Bioconductor's aCGH package

Jane Fridlyand[1] and Peter Dimitrov[2]

July 6, 2004

1. Department of Epidemiology and Biostatistics, and Comprehensive Cancer Center, University of California, San Francisco, jfridlyand@cc.ucsf.edu
2. Division of Biostatistics, University of California, Berkeley, dimitrov@stat.berkeley.edu

Contents

1 Overview **1**

2 Data **1**

3 Acknowledgements **12**

1 Overview

This document presents an overview of the `aCGH` package, which provides wide basic functions for reading, analyzing and plotting array Comparative Genomic Hybridization data (Snijders et al. (2001)). Specific example for reading data in is using output of the custom freely available programs, SPOT and SPROC (Jain et al. (2002)). These programs provide image quantification and pre-processing. Outputs of all the other image processing software need to be combined into a single file containing observed values for each clone and samples and then read in as a matrix.

2 Data

The data used in the example was generated in in lab of Dr. Fred Waldman at UCSF Comprehensive Cancer Center (Nakao et al. (2004)).Array CGH has been done on 125 colorectal fresh-frozen primary tumors and the associations with various phenotypes were analyzed. To reduce running time, only 40 samples are used in the examples.
This is where the examples start:
1. Creating aCGH object from log2.ratios and clone info files.
Each array CGH object has to contain the log2ratios reperesenting relative copy number along with the mapping information including but not limited to clone name, chromosome and kb relative to the chromosome. Optionally there may be phenotypes associated with each sample.

```
> library(aCGH)

Loading required package: cluster
Loading required package: repeated
Loading required package: rmutil
Loading required package: survival
Loading required package: multtest
Loading required package: ctest
Loading required package: sma

Attaching package 'aCGH':


        The following object(s) are masked from package:stats :

         heatmap

> datadir <- system.file("data", package = "aCGH")
> clones.info <- read.table(file = file.path(datadir, "clones.info.ex.csv"),
+     header = T, sep = "\t")
> log2.ratios <- read.table(file = file.path(datadir, "log2.ratios.ex.csv"),
+     header = T, sep = "\t")
> pheno.type <- read.table(file = file.path(datadir, "pheno.type.ex.csv"),
+     header = T, sep = "\t")
> ex.acgh <- create.aCGH(log2.ratios, clones.info, pheno.type)
```

2. Printing, summary and basic plotting (fig.1) for objects of class aCGH.

```
> data(colorectal)
> colorectal

aCGH object
Call: aCGH.read.Sprocs(sproclist[1:40], "human.clones.info.Jul03.csv",
    chrom.remove.threshold = 23)

Number of Arrays 40
Number of Clones 2031

> summary(colorectal)

aCGH object
Call: aCGH.read.Sprocs(sproclist[1:40], "human.clones.info.Jul03.csv",
    chrom.remove.threshold = 23)

Number of Arrays 40
Number of Clones 2031
Imputed data exist
HMM states assigned
```

```
samples standard deviations are computed
genomic events are assigned
phenotype exists
```

```
> plot(colorectal)
```

