aCGH-MAS: Analysis of aCGH by Means of Multi-agent System

Juan F. De Paz[1], Rocío Benito[2], Javier Bajo[1], Ana Rodríguez-Vicente[2], María Abáigar[2]

[1]Department of Computer Science, University of Salamanca, Address Plaza de la Merced s/n. Salamanca, Spain

{fcofds, jbajope}@usal.es

[2] IBMCC, Cancer Research Center, University of Salamanca-CSIC, Spain

{beniroc, anita82, mymary}@usal.es

** Corresponding Author: Juan F. De Paz Santana*

Faculty of Computer Sciences

Univ. Salamanca, Plaza de la Merced, s/n, 37008, Salamanca, Spain

Tel. + 34 923294400 Fax. + 34 923 294514

fcofds@usal.es

Abstract. There are currently different techniques, such as CGH Arrays, to study genetic variations in patients. Arrays CGH analyze gains and losses in different regions in the chromosomal. Regions with gains or losses in pathologies are important for selecting relevant genes, or CNVs (copy-number variations) associated to the variations detected within chromosomes. Information corresponding to mutations, genes, proteins, variations, CNVs and diseases can be found in different databases and it would be of interest to incorporate information of different sources to extract relevant information.. This work proposes a multi-agent to manage the information of aCGH arrays, with the aim of providing an intuitive and extensible system to analyze and interpret the results, . The agent roles integrate statistical techniques to select relevant variations and visualization techniques for the interpretation of the final results, and to extract relevant information from different sources of information by applying a CBR system.

Keywords: arrays CGH; knowledge extraction; visualization; CBR system

1 Introduction

There are various techniques for performing studies on genetic variations in patients, including expression arrays [6] [10], CGH (Comparative Genomic Hybridization) Arrays [45], and studies at the genetic sequence level. CGH arrays (aCGH) allow comparing the DNA of a patient with a control DNA and using this information to detect mutations [35] [30] based on gains, losses and amplifications [41]. Another kind of microarrays are the Expression arrays , which determine the expression of different genes with probes. aCGH are used to detect regions in the chromosomes with variations in certain pathologies. This information is taken into to account for sequencing these regions through the use of expression arrays and sequencers [3]. In these studies, the users have to work with a vast amount of information, which implies the development of systems oriented to improve the analysis of the data and to automatically extract information using databases [51]. For this reason, it is necessary to identify the exact location of those interesting genes in aCGH arrays before carrying out the sequencing.

There are currently various tools that provide a visual analysis of the information of aCGH. These tools typically represent the information but the interaction with the information is complex. The visual analysis is used to represent additional information about relevant regions. Some of these tools can be found in works [29] [25] [4] [43] [33] [37]. A visual analysis of these data is normally performed manually [37][48], which requires the participation of experts to select the relevant information. However, these tools lack usability and require the use of techniques that facilitate the automatic analysis and extraction of information from different sources. For this reason, it is necessary to incorporate a process that helps determine the interesting genes[50], proteins and relationships to diseases that must be analyzed and understood in a simpler way.

The distributed analysis of CGH data is performed by different laboratory personnel, from hybridating the chips to extracting the relevant variations and information associated to the chips. This work shows a multi-agent system specifically designed to analyze CGH data [2]. The funcionality of the multi-agent system is divided into layers and roles to carry out the analysis of aCGH arrays. The analysis is usually composed of several stages. The first stage is the segmentation process [39], which implements the subsequent analysis of the data

and is important to be able to represent a visualization of the data. The remaining stages depend on the analysis to be performed and include: clustering, classification, visualization, or extraction of information from databases. The proposed multi-agent system manages the analysis and the automatic interpretation of the data. The system can select the relevant genes and transcripts for the prior classification of pathologies. The information of the identified genes is obtained from public databases. The information management system is based on the CBR (Case-based reasoning) model [7][11] to detect the mutations, genes, proteins and diseases. Finally, visualization assists the user in reviewing the results.

This article is organized as follows: section 2 describes the state of the art in CGH arrays, section 3 describes the proposal, and section 4 presents the results and conclusions.

2 CGH Arrays

Array-based comparative genomic hybridization (aCGH) is a kind of microarray that analyzes areas of the genome to detect gains or losses. . Whereas traditional high-resolution chromosome analysis detects chromosome structure alterations at a resolution of 5 megabases (Mb) or greater, aCGH detects gains or losses of DNA that cannot be seen by traditional karyotyping and may sometimes be only thousands of base pairs in size [15]. aCGH has emerged as a powerful diagnostic technique for high resolution analysis of the human genome. It is a specific, sensitive, and rapid technique that can detect genomic arrangements and copy number changes. A variety of array CGH platforms are currently available, both commercially and in academic institutions. The choice of platform may depend on the type of data sought; however, the price, reproducibility, and standardization are crucial factors that need to be considered [17].

aCGH arrays incorporate segments of DNA, that are defined with genome databases. The clones are predominantly selected to target areas of the human genome that, when deleted or duplicated, are known or highly suspected to cause well-characterized genetic defects. Microarray printers attach the clones to a glass slide in an organized way to form a microarray. A typical microarray slide contains thousands of different clones representing targeted areas of the genome.

Fluorescently labeled DNA from both the patient and a known normal human control are applied to the slide and compete to attach or hybridize to their corresponding DNA segments. The fluorescent signals are analyzed and, depending of the values obtained, it is possible to detect areas with unequal hybridization of a patient versus control DNA.

The first whole genome microarray contains 2,400 large-insert genomic clones, primarily bacterial artificial chromosomes (BACs). With the total human genome covering approximately 3,000 Mb, the resolution of this array is on average close to 1 Mb, about one order of magnitude higher than that obtained with classical CGH [34]. For a full coverage resolution array, approximately 30,000 BACs have been arrayed [22], increasing the resolution with another order of magnitude. However, producing such large numbers of BACs for array CGH is expensive and time-consuming and, due to the large size of the BACs, the limits of BAC array CGH resolution have been reached. These problems can be overcome when oligonucleotides are used as targets in microarray experiments. Oligonucleotides allow a sheer infinite resolution, great flexibility, and are cost-effective [46]. Moreover, oligonucleotides allow for the generation of microarrays for any organism for which the genome has been sequenced. Attempts have been undertaken to increase the resolution of BAC arrays in other ways, but CGH cannot compete with the flexibility and versatility of oligonucleotides. Finally, oligonucleotide arrays are being used, designed and accepted for expression profiling and are thus widely available [16].

In conclusion, BAC arrays (array-based comparative genomic hybridization) have proved to be successful for the detection of submicroscopic DNA copy-number variations. Technological improvements to achieve a higher resolution have resulted in the generation of additional microarray platforms encompassing larger numbers of shorter DNA targets (oligonucleotides). Currently, both types (BAC and oligo arrays) have advantages and disadvantages. The BAC clone targets have been mapped to the human reference sequence produced by the International Human Genome Sequencing Consortium, allowing easy access to information in the related genomic databases. However, BACs, which are usually 100-200 kb, may miss alterations smaller than the size of a clone but are less likely to detect alterations of unclear clinical significance. For this reason, they are unable to distinguish deletions/amplification less than approximately 85 kb.

Oligonucleotides, which are much smaller probes, usually 25-60 bp, may detect small alterations that would not be seen using a BAC microarray, but oligonucleotide arrays are more likely to detect small alterations of unclear clinical significance.

3 Multi-agent System

The multi-agent system is composed of three layers: analysis, information management and visualization. Figure 2 shows the multiagent system architecture and the layers it comprises. The analysis layer performs the microarray analysis. It includes several algorithms that can be applied to the specific case study taken into consideration. The information management layer generates a local database using the information of several sources. The visualization layer manages the information and the algorithms. tIt displays the information and the results obtained after applying the existing algorithms at the analysis layer.

