

# Analysis of DNA Copy Number Aberrations with Chipster

Ilari Scheinin  
firstname.lastname@gmail.com

November 19, 2013

## Abstract

This tutorial covers analysis of DNA copy number aberrations from either array comparative genomic hybridization (aCGH) or next generation sequencing (NGS) data with [Chipster](#). The first section covers importing your data and is divided to separate sections for microarray and sequencing data. The rest covers the downstream analysis steps that are mostly common to both data types.

## Contents

<b>1</b>	<b>Importing your data into Chipster</b>	<b>2</b>
1.1	Microarrays . . . . .	2
1.1.1	Gene Expression Omnibus database . . . . .	2
1.1.2	CanGEM database . . . . .	2
1.1.3	Local files . . . . .	2
1.1.4	Quality control . . . . .	3
1.2	Next generation sequencing . . . . .	3
1.2.1	FASTQ files . . . . .	3
1.2.2	BAM files . . . . .	4
<b>2</b>	<b>Basic copy number analysis workflow</b>	<b>5</b>
2.1	Segmentation . . . . .	5
2.2	Calling gains and losses . . . . .	5
2.3	Identifying common regions . . . . .	5
2.4	Clustering . . . . .	7
2.5	Known copy number variations . . . . .	7
2.6	From features to genes . . . . .	8
2.7	Genome browser . . . . .	8
<b>3</b>	<b>Additional analysis steps</b>	<b>10</b>
3.1	Plotting copy number profiles . . . . .	10
3.2	Removing wavy artifacts from aCGH profiles . . . . .	10
3.3	Comparisons between groups . . . . .	10
3.4	Survival analysis . . . . .	10
3.5	Integration with expression . . . . .	12
3.6	Enriched Gene Ontology categories . . . . .	13
<b>4</b>	<b>Workflow diagrams</b>	<b>14</b>
4.1	Main copy number tools . . . . .	14
4.2	Copy number annotation tools . . . . .	15
4.3	Tools for integrating copy number and expression data . . . . .	16

# 1 Importing your data into Chipster

The first step is to import your data into the Chipster session. For copy number analysis, Chipster supports both microarray and next generation sequencing (NGS) data. The first section of this tutorial covers importing data, and is divided into two subsections for these two data types. The subsequent sections are common to both.

## 1.1 Microarrays

Microarray data can be imported from local files, or directly from GEO (1) or CanGEM (2) databases. Each option is outlined in their own section.

### 1.1.1 Gene Expression Omnibus database

For data stored in the [Gene Expression Omnibus \(GEO\) database](#) (1), you need the accession number of the data set, such as [GSE17181](#). A GEO Series (identified by an accession that starts with GSE) can contain Samples measured with one or more microarray Platforms. If there is more than one, you need to also specify the accession number of the Platform (starts with GPL) you want to import. In the example case of [GSE17181](#), there is only one Platform ([GPL8841](#)), so filling in the Platform accession is not needed. You can check this by going to the web page of the GEO Series and scrolling down to the Platform section.

Once you have the accession number of the Series (and the Platform if needed), run the tool **Copy number aberrations / Import from GEO** specifying the accession number(s) as tool parameters.

Bear in mind that data imported this way has already been normalized (and the normalization method might vary from one data set to another). If you want to be able to normalize the data yourself, the original raw data is often (but not always) available in the Supplementary files section on the web page for a Series. This file can be downloaded, uncompressed, and then imported as outlined below for local files.

Another important detail to remember is that Chipster imports the Platform annotations (such chromosome name, starting and ending base pair positions, cytoband, gene symbols and descriptions) that are stored in the GEO Platform entry. The genome build used for these annotations can vary from one Platform to another, so please check the Platform web page to check which one was used.

### 1.1.2 CanGEM database

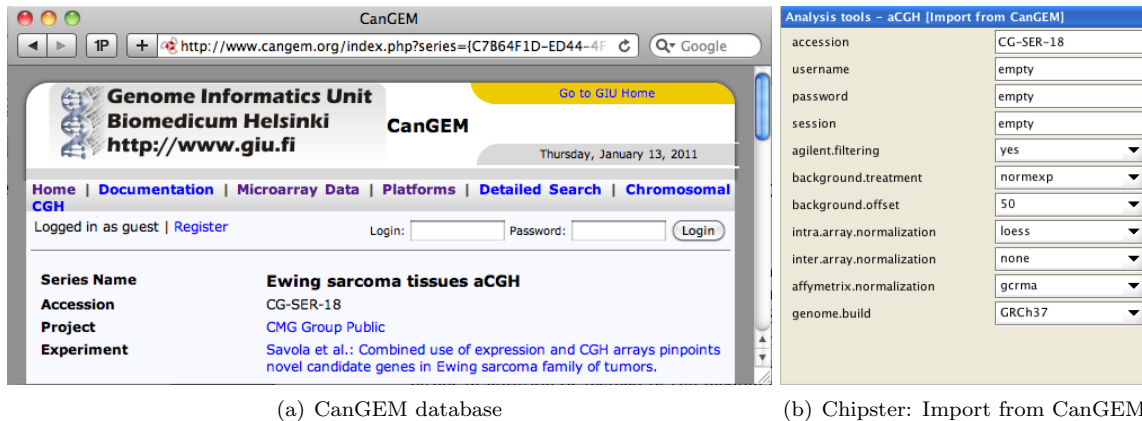
If the starting data is stored in the [CanGEM database](#) (2), the tool **Copy number aberrations / Import from CanGEM** download the original data files and performs normalization. Enter the accession number of the data in question, and change the normalization parameters if needed (the default values are recommended) and the genome build in case you do not want to use the latest one.

If your data is password-protected, there are two ways of accessing it. The first one is to enter your username and password into the corresponding parameters, but this will result in them being saved to any session and workflow files you create. A more secure approach is to log in on the CanGEM web site, locate the session ID on the bottom right corner of the page (the ID looks something like “ee8cbd9dcaa8284189f1582816531f46”), and copy&paste it into the session parameter in Chipster. This way Chipster can still download your data files. But after you log out (or the session times out after 24 minutes), saved sessions or workflows cannot access your private data anymore.