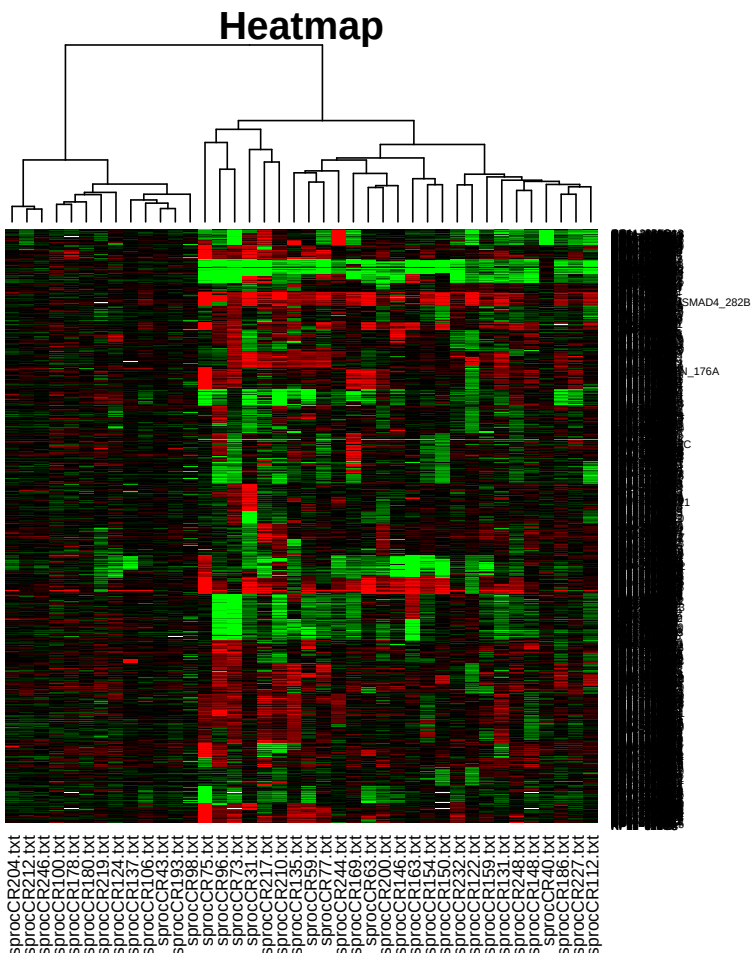

Figure 1: Heatmap of the clustered samples

```
> sample.names(colorectal)
```

```
 [1] "sprocCR31.txt"  "sprocCR40.txt"  "sprocCR43.txt"  "sprocCR59.txt"
 [5] "sprocCR63.txt"  "sprocCR73.txt"  "sprocCR75.txt"  "sprocCR77.txt"
 [9] "sprocCR96.txt"  "sprocCR98.txt"  "sprocCR100.txt" "sprocCR106.txt"
[13] "sprocCR112.txt" "sprocCR122.txt" "sprocCR124.txt" "sprocCR131.txt"
[17] "sprocCR135.txt" "sprocCR137.txt" "sprocCR146.txt" "sprocCR148.txt"
[21] "sprocCR150.txt" "sprocCR154.txt" "sprocCR159.txt" "sprocCR163.txt"
[25] "sprocCR169.txt" "sprocCR178.txt" "sprocCR180.txt" "sprocCR186.txt"
[29] "sprocCR193.txt" "sprocCR200.txt" "sprocCR204.txt" "sprocCR210.txt"
```

```
[33] "sprocCR212.txt" "sprocCR217.txt" "sprocCR219.txt" "sprocCR227.txt"
[37] "sprocCR232.txt" "sprocCR244.txt" "sprocCR246.txt" "sprocCR248.txt"
```

```
> phenotype(colorectal)[1:4, ]
```

```
  id age sex stage loc           hist diff gstm1 gstt1 nqo K12 K13 MTHFR ERCC1
1 31  70   0     1   0 Adenocarcinoma    1     0     1   1   1   2     2     1
2 40  71   0     1   1 Adenocarcinoma    1     1     1   1   2   2     2     2
3 43  59   1     1   0 Adenocarcinoma   NA     1     1   1   2   2     2     1
4 59  72   0     2   1 Adenocarcinoma    1     1     1   1   2   2     1    NA
  bat26 bat25 D5S346 D17S250 D2S123                         mi2      LOH k12
1     0     0      0       0      0           0/1 unstable loci negative   1
2     0     0      1       1      1 >2 loci unstable, (NCI def) negative   0
3     0     0      0       0      0           0/1 unstable loci negative   0
4     0     0      0       0      0           0/1 unstable loci negative   0
  K12AA k13 K13AA M677 M1298 p16 p14 mlh1 BAT26 mlh1c               mi misum
1   GTT   0     .    1     0   1   0    1     0     0 0/1 unstable loci     0
2     .   0     .    1     0   0   0    0     0     0  >2 loci unstable     3
3     .   0     .    1     0   2   0    0     0     0 0/1 unstable loci     0
4     .   0     .    0     1   0   1    0     0     0 0/1 unstable loci     0
   CGHSTAT
1 Complete
2 Complete
3 Complete
4 Not Done
```

