The Guide of waviCGH

September 27, 2010

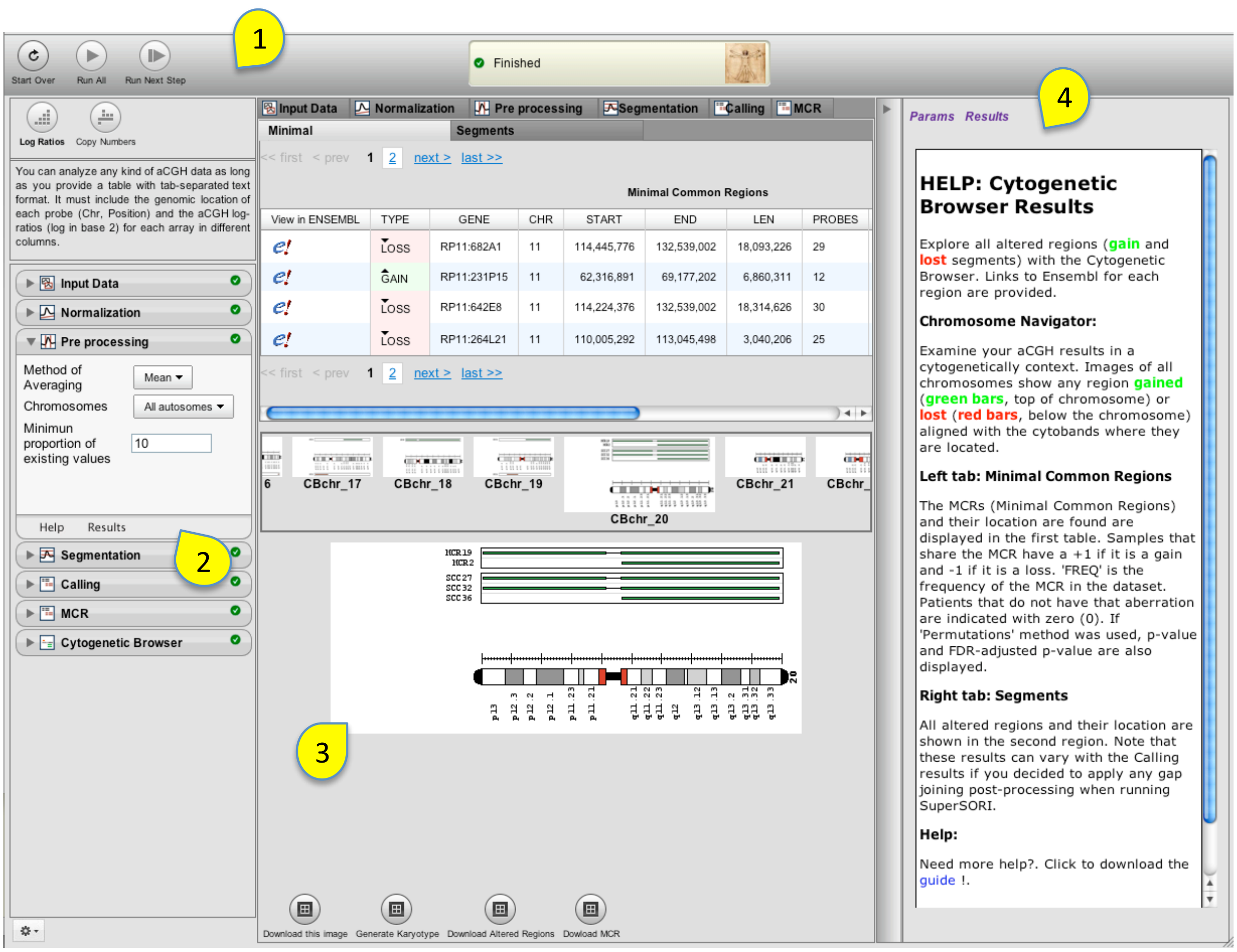
Angel Carro, Daniel Rico, Oscar M. Rueda, Ramón Díaz-Uriarte and David G. Pisano

CNIO - Centro Nacional de Investigaciones Oncológicas

Introduction

waviCGH is a versatile web-server application for the analysis and visualization of array-CGH (aCGH) data. The aim of this server is to provide easy access to the most usual computational methods in the analysis of aCGH data. We have also tried to guide researchers through a rational protocol, dividing the aCGH analysis in different steps.

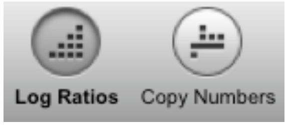
All steps are easily set by the expandable boxes at the left of the screen (Figure 1) which can be opened by clicking the triangles. The users can select the options at each step and run all steps in one go or run them step-by-step (see below). Each box contains also a "Help" button that opens waviCGH help for that step settings and a "Results" button that displays the results of that step in the main window (when they are ready) and the help for the step output.



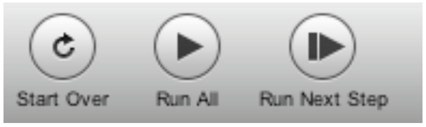
waviCGH user interface. The top bar is the Control Panel (1). Users can select the analysis protocol (depending on their input data type), run the complete protocol on one go, or step-by-step and see the state of the analysis on the central information screen. The parameters for the methods are in the Protocol Sidebar at the left (2), and are organized in expandable boxes, one for each of analysis step in the selected protocol. The main central frame is the Results Window (3). Results for each step appear in different tabs, so the results of the complete analysis workflow are easily traceable. Finally, the frame on the right contains Help about the parameters of each step and the interpretation of the results (4). The help frame can be opened or hidden at any time.

Protocols and Control Panel

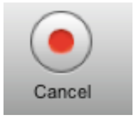
waviCGH is has been built as a logical multi-step but simple protocol to facilitate the usability of the most common aCGH analytical procedures. A protocol, or workflow, is simply a series of analysis steps. The number of analysis steps in the protocol depends on how data has been processed before using waviCGH. There are two protocol types: log-ratios and copy number data. Copy numbers are states represented as -1/1/0 (loss/gain/no-change) .Depending on your input data type, you can select the protocol you need by clicking on the corresponding button on the Control Panel:



We recommend beginners to run protocols step-by-step, clicking each time "Run Next Step", to examine and try to understand the results of each step before going to the next. Advanced users may already know all options they like so they have the possibility of running the complete workflow by clicking "Run All".



To run a new analysis from the beginning, click "Start Over". When a step is running a new button appears in case you want to cancel the process running:

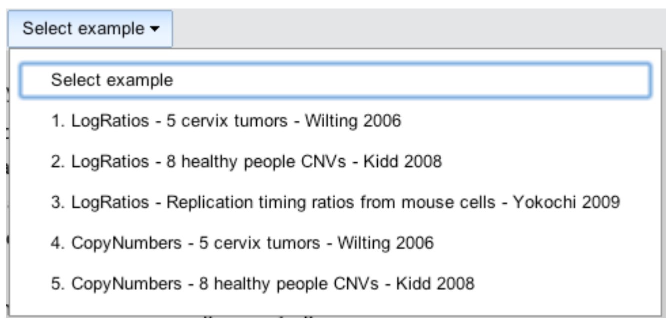


Projects

waviCGH analyses are organized in so-called projects. When a protocol is selected and the input data is uploaded into waviCGH, a new project is created and the URL of the project is provided. The results obtained in a project can be accessed for 5 days.

