MouseDivGeno - R package

Hyuna Yang

March 20, 2010

1 Introduction

`MouseDivGeno` is a R package specifically designed to genotype the Mouse Diversity Array (Yang., et al, 2008), an Affymetrix Mouse genotyping array equivalent to human SNP 6.0. MouseDivGeno can normalize Mouse Diversity Array, genotype, and identify probe sets potentially harboring a new mutation (variable intensity oligonucleotide or VINO, here we call vinotyping) or deletion. Normlization steps are highly customized and designed for the MouseDivGeno, and genotyping and vinotyping functions can be applied to other genotyping arrays for other species such as human, dog or horse. R package, updated annotation and further information can be obtained from the
`http://genomedynamics.org/tools/MouseDivGeno`.

2 Installation

`MouseDivGeno` was developed under the *R 2.10.0*, and it is assumed that you already install R. If not, visit `http://cran.r-project.org`.

First download the `MouseDivGeno` from the
`http://genomedynamics.org/tools/MouseDivGeno`. If you want to genotype the Mouse Diversity Array, you also need to obtain all annotation files under the
`http://genomedynamics.org/tools/MouseDivGeno/CDFfiles`. Those files are necessary for the normalization step. If you obtained normalized log2 intensities using other softwares, and want to use *MouseDivGeno* for genotyping purpose only, you do not need to obtain those annotation files. In that case, skip the section 3 and refer to section 4.

2.1 Installation - Windows(9x/NT/2000)

1. Start Rgui
2. Select Menu `Packages`, click `Install package from local zip file`. Choose the file `MouseDivGeno_*.tar.gz` and click 'OK'.

2.2 Installation - Linux/Unix

1. Go into the directory containing `MouseDivGeno_*.tar.gz`.

2. Type `R CMD INSTALL MouseDivGeno` to have the package installed in the standard location such like `/usr/lib/R/library`. You will have to be the superuser to do this. As a normal user, you can install the package in your own local directory. To do this, type `R CMD INSTALL -library=$LOCALRLIB MouseDivGeno_*.tar.gz`, where `$LOCALRLIB` is something like `/home/user/Rlib/`. Then you will need to create a file `.Renviron` in your home directory to contain the line `R_LIBS=/home/user/Rlib` so that R will know to search for packages in that directory.

You also need to install three R packages; `affyio`, `preprocessCore`, and `cluster`. Easiest way to do it would start R and type

```
R> install.packages('cluster')
R> source("http://bioconductor.org/biocLite.R")
R> biocLite("preprocessCore")
R> biocLite("affyid")
```

3 Qick Start - to genotype Mouse Diversity Genotyping Array

`MouseDivGenotype` is a R function highly customized for the Mouse Diversity Genotyping Array. It reads the .CEL files, normalized and genotype them. To learn more about each steps, refer to Section 4. To use this function, you need to install the `MouseDivGeno` R package and download all files under the
`http://genomedynamics.org/tools/MouseDivGeno/CDFfiles`. Next, you need to identify .CEL files that you want to genotype. This can be done in two ways; place all .CEL files under one directory and specify that directory at the `celfiledir`, or make a `celname` file, a tab delimited file listing all .CEL file names, and specify the name of the celname file at the `celfilename` and directory location at the at the `celfiledir`. If you do not specify the `celname` file at the `celfilename`, it reads all .CEL files under the `celfiledir`, and if `celfiledir` is not specified, it reads all .CEL files under the current working directory. Table 1 shows one example of `celname` file. The first column of the file must list .CEL file names, and the column header must be 'celfile'. Optionally gender information can be specified at the second column with a column header 'geneder'. Gender can be identifyed as 'female' or 'male' (case sensitive), and anything else is considered as un-konwn gender. If you do not specify the gender or gender column has at least one un-known gender, the software will compute the gender based on X chromosome and Y chromosome intensities.

celfile	gender
ex1.CEL	female
ex2.CEL	male
ex3.CEL	male
ex4.CEL	unknown
...	...

Table 1: Example of celname file, saved as 'filenames.txt'

You need to place annotation files `allid`, `ABid`, and `chrid` at a directory that you can load them. `allid` contains all indexes corresponding to SNP probes in the .CEL file. `ABid` contains indexes corresponding to A or B allele, and `chrid` contains indexes corresponding to each chromosome. `CGFLcorrection` and `reference` files are optional, and if they are specified, they will be used to normalized the Mouse Diversity Array. `CGFLcorrection` is to correct intensity variation due to C or G contents in 25mers and restriction enzyime fragment size, and containins a set of coefficients based on spline regression model fitting. `reference` contains one reference distribution used for the quantile normalization step. Note that all these files are specifically designed for the Mouse Diversity Genotyping Array, and if you want to genotype other types of array, refer to section 4.