3. Reading Sproc files

Here we demonstrate reading of the sproc files and combining them into one array CGH object. Sproc file format is specific to the custom SPROC processing software at UCSF Cancer Center.

```
> datadir <- system.file("examples", package = "aCGH")
> latest.mapping.file <- file.path(datadir, "human.clones.info.Jul03.csv")
> ex.acgh <- aCGH.read.Sprocs(system(paste("ls -1", file.path(datadir,
+     "*.txt")), intern = T), latest.mapping.file, chrom.remove.threshold = 23)
```

```
Trying to read  /usr/local/lib/R/library/aCGH/examples/sprocCR40.txt
Trying to read  /usr/local/lib/R/library/aCGH/examples/sprocCR43.txt

Averaging duplicated clones
CTB-102E19          751 752
CTB-112F7          1827 1828
CTB-142O24          1774 1775
CTB-339E12          1767 1768
CTB-36F16          1328 1329
CTD-2231J3          2066 2067
GS1-202O8          718 719
RP11-119J20          439 440
```

```
RP11-13C20          172 173
RP11-149G12          894 895
RP11-172D2          907 908
RP11-175H20          900 901
RP11-176L22          202 203
RP11-188C10          896 897
RP11-1L22          166 167
RP11-204M16          861 862
RP11-20K4          1319 1320
RP11-221P7          902 903
RP11-238H10          935 936
RP11-23G2          195 196
RP11-247E23          197 198
RP11-261B20          483 484
RP11-268N2          892 893
RP11-30M1          185 186
RP11-31B6          2027 2028
RP11-39A8          177 178
RP11-47E6          189 190
RP11-72C6          1098 1099
RP11-81L7          152 153
RP11-83O14          898 899
RP11-94M13          960 961
RP11-99M6          905 906
RP1-97B16          283 284
```

```
> ex.acgh

aCGH object
Call: aCGH.read.Sprocs(system(paste("ls -1", file.path(datadir, "*.txt")),
    intern = T), latest.mapping.file, chrom.remove.threshold = 23)

Number of Arrays 2
Number of Clones 2102
```

4. Basic heatmap plot for batch of aCGH Sproc files. (fig.2)
5. Subsetting example

```
> cr <- colorectal[, 1:3]
```

6. Basic plot for the ordered log2 ratios along the genome
The relative copy number is plotted along the genome with clones placed in the genomic order (fig. 3). Chromosome Y is excluded.
7. Plotting hmm states.
For a given sample, each chromosome is plotted on a separate page along with its smoothed values(figs. 4). The genomic events such as transitions, focal aberrations and amplifications are indicated. The outliers are also marked.

```
> plot(ex.acgh)
```

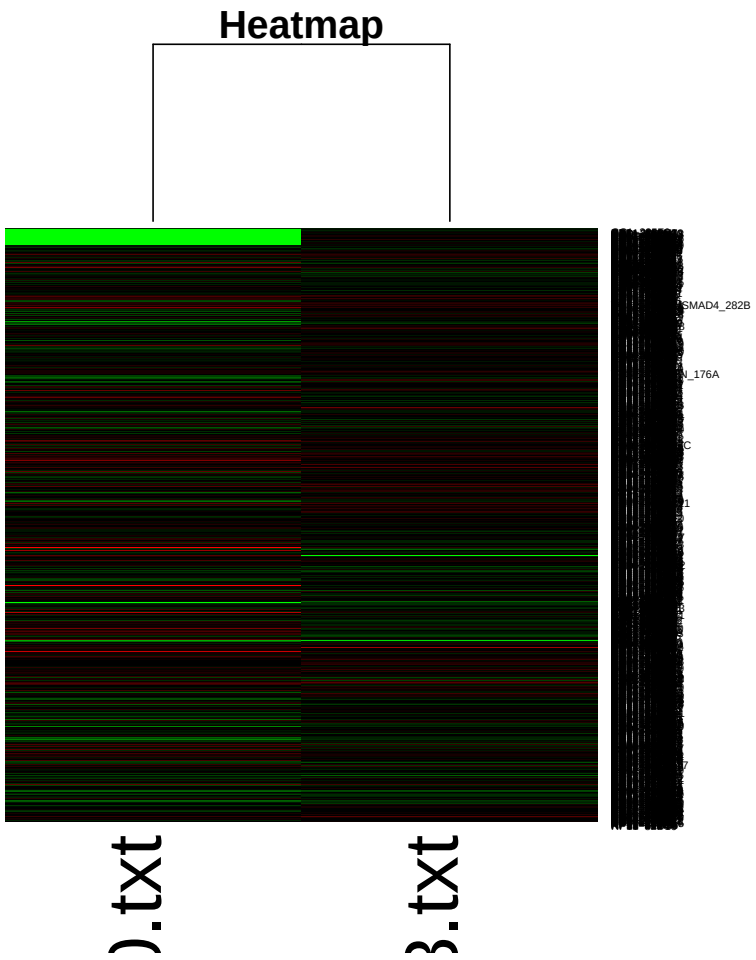

Figure 2: Basic heatmap plot for batch of aCGH Sproc files

```
> hmm(ex.acgh) <- find.hmm.states(ex.acgh)
> sd.samples(ex.acgh) <- computeSD.Samples(ex.acgh)
> genomic.events(ex.acgh) <- find.genomic.events(ex.acgh)

Finding outliers
Finding focal low level aberrations
Finding transitions
Finding focal amplifications
Processing chromosome  1
Processing chromosome  2
Processing chromosome  3
Processing chromosome  4
Processing chromosome  5
Processing chromosome  6
Processing chromosome  7
```

```
> plotGenome(ex.acgh, Y = FALSE)
```