Examples

We recommend to run the example projects to see what waviCGH can do. You can load the example projects directly from waviCGH home page:

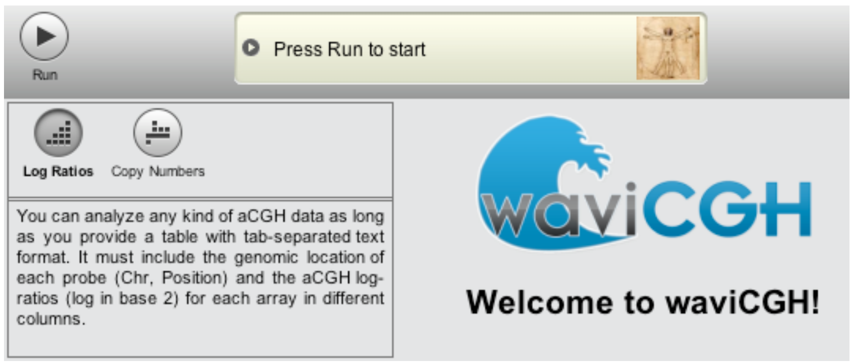


Then you will see how the results of the different steps corresponding to the selected protocol type example are loaded. We have included input data examples for the two types of starting material in waviCGH. For detailed information about these datasets and how they were generated, see the **Tutorials** section.

See next section to learn how to run an analysis in waviCGH. For a brief description of the options of each step click “Help”. This action will open the Help panel for the parameters (Param). For a brief description of the results of each step, click on “Results”. For a more detailed explanation of input and output, see the correspondent section of the manual.

Input Data

With waviCGH you can upload and process table with Log Ratios or with Copy Numbers. Just click on one of the top left buttons, select the file from your computer and click “Run”:



Protocol bottoms and Run bottom in the Upper Left of the server.

Reference DNA and colors

Depending on the laboratories, aCGH experiments can be carried out with the reference channel dyed using Cy3 or Cy5. If you upload a table of Log Ratios, we assume that Cy3 was the reference when log-ratios were calculated (if not, please calculate them again!).

waviCGH color for gains is green, losses are in red. Note that in expression arrays this is often the contrary.

Log Ratios

Click  to select “Log Ratios” type of input. waviCGH can read a table with aCGH log-ratios in this format:

"ProbeName", "Chr" (chromosome), "Position" of the probe, and then the names of the samples:

ProbeName	Chr	Position	Array1	Array2
A_55660	1	175	1.55	-1.03
A_98881	1	6035	1.21	-1.11
A_89770	2	300	0.03	-0.92

The data must be saved as a tab separated text file (compressed as .zip). Most people prepare microarray data in spreadsheet programs as Excel, but waviCGH does not recognize those format files. Please "save as" tab-separated text, and check that the columns names are correct. Finally, compress the file in zip format.

File size limit of 400 Mb. The maximum size file that waviCGH will process is 400 Mb, *AFTER the file is decompressed*. Please be aware that waviCGH needs to upload the file completely to uncompress the file and check its size.

Sexual chromosomes must be numerical: 23 (X) and 24 (Y) for human, 20 and 21 for mouse, and 21 for rat X.. Please note that sexual chromosomes must be numerical, otherwise the probes from X and Y will be removed at the Pre-processing step.

Copy Numbers

Click  to select "Copy Numbers" type of input. waviCGH can read a table with pre-calculated copy number states as -1/1/0 (loss/gain/no-change) in this format:

"ProbeName", "Chr" (chromosome), "Position" of the probe, and then the names of the samples:

ProbeName	Chr	Position	Array1	Array2
A_55660	1	175	-1	0
A_98881	1	6035	-1	0
A_89770	2	300	0	1

The data must be saved as a tab separated text file (compressed as .zip). Most people prepare microarray data in spreadsheet programs as Excel, but waviCGH does not recognize those format files. Please "save as" tab-separated text, and check that the columns names are correct. Finally, compress the file in zip format.

File size limit of 400 Mb. The maximum size file that waviCGH will process is 400 Mb, *AFTER the file is decompressed*. Please be aware that waviCGH needs to upload the file completely to uncompress the file and check its size. If you dataset is larger, we suggest to split it in different files and create different projects. If your dataset is really big and you are interested in a collaboration with us to analyze it, please contact us.

Sexual chromosomes must be numerical: 23 (X) and 24 (Y) for human, 20 and 21 for mouse, and 21 for rat X.. Please note that sexual chromosomes must be numerical, otherwise the probes from X and Y will be removed at the Pre-processing step.

SNP arrays

SNP microarray platforms like Illumina and Affymetrix allows to extract copy number information. In order to use data from these plataforms, you only need to calculate log-ratios using some sample (or some samples) as reference.

Illumina: you can export from Illumina software BeadStudio or GenomeStudio the R values (equivalent to single-channel intensities) or use directly the log-R-ratios (or LRR, see Illumina documentation for details, http://www.illumina.com/).

Affymetrix: you can use Affymetrix software (http://www.affymetrix.com) to export the intensities, or you can use dChip, free sofware for the analysis of Affymetrix arrays (Windows): http://www.hsph.harvard.edu/~cli/complab/dchip/copy.htm

A good alternative for R users is aroma.affymetrix: http://www.aroma-project.org

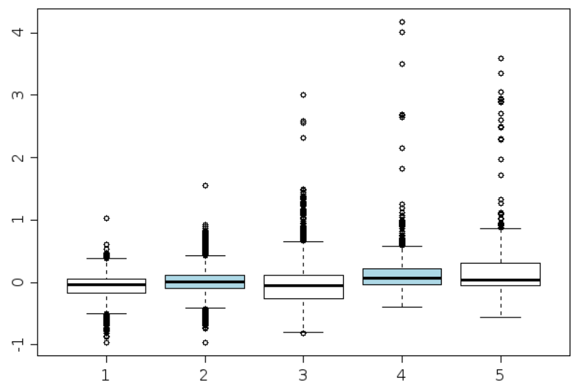
Diagnostic plots

Once your data have been uploaded into waviCGH, diagnostic plots are automatically generated. In the case of Log Ratios, they may help you to choose what kind of normalization is more appropriate to your data. Copy Numbers diagnostic plots show the distribution of the -1/0/1 states along the chromosomal position, using the information you displayed in the input data file.

Box plots

The box plots for each array show box-plots of the ratio for each array. In descriptive statistics, a boxplot (also known as a box-and-whisker diagram or plot) is a convenient way of graphically depicting groups of numerical data through their five-number summaries (the smallest observation, lower quartile (Q1), median (Q2), upper quartile (Q3), and largest observation). A boxplot may also indicate which observations, if any, might be considered outliers. We have used the standard boxplot function from R.

