Master research Internship

Bibliographic report

Biclustering: Quantitative Formal concept Analysis in Answer set Programming

Domain: Bioinformatics

Author:
Marie Salmon

Supervisor:
Jacques Nicolas
Dyliss

Abstract: This bibliographic report presents several approaches to define biclusters on numerical data, and more particularly gene expression data (GED), using Formal Concept Analysis (FCA). Biclusters can be defined on GED by either binarizing the data and perform a standard FCA on the resulting table or using a quantitative generalization of FCA working on numerical data. We will show that the binarizing stage usually leads to huge table making the first approach unpractical for huge dataset. Therefore, only quantitative FCA approaches are interesting for our biclustering task. We will also show that all approaches require the definition of some parameters from which the results are very dependant. A part of the internship will then consist to determine appropriate values for this parameters. Another part of the internship will consist to implement some of the quantitative FCA approaches presented. We will show that Answer Set Programming is a flexible framework adapted to this end.

keywords: *Biclustering, Formal Concept Analysis, Bioinformatics, Answer Set Programming*

Contents

1 Introduction

Bioinformatics is an interdisciplinary field that develops methods and software for assisting biologists in data analysis and modeling. We work more particularly on gene expression data (GED) which are representative of the genes' behaviour in different biological conditions (location, time, temperature). GED can be seen as a particular case of numerical dataset which are defined by a set of objects, attributes and attributes values, where objects are genes, attributes are conditions and attributes values are gene expression values. Usually, clustering techniques are used on numerical dataset to group objects into clusters based on a similarity criteria defined on their descriptions on all attributes. Clusters can be seen as global patterns since they are computed for all attributes. However, in biology, we are more interested in local pattern since a set of genes usually have common behaviour on a subset of biological conditions only. Therefore, pairs (set of object, set of attributes) called biclusters should be considered. Biclusters are usually defined on a numerical dataset using Formal Concept Analysis (FCA) [2]. FCA is a branch of applied lattice theory which allows to define in an exhaustive and non redundant way biclusters on boolean data. Intuitively, to perform FCA on numerical data we can either binarize the data, this process is called conceptual scaling, or generalize the FCA to work with numerical data. FCA has been generalized in different direction including triadic FCA [10, 7], pattern structures [5, 9, 7], fuzzy FCA [1] and $\mathcal{K}$-FCA [16, 6]. We will implement such quantitative generalizations of FCA in Answer Set Programming (ASP) [11], a formalism in non monotonic logic adapted to solve combinatorial problem.

The bibliographic report is organized as follows. First, we present the problem of biclustering gene expression data in Section 2. Then, we give an introduction to FCA in Section 3 and present different approaches to define biclusters using FCA in Section 4 and 5. We discuss the advantages and drawbacks of each approach in Section 6. Afterwards, we briefly present ASP and the way FCA can be model using this formalism in Section 7. Finally, we give a conclusion to this study and talk about issues of the coming internship in Section 8.

2 Problem definition: biclustering gene expression data

We consider the problem of extracting biclusters of the form (set of genes, set of conditions) from gene expression data based on positive on negative correlation of gene expression values. Gene expression values represent the activity of the gene. Finding correlation between gene expression is of highest importance since it is typical of underlying dependencies conveyed by regulation mechanisms between genes. There exist two types of regulations mechanisms: up-regulation and down-regulation. Up-regulation is a mechanism which results in the increase of genes expression. Conversely, down-regulation is a mechanism resulting in the decrease of expression of genes. In this section, we present gene expression data and the data mining task called biclustering.

2.1 Gene expression data (GED)

Gene expression refers to the process by which the information coded a gene is used for the synthesis of molecules having an active role in cellular functioning. Gene's behaviour is reflected by the concentration of RNA which depends on biological conditions (cell, time, temperature). The RNA concentration of a gene, also called *gene expression value*, is measured in several conditions providing what is called a *gene expression profile* for this gene. Gene expression data (GED) are usually represented by huge 2-dimensional numerical tables where rows represent genes and columns

	a_1	a_2	a_3
o_1	1	2	7
o_2	9	5	2
o_3	4	5	6
o_4	1	5	6

Table 1: A constant bicluster.

	a_1	a_2	a_3
o_1	$1(\searrow)$	$2(\searrow)$	$7(\nearrow)$
o_2	$9(\nearrow)$	$5(\searrow)$	$2(\searrow)$
o_3	$4(\searrow)$	$5(\nearrow)$	$6(\nearrow)$
o_4	$1(\searrow)$	$5(\nearrow)$	$6(\nearrow)$

Table 2: A bicluster of coherent evolution based on up($\nearrow$) / down($\searrow$)-regulation.

	a_1	a_2	a_3
o_1	1	2	7
o_2	9	5	2
o_3	4	5	6
o_4	1	5	6

Table 3: A bicluster of similar values ($\theta = 1$).

	a_1	a_2	a_3
o_1	$1(\searrow)$	$2(\searrow)$	$7(\nearrow)$
o_2	$9(\nearrow)$	$5(\searrow)$	$2(\searrow)$
o_3	$4(\searrow)$	$5(\nearrow)$	$6(\nearrow)$
o_4	$1(\searrow)$	$5(\nearrow)$	$6(\nearrow)$

Table 4: A bicluster of opposite evolution based on up($\nearrow$) / down($\searrow$)-regulation.

represent experimental conditions. A table entry (g, c) correspond to the gene expression value of gene g in condition c.

We will work on two huge GED: one concerning the gene expression values of the entire genome on one healthy and one cancerous tissue of 15 dogs; the other, concern two reproduction modes of the *pea aphid*, a well-known sap-sucking insect.

2.2 Biclustering

Biclustering is a method of data analysis applied to gene expression for the first time in 2000 by Cheng and Church [3] and has been considered since as the method of choice for the GED analysis. The main reason being that it is able to group similar genes on subsets of conditions unlike standard clustering techniques which define global patterns by considering all conditions. We use in the section definitions from [8].

A *numerical dataset* is defined as a tuple $(\mathcal{O}, \mathcal{A}, \mathcal{W}, \mathcal{I})$ made of a set of object $\mathcal{O}$, a set of attributes $\mathcal{A}$, a set of attribute values $\mathcal{W}$ and a relation $\mathcal{I}$ on $\mathcal{O} \times \mathcal{A} \times \mathcal{W}$ meaning that object $o \in \mathcal{O}$ takes the value $w \in \mathcal{W}$ for attribute $a \in \mathcal{A}$ which is written $(o, a, w) \in \mathcal{I}$ or $a(o) = w$. Note that in our case, objects are genes, attributes are conditions and values are gene expression values. Given a numerical dataset $(\mathcal{O}, \mathcal{A}, \mathcal{W}, \mathcal{I})$, a *bicluster* is a pair (X, Y) with $X \subseteq \mathcal{O}$ and $Y \subseteq \mathcal{A}$. Graphically it is represented as a rectangle on the table (under permutations of rows and columns).