### 1.1.3 Local files

Local files can be imported using the import tool as described in [this tutorial](#). For Agilent Feature Extraction files, choose ProbeName as the Identifier, and depending on the dyes used, either gMedianSignal/gBGMedianSignal for Sample/Sample BG and rMedianSignal/rBGMedianSignal for Control/Control BG, or *vice versa*. Depending on the settings of Feature Extraction, your files might contain columns for mean signals instead of medians (*e.g.* gMeanSignal), either in addition to or instead of the median signals. They can also be used.

The next step is normalization, with *e.g.* the **Normalization / Agilent 2-color** tool. The default parameter values are recommended.



**Figure 1:** Importing data from the CanGEM database using the **Copy number aberrations / Import from CanGEM** tool. Note the accession number on the CanGEM web site, and enter it into Chipster.

For all aCGH data analysis, it is crucial to know what locations in the genome the array probes hybridize to. These annotations can be downloaded from either GEO or CanGEM databases with the respective tools **Copy number aberrations / Fetch probe positions from GEO** and **Copy number aberrations / Fetch probe positions from CanGEM**.

GEO mappings are based on information from the platform manufacturers, and each Platform entry contains a single set of mappings to a specific build of the reference genome. Mappings in CanGEM have been obtained by manual alignment of probe sequences to the human reference genome (3), and are available for different builds of the human genome. The list of available array platforms can be found from these links for both [GEO](#) and [CanGEM](#). In Chipster, these annotations are saved to columns named chromosome, start and end. For the rest of the tutorial, it is assumed that these columns are present in your data.

### 1.1.4 Quality control

To evaluate if there are badly performing arrays that should be left out of the analysis, the following tools can be used. They (like most downstream analysis tools) require that there are no missing values in the data, so it might be necessary to run **Preprocessing / Impute missing values** first. Boxplots and intensity plots can be created with the **Quality control / Agilent 2-color** tool, and examined for outlier arrays. Principal component analysis (PCA) can be performed with **Statistics / PCA** (change parameter do.pca.on from “genes” to “chips”). When visualizing the result, choose “3D Scatterplot for PCA” from the Method pop-up menu. This is an interactive visualization that allows you to rotate the 3-dimensional plot, and also change the coloring of the individual samples based on variables in the phenodata table. This can be useful for preliminary screening of differences between sample groups.

## 1.2 Next generation sequencing

Copy number analysis for next generation sequencing (NGS) data can be performed starting from BAM or FASTQ files. If you already have BAM files, you can skip over the next section.

### 1.2.1 FASTQ files

Import your FASTQ files into the Chipster session by choosing Import files... from the File menu. Next you need to align the sequencing reads to the reference genome. For copy number analysis, the recommended aligner is BWA (4). The mappability data used later in the downstream analyses has been calculated allowing for two mismatches, so this is also recommended when aligning your reads. Also, increasing the Quality trimming threshold increases the total number of reads that can be aligned.

Recommended parameters are therefore: Maximum edit distance for the whole read: 2; Quality trimming threshold: 40.

### 1.2.2 BAM files

If you had BAM files to start with, import them into the Chipster session by choosing Import files... from the File menu. If your files are already sorted and you're also importing the corresponding index files (with file extension .bai), you can skip the preprocessing step when prompted.

Next select your BAM file and run the tool **CNA-seq / Count reads in fixed bins** with the desired bin size. Repeat this for each BAM file. Then select all the resulting files with binned read counts and run the tool **CNA-seq / Define CNA-seq experiment**. Read counts per bin are affected by their GC content, and should therefore be corrected by running the tool **CNA-seq / Correct for GC content**. Finally, normalize your data by running **CNA-seq / Normalize copy number data**.

## 2 Basic copy number analysis workflow

This section assumes a starting point of a normalized data set that contains the genomic position of each feature (*i.e.* has columns chromosome, start and end). The term feature can refer to probes on an aCGH microarray, or bins of NGS experiments.

### 2.1 Segmentation

This first step in copy number analysis is usually segmentation. Segments refer to non-overlapping areas that most likely share the same copy number and are separated by breakpoints. To do this in Chipster, use the tool **Segment copy number data**. The parameters allow one to set the minimum number of features per segment (default: 2), the minimum number of standard deviations required between segments (default: 0), and the significance level required between segments (default: 0.01 for microarrays, 1e-10 for sequencing data). Finding the optimal parameters is often an iterative process of performing the segmentation and plotting results with the tool **Plot copy number profiles** for evaluation.

Tool parameters also let you specify the number of chromosomes (for microarrays, this can be 24 in case of sex-matched reference samples or 23 otherwise; for sequencing data, usually 22).

Segmentation is performed using the DNACopy R package (5), which implements the circular binary segmentation (CBS) algorithm.

### 2.2 Calling gains and losses

The next step is to detect copy number aberrations, *i.e.* gains and losses. Sometimes higher-level amplifications and homozygous deletions can also be separated from gains and losses. To do this in Chipster, use the tool **Call copy number aberrations from segmented copy number data**. The parameters let you specify the number of copy number states (3 for loss/normal/gain, 4 for loss/normal/gain/amplification, or 5 for deletion/loss/normal/gain/amplification).

Calling assigns each segment a copy number call of a loss (represented with -1), normal (0), gain (1), or optionally separating amplifications (2) and homozygous deletions (-2). These are referred to as “hard calls”. As they are determined using a probabilistic model, each call also has an underlying probability, and these probabilities can be referred to as “soft calls”. For each feature, there are therefore three (or four or five) call probabilities that add up to 100%. If the probability of (*e.g.* a loss) is over 50%, the feature is called as a loss (-1). If none of the probabilities for aberrations exceed 50%, the call is normal (0).

The output from the tool is a big table with large number of columns. Usually there is no need to deal with these manually, but for information’s sake they are as follows: columns labeled chip.\* contain the original microarray log ratios, segmented.\* contain segmented log ratios, flag.\* contain copy number calls, and probdel.\*, probloss.\*, probnorm.\*, probgain.\* and probamp.\* contain the probabilities for the specific calls. In addition, the frequencies of aberrations are shown in columns loss.freq, gain.freq, and if needed, del.freq and amp.freq. In addition to the table, a frequency plot is also produced and can be seen in Figure 2(a).

