Bioconductor's aCGH package

Jane Fridlyand[1] and Peter Dimitrov[2]

November 23, 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 **2**

3 Examples **2**
3.1 Creating aCGH object from log2.ratios and clone info files 2
3.2 Filtering and imputation for objects of class aCGH 3
3.3 Printing, summary and basic plotting (fig. 1) for objects of class aCGH 3
3.4 Reading Sproc files 6
3.5 Basic plot for batch of aCGH Sproc files. (fig. 2) 7
3.6 Subsetting example 8
3.7 Basic plot for the ordered log2 ratios along the genome 9
3.8 Computing and plotting hmm states 10
3.9 Plotting summary of the tumor profiles(fig. **??**) 11
3.10 Overall frequency plot (fig. 5) 11
3.11 Testing association of clones with categorical, censored or continuous outcomes. 14
3.12 Clustering samples 24

4 Acknowledgements **25**

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.

3 Examples

3.1 Creating aCGH object from log2.ratios and clone info files

Each array CGH object has to contain the log2ratios representing 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: rmutil
Loading required package: survival
Loading required package: splines
Loading required package: multtest
Loading required package: sma

Attaching package 'aCGH':


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

         heatmap

> datadir <- system.file(package = "aCGH")
> datadir <- paste(datadir, "/examples", sep = "")
> clones.info <- read.table(file = file.path(datadir, "clones.info.ex.txt"),
+     header = T, sep = "\t", quote = "", comment.char = "")
> log2.ratios <- read.table(file = file.path(datadir, "log2.ratios.ex.txt"),
+     header = T, sep = "\t", quote = "", comment.char = "")
> pheno.type <- read.table(file = file.path(datadir, "pheno.type.ex.txt"),
+     header = T, sep = "\t", quote = "", comment.char = "")
> ex.acgh <- create.aCGH(log2.ratios, clones.info, pheno.type)
```

Note that when working with your own data, you will need to specify absolute path to those files ot the path relative to your working folder. For instance, if you are working in the folder *Project1* your data files are placed in the subfolder *Project1/Data*, then *datadir = "Data"* if you are using relative path.

3.2 Filtering and imputation for objects of class aCGH

Here we remove unmapped clones and clones mapping to Y chromosome, screen out clones missing in more than 25

```
> ex.acgh <- aCGH.process(ex.acgh, chrom.remove.threshold = 23,
+     prop.missing = 0.25, sample.quality.threshold = 0.4, unmapScreen = TRUE,
+     dupRemove = FALSE)
```

Here we impute missing observations using lowess approach. Note that occasionally, majority of the observations on chromosome Y may be missing causing imputing function to fail. Therefore, by default, the largest chromosome to be imputed is indexed as maxChrom=23 (X). Here we specify imputation for all chromosomes ; however, in this example there are no data on chromosome Y.

```
> log2.ratios.imputed(ex.acgh) <- impute.lowess(ex.acgh, maxChrom = 24)

Processing chromosome  1
Processing chromosome  2
Processing chromosome  3
Processing chromosome  4
Processing chromosome  5
Processing chromosome  6
Processing chromosome  7
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
```

3.3 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)
```

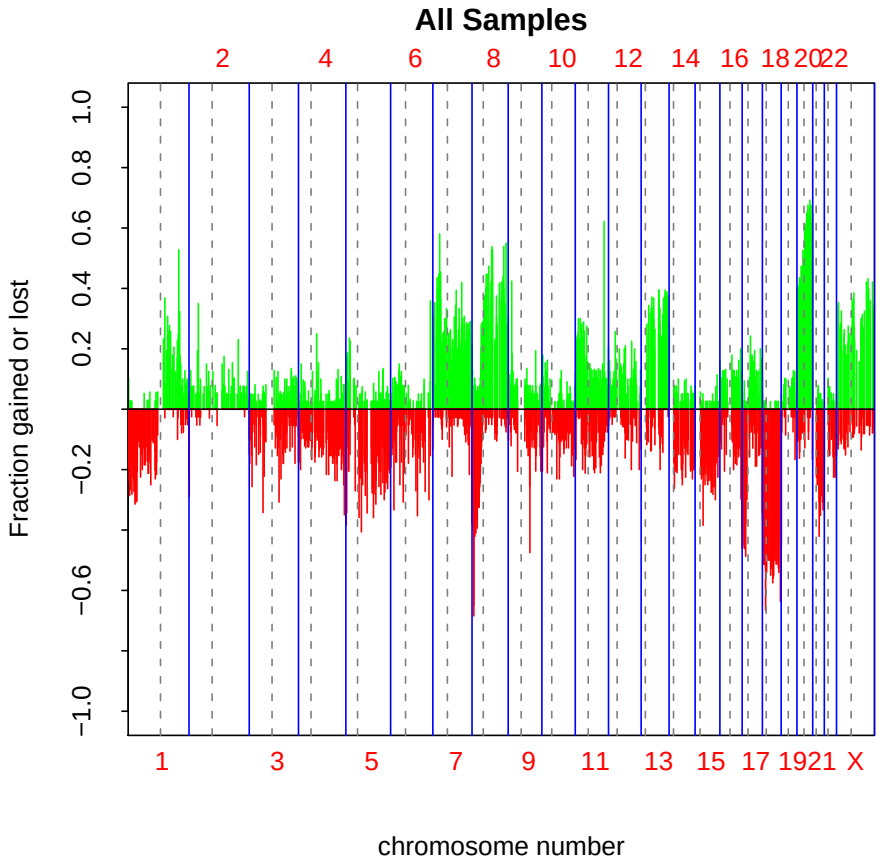

Figure 1: Basic Frequency Plot

```
> 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
```