Several types of bicluster have been proposed and are surveyed in [12]. We only focus here on a generalization of the constant bicluster and the bicluster of coherent evolution, which are more suited for our goal. A constant bicluster is composed of a subset of objects having a constant value on a subset of attributes. Constant biclusters are not adapted to GED since gene expression values are rarely equal. A better way to analyse GED is to define biclusters based on a similarity relation. Two values $v_1, v_2 \in \mathcal{W}$ are considered similar if their difference does not exceed a fixed parameter θ, i.e. $|v_1 - v_2| \leq \theta$ also written $v_1 \simeq_\theta v_2$. A bicluster (X, Y) is a *bicluster of similar values* if $\forall g_j, g_l \in X$ and $\forall m_i, m_k \in Y$, $m_i(g_j) \simeq_\theta m_k(g_l)$. An example of a bicluster of similar values (with $\theta = 1$) is given in Table 3. Biclusters able to capture coherent evolution of values across object and/or attributes are also interesting for GED. A coherent evolution is seen as a same sequence of states over line or column. In case of gene expression values, such states can be up and

down-regulated, an example of coherent evolution over conditions (column) based on such states (with mean expression value per gene) is given in Table 2. We are also interested by pair ((G_1, G_2) , C) which are representative of opposite behaviour between the two set of genes G_1 and G_2 in the set of conditions C. Biclusters of opposite evolution are intuitively the contrary of biclusters of coherent evolution and consist of sequences of opposite states. An example of such bicluster is represented in Table 4 meaning that gene o_2 has an opposite behaviour to genes o_3 and o_4 in conditions a_1, a_2 and a_3.

A bicluster (X, Y) of type t is said *maximal* if adding any attribute or an object leads to a bicluster which is not of type t. For example, $(\{o_3, o_4\}, \{a_2\})$ is a constant bicluster but is not maximal for the numerical data given in Table 1, since o_2 also have value 5 for attribute a_3. Note that all biclusters defined in Tables 1 to 4 are maximal.

3 Formal Concept Analysis (FCA)

Formal Concept Analysis (FCA) is a branch of applied lattice theory. From a binary relation between a set of objects and a set of attributes it defines groups of objects and attributes called formal concept. A formal concept consists of a maximal set of objects sharing a maximal set of attributes. We use in the following standard definitions from [2] with notation changes.

A (dyadic) *formal context* is a triple ($\mathcal{O}$,$\mathcal{A}$,$\mathcal{I}$) where $\mathcal{O}$ is a set of objects, $\mathcal{A}$ a set of attributes and $\mathcal{I}$ a relation between $\mathcal{O}$ and $\mathcal{A}$ which says when an object $o \in \mathcal{O}$ has the attribute $a \in \mathcal{A}$ which is written $o\mathcal{I}a$ or $(o, a) \in \mathcal{I}$. A formal context is usually represented as a 2-dimensional table filled with crosses. A cross in table entry (o, a) means that the object o has attribute a. The following derivation operators $(.)'$ are considered:

$$\begin{aligned} X' &= \{a \in \mathcal{A} \mid \forall\ x \in X,\ x\mathcal{I}a\} \quad \text{for } X \subseteq \mathcal{O} \\ Y' &= \{o \in \mathcal{O} \mid \forall\ y \in Y,\ o\mathcal{I}y\} \quad \text{for } Y \subseteq \mathcal{A} \end{aligned}$$

In other words, X' is the set of objects having all attributes of Y and Y' is the set of attributes shared by all objects of X. A *formal concept* is a pair (X, Y) for $X \subseteq \mathcal{O}$ and $Y \subseteq \mathcal{A}$ such that $X' = Y$ and $Y' = X$. X is called the *extent* and Y the *intent* of the formal concept. A formal concept is graphically represented on the context table by a rectangle full of crosses obtained under a finite number of permutations on row and column. ($\{o_3, o_4\}$,$\{a_2, a_3\}$) and ($\{o_2, o_3, o_4\}$,$\{a_2\}$) are two examples of formal concepts respectively represented in yellow and blue in Table 5. Concepts are partially ordered by $(X_1, Y_1) \leq (X_2, Y_2)$ if $X_1 \subseteq X_2$ or in an equivalent way $Y_1 \supseteq Y_2$. In that case, (X_1, Y_1) is said to be a *sub-concept* of (X_2, Y_2) and (X_2, Y_2) a *super-concept* of (X_1, Y_1). For example, ($\{o_3, o_4\}$,$\{a_2, a_3\}$) is a sub-concept of ($\{o_2, o_3, o_4\}$,$\{a_2\}$) since $\{o_3, o_4\} \subset \{o_2, o_3, o_4\}$. We denote by $\mathcal{B}(\mathcal{O}, \mathcal{A}, \mathcal{I})$ the set of all formal concepts defined on the context $(\mathcal{O}, \mathcal{A}, \mathcal{I})$. The ordered set $\mathcal{B}(\mathcal{O}, \mathcal{A}, \mathcal{I}, \leq)$ forms a *concept lattice* also called *Galois lattice*. Figure 1 shows two equivalent representations of the concept lattice associated with the formal context given in Table 5. Each node represents a formal concept and each line an order relation between two formal concepts. (2) is a differential representation of (1) where the extent of a formal concept is composed of objects attached to the concept and its sub-concepts; and the intent is composed of the attributes attached to the concept and the ones of its super-concepts.

	a_1	a_2	a_3
o_1			×
o_2	×	×	
o_3		×	×
o_4		×	×

Table 5: A formal context with two concepts (green boxes corresponds to the intersection of the yellow and blue concepts).

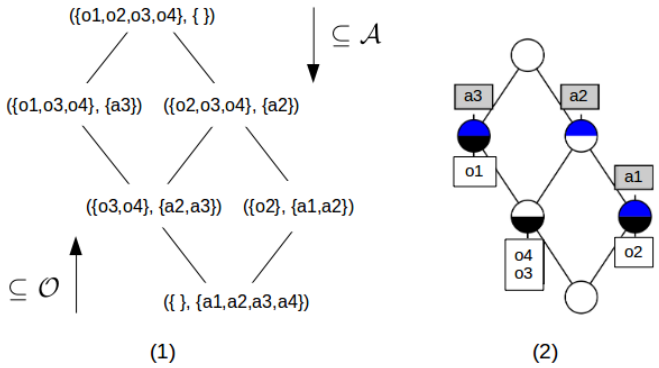

Figure 1: Two representations of the concept lattice associated with the formal context given in Table 5.

Formal Concept Analysis (FCA) can be considered as a particular case of biclustering defined on a boolean table, i.e. with table entry taking value in {0,1}, for which maximal constant biclusters of value 1 are defined. FCA allows to define all possible biclusters and to build a hierarchy on them.

4 Biclustering GED using conceptual scaling and FCA

To apply FCA on numerical data, it first needs to be transformed into a formal context, this process is called conceptual scaling. Depending on the data and the research goal, different scales may be chosen leading to different contexts and results. In this section, we present different approaches to conceptual scaling and the biclusters obtained on the resulting contexts.