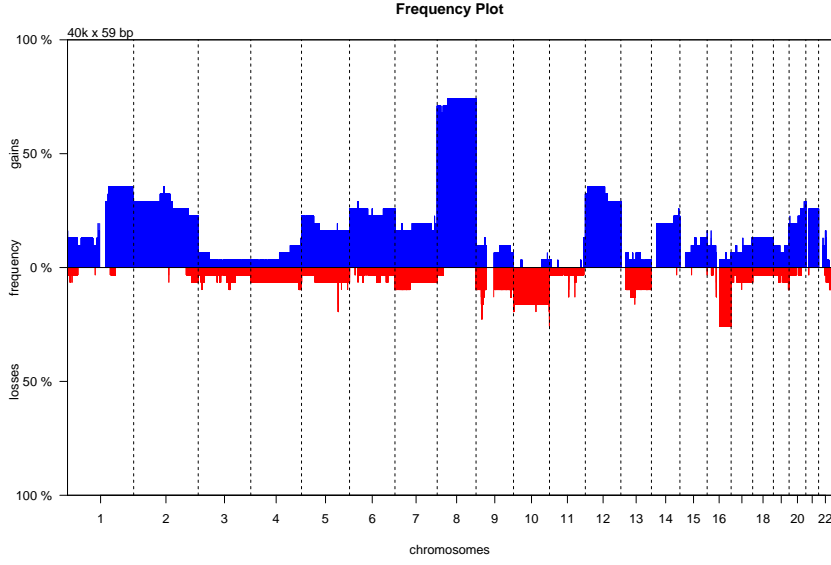
After the calling step, individual samples can be plotted with the **Plot copy number profiles** tool. Specify the number(s) of the sample(s), and chromosomes (0 means all chromosomes) to be plotted.

The implemented R packages is CGHcall (6).

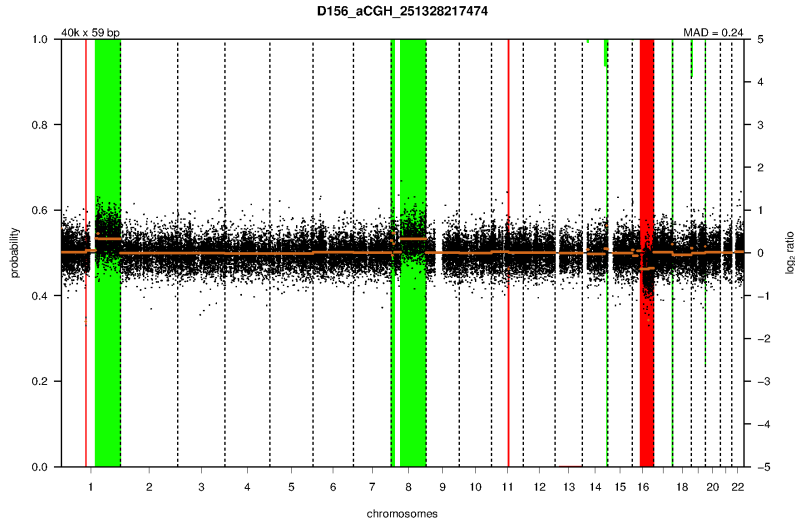
### 2.3 Identifying common regions

As copy number data typically contains long stretches of DNA without breakpoints and a shared copy number, its dimensionality can be greatly reduced after the calling. This makes the data more manageable and also reduces problems with multiple testing. In Chipster, this can be done with a tool called **Identify common regions from called copy number data**. These regions are what should be used for downstream analysis steps such as clustering, between-group comparisons, and survival analysis.

The output is a condensed table containing the same columns as in the input file, and also an additional one containing the number of features within each region. At this stage, the number of rows is usually also manageable, so that it is possible to order the table according to *e.g.* loss.freq or gain.freq



(a) frequency plot



(b) individual profile

**Figure 2:** Plots of DNA copy number. For both plots, chromosomes are along the x-axis. a) A frequency plot of all samples is generated with **Call copy number aberrations from segmented copy number data** and shows the frequencies of gains (blue) and losses (red) in the data set. b) A plot of on individual sample, produced with **Plot copy number profiles**. Original log ratios are shown in black and segmented log ratios in brown. The probabilities of losses are shown with red bars, and the values can be read directly from the y-axis. Probabilities of gains are shown in green, and the values can be read as  $1 - \text{the value on the y-axis}$ . Possible amplifications and deletions are shown with tick marks on the top and bottom edges. This plot can also be drawn for a subset of chromosomes.

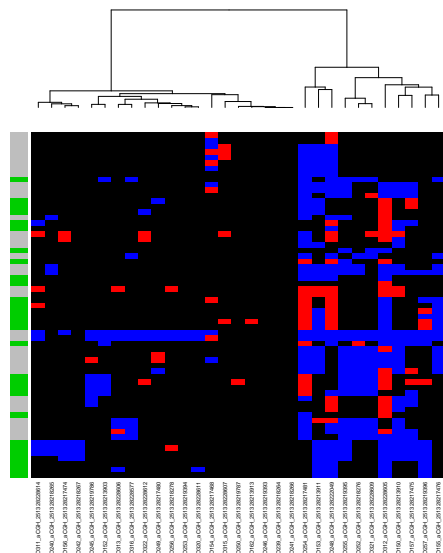
to see where the most frequent aberrations are. To also include information about karyotype bands, run the tool **Add cytogenetic bands**. In addition to the table, a frequency plot is produced.

The corresponding R package is called CGHregions (7).

## 2.4 Clustering

Using methods developed for expression data to cluster copy number samples does not yield optimal results. Therefore there is a separate tool for this purpose: **Cluster called copy number data**. It should be run after identifying the common regions. Otherwise meaningless long stretches of DNA without breakpoints will have more weight on the clustering than small, possibly very important, aberrations, as the long regions contain a larger number of features. Identifying the common regions compresses these regions into individual data points making the clustering more dependent on the actual differences between the samples.

Clustering can be performed both with hard or soft calls. Generally soft calls are recommended, as they not only include the hard calls, but also additional information about the reliability of these calls. The option to cluster using hard calls is provided mainly just for situations when soft calls are not available. In case you have analyzed your data with the **Call copy number aberrations from segmented copy number data** and **Identify common regions from called copy number data** tools, you will always have the soft calls available. Figure 3 shows an example of clustering results.



