Working notes

Prediction of the DNA methylation of MGMT with TCGA raw data (format IDAT) from Inifinium HM-27k platform `mgmtstp27` (document in preparation!)

BADY P.

October 3, 2014

Abstract

The document investigates the problem of discrepancy between two runs of prediction (level2 of new TCGA database 2014, and the Table S4 from [2]). We demosntrate that the present data from level 2 (version 2014) is not appropriate to predict DNA methylation state of MGMT promoter with MGMT-STP27 model ([2]). However, the combination of the data from level 1 with the use of R package `methylumi` ([3]) offers a very simple solution to reproduce the results of the Table S4.

Contents

1 Motivations

The objective on this document was to answer the question of Liu Qun given below:

Because there are more TCGA GBM methylation data available since the publish of your paper, I want to calculate the MGMT methylation status of all available TCGA samples. I downloaded the level2 HM-27K data of TCGA GBM samples since they have been preprocessed same as the Illumina GenomeStudio program. I calculated the M-value of the probe cg12434587 and cg12981137 using the equation M-value = log2((max(signal methylation,0)+1)/ (max(signal unmethylation,0)+1)). Then I calculated the methylation probability by inverse logit of the equation "4.3215+0.5271*cg12434587+0.9265*cg12981137" provided by your paper. But I can't get the same value "MGMT-STP27 (response)" in the supplementary Table S4 of your paper. For some samples, for example "TCGA-02-0009", I got different "MGMT-STP27 (class)" result given the cut-off of 0.358.

To investigate the problem of discrepancy between the two runs of prediction, we compare the prediction obtained in using the level 1 and level 2 of new version of TCGA GBM data from HM-27K platform and we compare the outputs with the table S4 from [2].

2 Data preprocessing and DNA methylation state of MGMT promoter

The HM-27K data (level1 and level2) comes from TCGA Data Portal (URL: `https://tcga-data.nci.nih.gov/tcga/`).

```
# file names and barcode for samples
infofile0 <- read.table("/export/scratch/data/monikaproject/TCGA8/DNAmethylation/file_manifest.txt",
                        h=TRUE,sep="\t")
infofile1 <- infofile0[infofile0$Level==1,]
infofile3 <- infofile1[,c("Sample","File.Name","Barcode")]
infofile3$File.Name <- sub("_Grn.idat$|_Red.idat$","",infofile3$File.Name)
infofile3$Patient <- substring(infofile3$Sample,1,12)
dim(infofile3)
infofile3 <- unique(infofile3)
```

```
rownames(infofile3) <- infofile3$File.Name
# i kept only the sample with one observations
w <- table(infofile3$Sample)
w[w>1]
dim(infofile3)
exclude1 <- rownames(infofile3[is.element(infofile3$Patient,c("TCGA-19-4065","TCGA-74-6573")),])
infofile3 <- infofile3[!is.element(infofile3$Patient,c("TCGA-19-4065","TCGA-74-6573")),]
dim(infofile3)
```

2.1 Importation of the level1 data

The set of functions contained in R package **minfi** ([1]) ware used to preprocess and to import raw HM-450K data (format IDAT). However, the function contains in this package don't work with HM-27K. Consequently, The function `methylumIDAT` from the R package `methylumi` ([3]) is used to import and to preprocess the raw HM-27K data.

```
### GBM from TCGA
datadir <- "/export/scratch/data/monikaproject/TCGA8/DNAmethylation/DNA_Methylation/JHU_USC__HumanMethylation2
sampletcga <- list.files(datadir)
sampletcga <- unique(sub("_Grn.idat$|_Red.idat$","",sampletcga))
commontcga <- intersect(sampletcga,rownames(infofile3))
infofile3 <- infofile3[commontcga,]
w <- table(infofile3$Patient)
w[w>1]
infofile3<- infofile3[!is.element(as.character(infofile3$Patient),names(w[w>1])),]
length(sampletcga)
# load R package methylumi
require(methylumi)
rgset0<- methylumIDAT(barcode=as.character(infofile3$File.Name),idatPath=datadir)
head(colnames(rgset0))
table(colnames(rgset0)==rownames(infofile3),useNA="always")
colnames(rgset0) <- infofile3[colnames(rgset0),"Patient"]
head(colnames(rgset0))
# no normalization for HM-27k, see help "For HumanMethylation27 data, the function does nothing"
norm27k <- normalizeMethyLumiSet(rgset0)
u27k <- unmethylated(norm27k)
m27k <- methylated(norm27k)
mvalue0 <- log2((m27k+1)/(u27k+1))
save(mvalue0 ,file="mvalue0.rda")
```