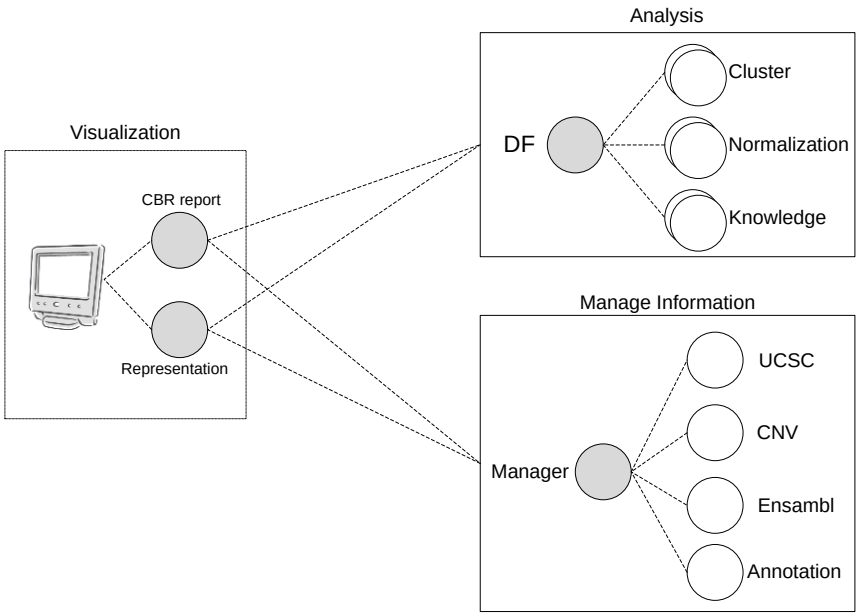


Figure 1 Multiagent system architecture

The analysis layer contains the DF (Directory Facilitator) agent, which registers the different types of agents that are contained in that layer and the services they provide. Each agent provides a set of services to perform certain functionalities that can be requested by other agents. This separation allows the inclusion of new functionalities in the application by modifying the services provided by the agents at the analysis layer. The

The information management layer contains an agent for each external source of information. Each agent is responsible for retrieving the information requested by the manager agent, which compiles the information from the agents and generates a local database.

The agents at the visualization layer consult with the DF to obtain the services provided for each existing process. Additionally, these agents can contact the manager in the information management layer to obtain relevant information that appears during the analysis.

The following subsections describe each one of the layers adapted to the case study for CGH arrays.

3.1 Analysis layer

The agents in these layers perform different tasks for processing information, specifically, the processing tasks that will be used in the case study and the chips. It is necessary to take into account that the algorithms can be adapted to each case study. In the particular case presented in this work, this layer contains agents for Normalization and segmentation, Knowledge extraction, and Clustering.

3.1.1 Normalization and Segmentation

During this process the data are preprocessed in order to segment them and reduce noise. This state is important to represent the information and extract relevant regions. While there are many algorithms capable of carrying out the preprocessing, the package snapCGH [39] R Server was used in this tool to normalize and segment the data. The package incorporates algorithms as aCGH, DNACopy, GLAD [42] [18]. It is important to use algorithms in order to compare the different results; additionaly, this package is widely used.

In order to compare all of the arrays simultaneously, the value is readjusted for all data that were previously processed by NimbleGen, which normalizes and segments each of the arrays. The data are then scaled according to the mad1dr (median absolute deviation) provided by NimbleGen. The process is defined according to the function (1) :

$$f(x,m,tl,tg)=\begin{cases} f_a(x,m,tl) & x \le 0 \\ f_b(x,m,tg) & x > 0 \end{cases} \qquad (1)$$

$$f_a(x,m,tl)=\begin{cases} f_1(x,m,tl) & f_l(x,m,tl)<0 \\ 0 & f_l(x,m,tl)\geq 0 \end{cases}$$

$$f_b(x,m,tl)=\begin{cases} f_g(x,m,tl) & f_g(x,m,tl)<0 \\ 0 & f_g(x,m,tl)\geq 0 \end{cases} \quad (2)$$

$$f_l(x,m,tl)=x-(-tl-m)-(-m-x)\,|-tl-m|/m$$

$$f_g(x,m,tg)=x+(tg-m)-(m-x)\,|tg-m|/m \quad (3)$$

where:

- x represents the value of the segment
- m is the value of the mad1dr for the given array
- tl is the loss threshold
- tg is the gain threshold

3.1.2 Knowledge Extraction and classifiers

There are currently several kinds of classifiers based on different technologies: decision rules and decision trees RIPPER [5], One-R [19], M5 [20], J48 [36], CART [6] (Classification and Regression Trees); probabilistic models naive Bayes [12], fuzzy models K-NN (K-Nearest Neighbors) [1], neural networks [8] etc. Some of these classifiers can be used to extract relevant information in order to obtain attributes, and this process can be carried out by traditional statistical techniques to compare values using the parametric or not parametric test ANOVA [9], Kruskal-Wallis [28] and Mann-Whitney U-test [47], or testing to compare the frequencies as parametric Chi Squared [24] or fisher exact test. The gain functions are a particular case of the techniques used in decision trees and decision rules for selecting the attributes, which is why they are not considered separately.

For this particular system, the decision trees were chosen to select the main genes of the most important pathologies, specifically the J48 [36] in its implementation for Weka [21]. However, if the system needs a generic selection, a gain functions or statistical test are chosen (specifically Chi Squared).

Chi squared [24] was selected because it can work with qualitative and nominal variables and it provides an easy way to select relevant regions depending on a p-

value. Fisher's Exact test [44] [13] is applied, which is the recommended method when the sample size is small and it is not possible to ensure that 80% of the expected frequency from a contingency table have a value greater than 5. Figure 2 displays the information used in the contingency tables to carry out the statistical tests.

	Gain	**Normal**	**Loss**	**Total**
Pathology	f_{11}	f_{12}	f_{13}	$f_{1.}$=Total pathology
Not Pathology	f_{21}	f_{22}	f_{23}	$f_{2.}$=Total not pathology
Total	$f_{.1}$=Total gains	$f_{.2}$=Total normals	$f_{.3}$=Total losses	$f_{..}$=total

Figure 2 Contingency table

The segments that were considered most relevant for each of the CGH arrays were selected for each pathology. Figure 2 displays the selection algorithm for the relevant segments used for the set of arrays and for the individuals with or without a particular pathology, as identified by the groups variable. The algorithm was applied repeatedly for each existing pathology.

Input: CGHArrays, Groups
Output: RelevantSegments

$RelevantSegments \leftarrow \emptyset$

foreach $segment \in CGHArrays/Groups\$pathologie$ **do**
  $ct \leftarrow calculateContingencyTable(segment, Groups, CGHArray)$;
  // Chi Squared test
  if *80% of the data from a* $ct > 5$ **then**
    $\chi^2 \leftarrow \sum_{i,j} \frac{f_{ij} - f_{i\cdot} * f_{\cdot j}/f_{\cdot\cdot}}{f_{i\cdot} * f_{\cdot j}/f_{\cdot\cdot}}$;
    $df \leftarrow (row(ct) - 1)(col(ct) - 1)$;
    p-value $\leftarrow calculatePValue(\chi^2, df)$;
  end
  // Fisher's Exact test
  else
    p-value $\leftarrow \frac{\binom{f_{11}+f_{21}}{f_{11}}\binom{f_{12}+f_{22}}{f_{12}}\binom{f_{13}+f_{23}}{f_{13}}}{\binom{f_{\cdot\cdot}}{f_{11}+f_{12}+f_{13}}}$;
  end
  // H0: pathology and the segments are independents
  if *p-value* < 0.05 **then**
    $RelevantSegments \leftarrow RelevantSegments \cup segment$;
  end
end

Figure 3 Automatic selection of segments

3.1.3 Cluster

There is a wide range of possibilities in data mining. Some of these techniques are artificial neural networks such as SOM [26] (self-organizing map), GNG [14] (Growing neural Gas) resulting from the union of techniques CHL [32] (Competitive Hebbian Learning) and NG [31] (neural gas), GCS [14] (Growing Cell Structure). There are other techniques with fewer computational costs that provide efficient results. Among them we can find the dendrogram and the PAM method [23] (Partitioning Around Medoids). A dendrogram [38] is an ascendant hierarchical method with a graphical representation that facilitates the interpretation of results and allows an easy way to establish groups without prior establishment. The PAM method requires selecting the number of clusters previous to its execution.