MouseDivGeno genotypes the SNPs based on contrast and summation dimension, and the function offers two different transformations to obtain contrast; CCS (contrast centers stretch) transformation proposed by BRLMM-P algorithm obtains contrast via asinh(K*(A-B)/(A+B))/asinh(K), where A and B is intensity of A and B allele, K is hyperparameter, and MA transformation obtains contrast by log2(A) - log2(B). In both cases summation is defined by (log2(A) + log2(B))/2

User can identify chromosomes that you want to genotype using `mchr`. Current default is `mchr = c(1:19, 'X', 'Y', 'M')`. MouseDivGeno also offers option `subset`, and if `subset` is TRUE, it only genotype 'good' probe sets. Those 'good' probe sets were defined based on previous experience, and may sensitive lab to lab variation. Current 'good' probe sets are trained based on .CEL files processed at the Jackson Laboratory. Current default if FALSE.

```
R> library(MouseDivGeno)
R> load('allid'); load('ABid'); load('chrid');
R> load('CGFLcorrection'); load('reference');
R> celfiledir = 'C://genotype/celfile'
R> outfiledir = 'C://genotype/outfile'
R> # if you want to genotype all .CEL files specified at the 'filenames.txt'
# based on MA transformation with C+G or fragment length correction
# and quantile normalization
R> MouseDivGenotype(celfiledir, outfiledir, allid, ABid, chrid,
   CGFLcorrection=CGFLcorrection, reference=reference,
  trans="MAtrans", celnamefile = 'filenames.txt' , mchr = c(1:19), subset = FALSE)
R> # if you want to genotype all .CEL files under the celfiledir based on
# CCS transformation with no C+G and fragment length, and quantile normalization
R> MouseDivGenotype(celfiledir, outfiledir, allid, ABid, chrid, CGFLcorrection=NULL,
  reference=NULL, trans=c("CCStrans") )
```

This will return the normalized intensities, genotype, vinotype, and confidence score.

4 Genotype - General steps

4.1 Quality check and normalization

Before genotype the data, it is always recommended to check the quality of data. The detail steps are designed for the Mouse Diversity Array, and the general ideas can be applied to other

types of array.

`imageplot` returns log2 intensity heatmap (Fig 1.A) and it is a good way to check if the array has a spatial distribution. Since we put four probes (two from sense and two from antisense strand) per probe set and those probes are randomly located across .CEL file, even if there is a spatially dimmer region, it does not affect to the overall intensity due to median summarization step. However if some .CEL files show unusually big dark spot, or overall dark image, check the array processing steps including reagent, scanner, etc. To use `imageplot`, you need to specify one .CEL filename that you want to draw impageplot, and provide plot name. If the plotname is not provided, the default is 'imageplot.jpg'

```
R> filename = '~/projects/cel/SNPex1.CEL'
R> densityplot(filename, plotname = 'myimageplot.jpg')
```

`densityplot` draw SNP density. Unlike `imageplot` which can be used only one .CEL file, `densityplot` can draw density plots of many .CEL files at the same time. Thus this will allow users to compare density of many .CEL files. This will guide you further normalization especially for quantile normalization. If the plotname is not provided, the default is 'density.jpg'

```
R> setwd(celfiledir)
R> filenames = c('ex1.CEL', 'ex2.CEL', 'ex3.CEL')
R> densityplot(filenames)
R> # or use filename.txt file
R> filenames = read.table('filenames.txt', header = TRUE)
R> filenames = filenames[,1]
R> densityplot(filenames, plotname ='mydensity.jpg')
```

4.2 Normalization

MouseDivGeno offers three steps of normalization: intensity variation due to C or G contents in probe sequences and restriction enzyme length correction, quantile normalization based on a reference distribution, and median summarization. Each probes of Mouse Diversity Array has different restriction enzyme fragment length and C or G contents, and it affects the intensity. To adjust those difference, we initally choose 350 .CEL files, fit a spline regression, and obtained the coefficients in each probes. These coefficitents were saved at the MouseDivGeno pacakge, and is used to normalize a new array. Quantile normalization is commonly used in microarray data to remove array specific noise, and reference distribution is often derived from the arrays that you study. However obtaining a reference distribution at each time introduces unnecessary batch effect (i.e. with which .CEL files it normliazed together, the reference distribution thus intensities will change), so we derive one reference distribution using the same 350 CEL files, and save the reference distribution to the MouseDivGeno package. Note that the quantile normalization can only be applied when samples have the same underlying distribution. If there is sample whose intensity distribution is different from that of classical inbred strains (such as C57BL/6J) then one should not use quantile normalization. `densityplot` can be a useful tool to check the intensity distribution. After these normalization steps, it summarizes intensities from probes to a probe set value. We compared intensities from sense and antisense and removed one strand extremly poorly performing strand. We removed 57,066 probes from sense strand and 61,196 probes from antisense strand. As a result for most probe set, it