As mentioned in the documentation, the function `normalizeMethyLumiSet` does nothing for HM-27K data(no normalization) that is similar to the function `preprocessRaw` from package `minfi`. To obtain a correct/coherent estimation of the DNA methylation state of MGMT promoter, the "raw" preprocessing (no normalisation) is required. The prediction can be directly provided by the function `MGMTpredict` from the R package `mgmtstp27` as defined in [2](URL: `http://lausanne.isb-sib.ch/~pbady/Rpackages.html`).The model and data are contains in an internal object `glm` called `MGMTSTP27`. An additional numerical vector called `perf` containing performance information and optimal cut-off (see [2]) was associated with this object.The model is described below:

```
# model description see bady et al. 2012
require(mgmtstp27)
data(MGMTSTP27)
MGMTSTP27
```

```
Call:  glm(formula = y ~ cg12434587 + cg12981137, family = binomial,
    data = tmp)

Coefficients:
(Intercept)   cg12434587   cg12981137
     4.3215       0.5271       0.9265

Degrees of Freedom: 67 Total (i.e. Null);  65 Residual
Null Deviance:            94.03
Residual Deviance: 30.14        AIC: 36.14
```

```
 summary(MGMTSTP27)
Call:
glm(formula = y ~ cg12434587 + cg12981137, family = binomial,
    data = tmp)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0674  -0.2682  -0.1469   0.2098   2.2753

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   4.3215     1.2200   3.542 0.000397
cg12434587    0.5271     0.3021   1.745 0.080988
cg12981137    0.9265     0.3018   3.069 0.002145

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 94.033  on 67  degrees of freedom
Residual deviance: 30.143  on 65  degrees of freedom
AIC: 36.143

Number of Fisher Scoring iterations: 6
 MGMTSTP27$perf
        cut    sens      spec       pvp      pvn      prev
1 0.3582476 0.96875 0.8888889 0.8857143 0.969697 0.4705882
 # prediction MGMT
 load("mvalue0.rda")
 data1 <- as.data.frame(t(mvalue0))
 lumiMgmtTcgaGBM27k <- MGMTpredict(data1)
 save(lumiMgmtTcgaGBM27k,file="lumiMgmtTcgaGBM27k.rda")
```

2.2 Importation of the level2 data

The data from level2 are prepared in using additional set of function contained the file `addfunc27k.R` (URL: `http://lausanne.isb-sib.ch/~pbady/files/rpackages/addfunc27k.R`).

```
 source("addfunc27k.R")
 # file list and data importation
 # Data importation (level 2)
 dirmethyl27 <- "/export/scratch/data/monikaproject/TCGA8/DNAmethylation/DNA_Methylation/JHU_USC__HumanMethylati
 wfiles1 <- list.files(dirmethyl27)
 str1 <- strsplit(wfiles1,"[.]")
 wsamplenames1 <- unlist(lapply(str1,function(x) x[6]))
 files1 <- paste(dirmethyl27,wfiles1,sep="/")
 names(files1) <- wsamplenames1
 filenames <- cbind(getTCGAnames(wsamplenames1),as.character(files1))
 colnames(filenames)[13] <- "FileName"
 dim(filenames)
 w <- table(filenames$Patient)
 w[w>1]
 filenames <- filenames[!is.element(as.character(filenames$Patient),names(w[w>1])),]
 dim(filenames)
 level2data27k  <- readTCGA.HM27k(filenames=as.character(filenames$FileName),samplenames=as.character(filenames
 save(level2data27k ,file="level2data27k.rda")
```