Dendrograms are hierarchical methods that initially define conglomerates for each available case. The algorithm is modified so that each coordinate stores the values -1, 0, 1 to indicate that the segment has a loss, no variation, or a gain, respectively. At each stage the method joins the two conglomerates with the least distance and then calculates the distance of the other conglomerate to this new one. The new distances are updated in the matrix of distances. The process finishes when there is only one conglomerate remaining (agglomerative method). The distance metric used in this paper was the average linkage, a metric that calculates the average distance of each pair of nodes for the two groups and, based on these distances, mergers the groups. The metric is known as the unweighted pair group method using arithmetic averages (UPGMA) [40]. This type of cluster was selected since it allows the grouping process to be easily reviewed by visualizing the results. The following algorithm is described in Figure 4.

```
Input: CGHArrays, thresholdLost, thresholdGain
Output: RootNodeTree
CHGArraysN ← CGHArrays
foreach chgArray ∈ CGHArraysN do
    foreach value ∈ cghArray do
        if value < thresholdLost then value ← −1;
        else if value > thresholdGain then value ← 1;
        else value ← 0;
    end
end
listNodes ← calculateLeafNodes(CHGArraysN);
while listNodes.size()>1 do
    distances ← calculateDistances(listNodes);
    [nodei, nodej] ← minDistance(distances);
    // Calculate a new node with the median of nodei and nodej
    treeNode ← fusion(nodei, nodej);
    listNodes.remove({nodei,nodej});
    listNodes.add(treeNode);
end
```

Figure 4 Dendrogram algorithm

3.2 Information management

The information management layer includes a different agent for each available source of information that is managed by the Manager agent. The specific agents used are UCSC, Ensembl, CNV (copy-number variations) and Annotation [49]. These agents download existing information from the databases managed by the

Manager agent to generate the local database. They specifically download information related to genes, proteins, pathologies, genomic variants, and CNV. This information is compiled by the Manager agent, who is responsible for generating a local database that will be used as a source of information. In addition to the information retrieved from the database, the system stores the annotations created by the system experts for future data analysis.

When the information in the GUI data base requires updating, the Manager agent orders the agents to download the updated information from the remote websites. Using this information, the agent then stores the data in the system's local database in order to improve performance. There are different local data bases for the different versions of HG that are being used. The data model used for each of the data bases follows the class model shown in Figure 5.

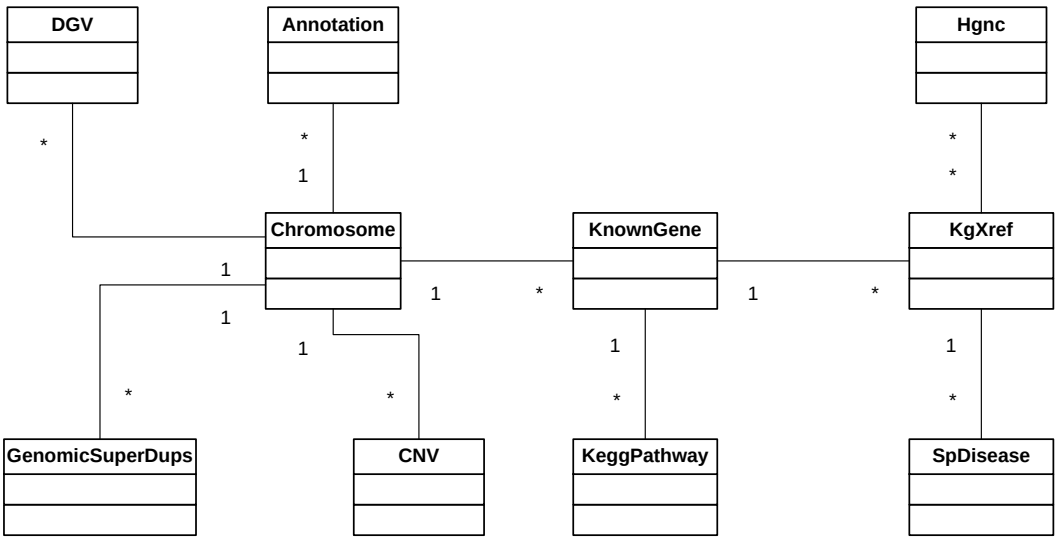


Figure 5 Class diagram with the information stored in the data bases

Although the information from the tables is downloaded from UCSC, the data model does not follow the same diagram; the information stored in the tables does, however, correspond to the information that can be found for the equivalent tables in UCSC

- DGV: Database of Genomic Variants
- Annotation: comments that are inserted into regions of the chromosomes
- Chromosome: Table with information for the chromosomes; this table only stores the chromosome identifier.
- CNV: table that stores information used to represent the copy number variations
- KnownGene: table with information about the genes

- KeggPathway: KEGG pathway cross reference
- KgXref: Link together a Known Gene ID and a gene alias; used to extract the information commonly used to identify genes.
- Hgnc: A cross-reference table between HUGO Gene Nomenclature Committee (HGNC) IDs and other database IDs.
- SpDisease: A cross-reference table between Swiss-Prot IDs and disease description.

The advantage is that all of the information is generated in a single data base and stored locally, which improves performance; additionally, new tables such as CNV can be added, or further annotations can be provided to the data base.

3.3 Visualization

aCGH is a technique to detect variations in patients who have different mutations in chromosomic regions. Usually the variations have already been catalogued, which is why the existing information can be used to catalogue and evaluate the mutation. In this case study, the cases were defined according to the segments in which the chromosomic regions have been fragmented. Therefore, in a CBR system [27], the retrieve and selection phase is adapted to get the most suitable information that solves the problem.

Cases involving stored memory correspond to the information for the region and the information associated with the region. There are cases associated with genes, pathologies, CNVs, annotations, variants and duplications. The algorithm selected for the retrieval of cases should be able to search the case base and select the genes, the known transcripts associated with the region, the variations to gains or losses etc. in the regions. The retrieved genes and transcripts are shown with each of the segments to validate the obtained results using the analysis techniques. The revise phase is carried out by an expert, and finally, the retained phase allows storing the information considered relevant. The analysis process followed by the system is shown in Figure 6

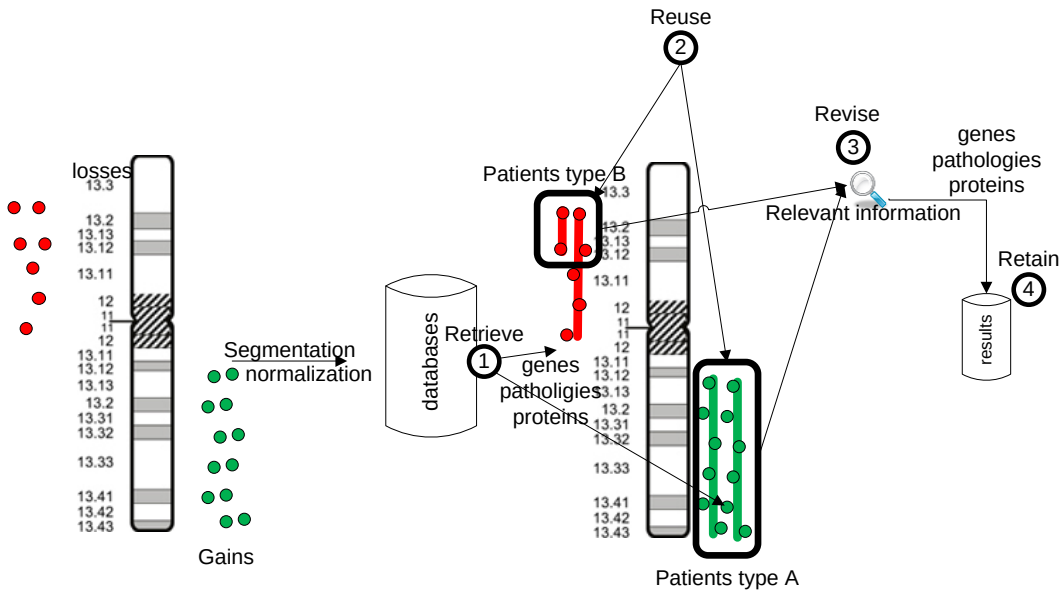


Figure 6 Chromosome 19 losses in red and gains in green

During the retrieval stage, the information previously stored by the Manager agent is retrieved from databases such as UCSC or ensembl. The retrieved information is that which is considered the most relevant and is organized in order to generate local databases, which are completed with other existing information, such as that originating from the copy-number variations.