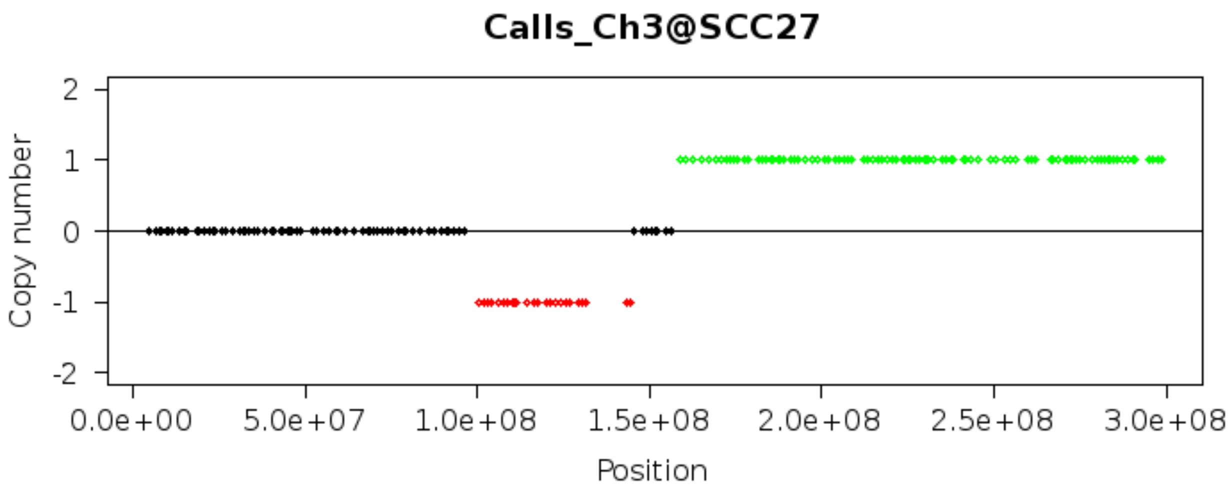
Boxplots allow you to assess if the general distribution and median (central line of the box-plot) of different arrays are comparable. If there are differences in scale among arrays, you might want to consider some type of normalization.



Diagnostic plots: Boxplots

Copy Numbers Plots

After loading a table of copy numbers the values along the chromosomal position are automatically plotted for each sample and chromosome. Gained probes (1) are shown in red, losses (-1) in green and no-changed probes (0) in black. These plots show the individual probe information before any post-processing.



Diagnostic plots: Copy numbers

Normalization

Normalization means to adjust microarray data for effects which arise from variation in the technology rather than from biological differences between the DNA samples or between the printed probes. waviCGH allows to do median normalization: to subtract the weighted median from the log-ratios for each array.

Median normalization method adjusts the log-ratios intra-array. Inter-array centering is a side effect: it assumes that most of the the data should be around zero (no different to reference DNA). Please use these procedure if the number of aberrations in your test sample is compatible with this assumption. Selecting normalization option “None” will leave the log-ratios as they are.

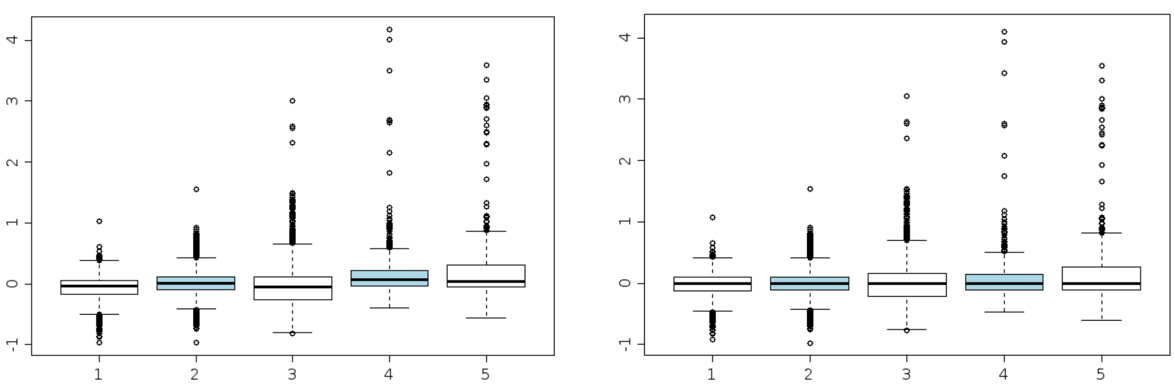
Genomic Coordinates

The position of a probe can vary between assemblies of the genome. You can convert the genomic coordinates of your probes from old human NCBI assemblies versions 34, 35 or 36 of the human genome to current Build 37 (GRCh37), and between from versions NCBIM34, M35 or M36 to the current NCBIM37.

This step is done before Pre-processing because the imputation of missing data depends on the position and distance between probes. In addition, the Waves adjust method uses GC content around probes that is taken from current genomic builds. Distance between probes can affect some segmentation methods too. New coordinates can be downloaded after the Pre-processing step.

Normalization Results

After normalization, box plots are generated again so you can examine the effect of it. If median normalization was chosen, the medians will be centered at zero.



Box plots before (left) and after median normalization (rigth).

Pre-processing

Pre-processing step englobes all the filtering and processing done after normalization. In fact, it could also be called (normalization) “post-processing” as normalization can also be considered a “processing step”. We call it Pre-processing, anyway, considering it the segmentation pre-processing step. Well, now that the terminology is clear, let’s see what this step does!

Average duplicated probes

Here we simply merge duplicated probes by the mean or the median. Duplicated probes was not an issue for normalization, but segmentation methods expect unique probes.

Chromosomes

You need to select whether you want to include sexual chromosomes in your analysis or not. Be aware than waviCGH does not know the gender of the reference DNA or any of the samples, so you will have to interpret your results according to the experimental design you did.

Missing values

You can filter missing values by setting the minimum proportion of existing values: if the proportion (percentage) of missing values in the total number of arrays is higher than this value, the probes will be removed. Otherwise, the values will be loess imputed using the method of Smith et. al (2007).

Genomic waves adjustment

You can optionally correct your log-ratios for genomic waves artifacts by selecting Wave adjust = Yes. waviCGH uses the method of Diskin et. al (2008). This method uses GC content information, and therefore is only available for human, mouse and rat data. Please note that your probe genomic position should be in the current assembly version, as we calculate the GC content around probes using latest builds. Remember that you can update your genomic positions by selecting the Genomic Coordinates=Yes option in the Input Data box.

Pre-processing Results

A summary table is generated after Pre-processing step:

field	value
Probes before filtering	3552
Duplicated probes	0
Probes with chromosome number higher than threshold	0
Probes without enough existing values for imputation	0
Probes after filtering	2054
Genomic waves adjusted	No

Preprocessing summary

- Probes before filtering: The number of probes before Pre-processing step.
- Duplicated probes: Number of probes that are duplicated (probes with same name).
- Probes with chromosome number higher than threshold: Number of probes which chromosome ("Chr") field was not numeric or a number higher than 22 (autosomes), 23 (including X) or 24 (including X and Y) depending on selection chosen.
- Probes without enough existing values for imputation: number of probes removed because they did not pass the minimum proportion value.
- Probes after filtering: Number of probes remaining after filtering and imputation.
- Genomics waves adjusted: Yes or No, depending on your selection.

In addition, new genomic coordinates can be downloaded if you selected this option, by clicking:

Download new Probe Position

Segmentation

We define ‘segmentation’ as the process of smoothing the observed/normalized log-ratios, so probes predicted to be in the same chromosomal segment have the same value. Translating these smoothed ratios into gain/no-change/loss states is a different step that we and others define as ‘calling’ (of gains and losses). Segmentation can have three main benefits: noise reduction, facilitate detection of lost and gained regions and make breakpoint analysis possible. We have made a selection from the most popular methods.

Segmentation methods

Here is a description of the segmentation methods and their modifiable parameters. We recommend to leave the default parameters unless you really know what you are doing. More detailed information about segmentation methods implementation is available in the final Appendix (pag, 30).

- **HaarSeg:** The recent and fast segmentation method by Ben-Yaacov and Eldar (2008). It is based on wavelet decomposition and thresholding, which detects significant breakpoints in the data. The only user option is the ***mad.threshold*** parameter, which controls calling (not segmentation).

- **DNAcopy:** Olshen & Venkatraman's popular algorithm, also known as circular binary segmentation (CBS). ***DNAcopy.min.width***: minimum number of markers for a changed segment (integer between 2 and 5). ***DNAcopy.alpha:*** Significance levels for the test to accept change-points (p-value: it should be higher than 0 and lower than 1). ***DNAcopy.nperm:*** Number of permutations used for p-value computation.