3.4 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.txt")
> ex.acgh <- aCGH.read.Sprocs(dir(path = datadir, pattern = paste("*",
+     "*sproc*", sep = ".", full.names = TRUE), latest.mapping.file,
+     chrom.remove.threshold = 23)

Trying to read  /home/jfridlyand/R.2.0/R-2.0.0/library/aCGH/examples/sprocCR40.txt
Trying to read  /home/jfridlyand/R.2.0/R-2.0.0/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(dir(path = datadir, pattern = paste("*", "*sproc*",
    sep = "."), full.names = TRUE), latest.mapping.file, chrom.remove.threshold = 23)

Number of Arrays 2
Number of Clones 2102
```

3.5 Basic plot for batch of aCGH Sproc files. (fig. 2)

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

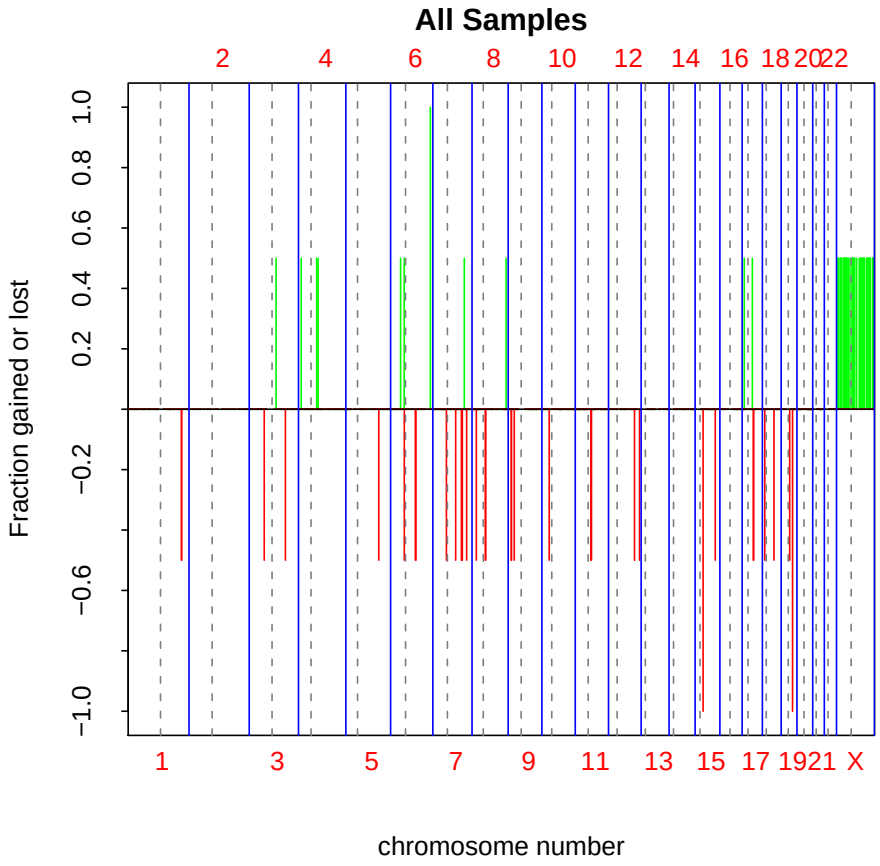

Figure 2: Basic plot for batch of aCGH Sproc files

3.6 Subsetting example

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

3.7 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. We are plotting sample 2 here. (fig. 3). Chromosome Y is explicitely excluded.

```
> plotGenome(ex.acgh, samples = 2, Y = FALSE)
```

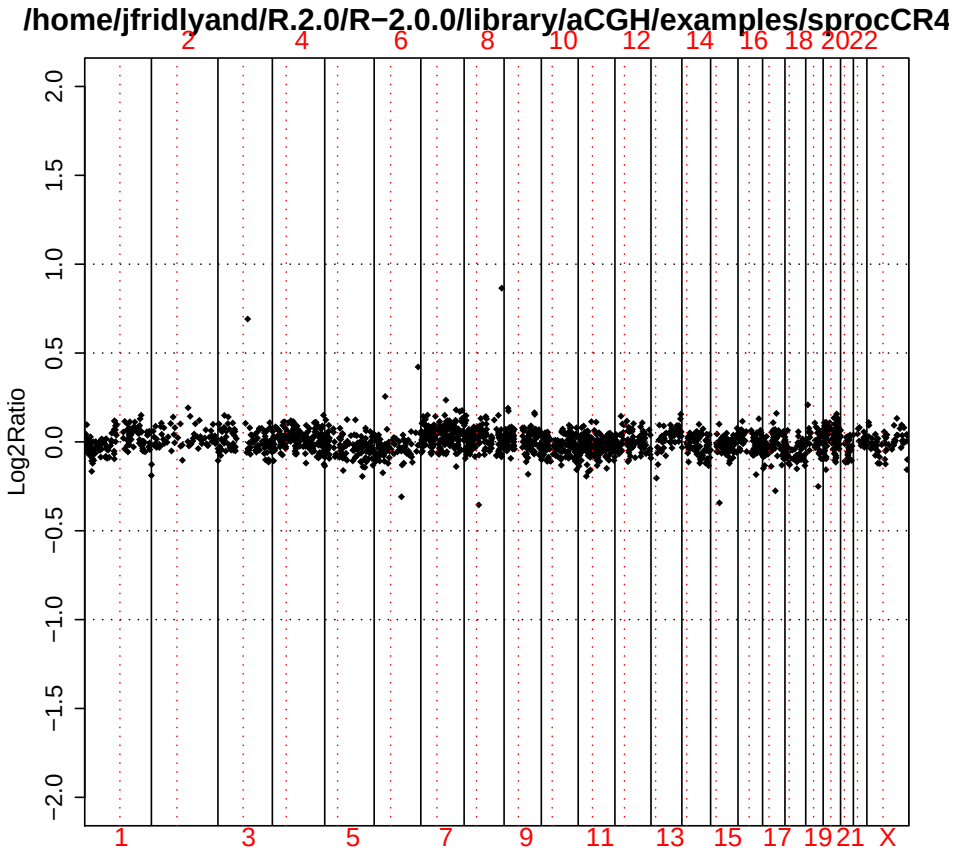

Figure 3: Basic plot for the ordered log2 ratios along the genome

3.8 Computing and plotting hmm states

Unsupervised hidden markov model is repeatedly fitted to each chromosome for varying number of states (2 , ..., 5). The number of states is determined after all fits are done usingmodel selection criterion such as AIC, BIC or delta-BIC. The model with minimal penalized negative log-likelihood is chosen for each selection criterion. Note, that some of the model fits are going to fail and are not going to be used in the final selection. Meanwhile , error message warning of the model fit failing will be printed during hmm runs. The user shoulld ignore those particular messages and related warnings.
For a given sample, each chromosome is plotted on a separate page along with its smoothed values(fig. 4). The genomic events such as transitions, focal aberrations and amplifications are indicated. The outliers are also marked.