The function `MGMTpredict` is used to estimate DNA methylation state of MGMT promoter as follow:

```
 load("level2data27k.rda")
 # prediction of DNA methylation status of MGMT promoter
 rawunmeth0 <- level2data27k$unmethyl
 rawmeth0 <- level2data27k$methyl
 # computation of M-values
 mvalue0 <- log2((rawmeth0+1)/(rawunmeth0+1))
 # data preparation
 mvalue0 <- as.data.frame(mvalue0)
 # prediction MGMT
 data1 <- as.data.frame(mvalue0)
 MgmtTcgaGBM27k <- MGMTpredict(data1)
 save(MgmtTcgaGBM27k,file="MgmtTcgaGBM27k.rda")
```

3 Comparison between the level1 and level2 from TCGA GBM database ([4])

The comparison analyses for level1 (called lumiMgmtTcgaGBM27k) and level2 (call MgmtTcgaGBM27k) are given below:

```
load("MgmtTcgaGBM27k.rda")
load("lumiMgmtTcgaGBM27k.rda")
intersectx <- intersect(rownames(MgmtTcgaGBM27k),rownames(lumiMgmtTcgaGBM27k))
table(level2=MgmtTcgaGBM27k[intersectx,"state"],level1=lumiMgmtTcgaGBM27k[intersectx,"state"])
```

```
      level1
level2   M   U
     M 130   0
     U  12 143
```

```
summary(MgmtTcgaGBM27k[intersectx,"pred"])
```

```
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
0.0004802 0.0058640 0.1299000 0.4440000 0.9780000 0.9996000
```

```
summary(lumiMgmtTcgaGBM27k[intersectx,"pred"])
```

```
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
0.001534 0.041550 0.336900 0.493000 0.979200 0.999500
```

```
cor(MgmtTcgaGBM27k[intersectx,"pred"],lumiMgmtTcgaGBM27k[intersectx,"pred"])
```

```
[1] 0.9885265
```

```
selectx <- !(MgmtTcgaGBM27k[intersectx,"state"]==lumiMgmtTcgaGBM27k[intersectx,"state"])
MgmtTcgaGBM27k[intersectx,][selectx,"pred"]
```

```
 [1] 0.0506447 0.1717732 0.3178457 0.2827167 0.3086338 0.3093379 0.1959487 0.1478479
 [9] 0.1849132 0.3566812 0.1286520 0.2425273
```

```
lumiMgmtTcgaGBM27k[intersectx,][selectx,"pred"]
```

```
 [1] 0.3583862 0.5349412 0.5342883 0.4305375 0.4655016 0.5291514 0.4134365 0.4142398
 [9] 0.4672015 0.5566097 0.4108445 0.4517754
```

The predictions from level1 and level2 are clerly different. The figure 1 illustrates these reulst without ambiguity: the probabilities trend to be underestimated by model when we directly work with level 2 of the last version of TCGA database.

```
plot(lumiMgmtTcgaGBM27k[intersectx,"pred"],MgmtTcgaGBM27k[intersectx,"pred"],xlab="probabilities from level1",
     ylab="probabilities from level2",panel.first=c(grid()),pch=19)
abline(0,1,col="red",lwd=2)
abline(h=MGMTSTP27$perf$cut,col="darkgreen",lty=2)
abline(v=MGMTSTP27$perf$cut,col="darkgreen",lty=2)
```