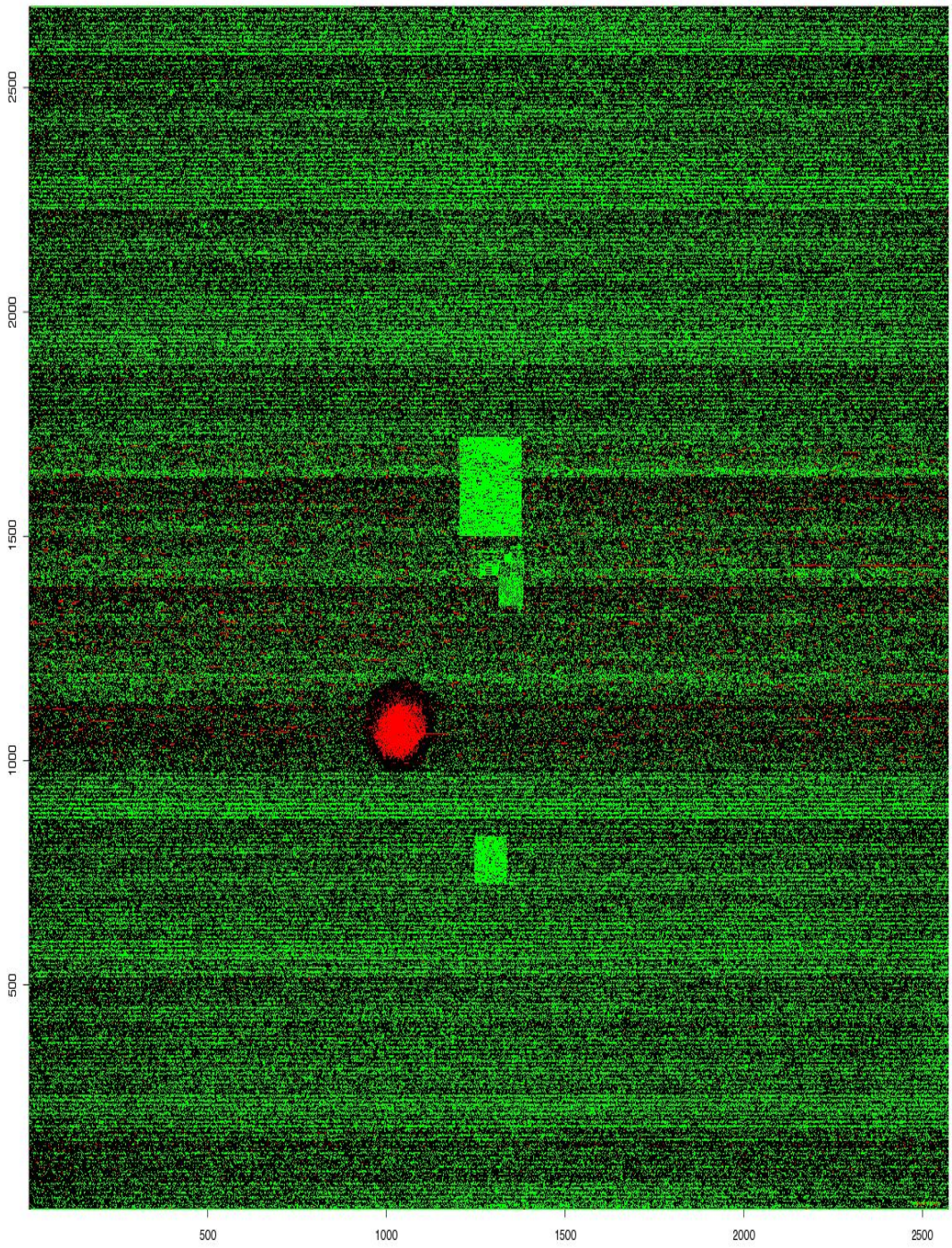

Figure 1: Example of `imageplot`.

summarizes four probe values, but for 118,262 probe sets, it only summarize 2 probes. Note that this does not affect the number of probe set itself. Transformation and subset options are the same as introduced at the Section 2. The example to obtain normalized intensities is following.

```
R> library(MouseDivGeno)
R> load('allid'); load('ABid'); load('chrid');
R> load('CGFLcorrection'); load('reference');
R> celfiledir = 'C://genotype/celfile'
R> outfiledir = 'C://genotype/outfile'
R> ReadCelFile(celfiledir, outfiledir, allid, ABid, chrid,
   CGFLcorrection=CGFLcorrection, reference=reference,
  trans="MAtrans", celnamefile = 'filenames.txt' , subset = FALSE)
```

This will save the normalized intensities under the `outfiledir`.