- **GLAD:** Huppe et al. build upon work of Polzehl and Spokoiny (2002) on non-parametric likelihood methods for breakpoint detection. ***GLAD.deltaN:*** Region with smoothing values in between the interval [-deltaN,+deltaN] are supposed to be normal. ***GLAD.forceGL2***: Level with smoothing value greater than this one are considered as gained. ***GLAD.forceGL1***: Level with smoothing value smaller than this one are considered as lost. ***GLAD.deletion***: Level (and outliers) with a smoothing value (log-ratio value) lower than this threshold are consider as deletion. ***GLAD.amplicon***: Level (and outliers) with a smoothing value (log-ratio value) greater than this threshold are consider as amplicon.

- **Wavelets:** This method was the first segmentation method that used wavelets for denoising the data, as described in Hsu et al. (2005). Its strategy is different to HaarSeg, and much slower - see the paper for details. After segmentation, we use calling using the MAD approach (see HaarSeg, above). For speed reasons, we do not recommend this method for high-density data.

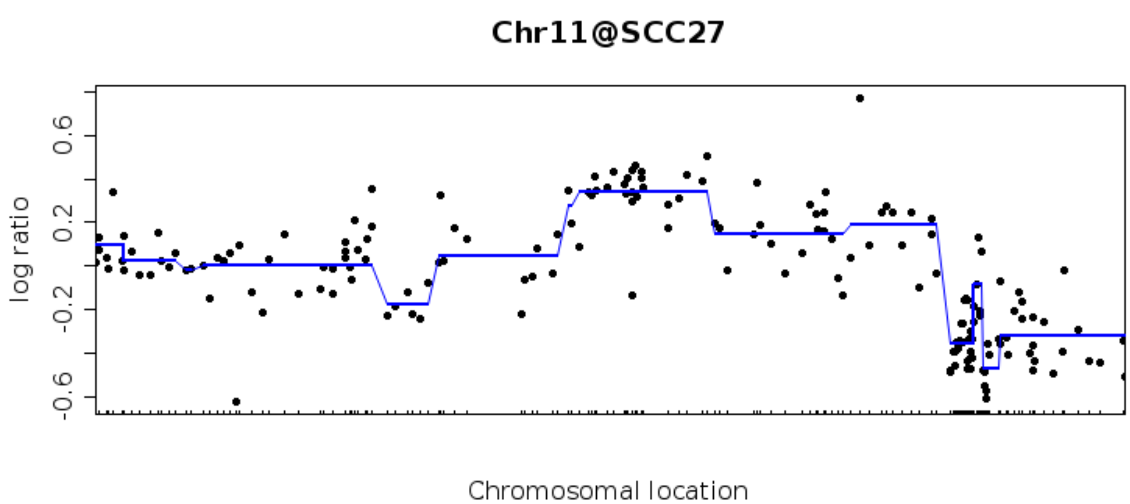
HMM: Fridlyand's and Dimitrov's (2007) Hidden Markov Model (HMM). Several homogeneous HMMs are fitted (with different numbers of underlying states) and then model selection, using AIC or BIC, is employed to select the number of states. ***AIC:*** Akaike Information Criterion. ***BIC***: Bayesian Information Criterion.

BioHMM: This method, developed by Marioni et al. (2006) is similar to HMM, except the models used are non-homogeneous HMMs that allow to incorporate distance between successive probes. ***AIC:*** Akaike Information Criterion. ***BIC***: Bayesian Information Criterion. For speed reasons, we do not recommend this method for high-density data.

CGHseg: Picard et al. (2005) model aCGH data as a random Gaussian process with abrupt changes mean. After segmentation, we use calling using the MAD approach (see HaarSeg, above). For speed reasons, we do not recommend this method for high-density data.

Segmentation Results

Post-segmentation plots show the Log Ratios before segmentation (black points scatterplot) and after segmentation (blue line) for all analyzed chromosomes in each sample. These plots can help to detect by eye big regions gained or lost, but we recommend to run the next step (Calling) to detect these regions based on statistical approaches.



Segmentation plot

Calling of gains and losses

After segmentation, you can select between two alternative calling methods: segmentation-based or probability-based. Basically, they try to translate your segmented ratios into meaningful biological estates. Both strategies will give calls in a numerical format of 0/-1/1 (and 2 for amplifications, optionally, in probability-based).

Segmentation based Calling

Segmentation-based calling is done using mergeLevels algorithm (Willenbrock et. al, 2005) except for GLAD segmentation case, that has its own region assignment algorithm (Hupé et. al, 2004) and HaarSeg, where were users can follow the recommendation from the authors of using a median absolute deviation (mad, HaarSeg.m parameter) cut-off of their choice. The original paper from Willembrock et. al does not map to a set of "gain/loss/no-change" levels. We have followed the same approach as in DNAcopy, and use here the mergeLevels procedure. It must be emphasized that this is an experimental procedure, not described in the original paper. Moreover, the wavelet-smoothing procedure returns smoothed values that rarely fool into a set of categories, so applying mergeLevels here often leads to non-sense results. Thus, we apply mergeLevels after running the original clustering procedure of this method with a very small threshold for merging (currently set to 0.05, or five times smaller than the default of 0.25); some preliminary trials show that the final outcome from mergeLevels is not sensitive to small variations around this threshold.

Probability based Calling

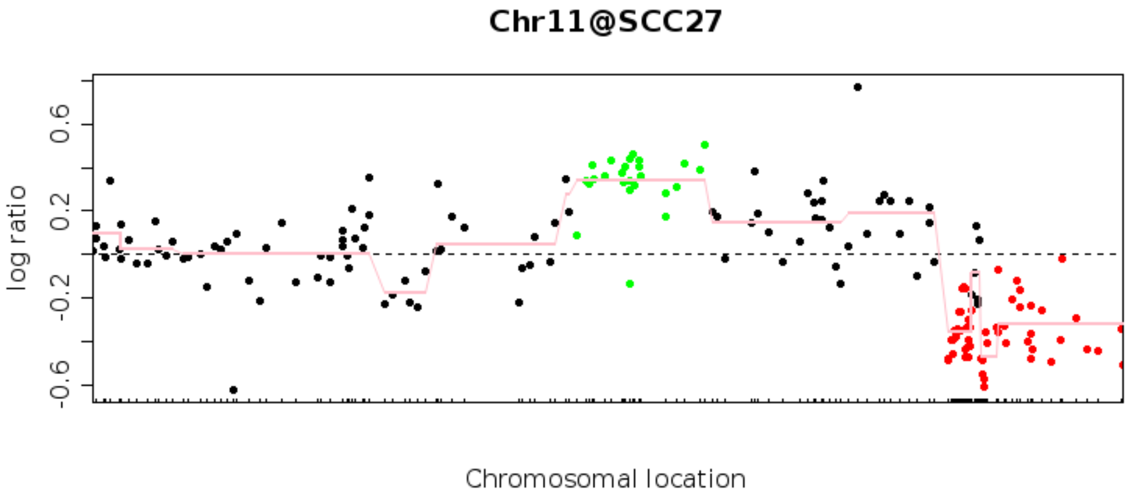
Probability-based calling is done using CGHcall package (VanDeWiel et al., 2007) from Bioconductor. CGHcall utilizes the segmented data only for breakpoint definition, and then calculates the probability of each segment of being no-changed, lost, gained or, optionally, amplified. This method can be applied after any segmentation method, in contrast to methods that are incorporated in the segmentation model itself.