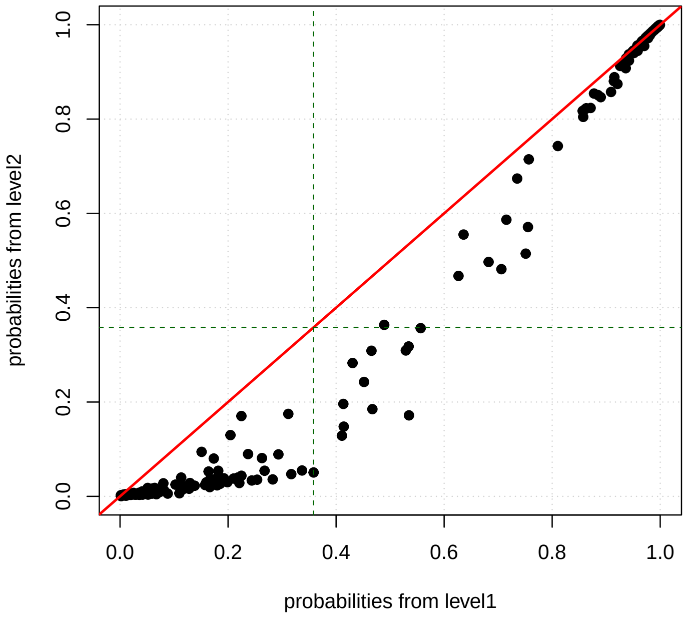

Figure 1: Comparison of the prediction from level1 and level2 from TCGA GBM database ([4]). The red line correspond to x=y and the dotted dark green lines identify the cut-off position.

.

4 Comparison between the level1 from TCGA GBM database and the Table S4 from [2]

The comparison analyses for level1 (called lumiMgmtTcgaGBM27k) and the table S4 (call TCGAgbm27) are given below:

```
# the Table S4 is available as data.frame in R package mgmtstp27
require(mgmtstp27)
data(TCGAgbm27)
head(TCGAgbm27)
```

```
             bcr_patient_barcode STP27response STP27class cg12434587 cg12981137
TCGA-19-0964        TCGA-19-0964     0.9983304          M   1.623601  1.3127016
TCGA-06-2565        TCGA-06-2565     0.9974861          M   1.120990  1.1559882
TCGA-06-0877        TCGA-06-0877     0.9085053          M  -4.514769  0.3817951
TCGA-02-0007        TCGA-02-0007     0.2789628          U  -3.414182 -3.7471048
TCGA-02-0009        TCGA-02-0009     0.5385067          M  -2.928746 -2.8317146
TCGA-02-0021        TCGA-02-0021     0.7546829          M  -2.962724 -1.7660180
```

```
 # intersection between the two datasets
 intersectx <- intersect(rownames(lumiMgmtTcgaGBM27k),rownames(TCGAgbm27))
 table(level1=lumiMgmtTcgaGBM27k[intersectx,"state"],tableS4=TCGAgbm27[intersectx,"STP27class"])
      tableS4
level1   M   U
     M 120   1
     U   0 120
 selectx <- !(lumiMgmtTcgaGBM27k[intersectx,"state"]==TCGAgbm27[intersectx,"STP27class"])
 lumiMgmtTcgaGBM27k[intersectx,][selectx,"pred"]
[1] 0.3583862
 TCGAgbm27[intersectx,][selectx,"STP27response"]
[1] 0.3542956
 r1 <- cor(TCGAgbm27[intersectx,"STP27response"],lumiMgmtTcgaGBM27k[intersectx,"pred"])
 r1
[1] 0.9999897
```

We detect only one discrepancy and it's just a problem related to number of decimal during the prediction step. The correlation between the probabilities is quasi-equal to 1 (r=0.999989664489721). The figure 2 illustrates the very strong concordance between the two sets of prediction.

```
 plot(lumiMgmtTcgaGBM27k[intersectx,"pred"],TCGAgbm27[intersectx,"STP27response"],xlab="probabilities from level
      ylab="probabilities from Table S4",panel.first=c(grid()),pch=19)
 abline(0,1,col="red",lwd=2)
 abline(h=MGMTSTP27$perf$cut,col="darkgreen",lty=2)
 abline(v=MGMTSTP27$perf$cut,col="darkgreen",lty=2)
```