4.1 Defining biclusters on context scaled using thresholds

A usual way to transform a numerical dataset $(\mathcal{O}, \mathcal{A}, \mathcal{W}, \mathcal{I})$ into a formal context consists to apply a cut on the whole numerical table representing the dataset using a single threshold t: each table entry having a value greater than t is encoded as 1 (or a cross) and 0 (or nothing) otherwise. For example, the formal context given in Table 5 can be obtained from Table 1 using 4 as threshold. In the case of GED, a cross is interpreted as up-regulation. After this binarization step a standard FCA can be applied to the context. Similarly, cut can be performed using one threshold per gene (row). For GED, each threshold t_g usually correspond to the mean expression values of genes g and is determined using confidence interval [13, 15]. Concepts defined on the resulting context correspond to genes up-regulated on the same subset of conditions. This can be viewed as a particular case of bicluster of coherent evolution with a constant state up-regulated or as a subset of maximal biclusters of similar values with $\theta \in [0, max(\mathcal{W}) - e]$, where $e \in \mathcal{W}$ is the minimal value such that $t < e$. However, not all maximal biclusters of similar values for a given θ are defined. For example, in our example the bicluster of similar values ($\theta = 0$) $(\{o_1, o_4\}, \{a_1\})$ is not defined using 4 as threshold on the Table 1. To define all biclusters of similar values we need to perform a cut and an FCA for all values of $\mathcal{W}$ as threshold.

In [15], the GED is transformed in a formal context in two stages. First, gene expression values

	a_1	a_2	a_3
o_1	1	2	7
o_2	9	5	2
o_3	4	5	6
o_4	1	5	6

(a)

↓

	a_1	a_2	a_3
o_1	-1	0	1
o_2	1	1	-1
o_3	-1	1	1
o_4	-1	1	1

(b)

→

	up-a_1	down-a_1	up-a_2	down-a_2	up-a_3	down-a_3
o_1		×			×	
o_2	×		×			×
o_3		×	×		×	
o_4		×	×		×	
$\neg o_1$	×					×
$\neg o_2$		×		×	×	
$\neg o_3$	×			×		×
$\neg o_4$	×			×		×

(c)

Figure 2: Discretization with $\alpha = 0.5$ (b) of a GED (a) then derived in the formal concept (c).

are discretized in {-1,0,1} using one threshold t_i per gene given by $t_i = m_i - \alpha\mu_i$ where m_i is the mean and μ_i the standard derivation of row i. α is a user fixed parameter used to tune the deviation from mean. Values -1, 1 and 0 thus represent respectively down-, up- and normal-regulation. Then, the table of size $n \times p$ is transformed into a formal context of size $2n \times 2p$ by adding the opposite gene expression profile of each gene g represented as $\neg g$, i.e. opposite values of each gene for each conditions, and by creating two boolean attributes for each previous conditions c: *is-up-regulated-in-c* and *is-down-regulated-in-c*. An example of such transformation is given in Figure 2. As there is redundancy in the formal context, different concepts having the same meaning are defined. For example, concepts $(\{o_2, \neg o_1, \neg o_3, \neg o_4\}, \{\text{up-}a1, \text{down-}a3\})$ and $(\{\neg o_2, o_1, o_3, o_4\}, \{\text{down-}a1, \text{up-}a3\})$, defined on the formal concept represented in Figure 2, both mean that o_2 is negatively correlated with o_1, o_3 and o_4 in conditions a_1 and a_3. The concept lattice is therefore filtered to remove such duplicates. Note that the concepts defined on the generated formal context can either correspond to biclusters of opposite or coherent evolution.

This technique is interesting as it allows to determine in the same way positive and negative correlation between set of genes but it has few drawbacks. Indeed, it generates redundant concepts due to the redundant information present in the formal context. However, this redundancy could be removed by creating only one of the attributes *is-up-regulated* or *is-down-regulated* for each condition. In addition, the number of lines of the formal context will still be twice the number of lines of the initial table leading to huge table when the initial one is already big.

4.2 Defining biclusters on context scaled using interordinal scaling

Ganter and Wille use a discretization procedure called *interordinal scaling* which allows to produce all possible intervals of values from a numerical dataset $(\mathcal{O}, \mathcal{A}, \mathcal{W}, \mathcal{I})$ without loss of information. For each attribute $a \in \mathcal{A}$ with range values $\mathcal{W}_a$ it constructs $2 \times |\mathcal{W}_a|$ binary attributes denoted "$a \leq w$" and "$a \geq w$" $\forall\ w \in \mathcal{W}_a$. This scaling technique produces the scaled context given in Table 6 from the numerical dataset given in Table 1. Sets of attributes can then be understood as a vector of intervals, each dimension representing the interval of values of each attribute. For example the

	≥ 1	≤ 4	≤ 9	≥ 1	≤ 4	≤ 9	≥ 2	≥ 5	≥ 2	≤ 5	≥ 2	≥ 6	≥ 7	≥ 2	≤ 6	≤ 7
	a_1	a_1	a_1	a_1	a_1	a_1	a_2	a_2	a_2	a_2	a_3	a_3	a_3	a_3	a_3	a_3
o_1	×	×	×	×			×	×	×				×	×	×	×
o_2			×	×	×	×		×	×	×	×	×	×	×		
o_3		×	×	×	×			×	×	×		×	×	×	×	
o_4	×	×	×	×				×	×	×		×	×	×	×	

Table 6: Context derived from the numerical data given in Table 1 using interordinal scaling.

set $\{a_1 \leq 4, a_1 \leq 9, a_2 \leq 2, a_3 \geq 5, a_3 \leq 6\}$ corresponds to the interval vector $\langle [1,4],[2,2],[5,6]\rangle$.

Concepts defined on this scaled context correspond to biclusters but they are not necessarily of similar values for a given θ. Indeed, a concept may contain intervals larger than θ and intervals of non similar values which respectively implies non similarity in column and lines. Moreover, the biclusters are not necessarily maximal. Therefore the following conditions must be added to extract biclusters (X, Y) from a concept (X, Z):

1. $\forall\ [a_i, b_i] \in v,\ a_i \simeq_\theta b_i$ (similarity in column)
2. $\forall\ i, j \in |Y|, [a_i, b_i] \simeq_\theta [a_j, b_j] \Leftrightarrow \min(a_i, a_j) \simeq_\theta \max(b_i, b_j)$ (similarity in row)
3. $\exists!(X', Z')$ such that $(X, Z) \leq (X', Z')$ with interval of maximal size θ defined on the same dimensions (maximality)

where $Y \in Z$ and Y can be represented as a vector of interval $v = \langle [a_i, b_i]\rangle_{i \in |Y|}$. Then, not any bicluster for $\theta = 2$ is extracted from concept $(\{o_3, o_4\}, \langle [1,4],[2,5],[6,6]\rangle)$ since the bicluster $(\{o_3, o_4\}, \{a_3\})$ is not maximal as object o_1 has value 7 for attributes a_3 which is defined by concept $(\{o_1, o_3, o_4\}, \langle [1,4],[2,5],[6,7]\rangle)$.