```
> hmm(ex.acgh) <- ex.acgh.hmm
> hmm.merged(ex.acgh) <- mergeHmmStates(ex.acgh, model.use = 1,
+     minDiff = 0.25)
> 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
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(colorectal, sample.ind = 1, chr = 1)
```

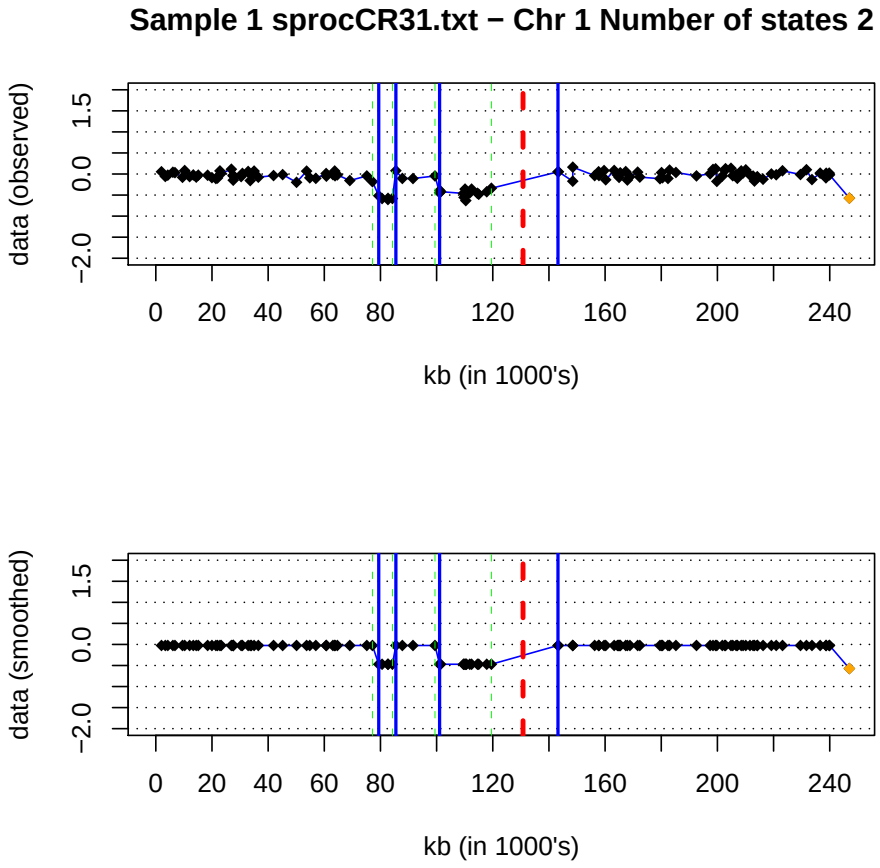

Figure 4: Plotting the hmm states found for colorectal data set.

3.9 Plotting summary of the tumor profiles(fig. ??)

Here the distribution of various genomic events as well as their frequency by location is displayed. Run the function plotSummaryProfile(colorectal) which produces multi-page figure. Necessary to write out as ps or pdf files.

3.10 Overall frequency plot (fig. 5)

