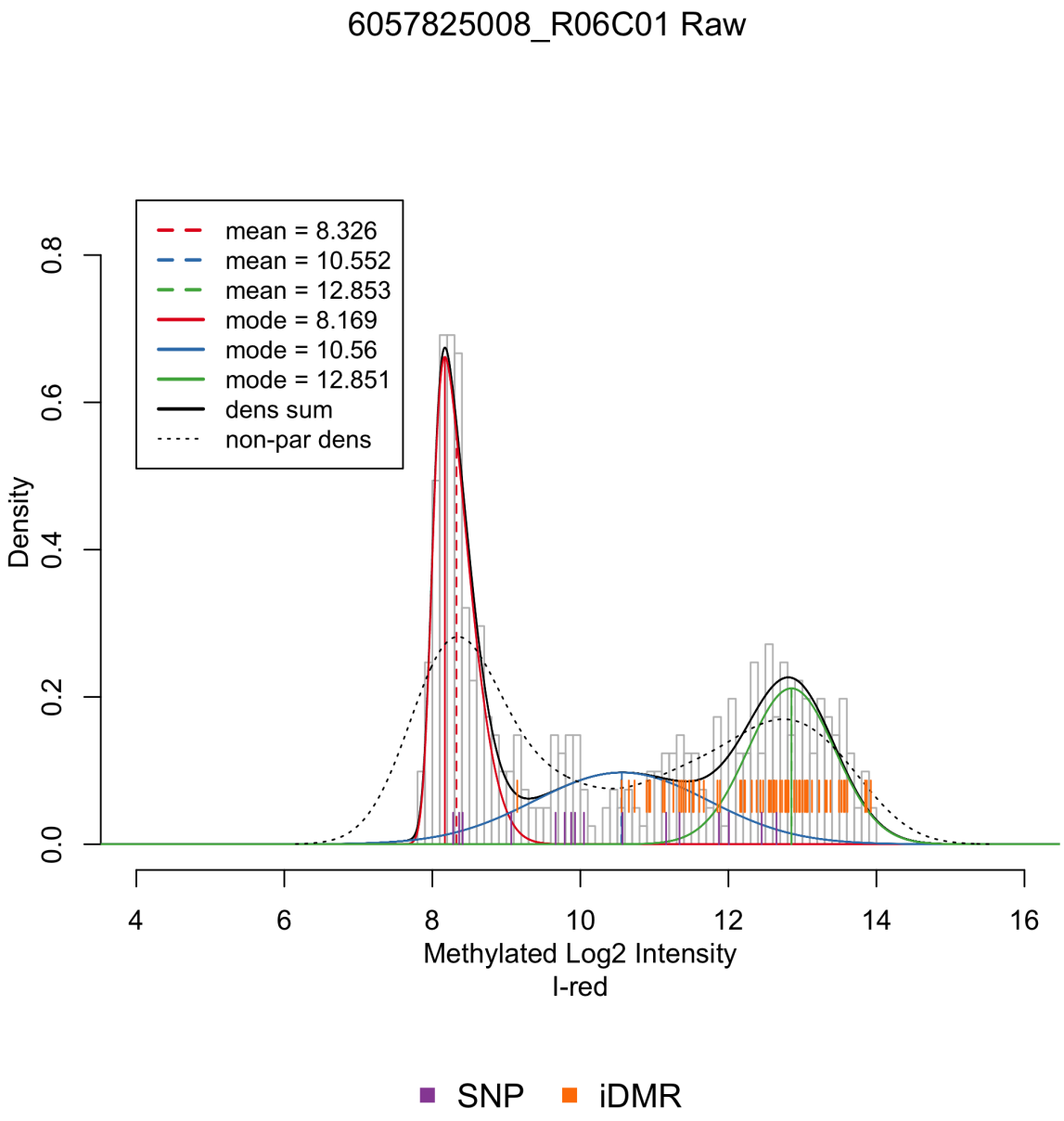

Figure 2: Single Frame Showing Skew-normal Components

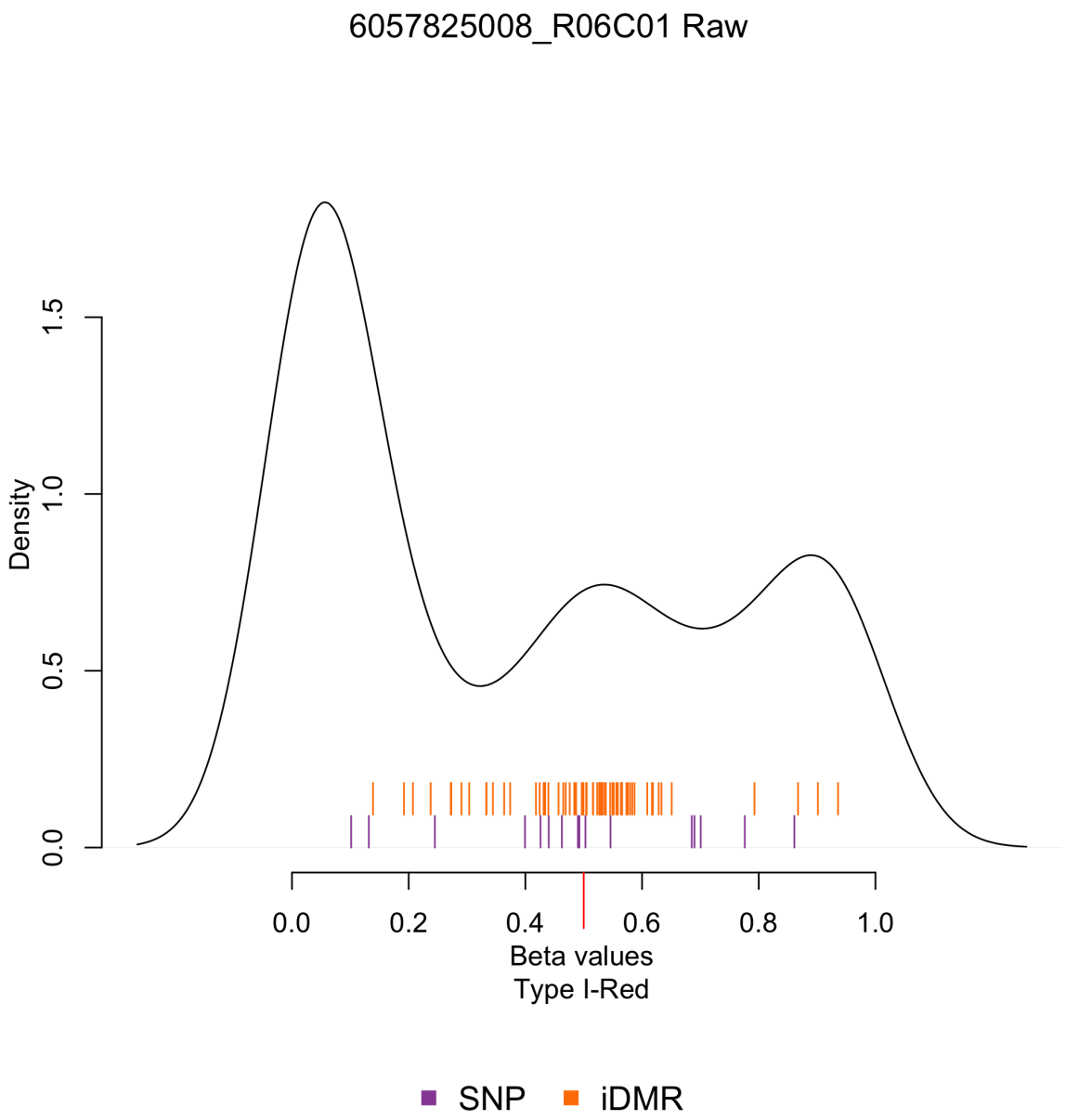

Figure 3: Single Frame Showing Beta Distribution for Type I Red

5 sessionInfo

- R version 3.5.0 (2018-04-23), `x86_64-apple-darwin15.6.0`
- Locale: `C/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8`
- Running under: `OS X El Capitan 10.11.6`
- Matrix products: default
- BLAS:
  `/Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRblas.0.dylib`
- LAPACK:
  `/Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRlapack.dylib`
- Base packages: base, datasets, grDevices, graphics, methods, parallel, stats, stats4, utils
- Other packages: AnnotationDbi 1.42.0, Biobase 2.40.0, BiocGenerics 0.26.0, BiocParallel 1.14.0, Biostrings 2.48.0, DelayedArray 0.6.0, FDb.InfiniumMethylation.hg19 2.2.0, GenomeInfoDb 1.16.0, GenomicFeatures 1.32.0, GenomicRanges 1.32.0, IRanges 2.14.1, IlluminaHumanMethylation450kanno.ilmn12.hg19 0.6.0, IlluminaHumanMethylation450kmanifest 0.4.0, ROC 