**Figure 3:** Clustering of samples, produced with **Cluster** called **copy number data**. Clustering using soft calls produces more reliable results and better shows the distances between samples than hard calls.

The implemented R package is WECCA (8).

## 2.5 Known copy number variations

The tool **Count overlapping CNVs** downloads a list of known copy number variations (CNVs) from the Database of Genomic Variants (9) and appends two new columns to the data set: `cnv.count` and `cnv.proportion`. The first one is a raw count of how many entries there are in the database that overlap with the area of interest (can be *probes/bins*, *regions*, or *genes*). The latter one is the proportion of overlap of the feature with known CNVs in the database. Value of 1 means complete overlap.

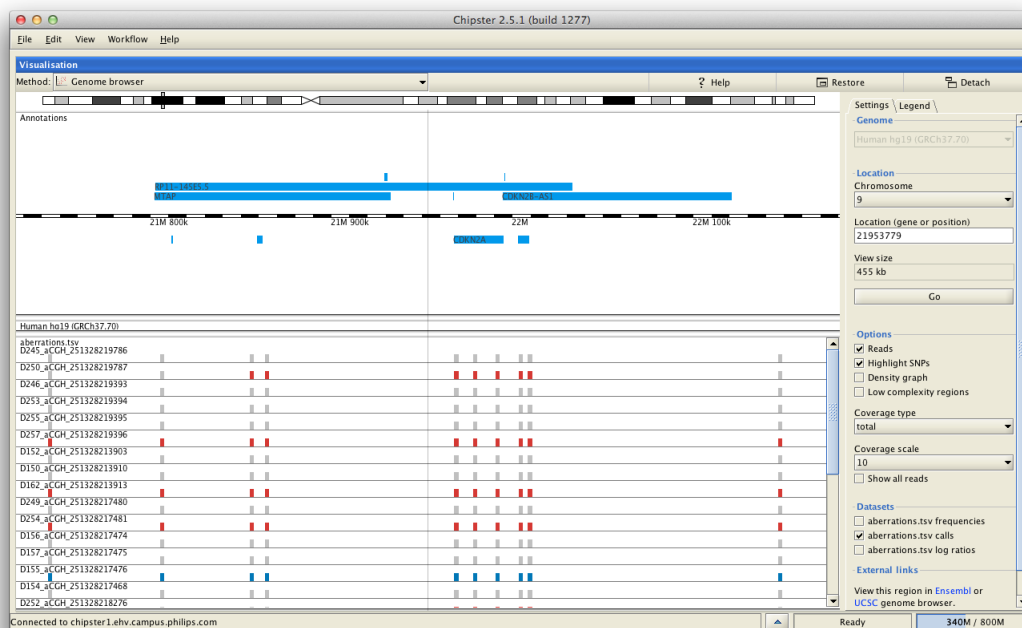
To evaluate the distribution of the values across the entire genome, run the tool **Statistics / Calculate descriptive statistics** and specify “chips” for the parameter `calculate.descriptives.for`.

## 2.6 From features to genes

In order to be able identify enriched Gene Ontology categories among gained/lost genes, we need to know the copy number of each gene. For this, we can use the **Detect genes from called copy number data** tool, which works as follows. First, the list of human genes is downloaded from the Ensembl database (10). Then for each gene, it is checked whether there are features that overlap the position of the gene. If yes, these feature(s) are used to derive the copy number call for this particular gene. If no, the last feature preceding and first one tailing the gene are used. Tool parameters can be used to choose between two methods for deriving the copy number call: “majority” means that in order to call the gene *e.g.* gained, more than 50% of the features in question have to show a gain. If “unambiguous” is chosen, the copy number of the gene is called as normal unless every one of the features gives the same aberrant call.

## 2.7 Genome browser

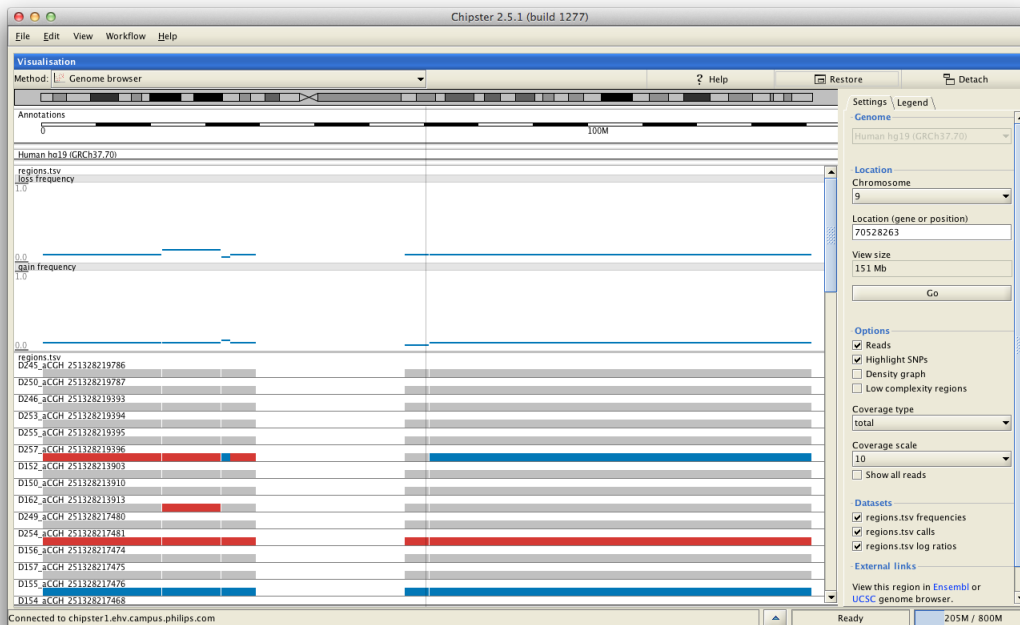
Copy number data can be visualized with the integrated Genome browser in three ways: 1) frequencies of gains and losses in the data set as scatterplots along the chromosomes, 2) gains and losses as horizontal bars for each sample, and 3) segmented copy number data as scatterplots per sample. First and second category require called data and are possible for the original features (probes on microarrays or bins for NGS experiments; from tool **Call aberrations from segmented copy number data**), common regions (**Identify common regions from called copy number data**), or genes (**Detect genes from called copy number data**). As the third category requires only segmented (and not necessarily called) data, output from tool **Segment copy number data** is also an option.