```
> plotFreqStat(colorectal, all = T)
```

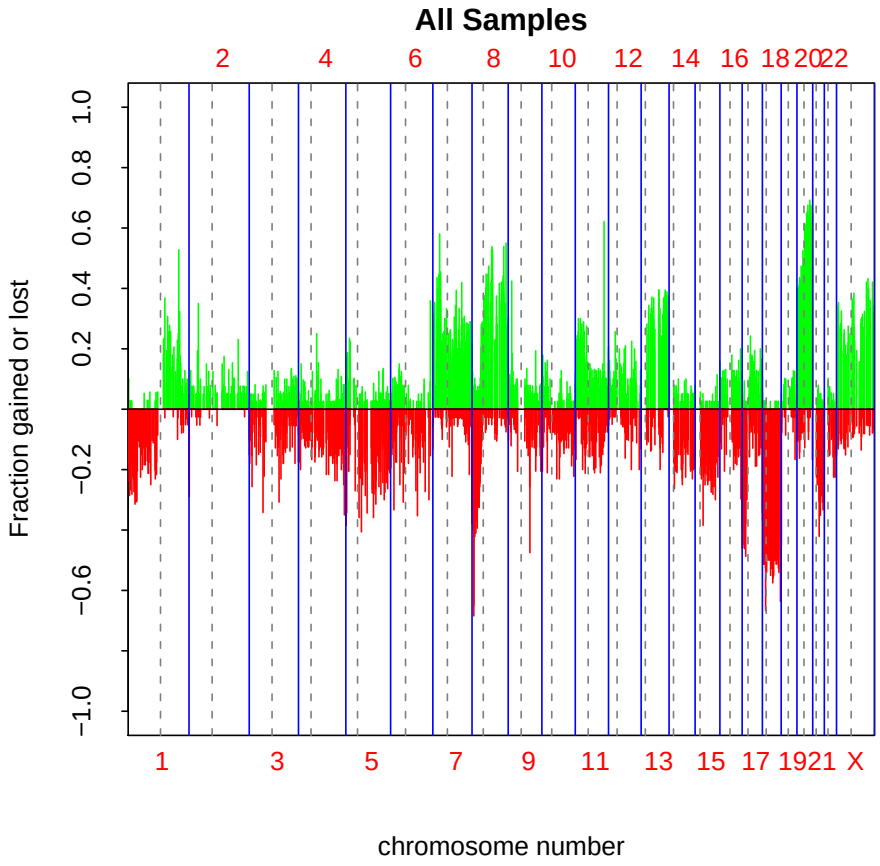

Figure 5: Overall frequency plot of the tumor profiles

summarize.clones() function is the text equivalent of plotFreqStat() - it summarizes the frequencies of changes for each clone accross tumors and includes results of statistical comparisons for each clone when available.

```
> summarize.clones(colorectal)[1:10, ]
```

	Clone	Target	Chrom	kb	NumPresent.All	NumGain.All
2	RP11-82D16	HumArray2H11_C9	1	2009	39	4
3	RP11-62M23	HumArray2H10_N30	1	3368	35	1
4	RP11-111O5	HumArray2H10_B18	1	4262	38	1
5	RP11-51B4	HumArray2H10_Q30	1	6069	35	0
6	RP11-60J11	HumArray2H10_T30	1	6817	36	1
7	RP11-813J5	HumArray2H10_B19	1	9498	30	0
8	RP11-199O1	HumArray2H10_W30	1	10284	39	1
9	RP11-188F7	HumArray2H9_C14	1	12042	36	1

```
10 RP11-178M15  HumArray2H9_F14     1 13349             35           1
11  RP11-219F4  HumArray2H9_I14     1 14391             39           1
   NumLost.All PropPresent.All PropGain.All PropLost.All
2            7            0.98         0.10         0.18
3            7            0.88         0.03         0.20
4            9            0.95         0.03         0.24
5           10            0.88         0.00         0.29
6            7            0.90         0.03         0.19
7            8            0.75         0.00         0.27
8            5            0.98         0.03         0.13
9            4            0.90         0.03         0.11
10           4            0.88         0.03         0.11
11           7            0.98         0.03         0.18
```

3.11 Testing association of clones with categorical, censored or continuous outcomes.

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.