This method has a major problem: it produces huge context difficult to process. In our example we start from a numerical table of dimension 4×3 and end with a formal context of size 4×16. When the initial data is already big the produced data table is huge. Such table is in addition very redundant. Indeed the same interval can be represented by several sets of attributes. For example the sets $\{a_1 \leq 4, a_1 \leq 9\}$, $\{a_1 \leq 4\}$, $\{a_1 \geq 1, a_1 \leq 4, a_1 \leq 9\}$ and $\{a_1 \geq 1, a_1 \leq 4\}$ represent the same interval [1,4] for attribute a_1.

5 Biclustering GED using quantitative FCA

As we saw in the previous section, conceptual scaling usually produce huge table hard to process. A better solution to apply FCA on numerical data therefore consists to generalize FCA to work directly on numerical data such generalization are refereed as quantitative FCA. We present in this section several quantitative approaches to FCA and their application to biclustering GED.

5.1 Defining biclusters with pattern structure

A pattern structure [5] can be seen as a generalization of a formal context in which the set of attributes is replaced by a meet-semilattice of object descriptions called patterns. We present the notions of patten structure and explain how it can be used to define biclusters on GED.

Pattern structure

A pattern structure is defined by a triple $(\mathcal{O}, \underline{D}, \delta)$ where $\underline{\mathrm{D}} = (D, \sqcap)$ is a meet-semilattice of object descriptions also called *pattern* with $\sqcap$ the infimum (meet) operator and $\delta : \mathcal{O} \to D$ is a mapping providing any object from $\mathcal{O}$ with a pattern. Patterns are ordered by a *subsumption* relation $\sqsubseteq$: $c \sqsubseteq d \Leftrightarrow c \sqcap d = c$. The following derivation operators are defined on pattern structure:

$$X^{\square} = \prod_{o \in X} \delta(o) \quad \text{for } X \subseteq \mathcal{O}$$
$$d^{\square} = \{o \in \mathcal{O} \mid d \sqsubseteq \delta(o)\} \quad \text{for } d \in D$$

The operators $(.)^{\square}$ form a Galois connection between the powerset of $\mathcal{O}$ and $(D, \sqsubseteq)$. A *pattern concept* is a pair (X, d) with $X \subseteq \mathcal{O}$ and $d \in (D, \sqcap)$ such that $X^{\square} = d$ and $d^{\square} = X$. In other words, d represent the maximal pattern describing all objects of X and X the maximal set of object sharing the pattern d. When ordered by $(X_1, d_1) \leq (X_2, d_2) \Leftrightarrow X_1 \subseteq X_2 (\Leftrightarrow d_2 \sqsubseteq d_1)$, the set of pattern concepts form a complete lattice called *pattern concept lattice* of $(\mathcal{O}, \underline{D}, \delta)$.

Biclustering GED with pattern structure using intervals as pattern

Biclusters of similar values are defined using pattern structure in [9, 7]. A numerical dataset $(\mathcal{O}, \mathcal{A}, \mathcal{W}, \mathcal{I})$ is represented as a pattern structure for which patterns are vectors of intervals, where the i^{th} dimension give an interval of values from $\mathcal{W}$ for attribute $a_i \in \mathcal{A}$. For example, object o_1 of Table 1 is described by the pattern $\delta(o_1) = \langle [1,1], [2,2], [7,7] \rangle$. The infimum $\sqcap$ of two intervals $[a_1, b_1]$ and $[a_2, b_2]$ is $[a_1, b_1] \sqcap [a_2, b_2] = [min(a_1, a_2), max(b_1, b_2)]$. In other words, $[a_1, b_1] \sqcap [a_2, b_2]$ is the largest interval containing both of them. Intervals are then ordered by

$$\begin{array}{rcl} [a_1, b_1] \sqsubseteq [a_2, b_2] & \Leftrightarrow & [a_1, b_1] \sqcap [a_2, b_2] = [a_1, b_1] \\ & \Leftrightarrow & [min(a_1, a_2), max(b_1, b_2)] = [a_1, b_1] \\ & \Leftrightarrow & [a_1, b_1] \supseteq [a_2, b_2] \end{array}$$

An object is described by an *interval pattern* which correspond to a p-dimensional vector of intervals, with $p = |\mathcal{A}|$. Given two vectors of intervals $e = \langle [a_i, b_i] \rangle_{i \in [1,p]}$ and $f = \langle [c_i, d_i] \rangle_{i \in [1,p]}$ their infimum $\sqcap$ and induced ordering relation $\sqsubseteq$ are given by:

$$\begin{array}{rcl} e \sqcap f = \langle [a_i, b_i] \rangle_{i \in [1,p]} \sqcap \langle [c_i, d_i] \rangle_{i \in [1,p]} & \Leftrightarrow & e \sqcap f = \langle [a_i, b_i] \sqcap [c_i, d_i] \rangle_{i \in [1,p]} \\ e \sqsubseteq f \Leftrightarrow \langle [a_i, b_i] \rangle_{i \in [1,p]} \sqsubseteq \langle [c_i, d_i] \rangle_{i \in [1,p]} & \Leftrightarrow & e \sqsubseteq f = [a_i, b_i] \sqsubseteq [c_i, d_i],\ \forall i \in [1, p] \end{array}$$

For example, on the context given in Table 5 we have

$$\begin{array}{rcl} \{o_2, o_3\}^{\square} & = & \delta(o_1) \sqcap \delta(o_3) = \langle [9,9], [5,5], [2,2] \rangle \sqcap \langle [4,4], [5,5], [6,6] \rangle \\ & = & \langle [9,9] \sqcap [4,4], [5,5] \sqcap [5,5], [2,2] \sqcap [6,6] \rangle = \langle [4,9], [5,5], [2,6] \rangle \\ \langle [4,9], [5,5], [2,6] \rangle^{\square} & = & \{o \in \mathcal{O} \mid \langle [4,9], [5,5], [2,6] \rangle \sqsubseteq \delta(o)\} = \{o_2, o_3\} \end{array}$$

Hence $(\{o_2, o_3\}, \langle [4,9], [5,5], [2,6] \rangle)$ is a pattern concept. Note that the pattern lattice associated to a pattern context is a complete lattice. Pattern concepts can be assimilated to biclusters. However, like concepts defined on a context scaled using interordinal scaling, they are not necessarily maximal biclusters of similar values for a given θ. Note that the two methods are strictly equivalent

but pattern structure allow to define biclusters without a binarization procedure and create more readable results.