**Figure 4:** Output from tool **Call aberrations from segmented copy number data** visualized in the Genome browser. The view is zoomed in around the CDKN2A gene in chromosome 9p21.3. Blue and red bars show microarray probes that have been called as gained or lost, respectively. Gray bars depict normal copy number. All samples in the data set are shown on their own line.



To view your data in the Genome browser, first select to appropriate file and then choose “Genome browser” from the Method pop-up menu in the Visualization section of the main Chipster window. After selecting the correct genome, *e.g.* Human hg19 (GRCh37.70), you can either choose a chromosome from the pop-up menu or type a gene name in the search box. Then press Go. If choosing to visualize a chromosome, please note that by default Chipster will zoom in to a 100 kbp around position 1,000,000 bp. To see the entire chromosome, you can zoom out with the scroll wheel of the mouse.



**Figure 5:** Genome browser view of chromosome 9 of regions from tool **Identify common regions from called copy number data**. The colored bars depict gains, losses, and normal copy number with blue, red, and gray, respectively. They are shown separately for each sample in the data set. Above the colored bars are gain and loss frequencies shown as scatterplots.

## 3 Additional analysis steps

### 3.1 Plotting copy number profiles

The tool **Plot copy number profiles** can be used to plot profiles for individual samples or the whole data set. Depending on whether the dataset has been segmented and/or called, the output of this tool varies. Data for the individual features will always be plotted in black. If the data has been segmented, segments will be shown in brown. And if it has also been called, call probabilities will be shown with colored bars. The probabilities of losses are shown with red bars, and the values can be read directly from the y-axis. Probabilities of gains are shown in green, and the values can be read as  $1 - \text{the value on the y-axis}$ . Possible amplifications and deletions are shown with tick marks on the top and bottom edges. The plot can also be drawn for a subset of chromosomes. An example of a profile with calls is shown in Figure 2(b).

### 3.2 Removing wavy artifacts from aCGH profiles

aCGH profiles sometimes contain a technical, wavy artifact (11). When analyzing cancer samples, it is possible to remove the effect of these waves by using clinical genetics samples as calibration data, as they are not expected to contain large aberrations. Preferably the calibration data should be measured with the same array platform as the data to be analyzed. Smoothing the waves generally leads to more accurate calling and improved reliability. The effects can be seen in Figure 6.

One important note about using the tool is that while selecting the two normalized data sets, first click on the cancer data, then on the calibration set. Otherwise Chipster will try to do it the wrong way.

The name of the implemented R package is NoWaves (12).

### 3.3 Comparisons between groups

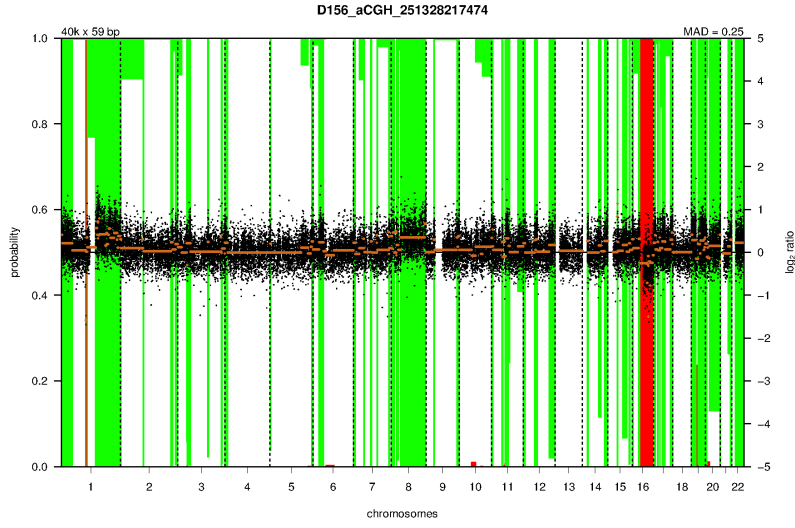
If your data set contains two or more groups, aberration frequencies can be compared visually with the tool **Plot copy number aberration frequencies**. And statistical testing for between-group differences can be performed using the **Group tests for called copy number data** tool. It should normally be run on *regions* (*i.e.* results from the **Identify common regions from called copy number data** tool), but can also be run on *probe/bin* or *gene*-based data as well, although running times are likely to be prohibitively long. A test statistic (either Chi-square, Wilcoxon or Kruskal-Wallis) is calculated for each region. As the distribution of the test statistic might be really skewed, significance is evaluated with a permutation-based approach instead of simple multiple testing correction. The group labels for individual arrays are randomly sampled, and the test statistics calculated for each repetition. Finally, p-values and false discovery rates (FDR) are calculated for each region based on how frequently test statistics as extreme as the calculated one were observed during the permutations. The number of permutations to run can be set in the tool parameters. The larger the number, the longer the execution takes. For final analysis, at least 10,000 permutations are recommended, but as this can take very long, it is good to first test with low values.

The implemented R package is CGHtest, which is an updated version of CGHMultiArray (13).