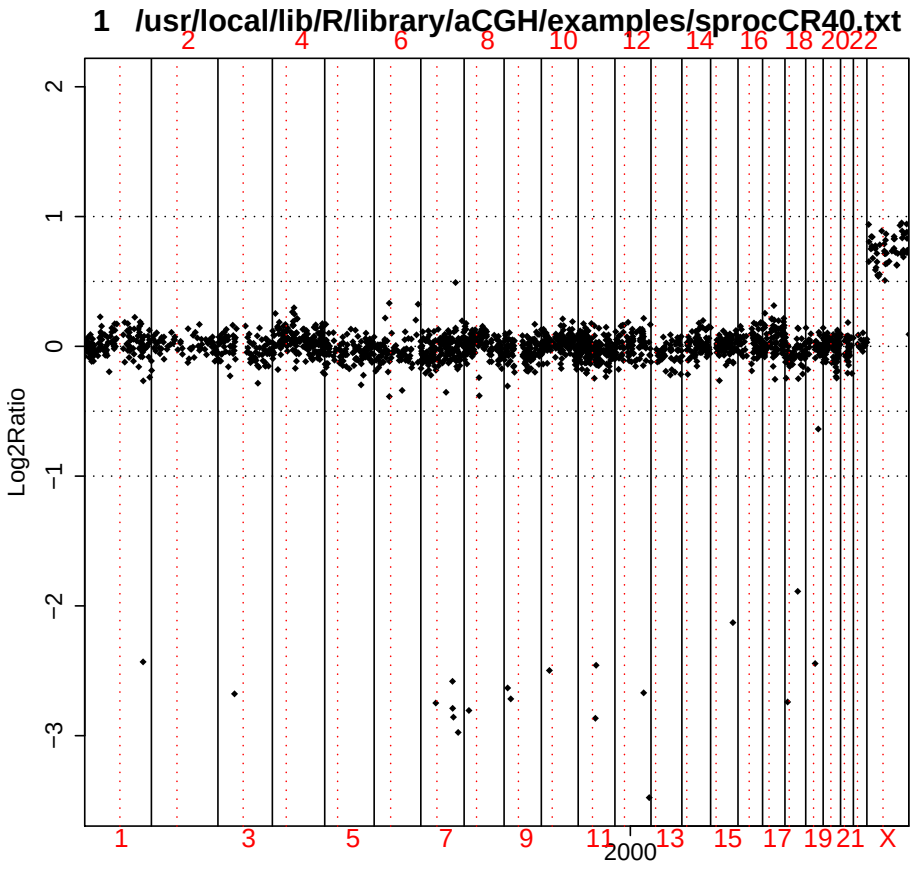

Figure 3: Basic plot for the ordered log2 ratios along the genome

```
Processing chromosome  8
Processing chromosome  9
Processing chromosome  10
Processing chromosome  11
Processing chromosome  12
Processing chromosome  13
Processing chromosome  14
Processing chromosome  15
Processing chromosome  16
Processing chromosome  17
Processing chromosome  18
Processing chromosome  19
Processing chromosome  20
Processing chromosome  21
Processing chromosome  22
```

```
Processing chromosome  23
```

```
> plotHmmStates(ex.acgh, sample.ind = 1)
```