Maximal biclusters of similar values are extracted from pattern concepts in the following steps. First, the similarity in column is insured by removing all interval $[a, b]$ of an interval pattern which do not satisfy $a \simeq_\theta b$. Then, biclusters (X, Y) are extracted from a pattern concept (X, d) where Y is a set of attributes for which the corresponding set of intervals $d_Y = \langle [a_i, b_i] \rangle_{i \in |Y|}$ satisfy the property: $\forall\ i, j \in |Y|, [a_i, b_i] \simeq_\theta [a_j, b_j] \Leftrightarrow min(a_i, a_j) \simeq_\theta max(b_i, b_j)$. Note that Several biclusters can be extracted from one pattern concept. For example, the concept $(\{o_2, o_3\}, \langle [4, 9], [5, 5], [2, 6] \rangle)$ correspond to two biclusters of similar values for $\theta = 5$: $(\{o_2, o_3\}, \{a_1, a_2\})$ and $(\{o_2, o_3\}, \{a_2, a_3\})$. Finally, only maximal biclusters are kept, i.e. concepts for which there is no super-concept with interval of maximal size θ defined on the same attributes. A visual example of this extraction process can be found in [7].

5.2 Defining biclusters with Triadic Concept Analysis

Triadic Concept Analysis (TCA) is an extension of FCA introduced by Lehmann and Wille in 1995 [10] to deal with more complex relations between objects an attributes. FCA is extended by adding a third dimension to the formal context, a set of conditions. In this section, we first formally define TCA and then explain how it can been applied to our biclustering task.

Triadic Concept Analysis (TCA)

A *triadic context* is defined by a tuple $(\mathcal{O}, \mathcal{A}, \mathcal{C}, \mathcal{J})$ where $\mathcal{O}$, $\mathcal{A}$ and $\mathcal{C}$ are respectively sets of objects, attributes and conditions and $\mathcal{J}$ a relation between $\mathcal{O} \times \mathcal{A} \times \mathcal{C}$. $(o, a, c) \in \mathcal{J}$ is interpreted as *"the object o has the attribute a under condition c"*. A *triadic concept* of a triadic context $(\mathcal{O}, \mathcal{A}, \mathcal{C}, \mathcal{J})$ is a triple (Y_1, Y_2, Y_3) with $Y_1 \subseteq \mathcal{O},\ Y_2 \subseteq \mathcal{A}$ and $Y_3 \subseteq \mathcal{C}$ such that $(Y_1, Y_2, Y_3) \subseteq \mathcal{J}$ and for $(X_1, X_2, X_3) \subseteq \mathcal{J}$

$$Y_1 \subseteq X_1, Y_2 \subseteq X_2, Y_3 \subseteq X_3 \Rightarrow (Y_1, Y_2, Y_3) = (X_1, X_2, X_3)$$

Y_1 is called the extent, Y_2 the intent and Y_3 the *modus* of the concept. A triadic context is represented by a three dimensional matrix filled with crosses and a triadic concept by a maximal cube of crosses. Triadic concepts are ordered in a *triadic diagram* which is a combination of a *geometric structure* represented by a triangular pattern and three *ordered structures* (one for each dimension), each represented by a Hasse diagram. A formal definition of triadic diagram and a representation example can be found in [7] (p.13-14).

Biclustering GED with TCA using intervals as conditions

In [7], TCA is applied to GED to determine biclusters of similar values using intervals as conditions. The set of conditions $\mathcal{C}$ of a triadic context $(\mathcal{O}, \mathcal{A}, \mathcal{C}, \mathcal{J})$ can be derived from the set of values $\mathcal{W}$ of a numerical dataset $(\mathcal{O}, \mathcal{A}, \mathcal{W}, \mathcal{I})$ using either interordinal scaling or classes of tolerance. A set of values $V \in \mathcal{W}$ is a *class of tolerance* over the set of values $\mathcal{W}$ with respect to a tolerance (or similarity) relation $\simeq_\theta$ if

$$\forall\ w_1, w_2 \in V,\ w_1 \simeq_\theta w_2 \ \text{ and } \ \forall\ w_1 \notin V,\ \exists w_2 \in V \text{ such that } w_1 \not\simeq_\theta w_2$$

	[1,2]			[4,5]			[5,6]			[6,7]		
	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3
o_1	×	×										×
o_2			×		×			×		×		
o_3				×	×			×	×			×
o_4	×				×			×	×			×

Table 7: A triadic context derived from Table 1 using tolerance classes over $\mathcal{W}$ and $\theta = 1$ with the triadic concept ($\{o_3,o_4\}$, $\{a_2,a_3\}$, $\{[5,6]\}$) in grey corresponding to a bicluster of similar values ($\{o_3,o_4\}$, $\{a_2,a_3\}$).

For example, {4,5,6} is the only class of tolerance defined on the set of values {1,4,5,6} for $\theta = 2$. Classes of tolerance can be renamed as the convex hull of their elements: {4,5,6} is for example renamed as [4,6]. An example of triadic context with classes of tolerance as conditions is given in Table 7. It has been derived from the numerical dataset given in Table 1 with $\theta = 1$.

The set of triadic concepts defined on a triadic context built using interordinal scaling correspond to all maximal biclusters of similar values for all possible values of θ, i.e. for $\theta \in [0, max(\mathcal{W}) - min(\mathcal{W})]$. Whereas, concepts defined on a triadic context built using classes of tolerance correspond to biclusters of similar values for a given θ and are not necessary maximal. For example the triadic concept $(\{o_3, o_4\}, \{a_3\}, \{[5, 6], [6, 7]\})$ defined on Table 7 correspond to a non maximal bicluster of similar values for $\theta = 1$ since o_1 has value 7 in condition a_3. Hence, the authors defined an algorithm named TriMax to extract maximal biclusters of similar values from such context. The algorithm, treat each class of tolerance separately, defining one dyadic formal context for each and then check if the produced concepts are still concepts in other dyadic contexts.

Using classes of tolerance avoid to look at intervals that may not be interesting, such as having a range size that is greater than θ or that may induced non maximal biclusters such as interval of the form $[n, n]$, $n \in \mathbb{R}$. This is particularly interesting since gene expression values may vary a lot depending on the experimental conditions and genes, leading to a huge set of possible intervals. However, the creation of a triadic context can be question when it is used to determine biclusters of similar values for a given θ since afterwards each conditions of the third dimension is treated (separately) as a dyadic formal context.

5.3 Defining biclusters with $\mathcal{K}$-FCA

In $\mathcal{K}$-FCA [17], the numerical data table take values in an idempotent semifield $\overline{\mathcal{K}}$ and the concepts are determined using a fixed parameter φ. We introduce the notion of $\mathcal{K}$-FCA and then explain how it has been applied to biclustering GED using max-plus and min-plus algebras.