1.56.0, S4Vectors 0.18.1, SummarizedExperiment 1.10.0, TxDb.Hsapiens.UCSC.hg19.knownGene 3.2.2, XVector 0.20.0, bumphunter 1.22.0, foreach 1.4.4, ggplot2 2.2.1, illuminaio 0.22.0, iterators 1.0.9, knitr 1.20, limma 3.36.0, locfit 1.5-9.1, lumi 2.32.0, matrixStats 0.53.1, methylumi 2.26.0, minfi 1.26.0, minfiData 0.26.0, mixsmsn 1.1-4, mvtnorm 1.0-7, org.Hs.eg.db 3.6.0, reshape2 1.4.3, scales 0.5.0, skewr 1.12.1, wateRmelon 1.24.0
- Loaded via a namespace (and not attached): BiocInstaller 1.30.0, DBI 1.0.0, DelayedMatrixStats 1.2.0, GEOquery 2.48.0, GenomeInfoDbData 1.1.0, GenomicAlignments 1.16.0, HDF5Array 1.8.0, KernSmooth 2.23-15, MASS 7.3-50, Matrix 1.2-14, R6 2.2.2, RColorBrewer 1.1-2, RCurl 1.95-4.10, RSQLite 2.1.0, Rcpp 0.12.16, Rhdf5lib 1.2.0, Rsamtools 1.32.0, XML 3.98-1.11, affy 1.58.0, affyio 1.50.0, annotate 1.58.0, assertthat 0.2.0, base64 2.0, beanplot 1.2, bindr 0.1.1, bindrcpp 0.2.2, biomaRt 2.36.0, bit 1.1-12, bit64 0.9-7, bitops 1.0-6, blob 1.1.1, codetools 0.2-15, colorspace 1.3-2, compiler 