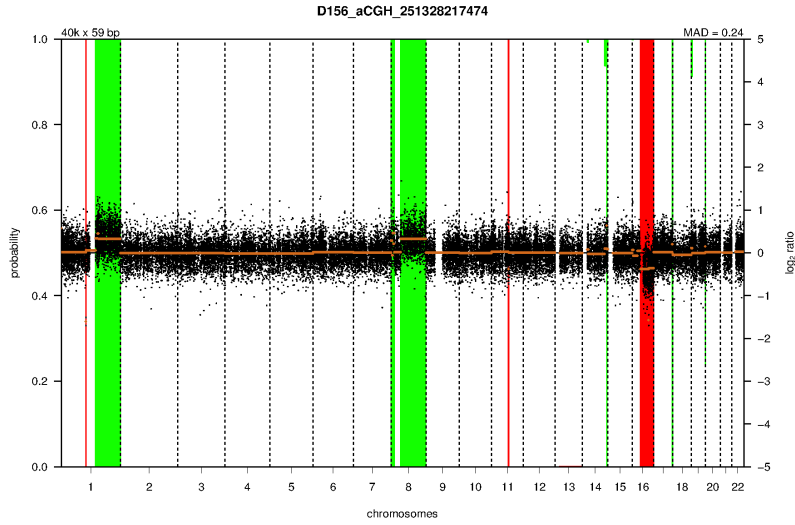
### 3.4 Survival analysis

When data for survival is available and entered into the phenodata file as two variables: one for the survival time and the second indicating patient status (1 means the patient is deceased, 0 lost to follow-up), survival analysis can be run with the tool **Survival test for called copy number data**. As with the group tests, it should be done on *regions* (*i.e.* results from the **Identify common regions from called copy number data** tool). Calculations are done using a permutation test, and the number of permutations can be set in the tool parameters. The larger the number, the longer the execution takes. For final analysis, at least 10,000 permutations are recommended, but as this can take very long, it is good to first test with low values. The tool will return a p-value and FDR for each region.

Tool **Plot survival curves for called copy number data** can be used to plot Kaplan-Meier curves, but one curve will be plotted for each row in the input file. It is therefore recommended that after running the **Survival test for called copy number data** tool, to look at the resulting output file ordered



(a) original profile plot



(b) smoothed profile plot

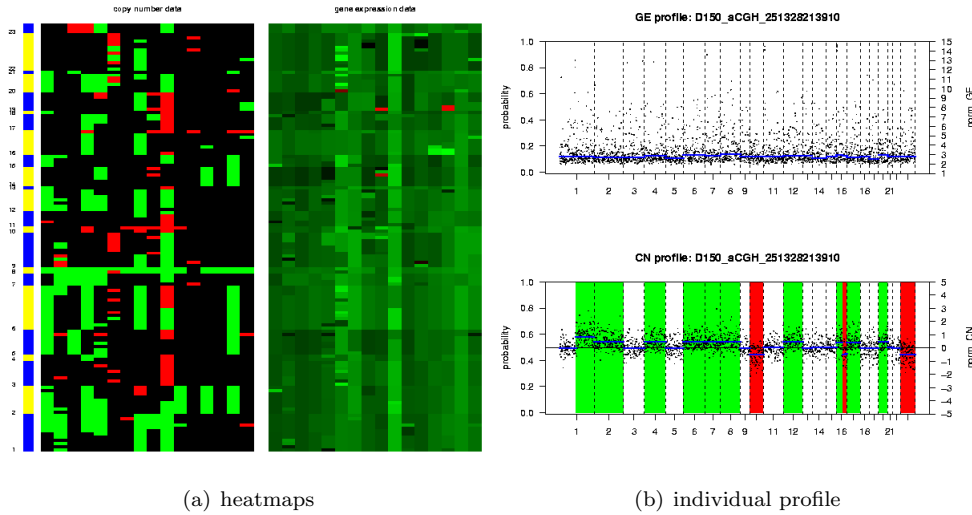
**Figure 6:** The effect of dewaving with **Copy number aberrations / Smooth waves from normalized aCGH data**. Profile plots of an individual sample are shown both for the a) original and b) smoothed data. Dewaving generally results in more confident calling (more probabilities close to 0% or 100%, instead of being around 50%).

by p-value/FDR and decide on a cutoff for which regions should be plotted. Tool **Preprocessing / Filter using a column value** can then be used to filter for only those regions, and plotting done on the resulting subset.

The implemented R package is CGHtest, which is an updated version of CGHMultiArray (13).

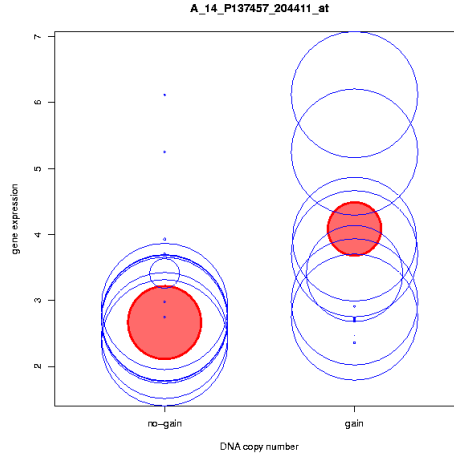
### 3.5 Integration with expression

Integrating copy number and expression data together is multi-step process involving four separate tools. The relationships between these tools are outlined in Figure 11. The first step is to run **Match copy number and expression features**, which takes two input files: the output of **Call copy number aberrations from segmented copy number data** and a normalized and filtered expression data set. To be able to pair the samples of the two data sets, the accompanying phenodata tables must have columns that contain common identifiers unambiguously identifying the pairs. When importing data from CanGEM, this is usually a column called Sample. The output is a table of matched features, and a plot showing heatmaps of both data sets (Figure 7(a)). This file can also be used to plot profiles of individual samples with **Plot profiles of matched copy number and expression**. Parameters allow the user to specify sample(s) and chromosome(s) to be plotted. The produced image (see Figure 7(b)) contains a copy number profile plot similar to Figure 2(b) and another plot showing expression levels.



**Figure 7:** a) Heatmaps of matched copy number and expression data, generated with **Match copy number and expression features**. The copy number data is on the left and samples are shown in the same order as in the expression heatmap on the right. Chromosomes are shown along the y-axis. b) Copy number and expression profiles of an individual sample, produced with **Plot profiles of matched copy number and expression**. Copy number data is shown on the bottom (for interpretation see Figure 2(b)), and expression profile on top. Expression levels of individual genes are shown with black dots, and blue lines show the mean expression levels of genes within *regions* defined by the copy number data.

To test the statistical significance of copy number changes on expression levels, run the **Test for copy-number-induced expression changes** tool. It divides samples into two groups for each expression feature based on the aberration profile for that particular feature. The comparison is either between ‘loss *vs.* no-loss (normals, gains and amplifications)’ or between ‘no-gain (losses, normals) *vs.* gain (gains and amplifications)’. Statistical testing is performed using a permutation test, and the tool parameters let the user specify how many permutations to run. 10,000 are recommended for final analysis, but take a long time. The resulting p-values can be found in the adj.p column of the resulting output table. Also contained within this file is a column labeled as gene.id, which contains IDs that are needed to plot visual representations of individual genes with the **Plot copy-number-induced gene expression** tool.