$\mathcal{K}$-FCA

A $\overline{\mathcal{K}}$-formal context is a tuple $(\mathcal{O}, \mathcal{A}, \mathcal{R})_{\overline{\mathcal{K}}}$ where $\mathcal{O}$ and $\mathcal{A}$ are as usual sets of objects and attributes and $\mathcal{R}$ is an incidence matrix relating object and attributes which has entries in $\overline{\mathcal{K}} = \langle K, \oplus, \otimes, \epsilon, e\rangle$. R_{oa} correspond to the degree to which object o has attribute a. $\mathcal{K}$-FCA add a parameter $\varphi \in K$ called the *threshold of existence* which describes a maximum degree value allowed for each pair of object and attribute. Objects and attributes are seen as vectors over the idempotent semifield

$X = \overline{\mathcal{K}}^{\mathcal{O}}$ and $Y = \overline{\mathcal{K}}^{\mathcal{A}}$. The following derivative operator $(.)^{\uparrow}_{R,\varphi}$ and $(.)^{\downarrow}_{R,\varphi}$ are considered

$$(x)^{\uparrow}_{R,\varphi} = \bigvee\{x \in X \mid x^T \otimes \mathcal{R} \otimes y \leq \varphi\}$$
$$(y)^{\downarrow}_{R,\varphi} = \bigvee\{y \in Y \mid x^T \otimes \mathcal{R} \otimes y \leq \varphi\}$$

Analogously to FCA, A pair (x, y) for $x \in \overline{\mathcal{K}}^{\mathcal{O}}$, $y \in \overline{\mathcal{K}}^{\mathcal{A}}$ such that $(x)^{\uparrow}_{R,\varphi} = y$ and $(y)^{\downarrow}_{R,\varphi} = x$ is called a φ-concept; φ-concepts are ordered by $(x_1, y_1) \leq (x_2, y_2) \Leftrightarrow x_1 \leq x_2$; and the ordered set of φ-concepts forms a φ-lattice. φ-concepts and the φ-lattice depend on the choice of algebraic structure $\overline{\mathcal{K}}$.

Biclustering GED with $\mathcal{K}$-FCA using max-plus and min-plus algebras

$\mathcal{K}$-FCA allows to defined biclusters on under- or over-expressed genes using adapted $\mathcal{K}$: max-plus and min-plus algebras denoted $\overline{\mathbb{R}}_{max,+}$ and $\overline{\mathbb{R}}_{min,+}$ [16, 6]. $\overline{\mathbb{R}}_{max,+}$ is equipped with maximum and addition as the two binary operations whereas $\overline{\mathbb{R}}_{min,+}$ is equipped with minimum and addition.

Before applying $\mathcal{K}$-FCA on GED the latter is usually normalized and log-compressed to be amenable with $\overline{\mathbb{R}}_{max,+}$ and $\overline{\mathbb{R}}_{min,+}$. One formal context is then defined on each algebraic structure. φ-concepts defined on the $\overline{\mathbb{R}}_{max,+}$-formal context correspond to maximal biclusters of under-expressed genes with regards to a threshold φ. However, φ-concepts defined on the $\overline{\mathbb{R}}_{min,+}$-formal context do not correspond to biclusters of over-expressed genes. In order to define such biclusters, new derivative operators (having a reverse order relation) $(.)^{\uparrow}_{R,\phi}$ and $(.)^{\downarrow}_{R,\phi}$ need to be defined:

$$(x)^{\uparrow}_{R,\phi} = \bigvee\{x \in X \mid x^T \otimes \mathcal{R} \otimes y \geq \phi\}$$
$$(y)^{\downarrow}_{R,\phi} = \bigvee\{y \in Y \mid x^T \otimes \mathcal{R} \otimes y \geq \phi\}$$

Then, ϕ-concepts defined on the $\overline{\mathbb{R}}_{min,+}$-formal context correspond to maximal biclusters of over-expressed genes with regards to a threshold ϕ. Note that the higher (resp. lower) is the value of ϕ (resp. φ) the more restrictive is the technique.

5.4 Defining biclusters with fuzzy FCA

FCA can be performed on numerical dataset by considering table entries as truth degrees in fuzzy logic and proceed in a similar way as in standard FCA. In this section, we first recall few notions of fuzzy logic. Then, present an overview of different approaches to fuzzy FCA. Finally, we explain how fuzzy FCA could be applied to biclustering GED.

Basic definitions of fuzzy logic

A *complete residuated lattice* is a structure of truth value commonly used in fuzzy logic. A residuated lattice is an algebra $\mathbf{L}$=$\langle L, \wedge, \vee, \otimes, \rightarrow, 0, 1\rangle$ such that (i) $\langle L, \wedge, \vee, 0, 1\rangle$ is a lattice with the least element 0 and the greatest element 1; (ii) $\langle L, \otimes, 1\rangle$ is a commutative monoïd i.e. $\otimes$ is commutative, associative and $a \otimes 1 = 1 \otimes a = a$ for each $a \in L$; and (iii) $\otimes, \rightarrow$ form an adjoint pair, i.e. $x \otimes y \leq Z \Leftrightarrow x \leq y \rightarrow z$ holds $\forall\, x, y, z \in L$. A Residuated lattice $\mathbf{L}$ is called complete if $\langle L, \wedge, \vee\rangle$ is a complete lattice. The collection of all fuzzy sets in a universe U is denoted by L^U.

Fuzzy Formal Concept Analysis

In fuzzy FCA, a table entry (o,a) corresponds to the *truth degree to which object o has attribute a*. Usually, degrees are taken from a scale $L \subseteq [0,1]$. Several approaches to fuzzy FCA have been proposed and are surveyed in [1]. They all shared the same notion of *fuzzy context* also called L-context which is defined as a triple $(\mathcal{O}, \mathcal{A}, \mathcal{R})$ where $\mathcal{O}$ and $\mathcal{A}$ are as usual sets of objects and attributes and $\mathcal{R} : \mathcal{O} \times \mathcal{A} \to L$ is a fuzzy relation between $\mathcal{O}$ and $\mathcal{A}$. L might be a support of some algebra $\mathbf{L}=\langle L, ... \rangle$ like a complete residuated lattice. Depending on the approach, the intent and the extent of fuzzy concepts can be both fuzzy, one fuzzy and the other crisp or both crisp. Due to a lack of space we will only talk about two approaches proposed by Yahia et al. in 2001 and Snášel et al. in 2002 which are respectively representative of two families of approach said *one-sided-fuzzy* and *threshold based*.

Snášel et al. proposed to derive classic formal contexts $(\mathcal{O}, \mathcal{A}, {}^{\alpha}\mathcal{R})$ from a L-context $(\mathcal{O}, \mathcal{A}, \mathcal{R})$ using α-*cut* of $\mathcal{R}$ defined as ${}^{\alpha}\mathcal{R} = \{\langle i, j \rangle \mid \mathcal{R}(i,j) \geq \alpha\}$ for $\alpha \in K$. $K \subseteq L$ is a subset of truth degree considered relevant or sufficiently covering L. A standard FCA is then applied on each of the formal context, defining a concept lattice $\mathcal{B}(\mathcal{O}, \mathcal{A}, {}^{\alpha}\mathcal{R})$. All lattices are then merged into a structure $\bigsqcup_{\alpha \in K} \mathcal{B}(\mathcal{O}, \mathcal{A}, {}^{\alpha}\mathcal{R})$ containing pairs (X, Y) where X is the extent of a formal concept of ${}^{\alpha}\mathcal{R}$, and Y is a multiset where the number of occurrences of element a in Y equals the number of cut-levels for which X is an extent defined on the context $(\mathcal{O}, \mathcal{A}, {}^{\alpha}\mathcal{R})$ and $a \in X^{\uparrow}$.