The stored information is reused during the reuse phase in order to generate the reports, which are provided to the end user, on the analysis of the regions that stand out during a visual analysis or automatic analysis of the data. Part of this information is also used to draw regions and non-relevant mutations, which helps the subsequent revision of the selected segments as relevant.

The revision phase is carried by an expert who determines the relevance of the selected regions according to the variations and the information retrieved from the reports. The expert also inputs any annotations considered relevant regarding the detected variations, which are stored during the learning phase for future analysis of new patients. Moreover, the revision is facilitated by representing information such as the CNVs and the annotations, which eliminates the variations that are not considered relevant.

4 Results

The multi-agent systems designed in this work was applied to the study of CGH arrays. A functionality was developed to study the different types of arrays. The system was applied on three different kinds of CGH arrays: BAC aCGH, Oligo

aCGH, and SNP CGH. Although these arrays are similar, the information provided by each differs considerably because the segments are defined in different ways.

The first step in the analysis of ACGH arrays is the segmentation and normalization process, Figure 7 shows the information obtained from the BAC arrays after this step. In this kind of array all patients are are represented by the same segments, shown as rows in the image. Each segment contains information about the chromosome, initial position and final position. For each, region, we have a value v_{ij} with the information of gains and losses for the segment i and patient j, these values represents gains or losses if they are greater or lower than a threshold.

	Patient 1	**Patient 2**	**Patient 3**	**...**	**Patient n**
Segment 1 (Chromosome-init-end)	v_{11}	v_{12}	v_{13}	…	v_{1n}
Segment 2 (Chromosome-init-end)	v_{21}	v_{22}	v_{23}	…	v_{2n}
...	…	…	…	…	…
Segment m (Chromosome-init-end)	v_{m1}	v_{m2}	v_{m3}	…	v_{mn}

Figure 7 BAC aCGH segmented and normalized

An analysis using Oligo aCGH shows that the available information is different. The information from these arrays is shown in Figure 8. The values v_{ij} represent gains and losses for segment i and patient j. Each patient has a different number of segments, which might not have the same initial value; this means that the initial or final value of segment i patient j can be different from segment i patient k.

Patient 1		**Patient 2**		**...**	**Patient n**	
Segment 11 (Chromosome-init-end)	v_{11}	**Segment 12 (Chromosome-init-end)**	v_{12}	**...**	**Segment 1n (Chromosome-init-end)**	v_{1n}
Segment 21 (Chromosome-init-end)	v_{21}	**Segment 22 (Chromosome-init-end)**	v_{22}	**...**	**Segment 2n (Chromosome-init-end)**	v_{2n}
...	…	**...**	…	**...**	**...**	…
Segment m1 (Chromosome-init-end)	v_{m1}	**Segment m2 (Chromosome-init-end)**	v_{m2}	**...**	**Segment mn (Chromosome-init-end)**	v_{mn}
		...	…	**...**	**...**	…
		Segment k2 (Chromosome-init-end)	v_{k2}	**...**	**Segment ln (Chromosome-init-end)**	v_{ln}

Figure 8 Oligo aCGH segmented and normalized

Finally, the system includes databases since the system extracts the information regarding genes, transcripts, CNV and local annotations.

Figure 9 shows the information from the BAC arrays cases, which includes 38 cases with 5 different pathologies. Only the information corresponding to chromosome 12 is shown. The green lines show gains regions in the chromosome, while red lines show losses. Therefore, the figure shows that the green patients have gains while the remainder present few variations. The most relevant segments are automatically highlighted as bright segments with the application of the hypothesis contrast Chi Squared. This technique allows the selection of the regions of interest.

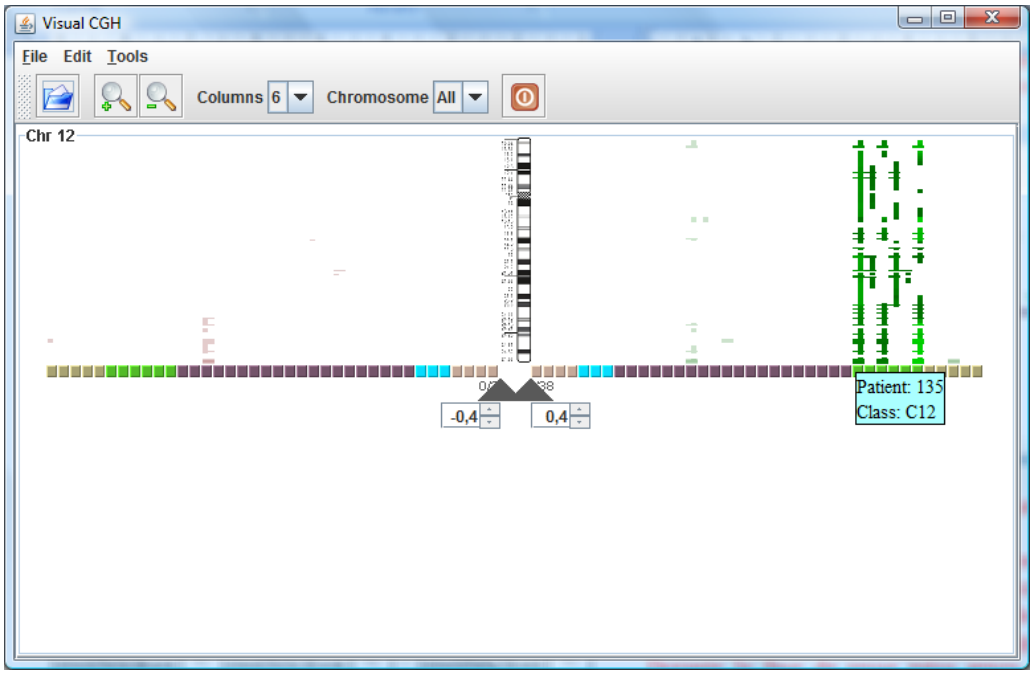


Figure 9 Automatic selection of segments

Once the data are represented, a CBR reasoning cycle is performed. During the retrieve phase, information regarding the catalogued genes and transcripts is recovered from the database. During the reuse phase, these genes are evaluated and valued according to the hypothesis contrast described in section 3. After selecting the segments, their relevance can be observed. Figure 10 shows the information from the genes that were recovered from the database and considered to be relevant.