- Classes: you can calculate the probability for each probe to be in three classes (lost, normal and gain) or four (differentiate one-copy gains and amplifications).

Calling Results

Segmentation-based Calling

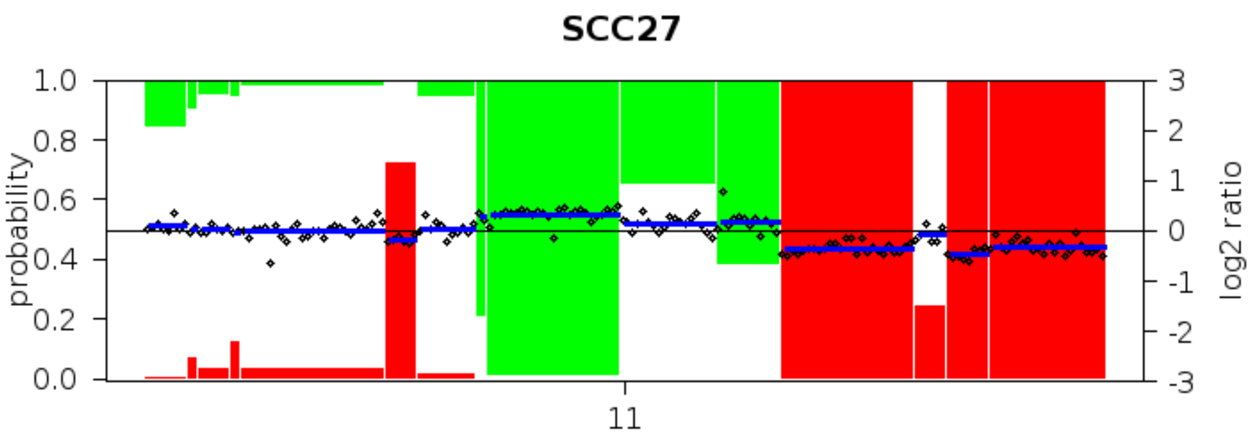
Post-calling plots are similar to post-segmentation plots, but gained (green) and lost (red) probes are highlighted.



Calling plot example (Segmentation-based)

Probability-based Calling

'Probability' post-calling plots show the probability of each part of the genome of each sample being gained (green) or lost (red) calculated by CGHcall (van den Wiel et. al, 2007) together with the information previously displayed in the post-segmentation plots: the Log Ratios before segmentation (black points scatterplot) and after segmentation (blue line) for all analyzed chromosomes in each sample. If the probability of an alteretion is higher than 0.5 (horizontal central line of the plot) the region is called gained (+1 for each probe in a gained region) or lost (-1). Normal probes are given a zero (0) value. When 4 levels have been used for calling, amplifications are indicated with a blue tickmark at the top of the plot.



Calling plot example (Probability-based)

The probability axis (left) is inverted for gains.

Additionally, he probabilities of each estate can be downloaded as tab-separated text file:

ProbeName	Chr	Position	log2ratio_AdCA10	loss_AdCA10	normal_AdCA10	gain_AdCA10
RP11-465B22	1	1051642	-0.133184165	6.63E-227	0.237335502	0.762664498
RP4-785P20	1	3284806	-0.08448131	6.63E-227	0.237335502	0.762664498
RP1-37J18	1	4542450	0.339021576	6.63E-227	0.237335502	0.762664498
RP3-438L4	1	7103442	0.341535645	6.63E-227	0.237335502	0.762664498
RP11-338N10	1	7674468	0.218027148	6.63E-227	0.237335502	0.762664498
RP3-510D11	1	9200696	0.205887546	6.63E-227	0.237335502	0.762664498
RP4-575L21	1	10122792	0.302520239	6.63E-227	0.237335502	0.762664498

Results table of Probability Calling, including probabilities for each state.

Minimal Common Regions

Two different strategies can be used to find minimal common regions (MCRs): 'SuperSORI' and 'Permutations'. The first method is faster because it assumes that the calls are real and simply looks for segments intersection. The second takes into account the frequency of calls in each sample and chromosome to calculate statistically significant MCRs, based on permutations of the calls.

SuperSORI

SuperSORI is a fast method that detect all MCRs shared by at least two individuals, also called smallest regions of imbalance (SORI) (Ferreira et. al, 2008). SuperSORI performs a curation of the calls to generate consistent segments before searching MCRs. First, it filters out segments of a given number of probes and joins segments separated by gaps of less than a given size (in base pairs or number of probes). Then, MCRs are found by obtaining the intersection of the curated segments. This step can be useful to avoid fragmentation of the results from noisy data, but should be used with caution.

Segment Filtering: Define minimum number of probes per segment. (Set to 1 if you don't want to filter any call out.)

Join Segments Gaps (by probes): Any two segments gained or lost (must be of same aberration class) separated by this number of probes or less will be merged into a unique segment.

Join Segments Gaps (by bases): Any two segments gained or lost (must be of same aberration class) separated by this number of bases (genomic distance) or less will be merged into a unique segment.

Permutations

'Permutations' method is a user friendly implementation similar to other approaches (Diskin et. al, 2006; Kim et. al, 2008), and computes for each probe a p-value testing the significance of the alteration of that probe across the set of samples. This p-value is based on a permutation test that assumes that the alterations found are randomly located in the genome. Then, the consecutive probes with p-values lower than a cut-off are joined in a common region.

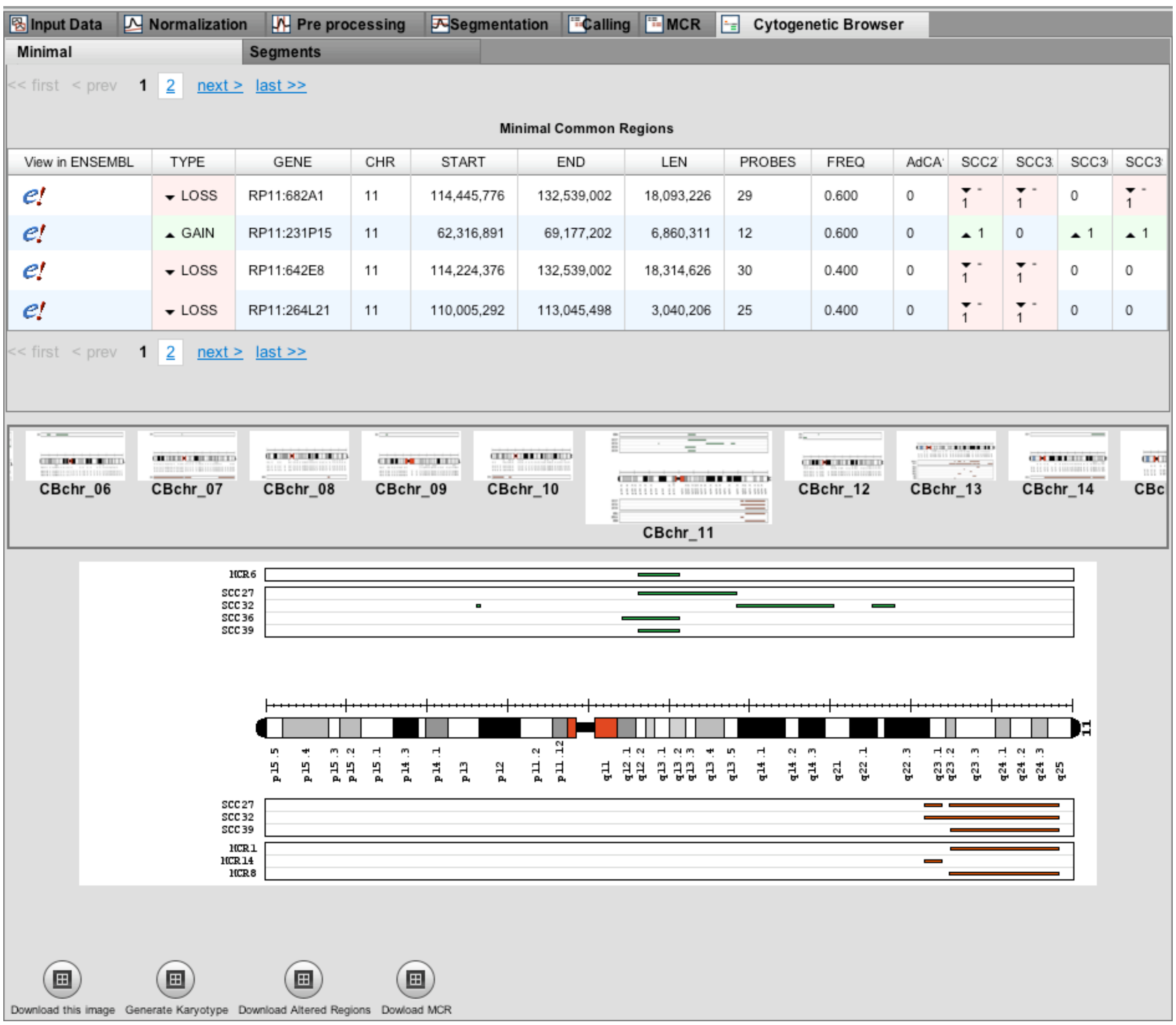
P-value cut-off: Only regions with a (not adjusted) p-value higher than cut-off will be displayed. FDR adjusted p-values will be also given.