**Figure 8:** A plot with matched copy number and expression data. The title shows the names of the copy number (A\_14\_P137457) and expression features (204411\_at). Based on the observed aberration frequencies, the test has been performed by comparing a “no-gain” group of samples (losses and normals) *vs.* a “gain” group of samples (gains and amplifications), as shown by the labels at the bottom. Expression levels of individual samples are shown with blue circles and the scale is along the y-axis. The radius of the circle represents the probability of the corresponding call. Each sample is therefore plotted on both columns, but using circles with different radii. Red circles represent mean values. This particular case had an adjusted p-value of 0.27.

The integration of copy number and expression data sets is implemented with the intCNGEan R package (14).

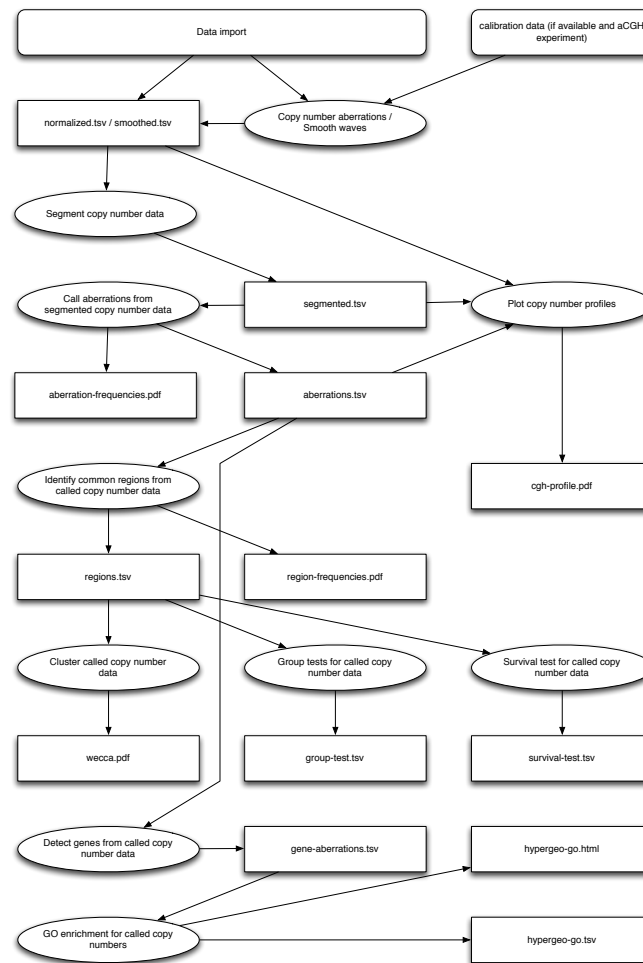
### 3.6 Enriched Gene Ontology categories

After the copy number data set has been converted from *probe/bin* to *gene*-based, the tool **GO enrichment for called copy numbers** can be used to detect Gene Ontology categories enriched among frequently aberrated genes. The user can choose to pick only genes that are frequently lost, gained or amplified, or combine all aberrations together (default). The minimum frequency of aberrations can also be specified (default is 50%). Genes showing more frequent aberrations than the threshold are then picked as the test list, and a hypergeometric test performed to see if certain Gene Ontology categories are enriched. The entire gene list is used as the reference. It should therefore be an unfiltered list, *i.e.* the direct output from **Detect genes from called copy number data**.

The rest of the parameters are the same as for the corresponding expression tools.