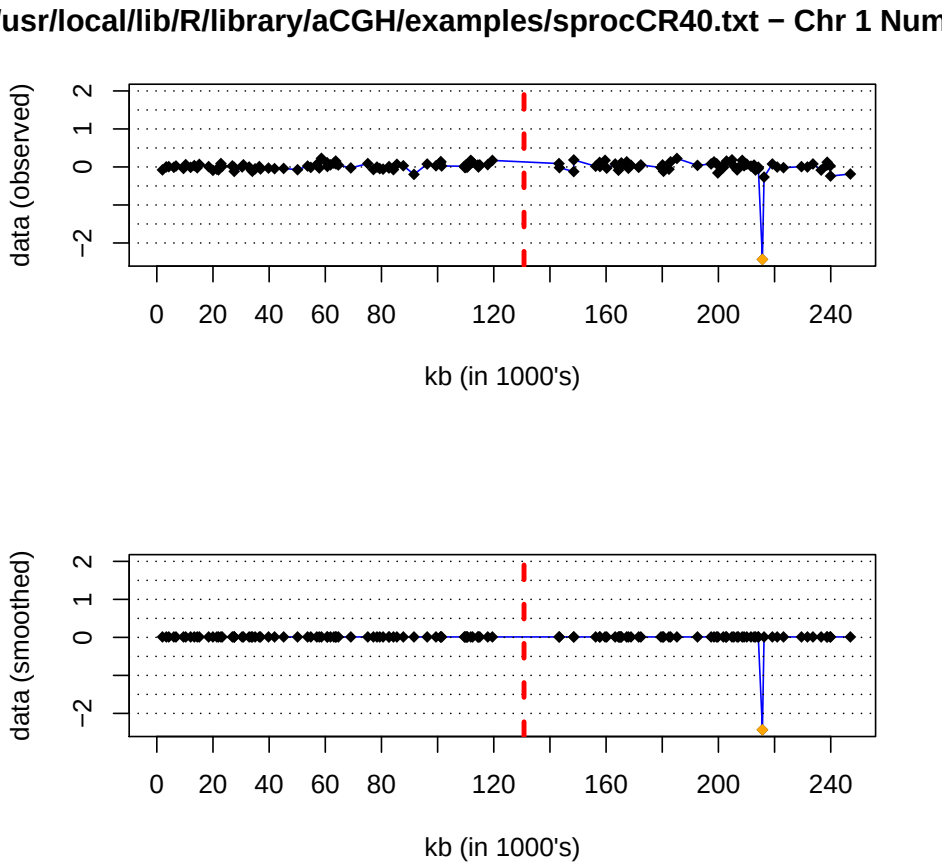

Figure 4: Plotting the hmm states found for ex.acgh object

8. Plotting summary of the tumor profiles(fig. 5). Here the distribution of various genomic events as well as their freqeuncy by location is displayed.
9. Testing association of clones with sex, followed by plotting the p.values. Use mt.maxT function from multtest package to test differences in group means for each clone grouped by sex. Plot the result along the genome displaying the frequencies of gains and losses as well well as height of the statistic correponsding to each clone(figs. 6 and 7.). The p-value can be adjusted and the horizontal lines indicate chosen level of significance.

```
> library(multtest)
> colnames(phenotype(colorectal))
```

```
 [1] "id"      "age"     "sex"     "stage"   "loc"     "hist"    "diff"
 [8] "gstm1"   "gstt1"   "nqo"     "K12"     "K13"     "MTHFR"   "ERCC1"
```

```
> plotSummaryProfile(colorectal)
```