```
> colnames(phenotype(colorectal))
```

```
 [1] "id"      "age"     "sex"     "stage"   "loc"     "hist"    "diff"
 [8] "gstm1"   "gstt1"   "nqo"     "K12"     "K13"     "MTHFR"   "ERCC1"
[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)$sex
> sex.na <- !is.na(sex)
> index.clones.use <- which(clones.info(colorectal)$Chrom < 23)
> colorectal.na <- colorectal[index.clones.use, sex.na, keep = TRUE]
> dat <- log2.ratios.imputed(colorectal.na)
> resT.sex <- mt.maxT(dat, sex[sex.na], test = "t.equalvar", 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], factor = 3,
+     titles = c("Female", "Male"), X = FALSE, Y = FALSE)
```

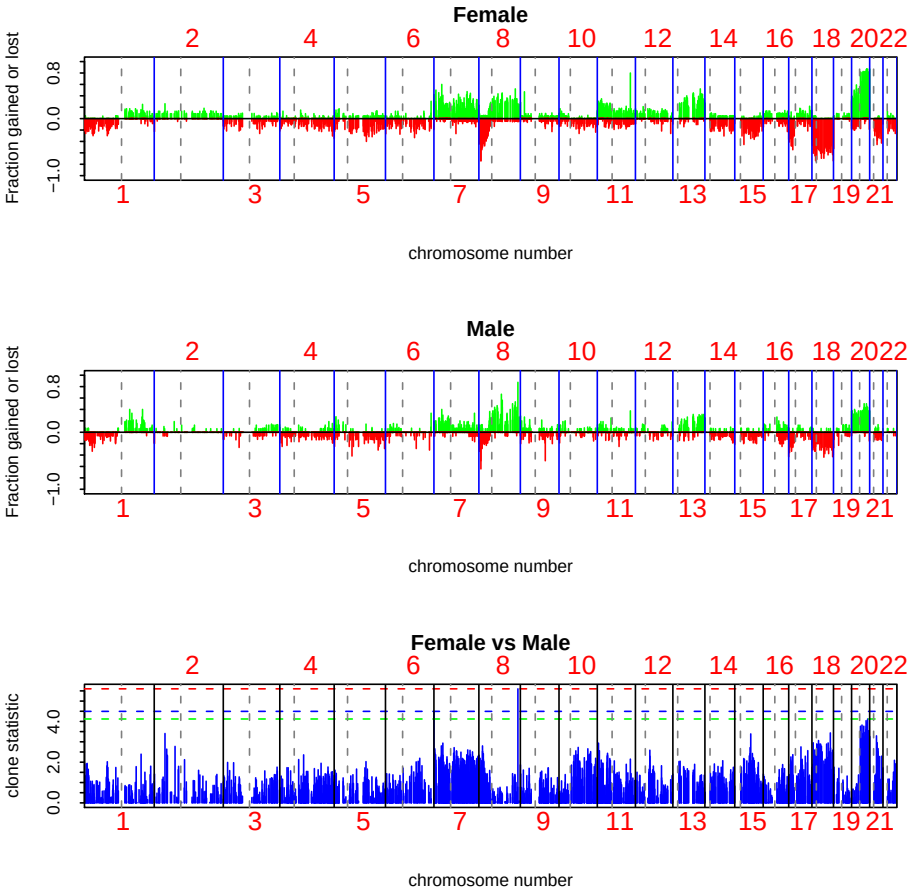

Figure 6: Frequency plots of the samples with respect to the sex groups

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

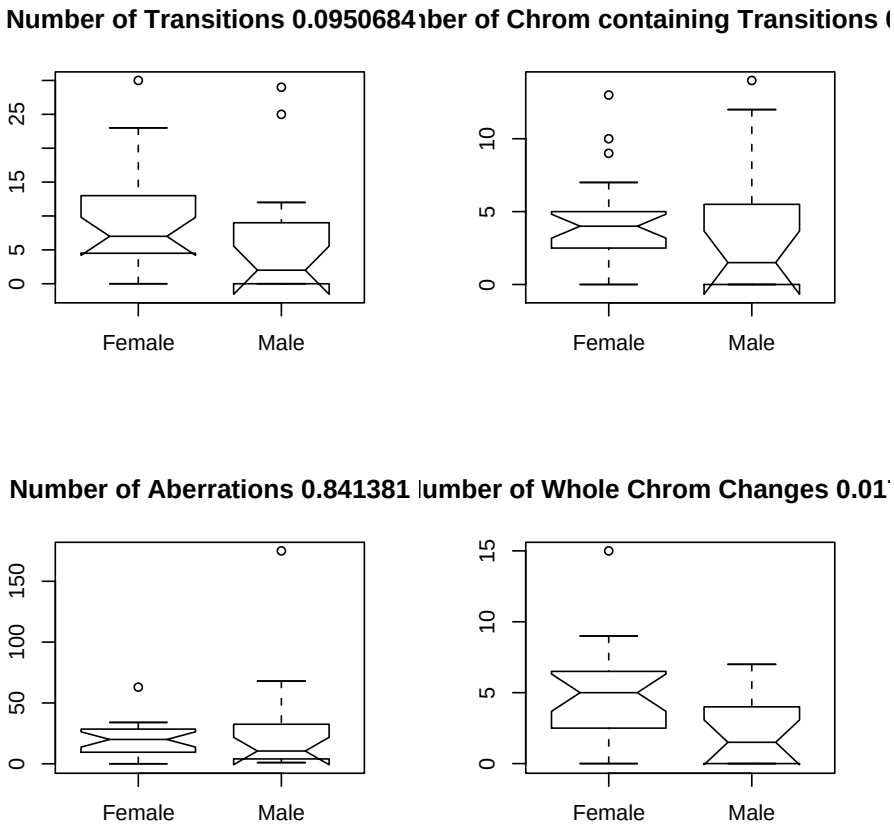

Figure 7: Plotting summary of the tumor profiles

Testing association of clones with categorical outcome for autosomal clones that are gained or lost in at least 10% of the samples. Note that the same dataset should be provided for creating *resT object and for plotting. Pay attention that HMM-related objects including sample variability do not get subsetted at the moment. Note that currently two-stage subsetting does not work for HMM slots, i.e two conditions (change and autosomal) need to be done in one iteration.*