Cytogenetic Browser

After segmentation and calling, you can send the altered regions (gain, lost or both type of segments) to the integrated Cytogenetic Browser. Currently, this feature is available for human, mouse and rat genomes.

Cytogenetic Browser Description

The browser displays chromosome images with alterations (at bottom), MCRs (left tab) and individual altered regions (right tab, you may need to scroll down to see it):



View in ENSEMBL	TYPE	GENE	CHR	START	END	LEN	PROBES	FREQ	AdCA	SCC2	SCC3	SCC3	SCC3
e!	LOSS	RP11:682A1	11	114,445,776	132,539,002	18,093,226	29	0.600	0	1	1	0	1
e!	GAIN	RP11:231P15	11	62,316,891	69,177,202	6,860,311	12	0.600	0	1	0	1	1
e!	LOSS	RP11:642E8	11	114,224,376	132,539,002	18,314,626	30	0.400	0	1	1	0	0
e!	LOSS	RP11:264L21	11	110,005,292	113,045,498	3,040,206	25	0.400	0	1	1	0	0

✱ Chromosome Navigator:

Examine your aCGH results in a cytogenetically context. Images of all chromosomes show any region gained (red bars) or lost (green bars) aligned with the cytobands where they are located. MCRs are numbered as MCR# to facilitate their location in the Minimal Common Regions Table. Segments are numbered as SampleName# as they appear in the Segments Table.

✱ Minimal Common Regions Table

The MCRs (Minimal Common Regions) and their location are found are displayed in the first table. Patients that share the MCR have a +1 if it is a gain and -1 if it is a loss. Patients that do not have that aberration are indicated with zero (0).

✱ Segments Table

All altered regions and their location are shown in the second Segments Table. Note that these results can vary with the Calling results if you decided to apply any gap joining post-processing. You may have to scroll down to see this table if the number of MCR is large.

In both tables you will find links to Ensembl genome browser for each region (either MCR or individual segments). Clicking them will open a new tab/window in your web broswer.

Tutorials

The following tutorials show all the steps that you should follow to obtain the results you can observe when the corresponding example projects are loaded.

1. Log Ratios - 5 cervix tumors

This example dataset for log-ratios (and the corresponding copy number data) is a subset of the squamous cell carcinomas and adenocarcinomas of the cervix presented in (Wilting et. al, 2006). The data have been obtained from the *CGHcall* package (van den Wiel et. al, 2007). You can download the input file ready to use in waviCGH from:

http://wavi.bioinfo.cnio.es/TUTORIALS/LogRatiosWilting.txt.zip

a. Input Data: Select *human* as the *Species*, and in *Genomic Coordinates* choose *Yes*: this action will open a new option box, where you can convert the genomic position from the NCBI build 35 corresponding to this array (Wilting and van den Wiel, personal communication) to the build GRCh37. Upload the file.

b. Normalization: you will see in the boxplots originated that the log-ratios distribution are not centered in zero, so we will median-normalized.

c. Preprocessing: we will leave the default parameters here.

d. Segmentation: we will use the *HaarSeg* mehod. HaarSeg.m is the median absolute deviation (mad) that waviCGH will use as calling cut-off for Segmentation-based calling. Leave the default of 2 deviations.

e. Calling: Use the Segmentation-based method, which is based on the previously set mad of 3. Clic on *Download Calls*: you will download a text file with calls in -1/0/1 format. This file can be saved and used anytime in a new Copy Numbers project (see Tutorial 4).

f. MCR: Use *SuperSORI* with the default parameters. Clic in FREQ to rank the MCRs by their frequency.

g. Explore the results in the Cytogenetic Browser.

2. Log Ratios - 8 healthy people CNVs

In this tutorial we will analyze log-ratios that were produced with Agilent aCGH 244K custom arrays for the validation of copy number variations (CNVs) discovered by clone-based sequencing in eight healthy individuals of diverse geographic ancestry (GEO accession [GSE10008](GSE10008) , see Kidd 2008 for details). We downloaded the raw data from GEO and selected samples with reference in Cy3 (green). Log-ratios were calculated using processed red and green signals. You can download the input file ready to use in waviCGH from:

http://wavi.bioinfo.cnio.es/TUTORIALS/LogRatiosKidd.txt.zip

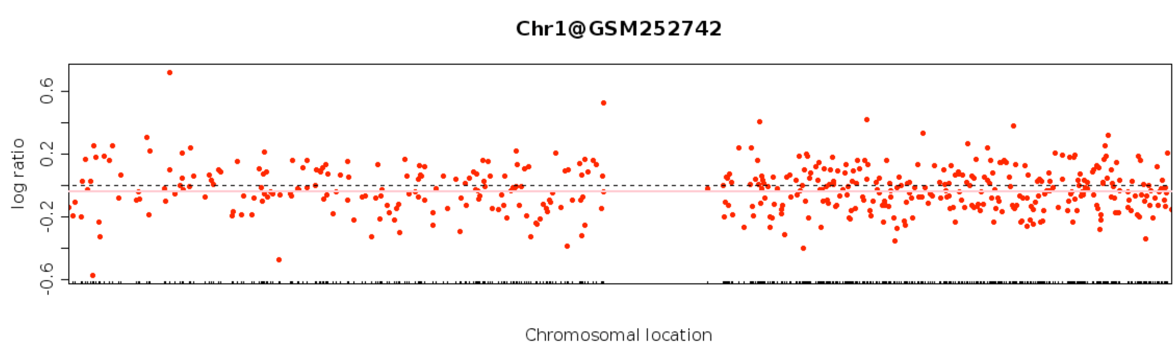
a. Input Data: Select *human* as the *Species*, and in *Genomic Coordinates* choose *Yes*: this action will open a new option box, where you can convert the genomic position from the NCBI build 35 corresponding to this array (GEO information, GPL5884) to the build GRCh37. Upload the file.

b. Normalization: you will see in the boxplots originated that the log-ratios distribution are not centered in zero, so we will median-normalized.