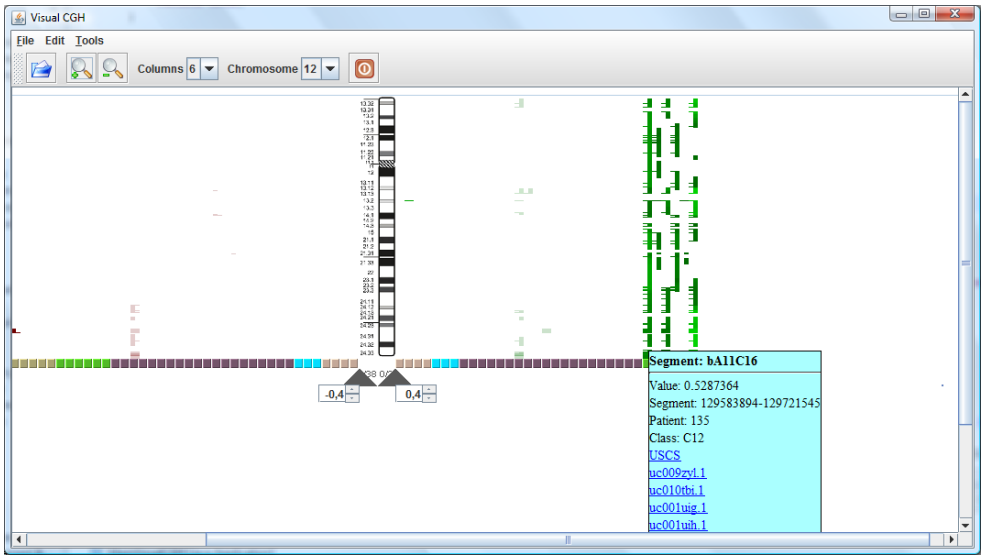
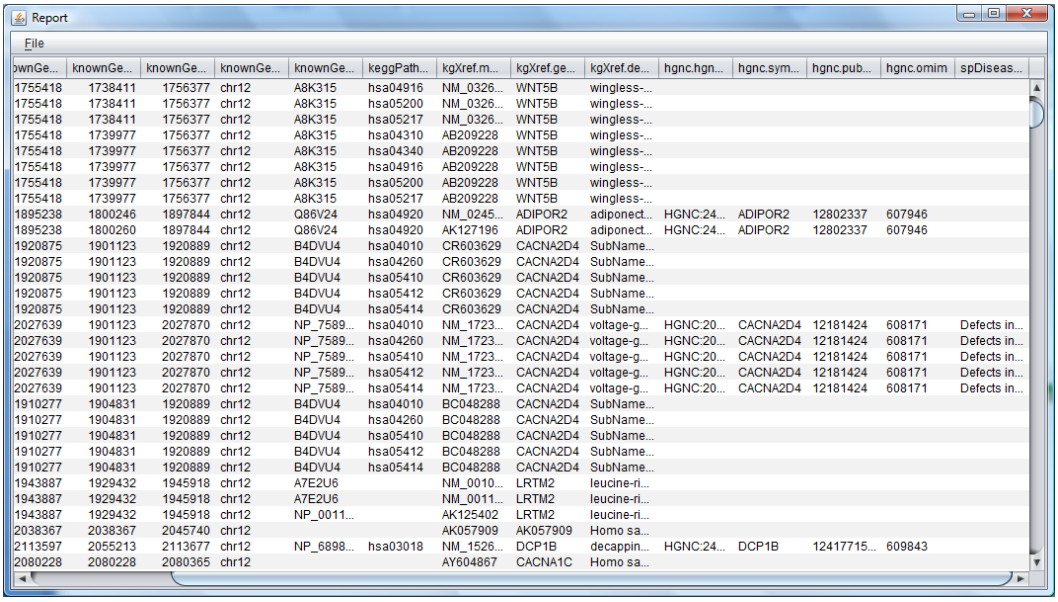


Figure 10 Automatic selection of segments and genes

In addition to visualizing the information for each of the different segments, it is also possible to generate reports automatically. These reports make it possible to quickly visualize potentially relevant information from the regions about proteins, genes and diseases. Figure 11 displays information related to proteins, genes and diseases. The system also makes it possible to generate reports on variants, duplications or CNVs.



ownGe...	knownGe...	knownGe...	knownGe...	knownGe...	keggPath...	kgXref.m...	kgXref.ge...	kgXref.de...	hgnc.hgn...	hgnc.sym...	hgnc.pub...	hgnc.omim	spDiseas...
1755418	1738411	1756377	chr12	A8K315	hsa04916	NM_0326...	WNT5B	wingless-...					
1755418	1738411	1756377	chr12	A8K315	hsa05200	NM_0326...	WNT5B	wingless-...					
1755418	1738411	1756377	chr12	A8K315	hsa05217	NM_0326...	WNT5B	wingless-...					
1755418	1739977	1756377	chr12	A8K315	hsa04310	AB209228	WNT5B	wingless-...					
1755418	1739977	1756377	chr12	A8K315	hsa04340	AB209228	WNT5B	wingless-...					
1755418	1739977	1756377	chr12	A8K315	hsa04916	AB209228	WNT5B	wingless-...					
1755418	1739977	1756377	chr12	A8K315	hsa05200	AB209228	WNT5B	wingless-...					
1755418	1739977	1756377	chr12	A8K315	hsa05217	AB209228	WNT5B	wingless-...					
1895238	1800246	1897844	chr12	Q86V24	hsa04920	NM_0245...	ADIPOR2	adiponect...	HGNC:24...	ADIPOR2	12802337	607946	
1895238	1800260	1897844	chr12	Q86V24	hsa04920	AK127196	ADIPOR2	adiponect...	HGNC:24...	ADIPOR2	12802337	607946	
1920875	1901123	1920889	chr12	B4DVU4	hsa04010	CR603629	CACNA2D4	SubName...					
1920875	1901123	1920889	chr12	B4DVU4	hsa04260	CR603629	CACNA2D4	SubName...					
1920875	1901123	1920889	chr12	B4DVU4	hsa05410	CR603629	CACNA2D4	SubName...					
1920875	1901123	1920889	chr12	B4DVU4	hsa05412	CR603629	CACNA2D4	SubName...					
1920875	1901123	1920889	chr12	B4DVU4	hsa05414	CR603629	CACNA2D4	SubName...					
2027639	1901123	2027870	chr12	NP_7589...	hsa04010	NM_1723...	CACNA2D4	voltage-g...	HGNC:20...	CACNA2D4	12181424	608171	Defects in...
2027639	1901123	2027870	chr12	NP_7589...	hsa04260	NM_1723...	CACNA2D4	voltage-g...	HGNC:20...	CACNA2D4	12181424	608171	Defects in...
2027639	1901123	2027870	chr12	NP_7589...	hsa05410	NM_1723...	CACNA2D4	voltage-g...	HGNC:20...	CACNA2D4	12181424	608171	Defects in...
2027639	1901123	2027870	chr12	NP_7589...	hsa05412	NM_1723...	CACNA2D4	voltage-g...	HGNC:20...	CACNA2D4	12181424	608171	Defects in...
2027639	1901123	2027870	chr12	NP_7589...	hsa05414	NM_1723...	CACNA2D4	voltage-g...	HGNC:20...	CACNA2D4	12181424	608171	Defects in...
1910277	1904831	1920889	chr12	B4DVU4	hsa04010	BC048288	CACNA2D4	SubName...					
1910277	1904831	1920889	chr12	B4DVU4	hsa04260	BC048288	CACNA2D4	SubName...					
1910277	1904831	1920889	chr12	B4DVU4	hsa05410	BC048288	CACNA2D4	SubName...					
1910277	1904831	1920889	chr12	B4DVU4	hsa05412	BC048288	CACNA2D4	SubName...					
1910277	1904831	1920889	chr12	B4DVU4	hsa05414	BC048288	CACNA2D4	SubName...					
1943887	1929432	1945918	chr12	A7E2U6		NM_0010...	LRTM2	leucine-ri...					
1943887	1929432	1945918	chr12	A7E2U6		NM_0011...	LRTM2	leucine-ri...					
1943887	1929432	1945918	chr12	NP_0011...		AK125402	LRTM2	leucine-ri...					
2038367	2038367	2045740	chr12			AK057909	AK057909	Homo sa...					
2113597	2055213	2113677	chr12	NP_6898...	hsa03018	NM_1526...	DCP1B	decappin...	HGNC:24...	DCP1B	12417715...	609843	
2080228	2080228	2080365	chr12			AY604867	CACNA1C	Homo sa...					