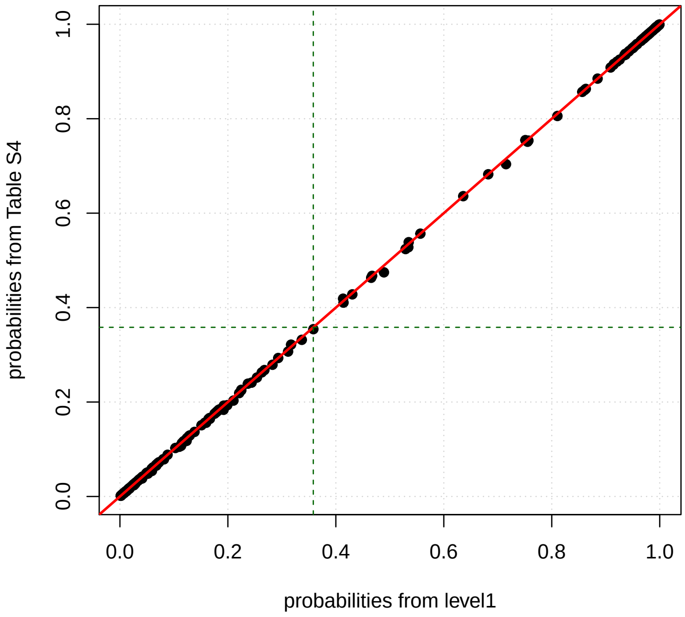

Figure 2: Comparison of the prediction from level1 from TCGA GBM database ([4]) and the Table S4 from [2]. The red line correspond to x=y and the dotted dark green lines identify the cut-off position.

.

5 Conclusion

In this document, we have shown that the level2 (2014) of the last version of TCGA GBM dataset don't correspond to the level 2 of the TCGA database used in [2]. TCGA have certainly (?) changed the procedure of the construction of the level 2 with additional of new/other methods of normalization (**or** Illumina have changed the default normalization procedure for HM-27K). However, no clear information about the data preprocessing is available in *wiki* (URL: https://wiki.nci.nih.gov/display/TCGA/DNA+methylation). Consequently, the users need to work with level 1 data (see the function methylumIDAT for data importation) to obtain unbiased prediction of DNA methylation state of MGMT promoter. To conclude, comments and instructions (as they come) related to the choice of normalization and importation of data (HM-450K or HM-27K) for using the model predicting the DNA methylation of the MGMT promoter, are given below:

- The model in Bady et al. (2012) requires to use the initial (raw) preprocessing proposed initialy by Illumina in *GenomeStudio* (2009-2011) that corresponds to the function preprocessRaw from R package minfi. For the HM-27K platform, the function contains in R package `minfi` ([1]) don't work, but it's possible to import and to preprocess data with functions from R package `methylumi` ([3]).
- The normalization can affect the prediction of the DNA methylation (it's not really a surprise).
- The generalization of the model can be affected by the new normalization proposed by Illumina (preprocessIllumina). The reference samples used during the normalization procedure were fixed within each dataset and they were not the same among the datasets.
- There are no problem/bias induced by chemistry type, because the two probes used in the model come from the chemistry I as in the HM-27K platform.

6 Acknowledgments

The results published here are in part based upon data generated by The Cancer TCGA Genome Atlas pilot project established by the NCI and NHGRI. Information about TCGA and the investigators and institutions who constitute the TCGA research network can be found at (http://cancergenome.nih.gov). The dbGaP accession number to the specific version of the TCGA data set is phs000178.v8.p7.

References

[1] 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*, 2014.

[2] Pierre Bady, Davide Sciuscio, Annie-Claire Diserens, Jocelyne Bloch, Martin J. van den Bent, Christine Marosi, Pierre-Yves Dietrich, Michael Weller, Luigi Mariani, Frank L. Heppner, David R. McDonald, Denis Lacombe, Roger Stupp, Mauro Delorenzi, and Monika E. Hegi. Mgmt methylation analysis of glioblastoma on the infinium methylation beadchip identifies two distinct cpg regions associated with gene silencing and outcome, yielding a prediction model for comparisons across datasets, tumor grades, and cimp-status. *Acta Neuropathologica*, 124(4):547–560, 2012. Times Cited: 6.