4.3 Genotype

Genotype is based on a method combining EM based clustering and single linkage hierachical clustering. Detail algorithm is following.

1. test two groups (N = 2)
   (a) Find center via EM based clustering using contrast.
      i. Initialization : μ_1 = maximum of contrast, μ_2 = minimum of contrast, $\sigma_1^2 = \sigma_2^2 = 0.1$
      ii. E step : calculate $P(j|i) = exp(-(x_i - \mu_j)^2/(2 * \sigma_j^2)$, where $j = 1, 2$, $i = 1, \cdots, n$ (n = number of samples)
      iii. M step : $\mu_j = \frac{\sum w_j y_i}{\sum w_j}$, $\sigma_j = \frac{\sum w_j * (y - \mu_j)^2}{\sum w_j}$, and $w_j = 1/N \sum p(j|i)$
   (b) Assign inital genotype. This step assigns genotype only for the samples having high probability $P(j|i)$. However, when the w_j is severely unbalanced, it often fails to assign at least one member to each group, so this initial genotype step also tries to assign at least one sample to each genotype.
      i. For the group having the higher w_j (let's call this group $j1$, and the other group $j2$) : Threshold = median of $P(j1|i)$ only using sample i whose $P(j2|i)$ <0.5. Assign genotype $j1$ to samples i if $P(j1|i)$ is bigger than threshold.
      ii. Assign genotype for group $j2$: Threshold = find biggest mode of $P(j2|i)$ only using i does not get assigned from the previous step. Also find median of $P(j2|i)$ only using i whose $P(j1|i)$ <0.5. Threshold is maximum of two values. Assign genotype $j2$ to i if $P(j2|i)$ is bigger than the threshold.
   (c) Genotype remaining samples using single linkage hierarchical clustering
      i. Find one unassigned sample having the smallest distance to assignment sample.
      ii. Assign the unassigned sample to the same genotype which the assigned sample belongs to.

iii. Repeat this procedure till every sample has genotype.

2 test N = 3 : same as N = 2 except now the initialization : μ_1 = maximum of contrast, μ_2 = obtain from hint file or 0, and μ_3 = minimum of contrast,

3 Finalize the genotype. Compare N = 3 vs. N = 2 using silhouette score and distance between 90th quantile of the previous genotype group and 10th quantile of the next genotype group. If N = 3 fails, it compares N = 2 vs. N = 1.

4.4 VINOtype

When probe sequence contains new mutation, hybridazation fails and it reduces the average intensity. Depending on the nature of new mutation and the genotype of target SNP, identifying some VINO is easier than the other. For instance when an inbred mouse has a new mutation right next to the target SNP, the hybridization failure gets noticable, thus easy to detect. One the other hand, if the new mutation occurs at the end of the probe sequence, the hybridization failure hardley notecialb, thus difficult to detect. VINOtyping is based on low intensities and more detail algorithm is following.

To find VINO (variant intensity oligonecleotide), MouseDivGeno calculates product of two probabilities. P(data is not a member of AA, AB, and BB) = 1 - P (data is a member of AA, AB or BB), and P(the intensity is low). P (data is a member of AA, AB or BB) is calculated by mohalanobis distance, and also served as a confidence score. Note that when there are many VINO it affects the mean and variance of each group, and to avoid this, first we remove outliers and secondary cluster at the average intensity dimension, if they exists. P(the intensity is low) is based on average intensity demension only and derived by normal distribution. Again we removed the outliers and secondary cluster based on each genotype, then merge data from all genotypes to obtain the mean and variance of average intensity. Using stringent threshold, it identifies samples having extremely low intensities and using single linkage based hierachical clustering, it identifies samples cluster to the one having extremely low intensities.

When MouseDivGeno genotype a probe set as VION, it implies two things. First, the genotype of the original target SNPs should be considered as no call. It is because the observed intensities reflect the dynamics between the nature of new mutation and original genotype, and both information is hidden thus it is not obvious to predict the original SNP genotype. Second, VINO indicates the hybridization fails in that probe seqeunce. Naturally, if great number of consecutive probes fail to hybridized, it indicates a deletion. Thus vinotyping along with simple HMM can be used to detect deletion.

4.5 Sexchromosome

When the gender information is not given, MouseDivGeno compute the gender based on Y and X chromosome intensities, and X and Y chromosome genotyping are done based on gender. Pseudoautosomal region at the end of X chromosomes are treated seperately.

5 Summary the result

: After the genotyping is done, you can find summary.txt file at the outfiledir directiory. The file contains basic summary statistics such as call rate, heterozygosity rate, VINO rate, and computed gender information.

6 citation

7 acknowlegement

8 reference