Figure 11 Automatic report with information from the highlighted segments

The system provides several visualizations to support the revision of the information by an expert. The system allows for information about the previously analyzed regions to be included, which makes it possible to eliminate regions that were not previously considered relevant. Additionally, the system can include

representations of different variations, which makes it possible to eliminate regions whose mutations have already been catalogued as not relevant. This helps the visual analysis and selection of relevant regions. The pink area highlighted in Figure 11 represents the regions presented by CNV; there is a gains area highlighted in green which corresponds to a CNV and should not be taken into consideration in the analysis. The information from the annotations inserted by laboratory personnel is likewise shown, with the color varying according to the user's selection. Another variation with respect to Figure 9 is that the data are represented as an accumulated amount as opposed to per individual patient, which allows the regions of gains or losses to be easily observed. The regions that were automatically selected by the analysis tests can be modified by using a mouse to mark the segments with a selection square, or by dragging the triangles located in the lower part of the chromosome

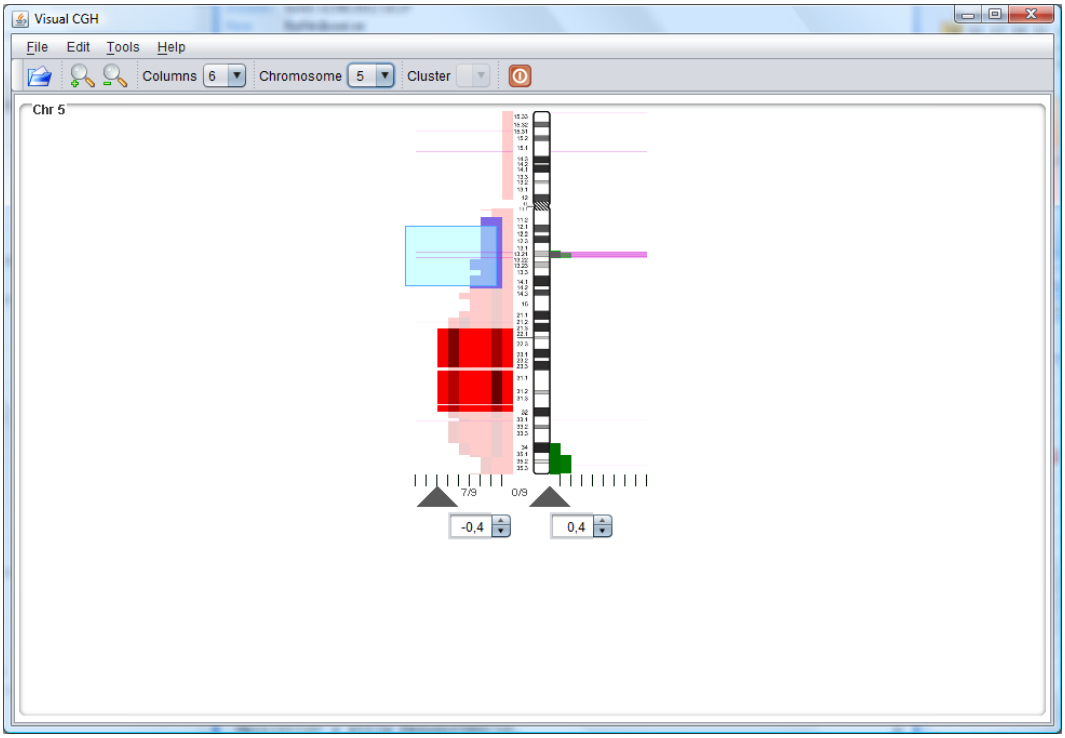


Figure 12 Accumulated view of chromosome 5 together with information from the CNVs

If there are any doubts, it is possible to consult the UCSC website to determine the relevance of a specific segment; to see the accumulated view, simply click on the segment and then select OK for the UCSC option for either the complete segment of the patient, or the minimum region into which each segment has been resegmented. Figure 13 shows the window from the UCSC site and the dialogue box that appears after clicking on a particular segment. This information is useful

to analyze the relevance of the regions by taking into consideration the different knowledge extraction techniques applied.

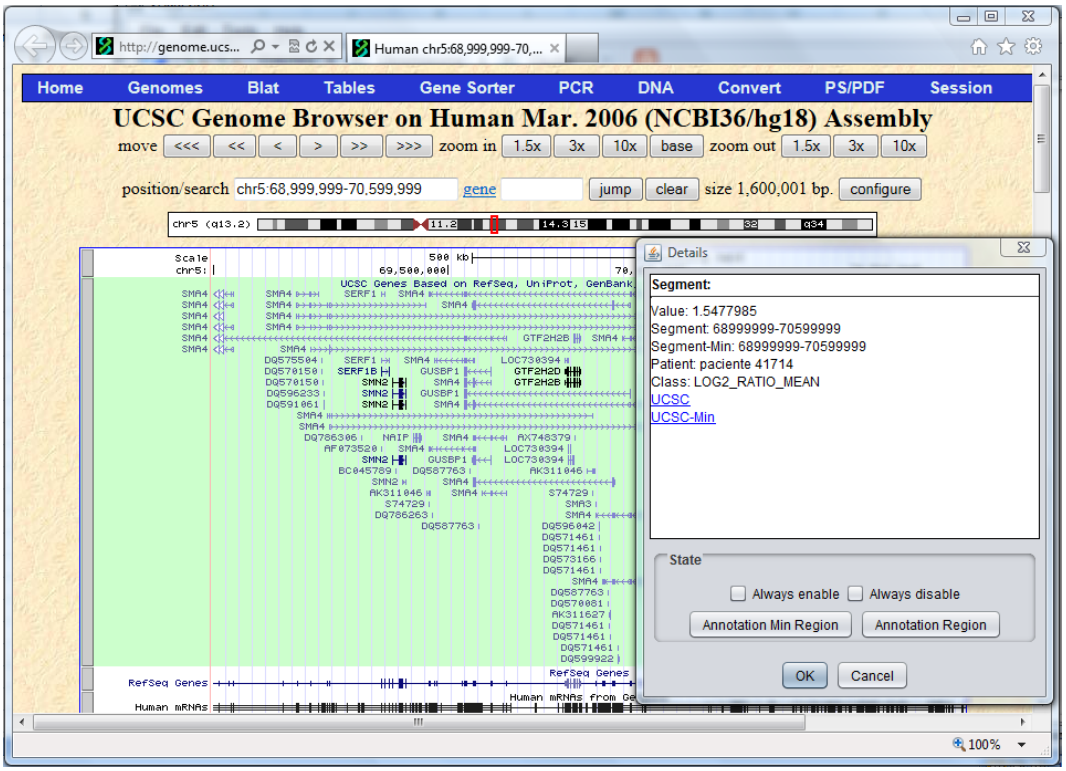


Figure 13 Information from UCSC for the selected segment

Figure 14 shows a representation in parallel coordinates and a bar graph. In the bar graph, one bar represents each individual and is divided into different segments with an amplitude proportional to the width of the segment. The color of the top rectangle represents the type of pathology the patient has. In parallel coordinates, each line is associated to a patient and the color represents the pathology type. Each coordinate represents a segment. If we select the patients from the Green category in the stacked bars, we can see how the other bars are deactivated, which indicates the patients have variations within different ranges; only the patients with variations within the range of the selected patients remain active, which makes it easy to see other similar patients. In the parallel coordinates, the values of each coordinate are adjusted to the maximum and minimum extremes for the selected individuals. The lines for each selected individual are highlighted while those not within the range of maximum and minimum values as established for each coordinate are marked in gray.

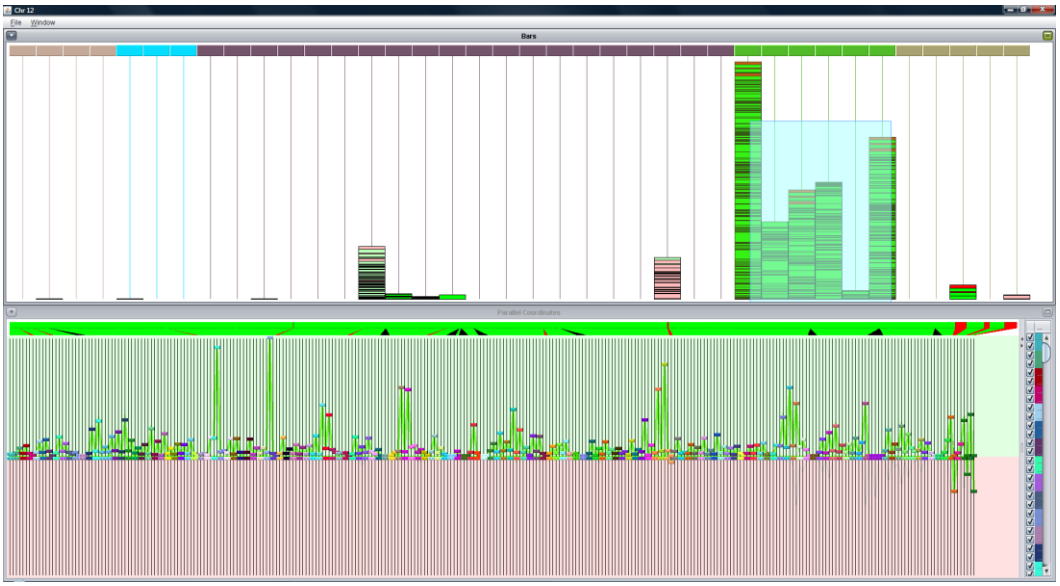


Figure 14 Clustering review with a bar graph and parallel coordinates

In the case of Figure 14, the number of segments selected was very high, which explains the appearance of so many coordinates. Having selected fewer, the number of coordinates would be lower, making it easier to see the range of variation for each of the coordinates.