[3] Sean Davis, Pan Du, Sven Bilke, Tim Triche, Jr., and Moiz Bootwalla. *methylumi: Handle Illumina methylation data*, 2014. R package version 2.10.0.

[4] The Cancer Genome Atlas Research Network. Comprehensive genomic characterization defines human glioblastoma genes and core pathways. *Nature*, 455(7216):1061–1068, 2008. 10.1038/nature07385.

7 Appendix

7.1 Session

```
print(sessionInfo(),locale=FALSE)
```

```
R version 3.1.1 (2014-07-10)
Platform: x86_64-pc-linux-gnu (64-bit)

attached base packages:
[1] datasets  utils     parallel  stats     graphics  grDevices methods   base

other attached packages:
 [1] mgmtstp27_0.1        methylumi_2.8.0      matrixStats_0.10.0
 [4] ggplot2_1.0.0        reshape2_1.4         scales_0.2.4
 [7] lumi_2.14.2          minfi_1.8.9          bumphunter_1.2.0
[10] locfit_1.5-9.1       iterators_1.0.7      foreach_1.4.2
[13] Biostrings_2.30.1    reshape_0.8.5        lattice_0.20-29
[16] Biobase_2.22.0       MASS_7.3-35          pixmap_0.4-11
[19] ade4_1.6-2           RColorBrewer_1.0-5   fortunes_1.5-2
[22] rtracklayer_1.22.7   GenomicRanges_1.14.4 XVector_0.2.0
[25] IRanges_1.20.7       BiocGenerics_0.8.0

loaded via a namespace (and not attached):
 [1] affy_1.40.0            affyio_1.30.0          annotate_1.40.1
 [4] AnnotationDbi_1.24.0   base64_1.1             beanplot_1.2
 [7] BiocInstaller_1.12.1   biomaRt_2.18.0         bitops_1.0-6
[10] BSgenome_1.30.0        codetools_0.2-9        colorspace_1.2-4
[13] DBI_0.3.1              digest_0.6.4           doRNG_1.6
[16] genefilter_1.44.0      GenomicFeatures_1.14.5 grid_3.1.1
[19] gtable_0.1.2           illuminaio_0.4.0       itertools_0.1-3
[22] KernSmooth_2.23-13     limma_3.18.13          Matrix_1.1-4
[25] mclust_4.4             mgcv_1.8-3             multtest_2.18.0
[28] munsell_0.4.2          nleqslv_2.5            nlme_3.1-117
[31] nor1mix_1.2-0          pkgmaker_0.22          plyr_1.8.1
[34] preprocessCore_1.24.0  proto_0.3-10           Rcpp_0.11.3
[37] RCurl_1.95-4.3         registry_0.2           R.methodsS3_1.6.1
[40] rngtools_1.2.4         Rsamtools_1.14.3       RSQLite_0.11.4
[43] siggenes_1.36.0        splines_3.1.1          stats4_3.1.1
[46] stringr_0.6.2          survival_2.37-7        tools_3.1.1
[49] XML_3.98-1.1           xtable_1.7-4           zlibbioc_1.8.0
```

7.2 Import raw HM-27K data (format .IDAT)

To import raw data (format **.IDAT**), the function contains in R package `minfi` ([1]) don't work with HM-27K. However, it's possible to import data with functions from R package `methylumi` ([3]).

```
require(methylumi)
rgset0<- methylumIDAT(barcode=as.character(File.Name),idatPath=datadir)
# no normalization for HM-27k,
# see help "For HumanMethylation27 data, the function does nothing"
norm27k <- normalizeMethyLumiSet(rgset0)
u27k <- unmethylated(norm27k)
m27k <- methylated(norm27k)
mvalue0 <- log2((m27k+1)/(u27k+1))
```