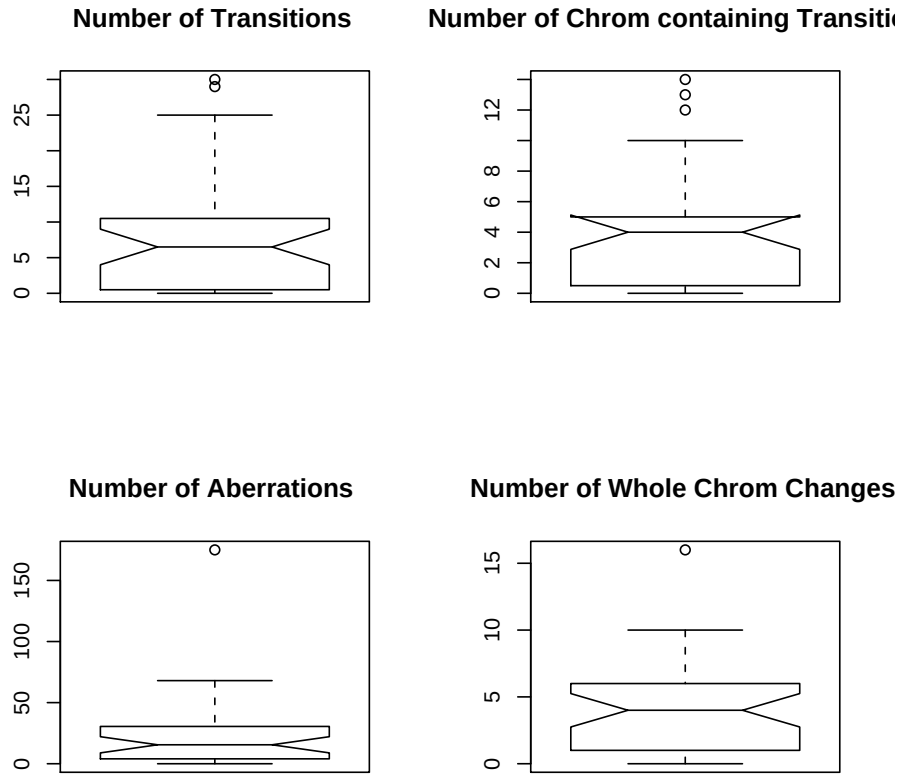

Figure 5: Plotting summary of the tumor profiles

```
[15] "bat26"   "bat25"   "D5S346"  "D17S250" "D2S123"  "mi2"     "LOH"
[22] "k12"     "K12AA"   "k13"     "K13AA"   "M677"    "M1298"   "p16"
[29] "p14"     "mlh1"    "BAT26"   "mlh1c"   "mi"      "misum"   "CGHSTAT"
```

```
> sex <- phenotype(colorectal)$sex
> sex.na <- !is.na(sex)
> colorectal.na <- colorectal[, sex.na]
> dat <- log2.ratios.imputed(colorectal.na)
> resT.sex <- mt.maxT(dat, sex[sex.na], test = "t", B = 1000)
```

```
b=10        b=20        b=30        b=40        b=50        b=60        b=70        b=80
b=110        b=120        b=130        b=140        b=150        b=160        b=170        b=
b=210        b=220        b=230        b=240        b=250        b=260        b=270        b=
b=310        b=320        b=330        b=340        b=350        b=360        b=370        b=
b=410        b=420        b=430        b=440        b=450        b=460        b=470        b=
```

```
b=510     b=520     b=530     b=540     b=550     b=560     b=570     b=
b=610     b=620     b=630     b=640     b=650     b=660     b=670     b=
b=710     b=720     b=730     b=740     b=750     b=760     b=770     b=
b=810     b=820     b=830     b=840     b=850     b=860     b=870     b=
b=910     b=920     b=930     b=940     b=950     b=960     b=970     b=
```

```
> plotFreqStat(colorectal.na, resT.sex, sex[sex.na], titles = c("Female",
+     "Male"), X = FALSE, Y = FALSE)
```