If there are no groups, the system can also create a cluster of individuals which can then be revised by selecting the individuals with a mouse and modifying the cluster to which they belong. Clusters can be made only according to the information from the chromosomes that can be seen on the screen, and only based on the highlighted segments, which makes it possible to create a group according to the information considered relevant. Figure 15 shows a cluster created from chromosomes 5 and 11 using all of the information from chromosome 11 and the highlighted information from chromosome 5. Once the dendrogram was created, the cluster was manually corrected by selecting individuals one by one.

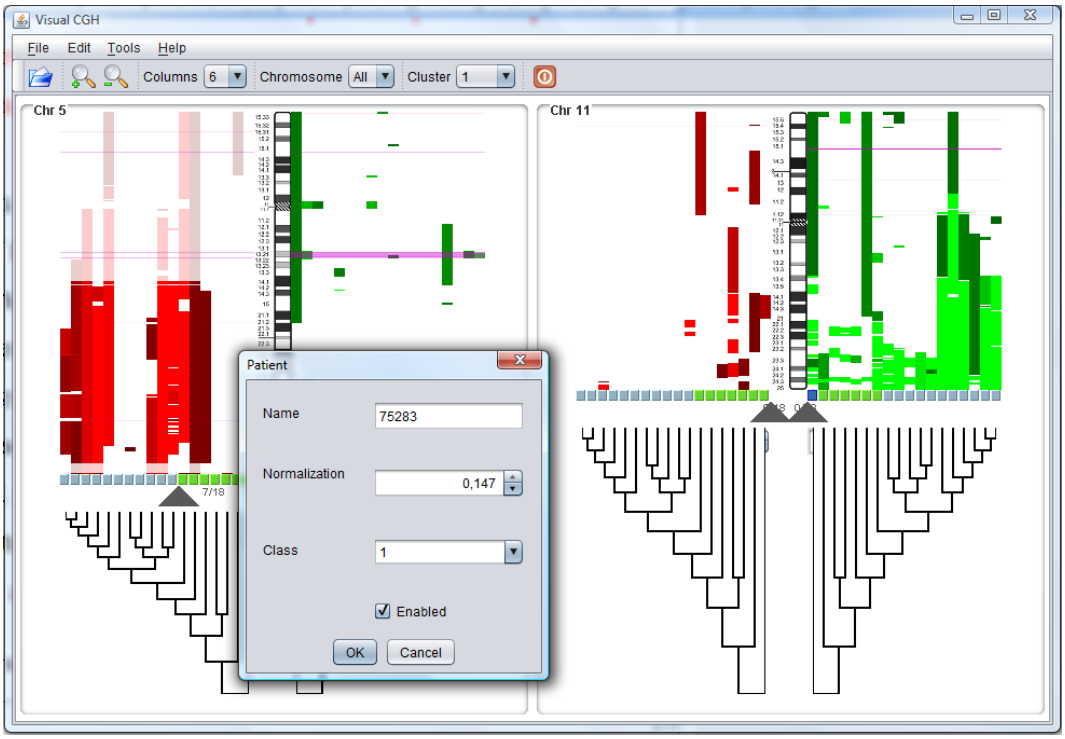


Figure 15 Grouped individuals according to the highlighted segments for each chromosome.

5 Conclusions

A visualization also makes it possible to carry out tasks, such as drag and drop, for each visualization, to export information in image format, to select thresholds, chromosomes to visualize, categories, individuals, to zoom, or to import Affymetrix (multiple tsv files) and NimbleGen (multiple txt files) data.

The multi-agent system can add agents that specialize in specific case studies, and allows the reuse of functionalities for specific layers. Furthermore, the independence of the different modules in this kind of system allows for the easy inclusion of new techniques. This case study used the aCGH data analysis to facilitate the addition and/or modification of existing techniques. The system provides easy access to information of several databases, improving the visual analysis of the information and proving relevant information of the selected regions of the chromosome. The system uses CBR to automatically select the genes that characterize pathologies. This CBR manages all the information of the databases and it allows the incorporation of new information that can be used in future analyses.

Finally, the different visualization can easily manage the data, thus improving the efficiency of the experts in the selection of relevant regions, its validation, and the access to information associated to these regions.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgements. This work has been carried out by the project Sociedades Humano-Agente en entornos Cloud Computing (Soha+C) . SA213U13. Project co-financed with Junta Castilla y León funds.

References

[1] Aha D., Kibler D., Albert, M.K.: Instance-based learning algorithms. Machine Learning. vol. 6, 37-66 (1991).

[2] Argente, E., Botti, V., Carrascosa, C., Giret, A., Julian, V. and Rebollo, M.: An abstract architecture for virtual organizations: The THOMAS approach. Knowledge and Information Systems. vol. 29 (2) pp. 379-403 (2011)

[3] Brown, P.O. and Botstein, D. Exploring the new world of the genome with DNA microarrays. Nature Genetics, vol. 21, 33–37 (1999).

[4] Chen, W., Erdogan, F., Ropers, H., Lenzner, S., Ullmann, R.: CGHPRO- a comprehensive data analysis tool for array CGH. BMC Bioinformatics. vol. 6 (85), 299-303 (2005).

[5] Cohen, W.W.: Fast effective rule induction. In Proceedings of the 12th International Conference on Machine Learning. Morgan Kaufmann. 115-123 (1995).

[6] Corchado J.M., De Paz J.F., Rodríguez S. and Bajo J. Model of experts for decision support in the diagnosis of leukemia patients. Artificial Intelligence in Medicine. vol. 46 (3), pp. 179-200 (2009).

[7] Glez-Pena D., Diaz F., Hernandez J.M., Corchado J.M. and Fdez-Riverola F., geneCBR: a translational tool for multiple-microarray analysis and integrative information retrieval for aiding diagnosis in cancer research. BMC Bioinformatics. vol. 10, 187 (2009).

[8] Glez-Peña, D., Díaz, F., Fdez-Riverola, F., Méndez, J.R., Corchado J.M., Fuzzy Patterns and GCS Networks to Clustering Gene Expression Data. Fuzzy Systems in Bioinformatics and Computational Biology 2009: 103-125.

[9] De Haan, J.R., Bauerschmidt, S., van Schaik, R.C., Piek E., Buydens, L.M.C., Wehrens R., Robust ANOVA for microarray data. Chemometrics and Intelligent Laboratory Systems. vol. 98 (1), 38-44 (2009).

[10] De Paz J.F., Bajo J., Vera V. and Corchado J.M.: MicroCBR: A case-based reasoning architecture for the classification of microarray data. Applied Soft Computing. vol. 11(8) 4496-4507 (2011).

[11] De Paz, J.F., Bajo, J., González, A., Rodríguez. S., and Corchado J.M.: Combining case-based reasoning systems and support vector regression to evaluate the atmosphere–ocean interaction. Knowledge and Information Systems. vol. 30 (1) pp. 155-177 (2012).

[12] Duda R.O., Hart P. Pattern classification and Scene Analysis. New York: John Wisley & Sons. (1973).

[13] Freeman, G.H. and Halton, J.H.: Note on an exact treatment of contingency, goodness of fit and other problems of significance, Biometrika, vol. 38, pp. 141-149 (1951).

[14] Fritzke, B.: A growing neural gas network learns topologies. Cambridge MA: Tesauro G, Touretzky D, Leen T (eds) Advances in Neural Information Processing Systems 7. pp. 625-632 (1995).