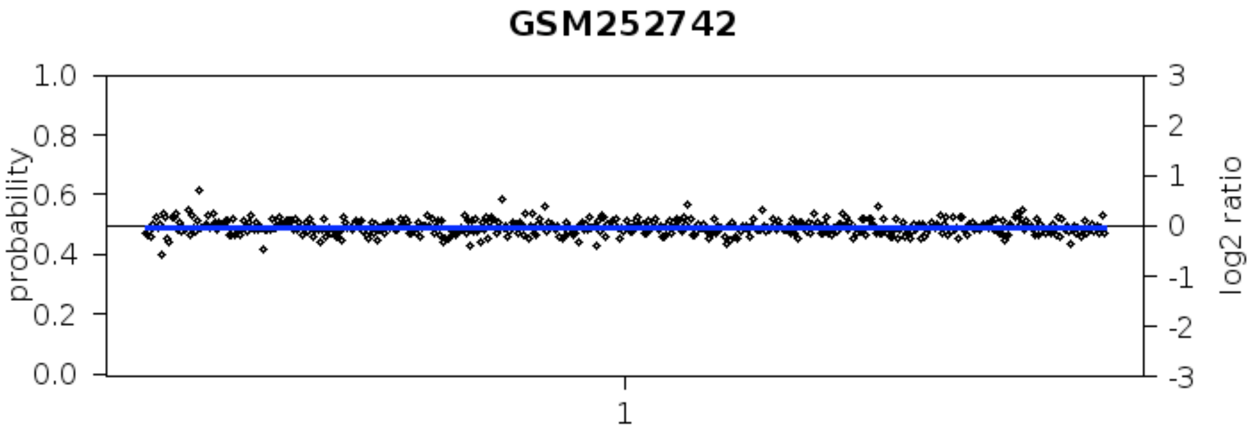
c. Preprocessing: we will leave the default parameters here.

d. Segmentation: we will use the *DNAcopy* mehod. *DNAcopy* is slower than HaarSeg, but is not as sensitive to outliers, and we find it more appropriate to detect germinal CNVs.

e. Calling: We will use the Probability-based Calling, with default parameter of three classes (loss, normal, gain). You can run a project in paralell using Segmentation-based calling, and you will find out that this method (*mergeLevels* is used) tends to call *entire* chromosomes as lost or gained. See for example chromosome 1 in GSM252742, where *mergeLevels* calls the whole chromosome as lost:



while the Probability-based CGHcall calls all probes as normal:



These results differences are probably caused by the design of the microarray (GPL5884) where probes are mainly targeting regions of CNVs. Clic on *Download Calls*: you will download a text file with calls in -1/0/1 format. This file can be saved and used anytime in a new Copy Numbers project (see Tutorial 5). You may also like to download the *Probabilities*.

f. MCR: Use *Permutations* with the default parameters. Clic in PVALADJ (p-value adjusted for multiple testing) to rank the MCRs by their significance.

g. Explore the results in the Cytogenetic Browser.

3. Log Ratios - Replication timing ratios from mouse

This example serves us to show how waviCGH can handle data from mouse. The data-set corresponds to replication timing Log Ratios from mouse cells, obtained using FSU NimbleGen Mouse 385K Whole Genome Tiling arrays (GSE18079). You can download the input file ready to use in waviCGH from:

http://wavi.bioinfo.cnio.es/TUTORIALS/LogRatiosYokochi_mouse.txt.zip

Check build:

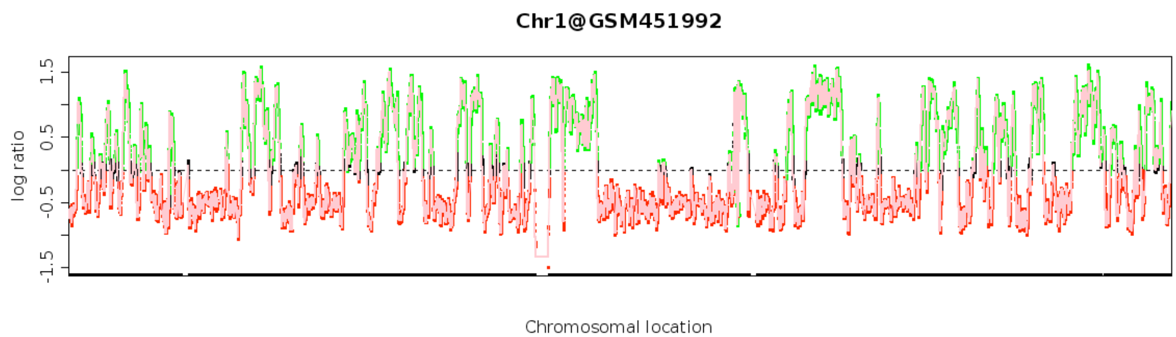
a. Input Data: Select *mouse* as the *Species*, and in *Genomic Coordinates* choose *Yes*; probe mapping correspond to mouse genome version MM8, NCBI 36, and we want to convert their positions to version M37.

b. Normalization: you will see in the boxplots originated that the log-ratios distribution are centered in zero, so we will select Normalization=None.

c. Preprocessing: we will leave the default parameters here.

d. Segmentation: we will use the *HaarSeg* mehod. Actually, we found that **HaarSeg was the only method that could segment these type of data, which have been previously smoothed.** HaarSeg mad.threshold is the median absolute deviation (mad) that waviCGH will use as calling cut-off for Segmentation-based calling. Leave the default of 2 deviations.

e. Calling: Use the Segmentation-based method, which is based on the previously set mad of 2:



Green gained regions correspond to early-replicating regions, while red lost regions to late-replication regions.

f. MCR: Use *SuperSORI* with the default parameters. Clic in FREQ to rank the MCRs by their frequency.

g. Explore the results in the Cytogenetic Browser.

4. Copy Numbers - 5 cervix tumors

You can use the *Calls* file from Tutorial 1, or download the input file ready to use in waviCGH from:

http://wavi.bioinfo.cnio.es/TUTORIALS/CopyNumbersWilting.txt.zip

You don't need to do Genomic Coordinates conversion, this was done already.

You can change the MCR method to *Permutations*, and compare the results with the ones obtained with *SuperSORI* in Tutorial 1.

5. Copy Numbers - 8 healthy people CNVs

You can use the *Calls* file from Tutorial 2, or download the input file ready to use in waviCGH from:

http://wavi.bioinfo.cnio.es/TUTORIALS/CopyNumbersKidd.txt.zip

You don't need to do Genomic Coordinates conversion, this was done already.

You can change the MCR method to *SuperSORI*, and compare the results with the ones obtained with *Permutations* in Tutorial 2.

References

Ben-Yaacob E and Eldar YC. (2008). A fast and flexible method for the segmentation of aCGH data. Bioinformatics, 24 ECCB 2008, i139-i145.

Diskin, S.J., Eck, T., Greshock, J., Mosse, Y.P., Naylor, T., Stoeckert, C.J., Weber, B.L., Maris, J.M. and Grant, G.R. (2006) STAC: A method for testing the significance of DNA copy number aberrations across multiple array-CGH experiments. Genome Res, 16, 1149-1158.