```
> factor <- 3
> minChanged <- 0.1
> gainloss <- gainLoss(log2.ratios(colorectal)[, sex.na], cols = 1:length(which(sex.na)),
+     thres = (factor * (sd.samples(colorectal)$madGenome))[sex.na])
> ind.clones.use <- which(gainloss$gainP >= minChanged | gainloss$lossP >=
+     minChanged & clones.info(colorectal)$Chrom < 23)
> colorectal.na <- colorectal[ind.clones.use, sex.na, keep = TRUE]
> dat <- log2.ratios.imputed(colorectal.na)
> resT.sex <- mt.maxT(dat, sex[sex.na], test = "t.equalvar", 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], factor = factor,
+     titles = c("Male", "Female"))
```

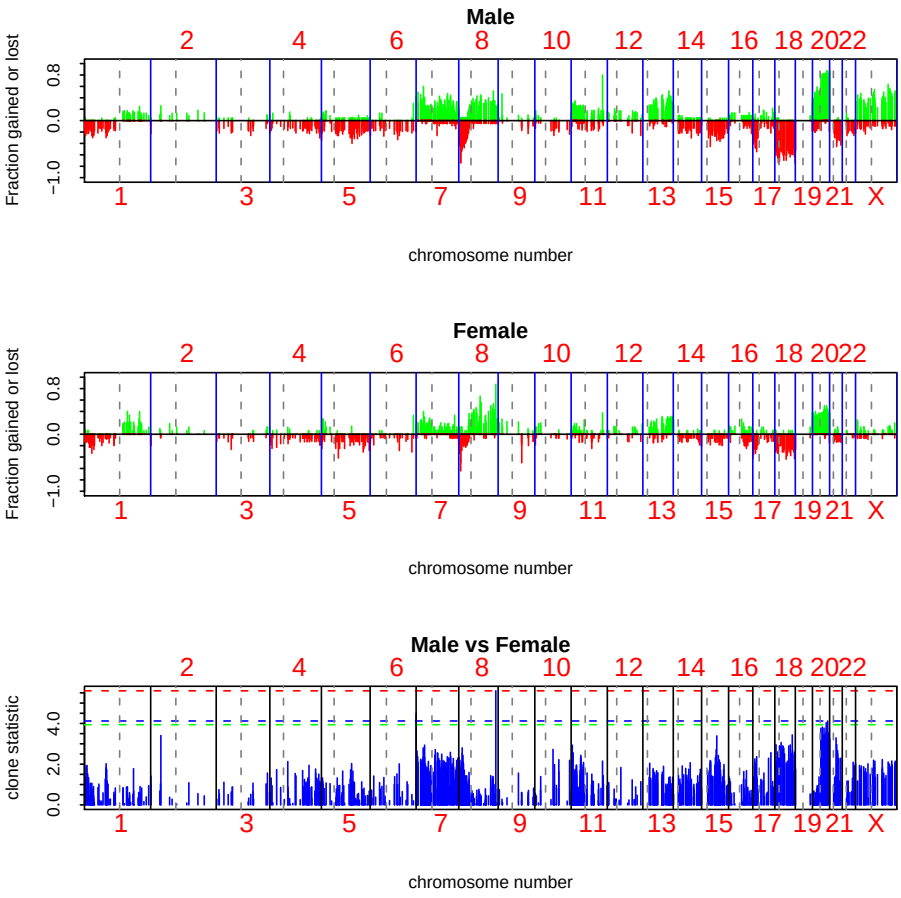

Figure 8: Frequency plots of the samples with respect to the sex groups for clones gained or lost in at least 10% of the samples

Testing association of clones with censored outcomes.Since there was no survival data available, we simulate data for a simple example to demonstrate creation and usage of basic survival object. We create an object equivalent to resT object that was created earlier. In the figure the samples are seprated into dead and alive/censored groups for ease of visualization. Nevertheless, statistic is computed and assessed for significance using proper survival object.

```
> time <- rexp(ncol(colorectal), rate = 1/12)
> events <- rbinom(ncol(colorectal), size = 1, prob = 0.5)
> surv.obj <- Surv(time, events)
> surv.obj

 [1]  0.5497257+  5.2601561   9.2027942+ 20.7781001  12.3795909  11.3925820
 [7]  7.5557326   1.3583758+  5.4146672+  1.0114897+  9.5970775+ 28.3606191
[13]  4.0165255+ 20.8023629   7.72611777 16.4934730+  6.4892360   7.8846814
[19]  1.4119679+ 14.2689133  39.0947980+ 22.2584729  16.4149341+ 15.3783722+
[25]  0.3189464+  4.4021582  21.7334862+ 22.8711022  14.0243722   0.2298490+
[31]  3.7771393+  7.0064322+ 10.2300580   5.5093099   9.4998265+ 12.6927404+
[37] 37.7226655+ 14.3638189  40.2094267+  8.7919403+

> stat.coxph <- aCGH.test(colorectal, surv.obj, test = "coxph",
+     p.adjust.method = "fdr")
> stat.coxph[1:10, ]

     index  teststat        rawp       adjp
1931  1931 -3.438029 0.000585966 0.5043327
1831  1831  3.199698 0.001375719 0.5043327
1834  1834  3.199055 0.001378788 0.5043327
1666  1666 -3.183891 0.001453097 0.5043327
390    390 -3.140434 0.001686977 0.5043327
1944  1944 -3.040147 0.002364628 0.5043327
1933  1933 -3.004409 0.002660972 0.5043327
1932  1932 -3.002839 0.002674743 0.5043327
1941  1941 -2.996101 0.002734562 0.5043327
1961  1961 -2.988994 0.002798975 0.5043327
```

```
> plotFreqStat(colorectal, stat.coxph, events, titles = c("Survived/Censored",
+     "Dead"), X = FALSE, Y = FALSE)
```

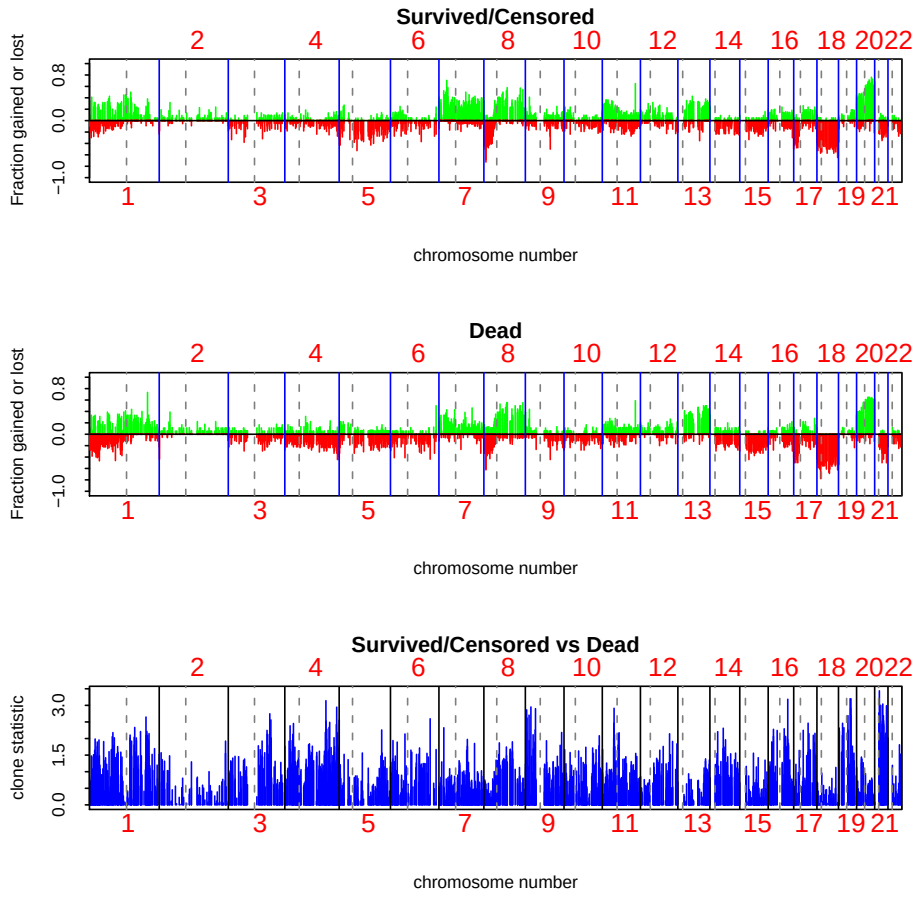

Figure 9: Frequency plots of the samples with respect to survival.

Deriving statistics and p-values for testing the linear association of age with the log2 ratios of each clone along the tumors. Here we repeat above two examples but using significance of linear regression coeffecient as a mesuare of association between genomic variable and continious outcome.