3.5.0, data.table 1.11.0, digest 0.6.15, doRNG 1.6.6, dplyr 0.7.4, evaluate 0.10.1, genefilter 1.62.0, glue 1.2.0, grid 3.5.0, gtable 0.2.0, highr 0.6, hms 0.4.2, httr 1.3.1, lattice 0.20-35, lazyeval 0.2.1, magrittr 1.5, mclust 5.4, memoise 1.1.0, mgcv 1.8-23, multtest 2.36.0, munsell 0.4.3, nleqslv 3.3.1, nlme 3.1-137, nor1mix 1.2-3, openssl 1.0.1, pillar 1.2.2, pkgconfig 2.0.1, pkgmaker 0.22, plyr 1.8.4, preprocessCore 1.42.0, prettyunits 1.0.2, progress 1.1.2, purrr 0.2.4, quadprog 1.5-5, readr 1.1.1, registry 0.5, reshape 0.8.7, rhdf5 2.24.0, rlang 0.2.0, rngtools 1.2.4, rtracklayer 1.40.0, siggenes 1.54.0, splines 3.5.0, stringi 1.2.2, stringr 1.3.0, survival 2.42-3, tibble 1.4.2, tidyr 0.8.0, tools 3.5.0, xml2 1.2.0,

xtable 1.8-2, zlibbioc 1.26.0

References

[1] Ruth Pidsley, Chloe C Y Wong, Manuela Volta, Katie Lunnon, Jonathan Mill, and Leonard C Schalkwyk. A data-driven approach to preprocessing illumina 450k methylation array data. *BMC Genomics*, 14:293, 2013. doi:10.1186/1471-2164-14-293.