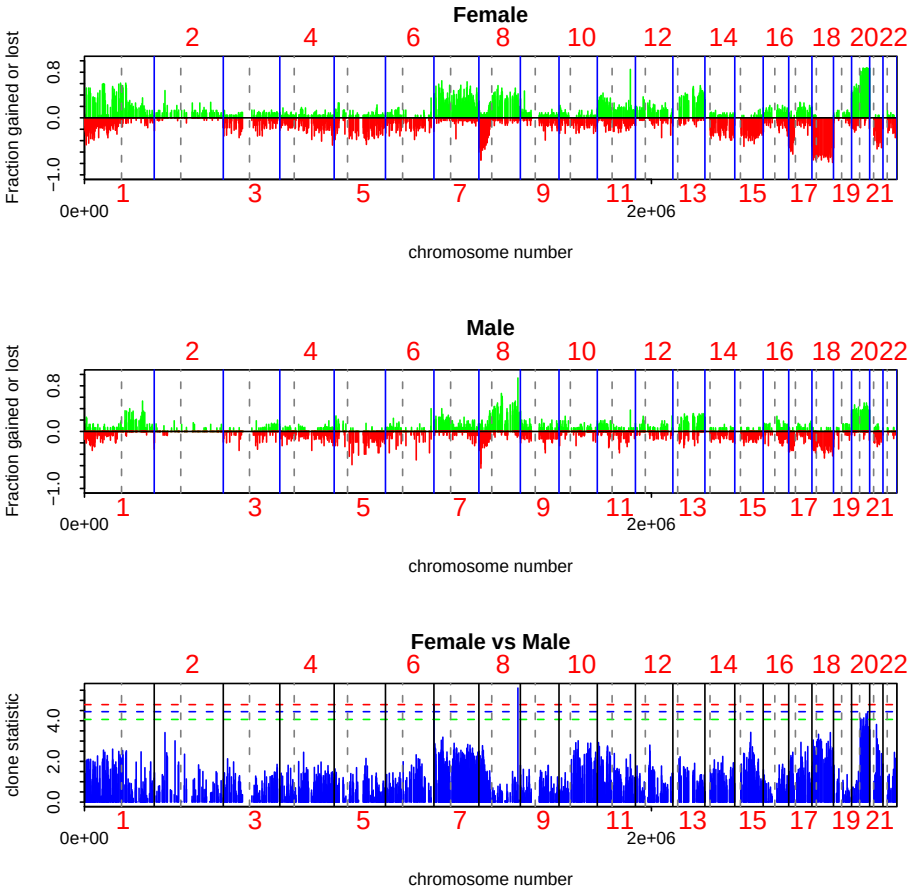

Figure 6: Frequency plots of the samples with respect to the sex groups

10. Derive statistics and p-values for testing the linear association of age with the log2 ratios of each clone along the tumors

```
> age <- phenotype(colorectal)$age
> age.na <- !is.na(age)
> colorectal.na <- colorectal[, age.na]
> dat <- log2.ratios.imputed(colorectal.na)
> stat.age <- sapply(1:nrow(dat), function(i) {
```

```
> plotSummaryProfile(colorectal, response = sex, titles = c("Female",
+     "Male"), X = FALSE, Y = FALSE, maxChrom = 22)
```

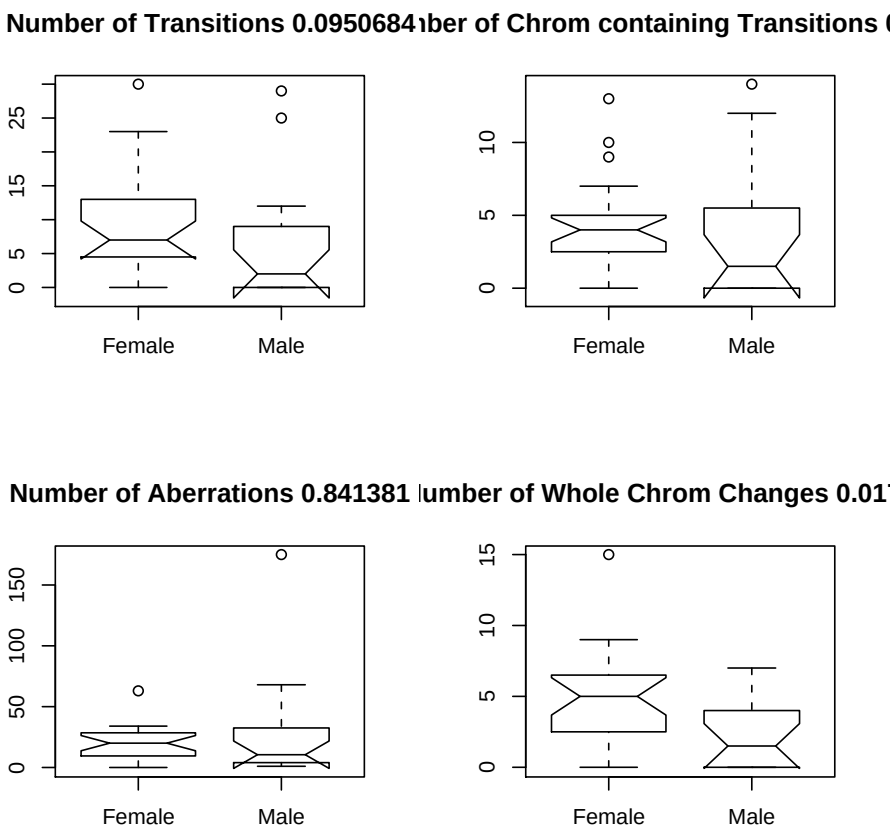

Figure 7: Plotting summary of the tumor profiles

```
+     if (i%%100 == 0)
+         cat(i, "\n")
+     lm.fit <- summary(lm(dat[i, ] ~ age[age.na]))
+     c(lm.fit$fstatistic[1], 1 - pf(lm.fit$fstatistic[1], lm.fit$fstatistic[2],
+         lm.fit$fstatistic[3]))
+ })

100
200
300
400
500
600
700
```

```
800
900
1000
1100
1200
1300
1400
1500
1600
1700
1800
1900
2000
```

11. Here we create a resT object using the values derived above.

```
> resT.age <- data.frame(index = 1:ncol(stat.age), teststat = stat.age[1,
+     ], rawp = stat.age[2, ], adjp = p.adjust(stat.age[2, ], "fdr"))
> resT.age <- resT.age[order(resT.age$adjp), ]
```

12. Here we cluster samples within each phenotype using chromosomes 4, 8 and 9 and display the phenotype labels, in this case, sex (fig. 8).