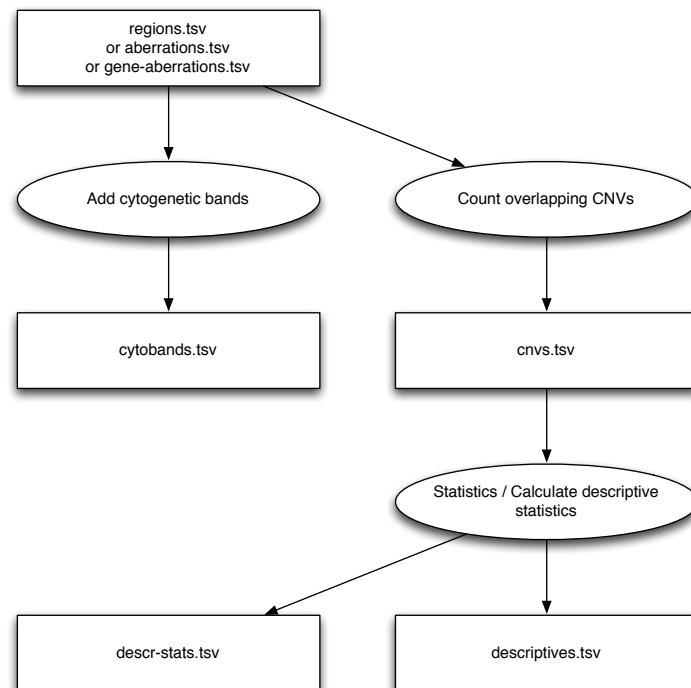
## 4 Workflow diagrams

### 4.1 Main copy number tools



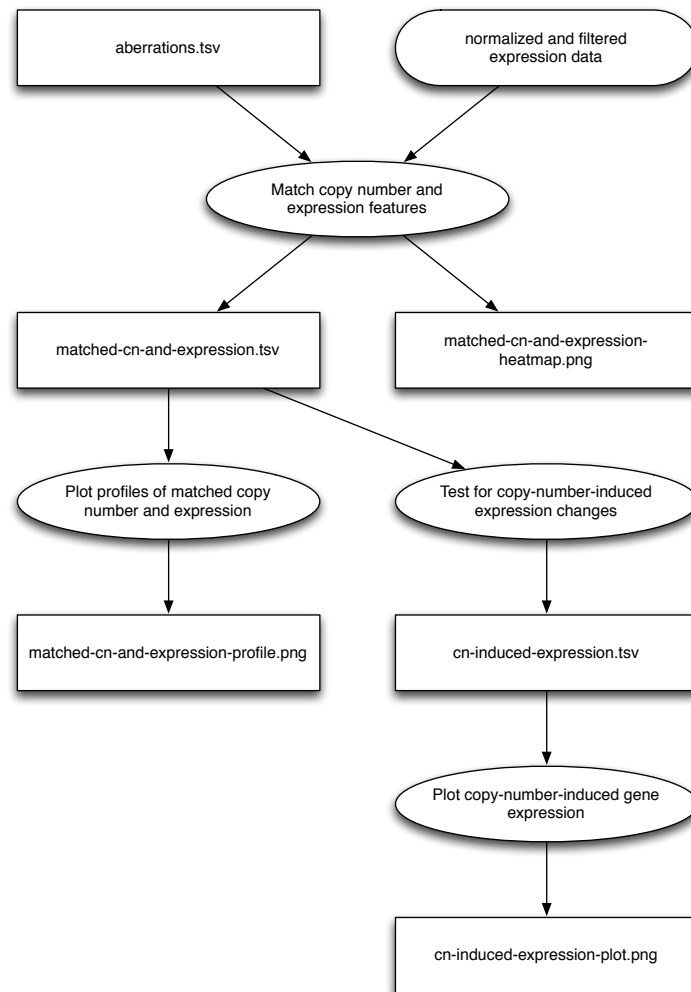
**Figure 9:** A diagram showing the order in which the copy number tools should be executed.

## 4.2 Copy number annotation tools



**Figure 10:** A diagram showing a typical use case of copy number annotation tools.

### 4.3 Tools for integrating copy number and expression data



**Figure 11:** A diagram showing how the different tools involved in integrating copy number and expression data are related to each other.



## References

- [1] T. Barrett, S. E. Wilhite, P. Ledoux, C. Evangelista, I. F. Kim, M. Tomashevsky, K. A. Marshall, K. H. Phillippy, P. M. Sherman, M. Holko, A. Yefanov, H. Lee, N. Zhang, C. L. Robertson, N. Serova, S. Davis, and A. Soboleva. Ncbi geo: archive for functional genomics data sets—update. *Nucleic Acids Res*, 41(Database issue):D991–5, Jan 2013.
- [2] I. Scheinin, S. Myllykangas, I. Borze, T. Bohling, S. Knuutila, and J. Saharinen. CanGEM: mining gene copy number changes in cancer. *Nucleic Acids Res*, 36(Database issue):D830–D835, 2008.
- [3] Z. Zhang, S. Schwartz, L. Wagner, and W. Miller. A greedy algorithm for aligning DNA sequences. *J Comput Biol*, 7(1-2):203–214, 2000.
- [4] H. Li and R. Durbin. Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics*, 25(14):1754–1760, Jul 2009.
- [5] E. S. Venkatraman and A. B. Olshen. A faster circular binary segmentation algorithm for the analysis of array CGH data. *Bioinformatics*, 23(6):657–663, Mar 2007.
- [6] M. A. van de Wiel, K. I. Kim, S. J. Vosse, W. N. van Wieringen, S. M. Wilting, and B. Ylstra. CGHcall: calling aberrations for array CGH tumor profiles. *Bioinformatics*, 23(7):892–894, 2007.
- [7] M. A. van de Wiel and W. N. van Wieringen. CGHregions: dimension reduction for array CGH data with minimal information loss. *Cancer Informatics*, 3:55–63, 2007.
- [8] W. N. van Wieringen, M. A. van de Wiel, and B. Ylstra. Weighted clustering of called array CGH data. *Biostatistics*, 9(3):484–500, Jul 2008.
- [9] A. J. Iafrate, L. Feuk, M. N. Rivera, M. L. Listewnik, P. K. Donahoe, Y. Qi, S. W. Scherer, and C. Lee. Detection of large-scale variation in the human genome. *Nat Genet*, 36(9):949–951, 2004.
- [10] P. Flicek, B. L. Aken, B. Ballester, K. Beal, E. Bragin, S. Brent, Y. Chen, P. Clapham, G. Coates, S. Fairley, S. Fitzgerald, J. Fernandez-Banet, L. Gordon, S. Gräf, S. Haider, M. Hammond, K. Howe, A. Jenkinson, N. Johnson, A. Kähäri, D. Keefe, S. Keenan, R. Kinsella, F. Kokocinski, G. Koscielny, E. Kulesha, D. Lawson, I. Longden, T. Massingham, W. McLaren, K. Megy, B. Overduin, B. Pritchard, D. Rios, M. Ruffier, M. Schuster, G. Slater, D. Smedley, G. Spudich, Y. A. Tang, S. Trevanion, A. Vilella, J. Vogel, S. White, S. P. Wilder, A. Zadissa, E. Birney, F. Cunningham, I. Dunham, R. Durbin, X. M. Fernández-Suarez, J. Herrero, T. J. P. Hubbard, A. Parker, G. Proctor, J. Smith, and S. M. J. Searle. Ensembl’s 10th year. *Nucleic Acids Res*, 38(Database issue):D557–62, Jan 2010.
- [11] J. C. Marioni, N. P. Thorne, A. Valsesia, T. Fitzgerald, R. Redon, H. Fiegler, T. D. Andrews, B. E. Stranger, A. G. Lynch, E. T. Dermitzakis, N. P. Carter, S. Tavaré, and M. E. Hurles. Breaking the waves: improved detection of copy number variation from microarray-based comparative genomic hybridization. *Genome Biol*, 8(10):R228, 2007.
- [12] M. A. van de Wiel, R. Brosens, P. H. C. Eilers, C. Kumps, G. A. Meijer, B. Menten, E. Sistermans, F. Speleman, M. E. Timmerman, and B. Ylstra. Smoothing waves in array CGH tumor profiles. *Bioinformatics*, 25(9):1099–1104, May 2009.
- [13] M. A. van de Wiel, S. J. Smeets, R. H. Brakenhoff, and B. Ylstra. CGHMultiArray: exact p-values for multi-array comparative genomic hybridization data. *Bioinformatics*, 21(14):3193–3194, 2005.
- [14] W. N. van Wieringen and M. A. van de Wiel. Nonparametric testing for DNA copy number induced differential mRNA gene expression. *Biometrics*, 65(1):19–29, Mar 2009.