[2] Pan Du, Xiao Zhang, Chiang-Ching Huang, Nadereh Jafari, Warren A Kibbe, Lifang Hou, and Simon M Lin. Comparison of Beta-value and M-value methods for quantifying methylation levels by microarray analysis. *BMC Bioinformatics*, 11:587, 2010. doi:10.1186/1471-2105-11-587, PMID:21118553.

[3] Marina Bibikova, Bret Barnes, Chan Tsan, Vincent Ho, Brandy Klotzle, Jennie M Le, David Delano, Lu Zhang, Gary P Schroth, Kevin L Gunderson, Jian-Bing Fan, and Richard Shen. High density DNA methylation array with single CpG site resolution. *Genomics*, 98(4):288–295, 2011. doi:10.1016/j.ygeno.2011.07.007, PMID:21839163.

[4] Sarah Dedeurwaerder, Matthieu Defrance, Emilie Calonne, Hélène Denis, Christos Sotiriou, and François Fuks. Evaluation of the Infinium Methylation 450K technology. *Epigenomics*, 3(6):771–784, 2011. doi:10.2217/epi.11.105, PMID:22126295.

[5] Sarah Dedeurwaerder, Matthieu Defrance, Martin Bizet, Emilie Calonne, Gianluca Bontempi, and François Fuks. A comprehensive overview of infinium humanmethylation450 data processing. *Briefings in Bioinformatics*, 15(6):929–941, 2014. doi:10.1093/bib/bbt054, PMID:23990268.

[6] Marcos Oliveira Prates, Celso Rômulo Barbosa Cabral, and Víctor Hugo Lachos. mixsmsn: Fitting finite mixture of scale mixture of skew-normal distributions. *Journal of Statistical Software*, 54(12):1–20, 2013. URL: http://www.jstatsoft.org/v54/i12/.

[7] Rodrigo M. Basso, Víctor H. Lachos, Celso Rômulo Barbosa Cabral, and Pulkak Ghosh. Robust mixture modeling based on scale mixtures of skew-normal distributions. *Computational Statistics and Data Analysis*, 54(12):2926–2941, 2010. doi:doi:10.1016/j.csda.2009.09.031.

[8] Sean Davis, Pan Du, Sven Bilke, Tim Triche, Jr., and Moiz Bootwalla. *methylumi: Handle Illumina methylation data*, 2014. R package version 2.12.0.

[9] Jovana Maksimovic, Lavinia Gordon, and Alicia Oshlack. SWAN: Subset quantile Within-Array Normalization for Illumina Infinium HumanMethylation450 BeadChips. *Genome Biology*, 13(6):R44, 2012. doi:10.1186/gb-2012-13-6-r44, PMID:22703947.

[10] Kasper Daniel Hansen, Martin Aryee, and Winston Timp. *minfiData: Example data for the Illumina Methylation 450k array*. R package version 0.7.1.

[11] Leonard C Schalkwyk, Ruth Pidsley, Chloe CY Wong, Nizar Touleimat, Matthieu Defrance, Andrew Teschendorff, and Jovana Maksimovic. *wateRmelon: Illumina 450 methylation array normalization and metrics*, 2013. R package version 1.5.1.

[12] Martin J Aryee, Andrew E Jaffe, Hector Corrada Bravo, Christine Ladd-Acosta, Andrew P Feinberg, Kasper D Hansen, and Rafael A Irizarry. Minfi: a flexible and comprehensive Bioconductor package for the analysis of Infinium DNA methylation microarrays. *Bioinformatics*, 30(10):1363–1369, 2014. doi:10.1093/bioinformatics/btu049, PMID:24478339.

[13] The Cancer Genome Atlas Research Network. Genomic and epigenomic landscapes of adult de novo acute myeloid leukemia. *New England Journal of Medicine*, 368(22):2059–74, May 30 2013. (data accessible at NCBI GEO database (Edgar et al., 2002), accession GSE49618). doi:DOI:10.1056/NEJMoa1301689, PMID:23634996.