Diskin SJ, Li M, Hou C, Yang S, Glessner J, Hakonarson H, Bucan M, Maris JM, Wang K (2008). Adjustment of genomic waves in signal intensities from whole-genome SNP genotyping platforms. Nucleic Acids Res, 36(19), e126.

Ferreira BI, Alonso J, Carrillo J, Acquadro F et al. (2008). Array CGH and gene-expression profiling reveals distinct genomic instability patterns associated with DNA repair and cell-cycle checkpoint pathways in Ewing's sarcoma. Oncogene, 27(14):2084-90.

Fridlyand J and Peter Dimitrov (2007). aCGH: Classes and functions for Array Comparative Genomic Hybridization data. R package version 1.12.0.

Greshock, J., Feng, B., Nogueira, C., Ivanova, E., Perna, I., Nathanson, K., Protopopov, A., Weber, B.L. and Chin, L. (2007) A comparison of DNA copy number profiling platforms. Cancer Res, 67, 10173-10180.

Hupe P, Stransky N, Thiery JP, Radvanyi F, and Barillot E. (2004). Analysis of array cgh data: from signal ratio to gain and loss of dna regions. Bioinformatics, 20:3413--3422.

Hsu L, Self SG, Grove D, Randolph T, Wang K, Delrow JJ, Loo L, Porter P. (2005) Denoising array-based comparative genomic hybridization data using wavelets. Biostatistics, 6:211-26.

Kidd JM, Cooper GM, Donahue WF, Hayden HS, Sampas N, Graves T, Hansen N, Teague B, Alkan C, et al. (2008). Mapping and sequencing of structural variation from eight human genomes. Nature, 453 (7191), 56-64.

Kim, T.M., Jung, Y.C., Rhyu, M.G., Jung, M.H. and Chung, Y.J. (2008) GEAR: genomic enrichment analysis of regional DNA copy number changes. Bioinformatics, 24, 420-421.

Marioni JC, Thorne NP, Tavaré S. (2006). Biohmm: a heterogeneous hidden markov model for segmenting array cgh data. Bioinformatics, 22:1144--1146.

Olshen AB, Venkatraman ES, Lucito R, Wigler M. (2004) Circular binary segmentation for the analysis of array-based DNA copy number data. Biostatistics, 5(4):557-72.

Picard P, Robin S, Lavielle M, Vaisse C, Daudin JJ. (2005). A statistical approach for array cgh data analysis. BMC Bioinformatics, 6:27, 2005.

Polzehl,J. and Spokoiny,S. (2002) Local likelihood modelling by adaptive weights smoothing. WIAS-Preprint 787.

Ritchie, M. E., Silver, J., Oshlack, A., Silver, J., Holmes, M., Diyagama, D., Holloway, A., and Smyth, G. K. (2007). A comparison of background correction methods for two-colour microarrays. Bioinformatics 23(20):2700-2707.

Smith ML. and John C. Marioni and Steven McKinney and Tom J. Hardcastle and Natalie P. Thorne (2007). snapCGH: Segmentation, normalisation and processing of aCGH data. R package version 1.6.0

van de Wiel MA., Kyung In Kim, Sjoerd J. Vosse, Wessel N. van Wieringen, Saskia M. Wilting and Bauke Ylstra (2007). CGHcall: calling aberrations for array CGH tumor profiles. Bioinformatics, 23, 892-894.

Willenbrock H, Fridlyand J. (2005). A comparison study: applying segmentation to array CGH data for downstream analyses Bioinformatics, 21: 4084-4091.

Wilting, S.M., Snijders, P.J., Meijer, G.A., Ylstra, B., van den Ijssel, P.R., Snijders, A.M., Albertson, D.G., Coffa, J., Schouten, J.P., et al. (2006) Increased gene copy numbers at chromosome 20q are frequent in both squamous cell carcinomas and adenocarcinomas of the cervix. J Pathol, 209, 220-230.

Yokochi T, Poduch K, Ryba T, Lu J, Hiratani I, Tachibana M, Shinkai Y, Gilbert DM. G9a selectively represses a class of late-replicating genes at the nuclear periphery. Proc Natl Acad Sci U S A. 2009 Nov 17;106(46):19363-8.

Appendix: Segmentation methods details

This section describes the technical details about the implementation of segmentation methods and the parameters that we used. For a description of the methods, the modifiable parameters and the results, please see the section **Segmentation (pag. 15)**.

We used R, version 2.10.0 Patched (2009-11-09 r50375). We have prepared a package (currently called ADaCGH21), which is a successor of our ADaCGH package in CRAN. This package parallelizes all calls to methods, and makes extensive usage of the ff package for memory-efficient storage of large data on disk. Details on the calls follow.

GLAD

We use version 2.6.0. Segmentation is done with the recommended daglad function (see vignette for GLAD). daglad is called using all the default options except, possibly, for the user-modifiable options of deltaN, forceGL, deletion, and amplicon.

HaarSeg

We use as the basic computational workhorse the original code from Erez Ben-Yaacov, downloaded from :

http://www.ee.technion.ac.il/people/YoninaEldar/Info/software/HaarSeg.htm

on May 2009, and corresponding to Version 1.0 of June 2008. Note that this code (both the R code and the C code) is identical to the currently (2010-04-08) available one as an R package, HaarSeg_0.0.2.tar.gz (packaged 2010-01-24). We use all the default options (the same as would be used in a call to the "haarSeg" function in the current R package). The only user option is the mad.threshold parameter, which controls calling (not segmentation). See p. i141 of the original reference.

DNAcopy

We use version 1.20.0 of the DNAcopy package. We use all the default parameters which would be equivalente to calling the "segment" function with default parameters except, possibly, for the user-modifiable options of min.width, alpha, and nperm. mergeLevels is always used on the output, using all the default options of mergeLevels, as implemented originally in snapCGH (v. 1.16.0), but with two bug-fixes for the Ansari test. Following the recommendations (see original paper and vignette) we use smoothing of the CNA object (by calling an internal version of smooth.CNA with all default parameters).

HMM

We use package aCGH, version 1.22.0, and use the function find.hmm.states with all default parameters except, possibly, aic and bic, which the user can choose. As with DNAcopy, and following Willenbrock and Fridlyand, we use mergeLevels on the output.

BioHMM

We use package snapCGH v. 1.16.0 and call the function fit.model with default parameters except, possibly, aic and bic, which the user can choose. We use mergeLevels on the output with all default parameters.

CGHseg

Note that, incorrectly, many papers state that they use CGHseg when using the segment function in the tilingArray package. That, however, is not correct, since it falls short of using the penalization procedure that Picard et al. developed and recommended. We have implemented CGHseg as described in the original paper, by using the segment function in tilingArray (v. 1.24.0) and then finding the optimal number of segments with the penalization scheme developed by Picard et al. The only parameter of this method is the threshold for the adaptive penalization in Picard et al.'s CGHseg. See p. 13 of the original paper. The default value used in the original reference is -0.5. However, our experience with the simulated data in Willenbrock and Fridlyand (2005) indicates that for those data values around -0.005 are more appropriate. We use here -0.05 as default. After segmentation, we use calling using the MAD approach (see HaarSeg, above).

Wavelets

We use R code originally provided by Li Hsu. We use all default parameters of this method. There is no R package, aside from our, that provides a canonical implementation of this method. We use the same default parameters as those available from our ADaCGH package (v. 1.5-3) from CRAN. (minDiff = 0.25, minMergeDiff = 0.05, thrLvl = 3, initClusterLevels = 10). After segmentation, we use calling using the MAD approach (see HaarSeg, above).

Availability:

All the code for our package and additional scripts are available from

http://launchpad.net/adacgh