3 Acknowledgements

The authors would like to express their gratitude to Drs. Fred Waldman and Kshama Mehta for sharing the data and to Dr. Taku Tokuyasu for quantifying the images. This work would not be possible without generous support and advice of Drs. Donna Albertson, Dan Pinkel and Ajay Jain. Antoine Snijders has played an integral role in developing ideas leading to the algorithms implemented in this package.

References

A. N. Jain, T. A. Tokuyasu, A. M. Snijders, R. Segraves, D. G. Albertson, and D. Pinkel. Fully automatic quantification of microarray image data. *Genome Research*, 12:325–332, 2002.

K. Nakao, K. E. Mehta, J. Fridlyand, D. H. Moore, A. N. Jain, A. Lafuente, J. W. Wiencke, J. P. Terdiman, and F. M. Waldman. High-resolution analysis of dna copy number alterations in colorectal cancer by array-based comparative genomic hybridization. *Carcinogenesis*, 2004. Epub in March.

A. M. Snijders, N. Nowak, R. Segraves, S. Blackwood, N. Brown, J. Conroy, G. Hamilton, A. K. Hindle, B. Huey, K. Kimura, S. Law, K. Myambo, J. Palmer, B. Ylstra, J. P. Yue, J. W. Gray, A. N. Jain, D. Pinkel, and D. G. Albertson. Assembly of microarrays for genome-wide measurement of dna copy number. *Nature Genetics*, 29, November 2001.

```
> plotvalGenome.func(colorectal, response = phenotype(colorectal)$sex,
+     titles = c("Female", "Male"), byclass = TRUE, showaber = TRUE,
+     vecchrom = c(4, 8, 9), dendPlot = FALSE, imp = FALSE)
```

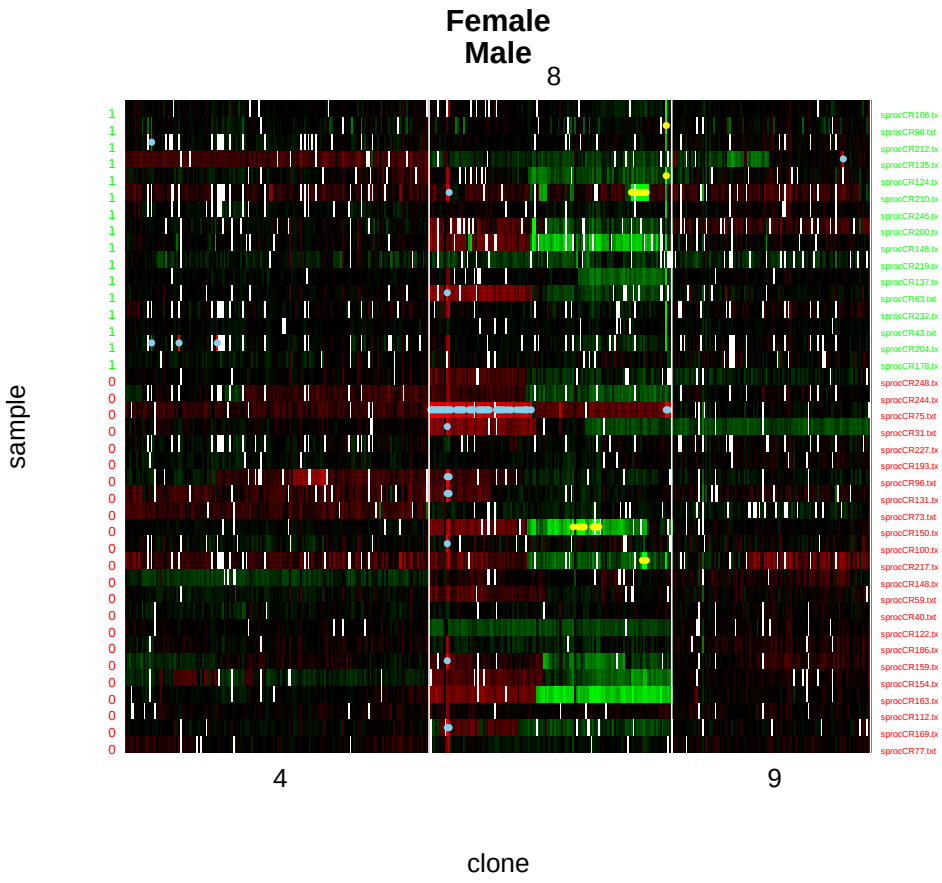

Figure 8: Clustering of the samples by sex