```
> age <- phenotype(colorectal)$age
> age.na <- which(!is.na(age))
> age <- age[age.na]
> colorectal.na <- colorectal[, age.na]
> stat.age <- aCGH.test(colorectal.na, age, test = "linear.regression",
+     p.adjust.method = "fdr")
> stat.age[1:10, ]
```

```
     index  teststat        rawp       adjp
1735  1735  3.259187 0.002399741 0.9952687
```

```
1739  1739  3.184326 0.002941084 0.9952687
685    685 -3.158061 0.003157117 0.9952687
1251  1251  3.144471 0.003274723 0.9952687
1718  1718  3.118281 0.003513183 0.9952687
1714  1714  3.112281 0.003570080 0.9952687
642    642 -3.082287 0.003867826 0.9952687
639    639 -3.012157 0.004658116 0.9952687
643    643 -2.937882 0.005659632 0.9952687
1744  1744  2.881404 0.006552898 0.9952687
```

```
> plotFreqStat(colorectal.na, stat.age, ifelse(age < 70, 0, 1),
+     titles = c("Young", "Old"), X = FALSE, Y = FALSE)
```

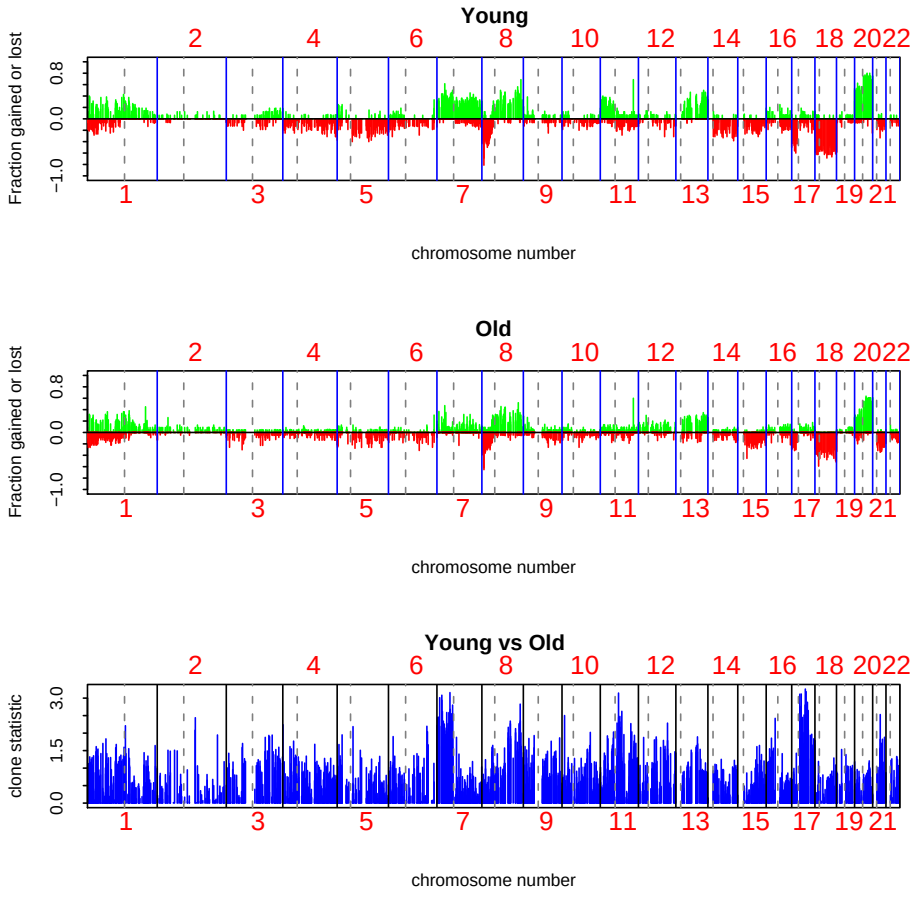

Figure 10: Frequency plots of the samples with respect to age.

Here we show example of how to create a table of results which can be later exported into other programs via write.table. *First, Males vs Females:*

```
> sex <- phenotype(colorectal)$sex
> sex.na <- !is.na(sex)
```

```
> index.clones.use <- which(clones.info(colorectal.na)$Chrom <
+     23)
> colorectal.na <- colorectal[index.clones.use, sex.na, keep = TRUE]
> dat <- log2.ratios.imputed(colorectal.na)
> resT.sex <- mt.maxT(dat, sex[sex.na], test = "t.equalvar", 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=
```