Yahia et al. proposed an approach which create fuzzy concepts with crisp extent and fuzzy intent. For a L-context $(\mathcal{O}, \mathcal{A}, \mathcal{R})$ with $L = [0,1]$, the derivation operators $f : 2^{\mathcal{O}} \to L^{\mathcal{A}}$ (assigning a fuzzy set of attributes to a set of object) and $h : L^{\mathcal{A}} \to 2^{\mathcal{O}}$ (assigning a set of object to a fuzzy set of attributes) are defined by:

$$\begin{aligned} f(X)(a) &= \bigwedge_{o \in X} \mathcal{R}(o,a) \\ h(Y) &= \{o \in \mathcal{O} \mid \text{for each } a \in \mathcal{A} : Y(a) \leq \mathcal{R}(o,a)\} \end{aligned}$$

Fuzzy concepts are defined and ordered in the same way as in standard FCA: a pair (X, Y) is a fuzzy concept for $X \subseteq 2^{\mathcal{O}}$, $Y \in L^{\mathcal{A}}$ if $f(X) = Y$ and $h(Y) = X$ and fuzzy concepts are ordered by $(X_1, Y_1) \leq (X_2, Y_2)$ if $X_1 \subseteq X_2 \Leftrightarrow Y_1 \supseteq Y_2$. Note that this approach can be modify to have fuzzy extents and crisp intents.

Another family of approach considered as more general, produce concepts with fuzzy extent and intent. This approaches usually produce large number of concepts which are sometimes hard to interpret, e.g. when all elements of the intent and the extent have truth value of 0.5. Such approach have been extended to what is called *crisply generated fuzzy concepts* keeping only concepts for which a crisp extent also exist. As the obtain lattice is isomorphic to the one-sided fuzzy concept lattice we do not consider this approach.

Biclustering GED with Fuzzy FCA

As far as we know, fuzzy FCA has never been used for biclustering GED or any numerical dataset. To be able to extract biclusters from L-concepts, we first need to define an appropriated algebra structure $\mathbf{L}$. To be able to apply the method proposed by Yahia et al. we consider a scale $L = [0,1]$ with $a \wedge b = min(a,b)$ and $a \vee b = max(a,b)$. Then, the GED need to be transformed into a L-context, i.e. transposed table entries to take values in $L = [0,1]$. This can be done by simply normalizing the GED. Once it has been done, a table entry (g, c) can be understood as

	a_1	a_2	a_3
o_1	0.1	0.2	0.8
o_2	1.0	0.6	0.2
o_3	0.4	0.6	0.7
o_4	0.1	0.6	0.7

Table 8: A fuzzy context

the expression degree of gene g in condition c. An example of fuzzy concept defined on the fuzzy context represented in Table 8 using Yahia et al. approach is then $(\{o_1, o_3, o_4\}, \{a_1^{0.1}, a_2^{0.2}, a_3^{0.7}\})$. It is understood as: all elements of the intent have an expression degree greater or equal than 0.1 in condition a_1, 0.2 in condition a_2 and 0.7 in condition a_3.

Maximal biclusters of genes simultaneously over-expressed with regards to a threshold t can then be extracted from fuzzy concept by selecting the set of objects and conditions in which the degree of expression is greater than t. The previous concept $(\{o_1, o_3, o_4\}, \{a_1^{0.1}, a_2^{0.2}, a_3^{0.7}\})$ corresponds then to the bicluster $(\{o_1, o_3, o_4\}, \{a_3\})$ for any $t > 0.2$. Biclusters of gene simultaneously under-expressed can be found by replacing the operator $\wedge$ by $\vee$ in the definition of f and changing the order in h:

$$\begin{aligned} f(X)(a) &= \bigvee_{o \in X} \mathcal{R}(o, a) \\ h(Y) &= \{o \in \mathcal{O} \mid \text{for each } a \in \mathcal{A} : Y(a) \geq \mathcal{R}(o, a)\} \end{aligned}$$

A bicluster on under-expressed genes (with regards to t) is then defined on a fuzzy concept by extracting attributes of the concept intent having a maximal expression degree less than t. For example, the bicluster $(\{o_1, o_3, o_4\}, \{a_1, a_2\})$ is extracted from concept $(\{o_1, o_3, o_4\}, \{a_1^{0.4}, a_2^{0.6}, a_3^{0.8}\})$ for any $t < 0.8$.

Biclusters can be defined in an analogous way using the method of Snášel et al. The actual definition giving biclusters of genes over-expressed and a modified definition (i.e. with order $\leq$) giving biclusters of genes under-expressed. All biclusters will then be ordered in two lattices structures.

6 Discussion

In the previous sections, we saw that biclustering can be performed on GED using FCA by either binarizing the GED to apply a standard FCA or using a quantitative generalization of FCA. The first family of approach has the advantage to be simple and to allow us to use existing FCA tools for the visualization and exploration of concept lattice. Nevertheless, conceptual scaling techniques that do not leads to loss of information leads to huge tables which make this approaches unpractical for our data. Some interesting ideas presented can however be kept. For example, we could discretize the data into {-1,0,1} [15] and then define positive correlation on the table by considering another generalization of FCA for which concepts correspond to maximal sets of object having the same value in column. The definition of negative correlation will required a more sophisticated generalization of FCA defining a concept as triple composed of two sets of objects and a set of conditions, objects of one set having opposed values to objects of the other.

The second family of approaches (using quantitative FCA) has the advantage that no scaling stage is required. However, when biclusters are defined with fuzzy FCA and $\mathcal{K}$-FCA a pre-processing

stage can still be needed to obtain values in the desired interval of values. $\mathcal{K}$-FCA and fuzzy FCA approaches to biclustering GED are very similar. They all explore possible threshold values to defined biclusters on genes under- or over-expressed based on their maximal or minimal expression values. However, they differ in the way the resulting concepts are ordered. In the fuzzy approach using α-cut, concepts are ordered in two lattice structures whereas $\mathcal{K}$-FCA produce a sequence of lattices for each value of φ and ϕ. Therefore, the α-cut approach seems more interesting as the exploration of all biclusters will be easier.

The two other quantitative approaches, based on pattern structures and triadic FCA, defined biclusters of similar values for a given θ following opposite strategies. Indeed, the triadic approach start by keeping only intervals of size θ whereas the pattern structure approach work on all possible intervals and select afterwards the biclusters defined for a given θ. If we use the first method we will work only on a subset of intervals which is particularly interesting since we could have a lot of different values but we will have to recompute everything for a different value of θ. On the contrary, with the second method we will have a huge pattern lattice from which we can extract afterwards all biclusters for any values of θ. Therefore, the most promising methods seems to be the one using α-cut and pattern structures.