[15] Hehir-Kwa, J.Y, Egmont-Petersen, M., Janssen, I.M., Smeets, D., van Kessel, A.G., Veltman, J.A.: Genome-wide Copy Number Profiling on High-density Bacterial Artificial Chromosomes, Single-nucleotide Polymorphisms, and Oligonucleotide Microarrays: A Platform Comparison based on Statistical Power Analysis. DNA Research vol.14 (1) pp. 1-11 (2007).
[16] Hermsen, M., Coffa, J., Meijer, Y.G., Morreau, H., van Eijk, R., Oosting, J., van Wezel, T.: High-Resolution Analysis of Genomic Copy Number Changes. Genomics Essential Methods. Ed. Wiley (2010).
[17] Hixson, P., Laritsky, E., Wang, X., Jiang, T., Cheung, S., Van Den Veyver, I., Cai, W.: Comparison between BAC and oligo array platforms in detecting submicroscopic genomic rearrangements [abstract]. American Society of Human Genetics, 2006 Annual Meeting, p. 239 (2006).
[18] Hofmann, W.A., Weigmann, A., Tauscher, M., Skawran, B., Focken, T., Buurman, R., Wingen, L.U., Schlegelberger, B., Steinemann, D.: Analysis of Array-CGH Data Using the R and Bioconductor Software Suite. Comparative and Functional Genomics,2009, Article ID 201325 (2009).
[19] Holmes, G., Hall, M., Prank, E.: Generating Rule Sets from Model Trees. Advanced Topics in Artificial Intelligence. vol. 1747/1999, 1-12 (2007).
[20] Holte, R.C.: Very simple classification rules perform well on most commonly used datasets. Machine Learning. vol. 11, 63-91 (1993).
[21] http://www.cs.waikato.ac.nz/ml/weka/
[22] Ishkanian, A.S., Malloff, C.A., Watson, S.K. et al.: Nature Genetics, vol. 36, pp. 299–303 (2004).
[23] Kaufman, L., Rousseeuw, P.: Finding Groups in Data: An Introduction to Cluster Analysis. Wiley, New York (1990).
[24] Kenney, J. F. and Keeping, E. S.: Mathematics of Statistics, Pt. 2, 2nd ed. Princeton, NJ: Van Nostrand, (1951).
[25] Kim, S.Y., Nam, S.W., Lee, S.H., Park, W.S., Yoo, N.J., Lee, J.Y., Chung, Y.J. ArrayCyGHt, a web application for analysis and visualization of array-CGH data. Bioinformatics. vol. 21(10),2554-2555 (2005).
[26] Kohonen, T.: Self-organized formation of topologically correct feature maps. Biological Cybernetics 59-69. (1982).
[27] Kolodner J.: Case-Based Reasoning. Morgan Kaufmann. (1993).
[28] Kruskal, W., and Wallis. W.: Use of ranks in one-criterion variance analysis, Journalof American Statistics Association (1952).
[29] Lingjaerde, O.C., Baumbush, L.O., Liestol, K., Glad, I.K., Borresen-Dale, A.L. 2005. CGH-explorer, a program for analysis of array-CGH data. Bioinformatics, vol. 21(6), 821-822 (2004).
[30] Mantripragada, K.K., Buckley, P.G., Diaz de Stahl, T., Dumanski, J.P.: Genomic microarrays in the spotlight. Trends Genetics. vol. 20 (2), 87-94 (2004).
[31] Martinetz, T., Schulten, K.: A neural-gas network learns topologies Kohonen T, Makisara K, Simula 0, Kangas J (eds) Artificial Neural Networks Amsterdam pp. 397-402 (1991).
[32] Martinetz, T.: Competitive Hebbian learning rule forms perfectly topology preserving maps. ICANN'93: International Conference on Artificial Neural Networks. Springer Amsterdam. pp. 427-434 (1993).
[33] Menten, B., Pattyn, F., De Preter, K., Robbrecht, P., Michels, E., Buysse, K., Mortier, G., De Paepe, A., van Vooren, S., Vermeesh, J., et al.: ArrayCGHbase: an analysis platform for comparative genomic hybridization microarrays. BMC Bioinformatics. vol. 6 (124) 179-187 (2006).
[34] Oostlander, A.E., Meijer, G.A., Ylstra, B.: Microarray-based comparative genomic hybridization and its applications in human genetics. Clinical Genetics, vol. 66, pp. 488–495 (2004).
[35] Pinkel, D. and Albertson, D.G. Array comparative genomic hybridization and its applications in cancer. Nature Genetics. vol. 37, 11–17 (2005).
[36] Quinlan, J.R.: C4.5: Programs For Machine Learning. Morgan Kaufmann Publishers Inc. (1993).
[37] Rosa, P., Viara, E., Hupé, P., Pierron, G., Liva, S., Neuvial, P., Brito, I., Lair, S., Servant, N., Robine, N., Manié, E., Brennetot, C., Janoueix-Lerosey, I., Raynal, V., Gruel, N., Rouveirol, C., Stransky, N., Stern, M., Delattre, O., Aurias, A., Radvanyi, F., Barillot, E.: VAMP: Visualization and analysis of array-CGH, transcriptome and other molecular profiles Bioinformatics. vol. 22 (17), 2066-2073 (2006).

[38] Saitou, N., Nie, M.: The neighbor-joining method: A new method for reconstructing phylogenetic trees. Molecular Biology and Evolution 4 pp. 406-425 (1987).
[39] Smith, M.L., Marioni, J.C., Hardcastle, T.J., Thorne, N.P.: snapCGH: Segmentation, Normalization and Processing of aCGH Data Users' Guide. Bioconductor, (2006).
[40] Sneath, P., Sokal, R.: Numerical Taxonomy. The Principles and Practice of Numerical Classification. W.H. Freeman Company, San Francisco (1973).
[41] Wang, P., Young,K., Pollack, J., Narasimham, B., Tibshirani, R.: A method for callong gains and losses in array CGH data. Biostat. vol. 6 (1), 45-58 (2005).
[42] Willenbrock, H. and Fridlyand, J.: A comparison study: applying segmentation to array CGH data for downstream analyses. Bioinformatics. vol. 21 (22), 4084–4091 (2005).
[43] Xia, X., McClelland, M., Wang, Y.: WebArray, an online platform for microarray data analysis. BMC Bionformatics. vol. 6 (306), 1737-1745 (2005).
[44] Yang X., Huang Y., Crowson M., Li J., Maitland M.L., Lussier Y.A.: Kinase inhibition-related adverse events predicted from in vitro kinome and clinical trial data. Journal of Biomedical Informatics. vol. 43 (3) pp. 376-384 (2010).
[45] Ylstra, B., Van den Ijssel, P., Carvalho, B. and Meijer, G.: BAC to the future! or oligonucleotides: a perspective for microarray comparative genomic hybridization (array CGH). Nucleic Acids Research. vol. 34, 445–450 (2006).
[46] Ylstra, B., van den IJssel, P., Carvalho, B. et al.: BAC to the future! or oligonucleotides: a perspective for micro array comparative genomic hybridization (array CGH). Nucleic Acids Res., 34, 445–450. Review of microarray CGH platforms (2006)
[47] Yue, S., Wang, C.: The influence of serial correlation on the Mann-Whitney test for detecting a shift in median, Advances in Water Resources. vol. 25 (3), 325-333 (2002).
[48] Zato, C., De Paz, J.F., de la Prieta, F., Martín, B., Supporting System for Detecting Pathologies, IWANN 2011, 6692,669-676 (2011).
[49] Choon, Y.W., Mohamad, M.S., Deris, S., Illias, R.M., Chong, C.K., Chai, L.E., Omatu, S., Corchado, J.M., Differential Bees Flux Balance Analysis with OptKnock for in silico Microbial Strains Optimization. (2014)
[50] Misman, M.F., Mohamad, M.S., Deris, S., Hashim. S.Z.M., A Group-Specific Tuning Parameter for Hybrid of SVM and SCAD in Identification of Informative Genes and Pathways. International Journal of Data Mining and Bioinformatics (IJDMB). 10, No. 2. 146-160. (2014).
[51] Choon, Y.W., Mohamad, M.S., Deris, S., Illias, R.M., Chong, C.K., Chai. L.E., A hybrid of bees algorithm and flux balance analysis with OptKnock as a platform for in silico optimization of microbial strains. Bioprocess and Biosystems Engineering. vol. 37 (3), 521-532 (2014).