```
> sex.tbl <- summarize.clones(colorectal.na, resT.sex, sex[sex.na],
+     titles = c("Male", "Female"))
> sex.tbl[1:5, ]
```

```
       Clone           Target Chrom   kb NumPresent.All NumGain.All NumLost.All
2 RP11-82D16  HumArray2H11_C9     1 2009             38           4           7
3 RP11-62M23 HumArray2H10_N30     1 3368             34           1           7
4 RP11-111O5 HumArray2H10_B18     1 4262             37           1           9
5  RP11-51B4 HumArray2H10_Q30     1 6069             34           0          10
6 RP11-60J11 HumArray2H10_T30     1 6817             35           1           7
  PropPresent.All PropGain.All PropLost.All NumPresent.Male NumGain.Male
2            0.97         0.11         0.18              23            1
3            0.87         0.03         0.21              20            1
4            0.95         0.03         0.24              23            0
5            0.87         0.00         0.29              19            0
6            0.90         0.03         0.20              20            0
  NumLost.Male PropPresent.Male PropGain.Male PropLost.Male NumPresent.Female
2            5             1.00          0.04          0.22                15
3            5             0.87          0.05          0.25                14
4            7             1.00          0.00          0.30                14
5            7             0.83          0.00          0.37                15
6            4             0.87          0.00          0.20                15
  NumGain.Female NumLost.Female PropPresent.Female PropGain.Female
2              3              2               0.94            0.20
3              0              2               0.88            0.00
4              1              2               0.88            0.07
5              0              3               0.94            0.00
6              1              3               0.94            0.07
  PropLost.Female      stat  rawp adjp
2            0.13 1.3456684 0.185    1
```

```
3          0.14 1.2966513 0.214    1
4          0.14 0.7545065 0.445    1
5          0.20 1.9207531 0.066    1
6          0.20 0.5052960 0.640    1
```

3.12 Clustering samples

Here we cluster samples while displaying phenotypes as well as within phenotypes using chromosomes 4, 8 and 9 and display the phenotype labels, in this case, sex. We also indicate high level amplifications and 2-copy deletions with yellow and blue colors. (fig. 11).

```
> par(mfrow = c(2, 1))
> clusterGenome(colorectal.na, response = sex[sex.na], titles = c("Female",
+     "Male"), byclass = FALSE, showaber = TRUE, vecchrom = c(4,
+     8, 9), dendPlot = FALSE, imp = FALSE)
> clusterGenome(colorectal.na, response = sex[sex.na], titles = c("Female",
+     "Male"), byclass = TRUE, showaber = TRUE, vecchrom = c(4,
+     8, 9), dendPlot = FALSE, imp = FALSE)
```

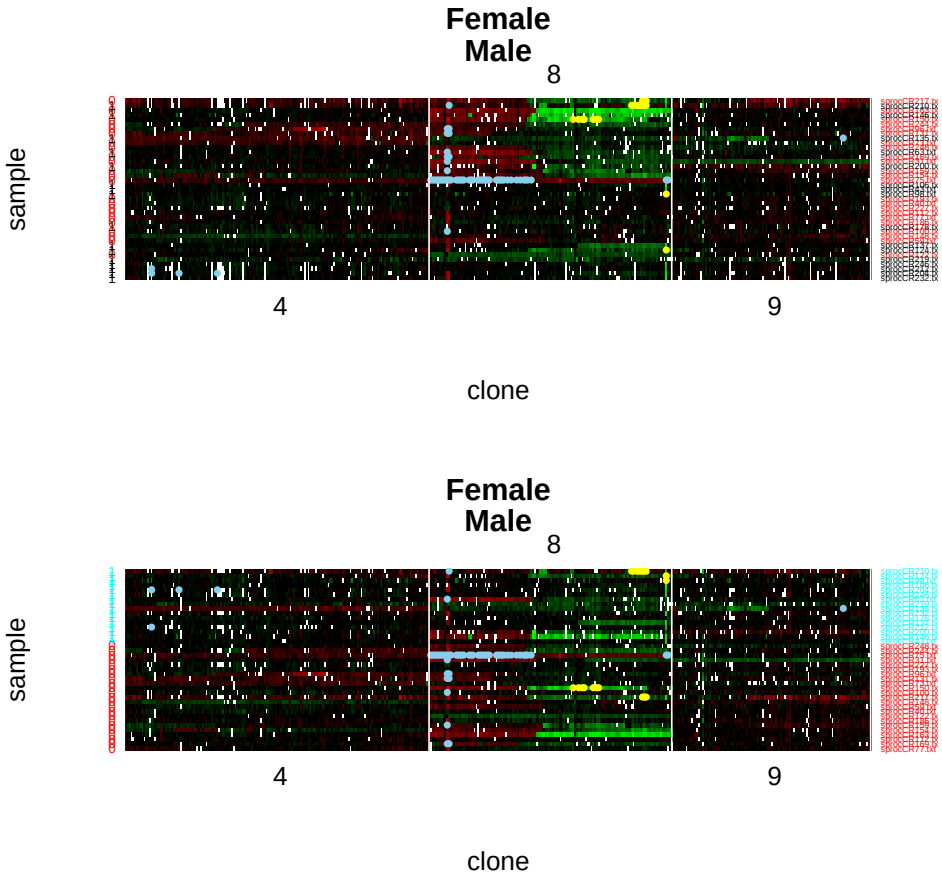

Figure 11: Clustering of the samples by sex

4 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.Many thanks to Ritu Roydasgupta for assistance in debugging.

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.*