The approaches defined either biclusters based on similarity of value or similarity of expression but it could be interesting to mix this two ideas to defined biclusters based on similar values and similar expression. We can note that for all approaches we need to fix some parameters (θ, α, φ, ϕ) from which the obtained results will be dependant. An important work will then consists to determined suited values for this parameters.

7 Modeling FCA in Answer Set Programming

We need a flexible framework to implement the various approaches available for biclustering. The team has been working since several years with success with Answer Set Programming (ASP) [11] modeling. ASP is an approach to declarative problem solving which combines an expressive logical representation language with high performance solving capacities. It is suited to solve combinatorial problems. The basic idea of ASP is to represent a problem by a logic program for which models (also called answer sets) correspond to solutions. A problem, represented by a program P, is solved in two stages: first, all variables of P are replaced by all their possible values, the resulting program called grounded program is then given to a boolean solver which find the models. A program consist of a finite set of rules of the form:

$$\underbrace{a_0}_{head} \leftarrow \underbrace{a_1, ..., a_m, \sim a_{m+1}, ..., \sim a_n}_{body}$$

where each a_i is an *atom* (propositional variable) for $0 \leq i \leq n$ and symbols $\leftarrow$, ',', $\sim$ respectively denote *if*, *and* and the *default negation*, i.e. the literal $\sim b$ is true if all the possible ways to prove b failed. Intuitively, the head of the rule must be true if the body holds.

A Formal Concept Analysis can be written in an elegant way in ASP using a few set of rules as showed in Figure 3. Note that `not` stand for $\sim$ and `:-` for $\leftarrow$. N-ary extensions of FCA can easily be derived from this rules adding two predicates (in and out) by dimension. Moreover, constraints on concept can easily be added to an ASP program by adding rules. This is particularly interesting for quantitative generalizations of FCA, as it allows to easily define constraints on similarity of values based on θ or on a minimal gene expression values taking by genes. In addition, FCA tends

```
1.  out_extent(O) :- object(O), in_intent(A), not relation(O,A).
2.  in_extent(O)  :- object(O), not out_extent(O).
3.  out_intent(A) :- attribute(A), in_extent(O), not relation(O,A).
4.  in_intent(A)  :- attribute(A), not out_intent(A).
```

Figure 3: ASP rules to perform FCA

to produce huge number of concepts when it is applied to big dataset and all concepts are not always meaningful or interesting depending on the task. Therefore, only a subset of concept verifying some constraints should be returned. In [14], membership constraints are used to navigate the concept space in ASP. Other constraints such as minimal support of object or attribute could be used. It make sens to apply such constraint for the analysis of gene expression data since concepts are interesting only if the element of the intent, i.e. genes, have common (or opposite) behaviour in at least a minimum number of conditions.

8 Conclusion

We have introduced several approaches to quantitative FCA and discussed their interest for the task of biclustering GED based on positive and negative correlation of gene expression values. We notice that negative correlation have not been much considered, a lot of biclustering methods focusing only on positive correlation. However, such correlation may imply important biological mechanism. Therefore, during the internship we will used different quantitative generalization of FCA to define biclusters based both on positive and negative correlation. This approaches will be implemented in the ASP formalism. A FCA can be modeled in an elegant way in ASP but its quantitative generalizations will require some adjustments. Moreover, the analysis will be applied to two huge data regarding expression values on dog melanoma and pea aphid reproduction. Therefore, data reduction on context and local discretization [4] will also be required in order to facilitate the task and speed up the solving process. Finally, we will have to find adapted values for the different parameters used such as the parameter θ defining similarity.

References

[1] Radim Belohlavek. What is a fuzzy concept lattice? ii. In *International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing*, pages 19–26. Springer, 2011.

[2] Claudio Carpineto and Giovanni Romano. *Concept data analysis: Theory and applications*. John Wiley & Sons, 2004.

[3] Yizong Cheng and George M Church. Biclustering of expression data. In *Ismb*, volume 8, pages 93–103, 2000.

[4] Sérgio M Dias and Newton J Vieira. Concept lattices reduction: Definition, analysis and classification. *Expert Systems with Applications*, 42(20):7084–7097, 2015.

[5] Bernhard Ganter and Sergei O Kuznetsov. Pattern structures and their projections. In *International Conference on Conceptual Structures*, pages 129–142. Springer, 2001.

[6] Jose M González-Calabozo, Francisco J Valverde-Albacete, and Carmen Peláez-Moreno. Interactive knowledge discovery and data mining on genomic expression data with numeric formal concept analysis. *BMC bioinformatics*, 17(1):374, 2016.

[7] Mehdi Kaytoue, Sergei O Kuznetsov, Juraj Macko, and Amedeo Napoli. Biclustering meets triadic concept analysis. *Annals of Mathematics and Artificial Intelligence*, 70(1-2):55–79, 2014.

[8] Mehdi Kaytoue, Sergei O Kuznetsov, and Amedeo Napoli. Biclustering numerical data in formal concept analysis. In *International Conference on Formal Concept Analysis*, pages 135–150. Springer, 2011.

[9] Mehdi Kaytoue, Sergei O Kuznetsov, Amedeo Napoli, and Sébastien Duplessis. Mining gene expression data with pattern structures in formal concept analysis. *Information Sciences*, 181(10):1989–2001, 2011.

[10] Fritz Lehmann and Rudolf Wille. *A triadic approach to formal concept analysis*. Springer, 1995.

[11] Gebser M, Kaminski R, Kaufmann, and Schaub T. *Answer Set Solving in Practice*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan and Claypool Publishers, 2012.

[12] Sara C Madeira and Arlindo L Oliveira. Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 1(1):24–45, 2004.

[13] Susanne Motameny, Beatrix Versmold, and Rita Schmutzler. Formal concept analysis for the identification of combinatorial biomarkers in breast cancer. In *International Conference on Formal Concept Analysis*, pages 229–240. Springer, 2008.

[14] Sebastian Rudolph, Christian Sacarea, and Diana Troanca. Membership constraints in formal concept analysis. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, 2015.

[15] Xudong Tu, Yuanliang Wang, Maolan Zhang, and Jinchuan Wu. Using formal concept analysis to identify negative correlations in gene expression data. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 13(2):380–391, 2016.

[16] Francisco J Valverde-Albacete, José María González-Calabozo, Anselmo Peñas, and Carmen Peláez-Moreno. Supporting scientific knowledge discovery with extended, generalized formal concept analysis. *Expert Systems with Applications*, 44:198–216, 2016.

[17] Francisco J Valverde-Albacete and Carmen Peláez-Moreno. Galois connections between semimodules and applications in data mining. In *International Conference on Formal Concept Analysis*, pages 181–196. Springer, 2007.